

MBS MacCG Plugin Documentation

Christian Schmitz

March 10, 2025

0.1 Introduction

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

0.2 Content

- 1 List of all topics 3
- 2 List of all classes 61
- 3 List of all global methods 65
- 4 All items in this plugin 67
- 13 List of Questions in the FAQ 661
- 14 The FAQ 671

Chapter 1

List of Topics

- 6 CoreGraphics 97
 - 11.1.1 class Application 653
 - * 11.1.3 OverlayApplicationDockTileImageMBS(pic as CGImageMBS) as boolean 653
 - * 11.1.4 RestoreApplicationDockTileImageMBS as boolean 654
 - * 11.1.5 SetApplicationDockTileImageMBS(pic as CGImageMBS) as boolean 654

• 5 CoreAnimation	69
– 5.1.1 class CALayerMBS	69
* 5.1.3 addSublayer(layer as CALayerMBS)	70
* 5.1.4 available as boolean	71
* 5.1.5 Constructor	71
* 5.1.6 display	71
* 5.1.7 displayIfNeeded	71
* 5.1.8 layer as CALayerMBS	72
* 5.1.9 layerFromHandle(Handle as Integer) as CALayerMBS	72
* 5.1.10 layerFromHandle(Handle as Ptr) as CALayerMBS	72
* 5.1.11 layoutIfNeeded	72
* 5.1.12 layoutSublayers	72
* 5.1.13 removeAllAnimations	73
* 5.1.14 removeFromSuperlayer	73
* 5.1.15 renderInContext(CGContextHandle as Integer) as boolean	73
* 5.1.16 renderInPicture(Pic as Picture) as boolean	73
* 5.1.17 setNeedsDisplay	74
* 5.1.18 setNeedsDisplayInRect(r as CGRectMBS)	74
* 5.1.19 setNeedsLayout	74
* 5.1.20 sublayers as CALayerMBS()	75
* 5.1.22 affineTransform as CGAffineTransformMBS	75
* 5.1.23 anchorPoint as CGRectMBS	75
* 5.1.24 anchorPointZ as Double	75
* 5.1.25 AutoresizingMask as Integer	76
* 5.1.26 backgroundColor as Variant	76
* 5.1.27 borderColor as Variant	76
* 5.1.28 borderWidth as Double	77
* 5.1.29 bounds as CGRectMBS	77
* 5.1.30 className as string	77
* 5.1.31 classPath as string	77
* 5.1.32 contents as Variant	78
* 5.1.33 contentsCenter as CGRectMBS	78
* 5.1.34 contentsRect as CGRectMBS	78
* 5.1.35 contentsScale as Double	79
* 5.1.36 cornerRadius as Double	79
* 5.1.37 DoubleSided as boolean	79
* 5.1.38 drawsAsynchronously as boolean	80
* 5.1.39 frame as CGRectMBS	80
* 5.1.40 Handle as Integer	80
* 5.1.41 Hidden as boolean	81
* 5.1.42 mask as CALayerMBS	81

	5
* 5.1.43 masksToBounds as Boolean	81
* 5.1.44 minificationFilterBias as Double	81
* 5.1.45 modelLayer as CALayerMBS	82
* 5.1.46 needsDisplay as boolean	82
* 5.1.47 needsDisplayOnBoundsChange as boolean	82
* 5.1.48 needsLayout as boolean	82
* 5.1.49 opacity as Double	83
* 5.1.50 Opaque as boolean	83
* 5.1.51 position as CGRectMBS	83
* 5.1.52 preferredFrameSize as CGSizeMBS	84
* 5.1.53 presentationLayer as CALayerMBS	84
* 5.1.54 rasterizationScale as Double	84
* 5.1.55 shadowColor as Variant	84
* 5.1.56 shadowOffset as CGSizeMBS	85
* 5.1.57 shadowOpacity as Double	85
* 5.1.58 shadowPath as Variant	85
* 5.1.59 shadowRadius as Double	86
* 5.1.60 shouldRasterize as Boolean	86
* 5.1.61 superlayer as CALayerMBS	86
* 5.1.62 zPosition as Double	87

- **10 Printing Mac** 615
 - ?? Globals ??
 - * 10.1.1 NewCPMPageFormatMBS as CPMPageFormatMBS 615
 - * 10.1.2 NewCPMPrintSessionMBS as CPMPrintSessionMBS 615
 - * 10.1.3 NewCPMPrintSettingsMBS as CPMPrintSettingsMBS 617

• 5 CoreAnimation	69
– 5.2.1 class CATextLayerMBS	88
* 5.2.3 Constructor	88
* 5.2.4 kCAAlignmentCenter as String	88
* 5.2.5 kCAAlignmentJustified as String	88
* 5.2.6 kCAAlignmentLeft as String	89
* 5.2.7 kCAAlignmentNatural as String	89
* 5.2.8 kCAAlignmentRight as String	89
* 5.2.9 kCATruncationEnd as String	89
* 5.2.10 kCATruncationMiddle as String	89
* 5.2.11 kCATruncationNone as String	89
* 5.2.12 kCATruncationStart as String	90
* 5.2.14 alignmentMode as String	90
* 5.2.15 allowsFontSubpixelQuantization as Boolean	90
* 5.2.16 font as Variant	90
* 5.2.17 fontSize as Double	90
* 5.2.18 foregroundColor as Variant	91
* 5.2.19 String as Variant	91
* 5.2.20 truncationMode as String	91
* 5.2.21 Wrapped as Boolean	91
– 5.3.1 class CATransactionMBS	92
* 5.3.3 animationDuration as Double	92
* 5.3.4 available as boolean	92
* 5.3.5 begin	93
* 5.3.6 commit	93
* 5.3.7 Constructor	93
* 5.3.8 flush	93
* 5.3.9 kCATransactionAnimationDuration as string	94
* 5.3.10 kCATransactionDisableActions as string	94
* 5.3.11 setAnimationDuration(value as Double)	94
* 5.3.12 setValue(value as Variant, key as string)	94
* 5.3.13 valueForKey(key as string) as Variant	94
* 5.3.15 Handle as Integer	95

• 6 CoreGraphics	97
– ?? Globals	??
* 6.1.4 CGBitmapContextCreateMBS(data as memoryblock, width as Integer, height as Integer, bitsPerComponent as Integer, bytesPerRow as Integer, colorspace as CGColorSpaceMBS, alphaInfo as Integer) as CGBitmapContextMBS	99
* 6.1.8 CGCreateImageFromJPEGDataProviderMBS(dataprovider as Variant, decode as memoryblock, shouldInterpolate as boolean, intent as Integer) as CGImageMBS	103
* 6.1.9 CGCreateImageFromPNGDataProviderMBS(dataprovider as Variant, decode as memoryblock, shouldInterpolate as boolean, intent as Integer) as CGImageMBS	104
* 6.1.10 CGCreateImageMBS(pic as picture) as CGImageMBS	105
* 6.1.11 CGCreateImageMBS(pic as picture, mask as picture) as CGImageMBS	105
* 6.1.12 CGMakePointMBS(x as Double, y as Double) as CGPointMBS	106
* 6.1.13 CGMakeRectMBS(left as Double, top as Double, width as Double, height as Double) as CGRectMBS	106
* 6.1.14 CGMakeSizeMBS(width as Double, height as Double) as CGSizeMBS	106
* 6.1.15 CGNewPDFDocumentMBS(consumer as CGDataConsumerMBS, MediaBox as CGRectMBS, title as string, author as string, creator as string) as CGPDFContextMBS	106
* 6.1.16 CGNewPDFDocumentMBS(file as folderitem, MediaBox as CGRectMBS, title as string, author as string, creator as string) as CGPDFContextMBS	107
* 6.1.1 CGNewPDFDocumentMBS(file as folderitem, MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean) as CGPDFContextMBS	97
* 6.1.2 CGNewPDFDocumentMBS(file as folderitem, MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean, KeyLength as Integer) as CGPDFContextMBS	98
* 6.1.5 CGOpenPDFDocumentMBS(dataprovider as CGDataProviderMBS) as CGPDFDocumentMBS	100
* 6.1.17 CGOpenPDFDocumentMBS(file as folderitem) as CGPDFDocumentMBS	107
* 6.1.6 CGShadingCreateAxialMBS(colorspace as CGColorSpaceMBS, startPoint as CGPointMBS, endPoint as CGPointMBS, func as CGFunctionMBS, extendStart as boolean, extendEnd as boolean) as CGShadingMBS	101
* 6.1.7 CGShadingCreateRadialMBS(colorspace as CGColorSpaceMBS, startPoint as CGPointMBS, startRadius as Double, endPoint as CGPointMBS, endRadius as Double, func as CGFunctionMBS, extendStart as boolean, extendEnd as boolean) as CGShadingMBS	102
* 6.1.3 GetCurrentCGContextMBS as CGContextMBS	98
– 6.2.1 class CGAffineTransformMBS	107
* 6.2.3 Binary as MemoryBlock	108
* 6.2.4 Concat(t as CGAffineTransformMBS) as CGAffineTransformMBS	108
* 6.2.5 Constructor	108
* 6.2.6 Constructor(a as double, b as double, c as double, d as double, tx as double = 0.0, ty as double = 0.0)	109

* 6.2.7	Constructor(p as Ptr)	109
* 6.2.8	Constructor(transform as CGAffineTransformMBS)	109
* 6.2.9	EqualToTransform(t as CGAffineTransformMBS) as boolean	110
* 6.2.10	Identity as CGAffineTransformMBS	110
* 6.2.11	Invert as CGAffineTransformMBS	110
* 6.2.12	IsIdentity as boolean	110
* 6.2.13	Make(a as double, b as double, c as double, d as double, tx as double = 0.0, ty as double = 0.0) as CGAffineTransformMBS	110
* 6.2.14	MakeRotation(angle as Double) as CGAffineTransformMBS	111
* 6.2.15	MakeScale(sx as Double, sy as Double) as CGAffineTransformMBS	111
* 6.2.16	MakeTranslation(tx as Double, ty as Double) as CGAffineTransformMBS	111
* 6.2.17	Operator__Convert as String	111
* 6.2.18	Operator__Multiply(other as CGAffineTransformMBS) as CGAffineTransformMBS	111
* 6.2.19	Operator__MultiplyRight(other as CGAffineTransformMBS) as CGAffineTransformMBS	111
* 6.2.20	Rotate(angle as Double) as CGAffineTransformMBS	112
* 6.2.21	Scale(sx as Double, sy as Double) as CGAffineTransformMBS	112
* 6.2.22	Translate(tx as Double, ty as Double) as CGAffineTransformMBS	112
* 6.2.24	A as Double	112
* 6.2.25	B as Double	112
* 6.2.26	C as Double	113
* 6.2.27	D as Double	113
* 6.2.28	StringValue as String	113
* 6.2.29	TX as Double	113
* 6.2.30	TY as Double	113
– 6.3.1	class CGBitmapContextMBS	114
* 6.3.3	CGImage(shouldInterpolate as boolean = false, intent as Integer = 0) as CGImageMBS	115
* 6.3.4	Constructor	115
* 6.3.5	Create(data as memoryblock, width as Integer, height as Integer, bitsPerComponent as Integer, bytesPerRow as Integer, colorspace as CGColorSpaceMBS, alphaInfo as Integer) as CGBitmapContextMBS	116
* 6.3.6	Create(Other as CGBitmapContextMBS, NewColorspace as CGColorSpaceMBS) as CGBitmapContextMBS	117
* 6.3.7	CreateImage as CGImageMBS	118
* 6.3.8	CreateRGB(data as memoryblock, width as Integer, height as Integer, bytesPerRow as Integer, colorspace as CGColorSpaceMBS = nil) as CGBitmapContextMBS	119
* 6.3.9	CreateWithPicture(Pic as Picture) as CGBitmapContextMBS	119
* 6.3.11	BitmapAlphaInfo as Integer	120
* 6.3.12	BitmapBitsPerComponent as Integer	120
* 6.3.13	BitmapBitsPerPixel as Integer	121
* 6.3.14	BitmapBytesPerRow as Integer	121

* 6.3.15	BitmapColorSpace as CGColorSpaceMBS	122
* 6.3.16	BitmapData as MemoryBlock	122
* 6.3.17	BitmapHeight as Integer	122
* 6.3.18	BitmapInfo as Integer	123
* 6.3.19	BitmapWidth as Integer	123
– 6.4.1	class CGColorMBS	125
* 6.4.3	Alpha as Double	125
* 6.4.4	Black as CGColorMBS	125
* 6.4.5	Clear as CGColorMBS	125
* 6.4.6	ColorSpace as CGColorSpaceMBS	125
* 6.4.7	Components as memoryblock	126
* 6.4.8	Copy as CGColorMBS	126
* 6.4.9	CopyWithAlpha(alpha as Double) as CGColorMBS	126
* 6.4.10	Create(colorspace as CGColorSpaceMBS, components as memoryblock) as CGColorMBS	126
* 6.4.11	Create(colorspace as CGColorSpaceMBS, components() as Double) as CGColorMBS	127
* 6.4.12	CreateDeviceCMYK(cyan as Double, magenta as Double, yellow as Double, black as Double, alpha as Double = 1.0) as CGColorMBS	127
* 6.4.13	CreateDeviceGray(gray as Double, alpha as Double = 1.0) as CGColorMBS	128
* 6.4.14	CreateDeviceRGB(red as Double, green as Double, blue as Double, alpha as Double = 1.0) as CGColorMBS	128
* 6.4.15	CreateGenericCMYK(cyan as Double, magenta as Double, yellow as Double, black as Double, alpha as Double = 1.0) as CGColorMBS	128
* 6.4.16	CreateGenericGray(gray as Double, alpha as Double = 1.0) as CGColorMBS	129
* 6.4.17	CreateGenericRGB(red as Double, green as Double, blue as Double, alpha as Double = 1.0) as CGColorMBS	129
* 6.4.18	Equal(secondColor as CGColorMBS) as boolean	129
* 6.4.19	NumberOfComponents as Integer	129
* 6.4.20	White as CGColorMBS	130
* 6.4.22	Handle as Integer	130
– 6.5.1	class CGColorSpaceMBS	131
* 6.5.3	CreateCalibratedGray(whitePoint() as Double, blackPoint() as Double, gamma as Double) as CGColorSpaceMBS	131
* 6.5.4	CreateCalibratedRGB(whitePoint() as Double, blackPoint() as Double, gamma() as Double, matrix() as Double) as CGColorSpaceMBS	131
* 6.5.5	CreateDeviceCMYK as CGColorSpaceMBS	132
* 6.5.6	CreateDeviceGray as CGColorSpaceMBS	132
* 6.5.7	CreateDeviceRGB as CGColorSpaceMBS	132
* 6.5.8	CreateLab(whitePoint() as Double, blackPoint() as Double, range() as Double) as CGColorSpaceMBS	132
* 6.5.9	CreatePattern(baseSpace as CGColorSpaceMBS) as CGColorSpaceMBS	133

* 6.5.10 CreateWithHandle(Handle as Integer) as CGColorSpaceMBS	133
* 6.5.11 CreateWithICCProfile(ICCProfileData as memoryblock) as CGColorSpaceMBS	133
* 6.5.12 CreateWithICCProfile(ICCProfileData as string) as CGColorSpaceMBS	134
* 6.5.13 CreateWithName(name as string) as CGColorSpaceMBS	134
* 6.5.14 CreateWithPlatformColorSpace(Handle as Integer) as CGColorSpaceMBS	134
* 6.5.15 ICCProfile as string	134
* 6.5.16 kCGColorSpaceACEScGLinear as string	134
* 6.5.17 kCGColorSpaceAdobeRGB1998 as string	135
* 6.5.18 kCGColorSpaceDCIP3 as string	135
* 6.5.19 kCGColorSpaceDisplayP3 as string	135
* 6.5.20 kCGColorSpaceGenericCMYK as string	135
* 6.5.21 kCGColorSpaceGenericGray as string	136
* 6.5.22 kCGColorSpaceGenericGrayGamma2_2 as string	136
* 6.5.23 kCGColorSpaceGenericRGB as string	136
* 6.5.24 kCGColorSpaceGenericRGBLinear as string	136
* 6.5.25 kCGColorSpaceGenericXYZ as string	136
* 6.5.26 kCGColorSpaceITUR_2020 as string	137
* 6.5.27 kCGColorSpaceITUR_709 as string	137
* 6.5.28 kCGColorSpaceROMMRGB as string	137
* 6.5.29 kCGColorSpaceSRGB as string	138
* 6.5.31 BaseColorSpace as CGColorSpaceMBS	138
* 6.5.32 ColorTableCount as Integer	138
* 6.5.33 Description as String	138
* 6.5.34 Handle as Integer	139
* 6.5.35 Model as Integer	139
* 6.5.36 ModelText as String	139
* 6.5.37 Name as String	139
* 6.5.38 NumberOfComponents as Integer	140
– 6.6.1 class CGContextMBS	142
* 6.6.3 AddArc(x as Double, y as Double, radius as Double, startangle as Double, endangle as Double, clockwise as boolean)	143
* 6.6.4 addArcToPath(x as Double, y as Double, w as Double, h as Double, startAngle as Integer, arcAngle as Integer)	143
* 6.6.5 AddArcToPoint(x1 as Double, y1 as Double, x2 as Double, y2 as Double, radius as Double)	144
* 6.6.6 AddCurveToPoint(cp1x as Double, cp1y as Double, cp2x as Double, cp2y as Double, x as Double, y as Double)	144
* 6.6.7 AddEllipseInRect(r as CGRectMBS)	144
* 6.6.8 AddLines(p() as CGPointMBS)	144
* 6.6.9 AddLineToPoint(x as Double, y as Double)	145
* 6.6.10 addOvalToPath(x as Double, y as Double, w as Double, h as Double)	145
* 6.6.11 AddPath(path as CGPathMBS)	145

* 6.6.12 AddQuadCurveToPoint(cpx as Double, cpy as Double, x as Double, y as Double)	145
* 6.6.13 AddRect(r as CGRectMBS)	145
* 6.6.14 AddRects(r() as CGRectMBS)	146
* 6.6.15 addRoundedRectToPath(x as Double, y as Double, w as Double, h as Double, arcWidth as Double, arcHeight as Double)	146
* 6.6.16 BeginPage(mediabox as CGRectMBS)	146
* 6.6.17 BeginPath	146
* 6.6.18 BeginTransparencyLayer(auxiliaryInfo as Dictionary = nil)	147
* 6.6.19 BeginTransparencyLayerWithRect(r as CGRectMBS, auxiliaryInfo as Dictionary = nil)	147
* 6.6.20 clearRect(rect as CGRectMBS)	147
* 6.6.21 clip	148
* 6.6.22 ClipToMask(rect as CGRectMBS, mask as CGImageMBS)	148
* 6.6.23 clipToRect(rect as CGRectMBS)	148
* 6.6.24 Close	149
* 6.6.25 closePath	149
* 6.6.26 ConcatCTM(transform as CGAffineTransformMBS)	149
* 6.6.27 Constructor(g as graphics)	149
* 6.6.28 Constructor(handle as Integer)	150
* 6.6.29 Constructor(pic as Picture)	150
* 6.6.30 contextWithCGContext(handle as Integer) as CGContextMBS	151
* 6.6.31 contextWithCGContext(handle as Ptr) as CGContextMBS	152
* 6.6.32 CopyPath as CGPathMBS	153
* 6.6.33 DrawCGPDFDocument(pdf as Variant, rect as CGRectMBS, page as Integer)	153
* 6.6.34 DrawLayerAtPoint(Point as CGPointMBS, layer as CGLayerMBS)	154
* 6.6.35 DrawLayerInRect(rect as CGRectMBS, layer as CGLayerMBS)	155
* 6.6.36 DrawLinearGradient(gradient as CGGradientMBS, startPoint as CGPointMBS, endPoint as CGPointMBS, options as Integer)	155
* 6.6.37 DrawPath(mode as Integer)	155
* 6.6.38 DrawPicture(pic as CGImageMBS, rect as CGRectMBS)	155
* 6.6.39 DrawRadialGradient(gradient as CGGradientMBS, startCenter as CGPointMBS, startRadius as Double, endCenter as CGPointMBS, endRadius as Double, options as Integer)	156
* 6.6.40 DrawShading(shading as CGShadingMBS)	157
* 6.6.41 DrawTiledImage(pic as CGImageMBS, rect as CGRectMBS)	157
* 6.6.42 EndPage	158
* 6.6.43 EndTransparencyLayer	158
* 6.6.44 EOClip	158
* 6.6.45 EOFillPath	158
* 6.6.46 FillEllipseInRect(rect as CGRectMBS)	158
* 6.6.47 FillPath	159
* 6.6.48 FillRect(rect as CGRectMBS)	159
* 6.6.49 fillRoundedRect(x as Double, y as Double, w as Double, h as Double, arcWidth as Double, arcHeight as Double)	160

* 6.6.50 Flush	160
* 6.6.51 frameArc(x as Double, y as Double, w as Double, h as Double, startAngle as Integer, arcAngle as Integer)	160
* 6.6.52 frameOval(x as Double, y as Double, w as Double, h as Double)	161
* 6.6.53 frameRect(x as Double, y as Double, w as Double, h as Double)	161
* 6.6.54 GetClipBoundingBox as CGRectMBS	162
* 6.6.55 GetCTM as CGAffineTransformMBS	162
* 6.6.56 GetPathBoundingBox as CGRectMBS	162
* 6.6.57 GetPathCurrentPoint as CGPointMBS	162
* 6.6.58 GetTextPosition as CGPointMBS	162
* 6.6.59 IsPathEmpty as boolean	163
* 6.6.60 MoveToPoint(x as Double, y as Double)	163
* 6.6.61 paintArc(x as Double, y as Double, w as Double, h as Double, startAngle as Integer, arcAngle as Integer)	163
* 6.6.62 paintOval(x as Double, y as Double, w as Double, h as Double)	163
* 6.6.63 paintRect(x as Double, y as Double, w as Double, h as Double)	164
* 6.6.64 PathContainsPoint(point as CGPointMBS, mode as Integer) as boolean	164
* 6.6.65 ReplacePathWithStrokedPath	165
* 6.6.66 RestoreGState	165
* 6.6.67 RotateCTM(angle as Double)	165
* 6.6.68 SaveGState	166
* 6.6.69 ScaleCTM(sx as Double, sy as Double)	166
* 6.6.70 SelectFont(name as string, size as Double, fontencoding as Integer)	166
* 6.6.71 SetAllowsAntialiasing(allowsAntialiasing as boolean)	166
* 6.6.72 SetAlpha(alpha as Double)	167
* 6.6.73 SetBlendMode(BlendMode as Integer)	167
* 6.6.74 SetCharacterSpacing(spacing as Double)	167
* 6.6.75 SetCMYKFillColor(cyan as Double, magenta as Double, yellow as Double, black as Double, alpha as Double = 1.0)	167
* 6.6.76 SetCMYKStrokeColor(cyan as Double, magenta as Double, yellow as Double, black as Double, alpha as Double = 1.0)	167
* 6.6.77 SetFillColor(color as CGColorMBS)	168
* 6.6.78 SetFillColorSpace(colorspace as CGColorSpaceMBS)	168
* 6.6.79 SetFlatness(flatness as Double)	168
* 6.6.80 SetFont(font as CGFontMBS)	168
* 6.6.81 SetFontSize(size as Double)	168
* 6.6.82 SetGrayFillColor(gray as Double, alpha as Double = 1.0)	168
* 6.6.83 SetGrayStrokeColor(gray as Double, alpha as Double = 1.0)	169
* 6.6.84 SetLineCap(cap as Integer)	169
* 6.6.85 SetLineDash(phase as Double, lengths as memoryblock, count as Integer)	169
* 6.6.86 SetLineJoin(join as Integer)	170
* 6.6.87 SetLineWidth(width as Double)	170

* 6.6.88 SetMiterLimit(limit as Double)	170
* 6.6.89 SetRenderingIntent(intent as Integer)	170
* 6.6.90 SetRGBFillColor(red as Double, green as Double, blue as Double, alpha as Double = 1.0)	171
* 6.6.91 SetRGBStrokeColor(red as Double, green as Double, blue as Double, alpha as Double = 1.0)	171
* 6.6.92 SetShadow(x as Double, y as Double, blur as Double)	171
* 6.6.93 SetShadowWithColor(x as Double, y as Double, blur as Double, colorvalue as CGColorMBS)	172
* 6.6.94 SetShouldAntialias(shouldAntialias as boolean)	172
* 6.6.95 SetShouldSmoothFonts(shouldSmoothFonts as boolean)	172
* 6.6.96 SetStrokeColor(color as CGColorMBS)	173
* 6.6.97 SetStrokeColorSpace(colorspace as CGColorSpaceMBS)	173
* 6.6.98 SetTextDrawingMode(mode as Integer)	173
* 6.6.99 ShowText(text as string)	173
* 6.6.100 ShowTextAtPoint(text as string, x as Double, y as Double)	174
* 6.6.101 StrokeEllipseInRect(rect as CGRectMBS)	174
* 6.6.102 StrokePath	174
* 6.6.103 StrokeRect(rect as CGRectMBS)	175
* 6.6.104 StrokeRectWithWidth(rect as CGRectMBS, width as Double)	175
* 6.6.105 strokeRoundedRect(x as Double, y as Double, w as Double, h as Double, arcWidth as Double, arcHeight as Double)	175
* 6.6.106 Synchronize	175
* 6.6.107 TranslateCTM(tx as Double, ty as Double)	175
* 6.6.109 handle as Integer	177
* 6.6.110 InterpolationQuality as Integer	177
* 6.6.111 RetainCount as Integer	177
* 6.6.112 TextMatrix as CGAffineTransformMBS	177
* 6.6.113 TextPosition as CGPointMBS	178
– 6.7.1 class CGDataConsumerMBS	181
* 6.7.3 Constructor	181
* 6.7.4 Constructor(file as folderitem)	181
* 6.7.5 Constructor(url as string)	181
* 6.7.6 CreateWithFile(file as folderitem) as CGDataConsumerMBS	182
* 6.7.7 CreateWithURL(url as string) as CGDataConsumerMBS	182
* 6.7.9 Handle as Integer	182
* 6.7.11 CloseConsumer	182
* 6.7.12 Put(data as string) as Integer	182
– 6.8.1 class CGDataProviderMBS	184
* 6.8.3 Constructor(data as string)	184
* 6.8.4 Constructor(file as folderitem)	184
* 6.8.5 CreateWithData(data as string) as CGDataProviderMBS	184

	15
* 6.8.6 CreateWithFile(file as folderitem) as CGDataProviderMBS	185
* 6.8.7 CreateWithURL(url as string) as CGDataProviderMBS	185
* 6.8.8 Data as string	185
* 6.8.10 Handle as Integer	185
– 6.9.1 class CGDisplayConfigMBS	186
* 6.9.3 Cancel	186
* 6.9.4 Complete(options as Integer)	186
* 6.9.5 DisplayMode(display as CGDisplayMBS, mode as CGDisplayModeMBS)	187
* 6.9.6 MirrorOfDisplay(display as CGDisplayMBS, master as CGDisplayMBS)	187
* 6.9.7 Mode(display as CGDisplayMBS, mode as Dictionary)	188
* 6.9.8 Origin(display as CGDisplayMBS, x as Integer, y as Integer)	188
* 6.9.9 RestorePermanentDisplayConfiguration	189
* 6.9.10 StereoOperation(display as CGDisplayMBS, stereo as Boolean, forceBlueLine as Boolean)	189
* 6.9.12 Handle as Integer	189
* 6.9.13 Lasterror as Integer	190
– 6.10.1 class CGDisplayMBS	191
* 6.10.3 AllDisplayModes(Options as Dictionary = nil) as CGDisplayModeMBS()	191
* 6.10.4 AvailableModes as Dictionary()	192
* 6.10.5 BestModeForParameters(BitsPerPixel as Integer, Width as Integer, Height as Integer, byref ExactMatch as boolean) as Dictionary	193
* 6.10.6 BestModeForParametersAndRefreshRate(BitsPerPixel as Integer, Width as Integer, Height as Integer, RefreshRate as Integer, byref ExactMatch as boolean) as Dictionary	193
* 6.10.7 BestModeForParametersAndRefreshRateWithProperty(BitsPerPixel as Integer, Width as Integer, Height as Integer, RefreshRate as Integer, propertyName as string, byref ExactMatch as boolean) as Dictionary	193
* 6.10.8 Capture as Integer	194
* 6.10.9 CaptureAllDisplays as Integer	194
* 6.10.10 CaptureAllDisplaysWithOptions(options as Integer) as Integer	195
* 6.10.11 CaptureWithOptions(options as Integer) as Integer	196
* 6.10.12 CreateImage as CGImageMBS	196
* 6.10.13 CreateImageAsync(receiverDelegate as CreateImageAsyncDelegateMBS, jpegQuality as Double = 0.9, tag as Variant = nil)	197
* 6.10.14 CreateImageForRect(rect as CGRectMBS) as CGImageMBS	197
* 6.10.15 DrawingContext as CGContextMBS	197
* 6.10.16 GetActiveDisplayList as CGDisplayMBS()	198
* 6.10.17 GetDisplaysWithOpenGLDisplayMask(mask as Integer) as CGDisplayMBS()	199
* 6.10.18 GetDisplaysWithPoint(cgpoint as CGPointMBS) as CGDisplayMBS()	199
* 6.10.19 GetDisplaysWithPoint(x as Double, y as Double) as CGDisplayMBS()	199
* 6.10.20 GetDisplaysWithRect(cgrect as CGRectMBS) as CGDisplayMBS()	200
* 6.10.21 GetDisplaysWithRect(x as Double, y as Double, w as Double, h as Double) as CGDisplayMBS()	201

* 6.10.22 GetDisplayTransferByTable(capacity as Integer, red as memoryblock, green as memoryblock, blue as memoryblock, byref samplecount as Integer) as Integer	201
* 6.10.23 GetDisplayTransferFormula(byref formula as CGDisplayTransferFormulaMBS) as Integer	201
* 6.10.24 GetLastMouseDelta(byref deltax as Integer, byref deltay as Integer)	202
* 6.10.25 GetOnlineDisplayList as CGDisplayMBS()	202
* 6.10.26 HideCursor as Integer	203
* 6.10.27 Info as Dictionary	203
* 6.10.28 InfoAsCFDictionary as Variant	203
* 6.10.29 IsCaptured as boolean	203
* 6.10.30 MainDisplay as CGDisplayMBS	204
* 6.10.31 MoveCursorToPoint(x as Double, y as Double) as Integer	204
* 6.10.32 OpenGLDisplayMask as Integer	204
* 6.10.33 Release as Integer	205
* 6.10.34 ReleaseAllDisplays as Integer	205
* 6.10.35 RestoreColorSyncSettings	206
* 6.10.36 SetDisplayMode(mode as CGDisplayModeMBS) as Integer	206
* 6.10.37 SetDisplayTransferByByteTable(count as Integer, red as memoryblock, green as memoryblock, blue as memoryblock) as Integer	206
* 6.10.38 SetDisplayTransferByTable(count as Integer, red as memoryblock, green as memoryblock, blue as memoryblock) as Integer	206
* 6.10.39 SetDisplayTransferFormula(formula as CGDisplayTransferFormulaMBS) as Integer	207
* 6.10.40 SetRotation(angle as Integer) as Integer	207
* 6.10.41 SetStereoOperation(stereo as boolean, forceBlueLine as boolean, option as Integer) as Integer	208
* 6.10.42 ShieldingWindowID as UInt32	208
* 6.10.43 ShieldingWindowLevel as Int32	208
* 6.10.44 ShowCursor as Integer	209
* 6.10.45 SwitchToMode(Mode as Dictionary) as Integer	209
* 6.10.46 WaitForBeamPositionOutsideLines(upperScanLine as UInt32, lowerScanLine as UInt32) as Integer	209
* 6.10.48 BeamPosition as UInt32	210
* 6.10.49 Bounds as CGRectMBS	210
* 6.10.50 Brightness as Double	211
* 6.10.51 CanSetPalette as boolean	211
* 6.10.52 ColorSpace as CGColorSpaceMBS	211
* 6.10.53 CurrentMode as Dictionary	212
* 6.10.54 DisplayMode as CGDisplayModeMBS	212
* 6.10.55 DisplayProductNames as Dictionary	212
* 6.10.56 Handle as Integer	213
* 6.10.57 IOServicePort as Integer	213
* 6.10.58 IsActive as boolean	213

* 6.10.59 IsAlwaysInMirrorSet as boolean	213
* 6.10.60 IsAsleep as boolean	214
* 6.10.61 IsBuiltin as boolean	214
* 6.10.62 IsInHWMirrorSet as boolean	215
* 6.10.63 IsInMirrorSet as boolean	215
* 6.10.64 IsMain as boolean	215
* 6.10.65 IsOnline as boolean	216
* 6.10.66 IsStereo as boolean	216
* 6.10.67 LastError as Integer	217
* 6.10.68 MirrorsDisplay as CGDisplayMBS	217
* 6.10.69 ModelNumber as Integer	217
* 6.10.70 PixelsHigh as Integer	218
* 6.10.71 PixelsWide as Integer	218
* 6.10.72 PrimaryDisplay as CGDisplayMBS	218
* 6.10.73 RefreshRate as Integer	219
* 6.10.74 Rotation as Double	219
* 6.10.75 ScreenSizeHeight as Double	219
* 6.10.76 ScreenSizeWidth as Double	220
* 6.10.77 SerialNumber as Integer	220
* 6.10.78 UnitNumber as Integer	221
* 6.10.79 UsesOpenGLAcceleration as boolean	222
* 6.10.80 VendorNumber as Integer	222
* 6.10.83 CreateImageAsyncDelegateMBS(img as CGImageMBS, JPEGData as MemoryBlock, Tag as Variant)	224
– 6.11.1 class CGDisplayModeMBS	225
* 6.11.3 Constructor	225
* 6.11.5 Handle as Integer	225
* 6.11.6 Height as Integer	225
* 6.11.7 IODisplayModeID as Int32	226
* 6.11.8 IOFlags as UInt32	226
* 6.11.9 IsUsableForDesktopGUI as boolean	226
* 6.11.10 PixelEncoding as string	226
* 6.11.11 PixelHeight as Integer	227
* 6.11.12 PixelWidth as Integer	227
* 6.11.13 RefreshRate as Double	227
* 6.11.14 Width as Integer	228
– 6.12.1 class CGDisplayReconfigurationEventMBS	229
* 6.12.3 DisplayReconfiguration(DisplayID as Integer, flags as Integer)	229
– 6.13.1 class CGDisplayStreamEventMBS	231
* 6.13.3 Constructor(DisplayHandle as Integer, outputWidth as Integer, outputHeight as Integer, pixelFormat as Integer = 0, properties as dictionary = nil)	231

* 6.13.4	kCGDisplayStreamColorSpace as String	231
* 6.13.5	kCGDisplayStreamDestinationRect as String	232
* 6.13.6	kCGDisplayStreamMinimumFrameTime as String	232
* 6.13.7	kCGDisplayStreamPreserveAspectRatio as String	232
* 6.13.8	kCGDisplayStreamQueueDepth as String	232
* 6.13.9	kCGDisplayStreamShowCursor as String	233
* 6.13.10	kCGDisplayStreamSourceRect as String	233
* 6.13.11	kCGDisplayStreamYCbCrMatrix as String	233
* 6.13.12	kCGDisplayStreamYCbCrMatrix_ITU_R_601_4 as String	233
* 6.13.13	kCGDisplayStreamYCbCrMatrix_ITU_R_709_2 as String	233
* 6.13.14	kCGDisplayStreamYCbCrMatrix_SMPTE_240M_1995 as String	234
* 6.13.15	Start	234
* 6.13.16	Stop	234
* 6.13.18	Handle as Integer	234
* 6.13.19	Lasterror as Integer	234
* 6.13.21	FrameAvailable(Status as Integer, displayTime as UInt64, frameSurfaceHandle as Integer, Update as CGDisplayStreamUpdateMBS)	235
– 6.14.1	class CGDisplayStreamUpdateMBS	236
* 6.14.3	getRects(type as Integer) as CGRectMBS()	236
* 6.14.5	CImage as Variant	236
* 6.14.6	DeltaX as Double	236
* 6.14.7	DeltaY as Double	237
* 6.14.8	DropCount as Integer	237
* 6.14.9	Handle as Integer	237
* 6.14.10	IOSurfaceHandle as Integer	237
– 6.15.1	class CGDisplayTransferFormulaMBS	238
* 6.15.3	BlueGamma as Double	238
* 6.15.4	BlueMax as Double	238
* 6.15.5	BlueMin as Double	238
* 6.15.6	GreenGamma as Double	238
* 6.15.7	GreenMax as Double	239
* 6.15.8	GreenMin as Double	239
* 6.15.9	RedGamma as Double	239
* 6.15.10	RedMax as Double	239
* 6.15.11	RedMin as Double	239
– 6.16.1	class CGFontMBS	240
* 6.16.3	CreateWithDataProvider(CGDataProvider as Variant) as CGFontMBS	240
* 6.16.4	CreateWithFontName(name as string) as CGFontMBS	240
* 6.16.5	CreateWithPlatformFont(ATSFonHandle as Integer) as CGFontMBS	241
* 6.16.7	Ascent as Integer	241
* 6.16.8	CapHeight as Integer	241

	19
* 6.16.9 Descent as Integer	242
* 6.16.10 FontBBox as CGRectMBS	242
* 6.16.11 FullName as String	242
* 6.16.12 Handle as Integer	243
* 6.16.13 ItalicAngle as Double	243
* 6.16.14 Leading as Integer	243
* 6.16.15 NumberOfGlyphs as UInt64	244
* 6.16.16 PostScriptName as String	244
* 6.16.17 StemV as Double	244
* 6.16.18 UnitsPerEm as Integer	244
* 6.16.19 XHeight as Integer	245
– 6.17.1 class CGFunctionMBS	246
* 6.17.3 Create(domainDimension as Integer, domain as memoryblock, rangeDimension as Integer, range as memoryblock)	246
* 6.17.5 Handle as Integer	247
* 6.17.7 Evaluate(Input as memoryblock, Output as memoryblock)	247
– 6.18.1 class CGGradientMBS	248
* 6.18.3 CreateWithColorComponents(colorSpace as CGColorSpaceMBS, components() as Double) as CGGradientMBS	248
* 6.18.4 CreateWithColorComponents(colorSpace as CGColorSpaceMBS, components() as Double, locations() as Double) as CGGradientMBS	249
* 6.18.5 CreateWithColors(colorSpace as CGColorSpaceMBS, colors() as CGColorMBS) as CGGradientMBS	251
* 6.18.6 CreateWithColors(colorSpace as CGColorSpaceMBS, colors() as CGColorMBS, locations() as Double) as CGGradientMBS	252
* 6.18.8 Handle as Integer	253
– 6.19.1 class CGImageDestinationMBS	255
* 6.19.3 AddImage(image as CGImageMBS, properties as dictionary=nil)	256
* 6.19.4 AddImageCF(image as CGImageMBS, properties as Variant = nil)	257
* 6.19.5 AddImageFromSource(source as CGImageSourceMBS, index as Integer, options as dictionary = nil)	257
* 6.19.6 AddImageFromSourceCF(source as CGImageSourceMBS, index as Integer, options as Variant = nil)	257
* 6.19.7 Constructor(file as folderitem, type as string, count as Integer = 1)	258
* 6.19.8 Constructor(type as string, count as Integer = 1)	258
* 6.19.9 Constructor(url as string, type as string, count as Integer = 1)	259
* 6.19.10 CreateWithData(type as string, count as Integer = 1) as CGImageDestinationMBS	259
* 6.19.11 CreateWithFile(file as folderitem, type as string, count as Integer = 1) as CGImageDestinationMBS	260
* 6.19.12 CreateWithURL(url as string, type as string, count as Integer = 1) as CGImageDestinationMBS	261

* 6.19.13 Data as string	261
* 6.19.14 Finalize as boolean	261
* 6.19.15 FinalizeMT as boolean	262
* 6.19.16 kCGImageDestinationBackgroundColor as string	262
* 6.19.17 kCGImageDestinationDateTime as string	263
* 6.19.18 kCGImageDestinationLossyCompressionQuality as string	263
* 6.19.19 kCGImageDestinationMergeMetadata as string	263
* 6.19.20 kCGImageDestinationMetadata as string	264
* 6.19.21 kCGImageDestinationOrientation as string	264
* 6.19.22 kCGImageMetadataShouldExcludeXMP as string	266
* 6.19.23 SetPropertyOptions(options as dictionary = nil)	266
* 6.19.24 SetPropertyOptionsCF(options as Variant)	267
* 6.19.25 TypeIdentifiers as string()	267
* 6.19.27 Handle as Integer	268
– 6.20.1 class CGImageMBS	269
* 6.20.3 AsNSImageMBS as variant	269
* 6.20.4 Constructor	270
* 6.20.5 Copy as CGImageMBS	270
* 6.20.6 Copy(r as CGRectMBS) as CGImageMBS	270
* 6.20.7 CopyWithColorSpace(profile as CGColorSpaceMBS) as CGImageMBS	270
* 6.20.8 CopyWithMask(mask as CGImageMBS) as CGImageMBS	271
* 6.20.9 CreateImage(pic as picture) as CGImageMBS	272
* 6.20.10 CreateImage(pic as picture, mask as picture) as CGImageMBS	272
* 6.20.11 CreateImageFromJPEGDataProvider(dataprovider as Variant, decode as memory-block, shouldInterpolate as boolean, intent as Integer) as CGImageMBS	273
* 6.20.12 CreateImageFromPNGDataProvider(dataprovider as Variant, decode as memory-block, shouldInterpolate as boolean, intent as Integer) as CGImageMBS	274
* 6.20.13 CreateImageWithFile(file as folderitem) as CGImageMBS	275
* 6.20.14 CreateImageWithHandle(handle as Integer) as CGImageMBS	275
* 6.20.15 DataProvider as Variant	275
* 6.20.16 DecodeArray as memoryblock	275
* 6.20.17 JPEGData(Compression as Integer = 90) as MemoryBlock	275
* 6.20.18 Picture(ColorSpace as CGColorSpaceMBS = nil) as Picture	276
* 6.20.19 PictureScaled(OutputWidth as Integer, OutputHeight as Integer, ColorSpace as CG-ColorSpaceMBS = nil) as Picture	276
* 6.20.20 PNGData as MemoryBlock	277
* 6.20.21 ReleaseHandle	277
* 6.20.22 RetainHandle	277
* 6.20.24 AlphaInfo as Integer	277
* 6.20.25 BitmapInfo as Integer	278
* 6.20.26 BitsPerComponent as Integer	278
* 6.20.27 BitsPerPixel as Integer	278

	21
* 6.20.28 BytesPerRow as Integer	278
* 6.20.29 ColorSpace as CGColorSpaceMBS	278
* 6.20.30 handle as Integer	279
* 6.20.31 height as Integer	279
* 6.20.32 ImageIsMask as boolean	279
* 6.20.33 RenderingIntent as Integer	280
* 6.20.34 RetainCount as Integer	280
* 6.20.35 ShouldInterpolate as boolean	280
* 6.20.36 width as Integer	280
– 6.21.1 class CGImageMetadataMBS	282
* 6.21.3 Constructor	282
* 6.21.4 Constructor(metadata as CGImageMetadataMBS)	282
* 6.21.5 CreateFromXMPData(data as MemoryBlock) as CGImageMetadataMBS	282
* 6.21.6 CreateXMPData(options as Dictionary = nil) as MemoryBlock	283
* 6.21.7 kCFErrorDomainCGImageMetadata as String	283
* 6.21.8 kCGImageMetadataEnumerateRecursively as String	283
* 6.21.9 kCGImageMetadataNamespaceDublinCore as String	283
* 6.21.10 kCGImageMetadataNamespaceExif as String	284
* 6.21.11 kCGImageMetadataNamespaceExifAux as String	284
* 6.21.12 kCGImageMetadataNamespaceExifEX as String	284
* 6.21.13 kCGImageMetadataNamespaceIPTCCore as String	284
* 6.21.14 kCGImageMetadataNamespaceIPTCExtension as String	284
* 6.21.15 kCGImageMetadataNamespacePhotoshop as String	284
* 6.21.16 kCGImageMetadataNamespaceTIFF as String	285
* 6.21.17 kCGImageMetadataNamespaceXMPBasic as String	285
* 6.21.18 kCGImageMetadataNamespaceXMPRights as String	285
* 6.21.19 kCGImageMetadataPrefixDublinCore as String	285
* 6.21.20 kCGImageMetadataPrefixExif as String	285
* 6.21.21 kCGImageMetadataPrefixExifAux as String	286
* 6.21.22 kCGImageMetadataPrefixExifEX as String	286
* 6.21.23 kCGImageMetadataPrefixIPTCCore as String	286
* 6.21.24 kCGImageMetadataPrefixIPTCExtension as String	286
* 6.21.25 kCGImageMetadataPrefixPhotoshop as String	286
* 6.21.26 kCGImageMetadataPrefixTIFF as String	286
* 6.21.27 kCGImageMetadataPrefixXMPBasic as String	287
* 6.21.28 kCGImageMetadataPrefixXMPRights as String	287
* 6.21.29 StringValueWithPath(parent as CGImageMetadataTagMBS = nil, path as String) as String	287
* 6.21.30 TagMatchingImageProperty(dictionaryName as String, propertyName as String) as Variant	287
* 6.21.31 Tags as CGImageMetadataTagMBS()	288

* 6.21.32	Tags(rootPath as string, options as Dictionary = nil) as CGImageMetadataTagMBS()	288
* 6.21.33	TagWithPath(parent as CGImageMetadataTagMBS = nil, path as string) as Variant	289
* 6.21.35	Handle as Integer	290
– 6.22.1	class CGImageMetadataTagMBS	291
* 6.22.3	Constructor	291
* 6.22.4	Create(XMLNS as String, Prefix as String, Name as String, Type as Integer, Value as Variant) as CGImageMetadataTagMBS	291
* 6.22.5	Qualifiers as CGImageMetadataTagMBS()	292
* 6.22.7	Handle as Integer	292
* 6.22.8	Name as String	292
* 6.22.9	Namespace_ as String	292
* 6.22.10	Prefix as String	293
* 6.22.11	Type as Integer	293
* 6.22.12	Value as Variant	293
– 6.23.1	class CGImageSourceMBS	295
* 6.23.3	Constructor(data as string, options as dictionary = nil)	296
* 6.23.4	Constructor(file as folderitem, options as dictionary = nil)	296
* 6.23.5	Constructor(options as dictionary = nil)	296
* 6.23.6	CreateImageAtIndex(index as Integer, options as dictionary = nil) as CGImageMBS	297
* 6.23.7	CreateIncremental(options as dictionary=nil) as CGImageSourceMBS	298
* 6.23.8	CreateThumbnailAtIndex(index as Integer, options as dictionary = nil) as CGImageMBS	298
* 6.23.9	CreateThumbnailMT(data as MemoryBlock, ImageIndex as integer = 0, Options as dictionary = nil) as CGImageMBS	299
* 6.23.10	CreateThumbnailMT(data as String, ImageIndex as integer = 0, Options as dictionary = nil) as CGImageMBS	299
* 6.23.11	CreateThumbnailMT(file as FolderItem, ImageIndex as integer = 0, Options as dictionary = nil) as CGImageMBS	300
* 6.23.12	CreateWithData(data as string, options as dictionary=nil) as CGImageSourceMBS	300
* 6.23.13	CreateWithFile(file as folderitem, options as dictionary=nil) as CGImageSourceMBS	301
* 6.23.14	CreateWithURL(url as string, options as dictionary=nil) as CGImageSourceMBS	301
* 6.23.15	kCGImageProperty8BIMDictionary as string	302
* 6.23.16	kCGImageProperty8BIMLayerNames as string	302
* 6.23.17	kCGImagePropertyCIFFCameraSerialNumber as string	302
* 6.23.18	kCGImagePropertyCIFFContinuousDrive as string	302
* 6.23.19	kCGImagePropertyCIFFDescription as string	303
* 6.23.20	kCGImagePropertyCIFFDictionary as string	303
* 6.23.21	kCGImagePropertyCIFFFirmware as string	303

* 6.23.22 kCGImagePropertyCIFFFlashExposureComp as string	303
* 6.23.23 kCGImagePropertyCIFFFocusMode as string	303
* 6.23.24 kCGImagePropertyCIFFImageFileName as string	304
* 6.23.25 kCGImagePropertyCIFFImageName as string	304
* 6.23.26 kCGImagePropertyCIFFImageSerialNumber as string	304
* 6.23.27 kCGImagePropertyCIFFLensMaxMM as string	304
* 6.23.28 kCGImagePropertyCIFFLensMinMM as string	304
* 6.23.29 kCGImagePropertyCIFFLensModel as string	304
* 6.23.30 kCGImagePropertyCIFFMeasuredEV as string	305
* 6.23.31 kCGImagePropertyCIFFMeteringMode as string	305
* 6.23.32 kCGImagePropertyCIFFOwnerName as string	305
* 6.23.33 kCGImagePropertyCIFFRecordID as string	305
* 6.23.34 kCGImagePropertyCIFFReleaseMethod as string	305
* 6.23.35 kCGImagePropertyCIFFReleaseTiming as string	306
* 6.23.36 kCGImagePropertyCIFFSelfTimingTime as string	306
* 6.23.37 kCGImagePropertyCIFFShootingMode as string	306
* 6.23.38 kCGImagePropertyCIFFWhiteBalanceIndex as string	306
* 6.23.39 kCGImagePropertyColorModel as string	306
* 6.23.40 kCGImagePropertyColorModelCMYK as string	307
* 6.23.41 kCGImagePropertyColorModelGray as string	307
* 6.23.42 kCGImagePropertyColorModelLab as string	307
* 6.23.43 kCGImagePropertyColorModelRGB as string	307
* 6.23.44 kCGImagePropertyDepth as string	307
* 6.23.45 kCGImagePropertyDNGBackwardVersion as string	307
* 6.23.46 kCGImagePropertyDNGCameraSerialNumber as string	308
* 6.23.47 kCGImagePropertyDNGDictionary as string	308
* 6.23.48 kCGImagePropertyDNGLensInfo as string	308
* 6.23.49 kCGImagePropertyDNGLocalizedCameraModel as string	308
* 6.23.50 kCGImagePropertyDNGUniqueCameraModel as string	308
* 6.23.51 kCGImagePropertyDNGVersion as string	309
* 6.23.52 kCGImagePropertyDPIHeight as string	309
* 6.23.53 kCGImagePropertyDPIWidth as string	309
* 6.23.54 kCGImagePropertyExifApertureValue as string	309
* 6.23.55 kCGImagePropertyExifAuxDictionary as string	309
* 6.23.56 kCGImagePropertyExifAuxFirmware as string	309
* 6.23.57 kCGImagePropertyExifAuxFlashCompensation as string	310
* 6.23.58 kCGImagePropertyExifAuxImageNumber as string	310
* 6.23.59 kCGImagePropertyExifAuxLensID as string	310
* 6.23.60 kCGImagePropertyExifAuxLensInfo as string	310
* 6.23.61 kCGImagePropertyExifAuxLensModel as string	310
* 6.23.62 kCGImagePropertyExifAuxLensSerialNumber as string	311
* 6.23.63 kCGImagePropertyExifAuxOwnerName as string	311

* 6.23.64 kCGImagePropertyExifAuxSerialNumber as string	311
* 6.23.65 kCGImagePropertyExifBrightnessValue as string	311
* 6.23.66 kCGImagePropertyExifCFAPattern as string	311
* 6.23.67 kCGImagePropertyExifColorSpace as string	312
* 6.23.68 kCGImagePropertyExifComponentsConfiguration as string	312
* 6.23.69 kCGImagePropertyExifCompressedBitsPerPixel as string	312
* 6.23.70 kCGImagePropertyExifContrast as string	312
* 6.23.71 kCGImagePropertyExifCustomRendered as string	312
* 6.23.72 kCGImagePropertyExifDateTimeDigitized as string	313
* 6.23.73 kCGImagePropertyExifDateTimeOriginal as string	313
* 6.23.74 kCGImagePropertyExifDeviceSettingDescription as string	313
* 6.23.75 kCGImagePropertyExifDictionary as string	313
* 6.23.76 kCGImagePropertyExifDigitalZoomRatio as string	313
* 6.23.77 kCGImagePropertyExifExposureBiasValue as string	313
* 6.23.78 kCGImagePropertyExifExposureIndex as string	314
* 6.23.79 kCGImagePropertyExifExposureMode as string	314
* 6.23.80 kCGImagePropertyExifExposureProgram as string	314
* 6.23.81 kCGImagePropertyExifExposureTime as string	314
* 6.23.82 kCGImagePropertyExifFileSource as string	314
* 6.23.83 kCGImagePropertyExifFlash as string	315
* 6.23.84 kCGImagePropertyExifFlashEnergy as string	315
* 6.23.85 kCGImagePropertyExifFlashPixVersion as string	315
* 6.23.86 kCGImagePropertyExifFNumber as string	315
* 6.23.87 kCGImagePropertyExifFocalLength as string	315
* 6.23.88 kCGImagePropertyExifFocalLenIn35mmFilm as string	315
* 6.23.89 kCGImagePropertyExifFocalPlaneResolutionUnit as string	316
* 6.23.90 kCGImagePropertyExifFocalPlaneXResolution as string	316
* 6.23.91 kCGImagePropertyExifFocalPlaneYResolution as string	316
* 6.23.92 kCGImagePropertyExifGainControl as string	316
* 6.23.93 kCGImagePropertyExifGamma as string	316
* 6.23.94 kCGImagePropertyExifImageUniqueID as string	317
* 6.23.95 kCGImagePropertyExifISOSpeedRatings as string	317
* 6.23.96 kCGImagePropertyExifLightSource as string	317
* 6.23.97 kCGImagePropertyExifMakerNote as string	317
* 6.23.98 kCGImagePropertyExifMaxApertureValue as string	317
* 6.23.99 kCGImagePropertyExifMeteringMode as string	317
* 6.23.100 kCGImagePropertyExifOECF as string	318
* 6.23.101 kCGImagePropertyExifPixelXDimension as string	318
* 6.23.102 kCGImagePropertyExifPixelYDimension as string	318
* 6.23.103 kCGImagePropertyExifRelatedSoundFile as string	319
* 6.23.104 kCGImagePropertyExifSaturation as string	319
* 6.23.105 kCGImagePropertyExifSceneCaptureType as string	319

* 6.23.106 kCGImagePropertyExifSceneType as string	319
* 6.23.107 kCGImagePropertyExifSensingMethod as string	319
* 6.23.108 kCGImagePropertyExifSharpness as string	319
* 6.23.109 kCGImagePropertyExifShutterSpeedValue as string	320
* 6.23.110 kCGImagePropertyExifSpatialFrequencyResponse as string	320
* 6.23.111 kCGImagePropertyExifSpectralSensitivity as string	320
* 6.23.112 kCGImagePropertyExifSubjectArea as string	320
* 6.23.113 kCGImagePropertyExifSubjectDistance as string	320
* 6.23.114 kCGImagePropertyExifSubjectDistRange as string	321
* 6.23.115 kCGImagePropertyExifSubjectLocation as string	321
* 6.23.116 kCGImagePropertyExifSubsecTime as string	321
* 6.23.117 kCGImagePropertyExifSubsecTimeDigitized as string	321
* 6.23.118 kCGImagePropertyExifSubsecTimeOriginal as string	321
* 6.23.119 kCGImagePropertyExifUserComment as string	321
* 6.23.120 kCGImagePropertyExifVersion as string	322
* 6.23.121 kCGImagePropertyExifWhiteBalance as string	322
* 6.23.122 kCGImagePropertyFileSize as string	322
* 6.23.123 kCGImagePropertyGIFDelayTime as string	322
* 6.23.124 kCGImagePropertyGIFDictionary as string	322
* 6.23.125 kCGImagePropertyGIFHasGlobalColorMap as string	323
* 6.23.126 kCGImagePropertyGIFImageColorMap as string	323
* 6.23.127 kCGImagePropertyGIFLoopCount as string	323
* 6.23.128 kCGImagePropertyGIFUnclampedDelayTime as string	323
* 6.23.129 kCGImagePropertyGPSAltitude as string	323
* 6.23.130 kCGImagePropertyGPSAltitudeRef as string	324
* 6.23.131 kCGImagePropertyGPSAreaInformation as string	324
* 6.23.132 kCGImagePropertyGPSDateStamp as string	324
* 6.23.133 kCGImagePropertyGPSDestBearing as string	324
* 6.23.134 kCGImagePropertyGPSDestBearingRef as string	324
* 6.23.135 kCGImagePropertyGPSDestDistance as string	324
* 6.23.136 kCGImagePropertyGPSDestDistanceRef as string	325
* 6.23.137 kCGImagePropertyGPSDestLatitude as string	325
* 6.23.138 kCGImagePropertyGPSDestLatitudeRef as string	325
* 6.23.139 kCGImagePropertyGPSDestLongitude as string	325
* 6.23.140 kCGImagePropertyGPSDestLongitudeRef as string	325
* 6.23.141 kCGImagePropertyGPSDictionary as string	326
* 6.23.142 kCGImagePropertyGPSDifferential as string	326
* 6.23.143 kCGImagePropertyGPSDOP as string	326
* 6.23.144 kCGImagePropertyGPSImgDirection as string	326
* 6.23.145 kCGImagePropertyGPSImgDirectionRef as string	326
* 6.23.146 kCGImagePropertyGPSLatitude as string	326
* 6.23.147 kCGImagePropertyGPSLatitudeRef as string	327

* 6.23.148 kCGImagePropertyGPSLongitude as string	327
* 6.23.149 kCGImagePropertyGPSLongitudeRef as string	327
* 6.23.150 kCGImagePropertyGPSMapDatum as string	327
* 6.23.151 kCGImagePropertyGPSMeasureMode as string	327
* 6.23.152 kCGImagePropertyGPSProcessingMethod as string	328
* 6.23.153 kCGImagePropertyGPSSatellites as string	328
* 6.23.154 kCGImagePropertyGPSSpeed as string	328
* 6.23.155 kCGImagePropertyGPSSpeedRef as string	328
* 6.23.156 kCGImagePropertyGPSStatus as string	328
* 6.23.157 kCGImagePropertyGPSTimeStamp as string	328
* 6.23.158 kCGImagePropertyGPSTrack as string	329
* 6.23.159 kCGImagePropertyGPSTrackRef as string	329
* 6.23.160 kCGImagePropertyGPSVersion as string	329
* 6.23.161 kCGImagePropertyHasAlpha as string	329
* 6.23.162 kCGImagePropertyIPTCActionAdvised as string	329
* 6.23.163 kCGImagePropertyIPTCByline as string	330
* 6.23.164 kCGImagePropertyIPTCBylineTitle as string	330
* 6.23.165 kCGImagePropertyIPTCCaptionAbstract as string	330
* 6.23.166 kCGImagePropertyIPTCCategory as string	330
* 6.23.167 kCGImagePropertyIPTCCity as string	330
* 6.23.168 kCGImagePropertyIPTCCContact as string	330
* 6.23.169 kCGImagePropertyIPTCCContactInfoAddress as string	331
* 6.23.170 kCGImagePropertyIPTCCContactInfoCity as string	331
* 6.23.171 kCGImagePropertyIPTCCContactInfoCountry as string	331
* 6.23.172 kCGImagePropertyIPTCCContactInfoEmails as string	331
* 6.23.173 kCGImagePropertyIPTCCContactInfoPhones as string	331
* 6.23.174 kCGImagePropertyIPTCCContactInfoPostalCode as string	332
* 6.23.175 kCGImagePropertyIPTCCContactInfoStateProvince as string	332
* 6.23.176 kCGImagePropertyIPTCCContactInfoWebURLs as string	332
* 6.23.177 kCGImagePropertyIPTCCContentLocationCode as string	332
* 6.23.178 kCGImagePropertyIPTCCContentLocationName as string	332
* 6.23.179 kCGImagePropertyIPTCCopyrightNotice as string	333
* 6.23.180 kCGImagePropertyIPTCCountryPrimaryLocationCode as string	333
* 6.23.181 kCGImagePropertyIPTCCountryPrimaryLocationName as string	333
* 6.23.182 kCGImagePropertyIPTCCreatorContactInfo as string	333
* 6.23.183 kCGImagePropertyIPTCCredit as string	333
* 6.23.184 kCGImagePropertyIPTCDateCreated as string	334
* 6.23.185 kCGImagePropertyIPTCDictionary as string	334
* 6.23.186 kCGImagePropertyIPTCDigitalCreationDate as string	335
* 6.23.187 kCGImagePropertyIPTCDigitalCreationTime as string	335
* 6.23.188 kCGImagePropertyIPTCEditorialUpdate as string	335
* 6.23.189 kCGImagePropertyIPTCEditStatus as string	336

* 6.23.190 kCGImagePropertyIPTCExpirationDate as string	336
* 6.23.191 kCGImagePropertyIPTCExpirationTime as string	336
* 6.23.192 kCGImagePropertyIPTCFixtureIdentifier as string	336
* 6.23.193 kCGImagePropertyIPTCHeadline as string	336
* 6.23.194 kCGImagePropertyIPTCImageOrientation as string	336
* 6.23.195 kCGImagePropertyIPTCImageType as string	337
* 6.23.196 kCGImagePropertyIPTCKeywords as string	337
* 6.23.197 kCGImagePropertyIPTCLanguageIdentifier as string	337
* 6.23.198 kCGImagePropertyIPTCObjectAttributeReference as string	337
* 6.23.199 kCGImagePropertyIPTCObjectCycle as string	337
* 6.23.200 kCGImagePropertyIPTCObjectName as string	338
* 6.23.201 kCGImagePropertyIPTCObjectTypeReference as string	338
* 6.23.202 kCGImagePropertyIPTCOriginalTransmissionReference as string	338
* 6.23.203 kCGImagePropertyIPTCOriginatingProgram as string	339
* 6.23.204 kCGImagePropertyIPTCProgramVersion as string	339
* 6.23.205 kCGImagePropertyIPTCProvinceState as string	339
* 6.23.206 kCGImagePropertyIPTCReferenceDate as string	339
* 6.23.207 kCGImagePropertyIPTCReferenceNumber as string	339
* 6.23.208 kCGImagePropertyIPTCReferenceService as string	339
* 6.23.209 kCGImagePropertyIPTCReleaseDate as string	340
* 6.23.210 kCGImagePropertyIPTCReleaseTime as string	340
* 6.23.211 kCGImagePropertyIPTCRightsUsageTerms as string	340
* 6.23.212 kCGImagePropertyIPTCScene as string	340
* 6.23.213 kCGImagePropertyIPTCSource as string	340
* 6.23.214 kCGImagePropertyIPTCSpecialInstructions as string	341
* 6.23.215 kCGImagePropertyIPTCStarRating as string	341
* 6.23.216 kCGImagePropertyIPTCSubjectReference as string	341
* 6.23.217 kCGImagePropertyIPTCSubLocation as string	341
* 6.23.218 kCGImagePropertyIPTCSupplementalCategory as string	341
* 6.23.219 kCGImagePropertyIPTCTimeCreated as string	341
* 6.23.220 kCGImagePropertyIPTCUrgency as string	342
* 6.23.221 kCGImagePropertyIPTCWriterEditor as string	342
* 6.23.222 kCGImagePropertyIsFloat as string	342
* 6.23.223 kCGImagePropertyIsIndexed as string	342
* 6.23.224 kCGImagePropertyJFIFDensityUnit as string	342
* 6.23.225 kCGImagePropertyJFIFDictionary as string	343
* 6.23.226 kCGImagePropertyJFIFIsProgressive as string	343
* 6.23.227 kCGImagePropertyJFIFVersion as string	343
* 6.23.228 kCGImagePropertyJFIFXDensity as string	343
* 6.23.229 kCGImagePropertyJFIFYDensity as string	343
* 6.23.230 kCGImagePropertyMakerCanonAspectRatioInfo as string	343
* 6.23.231 kCGImagePropertyMakerCanonCameraSerialNumber as string	344

* 6.23.232 kCGImagePropertyMakerCanonContinuousDrive as string	344
* 6.23.233 kCGImagePropertyMakerCanonDictionary as string	344
* 6.23.234 kCGImagePropertyMakerCanonFirmware as string	344
* 6.23.235 kCGImagePropertyMakerCanonFlashExposureComp as string	344
* 6.23.236 kCGImagePropertyMakerCanonImageSerialNumber as string	345
* 6.23.237 kCGImagePropertyMakerCanonLensModel as string	345
* 6.23.238 kCGImagePropertyMakerCanonOwnerName as string	345
* 6.23.239 kCGImagePropertyMakerFujiDictionary as string	345
* 6.23.240 kCGImagePropertyMakerMinoltaDictionary as string	345
* 6.23.241 kCGImagePropertyMakerNikonCameraSerialNumber as string	345
* 6.23.242 kCGImagePropertyMakerNikonColorMode as string	346
* 6.23.243 kCGImagePropertyMakerNikonDictionary as string	346
* 6.23.244 kCGImagePropertyMakerNikonDigitalZoom as string	346
* 6.23.245 kCGImagePropertyMakerNikonFlashExposureComp as string	346
* 6.23.246 kCGImagePropertyMakerNikonFlashSetting as string	346
* 6.23.247 kCGImagePropertyMakerNikonFocusDistance as string	347
* 6.23.248 kCGImagePropertyMakerNikonFocusMode as string	347
* 6.23.249 kCGImagePropertyMakerNikonImageAdjustment as string	347
* 6.23.250 kCGImagePropertyMakerNikonISOSelection as string	347
* 6.23.251 kCGImagePropertyMakerNikonISOSetting as string	347
* 6.23.252 kCGImagePropertyMakerNikonLensAdapter as string	347
* 6.23.253 kCGImagePropertyMakerNikonLensInfo as string	348
* 6.23.254 kCGImagePropertyMakerNikonLensType as string	348
* 6.23.255 kCGImagePropertyMakerNikonQuality as string	348
* 6.23.256 kCGImagePropertyMakerNikonSharpenMode as string	348
* 6.23.257 kCGImagePropertyMakerNikonShootingMode as string	348
* 6.23.258 kCGImagePropertyMakerNikonShutterCount as string	349
* 6.23.259 kCGImagePropertyMakerNikonWhiteBalanceMode as string	349
* 6.23.260 kCGImagePropertyMakerOlympusDictionary as string	349
* 6.23.261 kCGImagePropertyMakerPentaxDictionary as string	349
* 6.23.262 kCGImagePropertyOrientation as string	349
* 6.23.263 kCGImagePropertyPixelHeight as string	350
* 6.23.264 kCGImagePropertyPixelWidth as string	350
* 6.23.265 kCGImagePropertyPNGAuthor as string	351
* 6.23.266 kCGImagePropertyPNGChromaticities as string	351
* 6.23.267 kCGImagePropertyPNGCopyright as string	351
* 6.23.268 kCGImagePropertyPNGCreationTime as string	351
* 6.23.269 kCGImagePropertyPNGDescription as string	352
* 6.23.270 kCGImagePropertyPNGDictionary as string	352
* 6.23.271 kCGImagePropertyPNGGamma as string	352
* 6.23.272 kCGImagePropertyPNGInterlaceType as string	352
* 6.23.273 kCGImagePropertyPNGModificationTime as string	352

* 6.23.274 kCGImagePropertyPNGSoftware as string	353
* 6.23.275 kCGImagePropertyPNGsRGBIntent as string	353
* 6.23.276 kCGImagePropertyPNGTitle as string	353
* 6.23.277 kCGImagePropertyPNGXPixelsPerMeter as string	353
* 6.23.278 kCGImagePropertyPNGYPixelsPerMeter as string	353
* 6.23.279 kCGImagePropertyProfileName as string	353
* 6.23.280 kCGImagePropertyRawDictionary as string	354
* 6.23.281 kCGImagePropertyTIFFArtist as string	354
* 6.23.282 kCGImagePropertyTIFFCompression as string	354
* 6.23.283 kCGImagePropertyTIFFCopyright as string	354
* 6.23.284 kCGImagePropertyTIFFDateTime as string	354
* 6.23.285 kCGImagePropertyTIFFDictionary as string	355
* 6.23.286 kCGImagePropertyTIFFDocumentName as string	356
* 6.23.287 kCGImagePropertyTIFFHostComputer as string	356
* 6.23.288 kCGImagePropertyTIFFImageDescription as string	356
* 6.23.289 kCGImagePropertyTIFFMake as string	356
* 6.23.290 kCGImagePropertyTIFFModel as string	356
* 6.23.291 kCGImagePropertyTIFFOrientation as string	357
* 6.23.292 kCGImagePropertyTIFFPhotometricInterpretation as string	358
* 6.23.293 kCGImagePropertyTIFFPrimaryChromaticities as string	358
* 6.23.294 kCGImagePropertyTIFFResolutionUnit as string	358
* 6.23.295 kCGImagePropertyTIFFSoftware as string	358
* 6.23.296 kCGImagePropertyTIFFTransferFunction as string	358
* 6.23.297 kCGImagePropertyTIFFWhitePoint as string	359
* 6.23.298 kCGImagePropertyTIFFXResolution as string	359
* 6.23.299 kCGImagePropertyTIFFYResolution as string	359
* 6.23.300 kCGImageSourceCreateThumbnailFromImageAlways as string	359
* 6.23.301 kCGImageSourceCreateThumbnailFromImageIfAbsent as string	359
* 6.23.302 kCGImageSourceCreateThumbnailWithTransform as string	360
* 6.23.303 kCGImageSourceShouldAllowFloat as string	361
* 6.23.304 kCGImageSourceShouldCache as string	361
* 6.23.305 kCGImageSourceShouldCacheImmediately as string	361
* 6.23.306 kCGImageSourceThumbnailMaxPixelSize as string	361
* 6.23.307 kCGImageSourceTypeIdentifierHint as string	361
* 6.23.308 Properties(options as dictionary = nil) as dictionary	362
* 6.23.309 PropertiesAtIndex(index as Integer, options as dictionary = nil) as dictionary	362
* 6.23.310 PropertiesAtIndexCF(index as Integer, options as Variant = nil) as Variant	363
* 6.23.311 PropertiesCF(options as Variant = nil) as Variant	364
* 6.23.312 StatusAtIndex(index as Integer) as Integer	364
* 6.23.313 TypeIdentifiers as string()	364
* 6.23.314 UpdateData(data as string, final as boolean)	366
* 6.23.316 Count as Integer	366

* 6.23.317 Handle as Integer	366
* 6.23.318 Status as Integer	367
* 6.23.319 Type as string	367
– 6.24.1 class CGLayerMBS	369
* 6.24.3 Constructor(context as CGContextMBS, size as CGSizeMBS, auxiliaryInfo as dictionary = nil)	370
* 6.24.4 Constructor(context as CGContextMBS, width as Double, height as Double, auxiliaryInfo as dictionary = nil)	370
* 6.24.5 Context as CGContextMBS	371
* 6.24.6 Size as CGSizeMBS	371
* 6.24.8 Handle as Integer	371
– 6.25.1 class CGMutableImageMetadataMBS	372
* 6.25.3 Constructor	372
* 6.25.4 Constructor(metadata as CGImageMetadataMBS)	372
* 6.25.5 RegisterNamespaceForPrefix(parent as CGImageMetadataTagMBS = nil, xmlns as String, prefix as String, byref error as CFErrorMBS) as Boolean	372
* 6.25.6 RemoveTagWithPath(parent as CGImageMetadataTagMBS = nil, path as string) as Boolean	373
* 6.25.7 SetTagWithPath(parent as CGImageMetadataTagMBS = nil, path as string, value as CGImageMetadataTagMBS) as Boolean	373
* 6.25.8 SetValueMatchingImageProperty(dictionaryName as String, propertyName as String, value as CGImageMetadataTagMBS) as Boolean	374
* 6.25.9 SetValueWithPath(parent as CGImageMetadataTagMBS = nil, path as string, value as Variant) as Boolean	375
– 6.26.1 class CGMutablePathMBS	377
* 6.26.3 AddArc(transform as CGAffineTransformMBS, x as Double, y as Double, radius as Double, startAngle as Double, endAngle as Double, clockwise as boolean)	377
* 6.26.4 AddArcToPoint(transform as CGAffineTransformMBS, x as Double, y as Double, x2 as Double, y2 as Double, radius as Double)	377
* 6.26.5 AddCurveToPoint(transform as CGAffineTransformMBS, cpx1 as Double, cpy1 as Double, cpx2 as Double, cpy2 as Double, x as Double, y as Double)	377
* 6.26.6 AddEllipseInRect(transform as CGAffineTransformMBS, r as CGRectMBS)	378
* 6.26.7 AddLineToPoint(transform as CGAffineTransformMBS, x as Double, y as Double)	378
* 6.26.8 AddPath(transform as CGAffineTransformMBS, path as CGPathMBS)	378
* 6.26.9 AddQuadCurveToPoint(transform as CGAffineTransformMBS, cpx as Double, cpy as Double, x as Double, y as Double)	378
* 6.26.10 AddRect(transform as CGAffineTransformMBS, r as CGRectMBS)	379
* 6.26.11 CloseSubpath	379
* 6.26.12 Constructor	379
* 6.26.13 ContainsPoint(transform as CGAffineTransformMBS, point as CGPointMBS, eoFill as boolean) as boolean	379
* 6.26.14 MoveToPoint(transform as CGAffineTransformMBS, x as Double, y as Double)	379

	31
– 6.27.1 class CGPathElementMBS	381
* 6.27.3 Point(Index as Integer) as CGPointMBS	381
* 6.27.4 PointX(Index as Integer) as Double	381
* 6.27.5 PointY(Index as Integer) as Double	382
* 6.27.7 PointCount as Integer	382
* 6.27.8 Type as Integer	382
– 6.28.1 class CGPathMBS	383
* 6.28.3 BoundingBox as CGRectMBS	383
* 6.28.4 Copy as CGPathMBS	383
* 6.28.5 CurrentPoint as CGPointMBS	383
* 6.28.6 Elements as CGPathElementMBS()	384
* 6.28.7 EqualToPath(path as CGPathMBS) as boolean	384
* 6.28.8 IsEmpty as boolean	384
* 6.28.9 IsRect(byref rect as CGRectMBS) as boolean	384
* 6.28.10 MutableCopy as CGMutablePathMBS	384
* 6.28.12 Handle as Integer	385
– 6.29.1 class CGPDFArrayMBS	386
* 6.29.3 ArrayValue(index as Integer, byref value as CGPDFArrayMBS) as boolean	386
* 6.29.4 BooleanValue(index as Integer, byref value as boolean) as boolean	386
* 6.29.5 Count as Integer	386
* 6.29.6 DictionaryValue(index as Integer, byref value as CGPDFDictionaryMBS) as boolean	386
* 6.29.7 IntegerValue(index as Integer, byref value as Integer) as boolean	387
* 6.29.8 NameValue(index as Integer, byref value as string) as boolean	387
* 6.29.9 NullValue(index as Integer) as boolean	387
* 6.29.10 ObjectValue(index as Integer, byref value as CGPDFObjectMBS) as boolean	387
* 6.29.11 SingleValue(index as Integer, byref value as Double) as boolean	387
* 6.29.12 StreamValue(index as Integer, byref value as CGPDFStreamMBS) as boolean	388
* 6.29.13 StringValue(index as Integer, byref value as CGPDFStringMBS) as boolean	388
* 6.29.15 Document as CGPDFDocumentMBS	388
* 6.29.16 Handle as Integer	388
– 6.30.1 class CGPDFContextMBS	389
* 6.30.3 AddDestinationAtPoint(name as string, x as Double, y as Double)	390
* 6.30.4 BeginPage(pageInfo as dictionary)	390
* 6.30.5 Close	390
* 6.30.6 EndPage	390
* 6.30.7 kCGPDFContextAllowsCopying as string	390
* 6.30.8 kCGPDFContextAllowsPrinting as string	391
* 6.30.9 kCGPDFContextArtBox as string	391
* 6.30.10 kCGPDFContextAuthor as string	391
* 6.30.11 kCGPDFContextBleedBox as string	391

* 6.30.12	kCGPDFContextCreator as string	392
* 6.30.13	kCGPDFContextCropBox as string	392
* 6.30.14	kCGPDFContextEncryptionKeyLength as string	392
* 6.30.15	kCGPDFContextKeywords as string	392
* 6.30.16	kCGPDFContextMediaBox as string	393
* 6.30.17	kCGPDFContextOutputIntent as string	393
* 6.30.18	kCGPDFContextOutputIntents as string	393
* 6.30.19	kCGPDFContextOwnerPassword as string	393
* 6.30.20	kCGPDFContextSubject as string	394
* 6.30.21	kCGPDFContextTitle as string	394
* 6.30.22	kCGPDFContextTrimBox as string	394
* 6.30.23	kCGPDFContextUserPassword as string	394
* 6.30.24	kCGPDFXDestinationOutputProfile as string	395
* 6.30.25	kCGPDFXInfo as string	395
* 6.30.26	kCGPDFXOutputCondition as string	395
* 6.30.27	kCGPDFXOutputConditionIdentifier as string	395
* 6.30.28	kCGPDFXOutputIntentSubtype as string	396
* 6.30.29	kCGPDFXRegistryName as string	396
* 6.30.30	SetDestinationForRect(name as string, x as Double, y as Double, w as Double, h as Double)	396
* 6.30.31	SetURLForRect(url as string, x as Double, y as Double, w as Double, h as Double)	396
– 6.31.1	class CGPDFDictionaryListMBS	397
* 6.31.3	Close	397
* 6.31.4	Key(index as Integer) as string	397
* 6.31.5	Value(index as Integer) as CGPDFObjectMBS	397
* 6.31.7	Count as Integer	397
* 6.31.8	Document as CGPDFDocumentMBS	398
– 6.32.1	class CGPDFDictionaryMBS	399
* 6.32.3	ArrayValue(key as string, byref value as CGPDFArrayMBS) as boolean	399
* 6.32.4	BooleanValue(key as string, byref value as boolean) as boolean	399
* 6.32.5	Count as Integer	399
* 6.32.6	DictionaryValue(key as string, byref value as CGPDFDictionaryMBS) as boolean	399
* 6.32.7	IntegerValue(key as string, byref value as Integer) as boolean	400
* 6.32.8	List as CGPDFDictionaryListMBS	400
* 6.32.9	NameValue(key as string, byref value as string) as boolean	400
* 6.32.10	ObjectValue(key as string, byref value as CGPDFObjectMBS) as boolean	400
* 6.32.11	SingleValue(key as string, byref value as Double) as boolean	400
* 6.32.12	StreamValue(key as string, byref value as CGPDFStreamMBS) as boolean	400
* 6.32.13	StringValue(key as string, byref value as CGPDFStringMBS) as boolean	401
* 6.32.15	Document as CGPDFDocumentMBS	401

	33
* 6.32.16 Handle as Integer	401
– 6.33.1 class CGPDFDocumentMBS	402
* 6.33.3 ArtBox(page as Integer) as CRectMBS	403
* 6.33.4 BleedBox(page as Integer) as CRectMBS	403
* 6.33.5 Catalog as CGPDFDictionaryMBS	403
* 6.33.6 Constructor(dataProvider as CGDataProviderMBS)	403
* 6.33.7 Constructor(file as folderitem)	404
* 6.33.8 Constructor(Handle as Integer)	404
* 6.33.9 Constructor(url as string)	404
* 6.33.10 CreateWithData(data as Memoryblock) as CGPDFDocumentMBS	405
* 6.33.11 CreateWithData(data as string) as CGPDFDocumentMBS	405
* 6.33.12 CreateWithFile(file as folderitem) as CGPDFDocumentMBS	405
* 6.33.13 CreateWithProvider(dataProvider as CGDataProviderMBS) as CGPDFDocumentMBS	405
* 6.33.14 CreateWithURL(url as string) as CGPDFDocumentMBS	406
* 6.33.15 CropBox(page as Integer) as CRectMBS	406
* 6.33.16 GetID as CGPDFArrayMBS	406
* 6.33.17 GetInfo as CGPDFDictionaryMBS	406
* 6.33.18 MediaBox(page as Integer) as CRectMBS	407
* 6.33.19 Page(index as Integer) as CGPDFPageMBS	407
* 6.33.20 RotationAngle(page as Integer) as Integer	407
* 6.33.21 TrimBox(page as Integer) as CRectMBS	407
* 6.33.22 UnlockWithPassword(name as string) as boolean	408
* 6.33.24 AllowsCopying as Boolean	408
* 6.33.25 AllowsPrinting as Boolean	408
* 6.33.26 handle as Integer	408
* 6.33.27 IsEncrypted as Boolean	408
* 6.33.28 IsUnlocked as Boolean	409
* 6.33.29 MajorVersion as Integer	409
* 6.33.30 MinorVersion as Integer	409
* 6.33.31 PageCount as Integer	409
– 6.34.1 class CGPDFObjectMBS	411
* 6.34.3 ArrayValue(byref value as CGPDFArrayMBS) as boolean	411
* 6.34.4 BooleanValue(byref value as boolean) as boolean	411
* 6.34.5 DictionaryValue(byref value as CGPDFDictionaryMBS) as boolean	411
* 6.34.6 IntegerValue(byref value as Integer) as boolean	411
* 6.34.7 NameValue(byref value as string) as boolean	412
* 6.34.8 SingleValue(byref value as Double) as boolean	412
* 6.34.9 StreamValue(byref value as CGPDFStreamMBS) as boolean	412
* 6.34.10 StringValue(byref value as CGPDFStringMBS) as boolean	412
* 6.34.11 Type as Integer	412

* 6.34.13 Document as CGPDFDocumentMBS	413
* 6.34.14 Handle as Integer	413
– 6.35.1 class CGPDFPageMBS	415
* 6.35.3 ArtBox as CGRectMBS	415
* 6.35.4 BleedBox as CGRectMBS	415
* 6.35.5 CropBox as CGRectMBS	415
* 6.35.6 Dictionary as CGPDFDictionaryMBS	415
* 6.35.7 MediaBox as CGRectMBS	416
* 6.35.8 PageNumber as Integer	416
* 6.35.9 RotationAngle as Integer	416
* 6.35.10 TrimBox as CGRectMBS	416
* 6.35.12 Document as CGPDFDocumentMBS	416
* 6.35.13 Handle as Integer	417
– 6.36.1 class CGPDFStreamMBS	418
* 6.36.3 Data(byref format as Integer) as string	418
* 6.36.4 Dictionary as CGPDFDictionaryMBS	418
* 6.36.6 Document as CGPDFDocumentMBS	418
* 6.36.7 Handle as Integer	419
– 6.37.1 class CGPDFStringMBS	420
* 6.37.3 Bytes as MemoryBlock	420
* 6.37.4 Length as Integer	420
* 6.37.5 Text as string	420
* 6.37.7 Document as CGPDFDocumentMBS	420
* 6.37.8 Handle as Integer	421
– 6.38.1 class CGPointMBS	422
* 6.38.3 ApplyAffineTransform(p as CGAffineTransformMBS) as CGPointMBS	422
* 6.38.4 Binary as MemoryBlock	422
* 6.38.5 Constructor	423
* 6.38.6 Constructor(p as Ptr)	423
* 6.38.7 Constructor(source as CGPointMBS)	423
* 6.38.8 Constructor(x as Double, y as Double)	423
* 6.38.9 Equal(p as CGPointMBS) as boolean	424
* 6.38.10 Make(x as Double, y as Double) as CGPointMBS	424
* 6.38.11 multiply(factor as double) as CGPointMBS	424
* 6.38.12 multiply(x as double, y as double) as CGPointMBS	424
* 6.38.13 Zero as CGPointMBS	424
* 6.38.15 x as Double	425
* 6.38.16 y as Double	425
– 6.39.1 class CGPSConverterMBS	426
* 6.39.3 Abort as boolean	426

* 6.39.4	Constructor(options as Dictionary = nil)	426
* 6.39.5	Convert(provider as CGDataProviderMBS, consumer as CGDataConsumerMBS, options as Dictionary = nil) as boolean	426
* 6.39.6	IsConverting as boolean	427
* 6.39.8	Handle as Integer	427
* 6.39.10	BeginDocument	427
* 6.39.11	BeginPage(PageNumber as Integer, PageInfo as Dictionary)	427
* 6.39.12	EndDocument(success as boolean)	427
* 6.39.13	EndPage(PageNumber as Integer, PageInfo as Dictionary)	428
* 6.39.14	Finished	428
* 6.39.15	Message(message as string)	428
* 6.39.16	Progress	428
– 6.40.1	class CGRectMBS	429
* 6.40.3	ApplyAffineTransform(a as CGAffineTransformMBS) as CGRectMBS	429
* 6.40.4	Binary as MemoryBlock	430
* 6.40.5	Constructor	430
* 6.40.6	Constructor(p as Ptr)	431
* 6.40.7	Constructor(source as CGRectMBS)	431
* 6.40.8	Constructor(x as Double, y as Double, width as Double, height as Double)	431
* 6.40.9	ContainsPoint(r as CGPointMBS) as boolean	431
* 6.40.10	ContainsRect(r as CGRectMBS) as boolean	432
* 6.40.11	Divide(byref slice as CGRectMBS, byref remainder as CGRectMBS, amount as Double, edge as Integer)	432
* 6.40.12	Equal(r as CGRectMBS) as boolean	432
* 6.40.13	Infinite as CGRectMBS	432
* 6.40.14	Inset(dx as Double, dy as Double) as CGRectMBS	432
* 6.40.15	Integral as CGRectMBS	433
* 6.40.16	Intersection(r as CGRectMBS) as CGRectMBS	433
* 6.40.17	IntersectsRect(r as CGRectMBS) as boolean	433
* 6.40.18	IsEmpty as boolean	433
* 6.40.19	IsInfinite as boolean	433
* 6.40.20	IsNull as boolean	434
* 6.40.21	Make(x as Double, y as Double, width as Double, height as Double) as CGRectMBS	434
* 6.40.22	MaxX as Double	434
* 6.40.23	MaxY as Double	434
* 6.40.24	MidX as Double	435
* 6.40.25	MidY as Double	435
* 6.40.26	MinX as Double	435
* 6.40.27	MinY as Double	435
* 6.40.28	multiply(factor as double) as CGRectMBS	435
* 6.40.29	multiply(x as double, y as double) as CGRectMBS	436

* 6.40.30	Null as CGRectMBS	436
* 6.40.31	Offset(dx as Double, dy as Double) as CGRectMBS	436
* 6.40.32	Standardize as CGRectMBS	436
* 6.40.33	Union(r as CGRectMBS) as CGRectMBS	437
* 6.40.34	Zero as CGRectMBS	437
* 6.40.36	height as Double	437
* 6.40.37	left as Double	438
* 6.40.38	Origin as CGPointMBS	438
* 6.40.39	Size as CGSizeMBS	438
* 6.40.40	top as Double	438
* 6.40.41	width as Double	439
– 6.41.1	class CGSConnectionMBS	440
* 6.41.3	CGSWindow(w as window) as CGSWindowMBS	440
* 6.41.4	CGSWindowbyHandle(windowhandle as Integer) as CGSWindowMBS	440
* 6.41.5	FlushAllWindows	441
* 6.41.6	FlushAllWindowsForAllOtherProceses	441
* 6.41.7	FlushAllWindowsForAllProceses	441
* 6.41.8	GetOnScreenWindowList as CGSWindowListMBS	441
* 6.41.9	GetOnScreenWindowListForProcess(PID as Integer) as CGSWindowListMBS	441
* 6.41.10	GetWindowList as CGSWindowListMBS	442
* 6.41.11	GetWindowListForProcess(PID as Integer) as CGSWindowListMBS	442
* 6.41.12	NewTransition(request as CGSTransitionRequestMBS) as CGSTransitionMBS	442
* 6.41.13	RunTransition(request as CGSTransitionRequestMBS, duration as single)	442
* 6.41.14	SetWorkspaceWithTransition(workspace as CGSWorkspaceMBS, transition as Integer, type as Integer, time as single)	443
* 6.41.15	SetWorkspaceWithTransition(workspace as Integer, transition as Integer, type as Integer, time as single)	443
* 6.41.17	Handle as Integer	444
* 6.41.18	Lasterror as Integer	444
* 6.41.19	Workspace as CGSWorkspaceMBS	445
– 6.42.1	class CGScreenRefreshEventMBS	446
* 6.42.3	Initialized as Boolean	446
* 6.42.5	ScreenRefresh(rectCount as Integer, rects() as CGRectMBS)	446
– 6.43.1	class CGScreenUpdateMoveEventMBS	447
* 6.43.3	Initialized as Boolean	447
* 6.43.5	ScreenMove(deltaX as Integer, deltaY as Integer, rectCount as Integer, rects() as CGRectMBS)	447
– 6.44.1	class CGShadingMBS	448
* 6.44.3	Handle as Integer	448
– 6.45.1	class CGSizeMBS	449
* 6.45.3	ApplyAffineTransform(p as CGAffineTransformMBS) as CGSizeMBS	449

* 6.45.4 Binary as MemoryBlock	449
* 6.45.5 Constructor	450
* 6.45.6 Constructor(p as Ptr)	450
* 6.45.7 Constructor(source as CGSizeMBS)	450
* 6.45.8 Constructor(width as Double, height as Double)	450
* 6.45.9 Equal(p as CGSizeMBS) as boolean	451
* 6.45.10 Make(width as Double, height as Double) as CGSizeMBS	451
* 6.45.11 multiply(factor as double) as CGSizeMBS	451
* 6.45.12 multiply(x as double, y as double) as CGSizeMBS	451
* 6.45.13 Zero as CGSizeMBS	451
* 6.45.15 height as Double	452
* 6.45.16 width as Double	452
– 6.46.1 class CGSTransitionMBS	453
* 6.46.3 Invoke(duration as single)	453
* 6.46.4 Release	454
* 6.46.5 Run(duration as single)	454
* 6.46.6 Wait(duration as single)	455
* 6.46.8 Connection as CGSConnectionMBS	456
* 6.46.9 ConnectionHandle as Integer	456
* 6.46.10 Handle as Integer	456
– 6.47.1 class CGSTransitionRequestMBS	457
* 6.47.3 Run(duration as single) as boolean	457
* 6.47.5 Blue as Single	458
* 6.47.6 Green as Single	458
* 6.47.7 HasBackColor as Boolean	458
* 6.47.8 HasBackGround as Boolean	459
* 6.47.9 Red as Single	459
* 6.47.10 TransitionOption as Integer	459
* 6.47.11 TransitionType as Integer	460
* 6.47.12 Win as CGSWindowMBS	460
– 6.48.1 class CGSValueMBS	462
* 6.48.3 IntegerValue as Integer	462
* 6.48.4 StringValue as string	462
* 6.48.6 Connection as CGSConnectionMBS	462
* 6.48.7 ConnectionHandle as Integer	462
* 6.48.8 Handle as Integer	463
– 6.49.1 class CGSWindowListMBS	464
* 6.49.3 Item(index as Integer) as CGSWindowMBS	464
* 6.49.4 PreflightScreenCaptureAccess as Boolean	464
* 6.49.5 RequestScreenCaptureAccess as Boolean	465
* 6.49.7 Connection as CGSConnectionMBS	465

* 6.49.8 ConnectionHandle as Integer	465
* 6.49.9 Count as Integer	465
* 6.49.10 Handle as Integer	466
– 6.50.1 class CGSWindowMBS	467
* 6.50.3 Flush	467
* 6.50.4 Height as Double	467
* 6.50.5 Left as Double	468
* 6.50.6 Level as Integer	468
* 6.50.7 Move(byref x as single, byref y as single)	468
* 6.50.8 Order(mode as Integer, relativeToWindow as CGSWindowMBS=nil)	468
* 6.50.9 Title as string	469
* 6.50.10 Top as Double	469
* 6.50.11 Uncover	469
* 6.50.12 Warp(w as Integer, h as Integer, value as memoryblock)	469
* 6.50.13 Width as Double	470
* 6.50.14 Workspace as CGSWorkspaceMBS	470
* 6.50.16 Connection as CGSConnectionMBS	471
* 6.50.17 ConnectionHandle as Integer	471
* 6.50.18 Handle as Integer	471
* 6.50.19 LastError as Integer	471
* 6.50.20 AffineTransform as CGAffineTransformMBS	472
* 6.50.21 Alpha as single	473
* 6.50.22 EventMask as Integer	473
– 6.51.1 class CGSWorkspaceMBS	474
* 6.51.3 GetWorkspaceWindowList as CGSWindowListMBS	474
* 6.51.4 MoveWindows(target as CGSWorkspaceMBS)	474
* 6.51.6 Connection as CGSConnectionMBS	474
* 6.51.7 ConnectionHandle as Integer	475
* 6.51.8 Handle as Integer	475
* 6.51.9 LastError as Integer	475

	39
• 5 CoreAnimation	69
– 4.1.1 class Control	67
* 4.1.3 CALayerMBS as CALayerMBS	67

• 6 CoreGraphics	97
– ?? Globals	??
* 6.1.4 CGBitmapContextCreateMBS(data as memoryblock, width as Integer, height as Integer, bitsPerComponent as Integer, bytesPerRow as Integer, colorspace as CGColorSpaceMBS, alphaInfo as Integer) as CGBitmapContextMBS	99
* 6.1.8 CGCreateImageFromJPEGDataProviderMBS(dataprovider as Variant, decode as memoryblock, shouldInterpolate as boolean, intent as Integer) as CGImageMBS	103
* 6.1.9 CGCreateImageFromPNGDataProviderMBS(dataprovider as Variant, decode as memoryblock, shouldInterpolate as boolean, intent as Integer) as CGImageMBS	104
* 6.1.10 CGCreateImageMBS(pic as picture) as CGImageMBS	105
* 6.1.11 CGCreateImageMBS(pic as picture, mask as picture) as CGImageMBS	105
* 6.1.12 CGMakePointMBS(x as Double, y as Double) as CGPointMBS	106
* 6.1.13 CGMakeRectMBS(left as Double, top as Double, width as Double, height as Double) as CGRectMBS	106
* 6.1.14 CGMakeSizeMBS(width as Double, height as Double) as CGSizeMBS	106
* 6.1.15 CGNewPDFDocumentMBS(consumer as CGDataConsumerMBS, MediaBox as CGRectMBS, title as string, author as string, creator as string) as CGPDFContextMBS	106
* 6.1.16 CGNewPDFDocumentMBS(file as folderitem, MediaBox as CGRectMBS, title as string, author as string, creator as string) as CGPDFContextMBS	107
* 6.1.1 CGNewPDFDocumentMBS(file as folderitem, MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean) as CGPDFContextMBS	97
* 6.1.2 CGNewPDFDocumentMBS(file as folderitem, MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean, KeyLength as Integer) as CGPDFContextMBS	98
* 6.1.5 CGOpenPDFDocumentMBS(dataprovider as CGDataProviderMBS) as CGPDFDocumentMBS	100
* 6.1.17 CGOpenPDFDocumentMBS(file as folderitem) as CGPDFDocumentMBS	107
* 6.1.6 CGShadingCreateAxialMBS(colorspace as CGColorSpaceMBS, startPoint as CGPointMBS, endPoint as CGPointMBS, func as CGFunctionMBS, extendStart as boolean, extendEnd as boolean) as CGShadingMBS	101
* 6.1.7 CGShadingCreateRadialMBS(colorspace as CGColorSpaceMBS, startPoint as CGPointMBS, startRadius as Double, endPoint as CGPointMBS, endRadius as Double, func as CGFunctionMBS, extendStart as boolean, extendEnd as boolean) as CGShadingMBS	102
* 6.1.3 GetCurrentCGContextMBS as CGContextMBS	98

	41
• 7 CoreText	481
– 7.1.1 class CoreTextMBS	481
* 7.1.3 AvailableFontFamilyNames as string()	481
* 7.1.4 AvailableFontURLs as string()	481
* 7.1.5 AvailablePostScriptNames as string()	482
* 7.1.6 CompareFontFamilyNames(name1 as string, name2 as string) as Integer	482
* 7.1.7 Constructor	482
* 7.1.8 CoreTextVersion as Integer	482
* 7.1.9 CreateFontDescriptorFromData(data as memoryblock) as CTFontDescriptorMBS	483
* 7.1.10 CreateFontDescriptorFromData(data as string) as CTFontDescriptorMBS	483
* 7.1.11 CreateFontDescriptorsFromFile(file as folderitem) as CTFontDescriptorMBS()	483
* 7.1.12 CreateFontDescriptorsFromURL(URL as string) as CTFontDescriptorMBS()	484
* 7.1.13 Destructor	484
* 7.1.14 EnableFontDescriptors(descriptors() as CTFontDescriptorMBS, enable as boolean)	484
* 7.1.15 GetScopeForFile(file as folderitem) as Integer	484
* 7.1.16 GetScopeForURL(URL as string) as Integer	485
* 7.1.17 IsSupportedFontFile(file as folderitem) as boolean	485
* 7.1.18 IsSupportedFontURL(URL as string) as boolean	485
* 7.1.19 kCTBaselineClassAttributeName as string	485
* 7.1.20 kCTBaselineInfoAttributeName as string	486
* 7.1.21 kCTBaselineReferenceInfoAttributeName as string	486
* 7.1.22 kCTCharacterShapeAttributeName as string	486
* 7.1.23 kCTFontAttributeName as string	486
* 7.1.24 kCTFontManagerBundleIdentifier as string	486
* 7.1.25 kCTFontManagerErrorDomain as string	487
* 7.1.26 kCTFontManagerErrorFontURLsKey as string	487
* 7.1.27 kCTFontManagerRegisteredFontsChangedNotification as string	487
* 7.1.28 kCTFontSlantTrait as string	487
* 7.1.29 kCTFontSymbolicTrait as string	487
* 7.1.30 kCTFontWeightTrait as string	488
* 7.1.31 kCTFontWidthTrait as string	488
* 7.1.32 kCTForegroundColorAttributeName as string	488
* 7.1.33 kCTForegroundColorFromContextAttributeName as string	488
* 7.1.34 kCTGlyphInfoAttributeName as string	489
* 7.1.35 kCTKernAttributeName as string	489
* 7.1.36 kCTLanguageAttributeName as string	489
* 7.1.37 kCTLigatureAttributeName as string	489
* 7.1.38 kCTParagraphStyleAttributeName as string	490
* 7.1.39 kCTRunDelegateAttributeName as string	490
* 7.1.40 kCTStrokeColorAttributeName as string	490
* 7.1.41 kCTStrokeWidthAttributeName as string	491

- * 7.1.42 kCTSuperscriptAttributeName as string 491
- * 7.1.43 kCTUnderlineColorAttributeName as string 491
- * 7.1.44 kCTUnderlineStyleAttributeName as string 491
- * 7.1.45 kCTVerticalFormsAttributeName as string 492
- * 7.1.46 kCTWritingDirectionAttributeName as string 492
- * 7.1.47 MatchFontDescriptorsWithProgressHandler(descriptors() as CTFontDescriptorMBS, mandatoryAttributes() as string, tag as Variant = nil) as boolean 492
- * 7.1.48 RegisterFontsForFile(file as folderitem, scope as Integer, byref error as CFErrorMBS) as boolean 493
- * 7.1.49 RegisterFontsForFiles(files() as folderitem, scope as Integer, errors() as CFErrorMBS) as boolean 493
- * 7.1.50 RegisterFontsForURL(URL as string, scope as Integer, byref error as CFErrorMBS) as boolean 493
- * 7.1.51 RegisterFontsForURL(URLs() as string, scope as Integer, errors() as CFErrorMBS) as boolean 494
- * 7.1.52 RegisterGraphicsFont(font as CGFontMBS, byref error as CFErrorMBS) as boolean 494
- * 7.1.53 RequestFonts(descriptors() as CTFontDescriptorMBS, tag as variant = nil) 495
- * 7.1.54 UnregisterFontsForFile(file as folderitem, scope as Integer, byref error as CFErrorMBS) as boolean 495
- * 7.1.55 UnregisterFontsForFiles(files() as folderitem, scope as Integer, errors() as CFErrorMBS) as boolean 496
- * 7.1.56 UnregisterFontsForURL(URL as string, scope as Integer, byref error as CFErrorMBS) as boolean 496
- * 7.1.57 UnregisterFontsForURLs(URLs() as string, scope as Integer, errors() as CFErrorMBS) as boolean 496
- * 7.1.58 UnregisterGraphicsFont(font as CGFontMBS, byref error as CFErrorMBS) as boolean 497
- * 7.1.60 AutoActivationSetting(BundleID as string) as Integer 497
- * 7.1.62 FontCollectionSortDescriptors(first as CTFontDescriptorMBS, second as CTFontDescriptorMBS, tag as Variant) as Integer 497
- * 7.1.63 Progress(state as Integer, progressParameter as Dictionary, tag as Variant) as boolean 498
- * 7.1.64 RequestFontsCompleted(unresolvedFontDescriptors() as CTFontDescriptorMBS, tag as variant) 498

	43
• 10 Printing Mac	615
– 10.2.1 class CPMLanguageInfoMBS	617
* 10.2.3 Level as String	617
* 10.2.4 Release as String	617
* 10.2.5 Version as String	617
– 10.3.1 class CPMPageFormatMBS	619
* 10.3.3 AdjustedPageSize as CPMRectMBS	619
* 10.3.4 AdjustedPaperSize as CPMRectMBS	619
* 10.3.5 Constructor	619
* 10.3.6 CopySettings(Destination as CPMPageFormatMBS)	619
* 10.3.7 CreateDataRepresentation(Format as Integer = 0) as String	619
* 10.3.8 CreateWithDataRepresentation(Data as String) as CPMPageFormatMBS	620
* 10.3.9 PrinterID as String	620
* 10.3.10 UnadjustedPageSize as CPMRectMBS	620
* 10.3.11 UnadjustedPaperSize as CPMRectMBS	621
* 10.3.13 handle as Integer	621
* 10.3.14 Lasterror as Integer	621
* 10.3.15 release as boolean	621
* 10.3.16 Orientation as Integer	621
* 10.3.17 Scale as Double	622
– 10.4.1 class CPMPrinterMBS	624
* 10.4.3 Constructor(name as string)	624
* 10.4.4 CreateFromPrinterID(PrinterID as String) as CPMPrinterMBS	624
* 10.4.5 CreateGenericPrinter as CPMPrinterMBS	624
* 10.4.6 CreateLocalPrinterList as CPMPrinterMBS()	624
* 10.4.7 DescriptionURL as string	625
* 10.4.8 DeviceURI as string	625
* 10.4.9 DriverCreator as String	625
* 10.4.10 DriverReleaseInfo as CPMVersionMBS	626
* 10.4.11 HostName as string	626
* 10.4.12 ID as string	626
* 10.4.13 IndexedPrinterResolution(index as Integer) as CPMResolutionMBS	626
* 10.4.14 IsDefault as boolean	627
* 10.4.15 IsFavorite as boolean	627
* 10.4.16 IsPostScriptCapable as boolean	627
* 10.4.17 IsPostScriptPrinter as boolean	628
* 10.4.18 IsRemote as boolean	628
* 10.4.19 LanguageInfo as CPMLanguageInfoMBS	628
* 10.4.20 Location as string	629
* 10.4.21 MakeAndModelName as string	629
* 10.4.22 Name as string	629

* 10.4.23 ResolutionCount as Integer	629
* 10.4.24 SetDefault	629
* 10.4.25 State as Integer	630
* 10.4.27 handle as Integer	630
* 10.4.28 Lasterror as Integer	630
* 10.4.29 release as boolean	630
– 10.5.1 class CPMPrintSessionMBS	632
* 10.5.3 BeginDocument(settings as CPMPrintSettingsMBS, pageformat as CPMPageFormatMBS)	632
* 10.5.4 BeginDocumentNoDialog(settings as CPMPrintSettingsMBS, pageformat as CPMPageFormatMBS)	632
* 10.5.5 BeginPage(pageformat as CPMPageFormatMBS, rect as CPMRectMBS)	632
* 10.5.6 BeginPageNoDialog(pageformat as CPMPageFormatMBS, rect as CPMRectMBS)	633
* 10.5.7 Constructor	633
* 10.5.8 CreatePrinterList(list() as string)	633
* 10.5.9 CreatePrinterList(list() as string, byref index as Integer, byref currentprinter as CPMPrinterMBS)	633
* 10.5.10 DefaultPageFormat(pageformat as CPMPageFormatMBS)	634
* 10.5.11 DefaultPrintSettings(printsettings as CPMPrintSettingsMBS)	634
* 10.5.12 EndDocument	634
* 10.5.13 EndDocumentNoDialog	634
* 10.5.14 EndPage	635
* 10.5.15 EndPageNoDialog	635
* 10.5.16 GetDestinationFormat(printsettings as CPMPrintSettingsMBS) as String	635
* 10.5.17 GetDestinationLocation(printsettings as CPMPrintSettingsMBS) as String	635
* 10.5.18 GetDestinationType(printsettings as CPMPrintSettingsMBS) as Integer	635
* 10.5.19 kPMDocumentFormatDefault as String	636
* 10.5.20 kPMDocumentFormatPDF as String	636
* 10.5.21 kPMDocumentFormatPostScript as string	636
* 10.5.22 kPMGraphicsContextCoreGraphics as string	636
* 10.5.23 kPMGraphicsContextDefault as string	636
* 10.5.24 PageContext as CGContextMBS	636
* 10.5.25 PageSetupDialog(pageformat as CPMPageFormatMBS) as boolean	637
* 10.5.26 PrintDialog(settings as CPMPrintSettingsMBS, pageformat as CPMPageFormatMBS) as boolean	637
* 10.5.27 SetDestination(printsettings as CPMPrintSettingsMBS, desttype as Integer, destformat as String, desturl as String)	637
* 10.5.28 UseSheets(docWindow as window)	638
* 10.5.29 ValidatePageFormat(pageformat as CPMPageFormatMBS) as boolean	638
* 10.5.30 ValidatePrintSettings(printsettings as CPMPrintSettingsMBS) as boolean	638
* 10.5.32 handle as Integer	638
* 10.5.33 Lasterror as Integer	638

* 10.5.34	release as boolean	639
* 10.5.35	SheetTarget as Window	639
* 10.5.36	CurrentPrinter as CPMPrinterMBS	639
* 10.5.37	CurrentPrinterName as string	639
* 10.5.39	SheetDone(WindowHandle as Integer, accepted as boolean)	640
– 10.6.1	class CPMPrintSettingsMBS	641
* 10.6.3	Constructor	641
* 10.6.4	CopyPrintSettings(dest as CPMPrintSettingsMBS)	641
* 10.6.5	CreateDataRepresentation(Format as Integer = 0) as String	641
* 10.6.6	CreateWithDataRepresentation(Data as String) as CPMPrintSettingsMBS	641
* 10.6.7	Dictionary as Dictionary	642
* 10.6.8	GetPageRange(byref minPage as UInt32, byref maxPage as UInt32)	642
* 10.6.9	Keys as String()	642
* 10.6.10	SetPageRange(minPage as UInt32, maxPage as UInt32)	642
* 10.6.12	handle as Integer	642
* 10.6.13	Lasterror as Integer	643
* 10.6.14	release as boolean	643
* 10.6.15	Collate as boolean	643
* 10.6.16	Copies as Integer	643
* 10.6.17	Duplex as Integer	643
* 10.6.18	FirstPage as Integer	644
* 10.6.19	JobName as String	644
* 10.6.20	LastPage as Integer	644
* 10.6.21	Value(key as String) as Variant	646
– 10.7.1	class CPMRectMBS	647
* 10.7.3	Bottom as Double	647
* 10.7.4	Height as Double	647
* 10.7.5	Left as Double	647
* 10.7.6	Right as Double	647
* 10.7.7	Top as Double	648
* 10.7.8	Width as Double	648
– 10.8.1	class CPMResolutionMBS	649
* 10.8.3	Horizontal as Double	649
* 10.8.4	Vertical as Double	649
– 10.9.1	class CPMVersionMBS	650
* 10.9.3	CountryCode as Integer	650
* 10.9.4	LongVersion as String	650
* 10.9.5	ShortVersion as String	650
* 10.9.6	VersionMajor as Integer	650
* 10.9.7	VersionMinor as Integer	651
* 10.9.8	VersionRevision as Integer	651
* 10.9.9	VersionStage as Integer	651

• 7 CoreText	481
– 7.2.1 class CTFontCollectionMBS	502
* 7.2.3 Available as boolean	502
* 7.2.4 Constructor	502
* 7.2.5 CopyWithFontDescriptors(queryDescriptors() as CTFontDescriptorMBS, options as dictionary) as CTFontCollectionMBS	502
* 7.2.6 CreateCopyWithFontDescriptors(original as CTFontCollectionMBS, queryDescriptors() as CTFontDescriptorMBS, options as dictionary) as CTFontCollectionMBS	503
* 7.2.7 CreateFromAvailableFonts(options as Dictionary) as CTFontCollectionMBS	503
* 7.2.8 CreateWithFontDescriptors(queryDescriptors() as CTFontDescriptorMBS, options as dictionary) as CTFontCollectionMBS	503
* 7.2.9 ExclusionDescriptors as CTFontDescriptorMBS()	504
* 7.2.10 FontAttribute(attributeName as string, options as Integer) as Dictionary()	504
* 7.2.11 FontAttribute(attributeNames() as string, options as Integer) as Dictionary()	504
* 7.2.12 kCTFontCollectionDisallowAutoActivationOption as string	504
* 7.2.13 kCTFontCollectionIncludeDisabledFontsOption as string	505
* 7.2.14 kCTFontCollectionRemoveDuplicatesOption as string	505
* 7.2.15 MatchingFontDescriptors(options as dictionary = nil) as CTFontDescriptorMBS()	505
* 7.2.16 MatchingFontDescriptorsForFamily(familyName as string, options as dictionary = nil) as CTFontDescriptorMBS()	505
* 7.2.17 MatchingFontDescriptorsSorted(tag as Variant) as CTFontDescriptorMBS()	505
* 7.2.18 MutableCopy as CTMutableFontCollectionMBS	506
* 7.2.19 QueryDescriptors as CTFontDescriptorMBS()	506
– 7.3.1 class CTFontDescriptorMBS	507
* 7.3.3 AttributeValue(key as string) as Variant	507
* 7.3.4 AttributeValues as Dictionary	507
* 7.3.5 Available as boolean	507
* 7.3.6 Constructor	508
* 7.3.7 CopyWithAttributes(attributeValues as Dictionary) as CTFontDescriptorMBS	508
* 7.3.8 CopyWithFamily(family as String) as CTFontDescriptorMBS	508
* 7.3.9 CopyWithFeature(featureTypeIdentifier as Integer, featureSelectorIdentifier as Integer) as CTFontDescriptorMBS	508
* 7.3.10 CopyWithSymbolicTraits(symTraitValue as Integer, symTraitMask as Integer) as CTFontDescriptorMBS	509
* 7.3.11 CopyWithVariation(variationIdentifier as Integer, variationValue as Double) as CTFontDescriptorMBS	509
* 7.3.12 CreateCopyWithFamily(original as CTFontDescriptorMBS, family as String) as CTFontDescriptorMBS	509
* 7.3.13 CreateCopyWithSymbolicTraits(original as CTFontDescriptorMBS, symTraitValue as Integer, symTraitMask as Integer) as CTFontDescriptorMBS	510
* 7.3.14 CreateWithAttributes(attributeValues as Dictionary) as CTFontDescriptorMBS	511
* 7.3.15 CreateWithNameAndSize(Name as string, Size as Double = 0.0) as CTFontDescriptorMBS	511

* 7.3.16 kCTFontBaselineAdjustAttribute as string	511
* 7.3.17 kCTFontCascadeListAttribute as string	512
* 7.3.18 kCTFontCharacterSetAttribute as string	512
* 7.3.19 kCTFontDescriptorMatchingCurrentAssetSize as string	512
* 7.3.20 kCTFontDescriptorMatchingDescriptors as string	512
* 7.3.21 kCTFontDescriptorMatchingError as string	512
* 7.3.22 kCTFontDescriptorMatchingPercentage as string	513
* 7.3.23 kCTFontDescriptorMatchingResult as string	513
* 7.3.24 kCTFontDescriptorMatchingSourceDescriptor as string	513
* 7.3.25 kCTFontDescriptorMatchingTotalAssetSize as string	513
* 7.3.26 kCTFontDescriptorMatchingTotalDownloadedSize as string	513
* 7.3.27 kCTFontDisplayNameAttribute as string	514
* 7.3.28 kCTFontDownloadableAttribute as string	514
* 7.3.29 kCTFontEnabledAttribute as string	514
* 7.3.30 kCTFontFamilyNameAttribute as string	514
* 7.3.31 kCTFontFeaturesAttribute as string	514
* 7.3.32 kCTFontFeatureSettingsAttribute as string	515
* 7.3.33 kCTFontFixedAdvanceAttribute as string	515
* 7.3.34 kCTFontFormatAttribute as string	515
* 7.3.35 kCTFontLanguagesAttribute as string	515
* 7.3.36 kCTFontMacintoshEncodingsAttribute as string	516
* 7.3.37 kCTFontMatrixAttribute as string	516
* 7.3.38 kCTFontNameAttribute as string	516
* 7.3.39 kCTFontOrientationAttribute as string	516
* 7.3.40 kCTFontPriorityAttribute as string	517
* 7.3.41 kCTFontRegistrationScopeAttribute as string	517
* 7.3.42 kCTFontSizeAttribute as string	517
* 7.3.43 kCTFontStyleNameAttribute as string	517
* 7.3.44 kCTFontTraitsAttribute as string	518
* 7.3.45 kCTFontURLAttribute as string	518
* 7.3.46 kCTFontVariationAttribute as string	518
* 7.3.47 LocalizedAttributeValue(key as string, byref lang as string) as Variant	518
* 7.3.48 MatchingFontDescriptor(mandatoryAttributes() as String) as CTFontDescriptorMBS	519
* 7.3.49 MatchingFontDescriptors(mandatoryAttributes() as String) as CTFontDescriptorMBS()	519
* 7.3.51 DisplayName as String	519
* 7.3.52 FamilyName as String	520
* 7.3.53 File as FolderItem	520
* 7.3.54 FontSize as Double	520
* 7.3.55 Name as String	521
* 7.3.56 StyleName as String	521

* 7.3.57 URL as String	521
– 7.4.1 class CTFontMBS	524
* 7.4.3 AdvancesForGlyphs(orientation as Integer, glyphs() as Integer) as Double	524
* 7.4.4 AdvancesForGlyphs(orientation as Integer, glyphs() as Integer, boundingRects() as CGSizeMBS) as Double	525
* 7.4.5 AttributeValue(key as string) as Variant	525
* 7.4.6 Available as boolean	525
* 7.4.7 AvailableTables(options as Integer) as String()	526
* 7.4.8 BoundingRectsForGlyphs(orientation as Integer, glyphs() as Integer) as CGRectMBS	526
* 7.4.9 BoundingRectsForGlyphs(orientation as Integer, glyphs() as Integer, boundingRects() as CGRectMBS) as CGRectMBS	526
* 7.4.10 Constructor	527
* 7.4.11 CreateCopyWithAttributes(size as Double, Matrix as CGAffineTransformMBS, fontAttributes as CTFontDescriptorMBS) as CTFontMBS	527
* 7.4.12 CreateForString(text as string, location as Integer, length as Integer) as CTFontMBS	528
* 7.4.13 CreatePathForGlyph(glyph as Integer, transform as CGAffineTransformMBS) as CGPathMBS	529
* 7.4.14 CreateUIFontForLanguage(Type as Integer, size as Double = 0.0, language as string = "") as CTFontMBS	529
* 7.4.15 CreateWithFamily(size as Double, Matrix as CGAffineTransformMBS, family as string) as CTFontMBS	529
* 7.4.16 CreateWithFontDescriptor(descriptor as CTFontDescriptorMBS, size as Double = 0.0, matrix as CGAffineTransformMBS = nil, options as Integer = 0) as CTFontMBS	530
* 7.4.17 CreateWithGraphicsFont(graphicsFont as CGFontMBS, size as Double = 0.0, matrix as CGAffineTransformMBS = nil, attributeValues as CTFontDescriptorMBS = nil) as CTFontMBS	530
* 7.4.18 CreateWithName(name as string, size as Double = 0.0, matrix as CGAffineTransformMBS = nil, options as Integer = 0) as CTFontMBS	531
* 7.4.19 CreateWithPlatformFont(ATSTFontHandle as Integer, size as Double = 0.0, matrix as CGAffineTransformMBS = nil, attributeValues as CTFontDescriptorMBS = nil) as CTFontMBS	531
* 7.4.20 CreateWithSymbolicTraits(size as Double, Matrix as CGAffineTransformMBS, symTraitValue as Integer, symTraitMask as Integer) as CTFontMBS	532
* 7.4.21 DefaultCascadeListForLanguages(languagePrefList() as string) as String()	532
* 7.4.22 Draw(glyphs() as Integer, positions() as CGPointMBS, context as CGContextMBS)	533
* 7.4.23 Features as Dictionary()	533
* 7.4.24 FeatureSettings as Dictionary()	533
* 7.4.25 GlyphsForCharacters(characters() as Integer) as Integer()	534
* 7.4.26 GlyphWithName(name as string) as Integer	534
* 7.4.27 GraphicsFont(byref fontAttributes as CTFontDescriptorMBS) as CGFontMBS	534
* 7.4.28 kCTBaselineClassHanging as string	534

* 7.4.29 kCTBaselineClassIdeographicCentered as string	535
* 7.4.30 kCTBaselineClassIdeographicHigh as string	535
* 7.4.31 kCTBaselineClassIdeographicLow as string	535
* 7.4.32 kCTBaselineClassMath as string	535
* 7.4.33 kCTBaselineClassRoman as string	535
* 7.4.34 kCTBaselineOriginalFont as string	536
* 7.4.35 kCTBaselineReferenceFont as string	536
* 7.4.36 kCTFontCopyrightNameKey as string	536
* 7.4.37 kCTFontDescriptionNameKey as string	536
* 7.4.38 kCTFontDesignerNameKey as string	536
* 7.4.39 kCTFontDesignerURLNameKey as string	536
* 7.4.40 kCTFontFamilyNameKey as string	537
* 7.4.41 kCTFontFeatureSelectorDefaultKey as string	537
* 7.4.42 kCTFontFeatureSelectorIdentifierKey as string	537
* 7.4.43 kCTFontFeatureSelectorNameKey as string	537
* 7.4.44 kCTFontFeatureSelectorSettingKey as string	537
* 7.4.45 kCTFontFeatureTypeExclusiveKey as string	538
* 7.4.46 kCTFontFeatureTypeIdentifierKey as string	538
* 7.4.47 kCTFontFeatureTypeNameKey as string	538
* 7.4.48 kCTFontFeatureTypeSelectorsKey as string	538
* 7.4.49 kCTFontFullNameKey as string	538
* 7.4.50 kCTFontLicenseNameKey as string	539
* 7.4.51 kCTFontLicenseURLNameKey as string	539
* 7.4.52 kCTFontManufacturerNameKey as string	539
* 7.4.53 kCTFontPostScriptCIDNameKey as string	539
* 7.4.54 kCTFontPostScriptNameKey as string	539
* 7.4.55 kCTFontSampleTextNameKey as string	539
* 7.4.56 kCTFontStyleNameKey as string	540
* 7.4.57 kCTFontSubFamilyNameKey as string	540
* 7.4.58 kCTFontTrademarkNameKey as string	540
* 7.4.59 kCTFontUniqueNameKey as string	540
* 7.4.60 kCTFontVariationAxisDefaultValueKey as string	540
* 7.4.61 kCTFontVariationAxisIdentifierKey as string	541
* 7.4.62 kCTFontVariationAxisMaximumValueKey as string	541
* 7.4.63 kCTFontVariationAxisMinimumValueKey as string	541
* 7.4.64 kCTFontVariationAxisNameKey as string	541
* 7.4.65 kCTFontVendorURLNameKey as string	541
* 7.4.66 kCTFontVersionNameKey as string	541
* 7.4.67 LigatureCaretPositions(glyph as Integer) as Double()	542
* 7.4.68 Name(nameKey as string) as String	542
* 7.4.69 Name(nameKey as string, byref language as string) as String	542

* 7.4.70	OpticalBoundsForGlyphs(glyphs() as Integer, boundingRects() as CGRectMBS, options as Integer = 0) as CGRectMBS	543
* 7.4.71	OpticalBoundsForGlyphs(glyphs() as Integer, options as Integer = 0) as CGRectMBS	544
* 7.4.72	PlatformFont(byref fontAttributes as CTFontDescriptorMBS) as Integer	544
* 7.4.73	SupportedLanguages as String()	544
* 7.4.74	Table(table as string, options as Integer) as Memoryblock	545
* 7.4.75	VariationAxes as Dictionary()	545
* 7.4.76	VerticalTranslationsForGlyphs(glyphs() as Integer) as CGSizeMBS()	545
* 7.4.78	Ascent as Double	545
* 7.4.79	BoundingBox as CGRectMBS	546
* 7.4.80	CapHeight as Double	546
* 7.4.81	CharacterSet as Variant	546
* 7.4.82	Descent as Double	546
* 7.4.83	DisplayName as String	546
* 7.4.84	FamilyName as String	547
* 7.4.85	File as FolderItem	547
* 7.4.86	FontDescriptor as CTFontDescriptorMBS	547
* 7.4.87	FullName as String	548
* 7.4.88	GlyphCount as Integer	548
* 7.4.89	Leading as Double	548
* 7.4.90	Matrix as CGAffineTransformMBS	548
* 7.4.91	PostScriptName as String	548
* 7.4.92	Size as Double	549
* 7.4.93	SlantAngle as Double	549
* 7.4.94	StringEncoding as UInt32	549
* 7.4.95	SymbolicTraits as UInt32	549
* 7.4.96	Traits as Dictionary	550
* 7.4.97	UnderlinePosition as Double	550
* 7.4.98	UnderlineThickness as Double	550
* 7.4.99	UnitsPerEm as UInt64	550
* 7.4.100	URL as String	550
* 7.4.101	Variation as Dictionary	551
* 7.4.102	XHeight as Double	551
– 7.5.1	class CTFrameMBS	554
* 7.5.3	Available as boolean	555
* 7.5.4	Constructor	555
* 7.5.5	Draw(context as CGContextMBS)	555
* 7.5.6	kCTFrameClippingPathsAttributeName as string	555
* 7.5.7	kCTFramePathClippingPathAttributeName as string	556
* 7.5.8	kCTFramePathFillRuleAttributeName as string	556
* 7.5.9	kCTFramePathWidthAttributeName as string	556

	51
* 7.5.10 kCTFrameProgressionAttributeName as string	556
* 7.5.11 LineOrigins(location as Integer, length as Integer) as CGPointMBS()	557
* 7.5.12 Lines as CTLineMBS()	557
* 7.5.14 FrameAttributes as Dictionary	557
* 7.5.15 Path as CGPathMBS	558
* 7.5.16 StringRangeLength as Integer	558
* 7.5.17 StringRangeLocation as Integer	558
* 7.5.18 VisibleStringRangeLength as Integer	558
* 7.5.19 VisibleStringRangeLocation as Integer	558
– 7.6.1 class CTFramesetterMBS	560
* 7.6.3 Available as boolean	560
* 7.6.4 Constructor	560
* 7.6.5 CreateFrame(location as Integer, length as Integer, path as CGPathMBS, frameAttributes as dictionary = nil) as CTFrameMBS	560
* 7.6.6 CreateWithAttributedString(s as CFAttributedStringMBS) as CTFramesetterMBS	561
* 7.6.7 SuggestFrameSizeWithConstraints(location as Integer, length as Integer, frameAttributes as dictionary, constraints as CGSizeMBS, byref fitRangeLocation as Integer, byref fitRangeLength as Integer) as CGSizeMBS	561
* 7.6.9 TypeSetter as CTTypesetterMBS	562
– 7.7.1 class CTGlyphInfoMBS	563
* 7.7.3 Available as boolean	563
* 7.7.4 Constructor	563
* 7.7.5 CreateWithCharacterIdentifier(cid as Integer, collection as Integer, baseString as string) as CTGlyphInfoMBS	563
* 7.7.6 CreateWithGlyph(glyph as Integer, font as CTFontMBS, baseString as string) as CTGlyphInfoMBS	564
* 7.7.7 CreateWithGlyphName(glyphName as string, font as CTFontMBS, baseString as string) as CTGlyphInfoMBS	564
* 7.7.9 CharacterCollection as Integer	564
* 7.7.10 CharacterIdentifier as Integer	565
* 7.7.11 GlyphName as String	565
– 7.8.1 class CTLineMBS	566
* 7.8.3 Available as boolean	567
* 7.8.4 Bounds(options as Integer = 0) as CGRectMBS	567
* 7.8.5 Constructor	568
* 7.8.6 CreateJustifiedLine(justificationFactor as Double, justificationWidth as Double) as CTLineMBS	568
* 7.8.7 CreateTruncatedLine(width as Double, truncationType as Integer, truncationToken as CTLineMBS = nil) as CTLineMBS	568
* 7.8.8 CreateWithAttributedString(s as CFAttributedStringMBS) as CTLineMBS	569
* 7.8.9 Draw(context as CGContextMBS)	569
* 7.8.10 GlyphRuns as CTRunMBS()	569

* 7.8.11	ImageBounds(context as CGContextMBS) as CGRectMBS	570
* 7.8.12	OffsetForStringIndex(charIndex as Integer, byref secondaryOffset as Double) as Double	570
* 7.8.13	PenOffsetForFlush(flushFactor as Double, flushWidth as Double) as Double	570
* 7.8.14	StringIndexForPosition(position as CGPointMBS) as Integer	571
* 7.8.15	TypographicBounds(byref ascent as Double, byref descent as Double, byref leading as Double) as Double	571
* 7.8.17	GlyphCount as Integer	571
* 7.8.18	StringRangeLength as Integer	572
* 7.8.19	StringRangeLocation as Integer	572
* 7.8.20	TrailingWhitespaceWidth as Double	572
– 7.9.1	class CTMutableFontCollectionMBS	574
* 7.9.3	Constructor	574
* 7.9.4	SetExclusionDescriptors(descriptors() as CTFontDescriptorMBS)	574
* 7.9.5	SetQueryDescriptors(descriptors() as CTFontDescriptorMBS)	574
– 7.10.1	class CTParagraphStyleMBS	575
* 7.10.3	Available as boolean	575
* 7.10.4	Constructor	575
* 7.10.5	Create as CTParagraphStyleMBS	576
* 7.10.6	Create(settings() as CTParagraphStyleSettingMBS) as CTParagraphStyleMBS	576
* 7.10.7	CreateCopy as CTParagraphStyleMBS	576
* 7.10.8	CreateWithAlignment(Alignment as Integer) as CTParagraphStyleMBS	577
* 7.10.9	TabStops as CTTextTabMBS()	577
* 7.10.11	Alignment as Integer	577
* 7.10.12	BaseWritingDirection as Integer	578
* 7.10.13	DefaultTabInterval as Double	578
* 7.10.14	FirstLineHeadIndent as Double	578
* 7.10.15	HeadIndent as Double	578
* 7.10.16	LineBoundsOptions as Integer	578
* 7.10.17	LineBreakMode as Integer	579
* 7.10.18	LineHeightMultiple as Double	579
* 7.10.19	LineSpacingAdjustment as Double	579
* 7.10.20	MaximumLineHeight as Double	579
* 7.10.21	MaximumLineSpacing as Double	580
* 7.10.22	MinimumLineHeight as Double	580
* 7.10.23	MinimumLineSpacing as Double	580
* 7.10.24	ParagraphSpacing as Double	580
* 7.10.25	ParagraphSpacingBefore as Double	580
* 7.10.26	TailIndent as Double	581
– 7.11.1	class CTParagraphStyleSettingMBS	585
* 7.11.3	SetTextTabs(textTabs() as CTTextTabMBS)	585

	53
* 7.11.5 doubleValue as Double	585
* 7.11.6 intValue as Integer	585
* 7.11.7 Spec as Integer	585
– 7.12.1 class CTRunDelegateMBS	587
* 7.12.3 Available as boolean	587
* 7.12.4 Close	587
* 7.12.5 Constructor	587
* 7.12.7 Close	588
* 7.12.8 GetAscent as Double	588
* 7.12.9 GetDescent as Double	588
* 7.12.10 GetWidth as Double	588
– 7.13.1 class CTRunMBS	589
* 7.13.3 Advances as CGSizeMBS()	589
* 7.13.4 Available as boolean	589
* 7.13.5 Constructor	589
* 7.13.6 Draw(context as CGContextMBS, location as Integer, length as Integer = 0)	590
* 7.13.7 Glyphs as Integer()	590
* 7.13.8 ImageBounds(context as CGContextMBS, location as Integer, length as Integer) as CGRectMBS	590
* 7.13.9 Positions as CGPointMBS()	590
* 7.13.10 StringIndices as Integer()	591
* 7.13.11 TypographicBounds(location as Integer, length as Integer, byref ascent as Double, byref descent as Double, byref leading as Double) as Double	591
* 7.13.13 AttributeValues as Dictionary	591
* 7.13.14 GlyphCount as Integer	592
* 7.13.15 Status as Integer	592
* 7.13.16 StringRangeLength as Integer	592
* 7.13.17 StringRangeLocation as Integer	592
* 7.13.18 TextMatrix as CGAffineTransformMBS	593
– 7.14.1 class CTTextTabMBS	594
* 7.14.3 Available as boolean	594
* 7.14.4 Constructor	594
* 7.14.5 Create(alignment as Integer, location as Double, options as Dictionary = nil) as CTTextTabMBS	594
* 7.14.6 kCTTabColumnTerminatorsAttributeName as string	595
* 7.14.8 Alignment as Integer	595
* 7.14.9 Location as Double	595
* 7.14.10 Options as Dictionary	595
– 7.15.1 class CTTypesetterMBS	597
* 7.15.3 Available as boolean	597
* 7.15.4 Constructor	597

- * 7.15.5 CreateLine(location as Integer, length as Integer, offset as Double = 0.0) as CT-LineMBS 597
- * 7.15.6 CreateWithAttributedString(s as CFAttributedStringMBS) as CTypesetterMBS 598
- * 7.15.7 CreateWithAttributedString(s as CFAttributedStringMBS, options as dictionary) as CTypesetterMBS 598
- * 7.15.8 kCTTypesetterOptionDisableBidiProcessing as string 599
- * 7.15.9 kCTTypesetterOptionForcedEmbeddingLevel as string 599
- * 7.15.10 SuggestClusterBreak(startIndex as Integer, width as Double) as Integer 599
- * 7.15.11 SuggestClusterBreak(startIndex as Integer, width as Double, offset as Double) as Integer 600
- * 7.15.12 SuggestLineBreak(startIndex as Integer, width as Double) as Integer 600
- * 7.15.13 SuggestLineBreak(startIndex as Integer, width as Double, offset as Double) as Integer 601

	55
• 5 CoreAnimation	69
– 4.2.1 class DesktopControl	68
* 4.2.3 CALayerMBS as CALayerMBS	68

- **6 CoreGraphics** 97
 - 12.1.1 class DesktopWindow 657
 - * 12.1.3 CGContextMBS as CGContextMBS 657
 - * 12.1.5 CGColorSpaceMBS as CGColorSpaceMBS 658

	57
• 6 CoreGraphics	97
– ?? Globals	??
* 6.1.4 CGBitmapContextCreateMBS(data as memoryblock, width as Integer, height as Integer, bitsPerComponent as Integer, bytesPerRow as Integer, colorspace as CGColorSpaceMBS, alphaInfo as Integer) as CGBitmapContextMBS	99
* 6.1.8 CGCreateImageFromJPEGDataProviderMBS(dataprovider as Variant, decode as memoryblock, shouldInterpolate as boolean, intent as Integer) as CGImageMBS	103
* 6.1.9 CGCreateImageFromPNGDataProviderMBS(dataprovider as Variant, decode as memoryblock, shouldInterpolate as boolean, intent as Integer) as CGImageMBS	104
* 6.1.10 CGCreateImageMBS(pic as picture) as CGImageMBS	105
* 6.1.11 CGCreateImageMBS(pic as picture, mask as picture) as CGImageMBS	105
* 6.1.12 CGMakePointMBS(x as Double, y as Double) as CGPointMBS	106
* 6.1.13 CGMakeRectMBS(left as Double, top as Double, width as Double, height as Double) as CGRectMBS	106
* 6.1.14 CGMakeSizeMBS(width as Double, height as Double) as CGSizeMBS	106
* 6.1.15 CGNewPDFDocumentMBS(consumer as CGDataConsumerMBS, MediaBox as CGRectMBS, title as string, author as string, creator as string) as CGPDFContextMBS	106
* 6.1.16 CGNewPDFDocumentMBS(file as folderitem, MediaBox as CGRectMBS, title as string, author as string, creator as string) as CGPDFContextMBS	107
* 6.1.1 CGNewPDFDocumentMBS(file as folderitem, MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean) as CGPDFContextMBS	97
* 6.1.2 CGNewPDFDocumentMBS(file as folderitem, MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean, KeyLength as Integer) as CGPDFContextMBS	98
* 6.1.5 CGOpenPDFDocumentMBS(dataprovider as CGDataProviderMBS) as CGPDFDocumentMBS	100
* 6.1.17 CGOpenPDFDocumentMBS(file as folderitem) as CGPDFDocumentMBS	107
* 6.1.6 CGShadingCreateAxialMBS(colorspace as CGColorSpaceMBS, startPoint as CGPointMBS, endPoint as CGPointMBS, func as CGFunctionMBS, extendStart as boolean, extendEnd as boolean) as CGShadingMBS	101
* 6.1.7 CGShadingCreateRadialMBS(colorspace as CGColorSpaceMBS, startPoint as CGPointMBS, startRadius as Double, endPoint as CGPointMBS, endRadius as Double, func as CGFunctionMBS, extendStart as boolean, extendEnd as boolean) as CGShadingMBS	102
* 6.1.3 GetCurrentCGContextMBS as CGContextMBS	98

- **6 CoreGraphics** 97
 - 8.1.1 class FolderItem 603
 - * 8.1.3 NewCGPDFDocumentMBS(MediaBox as CGRectMBS, title as string, author as string, creator as string) as CGPDFContextMBS 603
 - * 8.1.4 NewCGPDFDocumentMBS(MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean) as CGPDFContextMBS 604
 - * 8.1.5 NewCGPDFDocumentMBS(MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean, KeyLength as Integer) as CGPDFContextMBS 605
 - * 8.1.6 NewCGPDFDocumentWithInfoMBS(MediaBox as CGRectMBS, info as object) as CGPDFContextMBS 606
 - * 8.1.7 OpenAsCGPDFDocumentMBS as CGPDFDocumentMBS 609

	59
• 6 CoreGraphics	97
– 9.1.1 class Graphics	611
* 9.1.3 DrawCGPDFDocumentMBS(pdf as CGPDFDocumentMBS, r as CGRectMBS, page as Integer)	611
* 9.1.4 DrawCGPDFDocumentMBS(pdf as CGPDFDocumentMBS, r as CGRectMBS, page as Integer, InterpolationQuality as Integer, Antialias as boolean, FontSmoothing as Boolean)	612
– 9.2.1 class Picture	614
* 9.2.3 CGColorSpaceMBS as CGColorSpaceMBS	614

• 6 CoreGraphics	97
– 6.52.1 class QDPictMBS	477
* 6.52.3 Constructor(dataProvider as CGDataProviderMBS)	477
* 6.52.4 Constructor(file as folderitem)	477
* 6.52.5 Constructor(url as string)	478
* 6.52.6 DrawToCGContext(context as CGContextMBS, r as CGRectMBS)	478
* 6.52.7 GetBounds as CGRectMBS	478
* 6.52.8 GetResolution(byref xRes as single, byref yRes as single)	478
* 6.52.9 Height as Double	479
* 6.52.10 HorizontalResolution as Double	479
* 6.52.11 VerticalResolution as Double	479
* 6.52.12 Width as Double	479
* 6.52.14 Handle as Integer	479
* 6.52.15 LastError as Integer	480
– 12.2.1 class Window	659
* 12.2.3 CGContextMBS as CGContextMBS	659
* 12.2.5 CGColorSpaceMBS as CGColorSpaceMBS	660

Chapter 2

List of all classes

• Application	653
• CALayerMBS	69
• CATextLayerMBS	88
• CATransactionMBS	92
• CGAffineTransformMBS	107
• CGBitmapContextMBS	114
• CGColorMBS	125
• CGColorSpaceMBS	131
• CGContextMBS	142
• CGDataConsumerMBS	181
• CGDataProviderMBS	184
• CGDisplayConfigMBS	186
• CGDisplayMBS	191
• CGDisplayModeMBS	225
• CGDisplayReconfigurationEventMBS	229
• CGDisplayStreamEventMBS	231
• CGDisplayStreamUpdateMBS	236
• CGDisplayTransferFormulaMBS	238
• CGFontMBS	240

• CGFunctionMBS	246
• CGGradientMBS	248
• CGImageDestinationMBS	255
• CGImageMBS	269
• CGImageMetadataMBS	282
• CGImageMetadataTagMBS	291
• CGImageSourceMBS	295
• CGLayerMBS	369
• CGMutableImageMetadataMBS	372
• CGMutablePathMBS	377
• CGPathElementMBS	381
• CGPathMBS	383
• CGPDFArrayMBS	386
• CGPDFContextMBS	389
• CGPDFDictionaryListMBS	397
• CGPDFDictionaryMBS	399
• CGPDFDocumentMBS	402
• CGPDFObjectMBS	411
• CGPDFPageMBS	415
• CGPDFStreamMBS	418
• CGPDFStringMBS	420
• CGPointMBS	422
• CGRectMBS	429
• CGSConnectionMBS	440
• CGScreenRefreshEventMBS	446
• CGScreenUpdateMoveEventMBS	447
• CGShadingMBS	448
• CGSizeMBS	449
• CGSValueMBS	462

	63
• CGSWindowListMBS	464
• CGSWindowMBS	467
• CGSWorkspaceMBS	474
• Control	67
• CoreTextMBS	481
• CPMLanguageInfoMBS	617
• CPMPageFormatMBS	619
• CPMPrinterMBS	624
• CPMPrintSessionMBS	632
• CPMPrintSettingsMBS	641
• CPMRectMBS	647
• CPMResolutionMBS	649
• CPMVersionMBS	650
• CTFontCollectionMBS	502
• CTFontDescriptorMBS	507
• CTFontMBS	524
• CTFrameMBS	554
• CTFramesetterMBS	560
• CTGlyphInfoMBS	563
• CTLineMBS	566
• CTMutableFontCollectionMBS	574
• CTParagraphStyleMBS	575
• CTParagraphStyleSettingMBS	585
• CTRunDelegateMBS	587
• CTRunMBS	589
• CTTextTabMBS	594
• CTTypesetterMBS	597
• DesktopControl	68
• DesktopWindow	657

• FolderItem	603
• Graphics	611
• Picture	614
• Window	659

Chapter 3

List of all global methods

- 6.1.4 `CGBitmapContextCreateMBS`(data as memoryblock, width as Integer, height as Integer, bitsPerComponent as Integer, bytesPerRow as Integer, colorspace as `CGColorSpaceMBS`, alphaInfo as Integer) as `CGBitmapContextMBS` 99
- 6.1.8 `CGCreateImageFromJPEGDataProviderMBS`(dataprovder as Variant, decode as memoryblock, shouldInterpolate as boolean, intent as Integer) as `CGImageMBS` 103
- 6.1.9 `CGCreateImageFromPNGDataProviderMBS`(dataprovder as Variant, decode as memoryblock, shouldInterpolate as boolean, intent as Integer) as `CGImageMBS` 104
- 6.1.10 `CGCreateImageMBS`(pic as picture) as `CGImageMBS` 105
- 6.1.11 `CGCreateImageMBS`(pic as picture, mask as picture) as `CGImageMBS` 105
- 6.1.12 `CGMakePointMBS`(x as Double, y as Double) as `CGPointMBS` 106
- 6.1.13 `CGMakeRectMBS`(left as Double, top as Double, width as Double, height as Double) as `CGRectMBS` 106
- 6.1.14 `CGMakeSizeMBS`(width as Double, height as Double) as `CGSizeMBS` 106
- 6.1.15 `CGNewPDFDocumentMBS`(consumer as `CGDataConsumerMBS`, MediaBox as `CGRectMBS`, title as string, author as string, creator as string) as `CGPDFContextMBS` 106
- 6.1.16 `CGNewPDFDocumentMBS`(file as folderitem, MediaBox as `CGRectMBS`, title as string, author as string, creator as string) as `CGPDFContextMBS` 107
- 6.1.1 `CGNewPDFDocumentMBS`(file as folderitem, MediaBox as `CGRectMBS`, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean) as `CGPDFContextMBS` 97
- 6.1.2 `CGNewPDFDocumentMBS`(file as folderitem, MediaBox as `CGRectMBS`, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean, KeyLength as Integer) as `CGPDFContextMBS` 98

- 6.1.5 CGOpenPDFDocumentMBS(dataprovider as CGDataProviderMBS) as CGPDFDocumentMBS
100
- 6.1.17 CGOpenPDFDocumentMBS(file as folderitem) as CGPDFDocumentMBS 107
- 6.1.6 CGShadingCreateAxialMBS(colorspace as CGColorSpaceMBS, startPoint as CGPointMBS, endPoint as CGPointMBS, func as CGFunctionMBS, extendStart as boolean, extendEnd as boolean) as CGShadingMBS 101
- 6.1.7 CGShadingCreateRadialMBS(colorspace as CGColorSpaceMBS, startPoint as CGPointMBS, startRadius as Double, endPoint as CGPointMBS, endRadius as Double, func as CGFunctionMBS, extendStart as boolean, extendEnd as boolean) as CGShadingMBS 102
- 6.1.3 GetCurrentCGContextMBS as CGContextMBS 98
- 10.1.1 NewCPMPPageFormatMBS as CPMPPageFormatMBS 615
- 10.1.2 NewCPMPrintSessionMBS as CPMPrintSessionMBS 615
- 10.1.3 NewCPMPrintSettingsMBS as CPMPrintSettingsMBS 617

Chapter 4

Controls

4.1 class Control

4.1.1 class Control

Plugin Version: 9.7, Platform: macOS, Targets: Desktop only.

Function: The built in Control class in Xojo.

4.1.2 Methods

4.1.3 CALayerMBS as CALayerMBS

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: Returns the Core Animation layer that the receiver uses as its backing store.

Notes: Works only in Cocoa target.

Also sets `wantsLayer` to true for the view to make sure it has a layer.

4.2 class DesktopControl

4.2.1 class DesktopControl

Plugin Version: 21.5, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: The built in Control class in Xojo.

4.2.2 Methods

4.2.3 CALayerMBS as CALayerMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Returns the Core Animation layer that the receiver uses as its backing store.

Notes: Works only in Cocoa target.

Also sets `wantsLayer` to true for the view to make sure it has a layer.

Chapter 5

CoreAnimation

5.1 class CALayerMBS

5.1.1 class CALayerMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The plugin class for a CoreAnimation.

Example:

```
// add gray layer over Xojo window. Or hide it on second call, show on third call, etc...
```

```
If view = Nil Then
```

```
// view is property in window/module
```

```
Var win As NSWindowMBS = Self.NSWindowMBS
```

```
view = New NSViewMBS(0, 0, Width, Height)
```

```
win.contentView.AddSubview(view)
```

```
view.wantsLayer = True
```

```
view.autoresizesSubviews = True
```

```
view.autoresizingMask = view.NSViewWidthSizable + view.NSViewHeightSizable
```

```
Var layer As CALayerMBS = view.layer
```

```
layer.backgroundColor = CGColorMBS.CreateDeviceGray(0.3, 0.5)
```

```
Else
```

```
// show or hide
```

```
view.isHidden = Not view.isHidden
```

End If

Notes: The `CALayer` class manages image-based content and allows you to perform animations on that content. Layers are often used to provide the backing store for views but can also be used without a view to display content. A layer's main job is to manage the visual content that you provide but the layer itself has visual attributes that can be set, such as a background color, border, and shadow. In addition to managing visual content, the layer also maintains information about the geometry of its content (such as its position, size, and transform) that is used to present that content onscreen. Modifying the properties of the layer is how you initiate animations on the layer's content or geometry. A layer object encapsulates the duration and pacing of a layer and its animations by adopting the `CAMediaTiming` protocol, which defines the layer's timing information.

If the layer object was created by a view, the view typically assigns itself as the layer's delegate automatically, and you should not change that relationship. For layers you create yourself, you can assign a delegate object and use that object to provide the contents of the layer dynamically and perform other tasks. A layer may also have a layout manager object (assigned to the `layoutManager` property) to manage the layout of subviews separately.

Available in OS X v10.5 and later.

Blog Entries

- [MBS Xojo Plugins, version 22.2pr2](#)
- [MBS Xojo Plugins, version 21.4pr2](#)
- [MBS Xojo / Real Studio Plugins, version 16.1pr3](#)
- [MBS Xojo / Real Studio Plugins, version 15.3pr3](#)
- [MBS Xojo / Real Studio Plugins, version 15.0pr12](#)
- [MBS Xojo / Real Studio Plugins, version 14.5pr2](#)
- [MBS Real Studio Plugins, version 13.1pr13](#)
- [MBS Real Studio Plugins, version 13.1pr9](#)

5.1.2 Methods

5.1.3 `addSublayer(layer as CALayerMBS)`

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Appends the layer to the layer's list of sublayers.

Notes: If the array in the `sublayers` property is nil, calling this method creates an array for that property and adds the specified layer to it.

5.1.4 available as boolean

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Returns true if CALayer is available.

Notes: True on Mac and False on Windows/Linux.

5.1.5 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Initializes the layer.

Notes: Available in OS X v10.5 and later.

5.1.6 display

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Reloads the content of this layer.

Notes: Do not call this method directly. The layer calls this method at appropriate times to update the layer's content. If the layer has a delegate object, this method attempts to call the delegate's Configuring the Layer's Rendering Behavior method, which the delegate can use to update the layer's contents. If the delegate does not implement the Configuring the Layer's Rendering Behavior method, this method creates a backing store and calls the layer's drawInContext: method to fill that backing store with content. The new backing store replaces the previous contents of the layer.

Subclasses can override this method and use it to set the layer's contents property directly. You might do this if your custom layer subclass handles layer updates differently.

5.1.7 displayIfNeeded

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Initiates the update process for a layer if it is currently marked as needing an update.

Notes: You can call this method as needed to force an update to your layer's contents outside of the normal update cycle. Doing so is generally not needed, though. The preferred way to update a layer is to call setNeedsDisplay and let the system update the layer during the next cycle.

Available in OS X v10.6 and later.

5.1.8 layer as CALayerMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Creates and returns an instance of the layer object.

Notes: Returns the initialized layer object or nil if initialization was not successful.

If you subclass CALayer, you may override this method and use it to provide an instance of your specific subclass.

5.1.9 layerFromHandle(Handle as Integer) as CALayerMBS

Plugin Version: 21.0, Platforms: macOS, iOS, Targets: All.

Function: Returns a new MBS layer object for the given handle.

Notes: Handle must be pointing to a CALayer, UIView or NSView.

Returns nil in case of an error.

See also:

- 5.1.10 layerFromHandle(Handle as Ptr) as CALayerMBS 72

5.1.10 layerFromHandle(Handle as Ptr) as CALayerMBS

Plugin Version: 21.0, Platforms: macOS, iOS, Targets: All.

Function: Returns a new MBS layer object for the given handle.

Notes: Handle must be pointing to a CALayer, UIView or NSView.

Returns nil in case of an error.

See also:

- 5.1.9 layerFromHandle(Handle as Integer) as CALayerMBS 72

5.1.11 layoutIfNeeded

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Recalculate the receiver's layout, if required.

Notes: When this message is received, the layer's super layers are traversed until a ancestor layer is found that does not require layout. Then layout is performed on the entire layer-tree beneath that ancestor.

5.1.12 layoutSublayers

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Tells the layer to update its layout.

Notes: Subclasses can override this method and use it to implement their own layout algorithm. Your implementation must set the frame of each sublayer managed by the receiver.

The default implementation of this method calls the `layoutSublayersOfLayer` method of the layer's delegate object. If there is no delegate object, or the delegate does not implement that method, this method calls the `layoutSublayersOfLayer` method of the object in the `layoutManager` property.

5.1.13 `removeAllAnimations`

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Remove all animations attached to the layer.

5.1.14 `removeFromSuperlayer`

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Detaches the layer from its parent layer.

Notes: You can use this method to remove a layer (and all of its sublayers) from a layer hierarchy. This method updates both the superlayer's list of sublayers and sets this layer's superlayer property to nil. Available in OS X v10.5 and later.

5.1.15 `renderInContext(CGContextHandle as Integer) as boolean`

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: Renders content of CALayer into given CGContext.

Notes: Renders the receiver and its sublayers into the specified context.

This method renders directly from the layer tree, ignoring any animations added to the render tree. Renders in the coordinate space of the layer.

Returns true if plugin called render command, so layer can draw itself.

Not all layers support drawing into context.

5.1.16 `renderInPicture(Pic as Picture) as boolean`

Plugin Version: 16.1, Platform: macOS, Targets: Desktop only.

Function: Renders content of CALayer into given Picture.

Notes: Renders the receiver and its sublayers into the specified context.

This method renders directly from the layer tree, ignoring any animations added to the render tree. Renders in the coordinate space of the layer.

Returns true if plugin called render command, so layer can draw itself.

Not all layers support drawing into context.

5.1.17 setNeedsDisplay

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Marks the layer's contents as needing to be updated.

Notes: Calling this method causes the layer to recache its content. This results in the layer potentially calling either the `displayLayer` or `drawLayer:inContext` method of its delegate. The existing content in the layer's contents property is removed to make way for the new content.

Available in OS X v10.5 and later.

5.1.18 setNeedsDisplayInRect(r as CGRectMBS)

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Marks the region within the specified rectangle as needing to be updated.

Notes: `r`: The rectangular region of the layer to mark as invalid. You must specify this rectangle in the layer's own coordinate system.

Available in OS X v10.5 and later.

5.1.19 setNeedsLayout

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Invalidates the layer's layout and marks it as needing an update.

Notes: You can call this method to indicate that the layout of a layer's sublayers has changed and must be updated. The system typically calls this method automatically when the layer's bounds change or when sublayers are added or removed. In OS X, if your layer's `layoutManager` property contains an object that implements the `invalidateLayoutOfLayer` method, that method is called too.

During the next update cycle, the system calls the `layoutSublayers` method of any layers requiring layout updates.

Available in OS X v10.5 and later.

5.1.20 sublayers as CALayerMBS()

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The array with sublayers.

5.1.21 Properties

5.1.22 affineTransform as CGAffineTransformMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The affine version of the layer's transform..

Notes: The affine transform structure that corresponds to the value in the layer's transform property.
(Read and Write property)

5.1.23 anchorPoint as CGRectMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Defines the anchor point of the layer's bounds rectangle. Animatable.

Notes: You specify the value for this property using the unit coordinate space. The default value of this property is (0.5, 0.5), which represents the center of the layer's bounds rectangle. All geometric manipulations to the view occur about the specified point. For example, applying a rotation transform to a layer with the default anchor point causes the layer to rotate around its center. Changing the anchor point to a different location would cause the layer to rotate around that new point.

For more information about the relationship between the frame, bounds, anchorPoint and position properties, see Core Animation Programming Guide.

Available in OS X v10.5 and later.

(Read and Write property)

5.1.24 anchorPointZ as Double

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The anchor point for the layer's position along the z axis. Animatable.

Notes: This property specifies the anchor point on the z axis around which geometric manipulations occur. The point is expressed as a distance (measured in points) along the z axis. The default value of this property is 0.

Available in OS X v10.6 and later.

(Read and Write property)

5.1.25 AutoresizingMask as Integer

Plugin Version: 13.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: A bitmask defining how the layer is resized when the bounds of its superlayer changes.

Notes: If your app does not use a layout manager or constraints to handle layout changes, you can assign a value to this property to adjust the layer's size in response to changes in the superlayer's bounds. For a list of possible values, see "Autoresizing Mask".

The default value of this property is `kCALayerNotSizable`.

(Read and Write property)

5.1.26 backgroundColor as Variant

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The background color of the receiver. Animatable.

Notes: Value is a `CGColorMBS`. We use variant to reduce dependency to other plugins.

The default value of this property is `nil`.

The value of this property is retained using the Core Foundation retain/release semantics. This behavior occurs despite the fact that the property declaration appears to use the default assign semantics for object retention.

(Read and Write property)

5.1.27 borderColor as Variant

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The color of the layer's border. Animatable.

Notes: Value is a `CGColorMBS`. We use variant to reduce dependency to other plugins.

The default value of this property is an opaque black color.

The value of this property is retained using the Core Foundation retain/release semantics. This behavior occurs despite the fact that the property declaration appears to use the default assign semantics for object retention.

(Read and Write property)

5.1.28 borderWidth as Double

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The width of the layer's border. Animatable.

Notes: When this value is greater than 0.0, the layer draws a border using the current `borderColor` value. The border is drawn inset from the receiver's bounds by the value specified in this property. It is composited above the receiver's contents and sublayers and includes the effects of the `cornerRadius` property.

The default value of this property is 0.0.

(Read and Write property)

5.1.29 bounds as CGRectMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The layer's bounds rectangle. Animatable.

Notes: The bounds rectangle is the origin and size of the layer in its own coordinate space. When you create a new standalone layer, the default value for this property is an empty rectangle, which you must change before using the layer. The values of each coordinate in the rectangle are measured in points.

For more information about the relationship between the `frame`, `bounds`, `anchorPoint` and `position` properties, see *Core Animation Programming Guide*.

(Read and Write property)

5.1.30 className as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The class name of the layer.

Notes: Useful for debugging.

(Read only property)

5.1.31 classPath as string

Plugin Version: 14.0, Platform: macOS, Targets: All.

Function: The path of this layer class.

Notes: Useful for debugging to know what super classes the layer has.
(Read only property)

5.1.32 contents as Variant

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The contents of the layer.

Notes: Value is a `CGImageMBS`. We use variant to reduce dependency to other plugins.
(Read and Write property)

5.1.33 contentsCenter as CGRectMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The rectangle that defines how the layer contents are scaled during a resizing operation. Animatable.

Notes: You can use this property to subdivide the layer's content into a 3x3 grid. The value in this property specifies the location and size of the center rectangle in that grid. If the layer's `contentsGravity` property is set to one of the resizing modes, resizing the layer causes scaling to occur differently in each rectangle of the grid. The center rectangle is stretched in both dimensions, the top-center and bottom-center rectangles are stretched only horizontally, the left-center and right-center rectangles are stretched only vertically, and the four corner rectangles are not stretched at all. Therefore, you can use this technique to implement stretchable backgrounds or images using a three-part or nine-part image.

The value in this property is set to the unit rectangle (0.0,0.0) (1.0,1.0) by default, which causes the entire image to scale in both dimensions. If you specify a rectangle that extends outside the unit rectangle, the result is undefined. The rectangle you specify is applied only after the `contentsRect` property has been applied to the image.

Note: If the width or height of the rectangle in this property is very small or 0, the value is implicitly changed to the width or height of a single source pixel centered at the specified location.
(Read and Write property)

5.1.34 contentsRect as CGRectMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The rectangle, in the unit coordinate space, that defines the portion of the layer's contents that should be used. Animatable.

Notes: Defaults to the unit rectangle (0.0, 0.0, 1.0, 1.0).

If pixels outside the unit rectangle are requested, the edge pixels of the contents image will be extended

outwards.

If an empty rectangle is provided, the results are undefined.
(Read and Write property)

5.1.35 contentsScale as Double

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The scale factor applied to the layer.

Notes: This value defines the mapping between the logical coordinate space of the layer (measured in points) and the physical coordinate space (measured in pixels). Higher scale factors indicate that each point in the layer is represented by more than one pixel at render time. For example, if the scale factor is 2.0 and the layer's bounds are 50 x 50 points, the size of the bitmap used to present the layer's content is 100 x 100 pixels.

The default value of this property is 1.0. For layers attached to a view, the view changes the scale factor automatically to a value that is appropriate for the current screen. For layers you create and manage yourself, you must set the value of this property yourself based on the resolution of the screen and the content you are providing. Core Animation uses the value you specify as a cue to determine how to render your content.

Available in OS X v10.7 and later.
(Read and Write property)

5.1.36 cornerRadius as Double

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The radius to use when drawing rounded corners for the layer's background. Animatable.

Notes: Setting the radius to a value greater than 0.0 causes the layer to begin drawing rounded corners on its background. By default, the corner radius does not apply to the image in the layer's contents property; it applies only to the background color and border of the layer. However, setting the masksToBounds property to true causes the content to be clipped to the rounded corners.

The default value of this property is 0.0.
(Read and Write property)

5.1.37 DoubleSided as boolean

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: A Boolean indicating whether the layer displays its content when facing away from the viewer. Animatable.

Notes: When the value in this property is false, the layer hides its content when it faces away from the viewer. The default value of this property is true.

Available in OS X v10.5 and later.

(Read and Write property)

5.1.38 `drawsAsynchronously` as boolean

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: A Boolean indicating whether drawing commands are deferred and processed asynchronously in a background thread.

Notes: When this property is set to true, the graphics context used to draw the layer's contents queues drawing commands and executes them on a background thread rather than executing them synchronously. Performing these commands asynchronously can improve performance in some apps. However, you should always measure the actual performance benefits before enabling this capability.

The default value for this property is false.

Available in OS X v10.8 and later.

(Read and Write property)

5.1.39 `frame` as `CGRectMBS`

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The layer's frame rectangle.

Notes: The frame rectangle is position and size of the layer specified in the superlayer's coordinate space. For layers, the frame rectangle is a computed property that is derived from the values in the `bounds`, `anchorPoint` and `position` properties. When you assign a new value to this property, the layer changes its position and bounds properties to match the rectangle you specified. The values of each coordinate in the rectangle are measured in points.

For more information about the relationship between the `frame`, `bounds`, `anchorPoint` and `position` properties, see Core Animation Programming Guide.

Note: The `frame` property is not directly animatable. Instead you should animate the appropriate combination of the `bounds`, `anchorPoint` and `position` properties to achieve the desired result.

(Read and Write property)

5.1.40 `Handle` as Integer

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The internal reference to the layer.

Notes: (Read and Write property)

5.1.41 Hidden as boolean

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: A Boolean indicating whether the layer is displayed. Animatable.

Notes: The default value of this property is false.

(Read and Write property)

5.1.42 mask as CALayerMBS

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The layer to be used as mask for this layer.

Notes: A layer whose alpha channel is used as a mask to select between the layer's background and the result of compositing the layer's contents with its filtered background. Defaults to nil. When used as a mask the layer's 'compositingFilter' and 'backgroundFilters' properties are ignored. When setting the mask to a new layer, the new layer must have a nil superlayer, otherwise the behavior is undefined. Nested masks (mask layers with their own masks) are unsupported.

(Read and Write property)

5.1.43 masksToBounds as Boolean

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Whether to mask by bounds.

Notes: When true an implicit mask matching the layer bounds is applied to the layer (including the effects of the 'cornerRadius' property). If both 'mask' and 'masksToBounds' are non-nil the two masks are multiplied to get the actual mask values. Defaults to false. Animatable.

(Read and Write property)

5.1.44 minificationFilterBias as Double

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The bias factor used by the minification filter to determine the levels of detail.

Notes: This value is used by the minificationFilter when it is set to kCAFilterTrilinear.

The default value of this property is 0.0.

Available in OS X v10.6 and later.

(Read and Write property)

5.1.45 `modelLayer` as `CALayerMBS`

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Returns the model layer object associated with the receiver, if any.

Notes: Calling this method on a layer in the presentation tree returns the corresponding layer object in the model tree. This method returns a value only when a transaction involving changes to the presentation layer is in progress. If no transaction is in progress, the results of calling this method are undefined.

Available in OS X v10.5 and later.

(Read only property)

5.1.46 `needsDisplay` as `boolean`

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Returns a Boolean indicating whether the layer has been marked as needing an update.

Notes: True if the layer needs to be updated.

Available in OS X v10.6 and later.

(Read only property)

5.1.47 `needsDisplayOnBoundsChange` as `boolean`

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: A Boolean indicating whether the layer contents must be updated when its bounds rectangle changes.

Notes: When this property is set to true, the layer automatically calls its `setNeedsDisplay` method whenever its bounds property changes. The default value of this property is false.

(Read and Write property)

5.1.48 `needsLayout` as `boolean`

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Returns a Boolean indicating whether the layer has been marked as needing a layout update.

Notes: True if the layer has been marked as requiring a layout update.

Available in OS X v10.6 and later.

(Read only property)

5.1.49 opacity as Double

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The opacity of the receiver. Animatable.

Notes: The value of this property must be in the range 0.0 (transparent) to 1.0 (opaque). Values outside that range are clamped to the minimum or maximum. The default value of this property is 1.0.

(Read and Write property)

5.1.50 Opaque as boolean

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: A Boolean value indicating whether the layer contains completely opaque content.

Notes: The default value of this property is false. If your app draws completely opaque content that fills the layer's bounds, setting this property to true lets the system optimize the rendering behavior for the layer. Specifically, when the layer creates the backing store for your drawing commands, Core Animation omits the alpha channel of that backing store. Doing so can improve the performance of compositing operations. If you set the value of this property to true, you must fill the layer's bounds with opaque content.

Setting this property affects only the backing store managed by Core Animation. If you assign an image with an alpha channel to the layer's contents property, that image retains its alpha channel regardless of the value of this property.

(Read and Write property)

5.1.51 position as CGRectMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The layer's position in its superlayer's coordinate space. Animatable.

Notes: The value of this property is specified in points and is always specified relative to the value in the anchorPoint property. For new standalone layers, the default position is set to (0.0, 0.0). Changing the frame property also updates the value in this property.

For more information about the relationship between the frame, bounds, anchorPoint and position properties, see Core Animation Programming Guide.

(Read and Write property)

5.1.52 preferredFrameSize as CGSizeMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Returns the preferred size of the layer in the coordinate space of its superlayer.

Notes: In OS X, the default implementation of this method calls the preferredSizeOfLayer method of its layout manager—that is, the object in its layoutManager property. If that object does not exist or does not implement that method, this method returns the size of the layer’s current bounds rectangle mapped into the coordinate space of its superlayer.

(Read only property)

5.1.53 presentationLayer as CALayerMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Returns a copy of the presentation layer object that represents the state of the layer as it currently appears onscreen.

Notes: The layer object returned by this method provides a close approximation of the layer that is currently being displayed onscreen. While an animation is in progress, you can retrieve this object and use it to get the current values for those animations.

The sublayers, mask, and superlayer properties of the returned layer return the corresponding objects from the presentation tree (not the model tree). This pattern also applies to any read-only layer methods. For example, the hitTest: method of the returned object queries the layer objects in the presentation tree.

Available in OS X v10.5 and later.

(Read only property)

5.1.54 rasterizationScale as Double

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: The scale at which the layer will be rasterized.

Notes: (when the shouldRasterize property has been set to true) relative to the coordinate space of the layer. Defaults to one. Animatable.

(Read and Write property)

5.1.55 shadowColor as Variant

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The color of the layer’s shadow. Animatable.

Notes: The default value of this property is an opaque black color.

The value of this property is retained using the Core Foundation retain/release semantics. This behavior occurs despite the fact that the property declaration appears to use the default assign semantics for object retention.

Value is a `CGColorMBS`. We use variant to reduce dependency to other plugins.
(Read and Write property)

5.1.56 shadowOffset as CGSizeMBS

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The offset (in points) of the layer's shadow. Animatable.

Notes: The default value of this property is (0.0, -3.0).
(Read and Write property)

5.1.57 shadowOpacity as Double

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The opacity of the layer's shadow. Animatable.

Notes: The value in this property must be in the range 0.0 (transparent) to 1.0 (opaque). The default value of this property is 0.0.

Available in OS X v10.5 and later.

(Read and Write property)

5.1.58 shadowPath as Variant

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The shadow path.

Notes: Value is a `CGPathMBS`. We use variant to reduce dependency to other plugins.

The default value of this property is nil, which causes the layer to use a standard shadow shape. If you specify a value for this property, the layer creates its shadow using the specified path instead of the layer's composited alpha channel. The path you provide defines the outline of the shadow. It is filled using the non-zero winding rule and the current shadow color, opacity, and blur radius.

Unlike most animatable properties, this property (as with all `CGPathRef` animatable properties) does not support implicit animation. However, the path object may be animated using any of the concrete subclasses

of `CAPROPERTYAnimation`. Paths will interpolate as a linear blend of the "on-line" points; "off-line" points may be interpolated non-linearly (to preserve continuity of the curve's derivative). If the two paths have a different number of control points or segments, the results are undefined. If the path extends outside the layer bounds it will not automatically be clipped to the layer, only if the normal layer masking rules cause that.

Specifying an explicit path usually improves rendering performance.

The value of this property is retained using the Core Foundation retain/release semantics. This behavior occurs despite the fact that the property declaration appears to use the default assign semantics for object retention.

(Read and Write property)

5.1.59 shadowRadius as Double

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The blur radius (in points) used to render the layer's shadow. Animatable.

Notes: You specify the radius The default value of this property is 3.0.

(Read and Write property)

5.1.60 shouldRasterize as Boolean

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Whether view should raster.

Notes: When true, the layer is rendered as a bitmap in its local coordinate space ("rasterized"), then the bitmap is composited into the destination (with the `minificationFilter` and `magnificationFilter` properties of the layer applied if the bitmap needs scaling). Rasterization occurs after the layer's filters and shadow effects are applied, but before the opacity modulation. As an implementation detail the rendering engine may attempt to cache and reuse the bitmap from one frame to the next. (Whether it does or not will have no affect on the rendered output.) When false the layer is composited directly into the destination whenever possible (however, certain features of the compositing model may force rasterization, e.g. adding filters). Defaults to false. Animatable.

(Read and Write property)

5.1.61 superlayer as CALayerMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The superlayer of the layer.

Notes: The superlayer manages the layout of its sublayers.

Available in OS X v10.5 and later.
(Read only property)

5.1.62 zPosition as Double

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The layer's position on the z axis. Animatable.

Notes: The default value of this property is 0. Changing the value of this property changes the the front-to-back ordering of layers onscreen. This can affect the visibility of layers whose frame rectangles overlap.

The value of this property is measured in points.
Available in OS X v10.5 and later.
(Read and Write property)

5.1.63 Constants

Edge constants

Constant	Value	Description
kCALayerBottomEdge	4	Specifies that the bottom edge of the receiver's content should be antialiased.
kCALayerLeftEdge	1	Specifies that the left edge of the receiver's content should be antialiased.
kCALayerRightEdge	2	Specifies that the right edge of the receiver's content should be antialiased.
kCALayerTopEdge	8	Specifies that the top edge of the receiver's content should be antialiased.

Autoresize Mask Constants

Constant	Value	Description
kCALayerHeightSizable	16	The receiver's height is flexible.
kCALayerMaxXMargin	4	The left margin between the receiver and its superview is flexible.
kCALayerMaxYMargin	32	The top margin between the receiver and its superview is flexible.
kCALayerMinXMargin	1	The left margin between the receiver and its superview is flexible.
kCALayerMinYMargin	8	The top margin between the receiver and its superview is flexible.
kCALayerNotSizable	0	The receiver cannot be resized.
kCALayerWidthSizable	2	The receiver's width is flexible.

5.2 class CATextLayerMBS

5.2.1 class CATextLayerMBS

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: The text layer provides simple text layout and rendering of plain or attributed strings.

Notes: The first line is aligned to the top of the layer.

Subclass of the CALayerMBS class.

Blog Entries

- [MonkeyBread Software Releases the MBS Xojo Plugins in version 18.4](#)
- [MBS Xojo Plugins, version 18.4pr1](#)

Xojo Developer Magazine

- [17.1, page 11: News](#)
- [16.6, page 9: News](#)

5.2.2 Methods

5.2.3 Constructor

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: The constructor.

5.2.4 kCAAlignmentCenter as String

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: One of the modes for alignment.

5.2.5 kCAAlignmentJustified as String

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: One of the modes for alignment.

5.2.6 kCAAlignmentLeft as String

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: One of the modes for alignment.

5.2.7 kCAAlignmentNatural as String

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: One of the modes for alignment.

5.2.8 kCAAlignmentRight as String

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: One of the modes for alignment.

5.2.9 kCATruncationEnd as String

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: One of the modes for truncation.

5.2.10 kCATruncationMiddle as String

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: One of the modes for truncation.

5.2.11 kCATruncationNone as String

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: One of the modes for truncation.

5.2.12 kCATruncationStart as String

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: One of the modes for truncation.

5.2.13 Properties

5.2.14 alignmentMode as String

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Describes how individual lines of text are aligned within the layer bounds.

Notes: The possible options are ‘natural’, ‘left’, ‘right’, ‘center’ and ‘justified’. Defaults to ‘natural’.
(Read and Write property)

5.2.15 allowsFontSubpixelQuantization as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Sets allowsFontSubpixelQuantization parameter of CGContextMBS passed to the drawInContext method.

Notes: Defaults to false.
(Read and Write property)

5.2.16 font as Variant

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: The font to use.

Notes: May be either a CTFontMBS, NSFontMBS, CGFontMBS, or a string naming the font. Defaults to the Helvetica font. Only used when the ‘string’ property is not an NSAttributedStringMBS object.
(Read and Write property)

5.2.17 fontSize as Double

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: The font size.

Notes: Defaults to 36. Only used when the ‘string’ property is not an NSAttributedStringMBS. Animatable

(Mac OS X 10.6 and later.)
(Read and Write property)

5.2.18 foregroundColor as Variant

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: The color object used to draw the text.

Notes: Defaults to opaque white.

Only used when the 'string' property is not an NSAttributedStringMBS object.

Animatable (Mac OS X 10.6 and later.)

(Read and Write property)

5.2.19 String as Variant

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: The text to be rendered, should be either a string or an NSAttributedString.

Notes: Defaults to nil.

(Read and Write property)

5.2.20 truncationMode as String

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: Describes how the string is truncated to fit within the layer bounds.

Notes: The possible options are 'none', 'start', 'middle' and 'end'. Defaults to 'none'.

(Read and Write property)

5.2.21 Wrapped as Boolean

Plugin Version: 18.4, Platform: macOS, Targets: All.

Function: When true the string is wrapped to fit within the layer bounds.

Notes: Defaults to false.

(Read and Write property)

5.3 class CATransactionMBS

5.3.1 class CATransactionMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The plugin class for a CoreAnimation transaction.

Notes: CATransaction is the Core Animation mechanism for batching multiple layer-tree operations into atomic updates to the render tree. Every modification to a layer tree must be part of a transaction. Nested transactions are supported.

Core Animation supports two types of transactions: implicit transactions and explicit transactions. Implicit transactions are created automatically when the layer tree is modified by a thread without an active transaction and are committed automatically when the thread's run-loop next iterates. Explicit transactions occur when the the application sends the CATransaction class a begin message before modifying the layer tree, and a commit message afterwards.

CATransaction allows you to override default animation properties that are set for animatable properties. You can customize duration, timing function, whether changes to properties trigger animations, and provide a handler that informs you when all animations from the transaction group are completed.

During a transaction you can temporarily acquire a recursive spin-lock for managing property atomicity. This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [MBS Real Studio Plugins, version 13.1pr9](#)

5.3.2 Methods

5.3.3 animationDuration as Double

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Returns the animation duration used by all animations within this transaction group.

5.3.4 available as boolean

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Returns true if transaction class is available.

Notes: Returns true on Mac and false on Linux/Windows.

5.3.5 begin

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Begin a new transaction for the current thread.

Notes: The transaction is nested within the thread's current transaction, if there is one. Available in OS X v10.5 and later.

5.3.6 commit

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Commit all changes made during the current transaction.

Notes: Raises an exception if no current transaction exists. Available in OS X v10.5 and later.

5.3.7 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The private constructor.

5.3.8 flush

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Flushes any extant implicit transaction.

Notes: Delays the commit until any nested explicit transactions have completed.

Flush is typically called automatically at the end of the current runloop, regardless of the runloop mode. If your application does not have a runloop, you must call this method explicitly.

However, you should attempt to avoid calling flush explicitly. By allowing flush to execute during the runloop your application will achieve better performance, atomic screen updates will be preserved, and transactions and animations that work from transaction to transaction will continue to function.

Available in OS X v10.5 and later.

5.3.9 `kCATransactionAnimationDuration` as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the keys for values of a transaction.

Notes: Duration, in seconds, for animations triggered within the transaction group. The value for this key must be a number.

Available in OS X v10.5 and later.

5.3.10 `kCATransactionDisableActions` as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the keys for values of a transaction.

Notes: If true, implicit actions for property changes made within the transaction group are suppressed. The value for this key must be a boolean.

Available in OS X v10.5 and later.

5.3.11 `setAnimationDuration(value as Double)`

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Sets the animation duration used by all animations within this transaction group.

Notes: Available in OS X v10.6 and later.

5.3.12 `setValue(value as Variant, key as string)`

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Sets the arbitrary keyed-data for the specified key.

Notes: value: The value for the key identified by key.

key: The name of one of the receiver's properties.

Nested transactions have nested data scope; setting a key always sets it in the innermost scope.

Available in OS X v10.5 and later.

5.3.13 `valueForKey(key as string) as Variant`

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Returns the arbitrary keyed-data specified by the given key.

Notes: key: The name of one of the receiver's properties.

Returns the value for the data specified by the key.

Nested transactions have nested data scope. Requesting a value for a key first searches the innermost scope, then the enclosing transactions.

Available in OS X v10.5 and later.

5.3.14 Properties

5.3.15 Handle as Integer

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The internal transaction handle.

Notes: (Read and Write property)

Chapter 6

CoreGraphics

6.1 Globals

6.1.1 CGNewPDFDocumentMBS(file as folderitem, MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean) as CGPDFContextMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Creates a PDF document.

Notes: Title, Author, Creator, Subject and Keywords parameters can be empty.

If OwnerPassword and UserPassword are filled in the PDF is encrypted and AllowsPrinting/AllowsCopy define what the user can do after he entered his password.

The passwords must be a string which can be represented in ASCII encoding; only the first 32 bytes will be used for the password.

Requires Mac OS X to work.

See also:

- 6.1.2 CGNewPDFDocumentMBS(file as folderitem, MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean, KeyLength as Integer) as CGPDFContextMBS 98

6.1.2 CGNewPDFDocumentMBS(file as folderitem, MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean, KeyLength as Integer) as CGPDFContextMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Creates a PDF document.

Notes: Title, Author, Creator, Subject and Keywords parameters can be empty.

If OwnerPassword and UserPassword are filled in the PDF is encrypted and AllowsPrinting/AllowsCopy define what the user can do after he entered his password. #

The passwords must be a string which can be represented in ASCII encoding; only the first 32 bytes will be used for the password.

Keylength must be a value between 48 bit and 128 bit in 8 bit steps. 0 uses default value.

Requires Mac OS X to work.

Xojo Developer Magazine

- [6.4, pages 30 to 31: Creating PDF Files, How to create PDFs using the MBS Plugins by Christian Schmitz](#)

See also:

- [6.1.1 CGNewPDFDocumentMBS\(file as folderitem, MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean\) as CGPDFContextMBS](#) 97

6.1.3 GetCurrentCGContextMBS as CGContextMBS

Plugin Version: 9.8, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a new CGContextMBS object for the current Cocoa graphics context.

Notes: Requires Mac OS X to work.

Returns nil on any error.

6.1.4 CGContextCreateMBS(data as memoryblock, width as Integer, height as Integer, bitsPerComponent as Integer, bytesPerRow as Integer, colorspace as CGContextMBS, alphaInfo as Integer) as CGContextMBS

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Create a bitmap context.

Notes: The context draws into a bitmap which is 'width' pixels wide and 'height' pixels high. The number of components for each pixel is specified by 'colorspace', which also may specify a destination color profile. The number of bits for each component of a pixel is specified by 'bitsPerComponent', which must be 1, 2, 4, or 8. Each row of the bitmap consists of 'bytesPerRow' bytes, which must be at least '(width * bitsPerComponent * number of components + 7)/8' bytes. 'data' points a block of memory at least 'bytesPerRow * height' bytes. 'alphaInfo' specifies whether the bitmap should contain an alpha channel, and how it's to be generated.

Fails if data=nil or colorspace=nil.

The memoryblock is not referenced and not stored, so keep it alive while using the CGContext object.

Returns nil on any error.

data

A pointer to the destination in memory where the drawing is to be rendered. The size of this memoryblock should be at least(bytesPerRow*height) bytes.

width

The width of the bitmap in pixels.

height

The height of the bitmap in pixels.

bitsPerComponent

The number of bits to use for each component of a pixel in memory. Allowable values are 4, 5, or 8. For example, for a 32-bit RGB(A) colorspace, you would specify a value of 8 bits per color component. In combination, the number of bits per component, the color space, and the alpha value determine which bitmap context formats Quartz supports.

bytesPerRow

The number of bytes of memory to use per row of the bitmap. This value must be at least the product of the width and bitsPerComponent parameters, times the number of components per pixel. The result should be divided by 8 and rounded up to the nearest whole number to obtain the number of bytes to use per row. That is, the value must be at least $((\text{width}) * (\text{bits per component}) * (\text{number of components per pixel}) + 7) / 8$ bytes. For a given row, Quartz stores bitmap data for the first width pixels and ignores any remaining bytes. The colorspace value referenced by the colorspace parameter specifies the number of components for each

pixel.

colorspace

The color space to use for the bitmap context.

alphaInfo

A `CGImageAlphaInfo` constant specifying whether the bitmap should contain an alpha channel and how it is to be generated. The alpha value determines the opacity of a pixel when it is drawn.

Supported pixel formats:

Pixel format	Color space	Bits per pixel	Bits per component	Alpha option
Gray_8	Grayscale	8	8	<code>kCGImageAlphaNone</code>
RGB555	RGB	16	5	<code>kCGImageAlphaNoneSkipFirst</code>
XRGB_32	RGB	32	8	<code>kCGImageAlphaNoneSkipFirst</code>
ARGB_32	RGB	32	8	<code>kCGImageAlphaPremultipliedFirst</code>
RGBX_32	RGB	32	8	<code>kCGImageAlphaNoneSkipLast</code>
RGBA_32	RGB	32	8	<code>kCGImageAlphaPremultipliedLast</code>

Quartz does not support the following formats in a bitmap context:

- 1-bit grayscale
- 24-bit RGB
- CMYK (any depth)

`CGImageAlphaInfo` constants:

<code>kCGImageAlphaNone</code>	0	
<code>kCGImageAlphaPremultipliedLast</code>	1	For example, premultiplied RGBA
<code>kCGImageAlphaPremultipliedFirst</code>	2	For example, premultiplied ARGB
<code>kCGImageAlphaLast</code>	3	For example, non-premultiplied RGBA
<code>kCGImageAlphaFirst</code>	4	For example, non-premultiplied ARGB
<code>kCGImageAlphaNoneSkipLast</code>	5	Equivalent to <code>kCGImageAlphaNone</code> .
<code>kCGImageAlphaNoneSkipFirst</code>	6	

6.1.5 `CGOpenPDFDocumentMBS(dataprovider as CGDataProviderMBS)` as `CGPDFDocumentMBS`

Plugin Version: 5.1, Platform: macOS, Targets: All.

Function: Opens a CG PDF Document from a data stream.

Example:

```

Sub Paint(g As Graphics)
// window.paint event
Var f as FolderItem
Var b as BinaryStream
Var s as string
Var d as CGDataProviderMBS
Var p as CGPDFDocumentMBS

// get a folderitem to a pdf file.
f=SpecialFolder.Desktop.Child("notes.pdf")

// load the content in a string variable
b=f.OpenAsBinaryFile(false)
s=b.Read(b.Length)
b.Close

// now make a CGDataProvider based on a string.
d=CGDataProviderMBS.CreateWithData(s)

// Open the PDF from the Data Provider
p=CGOpenPDFDocumentMBS(d)

// And play with it
g.DrawCGPDFDocumentMBS p,p.MediaBox(1),1

Exception
// trouble goes here.
End Sub

```

Notes: Returns nil on any error.

6.1.6 CGShadingCreateAxialMBS(colorspace as CGColorSpaceMBS, startPoint as CGPointMBS, endPoint as CGPointMBS, func as CGFunctionMBS, extendStart as boolean, extendEnd as boolean) as CGShadingMBS

Plugin Version: 6.2, Platform: macOS, Targets: All.

Function: Creates a Quartz axial shading.

Notes: Parameters:

colorspace: The color space in which color values are expressed. Quartz retains this object; upon return, you may safely release it.

startPoint: The starting point of the axis, in the shading's target coordinate space.

endPoint: The ending point of the axis, in the shading's target coordinate space.

function: A CGFunction object. This object refers to your function for creating an axial shading. Quartz

retains this object; upon return, you may safely release it.

`extendStart`: A Boolean value that specifies whether to extend the shading beyond the starting point of the axis.

`extendEnd`: A Boolean value that specifies whether to extend the shading beyond the ending point of the axis.

Returns a new Quartz axial shading or nil on any error.

Discussion

An axial shading is a color blend that varies along a linear axis between two endpoints and extends indefinitely perpendicular to that axis. When you are ready to draw the shading, call the function `CGContextMBS.DrawShading`.

Available in Mac OS X version 10.2 and later.

6.1.7 `CGShadingCreateRadialMBS(colorspace as CGColorSpaceMBS, startPoint as CGPointMBS, startRadius as Double, endPoint as CGPointMBS, endRadius as Double, func as CGFunctionMBS, extendStart as boolean, extendEnd as boolean) as CGShadingMBS`

Plugin Version: 6.2, Platform: macOS, Targets: All.

Function: Creates a Quartz radial shading.

Notes: Parameters

`colorspace`: The color space in which color values are expressed.

`startPoint`: The center of the starting circle, in the shading's target coordinate space.

`startRadius`: The radius of the starting circle, in the shading's target coordinate space.

`endPoint`: The center of the ending circle, in the shading's target coordinate space.

`endRadius`: The radius of the ending circle, in the shading's target coordinate space.

`function`: A `CGFunction` object. This object refers to your function for creating a radial shading.

`extendStart`: A Boolean value that specifies whether to extend the shading beyond the starting circle.

`extendEnd`: A Boolean value that specifies whether to extend the shading beyond the ending circle.

Returns a new Quartz radial shading or nil on any error.

A radial shading is a color blend that varies between two circles. To draw the shading, call the function `CGContextDrawShading`.

Available in Mac OS X version 10.2 and later.

6.1.8 CGCreateImageFromJPEGDataProviderMBS(dataprovider as Variant, decode as memoryblock, shouldInterpolate as boolean, intent as Integer) as CGImageMBS

Plugin Version: 3.3, Platform: macOS, Targets: All.

Function: Creates a new CGImage with JPEG data.

Example:

```
// Shows moon.jpg from the desktop folder
// shows in the window title if i,p or u is nil.

Sub Mainwindow.Paint(g As Graphics)
  Var f as FolderItem
  Var p as CGDataProviderMBS
  Var i as CGImageMBS

  f=SpecialFolder.Desktop.Child("moon.jpg")
  p=CGDataProviderMBS.CreateWithFile(f)
  if p=nil then
    Title="p=nil"
  else
    i=CGCreateImageFromJPEGDataProviderMBS(p,nil,true,0)

    if i=nil then
      Title="i=nil"
    else
      window1.CGContextMBS.DrawPicture i,CGMakeRectMBS(0,0,i.Width,i.Height)
    end if
  end if

End Sub
```

Notes: Dataprovider must be a CGDataProviderMBS object.

Parameters:

dataprovider:

A reference to a data provider supplying JPEG-encoded data.

decode:

Pass the decode array for the image. In the decode array, for each color component in the source color space, you provide a pair of values denoting the upper and lower limits of a range. For example, the decode array for a source image in the RGB color space would contain six entries total, consisting of one pair each for red, green, and blue. When the image is rendered, Quartz uses a linear transform to map the original component value into a relative number within your designated range that is appropriate for the destination

color space. If you do not want to allow remapping of the image's color values, pass nil for the decode array. The memoryblock for the array needs to be filled with double values.

shouldInterpolate:

Pass true if interpolation should occur; otherwise, pass false . The interpolation setting specifies whether Quartz should apply a pixel-smoothing algorithm to the image. If you pass false , the image may appear jagged or pixelated when drawn on an output device with higher resolution than the image data.

intent:

Pass a CGColorRenderingIntent value specifying how Quartz should display colors in the image that are not located within the current color space of the graphics context. The rendering intent determines the exact method used to map colors from one color space to another.

6.1.9 CGCreateImageFromPNGDataProviderMBS(dataprovider as Variant, decode as memoryblock, shouldInterpolate as boolean, intent as Integer) as CGImageMBS

Plugin Version: 3.3, Platform: macOS, Targets: All.

Function: Creates a new CGImage with PNG data.

Notes: Dataprovider must be a CGDataProviderMBS object.

Parameters:

dataprovider:

A reference to a data provider supplying JPEG-encoded data.

decode:

Pass the decode array for the image. In the decode array, for each color component in the source color space, you provide a pair of values denoting the upper and lower limits of a range. For example, the decode array for a source image in the RGB color space would contain six entries total, consisting of one pair each for red, green, and blue. When the image is rendered, Quartz uses a linear transform to map the original component value into a relative number within your designated range that is appropriate for the destination color space. If you do not want to allow remapping of the image's color values, pass nil for the decode array. The memoryblock for the array needs to be filled with double values.

shouldInterpolate:

Pass true if interpolation should occur; otherwise, pass false . The interpolation setting specifies whether Quartz should apply a pixel-smoothing algorithm to the image. If you pass false , the image may appear jagged or pixelated when drawn on an output device with higher resolution than the image data.

intent:

Pass a `CGColorRenderingIntent` value specifying how Quartz should display colors in the image that are not located within the current color space of the graphics context. The rendering intent determines the exact method used to map colors from one color space to another.

6.1.10 `CGCreateImageMBS(pic as picture)` as `CGImageMBS`

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Creates a new `CGImageMBS` from the given images.

Example:

```
Var c as CGImageMBS
Var pic, mask as Picture
// get picture and mask

c=CGCreateImageMBS(pic)
if c<>Nil then
// go on
end if
```

Notes: If the image has a mask, it is used.

See also:

- 6.1.11 `CGCreateImageMBS(pic as picture, mask as picture)` as `CGImageMBS`

105

6.1.11 `CGCreateImageMBS(pic as picture, mask as picture)` as `CGImageMBS`

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Creates a new `CGImageMBS` from the given images.

Example:

```
Var c as CGImageMBS
Var pic, mask as Picture
// get picture and mask

c=CGCreateImageMBS(pic, mask)
if c<>Nil then
// go on
end if
```

Notes: The mask is taken from the second image.

With 11.3 plugins we are deprecating to pass a mask. The plugin prefers to simply take the mask or alpha channel of the picture itself.

Blog Entries

- [Notes and Tips](#)
- [MonkeyBread Software Releases the MBS REALbasic plug-ins 8.6](#)

See also:

- 6.1.10 `CGCreateImageMBS(pic as picture)` as `CGImageMBS` 105

6.1.12 `CGMakePointMBS(x as Double, y as Double)` as `CGPointMBS`

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Returns a new `CGPointMBS` object.

6.1.13 `CGMakeRectMBS(left as Double, top as Double, width as Double, height as Double)` as `CGRectMBS`

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Returns a new `CGRectMBS` object with the given value.

Blog Entries

- [Colorspaces in MacOS with Xojo](#)
- [Automated video editing with Xojo](#)

Xojo Developer Magazine

- [6.4, page 30: Creating PDF Files, How to create PDFs using the MBS Plugins by Christian Schmitz](#)

6.1.14 `CGMakeSizeMBS(width as Double, height as Double)` as `CGSizeMBS`

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Returns a new `CGSizeMBS` object.

Blog Entries

- [Automated video editing with Xojo](#)

6.1.15 `CGNewPDFDocumentMBS(consumer as CGDataConsumerMBS, MediaBox as CGRectMBS, title as string, author as string, creator as string)` as `CGPDFContextMBS`

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Creates a PDF document.

Notes: Title, author and creator are all optional.

Requires Mac OS X to work.

Keep yourself a reference to the consumer object so RB does not release it resulting in a crash.

6.1.16 CGNewPDFDocumentMBS(file as folderitem, MediaBox as CGRectMBS, title as string, author as string, creator as string) as CGPDFContextMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Creates a PDF document.

Notes: Title, author and creator are all optional.

Requires Mac OS X to work.

6.1.17 CGOpenPDFDocumentMBS(file as folderitem) as CGPDFDocumentMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Opens a PDF document.

Notes: RB 4.5 should do this perfectly, but older RB versions may have problems with longer file names.

Requires Mac OS X to work.

Blog Entries

- [MBS REALbasic Plugins Version 10.4 release notes](#)
- [MBS REALbasic Plugins, version 10.4pr12](#)
- [MBS REALbasic Plugins, version 10.4pr11](#)

6.2 class CGAffineTransformMBS

6.2.1 class CGAffineTransformMBS

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: A class for an affine transformation.

Example:

```
Var af as CGAffineTransformMBS = CGAffineTransformMBS.Identity
```

```
MsgBox str(af.A)+" "+str(af.b)+" "+str(af.c)+" "+str(af.d)+" "+str(af.tx)+" "+str(af.ty)
```

```
af = af.Scale( 1, -1 )
```

```

MsgBox str(AF.A)+" "+str(AF.b)+" "+str(AF.c)+" "+str(AF.d)+" "+str(AF.tx)+" "+str(AF.ty)

af = af.Translate( 0, 100 )

MsgBox str(AF.A)+" "+str(AF.b)+" "+str(AF.c)+" "+str(AF.d)+" "+str(AF.tx)+" "+str(AF.ty)

Var r1 as CGRectMBS
Var r2 as CGRectMBS

r1=CGMakeRectMBS(100,100,100,100)

r2=r1.ApplyAffineTransform(af)

MsgBox "("+str(r1.Left)+", "+str(r1.top)+", "+str(r1.width)+", "+str(r1.height)+") => "("+str(r2.Left)+",
"+str(r2.top)+", "+str(r2.width)+", "+str(r2.height)+")"
```

Blog Entries

- [MBS Xojo Plugins, version 20.2pr2](#)
- [Automated video editing with Xojo](#)

6.2.2 Methods

6.2.3 Binary as MemoryBlock

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Returns the binary data of the object for toolbox calls.

6.2.4 Concat(t as CGAffineTransformMBS) as CGAffineTransformMBS

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Concatenate 't2' to 't1' and return the result: $t' = t1 * t2$

6.2.5 Constructor

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: This constructor creates the identity transform: $[1 0 0 1 0 0]$.

See also:

6.2. CLASS CGAFFINETRANSFORMMBS	109
• 6.2.6 Constructor(a as double, b as double, c as double, d as double, tx as double = 0.0, ty as double = 0.0)	109
• 6.2.7 Constructor(p as Ptr)	109
• 6.2.8 Constructor(transform as CGAffineTransformMBS)	109

6.2.6 Constructor(a as double, b as double, c as double, d as double, tx as double = 0.0, ty as double = 0.0)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates the transformation with the given values.

See also:

• 6.2.5 Constructor	108
• 6.2.7 Constructor(p as Ptr)	109
• 6.2.8 Constructor(transform as CGAffineTransformMBS)	109

6.2.7 Constructor(p as Ptr)

Plugin Version: 17.1, Platform: macOS, Targets: All.

Function: Creates a new instance using data at the pointer.

Notes: Make sure the pointer is valid and has the right data and size.

See also:

• 6.2.5 Constructor	108
• 6.2.6 Constructor(a as double, b as double, c as double, d as double, tx as double = 0.0, ty as double = 0.0)	109
• 6.2.8 Constructor(transform as CGAffineTransformMBS)	109

6.2.8 Constructor(transform as CGAffineTransformMBS)

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Creates a matrix object with copying the values.

Example:

```
Var m as new CGAffineTransformMBS(1,2,3,4,5,6)
```

```
Var n as new CGAffineTransformMBS(m)
```

```
MsgBox n
```

See also:

- 6.2.5 Constructor 108
- 6.2.6 Constructor(a as double, b as double, c as double, d as double, tx as double = 0.0, ty as double = 0.0) 109
- 6.2.7 Constructor(p as Ptr) 109

6.2.9 EqualToTransform(t as CGAffineTransformMBS) as boolean

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Checks whether two affine transforms are equal.

Notes: Returns true if t1 and t2 are equal, false otherwise.

Available in Mac OS X v10.4 and later.

6.2.10 Identity as CGAffineTransformMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The identity transform: $[1 \ 0 \ 0 \ 1 \ 0 \ 0]$.

6.2.11 Invert as CGAffineTransformMBS

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Invert 't' and return the result. If 't' has zero determinant, then 't' is returned unchanged.

6.2.12 IsIdentity as boolean

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Checks whether an affine transform is the identity transform.

Notes: Returns true if t is the identity transform, false otherwise.

Available in Mac OS X v10.4 and later.

6.2.13 Make(a as double, b as double, c as double, d as double, tx as double = 0.0, ty as double = 0.0) as CGAffineTransformMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Return the transform $[a \ b \ c \ d \ tx \ ty]$.

6.2.14 MakeRotation(angle as Double) as CGAffineTransformMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Return a transform which rotates by 'angle' radians: $t' = \begin{bmatrix} \cos(\text{angle}) & \sin(\text{angle}) & -\sin(\text{angle}) \\ \cos(\text{angle}) & 0 & 0 \end{bmatrix}$

6.2.15 MakeScale(sx as Double, sy as Double) as CGAffineTransformMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Return a transform which scales by '(sx, sy)': $t' = \begin{bmatrix} \text{sx} & 0 & 0 \\ 0 & \text{sy} & 0 \end{bmatrix}$

6.2.16 MakeTranslation(tx as Double, ty as Double) as CGAffineTransformMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Return a transform which translates by '(tx, ty)': $t' = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & \text{tx} \\ 0 & 0 & \text{ty} \end{bmatrix}$

6.2.17 Operator_Convert as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Returns string value.

6.2.18 Operator_Multiply(other as CGAffineTransformMBS) as CGAffineTransformMBS

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Multiplies both matrix where self is on the left side.

6.2.19 Operator_MultiplyRight(other as CGAffineTransformMBS) as CGAffineTransformMBS

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Multiplies both matrix where self is on the right side.

6.2.20 Rotate(angle as Double) as CGAffineTransformMBS

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Rotate 't' by 'angle' radians and return the result: $t' = [\cos(\text{angle}) \sin(\text{angle}) -\sin(\text{angle}) \cos(\text{angle}) 0 0] * t$

6.2.21 Scale(sx as Double, sy as Double) as CGAffineTransformMBS

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Scale 't' by '(sx, sy)' and return the result: $t' = [sx 0 0 sy 0 0] * t$

6.2.22 Translate(tx as Double, ty as Double) as CGAffineTransformMBS

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Translate 't' by '(tx, ty)' and return the result: $t' = [1 0 0 1 tx ty] * t$

Example:

```
Var t as new CGAffineTransformMBS(1,0,0,1,1,1)
t=t.Translate(2,3)
```

```
MsgBox str(T.A)+" "+str(t.B)+" "+str(t.C)+" "+str(t.D)+" "+str(t.TX)+" "+str(t.TY)
```

```
// shows 1 0 0 1 3 4
```

6.2.23 Properties

6.2.24 A as Double

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: The A value.

Notes: (Read and Write property)

6.2.25 B as Double

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: The B value.

Notes: (Read and Write property)

6.2.26 C as Double

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: The C value.

Notes: (Read and Write property)

6.2.27 D as Double

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: The D value.

Notes: (Read and Write property)

6.2.28 StringValue as String

Plugin Version: 20.2, Platform: macOS, Targets: All.

Function: Returns string value.

Notes: (Read only property)

6.2.29 TX as Double

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: The TX value.

Notes: (Read and Write property)

6.2.30 TY as Double

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: The TY value.

Notes: (Read and Write property)

6.3 class CGContextMBS

6.3.1 class CGContextMBS

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: A class to hold a CoreGraphics bitmap context.

Example:

```
Sub Paint(g As Graphics)
  Var c as CGContextMBS
  Var w as CGContextMBS // of window
  Var p,m as Picture

  c=new CGContextMBS(100,100)

  c.ClearRect CGMakeRectMBS(0,0,100,100)
  c.SetRGBFillColor 1,0,0,1
  c.FillRect CGMakeRectMBS(0,0,50,50)

  w=window1.CGContextMBS // we are inside paint event!
  // Draw using CGImage
  w.DrawPicture c.CGImage(false,0),CGMakeRectMBS(0,0,c.BitmapWidth,c.BitmapHeight)
  w.Flush

  // Draw using RB picture, so we can see whether it looks equal.
  m=c.CopyPictureMask
  p=New Picture(m.Width,m.Height,32)
  p.Graphics.DrawPicture c.CopyPicture,0,0
  p.Mask.Graphics.DrawPicture m,0,0
  g.DrawPicture p,0,0

End Sub
```

Notes: If the RB graphics class is like a CGContext, the RB picture class (created using NewPicture) is something like a CGContextMBS class.

Subclass of the CGContextMBS class.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [Colorspaces in MacOS with Xojo](#)
- [MBS Xojo / Real Studio Plugins, version 15.0pr5](#)
- [MBS Xojo / Real Studio Plugins, version 14.3pr11](#)
- [MBS Real Studio Plugins, version 13.1pr3](#)
- [MBS REALbasic Plugins, version 11.2pr1](#)

6.3.2 Methods

6.3.3 CGImage(shouldInterpolate as boolean = false, intent as Integer = 0) as CGImageMBS

Plugin Version: 4.1, Platform: macOS, Targets: All.

Function: Creates a CGImageMBS object referencing the CGContext object.

Example:

```

Var c as CGPictureContextMBS
Var w as CGContextMBS // of window

c=new CGPictureContextMBS(100,100)

c.SetRGBFillColor 1,0,0,1
c.FillRect CGMakeRectMBS(0,0,50,50)

w=window1.CGContextMBS // we are inside paint event!
w.DrawPicture c.CGImage(false,0),CGMakeRectMBS(0,0,c.BitmapWidth,c.BitmapHeight)
w.Flush

```

Notes: You will crash your application if you use this Image after the CGContext object was destroyed.

Changes made to the connected CGContext will be seen in the CGImage.

Returns nil on low memory.

Constants for intent:

kCGRenderingIntentDefault	0
kCGRenderingIntentAbsoluteColorimetric	1
kCGRenderingIntentRelativeColorimetric	2
kCGRenderingIntentPerceptual	3
kCGRenderingIntentSaturation	4

Set shouldInterpolate to true if the image should use interpolation.

6.3.4 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: The private constructor.

6.3.5 Create(data as memoryblock, width as Integer, height as Integer, bitsPerComponent as Integer, bytesPerRow as Integer, colorspace as CGColorSpaceMBS, alphaInfo as Integer) as CGContextMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Create a bitmap context.

Notes: The context draws into a bitmap which is 'width' pixels wide and 'height' pixels high. The number of components for each pixel is specified by 'colorspace', which also may specify a destination color profile. The number of bits for each component of a pixel is specified by 'bitsPerComponent', which must be 1, 2, 4, or 8. Each row of the bitmap consists of 'bytesPerRow' bytes, which must be at least '(width * bitsPerComponent * number of components + 7)/8' bytes. 'data' points a block of memory at least 'bytesPerRow * height' bytes. 'alphaInfo' specifies whether the bitmap should contain an alpha channel, and how it's to be generated.

Fails if data=nil or colorspace=nil.

The memoryblock is not referenced and not stored, so keep it alive while using the CGContext object.

Returns nil on any error.

data

A pointer to the destination in memory where the drawing is to be rendered. The size of this memoryblock should be at least(bytesPerRow*height) bytes. Can be nil to let system allocate it.

width

The width of the bitmap in pixels.

height

The height of the bitmap in pixels.

bitsPerComponent

The number of bits to use for each component of a pixel in memory. Allowable values are 4, 5, or 8. For example, for a 32-bit RGB(A) colorspace, you would specify a value of 8 bits per color component. In combination, the number of bits per component, the color space, and the alpha value determine which bitmap context formats Quartz supports.

bytesPerRow

The number of bytes of memory to use per row of the bitmap. This value must be at least the product of the width and bitsPerComponent parameters, times the number of components per pixel. The result should be divided by 8 and rounded up to the nearest whole number to obtain the number of bytes to use per row. That is, the value must be at least $((width) * (bits\ per\ component) * (number\ of\ components\ per\ pixel)) + 7 / 8$ bytes. For a given row, Quartz stores bitmap data for the first width pixels and ignores any remaining bytes. The colorspace value referenced by the colorspace parameter specifies the number of components for each pixel. Can be zero to let system decide (especially if data is nil)

colorspace

The color space to use for the bitmap context.

alphaInfo

A *CGImageAlphaInfo* constant specifying whether the bitmap should contain an alpha channel and how it is to be generated. The alpha value determines the opacity of a pixel when it is drawn.

Supported pixel formats:

Pixel format	Color space	Bits per pixel	Bits per component	Alpha option
Gray_8	Grayscale	8	8	<i>kCGImageAlphaNone</i>
RGB555	RGB	16	5	<i>kCGImageAlphaNoneSkipFirst</i>
XRGB_32	RGB	32	8	<i>kCGImageAlphaNoneSkipFirst</i>
ARGB_32	RGB	32	8	<i>kCGImageAlphaPremultipliedFirst</i>
RGBX_32	RGB	32	8	<i>kCGImageAlphaNoneSkipLast</i>
RGBA_32	RGB	32	8	<i>kCGImageAlphaPremultipliedLast</i>

Quartz does not support the following formats in a bitmap context:

- 1-bit grayscale
- 24-bit RGB
- CMYK (any depth)

CGImageAlphaInfo constants:

<i>kCGImageAlphaNone</i>	0	
<i>kCGImageAlphaPremultipliedLast</i>	1	For example, premultiplied RGBA
<i>kCGImageAlphaPremultipliedFirst</i>	2	For example, premultiplied ARGB
<i>kCGImageAlphaLast</i>	3	For example, non-premultiplied RGBA
<i>kCGImageAlphaFirst</i>	4	For example, non-premultiplied ARGB
<i>kCGImageAlphaNoneSkipLast</i>	5	Equivalent to <i>kCGImageAlphaNone</i> .
<i>kCGImageAlphaNoneSkipFirst</i>	6	

See also:

- 6.3.6 *Create(Other as CGBitmapContextMBS, NewColorspace as CGColorSpaceMBS) as CGBitmapContextMBS* 117

6.3.6 *Create(Other as CGBitmapContextMBS, NewColorspace as CGColorSpaceMBS) as CGBitmapContextMBS*

Plugin Version: 14.3, Platform: macOS, Targets: All.

Function: Creates a new bitmap object.

Example:

```

Var pic as new Picture(100,100) // some picture
Var ICCProfileData as memoryblock // get a IIC Profile somewhere

Var colorspace as CGColorSpaceMBS = CGColorSpaceMBS.CreateWithICCProfile(ICCProfileData)
Var bitmap as CGBitmapContextMBS = CGBitmapContextMBS.CreateWithPicture(pic)
Var zweiteBitmap as CGBitmapContextMBS = bitmap.Create(bitmap, colorspace)

```

Notes: The new bitmap object uses same data as existing object, just accesses the pixels using the new color space.

See also:

- 6.3.5 Create(data as memoryblock, width as Integer, height as Integer, bitsPerComponent as Integer, bytesPerRow as Integer, colorspace as CGColorSpaceMBS, alphaInfo as Integer) as CGBitmapContextMBS 116

6.3.7 CreateImage as CGImageMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Return an image containing a snapshot of the bitmap context.

Example:

```

// a new picture in RB
Var pic as new Picture(500, 500)

// and create CGBitmapContextMBS pointing to it
Var b as CGBitmapContextMBS = CGBitmapContextMBS.CreateWithPicture(pic)

// color set to full red
b.SetRGBFillColor 1.0, 0.0, 0.0, 1.0

// draw ellipse
Var r as CGRectMBS = CGRectMBS.Make(0, 0, 500, 500)
b.FillEllipseInRect r

// flush drawings
b.Flush

// now try CGImage creation
Var cgimage as CGImageMBS = b.CreateImage

// and display by converting to a new picture
Backdrop = cgimage.Picture

```

Notes: If context is not a bitmap context, or if the image cannot be created for any reason, this function returns NULL. This is a "copy" operation subsequent changes to context will not affect the contents of the returned image.

Note that in some cases the copy will actually follow "copy-on-write" semantics, so that the actual physical copy of the bits will only occur if the underlying data in the bitmap context is modified. As a consequence, you may wish to use the resulting image and release it before performing more drawing into the bitmap context; in this way, the actual physical copy of the data may be avoided.

6.3.8 CreateRGB(data as memoryblock, width as Integer, height as Integer, bytesPerRow as Integer, colorspace as CGColorSpaceMBS = nil) as CGBitmapContextMBS

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: Convenience function to handle RGB data.

Notes: Same as Create method, but for RGB data.

Converts data from 3 byte/pixel to 4 byte/pixel and then creates CGBitmapContextMBS.

Colorspace is optional and defaults to Generic RGB.

Returns nil on error, raises OutOfBounds exception for invalid parameters.

6.3.9 CreateWithPicture(Pic as Picture) as CGBitmapContextMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Creates a CGBitmapContext referencing the given picture.

Example:

```
// a new picture in RB
Var pic as new Picture(500, 500)

// and create CGBitmapContextMBS pointing to it
Var b as CGBitmapContextMBS = CGBitmapContextMBS.CreateWithPicture(pic)

// color set to full red
b.SetRGBFillColor 1.0, 0.0, 0.0, 1.0

// draw ellipse
Var r as CCGRectMBS = CCGRectMBS.Make(0, 0, 500, 500)
b.FillEllipseInRect r

// flush drawings
b.Flush
```

```
// and show
Backdrop = pic
```

Notes: Only for Cocoa target.
The plugin will do a clear cache on the picture in the destructor.

6.3.10 Properties

6.3.11 BitmapAlphaInfo as Integer

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Returns the alpha info of the bitmap.

Example:

```
// a new picture in RB
Var pic as new Picture(500, 500)

// and create CGContextMBS pointing to it
Var b as CGContextMBS = CGContextMBS.CreateWithPicture(pic)

// shows 2 for kCGImageAlphaPremultipliedFirst
MsgBox str(b.BitmapAlphaInfo)
```

Notes: Returns 0 on any error.

CGImageAlphaInfo constants:

kCGImageAlphaNone	0	
kCGImageAlphaPremultipliedLast	1	For example, premultiplied RGBA
kCGImageAlphaPremultipliedFirst	2	For example, premultiplied ARGB
kCGImageAlphaLast	3	For example, non-premultiplied RGBA
kCGImageAlphaFirst	4	For example, non-premultiplied ARGB
kCGImageAlphaNoneSkipLast	5	Equivalent to kCGImageAlphaNone.
kCGImageAlphaNoneSkipFirst	6	

(Read only property)

6.3.12 BitmapBitsPerComponent as Integer

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Returns the bits per component of the bitmap.

Example:

```
// a new picture in RB
Var pic as new Picture(500, 500)

// and create CGContextMBS pointing to it
Var b as CGContextMBS = CGContextMBS.CreateWithPicture(pic)

// shows 8
MsgBox str(b.BitmapBitsPerComponent)
```

Notes: Returns 0 on any error.
(Read only property)

6.3.13 BitmapBitsPerPixel as Integer

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Returns the bits per pixel of the bitmap.

Example:

```
// a new picture in RB
Var pic as new Picture(500, 500)

// and create CGContextMBS pointing to it
Var b as CGContextMBS = CGContextMBS.CreateWithPicture(pic)

// shows 32
MsgBox str(b.BitmapBitsPerPixel)
```

Notes: Returns 0 on any error.
(Read only property)

6.3.14 BitmapBytesPerRow as Integer

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Returns the bytes per row of the bitmap.

Example:

```
// a new picture in RB
Var pic as new Picture(500, 500)

// and create CGContextMBS pointing to it
```

```
Var b as CGContextMBS = CGContextMBS.CreateWithPicture(pic)
```

```
// shows 2000
MsgBox str(b.BitmapBytesPerRow)
```

Notes: Returns 0 on any error.
(Read only property)

6.3.15 BitmapColorSpace as CGColorSpaceMBS

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Returns the colorspace used for this bitmap.

Notes: Returns nil on any error.

This is not the same RB object used when you created the bitmap, but it will contain the same handle.
(Read only property)

6.3.16 BitmapData as MemoryBlock

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Returns a memoryblock for the data of the bitmap.

Notes: Returns nil on any error.

This is not the same RB memoryblock object as you passed to the Create function, but it will point to the same bytes in memory.
(Read only property)

6.3.17 BitmapHeight as Integer

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Returns the height of the bitmap.

Example:

```
// a new picture in RB
Var pic as new Picture(500, 500)

// and create CGContextMBS pointing to it
Var b as CGContextMBS = CGContextMBS.CreateWithPicture(pic)
```

```
// shows 500
MsgBox str(b.BitmapHeight)
```

Notes: Returns 0 on any error.
(Read only property)

6.3.18 BitmapInfo as Integer

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Obtains the bitmap information associated with a bitmap graphics context.

Example:

```
// a new picture in RB
Var pic as new Picture(500, 500)

// and create CGContextMBS pointing to it
Var b as CGContextMBS = CGContextMBS.CreateWithPicture(pic)

// shows info: 2 for kCGImageAlphaPremultipliedFirst
MsgBox str(b.BitmapInfo)
```

Notes: The bitmap info of the bitmap graphics context or 0 if c is not a bitmap graphics context. See CGImage Reference for a description of the Image Bitmap Information constants that can be returned.

The CGContextInfo data returned by the function specifies whether the bitmap contains an alpha channel and how the alpha channel is generated, along with whether the components are floating-point or integer.
(Read only property)

6.3.19 BitmapWidth as Integer

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Returns the width of the bitmap.

Example:

```
// a new picture in RB
Var pic as new Picture(500, 500)

// and create CGContextMBS pointing to it
Var b as CGContextMBS = CGContextMBS.CreateWithPicture(pic)

// show info: 500
```

MsgBox str(b.BitmapWidth)

Notes: Returns 0 on any error.
(Read only property)

6.4 class CGColorMBS

6.4.1 class CGColorMBS

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: A class for a CoreGraphics color object.

Blog Entries

- [Colorspaces in MacOS with Xojo](#)
- [MBS Real Studio Plugins, version 13.1pr16](#)
- [MBS REALbasic plug-ins version 9.5](#)

6.4.2 Methods

6.4.3 Alpha as Double

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: The alpha value of the color.

6.4.4 Black as CGColorMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The black color in the Generic gray color space.

6.4.5 Clear as CGColorMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The clear color in the Generic gray color space.

6.4.6 ColorSpace as CGColorSpaceMBS

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: The colorspace of this color.

Notes: May be nil if unkown.

6.4.7 Components as memoryblock

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: The components of the color as a memoryblock.

Notes: The memoryblock contains double properties.
m.double(0), m.double(4), etc.

6.4.8 Copy as CGColorMBS

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: Creates a copy of the color.

Notes: Returns nil on any error.

6.4.9 CopyWithAlpha(alpha as Double) as CGColorMBS

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: Makes a new copy of the color with a different alpha value.

Notes: Returns nil on any error.

6.4.10 Create(colorspace as CGColorSpaceMBS, components as memoryblock) as CGColorMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a new color with the given components.

Example:

```
Var c as color = &cFF0000
```

```
Var m As MemoryBlock
```

```
m=NewMemoryBlock(16)  
m.SingleValue(0)=c.Red/256  
m.SingleValue(4)=c.Green/256  
m.SingleValue(8)=c.Blue/256  
m.SingleValue(12)=1.0
```

```
Var colorspace as CGColorSpaceMBS
```

```
colorspace=CGColorSpaceMBS.CreateDeviceRGB
```

```

if colorspace=nil or colorspace.Handle=0 then
  MsgBox "Failed to get RGB color space!"
Return
end if

```

```

Var col As CGColorMBS

```

```

col=CGColorMBS.Create(colorspace, m)

```

```

if col=nil or col.Handle=0 then
  MsgBox "Failed to create color."
Return
end if

```

```

MsgBox str(Col.NumberOfComponents)

```

Notes: This method was called NewCGColorMBS in earlier MBS Plugins.
See also:

- 6.4.11 Create(colorspace as CGColorSpaceMBS, components() as Double) as CGColorMBS 127

6.4.11 Create(colorspace as CGColorSpaceMBS, components() as Double) as CGColorMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Creates a new color with the given components.

Notes: Supports up to 32 components.

Returns nil on any error.

See also:

- 6.4.10 Create(colorspace as CGColorSpaceMBS, components as memoryblock) as CGColorMBS 126

6.4.12 CreateDeviceCMYK(cyan as Double, magenta as Double, yellow as Double, black as Double, alpha as Double = 1.0) as CGColorMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Creates a color in the device CMYK color space.

Notes: cyan: A cyan value (0.0 - 1.0).

magenta: A magenta value (0.0 - 1.0).

yellow: A yellow value (0.0 - 1.0).

black: A black value (0.0 - 1.0).

alpha: An alpha value (0.0 - 1.0).

Returns a color object.

6.4.13 CreateDeviceGray(gray as Double, alpha as Double = 1.0) as CGColorMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Creates a color in the device gray color space.

Notes: gray: A grayscale value (0.0 - 1.0).

alpha: An alpha value (0.0 - 1.0).

Returns a color object.

6.4.14 CreateDeviceRGB(red as Double, green as Double, blue as Double, alpha as Double = 1.0) as CGColorMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Creates a color in the device RGB color space.

Notes: red: A red component value (0.0 - 1.0).

green: A green component value (0.0 - 1.0).

blue: A blue component value (0.0 - 1.0).

alpha: An alpha value (0.0 - 1.0).

Returns a new color object.

6.4.15 CreateGenericCMYK(cyan as Double, magenta as Double, yellow as Double, black as Double, alpha as Double = 1.0) as CGColorMBS

Plugin Version: 9.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a color in the Generic CMYK color space.

Notes: cyan: A cyan value (0.0 - 1.0).

magenta: A magenta value (0.0 - 1.0).

yellow: A yellow value (0.0 - 1.0).

black: A black value (0.0 - 1.0).

alpha: An alpha value (0.0 - 1.0).

Returns a color object.

Available in Mac OS X v10.5 and later.

6.4.16 CreateGenericGray(gray as Double, alpha as Double = 1.0) as CGColorMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a color in the Generic gray color space.

Notes: gray: A grayscale value (0.0 - 1.0).

alpha: An alpha value (0.0 - 1.0).

Returns a color object.

Available in Mac OS X v10.5 and later.

6.4.17 CreateGenericRGB(red as Double, green as Double, blue as Double, alpha as Double = 1.0) as CGColorMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a color in the Generic RGB color space.

Notes: red: A red component value (0.0 - 1.0).

green: A green component value (0.0 - 1.0).

blue: A blue component value (0.0 - 1.0).

alpha: An alpha value (0.0 - 1.0).

Returns a new color object.

Available in Mac OS X v10.5 and later.

6.4.18 Equal(secondColor as CGColorMBS) as boolean

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: True if both colors have equal values.

Notes: False on any error.

6.4.19 NumberOfComponents as Integer

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: The number of components.

Notes: Should be 3 for RGB and 4 for CMYK.

6.4.20 White as CGColorMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The white color in the Generic gray color space.

6.4.21 Properties

6.4.22 Handle as Integer

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: The handle to the CGColorRef used internal.

Notes: (Read and Write property)

6.5 class CGColorSpaceMBS

6.5.1 class CGColorSpaceMBS

Plugin Version: 3.3, Platform: macOS, Targets: All.

Function: A class for a CoreGraphics color space.

Blog Entries

- [MBS Xojo Plugins, version 19.3pr5](#)
- [Colorspaces in Xojo](#)
- [Colorspaces in MacOS with Xojo](#)
- [MBS Xojo / Real Studio Plugins, version 15.4pr7](#)
- [Tip of day: Load ICC profile on OS X for any image format with CGImageSource](#)
- [MBS Xojo / Real Studio Plugins, version 15.0pr5](#)
- [MBS Xojo / Real Studio plug-ins in version 13.4](#)
- [Notes from this week](#)
- [MBS REALbasic Plugins Version 10.4 release notes](#)
- [MBS REALbasic plug-ins version 9.5](#)

6.5.2 Methods

6.5.3 CreateCalibratedGray(whitePoint() as Double, blackPoint() as Double, gamma as Double) as CGColorSpaceMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a calibrated gray colorspace.

Notes: WhitePoint is an array of 3 numbers (type double) specifying the tristimulus value, in the CIE 1931 XYZ-space, of the diffuse white point.

BlackPoint is an array of 3 numbers (type double) specifying the tristimulus value, in CIE 1931 XYZ-space, of the diffuse black point.

Gamma defines the gamma for the gray component.

Returns nil on any error.

6.5.4 CreateCalibratedRGB(whitePoint() as Double, blackPoint() as Double, gamma() as Double, matrix() as Double) as CGColorSpaceMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a calibrated RGB colorspace.

Notes: WhitePoint is an array of 3 numbers (type double) specifying the tristimulus value, in the CIE 1931 XYZ-space, of the diffuse white point.

BlackPoint is an array of 3 numbers (type double) specifying the tristimulus value, in CIE 1931 XYZ-space, of the diffuse black point.

Gamma is an array of 3 numbers (type double) specifying the gamma for the red, green, and blue components of the color space.

Matrix is an array of 9 numbers (type double) specifying the linear interpretation of the gamma-modified RGB values of the colorspace with respect to the final XYZ representation.

Returns nil on any error.

6.5.5 CreateDeviceCMYK as CGColorSpaceMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a DeviceCMYK colorspace.

Notes: Returns nil on any error.

6.5.6 CreateDeviceGray as CGColorSpaceMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a DeviceGray colorspace.

Notes: Returns nil on any error.

6.5.7 CreateDeviceRGB as CGColorSpaceMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a DeviceRGB colorspace.

Notes: Returns nil on any error.

Old name: CGColorSpaceCreateDeviceRGBMBS

6.5.8 CreateLab(whitePoint() as Double, blackPoint() as Double, range() as Double) as CGColorSpaceMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates an L*a*b* colorspace.

Notes: WhitePoint is an array of 3 numbers (type double) specifying the tristimulus value, in the CIE 1931

XYZ-space, of the diffuse white point.

BlackPoint is an array of 3 numbers (type double) specifying the tristimulus value, in CIE 1931 XYZ-space, of the diffuse black point.

Range is an array of four numbers (type double) specifying the range of valid values for the a* and b* components of the color space.

Returns nil on any error.

6.5.9 CreatePattern(baseSpace as CGColorSpaceMBS) as CGColorSpaceMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a pattern colorspace.

Notes: BaseSpace is the underlying colorspace of the pattern colorspace. For colored patterns, baseSpace should be nil; for uncolored patterns, baseSpace specifies the colorspace of colors which will be painted through the pattern.

Returns nil on any error.

6.5.10 CreateWithHandle(Handle as Integer) as CGColorSpaceMBS

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: Creates a CGColorSpace for a handle.

Example:

```
Var n as NSColorSpaceMBS = NSColorSpaceMBS.genericCMYKColorSpace
Var c as CGColorSpaceMBS = CGColorSpaceMBS.CreateWithHandle(n.CGColorSpaceHandle)
MsgBox c.Name
```

Notes: Sometimes you need to create CGColorSpaceMBS from NSColorSpaceMBS or some handle you got from an OS function and then you can use this function.

Returns nil on any error. Retains the handle.

6.5.11 CreateWithICCProfile(ICCProfileData as memoryblock) as CGColorSpaceMBS

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Creates an ICC-based color space using the ICC profile contained in the specified data.

Notes: data: The data containing the ICC profile to set for the new color space.

Returns a new color space based on the specified profile.

See also:

- 6.5.12 `CreateWithICCProfile(ICCProfileData as string)` as `CGColorSpaceMBS` 134

6.5.12 `CreateWithICCProfile(ICCProfileData as string)` as `CGColorSpaceMBS`

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Creates an ICC-based color space using the ICC profile contained in the specified data.

Notes: data: The data containing the ICC profile to set for the new color space.

Returns a new color space based on the specified profile.

See also:

- 6.5.11 `CreateWithICCProfile(ICCProfileData as memoryblock)` as `CGColorSpaceMBS` 133

6.5.13 `CreateWithName(name as string)` as `CGColorSpaceMBS`

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Create a color space using name as the identifier for the color space.

Notes: Pass one of the `kCGColorSpace*` string constants.

6.5.14 `CreateWithPlatformColorSpace(Handle as Integer)` as `CGColorSpaceMBS`

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a `CGColorSpace` using a Colorsync Profile Handle.

Notes: Returns nil on any error.

6.5.15 `ICCProfile as string`

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns a copy of the ICC profile of the provided color space.

Notes: The ICC profile or "" if the color space does not have an ICC profile.

Available in Mac OS X v10.5 and later.

6.5.16 `kCGColorSpaceACESCGLinear as string`

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: One of the names for color spaces.

Notes: The name of the ACEScg color space. For more information, see „ÁúACEScg —A Working Space for CGI Render and Compositing,Àù, Version 1.0.1, Academy of Motion Picture Arts and Sciences (<http://www.oscars.org/science-technology/sci-tech-projects/aces>).

Available in OS X v10.11 and later.

6.5.17 kCGColorSpaceAdobeRGB1998 as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for color spaces.

Notes: The name of the Adobe RGB (1998) color space. For more information, see "Adobe RGB (1998) Color Image Encoding", Version 2005-05, Adobe Systems Inc. (<http://www.adobe.com>).

6.5.18 kCGColorSpaceDCIP3 as string

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: One of the names for color spaces.

Notes: The name of the DCI P3 color space, created by Digital Cinema Initiatives, LLC. This color space is the digital cinema standard.

Available in OS X v10.11 and later.

6.5.19 kCGColorSpaceDisplayP3 as string

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: One of the names for color spaces.

Notes: The name of the Display P3 color space, created by Apple Inc. This color space uses the DCI P3 primaries, a D65 white point, and the same gamma curve as the sRGB color space.

Available in OS X v10.10 and later.

6.5.20 kCGColorSpaceGenericCMYK as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for color spaces.

Notes: The name of the "Generic" CMYK color space.

6.5.21 `kCGColorSpaceGenericGray` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for color spaces.

Notes: The name of the "Generic" gray color space.

6.5.22 `kCGColorSpaceGenericGrayGamma2_2` as string

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: One of the names for color spaces.

Notes: The name of the generic gray color space with a gamma value of 2.2.

Available in OS X v10.6 and later.

6.5.23 `kCGColorSpaceGenericRGB` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for color spaces.

Notes: The name of the "Generic" RGB color space.

6.5.24 `kCGColorSpaceGenericRGBLinear` as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for color spaces.

Notes: The name of the "Generic" linear RGB color space. This is the same as `kCGColorSpaceGenericRGB` but with a 1.0 gamma.

6.5.25 `kCGColorSpaceGenericXYZ` as string

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: One of the names for color spaces.

Notes: The name of the generic XYZ color space.

Available in OS X v10.11 and later.

6.5.26 kCGColorSpaceITUR_2020 as string

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: One of the names for color spaces.

Notes: The name of the ITU-R Recommendation BT.2020 color space. For more information, see ,ÁúBT.2020 : Parameter values for ultra-high definition television systems for production and international programme exchange,Áù, Version 2014-06, International Telecommunication Union (<https://www.itu.int/rec/R-REC-BT.2020>).

Available in OS X v10.11 and later.

6.5.27 kCGColorSpaceITUR_709 as string

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: One of the names for color spaces.

Notes: The name of the ITU-R Recommendation BT.2020 color space. For more information, see ,ÁúBT.2020 : Parameter values for ultra-high definition television systems for production and international programme exchange,Áù, Version 2014-06, International Telecommunication Union (<https://www.itu.int/rec/R-REC-BT.2020>).

Available in OS X v10.11 and later.

6.5.28 kCGColorSpaceROMMRGB as string

Plugin Version: 16.0, Platform: macOS, Targets: All.

Function: One of the names for color spaces.

Notes: The name of the ROMM RGB color space. For more information, see ,ÁúReference Output Medium Metric RGB Color Space (ROMM RGB) White Paper,Áù, Version 2.1, Eastman Kodak Company (http://www.kodak.com/global/plugins/acrobat/en/professional/products/software/colorFlow/romm_rgb.pdf).

Available in OS X v10.11 and later.

6.5.29 kCGColorSpaceSRGB as string

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: One of the names for color spaces.

Notes: The name of the sRGB color space. The capitalization in the name, while strictly inaccurate, avoids interpretational ambiguity. For more information, see IEC 61966-2-1 (1999-10): "Multimedia systems and equipment - Colour measurement and management - Part 2-1: Colour management - Default RGB colour space - sRGB".

6.5.30 Properties

6.5.31 BaseColorSpace as CGColorSpaceMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns the base color space of a pattern or indexed color space.

Notes: The base color space if the space parameter is a pattern or indexed color space; otherwise, nil.

Available in Mac OS X v10.5 and later.

(Read only property)

6.5.32 ColorTableCount as Integer

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns the number of entries in the color table of an indexed color space.

Notes: The number of entries in the color table of the space parameter if the color space is an indexed color space; otherwise, returns 0.

Available in Mac OS X v10.5 and later.

(Read only property)

6.5.33 Description as String

Plugin Version: 15.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The description text.

Example:

```
Var c as CGColorSpaceMBS = window1.CGColorSpaceMBS
MsgBox c.description
```

Notes: (Read only property)

6.5.34 Handle as Integer

Plugin Version: 3.3, Platform: macOS, Targets: All.

Function: The handle to the color space.

Notes: (a CGColorSpaceRef)

(Read and Write property)

6.5.35 Model as Integer

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns the color space model of the provided color space.

Notes: Available in Mac OS X v10.5 and later.

Use this constants:

CGColorSpaceMBS.kCGColorSpaceModelUnknown

CGColorSpaceMBS.kCGColorSpaceModelCMYK

CGColorSpaceMBS.kCGColorSpaceModelDeviceN

CGColorSpaceMBS.kCGColorSpaceModelIndexed

CGColorSpaceMBS.kCGColorSpaceModelLab

CGColorSpaceMBS.kCGColorSpaceModelMonochrome

CGColorSpaceMBS.kCGColorSpaceModelPattern

CGColorSpaceMBS.kCGColorSpaceModelRGB

(Read only property)

6.5.36 ModelText as String

Plugin Version: 15.4, Platform: macOS, Targets: All.

Function: The model as text.

Notes: For viewing in debugger.

(Read only property)

6.5.37 Name as String

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Queries the name of the colorspace.

Notes: Not all CGColorspace objects have names. Some have only ICC Profile and you can get name via Name property in LCMS2ProfileMBS if you open the profile with LCMS.

(Read only property)

6.5.38 NumberOfComponents as Integer

Plugin Version: 3.3, Platform: macOS, Targets: All.

Function: Returns the number of components.

Notes: typical 1 for grayscale, 3 for RGB and 4 for CMYK.

(Read only property)

6.5.39 Constants

Constants

Constant	Value	Description
kCGColorSpaceModelCMYK	2	One of the colorspace model constants. A CMYK color space model. Available in Mac OS X v10.5 and later.
kCGColorSpaceModelDeviceN	4	One of the colorspace model constants. A DeviceN color space model. Available in Mac OS X v10.5 and later.
kCGColorSpaceModelIndexed	5	One of the colorspace model constants. An indexed color space model. Available in Mac OS X v10.5 and later.
kCGColorSpaceModelLab	3	One of the colorspace model constants. A Lab color space model. Available in Mac OS X v10.5 and later.
kCGColorSpaceModelMonochrome	0	One of the colorspace model constants. A monochrome color space model. Available in Mac OS X v10.5 and later.
kCGColorSpaceModelPattern	6	One of the colorspace model constants. A pattern color space model. Available in Mac OS X v10.5 and later.
kCGColorSpaceModelRGB	1	One of the colorspace model constants. An RGB color space model. Available in Mac OS X v10.5 and later.
kCGColorSpaceModelUnknown	-1	One of the colorspace model constants. An unknown color space model. Available in Mac OS X v10.5 and later.
kCGRenderingIntentAbsoluteColorimetric	1	One of the color rendering intents constants. Map colors outside of the gamut of the output device to the closest match inside the gamut of the output device. This can produce a perceptual effect, where two different color values in the gamut of the graphics context are mapped to the same color value in the output device's gamut. Unlike relative colorimetric, absolute colorimetric does not modify colors inside the gamut of the output device.
kCGRenderingIntentDefault	0	One of the color rendering intents constants. The default rendering intent for the graphics context.
kCGRenderingIntentPerceptual	3	One of the color rendering intents constants. Preserve the visual relationship between colors by compressing the gamut of the graphics context to fit inside the gamut of the output device. Perceptual intent is good for photographs and other complex, detailed images.
kCGRenderingIntentRelativeColorimetric	2	One of the color rendering intents constants. Map colors outside of the gamut of the output device to the closest match inside the gamut of the output device. This can produce a perceptual effect, where two different color values in the gamut of the graphics context are mapped to the same color value in the output device's gamut. Unlike relative colorimetric, relative colorimetric shifts all colors (including those within the gamut) to account for the difference between the white point of the graphics context and the white point of the output device.
kCGRenderingIntentSaturation	4	One of the color rendering intents constants. Preserve the relative saturation value of the colors when converting to the gamut of the output device. The result is an image with bright, saturated colors. Saturation intent is good for reproducing images with low detail, such as presentation charts and graphs.

6.6 class CGContextMBS

6.6.1 class CGContextMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: A class for a core graphics context.

Example:

```

Var p As New Picture(200, 200)
Var g As Graphics = p.Graphics

// Clear image with white
g.ForeColor = &cFFFFFF
g.FillRect 0, 0, g.Width, g.Height

Var c As New CGContextMBS(g)

c.SetRGBFillColor(1, 0, 0)

// coordinates are swapped in CGContext, so y = 20 means 20 pixels from bottom of picture!

Const x = 10
Const y = 20
Const w = 120
Const h = 100

c.FillRect New CGRectMBS(x, y, w, h)

Self.Backdrop = p

```

Notes: If the handle property has a non zero value, the destructor of this class will release the context reference.

If you create this in a method based on a window, please release it within that method. e.g. in a paint event, create it and let RB delete the last reference on the end of the method. Else you may see crashes as the context is still being around while the graphics port has been released.

Blog Entries

- [Several ways for picture to PDF in MBS Plugins](#)
- [Drawing with DirectDraw in Xojo](#)
- [MBS Xojo Plugins, version 20.3pr3](#)
- [MBS Xojo Plugins, version 19.6pr3](#)
- [Removing 32-bit Carbon GUI classes](#)

- [MBS Xojo Plugins, version 18.4pr1](#)
- [Colorspaces in MacOS with Xojo](#)
- [MBS Xojo / Real Studio Plugins, version 14.2pr8](#)
- [MBS Plugins 11.1 Release notes](#)
- [MonkeyBread Software Releases the MBS REALbasic plug-ins 8.4](#)

Xojo Developer Magazine

- [6.6, page 8: News](#)
- [6.4, pages 30 to 31: Creating PDF Files, How to create PDFs using the MBS Plugins by Christian Schmitz](#)

6.6.2 Methods

6.6.3 AddArc(x as Double, y as Double, radius as Double, startangle as Double, endangle as Double, clockwise as boolean)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Add an arc to the current path.

Example:

```
const r=100.0
Var c as CGContextMBS
// draws a circle with radius r
c.BeginPath
c.SetLineWidth 5
c.SetGrayStrokeColor 0,1
c.AddArc 250,150,r,0,360,false
c.StrokePath
```

Notes: Add an arc of a circle to the context's path, possibly preceded by a straight line segment. '(x, y)' is the center of the arc; 'radius' is its radius; 'startAngle' is the angle to the first endpoint of the arc; 'endAngle' is the angle to the second endpoint of the arc; and 'clockwise' is true if the arc is to be drawn clockwise, false otherwise. 'startAngle' and 'endAngle' are measured in radians.

6.6.4 addArcToPath(x as Double, y as Double, w as Double, h as Double, startAngle as Integer, arcAngle as Integer)

Plugin Version: 8.5, Platform: macOS, Targets: All.

Function: Adds an arc to the current path.

6.6.5 AddArcToPoint(x1 as Double, y1 as Double, x2 as Double, y2 as Double, radius as Double)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Add an arc to the current path.

Notes: Add an arc of a circle to the context's path, possibly preceded by a straight line segment. 'radius' is the radius of the arc. The arc is tangent to the line from the current point to '(x1, y1)', and the line from '(x1, y1)' to '(x2, y2)'.

6.6.6 AddCurveToPoint(cp1x as Double, cp1y as Double, cp2x as Double, cp2y as Double, x as Double, y as Double)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Append a cubic Bezier curve from the current point to (x,y), with control points (cp1x, cp1y) and (cp2x, cp2y)

6.6.7 AddEllipseInRect(r as CGRectMBS)

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Add an ellipse inside rect to the current path of context.

Notes: See the function CGContextMBS.AddEllipseInRect for more information on how the path for the ellipse is constructed.

Requires Mac OS X 10.4.

6.6.8 AddLines(p() as CGPointMBS)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Add a set of lines to the context's path.

Notes: Currently this function is not available to RB versions before 3.5.

Note that the p parameter is an array of CGPointMBS and not just one.

6.6.9 AddLineToPoint(x as Double, y as Double)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Append a straight line segment from the current point to (x, y).

Example:

```
Var c as CGContextMBS

c=window1.CGContextMBS

c.SetRGBStrokeColor 1,0,0,1
c.BeginPath
c.MoveToPoint 0,0
c.AddLineToPoint 100,100
c.StrokePath
c.Flush
```

6.6.10 addOvalToPath(x as Double, y as Double, w as Double, h as Double)

Plugin Version: 8.5, Platform: macOS, Targets: All.

Function: Adds an oval to the current path.

6.6.11 AddPath(path as CGPathMBS)

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Add path to the path of context.

Notes: The points in path are transformed by the CTM of context before they are added.

Requires Mac OS X 10.2.

6.6.12 AddQuadCurveToPoint(cpx as Double, cpy as Double, x as Double, y as Double)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Append a quadratic curve from the current point to (x, y), with control point (cpx, cpy).

6.6.13 AddRect(r as CGRectMBS)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Add a double rect to the context's path.

6.6.14 AddRects(r() as CGRectMBS)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Add a set of rects to the context's path.

Notes: Currently this function is not available to RB versions before 3.5.

Note that the r parameter is an array of CGRectMBS and not just one.

6.6.15 addRoundedRectToPath(x as Double, y as Double, w as Double, h as Double, arcWidth as Double, arcHeight as Double)

Plugin Version: 8.5, Platform: macOS, Targets: All.

Function: Adds a round rectangle to the current path.

6.6.16 BeginPage(mediabox as CGRectMBS)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Begin a new page.

6.6.17 BeginPath

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Begin a new path. The old path is discarded.

Example:

```
#If XojoVersion >= 2019.02 Then
Var p As ptr = g.Handle(Graphics.HandleTypes.CGContextRef)
Var c As CGContextMBS = CGContextMBS.contextWithCGContext(p)
#Else
Var h As Integer = g.Handle(g.HandleTypeCGContextRef)
Var c As CGContextMBS = CGContextMBS.contextWithCGContext(h)
#EndIf
```

```
c.SetGrayStrokeColor(0,1)
```

```
c.SetGrayFillColor(0,1)
```

```
c.BeginPath
```

```
c.SetLineWidth 0.5
```

```

c.MoveToPoint 50, 550
c.AddLineToPoint 100, 600
c.StrokePath
c.Flush
c = nil

```

Notes: Note that a context has a double path in use at any time: a path is not part of the graphics state.

6.6.18 BeginTransparencyLayer(auxiliaryInfo as Dictionary = nil)

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Begin a transparency layer in context.

Notes: All subsequent drawing operations until a corresponding EndTransparencyLayer are composited into a fully transparent backdrop (which is treated as a separate destination buffer from the context). After the transparency layer is ended, the result is composited into the context using the global alpha and shadow state of the context. This operation respects the clipping region of the context. After a call to this function, all of the parameters in the graphics state remain unchanged with the exception of the following:

- The global alpha is set to 1.
- The shadow is turned off.
- The blend mode is set to 'kCGBlendModeNormal'.

Ending the transparency layer restores these parameters to the values they had before BeginTransparencyLayer was called. Transparency layers may be nested.

6.6.19 BeginTransparencyLayerWithRect(r as CGRectMBS, auxiliaryInfo as Dictionary = nil)

Plugin Version: 12.1, Platform: macOS, Targets: All.

Function: Begin a transparency layer in context.

Notes: This function is identical to BeginTransparencyLayer except that the content of the transparency layer will be bounded by rect (specified in user space).

6.6.20 clearRect(rect as CGRectMBS)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Clears the background.

Notes: Remember that in CoreGraphics the position 0/0 is in the bottom left corner. In Xojo 0/0 is in the

top left corner.

6.6.21 clip

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Clips the current path.

Notes: Intersect the context's path with the current clip path and use the resulting path as the clip path for subsequent rendering operations. Use the winding-number fill rule for deciding what's inside the path.

6.6.22 ClipToMask(rect as CGRectMBS, mask as CGImageMBS)

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Maps a mask into the specified rectangle and intersects it with the current clipping area of the graphics context.

Notes: rect: The rectangle to map the mask parameter to.

mask: An image or an image mask. If mask is an image, then it must be in the DeviceGray color space, may not have an alpha component, and may not be masked by an image mask or masking color.

If the mask parameter is an image mask, then Quartz clips in a manner identical to the behavior seen with the function DrawImage—the mask indicates an area to be left unchanged when drawing. The source samples of the image mask determine which points of the clipping area are changed, acting as an "inverse alpha" value. If the value of a source sample in the image mask is S , then the corresponding point in the current clipping area is multiplied by an alpha value of $(1-S)$. For example, if S is 1 then the point in the clipping area becomes transparent. If S is 0, the point in the clipping area is unchanged.

If the mask parameter is an image, then mask acts like an alpha mask and is blended with the current clipping area. The source samples of mask determine which points of the clipping area are changed. If the value of the source sample in mask is S , then the corresponding point in the current clipping area is multiplied by an alpha of S . For example, if S is 0, then the point in the clipping area becomes transparent. If S is 1, the point in the clipping area is unchanged.

Available in Mac OS X v10.4 and later.

6.6.23 clipToRect(rect as CGRectMBS)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Clips the current path.

Notes: Intersect the current clipping path with 'rect'. Note that this function resets the context's path to

the empty path.

6.6.24 Close

Plugin Version: 15.2, Platform: macOS, Targets: All.

Function: Closes the context.

Notes: Same as destructor later, but running now when you call method.

6.6.25 closePath

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Close the current subpath of the context's path.

6.6.26 ConcatCTM(transform as CGAffineTransformMBS)

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: Concatenate the current graphics state's transformation matrix (the CTM) with the affine transform 'transform'.

6.6.27 Constructor(g as graphics)

Plugin Version: 20.3, Platform: macOS, Targets: All.

Function: Creates a new CGContextMBS object drawing to the existing graphics object.

Example:

```
Var p As New Picture(200, 200)
Var g As Graphics = p.Graphics
```

```
// Clear image with white
g.ForeColor = &cFFFFFF
g.FillRect 0, 0, g.Width, g.Height
```

```
Var c As New CGContextMBS(g)
```

```
c.SetRGBFillColor(1, 0, 0)
```

```
// coordinates are swapped in CGContext, so y = 20 means 20 pixels from bottom of picture!
```

```

Const x = 10
Const y = 20
Const w = 120
Const h = 100

```

```
c.FillRect New CGRectMBS(x, y, w, h)
```

```
Self.Backdrop = p
```

Notes: Raises exception if we fail to get the CGContext from Xojo.
Works on macOS and iOS.

See also:

- 6.6.28 Constructor(handle as Integer) 150
- 6.6.29 Constructor(pic as Picture) 150

6.6.28 Constructor(handle as Integer)

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Creates a new CGContextMBS object based on a CGContextRef.

Notes: The CGContext is retained.

See also:

- 6.6.27 Constructor(g as graphics) 149
- 6.6.29 Constructor(pic as Picture) 150

6.6.29 Constructor(pic as Picture)

Plugin Version: 20.5, Platform: macOS, Targets: All.

Function: Creates a new CGContextMBS object drawing to the existing Picture object.

Example:

```

Var p As New Picture(100,100)
Var pg As New CGContextMBS(p)
break // inspect in debugger

```

Notes: Raises exception, if the picture is not a bitmap picture.
Works on macOS and iOS.

See also:

- 6.6.27 Constructor(g as graphics) 149
- 6.6.28 Constructor(handle as Integer) 150

6.6.30 contextWithCGContext(handle as Integer) as CGContextMBS

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Creates a new CGContextMBS object based on a CGContextRef.

Example:

```
// load a picture
Var f as FolderItem = SpecialFolder.Desktop.Child("ColorSpin.jpg")
Var pic as Picture = picture.Open(f)

// open printer
Var g as Graphics = OpenPrinterDialog
if g = nil then Return

// draw
g.DrawPicture pic, 0, 0

// now load again
Var ImageSource as new CGImageSourceMBS(f)
Var img as CGImageMBS = ImageSource.CreateImageAtIndex(0)
Var cs as CGColorSpaceMBS = CGColorSpaceMBS.CreateDeviceRGB

// copy with replacing colorspace
img = img.CopyWithColorSpace(cs)

// and draw
#If XojoVersion >= 2019.02 Then
Var p As ptr = g.Handle(Graphics.HandleTypes.CGContextRef)
Var c As CGContextMBS = CGContextMBS.contextWithCGContext(p)
#Else
Var h As Integer = g.Handle(g.HandleTypeCGContextRef)
Var c As CGContextMBS = CGContextMBS.contextWithCGContext(h)
#EndIf
Var r as CRectMBS = CGMakeRectMBS(0, 0, img.Width, img.Height)

c.DrawPicture(img, r)
c.Flush
```

Notes: The CGContext is retained.

Returns nil on any error.

See also:

- 6.6.31 contextWithCGContext(handle as Ptr) as CGContextMBS

6.6.31 contextWithCGContext(handle as Ptr) as CGContextMBS

Plugin Version: 20.0, Platform: macOS, Targets: All.

Function: Creates a new CGContextMBS object based on a CGContextRef.

Example:

```
// load a picture
Var f as FolderItem = SpecialFolder.Desktop.Child("ColorSpin.jpg")
Var pic as Picture = picture.Open(f)

// open printer
Var g as Graphics = OpenPrinterDialog
if g = nil then Return

// draw
g.DrawPicture pic, 0, 0

// now load again
Var ImageSource as new CGImageSourceMBS(f)
Var img as CGImageMBS = ImageSource.CreateImageAtIndex(0)
Var cs as CGColorSpaceMBS = CGColorSpaceMBS.CreateDeviceRGB

// copy with replacing colorspace
img = img.CopyWithColorSpace(cs)

// and draw
#If XojoVersion >= 2019.02 Then
Var p As ptr = g.Handle(Graphics.HandleTypes.CGContextRef)
Var c As CGContextMBS = CGContextMBS.contextWithCGContext(p)
#Else
Var h As Integer = g.Handle(g.HandleTypeCGContextRef)
Var c As CGContextMBS = CGContextMBS.contextWithCGContext(h)
#EndIf
Var r as CGRectMBS = CGMakeRectMBS(0, 0, img.Width, img.Height)

c.DrawPicture(img, r)
c.Flush
```

Notes: The CGContext is retained.

Returns nil on any error.

See also:

- 6.6.30 contextWithCGContext(handle as Integer) as CGContextMBS

6.6.32 CopyPath as CGPathMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Returns a Quartz path object built from the current path information in a graphics context.

Notes: Available in Mac OS X v10.6 and later.

6.6.33 DrawCGPDFDocument(pdf as Variant, rect as CGRectMBS, page as Integer)

Plugin Version: 2.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: Draw 'page' in 'document' in the rectangular area specified by 'rect'.

Example:

```
'get a print session

// print this PDF
Var pathPrinted as FolderItem=GetFolderItem("test.pdf")

Var thePrintSession as CPMPrintSessionMBS = NewCPMPrintSessionMBS
if thePrintSession = nil then Return

'get default page format and print settings and attach it to the print settings
Var thePageFormat as CPMPageFormatMBS = NewCPMPageFormatMBS
Var thePrintSettings as CPMPrintSettingsMBS = NewCPMPrintSettingsMBS
thePrintSession.DefaultPageFormat thePageFormat
thePrintSession.DefaultPrintSettings thePrintSettings

'show the print dialog
if not thePrintSession.PrintDialog(thePrintSettings,thePageFormat) then return

'open the file which will be printed
Var thePdfDocument as CGPDFDocumentMBS = pathPrinted.OpenAsCGPDFDocumentMBS

' limit page counts to the one we have
Var LastPage as Integer = thePdfDocument.PageCount
if thePrintSettings.LastPage<lastpage then
lastpage=thePrintSettings.LastPage
end if

' you get better progress bar if you tell how many pages will come
thePrintSettings.LastPage=lastpage

'begin the printing
thePrintSession.BeginDocument(thePrintSettings, thePageFormat)
```

```

'loop over the number of copies
for currentCopy as Integer = 1 to thePrintSettings.Copies

'loop over the pages
for currentPage as Integer = thePrintSettings.FirstPage to LastPage

'prepage the page
Var PrintRect as CPMRectMBS =thePageFormat.AdjustedPageSize
Var CGRect as CGRectMBS =CGMakeRectMBS(PrintRect.left, PrintRect.top, PrintRect.Width, Print-
Rect.Height)
thePrintSession.BeginPage(thePageFormat, nil)
Var thePrintContext as CGContextMBS = thePrintSession.PageContext
if thePrintContext = Nil then return

'print the page
thePrintContext.DrawCGPDFDocument thePdfDocument, CGRect, currentPage

'end the page
thePrintContext = nil
thePrintSession.EndPage
next

next

'end the printing
thePrintSession.EndDocument

```

Notes: Pass a CGPDFDocumentMBS object for the pdf argument.
The media box of the page is scaled, if necessary, to fit into 'rect'.

6.6.34 DrawLayerAtPoint(Point as CGPointMBS, layer as CGLayerMBS)

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: Draws the contents of a CGLayer object at the specified point.

Notes: context: The graphics context associated with the layer.

point: The location, in current user space coordinates, to use as the origin for the drawing.

layer: The layer whose contents you want to draw.

Calling the function DrawLayerAtPoint is equivalent to calling the function DrawLayerInRect with a rectangle that has its origin at point and its size equal to the size of the layer.

Available in Mac OS X version 10.4 and later.

6.6.35 DrawLayerInRect(rect as CGRectMBS, layer as CGLayerMBS)

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: Draws the contents of a CGLayer object into the specified rectangle.

Notes: context: The graphics context associated with the layer.

rect: The rectangle, in current user space coordinates, to draw to.

layer: The layer whose contents you want to draw.

The contents are scaled, if necessary, to fit into the rectangle.

Available in Mac OS X version 10.4 and later.

6.6.36 DrawLinearGradient(gradient as CGGradientMBS, startPoint as CGPointMBS, endPoint as CGPointMBS, options as Integer)

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Paints a gradient fill that varies along the line defined by the provided starting and ending points.

Notes: gradient: A CGGradient object.

startPoint: The coordinate that defines the starting point of the gradient.

endPoint: The coordinate that defines the ending point of the gradient.

options: Option flags (kCGGradientDrawsBeforeStartLocation or kCGGradientDrawsAfterEndLocation) that control whether the fill is extended beyond the starting or ending point.

The color at location 0 in the CGGradient object is mapped to the starting point. The color at location 1 in the CGGradient object is mapped to the ending point. Colors are linearly interpolated between these two points based on the location values of the gradient. The option flags control whether the gradient is drawn before the start point or after the end point.

Available in Mac OS X v10.5 and later.

6.6.37 DrawPath(mode as Integer)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Draw the context's path using drawing mode 'mode'.

6.6.38 DrawPicture(pic as CGImageMBS, rect as CGRectMBS)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Draws a CGImageMBS at the given position.

Example:

```
// put inside window paint event

Var c as CGContextMBS
if TargetCocoa then
c = GetCurrentCGContextMBS
else
c = window1.CGContextMBS
end if

Var logo as Picture = logoMBS(500)
Var image as CGImageMBS = CGCreateImageMBS(logo)

Var r as CGRectMBS = CGMakeRectMBS(0,0,g.Width,g.Height)

c.DrawPicture image, r
```

Notes: Remember that in CoreGraphics the position 0/0 is in the bottom left corner. In Xojo 0/0 is in the top left corner.

6.6.39 DrawRadialGradient(**gradient** as CGGradientMBS, **startCenter** as CGPointMBS, **startRadius** as Double, **endCenter** as CGPointMBS, **endRadius** as Double, **options** as Integer)

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Paints a gradient fill that varies along the area defined by the provided starting and ending circles.

Notes: gradient: A CGGradient object.

startCenter: The coordinate that defines the center of the starting circle.

startRadius: The radius of the starting circle.

endCenter: The coordinate that defines the center of the ending circle.

endRadius: The radius of the ending circle.

options: Option flags (kCGGradientDrawsBeforeStartLocation or kCGGradientDrawsAfterEndLocation) that control whether the gradient is drawn before the starting circle or after the ending circle.

The color at location 0 in the CGGradient object is mapped to the circle defined by startCenter and startRadius. The color at location 1 in the CGGradient object is mapped to the circle defined by endCenter and endRadius. Colors are linearly interpolated between the starting and ending circles based on the location values of the gradient. The option flags control whether the gradient is drawn before the start point or after the end point.

Available in Mac OS X v10.5 and later.

6.6.40 DrawShading(shading as CGShadingMBS)

Plugin Version: 6.2, Platform: macOS, Targets: All.

Function: Fills the clipping path of a context with the specified shading.

Notes: shading: A Quartz shading. Quartz retains this object; upon return, you may safely release it. Available in Mac OS X version 10.2 and later.

6.6.41 DrawTiledImage(pic as CGImageMBS, rect as CGRectMBS)

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Repeatedly draws an image, scaled to the provided rectangle, to fill the current clip region.

Example:

```
// put inside window paint event

Var c as CGContextMBS
if TargetCocoa then
c = GetCurrentCGContextMBS
else
c = window1.CGContextMBS
end if

Var logo as Picture = logoMBS(50)
Var image as CGImageMBS = CGCreateImageMBS(logo)

Var r as CGRectMBS = CGMakeRectMBS(0,0,50,50)

c.DrawTiledImage image, r
```

Notes: rect: A rectangle that specifies the origin and size of the destination tile. Quartz scales the image—disproportionately, if necessary—to fit the bounds specified by the rect parameter.

image: The image to draw.

Quartz draws the scaled image starting at the origin of the rectangle in user space, then moves to a new point (horizontally by the width of the tile and/or vertically by the height of the tile), draws the scaled image, moves again, draws again, and so on, until the current clip region is tiled with copies of the image. Unlike patterns, the image is tiled in user space, so transformations applied to the CTM affect the final result.

Available in Mac OS X v10.5 and later.

6.6.42 EndPage

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: End the current page.

6.6.43 EndTransparencyLayer

Plugin Version: 11.1, Platform: macOS, Targets: All.

Function: End a transparency layer.

6.6.44 EOClip

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Clips the current path.

Notes: Intersect the context's path with the current clip path and use the resulting path as the clip path for subsequent rendering operations. Use the even-odd fill rule for deciding what's inside the path.

6.6.45 EOFillPath

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Fill the context's path using the even-odd fill rule. Any open subpath of the path is implicitly closed.

6.6.46 FillEllipseInRect(rect as CGRectMBS)

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Fill an ellipse (an oval) inside rect.

Example:

```
// a new picture in RB
Var pic as new Picture(500, 500)

// and create CGContextMBS pointing to it
```

```

Var b as CGContextMBS = CGContextMBS.CreateWithPicture(pic)

// color set to full red
b.SetRGBFillColor 1.0, 0.0, 0.0, 1.0

// draw ellipse
Var r as CGRectMBS = CGRectMBS.Make(0, 0, 500, 500)
b.FillEllipseInRect r

// flush drawings
b.Flush

// and show
Backdrop = pic

```

Notes: Requires Mac OS X 10.4.

6.6.47 FillPath

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Fill the context's path using the winding-number fill rule. Any open subpath of the path is implicitly closed.

6.6.48 FillRect(rect as CGRectMBS)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Fills the background with current fill color.

Example:

```

// put in window.paint event
Var c as CGContextMBS

c=window1.CGContextMBS

c.RotateCTM 0.1
c.SetRGBFillColor 0,0,1,0.5
c.FillRect CGMakeRectMBS(0,0,100,100)

c.Flush

```

Notes: Remember that in CoreGraphics the position 0/0 is in the bottom left corner. In Xojo 0/0 is in the top left corner.

6.6.49 fillRoundedRect(x as Double, y as Double, w as Double, h as Double, arcWidth as Double, arcHeight as Double)

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Fills a round rectangle.

Example:

```
Var c as CGContextMBS

c=window1.CGContextMBS

// fill in red
c.SetRGBFillColor 1,0,0,1
c.fillRoundedRect 100,100,100,100,20,20

// draw in green
c.SetRGBStrokeColor 0,1,0,1
c.strokeRoundedRect 100,100,100,100,20,20
```

6.6.50 Flush

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Updates the screen to show the current content.

Notes: Like UpdateNow in the RB window class.

6.6.51 frameArc(x as Double, y as Double, w as Double, h as Double, startAngle as Integer, arcAngle as Integer)

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Draws an arc.

Example:

```
Var c as CGContextMBS

c=window1.CGContextMBS

// fill in red
c.SetRGBFillColor 1,0,0,1
```

```
c.paintArc 100,100,100,100,50,90

// draw in green
c.SetRGBStrokeColor 0,1,0,1
c.frameArc 100,100,100,100,50,90
```

Notes: angles are in degree.

6.6.52 frameOval(x as Double, y as Double, w as Double, h as Double)

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Draws an oval.

Example:

```
Var c as CGContextMBS

c=window1.CGContextMBS

// fill in red
c.SetRGBFillColor 1,0,0,1
c.paintOval 100,100,100,100

// draw in green
c.SetRGBStrokeColor 0,1,0,1
c.frameOval 100,100,100,100
```

6.6.53 frameRect(x as Double, y as Double, w as Double, h as Double)

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Draws the frame for the rectangle.

Example:

```
Var c as CGContextMBS

c=window1.CGContextMBS

// fill in red
c.SetRGBFillColor 1,0,0,1
c.paintRect 100,100,100,100

// draw in green
c.SetRGBStrokeColor 0,1,0,1
c.frameRect 100,100,100,100
```

6.6.54 GetClipBoundingBox as CGRectMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Returns the bounding box of a clipping path.

Notes: Returns the bounding box of the clipping path, specified in user space.

The bounding box is the smallest rectangle completely enclosing all points in the clipping path, including control points for any Bezier curves in the path.

Available in Mac OS X v10.3 and later.

6.6.55 GetCTM as CGAffineTransformMBS

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: Return the current graphics state's transformation matrix.

Notes: Returns nil on any error.

6.6.56 GetPathBoundingBox as CGRectMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Return the bounding box of the context's path.

Notes: The bounding box is the smallest rectangle completely enclosing all points in the path, including control points for Bezier and quadratic curves.

6.6.57 GetPathCurrentPoint as CGPointMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Return the current point of the current subpath of the context's path.

6.6.58 GetTextPosition as CGPointMBS

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: Return the current user-space point at which text will be drawn to (x,y).

Notes: Returns nil on any problem.

6.6.59 IsPathEmpty as boolean

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Returns true if the context's path contains no elements.

6.6.60 MoveToPoint(x as Double, y as Double)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Append a straight line segment from the current point to (x, y).

6.6.61 paintArc(x as Double, y as Double, w as Double, h as Double, startAngle as Integer, arcAngle as Integer)

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Fills an arc.

Example:

```
Var c as CGContextMBS

c=window1.CGContextMBS

// fill in red
c.SetRGBFillColor 1,0,0,1
c.paintArc 100,100,100,100,50,90

// draw in green
c.SetRGBStrokeColor 0,1,0,1
c.frameArc 100,100,100,100,50,90
```

Notes: angles are in degree.

6.6.62 paintOval(x as Double, y as Double, w as Double, h as Double)

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Fills an oval.

Example:

```

Var c as CGContextMBS

c=window1.CGContextMBS

// fill in red
c.SetRGBFillColor 1,0,0,1
c.paintOval 100,100,100,100

// draw in green
c.SetRGBStrokeColor 0,1,0,1
c.frameOval 100,100,100,100

```

6.6.63 paintRect(x as Double, y as Double, w as Double, h as Double)

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Fills the rectangle.

Example:

```

Var c as CGContextMBS

c=window1.CGContextMBS

// fill in red
c.SetRGBFillColor 1,0,0,1
c.paintRect 100,100,100,100

// draw in green
c.SetRGBStrokeColor 0,1,0,1
c.frameRect 100,100,100,100

```

6.6.64 PathContainsPoint(point as CGPointMBS, mode as Integer) as boolean

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Return true if point is contained in the current path of context.

Notes: A point is contained within a contexts path if it is inside the painted region when the path is stroked or filled with opaque colors using the path drawing mode mode. point is specified is user space.

Requires Mac OS X 10.4.

6.6.65 ReplacePathWithStrokedPath

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Replace the path in context with the stroked version of the path, using the parameters of context to calculate the stroked path.

Notes: The resulting path is created such that filling it with the appropriate color will produce the same results as stroking the original path. You can use this path in the same way you can use the path of any context; for example, you can clip to the stroked version of a path by calling this function followed by a call to "ClipPath".

Requires Mac OS X 10.4.

6.6.66 RestoreGState

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Restores the last saved graphics state.

6.6.67 RotateCTM(angle as Double)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Rotate the current graphics state's transformation matrix (the CTM) by 'angle' radians.

Example:

// Rotate a picture not on the edge of the context:

```

Var c as CGImageMBS
Var f as FolderItem
Var d as CGDataProviderMBS
Var cg as CGContextMBS
Var r as CGRectMBS

f=SpecialFolder.Desktop.Child("IMAG0001.JPG")

d=CGDataProviderMBS.CreateWithFile(f)
c=CGCreateImageFromJPEGDataProviderMBS(d,nil,true,0)

cg=window1.CGContextMBS

r=CGMakeRectMBS(-c.Width/2,-c.Height/2,c.Width,c.Height)
cg.TranslateCTM Width/2,Height/2
cg.RotateCTM Slider1.Value/180.0*3.14
cg.DrawPicture c,r

```

cg.Flush

6.6.68 SaveGState

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Saves the current graphics state.

6.6.69 ScaleCTM(sx as Double, sy as Double)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Scale the current graphics state's transformation matrix (the CTM) by (sx, sy).

6.6.70 SelectFont(name as string, size as Double, fontencoding as Integer)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Attempts to find the font named 'name'. If successful, scales it to 'size' units in user space.

Notes: Name: string that contains the PostScript name of the font to set.

'textEncoding' specifies how to translate from bytes to glyphs.

```
kCGEncodingFontSpecific 0
kCGEncodingMacRoman    1
```

As "Comic Sans MS" works, but not ""Comic Sans ms" this functions seems to be case sensitive.

Matthias Buercher notes that sometimes a font is not selected if the RGBFillColor was not set before.

You may need to reset the textmatrix with some code like `c.TextMatrix = CGAffineTransformMBS.Identity`. If the text matrix is different, your text may be rotated, skewed or resized.

6.6.71 SetAllowsAntialiasing(allowsAntialiasing as boolean)

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Whether to allow antialiasing.

Notes: Allow antialiasing in context if `allowsAntialiasing` is true; don't allow it otherwise. This parameter is not part of the graphics state. A context will perform antialiasing if both `allowsAntialiasing` and the graphics state parameter `shouldAntialias` are true.

Requires Mac OS X 10.4 to work.

6.6.72 SetAlpha(alpha as Double)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Set the alpha value in the current graphics state to alpha.

6.6.73 SetBlendMode(BlendMode as Integer)

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Set the blend mode of context to mode.

Notes: Requires Mac OS X 10.4.

6.6.74 SetCharacterSpacing(spacing as Double)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Set the current character spacing to 'spacing'.

Notes: The character spacing is added to the displacement between the origin of one character and the origin of the next.

6.6.75 SetCMYKFillColor(cyan as Double, magenta as Double, yellow as Double, black as Double, alpha as Double = 1.0)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Sets the fill color to a CMYK color.

Notes: colors are from 0 to 1 and alpha is from 0 (transparent) to 1 (solid).

6.6.76 SetCMYKStrokeColor(cyan as Double, magenta as Double, yellow as Double, black as Double, alpha as Double = 1.0)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Sets the stroke color to a CMYK color.

Notes: colors are from 0 to 1 and alpha is from 0 (transparent) to 1 (solid).

6.6.77 SetFillColor(color as CGColorMBS)

Plugin Version: 17.2, Platform: macOS, Targets: All.

Function: Sets the given color as fill color.

6.6.78 SetFillColorSpace(colorspace as CGColorSpaceMBS)

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: Sets the colorspace used for the fill color of the graphics context.

6.6.79 SetFlatness(flatness as Double)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Set the path flatness parameter in the current graphics state to flatness.

6.6.80 SetFont(font as CGFontMBS)

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Sets the current font.

6.6.81 SetFontSize(size as Double)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Set the current font size to 'size'.

6.6.82 SetGrayFillColor(gray as Double, alpha as Double = 1.0)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Sets the fill color to a gray color.

Notes: gray is from 0 to 1 and alpha is from 0 (transparent) to 1 (solid).

6.6.83 SetGrayStrokeColor(gray as Double, alpha as Double = 1.0)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Sets the stroke color to a gray color.

Notes: gray is from 0 to 1 and alpha is from 0 (transparent) to 1 (solid).

6.6.84 SetLineCap(cap as Integer)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Set the line cap in the current graphics state to cap.

6.6.85 SetLineDash(phase as Double, lengths as memoryblock, count as Integer)

Plugin Version: 7.7, Platform: macOS, Targets: All.

Function: Sets the pattern for dashed lines in a graphics context.

Notes: phase:

A value that specifies how far into the dash pattern the line starts, in units of the user space. For example, passing a value of 3 means the line is drawn with the dash pattern starting at three units from its beginning. Passing a value of 0 draws a line starting with the beginning of a dash pattern.

lengths:

A memoryblock of float values that specify the lengths of the painted segments and unpainted segments, respectively, of the dash pattern—or nil for no dash pattern.

For example, passing a memoryblock with the values [2,3] sets a dash pattern that alternates between a 2-user-space-unit-long painted segment and a 3-user-space-unit-long unpainted segment. Passing the values [1,3,4,2] sets the pattern to a 1-unit painted segment, a 3-unit unpainted segment, a 4-unit painted segment, and a 2-unit unpainted segment.

count

If the lengths parameter specifies a memoryblock, pass the number of elements in the memoryblock. Otherwise, pass 0.

6.6.86 SetLineJoin(join as Integer)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Set the line join in the current graphics state to join.

6.6.87 SetLineWidth(width as Double)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Set the line width in the current graphics state to width.

Example:

```
#If XojoVersion >= 2019.02 Then
Var p As ptr = g.Handle(Graphics.HandleTypes.CGContextRef)
Var c As CGContextMBS = CGContextMBS.contextWithCGContext(p)
#Else
Var h As Integer = g.Handle(g.HandleTypeCGContextRef)
Var c As CGContextMBS = CGContextMBS.contextWithCGContext(h)
#EndIf
```

```
c.SetGrayStrokeColor(0,1)
c.SetGrayFillColor(0,1)
c.BeginPath
c.SetLineWidth 0.5
c.MoveToPoint 50, 550
c.AddLineToPoint 100, 600
c.StrokePath
c.Flush
c = nil
```

6.6.88 SetMiterLimit(limit as Double)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Set the miter limit in the current graphics state to limit.

6.6.89 SetRenderingIntent(intent as Integer)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Set the rendering intent in the graphics state to 'intent'.

Notes: Possible values for intent:

kCGRenderingIntentDefault	0
kCGRenderingIntentAbsoluteColorimetric	1
kCGRenderingIntentRelativeColorimetric	2
kCGRenderingIntentPerceptual	3
kCGRenderingIntentSaturation	4

6.6.90 SetRGBFillColor(red as Double, green as Double, blue as Double, alpha as Double = 1.0)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Sets the fill color to a RGB color.

Notes: colors are from 0 to 1 and alpha is from 0 (transparent) to 1 (solid).

6.6.91 SetRGBStrokeColor(red as Double, green as Double, blue as Double, alpha as Double = 1.0)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Sets the stroke color to a RGB color.

Notes: colors are from 0 to 1 and alpha is from 0 (transparent) to 1 (solid).

6.6.92 SetShadow(x as Double, y as Double, blur as Double)

Plugin Version: 8.5, Platform: macOS, Targets: All.

Function: Enables shadowing in a graphics context.

Example:

```
Sub Paint(g As Graphics)
  Var c as CGContextMBS
```

```
c=window1.CGContextMBS
```

```
c.SetShadow 5,5,0.5
```

```
c.SetRGBFillColor 1,0,0,1
```

```
c.FillRect CGMakeRectMBS(100,100,100,100)
```

```
c.Flush
```

```
End Sub
```

Notes: x/y: Specifies a translation of the context's coordinate system, to establish an offset for the shadow ({ 0,0 } specifies a light source immediately above the screen).

blur: A non-negative number specifying the amount of blur.

Shadow parameters are part of the graphics state in a context. After shadowing is set, all objects drawn are shadowed using a black color with 1/3 alpha (i.e., $\text{RGBA} = \{ 0, 0, 0, 1.0/3.0 \}$) in the DeviceRGB color space.

To turn off shadowing:

- Use the standard save/restore mechanism for the graphics state.
- Use `CGContextSetShadowWithColor` to set the shadow color to a fully transparent color (or pass nil as the color).

6.6.93 `SetShadowWithColor(x as Double, y as Double, blur as Double, color-value as CGColorMBS)`

Plugin Version: 8.5, Platform: macOS, Targets: All.

Function: Enables shadowing with color a graphics context.

Notes: x/y: Specifies a translation in base-space.

blur: A non-negative number specifying the amount of blur.

colorvalue: Specifies the color of the shadow, which may contain a non-opaque alpha value. If nil, then shadowing is disabled.

See also `SetShadow`.

6.6.94 `SetShouldAntialias(shouldAntialias as boolean)`

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Turn off antialiasing if 'shouldAntialias' is zero; turn it on otherwise.

Notes: This parameter is part of the graphics state.

6.6.95 `SetShouldSmoothFonts(shouldSmoothFonts as boolean)`

Plugin Version: 7.2, Platform: macOS, Targets: All.

Function: Whether font smoothing is enabled.

Notes: Turn on font smoothing if `shouldSmoothFonts` is true; turn it off otherwise. This parameter is part of the graphics state. Note that this doesn't guarantee that font smoothing will occur: not all destination contexts support font smoothing.

Requires Mac OS X 10.2.

6.6.96 SetStrokeColor(color as CGColorMBS)

Plugin Version: 17.2, Platform: macOS, Targets: All.

Function: Sets the given color as stroke color.

6.6.97 SetStrokeColorSpace(colorspace as CGColorSpaceMBS)

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: Sets the colorspace used for the stroke color of the graphics context.

6.6.98 SetTextDrawingMode(mode as Integer)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Set the text drawing mode to 'mode'.

Notes: Possible values:

kCGTextFill	0
kCGTextStroke	1
kCGTextFillStroke	2
kCGTextInvisible	3
kCGTextFillClip	4
kCGTextStrokeClip	5
kCGTextFillStrokeClip	6
kCGTextClip	7

6.6.99 ShowText(text as string)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Draw 'string' at the point specified by the current text matrix.

Notes: Each byte of the string is mapped through the encoding vector of the current font to obtain the glyph to display.

This function is more for quick and dirty text output, but not for serious drawing as it does not do most unicode strings correctly. Use ATS for better drawing.

Some RB 5.x versions show a bug that the text is not displayed on a CGContext in while running the application in debug mode.

6.6.100 ShowTextAtPoint(text as string, x as Double, y as Double)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Draw 'string' at the point '(x, y)', specified in user space.

Notes: Each byte of the string is mapped through the encoding vector of the current font to obtain the glyph to display.

This function is more for quick and dirty text output, but not for serious drawing as it does not do most unicode strings correctly. Use ATS for better drawing.

Some RB 5.x versions show a bug that the text is not displayed on a CGContext in while running the application in debug mode.

6.6.101 StrokeEllipseInRect(rect as CGRectMBS)

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Stroke an ellipse (an oval) inside rect.

Notes: Requires Mac OS X 10.4.

6.6.102 StrokePath

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Stroke the context's path.

Example:

```
Var c as CGContextMBS

c=window1.CGContextMBS

c.SetRGBStrokeColor 1,0,0,1
c.BeginPath
c.MoveToPoint 0,0
c.AddLineToPoint 100,100
c.StrokePath
c.Flush
```

6.6.103 StrokeRect(rect as CGRectMBS)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Stroke 'rect' with the current stroke color and the current linewidth.

6.6.104 StrokeRectWithWidth(rect as CGRectMBS, width as Double)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Stroke 'rect' with the current stroke color, using 'width' as the the line width.

6.6.105 strokeRoundedRect(x as Double, y as Double, w as Double, h as Double, arcWidth as Double, arcHeight as Double)

Plugin Version: 8.4, Platform: macOS, Targets: All.

Function: Draws the frame for the round rectangle.

Example:

```
Var c as CGContextMBS
```

```
c=window1.CGContextMBS
```

```
// fill in red
c.SetRGBFillColor 1,0,0,1
c.fillRoundedRect 100,100,100,100,20,20

// draw in green
c.SetRGBStrokeColor 0,1,0,1
c.strokeRoundedRect 100,100,100,100,20,20
```

6.6.106 Synchronize

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Synchronizes the context with the device.

6.6.107 TranslateCTM(tx as Double, ty as Double)

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Translate the current graphics state's transformation matrix (the CTM) by (tx,ty).

Example:

```
// Rotate a PDF page

// our files
Var sourcefile as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
Var destfile as FolderItem = SpecialFolder.Desktop.Child("rotated.pdf")

// open PDF
Var pdf as CGPDFDocumentMBS = sourcefile.OpenAsCGPDFDocumentMBS

// query media size of first page
Var r as CGRectMBS = pdf.MediaBox(1)

// create new PDF
Var c as CGContextMBS = destfile.NewCGPDFDocumentMBS(r,"title","Author","Creator")

// create rotated rectangle
Var 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
```

6.6.108 Properties**6.6.109 handle as Integer**

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: The handle for this context.

Notes: Handle is a CGContextRef.

(Read and Write property)

6.6.110 InterpolationQuality as Integer

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: The interpolation quality for image rendering of this context.

Notes: The interpolation quality is a gstate-parameter which controls the level of interpolation performed when an image is interpolated (for example, when scaling the image). Note that it is merely a hint to the context: not all contexts support all interpolation quality levels.

Possible values:

kCGInterpolationDefault	0	Let the context decide.
kCGInterpolationNone	1	Never interpolate.
kCGInterpolationLow	2	Fast, low quality.
kCGInterpolationHigh	3	Slow, high quality.

(Read and Write property)

6.6.111 RetainCount as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the reference counter of the object.

Notes: If the retain count falls below 1, the object is destroyed.

(Read only property)

6.6.112 TextMatrix as CGAffineTransformMBS

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: Return the text matrix.

Notes: Returns nil on any error.

(Read and Write computed property)

6.6.113 TextPosition as CGPointMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: User-space point at which text will be drawn to (x,y).

Notes: (Read and Write computed property)

6.6.114 Constants

Constants

Constant	Value	Description
kCGLineCapButt	0	A constant for the Line cap styles.
kCGLineCapRound	1	A constant for the Line cap styles.
kCGLineCapSquare	2	A constant for the Line cap styles.
kCGLineJoinBevel	2	A constant for the Line join styles.
kCGLineJoinMiter	0	A constant for the Line join styles.
kCGLineJoinRound	1	A constant for the Line join styles.
kCGPathEOFill	1	A constant for the Drawing modes for paths.
kCGPathEOFillStroke	4	A constant for the Drawing modes for paths.
kCGPathFill	0	A constant for the Drawing modes for paths.
kCGPathFillStroke	3	A constant for the Drawing modes for paths.
kCGPathStroke	2	A constant for the Drawing modes for paths.
kCGTextClip	7	A constant for the Drawing modes for text.
kCGTextFill	0	A constant for the Drawing modes for text.
kCGTextFillClip	4	A constant for the Drawing modes for text.
kCGTextFillStroke	2	A constant for the Drawing modes for text.
kCGTextFillStrokeClip	6	A constant for the Drawing modes for text.
kCGTextInvisible	3	A constant for the Drawing modes for text.
kCGTextStroke	1	A constant for the Drawing modes for text.
kCGTextStrokeClip	5	A constant for the Drawing modes for text.

Blend Mode Constants

Constant	Value	Description
kCGBlendModeClear	16	$R = 0$ Available in Mac OS X v10.5 and later.
kCGBlendModeColor	14	
kCGBlendModeColorBurn	7	
kCGBlendModeColorDodge	6	
kCGBlendModeCopy	17	$R = S$ Available in Mac OS X v10.5 and later.
kCGBlendModeDarken	4	
kCGBlendModeDestinationAtop	24	$R = S*(1 - D_a) + D*S_a$ Available in Mac OS X v10.5 and later.
kCGBlendModeDestinationIn	22	$R = D*S_a$ Available in Mac OS X v10.5 and later.
kCGBlendModeDestinationOut	23	$R = D*(1 - S_a)$ Available in Mac OS X v10.5 and later.
kCGBlendModeDestinationOver	21	$R = S*(1 - D_a) + D$ Available in Mac OS X v10.5 and later.
kCGBlendModeDifference	10	
kCGBlendModeExclusion	11	
kCGBlendModeHardLight	9	
kCGBlendModeHue	12	
kCGBlendModeLighten	5	
kCGBlendModeLuminosity	15	
kCGBlendModeMultiply	1	
kCGBlendModeNormal	0	
kCGBlendModeOverlay	3	
kCGBlendModePlusDarker	26	$R = \text{MAX}(0, (1 - D) + (1 - S))$ Available in Mac OS X v10.5 and later.
kCGBlendModePlusLighter	27	$R = \text{MIN}(1, S + D)$ Available in Mac OS X v10.5 and later.
kCGBlendModeSaturation	13	
kCGBlendModeScreen	2	
kCGBlendModeSoftLight	8	
kCGBlendModeSourceAtop	20	$R = S*D_a + D*(1 - S_a)$ Available in Mac OS X v10.5 and later.
kCGBlendModeSourceIn	18	$R = S*D_a$ Available in Mac OS X v10.5 and later.
kCGBlendModeSourceOut	19	$R = S*(1 - D_a)$ Available in Mac OS X v10.5 and later.
kCGBlendModeXOR	25	$R = S*(1 - D_a) + D*(1 - S_a)$. This XOR mode is only nominally related to the classical bitmap XOR operation, which is not supported by Quartz 2D. Available in Mac OS X v10.5 and later.

Encodings

Constant	Value	Description
kCGEncodingFontSpecific	0	Font Specific encoding
kCGEncodingMacRoman	1	Mac Roman Encoding

Interpolation Constants

Constant	Value	Description
kCGInterpolationDefault	0	Let the context decide.
kCGInterpolationHigh	3	Slower, higher quality.
kCGInterpolationLow	2	Faster, lower quality.
kCGInterpolationMedium	4	A medium level of interpolation quality. This setting is slower than the low setting but faster than the high setting. Available in Mac OS X v10.6 and later.
kCGInterpolationNone	1	Never interpolate.

6.7 class CGDataConsumerMBS

6.7.1 class CGDataConsumerMBS

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: A class for CoreGraphics to write data.

Blog Entries

- [MBS REALbasic plug-ins version 9.5](#)

6.7.2 Methods

6.7.3 Constructor

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a new event based data consumer.

Notes: On failure the handle property is zero.

See also:

- [6.7.4 Constructor\(file as folderitem\)](#) 181
- [6.7.5 Constructor\(url as string\)](#) 181

6.7.4 Constructor(file as folderitem)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a new CGDataConsumer to write data to the given file.

Notes: On failure the handle property is zero.

See also:

- [6.7.3 Constructor](#) 181
- [6.7.5 Constructor\(url as string\)](#) 181

6.7.5 Constructor(url as string)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a new CGDataConsumer to write data to the given url.

Notes: On failure the handle property is zero.

See also:

- [6.7.3 Constructor](#) 181

- 6.7.4 Constructor(file as folderitem)

6.7.6 CreateWithFile(file as folderitem) as CGDataConsumerMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a new CGDataConsumer to write data to the given file.

Notes: Returns nil on any error.

6.7.7 CreateWithURL(url as string) as CGDataConsumerMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a new CGDataConsumer to write data to the given url.

Notes: Returns nil on any error.

6.7.8 Properties

6.7.9 Handle as Integer

Plugin Version: 4.3, Platform: macOS, Targets: All.

Function: The handle to the CGDataConsumer object.

Notes: Data is a CGDataConsumerRef.

(Read and Write property)

6.7.10 Events

6.7.11 CloseConsumer

Plugin Version: 4.3, Platform: macOS, Targets: .

Function: Event called when the consumer is no longer needed.

6.7.12 Put(data as string) as Integer

Plugin Version: 4.3, Platform: macOS, Targets: .

Function: Some data needs to be stored.

Notes: Write the given data string into your backbuffer or file and return the number of bytes written.

Return 0 if you can't accept new input.

6.8 class CGDataProviderMBS

6.8.1 class CGDataProviderMBS

Plugin Version: 3.3, Platform: macOS, Targets: All.

Function: A class for a Coregraphics data provider.

Notes: Something like a textinputstream in RB, but for Coregraphics this stream provides binary data.

Blog Entries

- [MBS REALbasic plug-ins version 9.5](#)

6.8.2 Methods

6.8.3 Constructor(data as string)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a new data provider reading data from the given string.

Notes: On failure the handle property is zero.

Available in Mac OS X v10.4 and later.

See also:

- 6.8.4 Constructor(file as folderitem)

184

6.8.4 Constructor(file as folderitem)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a data provider using a CFUrl.

Notes: On failure the handle property is zero.

See also:

- 6.8.3 Constructor(data as string)

184

6.8.5 CreateWithData(data as string) as CGDataProviderMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a new data provider reading data from the given string.

Notes: On failure the handle property is zero.

Available in Mac OS X v10.4 and later.

6.8.6 CreateWithFile(file as folderitem) as CGDataProviderMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a data provider using a file.

Notes: Returns nil on any error.

6.8.7 CreateWithURL(url as string) as CGDataProviderMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a data provider using an URL.

Notes: Returns nil on any error.

6.8.8 Data as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns a copy of the provider's data.

Notes: Available in Mac OS X v10.5 and later.

6.8.9 Properties

6.8.10 Handle as Integer

Plugin Version: 3.3, Platform: macOS, Targets: All.

Function: The handle to CGDataProvider object.

Notes: (a CGDataProviderRef)

(Read and Write property)

6.9 class CGDisplayConfigMBS

6.9.1 class CGDisplayConfigMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The class for a display configuration.

Notes: The display reconfiguration process:

- Make all desired changes for all displays.
- Commit the changes using Complete(), or cancel with Cancel().

The resulting layout will be adjusted to remove gaps or overlaps from the requested layout, if needed. */

Blog Entries

- [MBS Plugins 11.1 Release notes](#)
- [MBS Real Studio Plugins, version 11.1pr8](#)

6.9.2 Methods

6.9.3 Cancel

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Cancel a set of display configuration changes.

Notes: On return, the configuration is cancelled and is no longer valid.

Calls CGCancelDisplayConfiguration.

6.9.4 Complete(options as Integer)

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Complete a set of display configuration changes.

Notes: On return, the configuration is no longer valid.

A configuration change can apply for the life of an application, the life of a login session, or permanently. If a request is made to make a change permanent, and the change cannot be supported by Mac OS X user interface, then the configuration change lasts only for the current login session.

A permanent configuration change also becomes the current session's configuration.

When the system reverts configurations at app termination, the configuration reverts to the session or permanent configuration setting.

When the system reverts configurations at session termination, the configuration reverts to the permanent configuration setting.

This operation may fail if an unsupported display mode is requested, or if another app is running in full-screen mode.

Calls `CGCompleteDisplayConfiguration`.

6.9.5 `DisplayMode(display as CGDisplayMBS, mode as CGDisplayModeMBS)`

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Configure the display mode of a display.

Notes: A display mode is a set of properties such as width, height, pixel depth, and refresh rate, and options such as stretched LCD panel filling.

If you use this function to change the mode of a display in a mirroring set, Quartz may adjust the bounds, resolutions, and depth of the other displays in the set to a safe mode, with matching depth and the smallest enclosing size.

Calls `CGConfigureDisplayWithDisplayMode`.

6.9.6 `MirrorOfDisplay(display as CGDisplayMBS, master as CGDisplayMBS)`

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Make a display a mirror of a master display.

Notes: Pass `nil` for the master display to disable mirroring.

Pass `MainDisplay` for the master display to mirror the main display.

Display mirroring and display matte generation are implemented either in hardware (preferred) or software, at the discretion of the device driver.

- Hardware mirroring

With hardware mirroring enabled, all drawing is directed to the primary display — see `PrimaryDisplay`.

If the device driver selects hardware matte generation, the display bounds and rowbytes values are adjusted to reflect the active drawable area.

- Software mirroring

In this form of mirroring, identical content is drawn into each display in the mirroring set. Applications that use the window system need not be concerned about mirroring, as the window system takes care of all flushing of window content to the appropriate displays.

Applications that draw directly to the display, as with display capture, must make sure to draw the same content to all mirrored displays in a software mirror set. When drawing to software mirrored displays using a full screen OpenGL context (not drawing through a window), you should create shared OpenGL contexts for each display and re-render for each display.

You can use the function `GetActiveDisplayList` to determine which displays are active, or drawable. This automatically gives your application the correct view of the current displays.

Calls `CGConfigureDisplayMirrorOfDisplay`.

6.9.7 `Mode(display as CGDisplayMBS, mode as Dictionary)`

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Sets the mode given the values in the dictionary.

Notes: Deprecated with Mac OS X 10.6.

Calls `CGConfigureDisplayMode`.

6.9.8 `Origin(display as CGDisplayMBS, x as Integer, y as Integer)`

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Configure the origin of a display in global display coordinates.

Notes: The new origin of the display is placed as close as possible to the requested location, without overlapping or leaving a gap between displays.

Any display whose origin is not explicitly set in a reconfiguration will be repositioned to a location as close as possible to its current location without overlapping or leaving a gap between displays.

Note that setting the origin of a display which is mirroring another display will remove that display from any mirroring set.

Calls `CGConfigureDisplayOrigin`.

6.9.9 RestorePermanentDisplayConfiguration

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Restore the permanent display configuration settings for the current user.

Notes: Calls CGRestorePermanentDisplayConfiguration.

6.9.10 StereoOperation(display as CGDisplayMBS, stereo as Boolean, force-BlueLine as Boolean)

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Enable or disable stereo operation for a display.

Notes: Note that the system normally detects the presence of a stereo window, and will automatically switch a display containing a stereo window to stereo operation. This function provides a mechanism to force a display to stereo operation, and to set options (such as blue line sync signal) when in stereo operation.

When in stereo operation, a display may need to generate a special stereo sync signal as part of the video output. The sync signal consists of a blue line which occupies the first 25% of the last scanline for the left eye view, and the first 75% of the last scanline for the right eye view. The remainder of the scanline is black. To force the display to generate this sync signal, pass true for forceBlueLine; otherwise, pass false.

Returns kCGErrorSuccess on success, or kCGErrorRangeCheck if the display does not support the stereo operation settings requested.

On success, the display resolution, mirroring mode, and available display modes may change due to hardware-specific capabilities and limitations. You should check these settings to verify that they are appropriate for your application.

Calls CGConfigureDisplayStereoOperation.

6.9.11 Properties

6.9.12 Handle as Integer

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The internal object reference.

Notes: (Read only property)

6.9.13 Lasterror as Integer

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The last error value.

Notes: (Read only property)

6.9.14 Constants

Options for Complete

Constant	Value	Description
kCGConfigureForAppOnly	0	For application only.
kCGConfigureForSession	1	For session only.
kCGConfigurePermanently	2	Permanently.

6.10 class CGDisplayMBS

6.10.1 class CGDisplayMBS

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: A class for a CoreGraphics Display object.

Example:

```
Var c as new CGDisplayMBS
Var lines(-1) as string
```

```
Var DPIWidth as Double = c.PixelsWide/(c.ScreenSizeWidth/10.0/2.54)
```

```
Var DPIHeight as Double = c.PixelsHigh/(c.ScreenSizeHeight/10.0/2.54)
```

```
lines.append str(c.ScreenSizeWidth)+" x "+str(c.ScreenSizeHeight)+" Millimeter with"
```

```
lines.append str(c.PixelsWide)+" x "+str(c.PixelsHigh)+" Pixel is"
```

```
lines.append str(DPIWidth)+" x "+str(DPIHeight)+" DPI"
```

```
MsgBox Join(lines,EndOfLine)
```

Blog Entries

- [MBS Xojo Plugins, version 21.5pr3](#)
- [MBS Xojo / Real Studio Plugins, version 15.4pr3](#)
- [MBS Xojo / Real Studio Plugins, version 15.3pr5](#)
- [MBS Xojo / Real Studio Plugins, version 15.1pr4](#)
- [MBS Xojo / Real Studio Plugins, version 15.0pr7](#)
- [MBS Xojo / Real Studio Plugins, version 14.4pr2](#)
- [Display Rotation](#)
- [MBS Plugins 11.1 Release notes](#)
- [MBS Plugins 10.3 Release Notes](#)
- [MonkeyBread Software Releases the MBS REALbasic plug-ins 8.6](#)

6.10.2 Methods

6.10.3 AllDisplayModes(Options as Dictionary = nil) as CGDisplayModeMBS()

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Return an array of all modes for the specified display.

Example:

```

Var d as CGDisplayMBS = CGDisplayMBS.MainDisplay

Var options As New Dictionary
options.Value(CGDisplayMBS.kCGDisplayShowDuplicateLowResolutionModes) = false

Var modes(-1) as string
for each mode as CGDisplayModeMBS in d.AllDisplayModes(options)
modes.append str(mode.Width)+" x "+str(mode.Height)
next
MsgBox Join(modes,EndOfLine)

```

Notes: Returns an empty array on any error.
Requires Mac OS X 10.6

6.10.4 AvailableModes as Dictionary()

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Returns an array filled with dictionaries (one for each mode).

Example:

```

// Display all available graphic modes:

Listbox1.DeleteAllRows
Listbox1.ColumnCount = 4

Var d as CGDisplayMBS = CGDisplayMBS.MainDisplay

Var a(-1) as Dictionary = d.AvailableModes

for each di as Dictionary in a

Listbox1.AddRow di.Value(d.kCGDisplayMode).stringValue
Listbox1.cell(Listbox1.LastIndex,1) = di.Value(d.kCGDisplayWidth)+" x "+di.Value(d.kCGDisplayHeight)
Listbox1.cell(Listbox1.LastIndex,2) = di.Value(d.kCGDisplayRefreshRate)
Listbox1.cell(Listbox1.LastIndex,3) = di.Value(d.kCGDisplayBitsPerPixel)

next

```

Notes: Returns empty error on any error.

6.10.5 BestModeForParameters(BitsPerPixel as Integer, Width as Integer, Height as Integer, byref ExactMatch as boolean) as Dictionary

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Returns a Dictionary with the best mode found for the given parameters.

Notes: Returns nil on any error.

Try to find a display mode of specified depth with dimensions equal or greater than specified. If no depth match is found, try for the next larger depth with dimensions equal or greater than specified. If no luck, then just return the current mode.

exactmatch is set to 'true' if an exact match in width, height, and depth is found, and 'false' otherwise. Returns nil if display is invalid.

6.10.6 BestModeForParametersAndRefreshRate(BitsPerPixel as Integer, Width as Integer, Height as Integer, RefreshRate as Integer, byref ExactMatch as boolean) as Dictionary

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Returns a Dictionary with the best mode found for the given parameters.

Notes: Returns nil on any error.

Try to find a display mode of specified depth with dimensions equal or greater than specified. If no depth match is found, try for the next larger depth with dimensions equal or greater than specified. If no luck, then just return the current mode.

exactmatch is set to 'true' if an exact match in width, height, and depth is found, and 'false' otherwise. Returns nil if display is invalid.

6.10.7 BestModeForParametersAndRefreshRateWithProperty(BitsPerPixel as Integer, Width as Integer, Height as Integer, RefreshRate as Integer, propertyName as string, byref ExactMatch as boolean) as Dictionary

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns a Dictionary with the best mode found for the given parameters.

Notes: Returns nil on any error.

Try to find a display mode of specified depth with dimensions equal or greater than specified.

If no depth match is found, try for the next larger depth with dimensions equal or greater than specified. If no luck, then just return the current mode.

exactmatch is set to 'true' if an exact match in width, height, and depth is found, and 'false' otherwise. Returns nil if display is invalid.

6.10.8 Capture as Integer

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Captures this display for your use.

Example:

```

Var d as CGDisplayMBS = CGDisplayMBS.MainDisplay

if d.Capture = 0 then

Var c as CGContextMBS = d.DrawingContext

// watch it drawing red
c.SetRGBFillColor 1.0, 0.0, 0.0, 1.0
c.FillRect CGMakeRectMBS(0,0,d.PixelsWide,d.PixelsHigh)

DelayMBS 3.0

call d.Release
else
MsgBox "Failed to capture displays."
end if

```

Notes: Don't forget to Release the display later. Returns an error code.

6.10.9 CaptureAllDisplays as Integer

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Capture all displays.

Example:

```

Var d as CGDisplayMBS = CGDisplayMBS.MainDisplay

if d.CaptureAllDisplays = 0 then

```

```

Var c as CGContextMBS = d.DrawingContext

// watch it drawing red
c.SetRGBFillColor 1.0, 0.0, 0.0, 1.0
c.FillRect CGMakeRectMBS(0,0,d.PixelsWide,d.PixelsHigh)

DelayMBS 3.0

call d.ReleaseAllDisplays
else
MsgBox "Failed to capture displays."
end if

```

Notes: This has the nice effect of providing an immersive environment, and preventing other apps from trying to adjust themselves to display changes only needed by your app.

Returns an error code.

6.10.10 CaptureAllDisplaysWithOptions(options as Integer) as Integer

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Capture all displays.

Example:

```

Var d as CGDisplayMBS = CGDisplayMBS.MainDisplay

if d.CaptureAllDisplaysWithOptions(d.kCGCaptureNoOptions) = 0 then

Var c as CGContextMBS = d.DrawingContext

// watch it drawing red
c.SetRGBFillColor 1.0, 0.0, 0.0, 1.0
c.FillRect CGMakeRectMBS(0,0,d.PixelsWide,d.PixelsHigh)

DelayMBS 3.0

call d.ReleaseAllDisplays
else
MsgBox "Failed to capture displays."
end if

```

Notes: This has the nice effect of providing an immersive environment, and preventing other apps from trying to adjust themselves to display changes only needed by your app.

Use `kCGCaptureNoOptions` and `kCGCaptureNoFill` for options parameter.
Returns an error code.

6.10.11 `CaptureWithOptions(options as Integer) as Integer`

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Captures this display for your use.

Example:

```

Var d as CGDisplayMBS = CGDisplayMBS.MainDisplay

if d.CaptureWithOptions(d.kCGCaptureNoOptions) = 0 then

  Var c as CGContextMBS = d.DrawingContext

  // watch it drawing red
  c.SetRGBFillColor 1.0, 0.0, 0.0, 1.0
  c.FillRect CGMakeRectMBS(0,0,d.PixelsWide,d.PixelsHigh)

  DelayMBS 3.0

  call d.Release
else
  MsgBox "Failed to capture displays."
end if

```

Notes: Don't forget to Release the display later.

Returns an error code.

Use `kCGCaptureNoOptions` and `kCGCaptureNoFill` for options parameter.

6.10.12 `CreateImage as CGImageMBS`

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Return an image containing the contents of the display.

Example:

```

Var d as CGDisplayMBS = CGDisplayMBS.MainDisplay
Var img as CGImageMBS = d.CreateImage

```

```

Backdrop = img.Picture

```

Notes: Requires Mac OS X 10.6.

6.10.13 CreateImageAsync(receiverDelegate as CreateImageAsyncDelegateMBS, jpegQuality as Double = 0.9, tag as Variant = nil)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: Asynchronously creates an image with screenshot.

Notes: If jpeg is ≥ 0 , we use it as quality for the jpeg compression and provide the JPEG data as memoryblock in the delegate.

Tag is passed through as is.

Delegate declaration:

```
CreateImageAsyncDelegateMBS(img as CGImageMBS, JPEGData as MemoryBlock, Tag as Variant)
```

6.10.14 CreateImageForRect(rect as CGRectMBS) as CGImageMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Return an image containing the contents of the rectangle rect, specified in display space, of the display identified by self.

Example:

```
Var r as CGRectMBS = CGMakeRectMBS(100,100,500,500)
Var d as CGDisplayMBS = CGDisplayMBS.MainDisplay
Var img as CGImageMBS = d.CreateImageForRect(r)
```

```
Backdrop = img.Picture
```

Notes: The actual rectangle used is the rectangle returned from `CGRectIntegral(rect)`.
Requires Mac OS X 10.6.

6.10.15 DrawingContext as CGContextMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Return a CGContext suitable for drawing to the captured display, or nil if display has not been captured.

Example:

```
Var d as CGDisplayMBS = CGDisplayMBS.MainDisplay
if d.Capture = 0 then
```

```

Var c as CGContextMBS = d.DrawingContext

// watch it drawing red
c.SetRGBFillColor 1.0, 0.0, 0.0, 1.0
c.FillRect CGMakeRectMBS(0,0,d.PixelsWide,d.PixelsHigh)

DelayMBS 3.0

call d.Release
else
MsgBox "Failed to capture displays."
end if

```

Notes: The context is owned by the device and should not be released by the caller.

The context remains valid while the display is captured and while the display configuration is unchanged. Releasing the captured display or reconfiguring the display invalidates the drawing context.

To determine when the display configuration is changing, use CGDisplayReconfigurationEventMBS class.

6.10.16 GetActiveDisplayList as CGDisplayMBS()

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns an array with the active displays.

Example:

```

// List all displays in a 2 column listbox:

Listbox1.DeleteAllRows

for each d as CGDisplayMBS in CGDisplayMBS.GetActiveDisplayList

Listbox1.AddRow hex(d.Handle)
Listbox1.cell(Listbox1.LastIndex,1)=str(d.PixelsWide)+" x "+str(d.PixelsHigh)

next

```

Notes: Returns an empty array on any error.

The first display returned in the list is the main display, the one with the menu bar. When mirroring, this will be the largest display, or if all are the same size, the one with the deepest pixel

depth.

This function was named CGGetActiveDisplayListMBS in older plugin versions.

6.10.17 GetDisplaysWithOpenGLDisplayMask(mask as Integer) as CGDisplayMBS()

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Find all displays for the given OpenGL mask.

Notes: Returns an empty array on any error.

6.10.18 GetDisplaysWithPoint(cgpoint as CGPointMBS) as CGDisplayMBS()

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns an array of CGDisplayMBS objects who are visible at the point on the virtual screen.

Example:

// List all displays in a 2 column listbox which match the point:

```
Listbox1.DeleteAllRows
```

```
Var p as CGPointMBS = CGMakePointMBS(100,100)
```

```
for each d as CGDisplayMBS in CGDisplayMBS.GetDisplaysWithPoint(p)
```

```
Listbox1.AddRow hex(d.Handle)
```

```
Listbox1.cell(Listbox1.LastIndex,1)=str(d.PixelsWide)+" x "+str(d.PixelsHigh)
```

```
next
```

Notes: Returns an empty array on any error.

This function was called CGGetDisplaysWithPointMBS in older plugin versions.

See also:

- 6.10.19 GetDisplaysWithPoint(x as Double, y as Double) as CGDisplayMBS()

199

6.10.19 GetDisplaysWithPoint(x as Double, y as Double) as CGDisplayMBS()

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns an array of CGDisplayMBS objects who are visible at the point on the virtual screen.

Example:

```
// List all displays in a 2 column listbox which match the point:

Listbox1.DeleteAllRows

for each d as CGDisplayMBS in CGDisplayMBS.GetDisplaysWithPoint(100,100)

Listbox1.AddRow hex(d.Handle)
Listbox1.cell(Listbox1.LastIndex,1)=str(d.PixelsWide)+" x "+str(d.PixelsHigh)

next
```

Notes: Returns an empty array on any error.

This function was called CGGetDisplaysWithPointMBS in older plugin versions.
See also:

- 6.10.18 GetDisplaysWithPoint(cgpoint as CGPointMBS) as CGDisplayMBS() 199

6.10.20 GetDisplaysWithRect(cgrect as CGRectMBS) as CGDisplayMBS()

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns an array of CGDisplayMBS objects who are visible within the rectangle on the virtual screen.

Example:

```
// List all displays in a 2 column listbox which match the rectangle:

Listbox1.DeleteAllRows

Var p as CGRectMBS = CGMakeRectMBS(100,100,100,100)
for each d as CGDisplayMBS in CGDisplayMBS.GetDisplaysWithRect(p)

Listbox1.AddRow hex(d.Handle)
Listbox1.cell(Listbox1.LastIndex,1)=str(d.PixelsWide)+" x "+str(d.PixelsHigh)

next
```

Notes: Returns an empty array on any error.

This function was called CGGetDisplaysWithRectMBS in older plugin versions.
See also:

- 6.10.21 GetDisplaysWithRect(x as Double, y as Double, w as Double, h as Double) as CGDisplayMBS()
201

6.10.21 GetDisplaysWithRect(x as Double, y as Double, w as Double, h as Double) as CGDisplayMBS()

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns an array of CGDisplayMBS objects who are visible within the rectangle on the virtual screen.

Example:

```
// List all displays in a 2 column listbox which match the rectangle:
```

```
Listbox1.DeleteAllRows
```

```
for each d as CGDisplayMBS in CGDisplayMBS.GetDisplaysWithRect(100,100,100,100)
```

```
Listbox1.AddRow hex(d.Handle)
```

```
Listbox1.cell(Listbox1.LastIndex,1)=str(d.PixelsWide)+" x "+str(d.PixelsHigh)
```

```
next
```

Notes: Returns an empty array on any error.

This function was called CGGetDisplaysWithRectMBS in older plugin versions.

See also:

- 6.10.20 GetDisplaysWithRect(cgrect as CRectMBS) as CGDisplayMBS()

200

6.10.22 GetDisplayTransferByTable(capacity as Integer, red as memoryblock, green as memoryblock, blue as memoryblock, byref samplecount as Integer) as Integer

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Get transfer tables.

Notes: Capacity should contain the number of samples each array can hold, and sampleCount is filled in with the number of samples actually copied in.

You must pass in Memoryblocks with the given size (capacity*4 Bytes).

Returns an error code.

6.10.23 GetDisplayTransferFormula(byref formula as CGDisplayTransferFormulaMBS) as Integer

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Queries the display for the current gamma formula.

Notes: Returns an error code.

6.10.24 GetLastMouseDelta(byref deltax as Integer, byref deltay as Integer)

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Report the mouse position change associated with the last mouse move event received by this application.

6.10.25 GetOnlineDisplayList as CGDisplayMBS()

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns an array with the online displays.

Example:

// List all online displays in a 2 column listbox:

```
Listbox1.DeleteAllRows
```

```
for each d as CGDisplayMBS in CGDisplayMBS.GetOnlineDisplayList
```

```
Listbox1.AddRow hex(d.Handle)
```

```
Listbox1.cell(Listbox1.LastIndex,1)=str(d.PixelsWide)+" x "+str(d.PixelsHigh)
```

```
next
```

Notes: Returns an empty array on any error.

The first display returned in the list is the main display, the one with the menu bar.

When mirroring, this will be the largest display, or if all are the same size, the one with the deepest pixel depth.

With hardware mirroring, a display may be on-line, but not necessarily active, or drawable. Programs which manipulate display settings such as the palette or gamma tables need access to all displays in use, including hardware mirrors which are not drawable.

This function was named CGGetOnlineDisplayListMBS in older plugin versions.

6.10.26 HideCursor as Integer

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Hides the mouse cursor.

Notes: Returns an error code.

Decrements hide cursor count.

6.10.27 Info as Dictionary

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the information CFDictionary about the display.

Example:

```
Var c as CGDisplayMBS = CGDisplayMBS.MainDisplay
```

```
Var d as Dictionary = c.Info
```

```
MsgBox d.Value("DisplaySerialNumber")
```

Notes: On any error the return value is nil.

This function leaks around 4 to 16 KB of memory on Mac OS X 10.4. Not in the versions 10.3 or 10.5.

6.10.28 InfoAsCFDictionary as Variant

Plugin Version: 14.4, Platform: macOS, Targets: Desktop only.

Function: Returns the information CFDictionary about the display.

Notes: On any error the return value is nil.

Returns a CFDictionaryMBS object. Returned as Variant to reduce plugin dependencies.

6.10.29 IsCaptured as boolean

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Returns true if you captured this display.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(d.IsCaptured)
```

6.10.30 MainDisplay as CGDisplayMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Return the display object of the current main display.

Example:

```
Var c as CGDisplayMBS = CGDisplayMBS.MainDisplay
```

```
MsgBox str(c.PixelsWide)+" x "+str(c.PixelsHigh)
```

6.10.31 MoveCursorToPoint(x as Double, y as Double) as Integer

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Move the cursor to the specified point relative to the display origin (the upper left corner of the display).

Example:

```
Function KeyDown(Key As String) As Boolean
```

```
// in a window keydown event:
```

```
if asc(key)=32 then
```

```
Var c as new CGDisplayMBS
```

```
Var error as Integer = c.MoveCursorToPoint(200,200)
```

```
Title=str(error) // zero on success
```

```
end if
```

```
End Function
```

Notes: Returns CGDisplayNoErr (0) on success.

No events are generated as a result of this move.

Points that would lie outside the desktop are clipped to the desktop.

6.10.32 OpenGLDisplayMask as Integer

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Return the OpenGL display mask for display, or 0 is display is an invalid display.

6.10.33 Release as Integer

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Releases the display.

Example:

```

Var d as CGDisplayMBS = CGDisplayMBS.MainDisplay

if d.Capture = 0 then

    Var c as CGContextMBS = d.DrawingContext

    // watch it drawing red
    c.SetRGBFillColor 1.0, 0.0, 0.0, 1.0
    c.FillRect CGMakeRectMBS(0,0,d.PixelsWide,d.PixelsHigh)

    DelayMBS 3.0

    call d.Release
else
    MsgBox "Failed to capture displays."
end if

```

Notes: Returns an error code.

6.10.34 ReleaseAllDisplays as Integer

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Release all captured displays.

Example:

```

Var d as CGDisplayMBS = CGDisplayMBS.MainDisplay

if d.CaptureAllDisplays = 0 then

    Var c as CGContextMBS = d.DrawingContext

    // watch it drawing red
    c.SetRGBFillColor 1.0, 0.0, 0.0, 1.0
    c.FillRect CGMakeRectMBS(0,0,d.PixelsWide,d.PixelsHigh)

    DelayMBS 3.0

    call d.ReleaseAllDisplays
else

```

```
MsgBox "Failed to capture displays."
end if
```

Notes: Release all captured displays, and restore the display modes to the user's preferences. May be used in conjunction with `Capture` or `CaptureAllDisplays`.

6.10.35 `RestoreColorSyncSettings`

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Restore gamma tables of system displays to the user's `ColorSync` specified values.

6.10.36 `SetDisplayMode(mode as CGDisplayModeMBS) as Integer`

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Switch the display mode to `mode`.

Notes: The selected display mode persists for the life of the program, and automatically reverts to the permanent setting when the program terminates.

When changing display modes of displays in a mirroring set, other displays in the mirroring set will be set to a display mode capable of mirroring the bounds of the largest display being explicitly set.

Note that after switching, display parameters and addresses may change.

6.10.37 `SetDisplayTransferByByteTable(count as Integer, red as memoryblock, green as memoryblock, blue as memoryblock) as Integer`

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Set a display gamma/transfer function using tables of data for each channel.

Notes: As a convenience, allow setting of the gamma table by byte values.

Returns an error code.

6.10.38 `SetDisplayTransferByTable(count as Integer, red as memoryblock, green as memoryblock, blue as memoryblock) as Integer`

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Set a display gamma/transfer function.

Notes: Set a display gamma/transfer function using tables of data for each channel.

Values within each table should have values in the range of 0.0 through 1.0.

The same table may be passed in for red, green, and blue channels. 'count' indicates the number of entries in each table.

The tables are interpolated as needed to generate the number of samples needed by hardware.

Returns an error code.

6.10.39 SetDisplayTransferFormula(formula as CGDisplayTransferFormulaMBS) as Integer

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Set a display gamma/transfer function.

Notes: Set a display gamma/transfer function from a formula specifying min and max values and a gamma for each channel.

Gamma values must be greater than 0.0.

To get an antigamma of 1.6, one would specify a value of (1.0 / 1.6)

Min values must be greater than or equal to 0.0 and less than 1.0.

Max values must be greater than 0.0 and less than or equal to 1.0.

Out of range values, or Max greater than or equal to Min result in a kCGSRangeCheck error.

Values are computed by sampling a function for a range of indices from 0 through 1: $value = Min + ((Max - Min) * pow(index, Gamma))$

The resulting values are converted to a machine specific format and loaded into hardware.

Returns an error code.

6.10.40 SetRotation(angle as Integer) as Integer

Plugin Version: 13.0, Platform: macOS, Targets: Desktop only.

Function: Rescans the bus of displays and changes the rotation of the display.

Example:

```
Var d as CGDisplayMBS = CGDisplayMBS.MainDisplay
```

```
Var e as Integer = d.SetRotation(90)
```

```
if e <> 0 then
```

```
MsgBox "Error: " + str(e)
```

```
end if
```

Notes: Returns IOKit error code. Zero means success.

Lasterror is set.

6.10.41 **SetStereoOperation(stereo as boolean, forceBlueLine as boolean, option as Integer) as Integer**

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Immediately enable or disable stereo operation for a display.

Notes: Note that the system normally detects the presence of a stereo window, and will automatically switch a display containing a stereo window to stereo operation. This function provides a mechanism to force a display to stereo operation, and to set options (such as blue line sync signal) when in stereo operation.

When in stereo operation, a display may need to generate a special stereo sync signal as part of the video output. The sync signal consists of a blue line which occupies the first 25% of the last scanline for the left eye view, and the first 75% of the last scanline for the right eye view. The remainder of the scanline is black. To force the display to generate this sync signal, pass true for forceBlueLine; otherwise, pass false.

Lasterror is set to kCGErrorSuccess on success, or kCGErrorRangeCheck if the display does not support the stereo operation settings requested.

On success, the display resolution, mirroring mode, and available display modes may change due to hardware-specific capabilities and limitations. You should check these settings to verify that they are appropriate for your application.

Lasterror is set.

6.10.42 **ShieldingWindowID as UInt32**

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns window ID of the shield window for the captured display, or 0 if the display is not not shielded.

6.10.43 **ShieldingWindowLevel as Int32**

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Returns the window level of the shield window for the captured display.

6.10.44 ShowCursor as Integer

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Shows the mouse cursor.

Notes: Returns an error code.

Increments hide cursor count.

6.10.45 SwitchToMode(Mode as Dictionary) as Integer

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Switch display mode.

Notes: Note that after switching, display parameters and addresses may change.

The selected display mode persists for the life of the program, and automatically reverts to the permanent setting made by Preferences when the program terminates.

The mode dictionary passed in must be a dictionary vended by other CGDirectDisplay APIs such as BestModeForParameters() and AvailableModes().

Returns an error code.

6.10.46 WaitForBeamPositionOutsideLines(upperScanLine as UInt32, lowerScanLine as UInt32) as Integer

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Wait until the beam position is outside the range specified.

Notes: Wait until the beam position is outside the range specified by upperScanLine and lowerScanLine.

Note that if upperScanLine and lowerScanLine encompass the entire display height, the function returns an error.

lowerScanLine must be greater than or equal to upperScanLine.

Some display systems may not conventional video vertical and horizontal sweep in painting.

These displays report a kCGDisplayRefreshRate of 0 in the CFDictionaryRef returned by CurrentMode().

On such displays, this function returns at once.

Some drivers may not implement support for this mechanism.

On such displays, this function returns at once.

Returns CGDisplayNoErr on success, and an error if display or upperScanLine and lowerScanLine are invalid.

The app should set the values of `upperScanLine` and `lowerScanLine` to allow enough lead time for the drawing operation to complete. A common strategy is to wait for the beam to pass the bottom of the drawing area, allowing almost a full vertical sweep period to perform drawing.

To do this, set `upperScanLine` to 0, and set `lowerScanLine` to the bottom of the bounding box:

```
lowerScanLine = (CGBeamPosition)(cgrect.origin.y + cgrect.size.height);
```

IOKit may implement this as a spin-loop on the beam position call used for `BeamPosition()`. On such system the function is CPU bound, and subject to all the usual scheduling pre-emption. In particular, attempting to wait for the beam to hit a specific scanline may be an exercise in frustration.

These functions are advisory in nature, and depend on IOKit and hardware specific drivers to implement support. If you need extremely precise timing, or access to vertical blanking interrupts, you should consider writing a device driver to tie into hardware-specific capabilities.

Returns an error code.

6.10.47 Properties

6.10.48 BeamPosition as UInt32

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Returns the current beam position on the display.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(d.BeamPosition)
```

Notes: If `display` is invalid, or the display does not implement conventional video vertical and horizontal sweep in painting, or the driver does not implement this functionality, 0 is returned.
(Read only property)

6.10.49 Bounds as CGRectMBS

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Return screen size and origin in global coords.

Notes: Empty rect or nil if display is invalid.

(Read only property)

6.10.50 Brightness as Double

Plugin Version: 15.0, Platform: macOS, Targets: Desktop only.

Function: Get/Set brightness of display.

Notes: Returns -1 if not supported.

Lasterror is set.

(Read and Write property)

6.10.51 CanSetPalette as boolean

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Returns true if the current display mode supports palettes.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(d.CanSetPalette)
```

Notes: (Read only property)

6.10.52 ColorSpace as CGColorSpaceMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the color space for a display.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(d.ColorSpace.Model) // typical 1 = CGColorSpaceMBS.kCGColorSpaceModelRGB
```

Notes: This function returns a display-dependent ICC-based color space. You can use this function when rendering content for a specific display in order to produce color-matched output for that display.

Available in Mac OS X v10.5 and later.

(Read only property)

6.10.53 CurrentMode as Dictionary

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Return a Dictionary describing the current display mode.

Example:

```
Var d as CGDisplayMBS = CGDisplayMBS.MainDisplay
Var dic as Dictionary = d.CurrentMode
```

```
MsgBox dic.Value(d.kCGDisplayWidth)+" x "+dic.Value(d.kCGDisplayHeight)
```

Notes: Returns nil on any error.

(Read only property)

6.10.54 DisplayMode as CGDisplayModeMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Return the current mode of the specified display.

Example:

```
Var d as CGDisplayMBS = CGDisplayMBS.MainDisplay
MsgBox str(d.DisplayMode.Width)+" x "+str(d.DisplayMode.Height)
```

Notes: Returns nil on any error.

Requires Mac OS X 10.6

(Read only property)

6.10.55 DisplayProductNames as Dictionary

Plugin Version: 14.4, Platform: macOS, Targets: Desktop only.

Function: Queries product names for display in all available languages.

Example:

```
Var display as CGDisplayMBS = CGDisplayMBS.MainDisplay
Var names as Dictionary = display.DisplayProductNames
```

```
MsgBox names.Lookup("en_US", "?")
```

Notes: (Read only property)

6.10.56 Handle as Integer

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: The handle to the current display.

Notes: Internally: The DisplayID.

0 is the main display (to keep it easier).

(Read only property)

6.10.57 IOServicePort as Integer

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Return the IOKit service port of a display.

Notes: (Read only property)

6.10.58 IsActive as boolean

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value indicating whether a display is active.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(d.IsActive)
```

Notes: If true, the specified display is active; otherwise, false.

An active display is connected, awake, and available for drawing. In a hardware mirroring set, only the primary display is active.

(Read only property)

6.10.59 IsAlwaysInMirrorSet as boolean

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value indicating whether a display is always in a mirroring set.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(d.IsAlwaysInMirrorSet)
```

Notes: If true, the specified display is in a mirroring set and cannot be removed from this set.

Some hardware configurations support the connection of auxiliary displays that always mirror the main display, and therefore cannot be removed from the mirroring set to which they belong.
(Read only property)

6.10.60 IsAsleep as boolean

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value indicating whether a display is sleeping (and is therefore not drawable.)

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(d.IsAsleep)
```

Notes: If true, the specified display is in sleep mode; otherwise, false.

A display is sleeping when its frame buffer and the attached monitor are in reduced power mode. A sleeping display is still considered to be a part of global display (desktop) space, but it is not drawable.
(Read only property)

6.10.61 IsBuiltin as boolean

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value indicating whether a display is built-in, such as the internal display in portable systems.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(d.IsBuiltin)
```

Notes: If true, the specified display is considered to be a built-in display; otherwise, false.

Portable systems typically identify the internal LCD panel as a built-in display.

Note that it is possible and reasonable for a system to have no displays marked as built-in. For example, a portable system running with the lid closed may report no built-in displays.
(Read only property)

6.10.62 IsInHWMirrorSet as boolean

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value indicating whether a display is in a hardware mirroring set.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(d.IsInHWMirrorSet)
```

Notes: If true, the specified display is a member of a hardware mirroring set; otherwise, false.

When hardware mirroring is enabled, the contents of a double frame buffer are rendered in all displays in the hardware mirroring set.
(Read only property)

6.10.63 IsInMirrorSet as boolean

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value indicating whether a display is in a mirroring set.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(d.IsInMirrorSet)
```

Notes: (Read only property)

6.10.64 IsMain as boolean

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value indicating whether a display is the main display.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(d.IsMain)
```

Notes: If true, the specified display is currently the main display; otherwise, false.
(Read only property)

6.10.65 IsOnline as boolean

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value indicating whether a display is connected or online.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(d.IsOnline)
```

Notes: Returns true if the specified display is connected; otherwise, false.

A display is considered connected or online when the frame buffer hardware is connected to a monitor.

You can use this function to determine if someone has hot-plugged a display to the system. Note that hot-plugging is a hardware feature that may not be present on all displays.
(Read only property)

6.10.66 IsStereo as boolean

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value indicating whether a display is running in a stereo graphics mode.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(d.IsStereo)
```

Notes: If true, the specified display is running in a stereo graphics mode; otherwise, false.

Available in Mac OS X v10.4 and later.
(Read only property)

6.10.67 LastError as Integer

Plugin Version: 15.4, Platform: macOS, Targets: Desktop only.

Function: The last error code.

Notes: Only set for a few properties like brightness where we do have an error code.
(Read only property)

6.10.68 MirrorsDisplay as CGDisplayMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: For a secondary display in a mirroring set, returns the primary display.

Notes: Returns the primary display in the mirroring set. Returns `kCGNullDirectDisplay` if the specified display is actually the primary display or is not in a mirroring set.
(Read only property)

6.10.69 ModelNumber as Integer

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the model number of a display monitor.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(d.ModelNumber)
```

Notes: A model number for the monitor associated with the specified display, or a constant to indicate an exception—see the discussion below.

This function uses I/O Kit to identify the monitor associated with the specified display. The return value depends on the following:

If I/O Kit can identify the monitor, the product ID code for the monitor is returned.

If I/O Kit can't identify the monitor, `kDisplayProductIDGeneric` is returned.

If no monitor is connected, a value of `0xFFFFFFFF` is returned.

(Read only property)

6.10.70 PixelsHigh as Integer

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Height of the display in pixels.

Notes: Var d as new CGDisplayMBS // pick main display

MsgBox str(D.PixelsHigh)
(Read only property)

6.10.71 PixelsWide as Integer

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: Width of the display in pixels.

Example:

Var d as new CGDisplayMBS // pick main display

MsgBox str(D.PixelsWide)

Notes: (Read only property)

6.10.72 PrimaryDisplay as CGDisplayMBS

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the primary display in a hardware mirroring set.

Notes: The primary display in the mirror set. If display is not hardware-mirrored, this function simply returns display.

In hardware mirroring, the contents of a double frame buffer are rendered in two or more displays simultaneously. The mirrored displays are said to be in a hardware mirroring set.

At the discretion of the device driver, one of the displays in a hardware mirroring set is designated as the primary display. The device driver binds the drawing engine, hardware accelerator, and 3D engine to the primary display, and directs all drawing operations to this display.

(Read only property)

6.10.73 RefreshRate as Integer

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Queries refresh rate in Hz for this display.

Example:

```
Var m as CGDisplayMBS = CGDisplayMBS.MainDisplay
MsgBox str(m.RefreshRate)+" Hz"
```

Notes: Returns 60 for LCD displays.

(Read only property)

6.10.74 Rotation as Double

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the rotation angle of a display in degrees.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(d.Rotation)
```

Notes: The rotation angle of the display in degrees, or 0 if the display is not valid.

This function returns the rotation angle of a display in a clockwise direction. For example, if the specified display is rotated clockwise 90 degrees then this function returns 90.0. After a 90 degree clockwise rotation, the physical bottom of the display is on the left side and the physical top is on the right side.

Available in Mac OS X v10.5 and later.

(Read only property)

6.10.75 ScreenSizeHeight as Double

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the height of a display in millimeters.

Example:

```
Var c as new CGDisplayMBS
```

```
Var lines(-1) as string
```

```
Var DPIWidth as Double = c.PixelsWide/(c.ScreenSizeWidth/10.0/2.54)
```

```
Var DPIHeight as Double = c.PixelsHigh/(c.ScreenSizeHeight/10.0/2.54)
```

```
lines.append str(c.ScreenSizeWidth)+" x "+str(c.ScreenSizeHeight)+" Millimeter with"
lines.append str(c.PixelsWide)+" x "+str(c.PixelsHigh)+" Pixel is"
lines.append str(DPIWidth)+" x "+str(DPIHeight)+" DPI"
```

```
MsgBox Join(lines,EndOfLine)
```

Notes: If Extended Display Identification Data (EDID) for the display device is not available, the size is estimated based on the device width and height in pixels from `CGDisplayBounds`, with an assumed resolution of 2.835 pixels/mm or 72 DPI, a reasonable guess for displays predating EDID support. (Read only property)

6.10.76 ScreenSizeWidth as Double

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the width of a display in millimeters.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(d.ScreenSizeHeight) // for example 400 on a 30" Apple Display
```

```
MsgBox str(d.ScreenSizeWidth) // for example 640 on a 30" Apple Display
```

Notes: If Extended Display Identification Data (EDID) for the display device is not available, the size is estimated based on the device width and height in pixels from `CGDisplayBounds`, with an assumed resolution of 2.835 pixels/mm or 72 DPI, a reasonable guess for displays predating EDID support. (Read only property)

6.10.77 SerialNumber as Integer

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the serial number of a display monitor.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(D.SerialNumber)
```

Notes: Returns the serial number for the monitor associated with the specified display, or a constant to indicate an exception—see the discussion below.

This function uses I/O Kit to identify the monitor associated with the specified display.

If I/O Kit can identify the monitor:

If the manufacturer has encoded a serial number for the monitor, the number is returned.
If there is no encoded serial number, 0x00000000 is returned.

If I/O Kit cannot identify the monitor:

If a monitor is connected to the display, 0x00000000 is returned.
If no monitor is connected to the display hardware, a value of 0xFFFFFFFF is returned.

Note that a serial number is meaningful only in conjunction with a specific vendor and product or model.
(Read only property)

6.10.78 UnitNumber as Integer

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the logical unit number of a display.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(D.UnitNumber)
```

Notes: A logical unit number for the specified display.

The logical unit number represents a particular node in the I/O Kit device tree associated with the display's frame buffer. For a particular hardware configuration, this value will not change when the attached monitor is changed.

The unit number will change if the I/O Kit device tree changes, as when hardware is reconfigured, drivers are replaced, or significant changes occur to I/O Kit, so it should not be assumed to be invariant across login sessions.

For more information about I/O Kit, see the Apple publication "I/O Kit Fundamentals".
(Read only property)

6.10.79 UsesOpenGLAcceleration as boolean

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns a Boolean value indicating whether Quartz is using OpenGL-based window acceleration (Quartz Extreme) to render in a display.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(D.UsesOpenGLAcceleration)
```

Notes: Return value: If true, Quartz Extreme is used to render in the specified display; otherwise, false.

Quartz Extreme is an OpenGL-based, hardware-accelerated window compositor available in Mac OS X version 10.2 and later. Quartz Extreme requires a minimum hardware configuration to operate.

The information this function provides is typically used to adjust the demands of drawing operations to the capabilities of the display hardware. For example, an application running on an unaccelerated system could disable live window-resizing.

(Read only property)

6.10.80 VendorNumber as Integer

Plugin Version: 8.6, Platform: macOS, Targets: Desktop only.

Function: Returns the vendor number of the specified display's monitor.

Example:

```
Var d as new CGDisplayMBS // pick main display
```

```
MsgBox str(D.VendorNumber)
// 1552 seems to be Apple
```

Notes: A vendor number for the monitor associated with the specified display, or a constant to indicate an exception—see the discussion below.

This function uses I/O Kit to identify the monitor associated with the specified display.

There are three cases:

If I/O Kit can identify the monitor, the vendor ID is returned.

If I/O Kit cannot identify the monitor, `kDisplayVendorIDUnknown` is returned.

If there is no monitor associated with the display, 0xFFFFFFFF is returned.
(Read only property)

6.10.81 Constants

Capture option flags.

Constant	Value	Description
kCGCaptureNoFill	1	Disables fill with black on capture.
kCGCaptureNoOptions	0	Default behavior.

Keys for mode dictionary

Constant	Value	Description
kCGDisplayBitsPerPixel	"BitsPerPixel"	The number of bits per pixel.
kCGDisplayBitsPerSample	"BitsPerSample"	The value for this key is a number in The number of bits per sample.
kCGDisplayBytesPerRow	"kCGDisplayBytesPerRow"	The value for this key is a number in
kCGDisplayHeight	"Height"	The display height.
kCGDisplayIOFlags	"IOFlags"	The value for this key is a number in
kCGDisplayMode	"Mode"	The value for this key is a number in
kCGDisplayModeIsInterlaced	"kCGDisplayModeIsInterlaced"	This key reflects interesting bits of th
kCGDisplayModeIsSafeForHardware	"kCGDisplayModeIsSafeForHardware"	Set if display mode doesn't need a co
kCGDisplayModeIsStretched	"kCGDisplayModeIsStretched"	This key reflects interesting bits of th
kCGDisplayModeIsTelevisionOutput	"kCGDisplayModeIsTelevisionOutput"	This key reflects interesting bits of th
kCGDisplayModeUsableForDesktopGUI	"UsableForDesktopGUI"	Whether this display can be used for The value for this key is a boolean in
kCGDisplayRefreshRate	"RefreshRate"	The refresh rate.
kCGDisplaySamplesPerPixel	"SamplesPerPixel"	The value for this key is a number in The number of samples per pixel.
kCGDisplayWidth	"Width"	The value for this key is a number in The width of the display.
kCGIODisplayModeID	"IODisplayModeID"	The value for this key is a number in

Keys for options dictionary

Constant	Value	Description
kCGDisplayShowDuplicateLowResolutionModes	"kCGDisplayResolution"	Pass to AllDisplayModes with boolean to in low resolution modes.

6.10.82 Delegates

6.10.83 CreateImageAsyncDelegateMBS(img as CGImageMBS, JPEGData as MemoryBlock, Tag as Variant)

Plugin Version: 15.3, Platform: macOS, Targets: Desktop only.

Function: The delegate for CreateImageAsync method.

6.11 class CGDisplayModeMBS

6.11.1 class CGDisplayModeMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The class for a display mode.

Notes: Requires Mac OS X 10.6

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [MBS Real Studio Plugins, version 12.3pr11](#)
- [MBS Plugins 11.1 Release notes](#)
- [MBS Real Studio Plugins, version 11.1pr8](#)

6.11.2 Methods

6.11.3 Constructor

Plugin Version: 13.1, Platform: macOS, Targets: Desktop only.

Function: The private constructor.

6.11.4 Properties

6.11.5 Handle as Integer

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The internal object reference.

Notes: (Read only property)

6.11.6 Height as Integer

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Return the height in pixels of the specified display mode.

Example:

```
Var display as CGDisplayMBS = CGDisplayMBS.MainDisplay
Var mode as CGDisplayModeMBS = Display.DisplayMode
```

```
MsgBox str(mode.Height)+" Pixel"
```

Notes: Call `CGDisplayModeGetHeight` function.
(Read only property)

6.11.7 IODisplayModeID as Int32

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Return the IOKit display mode ID of the specified display mode.

Notes: Call `CGDisplayModeGetIODisplayModeID` function.
(Read only property)

6.11.8 IOFlags as UInt32

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Return the IOKit flags of the specified display mode.

Example:

```
Var display as CGDisplayMBS = CGDisplayMBS.MainDisplay
Var mode as CGDisplayModeMBS = Display.DisplayMode
```

```
MsgBox str(mode.IOFlags)
```

Notes: Call `CGDisplayModeGetIOFlags` function.
(Read only property)

6.11.9 IsUsableForDesktopGUI as boolean

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Return true if the specified mode is usable for displaying the desktop GUI; false otherwise.

Notes: Call `CGDisplayModeIsUsableForDesktopGUI` function.
(Read only property)

6.11.10 PixelEncoding as string

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Return a string representing the pixel encoding of the specified display mode, expressed as a CFString containing an IOKit graphics mode.

Example:

```
Var display as CGDisplayMBS = CGDisplayMBS.MainDisplay
Var mode as CGDisplayModeMBS = Display.DisplayMode
```

```
MsgBox mode.PixelEncoding // shows e.g. ———RRRRRRRRRGGGGGGGGBBBBBBBB
```

Notes: Call CGDisplayModeCopyPixelEncoding function.
(Read only property)

6.11.11 PixelHeight as Integer

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: Return the height in pixels of the specified display mode.

Notes: Call CGDisplayModeGetPixelHeight function.
(Read only property)

6.11.12 PixelWidth as Integer

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: Return the width in pixels of the specified display mode.

Notes: Call CGDisplayModeGetPixelWidth function.
(Read only property)

6.11.13 RefreshRate as Double

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Return the refresh rate of the specified display mode.

Example:

```
Var display as CGDisplayMBS = CGDisplayMBS.MainDisplay
Var mode as CGDisplayModeMBS = Display.DisplayMode
```

```
MsgBox str(mode.RefreshRate)
```

Notes: Call CGDisplayModeGetRefreshRate function.
(Read only property)

6.11.14 Width as Integer

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: Return the width in pixels of the specified display mode.

Example:

```
Var display as CGDisplayMBS = CGDisplayMBS.MainDisplay
```

```
Var mode as CGDisplayModeMBS = Display.DisplayMode
```

```
MsgBox str(mode.Width)+" Pixel"
```

Notes: Call CGDisplayModeGetWidth function.

(Read only property)

6.12 class CGDisplayReconfigurationEventMBS

6.12.1 class CGDisplayReconfigurationEventMBS

Plugin Version: 11.1, Platform: macOS, Targets: Desktop only.

Function: The class to listen for display configuration changes.

Notes: In constructor the plugin registers for the event and in destructor unregisters. So keep the object reference alive to receive events.

events are invoked when the app is listening for events, on the event processing thread, or from within the display reconfiguration function when in the program that is driving the reconfiguration.

events should avoid changing display configurations, and should not raise exceptions or perform a non-local return such as calling longjmp.

Before display reconfiguration, a event fires to inform applications of a configuration change. The event runs once for each on-line display. The flag is set to kCGDisplayBeginConfigurationFlag. This event does not carry other per-display information, as details of how a reconfiguration affects a particular device rely on device-specific behaviors which may not be exposed by a device driver.

After display reconfiguration, at the time the event function is invoked, all display state reported by CoreGraphics, QuickDraw, and the Carbon Display Manager API will be up to date. This event runs after the Carbon Display Manager notification events. The event runs once for each added, removed, and currently on-line display. Note that in the case of removed displays, calls into the CoreGraphics API with the removed display ID will fail.

Blog Entries

- [MBS Plugins 11.1 Release notes](#)
- [MBS Real Studio Plugins, version 11.1pr8](#)

6.12.2 Events

6.12.3 DisplayReconfiguration(DisplayID as Integer, flags as Integer)

Plugin Version: 11.1, Platform: macOS, Targets: .

Function: This event is invoked whenever the configuration of a local display is changed.

6.12.4 Constants

Possible Flag Values

Constant	Value	Description
kCGDisplayAddFlag	16	Display Added
kCGDisplayBeginConfigurationFlag	1	Begin Configuration for display.
kCGDisplayDesktopShapeChangedFlag	4096	Desktop Shape Changed.
kCGDisplayDisabledFlag	512	Display disabled.
kCGDisplayEnabledFlag	256	Display enabled.
kCGDisplayMirrorFlag	1024	Mirror enabled
kCGDisplayMovedFlag	2	Display Moved
kCGDisplayRemoveFlag	32	Display Removed
kCGDisplaySetMainFlag	4	Set Main
kCGDisplaySetModeFlag	8	Set Mode
kCGDisplayUnMirrorFlag	2048	Mirror disabled

6.13 class CGDisplayStreamEventMBS

6.13.1 class CGDisplayStreamEventMBS

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: CGDisplayStreamEventMBS provides a streaming API for capturing display updates in a real-time manner.

Notes: It can also provide scaling and color space conversion services, as well as allow capturing sub regions of the display.

Requires OS X 10.8.

6.13.2 Methods

6.13.3 Constructor(DisplayHandle as Integer, outputWidth as Integer, outputHeight as Integer, pixelFormat as Integer = 0, properties as dictionary = nil)

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: Creates a new CGDisplayStream.

Notes: This function creates a new CGDisplayStream that is to be used to get a stream of frame updates from a particular display.

DisplayHandle: The CGDirectDisplayID to use as the source for generated frames. (Handle from CGDisplayMBS class)

outputWidth: The output width (in pixels, not points) of the frames to be generated. Must not be zero.

outputHeight: The output height (in pixels, not points) of the frames to be generated. Must not be zero.

pixelFormat: The desired CoreVideo/CoreMedia-style pixel format of the output IOSurfaces. If 0 the plugin will use RGBA.

properties: Any optional properties of the CGDisplayStream.

6.13.4 kCGDisplayStreamColorSpace as String

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: Set the desired CGColorSpace of the output frames.

By default the color space will be that of the display.

6.13.5 kCGDisplayStreamDestinationRect as String

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: This may be used to request where within the destination buffer the display updates should be placed. Use `CGRectCreateDictionaryRepresentation` to convert from a `CGRect` to the value used here.

Note: The coordinate system for the destination rectangle is always specified in output pixels to match the fact that the output buffer size is also specified in terms of pixels.

Defaults to entire buffer

6.13.6 kCGDisplayStreamMinimumFrameTime as String

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: Request that the delta between frame updates be at least as much specified by this value.

Number in seconds, defaults to zero.

6.13.7 kCGDisplayStreamPreserveAspectRatio as String

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: Enable/disable the work the Window Server will do to preserve the display aspect ratio. By default the Window Server will assume that it should preserve the original aspect ratio of the source display rect. If the aspect ratio of the source display and the display stream destination rect are not the same, black borders will be inserted at the top/bottom or right/left sides of the destination in order to preserve the source aspect ratio.

Boolean - defaults to true

6.13.8 kCGDisplayStreamQueueDepth as String

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: Controls how many frames deep the frame queue will be.

Defaults to 3.

6.13.9 kCGDisplayStreamShowCursor as String

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: Controls whether the cursor is embedded within the provided buffers or not.

6.13.10 kCGDisplayStreamSourceRect as String

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: This may be used to request a subregion of the display to be provided as the source of the display stream. Use `CGRectCreateDictionaryRepresentation` to convert from a `CGRect` to the value used here. N

Note: The coordinate system for the source rectangle is specified in display logical coordinates and not in pixels, in order to match the normal convention on HiDPI displays.

Defaults to entire display.

6.13.11 kCGDisplayStreamYCbCrMatrix as String

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: When outputting frames in 420v or 420f format, this key may be used to control which YCbCr matrix is used

The value should be one of the three `kCGDisplayStreamYCbCrMatrix` values specified below:

`kCGDisplayStreamYCbCrMatrix_SMPTE_240M_1995`

`kCGDisplayStreamYCbCrMatrix_ITU_R_709_2`

`kCGDisplayStreamYCbCrMatrix_ITU_R_601_4`

6.13.12 kCGDisplayStreamYCbCrMatrix_ITU_R_601_4 as String

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: A possible value for `kCGDisplayStreamYCbCrMatrix` key.

6.13.13 kCGDisplayStreamYCbCrMatrix_ITU_R_709_2 as String

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: A possible value for `kCGDisplayStreamYCbCrMatrix` key.

6.13.14 `kCGDisplayStreamYCbCrMatrix_SMPTE_240M_1995` as String

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: A possible value for `kCGDisplayStreamYCbCrMatrix` key.

6.13.15 `Start`

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: Begin delivering frame updates to the event.

6.13.16 `Stop`

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: End delivery of frame updates.

Notes: After this call returns, the `CGDisplayStream` callback function will eventually be called with a status of `StatusStopped`. After that point it is safe to release the object.

It is safe to call this function from within the event, but the previous caveat still applies.

6.13.17 `Properties`

6.13.18 `Handle as Integer`

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read only property)

6.13.19 `Lasterror as Integer`

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: The `lasterror` code.

Notes: (Read only property)

6.13.20 Events**6.13.21 FrameAvailable(Status as Integer, displayTime as UInt64, frameSurfaceHandle as Integer, Update as CGDisplayStreamUpdateMBS)**

Plugin Version: 16.1, Platform: macOS, Targets: .

Function: A new frame is available.

Notes: The event used for new frame delivery by CGDisplayStream objects

For each frame that is generated by the WindowServer for a particular display, the user provided event is invoked and provides the user with an IOSurface handle (inside update object), that contains the pixel data for the new frame, as well as a CGDisplayStreamUpdateMBS that contains all of the metadata associated with that IOSurface.

frameSurfaceHandle: The IOSurfaceRef for the current frame. May be zero/nil in some cases. The plugin retains it as part of Update object to keeps it alive.

displayTime: The mach absolute time of when the corresponding frame was to be displayed by the WindowServer

Update: The CGDisplayStreamUpdateMBS for the current frame. will be nil in cases when status is not kCGDisplayStreamFrameStatusFrameComplete.

6.13.22 Constants

Frame Status

Constant	Value	Description
StatusFrameBlank	2	As of displayTime, the display is has gone blank
StatusFrameComplete	0	A new frame has been generated by the Window Server for a particular display at time displayTime.
StatusFrameIdle	1	The Window Server did not generate a new frame for displayTime
StatusStopped	3	The display stream has stopped and no more calls will be made to the handler until the stream is started.

6.14 class CGDisplayStreamUpdateMBS

6.14.1 class CGDisplayStreamUpdateMBS

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: A single frame's extra metadata that describes useful frame delta information.

Notes: A CGDisplayStreamUpdate encapsulates information about what portions of a frame have changed relative to a previously delivered frame. This includes regions that were changed in any way, which ones were actually redrawn, and which regions were merely copied from one place to another.

6.14.2 Methods

6.14.3 getRects(type as Integer) as CGRectMBS()

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: Returns an array of CGRectMBS that describe what parts of the frame have changed relative to the previously delivered frame.

Notes: This rectangle list encapsulates both the update rectangles and movement rectangles.

6.14.4 Properties

6.14.5 CIIImage as Variant

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: Queries the IOSurface with frame as CIIImageMBS.

Notes: Value is a CIIImageMBS. We use variant to reduce dependency to other plugins.
(Read only property)

6.14.6 DeltaX as Double

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: The x component of the movement delta.

Notes: (Read only property)

6.14.7 DeltaY as Double

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: The y component of the movement delta.

Notes: (Read only property)

6.14.8 DropCount as Integer

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: Return how many frames (if any) have been dropped since the last call to the event.

Notes: This call is primarily useful for performance measurement to determine if the client is keeping up with all WindowServer updates.

(Read only property)

6.14.9 Handle as Integer

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read only property)

6.14.10 IOSurfaceHandle as Integer

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: The internal object reference for the IOSurface.

Notes: (Read only property)

6.14.11 Constants

Rectangle Selectors

Constant	Value	Description
UpdateDirtyRects	2	The union of both refreshed and moved rects.
UpdateMovedRects	1	The rectangles that were simply moved from one part of the display to another
UpdateReducedDirtyRects	3	A possibly simplified (but overstated) array of dirty rectangles.
UpdateRefreshedRects	0	The rectangles that were refreshed on the display, not counting moved rectangles

6.15 class CGDisplayTransferFormulaMBS

6.15.1 class CGDisplayTransferFormulaMBS

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: A class for a transfer formula.

6.15.2 Properties

6.15.3 BlueGamma as Double

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: The blue gamma value.

Notes: (Read and Write property)

6.15.4 BlueMax as Double

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: The maximum blue value.

Notes: (Read and Write property)

6.15.5 BlueMin as Double

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: The minimum blue value.

Notes: (Read and Write property)

6.15.6 GreenGamma as Double

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: The green gamma value.

Notes: (Read and Write property)

6.15.7 GreenMax as Double

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: The maximum green value.

Notes: (Read and Write property)

6.15.8 GreenMin as Double

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: The minimum green value.

Notes: (Read and Write property)

6.15.9 RedGamma as Double

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: The red gamma value.

Notes: (Read and Write property)

6.15.10 RedMax as Double

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: The maximum red value.

Notes: (Read and Write property)

6.15.11 RedMin as Double

Plugin Version: 3.3, Platform: macOS, Targets: Desktop only.

Function: The minimum red value.

Notes: (Read and Write property)

6.16 class CGFontMBS

6.16.1 class CGFontMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The CoreGraphics class for a font.

Notes: The CGFontRef opaque type encapsulates font information. A font is a set of shapes or glyphs associated with a character set. A glyph can represent a single character (such as ,Äòb,Äô), more than one character (such as the "Ö" Å" ligature), or a special character such as a space. Quartz retrieves the glyphs for the font from ATS (Apple Type Services) and paints the glyphs based on the relevant parameters of the current graphics state.

Quartz provides a limited, low-level interface for drawing text. For information on text-drawing functions, see CGContext Reference. For full Unicode and text-layout support, use the services provided by Core Text or ATSUI).

Blog Entries

- [MBS Xojo Plugins, version 22.5pr1](#)
- [MBS Xojo / Real Studio plug-ins in version 14.2](#)
- [MBS Xojo / Real Studio Plugins, version 14.2pr7](#)

Xojo Developer Magazine

- [12.4, page 9: News](#)

6.16.2 Methods

6.16.3 CreateWithDataProvider(CGDataProvider as Variant) as CGFontMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates a font object from data supplied from a data provider.

Notes: Dataprovider must be a CGDataProviderMBS object.

Returns the font object or nil if the font can't be created.

Before drawing text in a Quartz context, you must set the font in the current graphics state by calling the function CGContextMBS.SetFontSize.

6.16.4 CreateWithFontName(name as string) as CGFontMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates a font object corresponding to the font specified by a PostScript or full name.

Example:

```
Var c as CGFontMBS = CGFontMBS.CreateWithFontName("Courier")
MsgBox c.FullName
```

Notes: Returns the font object or nil if the font can't be created.

Before drawing text in a Quartz context, you must set the font in the current graphics state by calling the function `CGContextMBS.SetFont`.

6.16.5 CreateWithPlatformFont(ATSTFontHandle as Integer) as CGFontMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates a new font reference from an `ATSTFontRef`.

Notes: Create a `CGFont` using `platformFontReference`, a handle to a platform-specific font reference. For MacOS X, `platformFontReference` should be a handle to an `ATSTFontRef`.

6.16.6 Properties

6.16.7 Ascent as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the ascent of a font.

Example:

```
Var c as CGFontMBS = CGFontMBS.CreateWithFontName("Times")
MsgBox str(c.Ascent)
```

Notes: The ascent is the maximum distance above the baseline of glyphs in a font. The value is specified in glyph space units.

(Read only property)

6.16.8 CapHeight as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the cap height of a font.

Example:

```
Var c as CGFontMBS = CGFontMBS.CreateWithFontName("Times")
MsgBox str(c.CapHeight)
```

Notes: The cap height is the distance above the baseline of the top of flat capital letters of glyphs in a font. The value is specified in glyph space units.
(Read only property)

6.16.9 Descent as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the descent of a font.

Example:

```
Var c as CGFontMBS = CGFontMBS.CreateWithFontName("Times")
MsgBox str(c.Descent)
```

Notes: The descent is the maximum distance below the baseline of glyphs in a font. The value is specified in glyph space units.
(Read only property)

6.16.10 FontBBox as CGRectMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the bounding box of a font.

Example:

```
Var c as CGFontMBS = CGFontMBS.CreateWithFontName("Times")
Var r as CGRectMBS = c.FontBBox
MsgBox str(r.Width)+" "+str(r.Height)
```

Notes: The font bounding box is the union of all of the bounding boxes for all the glyphs in a font. The value is specified in glyph space units.
(Read only property)

6.16.11 FullName as String

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the full name associated with a font object.

Example:

```
Var c as CGFontMBS = CGFontMBS.CreateWithFontName("Courier")
```

MsgBox c.FullName

Notes: (Read only property)

6.16.12 Handle as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

6.16.13 ItalicAngle as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the italic angle of a font.

Example:

```
Var c as CGFontMBS = CGFontMBS.CreateWithFontName("Times")
MsgBox str(c.ItalicAngle)
```

Notes: The italic angle of the font, measured in degrees counter-clockwise from the vertical.
(Read only property)

6.16.14 Leading as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the leading of a font.

Example:

```
Var c as CGFontMBS = CGFontMBS.CreateWithFontName("Times")
MsgBox str(c.Leading)
```

Notes: The leading is the spacing between consecutive lines of text in a font. The value is specified in glyph space units.
(Read only property)

6.16.15 NumberOfGlyphs as UInt64

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the number of glyphs in a font.

Example:

```
Var c as CGFontMBS = CGFontMBS.CreateWithFontName("Times")
MsgBox str(c.NumberOfGlyphs)
```

Notes: (Read only property)

6.16.16 PostScriptName as String

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Obtains the PostScript name of a font.

Example:

```
Var c as CGFontMBS = CGFontMBS.CreateWithFontName("Courier")
MsgBox c.PostScriptName
```

Notes: (Read only property)

6.16.17 StemV as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the thickness of the dominant vertical stems of glyphs in a font.

Example:

```
Var c as CGFontMBS = CGFontMBS.CreateWithFontName("Times")
MsgBox str(c.StemV)
```

Notes: The thickness of the dominant vertical stems of glyphs in a font.
(Read only property)

6.16.18 UnitsPerEm as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the number of glyph space units per em for the provided font.

Example:

```
Var c as CGFontMBS = CGFontMBS.CreateWithFontName("Times")
MsgBox str(c.UnitsPerEm)
```

Notes: The number of glyph space units per em for the provided font.
(Read only property)

6.16.19 XHeight as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the x-height of a font.

Example:

```
Var c as CGFontMBS = CGFontMBS.CreateWithFontName("Times")
MsgBox str(c.XHeight)
```

Notes: The x-height is the distance above the baseline of the top of flat, non-ascending lowercase letters (such as x) of glyphs in a font. The value is specified in glyph space units.
(Read only property)

6.16.20 Constants

Constants

Constant	Value	Description
kCGFontIndexInvalid	65535	An invalid font index (a value which never represents a valid glyph).
kCGFontIndexMax	65534	The maximum allowed value for font index.
kCGGlyphMax	65534	The maximum value for a glyph.

PostScript Formats

Constant	Value	Description
kCGFontPostScriptFormatType1	1	This is documented in Adobe Type 1 Font Format, which is available from http://partners.adobe.com/ .
kCGFontPostScriptFormatType3	3	This is documented in PostScript Language Reference, 3rd edition, which is available from http://partners.adobe.com/ .
kCGFontPostScriptFormatType42	42	This is documented in Adobe Technical Note 5012, The Type 42 Font Format Specification, which is available from http://partners.adobe.com/ .

6.17 class CGFunctionMBS

6.17.1 class CGFunctionMBS

Plugin Version: 6.2, Platform: macOS, Targets: All.

Function: CGFunction provides a general facility for defining and using callback functions that take an arbitrary number of floating-point input values, and pass back an arbitrary number of floating-point output values.

Blog Entries

- [MBS Real Studio Plugins, version 13.1fc1](#)

6.17.2 Methods

6.17.3 Create(domainDimension as Integer, domain as memoryblock, rangeDimension as Integer, range as memoryblock)

Plugin Version: 6.2, Platform: macOS, Targets: All.

Function: Creates a Quartz function.

Notes: Parameters

domainDimension: The number of inputs.

domain: An array of ($2 * \text{domainDimension}$) floats used to specify the valid intervals of input values. For each k from 0 to ($\text{domainDimension} - 1$), $\text{domain} [2 * k]$ must be less than or equal to $\text{domain} [2 * k + 1]$, and the k th input value will be clipped to lie in the interval $\text{domain} [2 * k] , \hat{a} \text{ input} [k] , \hat{a} \text{ domain} [2 * k + 1]$. If this parameter is nil, then the input values are not clipped.

rangeDimension: The number of outputs.

range: An array of ($2 * \text{rangeDimension}$) floats that specifies the valid intervals of output values. For each k from 0 to ($\text{rangeDimension} - 1$), $\text{range} [2 * k]$ must be less than or equal to $\text{range} [2 * k + 1]$, and the k th output value will be clipped to lie in the interval $\text{range} [2 * k] , \hat{a} \text{ output} [k] , \hat{a} \text{ range} [2 * k + 1]$. If this parameter is nil, then the output values are not clipped.

Returns the new Quartz function or nil on any error.

Available in Mac OS X version 10.2 and later.

6.17.4 Properties

6.17.5 Handle as Integer

Plugin Version: 6.2, Platform: macOS, Targets: All.

Function: Handle to the CGFunctionRef.

Notes: (Read and Write property)

6.17.6 Events

6.17.7 Evaluate(Input as memoryblock, Output as memoryblock)

Plugin Version: 6.2, Platform: macOS, Targets: .

Function: Asks the CGFunction to calculate the values.

Notes: memoryblocks must be big enough.

6.18 class CGGradientMBS

6.18.1 class CGGradientMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: A gradient defines a smooth transition between colors across an area.

Notes: The CGGradientMBS class, and the functions that operate on it, make creating and using radial and axial gradient fills an easy task. A CGGradient object has a color space, two or more colors, and a location for each color. The color space cannot be a pattern or indexed color space, otherwise it can be any Quartz color space (CGColorSpaceMBS).

Colors can be provided as component values (such as red, green, blue) or as Quartz color objects (CGColorMBS). In Quartz, component can vary from 0.0 to 1.0, designating the proportion of the component present in the color.

A location is a normalized value. When it comes time to paint the gradient, Quartz maps the normalized location values to the points in coordinate space that you provide.

If you want more precise control over gradients, or if your application runs in versions of Mac OS X that are earlier than v10.5, see CGShadingMBS.

Requires Mac OS X 10.5.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 14.4pr6](#)
- [MBS Real Studio Plugins, version 13.1fc1](#)
- [MBS REALbasic Plugins, version 11.2pr1](#)

6.18.2 Methods

6.18.3 CreateWithColorComponents(colorSpace as CGColorSpaceMBS, components() as Double) as CGGradientMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Creates a CGGradient object from a color space and the provided color components and locations.

Notes: space: The color space to use for the gradient. You cannot use a pattern or indexed color space.
components: The color components for each color that defines the gradient. The components should be in the color space specified by space. If you are unsure of the number of components, you can call the function CGColorSpaceMBS.NumberOfComponents.

The number of items in this array should be the product of count and the number of components in the color space. For example, if the color space is an RGBA color space and you want to use two colors in the gradient (one for a starting location and another for an ending location), then you need to provide 8 values in components—red, green, blue, and alpha values for the first color, followed by red, green, blue, and alpha values for the second color.

locations: Optional, The location for each color provided in components. Each location must be a CGFloat value in the range of 0 to 1, inclusive. If 0 and 1 are not in the locations array, Quartz uses the colors provided that are closest to 0 and 1 for those locations.

If locations is not passed, the first color in colors is assigned to location 0, the last color in colors is assigned to location 1, and intervening colors are assigned locations that are at equal intervals in between.

Available in Mac OS X v10.5 and later.

See also:

- 6.18.4 CreateWithColorComponents(colorSpace as CGColorSpaceMBS, components() as Double, locations() as Double) as CGGradientMBS 249

6.18.4 CreateWithColorComponents(colorSpace as CGColorSpaceMBS, components() as Double, locations() as Double) as CGGradientMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Creates a CGGradient object from a color space and the provided color components and locations.

Example:

```
// put into paint event of window
Var c as CGContextMBS
if TargetCocoa then
c = GetCurrentCGContextMBS
else
c = window1.CGContextMBS
end if

Var locations() as Double = array(0.0, 0.6, 1.0)
Var components() as Double = array(1.0, 0.0, 0.0, 0.35, 0.0, 1.0, 0.0, 0.2, 0.0, 0.0, 1.0, 0.06) // Start color
and End color

Var rgbColorspace as CGColorSpaceMBS = CGColorSpaceMBS.CreateDeviceRGB
Var glossGradient as CGGradientMBS = CGGradientMBS.CreateWithColorComponents(rgbColorspace,
components, locations)
'dim glossGradient as CGGradientMBS = CGGradientMBS.CreateWithColorComponents(rgbColorspace,
components)

Var currentBounds as CGRectMBS = CGMakeRectMBS(0,0,g.Width,g.Height)
```

```

Var topCenter as CGPointMBS = CGMakePointMBS(g.width/2, 0.0)
Var midCenter as CGPointMBS = CGMakePointMBS(g.width/2, g.height)

c.SaveGState

// fill white
c.SetRGBFillColor 1.0, 1.0, 1.0, 1.0
c.FillRect currentBounds

// draw gradient
c.AddRect(currentBounds)
c.Clip
c.DrawLinearGradient(glossGradient, topCenter, midCenter, 0)
c.RestoreGState
c.Flush

```

Notes: *space*: The color space to use for the gradient. You cannot use a pattern or indexed color space.
components: The color components for each color that defines the gradient. The components should be in the color space specified by *space*. If you are unsure of the number of components, you can call the function `CGColorSpaceMBS.NumberOfComponents`.

The number of items in this array should be the product of *count* and the number of components in the color space. For example, if the color space is an RGBA color space and you want to use two colors in the gradient (one for a starting location and another for an ending location), then you need to provide 8 values in *components*—red, green, blue, and alpha values for the first color, followed by red, green, blue, and alpha values for the second color.

locations: Optional, The location for each color provided in *components*. Each location must be a `CGFloat` value in the range of 0 to 1, inclusive. If 0 and 1 are not in the *locations* array, Quartz uses the colors provided that are closest to 0 and 1 for those locations.

If *locations* is not passed, the first color in *colors* is assigned to location 0, the last color in *colors* is assigned to location 1, and intervening colors are assigned locations that are at equal intervals in between.

Available in Mac OS X v10.5 and later.

See also:

- 6.18.3 `CreateWithColorComponents(colorSpace as CGColorSpaceMBS, components() as Double) as CGGradientMBS` 248

6.18.5 CreateWithColors(colorSpace as CGColorSpaceMBS, colors() as CGColorMBS) as CGGradientMBS

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Creates a CGGradient object from a color space and the provided color objects and locations.

Example:

```
// put into window paint event
Var c as CGContextMBS
if TargetCocoa then
c = GetCurrentCGContextMBS
else
c = window1.CGContextMBS
end if

Var colors(-1) as CGColorMBS

colors.Append CGColorMBS.CreateGenericRGB(1.0, 0.0, 0.0, 0.35)
colors.Append CGColorMBS.CreateGenericRGB(0.0, 1.0, 0.0, 0.2)
colors.Append CGColorMBS.CreateGenericRGB(0.0, 0.0, 1.0, 0.06)

Var glossGradient as CGGradientMBS = CGGradientMBS.CreateWithColors(nil, colors)

Var topCenter as CGPointMBS = CGMakePointMBS(g.width/2, 0.0)
Var midCenter as CGPointMBS = CGMakePointMBS(g.width/2, g.height)

// fill white
c.SetRGBFillColor 1.0, 1.0, 1.0, 1.0
c.FillRect CGMakeRectMBS(0,0,g.Width,g.Height)

// gradient
Var currentBounds as CGRectMBS = CGMakeRectMBS(0,0,g.Width,g.Height)
c.SaveGState
c.AddRect(currentBounds)
c.Clip
c.DrawLinearGradient(glossGradient, topCenter, midCenter, 0)
c.RestoreGState

c.Flush
```

Notes: space: The color space to use for the gradient. You cannot use a pattern or indexed color space. Can be nil.

colors: A non-empty array of CGColorMBS objects that should be in the color space specified by space. If space is not nil, each color will be converted (if necessary) to that color space and the gradient will drawn in that color space. Otherwise, each color will be converted to and drawn in the GenericRGB color space.

locations: Optional. The location for each color provided in colors; each location must be a CGFloat value in the range of 0 to 1, inclusive. If 0 and 1 are not in the locations array, Quartz uses the colors provided

that are closest to 0 and 1 for those locations.

If `locations` is missing, the first color in `colors` is assigned to location 0, the last color in `colors` is assigned to location 1, and intervening colors are assigned locations that are at equal intervals in between. The `locations` array should contain the same number of items as the `colors` array.

Available in Mac OS X v10.5 and later.

See also:

- 6.18.6 `CreateWithColors(colorSpace as CGColorSpaceMBS, colors() as CGColorMBS, locations() as Double) as CGGradientMBS` 252

6.18.6 `CreateWithColors(colorSpace as CGColorSpaceMBS, colors() as CGColorMBS, locations() as Double) as CGGradientMBS`

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Creates a `CGGradient` object from a color space and the provided color objects and locations.

Example:

```
// put into window paint event
Var c as CGContextMBS
if TargetCocoa then
c = GetCurrentCGContextMBS
else
c = window1.CGContextMBS
end if

Var locations() as Double = array(0.0, 0.8, 1.0)

Var colors(-1) as CGColorMBS

colors.Append CGColorMBS.CreateGenericRGB(1.0, 0.0, 0.0, 0.35)
colors.Append CGColorMBS.CreateGenericRGB(0.0, 1.0, 0.0, 0.2)
colors.Append CGColorMBS.CreateGenericRGB(0.0, 0.0, 1.0, 0.06)

Var glossGradient as CGGradientMBS = CGGradientMBS.CreateWithColors(nil, colors, locations)

Var topCenter as CGPointMBS = CGMakePointMBS(g.width/2, 0.0)
Var midCenter as CGPointMBS = CGMakePointMBS(g.width/2, g.height)

// fill white
c.SetRGBFillColor 1.0, 1.0, 1.0, 1.0
c.FillRect CGMakeRectMBS(0,0,g.Width,g.Height)

// gradient
```

```

Var currentBounds as CGRectMBS = CGMakeRectMBS(0,0,g.Width,g.Height)
c.SaveGState
c.AddRect(currentBounds)
c.Clip
c.DrawLineGradient(glossGradient, topCenter, midCenter, 0)
c.RestoreGState

c.Flush

```

Notes: space: The color space to use for the gradient. You cannot use a pattern or indexed color space. Can be nil.

colors: A non-empty array of CGColorMBS objects that should be in the color space specified by space. If space is not nil, each color will be converted (if necessary) to that color space and the gradient will drawn in that color space. Otherwise, each color will be converted to and drawn in the GenericRGB color space.

locations: Optional. The location for each color provided in colors; each location must be a CGFloat value in the range of 0 to 1, inclusive. If 0 and 1 are not in the locations array, Quartz uses the colors provided that are closest to 0 and 1 for those locations.

If locations is missing, the first color in colors is assigned to location 0, the last color in colors is assigned to location 1, and intervening colors are assigned locations that are at equal intervals in between.

The locations array should contain the same number of items as the colors array.

Available in Mac OS X v10.5 and later.

See also:

- 6.18.5 CreateWithColors(colorSpace as CGColorSpaceMBS, colors() as CGColorMBS) as CGGradientMBS 251

6.18.7 Properties

6.18.8 Handle as Integer

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Handle to the CGGradientRef.

Notes: (Read and Write property)

6.18.9 Constants

Gradient Drawing Options

Constant	Value	Description
<code>kCGGradientDrawsAfterEndLocation</code>	2	The fill should extend beyond the ending location. The color that is drawn beyond the ending point is the solid color defined by the <code>CGGradient</code> to be at location 1. Available in Mac OS X v10.5 and later.
<code>kCGGradientDrawsBeforeStartLocation</code>	1	The fill should extend beyond the starting location. The color that is drawn beyond the starting point is the solid color defined by the <code>CGGradient</code> to be at location 0. Available in Mac OS X v10.5 and later.

6.19 class CGImageDestinationMBS

6.19.1 class CGImageDestinationMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: A class to write CGImages.

Example:

```

Var logo as Picture = LogoMBS(500)
Var image as CGImageMBS = CGCreateImageMBS(logo)

Var dic as new Dictionary

// 10%
'dic.Value(CGImageDestinationMBS.kCGImageDestinationLossyCompressionQuality)=0.1

// 100%
dic.Value(CGImageDestinationMBS.kCGImageDestinationLossyCompressionQuality)=1.0

Var file as FolderItem = SpecialFolder.Desktop.Child("logo.jpg")
Var d as new CGImageDestinationMBS(file, "public.jpeg", 1)

d.AddImage(image, dic)

if d.Finalize then
MsgBox "Saved"
else
MsgBox "Failed to save."
end if

```

Notes: CGImageDestination objects, available in Mac OS X v10.4 or later, abstract the data-writing task. An image destination can represent a single image or multiple images. It can contain thumbnail images as well as properties for each image.

The functions described in this reference can write data to three kinds of destinations: a file, an URL and a string. After creating a CGImageDestination object for the appropriate destination, you can add image data and set image properties. When you are finished adding data, call the function Finalize to write the image data and properties.

Blog Entries

- [Several ways for picture to PDF in MBS Plugins](#)
- [Multithreaded plugin functions can increase speed of Xojo application](#)
- [Four ways to save picture as Tiff in Xojo](#)
- [Problems with killing Xojo threads with plugin calls.](#)

- **Tip of the day:** Edit IPTC data in JPEG file with `CGImageSource/CGImageDestination`
- `MBS Xojo / Real Studio Plugins`, version 13.5pr8
- `MBS REALbasic plug-ins` version 9.5

6.19.2 Methods

6.19.3 `AddImage(image as CGImageMBS, properties as dictionary=nil)`

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Adds an image to an image destination.

Example:

```

Var inputFile as FolderItem = SpecialFolder.Desktop.Child("test.jpg")

// reading the picture
Var c as new CGImageSourceMBS(inputFile)
Var img as CGImageMBS = c.CreateImageAtIndex(0)
Var propertiesGlobal as Dictionary = c.Properties
Var propertiesImage as Dictionary = c.PropertiesAtIndex(0)

Var outputFile as FolderItem = SpecialFolder.Desktop.Child("output.jpg")
Var d as new CGImageDestinationMBS(outputFile, "public.jpeg", 1)

// writing the picture and include metadata
d.SetProperties(propertiesGlobal)
d.AddImage(img, propertiesImage)
if d.FinalizeMT then
outputFile.Launch
else
MsgBox "Failed to write jpeg."
end if

```

Notes: `image`: The image to add.

`properties`: An optional dictionary that specifies the properties of the added image.

For properties you can use those in the `CGImageSourceMBS` class, `kCGImageDestinationLossyCompressionQuality` and `kCGImageDestinationBackgroundColor`.

The function logs an error if you add more images than what you specified when you created the image destination.

Available in Mac OS X version 10.4 and later.

6.19.4 AddImageCF(image as CGImageMBS, properties as Variant = nil)

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Adds an image to an image destination.

Notes: image: The image to add.

properties: An optional dictionary or CFDictionaryMBS that specifies the properties of the added image.

For properties you can use those in the CGImageSourceMBS class, kCGImageDestinationLossyCompressionQuality and kCGImageDestinationBackgroundColor.

The function logs an error if you add more images than what you specified when you created the image destination.

Available in Mac OS X version 10.4 and later.

6.19.5 AddImageFromSource(source as CGImageSourceMBS, index as Integer, options as dictionary = nil)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Adds an image from an image source to an image destination.

Notes: source: An image source.

index: An index that specifies the location of the image in the image source. The index is zero-based.

properties: A dictionary that specifies properties to overwrite or add to the source image properties.

Available in Mac OS X version 10.4 and later.

6.19.6 AddImageFromSourceCF(source as CGImageSourceMBS, index as Integer, options as Variant = nil)

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Adds an image from an image source to an image destination.

Notes: source: An image source.

index: An index that specifies the location of the image in the image source. The index is zero-based.

properties: A dictionary or CFDictionaryMBS that specifies properties to overwrite or add to the source image properties.

Available in Mac OS X version 10.4 and later.

6.19.7 Constructor(file as folderitem, type as string, count as Integer = 1)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates an image destination that writes to a location specified by a folderitem.

Example:

```

Var p As Picture = LogoMBS(500)

// save as Photoshop

Var image As CGImageMBS = CGImageMBS.CreateImage(p)
Var file1 As FolderItem = SpecialFolder.Desktop.Child("test1.psd")
Var dest As New CGImageDestinationMBS(file1, "com.adobe.photoshop-image", 1)

dest.AddImage(image, Nil)

If dest.Finalize Then
// MsgBox "Saved"
Else
MsgBox "Failed to save."
End If

```

Notes: file: The file to write to. If the file already exists, the data at this location is overwritten.

type: The UTI (uniform type identifier) of the resulting image file. See Uniform Type Identifiers Overview for a list of system-declared and third-party UTIs.

count: The number of images (not including thumbnail images) that the image file will contain.

On success the handle value is not zero.

Available in Mac OS X version 10.4 and later.

See also:

- 6.19.8 Constructor(type as string, count as Integer = 1) 258
- 6.19.9 Constructor(url as string, type as string, count as Integer = 1) 259

6.19.8 Constructor(type as string, count as Integer = 1)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates an image destination that writes to a string.

Notes: type: The uniform type identifier (UTI) of the resulting image file. See Uniform Type Identifiers

Overview for a list of system-declared and third-party UTIs.

count: The number of images (not including thumbnail images) that the image file will contain.

On success the handle is not zero.

Available in Mac OS X version 10.4 and later.

See also:

- 6.19.7 Constructor(file as folderitem, type as string, count as Integer = 1) 258
- 6.19.9 Constructor(url as string, type as string, count as Integer = 1) 259

6.19.9 Constructor(url as string, type as string, count as Integer = 1)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates an image destination that writes to a location specified by a URL.

Notes: url: The URL to write to. If the URL already exists, the data at this location is overwritten.

type: The UTI (uniform type identifier) of the resulting image file. See Uniform Type Identifiers Overview for a list of system-declared and third-party UTIs.

count: The number of images (not including thumbnail images) that the image file will contain.

On success the handle value is not zero.

Available in Mac OS X version 10.4 and later.

See also:

- 6.19.7 Constructor(file as folderitem, type as string, count as Integer = 1) 258
- 6.19.8 Constructor(type as string, count as Integer = 1) 258

6.19.10 CreateWithData(type as string, count as Integer = 1) as CGImageDestinationMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates an image destination that writes to a string.

Notes: type: The uniform type identifier (UTI) of the resulting image file. See Uniform Type Identifiers Overview for a list of system-declared and third-party UTIs.

count: The number of images (not including thumbnail images) that the image file will contain.

On error the function returns nil.

Available in Mac OS X version 10.4 and later.

6.19.11 CreateWithFile(file as folderitem, type as string, count as Integer = 1) as CGImageDestinationMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates an image destination that writes to a location specified by a folderitem.

Example:

```

Var p As Picture = LogoMBS(500)

// save as tiff

Var image As CGImageMBS = CGImageMBS.CreateImage(p)
Var file2 As FolderItem = SpecialFolder.Desktop.Child("test2.tif")
Var dest As CGImageDestinationMBS = CGImageDestinationMBS.CreateWithFile(file2, "public.tiff", 1)

dest.AddImage(image, Nil)

If dest.Finalize Then
// MsgBox "Saved"
Else
MsgBox "Failed to save."
End If

```

Notes: file: The file to write to. If the file already exists, the data at this location is overwritten.

type: The UTI (uniform type identifier) of the resulting image file. See Uniform Type Identifiers Overview for a list of system-declared and third-party UTIs.

count: The number of images (not including thumbnail images) that the image file will contain.

On failure the function returns nil.

Available in Mac OS X version 10.4 and later.

6.19.12 CreateWithURL(url as string, type as string, count as Integer = 1) as CGImageDestinationMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates an image destination that writes to a location specified by a URL.

Notes: url: The URL to write to. If the URL already exists, the data at this location is overwritten.

type: The UTI (uniform type identifier) of the resulting image file. See Uniform Type Identifiers Overview for a list of system-declared and third-party UTIs.

count: The number of images (not including thumbnail images) that the image file will contain.

On error nil is returned.

Available in Mac OS X version 10.4 and later.

6.19.13 Data as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Makes a copy of the data.

Notes: The data is collected and after you called Finalize you can pick the data here.

6.19.14 Finalize as boolean

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Writes image data and properties to the data or URL associated with the image destination.

Example:

```
Var logo as Picture = LogoMBS(500)
Var image as CGImageMBS = CGCreateImageMBS(logo)

Var file as FolderItem = SpecialFolder.Desktop.Child("logo.png")
Var d as new CGImageDestinationMBS(file, "public.png", 1)

d.AddImage(image, nil)

if d.Finalize then
  MsgBox "Saved"
else
  MsgBox "Failed to save."
```

end if

Notes: Returns true if the image is successfully written; false otherwise.

You must call this function or the output of the image destination will not be valid. After calling this function, no additional data can be added to the image destination.

Available in Mac OS X version 10.4 and later.

6.19.15 FinalizeMT as boolean

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: Writes image data and properties to the data or URL associated with the image destination.

Notes: Returns true if the image is successfully written; false otherwise.

You must call this function or the output of the image destination will not be valid. After calling this function, no additional data can be added to the image destination.

Available in Mac OS X version 10.4 and later.

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.

6.19.16 kCGImageDestinationBackgroundColor as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the constant for the image destination properties.

Notes: The desired background color to composite against when writing an image that has an alpha component to a destination format that does not support alpha. If present, the value associated with this key must be a CGColorRef data type without an alpha component of its own. If not present, and if a background color is needed, a white color is used.

Available in Mac OS X v10.4 and later.

6.19.17 kCGImageDestinationDateTime as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the constant for the image destination properties.

Notes: Updates the DateTime parameters of the image metadata. Only values present in the original image will be updated. If present, the value should be a String or a Date. If String, the value must be in Exif DateTime or ISO 8601 DateTime format. This option is mutually exclusive with kCGImageDestinationMetadata.

6.19.18 kCGImageDestinationLossyCompressionQuality as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the constant for the image destination properties.

Notes: The desired compression quality to use when writing to an image destination. If present, the value associated with this key must be a double in the range 0.0 to 1.0. A value of 1.0 specifies to use lossless compression if destination format supports it. A value of 0.0 implies to use maximum compression.

Available in Mac OS X v10.4 and later.

6.19.19 kCGImageDestinationMergeMetadata as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the constant for the image destination properties.

Example:

```
// Change rotation in an image file

// files
Var dpath as folderitem = SpecialFolder.Desktop.Child("test.jpg")
Var opath as folderitem = dpath.parent.Child("output.jpg")

// open source
Var imageSource as new cgimagesourcembs(dpath)

// read image
Var img as cgimagembs = imageSource.createimageatindex(0)

// global properties
Var globalprop as dictionary = imageSource.properties
// per image properties
Var p as Dictionary = imageSource.PropertiesAtIndex(0)

Var imageDest as new CGImageDestinationMBS(opath,"public.jpeg",1)
```

```

'const orientation = 1 // top left
const orientation = 3 // bottom right

// set globals
imageDest.SetProperties(globalprop)

// now set new orientation
p.value(imageDest.kCGImageDestinationOrientation) = orientation
p.value(imageDest.kCGImageDestinationMergeMetadata) = true

// change tiff dictionary, if present
Var dTIFF as Dictionary = p.lookup(imageSource.kCGImagePropertyTIFFDictionary, nil)
if dTIFF <> nil then
dTIFF.value(imageSource.kCGImagePropertyTIFFOrientation) = orientation
end if

// change iptc dictionary, if present
Var dIPTC as Dictionary = p.lookup(imageSource.kCGImagePropertyIPTCDictionary, nil)
if dIPTC <> nil then
dIPTC.value(imageSource.kCGImagePropertyIPTCImageOrientation) = orientation
end if

// write out image
imageDest.AddImage(img,p)
call imageDest.Finalize

```

Notes: If true, The metadata will be copied from the source and merged with the tags specified in `kCGImageDestinationMetadata`. If a tag does not exist in the source, it will be added. If the tag exists in the source, it will be updated. A metadata tag can be removed by setting the tag's value to `nil`. If present, the value of this key is a Boolean. The default is `False`.

6.19.20 `kCGImageDestinationMetadata` as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the constant for the image destination properties.

Notes: Set the metadata tags for the image destination. If present, the value of this key is a `CGImageMetadata`. By default, all EXIF, IPTC, and XMP tags will be replaced. Use `kCGImageDestinationMergeMetadata` to merge the tags with the existing tags in the image source.

6.19.21 `kCGImageDestinationOrientation` as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the constant for the image destination properties.

Example:

```
// Change rotation in an image file

// files
Var dpath as folderitem = SpecialFolder.Desktop.Child("test.jpg")
Var opath as folderitem = dpath.parent.Child("output.jpg")

// open source
Var imageSource as new cgimagesourcembs(dpath)

// read image
Var img as cgimagembs = imageSource.createimageatindex(0)

// global properties
Var globalprop as dictionary = imageSource.properties
// per image properties
Var p as Dictionary = imageSource.PropertiesAtIndex(0)

Var imageDest as new CGImageDestinationMBS(opath,"public.jpeg",1)

'const orientation = 1 // top left
const orientation = 3 // bottom right

// set globals
imageDest.SetProperties(globalprop)

// now set new orientation
p.value(imageDest.kCGImageDestinationOrientation) = orientation
p.value(imageDest.kCGImageDestinationMergeMetadata) = true

// change tiff dictionary, if present
Var dTIFF as Dictionary = p.lookup(imageSource.kCGImagePropertyTIFFDictionary, nil)
if dTIFF <> nil then
dTIFF.value(imageSource.kCGImagePropertyTIFFOrientation) = orientation
end if

// change iptc dictionary, if present
Var dIPTC as Dictionary = p.lookup(imageSource.kCGImagePropertyIPTCDictionary, nil)
if dIPTC <> nil then
dIPTC.value(imageSource.kCGImagePropertyIPTCImageOrientation) = orientation
end if

// write out image
imageDest.AddImage(img,p)
call imageDest.Finalize
```

Notes: Updates the orientation in the image metadata. The image data itself will not be rotated. If present, the value should be a Integer from 1 to 8. This option is mutually exclusive with kCGImageDestination-Metadata.

6.19.22 kCGImageMetadataShouldExcludeXMP as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the constant for the image destination properties.

Notes: XMP data will not be written to the destination. If used in conjunction with kCGImageDestinationMetadata, EXIF and/or IPTC tags will be preserved, but an XMP packet will not be written to the file. If present, the value for this key is a Boolean. The default is False.

6.19.23 SetProperty(options as dictionary = nil)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Applies one or more properties to all images in an image destination.

Example:

```

Var inputFile as FolderItem = SpecialFolder.Desktop.Child("test.jpg")

// reading the picture
Var c as new CGImageSourceMBS(inputfile)
Var img as CGImageMBS = c.CreateImageAtIndex(0)
Var propertiesGlobal as Dictionary = c.Properties
Var propertiesImage as Dictionary = c.PropertiesAtIndex(0)

Var outputFile as FolderItem = SpecialFolder.Desktop.Child("output.jpg")
Var d as new CGImageDestinationMBS(outputFile, "public.jpeg", 1)

// writing the picture and include metadata
d.SetProperties(propertiesGlobal)
d.AddImage(img, propertiesImage)
if d.FinalizeMT then
outputFile.Launch
else
MsgBox "Failed to write jpeg."
end if

```

Notes: Available in Mac OS X version 10.4 and later.

6.19.24 SetPropertiesCF(options as Variant)

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Applies one or more properties to all images in an image destination.

Notes: Available in Mac OS X version 10.4 and later.

Options can be a Dictionary or CFDictionaryMBS.

6.19.25 TypeIdentifiers as string()

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns an array of the uniform type identifiers (UTIs) that are supported for image destinations.

Example:

```
Var s(-1) as string = CGImageDestinationMBS.TypeIdentifiers
```

```
MsgBox Join(s,EndOfLine)
```

```
// shows on Mac OS X 10.5:
//
// public.png
// public.jpeg
// com.compuserve.gif
// public.jpeg-2000
// public.tiff
// com.adobe.photoshop-image
// com.adobe.pdf
// com.microsoft.bmp
// com.apple.pict
// com.truevision.tga-image
// com.sgi.sgi-image
// com.ilm.openexr-image
```

Notes: Returns an array of the UTIs that are supported for image destinations. See Uniform Type Identifiers Overview for a list of system-declared and third-party UTIs that can be returned.

Available in Mac OS X version 10.4 and later.

6.19.26 Properties

6.19.27 Handle as Integer

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The internal reference.

Notes: (Read and Write property)

6.20 class CGImageMBS

6.20.1 class CGImageMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: A class for a core graphics image.

Notes: If the release property is true, the destructor of this class will release the image reference.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [News from the MBS Xojo Plugins Version 24.3](#)
- [Picking picture on macOS](#)
- [Several ways for picture to PDF in MBS Plugins](#)
- [Four ways to save picture as Tiff in Xojo](#)
- [CGImageSourceMBS CreateThumbnailMT method for Xojo](#)
- [MBS Xojo Plugins in version 19.0](#)
- [Colorspaces in MacOS with Xojo](#)
- [Tip of day: Load ICC profile on OS X for any image format with CGImageSource](#)
- [Notes from this week](#)
- [MBS REALbasic plug-ins version 9.5](#)

Xojo Developer Magazine

- [17.2, page 11: News](#)
- [12.2, page 45: Moving to AVFoundation, Leaving QuickTime behind by Christian Schmitz](#)

6.20.2 Methods

6.20.3 AsNSImageMBS as variant

Plugin Version: 24.3, Platform: macOS, Targets: All.

Function: Creates a NSImageMBS which references the given CGImage.

Notes: Returns nil on error.

Result declared as Variant to avoid plugin dependencies.

6.20.4 Constructor

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: The private constructor.

6.20.5 Copy as CGImageMBS

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Creates a copy of a bitmap image.

See also:

- 6.20.6 Copy(r as CGRectMBS) as CGImageMBS 270

6.20.6 Copy(r as CGRectMBS) as CGImageMBS

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Creates a copy of a bitmap image.

See also:

- 6.20.5 Copy as CGImageMBS 270

6.20.7 CopyWithColorSpace(profile as CGColorSpaceMBS) as CGImageMBS

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Returns a copy of the image with the new Colorspace included.

Example:

```
// load a picture
Var f as FolderItem = SpecialFolder.Desktop.Child("ColorSpin.jpg")
Var pic as Picture = picture.Open(f)

// open printer
Var g as Graphics = OpenPrinterDialog
if g = nil then Return

// draw
g.DrawPicture pic, 0, 0

// now load again
Var ImageSource as new CGImageSourceMBS(f)
Var img as CGImageMBS = ImageSource.CreateImageAtIndex(0)
```

```

Var cs as CGColorSpaceMBS = CGColorSpaceMBS.CreateDeviceRGB

// copy with replacing colorspace
img = img.CopyWithColorSpace(cs)

// and draw
#If XojoVersion >= 2019.02 Then
Var p As ptr = g.Handle(Graphics.HandleTypes.CGContextRef)
Var c As CGContextMBS = CGContextMBS.contextWithCGContext(p)
#Else
Var h As Integer = g.Handle(g.HandleTypeCGContextRef)
Var c As CGContextMBS = CGContextMBS.contextWithCGContext(h)
#EndIf
Var r as CGRectMBS = CGMakeRectMBS(0, 0, img.Width, img.Height)

c.DrawPicture(img, r)
c.Flush

```

Notes: profile must be a CGColorSpaceMBS.
Requires Mac OS X 10.3.

6.20.8 CopyWithMask(mask as CGImageMBS) as CGImageMBS

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: reates a bitmap image from an existing image and an image mask.

Notes: Parameters:

self: The image to apply the mask parameter to. This image must not be an image mask and may not have an image mask or masking color associated with it.

mask: A mask. If the mask is an image, it must be in the DeviceGray color space, must not have an alpha component, and may not itself be masked by an image mask or a masking color. If the mask is not the same size as the image specified by the image parameter, then Quartz scales the mask to fit the image.

Return Value: An image created by masking image with mask. You are responsible for releasing this object by calling CGImageRelease.

The resulting image depends on whether the mask parameter is an image mask or an image. If the mask parameter is an image mask, then the source samples of the image mask act as an inverse alpha value. That is, if the value of a source sample in the image mask is S, then the corresponding region in image is blended with the destination using an alpha value of (1-S). For example, if S is 1, then the region is not painted, while if S is 0, the region is fully painted.

If the mask parameter is an image, then it serves as an alpha mask for blending the image onto the destination. The source samples of mask' act as an alpha value. If the value of the source sample in mask is S, then the corresponding region in image is blended with the destination with an alpha of S. For example, if S is 0, then the region is not painted, while if S is 1, the region is fully painted.

6.20.9 CreateImage(pic as picture) as CGImageMBS

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: Creates a new CGImageMBS from the given images.

Example:

```

Var c as CGImageMBS
Var pic, mask as Picture
// get picture and mask

c = CGImageMBS.CreateImage(pic)
if c<>Nil then
// go on
end if

```

Notes: If the image has a mask, it is used.

See also:

- 6.20.10 CreateImage(pic as picture, mask as picture) as CGImageMBS

272

6.20.10 CreateImage(pic as picture, mask as picture) as CGImageMBS

Plugin Version: 13.5, Platform: macOS, Targets: Desktop only.

Function: Creates a new CGImageMBS from the given images.

Example:

```

Var c as CGImageMBS
Var pic, mask as Picture
// get picture and mask

c = CGImageMBS.CreateImage(pic, mask)
if c<>Nil then
// go on
end if

```

Notes: The mask is taken from the second image.

With 11.3 plugins we are deprecating to pass a mask. The plugin prefers to simply take the mask or alpha

channel of the picture itself.

See also:

- 6.20.9 CreateImage(pic as picture) as CGImageMBS

272

6.20.11 CreateImageFromJPEGDataProvider(dataprovider as Variant, decode as memoryblock, shouldInterpolate as boolean, intent as Integer) as CGImageMBS

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Creates a new CGImage with JPEG data.

Example:

```
// Shows moon.jpg from the desktop folder
// shows in the window title if i,p or u is nil.
```

```
Sub Mainwindow.Paint(g As Graphics)
  Var f as FolderItem
  Var p as CGDataProviderMBS
  Var i as CGImageMBS

  f=SpecialFolder.Desktop.Child("moon.jpg")
  p=CGDataProviderMBS.CreateWithFile(f)
  if p=nil then
    Title="p=nil"
  else
    i = CGImageMBS.CreateImageFromJPEGDataProvider(p,nil,true,0)

    if i=nil then
      Title="i=nil"
    else
      window1.CGContextMBS.DrawPicture i,CGMakeRectMBS(0,0,i.Width,i.Height)
    end if
  end if

End Sub
```

Notes: Parameters:

dataprovider:

A reference to a data provider supplying JPEG-encoded data.

decode:

Pass the decode array for the image. In the decode array, for each color component in the source color space, you provide a pair of values denoting the upper and lower limits of a range. For example, the decode

array for a source image in the RGB color space would contain six entries total, consisting of one pair each for red, green, and blue. When the image is rendered, Quartz uses a linear transform to map the original component value into a relative number within your designated range that is appropriate for the destination color space. If you do not want to allow remapping of the image's color values, pass nil for the decode array. The memoryblock for the array needs to be filled with double values.

`shouldInterpolate:`

Pass true if interpolation should occur; otherwise, pass false . The interpolation setting specifies whether Quartz should apply a pixel-smoothing algorithm to the image. If you pass false , the image may appear jagged or pixelated when drawn on an output device with higher resolution than the image data.

`intent:`

Pass a `CGColorRenderingIntent` value specifying how Quartz should display colors in the image that are not located within the current color space of the graphics context. The rendering intent determines the exact method used to map colors from one color space to another.

6.20.12 `CreateImageFromPNGDataProvider(dataprovider as Variant, decode as memoryblock, shouldInterpolate as boolean, intent as Integer) as CGImageMBS`

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Creates a new `CGImage` with PNG data.

Notes: Parameters:

`dataprovider:`

A reference to a data provider supplying JPEG-encoded data.

`decode:`

Pass the decode array for the image. In the decode array, for each color component in the source color space, you provide a pair of values denoting the upper and lower limits of a range. For example, the decode array for a source image in the RGB color space would contain six entries total, consisting of one pair each for red, green, and blue. When the image is rendered, Quartz uses a linear transform to map the original component value into a relative number within your designated range that is appropriate for the destination color space. If you do not want to allow remapping of the image's color values, pass nil for the decode array. The memoryblock for the array needs to be filled with double values.

`shouldInterpolate:`

Pass true if interpolation should occur; otherwise, pass false . The interpolation setting specifies whether Quartz should apply a pixel-smoothing algorithm to the image. If you pass false , the image may appear jagged or pixelated when drawn on an output device with higher resolution than the image data.

intent:

Pass a `CGColorRenderingIntent` value specifying how Quartz should display colors in the image that are not located within the current color space of the graphics context. The rendering intent determines the exact method used to map colors from one color space to another.

6.20.13 `CreateImageWithFile(file as folderitem)` as `CGImageMBS`

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: Opens a file as `CGImage` object.

Notes: Returns nil on any error.

6.20.14 `CreateImageWithHandle(handle as Integer)` as `CGImageMBS`

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Creates image from a `CGImageRef` handle value.

Notes: Internally the `CGImageMBS` retains this reference and releases it in destructor.

6.20.15 `DataProvider` as `Variant`

Plugin Version: 3.3, Platform: macOS, Targets: All.

Function: The data provider used for the image.

Notes: Value is a `CGDataProviderMBS` object.

Returns nil on any error.

6.20.16 `DecodeArray` as `memoryblock`

Plugin Version: 3.3, Platform: macOS, Targets: All.

Function: The decode array used for the image.

Notes: A `memoryblock` with an array of double variables.

Returns nil on any error.

6.20.17 `JPEGData(Compression as Integer = 90)` as `MemoryBlock`

Plugin Version: 14.3, Platform: macOS, Targets: All.

Function: Returns image compressed as JPEG file.

Notes: Compression defines the compression level from 0 to 100.
Returns nil on any error.

6.20.18 Picture(ColorSpace as CGColorSpaceMBS = nil) as Picture

Plugin Version: 11.3, Platform: macOS, Targets: All.

Function: Creates a copy of the CGImage as a picture.

Example:

```
// get CGImage
Var f as FolderItem = SpecialFolder.Desktop.Child("bild.jpg")
Var c as CGImageMBS = CGImageMBS.CreateImageWithFile(f)
// get picture
Var p as Picture = c.Picture
// save as jpeg
Var d as FolderItem = SpecialFolder.Desktop.Child("output.jpg")
p.Save(d, p.SaveAsJPEG, 80)
```

Notes: Colorspace: the optional CoreGraphics Colorspace to use for the bitmap conversion (CGColorSpaceMBS class).

Returns nil on any error.

6.20.19 PictureScaled(OutputWidth as Integer, OutputHeight as Integer, ColorSpace as CGColorSpaceMBS = nil) as Picture

Plugin Version: 17.1, Platform: macOS, Targets: All.

Function: Creates a copy of the CGImage as a picture scaled.

Example:

```
// get CGImage
Var f as FolderItem = SpecialFolder.Desktop.Child("bild.jpg")
Var c as CGImageMBS = CGImageMBS.CreateImageWithFile(f)
// get picture
Var p as Picture = c.PictureScaled(640,480)
// save as jpeg
Var d as FolderItem = SpecialFolder.Desktop.Child("output.jpg")
p.Save(d, p.SaveAsJPEG, 80)
```

Notes: Colorspace: the optional CoreGraphics Colorspace to use for the bitmap conversion (CGColorSpaceMBS class).

Returns nil on any error.

If output width and height are zero, we use the image sizes.

6.20.20 PNGData as MemoryBlock

Plugin Version: 14.3, Platform: macOS, Targets: All.

Function: Returns image compressed as PNG file.

Notes: Returns nil on any error.

6.20.21 ReleaseHandle

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Decrements the retain count of a `CGImageRef`.

Notes: Each retain must have a release. Too many releases and your app will crash, too many retains and it will leak memory.

Use only if you really know what you are doing.

6.20.22 RetainHandle

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Increments the retain count of a `CGImageRef`.

Notes: Each retain must have a release. Too many releases and your app will crash, too many retains and it will leak memory.

Use only if you really know what you are doing.

6.20.23 Properties

6.20.24 AlphaInfo as Integer

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Returns the alpha channel information for a bitmap image.

Notes: A `CGImageAlphaInfo` constant that specifies (1) whether the bitmap contains an alpha channel, (2) where the alpha bits are located in the image data, and (3) whether the alpha value is premultiplied. For possible values, see Constants. The function returns `kCGImageAlphaNone` if the image parameter refers to an image mask.

(Read only property)

6.20.25 BitmapInfo as Integer

Plugin Version: 15.3, Platform: macOS, Targets: All.

Function: Returns the bitmap information for a bitmap image.

Notes: This function returns a constant that specifies:

The type of bitmap data—floating point or integer. You use the constant `kCGBitmapFloatComponents` to extract this information.

Whether an alpha channel is in the data, and if so, how the alpha data is stored. You use the constant `kCGBitmapAlphaInfoMask` to extract the alpha information. Alpha information is specified as one of the constants listed in Alpha Information for Images.

(Read only property)

6.20.26 BitsPerComponent as Integer

Plugin Version: 3.3, Platform: macOS, Targets: All.

Function: Returns the number of bits/component of the image.

Notes: (Read only property)

6.20.27 BitsPerPixel as Integer

Plugin Version: 3.3, Platform: macOS, Targets: All.

Function: Returns the number of bits/pixel of image.

Notes: (Read only property)

6.20.28 BytesPerRow as Integer

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Returns the number of bytes allocated for a double row of a bitmap image.

Notes: (Read only property)

6.20.29 ColorSpace as CGColorSpaceMBS

Plugin Version: 3.3, Platform: macOS, Targets: All.

Function: The color space used for the image.

Example:

```
// get a picture file
Var file as FolderItem = SpecialFolder.Desktop.Child("sylwia.jpg")
// get image source
Var source as new CGImageSourceMBS(file)
// read image
Var image as CGImageMBS = source.CreateImageAtIndex(0)
// get Color space from image
Var profile as CGColorSpaceMBS = image.ColorSpace
// get ICC profile data
Var ICCProfile as string = profile.ICCProfile
// and parse it with LCMS and show name
Var LCMSProfile as LCMS2ProfileMBS = LCMS2ProfileMBS.OpenProfileFromString(ICCProfile)
MsgBox LCMSProfile.Name
```

Notes: Value is a CGColorSpaceMBS.

Returns nil on any error.

(Read only property)

6.20.30 handle as Integer

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: The handle for this image.

Notes: Handle is a CGImageRef.

(Read and Write property)

6.20.31 height as Integer

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Returns the image's height.

Notes: Returns 0 on error.

(Read only property)

6.20.32 ImageIsMask as boolean

Plugin Version: 3.3, Platform: macOS, Targets: All.

Function: Returns true if the image is an image mask, false otherwise.

Notes: (Read only property)

6.20.33 RenderingIntent as Integer

Plugin Version: 3.3, Platform: macOS, Targets: All.

Function: Returns the rendering intent of the image.

Notes: Constants:

kCGRenderingIntentDefault	0
kCGRenderingIntentAbsoluteColorimetric	1
kCGRenderingIntentRelativeColorimetric	2
kCGRenderingIntentPerceptual	3
kCGRenderingIntentSaturation	4

(Read only property)

6.20.34 RetainCount as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the retain count of the CGImageRef.

Notes: This is useful for debugging.

The retain count is for the CGImageRef, not the CGImageMBS object.

(Read only property)

6.20.35 ShouldInterpolate as boolean

Plugin Version: 3.3, Platform: macOS, Targets: All.

Function: Returns the interpolation parameter of image.

Notes: True if the image should use interpolation.

(Read only property)

6.20.36 width as Integer

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Returns the image's width.

Notes: Returns 0 on error.

(Read only property)

6.20.37 Constants

Bitmap Info

Constant	Value	Description
kCGBitmapAlphaInfoMask	&h1F	The alpha information mask. Use this to extract alpha information that specifies whether a bitmap contains an alpha channel and how the alpha channel is generated.
kCGBitmapByteOrder16Big	12288	16-bit, big endian format.
kCGBitmapByteOrder16Little	4096	16-bit, little endian format.
kCGBitmapByteOrder32Big	16384	32-bit, big endian format.
kCGBitmapByteOrder32Little	8192	32-bit, little endian format.
kCGBitmapByteOrderDefault	0	The default byte order.
kCGBitmapByteOrderMask	&h7000	The byte ordering of pixel formats.
kCGBitmapFloatComponents	256	The components of a bitmap are floating-point values.

Alpha Info

Constant	Value	Description
kCGImageAlphaFirst	4	The alpha component is stored in the most significant bits of each pixel. For example, non-premultiplied ARGB.
kCGImageAlphaLast	3	The alpha component is stored in the least significant bits of each pixel. For example, non-premultiplied RGBA.
kCGImageAlphaNone	0	There is no alpha channel. If the total size of the pixel is greater than the space required for the number of color components in the color space, the least significant bits are ignored. This value is equivalent to kCGImageAlphaNoneSkipLast.
kCGImageAlphaNoneSkipFirst	6	There is no alpha channel. If the total size of the pixel is greater than the space required for the number of color components in the color space, the most significant bits are ignored.
kCGImageAlphaNoneSkipLast	5	There is no alpha channel. If the total size of the pixel is greater than the space required for the number of color components in the color space, the least significant bits are ignored. This value is equivalent to kCGImageAlphaNoneSkipFirst.
kCGImageAlphaOnly	7	There is no color data, only an alpha channel.
kCGImageAlphaPremultipliedFirst	2	The alpha component is stored in the most significant bits of each pixel and the color components have already been multiplied by this alpha value. For example, premultiplied ARGB.
kCGImageAlphaPremultipliedLast	1	The alpha component is stored in the least significant bits of each pixel and the color components have already been multiplied by this alpha value. For example, premultiplied RGBA.

6.21 class CGImageMetadataMBS

6.21.1 class CGImageMetadataMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The class for a tag container.

Blog Entries

- [MBS Xojo Plugins, version 18.6pr1](#)
- [MBS Xojo Plugins, version 18.5pr8](#)

6.21.2 Methods

6.21.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Creates an empty metadata object.

See also:

- [6.21.4 Constructor\(metadata as CGImageMetadataMBS\)](#) 282

6.21.4 Constructor(metadata as CGImageMetadataMBS)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Copy constructor.

Notes: Creates a copy of the metadata.

See also:

- [6.21.3 Constructor](#) 282

6.21.5 CreateFromXMPData(data as MemoryBlock) as CGImageMetadataMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Creates a collection of CGImageMetadataTags from a block of XMP data.

Notes: Converts XMP data into a collection of metadata tags.

The data must be a complete XMP tree. XMP packet headers (<?xpacket .. ?>) are supported.

data: The XMP data.

Returns a collection of CGImageMetadata tags. Returns nil if an error occurred.

6.21.6 CreateXMPData(options as Dictionary = nil) as MemoryBlock

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Serializes the CGImageMetadataRef to XMP data.

Notes: This converts all of the metadata tags to a block of XMP data. Common uses include creating sidecar files that contain metadata for image formats that do not support embedded XMP, or cannot be edited due to other format restrictions (such as proprietary RAW camera formats).

options: should be nil. Options are currently not used, but may be used in future release.

Returns a Memoryblock containing an XMP representation of the metadata. Returns nil if an error occurred.

6.21.7 kCFErrorDomainCGImageMetadata as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Error domain for all errors originating in ImageIO for CGImageMetadata APIs.

6.21.8 kCGImageMetadataEnumerateRecursively as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: A key for the 'options' of Tags function.

Notes: If present, the value should be a CFBoolean. If true, tags will be enumerated recursively, if false, only the direct children of 'rootPath' will be enumerated. The default is non-recursive.

6.21.9 kCGImageMetadataNamespaceDublinCore as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the metadata namespaces.

Notes: Dublin Core

6.21.10 kCGImageMetadataNamespaceExif as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the metadata namespaces.

Notes: Exif

6.21.11 kCGImageMetadataNamespaceExifAux as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the metadata namespaces.

Notes: Exif Aux

6.21.12 kCGImageMetadataNamespaceExifEX as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the metadata namespaces.

Notes: ExifEX

6.21.13 kCGImageMetadataNamespaceIPTCCore as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the metadata namespaces.

Notes: IPTC Core

6.21.14 kCGImageMetadataNamespaceIPTCExtension as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the metadata namespaces.

Notes: IPTC Extension

6.21.15 kCGImageMetadataNamespacePhotoshop as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the metadata namespaces.

Notes: Photoshop

6.21.16 kCGImageMetadataNamespaceTIFF as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the metadata namespaces.

Notes: TIFF

6.21.17 kCGImageMetadataNamespaceXMPBasic as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the metadata namespaces.

Notes: XMP Basic

6.21.18 kCGImageMetadataNamespaceXMPRights as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the metadata namespaces.

Notes: XMP Rights

6.21.19 kCGImageMetadataPrefixDublinCore as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the common prefix names.

Notes: Dublin Core

6.21.20 kCGImageMetadataPrefixExif as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the common prefix names.

Notes: Exif

6.21.21 kCGImageMetadataPrefixExifAux as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the common prefix names.

Notes: Exif Aux

6.21.22 kCGImageMetadataPrefixExifEX as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the common prefix names.

Notes: Exif EX

6.21.23 kCGImageMetadataPrefixIPTCCore as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the common prefix names.

Notes: IPTC Core

6.21.24 kCGImageMetadataPrefixIPTCExtension as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the common prefix names.

Notes: IPTC Extension

6.21.25 kCGImageMetadataPrefixPhotoshop as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the common prefix names.

Notes: Photoshop

6.21.26 kCGImageMetadataPrefixTIFF as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the common prefix names.

Notes: TIFF

6.21.27 kCGImageMetadataPrefixXMPBasic as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the common prefix names.

Notes: XMP Basic

6.21.28 kCGImageMetadataPrefixXMPRights as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: One of the common prefix names.

Notes: XMP Rights

6.21.29 StringValueWithPath(parent as CGImageMetadataTagMBS = nil, path as String) as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Searches for a specific tag in a CGImageMetadataRef and returns its string value.

Notes: This is a convenience method for searching for a tag at path and extracting the string value.

parent: A parent tag. If nil, the path is relative to the root of the metadata (i.e. it is not a child of another property).

path: A string with the path to the desired tag. Please consult the documentation of TagWithPath for information about path syntax.

Returns a string from a CGImageMetadataTagMBS located at 'path'. The tag must be of type kTypeString or kTypeAlternateText.

For AlternateText tags, the element with the "x-default" language qualifier will be returned. For other types, "" will be returned.

6.21.30 TagMatchingImageProperty(dictionaryName as String, propertyName as String) as Variant

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Searches for a specific `CGImageMetadataTag` matching an image property constant.

Notes: Provides a bridge for values from `CGImageCopyPropertiesAtIndex`, simplifying access for properties defined in EXIF and IPTC standards, which have no notion of namespaces, prefixes, or XMP property types.

Metadata: Working Group guidance is factored into the mapping of `CGImage` properties to XMP compatible metadata tags.

For example, `kCGImagePropertyExifDateTimeOriginal` will get the value of the corresponding XMP tag, which is `photoshop:DateCreated`. Note that property values will still be in their XMP forms, such as "YYYY-MM-DDThh:mm:ss" for `DateTime`, rather than the EXIF or IPTC `DateTime` formats.

`dictionaryName`: the metadata subdictionary to which the image property belongs, such as `kCGImagePropertyExifDictionary` or `kCGImagePropertyIPTCDictionary`. Not all dictionaries and properties are supported at this time.

`propertyName`: the name of the property. This must be a defined property constant corresponding to the `'dictionaryName'`. For example, `kCGImagePropertyTIFFOrientation`, `kCGImagePropertyExifDateTimeOriginal`, or `kCGImagePropertyIPTCKeywords`. A warning will be logged if the `CGImageProperty` is unsupported by `CGImageMetadata`.

Returns a `CGImageMetadataTagRef` with the appropriate namespace, prefix, tag name, and XMP value for the corresponding `CGImageProperty`. Returns `nil` if the property could not be found.

6.21.31 Tags as `CGImageMetadataTagMBS()`

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Obtain an array of tags from a metadata object.

See also:

- 6.21.32 `Tags(rootPath as string, options as Dictionary = nil) as CGImageMetadataTagMBS()` 288

6.21.32 `Tags(rootPath as string, options as Dictionary = nil) as CGImageMetadataTagMBS()`

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns the tags matching the path and options.

Notes: This function iterates over all of the tags in a metadata to collect the matching one. The default behavior iterates over all top-level tags in the metadata.

`rootPath`: Iteration will occur for all children of the tag matching the root path. Please refer to `TagWithPath` for information about path syntax. If an empty string, the search will be executed for all top-level tags in

the metadata container.

options: A dictionary of options for iterating through the tags.

Currently the only supported option is `kCGImageMetadataEnumerateRecursively`, which should be set to a boolean.

Returns array with found items.

See also:

- 6.21.31 Tags as `CGImageMetadataTagMBS()`

288

6.21.33 TagWithPath(parent as CGImageMetadataTagMBS = nil, path as string) as Variant

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Searches for a specific `CGImageMetadataTagMBS` in a `CGImageMetadataMBS`.

Notes: This is the primary function for clients to obtain specific metadata properties from an image. The 'path' mechanism provides a way to access both simple top-level properties, such as Date & Time, or complex deeply-nested properties with ease.

parent: A parent tag. If nil, the path is relative to the root of the metadata (i.e. it is not a child of another property). If the parent is provided, the effective path will be the concatenation of the parent's path and the 'path' parameter. This is useful for accessing array elements or structure fields inside nested tags.

path: A string representing a path to the desired tag. Paths consist of a tag prefix (i.e. "exif") joined with a tag name (i.e. "Flash") by a colon (":"), such as "exif:Flash".

Elements of ordered and unordered arrays are accessed via 0-based indices inside square [] brackets.

Elements of alternate-text arrays are accessed by an RFC 3066 language code inside square [] brackets.

Fields of a structure are delimited by a period, ".".

Qualifiers are delimited by the '?' character. Only tags with string values (`kTypeString`) are allowed to have qualifiers - arrays and structures may not contain qualifiers.

If parent is nil, a prefix must be specified for the first tag. Prefixes for all subsequent tags are optional. If unspecified, the prefix is inherited from the nearest parent tag with a prefix. Custom prefixes must be registered using `RegisterNamespaceForPrefix` prior to use in any path-based functions.

Examples:

- 'path' = "xmp:CreateDate"
- 'path' = "exif:Flash.Fired"
- 'parent' = tag at path "exif:Flash", path = "exif:Fired" (equivalent to previous)

- 'path' = "exif:Flash.RedEyeMode"
- 'path' = "dc:title"
- 'path' = "dc:subject"
- 'path' = "dc:subject [2] "
- 'parent' = tag at path "dc:subject", path = " [2] " (equivalent to previous)
- 'path' = "dc:description [x-default]"
- 'path' = "dc.description [de]"
- 'path' = "dc.description [fr]"
- 'path' = "foo:product"
- 'path' = "foo:product?bar:manufacturer"

Returns a `CGImageMetadataTagMBS` matching 'path', or nil if no match is found. The copy of the tag's value is shallow. Tags copied from an immutable `CGImageMetadataMBS` are also immutable. Because this function returns a copy of the tag's value, any modification of the tag's value must be followed by a `SetTagWithPath` to commit the change to the metadata container.

6.21.34 Properties

6.21.35 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

6.21.36 Constants

Errors

Constant	Value	Description
<code>kErrorBadArgument</code>	2	Bad argument
<code>kErrorConflictingArguments</code>	3	Conflicting Arguments
<code>kErrorPrefixConflict</code>	4	Prefix conflict
<code>kErrorUnknown</code>	0	Unknown error
<code>kErrorUnsupportedFormat</code>	1	Unsupported Format

6.22 class CGImageMetadataTagMBS

6.22.1 class CGImageMetadataTagMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The class for a metadata tag.

Notes: This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [MBS Xojo Plugins, version 18.5pr8](#)

6.22.2 Methods

6.22.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The private constructor.

6.22.4 Create(XMLNS as String, Prefix as String, Name as String, Type as Integer, Value as Variant) as CGImageMetadataTagMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Creates a new CGImageMetadataTagMBS object.

Notes: xmlns: The namespace for the tag. The value can be a common XMP namespace defined above, such as kCGImageMetadataNamespaceExif, or a string with a custom namespace URI. Custom namespaces must be a valid XML namespace. By convention, namespaces should end with either '/' or '#'. For example, exif uses the namespace "http://ns.adobe.com/exif/1.0/".

prefix: An abbreviation for the XML namespace. The value can be empty if the namespace is defined as a constant. Custom prefixes must be a valid XML name. For example, the prefix used for "http://ns.adobe.com/exif/1.0/" is "exif". The XMP serialization of the tag will use the prefix. Prefixes are also important for path-based metadata functions, such as StringValueWithPath or SetValueWithPath.

name: The name of the tag. It must be a valid XMP name.

type: The type of the tag's value. Must be a kType* constant.

value: The value of the tag. Allowable types include string, double, integer, boolean, array, and dictionary. The type of 'value' must correspond to the 'type'. The elements of an array must be either a string or tags itself. The keys of a CFDictionary must be string with valid XMP names. The values of a dictionary must be either strings or tags. A shallow copy of the value is stored in the tag. Therefore, modifying a mutable value after the tag is created will not affect the tag's value.

Returns a new CGImageMetadataTagMBS object. Returns nil if a tag could not be created with the speci-

fied parameters.

6.22.5 Qualifiers as `CGImageMetadataTagMBS()`

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Return a copy of the tag's qualifiers.

Notes: XMP allows properties to contain supplemental properties called qualifiers. Qualifiers are themselves `CGImageMetadataTagMBS` objects with their own namespace, prefix, name, and value. A common use is the `xml:lang` qualifier for elements of an `alternate-text` array.

Returns a copy of the array of qualifiers. Elements of the array are `CGImageMetadataTagMBS`. Returns nil if the tag does not have any qualifiers.

6.22.6 Properties

6.22.7 Handle as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The internal object reference.

Notes: (Read and Write property)

6.22.8 Name as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns the tag's name.

Notes: (Read only property)

6.22.9 Namespace__ as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns the tag's namespace.

Notes: As `Namespace` is a reserved word in Xojo, we need to add a `_` to get it to compile. (Read only property)

6.22.10 Prefix as String

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns the tag's prefix.

Notes: (Read only property)

6.22.11 Type as Integer

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Get the type of the tag.

Notes: Returns a kType constant for the tag.

This is primarily used to determine how to interpret the tag's value.

(Read only property)

6.22.12 Value as Variant

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Returns a shallow copy of the tag's value.

Notes: This function should only be used to read the tag's value.

TagWithPath returns the tag (including the tag's value). Therefore mutating a tag's value returned from this function may not actually mutate the value in the CGImageMetadata. It is recommended to create a new tag followed by SetTagWithPath, or use SetValueWithPath to mutate a metadata value.

(Read only property)

6.22.13 Constants

Types

Constant	Value	Description
kTypeAlternateArray	4	An ordered array where all elements are alternates for the same value. Serialized in XMP as <rdf:Alt>.
kTypeAlternateText	5	A special case of an alternate array where all elements are different localized strings for the same value. Serialized in XMP as an alternate array of strings with xml:lang qualifiers.
kTypeArrayOrdered	3	An array where order is preserved. Serialized in XMP as <rdf:Seq>.
kTypeArrayUnordered	2	An array where order does not matter. Serialized in XMP as <rdf:Bag>.
kTypeDefault	0	The type is taken from the passed value for new tags.
kTypeInvalid	-1	Invalid.
kTypeString	1	A string value. Number and boolean values will be converted to a string.
kTypeStructure	6	A collection of keys and values. Unlike array elements, fields of a structure may belong to different namespaces.

6.23 class CGImageSourceMBS

6.23.1 class CGImageSourceMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The CoreGraphics class for image loading.

Example:

```
Var file as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
Var c as new CGImageSourceMBS(file)
Var img as CGImageMBS = c.CreateImageAtIndex(0)
```

```
Backdrop=img.Picture
```

Notes: CGImageSource objects, available in Mac OS X v10.4 or later, abstract the data-reading task. An image source can read image data from a URL, a file or a string.

After creating a CGImageSource object for the appropriate source, you can obtain images, thumbnails, image properties, and other image information using CGImageSource functions.

Blog Entries

- [Reading JPEG Thumbnails from EXIF](#)
- [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](#)
- [MonkeyBread Software Releases the MBS Xojo plug-ins in version 17.0](#)
- [Tip of day: Load ICC profile on OS X for any image format with CGImageSource](#)
- [Tip of the day: Edit IPTC data in JPEG file with CGImageSource/CGImageDestination](#)
- [Tip of the day: Quickly query image size](#)
- [MBS REALbasic plug-ins version 9.5](#)

Xojo Developer Magazine

- [17.3, page 11: News](#)
- [15.2, page 10: News](#)

6.23.2 Methods

6.23.3 Constructor(data as string, options as dictionary = nil)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates an image source that reads from a string.

Notes: data: The data string to read from.

options: A dictionary that specifies additional creation options. For example kCGImageSourceTypeIdentifierHint.

On success the handle value is not zero.

Available in Mac OS X version 10.4 and later.

See also:

- 6.23.4 Constructor(file as folderitem, options as dictionary = nil) 296
- 6.23.5 Constructor(options as dictionary = nil) 296

6.23.4 Constructor(file as folderitem, options as dictionary = nil)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates an image source that reads from a location specified by a file.

Notes: url: The URL to read from.

options: A dictionary that specifies additional creation options. For example kCGImageSourceTypeIdentifierHint.

On success the handle value is not zero.

Available in Mac OS X version 10.4 and later.

See also:

- 6.23.3 Constructor(data as string, options as dictionary = nil) 296
- 6.23.5 Constructor(options as dictionary = nil) 296

6.23.5 Constructor(options as dictionary = nil)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Create an incremental image source.

Notes: options: A dictionary that specifies additional creation options. For example kCGImageSource-

TypeIdentifierHint.

Returns an image source object.

The constructor creates an empty image source container to which you can add data later by calling the function UpdateData. You don't provide data when you call this function.

An incremental image is an image that is created in chunks, similar to the way large images viewed over the web are loaded piece by piece.

Available in Mac OS X version 10.4 and later.

See also:

- 6.23.3 Constructor(data as string, options as dictionary = nil) 296
- 6.23.4 Constructor(file as folderitem, options as dictionary = nil) 296

6.23.6 CreateImageAtIndex(index as Integer, options as dictionary = nil) as CGImageMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a CGImage object for the image data associated with the specified index in an image source.

Example:

```
Var file as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
Var c as new CGImageSourceMBS(file)
Var img as CGImageMBS = c.CreateImageAtIndex(0)
```

```
Backdrop=img.Picture
```

Notes: index: The index that specifies the location of the image. The index is zero-based.

options: A dictionary that specifies additional creation options.

Returns a CGImage object.

Available in Mac OS X version 10.4 and later.

6.23.7 CreateIncremental(options as dictionary=nil) as CGImageSourceMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Create an incremental image source.

Notes: options: A dictionary that specifies additional creation options. For example kCGImageSourceTypeIdentifierHint.

The function CreateIncremental creates an empty image source container to which you can add data later by calling the function UpdateData. You don't provide data when you call this function.

An incremental image is an image that is created in chunks, similar to the way large images viewed over the web are loaded piece by piece.

Available in Mac OS X version 10.4 and later.

6.23.8 CreateThumbnailAtIndex(index as Integer, options as dictionary = nil) as CGImageMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a thumbnail image of the image located at a specified location in an image source.

Example:

```
Var file as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
```

```
Var c as new CGImageSourceMBS(file)
```

```
Var d as new Dictionary
```

```
d.Value(c.kCGImageSourceThumbnailMaxPixelSize)=200
```

```
d.Value(c.kCGImageSourceCreateThumbnailFromImageIfAbsent)=true
```

```
Var img as CGImageMBS = c.CreateThumbnailAtIndex(0,d)
```

```
Backdrop=img.Picture
```

Notes: index: The index that specifies the location of the image. The index is zero-based.

options: A dictionary that specifies additional creation options.

Returns a CGImageMBS.

If the image source is a PDF, this function creates a 72 dpi image of the PDF page specified by the index that you pass. You must, however, pass an options dictionary that contains either the kCGImage-

SourceCreateThumbnailFromImageIfAbsent or kCGImageSourceCreateThumbnailFromImageAlways keys, with the value of the key set to TRUE.

Available in Mac OS X version 10.4 and later.

6.23.9 CreateThumbnailMT(data as MemoryBlock, ImageIndex as integer = 0, Options as dictionary = nil) as CGImageMBS

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Creates thumbnail for image thread friendly.

Notes: Data contains the image file data.

ImageIndex is the index of the image in the file to query.

Options contains various flags for options to creating image source and creating thumbnail.

Returns nil in case of any error.

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.

See also:

- 6.23.10 CreateThumbnailMT(data as String, ImageIndex as integer = 0, Options as dictionary = nil) as CGImageMBS 299
- 6.23.11 CreateThumbnailMT(file as FolderItem, ImageIndex as integer = 0, Options as dictionary = nil) as CGImageMBS 300

6.23.10 CreateThumbnailMT(data as String, ImageIndex as integer = 0, Options as dictionary = nil) as CGImageMBS

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Creates thumbnail for image thread friendly.

Notes: Data contains the image file data.

ImageIndex is the index of the image in the file to query.

Options contains various flags for options to creating image source and creating thumbnail.

Returns nil in case of any error.

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.

See also:

- 6.23.9 CreateThumbnailMT(data as MemoryBlock, ImageIndex as integer = 0, Options as dictionary = nil) as CGImageMBS 299
- 6.23.11 CreateThumbnailMT(file as FolderItem, ImageIndex as integer = 0, Options as dictionary = nil) as CGImageMBS 300

6.23.11 CreateThumbnailMT(file as FolderItem, ImageIndex as integer = 0, Options as dictionary = nil) as CGImageMBS

Plugin Version: 19.1, Platform: macOS, Targets: All.

Function: Creates thumbnail for image thread friendly.

Notes: File refers to image file on disk.

ImageIndex is the index of the image in the file to query.

Options contains various flags for options to creating image source and creating thumbnail.

Returns nil in case of any error.

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.

See also:

- 6.23.9 CreateThumbnailMT(data as MemoryBlock, ImageIndex as integer = 0, Options as dictionary = nil) as CGImageMBS 299
- 6.23.10 CreateThumbnailMT(data as String, ImageIndex as integer = 0, Options as dictionary = nil) as CGImageMBS 299

6.23.12 CreateWithData(data as string, options as dictionary=nil) as CGImageSourceMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates an image source that reads from a string.

Notes: data: The data string to read from.

options: A dictionary that specifies additional creation options. For example kCGImageSourceTypeIdentifierHint.

Returns an image source.

Available in Mac OS X version 10.4 and later.

6.23.13 CreateWithFile(file as folderitem, options as dictionary=nil) as CGImageSourceMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates an image source that reads from a location specified by a file.

Example:

```
// get a picture file
Var file as FolderItem = SpecialFolder.Desktop.Child("sylwia.jpg")
// get image source
Var source as new CGImageSourceMBS(file)
// read image
Var image as CGImageMBS = source.CreateImageAtIndex(0)
// get Color space from image
Var profile as CGColorSpaceMBS = image.ColorSpace
// get ICC profile data
Var ICCProfile as string = profile.ICCProfile
// and parse it with LCMS and show name
Var LCMSProfile as LCMS2ProfileMBS = LCMS2ProfileMBS.OpenProfileFromString(ICCProfile)
MsgBox LCMSProfile.Name
```

Notes: url: The URL to read from.

options: A dictionary that specifies additional creation options. For example kCGImageSourceTypeIdentifierHint.

Returns an image source.

Available in Mac OS X version 10.4 and later.

6.23.14 CreateWithURL(url as string, options as dictionary=nil) as CGImageSourceMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates an image source that reads from a location specified by a URL.

Notes: url: The URL to read from.

options: A dictionary that specifies additional creation options. For example kCGImageSourceTypeIdentifierHint.

On success the handle value is not zero.

Available in Mac OS X version 10.4 and later.

6.23.15 kCGImageProperty8BIMDictionary as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: A dictionary of key-value pairs for an Adobe Photoshop image.

6.23.16 kCGImageProperty8BIMLayerNames as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: The layer names for an Adobe Photoshop file.

6.23.17 kCGImagePropertyCIFFCameraSerialNumber as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The camera serial number.

6.23.18 kCGImagePropertyCIFFContinuousDrive as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The continuous drive mode.

6.23.19 kCGImagePropertyCIFFDescription as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The camera description..

6.23.20 kCGImagePropertyCIFFDictionary as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: A dictionary of key-value pairs for an image that uses Camera Image File Format (CIFF).

Available in Mac OS X v10.5 and later.

6.23.21 kCGImagePropertyCIFFFirmware as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The firmware version.

6.23.22 kCGImagePropertyCIFFFlashExposureComp as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The flash exposure compensation.

6.23.23 kCGImagePropertyCIFFFocusMode as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The focus mode.

6.23.24 kCGImagePropertyCIFFImageFileName as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The image file name.

6.23.25 kCGImagePropertyCIFFImageName as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The image name.

6.23.26 kCGImagePropertyCIFFImageSerialNumber as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The image serial number.

6.23.27 kCGImagePropertyCIFFLensMaxMM as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The maximum lens length.

6.23.28 kCGImagePropertyCIFFLensMinMM as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The minimum lens length.

6.23.29 kCGImagePropertyCIFFLensModel as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The lens model.

6.23.30 kCGImagePropertyCIFFMeasuredEV as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The measured EV.

6.23.31 kCGImagePropertyCIFFMeteringMode as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The metering mode.

6.23.32 kCGImagePropertyCIFFOwnerName as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The owner name.

6.23.33 kCGImagePropertyCIFFRecordID as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The record ID

6.23.34 kCGImagePropertyCIFFReleaseMethod as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The release method.

6.23.35 `kCGImagePropertyCIFFReleaseTiming` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The release timing.

6.23.36 `kCGImagePropertyCIFFSelfTimingTime` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The self timing time.

6.23.37 `kCGImagePropertyCIFFShootingMode` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The shooting mode.

6.23.38 `kCGImagePropertyCIFFWhiteBalanceIndex` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Camera Image File Format (CIFF).

Notes: The white balance index.

6.23.39 `kCGImagePropertyColorModel` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: The color model of the image such as, "RGB", "CMYK", "Gray", or "Lab". The value of this key is CFStringRef.

A color model describes how color values are represented mathematically. A color space is a color model combined with a definition of how to interpret values within the model.

6.23.40 kCGImagePropertyColorModelCMYK as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Values for the color model property.

Notes: A CMYK color model.

6.23.41 kCGImagePropertyColorModelGray as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Values for the color model property.

Notes: A Gray color model.

6.23.42 kCGImagePropertyColorModelLab as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Values for the color model property.

Notes: A Lab color model.

6.23.43 kCGImagePropertyColorModelRGB as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Values for the color model property.

Notes: An RGB color model.

6.23.44 kCGImagePropertyDepth as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: The number of bits in each color sample of each pixel. If present, this key is a CFNumber value.

6.23.45 kCGImagePropertyDNGBackwardVersion as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses the Digital Negative (DNG) archival format.

Notes: The oldest version for which a file is compatible.

6.23.46 `kCGImagePropertyDNGLensInfo` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses the Digital Negative (DNG) archival format.

Notes: The camera serial number.

6.23.47 `kCGImagePropertyDNGLensInfo` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: A dictionary of key-value pairs for an image that uses the Digital Negative (DNG) archival format.

6.23.48 `kCGImagePropertyDNGLensInfo` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses the Digital Negative (DNG) archival format.

Notes: Information about the lens used for the image.

6.23.49 `kCGImagePropertyDNGLensInfo` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses the Digital Negative (DNG) archival format.

Notes: The localized camera model name.

6.23.50 `kCGImagePropertyDNGLensInfo` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses the Digital Negative (DNG) archival format.

Notes: A unique, nonlocalized name for the camera mode.

6.23.51 kCGImagePropertyDNGVersion as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses the Digital Negative (DNG) archival format.

Notes: An encoding of the four-tier version number.

6.23.52 kCGImagePropertyDPIHeight as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: The resolution, in dots per inch, in the x dimension. If present, this key is a CFNumber value.

6.23.53 kCGImagePropertyDPIWidth as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: The resolution, in dots per inch, in the y dimension. If present, this key is a CFNumber value.

6.23.54 kCGImagePropertyExifApertureValue as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The aperture value.

6.23.55 kCGImagePropertyExifAuxDictionary as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: An auxiliary dictionary of key-value pairs for an image that uses Exchangeable Image File Format (EXIF).

6.23.56 kCGImagePropertyExifAuxFirmware as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the Exif Auxiliary dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: Firmware information.

6.23.57 `kCGImagePropertyExifAuxFlashCompensation` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the Exif Auxiliary dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: Flash compensation.

6.23.58 `kCGImagePropertyExifAuxImageNumber` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the Exif Auxiliary dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The image number.

6.23.59 `kCGImagePropertyExifAuxLensID` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the Exif Auxiliary dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The lens ID.

6.23.60 `kCGImagePropertyExifAuxLensInfo` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the Exif Auxiliary dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: Lens information.

6.23.61 `kCGImagePropertyExifAuxLensModel` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the Exif Auxiliary dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The lens model.

6.23.62 kCGImagePropertyExifAuxLensSerialNumber as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the Exif Auxiliary dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The lens serial number.

6.23.63 kCGImagePropertyExifAuxOwnerName as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the Exif Auxiliary dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The owner name.

6.23.64 kCGImagePropertyExifAuxSerialNumber as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the Exif Auxiliary dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The serial number.

6.23.65 kCGImagePropertyExifBrightnessValue as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The brightness value.

6.23.66 kCGImagePropertyExifCFAPattern as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The color filter array (CFA) pattern, which is the geometric pattern of the image sensor for a 1-chip color sensor area.

6.23.67 kCGImagePropertyExifColorSpace as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The color space.

6.23.68 kCGImagePropertyExifComponentsConfiguration as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The components configuration. For compressed data, specifies that the channels of each component are arranged in increasing numeric order (from first component to the fourth).

6.23.69 kCGImagePropertyExifCompressedBitsPerPixel as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The compressed bits per pixel.

6.23.70 kCGImagePropertyExifContrast as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The contrast applied to the image.

6.23.71 kCGImagePropertyExifCustomRendered as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: Special rendering performed on the image data.

6.23.72 kCGImagePropertyExifDateTimeDigitized as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The digitized date and time.

6.23.73 kCGImagePropertyExifDateTimeOriginal as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The original date and time.

6.23.74 kCGImagePropertyExifDeviceSettingDescription as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: For a particular camera mode, indicates the conditions for taking the picture.

6.23.75 kCGImagePropertyExifDictionary as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: A dictionary of key-value pairs for an image that uses Exchangeable Image File Format (EXIF).

6.23.76 kCGImagePropertyExifDigitalZoomRatio as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The digital zoom ratio.

6.23.77 kCGImagePropertyExifExposureBiasValue as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The exposure bias value.

6.23.78 `kCGImagePropertyExifExposureIndex` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The selected exposure index.

6.23.79 `kCGImagePropertyExifExposureMode` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The exposure mode setting.

6.23.80 `kCGImagePropertyExifExposureProgram` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The exposure program.

6.23.81 `kCGImagePropertyExifExposureTime` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The exposure time.

6.23.82 `kCGImagePropertyExifFileSource` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The image source.

6.23.83 kCGImagePropertyExifFlash as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The flash status when the image was shot.

6.23.84 kCGImagePropertyExifFlashEnergy as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The strobe energy when the image was captures, in beam candle power seconds.

6.23.85 kCGImagePropertyExifFlashPixVersion as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The FlashPix version supported by an FPXR file. FlashPix is a format for multi-resolution, tiled images, that facilitates fast onscreen viewing.

6.23.86 kCGImagePropertyExifFNumber as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The F number.

6.23.87 kCGImagePropertyExifFocalLength as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The focal length.

6.23.88 kCGImagePropertyExifFocalLenIn35mmFilm as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The equivalent focal length in 35 mm film.

6.23.89 `kCGImagePropertyExifFocalPlaneResolutionUnit` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The unit of measurement for the focal plane x and y tags.

6.23.90 `kCGImagePropertyExifFocalPlaneXResolution` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The number of image-width pixels (x) per focal plane resolution unit.

6.23.91 `kCGImagePropertyExifFocalPlaneYResolution` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The number of image-height pixels (y) per focal plane resolution unit.

6.23.92 `kCGImagePropertyExifGainControl` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The gain adjustment applied to the image.

6.23.93 `kCGImagePropertyExifGamma` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The gamma setting.

6.23.94 kCGImagePropertyExifImageUniqueID as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The unique ID of the image.

6.23.95 kCGImagePropertyExifISOSpeedRatings as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: ISO speed ratings.

6.23.96 kCGImagePropertyExifLightSource as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The light source.

6.23.97 kCGImagePropertyExifMakerNote as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: A maker note.

6.23.98 kCGImagePropertyExifMaxApertureValue as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The maximum aperture value.

6.23.99 kCGImagePropertyExifMeteringMode as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The metering mode.

6.23.100 kCGImagePropertyExifOECF as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The opto-electrical conversion function (OECF), which defines the relationship between the optical input of the camera and the image values.

6.23.101 kCGImagePropertyExifPixelXDimension as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The pixel x dimension.

6.23.102 kCGImagePropertyExifPixelYDimension as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The pixel y dimension.

6.23.103 kCGImagePropertyExifRelatedSoundFile as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: A related sound file.

6.23.104 kCGImagePropertyExifSaturation as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The saturation applied to the image.

6.23.105 kCGImagePropertyExifSceneCaptureType as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The scene capture type (standard, landscape, portrait, night).

6.23.106 kCGImagePropertyExifSceneType as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The scene type.

6.23.107 kCGImagePropertyExifSensingMethod as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The sensor type of the camera or input device.

6.23.108 kCGImagePropertyExifSharpness as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The sharpness applied to the image.

6.23.109 kCGImagePropertyExifShutterSpeedValue as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The shutter speed value.

6.23.110 kCGImagePropertyExifSpatialFrequencyResponse as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The spatial frequency table and spatial frequency response values in the direction of image width, image height, and diagonal directions. See ISO 12233..

6.23.111 kCGImagePropertyExifSpectralSensitivity as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The spectral sensitivity of each channel.

6.23.112 kCGImagePropertyExifSubjectArea as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The subject area.

6.23.113 kCGImagePropertyExifSubjectDistance as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).
Notes: The distance to the subject, in meters.

6.23.114 kCGImagePropertyExifSubjectDistRange as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The subject distance range.

6.23.115 kCGImagePropertyExifSubjectLocation as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The location of the scene's primary subject.

6.23.116 kCGImagePropertyExifSubsecTime as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The fraction of seconds for the date and time tag.

6.23.117 kCGImagePropertyExifSubsecTimeDigitized as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The fraction of seconds for the digitized time tag.

6.23.118 kCGImagePropertyExifSubsecTimeOriginal as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The fraction of seconds for the original date and time tag.

6.23.119 kCGImagePropertyExifUserComment as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: A user comment.

6.23.120 `kCGImagePropertyExifVersion` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The version.

6.23.121 `kCGImagePropertyExifWhiteBalance` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for the (EXIF) dictionary for an image that uses Exchangeable Image File Format (EXIF).

Notes: The white balance mode.

6.23.122 `kCGImagePropertyFileSize` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: The size of the image file in bytes, if known. If present, this key is a CFNumber value.

Available in Mac OS X v10.4 and later.

This key is used in the image source properties.

6.23.123 `kCGImagePropertyGIFDelayTime` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Graphics Interchange Format (GIF).

Notes: The delay time.

6.23.124 `kCGImagePropertyGIFDictionary` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: A dictionary of key-value pairs for an image that uses Graphics Interchange Format (GIF).

6.23.125 kCGImagePropertyGIFHasGlobalColorMap as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Graphics Interchange Format (GIF).

Notes: Whether or not the GIF has a global color map.

6.23.126 kCGImagePropertyGIFImageColorMap as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Graphics Interchange Format (GIF).

Notes: The image color map.

6.23.127 kCGImagePropertyGIFLoopCount as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Graphics Interchange Format (GIF).

Notes: The loop count.

6.23.128 kCGImagePropertyGIFUnclampedDelayTime as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: The unclamped delay time.

Available in OS X v10.7 and later.

6.23.129 kCGImagePropertyGPSAltitude as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The altitude.

6.23.130 kCGImagePropertyGPSAltitudeRef as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The reference altitude.

6.23.131 kCGImagePropertyGPSAreaInformation as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The name of the GPS area.

6.23.132 kCGImagePropertyGPSDateStamp as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The data and time information relative to Coordinated Universal Time (UTC).

6.23.133 kCGImagePropertyGPSDestBearing as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The bearing to the destination point.

6.23.134 kCGImagePropertyGPSDestBearingRef as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The reference for giving the bearing to the destination point.

6.23.135 kCGImagePropertyGPSDestDistance as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The distance to the destination point.

6.23.136 kCGImagePropertyGPSDestDistanceRef as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The units for expressing the distance to the destination point.

6.23.137 kCGImagePropertyGPSDestLatitude as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The latitude of the destination point.

6.23.138 kCGImagePropertyGPSDestLatitudeRef as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: Whether the latitude of the destination point is northern or southern.

6.23.139 kCGImagePropertyGPSDestLongitude as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The longitude of the destination point.

6.23.140 kCGImagePropertyGPSDestLongitudeRef as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: Whether the longitude of the destination point is east or west.

6.23.141 kCGImagePropertyGPSDictionary as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: A dictionary of key-value pairs for an image that has Global Positioning System (GPS) information.

6.23.142 kCGImagePropertyGPSDifferential as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: Whether differential correction is applied to the GPS receiver.

6.23.143 kCGImagePropertyGPSDOP as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The data degree of precision (DOP).

6.23.144 kCGImagePropertyGPSImgDirection as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The direction of the image.

6.23.145 kCGImagePropertyGPSImgDirectionRef as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The reference for the direction of the image.

6.23.146 kCGImagePropertyGPSLatitude as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The latitude.

6.23.147 kCGImagePropertyGPSLatitudeRef as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: Whether the latitude is northern or southern.

6.23.148 kCGImagePropertyGPSLongitude as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The longitude.

6.23.149 kCGImagePropertyGPSLongitudeRef as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: Whether the longitude is east or west.

6.23.150 kCGImagePropertyGPSMapDatum as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The geodetic survey data used by the GPS receiver.

6.23.151 kCGImagePropertyGPSMeasureMode as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The measurement mode.

6.23.152 kCGImagePropertyGPSProcessingMethod as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The name of the method used for finding a location.

6.23.153 kCGImagePropertyGPSSatellites as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The satellites used for GPS measurements.

6.23.154 kCGImagePropertyGPSSpeed as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The GPS receiver speed of movement.

6.23.155 kCGImagePropertyGPSSpeedRef as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The unit for expressing the GPS receiver speed of movement.

6.23.156 kCGImagePropertyGPSStatus as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The status of the GPS receiver.

6.23.157 kCGImagePropertyGPSTimeStamp as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The time as UTC (Coordinated Universal Time).

6.23.158 kCGImagePropertyGPSTrack as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The direction of GPS receiver movement.

6.23.159 kCGImagePropertyGPSTrackRef as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The reference for the direction of GPS receiver movement.

6.23.160 kCGImagePropertyGPSVersion as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that has Global Positioning System (GPS) information.

Notes: The version.

6.23.161 kCGImagePropertyHasAlpha as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: Whether or not the image has an alpha channel. The value of this key is `kCFBooleanTrue` if the image contains an alpha channel.

6.23.162 kCGImagePropertyIPTCActionAdvised as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The advised action.

6.23.163 kCGImagePropertyIPTCByline as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The byline.

6.23.164 kCGImagePropertyIPTCBylineTitle as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The byline title.

6.23.165 kCGImagePropertyIPTCCaptionAbstract as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The caption abstract.

6.23.166 kCGImagePropertyIPTCCategory as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The category.

6.23.167 kCGImagePropertyIPTCCity as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The city.

6.23.168 kCGImagePropertyIPTCContact as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: Contact information.

6.23.169 kCGImagePropertyIPTCContactInfoAddress as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: The address portion of the contact information.

Available in OS X v10.6 and later.

6.23.170 kCGImagePropertyIPTCContactInfoCity as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: The city portion of the contact information.

Available in OS X v10.6 and later.

6.23.171 kCGImagePropertyIPTCContactInfoCountry as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: The country portion of the contact information.

Available in OS X v10.6 and later.

6.23.172 kCGImagePropertyIPTCContactInfoEmails as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: Email addresses for the contact.

Available in OS X v10.6 and later.

6.23.173 kCGImagePropertyIPTCContactInfoPhones as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: Phone numbers for the contact.

Available in OS X v10.6 and later.

6.23.174 `kCGImagePropertyIPTCContactInfoPostalCode` as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: The postal code portion of the contact information.

Available in OS X v10.6 and later.

6.23.175 `kCGImagePropertyIPTCContactInfoStateProvince` as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: The state or province for the contact.

Available in OS X v10.6 and later.

6.23.176 `kCGImagePropertyIPTCContactInfoWebURLs` as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: Web addresses for the contact.

Available in OS X v10.6 and later.

6.23.177 `kCGImagePropertyIPTCContentLocationCode` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The content location code.

6.23.178 `kCGImagePropertyIPTCContentLocationName` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The content location name.

6.23.179 kCGImagePropertyIPTCCopyrightNotice as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The copyright notice.

6.23.180 kCGImagePropertyIPTCCountryPrimaryLocationCode as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The country primary location code.

6.23.181 kCGImagePropertyIPTCCountryPrimaryLocationName as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The country primary location name.

6.23.182 kCGImagePropertyIPTCCreatorContactInfo as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: The creator's contact info. See "IPTC Creator Contact Info Dictionary Keys."

Available in OS X v10.6 and later.

6.23.183 kCGImagePropertyIPTCCredit as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: Credit information.

6.23.184 kCGImagePropertyIPTCDateCreated as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The date created.

6.23.185 kCGImagePropertyIPTCDictionary as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Example:

```
// Change rotation in an image file

// files
Var dpath as folderitem = SpecialFolder.Desktop.Child("test.jpg")
Var opath as folderitem = dpath.parent.Child("output.jpg")

// open source
Var imageSource as new cgimagesourcembs(dpath)

// read image
Var img as cgimagembs = imageSource.createimageatindex(0)

// global properties
Var globalprop as dictionary = imageSource.properties
// per image properties
Var p as Dictionary = imageSource.PropertiesAtIndex(0)

Var imageDest as new CGImageDestinationMBS(opath,"public.jpeg",1)

'const orientation = 1 // top left
const orientation = 3 // bottom right

// set globals
imageDest.SetProperties(globalprop)

// now set new orientation
p.value(imageDest.kCGImageDestinationOrientation) = orientation
p.value(imageDest.kCGImageDestinationMergeMetadata) = true

// change tiff dictionary, if present
Var dTIFF as Dictionary = p.lookup(imageSource.kCGImagePropertyTIFFDictionary, nil)
if dTIFF <> nil then
dTIFF.value(imageSource.kCGImagePropertyTIFFOrientation) = orientation
```

end if

```
// change iptc dictionary, if present
Var dIPTC as Dictionary = p.lookup(imageSource.kCGImagePropertyIPTCDictionary, nil)
if dIPTC <> nil then
dIPTC.value(imageSource.kCGImagePropertyIPTCImageOrientation) = orientation
end if

// write out image
imageDest.AddImage(img,p)
call imageDest.Finalize
```

Notes: A dictionary of key-value pairs for an image that uses International Press Telecommunications Council (IPTC) metadata.

IPTC constants are metadata elements of the Information Interchange Model (IIM) used to provide information about images. The IIM was developed by the Newspaper Association of America (NAA) and the International Press Telecommunications Council (IPTC).

6.23.186 kCGImagePropertyIPTCDigitalCreationDate as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The digital creation date.

6.23.187 kCGImagePropertyIPTCDigitalCreationTime as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The digital creation time.

6.23.188 kCGImagePropertyIPTCEditorialUpdate as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: An editorial update.

6.23.189 kCGImagePropertyIPTCEditStatus as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The edit status.

6.23.190 kCGImagePropertyIPTCExpirationDate as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The expiration date.

6.23.191 kCGImagePropertyIPTCExpirationTime as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The expiration time.

6.23.192 kCGImagePropertyIPTCFixtureIdentifier as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: A fixture identifier.

6.23.193 kCGImagePropertyIPTCHeadline as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The headline.

6.23.194 kCGImagePropertyIPTCImageOrientation as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The image orientation.

6.23.195 kCGImagePropertyIPTCImageType as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The image type.

6.23.196 kCGImagePropertyIPTCKeywords as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: Keywords

6.23.197 kCGImagePropertyIPTCLanguageIdentifier as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The language identifier.

6.23.198 kCGImagePropertyIPTCObjectAttributeReference as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The object attribute.

6.23.199 kCGImagePropertyIPTCObjectCycle as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The object cycle.

6.23.200 kCGImagePropertyIPTCObjectName as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The object name.

6.23.201 kCGImagePropertyIPTCObjectTypeReference as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The object type.

6.23.202 kCGImagePropertyIPTCOriginalTransmissionReference as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The original transmission reference.

6.23.203 kCGImagePropertyIPTCOriginatingProgram as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The originating program.

6.23.204 kCGImagePropertyIPTCProgramVersion as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The program version.

6.23.205 kCGImagePropertyIPTCProvinceState as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The province or state.

6.23.206 kCGImagePropertyIPTCReferenceDate as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The reference date.

6.23.207 kCGImagePropertyIPTCReferenceNumber as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The reference number.

6.23.208 kCGImagePropertyIPTCReferenceService as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The reference service.

6.23.209 `kCGImagePropertyIPTCReleaseDate` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The release date.

6.23.210 `kCGImagePropertyIPTCReleaseTime` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The release time.

6.23.211 `kCGImagePropertyIPTCRightsUsageTerms` as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: The usage rights for the image.

Available in OS X v10.6 and later.

6.23.212 `kCGImagePropertyIPTCScene` as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: The scene codes for the image; a scene code is a six-digit string.

Available in OS X v10.6 and later.

6.23.213 `kCGImagePropertyIPTCSource` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The source.

6.23.214 kCGImagePropertyIPTCSpecialInstructions as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: Special instructions.

6.23.215 kCGImagePropertyIPTCStarRating as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The star rating.

6.23.216 kCGImagePropertyIPTCSubjectReference as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The subject.

6.23.217 kCGImagePropertyIPTCSubLocation as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The sublocation.

6.23.218 kCGImagePropertyIPTCSupplementalCategory as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: A supplemental category.

6.23.219 kCGImagePropertyIPTCTimeCreated as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The time created.

6.23.220 `kCGImagePropertyIPTCUrgency` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The urgency level.

6.23.221 `kCGImagePropertyIPTCWriterEditor` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses International Press Telecommunications Council (IPTC) metadata.

Notes: The writer or editor.

6.23.222 `kCGImagePropertyIsFloat` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: Whether or not the image contains floating-point pixel samples. The value of this key is `kCFBooleanTrue` if the image contains them.

6.23.223 `kCGImagePropertyIsIndexed` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: Whether or not the image contains indexed pixel samples (sometimes called paletted samples). The value of this key is `kCFBooleanTrue` if the image contains them.

6.23.224 `kCGImagePropertyJFIFDensityUnit` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses JPEG File Interchange Format (JFIF).

Notes: The density unit.

6.23.225 kCGImagePropertyJFIFDictionary as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: A dictionary of key-value pairs for an image that uses JPEG File Interchange Format (JFIF).

6.23.226 kCGImagePropertyJFIFIsProgressive as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses JPEG File Interchange Format (JFIF).

Notes: Whether or not the image is progressive.

6.23.227 kCGImagePropertyJFIFVersion as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses JPEG File Interchange Format (JFIF).

Notes: The version.

6.23.228 kCGImagePropertyJFIFXDensity as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses JPEG File Interchange Format (JFIF).

Notes: The x density.

6.23.229 kCGImagePropertyJFIFYDensity as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses JPEG File Interchange Format (JFIF).

Notes: The y density.

6.23.230 kCGImagePropertyMakerCanonAspectRatioInfo as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: The image aspect ratio.

6.23.231 `kCGImagePropertyMakerCanonCameraSerialNumber` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: The camera serial number.

6.23.232 `kCGImagePropertyMakerCanonContinuousDrive` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: The presence of a continuous drive.

6.23.233 `kCGImagePropertyMakerCanonDictionary` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: A dictionary of key-value pairs for an image from a Canon camera.

Available in Mac OS X v10.5 and later.

6.23.234 `kCGImagePropertyMakerCanonFirmware` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: The firmware version.

6.23.235 `kCGImagePropertyMakerCanonFlashExposureComp` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: The flash exposure compensation.

6.23.236 kCGImagePropertyMakerCanonImageSerialNumber as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: The image serial number.

6.23.237 kCGImagePropertyMakerCanonLensModel as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: The lens model.

6.23.238 kCGImagePropertyMakerCanonOwnerName as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: The owner name.

6.23.239 kCGImagePropertyMakerFujiDictionary as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: A dictionary of key-value pairs for an image from a Fuji camera.

Available in Mac OS X v10.5 and later.

6.23.240 kCGImagePropertyMakerMinoltaDictionary as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: A dictionary of key-value pairs for an image from a Minolta camera.

Available in Mac OS X v10.5 and later.

6.23.241 kCGImagePropertyMakerNikonCameraSerialNumber as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: The camera serial number.

6.23.242 `kCGImagePropertyMakerNikonColorMode` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: The color mode.

6.23.243 `kCGImagePropertyMakerNikonDictionary` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: A dictionary of key-value pairs for an image from a Nikon camera.

Available in Mac OS X v10.5 and later.

6.23.244 `kCGImagePropertyMakerNikonDigitalZoom` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: The digital zoom setting.

6.23.245 `kCGImagePropertyMakerNikonFlashExposureComp` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: The flash exposure compensation.

6.23.246 `kCGImagePropertyMakerNikonFlashSetting` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: The flash setting.

6.23.247 kCGImagePropertyMakerNikonFocusDistance as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: The focus distance.

6.23.248 kCGImagePropertyMakerNikonFocusMode as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: The focus mode.

6.23.249 kCGImagePropertyMakerNikonImageAdjustment as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: Image adjustment setting.

6.23.250 kCGImagePropertyMakerNikonISOSelection as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: The ISO selection.

6.23.251 kCGImagePropertyMakerNikonISOSetting as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: The ISO setting.

6.23.252 kCGImagePropertyMakerNikonLensAdapter as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: The lens adapter.

6.23.253 `kCGImagePropertyMakerNikonLensInfo` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: Lens information.

6.23.254 `kCGImagePropertyMakerNikonLensType` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: The lens type.

6.23.255 `kCGImagePropertyMakerNikonQuality` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: The quality setting.

6.23.256 `kCGImagePropertyMakerNikonSharpenMode` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: The sharpening mode.

6.23.257 `kCGImagePropertyMakerNikonShootingMode` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: The shooting mode.

6.23.258 kCGImagePropertyMakerNikonShutterCount as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: The shutter count.

6.23.259 kCGImagePropertyMakerNikonWhiteBalanceMode as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image from a Nikon camera.

Notes: The white balance mode.

6.23.260 kCGImagePropertyMakerOlympusDictionary as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: A dictionary of key-value pairs for an image from a Olympus camera.

Available in Mac OS X v10.5 and later.

6.23.261 kCGImagePropertyMakerPentaxDictionary as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: A dictionary of key-value pairs for an image from a Pentax camera.

Available in Mac OS X v10.5 and later.

6.23.262 kCGImagePropertyOrientation as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: The intended display orientation of the image. If present, this key is a CFNumber value with the same value as defined by the TIFF and EXIF specifications. The value specifies where the origin (0,0) of the image is located, as shown in Table 1. If not present, a value of 1 is assumed.

Value	Location of the origin of the image
1	Top, left
2	Top, right
3	Bottom, right
4	Bottom, left
5	Left, top
6	Right, top
7	Right, bottom
8	Left, bottom

6.23.263 kCGImagePropertyPixelHeight as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Example:

```
// open an image file
Var path as string = "/Library/Desktop Pictures/Galaxy.jpg"
Var f as FolderItem = GetFolderItem(path, FolderItem.PathTypeNative)
Var c As New CGImageSourceMBS(f)

// properties for first image
Var p as Dictionary = c.PropertiesAtIndex(0)
Var w as Integer = p.Value(c.kCGImagePropertyPixelWidth)
Var h as Integer = p.Value(c.kCGImagePropertyPixelHeight)

// show size
MsgBox str(w)+" x "+str(h)
```

Notes: The number of pixels in the y dimension. If present, this key is a CFNumber value.

6.23.264 kCGImagePropertyPixelWidth as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Example:

```
// open an image file
Var path as string = "/Library/Desktop Pictures/Galaxy.jpg"
Var f as FolderItem = GetFolderItem(path, FolderItem.PathTypeNative)
Var c As New CGImageSourceMBS(f)
```

```
// properties for first image
Var p as Dictionary = c.PropertiesAtIndex(0)
Var w as Integer = p.Value(c.kCGImagePropertyPixelWidth)
Var h as Integer = p.Value(c.kCGImagePropertyPixelHeight)

// show size
MsgBox str(w)+" x "+str(h)
```

Notes: The number of pixels in the x dimension. If present, this key is a CFNumber value.

6.23.265 kCGImagePropertyPNGAuthor as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: The content is a string that identifies the author of the image.
Available in OS X v10.7 and later.

6.23.266 kCGImagePropertyPNGChromaticities as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Portable Network Graphics (PNG) format.

Notes: The chromaticities.

6.23.267 kCGImagePropertyPNGCopyright as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: The content is a string that identifies the copyright of the image.
Available in OS X v10.7 and later.

6.23.268 kCGImagePropertyPNGCreationTime as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: The content is a string that identifies the date and time the image was created.

Available in OS X v10.7 and later.

6.23.269 `kCGImagePropertyPNGDescription` as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: The content is a string that describes the image.

Available in OS X v10.7 and later.

6.23.270 `kCGImagePropertyPNGDictionary` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: A dictionary of key-value pairs for an image that uses Portable Network Graphics (PNG) format.

6.23.271 `kCGImagePropertyPNGGamma` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Portable Network Graphics (PNG) format.

Notes: The gamma value.

6.23.272 `kCGImagePropertyPNGInterlaceType` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Portable Network Graphics (PNG) format.

Notes: The interlace type.

6.23.273 `kCGImagePropertyPNGModificationTime` as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: The content is a string that identifies the last date and time the image was modified.

Available in OS X v10.7 and later.

6.23.274 kCGImagePropertyPNGSoftware as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: The content is a string that identifies the software used to create the image. Available in OS X v10.7 and later.

6.23.275 kCGImagePropertyPNGsRGBIntent as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Portable Network Graphics (PNG) format.

Notes: The sRGB intent.

6.23.276 kCGImagePropertyPNGTitle as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: The content is a string that holds the image's title. Available in OS X v10.7 and later.

6.23.277 kCGImagePropertyPNGXPixelsPerMeter as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Portable Network Graphics (PNG) format.

Notes: The number of x pixels per meter.

6.23.278 kCGImagePropertyPNGYPixelsPerMeter as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Portable Network Graphics (PNG) format.

Notes: The number of y pixels per meter.

6.23.279 kCGImagePropertyProfileName as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: The name of the optional ICC profile embedded in the image, if known. If present, the value of this key is a CFStringRef.

6.23.280 kCGImagePropertyRawDictionary as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Notes: A dictionary of key-value pairs for an image that contains minimally processed, or raw, data.

6.23.281 kCGImagePropertyTIFFArtist as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: The artist.

6.23.282 kCGImagePropertyTIFFCompression as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: The compression scheme used on the image data.

6.23.283 kCGImagePropertyTIFFCopyright as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: Copyright information.

6.23.284 kCGImagePropertyTIFFDateTime as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: The date and time.

6.23.285 kCGImagePropertyTIFFDictionary as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: One of the keys for the properties dictionary.

Example:

```
// Change rotation in an image file

// files
Var dpath as folderitem = SpecialFolder.Desktop.Child("test.jpg")
Var opath as folderitem = dpath.parent.Child("output.jpg")

// open source
Var imageSource as new cgimagesourcembs(dpath)

// read image
Var img as cgimageembs = imageSource.createimageatindex(0)

// global properties
Var globalprop as dictionary = imageSource.properties
// per image properties
Var p as Dictionary = imageSource.PropertiesAtIndex(0)

Var imageDest as new CGImageDestinationMBS(opath,"public.jpeg",1)

'const orientation = 1 // top left
const orientation = 3 // bottom right

// set globals
imageDest.SetProperties(globalprop)

// now set new orientation
p.value(imageDest.kCGImageDestinationOrientation) = orientation
p.value(imageDest.kCGImageDestinationMergeMetadata) = true

// change tiff dictionary, if present
Var dTIFF as Dictionary = p.lookup(imageSource.kCGImagePropertyTIFFDictionary, nil)
if dTIFF <> nil then
dTIFF.value(imageSource.kCGImagePropertyTIFFOrientation) = orientation
end if

// change iptc dictionary, if present
Var dIPTC as Dictionary = p.lookup(imageSource.kCGImagePropertyIPTCDictionary, nil)
if dIPTC <> nil then
dIPTC.value(imageSource.kCGImagePropertyIPTCImageOrientation) = orientation
end if

// write out image
```

`imageDest.AddImage(img,p)`
`call imageDest.Finalize`

Notes: A dictionary of key-value pairs for an image that uses Tagged Image File Format (TIFF).

6.23.286 `kCGImagePropertyTIFFDocumentName` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: The document name.

6.23.287 `kCGImagePropertyTIFFHostComputer` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: The computer or operation system used when the image was created.

6.23.288 `kCGImagePropertyTIFFImageDescription` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: The image description.

6.23.289 `kCGImagePropertyTIFFMake` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: The camera or input device make.

6.23.290 `kCGImagePropertyTIFFModel` as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: A camera or input device model.

6.23.291 kCGImagePropertyTIFFOrientation as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Example:

```
// Change rotation in an image file

// files
Var dpath as folderitem = SpecialFolder.Desktop.Child("test.jpg")
Var opath as folderitem = dpath.parent.Child("output.jpg")

// open source
Var imageSource as new cgimagesourcembs(dpath)

// read image
Var img as cgimageembs = imageSource.createimageatindex(0)

// global properties
Var globalprop as dictionary = imageSource.properties
// per image properties
Var p as Dictionary = imageSource.PropertiesAtIndex(0)

Var imageDest as new CGImageDestinationMBS(opath,"public.jpeg",1)

'const orientation = 1 // top left
const orientation = 3 // bottom right

// set globals
imageDest.SetProperties(globalprop)

// now set new orientation
p.value(imageDest.kCGImageDestinationOrientation) = orientation
p.value(imageDest.kCGImageDestinationMergeMetadata) = true

// change tiff dictionary, if present
Var dTIFF as Dictionary = p.lookup(imageSource.kCGImagePropertyTIFFDictionary, nil)
if dTIFF <> nil then
dTIFF.value(imageSource.kCGImagePropertyTIFFOrientation) = orientation
end if

// change iptc dictionary, if present
Var dIPTC as Dictionary = p.lookup(imageSource.kCGImagePropertyIPTCDictionary, nil)
if dIPTC <> nil then
dIPTC.value(imageSource.kCGImagePropertyIPTCImageOrientation) = orientation
```

end if

```
// write out image
imageDest.AddImage(img,p)
call imageDest.Finalize
```

Notes: The image orientation.

6.23.292 kCGImagePropertyTIFFPhotometricInterpretation as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: The color space of the image data.

6.23.293 kCGImagePropertyTIFFPrimaryChromaticities as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: The chromaticities of the primaries of the image.

6.23.294 kCGImagePropertyTIFFResolutionUnit as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: The units of resolution.

6.23.295 kCGImagePropertyTIFFSoftware as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: The name and version of the software used for image creation.

6.23.296 kCGImagePropertyTIFFTransferFunction as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: The transfer function, in tabular format, used to map pixel components from a nonlinear form into a linear form.

6.23.297 kCGImagePropertyTIFFWhitePoint as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: The white point.

6.23.298 kCGImagePropertyTIFFXResolution as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: The number of pixels per resolution unit in the image width direction.

6.23.299 kCGImagePropertyTIFFYResolution as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Keys for an image that uses Tagged Image File Format (TIFF).

Notes: The number of pixels per resolution unit in the image height direction.

6.23.300 kCGImageSourceCreateThumbnailFromImageAlways as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: Specifies whether a thumbnail should be created from the full image even if a thumbnail is present in the image source file. The thumbnail will be created from the full image, subject to the limit specified by `kCGImageSourceThumbnailMaxPixelSize` — if a maximum pixel size isn't specified, then the thumbnail will be the size of the full image, which probably isn't what you want. The value of this key must be a Boolean; the default value of this key is `False`.

6.23.301 kCGImageSourceCreateThumbnailFromImageIfAbsent as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: Specifies whether a thumbnail should be automatically created for an image if a thumbnail isn't present in the image source file. The thumbnail will be created from the full image, subject to the limit specified by `kCGImageSourceThumbnailMaxPixelSize`—if a maximum pixel size isn't specified, then the thumbnail will be the size of the full image, which probably isn't what you want. The value of this key must be a Boolean; the default value of this key is `False`.

6.23.302 `kCGImageSourceCreateThumbnailWithTransform` as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: Specifies whether the thumbnail should be rotated and scaled according to the orientation and pixel aspect ratio of the full image. The value of this key must be a Boolean; the default value of this key is `False`.

6.23.303 kCGImageSourceShouldAllowFloat as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: Specifies whether the image should be returned as a floating point CGImage if supported by the file format. Extended range floating point CGImage may require additional processing to render pleasingly. The value of this key must be a Boolean; the default value is False.

6.23.304 kCGImageSourceShouldCache as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: Specifies whether the image should be cached in a decoded form. The value of this key must be a Boolean; the default value is False.

6.23.305 kCGImageSourceShouldCacheImmediately as string

Plugin Version: 17.0, Platform: macOS, Targets: All.

Function: One of the keys for the options dictionary.

Notes: Specifies whether image decoding and caching should happen at image creation time. The value of this key must be a boolean. The default value is kCFBooleanFalse (image decoding will happen at rendering time).

6.23.306 kCGImageSourceThumbnailMaxPixelSize as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: Specifies the maximum width and height in pixels of a thumbnail. If this key is not specified, the width and height of a thumbnail is not limited and thumbnails may be as big as the image itself. If present, this value of this key must be an Integer.

6.23.307 kCGImageSourceTypeIdentifierHint as string

Plugin Version: 13.1, Platform: macOS, Targets: All.

Function: One of the key names for the properties.

Notes: Specifies the "best guess" of the type identifier for the format of the image source file. If specified, the value of this key must be a String. For more information about type identifiers, see "UTType.h" in the

Application Services framework.

6.23.308 Properties(options as dictionary = nil) as dictionary

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns the properties of the image source.

Example:

```

Var inputfile as FolderItem = SpecialFolder.Desktop.Child("test.jpg")

// reading the picture
Var c as new CGImageSourceMBS(inputfile)
Var img as CGImageMBS = c.CreateImageAtIndex(0)
Var propertiesGlobal as Dictionary = c.Properties
Var propertiesImage as Dictionary = c.PropertiesAtIndex(0)

Var outputFile as FolderItem = SpecialFolder.Desktop.Child("output.jpg")
Var d as new CGImageDestinationMBS(outputFile, "public.jpeg", 1)

// writing the picture and include metadata
d.SetProperties(propertiesGlobal)
d.AddImage(img, propertiesImage)
if d.FinalizeMT then
outputFile.Launch
else
MsgBox "Failed to write jpeg."
end if

```

Notes: options: A dictionary you can use to request additional options.

Returns a dictionary that contains the properties associated with the image source container.

These properties apply to the container in general but not necessarily to any individual image contained in the image source.

Available in Mac OS X version 10.4 and later.

6.23.309 PropertiesAtIndex(index as Integer, options as dictionary = nil) as dictionary

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns the properties of the image at a specified location in an image source.

Example:

```

Var file as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
Var c as new CGImageSourceMBS(file)

```

```

Var d as Dictionary = c.PropertiesAtIndex(0)

Var e as Dictionary
e=d.Value(c.kCGImagePropertyExifDictionary)

if e<>Nil then
  Var keys(-1) as Variant = e.Keys
  Var lines(-1) as string

  for each key as string in keys
    lines.Append key+": "+e.Value(key)
  next

  MsgBox "Exif: "+EndOfLine+EndOfLine+Join(lines,EndOfLine)
end if

e=d.Value(c.kCGImagePropertyGPSDictionary)

if e<>Nil then
  Var keys(-1) as Variant = e.Keys
  Var lines(-1) as string

  for each key as string in keys
    lines.Append key+": "+e.Value(key)
  next

  MsgBox "GPS: "+EndOfLine+EndOfLine+Join(lines,EndOfLine)
end if

```

Notes: index: The index of the image whose properties you want to obtain. The index is zero-based.
options: A dictionary you can use to request additional options.
Returns a dictionary that contains the properties associated with the image.
Available in Mac OS X version 10.4 and later.

6.23.310 PropertiesAtIndexCF(index as Integer, options as Variant = nil) as Variant

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Returns the properties of the image at a specified location in an image source.
Notes: index: The index of the image whose properties you want to obtain. The index is zero-based.
options: A dictionary or CFDictionaryMBS you can use to request additional options.
Returns a CFDictionaryMBS that contains the properties associated with the image.
Available in Mac OS X version 10.4 and later.

6.23.311 PropertiesCF(options as Variant = nil) as Variant

Plugin Version: 13.5, Platform: macOS, Targets: All.

Function: Returns the properties of the image source.

Notes: options: A dictionary or NSDictionaryMBS you can use to request additional options.

Returns a NSDictionaryMBS that contains the properties associated with the image source container.

These properties apply to the container in general but not necessarily to any individual image contained in the image source.

Available in Mac OS X version 10.4 and later.

6.23.312 StatusAtIndex(index as Integer) as Integer

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns the current status of an image that is at a specified location in an image source.

Notes: index: The index of the image whose status you want to obtain. The index is zero-based.

Returns the current status of the image.

The status is particularly informative for incremental image sources, but may also be used by clients that provide non-incremental data.

Available in Mac OS X version 10.4 and later.

6.23.313 TypeIdentifiers as string()

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns an array of uniform type identifiers (UTIs) that are supported for image sources.

Example:

```
Var s(-1) as string = CGImageSourceMBS.TypeIdentifiers
```

```
MsgBox Join(s,EndOfLine)
```

```
// shows:
//
// public.png
// public.jpeg
// com.compuserve.gif
```

```
// public.jpeg-2000
// com.adobe.raw-image
// com.leafamerica.raw-image
// com.hasselblad.fff-raw-image
// com.hasselblad.3fr-raw-image
// com.nikon.raw-image
// com.nikon.nrw-raw-image
// com.pentax.raw-image
// com.sony.sr2-raw-image
// com.sony.arw-raw-image
// com.epson.raw-image
// com.kodak.raw-image
// public.tiff
// com.apple.icns
// com.canon.tif-raw-image
// com.canon.cr2-raw-image
// com.canon.crw-raw-image
// com.fuji.raw-image
// com.panasonic.raw-image
// com.panasonic.rw2-raw-image
// com.panasonic.rwl-raw-image
// com.leica.pwl-raw-image
// com.konicaminolta.raw-image
// com.olympus.raw-image
// com.olympus.raw-image
// com.sony.raw-image
// com.adobe.photoshop-image
// com.adobe.pdf
// com.adobe.illustrator.ai-image
// com.microsoft.ico
// com.microsoft.bmp
// public.xbitmap-image
// com.microsoft.cur
// com.apple.pict
// com.truevision.tga-image
// com.sgi.sgi-image
// com.apple.quicktime-image
// com.kodak.flashpix-image
// com.apple.macpaint-image
// com.ilm.openexr-image
// public.radiance
```

Notes: Returns an array of the UTIs that are supported for image sources.

See Uniform Type Identifiers Overview for a list of system-declared and third-party UTIs.

Available in Mac OS X version 10.4 and later.

6.23.314 UpdateData(data as string, final as boolean)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Updates an incremental image source with new data.

Notes: data: The data to add to the image source. Each time you call the function UpdateData, the data parameter must contain all of the image file data accumulated so far.

final: A value that specifies whether the data is the final set. Pass true if it is, false otherwise.

Available in Mac OS X version 10.4 and later.

6.23.315 Properties

6.23.316 Count as Integer

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns the number of images (not including thumbnails) in the image source.

Example:

```
Var file as FolderItem = SpecialFolder.Desktop.Child("test.jpg")
Var c as new CGImageSourceMBS(file)
```

```
MsgBox str(c.Count)+" images in this file."
```

Notes: The number of images. If the image source is a multilayered PSD file, the function returns 1.

This function does not extract the layers of a PSD file.

Available in Mac OS X version 10.4 and later.
(Read only property)

6.23.317 Handle as Integer

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The internal reference.

Notes: (Read and Write property)

6.23.318 Status as Integer

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Return the status of an image source.

Notes: Returns the current status of the image source.

The status is particularly informative for incremental image sources, but may also be used by clients that provide non-incremental data.

Available in Mac OS X version 10.4 and later.

(Read only property)

6.23.319 Type as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns the uniform type identifier of the source container.

Notes: The uniform type identifier of the image.

The uniform type identifier (UTI) of the source container can be different from the type of the images in the container. For example, the .icns format supports embedded JPEG2000. The type of the source container is "com.apple.icns" but type of the images is JPEG2000.

Available in Mac OS X version 10.4 and later.

(Read only property)

6.23.320 Constants

Constants

Constant	Value	Description
kCGImageStatusComplete	0	One of the status constants. The operation is complete.
kCGImageStatusIncomplete	-1	One of the status constants. The operation is not complete.
kCGImageStatusInvalidData	-4	One of the status constants. The data is not valid.
kCGImageStatusReadingHeader	-2	One of the status constants. In the process of reading the header.
kCGImageStatusUnexpectedEOF	-5	One of the status constants. The end of the file was encountered unexpectedly.
kCGImageStatusUnknownType	-3	One of the status constants. The image is an unknown type.

6.24 class CGLayerMBS

6.24.1 class CGLayerMBS

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: The CoreGraphics layer class, similar to a template.

Notes: CGLayer objects are useful for offscreen drawing and can be used in much the same way that a bitmap context can be used. In fact, a CGLayer object is a much better representation than a bitmap context.

Using CGLayer objects can improve performance, particularly when you need to capture a piece of drawing that you stamp repeatedly (using the same scale factor and orientation). Quartz can cache CGLayer objects to the video card, making drawing a CGLayer to a destination much faster than rendering the equivalent image constructed from a bitmap context.

A CGLayer object is created relative to a graphics context. Although layer uses this graphics context as a reference for initialization, you are not restricted to drawing the layer to this graphics context. You can draw the layer to other graphics contexts, although any limitations of the original context are imposed. For example, if you create a CGLayer object using a bitmap context, the layer is rendered as a bitmap when drawn to any other graphics context.

You can use a CGLayer when you want to apply a shadow to a group of objects (such as a group of circles) rather than to individual objects.

Use these layers in your code whenever you can, especially when:

- You need to reuse a filled or stroked shape.
- You are building a scene and at least some of it can be reused. Put the reusable drawing in its own CGLayer.

Any CG object that you draw repeatedly—including CGPath, CGShading, and CGPDFPage—benefit from improved performance if you draw it to a CGLayer object.

see also

https://developer.apple.com/library/mac/#documentation/graphicsimaging/conceptual/drawingwithquartz2d/dq_layers/dq_layers.html

Blog Entries

- [MBS Real Studio Plugins, version 12.2pr3](#)

6.24.2 Methods

6.24.3 Constructor(context as CGContextMBS, size as CGSizeMBS, auxiliary-Info as dictionary = nil)

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: Creates a CGLayer object that is associated with a graphics context.

Notes: context: The graphics context you want to create the layer relative to. The layer uses this graphics context as a reference for initialization.

size: The size, in default user space units, of the layer relative to the graphics context.

auxiliaryInfo: Reserved for future use. Pass nil.

On Success the handle property is not nil.

After you create a CGLayer object, you should reuse it whenever you can to facilitate the Quartz caching strategy. Quartz caches any objects that are reused, including CGLayer objects. Objects that are reused frequently remain in the cache. In contrast, objects that are used once in a while may be moved in and out of the cache according to their frequency of use. If you don't reuse CGLayer objects, Quartz won't cache them. This means that you lose an opportunity to improve the performance of your application.

Available in Mac OS X version 10.4 and later.

See also:

- 6.24.4 Constructor(context as CGContextMBS, width as Double, height as Double, auxiliaryInfo as dictionary = nil) 370

6.24.4 Constructor(context as CGContextMBS, width as Double, height as Double, auxiliaryInfo as dictionary = nil)

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: Creates a CGLayer object that is associated with a graphics context.

Notes: context: The graphics context you want to create the layer relative to. The layer uses this graphics context as a reference for initialization.

width/height: The size, in default user space units, of the layer relative to the graphics context.

auxiliaryInfo: Reserved for future use. Pass nil.

On Success the handle property is not nil.

After you create a CGLayer object, you should reuse it whenever you can to facilitate the Quartz caching strategy. Quartz caches any objects that are reused, including CGLayer objects. Objects that are reused frequently remain in the cache. In contrast, objects that are used once in a while may be moved in and out of the cache according to their frequency of use. If you don't reuse CGLayer objects, Quartz won't cache

them. This means that you lose an opportunity to improve the performance of your application.

Available in Mac OS X version 10.4 and later.

See also:

- 6.24.3 Constructor(context as CGContextMBS, size as CGSizeMBS, auxiliaryInfo as dictionary = nil)
370

6.24.5 Context as CGContextMBS

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: Returns the graphics context associated with a CGLayer object.

Notes: The context that's returned is the context for the layer itself, not the context that you specified when you created the layer.

6.24.6 Size as CGSizeMBS

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: Returns the width and height of a CGLayer object.

6.24.7 Properties

6.24.8 Handle as Integer

Plugin Version: 12.2, Platform: macOS, Targets: All.

Function: Internal object reference.

Notes: (Read and Write property)

6.25 class CGMutableImageMetadataMBS

6.25.1 class CGMutableImageMetadataMBS

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: The class for a mutable tag container.

Notes: Subclass of the CGImageMetadataMBS class.

Blog Entries

- [MBS Xojo Plugins, version 18.5pr8](#)

6.25.2 Methods

6.25.3 Constructor

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Creates an empty metadata object.

See also:

- [6.25.4 Constructor\(metadata as CGImageMetadataMBS\)](#) 372

6.25.4 Constructor(metadata as CGImageMetadataMBS)

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Copy constructor.

Notes: Creates a copy of the metadata.

See also:

- [6.25.3 Constructor](#) 372

6.25.5 RegisterNamespaceForPrefix(parent as CGImageMetadataTagMBS = nil, xmlns as String, prefix as String, byref error as CFErrorMBS) as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Associates an XMP namespace URI with a prefix string.

Notes: This allows ImageIO to create custom metadata when it encounters an unrecognized prefix in a path (see TagWithPath for more information about path syntax). A namespace must be registered before it can be used to add custom metadata. All namespaces found in the image's metadata, or defined as a constant above, will be pre-registered. Namespaces and prefixes must be unique.

Returns true if successful. Returns false and sets 'error' if an error or conflict occurs.

6.25.6 RemoveTagWithPath(parent as CGImageMetadataTagMBS = nil, path as string) as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Sets the tag at a specific path in a CGMutableImageMetadataMBS container or a parent tag

Notes: This is the primary function for adding new metadata tags to a metadata container, or updating existing tags. All tags required to reach the final tag (at the end of the path) will be created, if needed. Tags will be created with default types (ordered arrays). Creating tags will fail if a prefix is encountered that has not been registered. Use RegisterNamespaceForPrefix to associate a prefix with a namespace prior to using a path-based CGImageMetadataMBS function.

Note that if a parent tag is provided, the children of that tag reference will be modified, which may be a different reference from the tag stored in the metadata container. Since tags are normally obtained as a copy, it is typically necessary to use SetTagWithPath to commit the changed parent object back to the metadata container (using the parent's path and nil for the parent).

metadata: A mutable collection of metadata tags.

parent: A parent tag. If nil, the path is relative to the root of the metadata (i.e. it is not a child of another property).

Note that if a parent tag is provided, the children of that tag reference will be modified, which may be a different reference from the tag stored in the metadata container. Since tags are normally obtained as a copy, it is typically necessary to use SetTagWithPath to commit the changed parent object back to the metadata container (using the parent's path and nil for the parent).

path: A string with the path to the desired tag. Please consult the documentation of TagWithPath for information about path syntax.

Value: The CGImageMetadataTagMBS object to be added to the metadata. The tag will be retained.

Returns true if successful, false otherwise.

6.25.7 SetTagWithPath(parent as CGImageMetadataTagMBS = nil, path as string, value as CGImageMetadataTagMBS) as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Sets the value of the tag at a specific path in a CGMutableImageMetadataMBS container or a parent tag.

Notes: This function is used to add new metadata values to a metadata container, or update existing tag values. All tags required to reach the final tag (at the end of the path) are created, if needed. Tags are created with default types (i.e. arrays will be ordered). Creating tags will fail if a prefix is encountered that

has not been registered. Use `RegisterNamespaceForPrefix` to associate a prefix with a namespace prior to using a path-based `CGImageMetadata` function.

Examples

- `'path' = "xmp:CreateDate", 'value' = "2011-09-20T14:54:47-08:00")`
- `'path' = "dc:subject [0] ", 'value' = "San Francisco")`
- `'path' = "dc:subject [1] ", 'value' = "Golden Gate Bridge")`
- `'path' = "dc:description [en] " 'value' = "my image description")`
- `'path' = "dc:description [de] " 'value' = "meine bildbeschreibung")`

Note that if a parent tag is provided, the children of that tag reference will be modified, which may be a different reference from the tag stored in the metadata container. Since tags are normally obtained as a copy, it is typically necessary to use `SetTagWithPath` to commit the changed parent object back to the metadata container (using the parent's path and `""` for the parent).

parent: A parent tag. If nil, the path is relative to the root of the metadata (i.e. it is not a child of another property).

path: A string with the path to the desired tag. Please consult the documentation of `TagWithPath` for information about path syntax.

value: The value to be added to the `CGImageMetadataTag` matching the path.

The tag will be retained. The restrictions for the value are the same as in `Create`. Returns true if successful, false otherwise.

6.25.8 `SetValueMatchingImageProperty(dictionaryName as String, propertyName as String, value as CGImageMetadataTagMBS) as Boolean`

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Sets the value of the `CGImageMetadataTag` matching a `kCGImageProperty` constant.

Notes: Provides a bridge for values from `CGImageCopyPropertiesAtIndex`, simplifying changing property values defined in EXIF and IPTC standards, which have no notion of namespaces, prefixes, or XMP property types.

Metadata Working Group guidance is factored into the mapping of `CGImageProperties` to XMP compatible `CGImageMetadataTags`.

For example, setting `kCGImagePropertyExifDateTimeOriginal` will set the value of the corresponding XMP tag, which is `photoshop:DateCreated`. Note that property values should still be in their XMP forms, such as

”YYYY-MM-DDThh:mm:ss” for DateTime, rather than the EXIF or IPTC DateTime formats. Although this function will allow the caller to set custom values for these properties, you should consult the appropriate specifications for details about property value formats for EXIF and IPTC tags in XMP.

dictionaryName the metadata subdictionary to which the image property belongs, such as kCGImagePropertyExifDictionary or kCGImagePropertyIPTCDictionary. Not all dictionaries and properties are supported at this time.

propertyName the name of the property. This must be a defined property constant corresponding to the 'dictionaryName'. For example, kCGImagePropertyTIFFOrientation, kCGImagePropertyExifDateTimeOriginal, or kCGImagePropertyIPTCKeywords. A warning will be logged if the CGImageProperty is unsupported by CGImageMetadata.

value: A variant with the value for the tag. The same value restrictions apply as in Create.

Returns true if successful, false otherwise.

6.25.9 SetValueWithPath(parent as CGImageMetadataTagMBS = nil, path as string, value as Variant) as Boolean

Plugin Version: 18.5, Platform: macOS, Targets: All.

Function: Sets the value of the tag at a specific path in a CGMutableImageMetadataMBS container or a parent tag.

Notes: This function is used to add new metadata values to a metadata container, or update existing tag values. All tags required to reach the final tag (at the end of the path) are created, if needed. Tags are created with default types (i.e. arrays will be ordered). Creating tags will fail if a prefix is encountered that has not been registered. Use RegisterNamespaceForPrefix to associate a prefix with a namespace prior to using a path-based CGImageMetadata function.

Examples

- 'path' = "xmp:CreateDate", 'value' = "2011-09-20T14:54:47-08:00")
- 'path' = "dc:subject [0] ", 'value' = "San Francisco")
- 'path' = "dc:subject [1] ", 'value' = "Golden Gate Bridge")
- 'path' = "dc:description [en] " 'value' = "my image description")
- 'path' = "dc:description [de] " 'value' = "meine bildbeschreibung")

Note that if a parent tag is provided, the children of that tag reference will be modified, which may be a different reference from the tag stored in the metadata container. Since tags are normally obtained as a copy, it is typically necessary to use SetTagWithPath to commit the changed parent object back to the metadata

container (using the parent's path and "" for the parent).

parent: A parent tag. If nil, the path is relative to the root of the metadata (i.e. it is not a child of another property).

path: A string with the path to the desired tag. Please consult the documentation of TagWithPath for information about path syntax.

value: The value to be added to the CGImageMetadataTag matching the path.

The tag will be retained. The restrictions for the value are the same as in Create.

Returns true if successful, false otherwise.

6.26 class CGMutablePathMBS

6.26.1 class CGMutablePathMBS

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: A class for a CoreGraphics mutable path.

Notes: Subclass of the CGPathMBS class.

6.26.2 Methods

6.26.3 AddArc(transform as CGAffineTransformMBS, x as Double, y as Double, radius as Double, startAngle as Double, endAngle as Double, clockwise as boolean)

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Add an arc of a circle to path, possibly preceded by a straight line segment.

Notes: The arc is approximated by a sequence of cubic Bezier curves. (x, y) is the center of the arc; radius is its radius; startAngle is the angle to the first endpoint of the arc; endAngle is the angle to the second endpoint of the arc; and clockwise is true if the arc is to be drawn clockwise, false otherwise. startAngle and endAngle are measured in radians. If m is not nil, then the constructed Bezier curves representing the arc will be transformed by the matrix before they are added to path.

6.26.4 AddArcToPoint(transform as CGAffineTransformMBS, x as Double, y as Double, x2 as Double, y2 as Double, radius as Double)

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Add an arc of a circle to path, possibly preceded by a straight line segment.

Notes: The arc is approximated by a sequence of cubic Bezier curves. (x, y) is the center of the arc; radius is its radius; startAngle is the angle to the first endpoint of the arc; endAngle is the angle to the second endpoint of the arc; and clockwise is true if the arc is to be drawn clockwise, false otherwise. startAngle and endAngle are measured in radians. If m is not nil, then the constructed Bezier curves representing the arc will be transformed by the matrix before they are added to path.

6.26.5 AddCurveToPoint(transform as CGAffineTransformMBS, cpx1 as Double, cpy1 as Double, cpx2 as Double, cpy2 as Double, x as Double, y as Double)

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Append a cubic Bezier curve from the current point to (x,y) with control points (cp1x, cp1y) and (cp2x, cp2y) in path and move the current point to (x, y).

Notes: If transform is not nil, then transform all points by the matrix first.

6.26.6 AddEllipseInRect(transform as CGAffineTransformMBS, r as CGRectMBS)

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Add an ellipse (an oval) inside rect to path.

Notes: The ellipse is approximated by a sequence of Bezier curves. The center of the ellipse is the midpoint of rect. If rect is square, then the ellipse will be circular with radius equal to one-half the width (equivalently, one-half the height) of rect. If rect is rectangular, then the major- and minor-axes will be the width and height of rect. The ellipse forms a complete subpath of path — that is, it begins with a "move to" and ends with a "close subpath" — oriented in the clockwise direction. If transform is not nil, then the constructed Bezier curves representing the ellipse will be transformed by m before they are added to path.

6.26.7 AddLineToPoint(transform as CGAffineTransformMBS, x as Double, y as Double)

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Append a straight line segment from the current point to (x, y) in the path and move the current point to (x, y).

Notes: If transform is not nil, then transform (x, y) by the matrix first.

6.26.8 AddPath(transform as CGAffineTransformMBS, path as CGPathMBS)

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Add a path to the path.

Notes: If m is not nil, then the points in the new path will be transformed by the matrix before they are added to path1.

6.26.9 AddQuadCurveToPoint(transform as CGAffineTransformMBS, cpx as Double, cpy as Double, x as Double, y as Double)

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Append a quadratic curve from the current point to (x, y) with control point (cpx, cpy) in path and move the current point to (x, y).

Notes: If transform is not nil, then transform all points by the matrix first.

6.26.10 AddRect(transform as CGAffineTransformMBS, r as CRectMBS)

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Add rect to path.

Notes: If transform is not nil, then transform the rectangle by the matrix first.

6.26.11 CloseSubpath

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Append a line from the current point to the starting point of the current subpath of path and end the subpath.

6.26.12 Constructor

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Create a mutable path.

6.26.13 ContainsPoint(transform as CGAffineTransformMBS, point as CGPointMBS, eoFill as boolean) as boolean

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Return true if point is contained in path; false otherwise.

Notes: A point is contained in a path if it is inside the painted region when the path is filled; if eoFill is true, then the even-odd fill rule is used to evaluate the painted region of the path, otherwise, the winding-number fill rule is used. If m is not nil, then the point is transformed by m before determining whether the path contains it.

6.26.14 MoveToPoint(transform as CGAffineTransformMBS, x as Double, y as Double)

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Move the current point to (x, y) in path and begin a new subpath.

Notes: If transform is not nil, then transform (x, y) by the matrix first.

6.27 class CGPathElementMBS

6.27.1 class CGPathElementMBS

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: The class for CGPath elements.

Example:

```
Var m as new CGMutablePathMBS
```

```
m.MoveToPoint nil, 10, 10
```

```
m.AddLineToPoint nil, 20, 30
```

```
Var e() as CGPathElementMBS = m.Elements
```

```
Break // see in debugger
```

Blog Entries

- [MBS Releases the MBS Xojo / Real Studio plug-ins in version 15.1](#)
- [MBS Xojo / Real Studio Plugins, version 15.1pr7](#)

Xojo Developer Magazine

- [13.4, page 10: News](#)

6.27.2 Methods

6.27.3 Point(Index as Integer) as CGPointMBS

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: Returns point with given index.

6.27.4 PointX(Index as Integer) as Double

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: Returns point x with given index.

6.27.5 PointY(Index as Integer) as Double

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: Returns point y with given index.

6.27.6 Properties

6.27.7 PointCount as Integer

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: Returns number of points in this element.

Notes: (Read and Write property)

6.27.8 Type as Integer

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: The type of this element.

Notes: (Read and Write property)

6.27.9 Constants

Type Constants

Constant	Value	Description
kTypeAddCurveToPoint	3	The path element that adds a cubic curve from the current point to the specified point. The element holds two control points and a destination point.
kTypeAddLineToPoint	1	The path element that adds a line from the current point to a new point. The element holds a single point for the destination.
kTypeAddQuadCurveToPoint	2	The path element that adds a quadratic curve from the current point to the specified point. The element holds a control point and a destination point.
kTypeCloseSubpath	4	The path element that closes and completes a subpath. The element does not contain any points.
kTypeMoveToPoint	0	The path element that starts a new subpath. The element holds a single point for the destination.

6.28 class CGPathMBS

6.28.1 class CGPathMBS

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: A class for a CoreGraphics path.

Blog Entries

- [MBS Releases the MBS Xojo / Real Studio plug-ins in version 15.1](#)
- [MBS Xojo / Real Studio Plugins, version 15.1pr7](#)
- [MBS Xojo / Real Studio Plugins, version 14.4pr10](#)
- [MBS Real Studio Plugins, version 13.1fc1](#)

Xojo Developer Magazine

- [13.4, page 10: News](#)

6.28.2 Methods

6.28.3 BoundingBox as CCGRectMBS

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Return the bounding box of path.

Notes: The bounding box is the smallest rectangle completely enclosing all points in the path, including control points for Bezier and quadratic curves. If the path is empty, then return (0,0,0,0).

6.28.4 Copy as CGPathMBS

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Creates a copy of the path.

Notes: Returns nil on any error.

6.28.5 CurrentPoint as CGPointMBS

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Return the current point of the current subpath of path.

Notes: If there is no current point, then return (0,0).

6.28.6 Elements as CGPathElementMBS()

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: Queries all the elements in a CGPath.

Example:

```
Var m as new CGMutablePathMBS
```

```
m.MoveToPoint nil, 10, 10
```

```
m.AddLineToPoint nil, 20, 30
```

```
Var e() as CGPathElementMBS = m.Elements
```

```
Break // see in debugger
```

6.28.7 EqualToPath(path as CGPathMBS) as boolean

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Return true if the path self is equal to path; false otherwise.

Notes: Returns false if the path is nil.

6.28.8 IsEmpty as boolean

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Return true if path contains no elements, false otherwise.

6.28.9 IsRect(byref rect as CGRectMBS) as boolean

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Return true if path represents a rectangle, false otherwise.

6.28.10 MutableCopy as CGMutablePathMBS

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: Create a mutable copy of path.

Notes: Returns nil on any error.

6.28.11 Properties

6.28.12 Handle as Integer

Plugin Version: 8.0, Platform: macOS, Targets: All.

Function: The handle to the internal used object.

Notes: (Read and Write property)

6.29 class CGPDFArrayMBS

6.29.1 class CGPDFArrayMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: A class for a pdf array.

6.29.2 Methods

6.29.3 ArrayValue(index as Integer, byref value as CGPDFArrayMBS) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given index is a pdf array, this function will return it.

Notes: Returns true on success and false on failure.

6.29.4 BooleanValue(index as Integer, byref value as boolean) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given index is a boolean value, this function will return it.

Notes: Returns true on success and false on failure.

6.29.5 Count as Integer

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Returns the count of items in this array.

6.29.6 DictionaryValue(index as Integer, byref value as CGPDFDictionaryMBS) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given index is a pdf dictionary, this function will return it.

Notes: Returns true on success and false on failure.

6.29.7 IntegerValue(index as Integer, byref value as Integer) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given index is an integer value, this function will return it.

Notes: Returns true on success and false on failure.

Is function will return double and integer values. Conversion is done automatically.

6.29.8 NameValue(index as Integer, byref value as string) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given index is a name string, this function will return it.

Notes: Returns true on success and false on failure.

6.29.9 NullValue(index as Integer) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given index is a nil value, this function will return it.

Notes: Returns true on success and false on failure.

6.29.10 ObjectValue(index as Integer, byref value as CGPDFObjectMBS) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given index is a pdf object, this function will return it.

Notes: Returns true on success and false on failure.

6.29.11 SingleValue(index as Integer, byref value as Double) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given index is a floating point value, this function will return it.

Notes: Returns true on success and false on failure.

Is function will return double and integer values. Conversion is done automatically.

6.29.12 StreamValue(index as Integer, byref value as CGPDFStreamMBS) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given index is a pdf stream, this function will return it.

Notes: Returns true on success and false on failure.

6.29.13 StringValue(index as Integer, byref value as CGPDFStringMBS) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given index is a pdf string, this function will return it.

Notes: Returns true on success and false on failure.

6.29.14 Properties

6.29.15 Document as CGPDFDocumentMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The document this PDF Array belongs to.

Notes: (Read only property)

6.29.16 Handle as Integer

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: The handle of the pdf array.

Notes: (Read and Write property)

6.30 class CGPDFContextMBS

6.30.1 class CGPDFContextMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: A CoreGraphics context for PDF specific functions.

Example:

```
// create pdf
Var file as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
Var r as new CGRectMBS(0,0,500,500)
Var c as CGContextMBS = file.NewCGPDFDocumentMBS(r, "My Title", "My Author", "My Creator")

if c<<>Nil then

    // create page
    c.BeginPage r

    // draw something
    c.SetRGBFillColor(1.0, 0.0, 0.0, 1.0)
    c.FillRect CGMakeRectMBS(100,100,100,100)

    // close page
    c.EndPage

    // flush and show in PDF viewer
    c = nil
    file.Launch
end if
```

Notes: This class defines functions to create and get information about a Quartz PDF context. A CGPDFContext object is a type of CGContext that is used for drawing PDF content. The functions in this reference operate only on Quartz PDF graphics contexts created using the functions CGPDFContextCreate or CGPDFContextCreateWithURL.

When you draw to the PDF context using CGContext functions the drawing operations are recorded in PDF format. The PDF commands that represent the drawing are written to the destination specified when you create the PDF graphics context.

Subclass of the CGContextMBS class.

Blog Entries

- [MBS Xojo Plugins, version 19.1pr5](#)
- [MBS Real Studio Plugins, version 12.0pr6](#)

6.30.2 Methods

6.30.3 AddDestinationAtPoint(name as string, x as Double, y as Double)

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Sets a destination to jump to when a point in the current page of a PDF graphics context is clicked.

Notes: name: A destination name.

x/y: A location in the current page of the PDF graphics context.

Available in Mac OS X v10.4 and later.

6.30.4 BeginPage(pageInfo as dictionary)

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Begins a new page in a PDF graphics context.

Notes: pageInfo: A dictionary that contains key-value pairs that define the page properties.

You must call the function EndPage to signal the end of the page.

Available in Mac OS X v10.4 and later.

6.30.5 Close

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Closes a PDF document.

Notes: After closing the context, all pending data is written to the context destination, and the PDF file is completed. No additional data can be written to the destination context after the PDF document is closed.

6.30.6 EndPage

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Ends the current page in the PDF graphics context.

Notes: You can call EndPage only after you call the function BeginPage.

6.30.7 kCGPDFContextAllowsCopying as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that used to set up a PDF context.

Notes: Whether the document allows copying when unlocked with the user password. The value of this key must be a Boolean object. The default value of this key is true.

Available in Mac OS X v10.4 and later.

6.30.8 kCGPDFContextAllowsPrinting as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that used to set up a PDF context.

Notes: Whether the document allows printing when unlocked with the user password. The value of this key must be a boolean value. The default value of this key is true.

Available in Mac OS X v10.4 and later.

6.30.9 kCGPDFContextArtBox as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that specify various PDF boxes.

Notes: The art box for the document or for a given page. This key is optional. If present, the value of this key must be a CGRectMBS.

Available in Mac OS X v10.4 and later.

6.30.10 kCGPDFContextAuthor as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that used to set up a PDF context.

Notes: The corresponding value is a string that represents the name of the person who created the document. This key is optional.

Available in Mac OS X v10.4 and later.

6.30.11 kCGPDFContextBleedBox as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that specify various PDF boxes.

Notes: The bleed box for the document or for a given page. This key is optional. If present, the value of this key must be a CGRectMBS object.

Available in Mac OS X v10.4 and later.

6.30.12 `kCGPDFContextCreator` as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that used to set up a PDF context.

Notes: The corresponding value is a string that represents the name of the application used to produce the document. This key is optional.

Available in Mac OS X v10.4 and later.

6.30.13 `kCGPDFContextCropBox` as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that specify various PDF boxes.

Notes: The crop box for the document or for a given page. This key is optional. If present, the value of this key must be a `CGRect` object.

Available in Mac OS X v10.4 and later.

6.30.14 `kCGPDFContextEncryptionKeyLength` as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that used to set up a PDF context.

Notes: The encryption key length in bits; see Table 3.18 "Entries common to all encryption dictionaries", PDF Reference: Adobe PDF version 1.5 (4th ed.) for more information. Optional; if present, the value of this key must be a number with value which is a multiple of 8 between 40 and 128, inclusive. If this key is absent or invalid, the encryption key length defaults to 40 bits.

Available in Mac OS X v10.5 and later.

6.30.15 `kCGPDFContextKeywords` as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that used to set up a PDF context.

Notes: The keywords for this document. This key is optional. If the value of this key is a string, the `/Keywords` entry will be the specified string. If the value of this key is a array, then it must be an array of variants with strings. The `/Keywords` entry will, in this case, be the concatenation of the specified strings separated by commas (","). In addition, an entry with the key `/AAPL:Keywords` is stored in the document information dictionary; its value is an array consisting of each of the specified strings. The value of this key must be in one of the above forms; otherwise, this key is ignored.

Available in Mac OS X v10.5 and later.

6.30.16 kCGPDFContextMediaBox as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that specify various PDF boxes.

Notes: The media box for the document or for a given page. This key is optional. If present, the value of this key must be a CGRectMBS object.

Available in Mac OS X v10.4 and later.

6.30.17 kCGPDFContextOutputIntent as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that used to set up a PDF context.

Notes: The output intent PDF/X. This key is optional. If present, the value of this key must be a dictionary. The dictionary is added to the /OutputIntents entry in the PDF file document catalog. The keys and values contained in the dictionary must match those specified in section 9.10.4 of the PDF 1.4 specification, ISO/DIS 15930-3 document published by ISO/TC 130, and Adobe Technical Note #5413.

Available in Mac OS X v10.4 and later.

6.30.18 kCGPDFContextOutputIntents as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that used to set up a PDF context.

Notes: Output intent dictionaries. This key is optional. If present, the value must be an array of one or more kCGPDFContextOutputIntent dictionaries. The array is added to the PDF document in the /OutputIntents entry in the PDF file's document catalog. Each dictionary in the array must be of form specified for the kCGPDFContextOutputIntent key, except that only the first dictionary in the array is required to contain the "S" key with a value of GTS_PDFX. If both the kCGPDFContextOutputIntent and kCGPDFContextOutputIntents keys are specified, the former is ignored.

Available in Mac OS X v10.4 and later.

6.30.19 kCGPDFContextOwnerPassword as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that used to set up a PDF context.

Notes: The owner password of the PDF document. If this key is specified, the document is encrypted using the value as the owner password; otherwise, the document will not be encrypted. The value of this key must be a string that can be represented in ASCII encoding. Only the first 32 bytes are used for the password. There is no default value for this key. If the value of this key cannot be represented in ASCII, the document is not created and the creation function returns nil.

Available in Mac OS X v10.4 and later.

6.30.20 `kCGPDFContextSubject` as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that used to set up a PDF context.

Notes: The subject of a document. Optional; if present, the value of this key must be a string. Available in Mac OS X v10.5 and later.

6.30.21 `kCGPDFContextTitle` as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that used to set up a PDF context.

Notes: The corresponding value is a string that represents the title of the document. This key is optional. Available in Mac OS X v10.4 and later.

6.30.22 `kCGPDFContextTrimBox` as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that specify various PDF boxes.

Notes: The trim box for the document or for a given page. This key is optional. If present, the value of this key must be a memoryblock that contains a `CGRect` (stored by value, not by reference). Available in Mac OS X v10.4 and later.

6.30.23 `kCGPDFContextUserPassword` as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Keys that used to set up a PDF context.

Notes: The user password of the PDF document. If the document is encrypted, then the value of this key will be the user password for the document. If not specified, the user password is the empty string. The value of this key must be a string that can be represented in ASCII encoding; only the first 32 bytes will be used for the password. If the value of this key cannot be represented in ASCII, the document is not created and the creation function returns nil.

Available in Mac OS X v10.4 and later.

6.30.24 kCGPDFXDestinationOutputProfile as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Output Intent Dictionary Keys.

Notes: An ICC profile stream defining the transformation from the PDF document's source colors to output device colorants. This key is required if the value of kCGPDFXOutputConditionIdentifier does not specify a standard production condition. It is optional otherwise. If present, the value of this key must be an ICC-based color space specified as a CGColorSpaceMBS object.

Available in Mac OS X v10.4 and later.

6.30.25 kCGPDFXInfo as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Output Intent Dictionary Keys.

Notes: A human-readable text string containing additional information or comments about the intended target device or production condition. This key is required if the value of kCGPDFXOutputConditionIdentifier does not specify a standard production condition. It is optional otherwise. If present, the value of this key must be a string.

Available in Mac OS X v10.4 and later.

6.30.26 kCGPDFXOutputCondition as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Output Intent Dictionary Keys.

Notes: A text string identifying the intended output device or production condition in a human-readable form. This key is optional. If present, the value of this key must be a string.

Available in Mac OS X v10.4 and later.

6.30.27 kCGPDFXOutputConditionIdentifier as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Output Intent Dictionary Keys.

Notes: A string identifying the intended output device or production condition in a human- or machine-readable form. This key is required. The value of this key must be a string. For best results, the string should be restricted to characters in the ASCII character set.

Available in Mac OS X v10.4 and later.

6.30.28 kCGPDFXOutputIntentSubtype as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Output Intent Dictionary Keys.

6.30.29 kCGPDFXRegistryName as string

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: One of the Output Intent Dictionary Keys.

Notes: The output intent subtype. This key is required. The value of this key must be a String object equal to "GTS_PDFX"; otherwise, the dictionary is ignored.

Available in Mac OS X v10.4 and later.

6.30.30 SetDestinationForRect(name as string, x as Double, y as Double, w as Double, h as Double)

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Sets a destination to jump to when a rectangle in the current PDF page is clicked.

Notes: name: A destination name.

x,y,w,h: A rectangle that specifies an area of the current page of a PDF graphics context. The rectangle is specified in default user space (not device space).

Available in Mac OS X v10.4 and later.

6.30.31 SetURLForRect(url as string, x as Double, y as Double, w as Double, h as Double)

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Sets the URL associated with a rectangle in a PDF graphics context.

Example:

```
Var c as CGPDFContextMBS // your pdf context
c.SetURLForRect("http://www.apple.com/", 100, 100, 100, 100)
```

Notes: url: A string that specifies the destination of the contents associated with the rectangle.

rect: A rectangle specified in default user space (not device space).

Available in Mac OS X v10.4 and later.

6.31 class CGPDFDictionaryListMBS

6.31.1 class CGPDFDictionaryListMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: A class for a pdf dictionary list.

6.31.2 Methods

6.31.3 Close

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: The destructor.

Notes: There is no need to call this method except you want to free all resources of this object now without waiting for Xojo to do it for you.

6.31.4 Key(index as Integer) as string

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Returns the key with the give index.

Notes: Returns "" on any error.

6.31.5 Value(index as Integer) as CGPDFObjectMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: The value with the given index.

Notes: Returns nil on any error.

6.31.6 Properties

6.31.7 Count as Integer

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Returns the number of key&value pairs.

Notes: Returns 0 on any error.

(Read only property)

6.31.8 Document as CGPDFDocumentMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The document this PDF Dictionary belongs to.

Notes: (Read only property)

6.32 class CGPDFDictionaryMBS

6.32.1 class CGPDFDictionaryMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: A class for a pdf dictionary.

6.32.2 Methods

6.32.3 ArrayValue(key as string, byref value as CGPDFArrayMBS) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given key is a pdf array, this function will return it.

Notes: Returns true on success and false on failure.

6.32.4 BooleanValue(key as string, byref value as boolean) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given key is a boolean value, this function will return it.

Notes: Returns true on success and false on failure.

6.32.5 Count as Integer

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: The number of keys in this dictionary.

6.32.6 DictionaryValue(key as string, byref value as CGPDFDictionaryMBS) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given key is a pdf dictionary, this function will return it.

Notes: Returns true on success and false on failure.

6.32.7 IntegerValue(key as string, byref value as Integer) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given key is an integer value, this function will return it.

Notes: Returns true on success and false on failure.

Is function will return double and integer values. Conversion is done automatically.

6.32.8 List as CGPDFDictionaryListMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Returns a list of all key&value pairs in the dictionary.

Notes: Returns nil on any error.

6.32.9 NameValue(key as string, byref value as string) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given key is a name string, this function will return it.

Notes: Returns true on success and false on failure.

6.32.10 ObjectValue(key as string, byref value as CGPDFObjectMBS) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given key is a pdf object, this function will return it.

Notes: Returns true on success and false on failure.

6.32.11 SingleValue(key as string, byref value as Double) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given key is a floating point value, this function will return it.

Notes: Returns true on success and false on failure.

Is function will return double and integer values. Conversion is done automatically.

6.32.12 StreamValue(key as string, byref value as CGPDFStreamMBS) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given key is a pdf stream, this function will return it.

Notes: Returns true on success and false on failure.

6.32.13 StringValue(key as string, byref value as CGPDFStringMBS) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object with the given key is a pdf string, this function will return it.

Notes: Returns true on success and false on failure.

6.32.14 Properties

6.32.15 Document as CGPDFDocumentMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The document this PDF Dictionary belongs to.

Notes: (Read only property)

6.32.16 Handle as Integer

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: The handle of the pdf dictionary.

Notes: (Read and Write property)

6.33 class CGPDFDocumentMBS

6.33.1 class CGPDFDocumentMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: A class for a core graphics pdf document.

Example:

```

Var f as FolderItem
Var input as CGPDFDocumentMBS
Var output as CGContextMBS
Var r,c as CGRectMBS

f=SpecialFolder.Desktop.Child("test.pdf")
input=f.OpenAsCGPDFDocumentMBS

r=CGMakeRectMBS(0,0,200,200) // 200 by 200 Pixel page

f=SpecialFolder.Desktop.Child("output.pdf")
output=f.NewCGPDFDocumentMBS(r,"SomeTitle","SomeAuthor","SomeCreator")

// Create a new page
output.BeginPage r

// get size of input page one
c=input.CropBox(1)

// clip to a part of the new page leaving a border
r=CGMakeRectMBS(20,20,160,160)
output.ClipToRect r

// draw old pdf on a different position
c.top=-100
c.left=-100

// do the drawing of page one of input at new new output
output.DrawCGPDFDocument(input,c,1)

// cleanup
output.EndPage
output.Flush

```

Notes: If the release property is true, the destructor of this class will release the pdfdocument reference.

Blog Entries

- [Several ways for picture to PDF in MBS Plugins](#)

- [MBS Xojo Plugins, version 17.3pr7](#)
- [MBS Real Studio Plugins, version 12.5pr1](#)
- [MBS Real Studio Plugins, version 11.2pr7](#)

Xojo Developer Magazine

- [8.5, pages 20 to 21: Printing PDFs, Easy PDF printing on Mac and Windows](#)
- [6.4, page 30: Creating PDF Files, How to create PDFs using the MBS Plugins by Christian Schmitz](#)

6.33.2 Methods

6.33.3 ArtBox(page as Integer) as CGRectMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Returns the ArtBox of this page.

Notes: Returns nil on any error.

This function is deprecated by Apple in favor of using the CGPDFPageMBS class.

6.33.4 BleedBox(page as Integer) as CGRectMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Returns the BleedBox of this page.

Notes: Returns nil on any error.

This function is deprecated by Apple in favor of using the CGPDFPageMBS class.

6.33.5 Catalog as CGPDFDictionaryMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Return the document catalog of 'document'.

6.33.6 Constructor(dataProvider as CGDataProviderMBS)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a Quartz PDF document using a data provider.

Notes: Distributing individual pages of a PDF document to separate threads is not supported. If you want to use threads, consider creating a separate document for each thread and operating on a block of pages per

thread.

See also:

- 6.33.7 Constructor(file as folderitem) 404
- 6.33.8 Constructor(Handle as Integer) 404
- 6.33.9 Constructor(url as string) 404

6.33.7 Constructor(file as folderitem)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a Quartz PDF document using data specified by a file.

Notes: Distributing individual pages of a PDF document to separate threads is not supported. If you want to use threads, consider creating a separate document for each thread and operating on a block of pages per thread.

See also:

- 6.33.6 Constructor(dataProvider as CGDataProviderMBS) 403
- 6.33.8 Constructor(Handle as Integer) 404
- 6.33.9 Constructor(url as string) 404

6.33.8 Constructor(Handle as Integer)

Plugin Version: 11.2, Platform: macOS, Targets: All.

Function: Creates a Quartz PDF document object based on the given handle value.

Notes: Handle must not be zero and should be a valid CGPDFDocumentRef casted to integer.

See also:

- 6.33.6 Constructor(dataProvider as CGDataProviderMBS) 403
- 6.33.7 Constructor(file as folderitem) 404
- 6.33.9 Constructor(url as string) 404

6.33.9 Constructor(url as string)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a Quartz PDF document using data specified by a URL.

Notes: Distributing individual pages of a PDF document to separate threads is not supported. If you want to use threads, consider creating a separate document for each thread and operating on a block of pages per thread.

See also:

6.33. CLASS CGPDFDOCUMENTMBS	405
• 6.33.6 Constructor(dataProvider as CGDataProviderMBS)	403
• 6.33.7 Constructor(file as folderitem)	404
• 6.33.8 Constructor(Handle as Integer)	404

6.33.10 CreateWithData(data as Memoryblock) as CGPDFDocumentMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates a Quartz PDF document using a data.

Notes: Distributing individual pages of a PDF document to separate threads is not supported. If you want to use threads, consider creating a separate document for each thread and operating on a block of pages per thread.

See also:

- 6.33.11 CreateWithData(data as string) as CGPDFDocumentMBS 405

6.33.11 CreateWithData(data as string) as CGPDFDocumentMBS

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Creates a Quartz PDF document using a data.

Notes: Distributing individual pages of a PDF document to separate threads is not supported. If you want to use threads, consider creating a separate document for each thread and operating on a block of pages per thread.

See also:

- 6.33.10 CreateWithData(data as Memoryblock) as CGPDFDocumentMBS 405

6.33.12 CreateWithFile(file as folderitem) as CGPDFDocumentMBS

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Creates a Quartz PDF document using data specified by a file.

Notes: Distributing individual pages of a PDF document to separate threads is not supported. If you want to use threads, consider creating a separate document for each thread and operating on a block of pages per thread.

6.33.13 CreateWithProvider(dataProvider as CGDataProviderMBS) as CGPDFDocumentMBS

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Creates a Quartz PDF document using a data provider.

Notes: Distributing individual pages of a PDF document to separate threads is not supported. If you want to use threads, consider creating a separate document for each thread and operating on a block of pages per thread.

6.33.14 CreateWithURL(url as string) as CGPDFDocumentMBS

Plugin Version: 12.4, Platform: macOS, Targets: All.

Function: Creates a Quartz PDF document using data specified by a URL.

Notes: Distributing individual pages of a PDF document to separate threads is not supported. If you want to use threads, consider creating a separate document for each thread and operating on a block of pages per thread.

6.33.15 CropBox(page as Integer) as CGRectMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Returns the CropBox of this page.

Notes: Returns nil on any error.

This function is deprecated by Apple in favor of using the CGPDFPageMBS class.

6.33.16 GetID as CGPDFArrayMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Gets the file identifier for a PDF document.

Notes: A PDF file identifier is defined in the PDF specification as an array of two strings, the first of which is a permanent identifier that doesn't change even when the file is updated. The second string changes each time the file is updated. For more information, see PDF Reference: Version 1.3 (Second Edition), Adobe Systems Incorporated.

Available in Mac OS X v10.4 and later.

6.33.17 GetInfo as CGPDFDictionaryMBS

Plugin Version: 8.6, Platform: macOS, Targets: All.

Function: Queries the PDF info dictionary.

Example:

```
Var f as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
Var p as cgpdfdocumentMBS=f.OpenAsCGPDFDocumentMBS

Var Co as CGPDFStringMBS
Var info as CGPDFDictionaryMBS = p.GetInfo
if info.StringValue("Subject",co) then
MsgBox co.Text // shows subject
end if
```

Notes: Returns nil on any error.

6.33.18 MediaBox(page as Integer) as CGRectMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Returns the MediaBox of this page.

Notes: Returns nil on any error.

This function is deprecated by Apple in favor of using the CGPDFPageMBS class.

6.33.19 Page(index as Integer) as CGPDFPageMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Returns a page of the document.

6.33.20 RotationAngle(page as Integer) as Integer

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Returns the rotation of a page in degrees.

Notes: Returns 0 on any error.

6.33.21 TrimBox(page as Integer) as CGRectMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Returns the TrimBox of this page.

Notes: Returns nil on any error.

This function is deprecated by Apple in favor of using the CGPDFPageMBS class.

6.33.22 UnlockWithPassword(name as string) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Return true if 'document' is unlocked; false otherwise.

Notes: A document is unlocked if it isn't encrypted, or if it is encrypted and a valid password was previously specified with CGPDFDocumentUnlockWithPassword.

6.33.23 Properties

6.33.24 AllowsCopying as Boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Return true if 'document' allows copying; false otherwise.

Notes: Typically, this function returns false only if the document is encrypted and the document's current password doesn't grant permission to perform copying.

(Read only property)

6.33.25 AllowsPrinting as Boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Return true if 'document' allows printing; false otherwise.

Notes: Typically, this function returns false only if the document is encrypted and the document's current password doesn't grant permission to perform printing.

(Read only property)

6.33.26 handle as Integer

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: The handle for this pdfdocument.

Notes: Handle is a CGPDFDocumentRef.

(Read and Write property)

6.33.27 IsEncrypted as Boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Return true if the PDF file associated with 'document' is encrypted; false otherwise.

Notes: If the PDF file is encrypted, then a password must be supplied before certain operations are enabled; different passwords may enable different operations.
(Read only property)

6.33.28 IsUnlocked as Boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Return true if 'document' is unlocked; false otherwise.

Notes: A document is unlocked if it isn't encrypted, or if it is encrypted and a valid password was previously specified with `CGPDFDocumentUnlockWithPassword`.
(Read only property)

6.33.29 MajorVersion as Integer

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Returns the major version of the pdf document.

Notes: (Read only property)

6.33.30 MinorVersion as Integer

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Returns the minor version of the pdf document.

Notes: (Read only property)

6.33.31 PageCount as Integer

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Counts the pages inside the PDF document.

Example:

```
Var f as FolderItem
```

```
Var d as CGPDFDocumentMBS
```

```
f=SpecialFolder.Desktop.Child("test.pdf")
```

```
d=f.OpenAsCGPDFDocumentMBS
```

```
MsgBox str(d.PageCount)
```

Notes: Returns 0 on any error.
(Read only property)

6.34 class CGPDFObjectMBS

6.34.1 class CGPDFObjectMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: A class for a pdf object.

6.34.2 Methods

6.34.3 ArrayValue(byref value as CGPDFArrayMBS) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object contains a pdf array, this function will return it.

Notes: Returns true on success and false on failure.

6.34.4 BooleanValue(byref value as boolean) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object contains a boolean, this function will return it.

Notes: Returns true on success and false on failure.

6.34.5 DictionaryValue(byref value as CGPDFDictionaryMBS) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object contains a pdf dictionary, this function will return it.

Notes: Returns true on success and false on failure.

6.34.6 IntegerValue(byref value as Integer) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object contains an integer, this function will return it.

Notes: Returns true on success and false on failure.

Is function will return double and integer values. Conversion is done automatically.

6.34.7 NameValue(byref value as string) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object contains a name string, this function will return it.

Notes: Returns true on success and false on failure.

6.34.8 SingleValue(byref value as Double) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object contains a floating point number, this function will return it.

Notes: Returns true on success and false on failure.

Is function will return double and integer values. Conversion is done automatically.

6.34.9 StreamValue(byref value as CGPDFStreamMBS) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object contains a pdf stream, this function will return it.

Notes: Returns true on success and false on failure.

6.34.10 StringValue(byref value as CGPDFStringMBS) as boolean

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: If the object contains a pdfstring, this function will return it.

Notes: Returns true on success and false on failure.

6.34.11 Type as Integer

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Returns the type of the object content.

Notes: Some useful constants:

```
kCGPDFObjectTypeNull = 1
kCGPDFObjectTypeBoolean = 2
kCGPDFObjectTypeInteger = 3
kCGPDFObjectTypeReal = 4
kCGPDFObjectTypeName = 5
```

```
kCGPDFObjectTypeString = 6  
kCGPDFObjectTypeArray = 7  
kCGPDFObjectTypeDictionary = 8  
kCGPDFObjectTypeStream = 9
```

6.34.12 Properties

6.34.13 Document as CGPDFDocumentMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The document this PDF Object belongs to.

Notes: (Read only property)

6.34.14 Handle as Integer

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: The handle of the pdf object.

Notes: (Read and Write property)

6.34.15 Constants

Constants

Constant	Value	Description
kCGPDFObjectTypeArray	7	One of the type constants for a PDF object type. Type for a PDF array.
kCGPDFObjectTypeBoolean	2	One of the type constants for a PDF object type. The type for a PDF Boolean.
kCGPDFObjectTypeDictionary	8	One of the type constants for a PDF object type. The type for a PDF dictionary.
kCGPDFObjectTypeInteger	3	One of the type constants for a PDF object type. The type for a PDF integer.
kCGPDFObjectTypeName	5	One of the type constants for a PDF object type. Type for a PDF name.
kCGPDFObjectTypeNull	1	One of the type constants for a PDF object type. The type for a PDF null.
kCGPDFObjectTypeReal	4	One of the type constants for a PDF object type. The type for a PDF real.
kCGPDFObjectTypeStream	9	One of the type constants for a PDF object type. The type for a PDF stream.
kCGPDFObjectTypeString	6	One of the type constants for a PDF object type. The type for a PDF string.

6.35 class CGPDFPageMBS

6.35.1 class CGPDFPageMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: A class for a pdf page.

6.35.2 Methods

6.35.3 ArtBox as CGRectMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Returns the ArtBox of this page.

Notes: Returns nil on any error.

This is the value of the corresponding entry in the page's dictionary.

6.35.4 BleedBox as CGRectMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Returns the BleedBox of this page.

Notes: Returns nil on any error.

This is the value of the corresponding entry in the page's dictionary.

6.35.5 CropBox as CGRectMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Returns the BleedBox of this page.

Notes: Returns nil on any error.

This is the value of the corresponding entry in the page's dictionary.

6.35.6 Dictionary as CGPDFDictionaryMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: The information dictionary of this page.

Notes: Returns nil on any error.

6.35.7 MediaBox as CGRectMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Returns the MediaBox of this page.

Notes: Returns nil on any error.

This is the value of the corresponding entry in the page's dictionary.

6.35.8 PageNumber as Integer

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: The page number of this page.

6.35.9 RotationAngle as Integer

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Returns the rotation of a page in degrees.

Notes: This is the value of the /Rotate entry in the page's dictionary.

6.35.10 TrimBox as CGRectMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: Returns the TrimBox of this page.

Notes: Returns nil on any error.

This is the value of the corresponding entry in the page's dictionary.

6.35.11 Properties

6.35.12 Document as CGPDFDocumentMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The document this PDF Page belongs to.

Notes: (Read only property)

6.35.13 Handle as Integer

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: The handle of the pdf page.

Notes: (Read and Write property)

6.35.14 Constants

Constants

Constant	Value	Description
kCGPDFArtBox	4	One of the box types for a PDF page. The page art box—a rectangle, expressed in default user space units, defining the extent of the page’s meaningful content (including potential white space) as intended by the page’s creator. Available in Mac OS X v10.3 and later.
kCGPDFBleedBox	2	One of the box types for a PDF page. The page bleed box—a rectangle, expressed in default user space units, that defines the region to which the contents of the page should be clipped when output in a production environment Available in Mac OS X v10.3 and later.
kCGPDFCropBox	1	One of the box types for a PDF page. The page crop box—a rectangle, expressed in default user space units, that defines the visible region of default user space. When the page is displayed or printed, its contents are to be clipped to this rectangle. Available in Mac OS X v10.3 and later.
kCGPDFMediaBox	0	One of the box types for a PDF page. The page media box—a rectangle, expressed in default user space units, that defines the boundaries of the physical medium on which the page is intended to be displayed or printed Available in Mac OS X v10.3 and later.
kCGPDFTrimBox	3	One of the box types for a PDF page. The page trim box—a rectangle, expressed in default user space units, that defines the intended dimensions of the finished page after trimming. Available in Mac OS X v10.3 and later.

6.36 class CGPDFStreamMBS

6.36.1 class CGPDFStreamMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: A class for a pdf stream.

6.36.2 Methods

6.36.3 Data(byref format as Integer) as string

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The data of this pdf stream.

Notes: Format is set to one of the following constants:

CGPDFDataFormatRaw	0
CGPDFDataFormatJPEGEncoded	1
CGPDFDataFormatJPEG2000	2

Returns nil on any error.

6.36.4 Dictionary as CGPDFDictionaryMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: The information dictionary for this stream.

Notes: Returns nil on any error.

6.36.5 Properties

6.36.6 Document as CGPDFDocumentMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The document this PDF Stream belongs to.

Notes: (Read only property)

6.36.7 Handle as Integer

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: The handle of the pdf stream.

Notes: (Read and Write property)

6.36.8 Constants

Constants

Constant	Value	Description
CGPDFDataFormatJPEG2000	2	One of the data format constants.
CGPDFDataFormatJPEGEncoded	1	One of the data format constants.
CGPDFDataFormatRaw	0	One of the data format constants.

6.37 class CGPDFStringMBS

6.37.1 class CGPDFStringMBS

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: A class for a pdf string.

6.37.2 Methods

6.37.3 Bytes as MemoryBlock

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: The bytes of this pdf string.

Notes: Returns nil on any error.

This memoryblock is not encoding safe!

6.37.4 Length as Integer

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The length of the text.

6.37.5 Text as string

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: The text of the pdf string as a string.

Notes: In contrast to the bytes in the memoryblock, this String is encoding save.

Returns nil on any error.

6.37.6 Properties

6.37.7 Document as CGPDFDocumentMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The document this PDF String belongs to.

Notes: (Read only property)

6.37.8 Handle as Integer

Plugin Version: 4.2, Platform: macOS, Targets: All.

Function: The handle of the pdf string.

Notes: (Read and Write property)

6.38 class CGPointMBS

6.38.1 class CGPointMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: A class for a core graphics point.

Blog Entries

- [MBS Xojo Plugins, version 24.5pr3](#)
- [Basic Forms in MBS Xojo SceneKit Plugin](#)
- [MBS Real Studio Plugins, version 12.0pr6](#)
- [MBS REALbasic plug-ins version 9.5](#)

6.38.2 Methods

6.38.3 ApplyAffineTransform(p as CGAffineTransformMBS) as CGPointMBS

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Transform 'point' by 't' and return the result: p' = p * t where p = [x y 1] .

6.38.4 Binary as MemoryBlock

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Returns the internal data of the object as a CGPoint for use on Toolbox calls.

Example:

```
Var c as CGPointMBS
Var m as MemoryBlock

c=CGMakePointMBS(10,20)

m=c.Binary

MsgBox str(m.Size) // 8
MsgBox str(m.doubleValue( 0)) // 10
MsgBox str(m.doubleValue( 4)) // 20
```

6.38.5 Constructor

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a new point object with the given values.

See also:

- 6.38.6 Constructor(p as Ptr) 423
- 6.38.7 Constructor(source as CGPointMBS) 423
- 6.38.8 Constructor(x as Double, y as Double) 423

6.38.6 Constructor(p as Ptr)

Plugin Version: 17.1, Platform: macOS, Targets: All.

Function: Creates a new instance using data at the pointer.

Notes: Make sure the pointer is valid and has the right data and size.

See also:

- 6.38.5 Constructor 423
- 6.38.7 Constructor(source as CGPointMBS) 423
- 6.38.8 Constructor(x as Double, y as Double) 423

6.38.7 Constructor(source as CGPointMBS)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a new point object with the given values.

See also:

- 6.38.5 Constructor 423
- 6.38.6 Constructor(p as Ptr) 423
- 6.38.8 Constructor(x as Double, y as Double) 423

6.38.8 Constructor(x as Double, y as Double)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a new point object with the given values.

See also:

- 6.38.5 Constructor 423

- 6.38.6 Constructor(p as Ptr) 423
- 6.38.7 Constructor(source as CGPointMBS) 423

6.38.9 Equal(p as CGPointMBS) as boolean

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Returns true if both points are equal.

6.38.10 Make(x as Double, y as Double) as CGPointMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns a new CGPointMBS object.

6.38.11 multiply(factor as double) as CGPointMBS

Plugin Version: 24.5, Platform: macOS, Targets: All.

Function: Multiplies the point with this factors.

See also:

- 6.38.12 multiply(x as double, y as double) as CGPointMBS 424

6.38.12 multiply(x as double, y as double) as CGPointMBS

Plugin Version: 24.5, Platform: macOS, Targets: All.

Function: Multiplies the point with this factors.

See also:

- 6.38.11 multiply(factor as double) as CGPointMBS 424

6.38.13 Zero as CGPointMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns the zero point.

Notes: The "zero" point – equivalent to CGMakePointMBS(0, 0).

6.38.14 Properties

6.38.15 x as Double

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: The X property of the point.

Notes: (Read and Write property)

6.38.16 y as Double

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: The Y property of the point.

Notes: (Read and Write property)

6.39 class CGPSConverterMBS

6.39.1 class CGPSConverterMBS

Plugin Version: 4.3, Platform: macOS, Targets: Desktop, Console & Web.

Deprecated: This item is deprecated and should no longer be used. **Function:** The class in CoreGraphics to convert Postscript code to PDF data.

Notes: Requires Mac OS X 10.3.

Won't work in macOS Sonoma since Apple removed the feature from the OS.

Blog Entries

- [MBS Xojo Plugins, version 23.3pr2](#)
- [EPS support deprecated in macOS Sonoma](#)
- [MBS Real Studio Plugins, version 12.1pr10](#)

6.39.2 Methods

6.39.3 Abort as boolean

Plugin Version: 4.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Tell the converter to abort conversion at the next possible opportunity.

Notes: Returns false on any error and true on success.

6.39.4 Constructor(options as Dictionary = nil)

Plugin Version: 9.5, Platform: macOS, Targets: Desktop, Console & Web.

Function: Create a CGPSConverter.

Notes: Currently you should pass nil for options. (Mac OS X 10.3)

On failure the handle property is zero.

6.39.5 Convert(provider as CGDataProviderMBS, consumer as CGDataConsumerMBS, options as Dictionary = nil) as boolean

Plugin Version: 4.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Converts PostScript data to PDF data.

Notes: The PostScript data is supplied by provider; the resulting PDF is written to consumer. Returns true if the conversion succeeded; false otherwise.

6.39.6 IsConverting as boolean

Plugin Version: 4.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns true if the converter is currently converting data.

Notes: False on any error.

6.39.7 Properties

6.39.8 Handle as Integer

Plugin Version: 4.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The handle to the CGPSConverter object.

Notes: Datatype is CGPSConverterRef.

(Read and Write property)

6.39.9 Events

6.39.10 BeginDocument

Plugin Version: 4.3, Platform: macOS, Targets: .

Function: Is called at the beginning of the conversion of the PostScript document.

6.39.11 BeginPage(PageNumber as Integer, PageInfo as Dictionary)

Plugin Version: 4.3, Platform: macOS, Targets: .

Function: Called at the start of the conversion of each page in the PostScript document.

6.39.12 EndDocument(success as boolean)

Plugin Version: 4.3, Platform: macOS, Targets: .

Function: Called at the end of conversion of the PostScript document.

6.39.13 EndPage(PageNumber as Integer, PageInfo as Dictionary)

Plugin Version: 4.3, Platform: macOS, Targets: .

Function: Called at the end of the conversion of each page in the PostScript document.

6.39.14 Finished

Plugin Version: 4.3, Platform: macOS, Targets: .

Function: Called when the converter is no longer needed.

6.39.15 Message(message as string)

Plugin Version: 9.5, Platform: macOS, Targets: .

Function: Called to pass any messages that might result during the conversion.

6.39.16 Progress

Plugin Version: 4.3, Platform: macOS, Targets: .

Function: Called periodically during the conversion to indicate that conversion is proceeding.

6.40 class CGRectMBS

6.40.1 class CGRectMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: A class for a core graphics rectangle.

Blog Entries

- [Segment documents in images for Xojo](#)
- [MBS Xojo Plugins, version 24.5pr3](#)
- [Several ways for picture to PDF in MBS Plugins](#)
- [Automated video editing with Xojo](#)
- [MBS Real Studio Plugins, version 12.0pr6](#)
- [MBS REALbasic plug-ins version 9.5](#)

Xojo Developer Magazine

- [6.4, page 30: Creating PDF Files, How to create PDFs using the MBS Plugins by Christian Schmitz](#)

6.40.2 Methods

6.40.3 ApplyAffineTransform(a as CGAffineTransformMBS) as CGRectMBS

Plugin Version: 9.3, Platform: macOS, Targets: All.

Function: Applies an affine transform to the rectangle and returns the result.

Example:

```
Var af as CGAffineTransformMBS = CGAffineTransformMBS.Identity
```

```
MsgBox str(af.A)+" "+str(af.b)+" "+str(af.c)+" "+str(af.d)+" "+str(af.tx)+" "+str(af.ty)
```

```
af = af.Scale( 1, -1 )
```

```
MsgBox str(af.A)+" "+str(af.b)+" "+str(af.c)+" "+str(af.d)+" "+str(af.tx)+" "+str(af.ty)
```

```
af = af.Translate( 0, 100 )
```

```
MsgBox str(af.A)+" "+str(af.b)+" "+str(af.c)+" "+str(af.d)+" "+str(af.tx)+" "+str(af.ty)
```

```
Var r1 as CGRectMBS
```

```
Var r2 as CGRectMBS
```

```

r1=CGMakeRectMBS(100,100,100,100)

r2=r1.ApplyAffineTransform(af)

MsgBox "("+str(r1.Left)+", "+str(r1.top)+", "+str(r1.width)+", "+str(r1.height)+") => "("+str(r2.Left)+",
"+str(r2.top)+", "+str(r2.width)+", "+str(r2.height)+")"

```

Notes: Returns nil on any error.

6.40.4 Binary as MemoryBlock

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Returns the internal data of the object as a CGRect for use on Toolbox calls.

Example:

```

Var c as CGRectMBS
Var m as MemoryBlock

c=CGMakeRectMBS(10,20,30,40)

m=c.Binary

MsgBox str(m.Size) // 16
MsgBox str(m.doubleValue( 0)) // 10
MsgBox str(m.doubleValue( 4)) // 20
MsgBox str(m.doubleValue( 8)) // 30
MsgBox str(m.doubleValue(12)) // 40

```

6.40.5 Constructor

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The constructor to create a zero rectangle.

See also:

- 6.40.6 Constructor(p as Ptr) 431
- 6.40.7 Constructor(source as CGRectMBS) 431
- 6.40.8 Constructor(x as Double, y as Double, width as Double, height as Double) 431

6.40. CLASS CGRECTMBS 431

6.40.6 Constructor(p as Ptr)

Plugin Version: 17.1, Platform: macOS, Targets: All.

Function: Creates a new instance using data at the pointer.

Notes: Make sure the pointer is valid and has the right data and size.

See also:

- 6.40.5 Constructor 430
- 6.40.7 Constructor(source as CCGRectMBS) 431
- 6.40.8 Constructor(x as Double, y as Double, width as Double, height as Double) 431

6.40.7 Constructor(source as CCGRectMBS)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a new CCGRectMBS object by copying the values.

See also:

- 6.40.5 Constructor 430
- 6.40.6 Constructor(p as Ptr) 431
- 6.40.8 Constructor(x as Double, y as Double, width as Double, height as Double) 431

6.40.8 Constructor(x as Double, y as Double, width as Double, height as Double)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a new CCGRectMBS object with the given values.

See also:

- 6.40.5 Constructor 430
- 6.40.6 Constructor(p as Ptr) 431
- 6.40.7 Constructor(source as CCGRectMBS) 431

6.40.9 ContainsPoint(r as CCGPointMBS) as boolean

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Return true if 'point' is contained in the rect, false otherwise.

6.40.10 ContainsRect(r as CGRectMBS) as boolean

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Return true if 'rect2' is contained in 'rect1', false otherwise.

Notes: 'rect2' is contained in 'rect1' if the union of 'rect1' and 'rect2' is equal to 'rect1'.
(rect1 is the current rect and rect2 the rect you pass to this function)

6.40.11 Divide(byref slice as CGRectMBS, byref remainder as CGRectMBS, amount as Double, edge as Integer)

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Divides the rectangle.

Notes: Make two new rectangles, 'slice' and 'remainder', by dividing the rect with a line that's parallel to one of its sides, specified by 'edge' – either 'CGRectMinXEdge', 'CGRectMinYEdge', 'CGRectMaxXEdge', or 'CGRectMaxYEdge'. The size of 'slice' is determined by 'amount', which measures the distance from the specified edge.

6.40.12 Equal(r as CGRectMBS) as boolean

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Returns true if both rectangles are equal.

6.40.13 Infinite as CGRectMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: A rectangle that has infinite extent.

Example:

```
Var r as CGRectMBS = CGRectMBS.Infinite
MsgBox str(r.Left)+" "+str(R.Top)+" "+str(R.Width)+" "+str(r.Height)
```

Notes: Available in Mac OS X v10.4 and later.

6.40.14 Inset(dx as Double, dy as Double) as CGRectMBS

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Inset the rect by '(dx, dy)' – i.e., offset its origin by '(dx, dy)', and decrease its size by '(2*dx, 2*dy)'

6.40.15 Integral as CGRectMBS

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Expand the rect to the smallest rect containing it with integral origin and size.

6.40.16 Intersection(r as CGRectMBS) as CGRectMBS

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Return the intersection of both rectangles.

Notes: This may return a null rect.

6.40.17 IntersectsRect(r as CGRectMBS) as boolean

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Return true if 'rect1' intersects 'rect2', false otherwise.

Notes: 'rect1' intersect 'rect2' if the intersection of 'rect1' and 'rect2' is not the null rect.

6.40.18 IsEmpty as boolean

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Return true if the rect is empty – i.e., if it has zero width or height.

Notes: A null rect is defined to be empty.

6.40.19 IsInfinite as boolean

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns whether a rectangle is infinite.

Notes: Returns true if the specified rectangle is infinite, false otherwise.

An infinite rectangle is one that has no defined bounds. Infinite rectangles can be created as output from a tiling filter. For example, the Core Image framework perspective tile filter creates an image whose extent is

described by an infinite rectangle.

6.40.20 IsNull as boolean

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Return true if the rect is null – e.g., the result of intersecting two disjoint rectangles is a null rect.

6.40.21 Make(x as Double, y as Double, width as Double, height as Double) as CGRectMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns a new CGRectMBS object with the given values.

Example:

```
Var r as CGRectMBS = CGRectMBS.Make(1,2,3,4)
MsgBox str(r.Left)+" "+str(R.Top)+" "+str(R.Width)+" "+str(r.Height)
```

6.40.22 MaxX as Double

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Return the rightmost x-value of the rect.

Example:

```
Var r as CGRectMBS = CGMakeRectMBS(10, 20, 30, 40)
MsgBox str(r.MaxX) // 40
```

6.40.23 MaxY as Double

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Return the topmost y-value of the rect.

Example:

```
Var r as CGRectMBS = CGMakeRectMBS(10, 20, 30, 40)
MsgBox str(r.MaxY) // 60
```

6.40.24 MidX as Double

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Return the midpoint x-value of the rect.

Example:

```
Var r as CGRectMBS = CGMakeRectMBS(10, 20, 30, 40)
MsgBox str(r.MidX) // 25
```

6.40.25 MidY as Double

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Return the midpoint y-value of the rect.

Example:

```
Var r as CGRectMBS = CGMakeRectMBS(10, 20, 30, 40)
MsgBox str(r.MidY) // 40
```

6.40.26 MinX as Double

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Return the leftmost x-value of the rect.

Example:

```
Var r as CGRectMBS = CGMakeRectMBS(10, 20, 30, 40)
MsgBox str(r.MinX) // 10
```

6.40.27 MinY as Double

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Return the bottommost y-value of the rect.

Example:

```
Var r as CGRectMBS = CGMakeRectMBS(10, 20, 30, 40)
MsgBox str(r.MinY) // 20
```

6.40.28 multiply(factor as double) as CGRectMBS

Plugin Version: 24.5, Platform: macOS, Targets: All.

Function: Multiplies the rect with this factors.

See also:

- 6.40.29 multiply(x as double, y as double) as CGRectMBS 436

6.40.29 multiply(x as double, y as double) as CGRectMBS

Plugin Version: 24.5, Platform: macOS, Targets: All.

Function: Multiplies the rect with this factors.

See also:

- 6.40.28 multiply(factor as double) as CGRectMBS 435

6.40.30 Null as CGRectMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns the null rect.

Example:

```
Var r as CGRectMBS = CGRectMBS.Null
MsgBox str(r.Left)+" "+str(R.Top)+" "+str(R.Width)+" "+str(r.Height)
```

Notes: The "empty" rect. This is the rectangle returned when, for example, we intersect two disjoint rectangles.

Note that the null rect is not the same as the zero rect.

6.40.31 Offset(dx as Double, dy as Double) as CGRectMBS

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Offset the rect by '(dx, dy)'.

6.40.32 Standardize as CGRectMBS

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Standardize the rect – i.e., convert it to an equivalent rect which has positive width and height.

Example:

```
Var r as CGRectMBS = CGRectMBS.Make(5, 6, -3, -2)
MsgBox str(r.Left)+" "+str(R.Top)+" "+str(R.Width)+" "+str(r.Height)
```

```
r = r.Standardize
MsgBox str(r.Left)+" "+str(R.Top)+" "+str(R.Width)+" "+str(r.Height)
```

6.40.33 Union(r as CRectMBS) as CRectMBS

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Return the union of both rectangles.

Example:

```
Var r as CRectMBS = CRectMBS.Infinite
MsgBox str(r.Left)+" "+str(R.Top)+" "+str(R.Width)+" "+str(r.Height)
```

6.40.34 Zero as CRectMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: A rectangle constant with location (0,0), and width and height of 0.

Example:

```
Var r as CRectMBS = CRectMBS.Zero
MsgBox str(r.Left)+" "+str(R.Top)+" "+str(R.Width)+" "+str(r.Height)
```

Notes: The zero rectangle is equivalent to CRectMBS.Make(0,0,0,0).

6.40.35 Properties

6.40.36 height as Double

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: The height property of the rectangle.

Example:

```
Var r as CRectMBS = CGMakeRectMBS(10, 20, 30, 40)
MsgBox str(r.Left)+" "+str(R.Top)+" "+str(R.Width)+" "+str(r.Height)
```

Notes: (Read and Write property)

6.40.37 left as Double

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: The left property of the rectangle.

Notes: (Read and Write property)

6.40.38 Origin as CGPointMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: The origin of the rectangle.

Example:

```
Var r as CGRectMBS = CGRectMBS.Make(5, 6, 3, 2)
Var s as CGPointMBS = r.Origin
MsgBox str(s.x)+" x "+str(s.y)
```

Notes: (Read and Write property)

6.40.39 Size as CGSizeMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: The size of the rectangle.

Example:

```
Var r as CGRectMBS = CGRectMBS.Make(5, 6, 3, 2)
Var s as CGSizeMBS = r.Size
MsgBox str(s.Width)+" x "+str(s.Height)
```

Notes: (Read and Write property)

6.40.40 top as Double

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: The top property of the rectangle.

Example:

```
Var r as CGRectMBS = CGMakeRectMBS(10, 20, 30, 40)
MsgBox str(r.Left)+" "+str(R.Top)+" "+str(R.Width)+" "+str(r.Height)
```

Notes: (Read and Write property)

6.40.41 width as Double

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: The width property of the rectangle.

Example:

```
Var r as CGRectMBS = CGMakeRectMBS(10, 20, 30, 40)
MsgBox str(r.Left)+" "+str(R.Top)+" "+str(R.Width)+" "+str(r.Height)
```

Notes: (Read and Write property)

6.41 class CGSConnectionMBS

6.41.1 class CGSConnectionMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: A class for a connection to the CoreGraphics System.

Example:

```
Var c as new CGSConnectionMBS
Var list as CGSWindowListMBS = c.GetWindowList
```

```
Var names() as string
Var u as Integer = List.Count-1
for i as Integer = 0 to u
  Var w as CGSWindowMBS = List.Item(i)
  names.append w.Title
next
```

```
MsgBox str(List.Count)+" windows: "+Join(names, ", ")
```

Notes: Requires Mac OS X 10.4.

All functions used by the CGS classes are private APIs, so Apple does not guarantee that they work in future Mac OS X versions.

Blog Entries

- [MBS REALbasic Plugins, version 11.0fc1](#)
- [MonkeyBread Software Releases the MBS Plugins 8.1](#)

6.41.2 Methods

6.41.3 CGSWindow(w as window) as CGSWindowMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop only.

Function: Returns a CGSWindow reference to a normal RB window.

Notes: Can be used for the CGSTransitionRequestMBS.Win property.

6.41.4 CGSWindowbyHandle(windowhandle as Integer) as CGSWindowMBS

Plugin Version: 8.4, Platform: macOS, Targets: Desktop only.

Function: Returns a CGSWindow reference to any window where you can have a handle.

Notes: WindowHandle must be of C type WindowRef. Not a CGS Window Handle.

Can be used for the handle from an OverlayWindowMBS.

6.41.5 FlushAllWindows

Plugin Version: 8.0, Platform: macOS, Targets: Desktop only.

Function: Flushes all windows.

Notes: Walks over the window list (of the current process) and does a flush on each window.

Lasterror is set.

6.41.6 FlushAllWindowsForAllOtherProcesses

Plugin Version: 8.0, Platform: macOS, Targets: Desktop only.

Function: Flushes all windows.

Notes: Walks over the list of processes asking each for its window list and doing a flush on each window. The own process is ignored.

Lasterror is set.

6.41.7 FlushAllWindowsForAllProcesses

Plugin Version: 8.0, Platform: macOS, Targets: Desktop only.

Function: Flushes all windows.

Notes: Walks over the list of processes asking each for its window list and doing a flush on each window.

Lasterror is set.

6.41.8 GetOnScreenWindowList as CGSWindowListMBS

Plugin Version: 8.1, Platform: macOS, Targets: Desktop only.

Function: Returns the list of windows on screen for the current process.

Notes: Returns nil on any error.

6.41.9 GetOnScreenWindowListForProcess(PID as Integer) as CGSWindowListMBS

Plugin Version: 8.1, Platform: macOS, Targets: Desktop only.

Function: Returns the list of windows on screen for the process with the given Process ID.

Notes: Returns nil on any error.

6.41.10 GetWindowList as CGSWindowListMBS

Plugin Version: 8.1, Platform: macOS, Targets: Desktop only.

Function: Returns the list of windows for the current process.

Notes: Returns nil on any error.

6.41.11 GetWindowListForProcess(PID as Integer) as CGSWindowListMBS

Plugin Version: 8.1, Platform: macOS, Targets: Desktop only.

Function: Returns the list of windows for the process with the given Process ID.

Notes: Returns nil on any error.

6.41.12 NewTransition(request as CGSTransitionRequestMBS) as CGSTransitionMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a new transition.

Notes: Lasterror is set.

Request must be a valid object reference.

Transitions seem not to work in Carbon PEF builds. MachO works.

6.41.13 RunTransition(request as CGSTransitionRequestMBS, duration as single)

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: A method to run a transition for the given request and given time.

Example:

```
Var r as CGSTransitionRequestMBS
Var co as CGSConnectionMBS // global property
Var cw as CGSWindowMBS
```

```
co=new CGSConnectionMBS
```

```
cw=co.CGSWindow(window1)
```

```

r=new CGSTransitionRequestMBS
r.TransitionType=r.CGSFlip
r.TransitionOption=r.CGSLeft
r.HasBackGround=false
r.HasBackColor=false
r.Win=cw
co.RunTransition(r,2)

```

Notes: Lasterror is set.

Transitions seem not to work in Carbon PEF builds. MachO works.

6.41.14 SetWorkspaceWithTransition(workspace as CGSWorkspaceMBS, transition as Integer, type as Integer, time as single)

Plugin Version: 8.2, Platform: macOS, Targets: Desktop only.

Function: Performs a transition with a workspace specified by the workspace object.

Notes: The values for the parameters are not documented on the web except that the transition and type property may use the same transition constants as for the other transitions.

Lasterror is set.

See also:

- 6.41.15 SetWorkspaceWithTransition(workspace as Integer, transition as Integer, type as Integer, time as single) 443

6.41.15 SetWorkspaceWithTransition(workspace as Integer, transition as Integer, type as Integer, time as single)

Plugin Version: 8.2, Platform: macOS, Targets: Desktop only.

Function: Performs a transition with a workspace specified by the workspace number.

Example:

```

Var c as new CGSConnectionMBS
Var t as new CGSTransitionRequestMBS

// from space 1 to 2
c.SetWorkspaceWithTransition(2,t.CGSFade, t.CGSInOut, 5)

// but transition and time are ignored on testing Mac.

MsgBox str(c.Lasterror)

```

Notes: The values for the parameters are not documented on the web except that the transition and type property may use the same transition constants as for the other transitions.

Lasterror is set.

See also:

- 6.41.14 SetWorkspaceWithTransition(workspace as CGSWorkspaceMBS, transition as Integer, type as Integer, time as single) 443

6.41.16 Properties

6.41.17 Handle as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal handle for this connection.

Notes: (Read and Write property)

6.41.18 Lasterror as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Last error code reported.

Notes: The list of CoreGraphics error (from Mac OS X 10.4)

kCGErrorSuccess	= 0	Success
kCGErrorFailure	= 1000	Failure
kCGErrorIllegalArgument	= 1001	Illegal Argument
kCGErrorInvalidConnection	= 1002	Invalid Connection
kCGErrorInvalidContext	= 1003	Invalid Context
kCGErrorCannotComplete	= 1004	Cannot complete
kCGErrorNameTooLong	= 1005	Name is too long
kCGErrorNotImplemented	= 1006	Not implemented
kCGErrorRangeCheck	= 1007	Out of bounds
kCGErrorTypeCheck	= 1008	Type error
kCGErrorNoCurrentPoint	= 1009	No current point
kCGErrorInvalidOperation	= 1010	Invalid Operation
kCGErrorNoneAvailable	= 1011	internal errors have taken 1012 1013 and 1014
kCGErrorApplicationRequiresNewerSystem	= 1015	the application being launched says in it's bundle info that it requires a newer version of the system than is currently running.
kCGErrorApplicationNotPermittedToExecute	= 1016	Macintosh Manager is active and this application is not permitted to run
kCGErrorApplicationIncorrectExecutableFormatFound	= 1023	the application being launched does not have any executable code for the current system.
kCGErrorApplicationIsLaunching	= 1024	The application is in the process of launching but hasn't checked in yet. Any launch data provided will be given to the application when it does check in.
kCGErrorApplicationAlreadyRunning	= 1025	The application being launched was already running (and had already checked in) and so any launch data provided can not be delivered to in by CPS
kCGErrorApplicationCanOnlyBeRunInOneSessionAtATime	= 1026	The application being launched is incompatible with multiple user sessions and is already running in another session by another user.
kCGErrorClassicApplicationsMustBeLaunchedByClassic	= 1027	To avoid deadlock Classic can't launch another Classic application by going thru CPS. This error gets returned in that case and it signals TruBlueEnvironment that it must handle this launch on its own.
kCGErrorForkFailed	= 1028	CPS was unable to fork a new process in order to launch an application.

(Read and Write property)

6.41.19 Workspace as CGSWorkspaceMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The workspace this connection is pointing to.

Notes: (Read and Write computed property)

6.42 class CGScreenRefreshEventMBS

6.42.1 class CGScreenRefreshEventMBS

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: The class to receive events for screen updates.

6.42.2 Properties

6.42.3 Initialized as Boolean

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: Whether object is initialized correctly.

Notes: Should be true on OS X if initializing was done.

(Read only property)

6.42.4 Events

6.42.5 ScreenRefresh(rectCount as Integer, rects() as CGRectMBS)

Plugin Version: 16.1, Platform: macOS, Targets: .

Function: The event to inform you about a screen refresh.

Notes: When an area of the display is modified or refreshed, this event function will be invoked with a count of the number of rectangles in the refreshed areas, and a list of the refreshed rectangles. The rectangles are in global coordinates.

Quartz invokes this event when operations such as drawing, window movement, scrolling, or display reconfiguration occur on local displays.

Note that a single rectangle may occupy multiple displays, either by overlapping the displays or by residing on coincident displays when mirroring is active. You can use the function `CGDisplayMBS.GetDisplaysWithRect` to determine the displays a rectangle occupies.

6.43 class CGScreenUpdateMoveEventMBS

6.43.1 class CGScreenUpdateMoveEventMBS

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: The class to get event when something on screen changes.

6.43.2 Properties

6.43.3 Initialized as Boolean

Plugin Version: 16.1, Platform: macOS, Targets: All.

Function: Whether object is initialized correctly.

Notes: This should be true on Mac after calling constructor.

(Read only property)

6.43.4 Events

6.43.5 ScreenMove(deltaX as Integer, deltaY as Integer, rectCount as Integer, rects() as CGRectMBS)

Plugin Version: 16.1, Platform: macOS, Targets: .

Function: The event invoked when an area of the display is moved.

Notes: When an area of the display is moved, your callback function will be invoked with a count of the number of rectangles in the moved area, and a list of the moved rectangles. The rectangles are in global coordinates, and describe the area prior to the move operation.

A single rectangle may occupy multiple displays, either by overlapping the displays or by residing on coincident displays when mirroring is active. Use `CGDisplayMBS.GetDisplaysWithRect` to determine the displays a rectangle occupies.

DeltaX/DeltaY: The distance a region on the screen moves in pixel units.

The fields `deltaX` and `deltaY` describe the direction of movement. Positive values of `deltaX` indicate movement to the right; negative values indicate movement to the left. Positive values of `deltaY` indicate movement downward; negative values indicate movement upward.

6.44 class CGShadingMBS

6.44.1 class CGShadingMBS

Plugin Version: 6.2, Platform: macOS, Targets: All.

Function: To shade means to fill using a smooth transition between colors across an area. Quartz shadings simplify several common shading operations. Quartz shadings currently support radial and axial gradient fills.

Notes: To paint with a Quartz shading, you call `CGContextMBS.DrawShading`. This function fills the current clipping path using the specified color gradient, calling your parametric function repeatedly as it draws.

Blog Entries

- [MBS Real Studio Plugins, version 13.1fc1](#)

6.44.2 Properties

6.44.3 Handle as Integer

Plugin Version: 6.2, Platform: macOS, Targets: All.

Function: Handle to the `CGShadingRef`.

Notes: (Read and Write property)

6.45 class CGSizeMBS

6.45.1 class CGSizeMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: A class for a core graphics size.

Blog Entries

- [MBS Xojo Plugins, version 24.5pr3](#)
- [MBS Real Studio Plugins, version 12.0pr6](#)
- [MBS REALbasic plug-ins version 9.5](#)

Xojo Developer Magazine

- [12.2, page 45: Moving to AVFoundation, Leaving QuickTime behind by Christian Schmitz](#)

6.45.2 Methods

6.45.3 ApplyAffineTransform(p as CGAffineTransformMBS) as CGSizeMBS

Plugin Version: 3.4, Platform: macOS, Targets: All.

Function: Transform 'size' by 't' and return the result: $s' = s * t$ where $s = [\text{width height } 0]$.

6.45.4 Binary as MemoryBlock

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Returns the internal data of the object as a CGSize for use on Toolbox calls.

Example:

```
Var c as CGSizeMBS
Var m as MemoryBlock
```

```
c=CGMakeSizeMBS(10,20)
```

```
m=c.Binary
```

```
MsgBox str(m.Size) // 8
MsgBox str(m.doubleValue( 0)) // 10
MsgBox str(m.doubleValue( 4)) // 20
```

6.45.5 Constructor

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates an zero size object.

See also:

- 6.45.6 Constructor(p as Ptr) 450
- 6.45.7 Constructor(source as CGSizeMBS) 450
- 6.45.8 Constructor(width as Double, height as Double) 450

6.45.6 Constructor(p as Ptr)

Plugin Version: 17.1, Platform: macOS, Targets: All.

Function: Creates a new instance using data at the pointer.

Notes: Make sure the pointer is valid and has the right data and size.

See also:

- 6.45.5 Constructor 450
- 6.45.7 Constructor(source as CGSizeMBS) 450
- 6.45.8 Constructor(width as Double, height as Double) 450

6.45.7 Constructor(source as CGSizeMBS)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: The constructor which copies the values from the other size object.

See also:

- 6.45.5 Constructor 450
- 6.45.6 Constructor(p as Ptr) 450
- 6.45.8 Constructor(width as Double, height as Double) 450

6.45.8 Constructor(width as Double, height as Double)

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Creates a size object with the given values.

See also:

- 6.45.5 Constructor 450

6.45. CLASS CGSIZEMBS	451
• 6.45.6 Constructor(p as Ptr)	450
• 6.45.7 Constructor(source as CGSizeMBS)	450

6.45.9 Equal(p as CGSizeMBS) as boolean

Plugin Version: 3.2, Platform: macOS, Targets: All.

Function: Returns true if two CGSize objects are equal.

6.45.10 Make(width as Double, height as Double) as CGSizeMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns a new CGSizeMBS object with the given values.

6.45.11 multiply(factor as double) as CGSizeMBS

Plugin Version: 24.5, Platform: macOS, Targets: All.

Function: Multiplies the size with this factors.

See also:

- 6.45.12 multiply(x as double, y as double) as CGSizeMBS 451

6.45.12 multiply(x as double, y as double) as CGSizeMBS

Plugin Version: 24.5, Platform: macOS, Targets: All.

Function: Multiplies the size with this factors.

See also:

- 6.45.11 multiply(factor as double) as CGSizeMBS 451

6.45.13 Zero as CGSizeMBS

Plugin Version: 9.5, Platform: macOS, Targets: All.

Function: Returns the zero size.

Notes: The "zero" size – equivalent to CGSizeMBS.Make(0, 0).

6.45.14 Properties

6.45.15 height as Double

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: The height property of the point.

Notes: (Read and Write property)

6.45.16 width as Double

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: The width property of the point.

Notes: (Read and Write property)

6.46 class CGSTransitionMBS

6.46.1 class CGSTransitionMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Deprecated: This item is deprecated and should no longer be used. **Function:** A class for a transition to the CoreGraphics System.

Notes: Requires Mac OS X 10.4.

All functions used by the CGS classes are private APIs, so Apple does not guarantee that they work in future Mac OS X versions.

Blog Entries

- [MBS Xojo Plugins, version 23.3pr7](#)

6.46.2 Methods

6.46.3 Invoke(duration as single)

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Performs the transition.

Example:

```
// cube transition for the whole screen
Var r as CGSTransitionRequestMBS
Var co as CGSConnectionMBS // global property
Var ct as CGSTransitionMBS // global property

co=new CGSConnectionMBS

r=new CGSTransitionRequestMBS
r.TransitionType=r.CGSCube
r.TransitionOption=r.CGSLight
r.HasBackGround=true
r.HasBackColor=false

ct=co.NewTransition(r)
if ct<>Nil then
window1.refresh // draw new screen
ct.invoke 5
ct.wait 5
ct.release
else
MsgBox "Error creating the transition."
end if
```

Notes: duration is the time in seconds the duration will need to complete.
Lasterror is set.

6.46.4 Release

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Releases the transaction.

Example:

```
Var r as CGSTransitionRequestMBS
Var co as CGSConnectionMBS // global property
Var ct as CGSTransitionMBS // global property
Var cw as CGSWindowMBS
```

```
co=new CGSConnectionMBS
```

```
cw=co.CGSWindow(window1)
```

```
r=new CGSTransitionRequestMBS
r.TransitionType=r.CGSCube
r.TransitionOption=r.CGSLeft
r.HasBackGround=false
r.HasBackColor=false
r.Win=cw
```

```
ct=co.NewTransition(r)
if ct<>Nil then
window1.refresh // draw new screen
ct.invoke 5
ct.wait 5
ct.release
else
MsgBox "Error creating the transition."
end if
```

Notes: Lasterror is set.

Must be called after performing a transition, but not before the transition is finished.

6.46.5 Run(duration as single)

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Runs a transition effect for the given duration.

Example:

```

Var r as CGSTransitionRequestMBS
Var co as CGSConnectionMBS // global property
Var ct as CGSTransitionMBS // global property
Var cw as CGSWindowMBS

co=new CGSConnectionMBS

cw=co.CGSWindow(window1)

r=new CGSTransitionRequestMBS
r.TransitionType=r.CGSCube
r.TransitionOption=r.CGSLeft
r.HasBackGround=false
r.HasBackColor=false
r.Win=cw

ct=co.NewTransition(r)
if ct<>Nil then
window1.Refresh // draw new window

ct.run(2)
else
MsgBox "Error creating the transition."
end if

```

Notes: Same as calling Invoke, Wait and Release.

6.46.6 Wait(duration as single)

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Waits the given time in second.

Example:

// instead of wait(5) you can do this to give time to other threads and timers of your application:

```

Var ct as CGSTransitionMBS
Var t as Integer

t=ticks+300 // 60th of a second * 300 = 5 seconds

while ticks<t
ct.wait(0.1)

```

```
app.YieldToNextThread  
wend
```

Notes: All the CPU time is given away to other applications to give maximum performance to the transition.

6.46.7 Properties

6.46.8 Connection as CGSConnectionMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The reference to the connection used.

Notes: Every CGS class has a reference to the connection so the connection object stays in memory as long as one of the depending objects is being used.

(Read only property)

6.46.9 ConnectionHandle as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The handle to the CGS connection being used.

Notes: Most of the functions need this value and it needs to be not zero.

(Read and Write property)

6.46.10 Handle as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal handle for this transition.

Notes: (Read and Write property)

6.47 class CGSTransitionRequestMBS

6.47.1 class CGSTransitionRequestMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Deprecated: This item is deprecated and should no longer be used. **Function:** A class for a transition request to the CoreGraphics System.

Notes: Requires Mac OS X 10.4.

All functions used by the CGS classes are private APIs, so Apple does not guarantee that they work in future Mac OS X versions.

Blog Entries

- [MBS Xojo Plugins, version 23.3pr7](#)

6.47.2 Methods

6.47.3 Run(duration as single) as boolean

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Runs a transition.

Example:

```

Var r as CGSTransitionRequestMBS
Var co as CGSConnectionMBS // global property
Var cw as CGSWindowMBS // global property

co=new CGSConnectionMBS

cw=co.CGSWindow(window1)

r=new CGSTransitionRequestMBS
r.TransitionType=r.CGSCube
r.TransitionOption=r.CGSLeft
r.HasBackGround=false
r.HasBackColor=false
r.Win=cw
call r.run(5)

```

Notes: Lasterror is set.

Transitions seem not to work in Carbon PEF builds. MachO works.

6.47.4 Properties

6.47.5 Blue as Single

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Blue part of the bgcolor.

Notes: Range is from 0.0 (no color) to 1.0 (full color).
(Read and Write property)

6.47.6 Green as Single

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Green part of the bgcolor.

Notes: Range is from 0.0 (no color) to 1.0 (full color).
(Read and Write property)

6.47.7 HasBackColor as Boolean

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether the background on the transition has a color.

Example:

```
Var r as CGSTransitionRequestMBS
Var co as CGSConnectionMBS // global property
Var ct as CGSTransitionMBS // global property
```

```
r=new CGSTransitionRequestMBS
r.TransitionType=r.CGSWarpSwitch
r.TransitionOption=r.CGSLeft
r.HasBackGround=true
r.HasBackColor=true
r.red=0 // all zero is black, all one is white
r.Blue=0
r.Green=0
```

```
ct=co.NewTransition(r)
if ct<>Nil then
window1.Refresh
ct.Invoke(2)
ct.Wait(2)
ct.Release
else
```

```
MsgBox "Error creating the transition."  
end if
```

Notes: (Read and Write property)

6.47.8 HasBackGround as Boolean

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether you want to use the background or not.

Example:

```
Var r as new CGSTransitionRequestMBS  
r.HasBackGround=false // best for a window in place transition
```

Notes: If HasBackGround is true, a background is drawn behind the transition which may have a background color.

(Read and Write property)

6.47.9 Red as Single

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Red part of the bgcolor.

Notes: Range is from 0.0 (no color) to 1.0 (full color).

(Read and Write property)

6.47.10 TransitionOption as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The transition options to use.

Example:

```
Var r as CGSTransitionRequestMBS  
r.TransitionOption=r.CGSLeft
```

Notes: One of the constants:

(Read and Write property)

CGSDown	= 0	Old desktop moves down.
CGSLeft	= 1	Old desktop moves left.
CGSRight	= 2	Old desktop moves right.
CGSInRight	= 3	CGSSwap: Old desktop moves into screen, new comes from right.
CGSBottomLeft	= 5	CGSSwap: Old desktop moves to bottom left, new comes from top right.
CGSBottomRight	= 6	Old desktop to bottom right, New from top left.
CGSDownTopRight	= 7	CGSSwap: Old desktop moves down, new from top right.
CGSUp	= 8	Old desktop moves up.
CGSTopLeft	= 9	Old desktop moves top left.
CGSTopRight	= 10	CGSSwap: old to top right. new from bottom left.
CGSUpBottomRight	= 11	CGSSwap: old desktop up, new from bottom right.
CGSInBottom	= 12	CGSSwap: old in, new from bottom.
CGSLeftBottomRight	= 13	CGSSwap: old one moves left, new from bottom right.
CGSRightBottomLeft	= 14	CGSSwap: old one moves right, new from bottom left.
CGSInBottomRight	= 15	CGSSwap: onl one in, new from bottom right.
CGSInOut	= 16	CGSSwap: old in, new out.

6.47.11 TransitionType as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The transition to use.

Notes: One of the constants:

CGSNone	= 0	No transition effect.
CGSFade	= 1	Cross-fade.
CGSZoom	= 2	Zoom/fade towards us.
CGSReveal	= 3	Reveal new desktop under old.
CGSSlide	= 4	Slide old out and new in.
CGSWarpFade	= 5	Warp old and fade out revealing new.
CGSSwap	= 6	Swap desktops over graphically.
CGSCube	= 7	The well-known cube effect.
CGSWarpSwitch	= 8	Warp old, switch and un-warp.
CGSFlip	= 9	The flip effect known from Dashboard

(Read and Write property)

6.47.12 Win as CGSWindowMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The window to use for the transition.

Notes: Set to nil to do a full screen transition.

(Read and Write property)

6.47.13 Constants

Constants

Constant	Value	Description
CGSBottomLeft	5	CGSSwap: Old desktop moves to bottom left, new comes from top right.
CGSBottomRight	6	Old desktop to bottom right, New from top left.
CGSCube	7	The well-known cube effect.
CGSDown	0	Old desktop moves down.
CGSDownTopRight	7	CGSSwap: Old desktop moves down, new from top right.
CGSFade	1	Cross-fade effect.
CGSFlip	9	The flip transition effect. (like Dashboard)
CGSInBottom	12	CGSSwap: old in, new from bottom.
CGSInBottomRight	15	CGSSwap: old one in, new from bottom right.
CGSInOut	16	CGSSwap: old in, new out.
CGSInRight	3	CGSSwap: Old desktop moves into screen, new comes from right.
CGSLeft	1	Old desktop moves left.
CGSLeftBottomRight	13	CGSSwap: old one moves left, new from bottom right.
CGSNone	0	No transition effect.
CGSReveal	3	Reveal new desktop under old.
CGSRight	2	Old desktop moves right.
CGSRightBottomLeft	14	CGSSwap: old one moves right, new from bottom left.
CGSSlide	4	Slide old out and new in.
CGSSwap	6	Swap desktops over graphically.
CGSTopLeft	9	Old desktop moves top left.
CGSTopRight	10	CGSSwap: old to top right. new from bottom left.
CGSup	8	Old desktop moves up.
CGSupBottomRight	11	CGSSwap: old desktop up, new from bottom right.
CGSWarpFade	5	Warp old and fade out revealing new.
CGSWarpSwitch	8	Warp old, switch and un-warp.
CGSZoom	2	Zoom/fade towards us.

6.48 class CGSValueMBS

6.48.1 class CGSValueMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: A class for a value to the CoreGraphics System.

Notes: Requires Mac OS X 10.4.

All functions used by the CGS classes are private APIs, so Apple does not guarantee that they work in future Mac OS X versions.

6.48.2 Methods

6.48.3 IntegerValue as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The integer value of this object.

6.48.4 StringValue as string

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The string value of this object.

6.48.5 Properties

6.48.6 Connection as CGSConnectionMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The reference to the connection used.

Notes: Every CGS class has a reference to the connection so the connection object stays in memory as long as one of the depending objects is being used.

(Read only property)

6.48.7 ConnectionHandle as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The handle to the CGS connection being used.

Notes: Most of the functions need this value and it needs to be not zero.
(Read and Write property)

6.48.8 Handle as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal used handle.

Notes: (Read and Write property)

6.49 class CGSWindowListMBS

6.49.1 class CGSWindowListMBS

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: An array of CGSWindowMBS objects.

Example:

```
Var c as new CGSConnectionMBS
Var list as CGSWindowListMBS = c.GetWindowList

Var names() as string
Var u as Integer = List.Count-1
for i as Integer = 0 to u
  Var w as CGSWindowMBS = List.Item(i)
  names.append w.Title
next

MsgBox str(List.Count)+" windows: "+Join(names, ", ")
```

Blog Entries

- [New in MBS Xojo Plugins in version 23.1](#)
- [MonkeyBread Software Releases the MBS Xojo Plugins in version 23.1](#)
- [MBS Xojo Plugins, version 23.1pr3](#)
- [MBS Xojo / Real Studio Plugins, version 15.1pr1](#)
- [MonkeyBread Software Releases the MBS Plugins 8.1](#)

6.49.2 Methods

6.49.3 Item(index as Integer) as CGSWindowMBS

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the window with the given index.

Notes: Index is from 0 to count-1.

6.49.4 PreflightScreenCaptureAccess as Boolean

Plugin Version: 23.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Checks whether the current process already has screen capture access.

Notes: For macOS 10.15 and newer.
Always returns false on other targets.

6.49.5 RequestScreenCaptureAccess as Boolean

Plugin Version: 23.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Requests event listening access if absent, potentially prompting.

Notes: For macOS 10.15 and newer.
Always returns false on other targets.

6.49.6 Properties

6.49.7 Connection as CGSConnectionMBS

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The reference to the connection used.

Notes: Every CGS class has a reference to the connection so the connection object stays in memory as long as one of the depending objects is being used.
(Read only property)

6.49.8 ConnectionHandle as Integer

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The handle to the CGS connection being used.

Notes: Most of the functions need this value and it needs to be not zero.
(Read and Write property)

6.49.9 Count as Integer

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The number of CGSWindow objects in that array.

Notes: (Read and Write property)

6.49.10 Handle as Integer

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The handle to the window list.

Notes: (Read and Write property)

6.50 class CGSWindowMBS

6.50.1 class CGSWindowMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: A class for a window in the CoreGraphics System.

Notes: Requires Mac OS X 10.4.

All functions used by the CGS classes are private APIs, so Apple does not guarantee that they work in future Mac OS X versions.

To find all processes on Mac, use the ProcessMBS class.

To find all windows on Windows, use the WindowsListMBS class.

Not official supported by Apple and some effects don't work well on latest OS X versions!

Blog Entries

- [New in MBS Xojo Plugins in version 23.1](#)
- [MonkeyBread Software Releases the MBS Plugins 8.1](#)

6.50.2 Methods

6.50.3 Flush

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Flushes this window.

Example:

```
Var co as CGSConnectionMBS
```

```
Var cw as CGSWindowMBS
```

```
co=new CGSConnectionMBS
```

```
cw=co.CGSWindow(window1)
```

```
cw.Flush
```

Notes: Lasterror is set.

6.50.4 Height as Double

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The height of this window.

Example:

```
Var co as CGSConnectionMBS
```

```
Var cw as CGSWindowMBS
```

```
co=new CGSConnectionMBS  
cw=co.CGSWindow(window1)
```

```
MsgBox str(cw.height)
```

6.50.5 Left as Double

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The left position of this window.

Example:

```
Var co as CGSConnectionMBS
```

```
Var cw as CGSWindowMBS
```

```
co=new CGSConnectionMBS  
cw=co.CGSWindow(window1)
```

```
MsgBox str(cw.left)
```

6.50.6 Level as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The window level of this window.

Notes: See WindowGroupMBS.level or OverlayWindow.Level for details.

6.50.7 Move(byref x as single, byref y as single)

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Moves the window.

6.50.8 Order(mode as Integer, relativeToWindow as CGSWindowMBS=nil)

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Reorders the window.

Notes: Lasterror is set.

Mode must be kCGSOrderAbove, kCGSOrderBelow or kCGSOrderOut.

6.50.9 Title as string

Plugin Version: 8.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The window title.

6.50.10 Top as Double

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The top position of this window.

Example:

```
Var co as CGSConnectionMBS
```

```
Var cw as CGSWindowMBS
```

```
co=new CGSConnectionMBS
```

```
cw=co.CGSWindow(window1)
```

```
MsgBox str(cw.top)
```

Notes: In Pixel from the top of the screen.

6.50.11 Uncover

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Uncovers the window.

Notes: Lasterror is set.

6.50.12 Warp(w as Integer, h as Integer, value as memoryblock)

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Sets the wrap of a window.

Example:

```

Var w,h as Integer
Var m as memoryblock
Var localx,localy, gobalx,gobaly as Integer
Var offset as Integer

```

```

m=newmemoryblock(w*h*16)
// for each row
// for each point in row
m.SingleValue(offset)=localx
offset=offset+4
m.SingleValue(offset)=localy
offset=offset+4
m.SingleValue(offset)=gobalx
offset=offset+4
m.SingleValue(offset)=gobaly
offset=offset+4

```

Notes: the memoryblock must be of size $w*h*sizeof(single)*4$.
a single is 4 bytes, so this is $16*w*h$
w is the number of horizontal points you define.
h is the number of vertical points you define.

Basicly you define for a local point in the window the point onscreen where it is moved to. See the example project for more details.

6.50.13 Width as Double

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The width of this window.

Example:

```

Var co as CGSConnectionMBS
Var cw as CGSWindowMBS

```

```

co=new CGSConnectionMBS
cw=co.CGSWindow(window1)

```

```

MsgBox str(cw.width)

```

6.50.14 Workspace as CGSWorkspaceMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The workspace where this window is part of.

6.50.15 Properties

6.50.16 Connection as CGSConnectionMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The reference to the connection used.

Notes: Every CGS class has a reference to the connection so the connection object stays in memory as long as one of the depending objects is being used.

(Read only property)

6.50.17 ConnectionHandle as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The handle to the CGS connection being used.

Notes: Most of the functions need this value and it needs to be not zero.

(Read and Write property)

6.50.18 Handle as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The window handle.

Notes: This is not a WindowRef or a NSWindow pointer!

(Read and Write property)

6.50.19 LastError as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Last error code reported.

Notes: The list of CoreGraphics error (from Mac OS X 10.4)

(Read and Write property)

kCGErrorSuccess	= 0	Success
kCGErrorFailure	= 1000	Failure
kCGErrorIllegalArgument	= 1001	Illegal Argument
kCGErrorInvalidConnection	= 1002	Invalid Connection
kCGErrorInvalidContext	= 1003	Invalid Context
kCGErrorCannotComplete	= 1004	Cannot complete
kCGErrorNameTooLong	= 1005	Name is too long
kCGErrorNotImplemented	= 1006	Not implemented
kCGErrorRangeCheck	= 1007	Out of bounds
kCGErrorTypeCheck	= 1008	Type error
kCGErrorNoCurrentPoint	= 1009	No current point
kCGErrorInvalidOperation	= 1010	Invalid Operation
kCGErrorNoneAvailable	= 1011	internal errors have taken 1012 1013 and 1014
kCGErrorApplicationRequiresNewerSystem	= 1015	the application being launched says in it's bundle info that it requires a newer version of the system than is currently running.
kCGErrorApplicationNotPermittedToExecute	= 1016	Macintosh Manager is active and this application is not permitted to run
kCGErrorApplicationIncorrectExecutableFormatFound	= 1023	the application being launched does not have any executable code for the current system.
kCGErrorApplicationIsLaunching	= 1024	The application is in the process of launching but hasn't checked in yet. Any launch data provided will be given to the application when it does check in.
kCGErrorApplicationAlreadyRunning	= 1025	The application being launched was already running (and had already checked in) and so any launch data provided can not be delivered to in by CPS
kCGErrorApplicationCanOnlyBeRunInOneSessionAtATime	= 1026	The application being launched is incompatible with multiple user sessions and is already running in another session by another user.
kCGErrorClassicApplicationsMustBeLaunchedByClassic	= 1027	To avoid deadlock Classic can't launch another Classic application by going thru CPS. This error gets returned in that case and it signals TruBlueEnvironment that it must handle this launch on its own.
kCGErrorForkFailed	= 1028	CPS was unable to fork a new process in order to launch an application.

6.50.20 AffineTransform as CGAffineTransformMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The transformation of this window.

Example:

```
// rotates the window.
Var a as CGAffineTransformMBS

a=CGAffineTransformMBS.MakeRotation(1)

a=a.Translate(-Width,-Height)

Var co as CGSConnectionMBS
Var cw as CGSWindowMBS

co=new CGSConnectionMBS
cw=co.CGSWindow(window1)

cw.AffineTransform=a
```

Notes: Lasterror is set.
(Read and Write computed property)

6.50.21 Alpha as single

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The alpha value for this window.

Example:

```
Var cw as CGSWindowMBS
Var co as new CGSConnectionMBS
```

```
cw=co.CGSWindow(window1)
cw.Alpha=0.2
```

```
// same as:
```

```
// self.TransparencyMBS=0.2
```

Notes: 0 is invisible and 1.0 is visible.

You may prefer to use the window TransparencyMBS property instead as it uses documented APIs. (Read and Write computed property)

6.50.22 EventMask as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The event mask of this window.

Notes: (Read and Write computed property)

6.50.23 Constants

Constants

Constant	Value	Description
kCGSOrderAbove	1	Window is ordered above target.
kCGSOrderBelow	-1	Window is ordered below target.
kCGSOrderOut	0	Window is removed from the on-screen.

6.51 class CGSWorkspaceMBS

6.51.1 class CGSWorkspaceMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: A class for a workspace to the CoreGraphics System.

Notes: Requires Mac OS X 10.4.

All functions used by the CGS classes are private APIs, so Apple does not guarantee that they work in future Mac OS X versions.

Not official supported by Apple and some effects don't work well on latest OS X versions!

6.51.2 Methods

6.51.3 GetWorkspaceWindowList as CGSWindowListMBS

Plugin Version: 8.1, Platform: macOS, Targets: Desktop only.

Function: Returns the list of windows for this workspace.

Notes: Returns nil on any error.

6.51.4 MoveWindows(target as CGSWorkspaceMBS)

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Moves all windows from one workspace to another.

Notes: Lasterror is set.

6.51.5 Properties

6.51.6 Connection as CGSConnectionMBS

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The reference to the connection used.

Notes: Every CGS class has a reference to the connection so the connection object stays in memory as long as one of the depending objects is being used.

(Read only property)

6.51.7 ConnectionHandle as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The handle to the CGS connection being used.

Notes: Most of the functions need this value and it needs to be not zero.
(Read and Write property)

6.51.8 Handle as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal used handle to the workspace.

Notes: (Read and Write property)

6.51.9 LastError as Integer

Plugin Version: 7.4, Platform: macOS, Targets: Desktop, Console & Web.

Function: Last error code reported.

Notes: The list of CoreGraphics error (from Mac OS X 10.4)

kCGErrorSuccess	= 0	Success
kCGErrorFailure	= 1000	Failure
kCGErrorIllegalArgument	= 1001	Illegal Argument
kCGErrorInvalidConnection	= 1002	Invalid Connection
kCGErrorInvalidContext	= 1003	Invalid Context
kCGErrorCannotComplete	= 1004	Cannot complete
kCGErrorNameTooLong	= 1005	Name is too long
kCGErrorNotImplemented	= 1006	Not implemented
kCGErrorRangeCheck	= 1007	Out of bounds
kCGErrorTypeCheck	= 1008	Type error
kCGErrorNoCurrentPoint	= 1009	No current point
kCGErrorInvalidOperation	= 1010	Invalid Operation
kCGErrorNoneAvailable	= 1011	internal errors have taken 1012 1013 and 1014
kCGErrorApplicationRequiresNewerSystem	= 1015	the application being launched says in it's bundle info that it requires a newer version of the system than is currently running.
kCGErrorApplicationNotPermittedToExecute	= 1016	Macintosh Manager is active and this application is not permitted to run
kCGErrorApplicationIncorrectExecutableFormatFound	= 1023	the application being launched does not have any executable code for the current system.
kCGErrorApplicationIsLaunching	= 1024	The application is in the process of launching but hasn't checked in yet. Any launch data provided will be given to the application when it does check in.
kCGErrorApplicationAlreadyRunning	= 1025	The application being launched was already running (and had already checked in) and so any launch data provided can not be delivered to in by CPS
kCGErrorApplicationCanOnlyBeRunInOneSessionAtATime	= 1026	The application being launched is incompatible with multiple user sessions and is already running in another session by another user.
kCGErrorClassicApplicationsMustBeLaunchedByClassic	= 1027	To avoid deadlock Classic can't launch another Classic application by going thru CPS. This error gets returned in that case and it signals TruBlueEnvironment that it must handle this launch on its own.
kCGErrorForkFailed	= 1028	CPS was unable to fork a new process in order to launch an application.

(Read and Write property)

6.52 class QDPictMBS

6.52.1 class QDPictMBS

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Deprecated: This item is deprecated and should no longer be used. **Function:** A class to draw QuickDraw PICT data to a CoreGraphics context.

Notes: Note: QuickDraw picture data typically comes in two forms: a PICT resource that begins the picture header data at the beginning of the resource and PICT files that begin with 512 bytes of arbitrary data, followed by the picture header data. For this reason, the routines that create a QTPict object attempt to find the picture header data beginning at either the first byte of the data provided or at byte 513 of the data provided.

Additionally the Picture Bounds must not be an empty rect.

This class has been deprecated by Apple.

Blog Entries

- [MBS Real Studio Plugins, version 12.1pr10](#)

6.52.2 Methods

6.52.3 Constructor(dataProvider as CGDataProviderMBS)

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a new QTPictMBS object based on a data provider.

Notes: Create a QDPict object, using dataProvider to obtain the QDPict's data.

It is assumed that either the first byte or the 513th byte of data in the file referenced by the URL is the first byte of the picture header. If the URL does not begin PICT data at one of these places in the data fork then the handle property will be 0.

See also:

- 6.52.4 Constructor(file as folderitem) 477
- 6.52.5 Constructor(url as string) 478

6.52.4 Constructor(file as folderitem)

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a new QTPictMBS object based on a file.

See also:

- 6.52.3 Constructor(dataProvider as CGDataProviderMBS) 477
- 6.52.5 Constructor(url as string) 478

6.52.5 Constructor(url as string)

Plugin Version: 12.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a new QTPictMBS object based on an url.

Notes: Create a QDPict from an url.

It is assumed that either the first byte or the 513th byte of data in the file referenced by the URL is the first byte of the picture header. If the URL does not begin PICT data at one of these places in the data fork then the handle property will be 0.

See also:

- 6.52.3 Constructor(dataProvider as CGDataProviderMBS) 477
- 6.52.4 Constructor(file as folderitem) 477

6.52.6 DrawToCGContext(context as CGContextMBS, r as CGRectMBS)

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Draws the picture into the given context.

Notes: You can increase the drawing resolution by passing a bigger rectangle.

Lasterror is set. -1 if context or rectangle are invalid.

Draw picture in the rectangular area specified by r. The PICT bounds of the page is scaled, if necessary, to fit into the rectangle. To get unscaled results, supply a rect the size of the rect returned by GetBounds.

6.52.7 GetBounds as CGRectMBS

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Queries the bounds of the picture.

Notes: This are the bounds in pixel for 72 dpi.

Return the Picture Bounds of the QuickDraw picture represented by the picture. This rectangle is in the default user space with one unit = 1/72 inch.

6.52.8 GetResolution(byref xRes as single, byref yRes as single)

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: Queries the resolution of the picture.

Notes: Return the resolution of the QuickDraw picture represented by the picture.

This data, together with the CGRect returned by GetBounds, can be used to compute the size of the picture

in pixels, which is what QuickDraw really records into pictures.

6.52.9 Height as Double

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The height of the picture.

Notes: This value is using 72 dpi.

6.52.10 HorizontalResolution as Double

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The horizontal resolution.

6.52.11 VerticalResolution as Double

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The vertical resolution.

6.52.12 Width as Double

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The width of the picture.

Notes: This value is using 72 dpi.

6.52.13 Properties

6.52.14 Handle as Integer

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The internal handle.

Notes: (Read and Write property)

6.52.15 LastError as Integer

Plugin Version: 9.3, Platform: macOS, Targets: Desktop, Console & Web.

Function: The last error code.

Notes: Set by DrawToCGContext.
(Read and Write property)

Chapter 7

CoreText

7.1 class CoreTextMBS

7.1.1 class CoreTextMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The central CoreText class.

Notes: Some global methods, constants and events for CoreText.

Blog Entries

- [News from the MBS Xojo Plugins Version 25.0](#)
- [MBS Xojo Plugins, version 24.6pr6](#)
- [CoreText Completed](#)

7.1.2 Methods

7.1.3 AvailableFontFamilyNames as string()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns an array of visible font family names sorted for user interface display.

7.1.4 AvailableFontURLs as string()

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of font URLs.

7.1.5 AvailablePostScriptNames as string()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns an array of unique PostScript font names for the fonts.

7.1.6 CompareFontFamilyNames(name1 as string, name2 as string) as Integer

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: A comparator function to compare font family names and sort them according to Apple guidelines.

Notes: family1: The first localized font family name to compare, as a string.

family2: The second localized font family name to compare, as a string.

Returns a `CFComparisonResult` value indicating the sort order for the two family names. `kCFComparisonResultGreatherThan` (1) if family1 is greater than family2, `kCFComparisonResultLessThan` (-1) if family1 is less than family2, and `kCFComparisonResultEqualTo` (0) if they are equal.

This `CFComparatorFunction` function compares font family names and sorts them in the Apple preferred order, accounting for foundry prefix. Family names with recognized prefixes are sorted after the unprefix names in prefix order.

7.1.7 Constructor

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The constructor.

7.1.8 CoreTextVersion as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the version of the CoreText framework.

Notes: This function returns a number indicating the version of the CoreText framework. Note that framework version is not always an accurate indicator of feature availability. The recommended way to use this function is first to check that the function pointer is non nil (plugin will do and raise exceptions), followed by calling it and comparing its result to a defined constant (or constants).

Returns the version number. This value is for comparison with the constants beginning with kCTVersion-Number.

7.1.9 CreateFontDescriptorFromData(data as memoryblock) as CTFontDescriptorMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a font descriptor representing the font in the supplied data.

Notes: Note: the font descriptor is not available through font descriptor matching.

data: A memoryblock containing font data.

Returns a font descriptor created from the data, or nil on error.

See also:

- 7.1.10 CreateFontDescriptorFromData(data as string) as CTFontDescriptorMBS 483

7.1.10 CreateFontDescriptorFromData(data as string) as CTFontDescriptorMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a font descriptor representing the font in the supplied data.

Notes: Note: the font descriptor is not available through font descriptor matching.

data: A string containing font data.

Returns a font descriptor created from the data, or nil on error.

See also:

- 7.1.9 CreateFontDescriptorFromData(data as memoryblock) as CTFontDescriptorMBS 483

7.1.11 CreateFontDescriptorsFromFile(file as folderitem) as CTFontDescriptorMBS()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns an array of font descriptors representing each of the fonts in the specified file.

Example:

```
Var file as FolderItem = SpecialFolder.Desktop.Child("Didot.ttc")
```

```
Var descriptor() as CTFontDescriptorMBS
```

```

Var result() as string

descriptor = CoreTextMBS.CreateFontDescriptorsFromFile(file)

for i as Integer = 0 to descriptor.Ubound
result.append descriptor(i).Displayname
next

MsgBox Join(result, EndOfLine)

```

Notes: file: A folderitem referencing a valid font file.

7.1.12 CreateFontDescriptorsFromURL(URL as string) as CTFontDescriptorMBS()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns an array of font descriptors representing each of the fonts in the specified URL.

Notes: URL: A file system URL referencing a valid font file.

7.1.13 Destructor

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The destructor.

7.1.14 EnableFontDescriptors(descriptors() as CTFontDescriptorMBS, enable as boolean)

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Enables or disables the matching font descriptors for font descriptor matching.

Notes: descriptors: Array of font descriptors.

enable: Boolean value indicating whether the fonts matching descriptors should be enabled for font descriptor matching.

7.1.15 GetScopeForFile(file as folderitem) as Integer

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the registration scope of the specified folderitem.

Notes: The registration scope of the specified file or kCTFontManagerScopeNone if not currently registered.

Available in OS X v10.6 and later.

7.1.16 GetScopeForURL(URL as string) as Integer

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the registration scope of the specified URL.

Notes: The registration scope of the specified URL or `kCTFontManagerScopeNone` if not currently registered.

Available in OS X v10.6 and later.

7.1.17 IsSupportedFontFile(file as folderitem) as boolean

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Determines whether the referenced font data is supported on the current platform.

Notes: Returns true if the folderitem refers to a valid font that can be used on the current platform; false otherwise.

7.1.18 IsSupportedFontURL(URL as string) as boolean

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Determines whether the referenced font data (usually by file URL) is supported on the current platform.

Notes: Returns true if the URL refers to a valid font that can be used on the current platform; false otherwise.

7.1.19 kCTBaselineClassAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Key to reference a baseline class override.

Notes: Value must be one of the `kCTBaselineClass` constants. Normally, glyphs on the line will be assigned baseline classes according to the 'bsln' or 'BASE' table in the font. This attribute may be used to change this assignment.

see also: `kCTBaselineClassRoman`, `kCTBaselineClassIdeographicCentered`, `kCTBaselineClassIdeographicLow`, `kCTBaselineClassIdeographicHigh`, `kCTBaselineClassHanging`, `kCTBaselineClassMath`.

7.1.20 kCTBaselineInfoAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Key to reference a baseline info dictionary.

Notes: Value must be a CFDictionaryRef. Normally, baseline offsets will be assigned based on the 'bsln' or 'BASE' table in the font. This attribute may be used to assign different offsets. Each key in the dictionary is one of the kCTBaselineClass constants and the value is a number of the baseline offset in points. You only need to specify the offsets you wish to change.

7.1.21 kCTBaselineReferenceInfoAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Key to reference a baseline info dictionary for the reference baseline.

Notes: Value must be a Dictionary. All glyphs in a run are assigned a baseline class and then aligned to the offset for that class in the reference baseline baseline info. See the discussion of kCTBaselineInfoAttributeName for information about the contents of the dictionary. You can also use the kCTBaselineReferenceFont key to specify that the baseline offsets of a particular CTFontMBS should be used as the reference offsets.

7.1.22 kCTCharacterShapeAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Controls glyph selection.

Notes: Value must be a number. Default is value is 0 (disabled). A non-zero value is interpreted as an SFNT kCharacterShapeType selector + 1; see SFNTLayoutTypes.h for selectors. For example, an attribute value of 1 corresponds to kTraditionalCharactersSelector.

7.1.23 kCTFontAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The key for the font.

Notes: Value must be a CTFontMBS. Default is Helvetica 12.

7.1.24 kCTFontManagerBundleIdentifier as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: CTFontManager bundle identifier.

Notes: The CTFontManager bundle identifier to be used with get or set global auto-activation settings.

7.1.25 kCTFontManagerErrorDomain as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The error domain for CoreText.

Notes: CFError objects with this domain have error codes corresponding to one of the CTFontManagerError errors listed in "Font Registration Errors" and "Font Unregistration Errors."

7.1.26 kCTFontManagerErrorFontURLsKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: User info key to be used with CFError references returned from registration functions.

Notes: The value associated with this key in the user info dictionary of a CFError object is a CFArray of font URLs that failed with the given error.

7.1.27 kCTFontManagerRegisteredFontsChangedNotification as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Notification name for font registry changes.

Notes: This is the string to use as the notification name when subscribing to Core Text Font Manager notifications. This notification is posted when fonts are added to the font registry. The client is responsible for registered with the distributed notification center to receive notifications for changes to the session or user scopes, and with a local notification center for changes to the process scope.

7.1.28 kCTFontSlantTrait as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Dictionary key to access the slant trait value.

Notes: Use this key to access the normalized slant angle from the font traits dictionary. The value returned is a number representing a float value between -1.0 and 1.0 for normalized slant angle. The value of 0.0 corresponds to 0 degree clockwise rotation from the vertical and 1.0 corresponds to 30 degrees clockwise rotation.

7.1.29 kCTFontSymbolicTrait as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Dictionary key to access the symbolic traits value.

Notes: Use this key to access the symbolic traits value from the font traits dictionary. The value is returned as a number.

7.1.30 kCTFontWeightTrait as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Dictionary key to access the weight trait value.

Notes: Use this key to access the normalized weight trait from the font traits dictionary. The value returned is a number representing a float value between -1.0 and 1.0 for normalized weight. The value of 0.0 corresponds to the regular or medium font weight.

7.1.31 kCTFontWidthTrait as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Dictionary key to access the width (condense/expand) trait value.

Notes: Use this key to access the normalized proportion trait from the font traits dictionary. This value corresponds to the relative inter-glyph spacing for a given font. The value returned is a number representing a float between -1.0 and 1.0. The value of 0.0 corresponds to regular glyph spacing while negative values represent condensed glyph spacing.

7.1.32 kCTForegroundColorAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The key for the foreground color..

Notes: Value must be a CGColorMBS. Default value is black.

7.1.33 kCTForegroundColorFromContextAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for font attributes.

Notes: Never set a foreground color in the CGContext; use what is set as the context's fill color.

Value must be a boolean. Default is false. The reason why this exists is because an NSAttributedString defaults to a black color if no color attribute is set. This forces CoreText to set the color in the context. This will allow developers to sidestep this, making CoreText set nothing but font information in the CGContext. If set, this attribute also determines the color used by kCTUnderlineStyleAttributeName, in which case it

overrides the foreground color.

7.1.34 kCTGlyphInfoAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for font attributes.

Notes: Allows the use of unencoded glyphs.

Value must be a CTGlyphInfoMBS. The glyph specified by this CTGlyphInfo object is assigned to the entire attribute range, provided that its contents match the specified base string and that the specified glyph is available in the font specified by kCTFontAttributeName.

7.1.35 kCTKernAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for font attributes.

Notes: A kerning adjustment.

Value must be a float. Default is standard kerning. The kerning attribute indicate how many points the following character should be shifted from its default offset as defined by the current character's font in points; a positive kern indicates a shift farther along and a negative kern indicates a shift closer to the current character. If this attribute is not present, standard kerning will be used. If this attribute is set to 0.0, no kerning will be done at all.

7.1.36 kCTLanguageAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for font attributes.

Notes: Specifies text language.

Value must be a String containing a locale identifier. Default is unset. When this attribute is set to a valid identifier, it will be used to select localized glyphs (if supported by the font) and locale-specific line breaking rules.

7.1.37 kCTLigatureAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for font attributes.

Notes: Controls ligature formation.

Value must be a number. Default is int value 1. The ligature attribute determines what kinds of ligatures should be used when displaying the string. A value of 0 indicates that only ligatures essential for proper rendering of text should be used, 1 indicates that standard ligatures should be used, and 2 indicates that all available ligatures should be used. Which ligatures are standard depends on the script and possibly the font. Arabic text, for example, requires ligatures for many character sequences, but has a rich set of additional ligatures that combine characters. English text has no essential ligatures, and typically has only two standard ligatures, those for "fi" and "fl" – all others being considered more advanced or fancy.

On iOS releases prior to 6.0 essential ligatures are applied if the font contains glyphs for any of U+FB00 through U+FB04 and the font lacks AAT or OpenType shaping tables, but as of 6.0 shaping tables (or the lack thereof) are treated as definitive. This character-based shaping will still be performed if this attribute is explicitly specified with the default value of 1.

7.1.38 `kCTParagraphStyleAttributeName` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for font attributes.

Notes: A `CTParagraphStyle` object which is used to specify things like line alignment, tab rulers, writing direction, etc.

Value must be a `CTParagraphStyleMBS`. Default is an empty `CTParagraphStyle` object.

7.1.39 `kCTRunDelegateAttributeName` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for font attributes.

Notes: Allows customization of certain aspects of a range of text's appearance.

Value must be a `CTRunDelegateMBS`. The values returned by the embedded object for an attribute range apply to each glyph resulting from the text in that range. Because an embedded object is only a display-time modification, care should be taken to avoid applying this attribute to a range of text with complex behavior, such as a change of writing direction, combining marks, etc. Consequently, it is recommended that this attribute be applied to a range containing the single character U+FFFC.

7.1.40 `kCTStrokeColorAttributeName` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for font attributes.

Notes: The stroke color.
Value must be a CGColorMBS. Default is the foreground color.

7.1.41 kCTStrokeWidthAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for font attributes.

Notes: The stroke width.

Value must be a number. Default value is 0.0, or no stroke. This attribute, interpreted as a percentage of font point size, controls the text drawing mode: positive values effect drawing with stroke only; negative values are for stroke and fill. A typical value for outlined text is 3.0.

7.1.42 kCTSuperscriptAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for font attributes.

Notes: Controls vertical text positioning.

Value must be a number. Default is int value 0. If supported by the specified font, a value of 1 enables superscripting and a value of -1 enables subscripting.

7.1.43 kCTUnderlineColorAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for font attributes.

Notes: The underline color.

Value must be a CGColorMBS. Default is the foreground color.

7.1.44 kCTUnderlineStyleAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for font attributes.

Notes: Allows the setting of an underline to be applied at render time.

Value must be a number. Default is kCTUnderlineStyleNone. Set a value of something other than kCTUnderlineStyleNone to draw an underline. In addition, the CTUnderlineStyleModifiers can be used to modify the look of the underline. The underline color will be determined by the text's foreground color.

7.1.45 `kCTVerticalFormsAttributeName` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for font attributes.

Notes: Controls glyph orientation.

Value must be a boolean. Default is false. A value of false indicates that horizontal glyph forms are to be used, true indicates that vertical glyph forms are to be used.

7.1.46 `kCTWritingDirectionAttributeName` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for font attributes.

Notes: Specifies a bidirectional override or embedding.

Value must be a CFArray of CFNumberRefs, each of which should have a value of either `kCTWritingDirectionLeftToRight` or `kCTWritingDirectionRightToLeft`, plus one of `kCTWritingDirectionEmbedding` or `kCTWritingDirectionOverride`. This array represents a sequence of nested bidirectional embeddings or overrides, in order from outermost to innermost, with (`kCTWritingDirectionLeftToRight` | `kCTTextWritingDirectionEmbedding`) corresponding to a LRE/PDF pair in plain text or `` in HTML, (`kCTWritingDirectionRightToLeft` | `kCTTextWritingDirectionEmbedding`) corresponding to a RLE/PDF pair in plain text or a `` in HTML, (`kCTWritingDirectionLeftToRight` | `kCTTextWritingDirectionOverride`) corresponding to a LRO/PDF pair in plain text or `<bdo dir="ltr"></bdo>` in HTML, and (`kCTWritingDirectionRightToLeft` | `kCTTextWritingDirectionOverride`) corresponding to a RLO/PDF pair in plain text or `<bdo dir="rtl"></bdo>` in HTML.

7.1.47 `MatchFontDescriptorsWithProgressHandler(descriptors())` as `CTFontDescriptorMBS`, `mandatoryAttributes()` as string, `tag` as `Variant = nil` as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Matches font descriptors in background.

Notes: This function returns immediately, but can potentially take long time to process. The progress is notified via progress evnet.

`descriptors`: An array of descriptors to process.

`mandatoryAttributes`: some mandatory attributes.

Returns false if it couldn't start the work.

7.1.48 RegisterFontsForFile(file as folderitem, scope as Integer, byref error as CFErrorMBS) as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Registers fonts from the specified font URL with the Font Manager. Registered fonts are discoverable through font descriptor matching.

Notes: fontURL: The font URL.

scope: Scope constant defining the availability and lifetime of the registration. See "Font Registration Scope" for values to pass for this parameter.

error: a CFErrorMBS object which, in case of failed registration, contains error information.

Returns true if registration of the fonts was successful, otherwise false.

7.1.49 RegisterFontsForFiles(files() as folderitem, scope as Integer, errors() as CFErrorMBS) as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Registers fonts from the specified array of font URLs with the Font Manager. Registered fonts are discoverable through font descriptor matching.

Notes: files: Array of font files.

scope: Scope constant defining the availability and lifetime of the registration. See "Font Registration Scope" for values to pass for this parameter.

errors: An array of CFErrorMBS objects which, in case of failed registration, contain error information. Each error contains a CFArrayMBS of font URLs corresponding to kCTFontManagerErrorFontURLsKey. These URLs represent the font files that caused the error and were not successfully registered. The array must be released by the caller. Can be nil.

Returns true if registration of all font URLs was successful, otherwise false.

7.1.50 RegisterFontsForURL(URL as string, scope as Integer, byref error as CFErrorMBS) as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Registers fonts from the specified font URL with the Font Manager. Registered fonts are discoverable through font descriptor matching.

Notes: fontURL: The font URL.

scope: Scope constant defining the availability and lifetime of the registration. See "Font Registration Scope" for values to pass for this parameter.

error: a CFErrorMBS object which, in case of failed registration, contains error information.

Returns true if registration of the fonts was successful, otherwise false.

See also:

- 7.1.51 RegisterFontsForURL(URLs() as string, scope as Integer, errors() as CFErrorMBS) as boolean
494

7.1.51 RegisterFontsForURL(URLs() as string, scope as Integer, errors() as CFErrorMBS) as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Registers fonts from the specified array of font URLs with the Font Manager. Registered fonts are discoverable through font descriptor matching.

Notes: URLs: Array of font URLs.

scope: Scope constant defining the availability and lifetime of the registration. See "Font Registration Scope" for values to pass for this parameter.

errors: Array of CFErrorMBS objects which, in case of failed registration, contain error information. Each error contains a CFArrayMBS of font URLs corresponding to kCTFontManagerErrorFontURLsKey. These URLs represent the font files that caused the error and were not successfully registered. The array must be released by the caller. Can be nil.

Please pass in errors array empty as

Returns true if registration of all font URLs was successful, otherwise false.

See also:

- 7.1.50 RegisterFontsForURL(URL as string, scope as Integer, byref error as CFErrorMBS) as boolean
493

7.1.52 RegisterGraphicsFont(font as CGFontMBS, byref error as CFErrorMBS) as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Registers the specified graphics font with the font manager.

Notes: font: The graphics font to be registered.

error: Returns by indirection an error object in the case of failed registration.

Returns true if registration of the font was successful, otherwise false.

Registered fonts are discoverable through font descriptor matching. Any attempt to register a font that is either already registered or contains the same Postscript of an already registered font will fail. This behavior is useful for fonts that may be embedded in documents or constructed in memory. A graphics font is obtained by calling CGFontMBS.CreateWithDataProvider. Fonts that are backed by files should be registered using CoreTextMBS.RegisterFontsForURL.

7.1.53 RequestFonts(descriptors()) as CTFontDescriptorMBS, tag as variant = nil)

Plugin Version: 25.0, Platform: macOS, Targets: iOS only.

Function: Resolves font descriptors specified on input.

Example:

```
ct = New MyCoreTextMBS // your subclass with the event
Var descriptors() As CTFontDescriptorMBS

descriptors.add CTFontDescriptorMBS.CreateWithNameAndSize("testing", 12)

ct.RequestFonts descriptors
```

Notes: descriptors: An array of font descriptors to make available to the process. The keys for describing the fonts may be a combination of kCTFontNameAttribute, kCTFontFamilyNameAttribute, or kCTFontRegistrationUserInfoAttribute.

On iOS, fonts registered by font provider apps in the CTFontManagerScope.persistent scope aren't automatically available to other apps. Other apps must call this function to make the fonts available for font descriptor matching.

On iOS, if the font descriptors can't be found, the system presents the user with a dialog that indicates which fonts couldn't be resolved. The system may provide the user with a way to resolve the missing fonts, if the font manager has a way to enable them.

7.1.54 UnregisterFontsForFile(file as folderitem, scope as Integer, byref error as CFErrorMBS) as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Unregisters fonts from the specified font URL with the Font Manager. Unregistered fonts are no longer discoverable through font descriptor matching.

Notes: URL: The font URL.

scope: Scope constant defining the availability and lifetime of the registration. See "Font Registration Scope" for values to pass for this parameter.

error: A CFError object which, in case of failed registration, contains error information.

Returns true if unregistration of the fonts was successful, otherwise false.

7.1.55 `UnregisterFontsForFiles(files() as folderitem, scope as Integer, errors() as CFErrorMBS) as boolean`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Unregisters fonts from the specified array of font URLs with the Font Manager. Unregistered fonts are no longer discoverable through font descriptor matching.

Notes: files: Array of font folderitems.

scope: Scope constant defining the availability and lifetime of the registration. See "Font Registration Scope" for values to pass for this parameter.

errors: An array of CFError objects which, in case of failed registration, contain error information. Each error contains a CFArray of font URLs corresponding to `kCTFontManagerErrorFontURLsKey`. These URLs represent the font files that caused the error and were not successfully registered. The array must be released by the caller.

Returns true if unregistration of all font URLs was successful, otherwise false.

7.1.56 `UnregisterFontsForURL(URL as string, scope as Integer, byref error as CFErrorMBS) as boolean`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Unregisters fonts from the specified font URL with the Font Manager. Unregistered fonts are no longer discoverable through font descriptor matching.

Notes: URL: The font URL.

scope: Scope constant defining the availability and lifetime of the registration. See "Font Registration Scope" for values to pass for this parameter.

error: A CFError object which, in case of failed registration, contains error information.

Returns true if unregistration of the fonts was successful, otherwise false.

7.1.57 `UnregisterFontsForURLs(URLs() as string, scope as Integer, errors() as CFErrorMBS) as boolean`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Unregisters fonts from the specified array of font URLs with the Font Manager. Unregistered fonts are no longer discoverable through font descriptor matching.

Notes: URLs: Array of font URLs.

scope: Scope constant defining the availability and lifetime of the registration. See "Font Registration Scope" for values to pass for this parameter.

errors: An array of CFError objects which, in case of failed registration, contain error information. Each error contains a CFArray of font URLs corresponding to `kCTFontManagerErrorFontURLsKey`. These URLs

represent the font files that caused the error and were not successfully registered. The array must be released by the caller.

Returns true if unregistration of all font URLs was successful, otherwise false.

7.1.58 UnregisterGraphicsFont(font as CGFontMBS, byref error as CFErrorMBS) as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Unregisters the specified graphics font with the font manager.

Notes: font: The graphics font to be unregistered.

error: Returns by indirection an error object in the case of failed unregistration.

Returns true if unregistration of the font was successful, otherwise false.

Unregistered fonts are no longer discoverable through font descriptor matching. Fonts that are backed by files should be unregistered using `CTFontManagerUnregisterFontsForURL`.

7.1.59 Properties

7.1.60 AutoActivationSetting(BundleID as string) as Integer

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The auto-activation setting for the specified bundle identifier.

Notes: bundleID: The bundle identifier used to specify a particular application bundle. If "", the current application bundle is used. If `kCTFontManagerBundleIdentifier` is specified, sets global auto-activation.

(Read and Write computed property)

7.1.61 Events

7.1.62 FontCollectionSortDescriptors(first as CTFontDescriptorMBS, second as CTFontDescriptorMBS, tag as Variant) as Integer

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: The event to sort font descriptors..

Notes: This callback can be specified to obtain the matching font descriptors of a collection in sorted order.

Return the appropriate comparison result of first descriptor to second descriptor.

Return -1 if smaller, 0 if equal or 1 if bigger.

7.1.63 Progress(state as Integer, progressParameter as Dictionary, tag as Variant) as boolean

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Progress was made for a call to MatchFontDescriptorsWithProgressHandler.

Notes: Return true to continue, and return false to cancel the process.

7.1.64 RequestFontsCompleted(unresolvedFontDescriptors() as CTFontDescriptorMBS, tag as variant)

Plugin Version: 25.0, Platform: macOS, Targets: .

Function: An event called after the request operation completes.

Notes: This event takes a unresolvedFontDescriptors parameter that contains an array of descriptors that couldn't be resolved or found. The array can be empty if all descriptors resolved.

7.1.65 Constants

Font Class Constants

Constant	Value	Description
kCTFontClassClarendonSerifs	&H04000000	Clarendon Serifs
kCTFontClassFreeformSerifs	&H07000000	Freeform Serifs
kCTFontClassMaskShift	28	The font class shift. This is used to shift the font class to the upper most 4 bits of the s traits.
kCTFontClassModernSerifs	&H03000000	Modern Serifs
kCTFontClassOldStyleSerifs	&H01000000	OldStyleSerifs
kCTFontClassOrnamentals	&H09000000	Ornamentals
kCTFontClassSansSerif	&H08000000	Sans Serif
kCTFontClassScripts	&H0A000000	Scripts
kCTFontClassSlabSerifs	&H05000000	Slab Serifs
kCTFontClassSymbolic	&H0C000000	Symbolic
kCTFontClassTransitionalSerifs	&H02000000	Transitional Serifs
kCTFontClassUnknown	&H00000000	Unknown

Font Auto-Activation Settings

Constant	Value	Description
kCTFontManagerAutoActivationDefault	0	Default auto-activation setting. When specified, the application uses this setting.
kCTFontManagerAutoActivationDisabled	1	Disables auto-activation.
kCTFontManagerAutoActivationEnabled	2	Enables auto-activation.
kCTFontManagerAutoActivationPromptUser	3	Requires user input for auto-activation. A dialog is presented to confirm auto-activation of the font.

Font Registration Errors

Constant	Value	Description
kCTFontManagerErrorAlreadyRegistered	105	The file has already been registered in the specified scope.
kCTFontManagerErrorFileNotFound	101	The file does not exist at the specified URL.
kCTFontManagerErrorInsufficientPermissions	102	Cannot access the file due to insufficient permissions.
kCTFontManagerErrorInvalidFontData	104	The file contains invalid font data that could cause system problems.
kCTFontManagerErrorUnrecognizedFormat	103	The file is not a recognized or supported font file format.

Font Unregistration Errors

Constant	Value	Description
kCTFontManagerErrorInUse	202	The font file is actively in use and cannot be unregistered.
kCTFontManagerErrorNotRegistered	201	The file is not registered in the specified scope.
kCTFontManagerErrorSystemRequired	202	The file is required by the system and cannot be unregistered.

Font Registration Scope

Constant	Value	Description
kCTFontManagerScopeNone	0	No scope is defined.
kCTFontManagerScopeProcess	1	The font is available to the current process for the duration of the process unless directly unregistered.
kCTFontManagerScopeSession	3	The font is available to the current user session but will not be available in subsequent sessions.
kCTFontManagerScopeUser	2	The font is available to all processes for the current user session and will be available in subsequent sessions unless unregistered.

Font Traits

Constant	Value	Description
kCTFontTraitBold	2	Bold. Additional detail available via kCTFontWeightTrait
kCTFontTraitClassMask	4026531840	Mask for the font class
kCTFontTraitColorGlyphs	8192	Color bitmap glyphs are available.
kCTFontTraitComposite	16384	Composite The font is a CFR (Composite font reference), a cascade list is expected per font.
kCTFontTraitCondensed	64	Condensed Additional detail available via kCTFontWidthTrait
kCTFontTraitExpanded	32	Expanded Expanded and condensed traits are mutually exclusive
kCTFontTraitItalic	1	Italic Additional detail available via kCTFontSlantTrait
kCTFontTraitMonoSpace	1024	MonoSpace Use fixed-pitch glyphs if available. May have multiple glyph advances (most CJK glyphs may contain two spaces)
kCTFontTraitUIOptimized	4096	UI optimized Synthesize appropriate attributes for UI rendering such as control titles if necessary
kCTFontTraitVertical	2048	Vertical Use vertical glyph variants and metrics.

Underline Pattern

Constant	Value	Description
kCTUnderlinePatternDash	&h0200	Dash
kCTUnderlinePatternDashDot	&h0300	Dash Dot
kCTUnderlinePatternDashDotDot	&h0400	Dash Dot Dot
kCTUnderlinePatternDot	&h0100	Dot
kCTUnderlinePatternSolid	&h0000	Solid

Underline style specifiers.

Constant	Value	Description
kCTUnderlineStyleDouble	9	Double underlined.
kCTUnderlineStyleNone	0	Not underlined.
kCTUnderlineStyleSingle	1	Single underlined.
kCTUnderlineStyleThick	2	Thick underlined.

Version Constants

Constant	Value	Description
kCTVersionNumber10_5	&h00020000	Mac OS X 10.5
kCTVersionNumber10_5_2	&h00020001	Mac OS X 10.5.2
kCTVersionNumber10_5_3	&h00020002	Mac OS X 10.5.3
kCTVersionNumber10_5_5	&h00020003	Mac OS X 10.5.5
kCTVersionNumber10_6	&h00030000	Mac OS X 10.6
kCTVersionNumber10_7	&h00040000	Mac OS X 10.7
kCTVersionNumber10_8	&h00050000	Mac OS X 10.8
kCTVersionNumber10_9	&h00060000	Mac OS X 10.9

Writing Directions

Constant	Value	Description
kCTWritingDirectionEmbedding	0	
kCTWritingDirectionOverride	1	

7.2 class CTFontCollectionMBS

7.2.1 class CTFontCollectionMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The CTFontCollection opaque type represents a font collection, that is, a group of font descriptors taken together as a single object.

Notes: Font collections provide the capabilities of font enumeration, access to global and custom font collections, and access to the font descriptors comprising the collection.

Subclass of the CFontMBS class.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [CoreText Completed](#)

7.2.2 Methods

7.2.3 Available as boolean

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Whether this class is available.

Notes: Returns true in OS X v10.5 and later.

7.2.4 Constructor

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The private constructor.

7.2.5 CopyWithFontDescriptors(queryDescriptors() as CTFontDescriptorMBS, options as dictionary) as CTFontCollectionMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a copy of the original collection augmented with the given new font descriptors.

Notes: queryDescriptors: An array of font descriptors to augment those of the original collection.

options: The options dictionary.

Returns a copy of the original font collection augmented by the new font descriptors and options.

The new font descriptors are merged with the existing descriptors to create a single set.
Available in OS X v10.5 and later.

7.2.6 CreateCopyWithFontDescriptors(original as CTFontCollectionMBS, queryDescriptors() as CTFontDescriptorMBS, options as dictionary) as CTFontCollectionMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a copy of the original collection augmented with the given new font descriptors.

Notes: queryDescriptors: An array of font descriptors to augment those of the original collection.

options: The options dictionary.

Returns a copy of the original font collection augmented by the new font descriptors and options.

The new font descriptors are merged with the existing descriptors to create a single set.
Available in OS X v10.5 and later.

7.2.7 CreateFromAvailableFonts(options as Dictionary) as CTFontCollectionMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a new font collection containing all available fonts.

Notes: options: The options dictionary.

Returns a new collection containing all fonts available to the current application.

7.2.8 CreateWithFontDescriptors(queryDescriptors() as CTFontDescriptorMBS, options as dictionary) as CTFontCollectionMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a new collection based on the array of font descriptors.

Notes: An array of font descriptors to use for matching.

options: The options dictionary. See constant option keys.

Returns this function creates a new collection based on the provided font descriptors. The contents of this collection is defined by matching the provided descriptors against all available font descriptors.

7.2.9 ExclusionDescriptors as CTFontDescriptorMBS()

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the array of descriptors to exclude from the match.

7.2.10 FontAttribute(attributeName as string, options as Integer) as Dictionary()

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of font descriptor attribute values.

Notes: attributeName: The attribute to retrieve for each descriptor in the collection.

options: Options to alter the return value.

This function returns a retained reference to an array, or nil on error. The caller is responsible for releasing the array. The array contains one value for each descriptor, in the same order as the results from CreateMatchingDescriptors. When the kCTFontCollectionCopyUnique is set, duplicate values will be removed. When kCTFontCollectionCopyStandardSort is set, the values will be sorted in standard UI order.

See also:

- 7.2.11 FontAttribute(attributeNames() as string, options as Integer) as Dictionary() 504

7.2.11 FontAttribute(attributeNames() as string, options as Integer) as Dictionary()

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of dictionaries containing font descriptor attribute values.

Notes: This function returns a retained reference to a array, or nil on error. The caller is responsible for releasing the array. The array contains one value for each descriptor, in the same order as the results from CreateMatchingDescriptors. When the kCTFontCollectionCopyUnique is set, duplicate values will be removed. When kCTFontCollectionCopyStandardSort is set, the values will be sorted in standard UI order.

See also:

- 7.2.10 FontAttribute(attributeName as string, options as Integer) as Dictionary() 504

7.2.12 kCTFontCollectionDisallowAutoActivationOption as string

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the Collection Matching Options.

Notes: Option key to avoid auto-activating fonts.

Specify this option key in the options dictionary with a non-zero value to disallow searches for missing fonts

(font descriptors returning no results).

7.2.13 kCTFontCollectionIncludeDisabledFontsOption as string

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the Collection Matching Options.

Notes: Specify this option key in the options dictionary with a non-zero value to enable matching of disabled fonts. You can pass font descriptors specifying disabled fonts to CTFontManagerEnableFontDescriptors, but you cannot use such a font descriptor to query font attributes from the system database or create a CTFontMBS.

7.2.14 kCTFontCollectionRemoveDuplicatesOption as string

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: One of the Collection Matching Options.

Notes: Option key to specify filtering of duplicates.

7.2.15 MatchingFontDescriptors(options as dictionary = nil) as CTFontDescriptorMBS()

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of font descriptors matching the collection.

7.2.16 MatchingFontDescriptorsForFamily(familyName as string, options as dictionary = nil) as CTFontDescriptorMBS()

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns an array of font descriptors matching the specified family, one descriptor for each style in the collection.

7.2.17 MatchingFontDescriptorsSorted(tag as Variant) as CTFontDescriptorMBS()

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the array of matching font descriptors sorted with the callback function.

Notes: This function returns an array of font descriptors matching the criteria of the collection and sorted by the results of the sorting callback function.

Calls `FontCollectionSortDescriptors` in `CoreTextMBS` class.

7.2.18 MutableCopy as CTMutableFontCollectionMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns a mutable copy of the original collection.

7.2.19 QueryDescriptors as CTFontDescriptorMBS()

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Returns the array of descriptors to match.

7.2.20 Constants

Bulk attribute access

Constant	Value	Description
<code>kCTFontCollectionCopyDefaultOptions</code>	0	Passing this option indicates that defaults are to be used. Available in OS X v10.7 and later.
<code>kCTFontCollectionCopyStandardSort</code>	2	Passing this option indicates that the return values should be sorted in standard UI order, suitable for display to the user. This is the same sorting behavior used by <code>NSFontPanel</code> and <code>Font Book</code> . Available in OS X v10.7 and later.
<code>kCTFontCollectionCopyUnique</code>	1	Passing this option indicates that duplicate values should be removed from the results. Available in OS X v10.7 and later.

7.3 class CTFontDescriptorMBS

7.3.1 class CTFontDescriptorMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The CTFontDescriptor opaque type represents a font descriptor, that is, a dictionary of attributes (such as name, point size, and variation) that can completely specify a font.

Notes: A font descriptor can be an incomplete specification, in which case the system chooses the most appropriate font to match the given attributes.

Subclass of the CFontMBS class.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [CoreText Completed](#)

7.3.2 Methods

7.3.3 AttributeValue(key as string) as Variant

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the value associated with an arbitrary attribute.

Notes: key: The requested attribute.

Returns a attribute value, or nil if the requested attribute is not present.

7.3.4 AttributeValues as Dictionary

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the attributes dictionary of the font descriptor.

Notes: The font descriptor attributes dictionary. This dictionary contains the minimum number of attributes to specify fully this particular font descriptor.

7.3.5 Available as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Whether this class is available.

Notes: Returns true in OS X v10.5 and later.

7.3.6 Constructor

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The private constructor.

7.3.7 CopyWithAttributes(attributeValues as Dictionary) as CTFontDescriptorMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates a copy of the original font descriptor with new attributes.

Notes: attributes: A dictionary containing arbitrary attributes.

Returns a new copy of the original font descriptor with attributes augmented by those specified. If there are conflicts between attributes, the new attributes replace existing ones.

7.3.8 CopyWithFamily(family as String) as CTFontDescriptorMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a new font descriptor in the specified family based on the traits of the original descriptor.

Notes: family: The name of the desired family.

Returns a new font reference with the original traits in the given family, or nil if none found in the system.

7.3.9 CopyWithFeature(featureTypeIdentifier as Integer, featureSelectorIdentifier as Integer) as CTFontDescriptorMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Copies a font descriptor with new feature settings.

Notes: featureTypeIdentifier: The feature type identifier.

featureSelectorIdentifier: The feature selector identifier.

Returns a copy of the original font descriptor modified with the given feature settings. This is a convenience method to toggle more easily the state of individual features.

7.3.10 CopyWithSymbolicTraits(symTraitValue as Integer, symTraitMask as Integer) as CTFontDescriptorMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a new font descriptor based on the original descriptor having the specified symbolic traits.

Notes: symTraitValue: The value of the symbolic traits. This bitfield is used to indicate the desired value for the traits specified by the symTraitMask parameter. Used in conjunction, they can allow for trait removal as well as addition.

symTraitMask: The mask bits of the symbolic traits. This bitfield is used to indicate the traits that should be changed.

Returns a new font descriptor reference in the same family with the given symbolic traits, or nil if none found in the system.

7.3.11 CopyWithVariation(variationIdentifier as Integer, variationValue as Double) as CTFontDescriptorMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates a copy of the original font descriptor with a new variation instance.

Notes: variationIdentifier: The variation axis identifier. This is the four-character code of the variation axis as a number.

variationValue: The value corresponding with the variation instance.

Returns a copy of the original font descriptor with a new variation instance.

This is a convenience method for easily creating new variation font instances.

7.3.12 CreateCopyWithFamily(original as CTFontDescriptorMBS, family as String) as CTFontDescriptorMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a new font descriptor in the specified family based on the traits of the original descriptor.

Notes: original: The original font descriptor reference.

family: The name of the desired family.

Returns a new font reference with the original traits in the given family, or nil if none found in the system.

7.3.13 CreateCopyWithSymbolicTraits(original as CTFontDescriptorMBS, symTraitValue as Integer, symTraitMask as Integer) as CTFontDescriptorMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a new font descriptor based on the original descriptor having the specified symbolic traits.

Example:

```
// find PostScript names for a font
const nameOfAFont = "Arial"

Var FontName as string
Var FontFamilyName as string
Var FontPostscriptName as string
Var FontPostscriptNameBold as string
Var FontPostscriptNameItalic as string

Var f as CTFontMBS = CTFontMBS.CreateWithName(nameOfAFont, 13)

if f <> nil then
  FontName = f.FullName
  FontFamilyName = f.FamilyName
  FontPostscriptName = f.PostScriptName

  Var d as CTFontDescriptorMBS = f.FontDescriptor

  Var db as CTFontDescriptorMBS = CTFontDescriptorMBS.CreateCopyWithSymbolicTraits(d, CoreTextMBS.kCTFontTraitBold, CoreTextMBS.kCTFontTraitBold)
  Var di as CTFontDescriptorMBS = CTFontDescriptorMBS.CreateCopyWithSymbolicTraits(d, CoreTextMBS.kCTFontTraitItalic, CoreTextMBS.kCTFontTraitItalic)

  Var fb as CTFontMBS = f.CreateCopyWithAttributes(f.Size, nil, db)
  Var fi as CTFontMBS = f.CreateCopyWithAttributes(f.Size, nil, di)

  FontPostscriptNameBold = fb.PostScriptName
  FontPostscriptNameItalic = fi.PostScriptName

end if

Break // read names in debugger
```

Notes: original: The original font descriptor reference.

symTraitValue: The value of the symbolic traits. This bitfield is used to indicate the desired value for the traits specified by the symTraitMask parameter. Used in conjunction, they can allow for trait removal as well as addition.

symTraitMask: The mask bits of the symbolic traits. This bitfield is used to indicate the traits that should be changed.

Returns a new font descriptor reference in the same family with the given symbolic traits, or nil if none found in the system.

Requires Mac OS X 10.9 or newer.

7.3.14 CreateWithAttributes(attributeValues as Dictionary) as CTFontDescriptorMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates a new font descriptor reference from a dictionary of attributes.

Notes: The provided attribute dictionary can contain arbitrary attributes that are preserved; however, unrecognized attributes are ignored on font creation and may not be preserved over the round trip from descriptor to font and back to descriptor.

7.3.15 CreateWithNameAndSize(Name as string, Size as Double = 0.0) as CTFontDescriptorMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates a new font descriptor with the provided PostScript name and size.

Example:

```
Var c as CTFontDescriptorMBS = CTFontDescriptorMBS.CreateWithNameAndSize("Times", 12)
MsgBox c.AttributeValue(c.kCTFontFamilyNameAttribute)
```

Notes: name: The PostScript name to be used for the font descriptor as a string.

size: The point size. If 0.0, the font size attribute (kCTFontSizeAttribute) is omitted from the returned font descriptor.

Return sa new font descriptor reference with the given PostScript name and point size.

7.3.16 kCTFontBaselineAdjustAttribute as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key to specify or obtain the baseline adjustment for a font reference. This is primarily used when defining font descriptors for a cascade list to keep the baseline of all fonts even. The value associated with

this is a float represented as a number.

7.3.17 `kCTFontCascadeListAttribute` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key to specify or obtain the cascade list used for a font reference. The cascade list is a array containing `CTFontDescriptorMBS` elements. If unspecified, the global cascade list is used.

7.3.18 `kCTFontCharacterSetAttribute` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key to specify or obtain the Unicode character coverage set for a font reference. The value for this key is a `CharacterSetMBS` object. If specified, this attribute can be used to restrict the font to a subset of its actual character set. If unspecified, this attribute is ignored and the actual character set is used.

7.3.19 `kCTFontDescriptorMatchingCurrentAssetSize` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for progressParameter dictionary.

Notes: A number. Total downloaded byte size. Valid during Downloading state.

7.3.20 `kCTFontDescriptorMatchingDescriptors` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for progressParameter dictionary.

Notes: Array of descriptors to be queried.

Valid while downloading or when state is `kCTFontDescriptorMatchingWillBeginQuerying`.

7.3.21 `kCTFontDescriptorMatchingError` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for progressParameter dictionary.

Notes: A `CFErrorMBS`; Valid when state `kCTFontDescriptorMatchingDidFailWithError`.

7.3.22 kCTFontDescriptorMatchingPercentage as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for progressParameter dictionary.

Notes: A Number; Download progress in 0 - 100. Valid during Downloading state.

7.3.23 kCTFontDescriptorMatchingResult as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for progressParameter dictionary.

Notes: Array of matched font descriptors.

Valid when state is kCTFontDescriptorMatchingDidMatch or CTFontDescriptorMatchingEnd.

7.3.24 kCTFontDescriptorMatchingSourceDescriptor as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for progressParameter dictionary.

Notes: A CTFontDescriptorMBS; The current font descriptor.

Valid when state is kCTFontDescriptorMatchingDidMatch.

7.3.25 kCTFontDescriptorMatchingTotalAssetSize as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for progressParameter dictionary.

Notes: A Number; Total byte size to download.

Always valid, but may be Zero when information is not available.

7.3.26 kCTFontDescriptorMatchingTotalDownloadedSize as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys for progressParameter dictionary.

Notes: A Number; Total downloaded byte size. Valid during Downloading state.

7.3.27 `kCTFontDisplayNameAttribute` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key for accessing the name used to display the font. Most commonly this is the full name. The value associated with this key is a string. If the value is unspecified, it defaults to Helvetica, and if that font is unavailable, it falls back to the global font cascade list.

7.3.28 `kCTFontDownloadableAttribute` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: The font downloadable state.

The value associated with this key is a Boolean. If it is true, CoreText attempts to download a font if necessary when matching a descriptor.

7.3.29 `kCTFontEnabledAttribute` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key to obtain the font enabled state. The returned value is an integer represented as a number representing a Boolean value. Unregistered font descriptors return nil, which is equivalent to false. Available in OS X v10.6 and later.

7.3.30 `kCTFontFamilyNameAttribute` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key for accessing the font family name from the font descriptor. The value associated with this key is a string.

7.3.31 `kCTFontFeaturesAttribute` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key to specify or obtain the font features for a font reference. The value associated with this key is an array containing font feature dictionaries. This feature list contains the feature information from the 'feat'

table of the font. For more information, see Features.
Available in OS X v10.5 and later.

7.3.32 kCTFontFeatureSettingsAttribute as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key to specify or obtain the font features settings for a font reference. The value associated with this key is a CFArrayRef object containing font feature-setting dictionaries. A feature-setting dictionary contains a tuple of a kCTFontFeatureTypeIdentifierKey key-value pair and a kCTFontFeatureSelectorIdentifierKey key-value pair. Each setting dictionary indicates which setting should be turned on. In the case of duplicate or conflicting setting, the last setting in the list takes precedence. It is the caller's responsibility to handle exclusive and nonexclusive settings as necessary.

Available in OS X v10.5 and later.

7.3.33 kCTFontFixedAdvanceAttribute as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key to specify a fixed advance to be used for a font reference. If present and specified, this attribute is used to specify a constant advance to override any font values. The value associated with this key is a float represented as a number.

Available in OS X v10.5 and later.

7.3.34 kCTFontFormatAttribute as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key to specify or obtain the recognized format of the font. The value associated with this key is an integer represented as a number containing one of the constants in "Font Format Constants."

Available in OS X v10.6 and later.

7.3.35 kCTFontLanguagesAttribute as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key to specify or obtain a list of covered languages for a font reference. The value for this key is

an array containing string elements. If specified, this attribute restricts the search to matching fonts that support the specified languages. The language identifier string should conform to the RFC 3066bis standard. If unspecified, this attribute is ignored.

Available in OS X v10.5 and later.

7.3.36 `kCTFontMacintoshEncodingsAttribute` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key to specify or obtain the Macintosh encodings for a font reference. The value associated with this key is a number containing a bit field of the Macintosh encodings. This attribute is provided for legacy compatibility.

Available in OS X v10.5 and later.

7.3.37 `kCTFontMatrixAttribute` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key to specify the font transformation matrix when creating a font. If unspecified it defaults to the unit matrix. The value for this key is a Memoryblock object containing a `CGAffineTransform`.

7.3.38 `kCTFontNameAttribute` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key for accessing the PostScript name from the font descriptor. The value associated with this key is a string. If the value is unspecified, it defaults to Helvetica, and if that font is unavailable, it falls back to the global font cascade list.

7.3.39 `kCTFontOrientationAttribute` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key to specify a particular orientation for the glyphs of the font. The value associated with this key is an integer represented as a number containing one of the constants in "Font Orientation Constants." If you want to receive vertical metrics from a font for vertical rendering, specify `kCTFontVerticalOrientation`.

If unspecified, the font uses its native orientation.

7.3.40 kCTFontPriorityAttribute as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key to specify or obtain the font priority used by font descriptors when resolving duplicates and sorting match results. The value associated with this key is an integer represented as a CFNumberRef object containing one of the values enumerated in "Font Priority Constants." The higher the value, the higher the priority of the font. Only registered fonts have a priority. Unregistered font descriptors return nil. Available in OS X v10.6 and later.

7.3.41 kCTFontRegistrationScopeAttribute as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key to specify or obtain the font descriptor's registration scope. The value associated with this key is an integer represented as a number containing one of the CTFontManagerScope enumerated values. A value of nil can be returned for font descriptors that are not registered. Available in OS X v10.6 and later.

7.3.42 kCTFontSizeAttribute as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key to obtain or specify the font point size. Creating a font with this unspecified will default to a point size of 12.0. The value for this key is represented as a number.

7.3.43 kCTFontStyleNameAttribute as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key for accessing the style name of the font. This name represents the designer's description of the font's style. The value associated with this key is a string.

7.3.44 `kCTFontTraitsAttribute` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key for accessing the dictionary of font traits for stylistic information. See "Font Traits" for the list of font traits. The value associated with this key is a dictionary.

7.3.45 `kCTFontURLAttribute` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key for accessing the font URL from the font descriptor. The value associated with this key is an URL string.

Available in OS X v10.6 and later.

7.3.46 `kCTFontVariationAttribute` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants keys for accessing font attributes from a font descriptor.

Notes: Key to obtain the font variation dictionary instance as a dictionary object. If specified in a font descriptor, fonts with the specified axes are primary match candidates; if no such fonts exist, this attribute is ignored.

7.3.47 `LocalizedAttributeValue(key as string, byref lang as string)` as Variant

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a localized value for the requested attribute, if available.

Notes: key: The requested font attribute.

lang: On output, contains a reference to the matched language. The language identifier will conform to the RFC 3066bis standard.

Returns a localized attribute value based on the global language list.

This function passes back the matched language in language. If localization is not possible for the attribute, the behavior matches the value returned from `AttributeValue`. Generally, localization of attributes is applicable to name attributes of only a normalized font descriptor.

7.3.48 MatchingFontDescriptor(mandatoryAttributes() as String) as CTFontDescriptorMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the single preferred matching font descriptor based on the original descriptor and system precedence.

Notes: mandatoryAttributes: A set of attribute keys which must be identically matched in any returned font descriptors.

Returns a normalized font descriptor matching the attributes present in descriptor.

The original descriptor may be returned in normalized form. In the context of font descriptors, normalized infers that the input values were matched up with actual existing fonts, and the descriptors for those existing fonts are the returned normalized descriptors.

7.3.49 MatchingFontDescriptors(mandatoryAttributes() as String) as CTFontDescriptorMBS()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns an array of normalized font descriptors matching the provided descriptor.

Notes: mandatoryAttributes: A set of attribute keys that must be identically matched in any returned font descriptors.

Returns an array of normalized font descriptors matching the attributes present in descriptor.

If descriptor itself is normalized, then the array will contain only one item: the original descriptor. In the context of font descriptors, normalized infers that the input values were matched up with actual existing fonts, and the descriptors for those existing fonts are the returned normalized descriptors.

7.3.50 Properties

7.3.51 DisplayName as String

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The display name.

Example:

```
Var f as CTFontMBS = CTFontMBS.CreateWithName("Times")
Var d as CTFontDescriptorMBS = f.FontDescriptor
```

MsgBox d.DisplayName

Notes: (Read only property)

7.3.52 FamilyName as String

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The family name.

Example:

```
Var f as CTFontMBS = CTFontMBS.CreateWithName("Times")
Var d as CTFontDescriptorMBS = f.FontDescriptor
MsgBox d.FamilyName
```

Notes: (Read only property)

7.3.53 File as FolderItem

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The font file.

Example:

```
Var f as CTFontMBS = CTFontMBS.CreateWithName("Times")
Var d as CTFontDescriptorMBS = f.FontDescriptor
MsgBox d.file.NativePath
```

Notes: (Read only property)

7.3.54 FontSize as Double

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The font size.

Example:

```
Var f as CTFontMBS = CTFontMBS.CreateWithName("Times")
Var d as CTFontDescriptorMBS = f.FontDescriptor
MsgBox str(d.FontSize)
```

Notes: Can be zero if unknown.

(Read only property)

7.3.55 Name as String

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The PostScript name.

Example:

```
Var f as CTFontMBS = CTFontMBS.CreateWithName("Times")
Var d as CTFontDescriptorMBS = f.FontDescriptor
MsgBox d.Name
```

Notes: When matching, this is treated more generically: the system first tries to find fonts with this PostScript name. If none is found, the system tries to find fonts with this family name, and, finally, if still nothing, tries to find fonts with this display name.

(Read only property)

7.3.56 StyleName as String

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The style name.

Example:

```
Var f as CTFontMBS = CTFontMBS.CreateWithName("Times")
Var d as CTFontDescriptorMBS = f.FontDescriptor
MsgBox d.StyleName
```

Notes: This name represents the designer's description of the font's style.

Can be empty if unknown.

(Read only property)

7.3.57 URL as String

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The font URL.

Example:

```
Var f as CTFontMBS = CTFontMBS.CreateWithName("Times")
Var d as CTFontDescriptorMBS = f.FontDescriptor
```

MsgBox d.URL

Notes: (Read only property)

7.3.58 Constants

Font Matching State

Constant	Value	Description
kCTFontDescriptorMatchingDidBegin	0	called once at the beginning.
kCTFontDescriptorMatchingDidFailWithError	8	called when an error occurred. (may be called multiple times)
kCTFontDescriptorMatchingDidFinish	1	called once at the end.
kCTFontDescriptorMatchingDidFinishDownloading	6	Finished downloading a descriptor.
kCTFontDescriptorMatchingDidMatch	7	called when font descriptor is matched.
kCTFontDescriptorMatchingDownloading	5	Downloading a descriptor.
kCTFontDescriptorMatchingStalled	3	called when stalled. (e.g. while waiting for server response)
kCTFontDescriptorMatchingWillBeginDownloading	4	Starts downloading a descriptor.
kCTFontDescriptorMatchingWillBeginQuerying	2	Downloading part may be skipped if all the assets are already downloaded. Called once before talking to the server. Skipped if not needed.

Formats

Constant	Value	Description
kCTFontFormatBitmap	5	The font is a bitmap only format.
kCTFontFormatOpenTypePostScript	1	The font is an OpenType format containing PostScript data.
kCTFontFormatOpenTypeTrueType	2	The font is an OpenType format containing TrueType data.
kCTFontFormatPostScript	4	The font is a recognized PostScript format.
kCTFontFormatTrueType	3	The font is a recognized TrueType format.
kCTFontFormatUnrecognized	0	The font is not a recognized format.

Orientations

Constant	Value	Description
kCTFontOrientationDefault	0	Default
kCTFontOrientationHorizontal	1	Horizontal
kCTFontOrientationVertical	2	Vertical

Priorities

Constant	Value	Description
kCTFontPriorityComputer	30000	Priority of computer local fonts (located in /Library/Fonts).
kCTFontPriorityDynamic	50000	Priority of fonts registered dynamically, not located in a standard location (either kCTFontManagerScopeUser, or kCTFontManagerScopeSession).
kCTFontPriorityNetwork	20000	Priority of network fonts (located in /Network/Library/Fonts).
kCTFontPriorityProcess	60000	Priority of fonts registered for the process (kCTFontManagerScopeProcess).
kCTFontPrioritySystem	10000	Priority of system fonts (located in /System/Library/Fonts).
kCTFontPriorityUser	40000	Priority of local fonts (located in user's Library/Fonts).

7.4 class CTFontMBS

7.4.1 class CTFontMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The CTFont opaque type represents a Core Text font object.

Example:

```
Var c as CTFontMBS = CTFontMBS.CreateWithName("times", 10)
MsgBox c.FullName
```

Notes: Font objects represent fonts to an application, providing access to characteristics of the font, such as point size, transform matrix, and other attributes. Fonts provide assistance in laying out glyphs relative to one another and are used to establish the current font when drawing in a graphics context.

Subclass of the CFontMBS class.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [MBS Xojo Plugins, version 22.5pr1](#)
- [MBS Xojo Plugins, version 21.6pr1](#)
- [Removing 32-bit Carbon GUI classes](#)
- [MBS Xojo / Real Studio Plugins, version 15.0pr11](#)
- [CoreText Completed](#)

7.4.2 Methods

7.4.3 AdvancesForGlyphs(orientation as Integer, glyphs() as Integer) as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Calculates the advances for an array of glyphs and returns the summed advance.

Notes: font: The font reference.

orientation

The intended drawing orientation of the glyphs. Used to determine which glyph metrics to return.

glyphs: An array of count number of glyphs.

advances: An array of count number of CGSizeMBS objects to receive the computed glyph advances. Optional.

Returns the summed glyph advance of an array of glyphs.

Individual glyph advances are passed back via the advances parameter. These are the ideal metrics for each glyph scaled and transformed in font space.

See also:

- 7.4.4 `AdvancesForGlyphs(orientation as Integer, glyphs() as Integer, boundingRects() as CGSizeMBS) as Double` 525

7.4.4 `AdvancesForGlyphs(orientation as Integer, glyphs() as Integer, boundingRects() as CGSizeMBS) as Double`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Calculates the advances for an array of glyphs and returns the summed advance.

Notes: font: The font reference.

orientation

The intended drawing orientation of the glyphs. Used to determined which glyph metrics to return.

glyphs: An array of count number of glyphs.

advances: An array of count number of CGSizeMBS objects to receive the computed glyph advances. Optional.

Returns the summed glyph advance of an array of glyphs.

Individual glyph advances are passed back via the advances parameter. These are the ideal metrics for each glyph scaled and transformed in font space.

See also:

- 7.4.3 `AdvancesForGlyphs(orientation as Integer, glyphs() as Integer) as Double` 524

7.4.5 `AttributeValue(key as string) as Variant`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the value associated with an arbitrary attribute of the given font.

Notes: key: The requested attribute.

Returns attribute value or nil if the requested attribute is not present.

Available in OS X v10.5 and later.

7.4.6 `Available as boolean`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Whether this class is available.

Notes: Returns true in OS X v10.5 and later.

7.4.7 AvailableTables(options as Integer) as String()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns an array of font table tags.

Notes: options: The font table options.

Returns an array of Font Table Tag Constants values for the given font and the supplied options. Available in OS X v10.5 and later.

7.4.8 BoundingRectsForGlyphs(orientation as Integer, glyphs() as Integer) as CGRectMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Calculates the bounding rects for an array of glyphs and returns the overall bounding rectangle for the glyph run.

Notes: font: The font reference.

orientation: The intended drawing orientation of the glyphs. Used to determined which glyph metrics to return.

glyphs: An array of count number of glyphs.

boundingRects: Optional. On output, the computed glyph rectangles in an array of count number of CGRect objects.

Returns the overall bounding rectangle for an array or run of glyphs. Returns CGRectNull on error.

The bounding rectangles of the individual glyphs are returned through the boundingRects parameter. These are the design metrics from the font transformed in font space.

See also:

- 7.4.9 BoundingRectsForGlyphs(orientation as Integer, glyphs() as Integer, boundingRects() as CGRectMBS) as CGRectMBS 526

7.4.9 BoundingRectsForGlyphs(orientation as Integer, glyphs() as Integer, boundingRects() as CGRectMBS) as CGRectMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Calculates the bounding rects for an array of glyphs and returns the overall bounding rectangle for the glyph run.

Notes: font: The font reference.

orientation: The intended drawing orientation of the glyphs. Used to determined which glyph metrics to return.

glyphs: An array of count number of glyphs.

boundingRects: Optional. On output, the computed glyph rectangles in an array of count number of CGRect objects.

Returns the overall bounding rectangle for an array or run of glyphs. Returns CGRectNull on error.

The bounding rectangles of the individual glyphs are returned through the boundingRects parameter. These are the design metrics from the font transformed in font space.

See also:

- 7.4.8 BoundingRectsForGlyphs(orientation as Integer, glyphs() as Integer) as CGRectMBS 526

7.4.10 Constructor

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The private constructor.

7.4.11 CreateCopyWithAttributes(size as Double, Matrix as CGAffineTransformMBS, fontAttributes as CTFontDescriptorMBS) as CTFontMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a new font with additional attributes based on the original font.

Example:

```
// find PostScript names for a font
const nameOfAFont = "Arial"
```

```
Var FontName as string
Var FontFamilyName as string
Var FontPostscriptName as string
Var FontPostscriptNameBold as string
Var FontPostscriptNameItalic as string
```

```
Var f as CTFontMBS = CTFontMBS.CreateWithName(nameOfAFont, 13)
```

```
if f <> nil then
FontName = f.FullName
FontFamilyName = f.FamilyName
```

```
FontPostscriptName = f.PostScriptName
```

```
Var d as CTFontDescriptorMBS = f.FontDescriptor
```

```
Var db as CTFontDescriptorMBS = CTFontDescriptorMBS.CreateCopyWithSymbolicTraits(d, CoreTextMBS.kCTFontTraitBold, CoreTextMBS.kCTFontTraitBold)
```

```
Var di as CTFontDescriptorMBS = CTFontDescriptorMBS.CreateCopyWithSymbolicTraits(d, CoreTextMBS.kCTFontTraitItalic, CoreTextMBS.kCTFontTraitItalic)
```

```
Var fb as CTFontMBS = f.CreateCopyWithAttributes(f.Size, nil, db)
```

```
Var fi as CTFontMBS = f.CreateCopyWithAttributes(f.Size, nil, di)
```

```
FontPostscriptNameBold = fb.PostScriptName
```

```
FontPostscriptNameItalic = fi.PostScriptName
```

```
end if
```

```
Break // read names in debugger
```

Notes: size: The point size for the font reference. If 0.0 is specified, the original font's size is preserved.
matrix: The transformation matrix for the font. In most cases, set this parameter to be nil. If nil is specified, the original font's matrix is preserved.
fontAttributes: A font descriptor containing additional attributes that the new font should contain.

Returns a new font reference converted from the original with the specified attributes.

This function provides a mechanism to change attributes quickly on a given font reference in response to user actions. For instance, the size can be changed in response to a user manipulating a size slider. Available in OS X v10.5 and later.

7.4.12 CreateForString(text as string, location as Integer, length as Integer) as CTFontMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a new font reference that can best map the given string range based on the current font.

Notes: string: A unicode string containing characters that cannot be encoded by the current font.

location and length: The range of the string that needs to be mapped.

Returns the best substitute font from the cascade list of the current font that can encode the specified string range. If the current font is capable of encoding the string range, then it is retained and returned.

7.4.13 CreatePathForGlyph(glyph as Integer, transform as CGAffineTransformMBS) as CGPathMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates a path for the specified glyph.

Notes: glyph: The glyph.

transform: An affine transform applied to the path. Can be nil. If nil, CGAffineTransformIdentity is used.

Returns a CGPath object containing the glyph outlines, nil on error. Must be released by caller.

Creates a path from the outlines of the glyph for the specified font. The path reflects the font point size, matrix, and transform parameter, applied in that order. The transform parameter is most commonly be used to provide a translation to the desired glyph origin.

7.4.14 CreateUIFontForLanguage(Type as Integer, size as Double = 0.0, language as string = "") as CTFontMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the special user-interface font for the given language and user-interface type.

Notes: Type: A constant specifying the intended user-interface use for the requested font reference. See Enumerations for possible values.

size: The point size for the font reference. If 0.0 is specified, the default size for the requested user-interface type is used.

language: Language specifier string to select a font for a particular localization. If "" is specified, the current system language is used. The format of the language identifier should conform to the RFC 3066bis standard.

Returns the correct font for various user-interface uses.

The only required parameter is the Type selector; the other parameters have default values.

7.4.15 CreateWithFamily(size as Double, Matrix as CGAffineTransformMBS, family as string) as CTFontMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a new font in the specified family based on the traits of the original font.

Notes: size: The point size for the font reference. If 0.0 is specified, the original font's size is preserved.

matrix: The transformation matrix for the font. In most cases, set this parameter to be nil. If nil is specified, the original font's matrix is preserved.

family: The name of the desired family.

Returns a new font reference with the original traits in the given family, or nil if none is found in the system. Available in OS X v10.5 and later.

7.4.16 **CreateWithFontDescriptor(descriptor as CTFontDescriptorMBS, size as Double = 0.0, matrix as CGAffineTransformMBS = nil, options as Integer = 0) as CTFontMBS**

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a new font reference that best matches the given font descriptor.

Notes: descriptor: A font descriptor containing attributes that specify the requested font.

size: The point size for the font reference. If 0.0 is specified, the default font size of 12.0 is used. This parameter is optional.

matrix: The transformation matrix for the font. In most cases, set this parameter to be nil. If nil is specified, the identity matrix is used. This parameter is optional.

Returns a CTFontMBS that best matches the attributes provided with the font descriptor.

The size and matrix parameters override any specified in the font descriptor unless they are unspecified (0.0 for size and NULL for matrix). A best match font is always returned, and default values are used for any unspecified parameters.

7.4.17 **CreateWithGraphicsFont(graphicsFont as CGFontMBS, size as Double = 0.0, matrix as CGAffineTransformMBS = nil, attributeValues as CTFontDescriptorMBS = nil) as CTFontMBS**

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates a new font reference from an existing Core Graphics font reference.

Notes: graphicsFont: A valid Core Graphics font reference.

size: The point size for the font reference. If 0.0 is specified the default font size of 12.0 is used.

matrix: The transformation matrix for the font. In most cases, set this parameter to be nil. If nil, the identity matrix is used. Optional.

attributes: Additional attributes that should be matched. Optional.

Returns a new font reference for an existing CGFontRef object with the specified size, matrix, and additional attributes.

7.4.18 CreateWithName(name as string, size as Double = 0.0, matrix as CGAffineTransformMBS = nil, options as Integer = 0) as CTFontMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a new font reference for the given name.

Example:

```
Var c as CTFontMBS = CTFontMBS.CreateWithName("Times", 12)
MsgBox c.FullName
```

Notes: name: The font name for which you wish to create a new font reference. A valid PostScript name is preferred, although other font name types are matched in a fallback manner.

size: The point size for the font reference. If 0.0 is specified, the default font size of 12.0 is used. This parameter is optional.

matrix: The transformation matrix for the font. In most cases, set this parameter to be nil. If nil is specified, the identity matrix is used. This parameter is optional.

Returns a CTFontRef that best matches the name provided with size and matrix attributes.

The name parameter is the only required parameter, and default values are used for unspecified parameters (0.0 for size and nil for matrix). If all parameters cannot be matched identically, a best match is found.

7.4.19 CreateWithPlatformFont(ATSTFontHandle as Integer, size as Double = 0.0, matrix as CGAffineTransformMBS = nil, attributeValues as CTFontDescriptorMBS = nil) as CTFontMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Creates a new font reference from an ATS font reference.

Notes: ATSTFontHandle: A valid ATSTFontRef object.

size: The point size for the font reference. If 0.0 is specified the default font size of 12.0 is used.

matrix: The transformation matrix for the font. In most cases, set this parameter to be nil. If nil, the identity matrix is used. Optional.

attributes: A CTFontDescriptorMBS containing additional attributes that should be matched. Optional.

Returns a new font reference for an ATSTFontRef with the specified size, matrix, and additional attributes.

7.4.20 CreateWithSymbolicTraits(size as Double, Matrix as CGAffineTransformMBS, symTraitValue as Integer, symTraitMask as Integer) as CTFontMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a new font in the same font family as the original with the specified symbolic traits.

Example:

```
Var helveticaBold as CTFontMBS = CTFontMBS.CreateWithName("Helvetica-Bold", 12)
MsgBox helveticaBold.FullName

// now create similar fonts with

Var Trait as Integer = CoreTextMBS.kCTFontTraitItalic
Var TraitMask as Integer = CoreTextMBS.kCTFontTraitItalic+CoreTextMBS.kCTFontTraitBold

Var helveticaItalic as CTFontMBS = helveticaBold.CreateWithSymbolicTraits(12, nil, Trait, TraitMask)
MsgBox helveticaItalic.FullName
```

Notes: size: The point size for the font reference. If 0.0 is specified, the original font's size is preserved.
matrix: The transformation matrix for the font. In most cases, set this parameter to be nil. If nil is specified, the original font's matrix is preserved.

symTraitValue: The value of the symbolic traits.

symTraitMask: The mask bits of the symbolic traits.

Returns a new font reference in the same family with the given symbolic traits. or nil if none is found in the system.

Available in OS X v10.5 and later.

7.4.21 DefaultCascadeListForLanguages(languagePrefList() as string) as String()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Return an ordered list of CTFontDescriptorMBS's for font fallback derived from the system default fallback region according to the given language preferences.

Notes: The style of the given is also matched as well as the weight and width of the font is not one of the system UI font, otherwise the UI font fallback is applied.

languagePrefList: The language preference list - ordered array of CFStringRef's of ISO language codes.

The ordered list of fallback fonts - ordered array of CTFontDescriptors.

7.4.22 Draw(glyphs() as Integer, positions() as CGPointMBS, context as CGContextMBS)

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Renders the given glyphs of a font at the specified positions in the supplied graphics context.

Notes: font: The font with glyphs to render. If the font has a size or matrix attribute, context is set with these values.

glyphs: The glyphs to be rendered. The glyphs should be the result of proper Unicode text layout operations (such as with CTLine). Functions such as CTFontMBS.GetGlyphsForCharacters do not perform any Unicode text layout.

positions: The positions (origins) for each glyph in glyphs. The positions are in user space. The number of positions passed in must match the number of glyphs (in glyphs).

context: The graphics context used to render the glyphs.

This function modifies graphics state including font, text size, and text matrix if these attributes are specified in font. These attributes are not restored.

Available in OS X v10.7 and later.

7.4.23 Features as Dictionary()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns an array of font features.

Notes: Returns an array of font feature dictionaries for the font reference.

7.4.24 FeatureSettings as Dictionary()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns an array of font feature-setting tuples.

Notes: Returns a normalized array of font feature-setting dictionaries. The array contains only the non-default settings that should be applied to the font, or nil if the default settings should be used.

A feature-setting dictionary is a tuple of a kCTFontFeatureTypeIdentifierKey key-value pair and a kCTFontFeatureSelectorIdentifierKey key-value pair. Each setting dictionary indicates which setting is enabled. It is the caller's responsibility to handle exclusive and nonexclusive settings as necessary.

The feature settings are verified against those that the font supports and any that do not apply are removed. Further, feature settings that represent a default setting for the font are also removed.

7.4.25 GlyphsForCharacters(characters() as Integer) as Integer()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Provides basic Unicode encoding for the given font, returning by reference an array of CGGlyph values corresponding to a given array of Unicode characters for the given font.

Notes: If a glyph could not be encoded, a value of 0 is passed back at the corresponding index in the glyphs array and the function returns False. It is the responsibility of the caller to handle the Unicode properties of the input characters.

7.4.26 GlyphWithName(name as string) as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the CGGlyph value for the specified glyph name in the given font.

Notes: Name: The glyph name as a CFString object.

Returns the glyph value for the named glyph as a CGGlyph object, or if the glyph name is not recognized, the .notdef glyph index value.

The returned CGGlyph object can be used with any of the subsequent glyph data accessors or directly with Core Graphics.

7.4.27 GraphicsFont(byref fontAttributes as CTFontDescriptorMBS) as CGFontMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a Core Graphics font reference and attributes.

Notes: attributes: On output, points to a font descriptor containing additional attributes from the font.

Returns a CGFontMBS object for the given font reference.

7.4.28 kCTBaselineClassHanging as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Key to reference the Hanging baseline class.

Notes: This key can be used with a baseline info dictionary to offset to the Hanging baseline as a float. It can also be used as the value for kCTBaselineClassAttributeName.

7.4.29 kCTBaselineClassIdeographicCentered as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Key to reference the Ideographic Centered baseline class.

Notes: This key can be used with a baseline info dictionary to offset to the Ideographic Centered baseline as a float. It can also be used as the value for kCTBaselineClassAttributeName.

7.4.30 kCTBaselineClassIdeographicHigh as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Key to reference the Ideographic High baseline class.

Notes: This key can be used with a baseline info dictionary to offset to the Ideographic High baseline as a float. It can also be used as the value for kCTBaselineClassAttributeName.

7.4.31 kCTBaselineClassIdeographicLow as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Key to reference the Ideographic Low baseline class.

Notes: This key can be used with a baseline info dictionary to offset to the Ideographic Low baseline as a float. It can also be used as the value for kCTBaselineClassAttributeName.

7.4.32 kCTBaselineClassMath as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Key to reference the Math baseline class.

Notes: This key can be used with a baseline info dictionary to offset to the Math baseline as a float. It can also be used as the value for kCTBaselineClassAttributeName.

7.4.33 kCTBaselineClassRoman as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Key to reference the Roman baseline class.

Notes: This key can be used with a baseline info dictionary to offset to the Roman baseline as a float. It can also be used as the value for kCTBaselineClassAttributeName.

7.4.34 kCTBaselineOriginalFont as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Use the original font for setting the reference baseline.

Notes: This constant can be used as the value for kCTBaselineReferenceFont to specify that the original font should be used for the reference baseline.

7.4.35 kCTBaselineReferenceFont as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Key to reference a font for the reference baseline.

Notes: This key can be used to specify a font for the reference baseline. The value is a CTFontMBS or the kCTBaselineOriginalFont constant.

7.4.36 kCTFontCopyrightNameKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the copyright name.

7.4.37 kCTFontDescriptionNameKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the description name.

7.4.38 kCTFontDesignerNameKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the designer name.

7.4.39 kCTFontDesignerURLNameKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the vendor URL name.

7.4.40 kCTFontFamilyNameKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the family name.

7.4.41 kCTFontFeatureSelectorDefaultKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys to font feature dictionary values.

Notes: Key to be used with a selector dictionary to get the default indicator for the selector. This value is a boolean, which if present and true, indicates that this selector is the default setting for the current feature type.

7.4.42 kCTFontFeatureSelectorIdentifierKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys to font feature dictionary values.

Notes: Key to be used with a selector dictionary corresponding to a feature type to obtain the selector identifier value as a number.

7.4.43 kCTFontFeatureSelectorNameKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys to font feature dictionary values.

Notes: Key to be used with a selector dictionary to get the localized name string for the selector as a string.

7.4.44 kCTFontFeatureSelectorSettingKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys to font feature dictionary values.

Notes: Key to be used with a selector dictionary to get or specify the current setting for the selector. This

value is a Boolean to indicate whether this selector is on or off. If this key is not present, the default setting is used.

7.4.45 `kCTFontFeatureTypeExclusiveKey` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys to font feature dictionary values.

Notes: Key to get the font feature exclusive setting of the feature as a Boolean. The value associated with this key indicates whether the feature selectors associated with this type should be mutually exclusive.

7.4.46 `kCTFontFeatureTypeIdentifierKey` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys to font feature dictionary values.

Notes: Key to get the font feature type value as a number.

7.4.47 `kCTFontFeatureTypeNameKey` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys to font feature dictionary values.

Notes: Key to get the localized font feature type name as a string.

7.4.48 `kCTFontFeatureTypeSelectorsKey` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys to font feature dictionary values.

Notes: Key to get the the array of font feature selectors as an array. This is an array of selector dictionaries that contain the values for the font feature selector keys listed in this group.

7.4.49 `kCTFontFullNameKey` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the full name.

7.4.50 kCTFontLicenseNameKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the license name.

7.4.51 kCTFontLicenseURLNameKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the license URL name.

7.4.52 kCTFontManufacturerNameKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the manufacturer name.

7.4.53 kCTFontPostScriptCIDNameKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the PostScript character identifier (CID) font name.

7.4.54 kCTFontPostScriptNameKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the PostScript name.

7.4.55 kCTFontSampleTextNameKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the sample text name string.

7.4.56 `kCTFontStyleNameKey` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the style name.

7.4.57 `kCTFontSubFamilyNameKey` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the subfamily name.

7.4.58 `kCTFontTrademarkNameKey` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the trademark name.

7.4.59 `kCTFontUniqueNameKey` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the unique name.

7.4.60 `kCTFontVariationAxisDefaultValueKey` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys to font variation axis dictionary.

Notes: Key to get the variation axis default value as a number.

7.4.61 kCTFontVariationAxisIdentifierKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys to font variation axis dictionary.

Notes: Key to get the variation axis identifier value as a number.

7.4.62 kCTFontVariationAxisMaximumValueKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys to font variation axis dictionary.

Notes: Key to get the variation axis maximum value as a number.

7.4.63 kCTFontVariationAxisMinimumValueKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys to font variation axis dictionary.

Notes: Key to get the variation axis minimum value as a number.

7.4.64 kCTFontVariationAxisNameKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the keys to font variation axis dictionary.

Notes: Key to get the localized variation axis name string.

7.4.65 kCTFontVendorURLNameKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the vendor URL name.

7.4.66 kCTFontVersionNameKey as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the name specifier constants provide access to the different names associated with a font.

Notes: The name specifier for the version name.

7.4.67 `LigatureCaretPositions(glyph as Integer) as Double()`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns caret positions within a glyph.

Notes: glyph: A reference to the glyph.

This function is used to obtain caret positions for a specific glyph. The return value is the maximum number of positions possible, and the function will populate the caller's positions buffer with available positions if possible. This function might not be able to produce positions if the font does not have the appropriate data, in which case it will return 0.

7.4.68 `Name(nameKey as string) as String`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a reference to the requested name of the given font.

Example:

```
Var c as CTFontMBS = CTFontMBS.CreateWithName("Times", 12)
MsgBox c.name(CTFontMBS.kCTFontFullNameKey)
```

Notes: nameKey: The name specifier. See Name Specifier Constants for possible values.

Returns The requested name for the font, or "" if the font does not have an entry for the requested name. The Unicode version of the name is preferred, otherwise the first available version is returned.
See also:

- 7.4.69 `Name(nameKey as string, byref language as string) as String`

542

7.4.69 `Name(nameKey as string, byref language as string) as String`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a reference to a localized name for the given font.

Example:

```
Var c as CTFontMBS = CTFontMBS.CreateWithName("Times", 12)
Var lang as string
MsgBox c.name(CTFontMBS.kCTFontFullNameKey, lang)+EndOfLine+lang
```

// shows name and "en" as language.

Notes: nameKey: The name specifier. See Name Specifier Constants for possible values.

language: On output, points to the language string of the returned name string. The format of the language identifier conforms to the RFC 3066bis standard.

Returns a specific localized name from the font reference or "" if the font does not have an entry for the requested name key.

The name is localized based on the user's global language preference precedence. That is, the user's language preference is a list of languages in order of precedence. So, for example, if the list had Japanese and English, in that order, then a font that did not have Japanese name strings but had English strings would return the English strings.

Available in OS X v10.5 and later.

See also:

- 7.4.68 Name(nameKey as string) as String

542

7.4.70 OpticalBoundsForGlyphs(glyphs() as Integer, boundingRects() as CGRectMBS, options as Integer = 0) as CGRectMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Calculates the optical bounding rects for an array of glyphs and returns the overall optical bounding rect for the run.

Notes: Fonts may specify the optical edges of glyphs that can be used to make the edges of lines of text line up in a more visually pleasing way. This function returns bounding rects corresponding to this information if present in a font, otherwise it returns typographic bounding rects (composed of the font's ascent and descent and a glyph's advance width).

font: The font reference.

glyphs: An array of count number of glyphs.

boundingRects: An array of count number of CGRects to receive the computed glyph rects. Can be nil, in which case only the overall bounding rect is calculated.

options: Reserved, set to zero.

This function returns the overall bounding rectangle for an array or run of glyphs. The bounding rects of the individual glyphs are returned through the boundingRects parameter. These are the design metrics from the font transformed in font space.

See also:

- 7.4.71 OpticalBoundsForGlyphs(glyphs() as Integer, options as Integer = 0) as CGRectMBS

544

7.4.71 `OpticalBoundsForGlyphs(glyphs() as Integer, options as Integer = 0) as CGRectMBS`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Calculates the optical bounding rects for an array of glyphs and returns the overall optical bounding rect for the run.

Notes: Fonts may specify the optical edges of glyphs that can be used to make the edges of lines of text line up in a more visually pleasing way. This function returns bounding rects corresponding to this information if present in a font, otherwise it returns typographic bounding rects (composed of the font's ascent and descent and a glyph's advance width).

font: The font reference.

glyphs: An array of count number of glyphs.

boundingRects: An array of count number of CGRects to receive the computed glyph rects. Can be nil, in which case only the overall bounding rect is calculated.

options: Reserved, set to zero.

This function returns the overall bounding rectangle for an array or run of glyphs. The bounding rects of the individual glyphs are returned through the boundingRects parameter. These are the design metrics from the font transformed in font space.

See also:

- 7.4.70 `OpticalBoundsForGlyphs(glyphs() as Integer, boundingRects() as CGRectMBS, options as Integer = 0) as CGRectMBS` 543

7.4.72 `PlatformFont(byref fontAttributes as CTFontDescriptorMBS) as Integer`

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Deprecated: This item is deprecated and should no longer be used. **Function:** Returns an ATS font reference and attributes.

7.4.73 `SupportedLanguages as String()`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns an array of languages supported by the font.

Example:

```
Var c as CTFontMBS = CTFontMBS.CreateWithName("Times", 12)
MsgBox Join(c.SupportedLanguages, EndOfLine)
```

7.4.74 Table(table as string, options as Integer) as Memoryblock

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a reference to the font table data.

Notes: table: The font table identifier as a Font Table Tag Constants constant. See Font Table Tag Constants for possible values.

options: The font table options.

Returns a retained reference to the font table data as a Memoryblock.

7.4.75 VariationAxes as Dictionary()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns an array of variation axes.

Notes: An array of variation axes dictionaries. Each variation axis dictionary contains the five variation axis keys listed in Font Variation Axis Dictionary Keys.

7.4.76 VerticalTranslationsForGlyphs(glyphs() as Integer) as CGSizeMBS()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Calculates the offset from the default (horizontal) origin to the vertical origin for an array of glyphs.

Notes: glyphs: An array of count number of glyphs.

Returns the computed origin offsets in an array of count number of CGSizeMBS objects.

7.4.77 Properties

7.4.78 Ascent as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the scaled font-ascent metric of the given font.

Notes: The font-ascent metric scaled according to the point size and matrix of the font reference. (Read only property)

7.4.79 BoundingBox as CGRectMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the scaled bounding box of the given font.

Notes: The design bounding box of the font, which is the rectangle defined by xMin, yMin, xMax, and yMax values for the font. Returns CGRectNull on error.

(Read only property)

7.4.80 CapHeight as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the cap-height metric of the given font.

Notes: The font cap-height metric scaled according to the point size and matrix of the font reference.

(Read only property)

7.4.81 CharacterSet as Variant

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the Unicode character set of the font.

Notes: Value is a CFCharacterSetMBS. We use variant to reduce dependency to other plugins.

The returned character set covers the nominal referenced by the font's Unicode 'cmap,Äô table.

(Read only property)

7.4.82 Descent as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the scaled font-descent metric of the given font.

Notes: The font-descent metric scaled according to the point size and matrix of the font reference.

(Read only property)

7.4.83 DisplayName as String

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the display name of the given font.

Example:

```
Var c as CTFontMBS = CTFontMBS.CreateWithName("Times", 12)
MsgBox c.DisplayName
```

Notes: (Read only property)

7.4.84 FamilyName as String

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the family name of the given font.

Example:

```
Var c as CTFontMBS = CTFontMBS.CreateWithName("Times", 12)
MsgBox c.FamilyName
```

Notes: (Read only property)

7.4.85 File as FolderItem

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The font file.

Example:

```
Var f as CTFontMBS = CTFontMBS.CreateWithName("Times")
MsgBox f.file.NativePath
```

Notes: (Read only property)

7.4.86 FontDescriptor as CTFontDescriptorMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the normalized font descriptor for the given font reference.

Notes: A normalized font descriptor for a font containing enough information to recreate this font at a later time.

(Read only property)

7.4.87 FullName as String

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the full name of the given font.

Example:

```
Var c as CTFontMBS = CTFontMBS.CreateWithName("Times", 12)
MsgBox c.FullName
```

Notes: (Read only property)

7.4.88 GlyphCount as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the number of glyphs of the given font.

Notes: (Read only property)

7.4.89 Leading as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the scaled font-leading metric of the given font.

Notes: The font-leading metric scaled according to the point size and matrix of the font reference.
(Read only property)

7.4.90 Matrix as CGAffineTransformMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the transformation matrix of the given font.

Notes: The transformation matrix for the given font reference. This is the matrix that was provided when the font was created.

(Read only property)

7.4.91 PostScriptName as String

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the PostScript name of the given font.

Example:

```
Var c as CTFontMBS = CTFontMBS.CreateWithName("Times", 12)
MsgBox c.PostScriptName
```

Notes: (Read only property)

7.4.92 Size as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the point size of the given font.

Notes: This is the point size provided when the font was created.

(Read only property)

7.4.93 SlantAngle as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the slant angle of the given font.

Notes: The transformed slant angle of the font. This is equivalent to the italic or caret angle with any skew from the transformation matrix applied.

(Read only property)

7.4.94 StringEncoding as UInt32

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the best string encoding for legacy format support.

Notes: The best string encoding for the font.

(Read only property)

7.4.95 SymbolicTraits as UInt32

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the symbolic traits of the given font.

Notes: The symbolic traits of the font. This is equivalent to the kCTFontSymbolicTrait value of the traits dictionary.

(Read only property)

7.4.96 Traits as Dictionary

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the traits dictionary of the given font.

Notes: A retained reference to the font traits dictionary. Individual traits can be accessed with the trait key constants.

(Read only property)

7.4.97 UnderlinePosition as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the scaled underline position of the given font.

Notes: The font underline-position metric scaled according to the point size and matrix of the font reference.

(Read only property)

7.4.98 UnderlineThickness as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the scaled underline-thickness metric of the given font.

Notes: The font underline-thickness metric scaled according to the point size and matrix of the font reference.

(Read only property)

7.4.99 UnitsPerEm as UInt64

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the units-per-em metric of the given font.

Notes: (Read only property)

7.4.100 URL as String

Plugin Version: 15.0, Platform: macOS, Targets: All.

Function: The font URL.

Example:

```
Var f as CTFontMBS = CTFontMBS.CreateWithName("Times")
MsgBox f.URL
```

Notes: (Read only property)

7.4.101 Variation as Dictionary

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns a variation dictionary from the font reference.

Notes: The keys for each variation correspond to the variation identifier obtained via `kCTFontVariationAxisIdentifierKey`, which represents the four-character axis code as a number.

(Read only property)

7.4.102 XHeight as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the x-height metric of the given font.

Notes: The font x-height metric scaled according to the point size and matrix of the font reference.

(Read only property)

7.4.103 Constants

Font Option Constants

Constant	Value	Description
<code>kCTFontOptionsDefault</code>	0	Default options are used. Available in OS X v10.6 and later.
<code>kCTFontOptionsPreferSystemFont</code>	4	Font matching prefers to match Apple system fonts. Available in OS X v10.6 and later.
<code>kCTFontOptionsPreventAutoActivation</code>	1	Prevents automatic font activation. Available in OS X v10.6 and later.

Font Table Tag Constants

Constant	Value	Description
kCTFontTableAcnt	"acnt"	Font table tag for accent attachment.
kCTFontTableAnkr	"ankr"	Anchor points
kCTFontTableAvar	"avar"	Font table tag for axis variation.
kCTFontTableBASE	"BASE"	Font table tag for the font baseline.
kCTFontTableBdat	"bdat"	Font table tag for bitmap data.
kCTFontTableBhed	"bhed"	Font table tag for bitmap font header.
kCTFontTableBloc	"bloc"	Font table tag for bitmap location.
kCTFontTableBsln	"bsln"	Font table tag for baseline.
kCTFontTableCFF	"CFF "	Font table tag for a PostScript font program.
kCTFontTableCmap	"cmap"	Font table tag for character-to-glyph mapping.
kCTFontTableCvar	"cvar"	Font table tag for control value variation, or CVT variation.
kCTFontTableCvt	"cvt "	Font table tag for control value table.
kCTFontTableDSIG	"DSIG"	Font table tag for a digital signature.
kCTFontTableEBDT	"EBDT"	Font table tag for an embedded bitmap.
kCTFontTableEBLC	"EBLC"	Font table tag for the embedded bitmap location.
kCTFontTableEBSC	"EBSC"	Font table tag for embedded bitmap scaling.
kCTFontTableFdsc	"fdsc"	Font table tag for font descriptor.
kCTFontTableFeat	"feat"	Font table tag for layout feature.
kCTFontTableFmtx	"fmtx"	Font table tag for font metrics.
kCTFontTableFpgm	"fpgm"	Font table tag for font program.
kCTFontTableFvar	"fvar"	Font table tag for font variation.
kCTFontTableGasp	"gasp"	Font table tag for grid-fitting/scan-conversion.
kCTFontTableGDEF	"GDEF"	Font table tag for glyph definition.
kCTFontTableGlyf	"glyf"	Font table tag for glyph data.
kCTFontTableGPOS	"GPOS"	Font table tag for glyph positioning.
kCTFontTableGSUB	"GSUB"	Font table tag for glyph substitution.
kCTFontTableGvar	"gvar"	Font table tag for glyph variation.
kCTFontTableHdmx	"hdmx"	Font table tag for horizontal device metrics.
kCTFontTableHead	"head"	Font table tag for font header.
kCTFontTableHhea	"hhea"	Font table tag for horizontal header.
kCTFontTableHmtx	"hmtx"	Font table tag for horizontal metrics.
kCTFontTableHsty	"hsty"	Font table tag for horizontal style.
kCTFontTableJSTF	"JSTF"	Font table tag for justification.
kCTFontTableJust	"just"	Font table tag for justification.
kCTFontTableKern	"kern"	Font table tag for kerning.
kCTFontTableKerx	"kerx"	Font table tag for extended kerning.
kCTFontTableLcar	"lcar"	Font table tag for ligature caret.
kCTFontTableLoca	"loca"	Font table tag for index to location.
kCTFontTableLtag	"ltag"	Language tags
kCTFontTableLTSH	"LTSH"	Font table tag for linear threshold.
kCTFontTableMaxp	"maxp"	Font table tag for maximum profile.
kCTFontTableMort	"mort"	Font table tag for morph.
kCTFontTableMorx	"morx"	Font table tag for extended morph.
kCTFontTableName	"name"	Font table tag for naming table.
kCTFontTableOpbd	"opbd"	Font table tag for optical bounds.
kCTFontTableOS2	"OS/2"	Font table tag for OS/2 and Windows-specific metrics.
kCTFontTablePCLT	"PCLT"	Font table tag for PCL 5 data.
kCTFontTablePost	"post"	Font table tag for PostScript information.
kCTFontTablePrep	"prep"	Font table tag for control value program, 'prep' table.
kCTFontTableProp	"prop"	Font table tag for properties.
kCTFontTableSbit	"sbit"	Font table tag for bitmap data. Available in OS X v10.7 and later.
kCTFontTableSbix	"sbix"	Font table tag for extended bitmap data. Available in OS X v10.7 and later.
kCTFontTableTrak	"trak"	Font table tag for tracking.
kCTFontTableVDMX	"VDMX"	Font table tag for vertical device metrics.
kCTFontTableVhea	"vhea"	Font table tag for vertical header.
kCTFontTableVmtx	"vmtx"	Font table tag for vertical metrics.

Font Table Option Constants

Constant	Value	Description
kCTFontTableOptionExcludeSynthetic	1	The font table excludes synthetic font data. Available in OS X v10.5 and later. Deprecated in OS X v10.8.
kCTFontTableOptionNoOptions	0	No font table options are specified.

User Interface Type Constants

Constant	Value	Description
kCTFontUIFontAlertHeader	18	The font used for alert headers.
kCTFontUIFontApplication	9	The default font for text documents.
kCTFontUIFontControlContent	26	The font used for contents of user-interface controls.
kCTFontUIFontEmphasizedSystem	3	The system font used for emphasis in alerts.
kCTFontUIFontEmphasizedSystemDetail	20	The system font used for emphasis in details.
kCTFontUIFontLabel	10	The font used for labels and tick marks on full-size sliders.
kCTFontUIFontMenuItem	12	The font used for menu items.
kCTFontUIFontMenuItemCmdKey	14	The font used for menu-item command-key equivalents.
kCTFontUIFontMenuItemMark	13	The font used to draw menu item marks.
kCTFontUIFontMenuItemTitle	11	The font used for menu titles.
kCTFontUIFontMessage	23	The font used for standard interface items, such as button labels, menu items, and so on.
kCTFontUIFontMiniEmphasizedSystem	7	The small system font used for emphasis.
kCTFontUIFontMiniSystem	6	The standard small system font used for informative text in alerts, headings in lists, help tags, and small controls.
kCTFontUIFontNone	-1	The user-interface font type is not specified.
kCTFontUIFontPalette	24	The font used in tool palettes.
kCTFontUIFontPushButton	16	The font used for a push button (a rounded rectangular button with a label on it).
kCTFontUIFontSmallEmphasizedSystem	5	The system font used for emphasis in alerts.
kCTFontUIFontSmallSystem	4	The standard small system font used for informative text in alerts, headings in lists, help tags, and small controls.
kCTFontUIFontSmallToolbar	22	The small font used for labels of toolbar items.
kCTFontUIFontSystem	2	The system font used for standard user-interface items such as buttons, menu items, and so on.
kCTFontUIFontSystemDetail	19	The standard system font used for details.
kCTFontUIFontToolbar	21	The font used for labels of toolbar items.
kCTFontUIFontToolTip	25	The font used for tool tips.
kCTFontUIFontUser	0	The font used by default for documents and other text under the control (that is, text whose font the user can normally change).
kCTFontUIFontUserFixedPitch	1	The font used by default for documents and other text under the control when that font is fixed-pitch. Available in OS X v10.5 and later.
kCTFontUIFontUtilityWindowTitle	17	The font used for utility window titles.
kCTFontUIFontViews	8	The view font used as the default font of text in lists and tables.
kCTFontUIFontWindowTitle	15	The font used for utility window titles.

7.5 class CTFrameMBS

7.5.1 class CTFrameMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The CTFrame opaque type represents a frame containing multiple lines of text.

Example:

```

Var longText as string = "Lorem ipsum dolor sit amet..."

Var ct as new CFStringMBS(longText)
Var s as new CFAttributedStringMBS(ct, nil)

// layout master
Var framesetter as CTFramesetterMBS = CTFramesetterMBS.CreateWithAttributedString(s)

// a column
Var ColumnPath as new CGMutablePathMBS
Var Rect as new CGRectMBS(0, 0, g.Width, g.Height)
ColumnPath.AddRect nil, Rect

// context
#If XojoVersion >= 2019.02 Then
Var CGContextHandle As ptr = g.Handle(Graphics.HandleTypes.CGContextRef)
Var CGContext As CGContextMBS = CGContextMBS.contextWithCGContext(CGContextHandle)
#Else
Var CGContextHandle As Integer = g.Handle(g.HandleTypeCGContextRef)
Var CGContext As CGContextMBS = CGContextMBS.contextWithCGContext(CGContextHandle)
#EndIf

CGContext.SaveGState
// reset text matrix
Var a as CGAffineTransformMBS = CGAffineTransformMBS.Identity
CGContext.TextMatrix = a

Var Frame as CTFrameMBS = framesetter.CreateFrame(0, 0, ColumnPath, nil)

// draw
Frame.Draw(CGContext)

// cleanup
CGContext.RestoreGState
CGContext.Flush

```

Notes: The frame object is the output resulting from the text-framing process performed by a framesetter object.

You can draw the entire text frame directly into the current graphic context. The frame object contains an array of line objects that can be retrieved for individual rendering or to get glyph information.

Subclass of the CXObjectMBS class.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [CoreText Completed](#)

7.5.2 Methods

7.5.3 Available as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Whether this class is available.

Notes: Returns true in OS X v10.5 and later.

7.5.4 Constructor

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The private constructor.

7.5.5 Draw(context as CGContextMBS)

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Draws an entire frame into a context.

Notes: context: The context in which to draw the frame.

If both the frame and the context are valid, the frame is drawn in the context. This call can leave the context in any state and does not flush it after the draw operation.

7.5.6 kCTFrameClippingPathsAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the attributes keys.

Notes: Specifies array of paths to clip frame.

The value must be a array containing dictionaries. Each dictionary should have a `kCTFramePathClippingPathAttributeName` key-value pair, and can have a `kCTFramePathFillRuleAttributeName` key-value pair and `kCTFramePathFillRuleAttributeName` key-value pair as optional parameters.

Available in OS X v10.7 and later.

7.5.7 `kCTFramePathClippingPathAttributeName` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the attributes keys.

Notes: Specifies clipping path. This attribute is valid only in a dictionary contained in an array specified by `kCTFrameClippingPathsAttributeName`.

The value must be a `CGPathMBS` specifying a clipping pat. See `kCTFrameClippingPathsAttributeName`.

Available in OS X v10.7 and later.

7.5.8 `kCTFramePathFillRuleAttributeName` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The key used to specify the fill rule for a frame.

Notes: he value must be a `CFNumberRef` object containing a `CTFramePathFillRule` constant. See `CTFramePathFillRule Constants` for more information. The default value is `kCTFramePathFillEvenOdd`.

Available in OS X v10.7 and later.

7.5.9 `kCTFramePathWidthAttributeName` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the attributes keys.

Notes: The key used to specify the frame width.

The value must be a number containing a value specifying the frame width. The default width value is zero.

Available in OS X v10.7 and later.

7.5.10 `kCTFrameProgressionAttributeName` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the attributes keys.

Notes: Specifies progression for a frame.

A number containing a `CTFrameProgression` constant. The default is `kCTFrameProgressionTopToBottom`.

Available in OS X v10.5 and later.

This value determines the line-stacking behavior for a frame and does not affect the appearance of the glyphs within that frame.

7.5.11 LineOrigins(location as Integer, length as Integer) as CGPointMBS()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Copies a range of line origins for a frame.

Notes: location and length: The range of line origins you wish to copy. If the length of the range is 0, then the copy operation continues from the start index of the range to the last line origin.

Returns array with CGPoints. Empty array in case of errors.

Special Considerations

In versions of OS X prior to 10.7 and versions of iOS prior to 4.2, this function may function unpredictably if the frame is not rectangular.

7.5.12 Lines as CTLineMBS()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns an array of lines stored in the frame.

7.5.13 Properties

7.5.14 FrameAttributes as Dictionary

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the frame attributes used to create the frame.

Notes: Returns a reference to a NSDictionary object containing the frame attributes that were used to create the frame, or, if the frame was created without any frame attributes, nil.

You can create a frame with an attributes dictionary to control various aspects of the framing process. These attributes are different from the ones used to create an attributed string.
(Read only property)

7.5.15 Path as CGPathMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the path used to create the frame.

Notes: (Read only property)

7.5.16 StringRangeLength as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the range of characters originally requested to fill the frame.

Notes: (Read only property)

7.5.17 StringRangeLocation as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the range of characters originally requested to fill the frame.

Notes: (Read only property)

7.5.18 VisibleStringRangeLength as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the range of characters that actually fit in the frame.

Notes: This function can be used to cascade frames, because it returns the range of characters that can be seen in the frame. The next frame would start where this frame ends.

(Read only property)

7.5.19 VisibleStringRangeLocation as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the range of characters that actually fit in the frame.

Notes: This function can be used to cascade frames, because it returns the range of characters that can be seen in the frame. The next frame would start where this frame ends.

(Read only property)

7.5.20 Constants

Fill rules

Constant	Value	Description
kCTFramePathFillEvenOdd	0	Text is filled in the area that would be painted if the path were given to CGContextMBS.EOFillPath. Available in OS X v10.7 and later.
kCTFramePathFillWindingNumber	1	Text is fill in the area that would be painted if the path were given to CGContextMBS.FillPath. Available in OS X v10.7 and later.

Frame Progression Types

Constant	Value	Description
kCTFrameProgressionLeftToRight	2	Lines are stacked left to right for vertical text. The lines of text within a frame may be stacked for either horizontal or vertical text. Values are enumerated for each stacking type supported by CTFrame. Frames created with a progression type specifying vertical text rotate lines 90 degrees counterclockwise when drawing.
kCTFrameProgressionRightToLeft	1	Lines are stacked right to left for vertical text. The lines of text within a frame may be stacked for either horizontal or vertical text. Values are enumerated for each stacking type supported by CTFrame. Frames created with a progression type specifying vertical text rotate lines 90 degrees counterclockwise when drawing.
kCTFrameProgressionTopToBottom	0	Lines are stacked top to bottom for horizontal text. The lines of text within a frame may be stacked for either horizontal or vertical text. Values are enumerated for each stacking type supported by CTFrame. Frames created with a progression type specifying vertical text rotate lines 90 degrees counterclockwise when drawing.

7.6 class CTFramesetterMBS

7.6.1 class CTFramesetterMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The CTFramesetter opaque type is used to generate text frames. That is, CTFramesetter is an object factory for CTFrame objects.

Notes: The framesetter takes an attributed string object and a shape descriptor object and calls into the typesetter to create line objects that fill that shape. The output is a frame object containing an array of lines. The frame can then draw itself directly into the current graphic context.

Subclass of the CXObjectMBS class.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [CoreText Completed](#)

7.6.2 Methods

7.6.3 Available as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Whether this class is available.

Notes: Returns true in OS X v10.5 and later.

7.6.4 Constructor

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The private constructor.

7.6.5 CreateFrame(location as Integer, length as Integer, path as CGPathMBS, frameAttributes as dictionary = nil) as CTFrameMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates an immutable frame using a framesetter.

Notes: location and length: The range, of the attributed string that was used to create the framesetter, that is to be typeset in lines fitted into the frame. If the length portion of the range is set to 0, then the framesetter continues to add lines until it runs out of text or space.

path: A CGPath object that specifies the shape of the frame. The path may be non-rectangular in versions of OS X v10.7 or later and versions of iOS 4.2 or later.

frameAttributes: Additional attributes that control the frame filling process can be specified here, or nil if there are no such attributes.

Returns a reference to a new CTFrame object if the call was successful; otherwise, nil.

This call creates a frame full of glyphs in the shape of the path provided by the path parameter. The framesetter continues to fill the frame until it either runs out of text or it finds that text no longer fits.

Special Considerations

In versions of OS X prior to 10.7 and versions of iOS prior to 4.2, this function returns NULL if the CGPath specified by the path parameter is not rectangular.

7.6.6 CreateWithAttributedString(s as CFAttributedStringMBS) as CTFramesetterMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates an immutable framesetter object from an attributed string.

Notes: s: The attributed string with which to construct the framesetter object.

Returns a reference to a CTFramesetter object if the call was successful; otherwise, NULL.

The resultant framesetter object can be used to create and fill text frames with the CreateFrame call.

7.6.7 SuggestFrameSizeWithConstraints(location as Integer, length as Integer, frameAttributes as dictionary, constraints as CGSizeMBS, byref fitRangeLocation as Integer, byref fitRangeLength as Integer) as CGSizeMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Determines the frame size needed for a string range.

Notes: location and length: The string range to which the frame size applies. The string range is a range over the string used to create the framesetter. If the length portion of the range is set to 0, then the framesetter continues to add lines until it runs out of text or space.

frameAttributes: Additional attributes that control the frame filling process, or NULL if there are no such attributes.

constraints: The width and height to which the frame size is constrained. A value of CGFLOAT_MAX for either dimension indicates that it should be treated as unconstrained.

fitRange: On return, contains the range of the string that actually fit in the constrained size.

Returns the actual dimensions for the given string range and constraints.

This function can be used to determine how much space is needed to display a string, optionally by constraining the space along either dimension.

Available in OS X v10.5 and later.

7.6.8 Properties

7.6.9 TypeSetter as CTypesetterMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the typesetter object being used by the framesetter.

Notes: Return sa reference to a CTypesetter object if the call was successful; otherwise, nil. The framesetter maintains a reference to the returned object, which should not be released by the caller.

Each framesetter uses a typesetter internally to perform line breaking and other contextual analysis based on the characters in a string; this function returns the typesetter being used by a particular framesetter in case the caller would like to perform other operations on that typesetter.

Available in OS X v10.5 and later.

(Read only property)

7.7 class CTGlyphInfoMBS

7.7.1 class CTGlyphInfoMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The CTGlyphInfo opaque type enables you to override a font's specified mapping from Unicode to the glyph ID.

Notes: Subclass of the CFontMBS class.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [CoreText Completed](#)

7.7.2 Methods

7.7.3 Available as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Whether this class is available.

Notes: Returns true in OS X v10.5 and later.

7.7.4 Constructor

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The private constructor.

7.7.5 CreateWithCharacterIdentifier(cid as Integer, collection as Integer, baseString as string) as CTGlyphInfoMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates an immutable glyph info object with a character identifier.

Notes: cid: A character identifier.

collection: A character collection identifier.

baseString: The part of the string the returned object is intended to override.

Returns a valid reference to an immutable CTGlyphInfoMBS object if glyph info creation was successful; otherwise, nil.

This function creates an immutable glyph info object for a character identifier and a character collection. Available in OS X v10.5 and later.

7.7.6 `CreateWithGlyph(glyph as Integer, font as CTFontMBS, baseString as string) as CTGlyphInfoMBS`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates an immutable glyph info object with a glyph index.

Notes: `glyph`: The index of the glyph.

`font`: The font to be associated with the returned `CTGlyphInfoMBS` object.

`baseString`: The part of the string the returned object is intended to override.

Returns a valid reference to an immutable `CTGlyphInfoMBS` object, If glyph info creation was successful; otherwise, `nil`.

This function creates an immutable glyph info object for a glyph index using a specified font. Available in OS X v10.5 and later.

7.7.7 `CreateWithGlyphName(glyphName as string, font as CTFontMBS, baseString as string) as CTGlyphInfoMBS`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates an immutable glyph info object with a glyph name.

Notes: `glyphName`: The name of the glyph.

`font`: The font to be associated with the returned `CTGlyphInfo` object.

`baseString`: The part of the string the returned object is intended to override.

Returns a valid reference to an immutable `CTGlyphInfo` object if glyph info creation was successful; otherwise, `nil`.

This function creates an immutable glyph info object for a glyph name such as copyright using a specified font.

Available in OS X v10.5 and later.

7.7.8 Properties

7.7.9 `CharacterCollection as Integer`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Gets the character collection for a glyph info object.

Notes: If the glyph info object was created with a glyph name or a glyph index, its character collection is `kCTIdentityMappingCharacterCollection`.

(Read only property)

7.7.10 CharacterIdentifier as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Gets the character identifier for a glyph info object.

Notes: (Read only property)

7.7.11 GlyphName as String

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Gets the glyph name for a glyph info object if that object exists.

Notes: (Read only property)

7.7.12 Constants

Character Collection

Constant	Value	Description
<code>kCTAdobeCNS1CharacterCollection</code>	1	The Adobe-CNS1 mapping.
<code>kCTAdobeGB1CharacterCollection</code>	2	The Adobe-GB1 mapping.
<code>kCTAdobeJapan1CharacterCollection</code>	3	The Adobe-Japan1 mapping.
<code>kCTAdobeJapan2CharacterCollection</code>	4	The Adobe-Japan2 mapping.
<code>kCTAdobeKorea1CharacterCollection</code>	5	The Adobe-Korea1 mapping.
<code>kCTCharacterCollectionAdobeCNS1</code>	1	The Adobe-CNS1 mapping.
<code>kCTCharacterCollectionAdobeGB1</code>	2	The Adobe-GB1 mapping.
<code>kCTCharacterCollectionAdobeJapan1</code>	3	The Adobe-Japan1 mapping.
<code>kCTCharacterCollectionAdobeJapan2</code>	4	The Adobe-Japan2 mapping.
<code>kCTCharacterCollectionAdobeKorea1</code>	5	The Adobe-Korea1 mapping.
<code>kCTCharacterCollectionIdentityMapping</code>	0	The character identifier is equal to the <code>CGGlyph</code> glyph index.
<code>kCTIdentityMappingCharacterCollection</code>	0	The character identifier is equal to the <code>CGGlyph</code> glyph index.

7.8 class CTLineMBS

7.8.1 class CTLineMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The CTLine opaque type represents a line of text.

Example:

```
Sub Paint(g As Graphics)
// inside paint event of a Canvas

// create a font, quasi systemFontWithSize:24.0
Var sysUIFont as CTFontMBS = CTFontMBS.CreateUIFontForLanguage(CTFontMBS.kCTFontUIFontSystem, 24.0)

// create a naked string
Var text as string = "Some Text."

// blue
Var cgColor as CGColorMBS = CGColorMBS.CreateGenericRGB(0.0, 0.0, 1.0)

// single underline
Var underline as Integer = CoreTextMBS.kCTUnderlineStyleSingle

// pack it into attributes dictionary
Var attributesDict as new Dictionary
attributesDict.Value(CoreTextMBS.kCTFontAttributeName) = sysUIFont
attributesDict.Value(CoreTextMBS.kCTForegroundColorAttributeName) = cgColor
attributesDict.Value(CoreTextMBS.kCTUnderlineStyleAttributeName) = underline

// make the attributed string
Var cfDic as new CFDictionaryMBS(attributesDict)
Var cfStr as new CFStringMBS(text)
Var stringToDraw as CFAttributedStringMBS = CFAttributedStringMBS.Create( cfStr, cfDic)

// now for the actual drawing

#If XojoVersion >= 2019.02 Then
Var CGContextHandle As ptr = g.Handle(Graphics.HandleTypes.CGContextRef)
Var CGContext As CGContextMBS = CGContextMBS.contextWithCGContext(CGContextHandle)
#Else
Var CGContextHandle As Integer = g.Handle(g.HandleTypeCGContextRef)
Var CGContext As CGContextMBS = CGContextMBS.contextWithCGContext(CGContextHandle)
#EndIf

CGContext.SaveGState
```

```

// reset text matrix
Var a as CGAffineTransformMBS = CGAffineTransformMBS.Identity
CGContext.TextMatrix = a

// draw
Var line as CTLineMBS = CTLineMBS.CreateWithAttributedString(stringToDraw)

Var x as Integer = 10
Var y as Integer = 10

// plus text height
y = y + 24

// swap y
y = g.Height - y

CGContext.TextPosition = new CGPointMBS(x, y)
line.Draw(CGContext)

CGContext.RestoreGState
CGContext.Flush
End Sub

```

Notes: A CTLine object contains an array of glyph runs. Line objects are created by the typesetter during a framesetting operation and can draw themselves directly into a graphics context. Subclass of the CFObjektMBS class. This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [CoreText Completed](#)

7.8.2 Methods

7.8.3 Available as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Whether this class is available.

Notes: Returns true in OS X v10.5 and later.

7.8.4 Bounds(options as Integer = 0) as CGRectMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Queries bounds.

7.8.5 Constructor

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The private constructor.

7.8.6 CreateJustifiedLine(justificationFactor as Double, justificationWidth as Double) as CTLineMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates a justified line from an existing line.

Notes: line: The line from which to create a justified line.

justificationFactor: Full or partial justification. When set to 1.0 or greater, full justification is performed. If this parameter is set to less than 1.0, varying degrees of partial justification are performed. If it is set to 0 or less, no justification is performed.

justificationWidth: The width to which the resultant line is justified. If justificationWidth is less than the actual width of the line, then negative justification is performed (that is, glyphs are squeezed together).

Returns a reference to a justified CTLine object if the call was successful; otherwise, nil.

Available in OS X v10.5 and later.

7.8.7 CreateTruncatedLine(width as Double, truncationType as Integer, truncationToken as CTLineMBS = nil) as CTLineMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates a truncated line from an existing line.

Notes: line: The line from which to create a truncated line.

width: The width at which truncation begins. The line is truncated if its width is greater than the width passed in this parameter.

truncationType: The type of truncation to perform if needed. See constants for possible values.

truncationToken: This token is added at the point where truncation took place, to indicate that the line was truncated. Usually, the truncation token is the ellipsis character (U+2026). If this parameter is set to nil, then no truncation token is used and the line is simply cut off.

Returns a reference to a truncated CTLine object if the call was successful; otherwise, NULL.

The line specified in truncationToken should have a width less than the width specified by the width param-

eter. If the width of the line specified in `truncationToken` is greater than `width` and truncation is needed, the function returns `nil`.

Available in OS X v10.5 and later.

7.8.8 CreateWithAttributedString(s as CFAttributedStringMBS) as CTLineMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates a single immutable line object directly from an attributed string.

Notes: `s`: The string from which the line is created.

Returns a reference to a `CTLine` object if the call was successful; otherwise, `nil`.

This function allows clients who need very simple line generation to create a line without creating a typesetter object. The typesetting is done under the hood. Without a typesetter object, the line cannot be properly broken. However, for simple things like text labels, line breaking is not an issue.

Available in OS X v10.5 and later.

7.8.9 Draw(context as CGContextMBS)

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Draws a complete line.

Notes: `context`: The context into which the line is drawn.

This is a convenience function because the line could be drawn run-by-run by getting the glyph runs, getting the glyphs out of them, and calling a function such as `CGContextMBS.ShowGlyphsAtPositions`. This call can leave the graphics context in any state and does not flush the context after the draw operation.

Available in OS X v10.5 and later.

7.8.10 GlyphRuns as CTRunMBS()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the array of glyph runs that make up the line object.

Notes: Returns an array containing the `CTRunMBS` objects that make up the line.

Available in OS X v10.5 and later.

7.8.11 ImageBounds(context as CGContextMBS) as CGRectMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Calculates the image bounds for a line.

Notes: context: The context for which the image bounds are calculated. This is required because the context could have settings in it that would cause changes in the image bounds.

Returns a rectangle that tightly encloses the paths of the line's glyphs, or, if the line or context is invalid, CGRectNull.

Available in OS X v10.5 and later.

7.8.12 OffsetForStringIndex(charIndex as Integer, byref secondaryOffset as Double) as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Determines the graphical offset or offsets for a string index.

Notes: charIndex: The string index corresponding to the desired position.

secondaryOffset: On output, the secondary offset along the baseline for charIndex. When a single caret is sufficient for a string index, this value will be the same as the primary offset, which is the return value of this function. May be NULL.

Returns the primary offset along the baseline for charIndex, or 0.0 if the line does not support string access.

This function returns the graphical offset or offsets corresponding to a string index, suitable for movement between adjacent lines or for drawing a custom caret. For moving between adjacent lines, the primary offset can be adjusted for any relative indentation of the two lines; a CGPoint constructed with the adjusted offset for its x value and 0.0 for its y value is suitable for passing to CTLineGetStringIndexForPosition. For drawing a custom caret, the returned primary offset corresponds to the portion of the caret that represents the visual insertion location for a character whose direction matches the line's writing direction.

Available in OS X v10.5 and later.

7.8.13 PenOffsetForFlush(flushFactor as Double, flushWidth as Double) as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Gets the pen offset required to draw flush text.

Notes: flushFactor: Determines the type of flushness. A flushFactor of 0 or less indicates left flush. A flushFactor of 1.0 or more indicates right flush. Flush factors between 0 and 1.0 indicate varying degrees of

center flush, with a value of 0.5 being totally center flush.

flushWidth: Specifies the width to which the flushness operation should apply.

Returns the offset from the current pen position for the flush operation.

Available in OS X v10.5 and later.

7.8.14 StringIndexForPosition(position as CGPointMBS) as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Performs hit testing.

Notes: position: The location of the mouse click relative to the line's origin.

Returns the string index for the position, or if the line does not support string access, kCFNotFound. Relative to the line's string range, this value can be no less than the first string index and no greater than the last string index plus 1.

This function can be used to determine the string index for a mouse click or other event. This string index corresponds to the character before which the next character should be inserted. This determination is made by analyzing the string from which a typesetter was created and the corresponding glyphs as embodied by a particular line.

Available in OS X v10.5 and later.

7.8.15 TypographicBounds(byref ascent as Double, byref descent as Double, byref leading as Double) as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Calculates the typographic bounds of a line.

Notes: ascent: On output, the ascent of the line.

descent: On output, the descent of the line.

leading: On output, the leading of the line.

Returns the typographic width of the line. If the line is invalid, this function returns 0.

7.8.16 Properties

7.8.17 GlyphCount as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the total glyph count for the line object.

Notes: The total glyph count is equal to the sum of all of the glyphs in the glyph runs forming the line.
(Read only property)

7.8.18 StringRangeLength as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Gets the range of characters that originally spawned the glyphs in the line.

Notes: (Read only property)

7.8.19 StringRangeLocation as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Gets the range of characters that originally spawned the glyphs in the line.

Notes: (Read only property)

7.8.20 TrailingWhitespaceWidth as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the trailing whitespace width for a line.

Notes: The width of the line's trailing whitespace. If the line is invalid, this function will always return zero.

Creating a line for a width can result in a line that is actually longer than the desired width due to trailing whitespace. Although this is typically not an issue due to whitespace being invisible, this function can be used to determine what amount of a line's width is due to trailing whitespace.

Available in OS X v10.5 and later.

(Read only property)

7.8.21 Constants

Line Bounds

Constant	Value	Description
kCTLineBoundsExcludeTypographicLeading	1	Pass this option to exclude typographic leading.
kCTLineBoundsExcludeTypographicShifts	2	Pass this option to ignore cross-stream shifts due to positioning (such as kerning or baseline alignment).
kCTLineBoundsUseGlyphPathBounds	8	Pass this option to use glyph path bounds rather than the default text box bounds.
kCTLineBoundsUseHangingPunctuation	4	Normally line bounds include all glyphs; pass this option to truncate punctuation hanging off either end of the line as fully hanging.
kCTLineBoundsUseOpticalBounds	16	Pass this option to use optical bounds. This option overrides kCTLineBoundsUseGlyphPathBounds.

Truncate Types

Constant	Value	Description
kCTLineTruncationEnd	1	Truncate the end of the line, leaving the start portion visible.
kCTLineTruncationMiddle	2	Truncate the middle of the line, leaving both the start and the end portions visible.
kCTLineTruncationStart	0	Truncate the beginning of the line, leaving the end portion visible.

7.9 class CTMutableFontCollectionMBS

7.9.1 class CTMutableFontCollectionMBS

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The mutable font collection class.

Notes: Subclass of the CTFontCollectionMBS class.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [CoreText Completed](#)

7.9.2 Methods

7.9.3 Constructor

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: The private constructor.

7.9.4 SetExclusionDescriptors(descriptors()) as CTFontDescriptorMBS)

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Replaces the array of descriptors to exclude from the match.

7.9.5 SetQueryDescriptors(descriptors()) as CTFontDescriptorMBS)

Plugin Version: 14.2, Platform: macOS, Targets: Desktop, Console & Web.

Function: Replaces the array of descriptors to match.

7.10 class CTParagraphStyleMBS

7.10.1 class CTParagraphStyleMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The CTParagraphStyle opaque type represents paragraph or ruler attributes in an attributed string.

Notes: A paragraph style object represents a complex attribute value in an attributed string, storing a number of subattributes that affect paragraph layout for the characters of the string. Among these subattributes are alignment, tab stops, writing direction, line-breaking mode, and indentation settings.

Subclass of the CXObjectMBS class.

Blog Entries

- [CoreText Completed](#)

7.10.2 Methods

7.10.3 Available as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Whether this class is available.

Notes: Returns true in OS X v10.5 and later.

7.10.4 Constructor

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates an immutable paragraph style.

Notes: Returns a valid reference to an immutable CTParagraphStyle object, If the paragraph style creation was successful; otherwise, nil.

Using this function is the easiest and most efficient way to create a paragraph style. Paragraph styles should be kept immutable for totally lock-free operation. If an invalid paragraph style setting specifier is passed into the settings parameter, nothing bad will happen, but you will be unable to query for this value. The reason is to allow backward compatibility with style setting specifiers that may be introduced in future versions.

Available in OS X v10.5 and later.

7.10.5 Create as CTParagraphStyleMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates an immutable paragraph style.

Notes: Returns a valid reference to an immutable CTParagraphStyle object, If the paragraph style creation was successful; otherwise, nil.

Using this function is the easiest and most efficient way to create a paragraph style. Paragraph styles should be kept immutable for totally lock-free operation. If an invalid paragraph style setting specifier is passed into the settings parameter, nothing bad will happen, but you will be unable to query for this value. The reason is to allow backward compatibility with style setting specifiers that may be introduced in future versions.

Available in OS X v10.5 and later.

See also:

- 7.10.6 Create(settings() as CTParagraphStyleSettingMBS) as CTParagraphStyleMBS 576

7.10.6 Create(settings() as CTParagraphStyleSettingMBS) as CTParagraphStyleMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates an immutable paragraph style.

Notes: settings: The settings with which to preload the paragraph style.

Returns a valid reference to an immutable CTParagraphStyle object, If the paragraph style creation was successful; otherwise, nil.

Using this function is the easiest and most efficient way to create a paragraph style. Paragraph styles should be kept immutable for totally lock-free operation. If an invalid paragraph style setting specifier is passed into the settings parameter, nothing bad will happen, but you will be unable to query for this value. The reason is to allow backward compatibility with style setting specifiers that may be introduced in future versions.

Available in OS X v10.5 and later.

See also:

- 7.10.5 Create as CTParagraphStyleMBS 576

7.10.7 CreateCopy as CTParagraphStyleMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates an immutable copy of a paragraph style.

Notes: A valid reference to an immutable CTParagraphStyle object that is a copy of the one passed into paragraphStyle, If the paragraphStyle reference is valid; otherwise nil, if any error occurred, including being supplied with an invalid reference.

7.10.8 CreateWithAlignment(Alignment as Integer) as CTParagraphStyleMBS

Plugin Version: 15.1, Platform: macOS, Targets: All.

Function: Creates an immutable paragraph style.

Example:

```
Var a as Integer
```

```
Var t as CTParagraphStyleMBS
```

```
a = CTParagraphStyleMBS.kCTTextAlignmentRight
```

```
t = CTParagraphStyleMBS.CreateWithAlignment(a)
```

```
MsgBox str(T.Alignment) // shows 1
```

7.10.9 TabStops as CTTextTabMBS()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The CTTextTab objects, sorted by location, that define the tab stops for the paragraph style.

Notes: Type: Array of CTTextTabMBS. Default: 12 left-aligned tabs, spaced by 28.0 points. Application: CTFramesetter, CTTypesetter.

7.10.10 Properties

7.10.11 Alignment as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The text alignment.

Notes: Natural text alignment is realized as left or right alignment, depending on the line sweep direction of the first script contained in the paragraph. Type: CTTextAlignment. Default: kCTNaturalTextAlignment. Application: CTFramesetter.

(Read only property)

7.10.12 BaseWritingDirection as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The base writing direction of the lines.

Notes: Type: CTWritingDirection. Default: kCTWritingDirectionNatural. Application: CTFramesetter, CTypesetter.

(Read only property)

7.10.13 DefaultTabInterval as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The documentwide default tab interval.

Notes: Tabs after the last specified by kCTParagraphStyleSpecifierTabStops are placed at integer multiples of this distance (if positive). Type: CGFloat. Default: 0.0. Application: CTFramesetter, CTypesetter.

(Read only property)

7.10.14 FirstLineHeadIndent as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The distance, in points, from the leading margin of a frame to the beginning of the paragraph's first line.

Notes: This value is always nonnegative. Type: CGFloat. Default: 0.0. Application: CTFramesetter.

(Read only property)

7.10.15 HeadIndent as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The distance, in points, from the leading margin of a text container to the beginning of lines other than the first.

Notes: This value is always nonnegative. Type: CGFloat. Default: 0.0. Application: CTFramesetter.

(Read only property)

7.10.16 LineBoundsOptions as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The options controlling the alignment of the line edges with the leading and trailing margins.

Notes: Type: CTLineBoundsOptions

Default: 0 (no options)
Application: CTypesetter
(Read only property)

7.10.17 LineBreakMode as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The mode that should be used to break lines when laying out the paragraph's text.

Notes: Type: CTLineBreakMode. Default: kCTLineBreakByWordWrapping. Application: CTFramesetter
(Read only property)

7.10.18 LineHeightMultiple as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The line height multiple.

Notes: The natural line height of the receiver is multiplied by this factor (if positive) before being constrained by minimum and maximum line height. Type: CGFloat. Default: 0.0. Application: CTFramesetter.
(Read only property)

7.10.19 LineSpacingAdjustment as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The space in points added between lines within the paragraph (commonly known as leading).

Notes: Available in OS X v10.7 and later.

(Read only property)

7.10.20 MaximumLineHeight as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The maximum height that any line in the frame will occupy, regardless of the font size or size of any attached graphic.

Notes: Glyphs and graphics exceeding this height will overlap neighboring lines. A maximum height of 0 implies no line height limit. This value is always nonnegative. Type: CGFloat. Default: 0.0. Application: CTFramesetter.

(Read only property)

7.10.21 `MaximumLineSpacing` as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The maximum space in points between lines within the paragraph (commonly known as leading).

Notes: This value is always nonnegative.

Available in OS X v10.7 and later.

(Read only property)

7.10.22 `MinimumLineHeight` as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The minimum height that any line in the frame will occupy, regardless of the font size or size of any attached graphic.

Notes: This value is always nonnegative. Type: `CGFloat`. Default: 0.0. Application: `CTFramesetter`.

(Read only property)

7.10.23 `MinimumLineSpacing` as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The minimum space in points between lines within the paragraph (commonly known as leading).

Notes: This value is always nonnegative.

Available in OS X v10.7 and later.

(Read only property)

7.10.24 `ParagraphSpacing` as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The space added at the end of the paragraph to separate it from the following paragraph.

Notes: This value is always nonnegative and is determined by adding the previous paragraph's `kCTParagraphStyleSpecifierParagraphSpacing` setting and the current paragraph's `kCTParagraphStyleSpecifierParagraphSpacingBefore` setting. Type: `CGFloat`. Default: 0.0. Application: `CTFramesetter`.

(Read only property)

7.10.25 `ParagraphSpacingBefore` as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The distance between the paragraph’s top and the beginning of its text content.

Notes: Type: CGFloat. Default: 0.0. Application: CTFramesetter.

(Read only property)

7.10.26 TailIndent as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The distance, in points, from the margin of a frame to the end of lines.

Notes: If positive, this value is the distance from the leading margin (for example, the left margin in left-to-right text). If 0 or negative, it’s the distance from the trailing margin. Type: CGFloat. Default: 0.0. Application: CTFramesetter.

(Read only property)

7.10.27 Constants

Constants

Constant	Value	Description
kCTTextAlignmentCenter	2	One of the constants to specify text alignment. Text is visually center aligned.
kCTTextAlignmentJustified	3	One of the constants to specify text alignment. Text is fully justified. The last line in a paragraph is naturally aligned.
kCTTextAlignmentLeft	0	One of the constants to specify text alignment. Text is visually left aligned.
kCTTextAlignmentNatural	4	One of the constants to specify text alignment. Text uses the natural alignment of the text’s script.
kCTTextAlignmentRight	1	One of the constants to specify text alignment. Text is visually right aligned.

Line Breaking

Constant	Value	Description
kCTLineBreakByCharWrapping	1	Wrapping occurs before the first character that doesn’t fit.
kCTLineBreakByClipping	2	Lines are simply not drawn past the edge of the frame.
kCTLineBreakByTruncatingHead	3	Each line is displayed so that the end fits in the frame and the missing text is indicated by an ellipsis glyph.
kCTLineBreakByTruncatingMiddle	5	Each line is displayed so that the beginning and end fit in the container and the missing text is indicated by an ellipsis glyph in the middle.
kCTLineBreakByTruncatingTail	4	Each line is displayed so that the beginning fits in the container and the missing text is indicated by an ellipsis glyph.
kCTLineBreakByWordWrapping	0	Wrapping occurs at word boundaries unless the word itself doesn’t fit on a single line.

Constant	Value	Description
kCTParagraphStyleSpecifierAlignment	0	The text alignment. Natural text alignment is realized as justified text alignment, depending on the line sweep direction of the first sentence in the paragraph. Type: CTTextAlignment. Default: kCTNaturalTextAlignment. Application: CTFramesetter. Available in OS X v10.5 and later.
kCTParagraphStyleSpecifierBaseWritingDirection	13	The base writing direction of the lines. Type: CTWritingDirection. Default: kCTWritingDirectionNatural. Application: CTFramesetter.
kCTParagraphStyleSpecifierDefaultTabInterval	5	The documentwide default tab interval. Tabs after the last paragraph style specifier are placed at integer multiples of this interval (if positive). Type: CGFloat. Default: 0.0. Application: CTFramesetter, CTTypesetter.
kCTParagraphStyleSpecifierFirstLineHeadIndent	1	The distance, in points, from the leading margin of a frame to the paragraph's first line. This value is always nonnegative. Default: 0.0. Application: CTFramesetter.
kCTParagraphStyleSpecifierHeadIndent	2	The distance, in points, from the leading margin of a text block to the beginning of lines other than the first. This value is always nonnegative. Type: CGFloat. Default: 0.0. Application: CTFramesetter.
kCTParagraphStyleSpecifierLineBoundsOptions	17	The options controlling the alignment of the line edges with the leading and trailing margins. Type: CTLineBoundsOptions. Default: 0 (no options). Application: CTTypesetter.
kCTParagraphStyleSpecifierLineBreakMode	6	The mode that should be used to break lines when laying out text. Type: CTLineBreakMode. Default: kCTLineBreakModeNatural. Application: CTFramesetter.
kCTParagraphStyleSpecifierLineHeightMultiple	7	The line height multiple. The natural line height of the text is multiplied by this factor (if positive) before being constrained by minimum and maximum line height. Type: CGFloat. Default: 0.0. Application: CTFramesetter.
kCTParagraphStyleSpecifierLineSpacing	10	Deprecated. Use kCTParagraphStyleSpecifierMaximumLineSpacing, kCTParagraphStyleSpecifierMinimumLineSpacing, and kCTParagraphStyleSpecifierLineSpaceAdjustment to control space between lines. The space added between lines within the paragraph (commonly known as leading). This value is always nonnegative. Type: CGFloat. Default: 0.0. Application: CTFramesetter.
kCTParagraphStyleSpecifierLineSpacingAdjustment	16	The space in points added between lines within the paragraph (commonly known as leading). Available in OS X v10.7 and later.
kCTParagraphStyleSpecifierMaximumLineHeight	8	The maximum height that any line in the frame will occupy, expressed as a multiple of the font size or size of any attached graphic. Glyphs and graphics taller than this height will overlap neighboring lines. A maximum height of 0.0 indicates no height limit. This value is always nonnegative. Type: CGFloat. Application: CTFramesetter.
kCTParagraphStyleSpecifierMaximumLineSpacing	14	The maximum space in points between lines within the paragraph (commonly known as leading). This value is always nonnegative. Available in OS X v10.7 and later.
kCTParagraphStyleSpecifierMinimumLineHeight	9	The minimum height that any line in the frame will occupy, expressed as a multiple of the font size or size of any attached graphic. This value is always nonnegative. Type: CGFloat. Default: 0.0. Application: CTFramesetter.
kCTParagraphStyleSpecifierMinimumLineSpacing	15	The minimum space in points between lines within the paragraph (commonly known as leading). This value is always nonnegative. Available in OS X v10.7 and later.
kCTParagraphStyleSpecifierParagraphSpacing	11	The space added at the end of the paragraph to separate it from the previous paragraph. This value is always nonnegative and is defined relative to the previous paragraph's kCTParagraphStyleSpecifierParagraphSpacing and the current paragraph's kCTParagraphStyleSpecifierParagraphSpacingBefore setting. Type: CGFloat. Default: 0.0. Application: CTFramesetter.
kCTParagraphStyleSpecifierParagraphSpacingBefore	12	The distance between the paragraph's top and the beginning of the first line. Type: CGFloat. Default: 0.0. Application: CTFramesetter.

Writing Direction

Constant	Value	Description
kCTWritingDirectionLeftToRight	0	The writing direction is left to right.
kCTWritingDirectionNatural	-1	The writing direction is algorithmically determined using the Unicode Bidirectional Algorithm rules P2 and P3.
kCTWritingDirectionRightToLeft	1	The writing direction is right to left.

7.11 class CTParagraphStyleSettingMBS

7.11.1 class CTParagraphStyleSettingMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: This class is used to alter the paragraph style.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 14.2pr11](#)
- [CoreText Completed](#)

7.11.2 Methods

7.11.3 SetTextTabs(textTabs() as CTTextTabMBS)

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Sets the array of text tabs.

Notes: for use with kCTParagraphStyleSpecifierTabStops.

7.11.4 Properties

7.11.5 doubleValue as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The double value.

Notes: (Read and Write property)

7.11.6 intValue as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The integer value.

Notes: Also used for enumerations.

(Read and Write property)

7.11.7 Spec as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The specifier of the setting.

Notes: See `CTParagraphStyleMBS` for possible values.
(Read and Write property)

7.12 class CTRunDelegateMBS

7.12.1 class CTRunDelegateMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The class to get events for CTRun runs.

Notes: The CTRunDelegate opaque type represents a run delegate, which is assigned to a run (attribute range) to control typographic traits such as glyph ascent, glyph descent, and glyph width.

The events defined for CTRunDelegate are used to modify glyph metrics during layout. The values returned by the delegate are applied to each glyph in the run or runs corresponding to the attribute with that delegate. Subclass of the CXObjectMBS class.

Blog Entries

- [CoreText Completed](#)

7.12.2 Methods

7.12.3 Available as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Whether this class is available.

Notes: Returns true in OS X v10.5 and later.

7.12.4 Close

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Shuts down the delegate.

Notes: Please call this to explicitly end the delegate and avoid memory leaks.

See also:

- [7.12.7 Close](#)

7.12.5 Constructor

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates an immutable instance of a run delegate.

Notes: The run-delegate object can be used for reserving space in a line or for eliding the glyphs for a range of text altogether.

7.12.6 Events

7.12.7 Close

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Event called when item closes.

See also:

- 7.12.4 Close

587

7.12.8 GetAscent as Double

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Event that determines typographic ascent of glyphs in the run.

Notes: Returns the typographic ascent of glyphs in the run associated with the run delegate.

7.12.9 GetDescent as Double

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Event that determines typographic descent of glyphs in the run.

Notes: Returns the typographic descent of glyphs in the run associated with the run delegate.

7.12.10 GetWidth as Double

Plugin Version: 14.2, Platform: macOS, Targets: .

Function: Event that determines the typographic width of glyphs in the run.

Notes: Returns the typographic width of glyphs in the run associated with the run delegate. A value of 0.0 indicates that the glyphs should not be drawn.

7.13 class CTRunMBS

7.13.1 class CTRunMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The CTRun opaque type represents a glyph run, which is a set of consecutive glyphs sharing the same attributes and direction.

Notes: The typesetter creates glyph runs as it produces lines from character strings, attributes, and font objects. That is, a line is constructed of one or more glyphs runs. Glyph runs can draw themselves into a graphic context, if desired, although most users have no need to interact directly with glyph runs.

Subclass of the CXObjectMBS class.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [MBS Xojo Plugins, version 24.6pr1](#)
- [MBS Xojo Plugins, version 22.2pr3](#)
- [MBS Xojo / Real Studio Plugins, version 15.1pr7](#)
- [CoreText Completed](#)

7.13.2 Methods

7.13.3 Advances as CGSizeMBS()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Copies a range of glyph advances into an array.

Notes: Changed in v22.2 to use copy function to get array instead Ptr function, if no pointer is available.

7.13.4 Available as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Whether this class is available.

Notes: Returns true in OS X v10.5 and later.

7.13.5 Constructor

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The private constructor.

7.13.6 Draw(context as CGContextMBS, location as Integer, length as Integer = 0)

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Draws a complete run or part of one.

Notes: context: The context into which to draw the run.

range: The portion of the run to draw. If the length of the range is set to 0, then the draw operation continues from the start index of the range to the end of the run.

This is a convenience call, because the run could be drawn by accessing the glyphs. This call can leave the graphics context in any state and does not flush the context after the draw operation.

Available in OS X v10.5 and later.

7.13.7 Glyphs as Integer()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Copies a range of glyphs into an array.

Notes: Changed in v22.2 to use copy function to get array instead Ptr function, if no pointer is available.

7.13.8 ImageBounds(context as CGContextMBS, location as Integer, length as Integer) as CGRectMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Calculates the image bounds for a glyph range.

Notes: context: The context for the image bounds being calculated. This is required because the context could have settings in it that would cause changes in the image bounds.

range: The portion of the run to measure. If the length of the range is set to 0, then the measure operation continues from the start index of the range to the end of the run.

Returns a rectangle that tightly encloses the paths of the run's glyphs, or, if run, context, or range is invalid, CGRectNull.

Available in OS X v10.5 and later.

7.13.9 Positions as CGPointMBS()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Copies a range of glyph positions into an array.

Notes: Changed in v22.2 to use copy function to get array instead Ptr function, if no pointer is available.

7.13.10 StringIndices as Integer()

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Copies a range of string indices into an array.

Notes: The indices are the character indices that originally spawned the glyphs that make up the run. They can be used to map the glyphs in the run back to the characters in the backing store.

Changed in v22.2 to use copy function to get array instead Ptr function, if no pointer is available.

7.13.11 TypographicBounds(location as Integer, length as Integer, byref ascent as Double, byref descent as Double, byref leading as Double) as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Gets the typographic bounds of the run.

Notes: range: The portion of the run to measure. If the length of the range is set to 0, then the measure operation continues from the range's start index to the end of the run.

ascent: On output, the ascent of the run.

descent: On output, the descent of the run.

leading: On output, the leading of the run.

Returns the typographic width of the run, or if run or range is invalid, 0.

Available in OS X v10.5 and later.

7.13.12 Properties

7.13.13 AttributeValues as Dictionary

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the attribute dictionary that was used to create the glyph run.

Notes: Return a valid Dictionary or nil on error or if the run has no attributes.

The dictionary returned is either the same one that was set as an attribute dictionary on the original attributed string or a dictionary that has been manufactured by the layout engine. Attribute dictionaries can be manufactured in the case of font substitution or if the run is missing critical attributes.

(Read only property)

7.13.14 GlyphCount as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Gets the glyph count for the run.

Notes: run: The run for which to return the glyph count.

Returns the number of glyphs that the run contains, or if there are no glyphs in this run, a value of 0. Available in OS X v10.5 and later.

(Read only property)

7.13.15 Status as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the run's status.

Notes: Runs have status that can be used to expedite certain operations. Knowing the direction and ordering of a run's glyphs can aid in string index analysis, whereas knowing whether the positions reference the identity text matrix can avoid expensive comparisons. This status is provided as a convenience, because this information is not strictly necessary but can be helpful in some circumstances.

(Read only property)

7.13.16 StringRangeLength as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Gets the range of characters that originally spawned the glyphs in the run.

Notes: The range of characters that originally spawned the glyphs, or if run is invalid, an empty range.

(Read only property)

7.13.17 StringRangeLocation as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Gets the range of characters that originally spawned the glyphs in the run.

Notes: The range of characters that originally spawned the glyphs, or if run is invalid, an empty range.

(Read only property)

7.13.18 TextMatrix as CGAffineTransformMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the text matrix needed to draw this run.

Notes: To properly draw the glyphs in a run, the fields tx and ty of the CGAffineTransform returned by this function should be set to the current text position.

(Read only property)

7.13.19 Constants

Constants

Constant	Value	Description
kCTRunStatusHasNonIdentityMatrix	4	One of the possible values for the status bitfield. The run requires a specific text matrix to be set in the current Core Graphics context for proper drawing. Use BitwiseAnd() to check if the status has a given value.
kCTRunStatusNonMonotonic	2	One of the possible values for the status bitfield. The run has been reordered in some way such that the string indices associated with the glyphs are no longer strictly increasing (for left-to-right runs) or strictly decreasing (for right-to-left runs). Use BitwiseAnd() to check if the status has a given value.
kCTRunStatusNoStatus	0	One of the possible values for the status bitfield. The run has no special attributes.
kCTRunStatusRightToLeft	1	One of the possible values for the status bitfield. The run proceeds from right to left. Use BitwiseAnd() to check if the status has a given value.

7.14 class `CTTextTabMBS`

7.14.1 class `CTTextTabMBS`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The `CTTextTab` opaque type represents a tab in a paragraph style, storing an alignment type and location.

Notes: Core Text supports four alignment types: left, center, right, and decimal. These alignment types are absolute, not based on the line sweep direction of text. For example, tabbed text is always positioned to the left of a right-aligned tab, whether the line sweep direction is left to right or right to left. A tab's location, on the other hand, is relative to the back margin. A tab set at 1.5 inches, for example, is at 1.5 inches from the right in right-to-left text.

Subclass of the `CFOBJECTMBS` class.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 15.1pr7](#)
- [CoreText Completed](#)

7.14.2 Methods

7.14.3 Available as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Whether this class is available.

Notes: Returns true in OS X v10.5 and later.

7.14.4 Constructor

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The private constructor.

7.14.5 `Create(alignment as Integer, location as Double, options as Dictionary = nil)` as `CTTextTabMBS`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates and initializes a new text tab object.

Notes: alignment: The tab's alignment. This is used to determine the position of text inside the tab column.

This parameter must be set to a valid CTextAlignment value or this function returns nil.

location: The tab's ruler location, relative to the back margin.

options: Options to pass in when the tab is created. Currently, the only option available is kCTTabColumnTerminatorsAttributeName. This parameter is optional and can be set to nil if not needed.

Return a reference to a CTextTab object if the call was successful; otherwise, nil.

Available in OS X v10.5 and later.

7.14.6 kCTTabColumnTerminatorsAttributeName as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the options keys.

Notes: Specifies the terminating character for a tab column.

The value associated with this attribute is a CFCharacterSet object. The character set is used to determine the terminating character for a tab column. The tab and newline characters are implied even if they don't exist in the character set. This attribute can be used to implement decimal tabs, for instance. This attribute is optional.

7.14.7 Properties

7.14.8 Alignment as Integer

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the text alignment of the tab.

Notes: (Read only property)

7.14.9 Location as Double

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the tab's ruler location.

Notes: The tab's ruler location relative to the back margin.
(Read only property)

7.14.10 Options as Dictionary

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Returns the dictionary of attributes associated with the tab.

Notes: The dictionary of attributes associated with the tab, or if no dictionary is present, nil.

(Read only property)

7.15 class CTTypesetterMBS

7.15.1 class CTTypesetterMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The CTTypesetter opaque type represents a typesetter, which performs line layout.

Notes: Line layout includes word wrapping, hyphenation, and line breaking in either vertical or horizontal rectangles. A typesetter object takes as input an attributed string and produces a line of typeset glyphs (composed into glyph runs) in a CTLine object. The typesetter performs character-to-glyph encoding, glyph ordering, and positional operations, such as kerning, tracking, and baseline adjustments. If multiline layout is needed, it is performed by a framesetter object, which calls into the typesetter to generate the typeset lines to fill the frame.

A framesetter encapsulates a typesetter and provides a reference to it as a convenience, but a caller may also choose to create a freestanding typesetter.

Subclass of the CFOBJECTMBS class.

This is an abstract class. You can't create an instance, but you can get one from various plugin functions.

Blog Entries

- [CoreText Completed](#)

7.15.2 Methods

7.15.3 Available as boolean

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Whether this class is available.

Notes: Returns true in OS X v10.5 and later.

7.15.4 Constructor

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: The private constructor.

7.15.5 CreateLine(location as Integer, length as Integer, offset as Double = 0.0) as CTLineMBS

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates an immutable line from the typesetter at a specified line offset.

Notes: location and length: The string range on which the line is based. If the length portion of range is set to 0, then the typesetter continues to add glyphs to the line until it runs out of characters in the string. The location and length of the range must be within the bounds of the string, or the call will fail.

offset: The line position offset.

Returns a reference to a `CTLine` object if the call was successful; otherwise, `nil`.

The resultant line consists of glyphs in the correct visual order, ready to draw.

Available in OS X v10.6 and later.

7.15.6 `CreateWithAttributedString(s as CFAttributedStringMBS) as CTypesetterMBS`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates an immutable typesetter object using an attributed string.

Notes: `s`: The attributed string to typeset. This parameter must be filled in with a valid `CFAttributedString` object.

Returns a reference to a `CTypesetter` object if the call was successful; otherwise, `nil`.

The resultant typesetter can be used to create lines, perform line breaking, and do other contextual analysis based on the characters in the string.

Available in OS X v10.5 and later.

See also:

- 7.15.7 `CreateWithAttributedString(s as CFAttributedStringMBS, options as dictionary) as CTypesetterMBS` 598

7.15.7 `CreateWithAttributedString(s as CFAttributedStringMBS, options as dictionary) as CTypesetterMBS`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Creates an immutable typesetter object using an attributed string and a dictionary of options.

Notes: `s`: The attributed string to typeset. This parameter must be filled in with a valid `CFAttributedString` object.

`options`: A dictionary of typesetter options, or `nil` if there are none.

Returns a reference to a `CTypesetter` object if the call was successful; otherwise, `nil`.

The resultant typesetter can be used to create lines, perform line breaking, and do other contextual analysis based on the characters in the string.

Available in OS X v10.5 and later.

See also:

- 7.15.6 `CreateWithAttributedString(s as CFAttributedStringMBS) as CTypesetterMBS` 598

7.15.8 `kCTTypesetterOptionDisableBidiProcessing` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants to control aspects of the typesetter,Ãs bidirectional text processing.

Notes: Disables bidirectional processing. Value must be a `CFBoolean` object. Default value is false. Normally, typesetting applies the Unicode Bidirectional Algorithm as described in Unicode Standard Annex #9. If a typesetter is created with this option set to true, no directional reordering is performed, and any directional control characters are ignored.

Available in OS X v10.5 and later.

Deprecated in OS X v10.8.

7.15.9 `kCTTypesetterOptionForcedEmbeddingLevel` as string

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: One of the constants to control aspects of the typesetter,Ãs bidirectional text processing.

Notes: Specifies the embedding level. Value must be a `CFNumberRef` object. Default is unset. Normally, typesetting applies the Unicode Bidirectional Algorithm as described in Unicode Standard Annex #9. If present, this option specifies the embedding level, and any directional control characters are ignored.

Available in OS X v10.5 and later.

7.15.10 `SuggestClusterBreak(startIndex as Integer, width as Double) as Integer`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Suggests a cluster line breakpoint based on the width provided.

Notes: `startIndex`: The starting point for the typographic cluster-break calculations. The break calculations include the character starting at `startIndex`.

`width`: The requested typographic cluster-break width.

Return a count of the characters from `startIndex` that would cause the cluster break. The value returned can be used to construct a character range for `CTTypesetterCreateLine`.

This cluster break is similar to a character break, except that it does not break apart linguistic clusters. No other contextual analysis is done. This can be used by the caller to implement a different line-breaking

scheme, such as hyphenation. A typographic cluster break can also be triggered by a hard-break character in the stream. This function is equivalent to `SuggestClusterBreakWithOffset` with an offset of 0.0.

Available in OS X v10.5 and later.
See also:

- 7.15.11 `SuggestClusterBreak(startIndex as Integer, width as Double, offset as Double) as Integer` 600

7.15.11 `SuggestClusterBreak(startIndex as Integer, width as Double, offset as Double) as Integer`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Suggests a cluster line breakpoint based on the specified width and line offset.

Notes: `startIndex`: The starting point for the typographic cluster-break calculations. The break calculations include the character starting at `startIndex`.

`width`: The requested typographic cluster-break width.

`offset`: The line offset position.

Returns a count of the characters from `startIndex` that would cause the cluster break. The value returned can be used to construct a character range for `CreateLine`.

This cluster break is similar to a character break, except that it does not break apart linguistic clusters. No other contextual analysis is done. This can be used by the caller to implement a different line-breaking scheme, such as hyphenation. A typographic cluster break can also be triggered by a hard-break character in the stream.

Available in OS X v10.6 and later.
See also:

- 7.15.10 `SuggestClusterBreak(startIndex as Integer, width as Double) as Integer` 599

7.15.12 `SuggestLineBreak(startIndex as Integer, width as Double) as Integer`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Suggests a contextual line breakpoint based on the width provided.

Notes: `startIndex`: The starting point for the line-break calculations. The break calculations include the character starting at `startIndex`.

`width`: The requested line-break width.

Returns a count of the characters from `startIndex` that would cause the line break. The value returned can be used to construct a character range for `CreateLine`.

The line break can be triggered either by a hard-break character in the stream or by filling the specified width with characters. This function is equivalent to `SuggestLineBreakWithOffset` with an offset of 0.0.

Available in OS X v10.5 and later.

See also:

- 7.15.13 `SuggestLineBreak(startIndex as Integer, width as Double, offset as Double) as Integer` 601

7.15.13 `SuggestLineBreak(startIndex as Integer, width as Double, offset as Double) as Integer`

Plugin Version: 14.2, Platform: macOS, Targets: All.

Function: Suggests a contextual line breakpoint based on the width provided and the specified offset.

Notes: `startIndex`: The starting point for the line-break calculations. The break calculations include the character starting at `startIndex`.

`width`: The requested line-break width.

`offset`: The line position offset.

Returns a count of the characters from `startIndex` and `offset` that would cause the line break. The value returned can be used to construct a character range for `CreateLine`.

The line break can be triggered either by a hard-break character in the stream or by filling the specified width with characters.

Available in OS X v10.6 and later.

See also:

- 7.15.12 `SuggestLineBreak(startIndex as Integer, width as Double) as Integer` 600

Chapter 8

Files

8.1 class FolderItem

8.1.1 class FolderItem

Platforms: macOS, Linux, Windows, Targets: All.

Function: One of Xojo's base classes.

Notes: Handles access to files.

8.1.2 Methods

8.1.3 NewCGPDFDocumentMBS(MediaBox as CGRectMBS, title as string, author as string, creator as string) as CGPDFContextMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Creates a PDF document.

Example:

```
// create pdf
Var file as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
Var r as new CGRectMBS(0,0,500,500)
Var c as CGContextMBS = file.NewCGPDFDocumentMBS(r, "My Title", "My Author", "My Creator")

if c<>Nil then

// create page
c.BeginPage r
```

```
// draw something
c.SetRGBFillColor(1.0, 0.0, 0.0, 1.0)
c.FillRect CGMakeRectMBS(100,100,100,100)

// close page
c.EndPage

// flush and show in PDF viewer
c = nil
file.Launch
end if
```

Notes: Title, author and creator are all optional.

RB 4.5 should do this perfectly, but older RB versions may have problems.

(seems like the file must exist before calling this function)

Requires Mac OS X to work.

See also:

- 8.1.4 NewCGPDFDocumentMBS(MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean) as CGPDFContextMBS 604
- 8.1.5 NewCGPDFDocumentMBS(MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean, KeyLength as Integer) as CGPDFContextMBS 605

8.1.4 NewCGPDFDocumentMBS(MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean) as CGPDFContextMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Creates a PDF document.

Example:

```
// create pdf
Var file as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
Var r as new CGRectMBS(0,0,500,500)
Var c as CGContextMBS = file.NewCGPDFDocumentMBS(r, "My Title", "My Author", "My Creator",
"My Subject", "test, pdf, mac", "", "", true, true)

if c<>Nil then

// create page
c.BeginPage r
```

```
// draw something
c.SetRGBFillColor(1.0, 0.0, 0.0, 1.0)
c.FillRect CGMakeRectMBS(100,100,100,100)

// close page
c.EndPage

// flush and show in PDF viewer
c = nil
file.Launch
end if
```

Notes: Title, Author, Creator, Subject and Keywords parameters can be empty.

If OwnerPassword and UserPassword are filled in the PDF is encrypted and AllowsPrinting/AllowsCopy define what the user can do after he entered his password.

Requires Mac OS X to work.

See also:

- 8.1.3 NewCGPDFDocumentMBS(MediaBox as CGRectMBS, title as string, author as string, creator as string) as CGPDFContextMBS 603
- 8.1.5 NewCGPDFDocumentMBS(MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean, KeyLength as Integer) as CGPDFContextMBS 605

8.1.5 NewCGPDFDocumentMBS(MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean, KeyLength as Integer) as CGPDFContextMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Creates a PDF document.

Example:

```
// create pdf
Var file as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
Var r as new CGRectMBS(0,0,500,500)
Var c as CGContextMBS = file.NewCGPDFDocumentMBS(r, "My Title", "My Author", "My Creator",
"My Subject", "test,pdf,mac", "owner","user", false, false, 128)

if c<>Nil then
```

```
// create page
c.BeginPage r

// draw something
c.SetRGBFillColor(1.0, 0.0, 0.0, 1.0)
c.FillRect CGMakeRectMBS(100,100,100,100)

// close page
c.EndPage

// flush and show in PDF viewer
c = nil
file.Launch
end if
```

Notes: Title, Author, Creator, Subject and Keywords parameters can be empty.

If OwnerPassword and UserPassword are filled in the PDF is encrypted and AllowsPrinting/AllowsCopy define what the user can do after he entered his password.

Keylength must be a value between 48 bit and 128 bit in 8 bit steps. 0 uses default value.

Requires Mac OS X to work.

Blog Entries

- [Several ways for picture to PDF in MBS Plugins](#)

See also:

- 8.1.3 NewCGPDFDocumentMBS(MediaBox as CGRectMBS, title as string, author as string, creator as string) as CGPDFContextMBS 603
- 8.1.4 NewCGPDFDocumentMBS(MediaBox as CGRectMBS, title as string, author as string, creator as string, subject as string, keywords as string, OwnerPassword as string, UserPassword as string, AllowsPrinting as boolean, AllowsCopy as boolean) as CGPDFContextMBS 604

8.1.6 NewCGPDFDocumentWithInfoMBS(MediaBox as CGRectMBS, info as object) as CGPDFContextMBS

Plugin Version: 12.0, Platform: macOS, Targets: All.

Function: Creates a PDF document.

Example:

```

// create pdf
Var file as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
Var r as new CGRectMBS(0,0,500,500)

Var d as new CFMutableDictionaryMBS

d.Add NewCFStringMBS("kCGPDFContextTitle"), NewCFStringMBS("My Title")
d.Add NewCFStringMBS("kCGPDFContextAuthor"), NewCFStringMBS("My Author")
d.Add NewCFStringMBS("kCGPDFContextCreator"), NewCFStringMBS("My Creator")
d.Add NewCFStringMBS("kCGPDFContextSubject"), NewCFStringMBS("My Subject")
d.Add NewCFStringMBS("kCGPDFContextKeywords"), NewCFStringMBS("keyword,mac,pdf")

Var c as CGContextMBS = file.NewCGPDFDocumentWithInfoMBS(r, d)

if c<<>Nil then

// create page
c.BeginPage r

// draw something
c.SetRGBFillColor(1.0, 0.0, 0.0, 1.0)
c.FillRect CGMakeRectMBS(100,100,100,100)

// close page
c.EndPage

// flush and show in PDF viewer
c = nil
file.Launch
end if

```

Notes: The optional info parameter is a CFDictionaryMBS object and contains some information about the PDF file.

Keys for auxiliary info dictionary:

NewCFStringMBS("kCGPDFContextTitle")

The document's title. Optional; if present, the value must be a CFString.

NewCFStringMBS("kCGPDFContextAuthor")

The name of the person who created this document. Optional; if present, the value must be a CFString.

NewCFStringMBS("kCGPDFContextCreator")

The name of the application that created the original data used to create this document. Optional; if present, the value must be a CFString.

`NewCFStringMBS("kCGPDFContextOutputIntent")`

The document's output intent. Optional; if present, the value must be a `CFDictionaryMBS`. The dictionary is added to the PDF document in the `/OutputIntents` entry in the PDF file's document catalog. The keys and values contained in the dictionary must match those specified in section 9.10.4 of the PDF 1.4 specification, ISO/DIS 15930-3 document published by ISO/TC 130, and Adobe Technical Note #5413.

The following keys are supported:

"S" - The output intent subtype. This key is required; the value must be a `CFString` equal to "GTS_PDFX"; otherwise, the dictionary is ignored.

"OutputConditionIdentifier" - A string identifying the intended output device or production condition in a human- or machine-readable form. This key is required; the value must be a `CFString`. For best results, the string should be representable losslessly in ASCII encoding.

"OutputCondition" - A text string identifying the intended output device or production condition in a human-readable form. This key is optional; if present, the value must be a `CFString`.

"RegistryName" - A string identifying the registry in which the condition designated by "OutputConditionIdentifier" is defined. This key is optional; if present, the value must be a `CFString`. For best results, the string should be representable losslessly in ASCII encoding.

"Info" - A human-readable text string containing additional information or comments about the intended target device or production condition. This key is required if "OutputConditionIdentifier" does not specify a standard production condition; it is optional otherwise. If present, the value must be a `CFString`.

"DestOutputProfile" - An ICC profile stream defined the transformation from the PDF document's source colors to output device colorants. This key is required if "OutputConditionIdentifier" does not specify a standard production condition; it is optional otherwise. If present, the value must be a ICC-based `CGColorSpaceMBS`.

`NewCFStringMBS("kCGPDFContextOutputIntents")`

The document's output intents. Optional; if present, the value must be a `CFArrayMBS` containing one or more `CFDictionaryMBS`s. The array is added to the PDF document in the `/OutputIntents` entry in the PDF file's document catalog. Each dictionary in the array must be of form specified above for the "kCGPDFContextOutputIntent" key, except that only the first dictionary in the array is required to contain the "S" key with a value of "GTS_PDFX". If both the "kCGPDFContextOutputIntent" and "kCGPDFContextOutputIntents" keys are specified, the former will be ignored.

More keys in `CGPDFContext.h` coming with Xcode.

8.1.7 OpenAsCGPDFDocumentMBS as CGPDFDocumentMBS

Plugin Version: 2.7, Platform: macOS, Targets: All.

Function: Opens a PDF document.

Notes: RB 4.5 should do this perfectly, but older RB versions may have problems with longer file names.
Requires Mac OS X to work.

Xojo Developer Magazine

- [8.5, pages 20 to 21: Printing PDFs, Easy PDF printing on Mac and Windows](#)
- [6.4, page 30: Creating PDF Files, How to create PDFs using the MBS Plugins by Christian Schmitz](#)

Chapter 9

Graphics & Pictures

9.1 class Graphics

9.1.1 class Graphics

Platforms: macOS, Linux, Windows, Targets: All.

Function: Extends Xojo's Graphics Class.

9.1.2 Methods

9.1.3 DrawCGPDFDocumentMBS(pdf as CGPDFDocumentMBS, r as CGRectMBS, page as Integer)

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Draws a PDF on the graphics object inside the given rectangle.

Notes: Requires Mac OS X to work.

If destination is scaled page size, you need to also scale the rectangle here.

Seems broken in MacOS Mojave for us with drawing in print context.

See also:

- 9.1.4 DrawCGPDFDocumentMBS(pdf as CGPDFDocumentMBS, r as CGRectMBS, page as Integer, InterpolationQuality as Integer, Antialias as boolean, FontSmoothing as Boolean) 612

9.1.4 DrawCGPDFDocumentMBS(pdf as CGPDFDocumentMBS, r as CGRectMBS, page as Integer, InterpolationQuality as Integer, Antialias as boolean, FontSmoothing as Boolean)

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: Draws a PDF on the graphics object inside the given rectangle.

Notes: Requires Mac OS X to work.

You can pass for interpolation:

- 0 Let the context decide.
- 1 Never interpolate.
- 2 Low quality, fast interpolation.
- 4 Medium quality, slower than kCGInterpolationLow.
- 3 Highest quality, slower than kCGInterpolationMedium.

Set Antialias to true to allow anti aliasing and to false to disallow.

Set FontSmoothing to true to allow font smoothing and false to disallow it.

If destination is scaled page size, you need to also scale the rectangle here.

Seems broken in MacOS Mojave for us with drawing in print context.

Blog Entries

- [MBS Xojo Plugins, version 21.2pr1](#)
- [MBS Xojo Plugins, version 19.2pr1](#)
- [MBS Xojo Plugins, version 19.1pr7](#)
- [MBS Real Studio Plugins, version 12.3pr8](#)
- [MBS Plugins 11.1 Release notes](#)
- [MBS Real Studio Plugins, version 11.1pr10](#)

Xojo Developer Magazine

- [8.5, pages 20 to 22: Printing PDFs, Easy PDF printing on Mac and Windows](#)
- [6.4, page 30: Creating PDF Files, How to create PDFs using the MBS Plugins by Christian Schmitz](#)

See also:

- 9.1.3 DrawCGPDFDocumentMBS(pdf as CGPDFDocumentMBS, r as CGRectMBS, page as Integer)
611

9.2 class Picture

9.2.1 class Picture

Platforms: macOS, Linux, Windows, Targets: All.

Function: Extends Xojo's Picture Class.

9.2.2 Methods

9.2.3 CGColorSpaceMBS as CGColorSpaceMBS

Plugin Version: 13.4, Platform: macOS, Targets: All.

Function: Queries the colorspace profile for a picture.

Example:

```
Var p as new Picture(32,32,32)
Var c as CGColorSpaceMBS = p.CGColorSpaceMBS
MsgBox c.Name
```

Notes: Only for Cocoa targets.

In Version 19.3 changed to default to Generic RGB, which Xojo uses if none is set.

Chapter 10

Printing Mac

10.1 Globals

10.1.1 NewCPMPageFormatMBS as CPMPageFormatMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Creates a new page format object.

Notes: Returns nil on any error.

10.1.2 NewCPMPrintSessionMBS as CPMPrintSessionMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Creates a new print session object.

Example:

```
'get a print session

// print this PDF
Var pathPrinted as FolderItem=GetFolderItem("test.pdf")

Var thePrintSession as CPMPrintSessionMBS = NewCPMPrintSessionMBS
if thePrintSession = nil then Return

'get default page format and print settings and attach it to the print settings
Var thePageFormat as CPMPageFormatMBS = NewCPMPageFormatMBS
Var thePrintSettings as CPMPrintSettingsMBS = NewCPMPrintSettingsMBS
thePrintSession.DefaultPageFormat thePageFormat
thePrintSession.DefaultPrintSettings thePrintSettings
```

```
'show the print dialog
if not thePrintSession.PrintDialog(thePrintSettings,thePageFormat) then return

'open the file which will be printed
Var thePdfDocument as CGPDFDocumentMBS = pathPrinted.OpenAsCGPDFDocumentMBS

' limit page counts to the one we have
Var LastPage as Integer = thePdfDocument.PageCount
if thePrintSettings.LastPage<lastpage then
lastpage=thePrintSettings.LastPage
end if

' you get better progress bar if you tell how many pages will come
thePrintSettings.LastPage=lastpage

'begin the printing
thePrintSession.BeginDocument(thePrintSettings, thePageFormat)

'loop over the number of copies
for currentCopy as Integer = 1 to thePrintSettings.Copies

'loop over the pages
for currentPage as Integer = thePrintSettings.FirstPage to LastPage

'prepage the page
Var PrintRect as CPMRectMBS =thePageFormat.AdjustedPageSize
Var CGRect as CGRectMBS =CGMakeRectMBS(PrintRect.left, PrintRect.top, PrintRect.Width, Print-
Rect.Height)
thePrintSession.BeginPage(thePageFormat, nil)
Var thePrintContext as CGContextMBS = thePrintSession.PageContext
if thePrintContext = Nil then return

'print the page
thePrintContext.DrawCGPDFDocument thePdfDocument, CGRect, currentPage

'end the page
thePrintContext = nil
thePrintSession.EndPage
next

next

'end the printing
thePrintSession.EndDocument
```

Notes: Returns nil on any error.

10.1.3 NewCPMPrintSettingsMBS as CPMPrintSettingsMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Creates a new print settings object.

Notes: Returns nil on any error.

10.2 class CPMLanguageInfoMBS

10.2.1 class CPMLanguageInfoMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: A class for a language info.

Notes: Only a data class.

10.2.2 Properties

10.2.3 Level as String

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The level string of the printer.

Notes: (Read and Write property)

10.2.4 Release as String

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The release string of the printer.

Notes: (Read and Write property)

10.2.5 Version as String

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The version string of the printer.

Notes: (Read and Write property)

10.3 class CPMPageFormatMBS

10.3.1 class CPMPageFormatMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: A class for a page format.

10.3.2 Methods

10.3.3 AdjustedPageSize as CPMRectMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The adjusted page size.

Notes: Lasterror is set.

10.3.4 AdjustedPaperSize as CPMRectMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The adjusted page size.

Notes: Lasterror is set.

10.3.5 Constructor

Plugin Version: 9.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new page format object.

10.3.6 CopySettings(Destination as CPMPageFormatMBS)

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Copies the settings to another pageformat object.

10.3.7 CreateDataRepresentation(Format as Integer = 0) as String

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Returns a data representation of a PMPageFormat object as a data string.

Example:

```
Var p as new CPMPageFormatMBS
p.orientation = p.kPMLandscape
Var d as string = p.CreateDataRepresentation
break // check in debugger
```

Notes: Use CreateWithDataRepresentation to create a CPMPageFormatMBS from a string created by this call.

Format can be kPMDDataFormatXMLDefault, kPMDDataFormatXMLMinimal or kPMDDataFormatXMLCompressed.

10.3.8 CreateWithDataRepresentation(Data as String) as CPMPageFormatMBS

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Creates a PMPageFormat object from a data representation created with CreateDataRepresentation.

Example:

```
Var p as new CPMPageFormatMBS
p.orientation = p.kPMLandscape
Var d as string = p.CreateDataRepresentation
```

```
Var other as CPMPageFormatMBS = CPMPageFormatMBS.CreateWithDataRepresentation(d)
MsgBox str(other.Orientation)
```

Notes: Returns nil in case of error.

10.3.9 PrinterID as String

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Obtains the formatting printer for the pageformat.

Notes: Will either return the formatting printer for the pageformat or will return nil if the pageformat doesn't have that information.

10.3.10 UnadjustedPageSize as CPMRectMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The unadjusted page size.

Notes: Lasterror is set.

10.3.11 UnadjustedPaperSize as CPMRectMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The unadjusted page size.

Notes: Lasterror is set.

10.3.12 Properties

10.3.13 handle as Integer

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The handle for the page format.

Notes: It's a PMPageFormat which you find in CMTypes.h in the Toolbox.
(Read and Write property)

10.3.14 Lasterror as Integer

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The error code from the last command.

Notes: 0 if successful, -1 if function is not available or a required parameter was not set. Else a Mac OS error code.
(Read and Write property)

10.3.15 release as boolean

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: whether the destructor will release the handle.

Notes: (Read and Write property)

10.3.16 Orientation as Integer

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The page orientation.

Example:

```
Var p as new CPMPageFormatMBS
p.orientation = p.kPMLandscape
```

Notes: Lasterror is set.

Possible values:

kPMPortrait	1	
kPMLandscape	2	
kPMReversePortrait	3	will revert to kPortrait for current drivers
kPMReverseLandscape	4	will revert to kLandscape for current drivers

(Read and Write computed property)

10.3.17 Scale as Double

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The page scaling.

Notes: Lasterror is set.

Value is 100 for 100%.

(Read and Write computed property)

10.3.18 Constants

Data Formats

Constant	Value	Description
kPMPDataFormatXMLCompressed	2	kPMPDataFormatXMLCompressed is only compatible and usable with Mac OS X version 10.5 and later. Data in this format can be only be reconstituted into the equivalent printing manager object with the appropriate <code>PMXXXXCreateWithDataRepresentation</code> function. The data representation produced when using <code>kPMPDataFormatXMLCompressed</code> is approximately 20 times smaller than <code>kPMPDataFormatXMLDefault</code> . This format is a good choice when execution on versions of Mac OS X prior to 10.5 is not necessary and the minimum data size is important.
kPMPDataFormatXMLDefault	0	Specifies a data format that is compatible with all Mac OS X versions. Data in this format can be used with the <code>PMUnflattenXXX</code> routines present in all versions of Mac OS X prior to 10.5. However, this data representation is much larger than the more modern data representations described below.
kPMPDataFormatXMLMinimal	1	<code>XMLMinimal</code> is only compatible and usable with Mac OS X version 10.5 and later. Data in this format can be only be reconstituted into the equivalent printing manager object with the appropriate <code>CreateWithDataRepresentation</code> function. The data representation produced when using <code>kPMPDataFormatXMLMinimal</code> is approximately 3-5 times smaller than <code>kPMPDataFormatXMLDefault</code> . This format is a good choice when execution on versions of Mac OS X prior to 10.5 is not necessary and an uncompressed XML representation of the data is needed.

Orientation Constants

Constant	Value	Description
kPMPLandscape	2	Landscape
kPMPPortrait	1	Portrait
kPMPReverseLandscape	4	Reverse Landscape
kPMPReversePortrait	3	Reverse Portrait

10.4 class CPMPrinterMBS

10.4.1 class CPMPrinterMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: A class for a printer.

Blog Entries

- [MBS Real Studio Plugins, version 12.3pr5](#)
- [MonkeyBread Software Releases the MBS REALbasic plug-ins 9.2](#)

10.4.2 Methods

10.4.3 Constructor(name as string)

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: Initializes a new printer object given the printer name.

Notes: Raises exception if lookup fails.

This accepts both CUPS names and the Mac GUI names in printing control panel.

10.4.4 CreateFromPrinterID(PrinterID as String) as CPMPrinterMBS

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Create a PMPrinter instance from the supplied printerID.

Example:

```
Var p as CPMPrinterMBS = CPMPrinterMBS.CreateFromPrinterID("Brother_DCP_8085DN")
MsgBox p.name
```

10.4.5 CreateGenericPrinter as CPMPrinterMBS

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Creates a generic PMPrinter.

10.4.6 CreateLocalPrinterList as CPMPrinterMBS()

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Queries list of local printers.

Example:

```
Var a() as CPMPrinterMBS = CPMPrinterMBS.CreateLocalPrinterList
for each p as CPMPrinterMBS in a
  MsgBox p.Name
next
```

10.4.7 DescriptionURL as string

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Returns the description URL for this printer.

Example:

```
Var session as CPMPrintSessionMBS = NewCPMPrintSessionMBS
Var printer as CPMPrinterMBS = session.CurrentPrinter
```

```
MsgBox printer.DescriptionURL
```

Notes: Should be the path to a PPD file for a laser printer.
LastError is set.

10.4.8 DeviceURI as string

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Hand back the URI of the printer's device.

Example:

```
Var session as CPMPrintSessionMBS = NewCPMPrintSessionMBS
Var printer as CPMPrinterMBS = session.CurrentPrinter
```

```
MsgBox printer.DeviceURI
```

Notes: On success returns a CFURLMBS object describing the printer's device.
LastError is set.

10.4.9 DriverCreator as String

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Returns a 4 letter code for the creator of the printer.

Example:

```
Var session as CPMPrintSessionMBS = NewCPMPrintSessionMBS
Var printer as CPMPrinterMBS = session.CurrentPrinter
```

```
MsgBox "DriverCreator: "+str(printer.DriverCreator)
```

Notes: Lasterror is set.

10.4.10 DriverReleaseInfo as CPMVersionMBS

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Queries printer driver version.

10.4.11 HostName as string

Plugin Version: 9.2, Platform: macOS, Targets: Desktop only.

Function: Hand back the host name of the print server hosting the printer's print queue.

Example:

```
Var session as CPMPrintSessionMBS = NewCPMPrintSessionMBS
Var printer as CPMPrinterMBS = session.CurrentPrinter
```

```
MsgBox "HostName: "+str(printer.HostName)
```

Notes: Lasterror is set.

10.4.12 ID as string

Plugin Version: 9.2, Platform: macOS, Targets: Desktop only.

Function: Returns the ID of the string.

10.4.13 IndexedPrinterResolution(index as Integer) as CPMResolutionMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Returns a printer resolution.

Notes: Lasterror is set.

Index is from 1 to ResolutionCount.

10.4.14 IsDefault as boolean

Plugin Version: 9.2, Platform: macOS, Targets: Desktop only.

Function: Whether this printer is the default printer.

Example:

```
Var session as CPMPrintSessionMBS = NewCPMPrintSessionMBS
Var printer as CPMPrinterMBS = session.CurrentPrinter
```

```
MsgBox "IsDefault: "+str(printer.IsDefault)
```

10.4.15 IsFavorite as boolean

Plugin Version: 9.2, Platform: macOS, Targets: Desktop only.

Function: Return true if the printer is in the user's favorite printer list.

Example:

```
Var session as CPMPrintSessionMBS = NewCPMPrintSessionMBS
Var printer as CPMPrinterMBS = session.CurrentPrinter
```

```
MsgBox "IsFavorite: "+str(printer.IsFavorite)
```

10.4.16 IsPostScriptCapable as boolean

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Can this printer do postscript?

Example:

```
Var session as CPMPrintSessionMBS = NewCPMPrintSessionMBS
Var printer as CPMPrinterMBS = session.CurrentPrinter
```

```
MsgBox "IsPostScriptCapable: "+str(printer.IsPostScriptCapable)
```

Notes: Lasterror is set.

10.4.17 IsPostScriptPrinter as boolean

Plugin Version: 9.2, Platform: macOS, Targets: Desktop only.

Function: If the result is true if the printer is a PostScript printer.

Example:

```
Var session as CPMPrintSessionMBS = NewCPMPrintSessionMBS
Var printer as CPMPrinterMBS = session.CurrentPrinter
```

```
MsgBox "IsPostScriptPrinter: "+str(printer.IsPostScriptPrinter)
```

Notes: A PostScript printer is one whose driver takes PostScript directly.

10.4.18 IsRemote as boolean

Plugin Version: 9.2, Platform: macOS, Targets: Desktop only.

Function: Hand back a boolean indicating whether the printer is hosted by remote print server.

Example:

```
Var session as CPMPrintSessionMBS = NewCPMPrintSessionMBS
Var printer as CPMPrinterMBS = session.CurrentPrinter
```

```
MsgBox "IsRemote: "+str(printer.IsRemote)
```

Notes: If result is true, the print queue represents a printer hosted and managed by a remote print server.

If result is false, the print queue represents a directly connected printer, a network printer, or a remote printer that is locally managed. Consult the queue's device URI to determine the type of connection that is used to communicate with the printer.

Whether a printer is remote is derived from the CUPS printer-type attribute for the print queue.

10.4.19 LanguageInfo as CPMLanguageInfoMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Returns language information about the printer.

Notes: Lasterror is set.

10.4.20 Location as string

Plugin Version: 9.2, Platform: macOS, Targets: Desktop only.

Function: Returns the location of the printer.

10.4.21 MakeAndModelName as string

Plugin Version: 9.2, Platform: macOS, Targets: Desktop only.

Function: A string with the name for the model and the maker.

Notes: Lasterror is set.

Returns "" on any error.

10.4.22 Name as string

Plugin Version: 9.2, Platform: macOS, Targets: Desktop only.

Function: Returns the name of the string.

Example:

```
Var session as CPMPrintSessionMBS = NewCPMPrintSessionMBS
```

```
Var printer as CPMPrinterMBS = session.CurrentPrinter
```

```
MsgBox printer.name
```

10.4.23 ResolutionCount as Integer

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Returns how much resolutions are supported by the printer.

Notes: Lasterror is set.

10.4.24 SetDefault

Plugin Version: 9.2, Platform: macOS, Targets: Desktop only.

Function: Set the default printer for the current user.

Notes: It is not typical for an application to set the current default printer for the user; the printing system itself takes care of managing the default printer.

This function should be used only in rare circumstances.

Requires Mac OS X 10.5.

10.4.25 State as Integer

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Queries printer state.

Example:

```
Var p as CPMPrinterMBS = CPMPrinterMBS.CreateFromPrinterID("Brother_DCP_8085DN")
MsgBox str(p.State)
```

Notes: Can be kPMPrinterIdle, kPMPrinterProcessing or kPMPrinterStopped.

10.4.26 Properties

10.4.27 handle as Integer

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The handle for the printer.

Notes: It's a PMPrinter which you find in CMTypes.h in the Toolbox.
(Read and Write property)

10.4.28 Lasterror as Integer

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The error code from the last command.

Notes: 0 if successful, -1 if function is not available or a required parameter was not set. Else a Mac OS error code.
(Read and Write property)

10.4.29 release as boolean

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: whether the destructor will release the handle.

Notes: (Read and Write property)

10.4.30 Constants

Printer States

Constant	Value	Description
kPMPrinterIdle	3	Printer is idle.
kPMPrinterProcessing	4	Printer is processing.
kPMPrinterStopped	5	Printer is stopped.

10.5 class CPMPrintSessionMBS

10.5.1 class CPMPrintSessionMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: A class for a print session.

Notes: Works only in Carbon Stuff and can only be used with CGContext which works only on Mac OS X.

Blog Entries

- [MBS Real Studio Plugins, version 12.3pr5](#)

10.5.2 Methods

10.5.3 BeginDocument(settings as CPMPrintSettingsMBS, pageformat as CPMPageFormatMBS)

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Begins a new document with the given settings.

Notes: Lasterror is set.

10.5.4 BeginDocumentNoDialog(settings as CPMPrintSettingsMBS, pageformat as CPMPageFormatMBS)

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Begins a new document with the given settings.

Notes: Doesn't use the print progress dialog.

Lasterror is set.

10.5.5 BeginPage(pageformat as CPMPageFormatMBS, rect as CPMRectMBS)

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Begins a new page.

Notes: Lasterror is set.

10.5.6 BeginPageNoDialog(pageformat as CPMPageFormatMBS, rect as CPMRectMBS)

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Begins a new page.

Notes: Doesn't use the print progress dialog.

Lasterror is set.

10.5.7 Constructor

Plugin Version: 9.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new print session object.

10.5.8 CreatePrinterList(list() as string)

Plugin Version: 12.3, Platform: macOS, Targets: Desktop only.

Function: Creates a list of all printers.

Example:

```
Var session as CPMPrintSessionMBS = NewCPMPrintSessionMBS
```

```
Var list() as string
session.CreatePrinterList(list)
```

```
MsgBox join(list, EndOfLine)
```

Notes: List is an array of strings with the names of the printers.

Lasterror is set.

See also:

- 10.5.9 CreatePrinterList(list() as string, byref index as Integer, byref currentprinter as CPMPrinterMBS) 633

10.5.9 CreatePrinterList(list() as string, byref index as Integer, byref currentprinter as CPMPrinterMBS)

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Creates a list of all printers.

Example:

```
Var session as CPMPrintSessionMBS = NewCPMPrintSessionMBS
```

```
Var list() as string
```

```
Var currentprinter as CPMPrinterMBS
```

```
Var index as Integer
```

```
session.CreatePrinterList(list, index, currentprinter)
```

```
MsgBox join(list, EndOfLine)
```

Notes: List is an array of strings with the names of the printers.

Index is the index of the current printer inside this list.

CurrentPrinter is the current printer selected.

Lasterror is set.

See also:

- 10.5.8 CreatePrinterList(list() as string)

633

10.5.10 DefaultPageFormat(pageformat as CPMPageFormatMBS)

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Sets the page format object to the default settings.

Notes: Lasterror is set.

10.5.11 DefaultPrintSettings(printsettings as CPMPrintSettingsMBS)

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Sets the settings object to the default settings.

Notes: Lasterror is set.

10.5.12 EndDocument

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Ends the current document.

Notes: Lasterror is set.

10.5.13 EndDocumentNoDialog

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Ends the current document.

Notes: Doesn't use the print progress dialog.

Lasterror is set.

10.5.14 EndPage

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Ends the current page.

Notes: Lasterror is set.

10.5.15 EndPageNoDialog

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Ends the current page.

Notes: Doesn't use the print progress dialog.

Lasterror is set.

10.5.16 GetDestinationFormat(printsettings as CPMPrintSettingsMBS) as String

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Returns the destination format value.

Notes: Lasterror is set.

10.5.17 GetDestinationLocation(printsettings as CPMPrintSettingsMBS) as String

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Returns the destination location value.

Notes: Lasterror is set.

10.5.18 GetDestinationType(printsettings as CPMPrintSettingsMBS) as Integer

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Returns the destination type value.

Notes: Lasterror is set.

10.5.19 kPMDocumentFormatDefault as String

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: One of the constants for the document format.

10.5.20 kPMDocumentFormatPDF as String

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: One of the constants for the document format.

10.5.21 kPMDocumentFormatPostScript as string

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: One of the constants for the document format.

10.5.22 kPMGraphicsContextCoreGraphics as string

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: One of the constants for the graphics context.

10.5.23 kPMGraphicsContextDefault as string

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: One of the constants for the graphics context.

10.5.24 PageContext as CGContextMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Creates a new CGContext object for the current page.

Notes: Lasterror is set.

10.5.25 PageSetupDialog(pageformat as CPMPageFormatMBS) as boolean

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Shows the page setup dialog.

Notes: Lasterror is set.

10.5.26 PrintDialog(settings as CPMPrintSettingsMBS, pageformat as CPMPageFormatMBS) as boolean

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Shows the print dialog.

Notes: Lasterror is set.

10.5.27 SetDestination(printsettings as CPMPrintSettingsMBS, desttype as Integer, destformat as String, desturl as String)

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Sets a new destination.

Notes: Some destinations so you may need to specify a format, else you can pass "" for the default format. Some destinations require a URL to a file.

Destination type codes:

kPMDestinationInvalid	0
kPMDestinationPrinter	1
kPMDestinationFile	2
kPMDestinationFax	3
kPMDestinationPreview	4

Destformat strings:

Lasterror is set.

kPMDocumentFormatPDF	application/pdf
kPMDocumentFormatPICT	application/vnd.apple.printing-pict
kPMDocumentFormatPICTPS	application/vnd.apple.printing-pict-ps
kPMDocumentFormatPostScript	application/postscript

10.5.28 UseSheets(docWindow as window)

Plugin Version: 9.3, Platform: macOS, Targets: Desktop only.

Function: Specifies that a printing dialog should be displayed as a sheet.

Notes: docWindow: The window to which the sheet dialog should be attached.

Lasterror is set. (-1 for the docWindow parameter being nil)

10.5.29 ValidatePageFormat(pageformat as CPMPageFormatMBS) as boolean

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Validates the page format.

Notes: Lasterror is set.

10.5.30 ValidatePrintSettings(printsettings as CPMPrintSettingsMBS) as boolean

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Validates the print settings.

Notes: Lasterror is set.

10.5.31 Properties

10.5.32 handle as Integer

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The handle for the print session.

Notes: It's a PMPrintSession which you find in CMTypes.h in the Toolbox.
(Read and Write property)

10.5.33 Lasterror as Integer

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The error code from the last command.

Notes: 0 if successful, -1 if function is not available or a required parameter was not set. Else a Mac OS error code.

(Read and Write property)

10.5.34 release as boolean

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: whether the destructor will release the handle.

Notes: (Read and Write property)

10.5.35 SheetTarget as Window

Plugin Version: 9.3, Platform: macOS, Targets: Desktop only.

Function: The window set by UseSheets.

Notes: This property is to reference the window used for the sheets so Xojo does not destroy the window too early.

(Read and Write property)

10.5.36 CurrentPrinter as CPMPrinterMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The current printer for this session.

Notes: Lasterror is set.

Setting this can only be done on Mac OS X.

(Read and Write computed property)

10.5.37 CurrentPrinterName as string

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: The current printer for this session by name.

Notes: Lasterror is set.

(Read and Write computed property)

10.5.38 Events

10.5.39 SheetDone(WindowHandle as Integer, accepted as boolean)

Plugin Version: 9.3, Platform: macOS, Targets: .

Function: This event is called for a finished print sheet.

Notes: Accepted is false if the cancel button was used. True if OK was clicked.

10.5.40 Constants

Destination Types

Constant	Value	Description
kPMDestinationFax	3	Fax
kPMDestinationFile	2	File
kPMDestinationInvalid	0	Invalid
kPMDestinationPreview	4	Preview
kPMDestinationPrinter	1	Printer
kPMDestinationProcessPDF	5	Process to PDF
kPMDestinationTypeDefault	1	Default is printer.

10.6 class CPMPrintSettingsMBS

10.6.1 class CPMPrintSettingsMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: A class for a print session.

10.6.2 Methods

10.6.3 Constructor

Plugin Version: 9.3, Platform: macOS, Targets: Desktop only.

Function: Creates a new print settings object.

10.6.4 CopyPrintSettings(dest as CPMPrintSettingsMBS)

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Copies the settings to the other settings.

10.6.5 CreateDataRepresentation(Format as Integer = 0) as String

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Returns a data representation of a PMPageFormat object as a data string.

Example:

```
Var p as new CPMPrintSettingsMBS
p.Collate = true
p.Duplex = p.kPMDuplexTumble
Var data as string = p.CreateDataRepresentation
Break // check in debugger
```

10.6.6 CreateWithDataRepresentation(Data as String) as CPMPrintSettingsMBS

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Creates a PMPageFormat object from a data representation created with CreateDataRepresentation.

10.6.7 Dictionary as Dictionary

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Represent these print settings as a dictionary.

10.6.8 GetPageRange(byref minPage as UInt32, byref maxPage as UInt32)

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Queries the page range to print.

Notes: Obtains the valid range of pages that can be printed.

The default page range is 1 - (all pages). The page range is something that is set by the application. It is NOT the first and last page to print. It serves as limits for setting the first and last page.

10.6.9 Keys as String()

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Queries list of keys.

Example:

```
Var p as new CPMPrintSettingsMBS
p.Collate = true
p.Duplex = p.kPMDuplexTumble
Var keys() as string = p.keys
Break // check in debugger
```

10.6.10 SetPageRange(minPage as UInt32, maxPage as UInt32)

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: Sets the page range to print.

10.6.11 Properties

10.6.12 handle as Integer

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The handle for the print settings.

Notes: It's a PMPrintSettings which you find in CMTypes.h in the Toolbox.
(Read and Write property)

10.6.13 Lasterror as Integer

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The error code from the last command.

Notes: 0 if successful, -1 if function is not available or a required parameter was not set. Else a Mac OS error code.

(Read and Write property)

10.6.14 release as boolean

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: whether the destructor will release the handle.

Notes: (Read and Write property)

10.6.15 Collate as boolean

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: The collate setting.

Notes: (Read and Write computed property)

10.6.16 Copies as Integer

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The number of copies that the user requests to be printed.

Notes: Lasterror is set.

(Read and Write computed property)

10.6.17 Duplex as Integer

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: The duplex setting.

Notes: (Read and Write computed property)

10.6.18 FirstPage as Integer

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The number of the first page to print.

Notes: Lasterror is set.

(Read and Write computed property)

10.6.19 JobName as String

Plugin Version: 12.1, Platform: macOS, Targets: Desktop only.

Function: Specifies the name of a print job.

Notes: Lasterror is set.

(Read and Write computed property)

10.6.20 LastPage as Integer

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The number of the last page to print.

Example:

```
'get a print session

// print this PDF
Var pathPrinted as FolderItem=GetFolderItem("test.pdf")

Var thePrintSession as CPMPrintSessionMBS = NewCPMPrintSessionMBS
if thePrintSession = nil then Return

'get default page format and print settings and attach it to the print settings
Var thePageFormat as CPMPageFormatMBS = NewCPMPageFormatMBS
Var thePrintSettings as CPMPrintSettingsMBS = NewCPMPrintSettingsMBS
thePrintSession.DefaultPageFormat thePageFormat
thePrintSession.DefaultPrintSettings thePrintSettings

'show the print dialog
if not thePrintSession.PrintDialog(thePrintSettings,thePageFormat) then return
```

```

'open the file which will be printed
Var thePdfDocument as CGPDFDocumentMBS = pathPrinted.OpenAsCGPDFDocumentMBS

' limit page counts to the one we have
Var LastPage as Integer = thePdfDocument.PageCount
if thePrintSettings.LastPage<lastpage then
lastpage=thePrintSettings.LastPage
end if

' you get better progress bar if you tell how many pages will come
thePrintSettings.LastPage=lastpage

'begin the printing
thePrintSession.BeginDocument(thePrintSettings, thePageFormat)

'loop over the number of copies
for currentCopy as Integer = 1 to thePrintSettings.Copies

'loop over the pages
for currentPage as Integer = thePrintSettings.FirstPage to LastPage

'prepage the page
Var PrintRect as CPMRectMBS =thePageFormat.AdjustedPageSize
Var CGRect as CGRectMBS =CGMakeRectMBS(PrintRect.left, PrintRect.top, PrintRect.Width, Print-
Rect.Height)
thePrintSession.BeginPage(thePageFormat, nil)
Var thePrintContext as CGContextMBS = thePrintSession.PageContext
if thePrintContext = Nil then return

'print the page
thePrintContext.DrawCGPDFDocument thePdfDocument, CGRect, currentPage

'end the page
thePrintContext = nil
thePrintSession.EndPage
next

next

'end the printing
thePrintSession.EndDocument

```

Notes: Lasterror is set.
(Read and Write computed property)

10.6.21 Value(key as String) as Variant

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: The value in settings.

Example:

```
Var p as new CPMPrintSettingsMBS
p.Collate = true
Var value as Variant = p.value("com.apple.print.PrintSettings.PMCopyCollate")
Break // check in debugger
```

Notes: (Read and Write computed property)

10.6.22 Constants

Duplex Modes

Constant	Value	Description
kPMDuplexNone	1	Duplex off.
kPMDuplexNoTumble	2	Duplex with no tumble.
kPMDuplexTumble	3	Duplex with tumble.
kPMSimplexTumble	4	No duplex.

10.7 class CPMRectMBS

10.7.1 class CPMRectMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: A class for a rectangle.

Notes: Only a data class.

10.7.2 Properties

10.7.3 Bottom as Double

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The bottom distance to the border.

Notes: (Read and Write property)

10.7.4 Height as Double

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The height of the rectangle.

Notes: Setting the height changes the bottom property.

(Read and Write property)

10.7.5 Left as Double

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The left distance to the border.

Notes: (Read and Write property)

10.7.6 Right as Double

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The right distance to the border.

Notes: (Read and Write property)

10.7.7 Top as Double

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The top distance to the border.

Notes: (Read and Write property)

10.7.8 Width as Double

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The width of the rectangle.

Notes: Setting the height changes the right property.
(Read and Write property)

10.8 class CPMResolutionMBS

10.8.1 class CPMResolutionMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop, Console & Web.

Function: A class for a resolution.

Notes: Only a data class.

10.8.2 Properties

10.8.3 Horizontal as Double

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The horizontal resolution.

Notes: (Read and Write property)

10.8.4 Vertical as Double

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: The vertical resolution.

Notes: (Read and Write property)

10.9 class CPMVersionMBS

10.9.1 class CPMVersionMBS

Plugin Version: 15.1, Platform: macOS, Targets: Desktop, Console & Web.

Function: The class for version information.

10.9.2 Properties

10.9.3 CountryCode as Integer

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: The country code.

Notes: (Read and Write property)

10.9.4 LongVersion as String

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: The long version string.

Notes: (Read and Write property)

10.9.5 ShortVersion as String

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: The short version string.

Notes: (Read and Write property)

10.9.6 VersionMajor as Integer

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: The major version number.

Notes: (Read and Write property)

10.9.7 VersionMinor as Integer

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: The minor version number.

Notes: (Read and Write property)

10.9.8 VersionRevision as Integer

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: The reversion number.

Notes: (Read and Write property)

10.9.9 VersionStage as Integer

Plugin Version: 15.1, Platform: macOS, Targets: Desktop only.

Function: The version stage.

Notes: (Read and Write property)

Chapter 11

Process

11.1 class Application

11.1.1 class Application

Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: Extends the Application class inside Xojo.

11.1.2 Methods

11.1.3 OverlayApplicationDockTileImageMBS(pic as CGImageMBS) as boolean

Plugin Version: 2.6, Platform: macOS, Targets: Desktop only.

Function: Adds a picture on top of the application dock title.

Example:

```
static n as Integer // global property
Var p as picture
Var m as picture
Var g as graphics
Var s as string
Var w,h,x as Integer

n=n+1 // add one
s=str(n)
m=New Picture(128,128,32)
p=New Picture(128,128,32)
```

```

g=p.graphics
g.textsize=24
w=g.stringwidth(s)
h=g.stringheight(s,0)
x=125-w-10
g.foreColor=rgb(255,0,0)
g.fillRoundRect x,10,10+w,10+h,20,20
g.foreColor=rgb(0,0,0)
g.drawString s,x+5,15+g.textascent

```

```

g=m.graphics
g.foreColor=rgb(0,0,0)
g.fillRoundRect x,10,10+w,10+h,20,20

```

```

if app.OverlayApplicationDockTileImageMBS(CGCreateImageMBS(p,m)) then
end if

```

Notes: Changed in MBS Plugin 2.7 to use CGImages directly.
 Added support for 64-bit in plugin version 16.0.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 15.5pr2](#)

11.1.4 RestoreApplicationDockTileImageMBS as boolean

Plugin Version: 2.6, Platform: macOS, Targets: Desktop only.

Function: Restores the application dock tile image.

Example:

```

if app.RestoreApplicationDockTileImageMBS then
// ok
end if

```

Notes: Added support for 64-bit in plugin version 16.0.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 15.5pr2](#)

11.1.5 SetApplicationDockTileImageMBS(pic as CGImageMBS) as boolean

Plugin Version: 2.6, Platform: macOS, Targets: Desktop only.

Function: Sets the application dock tile image.

Example:

```
Var p as CGImageMBS = CGImageMBS.CreateImage(LogoMBS(500))  
  
if app.SetApplicationDockTileImageMBS(p) then  
// ok  
end if
```

Notes: Changed in MBS Plugin 2.7 to use CGImages directly.
Added support for 64-bit in plugin version 16.0.

Blog Entries

- [MBS Xojo / Real Studio Plugins, version 15.5pr2](#)

Chapter 12

Window

12.1 class DesktopWindow

12.1.1 class DesktopWindow

Plugin Version: 21.5, Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: Extends Xojo's Window Class.

Example:

```
window1.HasNoTitleBarMBS = true
```

Notes: In Xojo 2005 and newer you need to use self. in front of the method as the propertyname alone is not accepted.

12.1.2 Methods

12.1.3 CGContextMBS as CGContextMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Creates a new CGContextMBS object for the given Window.

Example:

```
Var CGContext as CGContextMBS
Var OverlayPic as Picture=LogoMBS(500)
Var contentpicture as CImageMBS= CGCreateImageMBS(OverlayPic)
```

```
CGContext=window1.CGContextMBS
```

```
CGContext.ClearRect CGMakeRectMBS(0, 0, OverlayPic.width, OverlayPic.height)
CGContext.DrawPicture contentpicture, CGMakeRectMBS(0, 0, OverlayPic.width, OverlayPic.height)
```

Notes: Requires Mac OS X to work.

Please make sure the CGContextMBS object is released in the paint event if you use it there, so all the drawings are flushed. Not releasing it may be visible in missing statictext or other strange graphics effects.

But on a Cocoa window the context is not always available, so in the window paint event, you can context from Graphics object.

12.1.4 Properties

12.1.5 CGColorSpaceMBS as CGColorSpaceMBS

Plugin Version: 21.5, Platform: macOS, Targets: Desktop only.

Function: Get or set the colorspace profile for this window.

Example:

```
Var c as CGColorSpaceMBS = window1.CGColorSpaceMBS
```

```
if c<>Nil then
  MsgBox str(c.NumberOfComponents)
else
  MsgBox "nil"
end if
```

Notes: Requires MacOS 10.6 or newer.

On Cocoa, the NSColorSpace is queried and the matching CGColorspace is returned. For setting a NSColorSpace is created for the given CGColorspace.

In Carbon applications, the CoreGraphics color space is passed directly.

(Read and Write computed property)

12.2 class Window

12.2.1 class Window

Platforms: macOS, Linux, Windows, Targets: Desktop only.

Function: Extends Xojo's Window Class.

Example:

```
window1.HasNoTitleBarMBS = true
```

Notes: In Xojo 2005 and newer you need to use self. in front of the method as the propertyname alone is not accepted.

12.2.2 Methods

12.2.3 CGContextMBS as CGContextMBS

Plugin Version: 2.7, Platform: macOS, Targets: Desktop only.

Function: Creates a new CGContextMBS object for the given Window.

Example:

```
Var CGContext as CGContextMBS
```

```
Var OverlayPic as Picture=LogoMBS(500)
```

```
Var contentpicture as CGImageMBS= CGCreateImageMBS(OverlayPic)
```

```
CGContext=window1.CGContextMBS
```

```
CGContext.ClearRect CGMakeRectMBS(0, 0, OverlayPic.width, OverlayPic.height)
```

```
CGContext.DrawPicture contentpicture, CGMakeRectMBS(0, 0, OverlayPic.width, OverlayPic.height)
```

Notes: Requires Mac OS X to work.

Please make sure the CGContextMBS object is released in the paint event if you use it there, so all the drawings are flushed. Not releasing it may be visible in missing statictext or other strange graphics effects.

But on a Cocoa window the context is not always available, so in the window paint event, you can context from Graphics object.

12.2.4 Properties

12.2.5 CGColorSpaceMBS as CGColorSpaceMBS

Plugin Version: 10.4, Platform: macOS, Targets: Desktop only.

Function: Get or set the colorspace profile for this window.

Example:

```
Var c as CGColorSpaceMBS = window1.CGColorSpaceMBS
```

```
if c<>Nil then
  MsgBox str(c.NumberOfComponents)
else
  MsgBox "nil"
end if
```

Notes: Requires MacOS 10.6 or newer.

On Cocoa, the NSColorSpace is queried and the matching CGColorspace is returned. For setting a NSColorSpace is created for the given CGColorspace.

In Carbon applications, the CoreGraphics color space is passed directly.

(Read and Write computed property)

Chapter 13

List of Questions in the FAQ

- 14.0.1 Can anyone help me convert seconds to time in this format hh:mm:ss? 671
- 14.0.2 Do you have plugins for Android? 672
- 14.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 672
- 14.0.4 How to catch delete key? 673
- 14.0.5 How to convert cmyk to rgb? 674
- 14.0.6 How to delete a folder? 675
- 14.0.7 How to detect if CPU is 64bit processor? 676
- 14.0.8 How to query variant type string for a variant? 677
- 14.0.9 How to refresh a htmlviewer on Windows? 678
- 14.0.10 Is there an example for vector graphics in Xojo? 679
- 14.0.11 Picture functions do not preserve resolution values? 680
- 14.0.12 A toolbox call needs a rect - how do I give it one? 680
- 14.0.13 API client not supported? 680
- 14.0.14 Can I access Access Database with Java classes? 681
- 14.0.15 Can I create PDF from Xojo Report using DynaPDF? 682
- 14.0.16 Can I use AppleScripts in a web application? 682
- 14.0.17 Can I use graphics class with DynaPDF? 682
- 14.0.18 Can I use sockets on a web application? 683
- 14.0.19 Can I use your ChartDirector plugin on a web application? 683

- 14.0.20 Can I use your DynaPDF plugin on a web application? 684
- 14.0.21 Can I use your plugin controls on a web application? 685
- 14.0.22 Can you get an unique machine ID? 685
- 14.0.23 ChartDirector: Alignment Specification 685
- 14.0.24 ChartDirector: Color Specification 686
- 14.0.25 ChartDirector: Font Specification 690
- 14.0.26 ChartDirector: Mark Up Language 693
- 14.0.27 ChartDirector: Parameter Substitution and Formatting 697
- 14.0.28 ChartDirector: Shape Specification 701
- 14.0.29 Copy styled text? 702
- 14.0.30 Do you have code to validate a credit card number? 703
- 14.0.31 Do you have plugins for X-Rite EyeOne, eXact or i1Pro? 703
- 14.0.32 Does SQL Plugin handle stored procedures with multiple result sets? 704
- 14.0.33 Does the plugin home home? 704
- 14.0.34 folderitem.absolutePath is limited to 255 chars. How can I get longer ones? 704
- 14.0.35 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? 705
- 14.0.36 How about Plugin support for older OS X? 706
- 14.0.37 How can I detect whether an Intel CPU is a 64bit CPU? 706
- 14.0.38 How can I disable the close box of a window on Windows? 707
- 14.0.39 How can I get all the environment variables from Windows? 708
- 14.0.40 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? 709
- 14.0.41 How can I get text from a PDF? 709
- 14.0.42 How can I get text from a Word Document? 709
- 14.0.43 How can I get the item string for a given file creator? 709
- 14.0.44 How can I launch an app using it's creator code? 711
- 14.0.45 How can I learn what shared libraries are required by a plugin on Linux? 711
- 14.0.46 How can I validate an email address? 712
- 14.0.47 How do I decode correctly an email subject? 713

	663
• 14.0.48 How do I enable/disable a single tab in a tabpanel?	714
• 14.0.49 How do I find the root volume for a file?	714
• 14.0.50 How do I get the current languages list?	715
• 14.0.51 How do I get the Mac OS Version?	716
• 14.0.52 How do I get the printer name?	717
• 14.0.53 How do I make a metal window if RB does not allow me this?	717
• 14.0.54 How do I make a smooth color transition?	718
• 14.0.55 How do I read the applications in the dock app?	719
• 14.0.56 How do I truncate a file?	720
• 14.0.57 How do update a Finder's windows after changing some files?	720
• 14.0.58 How to access a USB device directly?	720
• 14.0.59 How to add icon to file on Mac?	721
• 14.0.60 How to ask the Mac for the Name of the Machine?	721
• 14.0.61 How to automatically enable retina in my apps?	721
• 14.0.62 How to avoid leaks with Cocoa functions?	722
• 14.0.63 How to avoid trouble connecting to oracle database with SQL Plugin?	722
• 14.0.64 How to avoid ___NSAutoreleaseNoPool console messages in threads?	723
• 14.0.65 How to bring app to front?	723
• 14.0.66 How to bring my application to front?	724
• 14.0.67 How to catch Control-C on Mac or Linux in a console app?	724
• 14.0.68 How to change name of application menu?	725
• 14.0.69 How to change the name in the menubar of my app on Mac OS X?	725
• 14.0.70 How to check if a folder/directory has subfolders?	726
• 14.0.71 How to check if Macbook runs on battery or AC power?	726
• 14.0.72 How to check if Microsoft Outlook is installed?	727
• 14.0.73 How to check on Mac OS which country or language is currently selected?	728
• 14.0.74 How to code sign my app with plugins?	728
• 14.0.75 How to collapse a window?	729
• 14.0.76 How to compare two pictures?	729

- 14.0.77 How to compile PHP library? 731
- 14.0.78 How to convert a `BrowserType` to a `String` with `WebSession.Browser`? 732
- 14.0.79 How to convert a `EngineType` to a `String` with `WebSession.Engine`? 733
- 14.0.80 How to convert a `PlatformType` to a `String` with `WebSession.Platform`? 733
- 14.0.81 How to convert a text to iso-8859-1 using the `TextEncoder`? 734
- 14.0.82 How to convert `ChartTime` back to Xojo date? 735
- 14.0.83 How to convert line endings in text files? 735
- 14.0.84 How to convert picture to string and back? 736
- 14.0.85 How to copy an array? 737
- 14.0.86 How to copy an dictionary? 737
- 14.0.87 How to copy parts of a movie to another one? 738
- 14.0.88 How to create a birthday like calendar event? 738
- 14.0.89 How to create a GUID? 739
- 14.0.90 How to create a Mac picture clip file? 739
- 14.0.91 How to create a PDF file in Xojo? 740
- 14.0.92 How to create `EmailAttachment` for PDF Data in memory? 740
- 14.0.93 How to create PDF for image files? 741
- 14.0.94 How to CURL Options translate to Plugin Calls? 742
- 14.0.95 How to delete file with ftp and curl plugin? 743
- 14.0.96 How to detect display resolution changed? 743
- 14.0.97 How to detect retina? 744
- 14.0.98 How to disable force quit? 744
- 14.0.99 How to disable the error dialogs from Internet Explorer on javascript errors? 744
- 14.0.100 How to display a PDF file in Xojo? 744
- 14.0.101 How to do a lottery in RB? 745
- 14.0.102 How to do an asycron DNS lookup? 746
- 14.0.103 How to draw a dashed pattern line? 746
- 14.0.104 How to draw a nice antialiased line? 747
- 14.0.105 How to dump java class interface? 748

	665
• 14.0.106 How to duplicate a picture with mask or alpha channel?	748
• 14.0.107 How to enable assistive devices?	749
• 14.0.108 How to encrypt a file with Blowfish?	750
• 14.0.109 How to extract text from HTML?	750
• 14.0.110 How to find empty folders in a folder?	751
• 14.0.111 How to find iTunes on a Mac OS X machine fast?	751
• 14.0.112 How to find network interface for a socket by it's name?	752
• 14.0.113 How to find version of Microsoft Word?	753
• 14.0.114 How to fix CURL error 60/53 on connecting to server?	753
• 14.0.115 How to format double with n digits?	754
• 14.0.116 How to get a time converted to user time zone in a web app?	754
• 14.0.117 How to get an handle to the frontmost window on Windows?	755
• 14.0.118 How to get CFAbsoluteTime from date?	755
• 14.0.119 How to get client IP address on web app?	756
• 14.0.120 How to get fonts to load in charts on Linux?	756
• 14.0.121 How to get fonts to load in DynaPDF on Linux?	757
• 14.0.122 How to get GMT time and back?	757
• 14.0.123 How to get good crash reports?	758
• 14.0.124 How to get list of all threads?	758
• 14.0.125 How to get parameters from webpage URL in Xojo Web Edition?	759
• 14.0.126 How to get the color for disabled textcolor?	759
• 14.0.127 How to get the current free stack space?	760
• 14.0.128 How to get the current timezone?	760
• 14.0.129 How to get the current window title?	762
• 14.0.130 How to get the cursor blink interval time?	763
• 14.0.131 How to get the list of the current selected files in the Finder?	763
• 14.0.132 How to get the Mac OS system version?	764
• 14.0.133 How to get the Mac OS Version using System.Gestalt?	764
• 14.0.134 How to get the screensize excluding the task bar?	765

- 14.0.135 How to get the size of the frontmost window on Windows? 765
- 14.0.136 How to get the source code of a HTMLViewer? 766
- 14.0.137 How to get Xojo apps running Linux? 766
- 14.0.138 How to handle really huge images with GraphicsMagick or ImageMagick? 767
- 14.0.139 How to handle tab key for editable cells in listbox? 767
- 14.0.140 How to hard link MapKit framework? 768
- 14.0.141 How to have a PDF downloaded to the user in a web application? 769
- 14.0.142 How to hide all applications except mine? 769
- 14.0.143 How to hide script errors in HTMLViewer on Windows? 770
- 14.0.144 How to hide the grid/background/border in ChartDirector? 770
- 14.0.145 How to hide the mouse cursor on Mac? 770
- 14.0.146 How to insert image to NSTextView or TextArea? 770
- 14.0.147 How to jump to an anchor in a htmlviewer? 771
- 14.0.148 How to keep a movieplayer unclickable? 772
- 14.0.149 How to keep my web app from using 100% CPU time? 772
- 14.0.150 How to kill a process by name? 772
- 14.0.151 How to know how many CPUs are present? 773
- 14.0.152 How to know the calling function? 773
- 14.0.153 How to launch an app using it's creator code? 774
- 14.0.154 How to launch disc utility? 774
- 14.0.155 How to make a lot of changes to a REAL SQL Database faster? 775
- 14.0.156 How to make a NSImage object for my retina enabled app? 775
- 14.0.157 How to make a window borderless on Windows? 776
- 14.0.158 How to make an alias using AppleEvents? 776
- 14.0.159 How to make AppleScripts much faster? 777
- 14.0.160 How to make double clicks on a canvas? 777
- 14.0.161 How to make my Mac not sleeping? 779
- 14.0.162 How to make my own registration code scheme? 780
- 14.0.163 How to make small controls on Mac OS X? 780

	667
• 14.0.164 How to mark my Mac app as background only?	781
• 14.0.165 How to move a file or folder to trash?	781
• 14.0.166 How to move an application to the front using the creator code?	782
• 14.0.167 How to move file with ftp and curl plugin?	783
• 14.0.168 How to normalize string on Mac?	783
• 14.0.169 How to obscure the mouse cursor on Mac?	784
• 14.0.170 How to open icon file on Mac?	784
• 14.0.171 How to open PDF in acrobat reader?	784
• 14.0.172 How to open printer preferences on Mac?	785
• 14.0.173 How to open special characters panel on Mac?	785
• 14.0.174 How to optimize picture loading in Web Edition?	786
• 14.0.175 How to parse XML?	786
• 14.0.176 How to play audio in a web app?	787
• 14.0.177 How to pretty print xml?	788
• 14.0.178 How to print to PDF?	788
• 14.0.179 How to query Spotlight's Last Open Date for a file?	789
• 14.0.180 How to quit windows?	789
• 14.0.181 How to read a CSV file correctly?	790
• 14.0.182 How to read the command line on windows?	790
• 14.0.183 How to render PDF pages with PDF Kit?	791
• 14.0.184 How to restart a Mac?	792
• 14.0.185 How to resume ftp upload with curl plugin?	792
• 14.0.186 How to rotate a PDF page with CoreGraphics?	792
• 14.0.187 How to rotate image with CoreImage?	793
• 14.0.188 How to run a 32 bit application on a 64 bit Linux?	794
• 14.0.189 How to save HTMLViewer to PDF with landscape orientation?	794
• 14.0.190 How to save RTFD?	795
• 14.0.191 How to save RTFD?	795
• 14.0.192 How to scale a picture proportionally with mask?	796

- 14.0.193 How to scale a picture proportionally? 797
- 14.0.194 How to scale/resize a CIImageMBS? 798
- 14.0.195 How to scale/resize a picture? 798
- 14.0.196 How to search with regex and use unicode codepoints? 799
- 14.0.197 How to see if a file is invisible for Mac OS X? 800
- 14.0.198 How to set cache size for SQLite or REALSQLDatabase? 800
- 14.0.199 How to set the modified dot in the window? 801
- 14.0.200 How to show a PDF file to the user in a Web Application? 801
- 14.0.201 How to show Keyboard Viewer programmatically? 801
- 14.0.202 How to show the mouse cursor on Mac? 802
- 14.0.203 How to shutdown a Mac? 803
- 14.0.204 How to sleep a Mac? 803
- 14.0.205 How to speed up rasterizer for displaying PDFs with DynaPDF? 803
- 14.0.206 How to use PDFLib in my RB application? 804
- 14.0.207 How to use quotes in a string? 804
- 14.0.208 How to use Sybase in Web App? 804
- 14.0.209 How to use the Application Support folder? 805
- 14.0.210 How to use the IOPMCopyScheduledPowerEvents function in Xojo? 805
- 14.0.211 How to validate a GUID? 808
- 14.0.212 How to walk a folder hierarchie non recursively? 808
- 14.0.213 I got this error: PropVal, QDPictMBS.Name (property value), Type mismatch error. Expected CGDataProviderMBS, but got Variant, Name:QDPictMBS 809
- 14.0.214 I registered the MBS Plugins in my application, but later the registration dialog is shown. 809
- 14.0.215 I want to accept Drag & Drop from iTunes 810
- 14.0.216 I'm drawing into a listbox but don't see something. 812
- 14.0.217 I'm searching for a method or so to move a window from position x.y to somewhere else on the screen. 812
- 14.0.218 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? 813
- 14.0.219 Is the fn key on a powerbook keyboard down? 813

	669
• 14.0.220 Is there a case sensitive Dictionary?	813
• 14.0.221 Is there a way to use the MBS plugin to get only the visible item and folder count on a volume?	814
• 14.0.222 Is there an easy way I can launch the Displays preferences panel?	814
• 14.0.223 List of Windows Error codes?	815
• 14.0.224 Midi latency on Windows problem?	815
• 14.0.225 My Xojo Web App does not launch. Why?	815
• 14.0.226 SQLiteDatabase not initialized error?	816
• 14.0.227 Textconverter returns only the first x characters. Why?	816
• 14.0.228 The type translation between CoreFoundation/Foundation and Xojo data types.	817
• 14.0.229 Uploaded my web app with FTP, but it does not run on the server!	819
• 14.0.230 What classes to use for hotkeys?	819
• 14.0.231 What do I need for Linux to get picture functions working?	820
• 14.0.232 What does the NAN code mean?	820
• 14.0.233 What font is used as a 'small font' in typical Mac OS X apps?	821
• 14.0.234 What is last plugin version to run on Mac OS X 10.4?	821
• 14.0.235 What is last plugin version to run on PPC?	822
• 14.0.236 What is last version of the plugins for macOS 32-bit?	822
• 14.0.237 What is the difference between Timer and WebTimer?	822
• 14.0.238 What is the list of Excel functions?	822
• 14.0.239 What is the replacement for PluginMBS?	823
• 14.0.240 What to do on Xojo reporting a conflict?	824
• 14.0.241 What to do with a NSImageCacheException?	824
• 14.0.242 What to do with MySQL Error 2014?	824
• 14.0.243 What to do with SQL Plugin reporting Malformed string as error?	824
• 14.0.244 Where is CGGetActiveDisplayListMBS?	825
• 14.0.245 Where is CGGetDisplaysWithPointMBS?	825
• 14.0.246 Where is CGGetDisplaysWithRectMBS?	825
• 14.0.247 Where is CGGetOnlineDisplayListMBS?	825
• 14.0.248 Where is GetObjectClassNameMBS?	825

- 14.0.249 Where is NetworkAvailableMBS? 826
- 14.0.250 Where is StringHeight function in DynaPDF? 826
- 14.0.251 Where is XLSDocumentMBS class? 826
- 14.0.252 Where to get information about file formats? 827
- 14.0.253 Where to register creator code for my application? 827
- 14.0.254 Which Mac OS X frameworks are 64bit only? 827
- 14.0.255 Which plugins are 64bit only? 828
- 14.0.256 Why application doesn't launch because of a missing ddraw.dll!? 828
- 14.0.257 Why application doesn't launch because of a missing shlwapi.dll!? 828
- 14.0.258 Why do I hear a beep on keydown? 828
- 14.0.259 Why does folderitem.item return nil? 828
- 14.0.260 Why doesn't showurl work? 829
- 14.0.261 Why don't the picture functions not work on Linux? 829
- 14.0.262 Why have I no values in my chart? 829
- 14.0.263 Will application size increase with using plugins? 829
- 14.0.264 XLS: Custom format string guidelines 830
- 14.0.265 Xojo doesn't work with your plugins on Windows 98. 831
- 14.0.266 Xojo or my RB application itself crashes on launch on Mac OS Classic. Why? 831

Chapter 14

The FAQ

14.0.1 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"
```

```
Var hours, minutes, seconds as Integer
Var 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)

14.0.2 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.

14.0.3 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
Var theColor As NSColorMBS
If active Then
theColor = NSColorMBS.alternateSelectedControlColor
Else
theColor = NSColorMBS.secondarySelectedControlColor
End If

```

```

Var rgbColor As NSColorMBS = theColor.colorUsingColorSpaceName(NSColorSpaceMBS.NSCalibratedRGB-
BColorSpace)

```

```

If rgbColor <>Nil Then
Var red as Integer = rgbColor.redComponent * 255.0
Var green as Integer = rgbColor.greenComponent * 255.0
Var 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:

- 14.0.4 How to catch delete key? 673
- 14.0.5 How to convert cmyk to rgb? 674
- 14.0.6 How to delete a folder? 675
- 14.0.7 How to detect if CPU is 64bit processor? 676
- 14.0.8 How to query variant type string for a variant? 677
- 14.0.9 How to refresh a htmlviewer on Windows? 678

14.0.4 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:

- 14.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 672
- 14.0.5 How to convert cmyk to rgb? 674
- 14.0.6 How to delete a folder? 675

- 14.0.7 How to detect if CPU is 64bit processor? 676
- 14.0.8 How to query variant type string for a variant? 677
- 14.0.9 How to refresh a htmlviewer on Windows? 678

14.0.5 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
```

```
Var color_RGB as color
Var 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:

- 14.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 675
- 14.0.4 How to catch delete key? 672
- 14.0.6 How to delete a folder? 673
- 14.0.7 How to detect if CPU is 64bit processor? 675
- 14.0.8 How to query variant type string for a variant? 676
- 14.0.9 How to refresh a htmlviewer on Windows? 677

14.0.6 How to delete a folder?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: The following is the code that deletes a folder recursively.

Example:

```
Sub deletefolder(f as folderitem)
```

```
Var 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
```

```
Var 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:

- 14.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 672
- 14.0.4 How to catch delete key? 673
- 14.0.5 How to convert cmyk to rgb? 674
- 14.0.7 How to detect if CPU is 64bit processor? 676
- 14.0.8 How to query variant type string for a variant? 677
- 14.0.9 How to refresh a htmlviewer on Windows? 678

14.0.7 How to detect if CPU is 64bit processor?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Via CPUID you can ask CPU:

Example:

```
Var 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:

- 14.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 672
- 14.0.4 How to catch delete key? 673
- 14.0.5 How to convert cmyk to rgb? 674
- 14.0.6 How to delete a folder? 675
- 14.0.8 How to query variant type string for a variant? 677
- 14.0.9 How to refresh a htmlviewer on Windows? 678

14.0.8 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
Var type As Integer = GetVariantTypeMBS(v)

Var 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:

- 14.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 672
- 14.0.4 How to catch delete key? 673
- 14.0.5 How to convert cmyk to rgb? 674
- 14.0.6 How to delete a folder? 675
- 14.0.7 How to detect if CPU is 64bit processor? 676
- 14.0.9 How to refresh a htmlviewer on Windows? 678

14.0.9 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:

- 14.0.3 How do I get the proper highlight color on Mac OS X for active/inactive selection? 672
- 14.0.4 How to catch delete key? 673
- 14.0.5 How to convert cmyk to rgb? 674
- 14.0.6 How to delete a folder? 675

- 14.0.7 How to detect if CPU is 64bit processor?

- 14.0.8 How to query variant type string for a variant?

14.0.10 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:

```
Var v as Group2D
Var r as RectShape
Var 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
```

14.0.11 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:

```
Var l as Picture = LogoMBS(500)
```

```
l.HorizontalResolution = 300
```

```
l.VerticalResolution = 300
```

```
Var 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.

14.0.12 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:

```
Var 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
```

14.0.13 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.

14.0.14 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:

```

Var options(-1) as string

// load all the jar files we have in a folder called java:

Var appFolder as FolderItem = GetFolderItem("")

Var count as Integer = appFolder.Parent.Child("java").Count
Var libjs() as string
For i as Integer = 1 to count
Var 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
Var library as string = Join(libjs, "")
Var vm as new JavaVMMBS(library)

if vm.Handle = 0 then
MsgBox "Failed to initialize virtual machine"
else
// now make a new database connection with ucanaccess
Var d as new JavaDatabaseMBS(vm,"net.ucanaccess.jdbc.UcanaccessDriver")
Var DbFile as FolderItem = appFolder.Parent.Child("Database11.accdb")
Var j as JavaConnectionMBS = d.getConnection("jdbc:ucanaccess://" + DbFile.NativePath)

// select and show values
Var 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>

14.0.15 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.

14.0.16 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:

Var 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.

14.0.17 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](https://feedback.apple.com/showreport?report_id=11391)

14.0.18 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.

14.0.19 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
Var 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!
Var labels(-1) as string=array("Germany", "Italy", "France", "Spain", "UK", "Poland", "Russia")

// The colors to use for the sectors
Var colors(-1) as Integer

colors.Append &h66aaee
colors.Append &heebb22
colors.Append &hbbsbbb
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
Var c as new CDPieChartMBS(700, 600)

c.setBackground(c.linearGradientColor(0, 0, 0, c.getHeight(), &h0000cc, &h000044))
```

```

c.setRoundedFrame(&hffffff, 16)
Var tt as CDTextBoxMBS = c.addTitle("ChartDirector Demonstration", "timesbi.ttf", 18)
tt.setMargin(0, 0, 16, 0)
tt.setFontColor(&hFFFFFFF)

// 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)

Var t as CDTextBoxMBS = c.setLabelStyle("arialbd.ttf", 10, &h000000)
t.setBackground(CDPieChartMBS.kSameAsMainColor, CDPieChartMBS.kTransparent, CDPieChartMBS.soft-
Lighting(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

Var pic as picture = c.makeChartPicture
Var 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. Transferring 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.

14.0.20 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.

14.0.21 Can I use your plugin controls on a web application?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: No.

14.0.22 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.

14.0.23 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

<code>BottomLeft</code>	1	The leftmost point on the bottom line.
<code>BottomCenter</code>	2	The center point on the bottom line.
<code>BottomRight</code>	3	The rightmost point on the bottom line.
<code>Left</code>	4	The leftmost point on the middle horizontal line.
<code>Center</code>	5	The center point on the middle horizontal line.
<code>Right</code>	6	The rightmost point on the middle horizontal line.
<code>TopLeft</code>	7	The leftmost point on the top line.
<code>TopCenter</code>	8	The center point on the top line.
<code>TopRight</code>	9	The rightmost point on the top line.
<code>Bottom</code>	2	The center point on the bottom line. Same as <code>BottomCenter</code> .
<code>Top</code>	8	The center point on the top line. Same as <code>TopCenter</code> .
<code>TopLeft2</code>	10	An alternative top-left position used in <code>Axis.setTitlePos</code> for axis title positioning only. For a vertical axis, <code>TopLeft2</code> refers to the left of the top side, while <code>TopLeft</code> refers to the top of the left side. The reverse applies for a horizontal axis.
<code>TopRight2</code>	11	An alternative top-right position used in <code>Axis.setTitlePos</code> for axis title positioning only. For a vertical axis, <code>TopRight2</code> refers to the right of the top side, while <code>TopRight</code> refers to the top of the right side. The reverse applies for a horizontal axis.
<code>BottomLeft2</code>	12	An alternative bottom-left position used in <code>Axis.setTitlePos</code> for axis title positioning only. For a vertical axis, <code>BottomLeft2</code> refers to the left of the bottom side, while <code>BottomLeft</code> refers to the bottom of the left side. The reverse applies for a horizontal axis.
<code>BottomRight2</code>	13	An alternative bottom-right position used in <code>Axis.setTitlePos</code> for axis title positioning only. For a vertical axis, <code>BottomRight2</code> refers to the right of the bottom side, while <code>BottomRight</code> refers to the bottom of the right side. The reverse applies for a horizontal axis.

14.0.24 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 colors 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 AAR-RGGBB, 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 at all. This is equivalent to traditional RGB colors. A non-zero alpha transparency means 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 look 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

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.

14.0.25 ChartDirector: Font Specification

Plugin Version: 8.2, Platforms: macOS, Linux, Windows.

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 "FONTSPATH".

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 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.

14.0.26 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

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.

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".)

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.

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 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 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.

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.

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.

14.0.27 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 "&#nn;" (eg. ">" will become ">"). 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.

14.0.28 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
```

14.0.29 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.

14.0.30 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:

```

Var strNumber As String
Var nLength as Integer
Var nValue as Integer
Var nChecksum as Integer
Var 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)

14.0.31 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.
Then we can talk about the plugin.

14.0.32 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. Then 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.

14.0.33 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.

14.0.34 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
Var s as string
Var nf as FolderItem
nf = f
s = ""
while nf<>nil
s = nf.name + "." + s
nf = nf.parent
wend
Return s
End Function

```

14.0.35 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:

Var r as CGSTransitionRequestMBS
Var co as new CGSConnectionMBS
Var cw as CGSWindowMBS
Var 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.

14.0.36 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.

14.0.37 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
```

```

Var m as MemoryBlock = NewMemoryBlock(8)
Var family as Integer
Var s as string

m=SystemControlNameToMIBMBS("hw.cpubfamily")
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.

14.0.38 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
Var hSysMenu as Integer
hSysMenu = GetSystemMenu(me.WinHWND, 0)
RemoveMenu hSysMenu, &HF060, &H0
#endif

```

Notes: The window may not be updated directly.

14.0.39 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
Var m as memoryBlock
Var 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.

14.0.40 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:

```
Var 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.

14.0.41 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.

14.0.42 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.

14.0.43 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)
  Var result as Integer
  Var m, k as memoryBlock
  Var f as folderItem
  Var newType as string
  Var anIcon As picture
  Var 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
  end if
loop

End Sub

```

Notes: Change "Translation" to "CarbonLib" for Mac OS X.

14.0.44 How can I launch an app using its 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
Var A As AppleEvent
A = NewAppleEvent("aevt","odoc","MACS")
A.ObjectSpecifierParam("—") = GetUniqueIDObjectDescriptor("appf",nil,C)
return A.Send
End Function
```

14.0.45 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)
libpixmap-1.so.0 =>/usr/lib/i386-linux-gnu/libpixmap-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.

14.0.46 How can I validate an email address?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can try this code:

Example:

```

Var re As RegEx
re = New RegEx
Var 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>

14.0.47 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:

```
Var src as string // input

Var theRegex as Regex
Var theRegexMatch as RegexMatch
Var result, infoCharset, encodedPart as string
Var 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.

14.0.48 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.

14.0.49 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
Var 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

```

14.0.50 How do I get the current languages list?

Plugin Version: all, Platform: macOS.

Answer: Try this code:

Example:

```

Var p as new CFPREFERENCESMBS
Var a as CFArrayMBS
Var s as CFStringMBS
Var o as CFOBJECTMBS
Var sa(-1) as string

o=p.CopyAppValue("AppleLanguages",".GlobalPreferences")

if o<>Nil then
a=CFArrayMBS(o)

Var 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.

14.0.51 How do I get the Mac OS Version?

Plugin Version: all, Platform: macOS.

Answer: Try this code:

Example:

```
Var 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.

14.0.52 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:

```
Var s as String
Var 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 in 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.

14.0.53 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.

14.0.54 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)
  Var w,w1,x,p as Integer
  Var c1,c2,c as color
  Var 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.

14.0.55 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

Var pref as new CFPREFERENCESMBS

Var persistentapps as CFStringMBS = NewCFStringMBS("persistent-apps")
Var ApplicationID as CFStringMBS = NewCFStringMBS("com.apple.dock")
Var tiledata as CFStringMBS = NewCFStringMBS("tile-data")
Var filelabel as CFStringMBS = NewCFStringMBS("file-label")

// get the array of persistent applications from dock preferences
Var o as CFOBJECTMBS = pref.CopyValue(persistentapps, ApplicationID, pref.kCFPreferencesCurrentUser,
pref.kCFPreferencesAnyHost)

if o isa CFArrayMBS then
  Var a as CFArrayMBS = CFArrayMBS(o)

  // walk over all items in array
  Var 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
      Var 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
          Var name as string = CFStringMBS(o).str
          List.AddRow name
        end if
      end if
    end if
  next
end if
```

```
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.

14.0.56 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.

14.0.57 How do update a Finder's windows after changing some files?

Plugin Version: all, Platform: macOS.

Answer: Try this code:

Example:

```
Var f as folderitem // some file
Var 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.

14.0.58 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.

14.0.59 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.

14.0.60 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*

Var theEvent *as* AppleEvent

Var 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).

14.0.61 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:

```
Var 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.

14.0.62 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.

14.0.63 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>
```

14.0.64 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
Var 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

14.0.65 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:
Var 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.

14.0.66 How to bring my application to front?

Plugin Version: all, Platform: macOS.

Answer: This makes SimpleText (Code ttxt) to the frontmost application:

Example:

```
Var A As AppleEvent
A = NewAppleEvent("misc", "actv", "")
If Not A.Send then
Beep
end if
```

Notes: (Code is Mac only)

14.0.67 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)

Var 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 caught, a flag is set and you can ask later in your normal application flow for the result.

14.0.68 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:

```
Var mb as new MenubarMBS
Var 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.

14.0.69 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/> .

14.0.70 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
  Var c as Integer = folder.Count
```

```
  for i as Integer = 1 to c
    Var 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.

14.0.71 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
  Var p as new IOPowerSourcesMBS
```

```
  // check all power sources
  Var u as Integer = p.Count-1
  for i as Integer = 0 to u
    Var d as CFDictionaryMBS = p.Item(i)
    if d<>nil then
      // check if they have a power source state key:
      Var o as CFObjectMBS = d.Value(NewCFStringMBS("Power Source State"))
      if o isa CFStringMBS then
        Var 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.

14.0.72 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
Var 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

```

14.0.73 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:

Var 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.

14.0.74 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.

14.0.75 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)
```

```
  Var state, err as Integer
```

```
  Var 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.

14.0.76 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
```

```
  Var r, u as RGBSurface
```

```
  Var x, y, n, m, h, w as Integer
```

```
  Var w1, w2, h1, h2, d1, d2 as Integer
```

```
  Var 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.

14.0.77 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.

14.0.78 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

14.0.79 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

14.0.80 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

14.0.81 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 although it's not perfect.

You need to set lc to the current color you use.

Example:

```

Var outstring as string
Var theMac, thePC as textencoding
Var 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.

14.0.82 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
Var d2 as Double = CDBaseChartMBS.chartTime(2015, 1, 1)
Var da as new date(2015, 1, 1)
Var ts as Double = da.TotalSeconds
```

```
diff = ts - d2
end if
```

```
Var 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.

14.0.83 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:

```
Var inputfile as FolderItem = SpecialFolder.Desktop.Child("test.txt")
Var outputfile as FolderItem = SpecialFolder.Desktop.Child("output.txt")
Var it as TextInputStream = TextInputStream.Open(inputfile)
Var 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.

14.0.84 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

14.0.85 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()
  Var 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.

14.0.86 How to copy a 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
  Var 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 a dictionary of objects, you need to change code to also make a copy of those objects.

14.0.87 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:

```

Var f as FolderItem
Var md as EditableMovie
Var 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.

14.0.88 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
Var s as new CalCalendarStoreMBS

// needed for the error details
Var e as NSErrorMBS

Var r as CalRecurrenceRuleMBS = CalRecurrenceRuleMBS.initYearlyRecurrence(1, nil) // repeat every
year without end

Var a as new CalAlarmMBS // add alarm
a.action = a.CalAlarmActionDisplay
a.relativeTrigger = -3600*24 // 24 Hours before

// create a new calendar
Var c as new CalEventMBS

```

```

Var d as new date(2011, 04, 20) // the date

Var 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.

14.0.89 How to create a GUID?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use the UUIDMBS class for this.

14.0.90 How to create a Mac picture clip file?

Plugin Version: all, Platform: Windows.

Answer: You can use code like this one.

Example:

```

Var f As FolderItem
Var 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
```

Var 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
Var pictdata as string = p.PicHandleDataMBS
r.AddResource(pictdata,"PICT",256,"Picture")
```

Var 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

```
Var data as string = DecodeBase64("AQAAAAAAAAAAAAAAAAACAFRDRVIAAAAABAAAAAAAAAAAAAAAAABUQ0IQAAAAA")
r.AddResource(data,"drag",128,"") 'ditto
r.Close
```

Notes: In general Apple has deprecated this, but a few application still support clippings.

14.0.91 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.

14.0.92 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
Var 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.

14.0.93 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
Var pdf As New MyDynapdfMBS

// page width/height in MilliMeter
Var pdfWidth as Integer = PageWidth * 72 / 25.4
Var 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.

14.0.94 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 interpreted 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. Pass this additional header with the `SetOptionHTTPHeader` method.

```
curl -X PUT http://127.0.0.1:5984/appserials/f2f4e540bf8bb60f61cfd4328001c59 -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)

14.0.95 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:

```
Var d as CURLMBS // your curl object
```

```
// delete file
```

```
Var 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.

14.0.96 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".

14.0.97 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)
```

14.0.98 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.

14.0.99 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.

14.0.100 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.

14.0.101 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
Var n(0) as Integer ' all the numbers
Var m as Integer ' the highest field in the current array
Var 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

```



```

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.

14.0.104 How to draw a nice antialiased line?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

This code can help you although 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)
Var intX, intY, count, n, xDiff, yDiff as Integer
Var v, v1, floatX, floatY, xx, yy, xStep, yStep as Double
Var 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.

14.0.105 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

14.0.106 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:
    Var q as new Picture(p.Width, p.Height)
    q.Graphics.DrawPicture p,0,0

  Return q

  end if
  #endif

  // create new picture
  Var q as new Picture(p.Width, p.Height, 32)

  // get mask
  Var 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.

14.0.107 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.

14.0.108 How to encrypt a file with Blowfish?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use code like this:

Example:

```
Var fi as FolderItem = SpecialFolder.Desktop.Child("test.xojo_binary_project")
Var fo as FolderItem = SpecialFolder.Desktop.Child("test.encrypted")
```

```
// read input
Var bi as BinaryStream = BinaryStream.Open(fi)
Var si as string = bi.Read(bi.Length)
bi.Close

// encrypt
Var so as string = BlowfishMBS.Encrypt("MyKey",si)

// write output
Var 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.

14.0.109 How to extract text from HTML?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Use both RemoveHTMLTagsMBS and DecodingFromHTMLMBS like this:

Example:

```

Var html as string = "<p><B>Gr&uuml;l&szlig;e</B></P>"
Var htmltext as string = RemoveHTMLTagsMBS(html)
Var text as string = DecodingFromHTMLMBS(htmltext)

```

MsgBox text // shows: Gr√üë

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 ä to √§.

14.0.110 How to find empty folders in a folder?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Try this code:

Example:

```

Var folder as folderitem // your folder

Var c as Integer = folder.count
for i as Integer = 1 to c
  Var 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
end if
next

```

14.0.111 How to find iTunes on a Mac OS X machine fast?

Plugin Version: all, Platform: macOS.

Answer: You can try Launch Services.

Example:

```

Var f as FolderItem

f=LaunchServicesFindApplicationForInfoMBS("hook","com.apple.iTunes","iTunes.app")

MsgBox f.NativePath

```

14.0.112 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
Var u as Integer = System.NetworkInterfaceCount-1
for i as Integer = 0 to u
Var 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
Var interfaces() as NetworkInterfaceMBS = NetworkInterfaceMBS.AllInterfaces
Var map as new Dictionary

for each n as NetworkInterfaceMBS in interfaces
Var IPv4s() as string = n.IPv4s
Var 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
Var 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.

14.0.113 How to find version of Microsoft Word?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use code like this:

Example:

```
// find Word
Var f as FolderItem = LaunchServicesFindApplicationForInfoMBS("", "com.microsoft.Word", "")

// open bundle
Var c as new NSBundleMBS(f)

// read info
Var d as Dictionary = c.infoDictionary

// show version
MsgBox d.Lookup("CFBundleVersion", "")
```

Notes: Older versions of Word can be found with creator code "MSWD".

14.0.114 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:

```
Var 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:
Var 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/>

14.0.115 How to format double with n digits?

Plugin Version: all, Platform: macOS.

Answer: You can use the FormatMBS function for this.

Example:

```
Var 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.

14.0.116 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
Var d as new date
Var s as string = d.LongTime

// adjust to client GMT offset
d.GMTOffset = d.GMTOffset + Session.GMTOffset

Var t as string = D.LongTime

MsgBox s+EndOfLine+t
End Sub
```

14.0.117 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
```

14.0.118 How to get `CFAbsoluteTime` from date?

Plugin Version: all, Platforms: macOS, Windows.

Answer: Use code like this:

Example:

```
Var d as new date
Var t as CFTimeZoneMBS = SystemCFTimeZoneMBS
Var g as new CFGregorianCalendarMBS
g.Day = d.Day
g.Month = d.Month
g.Year = d.Year
g.Minute = d.Minute
g.Hour = d.Hour
g.Second = d.Second
```

```
Var at as CFAbsoluteTimeMBS = g.AbsoluteTime(t)
Var 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.

14.0.119 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
```

14.0.120 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:

Var 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.

14.0.121 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:

```
Var 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.

14.0.122 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
Var d as new date

// now in GMT
Var e as new date
e.GMTOffset = 0

// show
MsgBox str(d.TotalSeconds, "0.0")+ " " +str(e.TotalSeconds, "0.0")

Var GMTTimeStamp as Double = e.TotalSeconds

// restore
```

```
Var 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.

14.0.123 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>

14.0.124 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
Var t() as Thread

Var 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.

14.0.125 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.

14.0.126 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

Var i as Integer
Var 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" ...

14.0.127 How to get the current free stack space?

Plugin Version: all, Platform: macOS.

Answer: You can something like the code below:

Example:

```
Sub ShowStackSize()
  Var threadid as Integer
  Var size as Integer

  declare function GetCurrentThread lib "Carbon" (byref threadid as Integer) as short
  declare function ThreadCurrentStackSize lib "Carbon" (threadid as Integer, byref size as Integer) as short

  if GetCurrentThread(threadid)=0 then
  if 0=ThreadCurrentStackSize(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.

14.0.128 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

  Var result, bias, dayLightbias as Integer
```

```

Var info as memoryBlock
Var 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

```

14.0.129 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 Accessibility services are

Example:

```

Function CurrentWindowTitle() As string
// your application needs permissions for accessibility to make this work!

Var SystemWideElement,FocusedApplicationElement,FocusedWindowElement as AXUIElementMBS
Var FocusedApplication,FocusedWindow,Title as AXValueMBS
Var s as String
Var 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

```

14.0.130 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.

14.0.131 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:

```
Var ae as appleEvent
Var o1 as appleEventObjectSpecifier
Var f as folderItem
Var aList as appleEventdescList
Var i as Integer
Var 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.folderItem(i)

dateiname=f.name
// editfield1 with property "multiline=true"!
editfield1.text=editfield1.text + dateiname + chr(13)
next
end if

```

14.0.132 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:

```

Var first as Integer
Var second as Integer
Var third as Integer
Var 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

```

14.0.133 How to get the Mac OS Version using System.Gestalt?

Plugin Version: all, Platform: macOS.

Answer: Try this code:

Example:

```

Var s As String
Var b As Boolean
Var 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.

14.0.134 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.

14.0.135 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)
Var err as Integer
Var 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!

14.0.136 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
```

14.0.137 How to get Xojo apps running Linux?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You need to install some require 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.

14.0.138 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.

14.0.139 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.

14.0.140 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.

14.0.141 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:

```

Var 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.

14.0.142 How to hide all applications except mine?

Platform: macOS.

Answer: The code below will on Mac OS hide all applications except your one:

Example:

```

Var p as new ProcessMBS

```

```
p.GetFirstProcess
do
if not p.FrontProcess then
p.Visible=false
end if
loop until not p.GetNextProcess
```

14.0.143 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).

14.0.144 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.

14.0.145 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.

14.0.146 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
Var b as BinaryStream = BinaryStream.Open(f)
Var s as string = b.Read(b.Length)

// build wrapper
Var fileWrapper as NSFileWrapperMBS = NSFileWrapperMBS.initRegularFileWithContents(s)
fileWrapper.preferredFilename = f.name

// make attachment
Var fileAttachment as new NSTextAttachmentMBS(fileWrapper)
Var attributedString as NSAttributedStringMBS = NSAttributedStringMBS.attributedStringWithAttachment(fileAttachment)

// add to a NSTextViewMBS
textview.insertText attributedString

End Sub
```

Notes: For TextArea you can query the underlying NSTextViewMBS object via TextArea.NSTextViewMBS method.

14.0.147 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
```

14.0.148 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
```

14.0.149 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.

14.0.150 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:

```
Var 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.

14.0.151 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 then 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.

14.0.152 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
Var r as new NilObjectException
raise r
```

```
catch x as NilObjectException
```

```
// get stack
Var stack() as string = x.Stack
```

```
// pick function name and return
Var name as string = stack(2)
Return name
```

```
end try
End Function
```

Notes: You need to include function names in your application.

14.0.153 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:

```
Var a as AppleEvent
Var 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
```

14.0.154 How to launch disc utility?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use this code:

Example:

```
Var 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.

14.0.155 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:

```
Var db as Database // some database

db.SQLExecute "BEGIN TRANSACTION"
// Do some Stuff
db.SQLExecute "END TRANSACTION"
```

Notes: This can increase speed by some factors.

14.0.156 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
Var 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.

14.0.157 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
```

```
Var 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
```

14.0.158 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)
```

```
Var ev as AppleEvent
```

```
Var myResult as boolean
```

```
Var 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

14.0.159 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

```

14.0.160 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:

```
Var doubleClickTime, currentClickTicks as Integer

#if targetMacOS then
Declare Function GetDbfTime Lib "Carbon" () as Integer
doubleClickTime = GetDbfTime()
#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 occurred close enough together in space
if abs(X - lastClickX) <= 5 and abs(Y - LastClickY) <= 5 then
DoubleClick //a double click has occurred 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

14.0.161 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 HDAActivity = 3 // Delays hard drive spindown and idle sleep by small amount */
const IdleActivity = 4 // Delays idle sleep by timeout time */

```

```
Var 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.

14.0.162 How to make my own registration code scheme?

Plugin Version: all, Platform: Windows.

Answer: There are excellent articles about how to make a registration 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?

14.0.163 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 OS-
Type, size as Integer, data as ptr) as short

Var m as MemoryBlock

m=NewMemoryBlock(2)
m.UShort(0)=kControlSizeSmall

Title=str(SetControlData(CheckBox1.Handle, 0, kControlSizeTag, 2, m))

```

14.0.164 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:

```

Var 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.

14.0.165 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
Var r as FolderItem
Var e as Integer = MacFileOperationMBS.MoveObjectToTrashSync(f, r, MacFileOperationMBS.kFSFile-
OperationDefaultOptions)

```

```

if e = 0 then
Return true // Ok
end if

#elseif TargetWin32 then
Var w as new WindowsFileCopyMBS

Var 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.

14.0.166 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:

```

Var a as appleevent

a=newappleEvent("misc","actv","ttxt")

if a.send then

```

```
end if
```

Notes: (Code is Mac only)

14.0.167 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:

```
Var d as CURLMBS // your curl object
```

```
// rename/move file
```

```
Var ws() As String
```

```
ws.Append "RNFR Temp.txt"
```

```
ws.append "RNT0 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 RNT0 with the new file name. To delete use DELE and the file path.

14.0.168 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
```

```
Var s as CFStringMBS = NewCFStringMBS(t)
```

```
Var m as CFMutableStringMBS = s.Normalize(kCFStringNormalizationFormD)
```

```
Return m.str
```

```
End Function
```

Notes: This uses Apple's CFString functions to normalize unicode variants.

14.0.169 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.

14.0.170 How to open icon file on Mac?

Plugin Version: all, Platform: macOS.

Answer: Use the NSImageMBS class like this:

Example:

```
Var f as FolderItem = SpecialFolder.Desktop.Child("test.ico")
```

```
Var n as new NSImageMBS(f)
```

```
window1.Backdrop = n.CopyPictureWithMask
```

14.0.171 How to open PDF in acrobat reader?

Plugin Version: all, Platform: macOS.

Answer: Try this code:

Example:

```
Var pdf as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
```

```
// open PDF in Acrobat Reader on Mac:
```

```
// find app
```

```
Var bundleID as string = "com.adobe.Reader"
```

```
Var app as FolderItem = LaunchServicesFindApplicationForInfoMBS("", bundleID, "")
```

```

if app<>nil then

  // launch app with parameters

  Var docs() as FolderItem
  docs.Append pdf

  Var param as new LaunchServicesLaunchParameterMBS
  param.Defaults = true
  param.Application = app

  Var 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.

14.0.172 How to open printer preferences on Mac?

Plugin Version: all, Platform: macOS.

Answer: You can use our OpenMacOSXPreferencesPaneMBS function like this:

Example:

```

Var 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

```

14.0.173 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:

```
Var 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.

14.0.174 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.

14.0.175 How to parse XML?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: You can use code like this:

Example:

```
Var s as string = "<test><test /></test>"

try
Var 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.

14.0.176 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 javascript.

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:

```
Var urlo as string = app.audioFileOGG.URL
Var 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 ele-
ment.</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)+"");")
```

14.0.177 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>

14.0.178 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
Var p as new PrinterSetup

// now put it into NSPrintInfo to manipulate
Var n as new NSPrintInfoMBS
n.SetupString = p.SetupString

// change destination to file
Var f as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
n.SetSaveDestination(f)

// move back
p.SetupString = n.SetupString

// and print as usual
Var g as Graphics = OpenPrinter(p)
g.DrawString "Hello World", 20, 20
```

Notes: And you can use normal graphics class for that.

14.0.179 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
Var xMDItem as New MDItemMBS(F)
Var 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.

14.0.180 How to quit windows?

Plugin Version: all, Platform: Windows.

Answer: Try this code:

Example:

```
#if targetwin32 then
Var 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.

14.0.181 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:

```
Var f as FolderItem = SpecialFolder.Desktop.Child("test.csv")
Var t as TextInputStream = f.OpenAsTextFile

while not t.EOF
Var s as string = t.ReadLine(encodings.ASCII)

Var items() as string = SplitCommaSeparatedValuesMBS(s, ";", """")

List.AddRow ""
Var 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.

14.0.182 How to read the command line on windows?

Plugin Version: all, Platform: Windows.

Answer: Try this code:

Example:

```
#if targetwin32 then
Var line as string
Var 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.

14.0.183 How to render PDF pages with PDF Kit?

Plugin Version: all, Platform: Windows.

Answer: Try this code:

Example:

```
// choose a file
Var f as FolderItem = SpecialFolder.Desktop.Child("test.pdf")

// open it as PDF Document
Var sourceFile as New PDFDocumentMBS(f)

if sourceFile.handle <>0 then // it is a PDF file

// get upper bound of pages
Var c as Integer = sourceFile.pageCount-1

// from first to last page
for n as Integer = 0 to c

// pick that page
Var page as PDFPageMBS = sourceFile.pageAtIndex(n)

// render to image
Var 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.

14.0.184 How to restart a Mac?

Plugin Version: all, Platform: macOS.

Answer: Ask the Finder via Apple Events:

Example:

```
Var ae as appleevent
ae=newappleEvent("FNDR","rest","MACS")
if not ae.send then
msgBox "The computer couldn't be restarted."
end if
```

14.0.185 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.

14.0.186 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
Var sourcefile as FolderItem = SpecialFolder.Desktop.Child("test.pdf")
Var destfile as FolderItem = SpecialFolder.Desktop.Child("rotated.pdf")

// open PDF
Var pdf as CGPDFDocumentMBS = sourcefile.OpenAsCGPDFDocumentMBS
```

```

// query media size of first page
Var r as CGRectMBS = pdf.MediaBox(1)

// create new PDF
Var c as CGContextMBS = destfile.NewCGPDFDocumentMBS(r,"title","Author","Creator")

// create rotated rectangle
Var 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.

14.0.187 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

```

```

Var f as FolderItem = SpecialFolder.Desktop.Child("test.png")
Var image as new CImageMBS(f)

// rotate 45 degree
Var n as new NSAffineTransformMBS
n.rotateByDegrees(45)

Var TransformFilter as new CFilterAffineTransformMBS
TransformFilter.inputImage = image
TransformFilter.inputTransform = n

// get result
Var resultImage as CImageMBS = TransformFilter.outputImage

// for saving to file
Var outputImage as NSImageMBS = resultImage.RenderNSImage(false)

f = SpecialFolder.Desktop.Child("output.png")
Var b as BinaryStream = BinaryStream.Create(f, true)
b.Write outputImage.PNGRepresentation

// as Xojo picture object for display
Var pic as Picture = outputImage.CopyPictureWithMask

Backdrop = pic

```

14.0.188 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.

14.0.189 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
Var n as NSPrintInfoMBS = NSPrintInfoMBS.sharedPrintInfo
n.orientation = n.NSLandscapeOrientation

```

```
// save html to file
Var 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.

14.0.190 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
Var f as FolderItem = GetSaveFolderItem(FileTypes1.ApplicationRtf, "test.rtf")

if f = nil then Return

Var a as NSAttributedStringMBS = textView.textStorage
Var w as NSFileWrapperMBS = a.RTFDFileWrapperFromRange(0, a.length, DocumentAttributes)

Var e as NSErrorMBS
if w.writeToFile(f, e) then

else
MsgBox e.LocalizedDescription
end if
```

Notes: For TextArea you can query the underlying NSTextViewMBS object via TextArea.NSTextViewMBS method.

14.0.191 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)
Var 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
Var 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.

14.0.192 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

Var faktor as Double = min( Height / Pic.Height, Width / Pic.Width)

// Calculate new size
Var w as Integer = Pic.Width * faktor
Var h as Integer = Pic.Height * faktor

// create new picture
Var NewPic as new Picture(w,h,32)

// check if we have a mask and clear it
Var 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.

14.0.193 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

Var faktor as Double = min( Height / Pic.Height, Width / Pic.Width)

// Calculate new size
Var w as Integer = Pic.Width * faktor
Var h as Integer = Pic.Height * faktor

// create new picture
Var 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)

14.0.194 How to scale/resize a CImageMBS?

Plugin Version: all, Platform: Windows.

Answer: Use the CIFilterLanczosScaleTransform filter to scale down a picture to a specific size.

Example:

```

Var pic As Picture = LogoMBS(500)
Var image As CImageMBS = CImageMBS.imageWithPicture(pic)

Var filter As New CIFilterLanczosScaleTransformMBS

Const targetWidth = 600.0
Const targetHeight = 400.0

Var scale As Double = targetHeight / image.Extent.Height
Var aspect As Double = targetWidth / (image.Extent.Width * scale)

filter.inputImage = image
filter.inputScale = scale
filter.inputAspectRatio = aspect

Var result As Picture = filter.outputImage.RenderPicture

Backdrop = result

```

Notes: This is same code as our scaleTo convenience method.

14.0.195 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:

```

Var 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.

- CImageMBS 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.

14.0.196 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:

```
Var r as RegExMbs
Var s as string
Var c as Integer
```

```
s="123 √§√√° 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
```

14.0.197 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
Var TIS As TextInputStream
Var S,All As String
Var I as Integer
Var 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
```

14.0.198 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
Var 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.

14.0.199 How to set the modified dot in the window?

Plugin Version: all, Platform: macOS.

Answer: Try this declares:

Example:

```
window1.ModifiedMBS=true
```

14.0.200 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:

```
Var 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.

14.0.201 How to show Keyboard Viewer programmatically?

Platform: macOS.

Answer: Use Xojo or AppleScript to launch the KeyboardViewerServer.app.

Example:

```
Var a as new AppleScriptMBS
Var text as string
Var lines(-1) as string

lines.append "set theApplication to ""KeyboardViewerServer""
lines.append "set thePath to ""/System/Library/Components/KeyboardViewer.component/Contents/Shared-
Support/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/Key-
boardViewerServer.app"

```

```

set POSIXPath to ((POSIX file thePath) as string)
tell application "System Events" to set isRunning to 0 <(count (application processes whose name is theAp-
plication))
if isRunning then tell application POSIXPath to quit
delay 0.15

```

```

ignoring application responses
tell application POSIXPath to run
end ignoring

```

14.0.202 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.

14.0.203 How to shutdown a Mac?

Plugin Version: all, Platform: macOS.

Answer: Ask the Finder via Apple Events:

Example:

```
Var 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 immediatly: No document asked to be saved, all applications quitting without knowing).

```
Declare Sub ShutDwnPower Lib "Carbon" ()
ShutDwnPower
```

14.0.204 How to sleep a Mac?

Plugin Version: all, Platform: macOS.

Answer: Ask the Finder via Apple Events:

Example:

```
Var ae as appleevent
ae=newappleEvent("FNDR","slep","MACS")
if not ae.send then
msgBox "The computer doesn't want to sleep."
end if
```

14.0.205 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.

14.0.206 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.

14.0.207 How to use quotes in a string?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Just double them.

Example:

```
msgbox "This String contains ""quotes"""
```

14.0.208 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
```

14.0.209 How to use the Application Support folder?

Plugin Version: all, Platform: macOS.

Answer:

I was saving a registration code for an app to the Preference folder. 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:

```
Var 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!

14.0.210 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()
Var c as CFDateMBS
Var 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
Var s as SoftDeclareMBS
Var 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
Var s as SoftDeclareMBS
Var m as MemoryBlock

```

```

'/*
' * Types of power event
' * These are potential arguments to IOPMSchedulePowerEvent().
' * These are all potential values of the kIOPMPowerEventTypeKey in the CFDictionaryes
' * 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.

14.0.211 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
Var 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".

14.0.212 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)
Var folders() as FolderItem

```

```

folders.Append folder

```

```

while UBound(folders)>=0

```

```

Var currentFolder as FolderItem = folders.pop

Var c as Integer = currentFolder.Count
for i as Integer = 1 to c
Var 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.

14.0.213 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.

14.0.214 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 temporary 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 `~/Library/Caches` and your temp folders.

Finally make sure you use the right serial number. Not an older one or a misspelled one.

14.0.215 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)
Var s as string
```

```

Var f as folderItem
Var d as CFDictionaryMBS
Var o as CFObjectMBS
Var key as CFStringMBS
Var dl as CFDictionaryListMBS
Var i,c as Integer
Var u as CFURLMBS
Var 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=NewCFURLMBS CFStringMBS(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.

14.0.216 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
Var 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 clipping and an offset which the plugin doesn't know about.

14.0.217 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.

14.0.218 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)

14.0.219 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](#):

Var 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.

14.0.220 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

14.0.221 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:

```
Var 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.

14.0.222 Is there an easy way I can launch the Displays preferences panel?

Plugin Version: all, Platform: macOS.

Answer: Use the code below:

Example:

```
Var error as Integer
```

```

error=OpenMacOSXPreferencesPaneMBS("Displays")
if error<>0 then
MsgBox "Failed to launch QuickTime System Preferences panel."
end if

```

14.0.223 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>

14.0.224 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)

14.0.225 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?

14.0.226 SQLiteDatabase not initialized error?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Before you can use SQLiteDatabaseMBS, it must be initialized.

Example:

```
Var d as new SQLiteDatabaseMBS
```

Notes: This happens normally when you use "new SQLiteDatabaseMBS".

But if you just have a SQLConnectionMBS and get a recordset there, the initialization may not have happened, yet.

So please simply add a line "dim d as new SQLiteDatabaseMBS" to your app.open code after registration, so the plugin part can initialize and late provide recordsets.

14.0.227 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.

14.0.228 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 CTypeRef 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.

14.0.229 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.

14.0.230 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.

14.0.231 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.

14.0.232 What does the NAN code mean?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

14.0.233 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)
  Var err as Integer
  Var 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
```

14.0.234 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.

14.0.235 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.

14.0.236 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 use older versions if you use an old Xojo.

Xojo 2017r3 and newer load our 64-bit plugins.

14.0.237 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.

14.0.238 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, REFTTEXT, 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.

14.0.239 What is the replacement for PluginMBS?

Plugin Version: all, Platform: macOS.

Answer: Use the SoftDeclareMBS class to load libraries dynamically.

14.0.240 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.

14.0.241 What to do with a NSImageCacheException?

Plugin Version: all, Platforms: macOS, Windows.

Answer: You need to add exception handlers for NSExcptionMBS 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.

14.0.242 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.

14.0.243 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.

14.0.244 Where is CGGetActiveDisplayListMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetActiveDisplayList.

14.0.245 Where is CGGetDisplaysWithPointMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetDisplaysWithPoint.

14.0.246 Where is CGGetDisplaysWithRectMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetDisplaysWithRect.

14.0.247 Where is CGGetOnlineDisplayListMBS?

Plugin Version: all, Platform: Windows.

Answer: This is now CGDisplayMBS.GetOnlineDisplayList.

14.0.248 Where is GetObjectClassNameMBS?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer: Please use this replacement method:

Example:

```
Function GetObjectClassNameMBS(o as Object) As string
  Var t as Introspection.TypeInfo = Introspection.GetType(o)
  Return t.FullName
End Function
```

Notes: GetObjectClassNameMBS was removed from the plugins.

14.0.249 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.macs.w.de"

// resolve IPs
Var ip1 as string = DNSNameToAddressMBS(Domain1)
Var 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.

14.0.250 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.

14.0.251 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.

14.0.252 Where to get information about file formats?

Plugin Version: all, Platforms: macOS, Linux, Windows.

Answer:

Please visit this web page:
<http://www.wotsit.org>

14.0.253 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>

14.0.254 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.

14.0.255 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.

14.0.256 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.

14.0.257 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.

14.0.258 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.

14.0.259 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.

14.0.260 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.

14.0.261 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.

14.0.262 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.

14.0.263 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.

14.0.264 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

14.0.265 Xojo doesn't work with your plugins on Windows 98.

Plugin Version: all, Platform: Windows.

Answer: Please upgrade your Windows version.

**14.0.266 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 preferred 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 <code>Axis.setLabels</code> 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 <code>Layer.addExtraField</code> , <code>Layer.addExtraField2</code> , <code>BaseChart.addExtraField</code> or <code>BaseChart.addExtraField2</code> .

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 <code>LineStyle.setSymbolScale</code> .
zy	The symbol scale in the y dimension. Applicable for layers with symbol scales set by <code>LineStyle.setSymbolScale</code> .
z	The symbol scale without distinguishing the dimension to use. Applicable for layers with symbol scales set by <code>LineStyle.setSymbolScale</code> .

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 Polar-Layer.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 add 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,,