

MBS Real Studio WinICM Plugin Documentation

Christian Schmitz

May 15, 2012

0.1 Introduction

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

0.2 Content

- 1 List of all topics 3
- 2 All items in this plugin 13
- 3 List of all classes 109
- 4 List of all modules 111

Chapter 1

List of Topics

• 2 Windows ICM	13
– 2.2 class WindowsICMProfileMBS	21
* 2.2.1 ConvertColorNameToIndex(name as string) as integer	21
* 2.2.1 ConvertIndexToColorName(index as integer) as string	21
* 2.2.1 CountColorProfileElements as integer	22
* 2.2.1 GetColorProfileElement(tag as integer) as string	22
* 2.2.1 GetColorProfileElementTag(index as integer) as integer	23
* 2.2.1 GetNamedProfileInfo as WindowsICMNamedProfileInfoMBS	24
* 2.2.1 GetProfileData as string	24
* 2.2.1 IsColorProfileTagPresent(tag as integer) as boolean	24
* 2.2.1 IsValid as boolean	25
* 2.2.1 OpenProfileData(data as string, DesiredAccess as integer) as WindowsICMProfileMBS	25
* 2.2.1 OpenProfileFile(file as folderitem, DesiredAccess as integer, ShareMode as integer, CreationMode as integer) as WindowsICMProfileMBS	26
* 2.2.1 OpenProfilePath(path as string, DesiredAccess as integer, ShareMode as integer, CreationMode as integer) as WindowsICMProfileMBS	28
* 2.2.1 SetColorProfileHeader(header as WindowsICMProfileHeaderMBS) as boolean	30
* 2.2.2 ColorProfileHeader as WindowsICMProfileHeaderMBS	30
* 2.2.2 Handle as Integer	31
* 2.2.3 CREATE_ ALWAYS = 2	31
* 2.2.3 CREATE_ NEW = 1	31
* 2.2.3 FILE_ SHARE_ READ = 1	31
* 2.2.3 FILE_ SHARE_ WRITE = 2	32
* 2.2.3 OPEN_ ALWAYS = 4	32
* 2.2.3 OPEN_ EXISTING = 3	32

* 2.2.3 PROFILE_READ = 1	32
* 2.2.3 PROFILE_READWRITE = 2	32
* 2.2.3 TRUNCATE_EXISTING = 5	33
– 2.1 class WindowsICMSetupMBS	13
* 2.1.1 Setup as boolean	13
* 2.1.2 DisplayName as String	14
* 2.1.2 Flags as Integer	14
* 2.1.2 MonitorProfile as String	14
* 2.1.2 Parent as Window	15
* 2.1.2 PrinterName as String	15
* 2.1.2 PrinterProfile as String	15
* 2.1.2 ProofingIntent as Integer	15
* 2.1.2 RenderIntent as Integer	16
* 2.1.2 SourceName as String	16
* 2.1.2 TargetProfile as String	17
* 2.1.3 Apply	17
* 2.1.3 Idle	17
* 2.1.4 CMS_DISABLEICM = 1	17
* 2.1.4 CMS_DISABLEINTENT = 1024	18
* 2.1.4 CMS_DISABLERENDERINTENT = 2048	18
* 2.1.4 CMS_ENABLEPROOFING = 2	18
* 2.1.4 CMS_SETMONITORPROFILE = 16	18
* 2.1.4 CMS_SETPRINTERPROFILE = 32	18
* 2.1.4 CMS_SETPROOFINTENT = 5	19
* 2.1.4 CMS_SETRENDERINTENT = 4	19
* 2.1.4 CMS_SETTARGETPROFILE = 64	19
* 2.1.4 CMS_USEAPPLYCALLBACK = 256	19
* 2.1.4 CMS_USEDESCRIPTION = 512	20
* 2.1.4 CMS_USEHOOK = 128	20
* 2.1.4 INTENT_ABSOLUTE_COLORIMETRIC = 3	20
* 2.1.4 INTENT_PERCEPTUAL = 0	20
* 2.1.4 INTENT_RELATIVE_COLORIMETRIC = 1	20
* 2.1.4 INTENT_SATURATION = 2	21
– 2.7 class WindowsICMProfileHeaderMBS	77
* 2.7.1 Attributes0 as Integer	78
* 2.7.1 Attributes1 as Integer	78
* 2.7.1 Classes as Integer	78
* 2.7.1 CMMType as Integer	79
* 2.7.1 ConnectionSpace as Integer	79
* 2.7.1 Creator as Integer	80
* 2.7.1 DataColorSpace as Integer	80

* 2.7.1 DateTime0 as Integer	82
* 2.7.1 DateTime1 as Integer	82
* 2.7.1 DateTime2 as Integer	82
* 2.7.1 IlluminantX as Integer	82
* 2.7.1 IlluminantY as Integer	82
* 2.7.1 IlluminantZ as Integer	83
* 2.7.1 Manufacturer as Integer	83
* 2.7.1 Model as Integer	83
* 2.7.1 Platform as Integer	83
* 2.7.1 ProfileFlags as Integer	84
* 2.7.1 RenderingIntent as Integer	84
* 2.7.1 Signature as Integer	85
* 2.7.1 Version as Integer	85
* 2.7.2 ATTRIB_ MATTE = 2	86
* 2.7.2 ATTRIB_ TRANSPARENCY = 1	86
* 2.7.2 CLASS_ ABSTRACT = & h61627374	86
* 2.7.2 CLASS_ CAMP = & h6C616D70	86
* 2.7.2 CLASS_ COLORSPACE = & h73706163	86
* 2.7.2 CLASS_ GMMP = & h676D6D70	87
* 2.7.2 CLASS_ LINK = & h6C696E6B	87
* 2.7.2 CLASS_ MONITOR = & h6D6E7472	87
* 2.7.2 CLASS_ NAMED = & h6E6D636C	87
* 2.7.2 CLASS_ PRINTER = & h70727472	87
* 2.7.2 CLASS_ SCANNER = & h73636E72	88
* 2.7.2 FLAG_ DEPENDENTONDATA = 2	88
* 2.7.2 FLAG_ EMBEDDEDPROFILE = 1	88
* 2.7.2 FLAG_ ENABLE_ CHROMATIC_ ADAPTATION = & h02000000	88
* 2.7.2 SPACE_ 2_ CHANNEL = & h32434C52	88
* 2.7.2 SPACE_ 3_ CHANNEL = & h33434C52	89
* 2.7.2 SPACE_ 4_ CHANNEL = & h34434C52	89
* 2.7.2 SPACE_ 5_ CHANNEL = & h35434C52	89
* 2.7.2 SPACE_ 6_ CHANNEL = & h36434C52	89
* 2.7.2 SPACE_ 7_ CHANNEL = & h37434C52	89
* 2.7.2 SPACE_ 8_ CHANNEL = & h38434C52	90
* 2.7.2 SPACE_ CMY = & h434D5920	90
* 2.7.2 SPACE_ CMYK = & h434D594B	90
* 2.7.2 SPACE_ GRAY = & h47524159	90
* 2.7.2 SPACE_ HLS = & h484C5320	90
* 2.7.2 SPACE_ HSV = & h48535620	91
* 2.7.2 SPACE_ Lab = & h4C616220	91
* 2.7.2 SPACE_ Luv = & h4C757620	91
* 2.7.2 SPACE_ RGB = & h52474220	91

* 2.7.2 SPACE_ XYZ = & h58595A20	91
* 2.7.2 SPACE_ YCbCr = & h59436272	92
* 2.7.2 SPACE_ Yxy = & h59787920	92
– 2.3 class WindowsICMTransformMBS	33
* 2.3.1 CheckColors(InputColors() as WindowsICMColorMBS, ctInput as integer, Results() as integer) as boolean	33
* 2.3.1 Constructor(LogColorSpace as WindowsICMLogColorSpaceMBS, DestProfile as WindowsICMProfileMBS, TargetProfile as WindowsICMProfileMBS, Flags as integer)	34
* 2.3.1 Constructor(Profiles() as WindowsICMProfileMBS, Intents() as integer, Flags as integer, indexPreferredCMM as integer)	36
* 2.3.1 GetCMMInfo(what as integer) as integer	39
* 2.3.1 TranslateBitmapBits(SrcBits as memoryblock, InputType as integer, Width as integer, Height as integer, InputRowBytes as integer, DestBits as integer, DestType as integer, DestRowBytes as integer) as boolean	40
* 2.3.1 TranslateColors(InputColors() as WindowsICMColorMBS, ctInput as integer, OutputColors() as WindowsICMColorMBS, ctOutput as integer) as boolean	41
* 2.3.1 TranslatePictures(InputPicture as picture, OutputPicture as picture) as boolean	41
* 2.3.2 Handle as Integer	42
* 2.3.3 Progress(Maximum as integer, Current as integer) as boolean	42
* 2.3.4 BEST_ MODE = 3	42
* 2.3.4 BM_ 10b_ G3CH = & h0404	42
* 2.3.4 BM_ 10b_ Lab = & h0403	43
* 2.3.4 BM_ 10b_ RGB = 9	43
* 2.3.4 BM_ 10b_ XYZ = & h0401	43
* 2.3.4 BM_ 10b_ Yxy = & h0402	43
* 2.3.4 BM_ 16b_ G3CH = & h0504	43
* 2.3.4 BM_ 16b_ GRAY = & h0505	44
* 2.3.4 BM_ 16b_ Lab = & h0503	44
* 2.3.4 BM_ 16b_ RGB = 10	44
* 2.3.4 BM_ 16b_ XYZ = & h0501	44
* 2.3.4 BM_ 16b_ Yxy = & h0502	44
* 2.3.4 BM_ 32b_ scARGB = & h0602	45
* 2.3.4 BM_ 32b_ scRGB = & h0601	45
* 2.3.4 BM_ 565RGB = 1	45
* 2.3.4 BM_ 5CHANNEL = & h0205	45
* 2.3.4 BM_ 6CHANNEL = & h0206	45
* 2.3.4 BM_ 7CHANNEL = & h0207	46
* 2.3.4 BM_ 8CHANNEL = & h0208	46
* 2.3.4 BM_ BGRTRIPLETS = 4	46
* 2.3.4 BM_ CMYKQUADS = & h0020	46
* 2.3.4 BM_ G3CHTRIPLETS = & h0204	46
* 2.3.4 BM_ GRAY = & h0209	47

* 2.3.4 BM_ KYMCQUADS = & h0305	47
* 2.3.4 BM_ LabTRIPLETS = & h0203	47
* 2.3.4 BM_ NAMED_ INDEX = & h0405	47
* 2.3.4 BM_ R10G10B10A2 = & h0701	47
* 2.3.4 BM_ R10G10B10A2_ XR = & h0702	48
* 2.3.4 BM_ R16G16B16A16_ FLOAT = & h0703	48
* 2.3.4 BM_ RGBTRIPLETS = 2	48
* 2.3.4 BM_ S2DOT13FIXED_ scARGB = & h0604	48
* 2.3.4 BM_ S2DOT13FIXED_ scRGB = & h0603	48
* 2.3.4 BM_ x555G3CH = & h0104	49
* 2.3.4 BM_ x555Lab = & h0103	49
* 2.3.4 BM_ x555RGB = 0	49
* 2.3.4 BM_ x555XYZ = & h0101	49
* 2.3.4 BM_ x555Yxy = & h0102	49
* 2.3.4 BM_ xBGRQUADS = & h0010	50
* 2.3.4 BM_ xG3CHQUADS = & h0304	50
* 2.3.4 BM_ xRGBQUADS = & h0008	50
* 2.3.4 BM_ XYZTRIPLETS = & h0201	50
* 2.3.4 BM_ YxyTRIPLETS = & h0202	50
* 2.3.4 CMM_ DESCRIPTION = 5	51
* 2.3.4 CMM_ DLL_ VERSION = 3	51
* 2.3.4 CMM_ DRIVER_ VERSION = 2	51
* 2.3.4 CMM_ FROM_ PROFILE = 0	51
* 2.3.4 CMM_ IDENT = 1	51
* 2.3.4 CMM_ LOGOICON = 6	51
* 2.3.4 CMM_ VERSION = 4	52
* 2.3.4 CMM_ WINDOWS_ DEFAULT = & h57696E20	52
* 2.3.4 CMM_ WIN_ VERSION = 0	52
* 2.3.4 ENABLE_ GAMUT_ CHECKING = & h10000	52
* 2.3.4 FAST_ TRANSLATE = & h40000	52
* 2.3.4 INDEX_ DONT_ CARE = 0	53
* 2.3.4 INTENT_ ABSOLUTE_ COLORIMETRIC = 3	53
* 2.3.4 INTENT_ PERCEPTUAL = 0	53
* 2.3.4 INTENT_ RELATIVE_ COLORIMETRIC = 1	53
* 2.3.4 INTENT_ SATURATION = 2	53
* 2.3.4 NORMAL_ MODE = 2	54
* 2.3.4 PRESERVEBLACK = & h100000	54
* 2.3.4 PROOF_ MODE = 1	54
* 2.3.4 SEQUENTIAL_ TRANSFORM = & h80800000	54
* 2.3.4 USE_ RELATIVE_ COLORIMETRIC = & h20000	54
* 2.3.4 WCS_ ALWAYS = & h200000	55
- 2.5 class WindowsICMEnumMBS	62

* 2.5.1 Attributes0 as Integer	63
* 2.5.1 Attributes1 as Integer	63
* 2.5.1 Classes as Integer	64
* 2.5.1 CMMType as Integer	64
* 2.5.1 ConnectionSpace as Integer	65
* 2.5.1 Creator as Integer	65
* 2.5.1 DataColorSpace as Integer	65
* 2.5.1 DeviceClass as Integer	65
* 2.5.1 DeviceName as String	66
* 2.5.1 DitheringMode as Integer	66
* 2.5.1 Fields as Integer	66
* 2.5.1 Manufacturer as Integer	67
* 2.5.1 MediaType as Integer	67
* 2.5.1 Model as Integer	67
* 2.5.1 Platform as Integer	68
* 2.5.1 ProfileFlags as Integer	68
* 2.5.1 RenderingIntent as Integer	68
* 2.5.1 ResolutionX as Integer	69
* 2.5.1 ResolutionY as Integer	69
* 2.5.1 Signature as Integer	69
* 2.5.2 ATTRIB_ MATTE = 2	69
* 2.5.2 ATTRIB_ TRANSPARENCY = 1	70
* 2.5.2 CLASS_ ABSTRACT = & h61627374	70
* 2.5.2 CLASS_ CAMP = & h6C616D70	70
* 2.5.2 CLASS_ COLORSPACE = & h73706163	70
* 2.5.2 CLASS_ GMMP = & h676D6D70	70
* 2.5.2 CLASS_ LINK = & h6C696E6B	70
* 2.5.2 CLASS_ MONITOR = & h6D6E7472	71
* 2.5.2 CLASS_ NAMED = & h6E6D636C	71
* 2.5.2 CLASS_ PRINTER = & h70727472	71
* 2.5.2 CLASS_ SCANNER = & h73636E72	71
* 2.5.2 ET_ ATTRIBUTES = & h02000	71
* 2.5.2 ET_ CLASS = & h00020	71
* 2.5.2 ET_ CMMTYPE = & h00010	72
* 2.5.2 ET_ CONNECTIONSPACE = & h00080	72
* 2.5.2 ET_ CREATOR = & h08000	72
* 2.5.2 ET_ DATACOLORSPACE = & h00040	72
* 2.5.2 ET_ DEVICECLASS = & h10000	72
* 2.5.2 ET_ DEVICENAME = & h00001	72
* 2.5.2 ET_ DITHERMODE = & h00004	73
* 2.5.2 ET_ MANUFACTURER = & h00800	73
* 2.5.2 ET_ MEDIATYPE = & h00002	73

* 2.5.2 ET_ MODEL = & h01000	73
* 2.5.2 ET_ PLATFORM = & h00200	73
* 2.5.2 ET_ PROFILEFLAGS = & h00400	73
* 2.5.2 ET_ RENDERINGINTENT = & h04000	74
* 2.5.2 ET_ RESOLUTION = & h00008	74
* 2.5.2 ET_ SIGNATURE = & h00100	74
* 2.5.2 FLAG_ DEPENDENTONDATA = 2	74
* 2.5.2 FLAG_ EMBEDDEDPROFILE = 1	74
* 2.5.2 FLAG_ ENABLE_ CHROMATIC_ ADAPTATION = & h02000000	74
* 2.5.2 SigMacintosh = & h4150504C	75
* 2.5.2 SigMicrosoft = & h4D534654	75
* 2.5.2 SigSGI = & h53474920	75
* 2.5.2 SigSolaris = & h53554E57	75
* 2.5.2 SigTaligent = & h54474E54	75
* 2.5.2 SPACE_ Lab = & h4C616220	76
* 2.5.2 SPACE_ XYZ = & h58595A20	76
– 2.4 class WindowsICMColorMBS	55
* 2.4.1 Channel(index as integer) as integer	55
* 2.4.2 a as Integer	56
* 2.4.2 b as Integer	56
* 2.4.2 black as Integer	56
* 2.4.2 blue as Integer	56
* 2.4.2 ch1 as Integer	56
* 2.4.2 ch2 as Integer	57
* 2.4.2 ch3 as Integer	57
* 2.4.2 cyan as Integer	57
* 2.4.2 gray as Integer	57
* 2.4.2 green as Integer	57
* 2.4.2 Index as Integer	58
* 2.4.2 L as Integer	58
* 2.4.2 magenta as Integer	58
* 2.4.2 red as Integer	58
* 2.4.2 XYZ_ X as Integer	58
* 2.4.2 XYZ_ Y as Integer	59
* 2.4.2 XYZ_ Z as Integer	59
* 2.4.2 yellow as Integer	59
* 2.4.2 Yxy_ x as Integer	59
* 2.4.2 Yxy_ y as Integer	59
* 2.4.2 Yxy_ YY as Integer	60
* 2.4.3 COLOR_ 3_ CHANNEL = 6	60
* 2.4.3 COLOR_ 5_ CHANNEL = 8	60

* 2.4.3 COLOR_6_CHANNEL = 9	60
* 2.4.3 COLOR_7_CHANNEL = 10	60
* 2.4.3 COLOR_8_CHANNEL = 11	61
* 2.4.3 COLOR_CMYK = 7	61
* 2.4.3 COLOR_GRAY = 1	61
* 2.4.3 COLOR_Lab = 5	61
* 2.4.3 COLOR_NAMED = 12	61
* 2.4.3 COLOR_RGB = 2	62
* 2.4.3 COLOR_XYZ = 3	62
* 2.4.3 COLOR_Yxy = 4	62
* 2.4.3 MAX_COLOR_CHANNELS = 8	62
– 2.9 module WindowsICMModuleMBS	98
* 2.9.1 AssociateColorProfileWithDevice(ProfileName as string, DeviceName as string) as boolean	98
* 2.9.1 DisassociateColorProfileFromDevice(ProfileName as string, DeviceName as string) as boolean	99
* 2.9.1 EnumColorProfiles(criteria as WindowsICMEnumMBS) as string()	100
* 2.9.1 GetColorDirectory as folderitem	101
* 2.9.1 GetStandardColorSpaceProfile(ProfileID as integer) as string	102
* 2.9.1 InstallColorProfile(file as folderitem) as boolean	103
* 2.9.1 RegisterCMM(cmmID as integer, file as folderitem) as boolean	103
* 2.9.1 SelectCMM(cmmID as integer) as boolean	104
* 2.9.1 SetStandardColorSpaceProfile(ProfileID as integer, ProfileName as folderitem) as boolean	104
* 2.9.1 UninstallColorProfile(ProfileName as string, DeleteFile as boolean = true) as boolean	105
* 2.9.1 UnregisterCMM(cmmID as integer) as boolean	105
* 2.9.2 CMM_DESCRIPTION = 5	106
* 2.9.2 CMM_DLL_VERSION = 3	106
* 2.9.2 CMM_DRIVER_VERSION = 2	106
* 2.9.2 CMM_IDENT = 1	106
* 2.9.2 CMM_LOGOICON = 6	106
* 2.9.2 CMM_VERSION = 4	107
* 2.9.2 CMM_WIN_VERSION = 0	107
* 2.9.2 LCS_sRGB = & h73524742	107
* 2.9.2 LCS_WINDOWS_COLOR_SPACE = & h57696E20	107
– 2.6 class WindowsICMNamedProfileInfoMBS	76
* 2.6.1 Count as Integer	76
* 2.6.1 CountDevCoordinates as Integer	76
* 2.6.1 Flags as Integer	77
* 2.6.1 Prefix as String	77
* 2.6.1 Suffix as String	77

– 2.8 class WindowsICMLogColorSpaceMBS	92
* 2.8.1 CStype as Integer	93
* 2.8.1 EndpointsBX as Integer	93
* 2.8.1 EndpointsBY as Integer	94
* 2.8.1 EndpointsBZ as Integer	94
* 2.8.1 EndpointsGX as Integer	94
* 2.8.1 EndpointsGY as Integer	94
* 2.8.1 EndpointsGZ as Integer	94
* 2.8.1 EndpointsRX as Integer	95
* 2.8.1 EndpointsRY as Integer	95
* 2.8.1 EndpointsRZ as Integer	95
* 2.8.1 Filename as String	95
* 2.8.1 GammaBlue as Double	95
* 2.8.1 GammaGreen as Double	96
* 2.8.1 GammaRed as Double	96
* 2.8.1 Intent as Integer	96
* 2.8.2 INTENT_ ABSOLUTE_ COLORIMETRIC = 3	97
* 2.8.2 INTENT_ PERCEPTUAL = 0	97
* 2.8.2 INTENT_ RELATIVE_ COLORIMETRIC = 1	97
* 2.8.2 INTENT_ SATURATION = 2	97
* 2.8.2 LCS_ CALIBRATED_ RGB = 0	97
* 2.8.2 LCS_ sRGB = & h73524742	98
* 2.8.2 LCS_ WINDOWS_ COLOR_ SPACE = & h57696E20	98

Chapter 2

Windows ICM

2.1 class WindowsICMSetupMBS

class WindowsICMSetupMBS

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** This class is made to query color matching setup information from the user.

Notes: This class contains information that the Setup function uses to initialize the ColorManagement dialog box. After the user closes the dialog box, Setup returns information about the user's selection in this class.

2.1.1 Methods

Setup as boolean

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The Setup function creates a Color Management dialog box that lets the user choose whether to enable color management, and if so, provides control over the color profiles used and over the rendering intent.

Notes: Returns true if the user clicked OK and false on any error or when dialog was cancelled.

2.1.2 Properties

DisplayName as String

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** A string naming the monitor to be used for color management.

Notes:

If this is not the name of a valid monitor, the first enumerated monitor is used.
(Read and Write property)

Flags as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** A set of bit flags used to initialize the dialog box.

Notes:

If set to 0 on entry, all controls assume their default states.

When the dialog box returns, these flags are set to indicate the user's input.

See CMS_* flag constants.
(Read and Write property)

MonitorProfile as String

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** A string in which to place the name of the user-selected monitor profile.

Notes:

If the CMS_SETMONITORPROFILE flag is used, this flag can also be used to select a profile other than the monitor default when the dialog is first displayed.
(Read and Write property)

Parent as Window

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** The parent window.
Notes: (Read and Write property)

PrinterName as String

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** A string naming the printer on which the image is to be rendered.
Notes:

If this is not a valid printer name, the default printer is used and named in the dialog.
(Read and Write property)

PrinterProfile as String

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** A string in which to place the name of the user-selected printer profile.
Notes:

If the CMS_ SETPRINTERPROFILE flag is used, this flag can also be used to select a profile other than the printer default when the dialog is first displayed.
(Read and Write property)

ProofingIntent as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The type of color management desired for the proofed image.
Notes:

Valid values are:

INTENT_PERCEPTUAL
INTENT_SATURATION
INTENT_RELATIVE_COLORIMETRIC

INTENT_ABSOLUTE_COLORIMETRIC

For more information, see Rendering Intents.

[http://msdn.microsoft.com/en-us/library/dd372183\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/dd372183(v=vs.85).aspx)
(Read and Write property)

RenderIntent as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The type of color management desired.

Notes:

Valid values are:

INTENT_PERCEPTUAL
INTENT_SATURATION
INTENT_RELATIVE_COLORIMETRIC
INTENT_ABSOLUTE_COLORIMETRIC

For more information, see Rendering Intents.

[http://msdn.microsoft.com/en-us/library/dd372183\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/dd372183(v=vs.85).aspx)
(Read and Write property)

SourceName as String

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** An application-specified string which describes the source profile of the item for which color management is to be performed.

Notes:

If this is "", the Image Source control displays the name of the Windows default color profile.
(Read and Write property)

TargetProfile as String

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** A string in which to place the name of the user-selected target profile for proofing.

Notes:

If the CMS_ SETTARGETPROFILE flag is used, this flag can also be used to select a profile other than the printer default when the dialog is first displayed.

(Read and Write property)

2.1.3 Events

Apply

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** This event is invoked when the Apply button of the Color Management dialog box is selected.

Idle

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** This event is called regularly if the CMS_ USEHOOK flag is used.

2.1.4 Constants

CMS_ DISABLEICM = 1

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flag constants.

Notes: If set on entry, this flag indicates that the "Enable Color Management" check box is cleared, disabling all other controls. If set on exit, it means that the user does not wish color management performed.

CMS_DISABLEINTENT = 1024

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flag constants.

CMS_DISABLERENDERINTENT = 2048

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flag constants.

CMS_ENABLEPROOFING = 2

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flag constants.

Notes: If set on entry, this flag indicates that the Proofing controls are to be enabled, and the Proofing check box is checked. If set on exit, it means that the user wishes to perform color management for a different target device than the selected printer.

CMS_SETMONITORPROFILE = 16

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flag constants.

Notes: If set on entry, this flag indicates that the color management profile named in the MonitorProfile member is to be the initial selection in the monitor profile control. If the specified profile is not associated with the monitor, this flag is ignored, and the default profile for the monitor is used.

CMS_SETPRINTERPROFILE = 32

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flag constants.

Notes: If set on entry, this flag indicates that the color management profile named in the PrinterProfile member is to be the initial selection in the printer profile control. If the specified profile is not associated with the printer, this flag is ignored, and the default profile for the printer is used.

CMS_ SETPROOFINTENT = 5

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flag constants.

Notes: Ignored unless CMS_ENABLEPROOFING is also set. If set on entry, and CMS_ENABLEPROOFING is also set, this flag indicates that the ProofingIntent member is to be used to initialize the Target Rendering Intent control. Otherwise, the control defaults to Picture rendering. This flag is set on exit if proofing is enabled.

CMS_ SETRENDERINTENT = 4

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flag constants.

Notes: If set on entry, this flag indicates that the RenderIntent member contains the value to use to initialize the Rendering Intent control. Otherwise, the control defaults to Picture rendering. This flag is set on exit if WCS is enabled.

CMS_ SETTARGETPROFILE = 64

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flag constants.

Notes: If set on entry, this flag indicates that the color profile named in the TargetProfile member is to be the initial selection in the target profile control. If the specified profile is not installed, this flag is ignored, and the default profile for the printer is used. If the printer has no default profile, then the first profile in alphabetical order will be displayed.

CMS_ USEAPPLYCALLBACK = 256

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flag constants.

Notes: If set on entry, this flag indicates that the SetupColorMatching function should call the Apply event.

CMS_USEDESCRIPTION = 512

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flag constants.

Notes: If set on entry, this flag instructs the Setup function to retrieve the profile description contained in the profile description tags (See ICC Profile Format Specification v3.4). It will insert them into the Monitor Profile, Printer Profile, Emulated Device Profile edit boxes in the Color Management common dialog box.

CMS_USEHOOK = 128

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flag constants.

Notes: The idle event is called regularly if this flag is set.

INTENT_ABSOLUTE_COLORIMETRIC = 3

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the rendering intent constants.

Notes: Maintain the white point. Match the colors to their nearest color in the destination gamut.

INTENT_PERCEPTUAL = 0

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the rendering intent constants.

Notes: Maintain contrast. Used for photographs and natural images.

INTENT_RELATIVE_COLORIMETRIC = 1

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the rendering intent constants.

Notes: Maintain colorimetric match. Used for graphic designs and named colors.

INTENT_SATURATION = 2

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the rendering intent constants.

Notes: Maintain saturation. Used for business charts and other situations in which undithered colors are required.

2.2 class WindowsICMProfileMBS

class WindowsICMProfileMBS

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The class for a color profile.

2.2.1 Methods

ConvertColorNameToIndex(name as string) as integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The CMConvertColorNameToIndex function converts color names in a named color space to index numbers in a color profile.

Notes:

name: The name of the color.

Returns the color index.

This function is required in the default CMM. It is optional for all other CMMs.

ConvertIndexToColorName(index as integer) as string

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The ConvertIndexToColorName transforms indices in a color space to an array of names in a named color space.

Notes: This function is required in the default CMM. It is optional for all other CMMs.

CountColorProfileElements as integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The GetCountColorProfileElements function retrieves the number of tagged elements in a given color profile.

Example:

```
dim file as FolderItem = SpecialFolder.Desktop.Child("test.icc")
dim w as WindowsICMProfileMBS = WindowsICMProfileMBS.OpenProfileFile(file, WindowsICMProfileMBS.PROFILE_READ, WindowsICMProfileMBS.FILE_SHARE_READ, WindowsICMProfileMBS.OPEN_EXISTING)

MsgBox "CountColorProfileElements: " + str(w.CountColorProfileElements)
```

Notes:

Returns number of tagged elements in the profile or 0 on any error.
 This function will fail if hProfile is not a valid ICC profile.
 This function does not support Windows Color System (WCS) profiles CAMP, DMP, and GMMP.

GetColorProfileElement(tag as integer) as string

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The GetColorProfileElement function copies data from a specified tagged profile element of a specified color profile into a buffer.

Example:

```
dim file as FolderItem = SpecialFolder.Desktop.Child("sRGB Profile.icc")
dim w as WindowsICMProfileMBS = WindowsICMProfileMBS.OpenProfileFile(file, WindowsICMProfileMBS.PROFILE_READ, WindowsICMProfileMBS.FILE_SHARE_READ, WindowsICMProfileMBS.OPEN_EXISTING)

MsgBox w.GetColorProfileElement(& h64657363) // that's the code for desc, the description
```

Notes:

tag: Identifies the tagged element from which to copy.

This function will fail if Profile is not a valid International Color Consortium (ICC) profile.

This function does not support Windows Color System (WCS) profiles CAMP, DMP, and GMMP; because profile elements are implicitly associated with, and hard coded to, ICC tag types and there exist many robust XML parsing libraries.

GetColorProfileElementTag(index as integer) as integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The GetColorProfileElementTag function retrieves the tag name specified by dwIndex in the tag table of a given International Color Consortium (ICC) color profile, where Index is a one-based index into that table.

Example:

```
dim file as FolderItem = SpecialFolder.Desktop.Child("test.icc")
dim w as WindowsICMProfileMBS = WindowsICMProfileMBS.OpenProfileFile(file, WindowsICMProfileMBS.PRO-
FILE_ READ, WindowsICMProfileMBS.FILE_ SHARE_ READ, WindowsICMProfileMBS.OPEN_ EXIST-
ING)
dim list(-1) as string
dim c as integer = w.CountColorProfileElements-1

for i as integer = 0 to c
list.Append DecodingFromHexMBS(hex(w.GetColorProfileElementTag(i)))
next

MsgBox "Tags: "+Join(list, ", ")
```

Notes:

Index: Specifies the one-based index of the tag to retrieve.

This function will fail if Profile is not a valid ICC profile.

GetColorProfileElementTag can be used to enumerate all tags in a profile after getting the number of tags in the profile using GetCountColorProfileElements.

This function does not support Windows Color System (WCS) profiles CAMP, DMP, and GMMP; because profile elements are implicitly associated with, and hard coded to, ICC tag types and there exist many robust XML parsing libraries.

GetNamedProfileInfo as WindowsICMNamedProfileInfoMBS

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The GetNamedProfileInfo function retrieves information about the International Color Consortium (ICC) named color profile that is specified in the first parameter.

Notes:

This function will fail if hProfile is not a valid ICC profile.

This function does not support Windows Color System (WCS) profiles CAMP, DMP, and GMMP; because named profiles are explicit ICC profile types.

Returns nil on any error.

GetProfileData as string

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Given a handle to an open color profile, the GetColorProfileFromHandle function will copy the contents of the profile into a buffer supplied by the application.

Notes:

If the handle is a Windows Color System (WCS) handle, then the DMP is returned and the CAMP and GMMP associated with the HPROFILE are ignored.

Returns the data or an empty string on any error.

IsColorProfileTagPresent(tag as integer) as boolean

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The IsColorProfileTagPresent function reports whether a specified International Color Consortium (ICC) tag is present in the specified color profile.

Notes:

tag: Specifies the ICC tag to check.

Returns true if the tag is valid and false if not.

This function will fail if Profile is not a valid ICC profile.

This function does not support Windows Color System (WCS) profiles CAMP, DMP, and GMMP; because profile elements are implicitly associated with and hard coded to ICC tag types and there exist many robust XML parsing libraries.

IsValid as boolean

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The IsValid function reports whether the given profile is a valid ICC profile that can be used for color management.

Example:

```
dim file as FolderItem = SpecialFolder.Desktop.Child("test.icc")
dim w as WindowsICMProfileMBS = WindowsICMProfileMBS.OpenProfileFile(file, WindowsICMProfileMBS.PRO-
FILE_ READ, WindowsICMProfileMBS.FILE_ SHARE_ READ, WindowsICMProfileMBS.OPEN_ EXIST-
ING)

MsgBox "Valid: " + str(w.IsValid)
```

Notes:

Returns true if the profile is valid.

Only the Windows default CMM is required to export this function; it is optional for all other CMMs.

If a CMM does not support this function, Windows uses the default CMM to validate the profile.

OpenProfileData(data as string, DesiredAccess as integer) as WindowsICMProfileMBS

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The OpenColor-Profile function opens or create a color profile.

Notes:

file: The file where to load profile from.

DesiredAccess: Specifies how to access the given profile. This parameter must take one the following constant values.

Value	Meaning
PROFILE_READ	Opens the profile for read access.
PROFILE_READWRITE	Opens the profile for both read and write access. Has no effect for WCS XML profiles.

Returns the profile on success and nil on error.

For ICC and WCS profiles, a CAMP and GMMP are provided by the function based on the current default CAMP and GMMP in the registry.

When OpenColorProfile encounters an ICC profile with an embedded WCS profile, and if the dwType member within the Profile structure does not take the value DONT_USE_EMBEDDED_WCS_PROFILES, it should extract and use the WCS profile(s) contained in this WcsProfilesTag. The HPROFILE returned would be a WCS HPROFILE.

When the function opens the ICC profile, it will look for a WcsProfilesTag and, if there is one, it will extract and use the original WCS profiles contained therein. (See WcsCreateIccProfile.)

An profile with WCS profile information is derived from a DMP by acquiring the default CAMP and default GMMP from the registry. An HPROFILE is a composition of a DMP, CAMP and GMMP.

OpenProfileFile(file as folderitem, DesiredAccess as integer, ShareMode as integer, CreationMode as integer) as WindowsICMProfileMBS

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The OpenColorProfile function opens or create a color profile.

Notes:

file: The file where to load profile from.

DesiredAccess: Specifies how to access the given profile. This parameter must take one the following constant values.

ShareMode: Specifies how the profile should be shared, if the profile is contained in a file. A value of zero prevents the profile from being shared at all. The parameter can contain one or both of the following con-

Value	Meaning
PROFILE_READ	Opens the profile for read access.
PROFILE_READWRITE	Opens the profile for both read and write access. Has no effect for WCS XML profiles.

starts (combined by addition or logical OR).

Value	Meaning
FILE_SHARE_READ	Other open operations can be performed on the profile for read access.
FILE_SHARE_WRITE	Other open operations can be performed on the profile for write access. Has no effect for WCS XML profiles.

CreationMode: Specifies which actions to take on the profile while opening it, if it is contained in a file. This parameter must take one of the following constant values.

Value	Meaning
CREATE_NEW	Creates a new profile. Fails if the profile already exists.
CREATE_ALWAYS	Creates a new profile. Overwrites the profile if it exists.
OPEN_EXISTING	Opens the profile. Fails if it does not exist
OPEN_ALWAYS	Opens the profile if it exists. For ICC profiles, if the profile does not exist, creates the profile. For WCS XML profiles, if the profile does not exist, returns an error.
TRUNCATE_EXISTING	Opens the profile, and truncates it to zero bytes, returning a blank ICC profile. Fails if the profile doesn't exist.

Returns the profile on success and nil on error.

For ICC and WCS profiles, a CAMP and GMMP are provided by the function based on the current default CAMP and GMMP in the registry.

When OpenColorProfile encounters an ICC profile with an embedded WCS profile, and if the dwType member within the Profile structure does not take the value DONT_USE_EMBEDDED_WCS_PROFILES, it should extract and use the WCS profile(s) contained in this WcsProfilesTag. The HPROFILE returned would be a WCS HPROFILE.

CreationMode flags CREATE_NEW, CREATE_ALWAYS, and TRUNCATE_EXISTING, will always return blank ICC HPROFILES. If other CreationMode flags are present, InternalOpenColorProfile is called (using the flags as provided by the API) to determine whether the profile is ICC or WCS XML.

Within the ICC code path, an ICC Profile is returned using the requested sharing, access and creation flags as specified in the tables above.

Within the WCS path, the CreationMode flag `OPEN_ALWAYS` will fail if the profile doesn't exist, since WCS profiles cannot be created or edited within the WCS architecture (they must be edited outside of it, using MSXML6). For the same reason, `dwShareMode` flag `FILE_SHARE_WRITE`, and `dwDesiredAccess` flag `PROFILE_READWRITE` are ignored within the WCS path.

When the function opens the ICC profile, it will look for a `WcsProfilesTag` and, if there is one, it will extract and use the original WCS profiles contained therein. (See `WcsCreateIccProfile`.)

An profile with WCS profile information is derived from a DMP by acquiring the default CAMP and default GMMP from the registry. An HPROFILE is a composition of a DMP, CAMP and GMMP.

OpenProfilePath(path as string, DesiredAccess as integer, ShareMode as integer, CreationMode as integer) as WindowsICMProfileMBS

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The `OpenColorProfile` function opens or create a color profile.

Notes:

`file`: The file where to load profile from.

`DesiredAccess`: Specifies how to access the given profile. This parameter must take one the following constant values.

Value	Meaning
<code>PROFILE_READ</code>	Opens the profile for read access.
<code>PROFILE_READWRITE</code>	Opens the profile for both read and write access. Has no effect for WCS XML profiles.

`ShareMode`: Specifies how the profile should be shared, if the profile is contained in a file. A value of zero prevents the profile from being shared at all. The parameter can contain one or both of the following constants (combined by addition or logical OR).

Value	Meaning
<code>FILE_SHARE_READ</code>	Other open operations can be performed on the profile for read access.
<code>FILE_SHARE_WRITE</code>	Other open operations can be performed on the profile for write access. Has no effect for WCS XML profiles.

`CreationMode`: Specifies which actions to take on the profile while opening it, if it is contained in a file. This parameter must take one of the following constant values.

Value	Meaning
CREATE_NEW	Creates a new profile. Fails if the profile already exists.
CREATE_ALWAYS	Creates a new profile. Overwrites the profile if it exists.
OPEN_EXISTING	Opens the profile. Fails if it does not exist
OPEN_ALWAYS	Opens the profile if it exists. For ICC profiles, if the profile does not exist, creates the profile. For WCS XML profiles, if the profile does not exist, returns an error.
TRUNCATE_EXISTING	Opens the profile, and truncates it to zero bytes, returning a blank ICC profile. Fails if the profile doesn't exist.

Returns the profile on success and nil on error.

For ICC and WCS profiles, a CAMP and GMMP are provided by the function based on the current default CAMP and GMMP in the registry.

When `OpenColorProfile` encounters an ICC profile with an embedded WCS profile, and if the `dwType` member within the Profile structure does not take the value `DONT_USE_EMBEDDED_WCS_PROFILES`, it should extract and use the WCS profile(s) contained in this `WcsProfilesTag`. The `HPROFILE` returned would be a WCS `HPROFILE`.

CreationMode flags `CREATE_NEW`, `CREATE_ALWAYS`, and `TRUNCATE_EXISTING`, will always return blank ICC `HPROFILEs`. If other CreationMode flags are present, `InternalOpenColorProfile` is called (using the flags as provided by the API) to determine whether the profile is ICC or WCS XML.

Within the ICC code path, an ICC Profile is returned using the requested sharing, access and creation flags as specified in the tables above.

Within the WCS path, the CreationMode flag `OPEN_ALWAYS` will fail if the profile doesn't exist, since WCS profiles cannot be created or edited within the WCS architecture (they must be edited outside of it, using `MSXML6`). For the same reason, `dwShareMode` flag `FILE_SHARE_WRITE`, and `dwDesiredAccess` flag `PROFILE_READWRITE` are ignored within the WCS path.

When the function opens the ICC profile, it will look for a `WcsProfilesTag` and, if there is one, it will extract and use the original WCS profiles contained therein. (See `WcsCreateIccProfile`.)

An profile with WCS profile information is derived from a DMP by acquiring the default CAMP and default GMMP from the registry. An `HPROFILE` is a composition of a DMP, CAMP and GMMP.

SetColorProfileHeader(header as WindowsICMProfileHeaderMBS) as boolean

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The SetColorProfileHeader function sets the header data in a specified ICC color profile.

Notes:

Header: the profile header data to write to the specified profile.

Returns true on success.

This function will fail if Profile is not a valid ICC profile.

If the color profile was not opened with read/write permission, SetColorProfileHeader fails.

SetColorProfileHeader overwrites the current header in the ICC profile.

This function does not support Windows Color System (WCS) profiles CAMP, DMP, and GMMP; because profile elements are implicitly associated with and hard coded to ICC tag types and there exist many robust XML parsing libraries.

2.2.2 Properties

ColorProfileHeader as WindowsICMProfileHeaderMBS

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The GetColorProfileHeader function retrieves or derives ICC header structure from either ICC color profile or WCS XML profile.

Notes:

Drivers and applications should assume returning TRUE only indicates that a properly structured header is returned. Each tag will still need to be validated independently using either legacy ICM2 APIs or XML schema APIs.

To determine whether the header is derived from an ICC or DMP profile handle, check the header signature (header bytes 36-39). If the signature is "acsp" (big endian) then an ICC profile was used. If the signature is "cdmp" (big-endian) then a DMP was used.

The distinguishing features that identify a header as having been "synthesized" for a WCS DMP are:

```
WindowsICMPProfileHeaderMBS.Signature = 'pmdc' (little endian = big endian 'cdmp')
WindowsICMPProfileHeaderMBS.CMMType = '1scw' (little endian = big endian 'wcs1').
(Read only property)
```

Handle as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The internal reference to the object.

Notes: (Read and Write property)

2.2.3 Constants

CREATE_ALWAYS = 2

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the creation mode constants.

Notes: Creates a new profile. Overwrites the profile if it exists.

CREATE_NEW = 1

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the creation mode constants.

Notes: Creates a new profile. Fails if the profile already exists.

FILE_SHARE_READ = 1

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile sharing flags.

Notes: Other open operations can be performed on the profile for read access.

FILE_SHARE_WRITE = 2

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile sharing flags.

Notes: Other open operations can be performed on the profile for write access. Has no effect for WCS XML profiles.

OPEN_ALWAYS = 4

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the creation mode constants.

Notes: Opens the profile if it exists. For ICC profiles, if the profile does not exist, creates the profile. For WCS XML profiles, if the profile does not exist, returns an error.

OPEN_EXISTING = 3

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the creation mode constants.

Notes: Opens the profile. Fails if it does not exist

PROFILE_READ = 1

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile access flags.

Notes: Opens the profile for read access.

PROFILE_READWRITE = 2

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile access flags.

Notes: Opens the profile for both read and write access. Has no effect for WCS XML profiles.

TRUNCATE_EXISTING = 5

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the creation mode constants.

Notes: Opens the profile, and truncates it to zero bytes, returning a blank ICC profile. Fails if the profile doesn't exist.

2.3 class WindowsICMTransformMBS

class WindowsICMTransformMBS

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The class for a color transformation.

2.3.1 Methods

CheckColors(InputColors() as WindowsICMColorMBS, ctInput as integer, Results() as integer) as boolean

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The CheckColors function determines whether the colors in an array lie within the output gamut of a specified transform.

Notes:

InputColors: an array of colors.

ctInput: Specifies the input color type.

Results: An array of nColors bytes that receives the results of the test.

If this function succeeds, the return value is TRUE.

If this function fails, the return value is FALSE. For extended error information, call GetLastError.

Remarks

If the input color type is not compatible with the color transform, CheckColors fails.

The function places results of the tests in the array pointed to by paResult. Each byte in the array corresponds to a COLOR element in the array pointed to by paInputColors and has an unsigned value between 0 and 255. The value 0 denotes that the color is in gamut, while a nonzero value denotes that it is out of gamut. For any integer n such that $0 < n < 255$, a result value of n+1 indicates that the corresponding color is at least as far out of gamut as would be indicated by a result value of n.

The out-of-gamut information in the gamut tags created in WCS use the perceptual color distance in CIECAM02, which is the mean square root in CIECAM02 Jab space. The distance in the legacy ICC profile gamut tags is the mean square root in CIELAB space. We recommend that you use the CIECAM02 space when it is available because it provides more perceptually accurate distance metrics.

Constructor(LogColorSpace as WindowsICMLogColorSpaceMBS, DestProfile as WindowsICMProfileMBS, TargetProfile as WindowsICMProfileMBS, Flags as integer)

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The CreateColorTransform function creates a color transform that applications can use to perform color management.

Notes:

LogColorSpace: The input colorspace.

DestProfile: The profile of the destination device. The function determines whether the profile contains International Color Consortium (ICC) or Windows Color System (WCS) profile information.

TargetProfile: The profile of the target device. The function determines whether the profile contains ICC or WCS profile information.

Flags: Specifies flags to used control creation of the transform. See Remarks.

On success the handle property is not zero.

If the target profile is nil, the transform goes from the source logical color space to the destination profile.

If the target profile is given, the transform goes from the source logical color space to the target profile and then to the destination profile. This allows previewing output meant for the target device on the destination device.

The values in Flags are intended as hints only. The color management module must determine the best way to use them.

Windows Vista: Three new flags have been added that can be used with dwFlags:

PRESERVEBLACK	If this bit is set, the transform engine inserts the appropriate black generation GMMP as the last GMMP in the transform sequence. This flag only works in a pure WCS transform.
SEQUENTIAL_TRANSFORM	If this bit is set, each step in the WCS processing pipeline is performed for every pixel in the image and no optimized color transform is built. This flag only works in a pure WCS transform.

Restrictions: A transform created with the SEQUENTIAL_TRANSFORM flag set may only be used in the thread on which it was created and only for one color translation call at a time. COM must be initialized prior to creating the sequential transform and must remain initialized for the lifetime of the transform object.

WCS_ALWAYS If this bit is set, even all-ICC transforms will use the WCS code path.

For details, see CMM Transform Creation Flags. All of the flags mentioned there are supported for all types of transforms, except for FAST_TRANSLATE, which only works in a pure ICC-to-ICC transform.

The CreateColorTransform function is used outside of a device context. Colors may shift when transforming from a color profile to the same color profile. This is due to precision errors. Therefore, a color transform should not be performed under these circumstances.

The B2Ax tags are required for any profile that is the target of a transform.

WCS transform support for ICC ColorSpace profiles is limited to RGB colorspace profiles. The following ICC profile types cannot be used in a CITE-processed transform, either a mixed WCS/ICC transform or an all-ICC transform with WCS_ALWAYS set:

- Non-RGB ColorSpace profiles
- NamedColor profiles
- n-channel profiles (where n >8)
- DeviceLink profiles

Abstract profiles

See also:

- 2.3.1 Constructor(Profiles() as WindowsICMProfileMBS, Intents() as integer, Flags as integer, indexPreferredCMM as integer) 36

Constructor(Profiles() as WindowsICMProfileMBS, Intents() as integer, Flags as integer, indexPreferredCMM as integer)

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The CreateMultiProfileTransform function accepts an array of profiles or a single device link profile and creates a color transform that applications can use to perform color mapping.

Notes:

Profiles: An array of profiles to be used. The function determines whether the HPROFILEs contain International Color Consortium (ICC) or Windows Color System (WCS) profile information and processes them appropriately. When valid WCS profiles are returned by OpenColorProfile, these profiles contain the combination of DMP, CAMP, and GMMP profiles.

Intent: An array of intents to use. Each intent is one of the following values:

```
INTENT_PERCEPTUAL
INTENT_SATURATION
INTENT_RELATIVE_COLORIMETRIC
INTENT_ABSOLUTE_COLORIMETRIC
```

GMMPs are a generalization of intents. There are two possible sources of intents: the "destination" profile and the intent list parameter to CreateMultiProfileTransform. The term "destination" is not used since all but two of the profiles in the profile list parameter will serve as first destination and then source.

For more information, see Rendering Intents.

nIntents

Specifies the number of elements in the intents array: can either be 1 or the same value as nProfiles. For profile arrays that contain any WCS profiles, the first rendering intent is ignored and only nProfiles -1 elements are used for these profile arrays. The maximum number of nIntents is 10.

Flags: Specifies flags used to control creation of the transform.

indexPreferredCMM: Specifies the one-based index of the color profile that indicates what color management module (CMM) to use. The application developer may allow Windows to choose the CMM by setting this parameter to INDEX_DONT_CARE. See Using Color Management Modules (CMM) Third party CMMs are only available for ICC workflows. Profile arrays containing WCS profiles will ignore this flag. It is also ignored when only ICC profiles are used and when the WCS_ALWAYS flag is used.

On success the handle property is not zero.

If a device link profile is being used, the function will fail if Profiles contains more than one value.

The array of intents specifies how profiles should be combined. The nth intent is used for combining the nth profile in the array. If only one intent is specified, it is used for the first profile, and all other profiles are combined using Match intent.

The values in Flags are intended as hints only. The color management module must determine the best way to use them.

Windows Vista: Three new flags have been added that can be used with dwFlags:

PRESERVEBLACK	If this bit is set, the transform engine inserts the appropriate black generation GMMP as the last GMMP in the transform sequence. This flag only works in a pure WCS transform.
SEQUENTIAL_TRANSFORM	If this bit is set, each step in the WCS processing pipeline is performed for every pixel in the image and no optimized color transform is built. This flag only works in a pure WCS transform.

Restrictions: A transform created with the SEQUENTIAL_TRANSFORM flag set may only be used in the thread on which it was created and only for one color translation call at a time. COM must be initialized prior to creating the sequential transform and must remain initialized for the lifetime of the transform object.

WCS_ALWAYS If this bit is set, even all-ICC transforms will use the WCS code path.

For details, see CMM Transform Creation Flags. All of the flags mentioned there are supported for all types of transforms, except for FAST_TRANSLATE and USE_RELATIVE_COLORIMETRIC, which only work in a pure ICC-to-ICC transform.

The CreateMultiProfileTransform function is used outside of a device context. Colors may shift when trans-

forming from a color profile to the same color profile. This is due to precision errors. Therefore, a color transform should not be performed under these circumstances.

We recommend that there be only one GMMP between a source and destination DMP. Gamut boundary descriptions (GBDs) are created from the DMP/CAMP combinations. The subsequent GMMPs use the GDBs prior to them in the processing chain until there exists a DMP/CAMP GBD next in the sequence to be used. For example, assume a sequence DMP1, CAMP1, GMMP1, GMMP2, GMMP3, DMP2, CAMP2, GMMP4, GMMP5, CAMP3, DMP3. Then GMMP1, GMMP2 use GBD1 as their source and destination. Then GMMP3 uses GBD1 as source and GBD2 as destination. Then GMMP4 uses GBD2 as source and destination. Finally GMMP5 uses GBD2 as source and GBD3 as destination. This assumes no GMMP is identical to one next to it.

For WCS profiles, we recommend that the rendering intents be set to `DWORD_MAX` in order to use the GMMP within the WCS profile handle. This is because the array of rendering intents takes precedence over the rendering intents or gamut mapping models specified or contained in the profiles specified by the `PROFILES`. The array of rendering intents references the default GMMP for those rendering intents. Ideally, only one gamut mapping is performed between a source and destination device by setting one or the other GMMP to `NULL` when creating the `HPROFILE` with WCS profile information. Any legacy application that uses a WCS DMP will invoke a sequence of GMMPs. GDBs are chosen based on DMPs and CAMPs. For intermediate GMMP gamut boundaries, the source and destination GDBs are used.

In summary, if `ubound(Intent)=0`, then the first GMM is set based on the GMMP that is set as default* for the `padwIntent` value, unless that value is `DWORD_MAX`, in which case the embedded GMM information from the second profile is used (The embedded GMM information is either a GMMP or, in the case of an ICC profile, the baseline GMM corresponding to** the intent from the profile header). The remainder of the GMMs are set based on the GMMP that is set as default* for `RelativeColorimetric`.

If `ubound(Intent) = ubound(Profiles) - 1`, then each GMM is set based on the GMMP that is set as default* for the value in the `padwIntent` array at the corresponding index, except where `padwIntent` values are `DWORD_MAX`. For values in the `padwIntent` array that are `DWORD_MAX`, the GMMs at corresponding positions are set based on the embedded GMM information from the second of the two profiles whose gamuts are mapped by the GMM. (Again, the embedded GMM information is either a GMMP or, in the case of an ICC profile, the baseline GMM corresponding to** the intent from the profile header).

If `ubound(Intent) = ubound(Profiles)`, then first intent is ignored and function behaves as it does in the case when `ubound(Intent) = ubound(Profiles) - 1`.

Any other combination of `padwIntents` and `nIntents` will return an error.

* "set as default" means that the default GMMP is queried using `WcsGetDefaultColorProfile` with its `profileManagementScope` parameter set to `WCS_PROFILE_MANAGEMENT_SCOPE_CURRENT_USER`. This may return either current-user or system-wide defaults as described in the documentation for `WcsGetDefaultColorProfile`.

** "GMM corresponding to" does not mean "GMM from the GMMP set as default for". Instead it means "a constant association between ICC profile intents and baseline GMM algorithms."

WCS transform support for ICC ColorSpace profiles is limited to RGB colorspace profiles. The following ICC profile types cannot be used in a CITE-processed transform, either a mixed WCS/ICC transform or an all-ICC transform with `WCS_ALWAYS` set:

* Non-RGB ColorSpace profiles
 NamedColor profiles
 n-channel profiles (where $n > 8$)
 DeviceLink profiles
 Abstract profiles
 See also:

- 2.3.1 Constructor(LogColorSpace as WindowsICMLogColorSpaceMBS, DestProfile as WindowsICMProfileMBS, TargetProfile as WindowsICMProfileMBS, Flags as integer) 34

GetCMMInfo(what as integer) as integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The `GetCMMInfo` function retrieves various information about the color management module (CMM) that created the specified color transform.

Notes:

what: Specifies the information to be retrieved. This parameter can take one of the following constant values.

Value	Meaning
<code>CMM_WIN_VERSION</code>	Retrieves the version of Windows targeted by the color management module (CMM).
<code>CMM_DLL_VERSION</code>	Retrieves the version number of the CMM.
<code>CMM_IDENT</code>	Retrieves the CMM signature registered with the International Color Consortium (ICC).

If this function succeeds, the return value is the information specified in `What`.

If this function fails, the return value is zero.

TranslateBitmapBits(SrcBits as memoryblock, InputType as integer, Width as integer, Height as integer, InputRowBytes as integer, DestBits as integer, DestType as integer, DestRowBytes as integer) as boolean

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The TranslateBitmapBits function translates the colors of a bitmap having a defined format so as to produce another bitmap in a requested format.

Notes:

SrcBits: Pointer to the bitmap to translate.

InputType: Specifies the format of the input bitmap. Use one of the BM_* constants.

Width: Specifies the number of pixels per scan line in the input bitmap.

Height: Specifies the number of scan lines in the input bitmap.

InputRowBytes: Specifies the number of bytes from the beginning of one scan line to the beginning of the next in the input bitmap; if set to zero, the function assumes that scan lines are padded so as to be DWORD-aligned.

DestBits: Pointer to the buffer in which to place the translated bitmap.

DestType: Specifies the format of the output bitmap. Use one of the BM_* constants.

DestRowBytes: Specifies the number of bytes from the beginning of one scan line to the beginning of the next in the output bitmap; if set to zero, the function assumes that scan lines should be padded to be DWORD-aligned.

If this function succeeds, the return value is TRUE.

If this function fails, the return value is FALSE.

Remarks

If the input and output formats are not compatible with the color transform, this function fails.

When either of the floating point BMFORMATs, BM_32b_scARGB or BM_32b_scRGB are used, the color data being translated should not contain NaN or infinity. NaN and infinity are not considered to represent legitimate color component values, and the result of translating pixels containing NaN or infinity is meaningless in color terms. NaN or infinity values in the color data being processed will be handled silently, and an error will not be returned.

TranslateColors(InputColors() as WindowsICMColorMBS, ctInput as integer, OutputColors() as WindowsICMColorMBS, ctOutput as integer) as boolean

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The TranslateColors function translates an array of colors from the source color space to the destination color space as defined by a color transform.

Notes:

InputColors: The input color.

ctInput: Specifies the input color type.

OutputColors: The output color.

ctOutput: Specifies the output color type.

Returns true on success and false on failure.

TranslatePictures(InputPicture as picture, OutputPicture as picture) as boolean

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The TranslateBitmapBits function translates the colors of a bitmap having a defined format so as to produce another bitmap in a requested format.

Notes:

InputPicture: input picture

OutputPicture: output picture

If this function succeeds, the return value is TRUE.

If this function fails, the return value is FALSE.

Make sure the pictures have the same size.

If the input and output formats are not compatible with the color transform, this function fails.

2.3.2 Properties

Handle as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The internal handle for the transformation.

Notes: (Read and Write property)

2.3.3 Events

Progress(Maximum as integer, Current as integer) as boolean

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** The progress event for long operations.

2.3.4 Constants

BEST_MODE = 3

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flags passed when creating a transformation.

Notes: Transform will be used for the display of the highest-quality image possible on the target device.

BM_10b_G3CH = & h0404

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 32 bits per pixel. 10 bits are used for each color channel. The 2 most significant bits are ignored.

BM_10b_Lab = & h0403

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 32 bits per pixel. 10 bits are used for each color channel. The 2 most significant bits are ignored.

BM_10b_RGB = 9

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 32 bits per pixel. 10 bits are used for each color channel. The 2 most significant bits are ignored.

BM_10b_XYZ = & h0401

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 32 bits per pixel. 10 bits are used for each color channel. The 2 most significant bits are ignored.

BM_10b_Yxy = & h0402

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 32 bits per pixel. 10 bits are used for each color channel. The 2 most significant bits are ignored.

BM_16b_G3CH = & h0504

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 64 bits per pixel. 16 bits are used for the gray-scale value. Each of the three color channels uses 16 bits.

BM_16b_GRAY = & h0505

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 64 bits per pixel. 16 bits are used for the gray-scale value. All other bits are ignored.

BM_16b_Lab = & h0503

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 64 bits per pixel. 16 bits are used for the gray-scale value. Each of the three color channels uses 16 bits.

BM_16b_RGB = 10

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 64 bits per pixel. 16 bits are used for the gray-scale value. Each of the three color channels uses 16 bits.

BM_16b_XYZ = & h0501

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 64 bits per pixel. 16 bits are used for the gray-scale value. Each of the three color channels uses 16 bits.

BM_16b_Yxy = & h0502

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 64 bits per pixel. 16 bits are used for the gray-scale value. Each of the three color channels uses 16 bits.

BM_32b_scARGB = & h0602

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 128 bits per pixel. 32 bits are used for each color channel, as defined by the IEEE 32-bit floating point standard.

BM_32b_scRGB = & h0601

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 96 bits per pixel. 32 bits are used for each color channel, as defined by the IEEE 32-bit floating point standard.

BM_565RGB = 1

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 16 bits per pixel. 5 bits are used for red, 6 for green, and 5 for blue.

BM_5CHANNEL = & h0205

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 40 bits per pixel. 8 bits apiece are used for each channel.

BM_6CHANNEL = & h0206

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 48 bits per pixel. 8 bits apiece are used for each channel.

BM_7CHANNEL = & h0207

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 56 bits per pixel. 8 bits apiece are used for each channel.

BM_8CHANNEL = & h0208

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 64 bits per pixel. 8 bits apiece are used for each channel.

BM_BGRTRIPLETS = 4

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 24 bits per pixel maximum. For three channel colors, such as red, green, and blue, the total size is 24 bits per pixel. For single channel colors, such as gray, the total size is 8 bits per pixel.

BM_CMYKQUADS = & h0020

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 32 bits per pixel. 8 bits are used for each color channel.

BM_G3CHTRIPLETS = & h0204

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 24 bits per pixel maximum. For three channel values, the total size is 24 bits per pixel. For single channel gray scale, the total size is 8 bits per pixel.

BM_GRAY = & h0209

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 32 bits per pixel. Only the 8 bit gray-scale value is used.

BM_KYMCQUADS = & h0305

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 32 bits per pixel. 8 bits are used for each color channel.

BM_LabTRIPLETS = & h0203

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 24 bits per pixel maximum. For three channel, L, a, and b values, the total size is 24 bits per pixel. For single channel gray scale, the total size is 8 bits per pixel.

BM_NAMED_INDEX = & h0405

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 32 bits per pixel. Named color indices. Index numbering begins at one.

BM_R10G10B10A2 = & h0701

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: Only in Windows Vista.

BM_ R10G10B10A2_ XR = & h0702

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: Only in Windows Vista.

BM_ R16G16B16A16_ FLOAT = & h0703

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: Only in Windows Vista.

BM_ RGBTRIPLETS = 2

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 24 bits per pixel maximum. For three channel colors, such as red, green, and blue, the total size is 24 bits per pixel. For single channel colors, such as gray, the total size is 8 bits per pixel.

BM_ S2DOT13FIXED_ scARGB = & h0604

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 64 bits per pixel. Color data is stored as one 16-bit word per channel, with a fixed range of -4 to +4, inclusive. A signed format is used, with 1 bit for the sign, 2 bits for the integer portion, and 13 bits for the fractional portion.

BM_ S2DOT13FIXED_ scRGB = & h0603

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 48 bits per pixel. Color data is stored as one 16-bit word per channel, with a fixed range of -4 to +4, inclusive. A signed format is used, with 1 bit for the sign, 2 bits for the integer portion, and 13 bits for the fractional portion.

BM_ x555G3CH = & h0104

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 16 bits per pixel. G3CH color space. 5 bits per channel. The most significant bit is ignored.

BM_ x555Lab = & h0103

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 16 bits per pixel. Lab color space. 5 bits per channel. The most significant bit is ignored.

BM_ x555RGB = 0

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 16 bits per pixel. RGB color space. 5 bits per channel. The most significant bit is ignored.

BM_ x555XYZ = & h0101

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 16 bits per pixel. Yxy color space. 5 bits per channel. The most significant bit is ignored.

BM_ x555Yxy = & h0102

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 16 bits per pixel. Yxy color space. 5 bits per channel. The most significant bit is ignored.

BM_ xBGRQUADS = & h0010

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 32 bits per pixel. 8 bits are used for each color channel. The most significant byte is ignored.

BM_ xG3CHQUADS = & h0304

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 32 bits per pixel. 8 bits are used for each color channel. The most significant byte is ignored.

BM_ xRGBQUADS = & h0008

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 32 bits per pixel. 8 bits are used for each color channel. The most significant byte is ignored.

BM_ XYZTRIPLETS = & h0201

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 24 bits per pixel maximum. For three channel colors, such as red, green, and blue, the total size is 24 bits per pixel. For single channel colors, such as gray, the total size is 8 bits per pixel.

BM_ YxyTRIPLETS = & h0202

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the bitmap format constants.

Notes: 24 bits per pixel maximum. For three channel, Y, x, and y values, the total size is 24 bits per pixel. For single channel gray scale, the total size is 8 bits per pixel.

CMM_DESCRIPTION = 5

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the info selectors for GetInfo.

Notes: A text string that describes the color management module.

CMM_DLL_VERSION = 3

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the values for the GetCMMInfo function.

Notes: Retrieves the version number of the CMM.

CMM_DRIVER_VERSION = 2

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the info selectors for GetInfo.

CMM_FROM_PROFILE = 0

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the info selectors for GetInfo.

CMM_IDENT = 1

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the values for the GetCMMInfo function.

Notes: Retrieves the CMM signature registered with the International Color Consortium (ICC).

CMM_LOGOICON = 6

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the info selectors for GetInfo.

Notes: The logo icon for this CMM.

CMM_ VERSION = 4

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the info selectors for GetInfo.

Notes: Version of Windows supported.

CMM_ WINDOWS_ DEFAULT = & h57696E20

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the info selectors for GetInfo.

CMM_ WIN_ VERSION = 0

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the values for the GetCMMInfo function.

Notes: Retrieves the version of Windows targeted by the color management module (CMM).

ENABLE_ GAMUT_ CHECKING = & h10000

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flags passed when creating a transformation.

Notes: Use this transform for gamut checking.

FAST_ TRANSLATE = & h40000

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flags passed when creating a transformation.

Notes: Look up color only. Do not interpolate the color.

INDEX_DONT_CARE = 0

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** A special value for index in constructor.

Notes: The application developer may allow Windows to choose the CMM by setting the indexPreferred-CMM parameter to INDEX_DONT_CARE for the constructor.

INTENT_ABSOLUTE_COLORIMETRIC = 3

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the rendering intent constants.

Notes: Maintain the white point. Match the colors to their nearest color in the destination gamut.

INTENT_PERCEPTUAL = 0

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the rendering intent constants.

Notes: Maintain contrast. Used for photographs and natural images.

INTENT_RELATIVE_COLORIMETRIC = 1

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the rendering intent constants.

Notes: Maintain colorimetric match. Used for graphic designs and named colors.

INTENT_SATURATION = 2

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the rendering intent constants.

Notes: Maintain saturation. Used for business charts and other situations in which undithered colors are required.

NORMAL_ MODE = 2

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flags passed when creating a transformation.

Notes: Transform will be used for normal image display. Average image quality.

PRESERVEBLACK = & h100000

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flags passed when creating a transformation.

Notes: If this bit is set, the transform engine inserts the appropriate black generation GMMP as the last GMMP in the transform sequence. This flag only works in a pure WCS transform.

PROOF_ MODE = 1

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flags passed when creating a transformation.

Notes: Transform will be used to preview the image. Low image quality.

SEQUENTIAL_ TRANSFORM = & h80800000

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flags passed when creating a transformation.

Notes:

If this bit is set, each step in the WCS processing pipeline is performed for every pixel in the image and no optimized color transform is built. This flag only works in a pure WCS transform.

Restrictions: A transform created with the SEQUENTIAL_ TRANSFORM flag set may only be used in the thread on which it was created and only for one color translation call at a time. COM must be initialized prior to creating the sequential transform and must remain initialized for the lifetime of the transform object.

USE_ RELATIVE_ COLORIMETRIC = & h20000

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flags passed when creating a transformation.

WCS_ ALWAYS = & h200000

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the flags passed when creating a transformation.

Notes: If this bit is set, even all-ICC transforms will use the WCS code path.

2.4 class WindowsICMColorMBS

class WindowsICMColorMBS

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The class for a color value.

Notes:

A variable of type color may be accessed as any of the supported color space colors by accessing the appropriate member of the union.

see also:

[http://msdn.microsoft.com/en-us/library/dd371932\(v=VS.85\).aspx](http://msdn.microsoft.com/en-us/library/dd371932(v=VS.85).aspx)

2.4.1 Methods

Channel(index as integer) as integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The channel value of a colorspace with 3 to 8 channels.

Notes:

Index from 0 to 7.

(Read and Write computed property)

2.4.2 Properties

a as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The a color value of a Lab color.

Notes: (Read and Write property)

b as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The b color value of a Lab color.

Notes: (Read and Write property)

black as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The black color value of a CMYK color.

Notes: (Read and Write property)

blue as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The blue value.

Notes:

Range from 0 to 255.

(Read and Write property)

ch1 as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The first channel color value of a three channel color.

Notes: (Read and Write property)

ch2 as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The second channel color value of a three channel color.

Notes: (Read and Write property)

ch3 as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The third channel color value of a three channel color.

Notes: (Read and Write property)

cyan as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The cyan color value of a CMYK color.

Notes: (Read and Write property)

gray as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The gray color value.

Notes:

For grayscale color space.

(Read and Write property)

green as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The green value.

Notes:

Range from 0 to 255.

(Read and Write property)

Index as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The index value for an indexed color space.

Notes: (Read and Write property)

L as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The L color value of a Lab color.

Notes: (Read and Write property)

magenta as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The magenta color value of a CMYK color.

Notes: (Read and Write property)

red as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The red value.
Notes:

Range from 0 to 255.

(Read and Write property)

XYZ_ X as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The X value of a XYZ color.

Notes: (Read and Write property)

XYZ_ Y as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The Y value of a XYZ color.

Notes: (Read and Write property)

XYZ_ Z as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The Z value of a XYZ color.

Notes: (Read and Write property)

yellow as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The yellow color value of a CMYK color.

Notes: (Read and Write property)

Yxy_ x as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The x value of a Yxy color.

Notes: (Read and Write property)

Yxy_ y as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The y value of a Yxy color.

Notes:

Yxy_ YY is the first Y and Yxy_ Y the second one in a Yxy color.
(Read and Write property)

Yxy_ YY as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The y value of a Yxy color.

Notes:

Yxy_ YY is the first Y and Yxy_ Y the second one in a Yxy color.
(Read and Write property)

2.4.3 Constants

COLOR_ 3_ CHANNEL = 6

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color type constants.

Notes: The COLOR is in the GENERIC3CHANNEL color space.

COLOR_ 5_ CHANNEL = 8

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color type constants.

Notes: The COLOR is in a five channel color space.

COLOR_ 6_ CHANNEL = 9

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color type constants.

Notes: The COLOR is in a six channel color space.

COLOR_ 7_ CHANNEL = 10

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color type constants.

Notes: The COLOR is in a seven channel color space.

COLOR_ 8_ CHANNEL = 11

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color type constants.

Notes: The COLOR is in an eight channel color space.

COLOR_ CMYK = 7

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color type constants.

Notes: The COLOR is in the CMYKCOLOR color space.

COLOR_ GRAY = 1

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color type constants.

Notes: The COLOR is in the GRAYCOLOR color space.

COLOR_ Lab = 5

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color type constants.

Notes: The COLOR is in the LabCOLOR color space.

COLOR_ NAMED = 12

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color type constants.

Notes: The COLOR is in a named color space.

COLOR_ RGB = 2

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color type constants.

Notes: The COLOR is in the RGBCOLOR color space.

COLOR_ XYZ = 3

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color type constants.

Notes: The COLOR is in the XYZCOLOR color space.

COLOR_ Yxy = 4

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color type constants.

Notes: The COLOR is in the YxyCOLOR color space.

MAX_ COLOR_ CHANNELS = 8

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** The maximum number of supported color channels.

2.5 class WindowsICMEnumMBS

class WindowsICMEnumMBS

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The WindowsICMEnumMBS class contains information that defines the profile enumeration constraints.

Example:

```
dim c as new WindowsICMEnumMBS // no options set
```

```

dim a(-1) as string = WindowsICModuleMBS.EnumColorProfiles(c)

for each s as string in a
  MsgBox s
next

```

Notes:

see also

[http://msdn.microsoft.com/en-us/library/dd316895\(v=VS.85\).aspx](http://msdn.microsoft.com/en-us/library/dd316895(v=VS.85).aspx)

2.5.1 Properties**Attributes0 as Integer**

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Attributes of profile that can be any of the following values.

Notes:

Constant	Meaning
ATTRIB_TRANSPARENCY	Turns transparency on. If this flag is not used, the attribute is reflective by default.
ATTRIB_MATTE	Turns matte display on. If this flag is not used, the attribute is glossy by default.

(Read and Write property)

Attributes1 as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Attributes of profile that can be any of the following values.

Notes:

Constant	Meaning
ATTRIB_TRANSPARENCY	Turns transparency on. If this flag is not used, the attribute is reflective by default.
ATTRIB_MATTE	Turns matte display on. If this flag is not used, the attribute is glossy by default.

(Read and Write property)

Class as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Indicates the profile class.

Notes:

For a description of profile classes, see Using Device Profiles with WCS.
[http://msdn.microsoft.com/en-us/library/dd372213\(v=VS.85\).aspx](http://msdn.microsoft.com/en-us/library/dd372213(v=VS.85).aspx)

A profile class may have any of the following values.

Profile Class	Signature
Input Device Profile	CLASS_SCANNER
Display Device Profile	CLASS_MONITOR
Output Device Profile	CLASS_PRINTER
Device Link Profile	CLASS_LINK
Color Space Conversion Profile	CLASS_COLORSPACE
Abstract Profile	CLASS_ABSTRACT
Named Color Profile	CLASS_NAMED
Color Appearance Model Profile	CLASS_CAMP
Color Gamut Map Model Profile	CLASS_GMMP

(Read and Write property)

CMMType as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The identification number of the CMM that is used in the profile.

Notes:

Identification numbers are registered with the ICC.

(Read and Write property)

ConnectionSpace as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** A signature value that indicates the color space in which the profile connection space (PCS) is defined.

Notes:

Can be any of the following values: SPACE_ XYZ or SPACE_ Lab

When the Class member is set to CLASS_ LINK, the PCS is taken from the DataColorSpace member.
(Read and Write property)

Creator as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Signature of the software that created the profile.

Notes:

Signatures are registered with the ICC.
(Read and Write property)

DataColorSpace as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** A signature value that indicates the color space in which the profile data is defined.

Notes:

Can be any value from the SPACE_ * Constants.
(Read and Write property)

DeviceClass as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Indicates the device class.

Notes:

A device class may have one of the following values.

Profile Class	Signature
Input Device Profile	CLASS_ SCANNER
Display Device Profile	CLASS_ MONITOR
Output Device Profile	CLASS_ PRINTER

(Read and Write property)

DeviceName as String

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** User friendly name of the device.

Notes: (Read and Write property)

DitheringMode as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Indicates the style of dithering that will be used when an image is displayed.

Notes: (Read and Write property)

Fields as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Indicates which fields in this class are being used.

Notes:

Can be set to any combination of the following constant values:

```

ET_ DEVICENAME
ET_ MEDIATYPE
ET_ DITHERMODE
ET_ RESOLUTION
ET_ CMMTYPE
ET_ CLASS
ET_ DATACOLORSPACE
ET_ CONNECTIONSPACE

```

ET_SIGNATURE
ET_PLATFORM
ET_PROFILEFLAGS
ET_MANUFACTURER
ET_MODEL
ET_ATTRIBUTES
ET_RENDERINGINTENT
ET_CREATOR
ET_DEVICECLASS
(Read and Write property)

Manufacturer as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The identification number of the device profile manufacturer.

Notes:

All manufacturer identification numbers are registered with the ICC.
(Read and Write property)

MediaType as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Indicates which type of media is associated with the profile, such as a printer or screen.

Notes: (Read and Write property)

Model as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The device manufacturer's device model number.

Notes:

All model identification numbers are registered with the ICC.
(Read and Write property)

Platform as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The primary platform for which the profile was created.

Notes:

The member can be set to any of the following values.

Platform	Value
Apple Computer, Inc.	'APPL'
Microsoft Corp.	'MSFT'
Silicon Graphics, Inc.	'SGI'
Sun Microsystems, Inc.	'SUNW'
Taligent	'TGNT'

(Read and Write property)

ProfileFlags as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Bit flags containing hints that the CMM uses to interpret the profile data and can be set to one of the following values.

Notes:

Constant	Meaning
FLAG_EMBEDDEDPROFILE	The profile is embedded in a bitmap file.
FLAG_DEPENDENTONDATA	The profile can't be used independently of the embedded color data. Used for profiles that are embedded in bitmap files.

(Read and Write property)

RenderingIntent as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The profile rendering intent.

Notes:

The member can be set to one of the following values:

INTENT_PERCEPTUAL INTENT_SATURATION INTENT_RELATIVE_COLORIMETRIC INTENT_

ABSOLUTE_ COLORIMETRIC

For more information, see Rendering Intents.

[http://msdn.microsoft.com/en-us/library/dd372183\(v=VS.85\).aspx](http://msdn.microsoft.com/en-us/library/dd372183(v=VS.85).aspx)
(Read and Write property)

ResolutionX as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The horizontal resolution in pixels of the device on which the image will be displayed.

Notes: (Read and Write property)

ResolutionY as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The vertical resolution in pixels of the device on which the image will be displayed.

Notes: (Read and Write property)

Signature as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Reserved for internal use.

Notes: (Read and Write property)

2.5.2 Constants**ATTRIB_ MATTE = 2**

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the attribute constants.

Notes: Turns matte display on. If this flag is not used, the attribute is glossy by default.

ATTRIB_ TRANSPARENCY = 1

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the attribute constants.

Notes: Turns transparency on. If this flag is not used, the attribute is reflective by default.

CLASS_ ABSTRACT = & h61627374

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

CLASS_ CAMP = & h6C616D70

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

CLASS_ COLORSPACE = & h73706163

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

CLASS_ GMMP = & h676D6D70

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

CLASS_ LINK = & h6C696E6B

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

CLASS_ MONITOR = & h6D6E7472

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

CLASS_ NAMED = & h6E6D636C

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

CLASS_ PRINTER = & h70727472

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

CLASS_ SCANNER = & h73636E72

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

ET_ ATTRIBUTES = & h02000

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

ET_ CLASS = & h00020

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

ET_CMMTYPE = & h00010

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

ET_CONNECTIONSPACE = & h00080

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

ET_CREATOR = & h08000

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

ET_DATACOLORSPACE = & h00040

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

ET_DEVICECLASS = & h10000

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

ET_DEVICENAME = & h00001

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

ET_DITHERMODE = & h00004

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

ET_MANUFACTURER = & h00800

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

ET_MEDIATYPE = & h00002

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

ET_MODEL = & h01000

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

ET_PLATFORM = & h00200

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

ET_PROFILEFLAGS = & h00400

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

ET_RENDERINGINTENT = & h04000

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

ET_RESOLUTION = & h00008

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

ET_SIGNATURE = & h00100

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the constants to specify which property in this class is used.

FLAG_DEPENDENTONDATA = 2

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile flag constants.

Notes: The profile can't be used independently of the embedded color data. Used for profiles that are embedded in bitmap files.

FLAG_EMBEDDEDPROFILE = 1

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile flag constants.

Notes: The profile is embedded in a bitmap file.

FLAG_ENABLE_CHROMATIC_ADAPTATION = & h02000000

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile flag constants.

SigMacintosh = & h4150504C

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the platform signatures.

Notes: APPL = Apple Computer, Inc.

SigMicrosoft = & h4D534654

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the platform signatures.

Notes: MSFT = Microsoft Corp.

SigSGI = & h53474920

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the platform signatures.

Notes: SGI = Silicon Graphics, Inc.

SigSolaris = & h53554E57

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the platform signatures.

Notes: SUNW = Sun Microsystems, Inc.

SigTaligent = & h54474E54

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the platform signatures.

Notes: TGNT = Taligent

SPACE_Lab = & h4C616220

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: Lab

SPACE_XYZ = & h58595A20

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: XYZ

2.6 class WindowsICMNamedProfileInfoMBS

class WindowsICMNamedProfileInfoMBS

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** This class is used to store information about a named color profile.

2.6.1 Properties

Count as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Total number of named colors in the profile.

Notes: (Read and Write property)

CountDevCoordinates as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Total number of device coordinates for each named color.

Notes: (Read and Write property)

Flags as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Flags for this information record.

Notes:

Not currently used by the default CMM.
(Read and Write property)

Prefix as String

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** A string containing the prefix for each color name.

Notes: (Read and Write property)

Suffix as String

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** A string containing the suffix for each color name.

Notes: (Read and Write property)

2.7 class WindowsICMPProfileHeaderMBS

class **WindowsICMPProfileHeaderMBS**

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** This class contains information that describes the contents of a device profile file.

Notes: This header occurs at the beginning of a device profile file.

2.7.1 Properties

Attributes0 as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Attributes of profile.
Notes: (Read and Write property)

Attributes1 as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Attributes of profile.
Notes: (Read and Write property)

Class as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Indicates the profile class.

Example:

```
// some profile file
dim file as FolderItem = SpecialFolder.Desktop.Child("test.icc")

// open profile read only
dim w as WindowsICMProfileMBS = WindowsICMProfileMBS.OpenProfileFile(file, WindowsICMProfileMBS.PRO-
FILE_READ, WindowsICMProfileMBS.FILE_SHARE_READ, WindowsICMProfileMBS.OPEN_EXIST-
ING)

// get headers
dim h as WindowsICMProfileHeaderMBS = w.ColorProfileHeader

// show color space name
Select case h.Class
case WindowsICMProfileHeaderMBS.CLASS_MONITOR
msgbox "Monitor"
case WindowsICMProfileHeaderMBS.CLASS_PRINTER
msgbox "Printer"
case WindowsICMProfileHeaderMBS.CLASS_SCANNER
msgbox "Scanner"
case WindowsICMProfileHeaderMBS.CLASS_LINK
msgbox "Link"
case WindowsICMProfileHeaderMBS.CLASS_ABSTRACT
```

```

msgbox "Abstract"
case WindowsICMPProfileHeaderMBS.CLASS_ COLORSPACE
msgbox "Colorspace"
case WindowsICMPProfileHeaderMBS.CLASS_ NAMED
msgbox "Named"
case WindowsICMPProfileHeaderMBS.CLASS_ CAMP
msgbox "Camp"
case WindowsICMPProfileHeaderMBS.CLASS_ GMMP
msgbox "GNMP"
else
msgbox "Unknown: "+hex(h.Class)
end Select

```

Notes:

For a description of profile classes, see Using Device Profiles with WCS:
[http://msdn.microsoft.com/en-us/library/dd372213\(v=VS.85\).aspx](http://msdn.microsoft.com/en-us/library/dd372213(v=VS.85).aspx)

A profile class may have any of the values from the CLASS_ * constants.

Class is written with three s here as Class is a reserved word in Real Studio.
 (Read and Write property)

CMMType as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The identification number of the CMM that is used in the profile.

Notes:

Identification numbers are registered with the ICC.
 (Read and Write property)

ConnectionSpace as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** A signature value that indicates the color space in which the profile connection space (PCS) is defined.

Notes:

The member can be any of the following values: SPACE_ XYZ or SPACE_ Lab.
(Read and Write property)

Creator as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Signature of the software that created the profile.

Example:

```
dim file as FolderItem = SpecialFolder.Desktop.Child("test.icc")
dim w as WindowsICMProfileMBS = WindowsICMProfileMBS.OpenProfileFile(file, WindowsICMProfileMBS.PRO-
FILE_ READ, WindowsICMProfileMBS.FILE_ SHARE_ READ, WindowsICMProfileMBS.OPEN_ EXIST-
ING)
dim h as WindowsICMProfileHeaderMBS = w.ColorProfileHeader

MsgBox hex(h.Creator)
```

Notes:

Signatures are registered with the ICC.
(Read and Write property)

DataColorSpace as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** A signature value that indicates the color space in which the profile data is defined.

Example:

```
function Name() as string
dim w as WindowsICMProfileMBS // your profile
dim h as WindowsICMProfileHeaderMBS = w.ColorProfileHeader

if h=nil then Return "?"

Select case h.DataColorSpace
case WindowsICMProfileHeaderMBS.SPACE_ XYZ
Return "XYZ"
case WindowsICMProfileHeaderMBS.SPACE_ Lab
Return "Lab"
```

```

case WindowsICMPProfileHeaderMBS.SPACE_ Luv
Return "Luv"
case WindowsICMPProfileHeaderMBS.SPACE_ YCbCr
Return "YCbCr"
case WindowsICMPProfileHeaderMBS.SPACE_ Yxy
Return "Yxy"
case WindowsICMPProfileHeaderMBS.SPACE_ RGB
Return "RGB"
case WindowsICMPProfileHeaderMBS.SPACE_ GRAY
Return "GRAY"
case WindowsICMPProfileHeaderMBS.SPACE_ HSV
Return "HSV"
case WindowsICMPProfileHeaderMBS.SPACE_ HLS
Return "HLS"
case WindowsICMPProfileHeaderMBS.SPACE_ CMYK
Return "CMYK"
case WindowsICMPProfileHeaderMBS.SPACE_ CMY
Return "CMY"
case WindowsICMPProfileHeaderMBS.SPACE_ 2_ CHANNEL
Return "2 Channel"
case WindowsICMPProfileHeaderMBS.SPACE_ 3_ CHANNEL
Return "2 Channel"
case WindowsICMPProfileHeaderMBS.SPACE_ 4_ CHANNEL
Return "2 Channel"
case WindowsICMPProfileHeaderMBS.SPACE_ 5_ CHANNEL
Return "2 Channel"
case WindowsICMPProfileHeaderMBS.SPACE_ 6_ CHANNEL
Return "2 Channel"
case WindowsICMPProfileHeaderMBS.SPACE_ 7_ CHANNEL
Return "2 Channel"
case WindowsICMPProfileHeaderMBS.SPACE_ 8_ CHANNEL
Return "2 Channel"
else
Return "Unknown: "+hex(h.DataColorSpace)
end Select
end function

```

Notes:

The member can be any of value from the SPACE_ * Constants.
(Read and Write property)

DateTime0 as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The data and time that the profile was created.

Notes: (Read and Write property)

DateTime1 as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The data and time that the profile was created.

Notes: (Read and Write property)

DateTime2 as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The data and time that the profile was created.

Notes: (Read and Write property)

IlluminantX as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** X value of the Profile illuminant.

Notes: (Read and Write property)

IlluminantY as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Y value of the Profile illuminant.

Notes: (Read and Write property)

IlluminantZ as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Z value of the Profile illuminant.

Notes: (Read and Write property)

Manufacturer as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The identification number of the device profile manufacturer.

Notes:

All manufacturer identification numbers are registered with the ICC.
(Read and Write property)

Model as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The device manufacturer's device model number.

Notes:

All model identification numbers are registered with the ICC.
(Read and Write property)

Platform as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The primary platform for which the profile was created.

Example:

```
dim file as FolderItem = SpecialFolder.Desktop.Child("test.icc")
dim w as WindowsICMProfileMBS = WindowsICMProfileMBS.OpenProfileFile(file, WindowsICMProfileMBS.PROFILE_READ, WindowsICMProfileMBS.FILE_SHARE_READ, WindowsICMProfileMBS.OPEN_EXISTING)
dim h as WindowsICMProfileHeaderMBS = w.ColorProfileHeader
```

MsgBox DecodingFromHexMBS(hex(h.Platform)) // shows platform, e.g. "APPL"

Notes:

The primary platform can be set to any of the following values.

Platform	Value
Apple Computer, Inc.	'APPL'
Microsoft Corp.	'MSFT'
Silicon Graphics, Inc.	'SGI'
Sun Microsystems, Inc.	'SUNW'
Taligent	'TGNT'

(Read and Write property)

ProfileFlags as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Bit flags containing hints that the CMM uses to interpret the profile data.

Notes:

The member can be set to the following values.

Constant	Meaning
FLAG_EMBEDDEDPROFILE	The profile is embedded in a bitmap file.
FLAG_DEPENDENTONDATA	The profile can't be used independently of the embedded color data. Used for profiles that are embedded in bitmap files.

(Read and Write property)

RenderingIntent as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The profile rendering intent.

Notes:

The member can be set to one of the following values:

INTENT_ PERCEPTUALINTENT_ SATURATIONINTENT_ RELATIVE_ COLORIMETRICINTENT_ ABSOLUTE_ COLORIMETRIC

For more information, see Rendering Intents.

[http://msdn.microsoft.com/en-us/library/dd372183\(v=VS.85\).aspx](http://msdn.microsoft.com/en-us/library/dd372183(v=VS.85).aspx)

(Read and Write property)

Signature as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Reserved for internal use.

Notes: (Read and Write property)

Version as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The version number of the profile.

Notes:

The version number is determined by the ICC. The current major version number is & h02. The current minor version number is & h10. The major and minor version numbers are in binary coded decimal (BCD). They must be stored in the following format.

Byte Number	Contents
0	Major version number in BCD.
1	Minor version number in the most significant nibble of this byte. Bug fix version number in the least significant nibble.
2	Reserved. Must be set to 0.
3	Reserved. Must be set to 0.

(Read and Write property)

2.7.2 Constants

ATTRIB_ MATTE = 2

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the attribute constants.

Notes: Turns matte display on. If this flag is not used, the attribute is glossy by default.

ATTRIB_ TRANSPARENCY = 1

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the attribute constants.

Notes: Turns transparency on. If this flag is not used, the attribute is reflective by default.

CLASS_ ABSTRACT = & h61627374

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

Notes: Abstract Profile

CLASS_ CAMP = & h6C616D70

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

Notes: Color Appearance Model Profile

CLASS_ COLORSPACE = & h73706163

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

Notes: Color Space Conversion Profile

CLASS_ GMMP = & h676D6D70

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

Notes: Color Gamut Map Model Profile

CLASS_ LINK = & h6C696E6B

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

Notes: Device Link Profile

CLASS_ MONITOR = & h6D6E7472

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

Notes: Display Device Profile

CLASS_ NAMED = & h6E6D636C

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

Notes: Named Color Profile

CLASS_ PRINTER = & h70727472

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

Notes: Output Device Profile

CLASS_ SCANNER = & h73636E72

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile class constants.

Notes: Input Device Profile

FLAG_ DEPENDENTONDATA = 2

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile flags.

Notes: The profile can't be used independently of the embedded color data. Used for profiles that are embedded in bitmap files.

FLAG_ EMBEDDEDPROFILE = 1

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile flags.

Notes: The profile is embedded in a bitmap file.

FLAG_ ENABLE_ CHROMATIC_ ADAPTATION = & h02000000

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile flags.

SPACE_ 2_ CHANNEL = & h32434C52

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: Generic 2 channel

SPACE_3_CHANNEL = & h33434C52

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: Generic 3 channel

SPACE_4_CHANNEL = & h34434C52

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: Generic 4 channel

SPACE_5_CHANNEL = & h35434C52

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: Generic 5 channel

SPACE_6_CHANNEL = & h36434C52

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: Generic 6 channel

SPACE_7_CHANNEL = & h37434C52

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: Generic 7 channel

SPACE_8_CHANNEL = & h38434C52

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: Generic 8 channel

SPACE_CMY = & h434D5920

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: CMY

SPACE_CMYK = & h434D594B

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: CMYK

SPACE_GRAY = & h47524159

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: Gray scale

SPACE_HLS = & h484C5320

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: HLS

SPACE_ HSV = & h48535620

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: HSV

SPACE_ Lab = & h4C616220

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: Lab

SPACE_ Luv = & h4C757620

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: Luv

SPACE_ RGB = & h52474220

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: RGB

SPACE_ XYZ = & h58595A20

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: XYZ

SPACE_ YCbCr = & h59436272

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: YCbCr

SPACE_ Yxy = & h59787920

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space constants.

Notes: Yxy

2.8 class WindowsICMLogColorSpaceMBS

class WindowsICMLogColorSpaceMBS

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** This class contains information that defines a logical color space.

Notes:

If the CStype member is set to LCS_ sRGB or LCS_ WINDOWS_ COLOR_ SPACE, the other members of this structure are ignored, and WCS uses the sRGB color space. The Endpoints, GammaRed, GammaGreen, and GammaBlue members are used to describe the logical color space. The Endpoints member is a CIEXYZTRIPLE that contains the x, y, and z values of the color space's RGB endpoint.

The required DWORD bit format for the GammaRed, GammaGreen, and GammaBlue is an 8.8 fixed point integer left-shifted by 8 bits. The plugin takes care about that detail.

Whenever the Filename member contains a file name and the CStype member is set to LCS_ CALIBRATED_ RGB, WCS ignores the other members of this class. It uses the color space in the file as the color space to which this LOGCOLORSPACE structure refers.

The relation between tri-stimulus values X,Y,Z and chromaticity values x,y,z is as follows:

$$x = X/(X+Y+Z)$$

$$y = Y/(X+Y+Z)$$

$$z = Z/(X+Y+Z)$$

If the CStype member is set to LCS_ sRGB or LCS_ WINDOWS_ COLOR_ SPACE, the other members of this structure are ignored, and ICM uses the sRGB color space. Applications should still initialize the rest of the structure since CreateProfileFromLogColorSpace ignores CStype member and uses Endpoints, GammaRed, GammaGreen, GammaBlue members to create a profile, which may not be initialized in case of LCS_ sRGB or LCS_ WINDOWS_ COLOR_ SPACE color spaces.

2.8.1 Properties

CStype as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Color space type.
Notes:

The member can be one of the following values.

Value	Meaning
LCS_ CALIBRATED_ RGB	Color values are calibrated RGB values. The values are translated using the endpoints specified by the lcsEndpoints member before being passed to the device.
LCS_ sRGB	Color values are values are sRGB values.
LCS_ WINDOWS_ COLOR_ SPACE	Color values are Windows default color space color values.

(Read and Write property)

EndpointsBX as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The X value of the Blue Endpoint.

Notes: (Read and Write property)

EndpointsBY as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The Y value of the Blue Endpoint.

Notes: (Read and Write property)

EndpointsBZ as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The Z value of the Blue Endpoint.

Notes: (Read and Write property)

EndpointsGX as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The X value of the Green Endpoint.

Notes: (Read and Write property)

EndpointsGY as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The Y value of the Green Endpoint.

Notes: (Read and Write property)

EndpointsGZ as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The Z value of the Green Endpoint.

Notes: (Read and Write property)

EndpointsRX as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The X value of the Red Endpoint.

Notes: (Read and Write property)

EndpointsRY as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The Y value of the Red Endpoint.

Notes: (Read and Write property)

EndpointsRZ as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The Z value of the Red Endpoint.

Notes: (Read and Write property)

Filename as String

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** A string that names a color profile file.

Notes:

This member is typically set to "", but may be used to set the color space to be exactly as specified by the color profile. This is useful for devices that input color values for a specific printer, or when using an installable image color matcher. If a color profile is specified, all other members of this class should be set to reasonable values, even if the values are not completely accurate.

(Read and Write property)

GammaBlue as Double

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Scale of the blue coordinate.

Notes: (Read and Write property)

GammaGreen as Double

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Scale of the green coordinate.

Notes: (Read and Write property)

GammaRed as Double

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** Scale of the red coordinate.

Notes: (Read and Write property)

Intent as Integer

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The gamut mapping method.

Notes:

This member can be one of the following values.

Value	Value	Intent	ICC Name	Meaning
LCS_GM_BUSINESS	1	Graphic	Saturation	Maintain saturation. Used for business charts and other situations in which undithered colors are required.
LCS_GM_GRAPHICS	2	Proof	Relative Colorimetric	Maintain colorimetric match. Used for graphic designs and named colors.
LCS_GM_IMAGES	4	Picture	Perceptual	Maintain contrast. Used for photographs and natural images.
LCS_GM_ABS_COLORIMETRIC	8	Match	Absolute Colorimetric	Maintain the white point. Match the colors to their nearest color in the destination gamut.

(Read and Write property)

2.8.2 Constants

INTENT_ABSOLUTE_COLORIMETRIC = 3

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the rendering intent constants.

Notes: Maintain the white point. Match the colors to their nearest color in the destination gamut.

INTENT_PERCEPTUAL = 0

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the rendering intent constants.

Notes: Maintain contrast. Used for photographs and natural images.

INTENT_RELATIVE_COLORIMETRIC = 1

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the rendering intent constants.

Notes: Maintain colorimetric match. Used for graphic designs and named colors.

INTENT_SATURATION = 2

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the rendering intent constants.

Notes: Maintain saturation. Used for business charts and other situations in which undithered colors are required.

LCS_CALIBRATED_RGB = 0

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space type constants.

Notes: Color values are calibrated RGB values. The values are translated using the endpoints specified by the Endpoints member before being passed to the device.

LCS_ sRGB = & h73524742

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space type constants.

Notes: Color values are values are sRGB values.

LCS_ WINDOWS_ COLOR_ SPACE = & h57696E20

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the color space type constants.

Notes: Color values are Windows default color space color values.

2.9 module WindowsICMModuleMBS

module WindowsICMModuleMBS

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The main module for Windows Image Color Matching functions.

2.9.1 Methods

AssociateColorProfileWithDevice(ProfileName as string, DeviceName as string) as boolean

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The AssociateColorProfileWithDevice function associates a specified color profile with a specified device.

Notes:

ProfileName: The file name of the profile to associate.

DeviceName: The name of the device to associate.

Returns true on success and false on failure.

The AssociateColorProfileWithDevice function will fail if the profile has not been installed on the computer

using the `InstallColorProfile` function.

Note that under Windows (Windows 95 or later), the PostScript device driver for printers assumes a CMYK color model. Therefore, all PostScript printers must use a CMYK color profile. Windows 2000 does not have this limitation.

If the specified device is a monitor, this function updates the default profile.

Several profiles are typically associated with printers, based on paper and ink types. There is no default. The GDI selects the best one from the associated profiles when your application creates a device context (DC).

Scanners also have no default profile. However, it is atypical to associate more than one profile with a scanner.

`AssociateColorProfileWithDevice` always adds the specified profile to the current user's per-user profile association list for the specified device. Before adding the profile to the list, `AssociateColorProfileWithDevice` determines whether the user has previously expressed the desire to use a per-user profile association list for the device. If so, then `AssociateColorProfileWithDevice` simply adds the specified profile to the existing per-user profile association list for the device. If not, then `AssociateColorProfileWithDevice` creates a new per-user profile association list for the device by copying the system-wide association list for that device. It then appends the specified profile to the per-user list. From that point on, the current user will be using a per-user profile association list for the specified device, as if `SetUsePerUserProfiles` had been called for Device with the `usePerUserProfiles` parameter set to `TRUE`.

`DisassociateColorProfileFromDevice(ProfileName as string, DeviceName as string)` as boolean

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The `DisassociateColorProfileFromDevice` function disassociates a specified color profile with a specified device on a specified computer.

Notes:

Returns true on success and false on failure.

`ProfileName`: The file name of the profile to disassociate.

`DeviceName`: The name of the device to disassociate.

If more than one profile is associated with a device, WCS uses the last one associated as the default. That is,

if your application sequentially associates three profiles with a device, WCS will use the last one associated as the default. If your application then calls the `DisassociateColorProfileFromDevice` function to disassociate the third profile (which is the default in this example), the WCS will use the second profile as the default.

If your application disassociates all profiles from a device, WCS uses the sRGB profile as the default.

`DisassociateColorProfileFromDevice` always removes the specified profile from the current user's per-user profile association list for the specified device. Before removing the profile from the list, `DisassociateColorProfileFromDevice` determines whether the user has previously expressed the desire to use a per-user profile association list for the device. If so, then `DisassociateColorProfileFromDevice` simply removes the specified profile from the existing per-user profile association list for the device. If not, then `DisassociateColorProfileFromDevice` creates a new per-user profile association list for the device by copying the system-wide association list for that device. It then removes the specified profile from the per-user list. From that point on, the current user will be using a per-user profile association list for the specified device, as if `SetUsePerUserProfiles` had been called for Device with the `usePerUserProfiles` parameter set to `TRUE`.

EnumColorProfiles(criteria as WindowsICMEnumMBS) as string()

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The `EnumColorProfiles` function enumerates all the profiles satisfying the given enumeration criteria.

Example:

```
dim c as new WindowsICMEnumMBS // no options set

dim a(-1) as string = WindowsICMModuleMBS.EnumColorProfiles(c)

for each s as string in a
  MsgBox s
next
```

Notes:

On success the function returns an array of profile names.

Several profiles are typically associated with printers, based on the paper and ink types. There is a default profile for each device. For International Color Consortium (ICC) profiles, GDI selects the best one from the ICC-associated profiles when your application creates a device context (DC).

Do not attempt to use `EnumColorProfiles` to determine the default profile for a device. Instead, create a device context for the device and then invoke the `GetICMPProfile` function. On Windows Vista and Windows 7, the `WcsGetDefaultColorProfile` function can also be used to determine a device's default color profile.

If the `Fields` member of `WindowsICMEnumMBS` that is pointed to by the `criteria` parameter is set to `ET_DEVICENAME`, this function will enumerate all of the color profiles associated with all types of devices attached to the user's computer, regardless of the device class. If the `Fields` member is set to `ET_DEVICENAME` or `ET_DEVICECLASS` and a device class is specified in the `DeviceClass` member, this function will only enumerate the profiles associated with the specified device class. If the `Fields` member is set only to `ET_DEVICECLASS`, the `EnumColorProfiles` function will enumerate all profiles that can be associated with that type of device.

Whenever `EnumColorProfiles` is examining the profiles associated with a specific device, the results depend on whether the user has chosen to use the system-wide list of profiles associated with that device, or his or her own ("per-user") list. Calling `SetUsePerUserProfiles` with its `usePerUserProfiles` parameter set to `TRUE` causes future calls to `EnumColorProfiles` to look at only the current user's per-user list of profile associations for the specified device. Calling `WcsSetUsePerUserProfiles` with its `usePerUserProfiles` parameter set to `FALSE` causes future calls to `EnumColorProfiles` to look at the system-wide list of profile associations for the specified device. If `SetUsePerUserProfiles` has never been called for the current user, `EnumColorProfiles` examines the system-wide list.

This function will provide the information for converting WCS-specific DMP information to the legacy `EnumType` record in `enable` consistent profile enumeration. The defaults will be the same as ICC if this information is not present.

Per-user/LUA support

The enumeration is specific to current user. Both system wide and current user device associations are considered. For default profile configuration, current user settings override system wide ones.

GetColorDirectory as folderitem

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The `GetColorDirectory` function retrieves the path of the Windows COLOR directory on a specified machine.

Example:

```
dim f as FolderItem = WindowsICMModuleMBS.GetColorDirectory
```

```
if f=nil then
```

```
MsgBox "No path?"  
else  
MsgBox f.AbsolutePath  
end if
```

Notes: On success returns folderitem for color directory. Returns nil on any error.

GetStandardColorSpaceProfile(ProfileID as integer) as string

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The GetStandardColorSpaceProfile function retrieves the color profile registered for the specified standard color space.

Example:

```
MsgBox WindowsICModuleMBS.GetStandardColorSpaceProfile(WindowsICModuleMBS.LCS_ WINDOWS_ COLOR_ SPACE)
```

Notes:

ProfileID: Specifies the ID value of the standard color space for which to retrieve the profile. The only valid values for this parameter are LCS_ sRGB and LCS_ WINDOWS_ COLOR_ SPACE.

This function supports Windows Color System (WCS) device model profiles (DMPs) in addition to International Color Consortium (ICC) profiles. It does not support WCS CAMP or GMMP profiles and will return an error if such profiles are used.

Overview of Windows Vista Specific Functionality

This will support WCS DMPs in addition to ICC profiles. It will not support WCS CAMP or GMMP profiles and will return an error if such profiles are used with this API.

Per-user/LUA support

This will retrieve the color profile registered for the given standard color space for current user. If there is no such setting for the current user, it retrieves the system wide setting.

This uses `WcsGetDefaultColorProfile` with `WCS_PROFILE_MANAGEMENT_SCOPE_CURRENT_USER`.

This is executable in LUA context.

InstallColorProfile(file as folderitem) as boolean

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The `InstallColorProfile` function installs a given profile for use on a specified machine.

Notes:

The profile is also copied to the `COLOR` directory.
Returns true on success and false on failure.

RegisterCMM(cmmID as integer, file as folderitem) as boolean

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** `RegisterCMM` associates a specified identification value with the specified color management module dynamic link library (CMM DLL).

Notes:

When this ID appears in a color profile, Windows can then locate the corresponding CMM so as to create a transform.

`cmmID`: Specifies the ID signature of the CMM registered with the International Color Consortium (ICC).

`file`: Points to the CMM DLL.

Returns true on success and false on failure.

SelectCMM(cmmID as integer) as boolean

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** SelectCMM allows an application to select the preferred color management module (CMM) to use.

Notes:

cmmID: Specifies the signature of the desired CMM as registered with the International Color Consortium (ICC).

Windows 2000 only: Setting this parameter to 0 causes the WCS system to select the default CMM.

Returns true on success and false on failure.

SetStandardColorSpaceProfile(ProfileID as integer, ProfileName as folderitem) as boolean

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The SetStandardColorSpaceProfile function registers a specified profile for a given standard color space.

Notes:

The profile can be queried using GetStandardColorSpaceProfile.

ProfileID: Specifies the ID value of the standard color space that the given profile represents.

ProfileName: path to the profile file.

Returns true on success and false on failure.

The profile must already be installed on the system before it can be registered for a standard color space.

This function supports Windows Color System (WCS) device model profiles (DMPs) in addition to International Color Consortium (ICC) profiles. It does not support WCS CAMP or GMMP profiles and will return an error if such profiles are used.

Per-user/LUA support

This will register a specified profile for a given standard color space only for current user.

This uses `SetDefaultColorProfile` with `WCS_PROFILE_MANAGEMENT_SCOPE_CURRENT_USER`.

This is executable in LUA context if the profile is already installed, fails otherwise with access denied since install is system-wide and requires administrator privileges.

UninstallColorProfile(ProfileName as string, DeleteFile as boolean = true) as boolean

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** `UninstallColorProfile` removes a specified color profile from a specified computer. Associated files are optionally deleted from the system.

Notes:

`ProfileName`: Points to the file name of the profile to uninstall.

`DeleteFile`: If set to true, the function deletes the profile from the COLOR directory. If set to false, this function has no effect.

Returns true on success and false on failure.

UnregisterCMM(cmmID as integer) as boolean

Plugin Version: 11.1 Console & Web: Yes Mac: No, Win: Yes, Linux: No, . **Function:** The `UnregisterCMM` function dissociates a specified ID value from a given color management module dynamic-link library (CMM DLL).

Notes:

`cmmID`: Specifies the ID value identifying the CMM whose registration is to be removed. This is the signature of the CMM registered with the International Color Consortium (ICC).

Returns true on success and false on failure.

2.9.2 Constants

CMM_DESCRIPTION = 5

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the info selectors for GetInfo.

Notes: A text string that describes the color management module.

CMM_DLL_VERSION = 3

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the info selectors for GetInfo.

Notes: Version number of the CMM DLL.

CMM_DRIVER_VERSION = 2

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the info selectors for GetInfo.

CMM_IDENT = 1

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the info selectors for GetInfo.

Notes: The CMM identification signature registered with the International Color Consortium (ICC).

CMM_LOGOICON = 6

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the info selectors for GetInfo.

Notes: The logo icon for this CMM.

CMM_ VERSION = 4

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the info selectors for GetInfo.

Notes: Version of Windows supported.

CMM_ WIN_ VERSION = 0

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the info selectors for GetInfo.

Notes: Backward compatibility with Windows 95.

LCS_ sRGB = & h73524742

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile IDs for GetStandardColorSpaceProfile.

Example:

```
MsgBox WindowsICMModuleMBS.GetStandardColorSpaceProfile(WindowsICMModuleMBS.LCS_ sRGB)
```

LCS_ WINDOWS_ COLOR_ SPACE = & h57696E20

Plugin Version: 11.1 Console & Web: No Mac: No, Win: Yes, Linux: No, . **Function:** One of the profile IDs for GetStandardColorSpaceProfile.

Example:

```
MsgBox WindowsICMModuleMBS.GetStandardColorSpaceProfile(WindowsICMModuleMBS.LCS_ WINDOWS_ COLOR_ SPACE)
```


Chapter 3

List of all classes

• WindowsICMColorMBS	55
• WindowsICMEnumMBS	62
• WindowsICMLogColorSpaceMBS	92
• WindowsICMNamedProfileInfoMBS	76
• WindowsICMProfileHeaderMBS	77
• WindowsICMProfileMBS	21
• WindowsICMSetupMBS	13
• WindowsICMTransformMBS	33

Chapter 4

List of all modules

- WindowsICModuleMBS

98