

kolrabi's another Image Library  
1.8.3

Generated by Doxygen 1.8.4

Mon Jun 2 2014 11:40:45



# Contents



# Chapter 1

## Module Index

### 1.1 Modules

Here is a list of all modules:

Global State . . . . .	??
Initialization / Deinitialization . . . . .	??
Image Management . . . . .	??
Image Manipulation . . . . .	??



## Chapter 2

# Hierarchical Index

### 2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

_iff_chunk . . . . .	??
ALPHA_CHUNK . . . . .	??
ALPHAINFO_CHUNK . . . . .	??
BITFILE . . . . .	??
BLOCKHEAD . . . . .	??
BLP1HEAD . . . . .	??
BLP2HEAD . . . . .	??
BMHD . . . . .	??
BMPHEAD . . . . .	??
Box . . . . .	??
BUCKET . . . . .	??
CHANNEL . . . . .	??
CHANNEL_CHUNK . . . . .	??
CLIST . . . . .	??
COL_CUBE . . . . .	??
Color565 . . . . .	??
Color888 . . . . .	??
Color8888 . . . . .	??
CONTRIB . . . . .	??
CUT_HEAD . . . . .	??
DCXHEAD . . . . .	??
DDS_CONTEXT . . . . .	??
DDSHEAD . . . . .	??
DICOMHEAD . . . . .	??
DOOM_HEAD . . . . .	??
DPX_FILE_INFO . . . . .	??
DPX_IMAGE_ELEMENT . . . . .	??
DPX_IMAGE_INFO . . . . .	??
DPX_IMAGE_ORIENT . . . . .	??
DPX_MOTION_PICTURE_HEAD . . . . .	??
DPX_TELEVISION_HEAD . . . . .	??
DXTAlphaBlock3BitLinear . . . . .	??
DXTAlphaBlockExplicit . . . . .	??
DXTColBlock . . . . .	??
Edge . . . . .	??
EXRHEAD . . . . .	??
FITSHEAD . . . . .	??
FORM_HEAD . . . . .	??

FTX_HEAD . . . . .	??
GENATT_CHUNK . . . . .	??
GifGraphicControlExtension . . . . .	??
GifImageDescriptor . . . . .	??
GifLoadingContext . . . . .	??
GifLogicalScreenDescriptor . . . . .	??
GifSignature . . . . .	??
HALOHEAD . . . . .	??
ICNSDATA . . . . .	??
ICNSHEAD . . . . .	??
ICODIR . . . . .	??
ICODIRENTRY . . . . .	??
ICOIMAGE . . . . .	??
IconData . . . . .	??
ICONDIR . . . . .	??
ICONDIRENTRY . . . . .	??
iff_chunk_stack . . . . .	??
iFormatL . . . . .	??
iFormatS . . . . .	??
iFree . . . . .	??
IL_HINTS . . . . .	??
IL_STATES . . . . .	??
ilError . . . . .	??
ilFilters . . . . .	??
ILformat . . . . .	??
ILformatEntry . . . . .	??
ilImage . . . . .	??
ILimage . . . . .	??
ILpal . . . . .	??
ilState . . . . .	??
ILUinfo . . . . .	??
ILUpointf . . . . .	??
ILUpointi . . . . .	??
ILUT_STATES . . . . .	??
ilValidate . . . . .	??
INFOHEAD . . . . .	??
iread_mgr . . . . .	??
iSgiHeader . . . . .	??
IStream	
ilIStream . . . . .	??
IWIHEAD . . . . .	??
iwrite_mgr . . . . .	??
LAYERBITMAP_CHUNK . . . . .	??
LAYERINFO_CHUNK . . . . .	??
LIF_HEAD . . . . .	??
LZWInputStream . . . . .	??
MDL_HEAD . . . . .	??
MP3HEAD . . . . .	??
NeuQuantContext . . . . .	??
OS2_HEAD . . . . .	??
OStream	
ilOStream . . . . .	??
PCXHEAD . . . . .	??
PIC_HEAD . . . . .	??
PIXHEAD . . . . .	??
PNGData . . . . .	??
PPMINFO . . . . .	??
PSDHEAD . . . . .	??



PSP_CTX . . . . .	??
PSPHEAD . . . . .	??
PSPRECT . . . . .	??
R32 . . . . .	??
RAW_HEAD . . . . .	??
rgbe_header_info . . . . .	??
ROT_HEAD . . . . .	??
SIO . . . . .	??
SUNHEAD . . . . .	??
TARGAEXT . . . . .	??
TARGAFOOTER . . . . .	??
TARGAHEAD . . . . .	??
TEX_HEAD . . . . .	??
TEX_INFO . . . . .	??
TPLHEAD . . . . .	??
UTXENTRYNAME . . . . .	??
UTXEXPORTTABLE . . . . .	??
UTXHEADER . . . . .	??
UTXIMPORTTABLE . . . . .	??
UTXPALETTE . . . . .	??
VTFHEAD . . . . .	??
WALHEAD . . . . .	??
WDPDCQUANT . . . . .	??
WDPGUID . . . . .	??
WDPHEAD . . . . .	??
WDPIFD . . . . .	??
WDPIMGHEAD . . . . .	??
WDPIMGPLANE . . . . .	??
WDPTILE . . . . .	??
XPMHASHENTRY . . . . .	??



## Chapter 3

# Data Structure Index

### 3.1 Data Structures

Here are the data structures with brief descriptions:

<a href="#">_iff_chunk</a>	??
<a href="#">ALPHA_CHUNK</a>	??
<a href="#">ALPHAINFO_CHUNK</a>	??
<a href="#">BITFILE</a>	??
<a href="#">BLOCKHEAD</a>	??
<a href="#">BLP1HEAD</a>	??
<a href="#">BLP2HEAD</a>	??
<a href="#">BMHD</a>	??
<a href="#">BMPHEAD</a>	??
<a href="#">Box</a>	??
<a href="#">BUCKET</a>	??
<a href="#">CHANNEL</a>	??
<a href="#">CHANNEL_CHUNK</a>	??
<a href="#">CLIST</a>	??
<a href="#">COL_CUBE</a>	??
<a href="#">Color565</a>	??
<a href="#">Color888</a>	??
<a href="#">Color8888</a>	??
<a href="#">CONTRIB</a>	??
<a href="#">CUT_HEAD</a>	??
<a href="#">DCXHEAD</a>	??
<a href="#">DDS_CONTEXT</a>	??
<a href="#">DDSHEAD</a>	??
<a href="#">DICOMHEAD</a>	??
<a href="#">DOOM_HEAD</a>	??
<a href="#">DPX_FILE_INFO</a>	??
<a href="#">DPX_IMAGE_ELEMENT</a>	??
<a href="#">DPX_IMAGE_INFO</a>	??
<a href="#">DPX_IMAGE_ORIENT</a>	??
<a href="#">DPX_MOTION_PICTURE_HEAD</a>	??
<a href="#">DPX_TELEVISION_HEAD</a>	??
<a href="#">DXTAlphaBlock3BitLinear</a>	??
<a href="#">DXTAlphaBlockExplicit</a>	??
<a href="#">DXTColBlock</a>	??
<a href="#">Edge</a>	??
<a href="#">EXRHEAD</a>	??
<a href="#">FITSHEAD</a>	??
<a href="#">FORM_HEAD</a>	??

FTX_HEAD	??
GENATT_CHUNK	??
GifGraphicControlExtension	??
GifImageDescriptor	??
GifLoadingContext	??
GifLogicalScreenDescriptor	??
GifSignature	??
HALOHEAD	??
ICNSDATA	??
ICNSHEAD	??
ICODIR	??
ICODIRENTRY	??
ICOIMAGE	??
IconData	??
ICONDIR	??
ICONDIRENTRY	??
iff_chunk_stack	??
iFormatL	??
iFormatS	??
iFree	??
IL_HINTS	??
IL_STATES	??
ilError	??
ilFilters	??
ILformat	??
ILformatEntry	??
ilImage	??
ILimage	??
The Fundamental Image structure	??
ilIStream	??
ilOStream	??
ILpal	??
Basic Palette struct	??
ilState	??
ILUinfo	??
ILUpointf	??
ILUpointi	??
ILUT_STATES	??
ilValidate	??
INFOHEAD	??
iread_mgr	??
iSgiHeader	??
IWIHEAD	??
iwrite_mgr	??
LAYERBITMAP_CHUNK	??
LAYERINFO_CHUNK	??
LIF_HEAD	??
LZWInputStream	??
MDL_HEAD	??
MP3HEAD	??
NeuQuantContext	??
OS2_HEAD	??
PCXHEAD	??
PIC_HEAD	??
PIXHEAD	??
PNGData	??
PPMINFO	??
PSDHEAD	??

PSP_CTX	??
PSPHEAD	??
PSPRECT	??
R32	??
RAW_HEAD	??
rgbe_header_info	??
ROT_HEAD	??
SIO	??
SUNHEAD	??
TARGAEXT	??
TARGAFOOTER	??
TARGAHEAD	??
TEX_HEAD	??
TEX_INFO	??
TPLHEAD	??
UTXENTRYNAME	??
UTXEXPORTTABLE	??
UTXHEADER	??
UTXIMPORTTABLE	??
UTXPALETTE	??
VTFHEAD	??
WALHEAD	??
WDPDCQUANT	??
WDPGUID	??
WDPHEAD	??
WDPIFD	??
WDPIMGHEAD	??
WDPIMGPLANE	??
WDPTILE	??
XPMHASHENTRY	??



## Chapter 4

# File Index

### 4.1 File List

Here is a list of all files with brief descriptions:

include/IL/devil_cpp_wrapper.hpp	??
include/IL/devil_internal_exports.h	??
include/IL/il.h	??
include/IL/ilu.h	??
include/IL/ilu_region.h	??
include/IL/ilut.h	??
src/IL/il_alloc.c	??
src/IL/il_alloc.h	??
src/IL/il_api.c	
Contains public IL entry functions	??
src/IL/il_bits.c	??
src/IL/il_bits.h	??
src/IL/il_endian.c	??
src/IL/il_endian.h	??
src/IL/il_error.c	??
src/IL/il_files.c	??
src/IL/il_files.h	??
src/IL/il_formats.c	??
src/IL/il_formats.h	??
src/IL/il_internal.c	??
src/IL/il_internal.h	??
src/IL/il_io.c	??
src/IL/il_kail.c	??
src/IL/il_main.c	??
src/IL/il_manip.c	??
src/IL/il_manip.h	??
src/IL/il_pal.c	??
src/IL/il_pal.h	??
src/IL/il_profiles.c	??
src/IL/il_register.c	??
src/IL/il_register.h	??
src/IL/il_size.c	??
src/IL/il_skia.cc	??
src/IL/il_stack.c	??
src/IL/il_stack.h	??
src/IL/il_states.c	??
src/IL/il_states.h	??
src/IL/il_string.c	??

src/IL/il_string.h	??
src/IL/il_utility.c	??
src/IL/pack_pop.h	??
src/IL/pack_push.h	??
src/IL/algo/il_neuquant.c	??
src/IL/algo/il_nvidia.cc	??
src/IL/algo/il_quantizer.c	??
src/IL/algo/il_rle.c	??
src/IL/algo/il_rle.h	??
src/IL/algo/il_squish.c	??
src/IL/altivec/common.c	??
src/IL/altivec/common.h	??
src/IL/altivec/typeconversion.c	??
src/IL/altivec/typeconversion.h	??
src/IL/conv/il_color.h	??
src/IL/conv/il_convbuff.c	??
src/IL/conv/il_convert.c	??
src/IL/conv/il_fastconv.c	??
src/IL/formats/il_blp.c	??
src/IL/formats/il_bmp.c	??
src/IL/formats/il_bmp.h	??
src/IL/formats/il_cut.c	??
src/IL/formats/il_dcx.c	??
src/IL/formats/il_dcx.h	??
src/IL/formats/il_dds-save.c	??
src/IL/formats/il_dds.c	??
src/IL/formats/il_dds.h	??
src/IL/formats/il_dicom.c	??
src/IL/formats/il_doom.c	??
src/IL/formats/il_doompal.h	??
src/IL/formats/il_dpx.c	??
src/IL/formats/il_dpx.h	??
src/IL/formats/il_exr.c	??
src/IL/formats/il_exr.h	??
src/IL/formats/il_fits.c	??
src/IL/formats/il_ftx.c	??
src/IL/formats/il_gif.c	??
src/IL/formats/il_gif.h	??
src/IL/formats/il_hdr.c	??
src/IL/formats/il_header.c	??
src/IL/formats/il_icns.c	??
src/IL/formats/il_icns.h	??
src/IL/formats/il_icon.c	??
src/IL/formats/il_icon.h	??
src/IL/formats/il_iff.c	??
src/IL/formats/il_ilbm.c	??
src/IL/formats/il_iwi.c	??
src/IL/formats/il_jp2.c	??
src/IL/formats/il_jp2.h	??
src/IL/formats/il_jpeg.c	??
src/IL/formats/il_jpeg.h	??
src/IL/formats/il_lif.c	??
src/IL/formats/il_lif.h	??
src/IL/formats/il_md1.c	??
src/IL/formats/il_md1.h	??
src/IL/formats/il_mng.c	??
src/IL/formats/il_mp3.c	??
src/IL/formats/il_pal_act.c	??



src/IL/formats/il_pal_col.c	??
src/IL/formats/il_pal_halo.c	??
src/IL/formats/il_pal_jasc.c	??
src/IL/formats/il_pal_plt.c	??
src/IL/formats/il_pcd.c	??
src/IL/formats/il_pcx.c	??
src/IL/formats/il_pcx.h	??
src/IL/formats/il_pic.c	??
src/IL/formats/il_pic.h	??
src/IL/formats/il_pix.c	??
src/IL/formats/il_png.c	??
src/IL/formats/il_pnm.c	??
src/IL/formats/il_pnm.h	??
src/IL/formats/il_psd.c	??
src/IL/formats/il_psd.h	??
src/IL/formats/il_psp.c	??
src/IL/formats/il_psp.h	??
src/IL/formats/il_pxr.c	??
src/IL/formats/il_q2pal.h	??
src/IL/formats/il_raw.c	??
src/IL/formats/il_rawdata.c	??
src/IL/formats/il_rot.c	??
src/IL/formats/il_sgi.c	??
src/IL/formats/il_sgi.h	??
src/IL/formats/il_sun.c	??
src/IL/formats/il_targa.c	??
src/IL/formats/il_targa.h	??
src/IL/formats/il_texture.c	??
src/IL/formats/il_tiff.c	??
src/IL/formats/il_tpl.c	??
src/IL/formats/il_utx.c	??
src/IL/formats/il_utx.h	??
src/IL/formats/il_vtf.c	??
src/IL/formats/il_vtf.h	??
src/IL/formats/il_wal.c	??
src/IL/formats/il_wbmp.c	??
src/IL/formats/il_wdp.c	??
src/IL/formats/il_wdp.h	??
src/IL/formats/il_xpm.c	??
src/ILU/ilu_alloc.c	??
src/ILU/ilu_alloc.h	??
src/ILU/ilu_error.c	??
src/ILU/ilu_filter.c	??
src/ILU/ilu_filter.h	??
src/ILU/ilu_filter_rcg.c	??
src/ILU/ilu_internal.c	??
src/ILU/ilu_internal.h	??
src/ILU/ilu_main.c	??
src/ILU/ilu_manip.c	??
src/ILU/ilu_mipmap.c	??
src/ILU/ilu_noise.c	??
src/ILU/ilu_region.c	??
src/ILU/ilu_region.h	??
src/ILU/ilu_rotate.c	??
src/ILU/ilu_scale.c	??
src/ILU/ilu_scale2d.c	??
src/ILU/ilu_scale3d.c	??
src/ILU/ilu_scaling.c	??

src/ILU/ilu_states.c	??
src/ILU/ilu_states.h	??
src/ILU/ilu_utilities.c	??
src/ILU/ilu_error/ilu_err-arabic.h	??
src/ILU/ilu_error/ilu_err-dutch.h	??
src/ILU/ilu_error/ilu_err-english.h	??
src/ILU/ilu_error/ilu_err-french.h	??
src/ILU/ilu_error/ilu_err-german.h	??
src/ILU/ilu_error/ilu_err-japanese.h	??
src/ILU/ilu_error/ilu_err-spanish.h	??
src/ILUT/ilut_allegro.cc	??
src/ILUT/ilut_allegro.h	??
src/ILUT/ilut_directx.c	??
src/ILUT/ilut_directx9.c	??
src/ILUT/ilut_internal.c	??
src/ILUT/ilut_internal.h	??
src/ILUT/ilut_main.c	??
src/ILUT/ilut_opengl.c	??
src/ILUT/ilut_opengl.h	??
src/ILUT/ilut_sdlsurface.c	??
src/ILUT/ilut_states.c	??
src/ILUT/ilut_states.h	??
src/ILUT/ilut_win32.c	??
src/ILUT/ilut_x11.c	??
src/test/iltest-format-load.c	??
src/test/iltest-io.c	??
src/test/iltest-memory.c	??

## Chapter 5

# Module Documentation

### 5.1 Global State

#### Functions

- `void ILAPIENTRY ilBindImage (ILuint Image)`  
*Makes Image the current active image - similar to `glBindTexture()`.*
- `ILboolean ILAPIENTRY ilDisable (ILenum Mode)`  
*Disables a mode.*
- `ILboolean ILAPIENTRY ilEnable (ILenum Mode)`  
*Enables a mode.*
- `ILboolean ILAPIENTRY ilFormatFunc (ILenum Mode)`  
*Set the default image format to use.*
- `ILboolean ILAPIENTRY ilGetBoolean (ILenum Mode)`  
*Returns the current value of the Mode.*
- `void ILAPIENTRY ilGetBooleanv (ILenum Mode, ILboolean *Param)`  
*Sets Param equal to the current value of the Mode.*
- `ILenum ILAPIENTRY ilGetError (void)`  
*Gets the last error on the error stack.*
- `ILint ILAPIENTRY ilGetInteger (ILenum Mode)`  
*Returns the current value of the Mode.*
- `void ILAPIENTRY ilGetIntegerv (ILenum Mode, ILint *Param)`  
*Sets Param equal to the current value of the Mode.*
- `ILconst_string ILAPIENTRY ilGetString (ILenum StringName)`  
*Returns a constant string detailing aspects about this library.*
- `void ILAPIENTRY ilHint (ILenum Target, ILenum Mode)`  
*Specifies implementation-dependent performance hints.*
- `ILboolean ILAPIENTRY ilIsDisabled (ILenum Mode)`  
*Checks whether a Mode is not enabled.*
- `ILboolean ILAPIENTRY ilIsEnabled (ILenum Mode)`  
*Checks whether a Mode is enabled.*
- `void ILAPIENTRY ilSetInteger (ILenum Mode, ILint Param)`  
*Sets a parameter value for a Mode.*
- `void ILAPIENTRY ilSetString (ILenum StringName, const char *String)`  
*Sets a string detailing aspects about this library.*

### 5.1.1 Detailed Description

### 5.1.2 Function Documentation

#### 5.1.2.1 void ILAPIENTRY ilBindImage ( ILuint Image )

Makes Image the current active image - similar to glBindTexture().

This automatically resets the state to the first sub image, face (if applicable) and top level mipmap.

Parameters

<i>Image</i>	Name of image to bind.
--------------	------------------------

#### 5.1.2.2 ILboolean ILAPIENTRY ilDisable ( ILenum Mode )

Disables a mode.

Parameters

<i>Mode</i>	Mode to disable
-------------	-----------------

Returns

IL\_TRUE if successful.

See Also

[ilEnable](#) for a list of valid modes.

[ilIsEnabled](#)

#### 5.1.2.3 ILboolean ILAPIENTRY ilEnable ( ILenum Mode )

Enables a mode.

Valid modes are:

Mode	Default	Description
<i>IL_ORIGIN_SET</i>	<i>IL_FALSE</i>	Flip image on load to match <i>IL_ORIGIN_MODE</i> .
<i>IL_FORMAT_SET</i>	<i>IL_FALSE</i>	Convert image format on load to match the format set by <a href="#">ilFormatFunc()</a> .
<i>IL_TYPE_SET</i>	<i>IL_FALSE</i>	Convert image data type on load to match <i>IL_TYPE_MODE</i> .
<i>IL_CONV_PAL</i>	<i>IL_FALSE</i>	Convert images that use palettes on load to 24 bit RGBA.
<i>IL_NVIDIA_COMPRESS</i>	<i>IL_FALSE</i>	Use NVIDIA texture tools for compressing DXT formats if available.
<i>IL_SQUISH_COMPRESS</i>	<i>IL_FALSE</i>	Use libsquish for compressing DXT formats if available.

Parameters

<i>Mode</i>	Mode to enable
-------------	----------------

**Returns**

IL\_TRUE if successful.

**See Also**

[iIsEnabled](#)

**5.1.2.4 ILboolean ILAPIENTRY ilFormatFunc ( IEnum Mode )**

Set the default image format to use.

Default value IL\_BGRA. The current value can be retrieved by calling *ilGetInteger* with the parameter *IL\_FORMAT\_MODE*.

**Parameters**

<i>Mode</i>	New default format.
-------------	---------------------

**Returns**

IL\_TRUE if successful, on failure IL\_FALSE is returned and an error is set.

**5.1.2.5 ILboolean ILAPIENTRY ilGetBoolean ( IEnum Mode )**

Returns the current value of the *Mode*.

**5.1.2.6 void ILAPIENTRY ilGetBooleanv ( IEnum Mode, ILboolean \* Param )**

Sets *Param* equal to the current value of the *Mode*.

**5.1.2.7 IEnum ILAPIENTRY ilGetError ( void )**

Gets the last error on the error stack.

**Returns**

An enum describing the last error.

**5.1.2.8 ILint ILAPIENTRY ilGetInteger ( IEnum Mode )**

Returns the current value of the *Mode*.

Valid *Modes* are:

Mode	R/W	Default	Description
IL_CUR_IMAGE	R	0	The name of the currently bound image set by <a href="#">ilBindImage()</a> .

IL_FORMAT_MODE	RW	IL_BGRA	The format to convert loaded images into if IL_FORMAT_SET is enabled.
IL_KEEP_DXTC_DATA	RW	IL_FALSE	When loading DXTC compressed images, keep a copy of the original data around.
IL_ORIGIN_MODE	RW	IL_ORIGIN_LOWER_LEFT	Specify the origin of the image, can be <i>IL_ORIGIN_LOWER_LEFT</i> or <i>IL_ORIGIN_UPPER_LEFT</i> .
IL_MAX_QUANT_INDICES	RW	256	Maximum number of colour indices to use when quantizing images.
IL_NEU_QUANT_SAMPLE	RW	15	Number of samples to use when quantizing with NeuQuant.
IL_QUANTIZATION_MODE	RW	IL_WU_QUANT	Quantizer to use, can be <i>IL_WU_QUANT</i> or <i>IL_NEU_QUANT</i> .
IL_TYPE_MODE	RW	IL_UNSIGNED_BYTE	The type to convert loaded images into if IL_TYPE_SET is enabled.
IL_VERSION_NUM	R	IL_VERSION	The version of the image library.
IL_ACTIVE_IMAGE	R	0	The currently active sub image, set by <a href="#">ilActiveImage()</a> .
IL_ACTIVE_MIPMAP	R	0	The currently active mipmap, set by <a href="#">ilActiveMipmap()</a> .
IL_ACTIVE_LAYER	R	0	The currently active layer, set by <a href="#">ilActiveLayer()</a> .
IL_BMP_RLE	RW	IL_FALSE	Use RLE when writing BMPs.
IL_DXTC_FORMAT	RW	IL_DXT1	DXTC format to use when compressing, can be IL_DXT1, IL_DXT3, IL_DXT4, IL_DXT5, IL_DXT_NO_COMP.
IL_JPG_QUALITY	RW	99	JPEG compression quality used when writing JPEG files.
IL_PCD_PICNUM	RW	2	Select picture resolution for Kodak Photo CD files.
IL_PNG_ALPHA_INDEX	RW	-1	Define a colour index as transparent, saved in the tRNS chunk.

IL_PNG_INTERLACE	RW	IL_FALSE	Use interlacing when writing PNG files.
IL_SGI_RLE	RW	IL_FALSE	Use RLE when writing SGI files.

IL\_TGA\_CREATE\_STAMP | RW IL\_TGA\_RLE | RW | IL\_FALSE | Use RLE when writing Targa files. IL\_VTF\_COMP | RW | IL\_DXT\_NO\_COMP | Compression to use when writing VTF files, can be IL\_DXT\_NO\_COMP, IL\_DXT1, IL\_DXT3, IL\_DXT4, IL\_DXT5.

See Also

[ilGetIntegerImage](#) for image specific modes.

#### 5.1.2.9 void ILAPIENTRY ilGetInterv ( IEnum Mode, IInt \* Param )

Sets *Param* equal to the current value of the *Mode*.

See Also

[ilGetInteger](#)

#### 5.1.2.10 ILconst\_string ILAPIENTRY ilGetString ( IEnum StringName )

Returns a constant string detailing aspects about this library.

Valid StringNames are:

Name	R/W	Description
IL_VENDOR	R	The name of the vendor of this version of the IL implementation.
IL_VERSION_NUM	R	Current version string of the IL implementation.
IL_LOAD_EXT	R	A string containing extensions of all files that can be loaded.
IL_SAVE_EXT	R	A string containing extensions of all files that can be saved.
IL_TGA_ID_STRING	RW	Identifier string to be used when writing Targa image files.
IL_TGA_AUTHNAME_STRING	RW	Author name to be used when writing Targa image files.
IL_TGA_AUTHCOMMENT_STRING	RW	Author comment to be used when writing Targa image files.
IL_PNG_AUTHNAME_STRING	RW	Author name to be used when writing PNG image files.
IL_PNG_TITLE_STRING	RW	Image title to be used when writing PNG image files.
IL_PNG_DESCRIPTION_STRING	RW	Image description to be used when writing PNG image files.
IL_TIF_DESCRIPTION_STRING	RW	Image description to be used when writing TIFF image files.
IL_TIF_HOSTCOMPUTER_STRING	RW	Name of host computer to be used when writing TIFF image files.

<i>IL_TIF_DOCUMENTNAME_STRING</i>	RW	Document name to be used when writing TIFF image files.
<i>IL_TIF_AUTHNAME_STRING</i>	RW	Author name to be used when writing TIFF image files.
<i>IL_CHEAD_HEADER_STRING</i>	RW	Variable name to use when writing C headers.

Strings marked with RW can also be set using [ilSetString\(\)](#);

#### Parameters

<i>StringName</i>	String to get.
-------------------	----------------

#### 5.1.2.11 void ILAPIENTRY ilHint ( IEnum Target, IEnum Mode )

Specifies implementation-dependent performance hints.

These are only recommendations for the image library and it is free to ignore them.

Valid *Targets* are:

Target	Default	Description
<i>IL_MEM_SPEED_HINT</i>	IL_FASTEST	Preference between speed and memory usage. Can be <i>IL_LESS_MEM</i> or <i>IL_FASTEST</i> .
<i>IL_USE_COMPRESSION</i>	IL_NO_COMPRESSION	Whether to use compression when writing formats that support it optionally. Can be <i>IL_USE_COMPRESSION</i> or <i>IL_NO_COMPRESSION</i> .

#### 5.1.2.12 ILboolean ILAPIENTRY ilIsDisabled ( IEnum Mode )

Checks whether a *Mode* is not enabled.

#### See Also

[ilEnable](#)  
[ilDisable](#)

#### 5.1.2.13 ILboolean ILAPIENTRY ilIsEnabled ( IEnum Mode )

Checks whether a *Mode* is enabled.

#### See Also

[ilEnable](#)  
[ilDisable](#)

#### 5.1.2.14 void ILAPIENTRY ilSetInteger ( IEnum Mode, IInt Param )

Sets a parameter value for a *Mode*.

#### See Also

[ilGetInteger](#) for a list of valid *Modes*.



5.1.2.15 void ILAPIENTRY ilSetString ( IEnum *StringName*, const char \* *String* )

Sets a string detailing aspects about this library.

## Parameters

<i>StringName</i>	Name of string to set.
<i>String</i>	New string value, will be automatically converted to ILchar if necessary.

## 5.2 Initialization / Deinitialization

### Functions

- `void ILAPIENTRY ilInit (void)`  
*Initialize the image library.*
- `void ILAPIENTRY ilSetMemory (mAlloc mallocFunc, mFree freeFunc)`  
*Sets the memory allocation and deallocation functions.*
- `void ILAPIENTRY ilShutDown (void)`  
*Shuts down the image library.*

### 5.2.1 Detailed Description

### 5.2.2 Function Documentation

#### 5.2.2.1 `void ILAPIENTRY ilInit ( void )`

Initialize the image library.

This must be called before calling any other IL functions or their behaviour is undefined.

#### 5.2.2.2 `void ILAPIENTRY ilSetMemory ( mAlloc mallocFunc, mFree freeFunc )`

Sets the memory allocation and deallocation functions.

When changing the *freeFunc* all allocated memory up to that point will still be freed by the function that was set when that memory was allocated. This means the correct function will be called for every allocated object.

#### Parameters

<i>mallocFunc</i>	The function to call to allocate memory or NULL to reset to the default.
<i>freeFunc</i>	The function to call to free memory or NULL to reset to the default.

#### 5.2.2.3 `void ILAPIENTRY ilShutDown ( void )`

Shuts down the image library.

## 5.3 Image Management

### Functions

- [ILuint ILAPIENTRY ilCloneCurlImage \(\)](#)  
*Creates a duplicate of the currently bound image.*
- [ILboolean ILAPIENTRY ilCopyImage \(ILuint Src\)](#)  
*Copies everything from Src to the current bound image.*
- [ILint ILAPIENTRY ilGetIntegerImage \(ILuint Image, ILenum Mode\)](#)  
*Get a value about a specific image.*
- [ILboolean ILAPIENTRY ilIsImage \(ILuint Image\)](#)  
*Checks whether a given Image name is valid.*

#### 5.3.1 Detailed Description

#### 5.3.2 Function Documentation

##### 5.3.2.1 ILuint ILAPIENTRY ilCloneCurlImage ( void )

Creates a duplicate of the currently bound image.

##### 5.3.2.2 ILboolean ILAPIENTRY ilCopyImage ( ILuint Src )

Copies everything from Src to the current bound image.

##### Parameters

<i>Src</i>	Name of source image from which to copy.
------------	--

##### 5.3.2.3 ILint ILAPIENTRY ilGetIntegerImage ( ILuint Image, ILenum Mode )

Get a value about a specific image.

The *Modes* listed here can also be retrieved for the currently bound image by calling [ilGetInteger\(\)](#).

Valid Modes are:

Mode	Description
IL_DXTC_DATA_FORMAT	Format of the retained compressed DXTC data (if IL_KEEP_DXTC_DATA is enabled on load).
IL_IMAGE_BITS_PER_PIXEL	Bits per pixel.
IL_IMAGE_BYTES_PER_PIXEL	Bytes per pixel.
IL_IMAGE_BPC	Bytes per channel.
IL_IMAGE_CHANNELS	Image colour channel count.
IL_IMAGE_CUBEFLAGS	Cubemap face of image if it is a cubemap.
IL_IMAGE_DEPTH	Depth of image in pixels (number 2d images along the Z axis).
IL_IMAGE_DURATION	Duration of the image in an animation in milliseconds.
IL_IMAGE_FORMAT	Pixel format of image.
IL_IMAGE_HEIGHT	Height of image in pixels.
IL_IMAGE_SIZE_OF_DATA	Total number of bytes in image data buffer.

IL_IMAGE_OFFX	X offset of image in pixels.
IL_IMAGE_OFFY	Y offset of image in pixels.
IL_IMAGE_ORIGIN	Origin of image.
IL_IMAGE_PLANESIZE	Size of one 2d image plane in bytes.
IL_IMAGE_TYPE	Data type of bytes in data buffer.
IL_IMAGE_WIDTH	Width of image in pixels.
IL_NUM_FACES	Number of faces (== 5 for cubemaps).
IL_NUM_IMAGES	Number of following sub images (eg. in an animation).
IL_NUM_LAYERS	Number of layers in image.
IL_NUM_MIPMAPS	Number of mipmaps contained in image.
IL_PALETTE_TYPE	Type of palette data if any.
IL_PALETTE_BPP	Bytes pro palette entry.
IL_PALETTE_NUM_COLS	Total number of palette entries.
IL_PALETTE_BASE_TYPE	Pixel format for all palette entries.

#### 5.3.2.4 ILboolean ILAPIENTRY illImage ( ILuint *Image* )

Checks whether a given *Image* name is valid.

## 5.4 Image Manipulation

### Functions

- **ILuint ILAPIENTRY ilCopyPixels** (ILuint XOff, ILuint YOff, ILuint ZOff, ILuint Width, ILuint Height, ILuint Depth, ILenum Format, ILenum Type, void \*Data)  
*Copy the pixels of a region of the currently bound image to a buffer.*
- **ILuint ILAPIENTRY ilCreateSubImage** (ILenum Type, ILuint Num)  
*Creates sub images of the given type for the currently bound image.*
- **ILboolean ILAPIENTRY ilDefaultImage** ()  
*Creates an ugly 64x64 black and yellow checkerboard image.*

#### 5.4.1 Detailed Description

#### 5.4.2 Function Documentation

##### 5.4.2.1 ILuint ILAPIENTRY ilCopyPixels ( ILuint XOff, ILuint YOff, ILuint ZOff, ILuint Width, ILuint Height, ILuint Depth, ILenum Format, ILenum Type, void \* Data )

Copy the pixels of a region of the currently bound image to a buffer.

##### Parameters

<i>XOff</i>	Left border of image subregion to copy in pixels.
<i>YOff</i>	Top border of image subregion to copy in pixels.
<i>ZOff</i>	Front border of image subregion to copy in slices.
<i>Width</i>	Width of region to copy in pixels.
<i>Height</i>	Height of region to copy in pixels.
<i>Depth</i>	Depth of region to copy in slices.
<i>Format</i>	Format of pixel data to get.
<i>Type</i>	Underlying pixel data type.
<i>Data</i>	Buffer to receive pixel data.

##### 5.4.2.2 ILuint ILAPIENTRY ilCreateSubImage ( ILenum Type, ILuint Num )

Creates sub images of the given type for the currently bound image.

The new sub images will be empty. Existing sub images of the type will be replaced. The current image binding will not be changed.

##### Parameters

<i>Type</i>	Sub image type, can be <i>IL_SUB_NEXT</i> to create animation frames after the current image, <i>IL_SUB_MIPMAP</i> to create mipmaps or <i>IL_SUB_LAYER</i> to create layers.
<i>Num</i>	The number of images to create.

##### Returns

The number of images actually created.

##### Note

The original version behaved a little differently, it only created one sub image of the given type and the rest were added as frames in the animation chain. I believe this was a bug and fixed it. However if your program relied on that behaviour, it might be broken now. Be aware of that.

#### 5.4.2.3 ILboolean ILAPIENTRY ilDefaultImage ( void )

Creates an ugly 64x64 black and yellow checkerboard image.





## Chapter 6

# Data Structure Documentation

### 6.1 \_iff\_chunk Struct Reference

#### Data Fields

- [ILuint chunkType](#)
- [ILuint size](#)
- [ILuint start](#)
- [ILuint tag](#)

#### 6.1.1 Field Documentation

##### 6.1.1.1 ILuint chunkType

##### 6.1.1.2 ILuint size

##### 6.1.1.3 ILuint start

##### 6.1.1.4 ILuint tag

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_iff.c](#)

### 6.2 ALPHA\_CHUNK Struct Reference

```
#include <il_psp.h>
```

#### Data Fields

- [ILushort BitmapCount](#)
- [ILushort ChannelCount](#)

#### 6.2.1 Field Documentation

##### 6.2.1.1 ILushort BitmapCount

#### 6.2.1.2 ILushort ChannelCount

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_psp.h](#)

### 6.3 ALPHAINFO\_CHUNK Struct Reference

```
#include <il_psp.h>
```

#### Data Fields

- [PSPRECT AlphaRect](#)
- [PSPRECT AlphaSavedRect](#)

#### 6.3.1 Field Documentation

##### 6.3.1.1 PSPRECT AlphaRect

##### 6.3.1.2 PSPRECT AlphaSavedRect

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_psp.h](#)

### 6.4 BITFILE Struct Reference

```
#include <il_bits.h>
```

#### Data Fields

- [ILuint BitPos](#)
- [ILubyte Buff](#)
- [ILint ByteBitOff](#)
- [SIO \\* io](#)

#### 6.4.1 Field Documentation

##### 6.4.1.1 ILuint BitPos

##### 6.4.1.2 ILubyte Buff

##### 6.4.1.3 ILint ByteBitOff

##### 6.4.1.4 SIO\* io

The documentation for this struct was generated from the following file:

- [src/IL/il\\_bits.h](#)

## 6.5 BLOCKHEAD Struct Reference

```
#include <il_psp.h>
```

### Data Fields

- [ILushort BlockID](#)
- [ILuint BlockLen](#)
- [ILubyte HeadID](#) [4]

### 6.5.1 Field Documentation

#### 6.5.1.1 ILushort BlockID

#### 6.5.1.2 ILuint BlockLen

#### 6.5.1.3 ILubyte HeadID[4]

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_psp.h](#)

## 6.6 BLP1HEAD Struct Reference

### Data Fields

- [ILuint Compression](#)
- [ILuint Flags](#)
- [ILuint Height](#)
- [ILuint MipLengths](#) [16]
- [ILuint MipOffsets](#) [16]
- [ILuint PictureSubType](#)
- [ILuint PictureType](#)
- [ILubyte Sig](#) [4]
- [ILuint Width](#)

### 6.6.1 Field Documentation

#### 6.6.1.1 ILuint Compression

#### 6.6.1.2 ILuint Flags

#### 6.6.1.3 ILuint Height

#### 6.6.1.4 ILuint MipLengths[16]

#### 6.6.1.5 ILuint MipOffsets[16]

#### 6.6.1.6 ILuint PictureSubType

#### 6.6.1.7 ILuint PictureType

#### 6.6.1.8 ILubyte Sig[4]

#### 6.6.1.9 ILuint Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_blp.c](#)

## 6.7 BLP2HEAD Struct Reference

### Data Fields

- [ILubyte AlphaBits](#)
- [ILubyte AlphaType](#)
- [ILubyte Compression](#)
- [ILubyte HasMips](#)
- [ILuint Height](#)
- [ILuint MipLengths](#) [16]
- [ILuint MipOffsets](#) [16]
- [ILubyte Sig](#) [4]
- [ILuint Type](#)
- [ILuint Width](#)

### 6.7.1 Field Documentation

#### 6.7.1.1 ILubyte AlphaBits

#### 6.7.1.2 ILubyte AlphaType

#### 6.7.1.3 ILubyte Compression

#### 6.7.1.4 ILubyte HasMips

#### 6.7.1.5 ILuint Height

#### 6.7.1.6 ILuint MipLengths[16]

#### 6.7.1.7 ILuint MipOffsets[16]

#### 6.7.1.8 ILubyte Sig[4]

#### 6.7.1.9 ILuint Type

#### 6.7.1.10 ILuint Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_blp.c](#)

## 6.8 BMHD Struct Reference

## Data Fields

- [ILushort h](#)
- [ILshort Hpage](#)
- [ILshort Lpage](#)
- [ILubyte mask](#)
- [ILubyte pad1](#)
- [ILubyte planes](#)
- [ILushort tcolor](#)
- [ILubyte tcomp](#)
- [ILushort w](#)
- [ILshort x](#)
- [ILubyte xAspect](#)
- [ILshort y](#)
- [ILubyte yAspect](#)

### 6.8.1 Field Documentation

#### 6.8.1.1 ILushort h

#### 6.8.1.2 ILshort Hpage

#### 6.8.1.3 ILshort Lpage

#### 6.8.1.4 ILubyte mask

#### 6.8.1.5 ILubyte pad1

#### 6.8.1.6 ILubyte planes

#### 6.8.1.7 ILushort tcolor

#### 6.8.1.8 ILubyte tcomp

#### 6.8.1.9 ILushort w

#### 6.8.1.10 ILshort x

#### 6.8.1.11 ILubyte xAspect

#### 6.8.1.12 ILshort y

#### 6.8.1.13 ILubyte yAspect

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_ilbm.c](#)

## 6.9 BMPHEAD Struct Reference

```
#include <il_bmp.h>
```

## Data Fields

- [ILint bfDataOff](#)
- [Luint bfReserved](#)
- [ILint bfSize](#)
- [ILbyte bfType \[2\]](#)
- [ILshort biBitCount](#)
- [ILint biClrImportant](#)
- [ILint biClrUsed](#)
- [ILint biCompression](#)
- [ILint biHeight](#)
- [ILshort biPlanes](#)
- [ILint biSize](#)
- [ILint biSizeImage](#)
- [ILint biWidth](#)
- [ILint biXPelsPerMeter](#)
- [ILint biYPelsPerMeter](#)

### 6.9.1 Field Documentation

6.9.1.1 [ILint bfDataOff](#)

6.9.1.2 [Luint bfReserved](#)

6.9.1.3 [ILint bfSize](#)

6.9.1.4 [ILbyte bfType\[2\]](#)

6.9.1.5 [ILshort biBitCount](#)

6.9.1.6 [ILint biClrImportant](#)

6.9.1.7 [ILint biClrUsed](#)

6.9.1.8 [ILint biCompression](#)

6.9.1.9 [ILint biHeight](#)

6.9.1.10 [ILshort biPlanes](#)

6.9.1.11 [ILint biSize](#)

6.9.1.12 [ILint biSizeImage](#)

6.9.1.13 [ILint biWidth](#)

6.9.1.14 [ILint biXPelsPerMeter](#)

6.9.1.15 [ILint biYPelsPerMeter](#)

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_bmp.h](#)

## 6.10 Box Struct Reference

### Data Fields

- [ILint b0](#)
- [ILint b1](#)
- [ILint g0](#)
- [ILint g1](#)
- [ILint r0](#)
- [ILint r1](#)
- [ILint vol](#)

### 6.10.1 Field Documentation

#### 6.10.1.1 [ILint b0](#)

#### 6.10.1.2 [ILint b1](#)

#### 6.10.1.3 [ILint g0](#)

#### 6.10.1.4 [ILint g1](#)

#### 6.10.1.5 [ILint r0](#)

#### 6.10.1.6 [ILint r1](#)

#### 6.10.1.7 [ILint vol](#)

The documentation for this struct was generated from the following file:

- [src/IL/algo/il\\_quantizer.c](#)

## 6.11 BUCKET Struct Reference

### Data Fields

- [ILubyte Colours](#) [4]
- struct [BUCKET](#) \* [Next](#)

### 6.11.1 Field Documentation

#### 6.11.1.1 [ILubyte Colours](#)[4]

#### 6.11.1.2 struct [BUCKET](#)\* [Next](#)

The documentation for this struct was generated from the following file:

- [src/ILU/ilu\\_manip.c](#)

## 6.12 CHANNEL Struct Reference

```
#include <il_pic.h>
```

## Data Fields

- [ILubyte Chan](#)
- [void \\* Next](#)
- [ILubyte Size](#)
- [ILubyte Type](#)

### 6.12.1 Field Documentation

#### 6.12.1.1 ILubyte Chan

#### 6.12.1.2 void\* Next

#### 6.12.1.3 ILubyte Size

#### 6.12.1.4 ILubyte Type

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_pic.h](#)

## 6.13 CHANNEL\_CHUNK Struct Reference

```
#include <il_psp.h>
```

## Data Fields

- [ILushort BitmapType](#)
- [ILushort ChanType](#)
- [ILuint CompLen](#)
- [ILuint Length](#)

### 6.13.1 Field Documentation

#### 6.13.1.1 ILushort BitmapType

#### 6.13.1.2 ILushort ChanType

#### 6.13.1.3 ILuint CompLen

#### 6.13.1.4 ILuint Length

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_psp.h](#)

## 6.14 CLIST Struct Reference

## Data Fields

- [int n](#)
- [CONTRIB \\* p](#)



### 6.14.1 Field Documentation

6.14.1.1 `int n`

6.14.1.2 `CONTRIB* p`

The documentation for this struct was generated from the following file:

- [src/ILU/ilu\\_filter\\_rcg.c](#)

## 6.15 COL\_CUBE Struct Reference

### Data Fields

- [ILubyte Max](#) [3]
- [ILubyte Min](#) [3]
- [ILubyte Val](#) [3]

### 6.15.1 Field Documentation

6.15.1.1 `ILubyte Max[3]`

6.15.1.2 `ILubyte Min[3]`

6.15.1.3 `ILubyte Val[3]`

The documentation for this struct was generated from the following file:

- [src/IL/il\\_pal.c](#)

## 6.16 Color565 Struct Reference

```
#include <il_dds.h>
```

### Data Fields

- unsigned [nBlue](#): 5
- unsigned [nGreen](#): 6
- unsigned [nRed](#): 5

### 6.16.1 Field Documentation

6.16.1.1 `unsigned nBlue`

6.16.1.2 `unsigned nGreen`

6.16.1.3 `unsigned nRed`

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dds.h](#)

## 6.17 Color888 Struct Reference

```
#include <il_dds.h>
```

### Data Fields

- [ILubyte b](#)
- [ILubyte g](#)
- [ILubyte r](#)

### 6.17.1 Field Documentation

#### 6.17.1.1 ILubyte b

#### 6.17.1.2 ILubyte g

#### 6.17.1.3 ILubyte r

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dds.h](#)

## 6.18 Color8888 Struct Reference

```
#include <il_dds.h>
```

### Data Fields

- [ILubyte a](#)
- [ILubyte b](#)
- [ILubyte g](#)
- [ILubyte r](#)

### 6.18.1 Field Documentation

#### 6.18.1.1 ILubyte a

#### 6.18.1.2 ILubyte b

#### 6.18.1.3 ILubyte g

#### 6.18.1.4 ILubyte r

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dds.h](#)

## 6.19 CONTRIB Struct Reference

### Data Fields

- int [pixel](#)
- double [weight](#)

#### 6.19.1 Field Documentation

##### 6.19.1.1 int pixel

##### 6.19.1.2 double weight

The documentation for this struct was generated from the following file:

- [src/ILU/ilu\\_filter\\_rcg.c](#)

## 6.20 CUT\_HEAD Struct Reference

### Data Fields

- [ILushort Dummy](#)
- [ILushort Height](#)
- [ILushort Width](#)

#### 6.20.1 Field Documentation

##### 6.20.1.1 ILushort Dummy

##### 6.20.1.2 ILushort Height

##### 6.20.1.3 ILushort Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_cut.c](#)

## 6.21 DCXHEAD Struct Reference

```
#include <il_dcx.h>
```

### Data Fields

- [ILubyte Bpp](#)
- [ILushort Bps](#)
- [ILubyte ColMap](#) [48]
- [ILubyte Encoding](#)
- [ILubyte Filler](#) [54]
- [ILushort HDpi](#)
- [ILushort HScreenSize](#)

- [ILubyte Manufacturer](#)
- [ILubyte NumPlanes](#)
- [ILushort PaletteInfo](#)
- [ILubyte Reserved](#)
- [ILushort VDpi](#)
- [ILubyte Version](#)
- [ILushort VScreenSize](#)
- [ILushort Xmax](#)
- [ILushort Xmin](#)
- [ILushort Ymax](#)
- [ILushort Ymin](#)

### 6.21.1 Field Documentation

6.21.1.1 **ILubyte Bpp**

6.21.1.2 **ILushort Bps**

6.21.1.3 **ILubyte ColMap[48]**

6.21.1.4 **ILubyte Encoding**

6.21.1.5 **ILubyte Filler[54]**

6.21.1.6 **ILushort HDpi**

6.21.1.7 **ILushort HScreenSize**

6.21.1.8 **ILubyte Manufacturer**

6.21.1.9 **ILubyte NumPlanes**

6.21.1.10 **ILushort PaletteInfo**

6.21.1.11 **ILubyte Reserved**

6.21.1.12 **ILushort VDpi**

6.21.1.13 **ILubyte Version**

6.21.1.14 **ILushort VScreenSize**

6.21.1.15 **ILushort Xmax**

6.21.1.16 **ILushort Xmin**

6.21.1.17 **ILushort Ymax**

6.21.1.18 **ILushort Ymin**

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dcx.h](#)

## 6.22 DDS\_CONTEXT Struct Reference

### Data Fields

- [ILImage \\*](#) [BaseImage](#)
- [ILubyte \\*](#) [CompData](#)
- [ILuint](#) [CompSize](#)
- [ILint](#) [Depth](#)
- [ILboolean](#) [Has16BitComponents](#)
- [DDSHEAD](#) [Head](#)
- [ILint](#) [Height](#)
- [ILImage \\*](#) [Image](#)
- [ILint](#) [Width](#)

### 6.22.1 Field Documentation

#### 6.22.1.1 [ILImage \\*](#) [BaseImage](#)

#### 6.22.1.2 [ILubyte\\*](#) [CompData](#)

#### 6.22.1.3 [ILuint](#) [CompSize](#)

#### 6.22.1.4 [ILint](#) [Depth](#)

#### 6.22.1.5 [ILboolean](#) [Has16BitComponents](#)

#### 6.22.1.6 [DDSHEAD](#) [Head](#)

#### 6.22.1.7 [ILint](#) [Height](#)

#### 6.22.1.8 [ILImage\\*](#) [Image](#)

#### 6.22.1.9 [ILint](#) [Width](#)

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dds.c](#)

## 6.23 DDSHEAD Struct Reference

```
#include <il_dds.h>
```

### Data Fields

- [ILuint](#) [AlphaBitDepth](#)
- [ILuint](#) [BBitMask](#)
- [ILuint](#) [ddsCaps1](#)
- [ILuint](#) [ddsCaps2](#)
- [ILuint](#) [ddsCaps3](#)
- [ILuint](#) [ddsCaps4](#)
- [ILuint](#) [Depth](#)
- [ILuint](#) [Flags1](#)
- [ILuint](#) [Flags2](#)

- [ILuint FourCC](#)
- [ILuint GBitMask](#)
- [ILuint Height](#)
- [ILuint LinearSize](#)
- [ILuint MipMapCount](#)
- [ILuint NotUsed](#) [10]
- [ILuint RBitMask](#)
- [ILuint RGBAAlphaBitMask](#)
- [ILuint RGBBitCount](#)
- [ILbyte Signature](#) [4]
- [ILuint Size1](#)
- [ILuint Size2](#)
- [ILuint TextureStage](#)
- [ILuint Width](#)

### 6.23.1 Field Documentation

6.23.1.1 [ILuint AlphaBitDepth](#)

6.23.1.2 [ILuint BBitMask](#)

6.23.1.3 [ILuint ddsCaps1](#)

6.23.1.4 [ILuint ddsCaps2](#)

6.23.1.5 [ILuint ddsCaps3](#)

6.23.1.6 [ILuint ddsCaps4](#)

6.23.1.7 [ILuint Depth](#)

6.23.1.8 [ILuint Flags1](#)

6.23.1.9 [ILuint Flags2](#)

6.23.1.10 [ILuint FourCC](#)

6.23.1.11 [ILuint GBitMask](#)

6.23.1.12 [ILuint Height](#)

6.23.1.13 [ILuint LinearSize](#)

6.23.1.14 [ILuint MipMapCount](#)

6.23.1.15 [ILuint NotUsed](#)[10]

6.23.1.16 [ILuint RBitMask](#)

6.23.1.17 [ILuint RGBAAlphaBitMask](#)

6.23.1.18 [ILuint RGBBitCount](#)

6.23.1.19 [ILbyte Signature](#)[4]

6.23.1.20 ILuint Size1

6.23.1.21 ILuint Size2

6.23.1.22 ILuint TextureStage

6.23.1.23 ILuint Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dds.h](#)

## 6.24 DICOMHEAD Struct Reference

### Data Fields

- [ILboolean BigEndian](#)
- [ILuint BitsAllocated](#)
- [ILuint BitsStored](#)
- [ILuint DataLen](#)
- [ILuint Depth](#)
- [ILenum Encoding](#)
- [ILenum Format](#)
- [ILuint Height](#)
- [ILuint Samples](#)
- [ILubyte Signature](#) [4]
- [ILenum Type](#)
- [ILuint Version](#)
- [ILuint Width](#)

### 6.24.1 Field Documentation

6.24.1.1 ILboolean BigEndian

6.24.1.2 ILuint BitsAllocated

6.24.1.3 ILuint BitsStored

6.24.1.4 ILuint DataLen

6.24.1.5 ILuint Depth

6.24.1.6 ILenum Encoding

6.24.1.7 ILenum Format

6.24.1.8 ILuint Height

6.24.1.9 ILuint Samples

6.24.1.10 ILubyte Signature[4]

6.24.1.11 ILenum Type

#### 6.24.1.12 ILuint Version

#### 6.24.1.13 ILuint Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dicom.c](#)

### 6.25 DOOM\_HEAD Struct Reference

#### Data Fields

- [ILuint graphic\\_header](#)
- [ILushort height](#)
- [ILushort width](#)

#### 6.25.1 Field Documentation

##### 6.25.1.1 ILuint graphic\_header

##### 6.25.1.2 ILushort height

##### 6.25.1.3 ILushort width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_doom.c](#)

### 6.26 DPX\_FILE\_INFO Struct Reference

```
#include <il_dpx.h>
```

#### Data Fields

- [ILbyte Copyright](#) [200]
- [ILbyte CreateTime](#) [24]
- [ILbyte Creator](#) [100]
- [ILuint DittoKey](#)
- [ILbyte FileName](#) [100]
- [ILuint FileSize](#)
- [ILuint GenHdrSize](#)
- [ILuint IndHdrSize](#)
- [ILuint Key](#)
- [ILubyte Magic](#) [4]
- [ILuint Offset](#)
- [ILbyte Project](#) [200]
- [ILbyte Reserved](#) [104]
- [ILuint UserDataSize](#)
- [ILbyte Vers](#) [8]



### 6.26.1 Field Documentation

6.26.1.1 ILbyte Copyright[200]

6.26.1.2 ILbyte CreateTime[24]

6.26.1.3 ILbyte Creator[100]

6.26.1.4 ILuint DittoKey

6.26.1.5 ILbyte FileName[100]

6.26.1.6 ILuint FileSize

6.26.1.7 ILuint GenHdrSize

6.26.1.8 ILuint IndHdrSize

6.26.1.9 ILuint Key

6.26.1.10 ILubyte Magic[4]

6.26.1.11 ILuint Offset

6.26.1.12 ILbyte Project[200]

6.26.1.13 ILbyte Reserved[104]

6.26.1.14 ILuint UserDataSize

6.26.1.15 ILbyte Vers[8]

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dpx.h](#)

## 6.27 DPX\_IMAGE\_ELEMENT Struct Reference

```
#include <il_dpx.h>
```

### Data Fields

- [ILubyte BitSize](#)
- [ILubyte Colorimetric](#)
- [ILuint DataOffset](#)
- [ILuint DataSign](#)
- [ILbyte Description](#) [32]
- [ILubyte Descriptor](#)
- [ILushort Encoding](#)
- [ILuint EolImagePadding](#)
- [ILuint EolPadding](#)
- [ILushort Packing](#)
- [ILuint RefHighData](#)
- [R32 RefHighQuantity](#)

- [ILuint RefLowData](#)
- [R32 RefLowQuantity](#)
- [ILubyte Transfer](#)

## 6.27.1 Field Documentation

### 6.27.1.1 ILubyte BitSize

### 6.27.1.2 ILubyte Colorimetric

### 6.27.1.3 ILuint DataOffset

### 6.27.1.4 ILuint DataSign

### 6.27.1.5 ILbyte Description[32]

### 6.27.1.6 ILubyte Descriptor

### 6.27.1.7 ILushort Encoding

### 6.27.1.8 ILuint EolImagePadding

### 6.27.1.9 ILuint EolPadding

### 6.27.1.10 ILushort Packing

### 6.27.1.11 ILuint RefHighData

### 6.27.1.12 R32 RefHighQuantity

### 6.27.1.13 ILuint RefLowData

### 6.27.1.14 R32 RefLowQuantity

### 6.27.1.15 ILubyte Transfer

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dpx.h](#)

## 6.28 DPX\_IMAGE\_INFO Struct Reference

```
#include <il_dpx.h>
```

### Data Fields

- [ILuint Height](#)
- [DPX\\_IMAGE\\_ELEMENT ImageElement](#) [8]
- [ILushort NumElements](#)
- [ILushort Orientation](#)
- [ILubyte reserved](#) [52]
- [ILuint Width](#)

### 6.28.1 Field Documentation

6.28.1.1 ILuint Height

6.28.1.2 DPX\_IMAGE\_ELEMENT ImageElement[8]

6.28.1.3 ILushort NumElements

6.28.1.4 ILushort Orientation

6.28.1.5 ILubyte reserved[52]

6.28.1.6 ILuint Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dpx.h](#)

## 6.29 DPX\_IMAGE\_ORIENT Struct Reference

```
#include <il_dpx.h>
```

### Data Fields

- [ILushort Border](#) [4]
- [ILbyte CreationTime](#) [24]
- [ILbyte FileName](#) [100]
- [ILbyte InputDev](#) [32]
- [ILbyte InputSerial](#) [32]
- [ILuint PixelAspect](#) [2]
- [ILubyte Reserved](#) [28]
- [R32 XCenter](#)
- [ILuint XOffset](#)
- [ILuint XOrigSize](#)
- [R32 YCenter](#)
- [ILuint YOffset](#)
- [ILuint YOrigSize](#)

### 6.29.1 Field Documentation

6.29.1.1 ILushort Border[4]

6.29.1.2 ILbyte CreationTime[24]

6.29.1.3 ILbyte FileName[100]

6.29.1.4 ILbyte InputDev[32]

6.29.1.5 ILbyte InputSerial[32]

6.29.1.6 ILuint PixelAspect[2]

6.29.1.7 ILubyte Reserved[28]

6.29.1.8 R32 XCenter

6.29.1.9 ILuint XOffset

6.29.1.10 ILuint XOrigSize

6.29.1.11 R32 YCenter

6.29.1.12 ILuint YOffset

6.29.1.13 ILuint YOrigSize

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dpx.h](#)

## 6.30 DPX\_MOTION\_PICTURE\_HEAD Struct Reference

```
#include <il_dpx.h>
```

### Data Fields

- [ILbyte count](#) [4]
- [ILbyte film\\_mfg\\_id](#) [2]
- [ILbyte film\\_type](#) [2]
- [ILbyte format](#) [32]
- [ILbyte frame\\_id](#) [32]
- [ILuint frame\\_position](#)
- [R32 frame\\_rate](#)
- [ILuint held\\_count](#)
- [ILbyte offset](#) [2]
- [ILbyte prefix](#) [6]
- [ILubyte reserved](#) [56]
- [ILuint sequence\\_len](#)
- [R32 shutter\\_angle](#)
- [ILbyte slate\\_info](#) [100]

### 6.30.1 Field Documentation

6.30.1.1 ILbyte count[4]

6.30.1.2 ILbyte film\_mfg\_id[2]

6.30.1.3 ILbyte film\_type[2]

6.30.1.4 ILbyte format[32]

6.30.1.5 ILbyte frame\_id[32]

6.30.1.6 ILuint frame\_position

6.30.1.7 R32 frame\_rate

6.30.1.8 ILuint held\_count

6.30.1.9 ILbyte offset[2]

6.30.1.10 ILbyte prefix[6]

6.30.1.11 ILubyte reserved[56]

6.30.1.12 ILuint sequence\_len

6.30.1.13 R32 shutter\_angle

6.30.1.14 ILbyte slate\_info[100]

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dpx.h](#)

## 6.31 DPX\_TELEVISION\_HEAD Struct Reference

```
#include <il_dpx.h>
```

### Data Fields

- [R32 black\\_gain](#)
- [R32 black\\_level](#)
- [R32 break\\_point](#)
- [ILubyte field\\_num](#)
- [R32 frame\\_rate](#)
- [R32 gamma](#)
- [R32 hor\\_sample\\_rate](#)
- [R32 integration\\_times](#)
- [ILubyte interlace](#)
- [ILubyte reserved \[76\]](#)
- [ILuint tim\\_code](#)
- [R32 time\\_offset](#)
- [ILubyte unused](#)
- [ILuint userBits](#)
- [R32 ver\\_sample\\_rate](#)
- [ILubyte video\\_signal](#)
- [R32 white\\_level](#)

### 6.31.1 Field Documentation

6.31.1.1 R32 black\_gain

6.31.1.2 R32 black\_level

6.31.1.3 R32 break\_point

6.31.1.4 ILubyte field\_num

6.31.1.5 R32 frame\_rate

6.31.1.6 R32 gamma

6.31.1.7 R32 hor\_sample\_rate

6.31.1.8 R32 integration\_times

6.31.1.9 ILubyte interlace

6.31.1.10 ILubyte reserved[76]

6.31.1.11 ILuint tim\_code

6.31.1.12 R32 time\_offset

6.31.1.13 ILubyte unused

6.31.1.14 ILuint userBits

6.31.1.15 R32 ver\_sample\_rate

6.31.1.16 ILubyte video\_signal

6.31.1.17 R32 white\_level

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dpx.h](#)

## 6.32 DXTAlphaBlock3BitLinear Struct Reference

```
#include <il_dds.h>
```

### Data Fields

- [ILbyte alpha0](#)
- [ILbyte alpha1](#)
- [ILbyte stuff](#) [6]

### 6.32.1 Field Documentation

6.32.1.1 ILbyte alpha0

6.32.1.2 ILbyte alpha1

6.32.1.3 ILbyte stuff[6]

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dds.h](#)

## 6.33 DXAlphaBlockExplicit Struct Reference

```
#include <il_dds.h>
```

### Data Fields

- [ILshort row](#) [4]

#### 6.33.1 Field Documentation

##### 6.33.1.1 ILshort row[4]

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dds.h](#)

## 6.34 DXColBlock Struct Reference

```
#include <il_dds.h>
```

### Data Fields

- [ILshort col0](#)
- [ILshort col1](#)
- [ILbyte row](#) [4]

#### 6.34.1 Field Documentation

##### 6.34.1.1 ILshort col0

##### 6.34.1.2 ILshort col1

##### 6.34.1.3 ILbyte row[4]

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dds.h](#)

## 6.35 Edge Struct Reference

```
#include <ilu_region.h>
```

### Data Fields

- [ILfloat dxPerScan](#)
- [struct Edge \\* next](#)
- [ILfloat xIntersect](#)
- [ILint yUpper](#)

### 6.35.1 Field Documentation

6.35.1.1 `ILfloat dxPerScan`

6.35.1.2 `struct Edge * next`

6.35.1.3 `ILfloat xIntersect`

6.35.1.4 `ILint yUpper`

The documentation for this struct was generated from the following files:

- [src/ILU/ilu\\_region.h](#)
- [include/IL/ilu\\_region.h](#)

## 6.36 EXRHEAD Struct Reference

```
#include <il_exr.h>
```

### Data Fields

- [ILuint MagicNumber](#)
- [ILuint Version](#)

### 6.36.1 Field Documentation

6.36.1.1 `ILuint MagicNumber`

6.36.1.2 `ILuint Version`

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_exr.h](#)

## 6.37 FITSHEAD Struct Reference

### Data Fields

- [ILint BitsPixel](#)
- [ILint Depth](#)
- [ILenum Format](#)
- [ILint Height](#)
- [ILboolean IsSimple](#)
- [ILint NumAxes](#)
- [ILint NumChans](#)
- [ILenum Type](#)
- [ILint Width](#)



### 6.37.1 Field Documentation

#### 6.37.1.1 ILint BitsPixel

#### 6.37.1.2 ILint Depth

#### 6.37.1.3 ILenum Format

#### 6.37.1.4 ILint Height

#### 6.37.1.5 ILboolean IsSimple

#### 6.37.1.6 ILint NumAxes

#### 6.37.1.7 ILint NumChans

#### 6.37.1.8 ILenum Type

#### 6.37.1.9 ILint Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_fits.c](#)

## 6.38 FORM\_HEAD Struct Reference

### Data Fields

- [ILuint FORM](#)
- [ILuint FormLength](#)
- [ILuint FormName](#)

### 6.38.1 Field Documentation

#### 6.38.1.1 ILuint FORM

#### 6.38.1.2 ILuint FormLength

#### 6.38.1.3 ILuint FormName

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_rot.c](#)

## 6.39 FTX\_HEAD Struct Reference

### Data Fields

- [ILuint HasAlpha](#)
- [ILuint Height](#)
- [ILuint Width](#)

### 6.39.1 Field Documentation

#### 6.39.1.1 ILuint HasAlpha

#### 6.39.1.2 ILuint Height

#### 6.39.1.3 ILuint Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_ftx.c](#)

## 6.40 GENATT\_CHUNK Struct Reference

```
#include <il_psp.h>
```

### Data Fields

- [ILint ActiveLayer](#)
- [ILushort BitDepth](#)
- [ILuint ColourCount](#)
- [ILushort Compression](#)
- [ILuint GraphicContents](#)
- [ILubyte GreyscaleFlag](#)
- [ILint Height](#)
- [ILushort LayerCount](#)
- [ILushort PlaneCount](#)
- [ILubyte ResMetric](#)
- [ILdouble Resolution](#)
- [ILuint SizeOfImage](#)
- [ILint Width](#)

### 6.40.1 Field Documentation

#### 6.40.1.1 ILint ActiveLayer

#### 6.40.1.2 ILushort BitDepth

#### 6.40.1.3 ILuint ColourCount

#### 6.40.1.4 ILushort Compression

#### 6.40.1.5 ILuint GraphicContents

#### 6.40.1.6 ILubyte GreyscaleFlag

#### 6.40.1.7 ILint Height

#### 6.40.1.8 ILushort LayerCount

#### 6.40.1.9 ILushort PlaneCount

#### 6.40.1.10 ILubyte ResMetric

6.40.1.11 ILdouble Resolution

6.40.1.12 ILuint SizeOfImage

6.40.1.13 ILint Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_psp.h](#)

## 6.41 GifGraphicControlExtension Struct Reference

```
#include <il_gif.h>
```

### Data Fields

- [ILushort Delay](#)
- [ILubyte Flags](#)
- [ILubyte TransparentColor](#)

#### 6.41.1 Field Documentation

6.41.1.1 ILushort Delay

6.41.1.2 ILubyte Flags

6.41.1.3 ILubyte TransparentColor

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_gif.h](#)

## 6.42 GifImageDescriptor Struct Reference

```
#include <il_gif.h>
```

### Data Fields

- [ILubyte Flags](#)
- [ILushort Height](#)
- [ILushort Left](#)
- [ILushort Top](#)
- [ILushort Width](#)

#### 6.42.1 Field Documentation

6.42.1.1 ILubyte Flags

6.42.1.2 ILushort Height

#### 6.42.1.3 ILushort Left

#### 6.42.1.4 ILushort Top

#### 6.42.1.5 ILushort Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_gif.h](#)

### 6.43 GifLoadingContext Struct Reference

```
#include <il_gif.h>
```

#### Data Fields

- [ILushort Colors](#)
- [ILuint Delay](#)
- [ILubyte DisposalMethod](#)
- [ILuint Frame](#)
- [ILpal GlobalPal](#)
- [ILimage \\* Image](#)
- [ILboolean IsInterlaced](#)
- [ILpal LocalPal](#)
- [ILubyte LZWCodeSize](#)
- [ILubyte NextDisposalMethod](#)
- [ILimage \\* PrevImage](#)
- [GifLogicalScreenDescriptor Screen](#)
- [ILimage \\* Target](#)
- [ILubyte TransparentColor](#)
- [ILboolean UseLocalPal](#)
- [ILboolean UseTransparentColor](#)

#### 6.43.1 Field Documentation

##### 6.43.1.1 ILushort Colors

##### 6.43.1.2 ILuint Delay

##### 6.43.1.3 ILubyte DisposalMethod

##### 6.43.1.4 ILuint Frame

##### 6.43.1.5 ILpal GlobalPal

##### 6.43.1.6 ILimage\* Image

##### 6.43.1.7 ILboolean IsInterlaced

##### 6.43.1.8 ILpal LocalPal

##### 6.43.1.9 ILubyte LZWCodeSize

6.43.1.10 ILubyte NextDisposalMethod

6.43.1.11 ILImage\* PrevImage

6.43.1.12 GifLogicalScreenDescriptor Screen

6.43.1.13 ILImage\* Target

6.43.1.14 ILubyte TransparentColor

6.43.1.15 ILboolean UseLocalPal

6.43.1.16 ILboolean UseTransparentColor

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_gif.h](#)

## 6.44 GifLogicalScreenDescriptor Struct Reference

```
#include <il_gif.h>
```

### Data Fields

- [ILubyte Background](#)
- [ILubyte Flags](#)
- [ILushort Height](#)
- [ILubyte PixelAspect](#)
- [ILushort Width](#)

### 6.44.1 Field Documentation

6.44.1.1 ILubyte Background

6.44.1.2 ILubyte Flags

6.44.1.3 ILushort Height

6.44.1.4 ILubyte PixelAspect

6.44.1.5 ILushort Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_gif.h](#)

## 6.45 GifSignature Struct Reference

```
#include <il_gif.h>
```

## Data Fields

- char [Magic](#) [6]

### 6.45.1 Field Documentation

#### 6.45.1.1 char Magic[6]

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_gif.h](#)

## 6.46 HALOHEAD Struct Reference

### Data Fields

- [ILbyte Filetype](#)
- [ILbyte Filler](#) [20]
- [ILubyte Id](#) [2]
- [ILint Ignored](#)
- [ILushort MaxBlue](#)
- [ILushort MaxGreen](#)
- [ILushort MaxIndex](#)
- [ILushort MaxRed](#)
- [ILshort Size](#)
- [ILbyte Subtype](#)
- [ILshort Version](#)

### 6.46.1 Field Documentation

#### 6.46.1.1 ILbyte Filetype

#### 6.46.1.2 ILbyte Filler[20]

#### 6.46.1.3 ILubyte Id[2]

#### 6.46.1.4 ILint Ignored

#### 6.46.1.5 ILushort MaxBlue

#### 6.46.1.6 ILushort MaxGreen

#### 6.46.1.7 ILushort MaxIndex

#### 6.46.1.8 ILushort MaxRed

#### 6.46.1.9 ILshort Size

#### 6.46.1.10 ILbyte Subtype

#### 6.46.1.11 ILshort Version

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_pal\\_halo.c](#)

## 6.47 ICNSDATA Struct Reference

```
#include <il_icns.h>
```

### Data Fields

- [char ID \[4\]](#)
- [ILint Size](#)

### 6.47.1 Field Documentation

6.47.1.1 [char ID\[4\]](#)

6.47.1.2 [ILint Size](#)

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_icns.h](#)

## 6.48 ICNSHEAD Struct Reference

```
#include <il_icns.h>
```

### Data Fields

- [char Head \[4\]](#)
- [ILint Size](#)

### 6.48.1 Field Documentation

6.48.1.1 [char Head\[4\]](#)

6.48.1.2 [ILint Size](#)

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_icns.h](#)

## 6.49 ICODIR Struct Reference

```
#include <il_icon.h>
```

### Data Fields

- [ILshort Count](#)
- [ILshort Reserved](#)
- [ILshort Type](#)

### 6.49.1 Field Documentation

#### 6.49.1.1 ILshort Count

#### 6.49.1.2 ILshort Reserved

#### 6.49.1.3 ILshort Type

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_icon.h](#)

## 6.50 ICODIRENTRY Struct Reference

```
#include <il_icon.h>
```

### Data Fields

- [ILshort Bpp](#)
- [ILubyte Height](#)
- [ILubyte NumColours](#)
- [ILuint Offset](#)
- [ILshort Planes](#)
- [ILubyte Reserved](#)
- [ILuint SizeOfData](#)
- [ILubyte Width](#)

### 6.50.1 Field Documentation

#### 6.50.1.1 ILshort Bpp

#### 6.50.1.2 ILubyte Height

#### 6.50.1.3 ILubyte NumColours

#### 6.50.1.4 ILuint Offset

#### 6.50.1.5 ILshort Planes

#### 6.50.1.6 ILubyte Reserved

#### 6.50.1.7 ILuint SizeOfData

#### 6.50.1.8 ILubyte Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_icon.h](#)

## 6.51 ICOIMAGE Struct Reference

```
#include <il_icon.h>
```



## Data Fields

- [ILubyte \\* AND](#)
- [ILubyte \\* Data](#)
- [INFOHEAD Head](#)
- [ILubyte \\* Pal](#)

### 6.51.1 Field Documentation

#### 6.51.1.1 ILubyte\* AND

#### 6.51.1.2 ILubyte\* Data

#### 6.51.1.3 INFOHEAD Head

#### 6.51.1.4 ILubyte\* Pal

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_icon.h](#)

## 6.52 IconData Struct Reference

## Data Fields

- [ILint ico\\_color\\_type](#)
- [png\\_infop ico\\_info\\_ptr](#)
- [png\\_structp ico\\_png\\_ptr](#)

### 6.52.1 Field Documentation

#### 6.52.1.1 ILint ico\_color\_type

#### 6.52.1.2 png\_infop ico\_info\_ptr

#### 6.52.1.3 png\_structp ico\_png\_ptr

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_icon.c](#)

## 6.53 ICONDIR Struct Reference

## Data Fields

- [WORD iconCount](#)
- [WORD iconType](#)
- [WORD reserved](#)

### 6.53.1 Field Documentation

6.53.1.1 **WORD** iconCount

6.53.1.2 **WORD** iconType

6.53.1.3 **WORD** reserved

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_icon.c](#)

## 6.54 ICONDIRENTRY Struct Reference

### Data Fields

- [BYTE](#) colorsInPalette
- [DWORD](#) dataOffset
- [DWORD](#) dataSize
- [BYTE](#) height
- [BYTE](#) reserved
- [WORD](#) variant1
- [WORD](#) variant2
- [BYTE](#) width

### 6.54.1 Field Documentation

6.54.1.1 **BYTE** colorsInPalette

6.54.1.2 **DWORD** dataOffset

6.54.1.3 **DWORD** dataSize

6.54.1.4 **BYTE** height

6.54.1.5 **BYTE** reserved

6.54.1.6 **WORD** variant1

6.54.1.7 **WORD** variant2

6.54.1.8 **BYTE** width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_icon.c](#)

## 6.55 iff\_chunk\_stack Struct Reference

### Data Fields

- [int](#) chunkDepth
- [iff\\_chunk](#) chunkStack [[CHUNK\\_STACK\\_SIZE](#)]

### 6.55.1 Field Documentation

6.55.1.1 `int chunkDepth`

6.55.1.2 `iff_chunk chunkStack[CHUNK_STACK_SIZE]`

The documentation for this struct was generated from the following file:

- `src/IL/formats/il_iff.c`

## 6.56 iFormatL Struct Reference

```
#include <il_register.h>
```

### Data Fields

- [ILstring Ext](#)
- [IL\\_LOADPROC Load](#)
- struct [iFormatL](#) \* [Next](#)

### 6.56.1 Field Documentation

6.56.1.1 `ILstring Ext`

6.56.1.2 `IL_LOADPROC Load`

6.56.1.3 `struct iFormatL* Next`

The documentation for this struct was generated from the following file:

- `src/IL/il_register.h`

## 6.57 iFormatS Struct Reference

```
#include <il_register.h>
```

### Data Fields

- [ILstring Ext](#)
- struct [iFormatS](#) \* [Next](#)
- [IL\\_SAVEPROC Save](#)

### 6.57.1 Field Documentation

6.57.1.1 `ILstring Ext`

6.57.1.2 `struct iFormatS* Next`

6.57.1.3 `IL_SAVEPROC Save`

The documentation for this struct was generated from the following file:

- [src/IL/il\\_register.h](#)

## 6.58 iFree Struct Reference

```
#include <il_stack.h>
```

### Data Fields

- [ILuint Name](#)
- [void \\* Next](#)

### 6.58.1 Field Documentation

#### 6.58.1.1 ILuint Name

#### 6.58.1.2 void\* Next

The documentation for this struct was generated from the following file:

- [src/IL/il\\_stack.h](#)

## 6.59 IL\_HINTS Struct Reference

```
#include <il_states.h>
```

### Data Fields

- [ILenum CompressHint](#)
- [ILenum MemVsSpeedHint](#)

### 6.59.1 Field Documentation

#### 6.59.1.1 ILenum CompressHint

#### 6.59.1.2 ILenum MemVsSpeedHint

The documentation for this struct was generated from the following file:

- [src/IL/il\\_states.h](#)

## 6.60 IL\_STATES Struct Reference

```
#include <il_states.h>
```

## Data Fields

- [ILboolean ilAutoConvPal](#)
- [ILboolean ilBlitBlend](#)
- [ILboolean ilBmpRle](#)
- [ILchar \\* ilCHeader](#)
- [ILenum ilCompression](#)
- [ILboolean ilDefaultOnFail](#)
- [ILenum ilDxtcFormat](#)
- [ILenum ilFormatMode](#)
- [ILboolean ilFormatSet](#)
- [ILenum ilInterlace](#)
- [ILenum ilJpgFormat](#)
- [ILboolean ilJpgProgressive](#)
- [ILuint ilJpgQuality](#)
- [ILboolean ilKeepDxtcData](#)
- [ILuint ilNeuSample](#)
- [ILenum ilOriginMode](#)
- [ILboolean ilOriginSet](#)
- [ILboolean ilOverWriteFiles](#)
- [ILenum ilPcdPicNum](#)
- [ILint ilPngAlphaIndex](#)
- [ILchar \\* ilPngAuthName](#)
- [ILchar \\* ilPngDescription](#)
- [ILboolean ilPngInterlace](#)
- [ILchar \\* ilPngTitle](#)
- [ILuint ilQuantMaxIndexs](#)
- [ILenum ilQuantMode](#)
- [ILboolean ilSgiRle](#)
- [ILchar \\* iTgaAuthComment](#)
- [ILchar \\* iTgaAuthName](#)
- [ILboolean iTgaCreateStamp](#)
- [ILchar \\* iTgalD](#)
- [ILboolean iTgaRle](#)
- [ILchar \\* iTifAuthName](#)
- [ILchar \\* iTifDescription](#)
- [ILchar \\* iTifDocumentName](#)
- [ILchar \\* iTifHostComputer](#)
- [ILenum ilTypeMode](#)
- [ILboolean ilTypeSet](#)
- [ILboolean ilUseKeyColour](#)
- [ILboolean ilUseNVidiaDXT](#)
- [ILboolean ilUseSquishDXT](#)
- [ILenum ilVtfCompression](#)

### 6.60.1 Field Documentation

6.60.1.1 [ILboolean ilAutoConvPal](#)

6.60.1.2 [ILboolean ilBlitBlend](#)

6.60.1.3 [ILboolean ilBmpRle](#)

6.60.1.4 [ILchar\\* ilCHeader](#)

- 6.60.1.5 **ILenum** ilCompression
- 6.60.1.6 **ILboolean** ilDefaultOnFail
- 6.60.1.7 **ILenum** ilDxtcFormat
- 6.60.1.8 **ILenum** ilFormatMode
- 6.60.1.9 **ILboolean** ilFormatSet
- 6.60.1.10 **ILenum** ilInterlace
- 6.60.1.11 **ILenum** ilJpgFormat
- 6.60.1.12 **ILboolean** ilJpgProgressive
- 6.60.1.13 **ILuint** ilJpgQuality
- 6.60.1.14 **ILboolean** ilKeepDxtcData
- 6.60.1.15 **ILuint** ilNeuSample
- 6.60.1.16 **ILenum** ilOriginMode
- 6.60.1.17 **ILboolean** ilOriginSet
- 6.60.1.18 **ILboolean** ilOverWriteFiles
- 6.60.1.19 **ILenum** ilPcdPicNum
- 6.60.1.20 **ILint** ilPngAlphaIndex
- 6.60.1.21 **ILchar\*** ilPngAuthName
- 6.60.1.22 **ILchar\*** ilPngDescription
- 6.60.1.23 **ILboolean** ilPngInterlace
- 6.60.1.24 **ILchar\*** ilPngTitle
- 6.60.1.25 **ILuint** ilQuantMaxIndexs
- 6.60.1.26 **ILenum** ilQuantMode
- 6.60.1.27 **ILboolean** ilSgiRle
- 6.60.1.28 **ILchar\*** iTgaAuthComment
- 6.60.1.29 **ILchar\*** iTgaAuthName
- 6.60.1.30 **ILboolean** iTgaCreateStamp
- 6.60.1.31 **ILchar\*** iTgald
- 6.60.1.32 **ILboolean** iTgaRle

- 6.60.1.33 ILchar\* iTifAuthName
- 6.60.1.34 ILchar\* iTifDescription
- 6.60.1.35 ILchar\* iTifDocumentName
- 6.60.1.36 ILchar\* iTifHostComputer
- 6.60.1.37 ILenum ilTypeMode
- 6.60.1.38 ILboolean ilTypeSet
- 6.60.1.39 ILboolean ilUseKeyColour
- 6.60.1.40 ILboolean ilUseNVidiaDXT
- 6.60.1.41 ILboolean ilUseSquishDXT
- 6.60.1.42 ILenum ilVtfCompression

The documentation for this struct was generated from the following file:

- [src/IL/il\\_states.h](#)

## 6.61 ilError Class Reference

```
#include <devil_cpp_wrapper.hpp>
```

### Static Public Member Functions

- static [void Check](#) ([void](#)(\*Callback)(const char \*))
- static [void Check](#) ([void](#)(\*Callback)(ILenum))
- static [ILenum Get](#) ([void](#))
- static const char \* [String](#) ([void](#))
- static const char \* [String](#) (ILenum)

### 6.61.1 Member Function Documentation

- 6.61.1.1 [void Check](#) ( [void](#)(\*)(const char \*) *Callback* ) [static]
- 6.61.1.2 [void Check](#) ( [void](#)(\*)(ILenum) *Callback* ) [static]
- 6.61.1.3 [ILenum Get](#) ( [void](#) ) [static]
- 6.61.1.4 [const char \\* String](#) ( [void](#) ) [static]
- 6.61.1.5 [const char \\* String](#) ( [ILenum Error](#) ) [static]

The documentation for this class was generated from the following file:

- [include/IL/devil\\_cpp\\_wrapper.hpp](#)

## 6.62 iIFilters Class Reference

```
#include <devil_cpp_wrapper.hpp>
```

### Static Public Member Functions

- static [ILboolean Alienify](#) ([iImage](#) &)
- static [ILboolean BlurAvg](#) ([iImage](#) &, [ILuint Iter](#))
- static [ILboolean BlurGaussian](#) ([iImage](#) &, [ILuint Iter](#))
- static [ILboolean Contrast](#) ([iImage](#) &, [ILfloat Contrast](#))
- static [ILboolean EdgeDetectE](#) ([iImage](#) &)
- static [ILboolean EdgeDetectP](#) ([iImage](#) &)
- static [ILboolean EdgeDetectS](#) ([iImage](#) &)
- static [ILboolean Emboss](#) ([iImage](#) &)
- static [ILboolean Gamma](#) ([iImage](#) &, [ILfloat Gamma](#))
- static [ILboolean Negative](#) ([iImage](#) &)
- static [ILboolean Noisify](#) ([iImage](#) &, [ILubyte Factor](#))
- static [ILboolean Pixelize](#) ([iImage](#) &, [ILuint PixSize](#))
- static [ILboolean Saturate](#) ([iImage](#) &, [ILfloat Saturation](#))
- static [ILboolean Saturate](#) ([iImage](#) &, [ILfloat r](#), [ILfloat g](#), [ILfloat b](#), [ILfloat Saturation](#))
- static [ILboolean ScaleColours](#) ([iImage](#) &, [ILfloat r](#), [ILfloat g](#), [ILfloat b](#))
- static [ILboolean Sharpen](#) ([iImage](#) &, [ILfloat Factor](#), [ILuint Iter](#))

### 6.62.1 Member Function Documentation

6.62.1.1 [ILboolean Alienify](#) ( [iImage](#) & *Image* ) [static]

6.62.1.2 [ILboolean BlurAvg](#) ( [iImage](#) & *Image*, [ILuint Iter](#) ) [static]

6.62.1.3 [ILboolean BlurGaussian](#) ( [iImage](#) & *Image*, [ILuint Iter](#) ) [static]

6.62.1.4 [ILboolean Contrast](#) ( [iImage](#) & *Image*, [ILfloat Contrast](#) ) [static]

6.62.1.5 [ILboolean EdgeDetectE](#) ( [iImage](#) & *Image* ) [static]

6.62.1.6 [ILboolean EdgeDetectP](#) ( [iImage](#) & *Image* ) [static]

6.62.1.7 [ILboolean EdgeDetectS](#) ( [iImage](#) & *Image* ) [static]

6.62.1.8 [ILboolean Emboss](#) ( [iImage](#) & *Image* ) [static]

6.62.1.9 [ILboolean Gamma](#) ( [iImage](#) & *Image*, [ILfloat Gamma](#) ) [static]

6.62.1.10 [ILboolean Negative](#) ( [iImage](#) & *Image* ) [static]

6.62.1.11 [ILboolean Noisify](#) ( [iImage](#) & *Image*, [ILubyte Factor](#) ) [static]

6.62.1.12 [ILboolean Pixelize](#) ( [iImage](#) & *Image*, [ILuint PixSize](#) ) [static]

6.62.1.13 [ILboolean Saturate](#) ( [iImage](#) & *Image*, [ILfloat Saturation](#) ) [static]

6.62.1.14 [ILboolean Saturate](#) ( [iImage](#) & *Image*, [ILfloat r](#), [ILfloat g](#), [ILfloat b](#), [ILfloat Saturation](#) ) [static]

6.62.1.15 [ILboolean ScaleColours](#) ( [iImage](#) & *Image*, [ILfloat r](#), [ILfloat g](#), [ILfloat b](#) ) [static]



#### 6.62.1.16 ILboolean Sharpen ( *ilImage & Image*, *ILfloat Factor*, *ILuint Iter* ) [static]

The documentation for this class was generated from the following file:

- [include/IL/devil\\_cpp\\_wrapper.hpp](#)

## 6.63 ILformat Struct Reference

```
#include <il_formats.h>
```

### Data Fields

- [ILconst\\_string \\* Exts](#)
- [ILformatLoadFunc Load](#)
- [ILformatSaveFunc Save](#)
- [ILformatValidateFunc Validate](#)

#### 6.63.1 Field Documentation

##### 6.63.1.1 ILconst\_string\* Exts

##### 6.63.1.2 ILformatLoadFunc Load

##### 6.63.1.3 ILformatSaveFunc Save

##### 6.63.1.4 ILformatValidateFunc Validate

The documentation for this struct was generated from the following file:

- [src/IL/il\\_formats.h](#)

## 6.64 ILformatEntry Struct Reference

### Data Fields

- [const ILformat \\* format](#)
- [ILenum id](#)
- [const char \\* name](#)
- [struct ILformatEntry \\* next](#)

#### 6.64.1 Field Documentation

##### 6.64.1.1 const ILformat\* format

##### 6.64.1.2 ILenum id

##### 6.64.1.3 const char\* name

##### 6.64.1.4 struct ILformatEntry\* next

The documentation for this struct was generated from the following file:

- [src/IL/il\\_formats.c](#)

## 6.65 illImage Class Reference

```
#include <devil_cpp_wrapper.hpp>
```

### Public Member Functions

- [illImage \(\)](#)
- [illImage \(ILconst\\_string\)](#)
- [illImage \(const illImage &\)](#)
- [virtual ~illImage \(\)](#)
- [ILboolean ActiveImage \(ILuint\)](#)
- [ILboolean ActiveLayer \(ILuint\)](#)
- [ILboolean ActiveMipmap \(ILuint\)](#)
- [void Bind \(void\) const](#)
- [void Bind \(ILuint\)](#)
- [ILuint BindImage \(void\)](#)
- [ILuint BindImage \(ILenum\)](#)
- [ILubyte Bitpp \(void\)](#)
- [ILubyte Bpp \(void\)](#)
- [ILboolean Clear \(void\)](#)
- [void ClearColour \(ILclampf, ILclampf, ILclampf, ILclampf\)](#)
- [void Close \(void\)](#)
- [ILboolean Convert \(ILenum\)](#)
- [ILboolean Copy \(ILuint\)](#)
- [ILboolean Default \(void\)](#)
- [void Delete \(void\)](#)
- [ILuint Depth \(void\)](#)
- [ILboolean Flip \(void\)](#)
- [ILenum Format \(void\)](#)
- [ILubyte \\* GetData \(void\)](#)
- [ILuint GetId \(void\) const](#)
- [ILenum GetOrigin \(void\)](#)
- [ILubyte \\* GetPalette \(void\)](#)
- [ILuint Height \(void\)](#)
- [void iGenBind \(\)](#)
- [ILboolean Load \(ILconst\\_string\)](#)
- [ILboolean Load \(ILconst\\_string, ILenum\)](#)
- [ILuint NumImages \(void\)](#)
- [ILuint NumMipmaps \(void\)](#)
- [illImage & operator= \(ILuint\)](#)
- [illImage & operator= \(const illImage &\)](#)
- [ILenum PaletteAlphaIndex \(\)](#)
- [ILenum PaletteType \(void\)](#)
- [ILboolean Resize \(ILuint, ILuint, ILuint\)](#)
- [ILboolean Save \(ILconst\\_string\)](#)
- [ILboolean Save \(ILconst\\_string, ILenum\)](#)
- [ILboolean SwapColours \(void\)](#)
- [ILboolean TexImage \(ILuint, ILuint, ILuint, ILubyte, ILenum, ILenum, void \\*\)](#)
- [ILenum Type \(void\)](#)
- [ILuint Width \(void\)](#)

## Protected Attributes

- [ILuint Id](#)

## 6.65.1 Constructor & Destructor Documentation

6.65.1.1 `illImage ( )`

6.65.1.2 `illImage ( ILconst_string FileName )`

6.65.1.3 `illImage ( const illImage & Image )`

6.65.1.4 `~illImage ( )` `[virtual]`

## 6.65.2 Member Function Documentation

6.65.2.1 `ILboolean ActiveImage ( ILuint Number )`

6.65.2.2 `ILboolean ActiveLayer ( ILuint Number )`

6.65.2.3 `ILboolean ActiveMipmap ( ILuint Number )`

6.65.2.4 `void Bind ( void ) const`

6.65.2.5 `void Bind ( ILuint Image )`

6.65.2.6 `ILuint BindImage ( void )`

6.65.2.7 `ILuint BindImage ( ILenum )`

6.65.2.8 `ILubyte Bitpp ( void )`

6.65.2.9 `ILubyte Bpp ( void )`

6.65.2.10 `ILboolean Clear ( void )`

6.65.2.11 `void ClearColour ( ILclampf Red, ILclampf Green, ILclampf Blue, ILclampf Alpha )`

6.65.2.12 `void Close ( void )` `[inline]`

6.65.2.13 `ILboolean Convert ( ILenum NewFormat )`

6.65.2.14 `ILboolean Copy ( ILuint Src )`

6.65.2.15 `ILboolean Default ( void )`

6.65.2.16 `void Delete ( void )`

6.65.2.17 `ILuint Depth ( void )`

6.65.2.18 `ILboolean Flip ( void )`

6.65.2.19 `ILenum Format ( void )`

6.65.2.20 `ILubyte * GetData ( void )`

- 6.65.2.21 `ILuint GetId ( void ) const`
- 6.65.2.22 `ILenum GetOrigin ( void )`
- 6.65.2.23 `ILubyte * GetPalette ( void )`
- 6.65.2.24 `ILuint Height ( void )`
- 6.65.2.25 `void iGenBind ( )`
- 6.65.2.26 `ILboolean Load ( ILconst_string FileName )`
- 6.65.2.27 `ILboolean Load ( ILconst_string FileName, ILenum Type )`
- 6.65.2.28 `ILenum NumImages ( void )`
- 6.65.2.29 `ILenum NumMipmaps ( void )`
- 6.65.2.30 `ilImage & operator= ( ILuint Image )`
- 6.65.2.31 `ilImage & operator= ( const ilImage & Image )`
- 6.65.2.32 `ILenum PaletteAlphaIndex ( )`
- 6.65.2.33 `ILenum PaletteType ( void )`
- 6.65.2.34 `ILboolean Resize ( ILuint Width, ILuint Height, ILuint Depth )`
- 6.65.2.35 `ILboolean Save ( ILconst_string FileName )`
- 6.65.2.36 `ILboolean Save ( ILconst_string FileName, ILenum Type )`
- 6.65.2.37 `ILboolean SwapColours ( void )`
- 6.65.2.38 `ILboolean TexImage ( ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, ILenum Format, ILenum Type, void * Data )`
- 6.65.2.39 `ILenum Type ( void )`
- 6.65.2.40 `ILuint Width ( void )`

### 6.65.3 Field Documentation

- 6.65.3.1 `ILuint Id` `[protected]`

The documentation for this class was generated from the following file:

- [include/IL/devil\\_cpp\\_wrapper.hpp](#)

## 6.66 IImage Struct Reference

The Fundamental Image structure.

```
#include <devil_internal_exports.h>
```

## Data Fields

- [ILuint \\* AnimList](#)  
*animation list*
- [ILuint AnimSize](#)  
*animation list size*
- [ILubyte Bpc](#)  
*bytes per channel*
- [ILubyte Bpp](#)  
*bytes per pixel (now number of channels)*
- [ILuint Bps](#)  
*bytes per scanline (components for IL)*
- [ILenum CubeFlags](#)  
*cube map flags for sides present in chain*
- [ILubyte \\* Data](#)  
*the image data*
- [ILuint Depth](#)  
*the image's depth*
- [ILuint Duration](#)  
*length of the time to display this "frame"*
- [ILubyte \\* DxtcData](#)  
*compressed data*
- [ILenum DxtcFormat](#)  
*compressed data format*
- [ILuint DxtcSize](#)  
*compressed data size*
- [struct IImage \\* Faces](#)  
*next cubemap face in the chain - usu. NULL*
- [ILenum Format](#)  
*image format (in IL enum style)*
- [ILuint Height](#)  
*the image's height*
- [SIO io](#)
- [struct IImage \\* Layers](#)  
*subsequent layers in the chain - usu. NULL*
- [struct IImage \\* Mipmaps](#)  
*mipmapped versions of this image terminated by a NULL - usu. NULL*
- [struct IImage \\* Next](#)  
*next image in the chain - usu. NULL*
- [ILuint OffX](#)  
*x-offset of the image*
- [ILuint OffY](#)  
*y-offset of the image*
- [ILenum Origin](#)  
*origin of the image*
- [ILpal Pal](#)  
*palette details*
- [void \\* Profile](#)  
*colour profile*
- [ILuint ProfileSize](#)  
*colour profile size*

- [ILuint SizeOfData](#)  
*the total size of the data (in bytes)*
- [ILuint SizeOfPlane](#)  
*SizeOfData in a 2d image, size of each plane slice in a 3d image (in bytes)*
- [ILenum Type](#)  
*image type (in IL enum style)*
- [ILuint Width](#)  
*the image's width*

### 6.66.1 Detailed Description

The Fundamental Image structure.

Every bit of information about an image is stored in this internal structure.

### 6.66.2 Field Documentation

#### 6.66.2.1 ILuint\* AnimList

animation list

#### 6.66.2.2 ILuint AnimSize

animation list size

#### 6.66.2.3 ILubyte Bpc

bytes per channel

#### 6.66.2.4 ILubyte Bpp

bytes per pixel (now number of channels)

#### 6.66.2.5 ILuint Bps

bytes per scanline (components for IL)

#### 6.66.2.6 ILenum CubeFlags

cube map flags for sides present in chain

#### 6.66.2.7 ILubyte\* Data

the image data

#### 6.66.2.8 ILuint Depth

the image's depth

**6.66.2.9 ILuint Duration**

length of the time to display this "frame"

**6.66.2.10 ILubyte\* DxtcData**

compressed data

**6.66.2.11 ILenum DxtcFormat**

compressed data format

**6.66.2.12 ILuint DxtcSize**

compressed data size

**6.66.2.13 struct IImage\* Faces**

next cubemap face in the chain - usu. NULL

**6.66.2.14 ILenum Format**

image format (in IL enum style)

**6.66.2.15 ILuint Height**

the image's height

**6.66.2.16 SIO io****6.66.2.17 struct IImage\* Layers**

subsequent layers in the chain - usu. NULL

**6.66.2.18 struct IImage\* Mipmaps**

mipmapped versions of this image terminated by a NULL - usu. NULL

**6.66.2.19 struct IImage\* Next**

next image in the chain - usu. NULL

**6.66.2.20 ILuint OffX**

x-offset of the image

**6.66.2.21 ILuint OffY**

y-offset of the image

#### 6.66.2.22 ILenum Origin

origin of the image

#### 6.66.2.23 ILpal Pal

palette details

#### 6.66.2.24 void\* Profile

colour profile

#### 6.66.2.25 ILuint ProfileSize

colour profile size

#### 6.66.2.26 ILuint SizeOfData

the total size of the data (in bytes)

#### 6.66.2.27 ILuint SizeOfPlane

SizeOfData in a 2d image, size of each plane slice in a 3d image (in bytes)

#### 6.66.2.28 ILenum Type

image type (in IL enum style)

#### 6.66.2.29 ILuint Width

the image's width

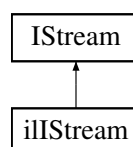
The documentation for this struct was generated from the following file:

- [include/IL/devil\\_internal\\_exports.h](#)

## 6.67 iIStream Class Reference

```
#include <il_exr.h>
```

Inheritance diagram for iIStream:





## Public Member Functions

- [iStream](#) ()
- virtual [void clear](#) ()
- virtual bool [read](#) (char c[], int n)
- virtual [void seekg](#) (Imf::Int64 Pos)
- virtual Imf::Int64 [tellg](#) ()

### 6.67.1 Constructor & Destructor Documentation

#### 6.67.1.1 iStream ( )

### 6.67.2 Member Function Documentation

#### 6.67.2.1 void clear ( ) [virtual]

#### 6.67.2.2 bool read ( char c[], int n ) [virtual]

#### 6.67.2.3 void seekg ( Imf::Int64 Pos ) [virtual]

#### 6.67.2.4 Imf::Int64 tellg ( ) [virtual]

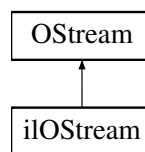
The documentation for this class was generated from the following files:

- src/IL/formats/[il\\_exr.h](#)
- src/IL/formats/[il\\_exr.c](#)

## 6.68 iostream Class Reference

```
#include <il_exr.h>
```

Inheritance diagram for iostream:



## Public Member Functions

- [iStream](#) ()
- virtual [void seekp](#) (Imf::Int64 Pos)
- virtual Imf::Int64 [tellp](#) ()
- virtual [void write](#) (const char c[], int n)

### 6.68.1 Constructor & Destructor Documentation

#### 6.68.1.1 iStream ( )

### 6.68.2 Member Function Documentation

6.68.2.1 `void seekp ( Imf::Int64 Pos ) [virtual]`

6.68.2.2 `Imf::Int64 tellp ( ) [virtual]`

6.68.2.3 `void write ( const char c[], int n ) [virtual]`

The documentation for this class was generated from the following files:

- [src/IL/formats/il\\_exr.h](#)
- [src/IL/formats/il\\_exr.c](#)

## 6.69 ILpal Struct Reference

Basic Palette struct.

```
#include <devil_internal_exports.h>
```

### Data Fields

- [ILubyte \\* Palette](#)  
*the image palette (if any)*
- [ILuint PalSize](#)  
*size of the palette (in bytes)*
- [ILenum PalType](#)  
*the palette types in [il.h](#) (0x0500 range)*

### 6.69.1 Detailed Description

Basic Palette struct.

### 6.69.2 Field Documentation

#### 6.69.2.1 ILubyte\* Palette

the image palette (if any)

#### 6.69.2.2 ILuint PalSize

size of the palette (in bytes)

#### 6.69.2.3 ILenum PalType

the palette types in [il.h](#) (0x0500 range)

The documentation for this struct was generated from the following file:

- [include/IL/devil\\_internal\\_exports.h](#)

## 6.70 ilState Class Reference

```
#include <devil_cpp_wrapper.hpp>
```

## Static Public Member Functions

- static [ILboolean Disable](#) ([ILenum](#))
- static [ILboolean Enable](#) ([ILenum](#))
- static [void Get](#) ([ILenum](#), [ILboolean &](#))
- static [void Get](#) ([ILenum](#), [ILint &](#))
- static [ILboolean GetBool](#) ([ILenum](#))
- static [ILint GetInt](#) ([ILenum](#))
- static const char \* [GetString](#) ([ILenum](#))
- static [ILboolean IsDisabled](#) ([ILenum](#))
- static [ILboolean IsEnabled](#) ([ILenum](#))
- static [ILboolean Origin](#) ([ILenum](#))
- static [void Pop](#) ([void](#))
- static [void Push](#) ([ILuint](#))

## 6.70.1 Member Function Documentation

6.70.1.1 [ILboolean Disable](#) ( [ILenum State](#) ) [static]

6.70.1.2 [ILboolean Enable](#) ( [ILenum State](#) ) [static]

6.70.1.3 [void Get](#) ( [ILenum Mode](#), [ILboolean & Param](#) ) [static]

6.70.1.4 [void Get](#) ( [ILenum Mode](#), [ILint & Param](#) ) [static]

6.70.1.5 [ILboolean GetBool](#) ( [ILenum Mode](#) ) [static]

6.70.1.6 [ILint GetInt](#) ( [ILenum Mode](#) ) [static]

6.70.1.7 const char \* [GetString](#) ( [ILenum StringName](#) ) [static]

6.70.1.8 [ILboolean IsDisabled](#) ( [ILenum Mode](#) ) [static]

6.70.1.9 [ILboolean IsEnabled](#) ( [ILenum Mode](#) ) [static]

6.70.1.10 [ILboolean Origin](#) ( [ILenum Mode](#) ) [static]

6.70.1.11 [void Pop](#) ( [void](#) ) [static]

6.70.1.12 [void Push](#) ( [ILuint Bits = IL\\_ALL\\_ATTRIB\\_BITS](#) ) [static]

The documentation for this class was generated from the following file:

- [include/IL/devil\\_cpp\\_wrapper.hpp](#)

## 6.71 ILUinfo Struct Reference

```
#include <ilu.h>
```

## Data Fields

- [ILubyte Bpp](#)
- [ILenum CubeFlags](#)

- [ILubyte \\* Data](#)
- [ILuint Depth](#)
- [ILenum Format](#)
- [ILuint Height](#)
- [ILuint Id](#)
- [ILuint NumLayers](#)
- [ILuint NumMips](#)
- [ILuint NumNext](#)
- [ILenum Origin](#)
- [ILubyte \\* Palette](#)
- [ILuint PalSize](#)
- [ILenum PalType](#)
- [ILuint SizeOfData](#)
- [ILenum Type](#)
- [ILuint Width](#)

### 6.71.1 Field Documentation

#### 6.71.1.1 ILubyte Bpp

#### 6.71.1.2 ILenum CubeFlags

#### 6.71.1.3 ILubyte\* Data

#### 6.71.1.4 ILuint Depth

#### 6.71.1.5 ILenum Format

#### 6.71.1.6 ILuint Height

#### 6.71.1.7 ILuint Id

#### 6.71.1.8 ILuint NumLayers

#### 6.71.1.9 ILuint NumMips

#### 6.71.1.10 ILuint NumNext

#### 6.71.1.11 ILenum Origin

#### 6.71.1.12 ILubyte\* Palette

#### 6.71.1.13 ILuint PalSize

#### 6.71.1.14 ILenum PalType

#### 6.71.1.15 ILuint SizeOfData

#### 6.71.1.16 ILenum Type

#### 6.71.1.17 ILuint Width

The documentation for this struct was generated from the following file:

- [include/IL/ilu.h](#)

## 6.72 ILUpointf Struct Reference

```
#include <ilu.h>
```

### Data Fields

- [ILfloat x](#)
- [ILfloat y](#)

### 6.72.1 Field Documentation

#### 6.72.1.1 ILfloat x

#### 6.72.1.2 ILfloat y

The documentation for this struct was generated from the following file:

- [include/IL/ilu.h](#)

## 6.73 ILUpointi Struct Reference

```
#include <ilu.h>
```

### Data Fields

- [ILint x](#)
- [ILint y](#)

### 6.73.1 Field Documentation

#### 6.73.1.1 ILint x

#### 6.73.1.2 ILint y

The documentation for this struct was generated from the following file:

- [include/IL/ilu.h](#)

## 6.74 ILUT\_STATES Struct Reference

```
#include <ilut_states.h>
```

### Data Fields

- [ILint D3DAlphaKeyColor](#)
- [ILuint D3DMipLevels](#)
- [ILenum D3DPool](#)
- [ILboolean ilutAutodetectTextureTarget](#)
- [ILenum ilutDXTCFormat](#)

- [ILboolean ilutForceIntegerFormat](#)
- [ILboolean ilutGenS3TC](#)
- [ILboolean ilutOglConv](#)
- [ILboolean ilutUsePalettes](#)
- [ILboolean ilutUseS3TC](#)
- [ILint MaxTexD](#)
- [ILint MaxTexH](#)
- [ILint MaxTexW](#)

### 6.74.1 Field Documentation

6.74.1.1 [ILint D3DAlphaKeyColor](#)

6.74.1.2 [ILuint D3DMipLevels](#)

6.74.1.3 [ILenum D3DPool](#)

6.74.1.4 [ILboolean ilutAutodetectTextureTarget](#)

6.74.1.5 [ILenum ilutDXTCFormat](#)

6.74.1.6 [ILboolean ilutForceIntegerFormat](#)

6.74.1.7 [ILboolean ilutGenS3TC](#)

6.74.1.8 [ILboolean ilutOglConv](#)

6.74.1.9 [ILboolean ilutUsePalettes](#)

6.74.1.10 [ILboolean ilutUseS3TC](#)

6.74.1.11 [ILint MaxTexD](#)

6.74.1.12 [ILint MaxTexH](#)

6.74.1.13 [ILint MaxTexW](#)

The documentation for this struct was generated from the following file:

- [src/ILUT/ilut\\_states.h](#)

## 6.75 ilValidate Class Reference

```
#include <devil_cpp_wrapper.hpp>
```

### Static Public Member Functions

- static [ILboolean Valid](#) ([ILenum](#), [ILconst\\_string](#))
- static [ILboolean Valid](#) ([ILenum](#), FILE \*)
- static [ILboolean Valid](#) ([ILenum](#), void \*, [ILuint](#))

### 6.75.1 Member Function Documentation

6.75.1.1 `ILboolean Valid ( IEnum Type, ILconst_string FileName )` `[static]`

6.75.1.2 `ILboolean Valid ( IEnum Type, FILE * File )` `[static]`

6.75.1.3 `ILboolean Valid ( IEnum Type, void * Lump, ILuint Size )` `[static]`

The documentation for this class was generated from the following file:

- [include/IL/devil\\_cpp\\_wrapper.hpp](#)

## 6.76 INFOHEAD Struct Reference

```
#include <il_icon.h>
```

### Data Fields

- [ILshort BitCount](#)
- [ILint ColourImportant](#)
- [ILint ColourUsed](#)
- [ILint Compression](#)
- [ILint Height](#)
- [ILshort Planes](#)
- [ILint Size](#)
- [ILint SizeImage](#)
- [ILint Width](#)
- [ILint XPixPerMeter](#)
- [ILint YPixPerMeter](#)

### 6.76.1 Field Documentation

6.76.1.1 `ILshort BitCount`

6.76.1.2 `ILint ColourImportant`

6.76.1.3 `ILint ColourUsed`

6.76.1.4 `ILint Compression`

6.76.1.5 `ILint Height`

6.76.1.6 `ILshort Planes`

6.76.1.7 `ILint Size`

6.76.1.8 `ILint SizeImage`

6.76.1.9 `ILint Width`

6.76.1.10 `ILint XPixPerMeter`

#### 6.76.1.11 ILint YPixPerMeter

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_icon.h](#)

## 6.77 iread\_mgr Struct Reference

### Data Fields

- JOCTET \* [buffer](#)
- SIO \* [io](#)
- struct jpeg\_source\_mgr [pub](#)
- boolean [start\\_of\\_file](#)

### 6.77.1 Field Documentation

#### 6.77.1.1 JOCTET\* buffer

#### 6.77.1.2 SIO\* io

#### 6.77.1.3 struct jpeg\_source\_mgr pub

#### 6.77.1.4 boolean start\_of\_file

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_jpeg.c](#)

## 6.78 iSgiHeader Struct Reference

```
#include <il_sgi.h>
```

### Data Fields

- ILbyte [Bpc](#)
- ILint [ColMap](#)
- ILushort [Dim](#)
- ILbyte [Dummy](#) [404]
- ILint [Dummy1](#)
- ILshort [MagicNum](#)
- ILbyte [Name](#) [80]
- ILint [PixMax](#)
- ILint [PixMin](#)
- ILbyte [Storage](#)
- ILushort [XSize](#)
- ILushort [YSize](#)
- ILushort [ZSize](#)



### 6.78.1 Field Documentation

6.78.1.1 ILbyte Bpc

6.78.1.2 ILint ColMap

6.78.1.3 ILushort Dim

6.78.1.4 ILbyte Dummy[404]

6.78.1.5 ILint Dummy1

6.78.1.6 ILshort MagicNum

6.78.1.7 ILbyte Name[80]

6.78.1.8 ILint PixMax

6.78.1.9 ILint PixMin

6.78.1.10 ILbyte Storage

6.78.1.11 ILushort XSize

6.78.1.12 ILushort YSize

6.78.1.13 ILushort ZSize

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_sgi.h](#)

## 6.79 IWIHEAD Struct Reference

### Data Fields

- [ILubyte Flags](#)
- [ILubyte Format](#)
- [ILushort Height](#)
- [ILuint Signature](#)
- [ILubyte Unknown](#) [18]
- [ILushort Width](#)

### 6.79.1 Field Documentation

6.79.1.1 ILubyte Flags

6.79.1.2 ILubyte Format

6.79.1.3 ILushort Height

6.79.1.4 ILuint Signature

#### 6.79.1.5 ILubyte Unknown[18]

#### 6.79.1.6 ILushort Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_iwi.c](#)

### 6.80 iwrite\_mgr Struct Reference

#### Data Fields

- [ILboolean bah](#)
- [JOCTET \\* buffer](#)
- [SIO \\* io](#)
- [struct jpeg\\_destination\\_mgr pub](#)

#### 6.80.1 Field Documentation

##### 6.80.1.1 ILboolean bah

##### 6.80.1.2 JOCTET\* buffer

##### 6.80.1.3 SIO\* io

##### 6.80.1.4 struct jpeg\_destination\_mgr pub

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_jpeg.c](#)

### 6.81 LAYERBITMAP\_CHUNK Struct Reference

```
#include <il_psp.h>
```

#### Data Fields

- [ILushort NumBitmaps](#)
- [ILushort NumChannels](#)

#### 6.81.1 Field Documentation

##### 6.81.1.1 ILushort NumBitmaps

##### 6.81.1.2 ILushort NumChannels

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_psp.h](#)

## 6.82 LAYERINFO\_CHUNK Struct Reference

```
#include <il_psp.h>
```

### Data Fields

- [ILubyte BlendingMode](#)
- [ILushort BlendRange](#)
- [ILubyte DestBlend1](#) [4]
- [ILubyte DestBlend2](#) [4]
- [ILubyte DestBlend3](#) [4]
- [ILubyte DestBlend4](#) [4]
- [ILubyte DestBlend5](#) [4]
- [PSPRECT ImageRect](#)
- [ILubyte InvertMaskBlend](#)
- [ILubyte LayerFlags](#)
- [ILubyte LayerType](#)
- [ILubyte LinkID](#)
- [ILubyte MaskDisabled](#)
- [ILubyte MaskLinked](#)
- [PSPRECT MaskRect](#)
- [ILubyte Opacity](#)
- [PSPRECT SavedImageRect](#)
- [PSPRECT SavedMaskRect](#)
- [ILubyte SourceBlend1](#) [4]
- [ILubyte SourceBlend2](#) [4]
- [ILubyte SourceBlend3](#) [4]
- [ILubyte SourceBlend4](#) [4]
- [ILubyte SourceBlend5](#) [4]
- [ILubyte TransProtFlag](#)

### 6.82.1 Field Documentation

6.82.1.1 [ILubyte BlendingMode](#)

6.82.1.2 [ILushort BlendRange](#)

6.82.1.3 [ILubyte DestBlend1](#)[4]

6.82.1.4 [ILubyte DestBlend2](#)[4]

6.82.1.5 [ILubyte DestBlend3](#)[4]

6.82.1.6 [ILubyte DestBlend4](#)[4]

6.82.1.7 [ILubyte DestBlend5](#)[4]

6.82.1.8 [PSPRECT ImageRect](#)

6.82.1.9 [ILubyte InvertMaskBlend](#)

6.82.1.10 [ILubyte LayerFlags](#)

- 6.82.1.11 ILubyte LayerType
- 6.82.1.12 ILubyte LinkID
- 6.82.1.13 ILubyte MaskDisabled
- 6.82.1.14 ILubyte MaskLinked
- 6.82.1.15 PSPRECT MaskRect
- 6.82.1.16 ILubyte Opacity
- 6.82.1.17 PSPRECT SavedImageRect
- 6.82.1.18 PSPRECT SavedMaskRect
- 6.82.1.19 ILubyte SourceBlend1[4]
- 6.82.1.20 ILubyte SourceBlend2[4]
- 6.82.1.21 ILubyte SourceBlend3[4]
- 6.82.1.22 ILubyte SourceBlend4[4]
- 6.82.1.23 ILubyte SourceBlend5[4]
- 6.82.1.24 ILubyte TransProtFlag

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_esp.h](#)

## 6.83 LIF\_HEAD Struct Reference

```
#include <il_lif.h>
```

### Data Fields

- [ILuint Flags](#)
- [ILuint Height](#)
- [char Id \[8\]](#)
- [ILuint ImageCRC](#)
- [ILuint PaletteCRC](#)
- [ILuint PalOffset](#)
- [ILuint TeamEffect0](#)
- [ILuint TeamEffect1](#)
- [ILuint Version](#)
- [ILuint Width](#)

### 6.83.1 Field Documentation

#### 6.83.1.1 ILuint Flags

6.83.1.2 ILuint Height

6.83.1.3 char Id[8]

6.83.1.4 ILuint ImageCRC

6.83.1.5 ILuint PaletteCRC

6.83.1.6 ILuint PalOffset

6.83.1.7 ILuint TeamEffect0

6.83.1.8 ILuint TeamEffect1

6.83.1.9 ILuint Version

6.83.1.10 ILuint Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_lif.h](#)

## 6.84 LZWInputStream Struct Reference

```
#include <il_gif.h>
```

### Data Fields

- [ILuint ClearCode](#)
- [ILubyte CodeSize](#)
- [GifLoadingContext \\* Ctx](#)
- [ILuint EndCode](#)
- [ILubyte InputAvail](#)
- [ILuint InputBitCount](#)
- [ILuint InputBits](#)
- [ILubyte InputBuffer \[256\]](#)
- [ILubyte InputPos](#)
- [ILuint NextCode](#)
- [ILubyte OriginalCodeSize](#)
- [ILubyte OutputBuffer \[4096\]](#)
- [ILuint OutputBufferLen](#)
- [ILuint \\* Phrases \[4096\]](#)
- [ILuint PrevCode](#)

### 6.84.1 Field Documentation

6.84.1.1 ILuint ClearCode

6.84.1.2 ILubyte CodeSize

6.84.1.3 GifLoadingContext\* Ctx

6.84.1.4 ILuint EndCode

- 6.84.1.5 ILubyte InputAvail
- 6.84.1.6 ILuint InputBitCount
- 6.84.1.7 ILuint InputBits
- 6.84.1.8 ILubyte InputBuffer[256]
- 6.84.1.9 ILubyte InputPos
- 6.84.1.10 ILuint NextCode
- 6.84.1.11 ILubyte OriginalCodeSize
- 6.84.1.12 ILubyte OutputBuffer[4096]
- 6.84.1.13 ILuint OutputBufferLen
- 6.84.1.14 ILuint\* Phrases[4096]
- 6.84.1.15 ILuint PrevCode

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_gif.h](#)

## 6.85 MDL\_HEAD Struct Reference

```
#include <il_md1.h>
```

### Data Fields

- [ILubyte Magic](#) [4]
- [ILuint Version](#)

### 6.85.1 Field Documentation

- 6.85.1.1 ILubyte Magic[4]
- 6.85.1.2 ILuint Version

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_md1.h](#)

## 6.86 MP3HEAD Struct Reference

### Data Fields

- [ILubyte Flags](#)
- [ILuint Length](#)
- char [Signature](#) [3]

- [ILubyte VersionMajor](#)
- [ILubyte VersionMinor](#)

### 6.86.1 Field Documentation

#### 6.86.1.1 ILubyte Flags

#### 6.86.1.2 ILuint Length

#### 6.86.1.3 char Signature[3]

#### 6.86.1.4 ILubyte VersionMajor

#### 6.86.1.5 ILubyte VersionMinor

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_mp3.c](#)

## 6.87 NeuQuantContext Struct Reference

### Data Fields

- [ILint alphadec](#)
- int [bias](#) [[netsize](#)]
- int [freq](#) [[netsize](#)]
- int [lengthcount](#)
- int [netindex](#) [256]
- int [netsizethink](#)
- [pixel network](#) [[netsize](#)]
- int [radpower](#) [[initrad](#)]
- int [samplefac](#)
- unsigned char \* [thepicture](#)

### 6.87.1 Field Documentation

#### 6.87.1.1 ILint alphadec

#### 6.87.1.2 int bias[netsize]

#### 6.87.1.3 int freq[netsize]

#### 6.87.1.4 int lengthcount

#### 6.87.1.5 int netindex[256]

#### 6.87.1.6 int netsizethink

#### 6.87.1.7 pixel network[netsize]

#### 6.87.1.8 int radpower[initrad]

6.87.1.9 int samplefac

6.87.1.10 unsigned char\* thepicture

The documentation for this struct was generated from the following file:

- [src/IL/algo/il\\_neuquant.c](#)

## 6.88 OS2\_HEAD Struct Reference

```
#include <il_bmp.h>
```

### Data Fields

- [ILushort bfType](#)
- [ILuint biSize](#)
- [ILuint cbFix](#)
- [ILushort cBitCount](#)
- [ILushort cPlanes](#)
- [ILushort cx](#)
- [ILushort cy](#)
- [ILuint DataOff](#)
- [ILshort xHotspot](#)
- [ILshort yHotspot](#)

### 6.88.1 Field Documentation

6.88.1.1 ILushort bfType

6.88.1.2 ILuint biSize

6.88.1.3 ILuint cbFix

6.88.1.4 ILushort cBitCount

6.88.1.5 ILushort cPlanes

6.88.1.6 ILushort cx

6.88.1.7 ILushort cy

6.88.1.8 ILuint DataOff

6.88.1.9 ILshort xHotspot

6.88.1.10 ILshort yHotspot

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_bmp.h](#)



## 6.89 PCXHEAD Struct Reference

```
#include <il_pcx.h>
```

### Data Fields

- [ILubyte Bpp](#)
- [ILushort Bps](#)
- [ILubyte ColMap \[48\]](#)
- [ILubyte Encoding](#)
- [ILubyte Filler \[54\]](#)
- [ILushort HDpi](#)
- [ILushort HScreenSize](#)
- [ILubyte Manufacturer](#)
- [ILubyte NumPlanes](#)
- [ILushort PaletteInfo](#)
- [ILubyte Reserved](#)
- [ILushort VDpi](#)
- [ILubyte Version](#)
- [ILushort VScreenSize](#)
- [ILushort Xmax](#)
- [ILushort Xmin](#)
- [ILushort Ymax](#)
- [ILushort Ymin](#)

### 6.89.1 Field Documentation

#### 6.89.1.1 ILubyte Bpp

#### 6.89.1.2 ILushort Bps

#### 6.89.1.3 ILubyte ColMap[48]

#### 6.89.1.4 ILubyte Encoding

#### 6.89.1.5 ILubyte Filler[54]

#### 6.89.1.6 ILushort HDpi

#### 6.89.1.7 ILushort HScreenSize

#### 6.89.1.8 ILubyte Manufacturer

#### 6.89.1.9 ILubyte NumPlanes

#### 6.89.1.10 ILushort PaletteInfo

#### 6.89.1.11 ILubyte Reserved

#### 6.89.1.12 ILushort VDpi

#### 6.89.1.13 ILubyte Version

#### 6.89.1.14 ILushort VScreenSize

6.89.1.15 ILushort Xmax

6.89.1.16 ILushort Xmin

6.89.1.17 ILushort Ymax

6.89.1.18 ILushort Ymin

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_pcx.h](#)

## 6.90 PIC\_HEAD Struct Reference

```
#include <il_pic.h>
```

### Data Fields

- [ILbyte Comment](#) [80]
- [ILshort Fields](#)
- [ILshort Height](#)
- [ILbyte Id](#) [4]
- [ILint Magic](#)
- [ILshort Padding](#)
- [ILfloat Ratio](#)
- [ILfloat Version](#)
- [ILshort Width](#)

### 6.90.1 Field Documentation

6.90.1.1 ILbyte Comment[80]

6.90.1.2 ILshort Fields

6.90.1.3 ILshort Height

6.90.1.4 ILbyte Id[4]

6.90.1.5 ILint Magic

6.90.1.6 ILshort Padding

6.90.1.7 ILfloat Ratio

6.90.1.8 ILfloat Version

6.90.1.9 ILshort Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_pic.h](#)

## 6.91 PIXHEAD Struct Reference

### Data Fields

- [ILushort Bpp](#)
- [ILubyte Bpp](#)
- [ILushort Height](#)
- [ILushort OffX](#)
- [ILushort OffY](#)
- [ILubyte Reserved1](#) [413]
- [ILubyte Reserved2](#) [4]
- [ILubyte Signature](#) [2]
- [ILushort Width](#)

### 6.91.1 Field Documentation

#### 6.91.1.1 ILushort Bpp

#### 6.91.1.2 ILubyte Bpp

#### 6.91.1.3 ILushort Height

#### 6.91.1.4 ILushort OffX

#### 6.91.1.5 ILushort OffY

#### 6.91.1.6 ILubyte Reserved1[413]

#### 6.91.1.7 ILubyte Reserved2[4]

#### 6.91.1.8 ILubyte Signature[2]

#### 6.91.1.9 ILushort Width

The documentation for this struct was generated from the following files:

- [src/IL/formats/il\\_pix.c](#)
- [src/IL/formats/il\\_pxr.c](#)

## 6.92 PNGData Struct Reference

### Data Fields

- [png\\_infop](#) [info\\_ptr](#)
- [ILint](#) [png\\_color\\_type](#)
- [png\\_structp](#) [png\\_ptr](#)

### 6.92.1 Field Documentation

#### 6.92.1.1 png\_infop info\_ptr

#### 6.92.1.2 ILint png\_color\_type

### 6.92.1.3 png\_structp png\_ptr

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_png.c](#)

## 6.93 PPMINFO Struct Reference

```
#include <il_pnm.h>
```

### Data Fields

- [ILubyte Bpp](#)
- [ILuint Height](#)
- [ILuint MaxColour](#)
- [ILenum Type](#)
- [ILuint Width](#)

### 6.93.1 Field Documentation

#### 6.93.1.1 ILubyte Bpp

#### 6.93.1.2 ILuint Height

#### 6.93.1.3 ILuint MaxColour

#### 6.93.1.4 ILenum Type

#### 6.93.1.5 ILuint Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_pnm.h](#)

## 6.94 PSDHEAD Struct Reference

```
#include <il_psd.h>
```

### Data Fields

- [ILushort Channels](#)
- [ILushort Depth](#)
- [ILuint Height](#)
- [ILushort Mode](#)
- [ILubyte Reserved](#) [6]
- [ILubyte Signature](#) [4]
- [ILushort Version](#)
- [ILuint Width](#)

### 6.94.1 Field Documentation

#### 6.94.1.1 ILushort Channels

#### 6.94.1.2 ILushort Depth

#### 6.94.1.3 ILuint Height

#### 6.94.1.4 ILushort Mode

#### 6.94.1.5 ILubyte Reserved[6]

#### 6.94.1.6 ILubyte Signature[4]

#### 6.94.1.7 ILushort Version

#### 6.94.1.8 ILuint Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_psd.h](#)

## 6.95 PSP\_CTX Struct Reference

### Data Fields

- [ILubyte \\* Alpha](#)
- [GENATT\\_CHUNK AttChunk](#)
- [ILubyte \\*\\* Channels](#)
- [PSPHEAD Header](#)
- [ILimage \\* Image](#)
- [SIO \\* io](#)
- [ILuint NumChannels](#)
- [ILpal Pal](#)

### 6.95.1 Field Documentation

#### 6.95.1.1 ILubyte\* Alpha

#### 6.95.1.2 GENATT\_CHUNK AttChunk

#### 6.95.1.3 ILubyte\*\* Channels

#### 6.95.1.4 PSPHEAD Header

#### 6.95.1.5 ILimage\* Image

#### 6.95.1.6 SIO\* io

#### 6.95.1.7 ILuint NumChannels

#### 6.95.1.8 ILpal Pal

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_psp.c](#)

## 6.96 PSPHEAD Struct Reference

```
#include <il_psp.h>
```

### Data Fields

- char [FileSig](#) [32]
- [ILushort MajorVersion](#)
- [ILushort MinorVersion](#)

### 6.96.1 Field Documentation

6.96.1.1 char [FileSig](#)[32]

6.96.1.2 [ILushort MajorVersion](#)

6.96.1.3 [ILushort MinorVersion](#)

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_psp.h](#)

## 6.97 PSPRECT Struct Reference

```
#include <il_psp.h>
```

### Data Fields

- [ILuint x1](#)
- [ILuint x2](#)
- [ILuint y1](#)
- [ILuint y2](#)

### 6.97.1 Field Documentation

6.97.1.1 [ILuint x1](#)

6.97.1.2 [ILuint x2](#)

6.97.1.3 [ILuint y1](#)

6.97.1.4 [ILuint y2](#)

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_psp.h](#)

## 6.98 R32 Struct Reference

```
#include <il_dpx.h>
```

### Data Fields

- [ILubyte a](#)
- [ILubyte b](#)
- [ILubyte g](#)
- [ILubyte r](#)

### 6.98.1 Field Documentation

6.98.1.1 [ILubyte a](#)

6.98.1.2 [ILubyte b](#)

6.98.1.3 [ILubyte g](#)

6.98.1.4 [ILubyte r](#)

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_dpx.h](#)

## 6.99 RAW\_HEAD Struct Reference

### Data Fields

- [ILubyte Bpc](#)
- [ILubyte Bpp](#)
- [ILuint Depth](#)
- [ILuint Height](#)
- [ILuint Width](#)

### 6.99.1 Field Documentation

6.99.1.1 [ILubyte Bpc](#)

6.99.1.2 [ILubyte Bpp](#)

6.99.1.3 [ILuint Depth](#)

6.99.1.4 [ILuint Height](#)

6.99.1.5 [ILuint Width](#)

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_raw.c](#)

## 6.100 rgbe\_header\_info Struct Reference

### Data Fields

- [ILfloat exposure](#)
- [ILfloat gamma](#)
- [ILbyte programtype](#) [16]
- [ILuint valid](#)

### 6.100.1 Field Documentation

#### 6.100.1.1 ILfloat exposure

#### 6.100.1.2 ILfloat gamma

#### 6.100.1.3 ILbyte programtype[16]

#### 6.100.1.4 ILuint valid

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_hdr.c](#)

## 6.101 ROT\_HEAD Struct Reference

### Data Fields

- [ILuint Format](#)
- [FORM\\_HEAD FormHead](#)
- [ILuint Height](#)
- [ILuint Width](#)

### 6.101.1 Field Documentation

#### 6.101.1.1 ILuint Format

#### 6.101.1.2 FORM\_HEAD FormHead

#### 6.101.1.3 ILuint Height

#### 6.101.1.4 ILuint Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_rot.c](#)

## 6.102 SIO Struct Reference

```
#include <devil_internal_exports.h>
```



## Data Fields

- fCloseProc [close](#)
- fEofProc [eof](#)
- fGetcProc [getchar](#)
- ILHANDLE [handle](#)
- const void \* [lump](#)
- ILint64 [lumpPos](#)
- ILuint [lumpSize](#)
- fOpenProc [openReadOnly](#)
- fOpenProc [openWrite](#)
- fPutcProc [putchar](#)
- fReadProc [read](#)
- ILuint [ReadFileStart](#)
- fSeekProc [seek](#)
- fTellProc [tell](#)
- fWriteProc [write](#)
- ILuint [WriteFileStart](#)

### 6.102.1 Field Documentation

6.102.1.1 fCloseProc [close](#)

6.102.1.2 fEofProc [eof](#)

6.102.1.3 fGetcProc [getchar](#)

6.102.1.4 ILHANDLE [handle](#)

6.102.1.5 const void\* [lump](#)

6.102.1.6 ILint64 [lumpPos](#)

6.102.1.7 ILuint [lumpSize](#)

6.102.1.8 fOpenProc [openReadOnly](#)

6.102.1.9 fOpenProc [openWrite](#)

6.102.1.10 fPutcProc [putchar](#)

6.102.1.11 fReadProc [read](#)

6.102.1.12 ILuint [ReadFileStart](#)

6.102.1.13 fSeekProc [seek](#)

6.102.1.14 fTellProc [tell](#)

6.102.1.15 fWriteProc [write](#)

6.102.1.16 ILuint [WriteFileStart](#)

The documentation for this struct was generated from the following file:

- [include/IL/devil\\_internal\\_exports.h](#)

## 6.103 SUNHEAD Struct Reference

### Data Fields

- [ILuint BitsPerPixel](#)
- [ILuint ColorMapLength](#)
- [ILuint ColorMapType](#)
- [ILuint DataSize](#)
- [ILuint Height](#)
- [ILuint MagicNumber](#)
- [ILuint Type](#)
- [ILuint Width](#)

### 6.103.1 Field Documentation

#### 6.103.1.1 ILuint BitsPerPixel

#### 6.103.1.2 ILuint ColorMapLength

#### 6.103.1.3 ILuint ColorMapType

#### 6.103.1.4 ILuint DataSize

#### 6.103.1.5 ILuint Height

#### 6.103.1.6 ILuint MagicNumber

#### 6.103.1.7 ILuint Type

#### 6.103.1.8 ILuint Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_sun.c](#)

## 6.104 TARGAEXT Struct Reference

```
#include <il_targa.h>
```

### Data Fields

- [ILbyte AuthComments](#) [324]
- [ILbyte AuthName](#) [41]
- [ILshort Day](#)
- [ILshort Hour](#)
- [ILshort JobHour](#)
- [ILbyte JobID](#) [41]
- [ILshort JobMin](#)
- [ILshort JobSecs](#)
- [ILint KeyColor](#)
- [ILshort Minute](#)
- [ILshort Month](#)

- [ILshort Second](#)
- [ILshort Size](#)
- [ILbyte SoftwareID \[41\]](#)
- [ILshort SoftwareVer](#)
- [ILbyte SoftwareVerByte](#)
- [ILshort Year](#)

### 6.104.1 Field Documentation

6.104.1.1 [ILbyte AuthComments\[324\]](#)

6.104.1.2 [ILbyte AuthName\[41\]](#)

6.104.1.3 [ILshort Day](#)

6.104.1.4 [ILshort Hour](#)

6.104.1.5 [ILshort JobHour](#)

6.104.1.6 [ILbyte JobID\[41\]](#)

6.104.1.7 [ILshort JobMin](#)

6.104.1.8 [ILshort JobSecs](#)

6.104.1.9 [ILint KeyColor](#)

6.104.1.10 [ILshort Minute](#)

6.104.1.11 [ILshort Month](#)

6.104.1.12 [ILshort Second](#)

6.104.1.13 [ILshort Size](#)

6.104.1.14 [ILbyte SoftwareID\[41\]](#)

6.104.1.15 [ILshort SoftwareVer](#)

6.104.1.16 [ILbyte SoftwareVerByte](#)

6.104.1.17 [ILshort Year](#)

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_targa.h](#)

## 6.105 TARGAFOOTER Struct Reference

```
#include <il_targa.h>
```

## Data Fields

- [ILuint DevDirOff](#)
- [ILuint ExtOff](#)
- [ILbyte NullChar](#)
- [ILbyte Reserved](#)
- [ILbyte Signature](#) [16]

### 6.105.1 Field Documentation

6.105.1.1 [ILuint DevDirOff](#)

6.105.1.2 [ILuint ExtOff](#)

6.105.1.3 [ILbyte NullChar](#)

6.105.1.4 [ILbyte Reserved](#)

6.105.1.5 [ILbyte Signature](#)[16]

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_targa.h](#)

## 6.106 TARGAHEAD Struct Reference

```
#include <il_targa.h>
```

## Data Fields

- [ILubyte Bpp](#)
- [ILubyte ColMapEntSize](#)
- [ILshort ColMapLen](#)
- [ILubyte ColMapPresent](#)
- [ILshort FirstEntry](#)
- [ILushort Height](#)
- [ILubyte IDLen](#)
- [ILubyte ImageDesc](#)
- [ILubyte ImageType](#)
- [ILshort OriginX](#)
- [ILshort OriginY](#)
- [ILushort Width](#)

### 6.106.1 Field Documentation

6.106.1.1 [ILubyte Bpp](#)

6.106.1.2 [ILubyte ColMapEntSize](#)

6.106.1.3 [ILshort ColMapLen](#)

6.106.1.4 [ILubyte ColMapPresent](#)

6.106.1.5 ILshort FirstEntry

6.106.1.6 ILushort Height

6.106.1.7 ILubyte IDLen

6.106.1.8 ILubyte ImageDesc

6.106.1.9 ILubyte ImageType

6.106.1.10 ILshort OriginX

6.106.1.11 ILshort OriginY

6.106.1.12 ILushort Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_targa.h](#)

## 6.107 TEX\_HEAD Struct Reference

```
#include <il_md1.h>
```

### Data Fields

- [ILuint Flags](#)
- [ILuint Height](#)
- [char Name](#) [64]
- [ILuint Offset](#)
- [ILuint Width](#)

### 6.107.1 Field Documentation

6.107.1.1 ILuint Flags

6.107.1.2 ILuint Height

6.107.1.3 [char Name](#)[64]

6.107.1.4 ILuint Offset

6.107.1.5 ILuint Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_md1.h](#)

## 6.108 TEX\_INFO Struct Reference

```
#include <il_md1.h>
```

## Data Fields

- [ILuint NumTex](#)
- [ILuint TexDataOff](#)
- [ILuint TexOff](#)

### 6.108.1 Field Documentation

#### 6.108.1.1 ILuint NumTex

#### 6.108.1.2 ILuint TexDataOff

#### 6.108.1.3 ILuint TexOff

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_md1.h](#)

## 6.109 TPLHEAD Struct Reference

## Data Fields

- [ILuint HeaderSize](#)
- [ILuint Magic](#)
- [ILuint nTextures](#)

### 6.109.1 Field Documentation

#### 6.109.1.1 ILuint HeaderSize

#### 6.109.1.2 ILuint Magic

#### 6.109.1.3 ILuint nTextures

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_tpl.c](#)

## 6.110 UTXENTRYNAME Struct Reference

```
#include <il_utx.h>
```

## Data Fields

- [ILuint Flags](#)
- `char *` [Name](#)

### 6.110.1 Field Documentation

#### 6.110.1.1 ILuint Flags

#### 6.110.1.2 char\* Name

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_utx.h](#)

## 6.111 UTXEXPORTTABLE Struct Reference

```
#include <il_utx.h>
```

### Data Fields

- [ILint Class](#)
- [ILboolean ClassImported](#)
- [ILint Group](#)
- [ILboolean GroupImported](#)
- [ILuint ObjectFlags](#)
- [ILint ObjectName](#)
- [ILint SerialOffset](#)
- [ILint SerialSize](#)
- [ILint Super](#)
- [ILboolean SuperImported](#)

### 6.111.1 Field Documentation

#### 6.111.1.1 ILint Class

#### 6.111.1.2 ILboolean ClassImported

#### 6.111.1.3 ILint Group

#### 6.111.1.4 ILboolean GroupImported

#### 6.111.1.5 ILuint ObjectFlags

#### 6.111.1.6 ILint ObjectName

#### 6.111.1.7 ILint SerialOffset

#### 6.111.1.8 ILint SerialSize

#### 6.111.1.9 ILint Super

#### 6.111.1.10 ILboolean SuperImported

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_utx.h](#)

## 6.112 UTXHEADER Struct Reference

```
#include <il_utx.h>
```

### Data Fields

- [ILuint ExportCount](#)
- [ILuint ExportOffset](#)
- [ILuint Flags](#)
- [ILuint ImportCount](#)
- [ILuint ImportOffset](#)
- [ILushort LicenseMode](#)
- [ILuint NameCount](#)
- [ILuint NameOffset](#)
- [ILuint Signature](#)
- [ILushort Version](#)

### 6.112.1 Field Documentation

6.112.1.1 [ILuint ExportCount](#)

6.112.1.2 [ILuint ExportOffset](#)

6.112.1.3 [ILuint Flags](#)

6.112.1.4 [ILuint ImportCount](#)

6.112.1.5 [ILuint ImportOffset](#)

6.112.1.6 [ILushort LicenseMode](#)

6.112.1.7 [ILuint NameCount](#)

6.112.1.8 [ILuint NameOffset](#)

6.112.1.9 [ILuint Signature](#)

6.112.1.10 [ILushort Version](#)

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_utx.h](#)

## 6.113 UTXIMPORTTABLE Struct Reference

```
#include <il_utx.h>
```

### Data Fields

- [ILint ClassName](#)
- [ILint ClassPackage](#)
- [ILint ObjectName](#)



- [ILint Package](#)
- [ILboolean PackageImported](#)

### 6.113.1 Field Documentation

6.113.1.1 [ILint ClassName](#)

6.113.1.2 [ILint ClassPackage](#)

6.113.1.3 [ILint ObjectName](#)

6.113.1.4 [ILint Package](#)

6.113.1.5 [ILboolean PackageImported](#)

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_utx.h](#)

## 6.114 UTXPALETTE Struct Reference

```
#include <il_utx.h>
```

### Data Fields

- [ILuint Count](#)
- [ILint Name](#)
- [ILubyte \\* Pal](#)

### 6.114.1 Field Documentation

6.114.1.1 [ILuint Count](#)

6.114.1.2 [ILint Name](#)

6.114.1.3 [ILubyte\\* Pal](#)

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_utx.h](#)

## 6.115 VTFHEAD Struct Reference

```
#include <il_vtf.h>
```

### Data Fields

- [ILfloat BumpmapScale](#)
- [ILushort Depth](#)
- [ILushort FirstFrame](#)

- [ILuint Flags](#)
- [ILushort Frames](#)
- [ILuint HeaderSize](#)
- [ILushort Height](#)
- [ILuint HighResImageFormat](#)
- [ILuint LowResImageFormat](#)
- [ILubyte LowResImageHeight](#)
- [ILubyte LowResImageWidth](#)
- [ILubyte MipmapCount](#)
- [ILubyte Padding0 \[4\]](#)
- [ILubyte Padding1 \[4\]](#)
- [ILfloat Reflectivity \[3\]](#)
- [ILubyte Signature \[4\]](#)
- [ILuint Version \[2\]](#)
- [ILushort Width](#)

### 6.115.1 Field Documentation

6.115.1.1 **ILfloat BumpmapScale**

6.115.1.2 **ILushort Depth**

6.115.1.3 **ILushort FirstFrame**

6.115.1.4 **ILuint Flags**

6.115.1.5 **ILushort Frames**

6.115.1.6 **ILuint HeaderSize**

6.115.1.7 **ILushort Height**

6.115.1.8 **ILuint HighResImageFormat**

6.115.1.9 **ILuint LowResImageFormat**

6.115.1.10 **ILubyte LowResImageHeight**

6.115.1.11 **ILubyte LowResImageWidth**

6.115.1.12 **ILubyte MipmapCount**

6.115.1.13 **ILubyte Padding0[4]**

6.115.1.14 **ILubyte Padding1[4]**

6.115.1.15 **ILfloat Reflectivity[3]**

6.115.1.16 **ILubyte Signature[4]**

6.115.1.17 **ILuint Version[2]**

6.115.1.18 **ILushort Width**

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_vtf.h](#)

## 6.116 WALHEAD Struct Reference

### Data Fields

- [ILbyte AnimName](#) [32]
- [ILuint Contents](#)
- [ILbyte FileName](#) [32]
- [ILuint Flags](#)
- [ILuint Height](#)
- [ILuint Offsets](#) [4]
- [ILuint Value](#)
- [ILuint Width](#)

### 6.116.1 Field Documentation

#### 6.116.1.1 ILbyte AnimName[32]

#### 6.116.1.2 ILuint Contents

#### 6.116.1.3 ILbyte FileName[32]

#### 6.116.1.4 ILuint Flags

#### 6.116.1.5 ILuint Height

#### 6.116.1.6 ILuint Offsets[4]

#### 6.116.1.7 ILuint Value

#### 6.116.1.8 ILuint Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_wal.c](#)

## 6.117 WDPDCQUANT Struct Reference

```
#include <il_wdp.h>
```

### Data Fields

- [ILubyte ChMode](#)
- [ILubyte DcQuant](#)
- [ILubyte DcQuantChan](#)
- [ILubyte DcQuantUV](#)
- [ILubyte DcQuantY](#)

### 6.117.1 Field Documentation

6.117.1.1 ILubyte ChMode

6.117.1.2 ILubyte DcQuant

6.117.1.3 ILubyte DcQuantChan

6.117.1.4 ILubyte DcQuantUV

6.117.1.5 ILubyte DcQuantY

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_wdp.h](#)

## 6.118 WDPGUID Struct Reference

```
#include <il_wdp.h>
```

### Data Fields

- [ILuint Fifth](#)
- [ILuint First](#)
- [ILuint Fourth](#)
- [ILushort Second](#)
- [ILushort Third](#)

### 6.118.1 Field Documentation

6.118.1.1 ILuint Fifth

6.118.1.2 ILuint First

6.118.1.3 ILuint Fourth

6.118.1.4 ILushort Second

6.118.1.5 ILushort Third

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_wdp.h](#)

## 6.119 WDPHEAD Struct Reference

```
#include <il_wdp.h>
```

## Data Fields

- [ILubyte Encoding](#) [2]
- [ILuint Offset](#)
- [ILubyte UniqueID](#)
- [ILubyte Version](#)

### 6.119.1 Field Documentation

#### 6.119.1.1 ILubyte Encoding[2]

#### 6.119.1.2 ILuint Offset

#### 6.119.1.3 ILubyte UniqueID

#### 6.119.1.4 ILubyte Version

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_wdp.h](#)

## 6.120 WDPIFD Struct Reference

```
#include <il_wdp.h>
```

## Data Fields

- [ILuint Count](#)
- [ILuint NextOff](#)
- [ILushort Tag](#)
- [ILushort Type](#)
- [ILuint ValOff](#)

### 6.120.1 Field Documentation

#### 6.120.1.1 ILuint Count

#### 6.120.1.2 ILuint NextOff

#### 6.120.1.3 ILushort Tag

#### 6.120.1.4 ILushort Type

#### 6.120.1.5 ILuint ValOff

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_wdp.h](#)

## 6.121 WDPIMGHEAD Struct Reference

```
#include <il_wdp.h>
```

## Data Fields

- [ILuint Codec](#)
- [ILubyte ExtraPixels](#) [3]
- [ILubyte Flags](#) [2]
- [ILubyte Format](#)
- [ILuint GDISignature](#) [2]
- [ILuint Height](#)
- [ILuint HorzTiles](#)
- [ILushort \\* TileHeight](#)
- [ILubyte \\* TileStretch](#)
- [ILushort \\* TileWidth](#)
- [ILuint VertTiles](#)
- [ILuint Width](#)

### 6.121.1 Field Documentation

#### 6.121.1.1 ILuint Codec

#### 6.121.1.2 ILubyte ExtraPixels[3]

#### 6.121.1.3 ILubyte Flags[2]

#### 6.121.1.4 ILubyte Format

#### 6.121.1.5 ILuint GDISignature[2]

#### 6.121.1.6 ILuint Height

#### 6.121.1.7 ILuint HorzTiles

#### 6.121.1.8 ILushort \* TileHeight

#### 6.121.1.9 ILubyte\* TileStretch

#### 6.121.1.10 ILushort\* TileWidth

#### 6.121.1.11 ILuint VertTiles

#### 6.121.1.12 ILuint Width

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_wdp.h](#)

## 6.122 WDPIMGPLANE Struct Reference

```
#include <il_wdp.h>
```

## Data Fields

- [ILubyte Bayer](#)
- [ILubyte Color](#)

- [ILubyte Expbias](#)
- [ILubyte Flags1](#)
- [ILubyte Flags2](#)
- [ILubyte Mantissa](#)
- [ILubyte NumChannels](#)
- [ILubyte ShiftBits](#)

### 6.122.1 Field Documentation

#### 6.122.1.1 ILubyte Bayer

#### 6.122.1.2 ILubyte Color

#### 6.122.1.3 ILubyte Expbias

#### 6.122.1.4 ILubyte Flags1

#### 6.122.1.5 ILubyte Flags2

#### 6.122.1.6 ILubyte Mantissa

#### 6.122.1.7 ILubyte NumChannels

#### 6.122.1.8 ILubyte ShiftBits

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_wdp.h](#)

## 6.123 WDPTILE Struct Reference

```
#include <il_wdp.h>
```

### Data Fields

- [ILubyte HashAndType](#)
- [ILuint StartCode](#)

### 6.123.1 Field Documentation

#### 6.123.1.1 ILubyte HashAndType

#### 6.123.1.2 ILuint StartCode

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_wdp.h](#)

## 6.124 XPMHASHENTRY Struct Reference

### Data Fields

- [ILubyte ColourName](#) [[XPM\\_MAX\\_CHAR\\_PER\\_PIXEL](#)]
- [XpmPixel ColourValue](#)
- struct [XPMHASHENTRY](#) \* [Next](#)

### 6.124.1 Field Documentation

6.124.1.1 [ILubyte ColourName](#)[[XPM\\_MAX\\_CHAR\\_PER\\_PIXEL](#)]

6.124.1.2 [XpmPixel ColourValue](#)

6.124.1.3 struct [XPMHASHENTRY](#)\* [Next](#)

The documentation for this struct was generated from the following file:

- [src/IL/formats/il\\_xpm.c](#)



## Chapter 7

# File Documentation

### 7.1 include/IL/devil\_cpp\_wrapper.hpp File Reference

```
#include <IL/ilut.h>
```

#### Data Structures

- class [iLError](#)
- class [iFilters](#)
- class [iImage](#)
- class [iState](#)
- class [iValidate](#)

### 7.2 include/IL/devil\_internal\_exports.h File Reference

```
#include <IL/il.h>  
#include <string.h>  
#include <wchar.h>
```

#### Data Structures

- struct [ILimage](#)  
*The Fundamental Image structure.*
- struct [ILpal](#)  
*Basic Palette struct.*
- struct [SIO](#)

#### Macros

- #define [assert](#)(x)
- #define [iaalloc](#)(T, n) (T\*)[icalloc](#)(sizeof(T), (n))  
*Allocate an array of type T with n elements.*
- #define [iCharStrCpy](#) strcpy
- #define [iCharStrICmp](#) strcasecmp
- #define [iCharStrLen](#) strlen

- `#define INLINE inline`
- `#define ioalloc(T) iaalloc(T, 1)`  
*Allocate one object of type T.*
- `#define iStrCat strcat`
- `#define iStrCmp strcmp`
- `#define iStrCpy strcpy`
- `#define iStrlcmp iCharStrlCmp`
- `#define iStrLen strlen`
- `#define SIOclose(io) { if ((io)->close) (io)->close((io)->handle); (io)->handle = NULL; }`
- `#define SIOeof(io) (io)->eof ((io)->handle )`
- `#define SIOgetc(io) (io)->getchar((io)->handle )`
- `#define SIOopenRO(io,f) ((io)->openReadOnly ? (io)->openReadOnly((io)->handle) : NULL)`
- `#define SIOopenWR(io,f) ((io)->openWrite ? (io)->openWrite((io)->handle) : NULL)`
- `#define SIOpad(io,n) for (ILuint i=0; i<n; i++) SIOputc((io), 0);`
- `#define SIOputc(io,c) (io)->putchar((c), (io)->handle )`
- `#define SIOputs(io,s) SIOwrite(io, s, strlen(s), 1)`
- `#define SIOread(io, p, s, n) (io)->read ((io)->handle, (p), (s), (n))`
- `#define SIOseek(io,s, w) (io)->seek ((io)->handle, (s), (w))`
- `#define SIOTell(io) (io)->tell ((io)->handle )`
- `#define SIOwrite(io, p, s, n) (io)->write ((p), (s), (n), (io)->handle)`

## Typedefs

- `typedef struct ILImage ILImage`  
*The Fundamental Image structure.*
- `typedef struct ILpal ILpal`  
*Basic Palette struct.*
- `typedef struct SIO SIO`

## Functions

- `ILAPI void *ILAPIENTRY ialloc (ILsizei Size)`
- `ILAPI void ILAPIENTRY iBindImageTemp (void)`
- `ILAPI void *ILAPIENTRY icalloc (const ILsizei Size, const ILsizei Num)`
- `ILAPI char *ILAPIENTRY iCharStrDup (const char *Str)`
- `ILAPI ILboolean ILAPIENTRY iCheckExtension (ILconst_string Arg, ILconst_string Ext)`
- `ILAPI ILImage *ILAPIENTRY iConvertImage (ILImage *Image, ILenum DestFormat, ILenum DestType)`
- `ILAPI ILpal *ILAPIENTRY iConvertPal (ILpal *Pal, ILenum DestFormat)`
- `ILAPI ILpal *ILAPIENTRY iCopyPal (ILImage *Image)`
- `ILAPI void ILAPIENTRY iFlipBuffer (ILubyte *buff, ILuint depth, ILuint line_size, ILuint line_num)`
- `ILAPI ILboolean ILAPIENTRY iFlipImage (ILImage *Image)`
- `ILAPI void ILAPIENTRY ifree (void *Ptr)`
- `ILAPI ILImage *ILAPIENTRY iGetBaseImage (void)`
- `ILAPI ILImage *ILAPIENTRY iGetCurImage (void)`
- `ILAPI ILstring ILAPIENTRY iGetExtension (ILconst_string FileName)`
- `ILAPI ILubyte *ILAPIENTRY iGetFlipped (ILImage *Image)`
- `ILAPI ILImage *ILAPIENTRY iGetImage (ILuint Image)`
- `ILAPI ILint ILAPIENTRY iGetIntegerImage (ILImage *Image, ILenum Mode)`  
*Sets Param equal to the current value of the Mode.*
- `ILAPI ILImage *ILAPIENTRY iGetMipmap (ILImage *Image, ILuint Number)`
- `ILAPI ILImage *ILAPIENTRY iGetSubImage (ILImage *Image, ILuint Number)`  
*Used for setting the current image if it is an animation.*
- `ILAPI ILboolean ILAPIENTRY iClearImage\_ (ILImage *Image)`

- `ILAPI void ILAPIENTRY ilCloseImage (ILImage *Image)`  
*Closes Image and frees all memory associated with it.*
- `ILAPI void ILAPIENTRY ilClosePal (ILpal *Palette)`  
*Closes Palette and frees all memory associated with it.*
- `ILAPI void *ILAPIENTRY ilConvertBuffer (ILuint SizeOfData, ILenum SrcFormat, ILenum DestFormat, ILenum SrcType, ILenum DestType, ILpal *SrcPal, void *Buffer)`
- `ILAPI ILImage *ILAPIENTRY ilCopyImage_ (ILImage *Src)`
- `ILAPI ILboolean ILAPIENTRY ilCopyImageAttr (ILImage *Dest, ILImage *Src)`
- `ILAPI ILubyte ILAPIENTRY ilGetBpcType (ILenum Type)`
- `ILAPI ILubyte ILAPIENTRY ilGetBppFormat (ILenum Format)`
- `ILAPI ILubyte ILAPIENTRY ilGetBppPal (ILenum PalType)`
- `ILAPI void ILAPIENTRY ilGetClear (void *Colours, ILenum Format, ILenum Type)`
- `ILAPI ILuint ILAPIENTRY ilGetCurName (void)`
- `ILAPI ILenum ILAPIENTRY ilGetFormatBpp (ILubyte Bpp)`
- `ILAPI ILenum ILAPIENTRY ilGetPalBaseType (ILenum PalType)`
- `ILAPI ILenum ILAPIENTRY ilGetTypeBpc (ILubyte Bpc)`
- `ILAPI ILboolean ILAPIENTRY ilInitImage (ILImage *Image, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, ILenum Format, ILenum Type, void *Data)`
- `ILAPI ILboolean ILAPIENTRY ilIsValidPal (ILpal *Palette)`
- `ILAPI ILImage *ILAPIENTRY ilNewImage (ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, ILubyte Bpc)`
- `ILAPI ILImage *ILAPIENTRY ilNewImageFull (ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, ILenum Format, ILenum Type, void *Data)`
- `ILAPI ILuint ILAPIENTRY ilNextPower2 (ILuint Num)`
- `ILAPI void ILAPIENTRY ilReplaceCurlImage (ILImage *Image)`
- `ILAPI ILboolean ILAPIENTRY ilResizImage (ILImage *Image, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, ILubyte Bpc)`
- `ILAPI void ILAPIENTRY ilSetCurlImage (ILImage *Image)`
- `ILAPI ILboolean ILAPIENTRY ilTexImage_ (ILImage *Image, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, ILenum Format, ILenum Type, void *Data)`
- `ILAPI ILboolean ILAPIENTRY ilTexSubImage_ (ILImage *Image, void *Data)`
- `ILAPI ILenum ILAPIENTRY ilTypeFromExt (ILconst_string FileName)`
- `ILAPI ILImage *ILAPIENTRY iluRotate3D_ (ILImage *Image, ILfloat x, ILfloat y, ILfloat z, ILfloat Angle)`
- `ILAPI ILImage *ILAPIENTRY iluRotate_ (ILImage *Image, ILfloat Angle)`  
*Rotates a bitmap any angle.*
- `ILAPI ILImage *ILAPIENTRY iluScale_ (ILImage *Image, ILuint Width, ILuint Height, ILuint Depth)`
- `ILAPI void ILAPIENTRY iMemSwap (ILubyte *, ILubyte *, const ILuint)`
- `ILAPI ILboolean ILAPIENTRY iMirrorImage (ILImage *Image)`  
*Mirrors an image over its y axis.*
- `ILAPI char *ILAPIENTRY iMultiByteFromWide (const wchar_t *Wide)`
- `ILAPI void ILAPIENTRY iResetRead (ILImage *image)`
- `ILAPI void ILAPIENTRY iResetWrite (ILImage *image)`
- `ILAPI void ILAPIENTRY iSetError (ILenum Error)`
- `ILAPI void ILAPIENTRY iSetPal (ILImage *Image, ILpal *Pal)`
- `ILAPI ILstring ILAPIENTRY iStrDup (ILconst_string Str)`  
*Glut's portability.txt says to use this...*
- `ILAPI wchar_t *ILAPIENTRY iWideFromMultiByte (const char *Multi)`
- `ILAPI char *ILAPIENTRY SIOgets (SIO *io, char *buffer, ILuint maxlen)`
- `ILAPI char *ILAPIENTRY SIOgetw (SIO *io, char *buffer, ILuint MaxLen)`

## 7.2.1 Macro Definition Documentation

7.2.1.1 `#define assert( x )`

7.2.1.2 `#define ialloc( T, n ) (T*)calloc(sizeof(T), (n))`

Allocate an array of type *T* with *n* elements.

7.2.1.3 **#define iCharStrCpy strcpy**

7.2.1.4 **#define iCharStrlCmp strcasecmp**

7.2.1.5 **#define iCharStrLen strlen**

7.2.1.6 **#define INLINE inline**

7.2.1.7 **#define ialloc( T ) iaalloc(T, 1)**

Allocate one object of type *T*.

7.2.1.8 **#define iStrCat strcat**

7.2.1.9 **#define iStrCmp strcmp**

7.2.1.10 **#define iStrCpy strcpy**

7.2.1.11 **#define iStrlcmp iCharStrlCmp**

7.2.1.12 **#define iStrLen strlen**

7.2.1.13 **#define SIOclose( io ) { if ((io)->close) (io)->close((io)->handle); (io)->handle = NULL; }**

7.2.1.14 **#define SIOeof( io ) (io)->eof ((io)->handle )**

7.2.1.15 **#define SIOgetc( io ) (io)->getchar((io)->handle )**

7.2.1.16 **#define SIOOpenRO( io, f ) ((io)->openReadOnly ? (io)->openReadOnly((io)->handle) : NULL)**

7.2.1.17 **#define SIOOpenWR( io, f ) ((io)->openWrite ? (io)->openWrite((io)->handle) : NULL)**

7.2.1.18 **#define SIOpad( io, n ) for (ILuint i=0; i<n; i++) SIOputc((io), 0);**

7.2.1.19 **#define SIOputc( io, c ) (io)->putchar((c), (io)->handle )**

7.2.1.20 **#define SIOputs( io, s ) SIOwrite(io, s, strlen(s), 1)**

7.2.1.21 **#define SIOread( io, p, s, n ) (io)->read ((io)->handle, (p), (s), (n))**

7.2.1.22 **#define SIOseek( io, s, w ) (io)->seek ((io)->handle, (s), (w))**

7.2.1.23 **#define SIOTell( io ) (io)->tell ((io)->handle )**

7.2.1.24 **#define SIOwrite( io, p, s, n ) (io)->write ((p), (s), (n), (io)->handle)**

## 7.2.2 Typedef Documentation

7.2.2.1 **typedef struct ILImage ILImage**

The Fundamental Image structure.

Every bit of information about an image is stored in this internal structure.

## 7.2.2.2 typedef struct ILpal ILpal

Basic Palette struct.

## 7.2.2.3 typedef struct SIO SIO

## 7.2.3 Function Documentation

7.2.3.1 ILAPI void\* ILAPIENTRY ialloc ( ILsizei *Size* )

7.2.3.2 ILAPI void ILAPIENTRY iBindImageTemp ( void )

7.2.3.3 ILAPI void\* ILAPIENTRY icalloc ( const ILsizei *Size*, const ILsizei *Num* )

7.2.3.4 ILAPI char\* ILAPIENTRY iCharStrDup ( const char \* *Str* )

7.2.3.5 ILAPI ILboolean ILAPIENTRY iCheckExtension ( ILconst\_string *Arg*, ILconst\_string *Ext* )

7.2.3.6 ILAPI ILimage\* ILAPIENTRY iConvertImage ( ILimage \* *Image*, IEnum *DestFormat*, IEnum *DestType* )

7.2.3.7 ILAPI ILpal\* ILAPIENTRY iConvertPal ( ILpal \* *Pal*, IEnum *DestFormat* )

7.2.3.8 ILAPI ILpal\* ILAPIENTRY iCopyPal ( ILimage \* *Image* )

7.2.3.9 ILAPI void ILAPIENTRY iFlipBuffer ( ILubyte \* *buff*, ILuint *depth*, ILuint *line\_size*, ILuint *line\_num* )

7.2.3.10 ILAPI ILboolean ILAPIENTRY iFlipImage ( ILimage \* *Image* )

7.2.3.11 ILAPI void ILAPIENTRY ifree ( void \* *Ptr* )

7.2.3.12 ILAPI ILimage\* ILAPIENTRY iGetBaseImage ( void )

7.2.3.13 ILAPI ILimage\* ILAPIENTRY iGetCurlImage ( void )

7.2.3.14 ILAPI ILstring ILAPIENTRY iGetExtension ( ILconst\_string *FileName* )

7.2.3.15 ILAPI ILubyte\* ILAPIENTRY iGetFlipped ( ILimage \* *Image* )

7.2.3.16 ILAPI ILimage\* ILAPIENTRY iGetImage ( ILuint *Image* )

7.2.3.17 ILAPI ILint ILAPIENTRY iGetIntegerImage ( ILimage \* *Image*, IEnum *Mode* )

Sets Param equal to the current value of the Mode.

7.2.3.18 ILAPI ILimage\* ILAPIENTRY iGetMipmap ( ILimage \* *Image*, ILuint *Number* )

7.2.3.19 ILAPI ILimage\* ILAPIENTRY iGetSubImage ( ILimage \* *Image*, ILuint *Number* )

Used for setting the current image if it is an animation.

7.2.3.20 ILAPI ILboolean ILAPIENTRY iClearImage\_ ( ILimage \* *Image* )

7.2.3.21 ILAPI void ILAPIENTRY iCloseImage ( ILimage \* *Image* )

Closes Image and frees all memory associated with it.

7.2.3.22 ILAPI void ILAPIENTRY ilClosePal ( ILpal \* *Palette* )

Closes Palette and frees all memory associated with it.

7.2.3.23 ILAPI void\* ILAPIENTRY ilConvertBuffer ( ILuint *SizeOfData*, IEnum *SrcFormat*, IEnum *DestFormat*, IEnum *SrcType*, IEnum *DestType*, ILpal \* *SrcPal*, void \* *Buffer* )

7.2.3.24 ILAPI IImage\* ILAPIENTRY ilCopyImage\_ ( IImage \* *Src* )

7.2.3.25 ILAPI ILboolean ILAPIENTRY ilCopyImageAttr ( IImage \* *Dest*, IImage \* *Src* )

7.2.3.26 ILAPI ILubyte ILAPIENTRY ilGetBpcType ( IEnum *Type* )

7.2.3.27 ILAPI ILubyte ILAPIENTRY ilGetBppFormat ( IEnum *Format* )

7.2.3.28 ILAPI ILubyte ILAPIENTRY ilGetBppPal ( IEnum *PalType* )

7.2.3.29 ILAPI void ILAPIENTRY ilGetClear ( void \* *Colours*, IEnum *Format*, IEnum *Type* )

7.2.3.30 ILAPI ILuint ILAPIENTRY ilGetCurName ( void )

7.2.3.31 ILAPI IEnum ILAPIENTRY ilGetFormatBpp ( ILubyte *Bpp* )

7.2.3.32 ILAPI IEnum ILAPIENTRY ilGetPalBaseType ( IEnum *PalType* )

7.2.3.33 ILAPI IEnum ILAPIENTRY ilGetTypeBpc ( ILubyte *Bpc* )

7.2.3.34 ILAPI ILboolean ILAPIENTRY ilInitImage ( IImage \* *Image*, ILuint *Width*, ILuint *Height*, ILuint *Depth*, ILubyte *Bpp*, IEnum *Format*, IEnum *Type*, void \* *Data* )

7.2.3.35 ILAPI ILboolean ILAPIENTRY ilIsValidPal ( ILpal \* *Palette* )

7.2.3.36 ILAPI IImage\* ILAPIENTRY ilNewImage ( ILuint *Width*, ILuint *Height*, ILuint *Depth*, ILubyte *Bpp*, ILubyte *Bpc* )

7.2.3.37 ILAPI IImage\* ILAPIENTRY ilNewImageFull ( ILuint *Width*, ILuint *Height*, ILuint *Depth*, ILubyte *Bpp*, IEnum *Format*, IEnum *Type*, void \* *Data* )

7.2.3.38 ILAPI ILuint ILAPIENTRY ilNextPower2 ( ILuint *Num* )

7.2.3.39 ILAPI void ILAPIENTRY ilReplaceCurlImage ( IImage \* *Image* )

7.2.3.40 ILAPI ILboolean ILAPIENTRY ilResizImage ( IImage \* *Image*, ILuint *Width*, ILuint *Height*, ILuint *Depth*, ILubyte *Bpp*, ILubyte *Bpc* )

7.2.3.41 ILAPI void ILAPIENTRY ilSetCurlImage ( IImage \* *Image* )

7.2.3.42 ILAPI ILboolean ILAPIENTRY ilTexImage\_ ( IImage \* *Image*, ILuint *Width*, ILuint *Height*, ILuint *Depth*, ILubyte *Bpp*, IEnum *Format*, IEnum *Type*, void \* *Data* )

7.2.3.43 ILAPI ILboolean ILAPIENTRY ilTexSubImage\_ ( IImage \* *Image*, void \* *Data* )

7.2.3.44 ILAPI IEnum ILAPIENTRY ilTypeFromExt ( ILconst\_string *FileName* )

7.2.3.45 ILAPI IImage\* ILAPIENTRY iluRotate3D\_ ( IImage \* *Image*, ILfloat *x*, ILfloat *y*, ILfloat *z*, ILfloat *Angle* )

7.2.3.46 ILAPI IImage\* ILAPIENTRY iluRotate\_ ( IImage \* *Image*, ILfloat *Angle* )

Rotates a bitmap any angle.

7.2.3.47 ILAPI IImage\* ILAPIENTRY iluScale\_ ( IImage \* *Image*, ILuint *Width*, ILuint *Height*, ILuint *Depth* )

7.2.3.48 ILAPI void ILAPIENTRY iMemSwap ( ILubyte \* , ILubyte \* , const ILuint )

7.2.3.49 ILAPI ILboolean ILAPIENTRY iMirrorImage ( IImage \* *Image* )

Mirrors an image over its y axis.

7.2.3.50 ILAPI char\* ILAPIENTRY iMultiByteFromWide ( const wchar\_t \* *Wide* )

7.2.3.51 ILAPI void ILAPIENTRY iResetRead ( IImage \* *image* )

7.2.3.52 ILAPI void ILAPIENTRY iResetWrite ( IImage \* *image* )

7.2.3.53 ILAPI void ILAPIENTRY iSetError ( IEnum *Error* )

7.2.3.54 ILAPI void ILAPIENTRY iSetPal ( IImage \* *Image*, ILpal \* *Pal* )

7.2.3.55 ILAPI ILstring ILAPIENTRY iStrDup ( ILconst\_string *Str* )

Glut's portability.txt says to use this...

7.2.3.56 ILAPI wchar\_t\* ILAPIENTRY iWideFromMultiByte ( const char \* *Multi* )

7.2.3.57 ILAPI char\* ILAPIENTRY SIOgets ( SIO \* *io*, char \* *buffer*, ILuint *maxlen* )

7.2.3.58 ILAPI char\* ILAPIENTRY SIOgetw ( SIO \* *io*, char \* *buffer*, ILuint *MaxLen* )

## 7.3 include/IL/il.h File Reference

```
#include <IL/config.h>
#include <stdio.h>
#include <limits.h>
```

### Macros

- #define `__il_h__`
- #define `__IL_H__`
- #define `CLAMP_DOUBLES` 1
- #define `CLAMP_FLOATS` 1
- #define `CLAMP_HALF` 1
- #define `CONST_RESTRICT` const
- #define `IL_3DC` 0x070E
- #define `IL_ACT_PAL` 0x0477
- #define `IL_ACTIVE_FACE` 0x0E00
- #define `IL_ACTIVE_IMAGE` 0x0DF4
- #define `IL_ACTIVE_LAYER` 0x0DF6
- #define `IL_ACTIVE_MIPMAP` 0x0DF5

- #define [IL\\_ALL\\_ATTRIB\\_BITS](#) 0x000FFFFF
- #define [IL\\_ALPHA](#) 0x1906
- #define [IL\\_ATI1N](#) 0x0710
- #define [IL\\_BAD\\_DIMENSIONS](#) 0x0511
- #define [IL\\_BGR](#) 0x80E0
- #define [IL\\_BGRA](#) 0x80E1
- #define [IL\\_BLIT\\_BLEND](#) 0x0636
- #define [IL\\_BLP](#) 0x044C
- Blizzard Texture Format - .blp extension.*
- #define [IL\\_BMP](#) 0x0420
- Microsoft Windows Bitmap - .bmp extension.*
- #define [IL\\_BMP\\_RLE](#) 0x0714
- #define [IL\\_BYTE](#) 0x1400
- Data types Types.*
- #define [IL\\_CHEAD](#) 0x042F
- C-Style Header - .h extension.*
- #define [IL\\_CHEAD\\_HEADER\\_STRING](#) 0x0722
- #define [IL\\_CLAMP\(x\) IL\\_LIMIT\(\(x\),0,1\)](#)
- #define [IL\\_COL\\_PAL](#) 0x0478
- #define [IL\\_COLOR\\_INDEX](#) 0x1900
- #define [IL\\_COLOUR\\_INDEX](#) 0x1900
- Data formats Formats.*
- #define [IL\\_COMPRESS\\_BIT](#) 0x00000020
- #define [IL\\_COMPRESS\\_LZO](#) 0x0703
- #define [IL\\_COMPRESS\\_MODE](#) 0x0700
- #define [IL\\_COMPRESS\\_NONE](#) 0x0701
- #define [IL\\_COMPRESS\\_RLE](#) 0x0702
- #define [IL\\_COMPRESS\\_ZLIB](#) 0x0704
- #define [IL\\_COMPRESSION\\_HINT](#) 0x0668
- #define [IL\\_CONV\\_PAL](#) 0x0630
- #define [IL\\_COULD\\_NOT\\_OPEN\\_FILE](#) 0x050A
- #define [IL\\_CUBEMAP\\_NEGATIVEX](#) 0x00000800
- #define [IL\\_CUBEMAP\\_NEGATIVEY](#) 0x00002000
- #define [IL\\_CUBEMAP\\_NEGATIVEZ](#) 0x00008000
- #define [IL\\_CUBEMAP\\_POSITIVEX](#) 0x00000400
- #define [IL\\_CUBEMAP\\_POSITIVEY](#) 0x00001000
- #define [IL\\_CUBEMAP\\_POSITIVEZ](#) 0x00004000
- #define [IL\\_CUR\\_IMAGE](#) 0x0DF7
- #define [IL\\_CUT](#) 0x0421
- Dr. Halo - .cut extension.*
- #define [IL\\_DCX](#) 0x0438
- ZSoft Multi-PCX - .dcx extension.*
- #define [IL\\_DDS](#) 0x0437
- DirectDraw Surface - .dds extension.*
- #define [IL\\_DEFAULT\\_ON\\_FAIL](#) 0x0632
- #define [IL\\_DEPRECATED](#) (D) D
- #define [IL\\_DICOM](#) 0x044A
- Digital Imaging and Communications in Medicine (DICOM) - .dcm and .dicom extensions.*
- #define [IL\\_DONT\\_CARE](#) 0x0662
- #define [IL\\_DOOM](#) 0x0422
- DooM walls - no specific extension.*
- #define [IL\\_DOOM\\_FLAT](#) 0x0423
- DooM flats - no specific extension.*



- #define `IL_DOUBLE` 0x140A
- #define `IL_DPX` 0x0450
  - Digital Picture Exchange - .dpx extension.*
- #define `IL_DXT1` 0x0706
- #define `IL_DXT1A` 0x0711
- #define `IL_DXT2` 0x0707
- #define `IL_DXT3` 0x0708
- #define `IL_DXT4` 0x0709
- #define `IL_DXT5` 0x070A
- #define `IL_DXT_NO_COMP` 0x070B
- #define `IL_DXTC_DATA_FORMAT` 0x070D
- #define `IL_DXTC_FORMAT` 0x0705
- #define `IL_EOF` -1
- #define `IL_EXIF` 0x043A
- #define `IL_EXR` 0x0442
  - OpenEXR - .exr extension.*
- #define `IL_FALSE` 0
- #define `IL_FASTEST` 0x0660
- #define `IL_FILE_ALREADY_EXISTS` 0x050C
- #define `IL_FILE_BIT` 0x00000002
- #define `IL_FILE_MODE` 0x0621
- #define `IL_FILE_OVERWRITE` 0x0602
- #define `IL_FILE_READ_ERROR` 0x0512
- #define `IL_FILE_WRITE_ERROR` 0x0512
- #define `IL_FITS` 0x0449
  - Flexible Image Transport System - .fit and .fits extensions.*
- #define `IL_FLOAT` 0x1406
- #define `IL_FORMAT_BIT` 0x00000008
- #define `IL_FORMAT_MODE` 0x0611
- #define `IL_FORMAT_NOT_SUPPORTED` 0x0503
- #define `IL_FORMAT_SET` 0x0610
- #define `IL_FORMAT_SPECIFIC_BIT` 0x00000080
- #define `IL_FTX` 0x044D
  - Heavy Metal: FAKK2 Texture - .ftx extension.*
- #define `IL_GIF` 0x0436
  - Graphics Interchange Format - .gif extension.*
- #define `IL_HALF` 0x140B
- #define `IL_HALO_PAL` 0x0476
  - Dr. Halo Palette.*
- #define `IL_HDR` 0x043F
  - Radiance High Dynamic Range - .hdr extension.*
- #define `IL_ICNS` 0x0440
  - Macintosh Icon - .icns extension.*
- #define `IL_ICO` 0x0424
  - Microsoft Windows Icons and Cursors - .ico and .cur extensions.*
- #define `IL_IFF` 0x0447
  - Interchange File Format - .iff extension.*
- #define `IL_ILBM` 0x0426
  - Amiga IFF (FORM ILBM) - .iff, .ilbm, .lbm extensions.*
- #define `IL_ILLEGAL_FILE_VALUE` 0x0507
- #define `IL_ILLEGAL_OPERATION` 0x0506
- #define `IL_IMAGE_BITS_PER_PIXEL` 0x0DE9
- #define `IL_IMAGE_BPC` 0x0DFA

- #define `IL_IMAGE_BPP` 0x0DE8
- #define `IL_IMAGE_BPP` 0x0DE8
- #define `IL_IMAGE_BYTES_PER_PIXEL` 0x0DE8
- #define `IL_IMAGE_CHANNELS` 0x0DFF
- #define `IL_IMAGE_CUBEFLAGS` 0x0DFD
- #define `IL_IMAGE_DEPTH` 0x0DE6
- #define `IL_IMAGE_DURATION` 0x0DF8
- #define `IL_IMAGE_FORMAT` 0x0DEA
- #define `IL_IMAGE_HEIGHT` 0x0DE5
- #define `IL_IMAGE_OFFX` 0x0DFB
- #define `IL_IMAGE_OFFY` 0x0DFC
- #define `IL_IMAGE_ORIGIN` 0x0DFE
- #define `IL_IMAGE_PLANESIZE` 0x0DF9
- #define `IL_IMAGE_SIZE_OF_DATA` 0x0DE7
- #define `IL_IMAGE_TYPE` 0x0DEB
- #define `IL_IMAGE_WIDTH` 0x0DE4
- #define `IL_INT` 0x1404
- #define `IL_INTERLACE_MODE` 0x063A
- #define `IL_INTERNAL_ERROR` 0x0504
- #define `IL_INVALID_CONVERSION` 0x0510
- #define `IL_INVALID_ENUM` 0x0501
- #define `IL_INVALID_EXTENSION` 0x050B
- #define `IL_INVALID_FILE_HEADER` 0x0508
- #define `IL_INVALID_PARAM` 0x0509
- #define `IL_INVALID_VALUE` 0x0505
- #define `IL_IWI` 0x044B
  - *Call of Duty Infinity Ward Image - .iwi extension.*
- #define `IL_JASC_PAL` 0x0475
  - *PaintShop Pro Palette.*
- #define `IL_JFIF` 0x0425
- #define `IL_JNG` 0x0435
- #define `IL_JP2` 0x0441
  - *Jpeg 2000 - .jp2 extension.*
- #define `IL_JPG` 0x0425
  - *JPEG - .jpg, .jpe and .jpeg extensions.*
- #define `IL_JPG_PROGRESSIVE` 0x0725
- #define `IL_JPG_QUALITY` 0x0711
- #define `IL_JPG_SAVE_FORMAT` 0x0721
- #define `IL_KEEP_DXTC_DATA` 0x070C
- #define `IL_LESS_MEM` 0x0661
- #define `IL_LIB_EXR_ERROR` 0x05E7
- #define `IL_LIB_GIF_ERROR` 0x05E1
- #define `IL_LIB_JP2_ERROR` 0x05E6
- #define `IL_LIB_JPEG_ERROR` 0x05E2
- #define `IL_LIB_MNG_ERROR` 0x05E5
- #define `IL_LIB_PNG_ERROR` 0x05E3
- #define `IL_LIB_TIFF_ERROR` 0x05E4
- #define `IL_LIF` 0x0434
  - *Homeworld Texture - .lif extension.*
- #define `IL_LIMIT(x, m, M) ((x)<(m)?(m):((x)>(M)?(M):(x)))`
- #define `IL_LOAD_EXT` 0x1F01
- #define `IL_LOADFAIL_BIT` 0x00000040
- #define `IL_LUMINANCE` 0x1909
- #define `IL_LUMINANCE_ALPHA` 0x190A

- #define [IL\\_MAX\(a, b\) \(\(\(a\) > \(b\)\) ? \(a\) : \(b\)\)](#)
- #define [IL\\_MAX\\_BYTE](#) SCHAR\_MAX
- #define [IL\\_MAX\\_INT](#) INT\_MAX
- #define [IL\\_MAX\\_QUANT\\_INDEXES](#) 0x0644
- #define [IL\\_MAX\\_QUANT\\_INDICES](#) 0x0644
- #define [IL\\_MAX\\_SHORT](#) SHRT\_MAX
- #define [IL\\_MAX\\_UNSIGNED\\_BYTE](#) UCHAR\_MAX
- #define [IL\\_MAX\\_UNSIGNED\\_INT](#) UINT\_MAX
- #define [IL\\_MAX\\_UNSIGNED\\_SHORT](#) USHRT\_MAX
- #define [IL\\_MDL](#) 0x0431

*Half-Life Model Texture - .mdl extension.*

- #define [IL\\_MEM\\_SPEED\\_HINT](#) 0x0665
- #define [IL\\_MIN\(a, b\) \(\(\(a\) < \(b\)\) ? \(a\) : \(b\)\)](#)
- #define [IL\\_MNG](#) 0x0435

*Multiple-image Network Graphics - .mng extension.*

- #define [IL\\_MP3](#) 0x0452

*MPEG-1 Audio Layer 3 - .mp3 extension.*

- #define [IL\\_NEU\\_QUANT](#) 0x0642
- #define [IL\\_NEU\\_QUANT\\_SAMPLE](#) 0x0643
- #define [IL\\_NO\\_COMPRESSION](#) 0x0667
- #define [IL\\_NO\\_ERROR](#) 0x0000
- #define [IL\\_NUM\\_FACES](#) 0x0DE1
- #define [IL\\_NUM\\_IMAGES](#) 0x0DF1
- #define [IL\\_NUM\\_LAYERS](#) 0x0DF3
- #define [IL\\_NUM\\_MIPMAPS](#) 0x0DF2
- #define [IL\\_NVIDIA\\_COMPRESS](#) 0x0670
- #define [IL\\_ORIGIN\\_BIT](#) 0x00000001
- #define [IL\\_ORIGIN\\_LOWER\\_LEFT](#) 0x0601
- #define [IL\\_ORIGIN\\_MODE](#) 0x0603
- #define [IL\\_ORIGIN\\_SET](#) 0x0600
- #define [IL\\_ORIGIN\\_UPPER\\_LEFT](#) 0x0602
- #define [IL\\_OUT\\_FORMAT\\_SAME](#) 0x050D
- #define [IL\\_OUT\\_OF\\_MEMORY](#) 0x0502
- #define [IL\\_PAL\\_BGR24](#) 0x0404
- #define [IL\\_PAL\\_BGR32](#) 0x0405
- #define [IL\\_PAL\\_BGRA32](#) 0x0406
- #define [IL\\_PAL\\_BIT](#) 0x00000004
- #define [IL\\_PAL\\_NONE](#) 0x0400
- #define [IL\\_PAL\\_RGB24](#) 0x0401
- #define [IL\\_PAL\\_RGB32](#) 0x0402
- #define [IL\\_PAL\\_RGBA32](#) 0x0403
- #define [IL\\_PALETTE\\_BASE\\_TYPE](#) 0x0DF0
- #define [IL\\_PALETTE\\_BPP](#) 0x0DEE
- #define [IL\\_PALETTE\\_NUM\\_COLS](#) 0x0DEF
- #define [IL\\_PALETTE\\_SIZE](#) 0x0DED
- #define [IL\\_PALETTE\\_TYPE](#) 0x0DEC
- #define [IL\\_PCD](#) 0x0427

*Kodak PhotoCD - .pcd extension.*

- #define [IL\\_PCD\\_PICNUM](#) 0x0723
- #define [IL\\_PCX](#) 0x0428

*ZSoft PCX - .pcx extension.*

- #define [IL\\_PIC](#) 0x0429

*PIC - .pic extension.*

- #define [IL\\_PIX](#) 0x043C

*PIX - .pix extension.*

- #define [IL\\_PLT\\_PAL](#) 0x0479
- #define [IL\\_PNG](#) 0x042A

*Portable Network Graphics - .png extension.*

- #define [IL\\_PNG\\_ALPHA\\_INDEX](#) 0x0724
- #define [IL\\_PNG\\_AUTHNAME\\_STRING](#) 0x071A
- #define [IL\\_PNG\\_DESCRIPTION\\_STRING](#) 0x071C
- #define [IL\\_PNG\\_INTERLACE](#) 0x0712
- #define [IL\\_PNG\\_TITLE\\_STRING](#) 0x071B
- #define [IL\\_PNM](#) 0x042B

*Portable Any Map - .pbm, .pgm, .ppm and .pnm extensions.*

- #define [IL\\_PSD](#) 0x0439

*Adobe PhotoShop - .psd extension.*

- #define [IL\\_PSP](#) 0x043B

*PaintShop Pro - .psp extension.*

- #define [IL\\_PXR](#) 0x043D

*Pixar - .pxr extension.*

- #define [IL\\_QUANTIZATION\\_MODE](#) 0x0640
- #define [IL\\_RAW](#) 0x0430

*Raw Image Data - any extension.*

- #define [IL\\_RGB](#) 0x1907
- #define [IL\\_RGBA](#) 0x1908
- #define [IL\\_ROT](#) 0x044E

*Homeworld 2 - Relic Texture - .rot extension.*

- #define [IL\\_RXGB](#) 0x070F
- #define [IL\\_SAVE\\_EXT](#) 0x1F02
- #define [IL\\_SAVE\\_INTERLACED](#) 0x0639
- #define [IL\\_SEEK\\_CUR](#) 1
- #define [IL\\_SEEK\\_END](#) 2
- #define [IL\\_SEEK\\_SET](#) 0
- #define [IL\\_SFMT](#) "%s"
- #define [IL\\_SGI](#) 0x042C

*Silicon Graphics - .sgi, .bw, .rgb and .rgba extensions.*

- #define [IL\\_SGI\\_RLE](#) 0x0715
- #define [IL\\_SHORT](#) 0x1402
- #define [IL\\_SPHEREMAP](#) 0x00010000
- #define [IL\\_SQUISH\\_COMPRESS](#) 0x0671
- #define [IL\\_STACK\\_OVERFLOW](#) 0x050E
- #define [IL\\_STACK\\_UNDERFLOW](#) 0x050F
- #define [IL\\_SUB\\_LAYER](#) 0x0682
- #define [IL\\_SUB\\_MIPMAP](#) 0x0681
- #define [IL\\_SUB\\_NEXT](#) 0x0680
- #define [IL\\_SUN](#) 0x0446

*Sun Raster - .sun, .ras, .rs, .im1, .im8, .im24 and .im32 extensions.*

- #define [IL\\_TEXT\(s\)](#) s
- #define [IL\\_TEXTURE](#) 0x044F

*Medieval II: Total War Texture - .texture extension.*

- #define [IL\\_TGA](#) 0x042D

*TrueVision Targa File - .tga, .vda, .icb and .vst extensions.*

- #define [IL\\_TGA\\_AUTHCOMMENT\\_STRING](#) 0x0719
- #define [IL\\_TGA\\_AUTHNAME\\_STRING](#) 0x0718
- #define [IL\\_TGA\\_CREATE\\_STAMP](#) 0x0710
- #define [IL\\_TGA\\_ID\\_STRING](#) 0x0717

- #define `IL_TGA_RLE` 0x0713
- #define `IL_TIF` 0x042E
  - Tagged Image File Format - .tif and .tiff extensions.*
- #define `IL_TIF_AUTHNAME_STRING` 0x0720
- #define `IL_TIF_DESCRIPTION_STRING` 0x071D
- #define `IL_TIF_DOCUMENTNAME_STRING` 0x071F
- #define `IL_TIF_HOSTCOMPUTER_STRING` 0x071E
- #define `IL_TPL` 0x0448
  - Gamecube Texture - .tpl extension.*
- #define `IL_TRUE` 1
- #define `IL_TYPE_BIT` 0x00000010
- #define `IL_TYPE_MODE` 0x0613
- #define `IL_TYPE_SET` 0x0612
- #define `IL_TYPE_UNKNOWN` 0x0000
- #define `IL_UNKNOWN_ERROR` 0x05FF
- #define `IL_UNSIGNED_BYTE` 0x1401
- #define `IL_UNSIGNED_INT` 0x1405
- #define `IL_UNSIGNED_SHORT` 0x1403
- #define `IL_USE_COMPRESSION` 0x0666
- #define `IL_USE_KEY_COLOR` 0x0635
- #define `IL_USE_KEY_COLOUR` 0x0635
- #define `IL_UTX` 0x0451
  - Unreal (and Unreal Tournament) Texture - .utx extension.*
- #define `IL_VARIANT_KAIL`
- #define `IL_VENDOR` 0x1F00
- #define `IL_VERSION` 183
- #define `IL_VERSION_1_8_2`
- #define `IL_VERSION_1_8_3`
- #define `IL_VERSION_NUM` 0x0DE2
- #define `IL_VTF` 0x0444
  - Valve Texture Format - .vtf extension.*
- #define `IL_VTF_COMP` 0x0726
- #define `IL_WAL` 0x0432
  - Quake 2 Texture - .wal extension.*
- #define `IL_WBMP` 0x0445
  - Wireless Bitmap - .wbmp extension.*
- #define `IL_WDP` 0x0443
  - Microsoft HD Photo - .wdp and .hdp extension.*
- #define `IL_WU_QUANT` 0x0641
- #define `IL_XPM` 0x043E
  - X Pixel Map - .xpm extension.*
- #define `ILAPI`
- #define `ILAPIENTRY`
- #define `ILchar` char
- #define `ilClearColor` `ilClearColour`
- #define `ILconst_string` `ILchar` const \*
- #define `ilKeyColor` `ilKeyColour`
- #define `ILstring` `ILchar` \*
- #define `RESTRICT`

## Typedefs

- typedef unsigned int [ILbitfield](#)
- typedef unsigned char [ILboolean](#)
- typedef signed char [ILbyte](#)
- typedef double [ILclampd](#)
- typedef float [ILclampf](#)
- typedef double [ILdouble](#)
- typedef unsigned int [ILenum](#)
- typedef float [ILfloat](#)
- typedef void \* [ILHANDLE](#)
- typedef int [ILint](#)
- typedef long long int [ILint64](#)
- typedef signed short [ILshort](#)
- typedef size\_t [ILsizei](#)
- typedef unsigned char [ILubyte](#)
- typedef unsigned int [ILuint](#)
- typedef long long unsigned int [ILuint64](#)
- typedef unsigned short [ILushort](#)
- typedef void \* [ILAPIENTRY](#) \* [mAlloc](#) ([ILsizei](#))

## Functions

- [ILAPI ILboolean ILAPIENTRY IL\\_DEPRECATED](#) ([ilCompressFunc](#)([ILenum](#) Mode))
- [ILAPI void ILAPIENTRY IL\\_DEPRECATED](#) ([ilResetMemory](#)(void))
- [ILAPI ILboolean ILAPIENTRY ilActiveFace](#) ([ILuint](#) Number)  
*Used for setting the current face if it is a cubemap.*
- [ILAPI ILboolean ILAPIENTRY ilActiveImage](#) ([ILuint](#) Number)  
*Used for setting the current image if it is an animation.*
- [ILAPI ILboolean ILAPIENTRY ilActiveLayer](#) ([ILuint](#) Number)  
*Used for setting the current layer if layers exist.*
- [ILAPI ILboolean ILAPIENTRY ilActiveMipmap](#) ([ILuint](#) Number)  
*Sets the current mipmap level.*
- [ILAPI ILboolean ILAPIENTRY ilApplyPal](#) ([ILconst\\_string](#) FileName)
- [ILAPI ILboolean ILAPIENTRY ilApplyProfile](#) ([ILstring](#) InProfile, [ILstring](#) OutProfile)
- [ILAPI void ILAPIENTRY ilBindImage](#) ([ILuint](#) Image)  
*Makes Image the current active image - similar to [glBindTexture\(\)](#).*
- [ILAPI ILboolean ILAPIENTRY ilBlit](#) ([ILuint](#) Source, [ILint](#) DestX, [ILint](#) DestY, [ILint](#) DestZ, [ILuint](#) SrcX, [ILuint](#) SrcY, [ILuint](#) SrcZ, [ILuint](#) Width, [ILuint](#) Height, [ILuint](#) Depth)
- typedef [ILboolean](#) ([ILAPIENTRY](#) \*[fEofProc](#))([ILHANDLE](#))
- [ILAPI ILboolean ILAPIENTRY ilClampNTSC](#) (void)  
*Clamps data values of unsigned bytes from 16 to 235 for display on an.*
- [ILAPI void ILAPIENTRY ilClearColour](#) ([ILclampf](#) Red, [ILclampf](#) Green, [ILclampf](#) Blue, [ILclampf](#) Alpha)
- [ILAPI ILboolean ILAPIENTRY ilClearImage](#) (void)  
*Clears the current bound image to the values specified in [ilClearColour](#).*
- [ILAPI ILuint ILAPIENTRY ilCloneCurlImage](#) (void)  
*Creates a duplicate of the currently bound image.*
- [ILAPI ILubyte \\*ILAPIENTRY ilCompressDXT](#) ([ILubyte](#) \*Data, [ILuint](#) Width, [ILuint](#) Height, [ILuint](#) Depth, [ILenum](#) DXTCFormat, [ILuint](#) \*DXTCSize)  
*Compresses data to a DXT format using different methods.*
- [ILAPI ILboolean ILAPIENTRY ilConvertImage](#) ([ILenum](#) DestFormat, [ILenum](#) DestType)  
*Converts the current image to the DestFormat format.*
- [ILAPI ILboolean ILAPIENTRY ilConvertPal](#) ([ILenum](#) DestFormat)

- Converts the current image to the DestFormat format.*
- [ILAPI ILboolean ILAPIENTRY ilCopyImage \(ILuint Src\)](#)  
*Copies everything from Src to the current bound image.*
- [ILAPI ILuint ILAPIENTRY ilCopyPixels \(ILuint XOff, ILuint YOff, ILuint ZOff, ILuint Width, ILuint Height, ILuint Depth, ILenum Format, ILenum Type, void \\*Data\)](#)  
*Copy the pixels of a region of the currently bound image to a buffer.*
- [ILAPI ILuint ILAPIENTRY ilCreateSubImage \(ILenum Type, ILuint Num\)](#)  
*Creates sub images of the given type for the currently bound image.*
- [ILAPI ILboolean ILAPIENTRY ilDefaultImage \(void\)](#)  
*Creates an ugly 64x64 black and yellow checkerboard image.*
- [ILAPI void ILAPIENTRY ilDeleteImage \(const ILuint Num\)](#)
- [ILAPI void ILAPIENTRY ilDeleteImages \(ILsizei Num, const ILuint \\*Images\)](#)  
*Deletes Num images from the image stack - similar to glDeleteTextures().*
- [ILAPI ILuint ILAPIENTRY ilDetermineSize \(ILenum Type\)](#)  
*Returns the size of the memory buffer needed to save the current image into this Type.*
- [ILAPI ILenum ILAPIENTRY ilDetermineType \(ILconst\\_string FileName\)](#)
- [ILAPI ILenum ILAPIENTRY ilDetermineTypeF \(ILHANDLE File\)](#)
- [ILAPI ILenum ILAPIENTRY ilDetermineTypeFuncs \(\)](#)
- [ILAPI ILenum ILAPIENTRY ilDetermineTypeL \(const void \\*Lump, ILuint Size\)](#)
- [ILAPI ILboolean ILAPIENTRY ilDisable \(ILenum Mode\)](#)  
*Disables a mode.*
- [ILAPI ILboolean ILAPIENTRY ilDxtcDataToImage \(void\)](#)
- [ILAPI ILboolean ILAPIENTRY ilDxtcDataToSurface \(void\)](#)
- [ILAPI ILboolean ILAPIENTRY ilEnable \(ILenum Mode\)](#)  
*Enables a mode.*
- [typedef ILenum \(ILAPIENTRY \\*IL\\_LOADPROC\)\(ILconst\\_string\)](#)
- [ILAPI void ILAPIENTRY ilFlipSurfaceDxtcData \(void\)](#)
- [ILAPI ILboolean ILAPIENTRY ilFormatFunc \(ILenum Mode\)](#)  
*Set the default image format to use.*
- [ILAPI ILuint ILAPIENTRY ilGenImage \(void\)](#)
- [ILAPI void ILAPIENTRY ilGenImages \(ILsizei Num, ILuint \\*Images\)](#)  
*Creates Num images and puts their index in Images - similar to glGenTextures().*
- [ILAPI ILubyte \\*ILAPIENTRY ilGetAlpha \(ILenum Type\)](#)
- [ILAPI ILboolean ILAPIENTRY ilGetBoolean \(ILenum Mode\)](#)  
*Returns the current value of the Mode.*
- [ILAPI void ILAPIENTRY ilGetBooleanv \(ILenum Mode, ILboolean \\*Param\)](#)  
*Sets Param equal to the current value of the Mode.*
- [ILAPI ILubyte \\*ILAPIENTRY ilGetData \(void\)](#)  
*Returns a pointer to the current image's data.*
- [ILAPI ILuint ILAPIENTRY ilGetDXTCData \(void \\*Buffer, ILuint BufferSize, ILenum DXTCFormat\)](#)
- [ILAPI ILenum ILAPIENTRY ilGetError \(void\)](#)  
*Gets the last error on the error stack.*
- [ILAPI ILint ILAPIENTRY ilGetInteger \(ILenum Mode\)](#)  
*Returns the current value of the Mode.*
- [ILAPI ILint ILAPIENTRY ilGetIntegerImage \(ILuint Image, ILenum Mode\)](#)  
*Get a value about a specific image.*
- [ILAPI void ILAPIENTRY ilGetIntegerv \(ILenum Mode, ILint \\*Param\)](#)  
*Sets Param equal to the current value of the Mode.*
- [ILAPI ILuint64 ILAPIENTRY ilGetLumpPos \(void\)](#)
- [ILAPI ILubyte \\*ILAPIENTRY ilGetPalette \(void\)](#)  
*Returns a pointer to the current image's palette data.*
- [ILAPI ILconst\\_string ILAPIENTRY ilGetString \(ILenum StringName\)](#)

- Returns a constant string detailing aspects about this library.*
- typedef ILHANDLE (ILAPIENTRY \*fOpenProc)(ILconst\_string)
- ILAPI void ILAPIENTRY ilHint (ILenum Target, ILenum Mode)
- Specifies implementation-dependent performance hints.*
- ILAPI ILboolean ILAPIENTRY ilImageToDxtcData (ILenum Format)
- ILAPI void ILAPIENTRY ilInit (void)
- Initialize the image library.*
- typedef ILint (ILAPIENTRY \*fGetcProc)(ILHANDLE)
- ILAPI ILboolean ILAPIENTRY ilInvertSurfaceDxtcDataAlpha (void)
- ILAPI ILboolean ILAPIENTRY ilIsDisabled (ILenum Mode)
- Checks whether a Mode is not enabled.*
- ILAPI ILboolean ILAPIENTRY ilIsEnabled (ILenum Mode)
- Checks whether a Mode is enabled.*
- ILAPI ILboolean ILAPIENTRY ilIsImage (ILuint Image)
- Checks whether a given Image name is valid.*
- ILAPI ILboolean ILAPIENTRY ilIsValid (ILenum Type, ILconst\_string FileName)
- ILAPI ILboolean ILAPIENTRY ilIsValidF (ILenum Type, ILHANDLE File)
- ILAPI ILboolean ILAPIENTRY ilIsValidL (ILenum Type, void \*Lump, ILuint Size)
- ILAPI void ILAPIENTRY ilKeyColour (ILclampf Red, ILclampf Green, ILclampf Blue, ILclampf Alpha)
- ILAPI ILboolean ILAPIENTRY ilLoad (ILenum Type, ILconst\_string FileName)
- Attempts to load an image from a file. The file format is specified by the user.*
- ILAPI ILboolean ILAPIENTRY ilLoadData (ILconst\_string FileName, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp)
- Reads a raw data file.*
- ILAPI ILboolean ILAPIENTRY ilLoadDataF (ILHANDLE File, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp)
- Reads an already-opened raw data file.*
- ILAPI ILboolean ILAPIENTRY ilLoadDataL (void \*Lump, ILuint Size, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp)
- Reads from a raw data memory "lump".*
- ILAPI ILboolean ILAPIENTRY ilLoadF (ILenum Type, ILHANDLE File)
- Attempts to load an image from a file stream. The file format is specified by the user.*
- ILAPI ILboolean ILAPIENTRY ilLoadFuncs (ILenum Type)
- Attempts to load an image using the currently set IO functions. The file format is specified by the user.*
- ILAPI ILboolean ILAPIENTRY ilLoadImage (ILconst\_string FileName)
- Attempts to load an image from a file with various different methods before failing - very generic.*
- ILAPI ILboolean ILAPIENTRY ilLoadL (ILenum Type, const void \*Lump, ILuint Size)
- Attempts to load an image from a memory buffer. The file format is specified by the user.*
- ILAPI ILboolean ILAPIENTRY ilLoadPal (ILconst\_string FileName)
- Loads a palette from FileName into the current image's palette.*
- ILAPI void ILAPIENTRY ilModAlpha (ILdouble AlphaValue)
- ILAPI ILboolean ILAPIENTRY ilOriginFunc (ILenum Mode)
- Sets the default origin to be used.*
- ILAPI ILboolean ILAPIENTRY ilOverlayImage (ILuint Source, ILint XCoord, ILint YCoord, ILint ZCoord)
- Overlays the image found in Src on top of the current bound image at the coords specified.*
- ILAPI void ILAPIENTRY ilPopAttrib (void)
- Pops the last entry off the state stack into the current states.*
- ILAPI void ILAPIENTRY ilPushAttrib (ILuint Bits)
- Pushes the states indicated by Bits onto the state stack.*
- ILAPI void ILAPIENTRY ilRegisterFormat (ILenum Format)
- ILAPI ILboolean ILAPIENTRY ilRegisterLoad (ILconst\_string Ext, IL\_LOADPROC Load)
- ILAPI ILboolean ILAPIENTRY ilRegisterMipNum (ILuint Num)



- `ILAPI ILboolean ILAPIENTRY ilRegisterNumFaces (ILuint Num)`
- `ILAPI ILboolean ILAPIENTRY ilRegisterNumImages (ILuint Num)`
- `ILAPI void ILAPIENTRY ilRegisterOrigin (ILenum Origin)`
- `ILAPI void ILAPIENTRY ilRegisterPal (void *Pal, ILuint Size, ILenum Type)`
- `ILAPI ILboolean ILAPIENTRY ilRegisterSave (ILconst_string Ext, IL_SAVEPROC Save)`
- `ILAPI void ILAPIENTRY ilRegisterType (ILenum Type)`
- `ILAPI ILboolean ILAPIENTRY ilRemoveLoad (ILconst_string Ext)`  
*Unregisters a load extension - doesn't have to be called.*
- `ILAPI ILboolean ILAPIENTRY ilRemoveSave (ILconst_string Ext)`  
*Unregisters a save extension - doesn't have to be called.*
- `ILAPI void ILAPIENTRY ilResetRead (void)`
- `ILAPI void ILAPIENTRY ilResetWrite (void)`
- `ILAPI ILboolean ILAPIENTRY ilSave (ILenum Type, ILconst_string FileName)`  
*Attempts to save an image to a file. The file format is specified by the user.*
- `ILAPI ILboolean ILAPIENTRY ilSaveData (ILconst_string FileName)`  
*Save the current image to FileName as raw data.*
- `ILAPI ILuint ILAPIENTRY ilSaveF (ILenum Type, ILHANDLE File)`  
*Attempts to save an image to a file stream. The file format is specified by the user.*
- `ILAPI ILboolean ILAPIENTRY ilSaveFuncs (ILenum type)`
- `ILAPI ILboolean ILAPIENTRY ilSaveImage (ILconst_string FileName)`  
*Saves the current image based on the extension given in FileName.*
- `ILAPI ILuint ILAPIENTRY ilSaveL (ILenum Type, void *Lump, ILuint Size)`  
*Attempts to save an image to a memory buffer. The file format is specified by the user.*
- `ILAPI ILboolean ILAPIENTRY ilSavePal (ILconst_string FileName)`
- `ILAPI ILboolean ILAPIENTRY ilSetAlpha (ILdouble AlphaValue)`
- `ILAPI ILboolean ILAPIENTRY ilSetData (void *Data)`  
*Uploads Data of the same size to replace the current image's data.*
- `ILAPI ILboolean ILAPIENTRY ilSetDuration (ILuint Duration)`
- `ILAPI void ILAPIENTRY ilSetInteger (ILenum Mode, ILint Param)`  
*Sets a parameter value for a Mode.*
- `ILAPI void ILAPIENTRY ilSetMemory (mAlloc, mFree)`  
*Sets the memory allocation and deallocation functions.*
- `ILAPI void ILAPIENTRY ilSetPixels (ILint XOff, ILint YOff, ILint ZOff, ILuint Width, ILuint Height, ILuint Depth, ILenum Format, ILenum Type, void *Data)`
- `ILAPI ILboolean ILAPIENTRY ilSetRead (fOpenProc, fCloseProc, fEofProc, fGetcProc, fReadProc, fSeekProc, fTellProc)`  
*Allows you to override the default file-reading functions.*
- `ILAPI void ILAPIENTRY ilSetString (ILenum Mode, const char *String)`  
*Sets a string detailing aspects about this library.*
- `ILAPI ILboolean ILAPIENTRY ilSetWrite (fOpenProc, fCloseProc, fPutcProc, fSeekProc, fTellProc, fWriteProc)`  
*Allows you to override the default file-writing functions.*
- `ILAPI void ILAPIENTRY ilShutDown (void)`  
*Shuts down the image library.*
- `ILAPI ILboolean ILAPIENTRY ilSurfaceToDxtcData (ILenum Format)`
- `ILAPI ILboolean ILAPIENTRY ilTexImage (ILuint Width, ILuint Height, ILuint Depth, ILubyte NumChannels, ILenum Format, ILenum Type, void *Data)`  
*Changes the current bound image to use these new dimensions (current data is destroyed).*
- `ILAPI ILboolean ILAPIENTRY ilTexImageDxtc (ILint w, ILint h, ILint d, ILenum DxtFormat, const ILubyte *data)`
- `ILAPI ILenum ILAPIENTRY ilTypeFromExt (ILconst_string FileName)`
- `ILAPI ILboolean ILAPIENTRY ilTypeFunc (ILenum Mode)`  
*Sets the default type to be used.*
- `typedef ILuint (ILAPIENTRY *fReadProc)(ILHANDLE`
- `typedef void (ILAPIENTRY *fCloseProc)(ILHANDLE`

### 7.3.1 Detailed Description

The main include file for DevIL

### 7.3.2 Macro Definition Documentation

7.3.2.1 `#define __il_h_`

7.3.2.2 `#define __IL_H__`

7.3.2.3 `#define CLAMP_DOUBLES 1`

7.3.2.4 `#define CLAMP_FLOATS 1`

7.3.2.5 `#define CLAMP_HALF 1`

7.3.2.6 `#define CONST_RESTRICT const`

7.3.2.7 `#define IL_3DC 0x070E`

7.3.2.8 `#define IL_ACT_PAL 0x0477`

7.3.2.9 `#define IL_ACTIVE_FACE 0x0E00`

7.3.2.10 `#define IL_ACTIVE_IMAGE 0x0DF4`

7.3.2.11 `#define IL_ACTIVE_LAYER 0x0DF6`

7.3.2.12 `#define IL_ACTIVE_MIPMAP 0x0DF5`

7.3.2.13 `#define IL_ALL_ATTRIB_BITS 0x000FFFFF`

7.3.2.14 `#define IL_ALPHA 0x1906`

7.3.2.15 `#define IL_ATI1N 0x0710`

7.3.2.16 `#define IL_BAD_DIMENSIONS 0x0511`

7.3.2.17 `#define IL_BGR 0x80E0`

7.3.2.18 `#define IL_BGRA 0x80E1`

7.3.2.19 `#define IL_BLIT_BLEND 0x0636`

7.3.2.20 `#define IL_BLP 0x044C`

Blizzard Texture Format - .blp extension.

7.3.2.21 `#define IL_BMP 0x0420`

Microsoft Windows Bitmap - .bmp extension.

7.3.2.22 `#define IL_BMP_RLE 0x0714`

7.3.2.23 `#define IL_BYTE 0x1400`

Data types Types.

7.3.2.24 `#define IL_CHEAD 0x042F`

C-Style Header - .h extension.

7.3.2.25 `#define IL_CHEAD_HEADER_STRING 0x0722`

7.3.2.26 `#define IL_CLAMP( x ) IL_LIMIT((x),0,1)`

7.3.2.27 `#define IL_COL_PAL 0x0478`

7.3.2.28 `#define IL_COLOR_INDEX 0x1900`

7.3.2.29 `#define IL_COLOUR_INDEX 0x1900`

Data formats Formats.

7.3.2.30 `#define IL_COMPRESS_BIT 0x00000020`

7.3.2.31 `#define IL_COMPRESS_LZO 0x0703`

7.3.2.32 `#define IL_COMPRESS_MODE 0x0700`

7.3.2.33 `#define IL_COMPRESS_NONE 0x0701`

7.3.2.34 `#define IL_COMPRESS_RLE 0x0702`

7.3.2.35 `#define IL_COMPRESS_ZLIB 0x0704`

7.3.2.36 `#define IL_COMPRESSION_HINT 0x0668`

7.3.2.37 `#define IL_CONV_PAL 0x0630`

7.3.2.38 `#define IL_COULD_NOT_OPEN_FILE 0x050A`

7.3.2.39 `#define IL_CUBEMAP_NEGATIVEX 0x00000800`

7.3.2.40 `#define IL_CUBEMAP_NEGATIVEY 0x00002000`

7.3.2.41 `#define IL_CUBEMAP_NEGATIVEZ 0x00008000`

7.3.2.42 `#define IL_CUBEMAP_POSITIVEX 0x00000400`

7.3.2.43 `#define IL_CUBEMAP_POSITIVEY 0x00001000`

7.3.2.44 `#define IL_CUBEMAP_POSITIVEZ 0x00004000`

7.3.2.45 `#define IL_CUR_IMAGE 0x0DF7`

7.3.2.46 `#define IL_CUT 0x0421`

Dr. Halo - .cut extension.

7.3.2.47 `#define IL_DCX 0x0438`

ZSoft Multi-PCX - .dcx extension.

7.3.2.48 `#define IL_DDS 0x0437`

DirectDraw Surface - .dds extension.

7.3.2.49 `#define IL_DEFAULT_ON_FAIL 0x0632`

7.3.2.50 `#define IL_DEPRECATED (D) D`

7.3.2.51 `#define IL_DICOM 0x044A`

Digital Imaging and Communications in Medicine (DICOM) - .dcm and .dicom extensions.

7.3.2.52 `#define IL_DONT_CARE 0x0662`

7.3.2.53 `#define IL_DOOM 0x0422`

DooM walls - no specific extension.

7.3.2.54 `#define IL_DOOM_FLAT 0x0423`

DooM flats - no specific extension.

7.3.2.55 `#define IL_DOUBLE 0x140A`

7.3.2.56 `#define IL_DPX 0x0450`

Digital Picture Exchange - .dpx extension.

7.3.2.57 `#define IL_DXT1 0x0706`

7.3.2.58 `#define IL_DXT1A 0x0711`

7.3.2.59 `#define IL_DXT2 0x0707`

7.3.2.60 `#define IL_DXT3 0x0708`

7.3.2.61 `#define IL_DXT4 0x0709`

7.3.2.62 `#define IL_DXT5 0x070A`

7.3.2.63 `#define IL_DXT_NO_COMP 0x070B`

7.3.2.64 `#define IL_DXTC_DATA_FORMAT 0x070D`

7.3.2.65 `#define IL_DXTC_FORMAT 0x0705`

7.3.2.66 `#define IL_EOF -1`

7.3.2.67 `#define IL_EXIF 0x043A`

7.3.2.68 `#define IL_EXR 0x0442`

OpenEXR - .exr extension.

7.3.2.69 `#define IL_FALSE 0`

7.3.2.70 `#define IL_FASTEST 0x0660`

7.3.2.71 `#define IL_FILE_ALREADY_EXISTS 0x050C`

7.3.2.72 `#define IL_FILE_BIT 0x00000002`

7.3.2.73 `#define IL_FILE_MODE 0x0621`

7.3.2.74 `#define IL_FILE_OVERWRITE 0x0602`

7.3.2.75 `#define IL_FILE_READ_ERROR 0x0512`

7.3.2.76 `#define IL_FILE_WRITE_ERROR 0x0512`

7.3.2.77 `#define IL_FITS 0x0449`

Flexible Image Transport System - .fit and .fits extensions.

7.3.2.78 `#define IL_FLOAT 0x1406`

7.3.2.79 `#define IL_FORMAT_BIT 0x00000008`

7.3.2.80 `#define IL_FORMAT_MODE 0x0611`

7.3.2.81 `#define IL_FORMAT_NOT_SUPPORTED 0x0503`

7.3.2.82 `#define IL_FORMAT_SET 0x0610`

7.3.2.83 `#define IL_FORMAT_SPECIFIC_BIT 0x00000080`

7.3.2.84 `#define IL_FTX 0x044D`

Heavy Metal: FAKK2 Texture - .ftx extension.

7.3.2.85 `#define IL_GIF 0x0436`

Graphics Interchange Format - .gif extension.

7.3.2.86 `#define IL_HALF 0x140B`

7.3.2.87 `#define IL_HALO_PAL 0x0476`

Dr. Halo Palette.

### 7.3.2.88 #define IL\_HDR 0x043F

Radiance High Dynamic Range - .hdr extension.

### 7.3.2.89 #define IL\_ICNS 0x0440

Macintosh Icon - .icns extension.

### 7.3.2.90 #define IL\_ICO 0x0424

Microsoft Windows Icons and Cursors - .ico and .cur extensions.

### 7.3.2.91 #define IL\_IFF 0x0447

Interchange File Format - .iff extension.

### 7.3.2.92 #define IL\_ILBM 0x0426

Amiga IFF (FORM ILBM) - .iff, .ilbm, .lbm extensions.

### 7.3.2.93 #define IL\_ILLEGAL\_FILE\_VALUE 0x0507

### 7.3.2.94 #define IL\_ILLEGAL\_OPERATION 0x0506

### 7.3.2.95 #define IL\_IMAGE\_BITS\_PER\_PIXEL 0x0DE9

### 7.3.2.96 #define IL\_IMAGE\_BPC 0x0DFA

### 7.3.2.97 #define IL\_IMAGE\_BPP 0x0DE8

### 7.3.2.98 #define IL\_IMAGE\_BPP 0x0DE8

### 7.3.2.99 #define IL\_IMAGE\_BYTES\_PER\_PIXEL 0x0DE8

### 7.3.2.100 #define IL\_IMAGE\_CHANNELS 0x0DFF

### 7.3.2.101 #define IL\_IMAGE\_CUBEFLAGS 0x0DFD

### 7.3.2.102 #define IL\_IMAGE\_DEPTH 0x0DE6

### 7.3.2.103 #define IL\_IMAGE\_DURATION 0x0DF8

### 7.3.2.104 #define IL\_IMAGE\_FORMAT 0x0DEA

### 7.3.2.105 #define IL\_IMAGE\_HEIGHT 0x0DE5

### 7.3.2.106 #define IL\_IMAGE\_OFFX 0x0DFB

### 7.3.2.107 #define IL\_IMAGE\_OFFY 0x0DFC

### 7.3.2.108 #define IL\_IMAGE\_ORIGIN 0x0DFE

### 7.3.2.109 #define IL\_IMAGE\_PLANESIZE 0x0DF9

7.3.2.110 `#define IL_IMAGE_SIZE_OF_DATA 0x0DE7`

7.3.2.111 `#define IL_IMAGE_TYPE 0x0DEB`

7.3.2.112 `#define IL_IMAGE_WIDTH 0x0DE4`

7.3.2.113 `#define IL_INT 0x1404`

7.3.2.114 `#define IL_INTERLACE_MODE 0x063A`

7.3.2.115 `#define IL_INTERNAL_ERROR 0x0504`

7.3.2.116 `#define IL_INVALID_CONVERSION 0x0510`

7.3.2.117 `#define IL_INVALID_ENUM 0x0501`

7.3.2.118 `#define IL_INVALID_EXTENSION 0x050B`

7.3.2.119 `#define IL_INVALID_FILE_HEADER 0x0508`

7.3.2.120 `#define IL_INVALID_PARAM 0x0509`

7.3.2.121 `#define IL_INVALID_VALUE 0x0505`

7.3.2.122 `#define IL_IWI 0x044B`

Call of Duty Infinity Ward Image - .iwi extension.

7.3.2.123 `#define IL_JASC_PAL 0x0475`

PaintShop Pro Palette.

7.3.2.124 `#define IL_JFIF 0x0425`

7.3.2.125 `#define IL_JNG 0x0435`

7.3.2.126 `#define IL_JP2 0x0441`

Jpeg 2000 - .jp2 extension.

7.3.2.127 `#define IL_JPG 0x0425`

JPEG - .jpg, .jpe and .jpeg extensions.

7.3.2.128 `#define IL_JPG_PROGRESSIVE 0x0725`

7.3.2.129 `#define IL_JPG_QUALITY 0x0711`

7.3.2.130 `#define IL_JPG_SAVE_FORMAT 0x0721`

7.3.2.131 `#define IL_KEEP_DXTC_DATA 0x070C`

7.3.2.132 `#define IL_LESS_MEM 0x0661`

7.3.2.133 `#define IL_LIB_EXR_ERROR 0x05E7`

7.3.2.134 `#define IL_LIB_GIF_ERROR 0x05E1`

7.3.2.135 `#define IL_LIB_JP2_ERROR 0x05E6`

7.3.2.136 `#define IL_LIB_JPEG_ERROR 0x05E2`

7.3.2.137 `#define IL_LIB_MNG_ERROR 0x05E5`

7.3.2.138 `#define IL_LIB_PNG_ERROR 0x05E3`

7.3.2.139 `#define IL_LIB_TIFF_ERROR 0x05E4`

7.3.2.140 `#define IL_LIF 0x0434`

Homeworld Texture - .lif extension.

7.3.2.141 `#define IL_LIMIT( x, m, M ) ((x)<(m)?(m):((x)>(M)?(M):(x)))`

7.3.2.142 `#define IL_LOAD_EXT 0x1F01`

7.3.2.143 `#define IL_LOADFAIL_BIT 0x00000040`

7.3.2.144 `#define IL_LUMINANCE 0x1909`

7.3.2.145 `#define IL_LUMINANCE_ALPHA 0x190A`

7.3.2.146 `#define IL_MAX( a, b ) (((a) > (b)) ? (a) : (b))`

7.3.2.147 `#define IL_MAX_BYTE SCHAR_MAX`

7.3.2.148 `#define IL_MAX_INT INT_MAX`

7.3.2.149 `#define IL_MAX_QUANT_INDEXES 0x0644`

7.3.2.150 `#define IL_MAX_QUANT_INDICES 0x0644`

7.3.2.151 `#define IL_MAX_SHORT SHRT_MAX`

7.3.2.152 `#define IL_MAX_UNSIGNED_BYTE UCHAR_MAX`

7.3.2.153 `#define IL_MAX_UNSIGNED_INT UINT_MAX`

7.3.2.154 `#define IL_MAX_UNSIGNED_SHORT USHRT_MAX`

7.3.2.155 `#define IL_MDL 0x0431`

Half-Life Model Texture - .mdl extension.

7.3.2.156 `#define IL_MEM_SPEED_HINT 0x0665`

7.3.2.157 `#define IL_MIN( a, b ) (((a) < (b)) ? (a) : (b))`



7.3.2.158 `#define IL_MNG 0x0435`

Multiple-image Network Graphics - .mng extension.

7.3.2.159 `#define IL_MP3 0x0452`

MPEG-1 Audio Layer 3 - .mp3 extension.

7.3.2.160 `#define IL_NEU_QUANT 0x0642`

7.3.2.161 `#define IL_NEU_QUANT_SAMPLE 0x0643`

7.3.2.162 `#define IL_NO_COMPRESSION 0x0667`

7.3.2.163 `#define IL_NO_ERROR 0x0000`

7.3.2.164 `#define IL_NUM_FACES 0x0DE1`

7.3.2.165 `#define IL_NUM_IMAGES 0x0DF1`

7.3.2.166 `#define IL_NUM_LAYERS 0x0DF3`

7.3.2.167 `#define IL_NUM_MIPMAPS 0x0DF2`

7.3.2.168 `#define IL_NVIDIA_COMPRESS 0x0670`

7.3.2.169 `#define IL_ORIGIN_BIT 0x00000001`

7.3.2.170 `#define IL_ORIGIN_LOWER_LEFT 0x0601`

7.3.2.171 `#define IL_ORIGIN_MODE 0x0603`

7.3.2.172 `#define IL_ORIGIN_SET 0x0600`

7.3.2.173 `#define IL_ORIGIN_UPPER_LEFT 0x0602`

7.3.2.174 `#define IL_OUT_FORMAT_SAME 0x050D`

7.3.2.175 `#define IL_OUT_OF_MEMORY 0x0502`

7.3.2.176 `#define IL_PAL_BGR24 0x0404`

7.3.2.177 `#define IL_PAL_BGR32 0x0405`

7.3.2.178 `#define IL_PAL_BGRA32 0x0406`

7.3.2.179 `#define IL_PAL_BIT 0x00000004`

7.3.2.180 `#define IL_PAL_NONE 0x0400`

7.3.2.181 `#define IL_PAL_RGB24 0x0401`

7.3.2.182 `#define IL_PAL_RGB32 0x0402`

7.3.2.183 `#define IL_PAL_RGBA32 0x0403`

7.3.2.184 `#define IL_PALETTE_BASE_TYPE 0x0DF0`

7.3.2.185 `#define IL_PALETTE_BPP 0x0DEE`

7.3.2.186 `#define IL_PALETTE_NUM_COLS 0x0DEF`

7.3.2.187 `#define IL_PALETTE_SIZE 0x0DED`

7.3.2.188 `#define IL_PALETTE_TYPE 0x0DEC`

7.3.2.189 `#define IL_PCD 0x0427`

Kodak PhotoCD - .pcd extension.

7.3.2.190 `#define IL_PCD_PICNUM 0x0723`

7.3.2.191 `#define IL_PCX 0x0428`

ZSoft PCX - .pcx extension.

7.3.2.192 `#define IL_PIC 0x0429`

PIC - .pic extension.

7.3.2.193 `#define IL_PIX 0x043C`

PIX - .pix extension.

7.3.2.194 `#define IL_PLT_PAL 0x0479`

7.3.2.195 `#define IL_PNG 0x042A`

Portable Network Graphics - .png extension.

7.3.2.196 `#define IL_PNG_ALPHA_INDEX 0x0724`

7.3.2.197 `#define IL_PNG_AUTHNAME_STRING 0x071A`

7.3.2.198 `#define IL_PNG_DESCRIPTION_STRING 0x071C`

7.3.2.199 `#define IL_PNG_INTERLACE 0x0712`

7.3.2.200 `#define IL_PNG_TITLE_STRING 0x071B`

7.3.2.201 `#define IL_PNM 0x042B`

Portable Any Map - .pbm, .pgm, .ppm and .pnm extensions.

7.3.2.202 `#define IL_PSD 0x0439`

Adobe PhotoShop - .psd extension.

7.3.2.203 `#define IL_PSP 0x043B`

PaintShop Pro - .psp extension.

7.3.2.204 `#define IL_PXR 0x043D`

Pixar - .pxr extension.

7.3.2.205 `#define IL_QUANTIZATION_MODE 0x0640`

7.3.2.206 `#define IL_RAW 0x0430`

Raw Image Data - any extension.

7.3.2.207 `#define IL_RGB 0x1907`

7.3.2.208 `#define IL_RGBA 0x1908`

7.3.2.209 `#define IL_ROT 0x044E`

Homeworld 2 - Relic Texture - .rot extension.

7.3.2.210 `#define IL_RXGB 0x070F`

7.3.2.211 `#define IL_SAVE_EXT 0x1F02`

7.3.2.212 `#define IL_SAVE_INTERLACED 0x0639`

7.3.2.213 `#define IL_SEEK_CUR 1`

7.3.2.214 `#define IL_SEEK_END 2`

7.3.2.215 `#define IL_SEEK_SET 0`

7.3.2.216 `#define IL_SFMT "%s"`

7.3.2.217 `#define IL_SGI 0x042C`

Silicon Graphics - .sgi, .bw, .rgb and .rgba extensions.

7.3.2.218 `#define IL_SGI_RLE 0x0715`

7.3.2.219 `#define IL_SHORT 0x1402`

7.3.2.220 `#define IL_SPHEREMAP 0x00010000`

7.3.2.221 `#define IL_SQUISH_COMPRESS 0x0671`

7.3.2.222 `#define IL_STACK_OVERFLOW 0x050E`

7.3.2.223 `#define IL_STACK_UNDERFLOW 0x050F`

7.3.2.224 `#define IL_SUB_LAYER 0x0682`

7.3.2.225 `#define IL_SUB_MIPMAP 0x0681`

7.3.2.226 `#define IL_SUB_NEXT 0x0680`

7.3.2.227 `#define IL_SUN 0x0446`

Sun Raster - .sun, .ras, .rs, .im1, .im8, .im24 and .im32 extensions.

7.3.2.228 `#define IL_TEXT( s ) s`

7.3.2.229 `#define IL_TEXTURE 0x044F`

Medieval II: Total War Texture - .texture extension.

7.3.2.230 `#define IL_TGA 0x042D`

TrueVision Targa File - .tga, .vda, .icb and .vst extensions.

7.3.2.231 `#define IL_TGA_AUTHCOMMENT_STRING 0x0719`

7.3.2.232 `#define IL_TGA_AUTHNAME_STRING 0x0718`

7.3.2.233 `#define IL_TGA_CREATE_STAMP 0x0710`

7.3.2.234 `#define IL_TGA_ID_STRING 0x0717`

7.3.2.235 `#define IL_TGA_RLE 0x0713`

7.3.2.236 `#define IL_TIF 0x042E`

Tagged Image File Format - .tif and .tiff extensions.

7.3.2.237 `#define IL_TIF_AUTHNAME_STRING 0x0720`

7.3.2.238 `#define IL_TIF_DESCRIPTION_STRING 0x071D`

7.3.2.239 `#define IL_TIF_DOCUMENTNAME_STRING 0x071F`

7.3.2.240 `#define IL_TIF_HOSTCOMPUTER_STRING 0x071E`

7.3.2.241 `#define IL_TPL 0x0448`

Gamecube Texture - .tpl extension.

7.3.2.242 `#define IL_TRUE 1`

7.3.2.243 `#define IL_TYPE_BIT 0x00000010`

7.3.2.244 `#define IL_TYPE_MODE 0x0613`

7.3.2.245 `#define IL_TYPE_SET 0x0612`

7.3.2.246 `#define IL_TYPE_UNKNOWN 0x0000`

7.3.2.247 `#define IL_UNKNOWN_ERROR 0x05FF`

7.3.2.248 `#define IL_UNSIGNED_BYTE 0x1401`

7.3.2.249 `#define IL_UNSIGNED_INT 0x1405`

7.3.2.250 `#define IL_UNSIGNED_SHORT 0x1403`

7.3.2.251 `#define IL_USE_COMPRESSION 0x0666`

7.3.2.252 `#define IL_USE_KEY_COLOR 0x0635`

7.3.2.253 `#define IL_USE_KEY_COLOUR 0x0635`

7.3.2.254 `#define IL_UTX 0x0451`

Unreal (and Unreal Tournament) Texture - .utx extension.

7.3.2.255 `#define IL_VARIANT_KAIL`

7.3.2.256 `#define IL_VENDOR 0x1F00`

7.3.2.257 `#define IL_VERSION 183`

7.3.2.258 `#define IL_VERSION_1_8_2`

7.3.2.259 `#define IL_VERSION_1_8_3`

7.3.2.260 `#define IL_VERSION_NUM 0x0DE2`

7.3.2.261 `#define IL_VTF 0x0444`

Valve Texture Format - .vtf extension.

7.3.2.262 `#define IL_VTF_COMP 0x0726`

7.3.2.263 `#define IL_WAL 0x0432`

Quake 2 Texture - .wal extension.

7.3.2.264 `#define IL_WBMP 0x0445`

Wireless Bitmap - .wbmp extension.

7.3.2.265 `#define IL_WDP 0x0443`

Microsoft HD Photo - .wdp and .hdp extension.

7.3.2.266 `#define IL_WU_QUANT 0x0641`

7.3.2.267 `#define IL_XPM 0x043E`

X Pixel Map - .xpm extension.

7.3.2.268 `#define ILAPI`

7.3.2.269 `#define ILAPIENTRY`

7.3.2.270 `#define ILchar char`

7.3.2.271 `#define ilClearColor ilClearColour`

7.3.2.272 `#define ILconst_string ILchar const *`

7.3.2.273 `#define ilKeyColor ilKeyColour`

7.3.2.274 `#define ILstring ILchar *`

7.3.2.275 `#define RESTRICT`

### 7.3.3 Typedef Documentation

7.3.3.1 `typedef unsigned int ILbitfield`

7.3.3.2 `typedef unsigned char ILboolean`

7.3.3.3 `typedef signed char ILbyte`

7.3.3.4 `typedef double ILclampd`

7.3.3.5 `typedef float ILclampf`

7.3.3.6 `typedef double ILdouble`

7.3.3.7 `typedef ILenum`

7.3.3.8 `typedef float ILfloat`

7.3.3.9 `typedef ILHANDLE`

7.3.3.10 `typedef ILint`

7.3.3.11 `typedef long long int ILint64`

7.3.3.12 `typedef signed short ILshort`

7.3.3.13 `typedef size_t ILsizei`

7.3.3.14 `typedef unsigned char ILubyte`

7.3.3.15 `typedef ILuint`

7.3.3.16 `typedef long long unsigned int ILuint64`

7.3.3.17 `typedef unsigned short ILushort`

7.3.3.18 `typedef void* ILAPIENTRY* mAlloc(ILsizei)`

### 7.3.4 Function Documentation

7.3.4.1 ILAPI ILboolean ILAPIENTRY IL\_DEPRECATED ( ilCompressFunc(ILenum Mode) )

7.3.4.2 ILAPI void ILAPIENTRY IL\_DEPRECATED ( ilResetMemory(void) )

7.3.4.3 ILAPI ILboolean ILAPIENTRY ilActiveFace ( ILuint *Number* )

Used for setting the current face if it is a cubemap.

7.3.4.4 ILAPI ILboolean ILAPIENTRY ilActiveImage ( ILuint *Number* )

Used for setting the current image if it is an animation.

7.3.4.5 ILAPI ILboolean ILAPIENTRY ilActiveLayer ( ILuint *Number* )

Used for setting the current layer if layers exist.

7.3.4.6 ILAPI ILboolean ILAPIENTRY ilActiveMipmap ( ILuint *Number* )

Sets the current mipmap level.

7.3.4.7 ILAPI ILboolean ILAPIENTRY ilApplyPal ( ILconst\_string *FileName* )

7.3.4.8 ILAPI ILboolean ILAPIENTRY ilApplyProfile ( ILstring *InProfile*, ILstring *OutProfile* )

7.3.4.9 ILAPI ILboolean ILAPIENTRY ilBlit ( ILuint *Source*, ILint *DestX*, ILint *DestY*, ILint *DestZ*, ILuint *SrcX*, ILuint *SrcY*, ILuint *SrcZ*, ILuint *Width*, ILuint *Height*, ILuint *Depth* )

7.3.4.10 typedef ILboolean ( ILAPIENTRY \* *fEofProc* )

7.3.4.11 ILAPI ILboolean ILAPIENTRY ilClampNTSC ( void )

Clamps data values of unsigned bytes from 16 to 235 for display on an.

7.3.4.12 ILAPI void ILAPIENTRY ilClearColour ( ILclampf *Red*, ILclampf *Green*, ILclampf *Blue*, ILclampf *Alpha* )

7.3.4.13 ILAPI ILboolean ILAPIENTRY ilClearImage ( void )

Clears the current bound image to the values specified in ilClearColour.

7.3.4.14 ILAPI ILubyte\* ILAPIENTRY ilCompressDXT ( ILubyte \* *Data*, ILuint *Width*, ILuint *Height*, ILuint *Depth*, ILenum *DXTCFormat*, ILuint \* *DXTCSize* )

Compresses data to a DXT format using different methods.

7.3.4.15 ILAPI ILboolean ILAPIENTRY ilConvertImage ( ILenum *DestFormat*, ILenum *DestType* )

Converts the current image to the DestFormat format.

## Parameters

<i>DestFormat</i>	An enum of the desired output format. Any format values are accepted.
<i>DestType</i>	An enum of the desired output type. Any type values are accepted.

## Exceptions

<i>IL_ILLEGAL_OPERATION</i>	No currently bound image
<i>IL_INVALID_CONVERSION</i>	DestFormat or DestType was an invalid identifier.
<i>IL_OUT_OF_MEMORY</i>	Could not allocate enough memory.

## Returns

Boolean value of failure or success

#### 7.3.4.16 ILAPI ILboolean ILAPIENTRY ilConvertPal ( ILenum *DestFormat* )

Converts the current image to the DestFormat format.

#### 7.3.4.17 ILAPI void ILAPIENTRY ilDeleteImage ( const ILuint *Num* )

#### 7.3.4.18 ILAPI void ILAPIENTRY ilDeleteImages ( ILsizei *Num*, const ILuint \* *Images* )

Deletes Num images from the image stack - similar to glDeleteTextures().

#### 7.3.4.19 ILAPI ILuint ILAPIENTRY ilDetermineSize ( ILenum *Type* )

Returns the size of the memory buffer needed to save the current image into this Type.

#### 7.3.4.20 ILAPI ILenum ILAPIENTRY ilDetermineType ( ILconst\_string *FileName* )

#### 7.3.4.21 ILAPI ILenum ILAPIENTRY ilDetermineTypeF ( ILHANDLE *File* )

#### 7.3.4.22 ILAPI ILenum ILAPIENTRY ilDetermineTypeFuncs ( )

#### 7.3.4.23 ILAPI ILenum ILAPIENTRY ilDetermineTypeL ( const void \* *Lump*, ILuint *Size* )

#### 7.3.4.24 ILAPI ILboolean ILAPIENTRY ilDxtcDataToImage ( void )

#### 7.3.4.25 ILAPI ILboolean ILAPIENTRY ilDxtcDataToSurface ( void )

#### 7.3.4.26 typedef ILenum ( ILAPIENTRY \* *IL\_LOADPROC* ) const

#### 7.3.4.27 ILAPI void ILAPIENTRY ilFlipSurfaceDxtcData ( void )

#### 7.3.4.28 ILAPI ILuint ILAPIENTRY ilGenImage ( void )

#### 7.3.4.29 ILAPI void ILAPIENTRY ilGenImages ( ILsizei *Num*, ILuint \* *Images* )

Creates Num images and puts their index in Images - similar to glGenTextures().

#### 7.3.4.30 ILAPI ILubyte\* ILAPIENTRY ilGetAlpha ( ILenum *Type* )



**7.3.4.31 ILAPI ILubyte\* ILAPIENTRY ilGetData ( void )**

Returns a pointer to the current image's data.

The pointer to the image data returned by this function is only valid until any operations are done on the image. After any operations, this function should be called again. The pointer can be cast to other types for images that have more than one byte per channel for easier access to data.

**Exceptions**

<i>IL_ILLEGAL_OPERATION</i>	No currently bound image
-----------------------------	--------------------------

**Returns**

ILubyte pointer to image data.

**7.3.4.32 ILAPI ILuint ILAPIENTRY ilGetDXTCData ( void \* *Buffer*, ILuint *BufferSize*, ILenum *DXTCFormat* )****7.3.4.33 ILAPI ILuint64 ILAPIENTRY ilGetLumpPos ( void )****7.3.4.34 ILAPI ILubyte\* ILAPIENTRY ilGetPalette ( void )**

Returns a pointer to the current image's palette data.

The pointer to the image palette data returned by this function is only valid until any operations are done on the image. After any operations, this function should be called again.

**Exceptions**

<i>IL_ILLEGAL_OPERATION</i>	No currently bound image
-----------------------------	--------------------------

**Returns**

ILubyte pointer to image palette data.

**7.3.4.35 typedef ILHANDLE ( ILAPIENTRY \* *fOpenProc* ) const****7.3.4.36 ILAPI ILboolean ILAPIENTRY ilImageToDxtcData ( ILenum *Format* )****7.3.4.37 typedef ILint ( ILAPIENTRY \* *fGetcProc* )****7.3.4.38 ILAPI ILboolean ILAPIENTRY ilInvertSurfaceDxtcDataAlpha ( void )****7.3.4.39 ILAPI ILboolean ILAPIENTRY ilIsValid ( ILenum *Type*, ILconst\_string *FileName* )****7.3.4.40 ILAPI ILboolean ILAPIENTRY ilIsValidF ( ILenum *Type*, ILHANDLE *File* )****7.3.4.41 ILAPI ILboolean ILAPIENTRY ilIsValidL ( ILenum *Type*, void \* *Lump*, ILuint *Size* )****7.3.4.42 ILAPI void ILAPIENTRY ilKeyColour ( ILclampf *Red*, ILclampf *Green*, ILclampf *Blue*, ILclampf *Alpha* )****7.3.4.43 ILAPI ILboolean ILAPIENTRY ilLoad ( ILenum *Type*, ILconst\_string *FileName* )**

Attempts to load an image from a file. The file format is specified by the user.

## Parameters

<i>Type</i>	Format of this file. Acceptable values are IL_BLP, IL_BMP, IL_CUT, IL_DCX, IL_DDS, IL_-DICOM, IL_DOOM, IL_DOOM_FLAT, IL_DPX, IL_EXR, IL_FITS, IL_FTX, IL_GIF, IL_HDR, IL_ICO, IL_ICNS, IL_IFF, IL_IWI, IL_JP2, IL_JPG, IL_LIF, IL_MDL, IL_MNG, IL_MP3, IL_PCX, IL_PIX, IL_PNG, IL_PNM, IL_PSD, IL_PSP, IL_PXR, IL_ROT, IL_SGI, IL_SUN, IL_TEXTURE, IL_TGA, IL_TIF, IL_TPL, IL_UTX, IL_VTF, IL_WAL, IL_WBMP, IL_XPM, IL_RAW, IL_JASC_PAL and IL_TYPE_UNKNOWN. If IL_TYPE_UNKNOWN is specified, ilLoad will try to determine the type of the file and load it.
<i>FileName</i>	Ansi or Unicode string, depending on the compiled version of DevIL, that gives the filename of the file to load.

## Returns

Boolean value of failure or success. Returns IL\_FALSE if all three loading methods have been tried and failed.

#### 7.3.4.44 ILAPI ILboolean ILAPIENTRY ilLoadData ( ILconst\_string FileName, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp )

Reads a raw data file.

#### 7.3.4.45 ILAPI ILboolean ILAPIENTRY ilLoadDataF ( ILHANDLE File, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp )

Reads an already-opened raw data file.

#### 7.3.4.46 ILAPI ILboolean ILAPIENTRY ilLoadDataL ( void \* Lump, ILuint Size, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp )

Reads from a raw data memory "lump".

#### 7.3.4.47 ILAPI ILboolean ILAPIENTRY ilLoadF ( ILenum Type, ILHANDLE File )

Attempts to load an image from a file stream. The file format is specified by the user.

## Parameters

<i>Type</i>	Format of this file. Acceptable values are IL_BLP, IL_BMP, IL_CUT, IL_DCX, IL_DDS, IL_-DICOM, IL_DOOM, IL_DOOM_FLAT, IL_DPX, IL_EXR, IL_FITS, IL_FTX, IL_GIF, IL_HDR, IL_ICO, IL_ICNS, IL_IFF, IL_IWI, IL_JP2, IL_JPG, IL_LIF, IL_MDL, IL_MNG, IL_MP3, IL_PCX, IL_PIX, IL_PNG, IL_PNM, IL_PSD, IL_PSP, IL_PXR, IL_ROT, IL_SGI, IL_SUN, IL_TEXTURE, IL_TGA, IL_TIF, IL_TPL, IL_UTX, IL_VTF, IL_WAL, IL_WBMP, IL_XPM, IL_RAW, IL_JASC_PAL and IL_TYPE_UNKNOWN. If IL_TYPE_UNKNOWN is specified, ilLoadF will try to determine the type of the file and load it.
-------------	---

<i>File</i>	File stream to load from. The caller is responsible for closing the handle.
-------------	---

**Returns**

Boolean value of failure or success. Returns IL\_FALSE if loading fails.

**7.3.4.48 ILAPI ILboolean ILAPIENTRY ilLoadFuncs ( ILenum *type* )**

Attempts to load an image using the currently set IO functions. The file format is specified by the user.

**Parameters**

<i>Type</i>	Format of this file. Acceptable values are IL_BLP, IL_BMP, IL_CUT, IL_DCX, IL_DDS, IL_DICOM, IL_DOOM, IL_DOOM_FLAT, IL_DPX, IL_EXR, IL_FITS, IL_FTX, IL_GIF, IL_HDR, IL_ICO, IL_ICNS, IL_IFF, IL_IWI, IL_JP2, IL_JPG, IL_LIF, IL_MDL, IL_MNG, IL_MP3, IL_PCD, IL_PCX, IL_PIX, IL_PNG, IL_PNM, IL_PSD, IL_PSP, IL_PXR, IL_ROT, IL_SGI, IL_SUN, IL_TEXTURE, IL_TGA, IL_TIF, IL_TPL, IL_UTX, IL_VTF, IL_WAL, IL_WBMP, IL_XPM, IL_RAW, IL_JASC_PAL and IL_TYPE_UNKNOWN. If IL_TYPE_UNKNOWN is specified, ilLoadFuncs fails.
<i>File</i>	File stream to load from.

**Returns**

Boolean value of failure or success. Returns IL\_FALSE if loading fails.

**7.3.4.49 ILAPI ILboolean ILAPIENTRY ilLoadImage ( ILconst\_string *FileName* )**

Attempts to load an image from a file with various different methods before failing - very generic.

The ilLoadImage function allows a general interface to the specific internal file-loading routines. First, it finds the extension and checks to see if any user-registered functions (registered through ilRegisterLoad) match the extension. If nothing matches, it takes the extension and determines which function to call based on it. Lastly, it attempts to identify the image based on various image header verification functions, such as illValidPngF. If all this checking fails, IL\_FALSE is returned with no modification to the current bound image.

**Parameters**

<i>FileName</i>	Ansi or Unicode string, depending on the compiled version of DevIL, that gives the filename of the file to load.
-----------------	--

**Returns**

Boolean value of failure or success. Returns IL\_FALSE if all three loading methods have been tried and failed.

**7.3.4.50 ILAPI ILboolean ILAPIENTRY ilLoadL ( ILenum *Type*, const void \* *Lump*, ILuint *Size* )**

Attempts to load an image from a memory buffer. The file format is specified by the user.

**Parameters**

<i>Type</i>	Format of this file. Acceptable values are IL_BLP, IL_BMP, IL_CUT, IL_DCX, IL_DDS, IL_DICOM, IL_DOOM, IL_DOOM_FLAT, IL_DPX, IL_EXR, IL_FITS, IL_FTX, IL_GIF, IL_HDR, IL_ICO, IL_ICNS, IL_IFF, IL_IWI, IL_JP2, IL_JPG, IL_LIF, IL_MDL, IL_MNG, IL_MP3, IL_PCD, IL_PCX, IL_PIX, IL_PNG, IL_PNM, IL_PSD, IL_PSP, IL_PXR, IL_ROT, IL_SGI, IL_SUN, IL_TEXTURE, IL_TGA, IL_TIF, IL_TPL, IL_UTX, IL_VTF, IL_WAL, IL_WBMP, IL_XPM, IL_RAW, IL_JASC_PAL and IL_TYPE_UNKNOWN. If IL_TYPE_UNKNOWN is specified, ilLoadL will try to determine the type of the file and load it.
<i>Lump</i>	The buffer where the file data is located
<i>Size</i>	Size of the buffer

**Returns**

Boolean value of failure or success. Returns IL\_FALSE if loading fails.

**7.3.4.51 ILAPI ILboolean ILAPIENTRY ilLoadPal ( ILconst\_string *FileName* )**

Loads a palette from *FileName* into the current image's palette.

**7.3.4.52 ILAPI void ILAPIENTRY ilModAlpha ( ILdouble *AlphaValue* )****7.3.4.53 ILAPI ILboolean ILAPIENTRY ilOriginFunc ( ILenum *Mode* )**

Sets the default origin to be used.

**7.3.4.54 ILAPI ILboolean ILAPIENTRY ilOverlayImage ( ILuint *Source*, ILint *XCoord*, ILint *YCoord*, ILint *ZCoord* )**

Overlays the image found in *Src* on top of the current bound image at the coords specified.

TODO: move to [il\\_api.c](#)

**7.3.4.55 ILAPI void ILAPIENTRY ilPopAttrib ( void )**

Pops the last entry off the state stack into the current states.

**7.3.4.56 ILAPI void ILAPIENTRY ilPushAttrib ( ILuint *Bits* )**

Pushes the states indicated by *Bits* onto the state stack.

**7.3.4.57 ILAPI void ILAPIENTRY ilRegisterFormat ( ILenum *Format* )****7.3.4.58 ILAPI ILboolean ILAPIENTRY ilRegisterLoad ( ILconst\_string *Ext*, IL\_LOADPROC *Load* )****7.3.4.59 ILAPI ILboolean ILAPIENTRY ilRegisterMipNum ( ILuint *Num* )****7.3.4.60 ILAPI ILboolean ILAPIENTRY ilRegisterNumFaces ( ILuint *Num* )****7.3.4.61 ILAPI ILboolean ILAPIENTRY ilRegisterNumImages ( ILuint *Num* )****7.3.4.62 ILAPI void ILAPIENTRY ilRegisterOrigin ( ILenum *Origin* )****7.3.4.63 ILAPI void ILAPIENTRY ilRegisterPal ( void \* *Pal*, ILuint *Size*, ILenum *Type* )****7.3.4.64 ILAPI ILboolean ILAPIENTRY ilRegisterSave ( ILconst\_string *Ext*, IL\_SAVEPROC *Save* )****7.3.4.65 ILAPI void ILAPIENTRY ilRegisterType ( ILenum *Type* )****7.3.4.66 ILAPI ILboolean ILAPIENTRY ilRemoveLoad ( ILconst\_string *Ext* )**

Unregisters a load extension - doesn't have to be called.

**7.3.4.67 ILAPI ILboolean ILAPIENTRY ilRemoveSave ( ILconst\_string *Ext* )**

Unregisters a save extension - doesn't have to be called.

**7.3.4.68 ILAPI void ILAPIENTRY ilResetRead ( void )**

**7.3.4.69 ILAPI void ILAPIENTRY ilResetWrite ( void )**

**7.3.4.70 ILAPI ILboolean ILAPIENTRY ilSave ( ILenum type, ILconst\_string FileName )**

Attempts to save an image to a file. The file format is specified by the user.

**Parameters**

<i>Type</i>	Format of this file. Acceptable values are IL_BMP, IL_CHEAD, IL_DDS, IL_EXR, IL_HDR, IL_JP2, IL_JPG, IL_PCX, IL_PNG, IL_PNM, IL_PSD, IL_RAW, IL_SGI, IL_TGA, IL_TIF, IL_VTF, IL_WBMP and IL_JASC_PAL.
<i>FileName</i>	Ansi or Unicode string, depending on the compiled version of DevIL, that gives the filename to save to.

**Returns**

Boolean value of failure or success. Returns IL\_FALSE if saving failed.

**7.3.4.71 ILAPI ILboolean ILAPIENTRY ilSaveData ( ILconst\_string FileName )**

Save the current image to FileName as raw data.

**7.3.4.72 ILAPI ILuint ILAPIENTRY ilSaveF ( ILenum type, ILHANDLE File )**

Attempts to save an image to a file stream. The file format is specified by the user.

**Parameters**

<i>Type</i>	Format of this file. Acceptable values are IL_BMP, IL_CHEAD, IL_DDS, IL_EXR, IL_HDR, IL_JP2, IL_JPG, IL_PCX, IL_PNG, IL_PNM, IL_PSD, IL_RAW, IL_SGI, IL_TGA, IL_TIF, IL_VTF, IL_WBMP and IL_JASC_PAL.
<i>File</i>	File stream to save to.

**Returns**

Boolean value of failure or success. Returns IL\_FALSE if saving failed.

**7.3.4.73 ILAPI ILboolean ILAPIENTRY ilSaveFuncs ( ILenum type )**

**7.3.4.74 ILAPI ILboolean ILAPIENTRY ilSaveImage ( ILconst\_string FileName )**

Saves the current image based on the extension given in FileName.

**Parameters**

<i>FileName</i>	Ansi or Unicode string, depending on the compiled version of DevIL, that gives the filename to save to.
-----------------	---

**Returns**

Boolean value of failure or success. Returns IL\_FALSE if saving failed.

**7.3.4.75 ILAPI ILuint ILAPIENTRY ilSaveL ( ILenum Type, void \* Lump, ILuint Size )**

Attempts to save an image to a memory buffer. The file format is specified by the user.

## Parameters

<i>Type</i>	Format of this image file. Acceptable values are IL_BMP, IL_CHEAD, IL_DDS, IL_EXR, IL_HDR, IL_JP2, IL_JPG, IL_PCX, IL_PNG, IL_PNM, IL_PSD, IL_RAW, IL_SGI, IL_TGA, IL_TIF, IL_VTF, IL_WBMP and IL_JASC_PAL.
<i>Lump</i>	Memory buffer to save to
<i>Size</i>	Size of the memory buffer

## Returns

The number of bytes written to the lump, or 0 in case of failure

**7.3.4.76 ILAPI ILboolean ILAPIENTRY ilSavePal ( ILconst\_string FileName )**

**7.3.4.77 ILAPI ILboolean ILAPIENTRY ilSetAlpha ( ILdouble AlphaValue )**

**7.3.4.78 ILAPI ILboolean ILAPIENTRY ilSetData ( void \* Data )**

Uploads Data of the same size to replace the current image's data.

## Parameters

<i>Data</i>	New image data to update the currently bound image
-------------	--

## Exceptions

<i>IL_ILLEGAL_OPERATION</i>	No currently bound image
<i>IL_INVALID_PARAM</i>	Data was NULL.

## Returns

Boolean value of failure or success

**7.3.4.79 ILAPI ILboolean ILAPIENTRY ilSetDuration ( ILuint Duration )**

**7.3.4.80 ILAPI void ILAPIENTRY ilSetPixels ( ILint XOff, ILint YOff, ILint ZOff, ILuint Width, ILuint Height, ILuint Depth, ILenum Format, ILenum Type, void \* Data )**

**7.3.4.81 ILAPI ILboolean ILAPIENTRY ilSetRead ( fOpenProc , fCloseProc , fEofProc , fGetcProc , fReadProc , fSeekProc , fTellProc )**

Allows you to override the default file-reading functions.

**7.3.4.82 ILAPI ILboolean ILAPIENTRY ilSetWrite ( fOpenProc , fCloseProc , fPutcProc , fSeekProc , fTellProc , fWriteProc )**

Allows you to override the default file-writing functions.

**7.3.4.83 ILAPI ILboolean ILAPIENTRY ilSurfaceToDxtcData ( ILenum Format )**

**7.3.4.84 ILAPI ILboolean ILAPIENTRY ilTexImage ( ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, ILenum Format, ILenum Type, void \* Data )**

Changes the current bound image to use these new dimensions (current data is destroyed).

## Parameters

<i>Width</i>	Specifies the new image width. This cannot be 0.
<i>Height</i>	Specifies the new image height. This cannot be 0.
<i>Depth</i>	Specifies the new image depth. This cannot be 0.
<i>Bpp</i>	Number of channels (ex. 3 for RGB)
<i>Format</i>	Enum of the desired format. Any format values are accepted.
<i>Type</i>	Enum of the desired type. Any type values are accepted.
<i>Data</i>	Specifies data that should be copied to the new image. If this parameter is NULL, no data is copied, and the new image data consists of undefined values.

## Exceptions

<i>IL_ILLEGAL_OPERATION</i>	No currently bound image.
<i>IL_INVALID_PARAM</i>	One of the parameters is incorrect, such as one of the dimensions being 0.
<i>IL_OUT_OF_MEMORY</i>	Could not allocate enough memory.

## Returns

Boolean value of failure or success

**7.3.4.85** `ILAPI ILboolean ILAPIENTRY ilTexImageDxtc ( GLint w, GLint h, GLint d, IEnum DxtFormat, const ILubyte * data )`

**7.3.4.86** `ILAPI IEnum ILAPIENTRY ilTypeFromExt ( ILconst_string FileName )`

**7.3.4.87** `ILAPI ILboolean ILAPIENTRY ilTypeFunc ( IEnum Mode )`

Sets the default type to be used.

**7.3.4.88** `typedef ILuint ( ILAPIENTRY * fReadProc )`

**7.3.4.89** `typedef void ( ILAPIENTRY * fCloseProc )`

## 7.4 include/IL/ilu.h File Reference

```
#include <IL/il.h>
```

## Data Structures

- struct [ILUinfo](#)
- struct [ILUpointf](#)
- struct [ILUpointi](#)

## Macros

- #define [\\_\\_ilu\\_h\\_\\_](#)
- #define [\\_\\_ILU\\_H\\_\\_](#)
- #define [ILU\\_ARABIC](#) 0x0801
- #define [ILU\\_BILINEAR](#) 0x2603
- #define [ILU\\_CENTER](#) 0x0705
- #define [ILU\\_CONVOLUTION\\_MATRIX](#) 0x0710

- `#define ILU_DUTCH 0x0802`
- `#define ILU_ENGLISH 0x0800`
- `#define ILU_FILTER 0x2600`
- `#define ILU_FRENCH 0x0806`
- `#define ILU_GERMAN 0x0805`
- `#define ILU_ILLEGAL_OPERATION 0x0506`
- `#define ILU_INTERNAL_ERROR 0x0504`
- `#define ILU_INVALID_ENUM 0x0501`
- `#define ILU_INVALID_PARAM 0x0509`
- `#define ILU_INVALID_VALUE 0x0505`
- `#define ILU_JAPANESE 0x0803`
- `#define ILU_LINEAR 0x2602`
- `#define ILU_LOWER_LEFT 0x0701`
- `#define ILU_LOWER_RIGHT 0x0702`
- `#define ILU_NEAREST 0x2601`
- `#define ILU_OUT_OF_MEMORY 0x0502`
- `#define ILU_PLACEMENT 0x0700`
- `#define ILU_SCALE_BELL 0x2606`
- `#define ILU_SCALE_BOX 0x2604`
- `#define ILU_SCALE_BSPLINE 0x2607`
- `#define ILU_SCALE_LANCZOS3 0x2608`
- `#define ILU_SCALE_MITCHELL 0x2609`
- `#define ILU_SCALE_TRIANGLE 0x2605`
- `#define ILU_SPANISH 0x0804`
- `#define ILU_UPPER_LEFT 0x0703`
- `#define ILU_UPPER_RIGHT 0x0704`
- `#define ILU_VENDOR IL_VENDOR`
- `#define ILU_VERSION 183`
- `#define ILU_VERSION_1_7_8 1`
- `#define ILU_VERSION_1_8_3 1`
- `#define ILU_VERSION_NUM IL_VERSION_NUM`
- `#define iluColorsUsed iluColoursUsed`
- `#define iluReplaceColor iluReplaceColour`
- `#define iluScaleColor iluScaleColour`
- `#define iluSwapColors iluSwapColours`

## Typedefs

- `typedef ILUinfo ILinfo`
- `typedef ILUpointf ILpointf`
- `typedef ILUpointi ILpointi`
- `typedef struct ILUinfo ILUinfo`
- `typedef struct ILUpointf ILUpointf`
- `typedef struct ILUpointi ILUpointi`

## Functions

- `ILAPI void ILAPIENTRY IL_DEPRECATED (iluDeleteImage(ILuint Id))`
- `ILAPI ILuint ILAPIENTRY IL_DEPRECATED (iluGenImage(void))`
- `ILAPI ILboolean ILAPIENTRY iluAlienify (void)`

*Funny as hell filter that I stumbled upon accidentally.*

- `ILAPI ILboolean ILAPIENTRY iluBlurAvg (ILuint lter)`
- `ILAPI ILboolean ILAPIENTRY iluBlurGaussian (ILuint lter)`



- ILAPI ILboolean ILAPIENTRY iluBuildMipmaps (void)
- ILAPI ILuint ILAPIENTRY iluColoursUsed (void)
- ILAPI ILboolean ILAPIENTRY iluCompareImage (ILuint Comp)
- ILAPI ILboolean ILAPIENTRY iluContrast (ILfloat Contrast)
- ILAPI ILboolean ILAPIENTRY iluConvolution (ILint \*matrix, ILint scale, ILint bias)
- ILAPI ILboolean ILAPIENTRY iluCrop (ILuint XOff, ILuint YOff, ILuint ZOff, ILuint Width, ILuint Height, ILuint Depth)
- ILAPI ILboolean ILAPIENTRY iluEdgeDetectE (void)
- ILAPI ILboolean ILAPIENTRY iluEdgeDetectP (void)
- ILAPI ILboolean ILAPIENTRY iluEdgeDetectS (void)
- ILAPI ILboolean ILAPIENTRY iluEmboss (void)
- ILAPI ILboolean ILAPIENTRY iluEnlargeCanvas (ILuint Width, ILuint Height, ILuint Depth)  
*Enlarges the canvas.*
- ILAPI ILboolean ILAPIENTRY iluEnlargeImage (ILfloat XDim, ILfloat YDim, ILfloat ZDim)
- ILAPI ILboolean ILAPIENTRY iluEqualize (void)
- ILAPI ILconst\_string ILAPIENTRY iluErrorString (ILenum Error)
- ILAPI ILboolean ILAPIENTRY iluFlipImage (void)  
*Flips an image over its x axis.*
- ILAPI ILboolean ILAPIENTRY iluGammaCorrect (ILfloat Gamma)
- ILAPI void ILAPIENTRY iluGetImageInfo (ILUinfo \*Info)  
*Retrieves information about the current bound image.*
- ILAPI ILint ILAPIENTRY iluGetInteger (ILenum Mode)
- ILAPI void ILAPIENTRY iluGetIntegerv (ILenum Mode, ILint \*Param)
- ILAPI ILstring ILAPIENTRY iluGetString (ILenum StringName)
- ILAPI void ILAPIENTRY iluImageParameter (ILenum PName, ILenum Param)
- ILAPI void ILAPIENTRY iluInit (void)
- ILAPI ILboolean ILAPIENTRY iluInvertAlpha (void)  
*Inverts the alpha in the image.*
- ILAPI ILuint ILAPIENTRY iluLoadImage (ILconst\_string FileName)
- ILAPI ILboolean ILAPIENTRY iluMirror (void)  
*Mirrors an image over its y axis.*
- ILAPI ILboolean ILAPIENTRY iluNegative (void)  
*Inverts the colours in the image.*
- ILAPI ILboolean ILAPIENTRY iluNoisify (ILclampf Tolerance)
- ILAPI ILboolean ILAPIENTRY iluPixelize (ILuint PixSize)  
*Pixelizes an image.*
- ILAPI void ILAPIENTRY iluRegionfv (ILUpointf \*Points, ILuint n)
- ILAPI void ILAPIENTRY iluRegioniv (ILUpointi \*Points, ILuint n)
- ILAPI ILboolean ILAPIENTRY iluReplaceColour (ILubyte Red, ILubyte Green, ILubyte Blue, ILfloat Tolerance)
- ILAPI ILboolean ILAPIENTRY iluRotate (ILfloat Angle)
- ILAPI ILboolean ILAPIENTRY iluRotate3D (ILfloat x, ILfloat y, ILfloat z, ILfloat Angle)
- ILAPI ILboolean ILAPIENTRY iluSaturate1f (ILfloat Saturation)
- ILAPI ILboolean ILAPIENTRY iluSaturate4f (ILfloat r, ILfloat g, ILfloat b, ILfloat Saturation)
- ILAPI ILboolean ILAPIENTRY iluScale (ILuint Width, ILuint Height, ILuint Depth)
- ILAPI ILboolean ILAPIENTRY iluScaleAlpha (ILfloat scale)
- ILAPI ILboolean ILAPIENTRY iluScaleColours (ILfloat r, ILfloat g, ILfloat b)  
*Scales image colours.*
- ILAPI ILboolean ILAPIENTRY iluSetLanguage (ILenum Language)
- ILAPI ILboolean ILAPIENTRY iluSharpen (ILfloat Factor, ILuint Iter)
- ILAPI ILboolean ILAPIENTRY iluSwapColours (void)
- ILAPI ILboolean ILAPIENTRY iluWave (ILfloat Angle)

### 7.4.1 Detailed Description

The main include file for ILU

### 7.4.2 Macro Definition Documentation

7.4.2.1 `#define __ilu_h__`

7.4.2.2 `#define __ILU_H__`

7.4.2.3 `#define ILU_ARABIC 0x0801`

7.4.2.4 `#define ILU_BILINEAR 0x2603`

7.4.2.5 `#define ILU_CENTER 0x0705`

7.4.2.6 `#define ILU_CONVOLUTION_MATRIX 0x0710`

7.4.2.7 `#define ILU_DUTCH 0x0802`

7.4.2.8 `#define ILU_ENGLISH 0x0800`

7.4.2.9 `#define ILU_FILTER 0x2600`

7.4.2.10 `#define ILU_FRENCH 0x0806`

7.4.2.11 `#define ILU_GERMAN 0x0805`

7.4.2.12 `#define ILU_ILLEGAL_OPERATION 0x0506`

7.4.2.13 `#define ILU_INTERNAL_ERROR 0x0504`

7.4.2.14 `#define ILU_INVALID_ENUM 0x0501`

7.4.2.15 `#define ILU_INVALID_PARAM 0x0509`

7.4.2.16 `#define ILU_INVALID_VALUE 0x0505`

7.4.2.17 `#define ILU_JAPANESE 0x0803`

7.4.2.18 `#define ILU_LINEAR 0x2602`

7.4.2.19 `#define ILU_LOWER_LEFT 0x0701`

7.4.2.20 `#define ILU_LOWER_RIGHT 0x0702`

7.4.2.21 `#define ILU_NEAREST 0x2601`

7.4.2.22 `#define ILU_OUT_OF_MEMORY 0x0502`

7.4.2.23 `#define ILU_PLACEMENT 0x0700`

7.4.2.24 `#define ILU_SCALE_BELL 0x2606`

7.4.2.25 `#define ILU_SCALE_BOX 0x2604`

7.4.2.26 `#define ILU_SCALE_BSPLINE 0x2607`

7.4.2.27 `#define ILU_SCALE_LANCZOS3 0x2608`

7.4.2.28 `#define ILU_SCALE_MITCHELL 0x2609`

7.4.2.29 `#define ILU_SCALE_TRIANGLE 0x2605`

7.4.2.30 `#define ILU_SPANISH 0x0804`

7.4.2.31 `#define ILU_UPPER_LEFT 0x0703`

7.4.2.32 `#define ILU_UPPER_RIGHT 0x0704`

7.4.2.33 `#define ILU_VENDOR IL_VENDOR`

7.4.2.34 `#define ILU_VERSION 183`

7.4.2.35 `#define ILU_VERSION_1_7_8 1`

7.4.2.36 `#define ILU_VERSION_1_8_3 1`

7.4.2.37 `#define ILU_VERSION_NUM IL_VERSION_NUM`

7.4.2.38 `#define iluColorsUsed iluColoursUsed`

7.4.2.39 `#define iluReplaceColor iluReplaceColour`

7.4.2.40 `#define iluScaleColor iluScaleColour`

7.4.2.41 `#define iluSwapColors iluSwapColours`

### 7.4.3 Typedef Documentation

7.4.3.1 `typedef ILUinfo ILInfo`

7.4.3.2 `typedef ILUpointf ILpointf`

7.4.3.3 `typedef ILUpointi ILpointi`

7.4.3.4 `typedef struct ILUinfo ILUinfo`

7.4.3.5 `typedef struct ILUpointf ILUpointf`

7.4.3.6 `typedef struct ILUpointi ILUpointi`

### 7.4.4 Function Documentation

7.4.4.1 `ILAPI void ILAPIENTRY IL_DEPRECATED ( iluDeleteImage(ILuint Id) )`

7.4.4.2 `ILAPI ILuint ILAPIENTRY IL_DEPRECATED ( iluGenImage(void) )`

7.4.4.3 `ILAPI ILboolean ILAPIENTRY iluAlienify ( void )`

Funny as hell filter that I stumbled upon accidentally.

- 7.4.4.4 ILAPI ILboolean ILAPIENTRY iluBlurAvg ( ILuint *Iter* )
- 7.4.4.5 ILAPI ILboolean ILAPIENTRY iluBlurGaussian ( ILuint *Iter* )
- 7.4.4.6 ILAPI ILboolean ILAPIENTRY iluBuildMipmaps ( void )
- 7.4.4.7 ILAPI ILuint ILAPIENTRY iluColoursUsed ( void )
- 7.4.4.8 ILAPI ILboolean ILAPIENTRY iluCompareImage ( ILuint *Comp* )
- 7.4.4.9 ILAPI ILboolean ILAPIENTRY iluContrast ( ILfloat *Contrast* )
- 7.4.4.10 ILAPI ILboolean ILAPIENTRY iluConvolution ( ILint \* *matrix*, ILint *scale*, ILint *bias* )
- 7.4.4.11 ILAPI ILboolean ILAPIENTRY iluCrop ( ILuint *XOff*, ILuint *YOff*, ILuint *ZOff*, ILuint *Width*, ILuint *Height*, ILuint *Depth* )
- 7.4.4.12 ILAPI ILboolean ILAPIENTRY iluEdgeDetectE ( void )
- 7.4.4.13 ILAPI ILboolean ILAPIENTRY iluEdgeDetectP ( void )
- 7.4.4.14 ILAPI ILboolean ILAPIENTRY iluEdgeDetectS ( void )
- 7.4.4.15 ILAPI ILboolean ILAPIENTRY iluEmboss ( void )
- 7.4.4.16 ILAPI ILboolean ILAPIENTRY iluEnlargeCanvas ( ILuint *Width*, ILuint *Height*, ILuint *Depth* )
- Enlarges the canvas.
- 7.4.4.17 ILAPI ILboolean ILAPIENTRY iluEnlargeImage ( ILfloat *XDim*, ILfloat *YDim*, ILfloat *ZDim* )
- 7.4.4.18 ILAPI ILboolean ILAPIENTRY iluEqualize ( void )
- 7.4.4.19 ILAPI ILconst\_string ILAPIENTRY iluErrorString ( ILenum *Error* )
- 7.4.4.20 ILAPI ILboolean ILAPIENTRY iluFlipImage ( void )
- Flips an image over its x axis.
- 7.4.4.21 ILAPI ILboolean ILAPIENTRY iluGammaCorrect ( ILfloat *Gamma* )
- 7.4.4.22 ILAPI void ILAPIENTRY iluGetImageInfo ( ILUinfo \* *Info* )
- Retrieves information about the current bound image.
- 7.4.4.23 ILAPI ILint ILAPIENTRY iluGetInteger ( ILenum *Mode* )
- 7.4.4.24 ILAPI void ILAPIENTRY iluGetIntegerv ( ILenum *Mode*, ILint \* *Param* )
- 7.4.4.25 ILAPI ILstring ILAPIENTRY iluGetString ( ILenum *StringName* )
- 7.4.4.26 ILAPI void ILAPIENTRY iluImageParameter ( ILenum *PName*, ILenum *Param* )
- 7.4.4.27 ILAPI void ILAPIENTRY iluInit ( void )

**7.4.4.28 ILAPI ILboolean ILAPIENTRY iluInvertAlpha ( void )**

Inverts the alpha in the image.

**7.4.4.29 ILAPI ILuint ILAPIENTRY iluLoadImage ( ILconst\_string FileName )****7.4.4.30 ILAPI ILboolean ILAPIENTRY iluMirror ( void )**

Mirrors an image over its y axis.

**7.4.4.31 ILAPI ILboolean ILAPIENTRY iluNegative ( void )**

Inverts the colours in the image.

**7.4.4.32 ILAPI ILboolean ILAPIENTRY iluNoisify ( ILclampf Tolerance )****7.4.4.33 ILAPI ILboolean ILAPIENTRY iluPixelize ( ILuint PixSize )**

Pixelizes an image.

**7.4.4.34 ILAPI void ILAPIENTRY iluRegionfv ( ILUpointf \* Points, ILuint n )****7.4.4.35 ILAPI void ILAPIENTRY iluRegioniv ( ILUpointi \* Points, ILuint n )****7.4.4.36 ILAPI ILboolean ILAPIENTRY iluReplaceColour ( ILubyte Red, ILubyte Green, ILubyte Blue, ILfloat Tolerance )****7.4.4.37 ILAPI ILboolean ILAPIENTRY iluRotate ( ILfloat Angle )****7.4.4.38 ILAPI ILboolean ILAPIENTRY iluRotate3D ( ILfloat x, ILfloat y, ILfloat z, ILfloat Angle )****7.4.4.39 ILAPI ILboolean ILAPIENTRY iluSaturate1f ( ILfloat Saturation )****7.4.4.40 ILAPI ILboolean ILAPIENTRY iluSaturate4f ( ILfloat r, ILfloat g, ILfloat b, ILfloat Saturation )****7.4.4.41 ILAPI ILboolean ILAPIENTRY iluScale ( ILuint Width, ILuint Height, ILuint Depth )****7.4.4.42 ILAPI ILboolean ILAPIENTRY iluScaleAlpha ( ILfloat scale )****7.4.4.43 ILAPI ILboolean ILAPIENTRY iluScaleColours ( ILfloat r, ILfloat g, ILfloat b )**

Scales image colours.

**7.4.4.44 ILAPI ILboolean ILAPIENTRY iluSetLanguage ( ILenum Language )****7.4.4.45 ILAPI ILboolean ILAPIENTRY iluSharpen ( ILfloat Factor, ILuint Iter )****7.4.4.46 ILAPI ILboolean ILAPIENTRY iluSwapColours ( void )****7.4.4.47 ILAPI ILboolean ILAPIENTRY iluWave ( ILfloat Angle )**

## 7.5 include/IL/ilut.h File Reference

```
#include <IL/il.h>
#include <IL/ilu.h>
#include <IL/ilut_config.h>
```

### Macros

- `#define __ilut_h_`
- `#define __ILUT_H__`
- `#define ILUT_ALL_ATTRIB_BITS 0x000FFFFF`
- `#define ILUT_ALLEGRO 1`
- `#define ILUT_BAD_DIMENSIONS 0x0511`
- `#define ILUT_COULD_NOT_OPEN_FILE 0x050A`
- `#define ILUT_D3D_ALPHA_KEY_COLOR 0x0707`
- `#define ILUT_D3D_ALPHA_KEY_COLOUR 0x0707`
- `#define ILUT_D3D_BIT 0x00000002`
- `#define ILUT_D3D_GEN_DXTC 0x0635`
- `#define ILUT_D3D_MIPLEVELS 0x0620`
- `#define ILUT_D3D_POOL 0x0706`
- `#define ILUT_D3D_USE_DXTC 0x0634`
- `#define ILUT_DIRECT3D10 6`
- `#define ILUT_DIRECT3D8 3`
- `#define ILUT_DIRECT3D9 4`
- `#define ILUT_DXTC_FORMAT 0x0705`
- `#define ILUT_FORCE_INTEGER_FORMAT 0x0636`
- `#define ILUT_GL_AUTODETECT_TEXTURE_TARGET 0x0807`
- `#define ILUT_GL_GEN_S3TC 0x0635`
- `#define ILUT_GL_USE_S3TC 0x0634`
- `#define ILUT_ILLEGAL_OPERATION 0x0506`
- `#define ILUT_INVALID_ENUM 0x0501`
- `#define ILUT_INVALID_PARAM 0x0509`
- `#define ILUT_INVALID_VALUE 0x0505`
- `#define ILUT_MAXTEX_DEPTH 0x0632`
- `#define ILUT_MAXTEX_HEIGHT 0x0631`
- `#define ILUT_MAXTEX_WIDTH 0x0630`
- `#define ILUT_NOT_SUPPORTED 0x0550`
- `#define ILUT_OPENGL 0`
- `#define ILUT_OPENGL_BIT 0x00000001`
- `#define ILUT_OPENGL_CONV 0x0610`
- `#define ILUT_OUT_OF_MEMORY 0x0502`
- `#define ILUT_PALETTE_MODE 0x0600`
- `#define ILUT_S3TC_FORMAT 0x0705`
- `#define ILUT_STACK_OVERFLOW 0x050E`
- `#define ILUT_STACK_UNDERFLOW 0x050F`
- `#define ILUT_VENDOR_IL_VENDOR`
- `#define ILUT_VERSION 183`
- `#define ILUT_VERSION_1_7_8 1`
- `#define ILUT_VERSION_1_8_3 1`
- `#define ILUT_VERSION_NUM IL_VERSION_NUM`
- `#define ILUT_WIN32 2`
- `#define ILUT_X11 5`

## Functions

- [ILAPI ILboolean ILAPIENTRY ilutDisable \(ILenum Mode\)](#)
- [ILAPI ILboolean ILAPIENTRY ilutEnable \(ILenum Mode\)](#)
- [ILAPI ILboolean ILAPIENTRY ilutGetBoolean \(ILenum Mode\)](#)
- [ILAPI void ILAPIENTRY ilutGetBooleanv \(ILenum Mode, ILboolean \\*Param\)](#)
- [ILAPI ILint ILAPIENTRY ilutGetInteger \(ILenum Mode\)](#)
- [ILAPI void ILAPIENTRY ilutGetInterv \(ILenum Mode, ILint \\*Param\)](#)
- [ILAPI ILstring ILAPIENTRY ilutGetString \(ILenum StringName\)](#)
- [ILAPI void ILAPIENTRY ilutInit \(void\)](#)
- [ILAPI ILboolean ILAPIENTRY ilutIsDisabled \(ILenum Mode\)](#)
- [ILAPI ILboolean ILAPIENTRY ilutIsEnabled \(ILenum Mode\)](#)
- [ILAPI void ILAPIENTRY ilutPopAttrib \(void\)](#)
- [ILAPI void ILAPIENTRY ilutPushAttrib \(ILuint Bits\)](#)
- [ILAPI ILboolean ILAPIENTRY ilutRenderer \(ILenum Renderer\)](#)
- [ILAPI void ILAPIENTRY ilutSetInteger \(ILenum Mode, ILint Param\)](#)

### 7.5.1 Detailed Description

The main include file for ILUT

### 7.5.2 Macro Definition Documentation

7.5.2.1 `#define __ilut_h_`

7.5.2.2 `#define __ILUT_H__`

7.5.2.3 `#define ILUT_ALL_ATTRIB_BITS 0x000FFFFF`

7.5.2.4 `#define ILUT_ALLEGRO 1`

7.5.2.5 `#define ILUT_BAD_DIMENSIONS 0x0511`

7.5.2.6 `#define ILUT_COULD_NOT_OPEN_FILE 0x050A`

7.5.2.7 `#define ILUT_D3D_ALPHA_KEY_COLOR 0x0707`

7.5.2.8 `#define ILUT_D3D_ALPHA_KEY_COLOUR 0x0707`

7.5.2.9 `#define ILUT_D3D_BIT 0x00000002`

7.5.2.10 `#define ILUT_D3D_GEN_DXTC 0x0635`

7.5.2.11 `#define ILUT_D3D_MIPLEVELS 0x0620`

7.5.2.12 `#define ILUT_D3D_POOL 0x0706`

7.5.2.13 `#define ILUT_D3D_USE_DXTC 0x0634`

7.5.2.14 `#define ILUT_DIRECT3D10 6`

7.5.2.15 `#define ILUT_DIRECT3D8 3`

7.5.2.16 `#define ILUT_DIRECT3D9 4`

7.5.2.17 `#define ILUT_DXTC_FORMAT 0x0705`

7.5.2.18 `#define ILUT_FORCE_INTEGER_FORMAT 0x0636`

7.5.2.19 `#define ILUT_GL_AUTODETECT_TEXTURE_TARGET 0x0807`

7.5.2.20 `#define ILUT_GL_GEN_S3TC 0x0635`

7.5.2.21 `#define ILUT_GL_USE_S3TC 0x0634`

7.5.2.22 `#define ILUT_ILLEGAL_OPERATION 0x0506`

7.5.2.23 `#define ILUT_INVALID_ENUM 0x0501`

7.5.2.24 `#define ILUT_INVALID_PARAM 0x0509`

7.5.2.25 `#define ILUT_INVALID_VALUE 0x0505`

7.5.2.26 `#define ILUT_MAXTEX_DEPTH 0x0632`

7.5.2.27 `#define ILUT_MAXTEX_HEIGHT 0x0631`

7.5.2.28 `#define ILUT_MAXTEX_WIDTH 0x0630`

7.5.2.29 `#define ILUT_NOT_SUPPORTED 0x0550`

7.5.2.30 `#define ILUT_OPENGL 0`

7.5.2.31 `#define ILUT_OPENGL_BIT 0x00000001`

7.5.2.32 `#define ILUT_OPENGL_CONV 0x0610`

7.5.2.33 `#define ILUT_OUT_OF_MEMORY 0x0502`

7.5.2.34 `#define ILUT_PALETTE_MODE 0x0600`

7.5.2.35 `#define ILUT_S3TC_FORMAT 0x0705`

7.5.2.36 `#define ILUT_STACK_OVERFLOW 0x050E`

7.5.2.37 `#define ILUT_STACK_UNDERFLOW 0x050F`

7.5.2.38 `#define ILUT_VENDOR IL_VENDOR`

7.5.2.39 `#define ILUT_VERSION 183`

7.5.2.40 `#define ILUT_VERSION_1_7_8 1`

7.5.2.41 `#define ILUT_VERSION_1_8_3 1`

7.5.2.42 `#define ILUT_VERSION_NUM IL_VERSION_NUM`

7.5.2.43 `#define ILUT_WIN32 2`

7.5.2.44 `#define ILUT_X11 5`



### 7.5.3 Function Documentation

- 7.5.3.1 ILAPI ILboolean ILAPIENTRY ilutDisable ( *ILenum Mode* )
- 7.5.3.2 ILAPI ILboolean ILAPIENTRY ilutEnable ( *ILenum Mode* )
- 7.5.3.3 ILAPI ILboolean ILAPIENTRY ilutGetBoolean ( *ILenum Mode* )
- 7.5.3.4 ILAPI void ILAPIENTRY ilutGetBooleanv ( *ILenum Mode*, *ILboolean \* Param* )
- 7.5.3.5 ILAPI ILint ILAPIENTRY ilutGetInteger ( *ILenum Mode* )
- 7.5.3.6 ILAPI void ILAPIENTRY ilutGetIntegerv ( *ILenum Mode*, *ILint \* Param* )
- 7.5.3.7 ILAPI ILstring ILAPIENTRY ilutGetString ( *ILenum StringName* )
- 7.5.3.8 ILAPI void ILAPIENTRY ilutInit ( *void* )
- 7.5.3.9 ILAPI ILboolean ILAPIENTRY ilutIsDisabled ( *ILenum Mode* )
- 7.5.3.10 ILAPI ILboolean ILAPIENTRY ilutIsEnabled ( *ILenum Mode* )
- 7.5.3.11 ILAPI void ILAPIENTRY ilutPopAttrib ( *void* )
- 7.5.3.12 ILAPI void ILAPIENTRY ilutPushAttrib ( *ILuint Bits* )
- 7.5.3.13 ILAPI ILboolean ILAPIENTRY ilutRenderer ( *ILenum Renderer* )
- 7.5.3.14 ILAPI void ILAPIENTRY ilutSetInteger ( *ILenum Mode*, *ILint Param* )

## 7.6 src/IL/algo/il\_neuquant.c File Reference

```
#include "il_internal.h"
```

### Data Structures

- struct [NeuQuantContext](#)

### Macros

- #define [alphabiasshift](#) 10
- #define [alpharadbias](#) (((*ILint*) 1)<<[alpharadbshift](#))
- #define [alpharadbshift](#) ([alphabiasshift](#)+[radbiasshift](#))
- #define [beta](#) ([intbias](#)>>[betashift](#))
- #define [betagamma](#) ([intbias](#)<<( [gammashift](#)-[betashift](#)))
- #define [betashift](#) 10
- #define [gamma](#) (((*ILint*) 1)<<[gammashift](#))
- #define [gammashift](#) 10
- #define [initalpha](#) (((*ILint*) 1)<<[alphabiasshift](#))
- #define [initrad](#) ([netsize](#)>>3)
- #define [initradius](#) ([initrad](#)\*[radiusbias](#))
- #define [intbias](#) (((*ILint*) 1)<<[intbiasshift](#))
- #define [intbiasshift](#) 16

- `#define maxnetpos(ctx) ((ctx)->netsizethink-1)`
- `#define minpicturebytes (3*prime4)`
- `#define ncycles 100`
- `#define netbiasshift 4`
- `#define netsize 256`
- `#define prime1 499`
- `#define prime2 491`
- `#define prime3 487`
- `#define prime4 503`
- `#define radbias (((ILint) 1)<<radbiasshift)`
- `#define radbiasshift 8`
- `#define radiusbias (((ILint) 1)<<radiusbiasshift)`
- `#define radiusbiasshift 6`
- `#define radiusdec 30`

## Typedefs

- `typedef int pixel [4]`

## Functions

- `void alterneigh (NeuQuantContext *ctx, ILint rad, ILint i, ILint b, ILint g, ILint r)`
- `void altersingle (NeuQuantContext *ctx, ILint alpha, ILint i, ILint b, ILint g, ILint r)`
- `ILint contest (NeuQuantContext *ctx, ILint b, ILint g, ILint r)`
- `ILImage * iNeuQuant (ILImage *Image, ILuint NumCols)`
- `void inxbuild (NeuQuantContext *ctx)`
- `ILubyte inxsearch (NeuQuantContext *ctx, ILint b, ILint g, ILint r)`
- `void learn (NeuQuantContext *ctx)`
- `void unbiasednet (NeuQuantContext *ctx)`

## 7.6.1 Macro Definition Documentation

7.6.1.1 `#define alphabiasshift 10`

7.6.1.2 `#define alphasradbias (((ILint) 1)<<alpharadbshift)`

7.6.1.3 `#define alphasradbshift (alphabiasshift+radbiasshift)`

7.6.1.4 `#define beta (intbias>>betashift)`

7.6.1.5 `#define betagamma (intbias<<(gammashift-betashift))`

7.6.1.6 `#define betashift 10`

7.6.1.7 `#define gamma (((ILint) 1)<<gammashift)`

7.6.1.8 `#define gammashift 10`

7.6.1.9 `#define initalpha (((ILint) 1)<<alphabiasshift)`

7.6.1.10 `#define initrad (netsize>>3)`

7.6.1.11 `#define initradius (initrad*radiusbias)`

7.6.1.12 `#define intbias (((ILint) 1)<<intbiasshift)`

7.6.1.13 `#define intbiasshift 16`

7.6.1.14 `#define maxnetpos( ctx ) ((ctx)->netsizethink-1)`

7.6.1.15 `#define minpicturebytes (3*prime4)`

7.6.1.16 `#define ncycles 100`

7.6.1.17 `#define netbiasshift 4`

7.6.1.18 `#define netsize 256`

7.6.1.19 `#define prime1 499`

7.6.1.20 `#define prime2 491`

7.6.1.21 `#define prime3 487`

7.6.1.22 `#define prime4 503`

7.6.1.23 `#define radbias (((ILint) 1)<<radbiasshift)`

7.6.1.24 `#define radbiasshift 8`

7.6.1.25 `#define radiusbias (((ILint) 1)<<radiusbiasshift)`

7.6.1.26 `#define radiusbiasshift 6`

7.6.1.27 `#define radiusdec 30`

## 7.6.2 Typedef Documentation

7.6.2.1 `typedef int pixel[4]`

## 7.6.3 Function Documentation

7.6.3.1 `void alterneigh ( NeuQuantContext * ctx, ILint rad, ILint i, ILint b, ILint g, ILint r )`

7.6.3.2 `void altersingle ( NeuQuantContext * ctx, ILint alpha, ILint i, ILint b, ILint g, ILint r )`

7.6.3.3 `ILint contest ( NeuQuantContext * ctx, ILint b, ILint g, ILint r )`

7.6.3.4 `ILImage* iNeuQuant ( ILImage * Image, ILuint NumCols )`

7.6.3.5 `void inxbuild ( NeuQuantContext * ctx )`

7.6.3.6 `ILubyte inxsearch ( NeuQuantContext * ctx, ILint b, ILint g, ILint r )`

7.6.3.7 `void learn ( NeuQuantContext * ctx )`

7.6.3.8 `void unbiasednet ( NeuQuantContext * ctx )`

## 7.7 src/IL/algo/il\_nvidia.cc File Reference

```
#include "il_internal.h"
#include "il_dds.h"
#include "il_manip.h"
#include <limits.h>
```

### Functions

- [ILAPI ILubyte\\* ILAPIENTRY ilNVidiaCompressDXT](#) ([ILubyte](#) \*Data, [ILuint](#) Width, [ILuint](#) Height, [ILuint](#) Depth, [ILenum](#) DxtFormat, [ILuint](#) \*DxtSize)

### 7.7.1 Function Documentation

**7.7.1.1** [ILAPI ILubyte\\* ILAPIENTRY ilNVidiaCompressDXT](#) ( [ILubyte](#) \* *Data*, [ILuint](#) *Width*, [ILuint](#) *Height*, [ILuint](#) *Depth*, [ILenum](#) *DxtFormat*, [ILuint](#) \* *DxtSize* )

## 7.8 src/IL/algo/il\_quantizer.c File Reference

```
#include "il_internal.h"
```

### Data Structures

- struct [Box](#)

### Macros

- [#define BLUE](#) 0
- [#define GREEN](#) 1
- [#define MAXCOLOR](#) 256
- [#define RED](#) 2

### Typedefs

- typedef struct [Box](#) [Box](#)

### Functions

- [ILint Bottom](#) ([Box](#) \*cube, [ILubyte](#) dir, [ILint](#) mmt[33][33][33])
- [ILint Cut](#) ([Box](#) \*set1, [Box](#) \*set2)
- [ILboolean Hist3d](#) ([ILubyte](#) \*Ir, [ILubyte](#) \*Ig, [ILubyte](#) \*Ib, [ILint](#) \*vwt, [ILint](#) \*vmr, [ILint](#) \*vmg, [ILint](#) \*vmb, [ILfloat](#) \*m2)
- [ILimage](#) \* [iQuantizeImage](#) ([ILimage](#) \*Image, [ILuint](#) NumCols)
- [void M3d](#) ([ILint](#) \*vwt, [ILint](#) \*vmr, [ILint](#) \*vmg, [ILint](#) \*vmb, [ILfloat](#) \*m2)
- [void Mark](#) (struct [Box](#) \*cube, int label, unsigned char \*tag)
- [ILfloat Maximize](#) ([Box](#) \*cube, [ILubyte](#) dir, [ILint](#) first, [ILint](#) last, [ILint](#) \*cut, [ILint](#) whole\_r, [ILint](#) whole\_g, [ILint](#) whole\_b, [ILint](#) whole\_w)
- [ILuint n2](#) ([ILint](#) s)
- [ILint Top](#) ([Box](#) \*cube, [ILubyte](#) dir, [ILint](#) pos, [ILint](#) mmt[33][33][33])

- [ILfloat Var](#) ([Box](#) \*cube)
- [ILint Vol](#) ([Box](#) \*cube, [ILint](#) mmt[33][33][33])

## Variables

- [ILubyte](#) \* buffer
- [ILfloat gm2](#) [33][33][33]
- [ILint](#) i
- [ILint](#) K
- [ILint mb](#) [33][33][33]
- [ILint mg](#) [33][33][33]
- [ILint mr](#) [33][33][33]
- [ILushort](#) \* Qadd
- [ILuint](#) size
- [ILint](#) WindD
- [ILint](#) WindH
- [ILint](#) WindW
- [ILint wt](#) [33][33][33]

## 7.8.1 Macro Definition Documentation

7.8.1.1 `#define BLUE 0`

7.8.1.2 `#define GREEN 1`

7.8.1.3 `#define MAXCOLOR 256`

7.8.1.4 `#define RED 2`

## 7.8.2 Typedef Documentation

7.8.2.1 `typedef struct Box Box`

## 7.8.3 Function Documentation

7.8.3.1 `ILint Bottom ( Box * cube, ILubyte dir, ILint mmt[33][33][33] )`

7.8.3.2 `ILint Cut ( Box * set1, Box * set2 )`

7.8.3.3 `ILboolean Hist3d ( ILubyte * lr, ILubyte * lg, ILubyte * lb, ILint * vwt, ILint * vmr, ILint * vmg, ILint * vmb, ILfloat * m2 )`

7.8.3.4 `ILimage* iQuantizeImage ( ILimage * Image, ILuint NumCols )`

7.8.3.5 `void M3d ( ILint * vwt, ILint * vmr, ILint * vmg, ILint * vmb, ILfloat * m2 )`

7.8.3.6 `void Mark ( struct Box * cube, int label, unsigned char * tag )`

7.8.3.7 `ILfloat Maximize ( Box * cube, ILubyte dir, ILint first, ILint last, ILint * cut, ILint whole_r, ILint whole_g, ILint whole_b, ILint whole_w )`

7.8.3.8 `ILuint n2 ( ILint s )`

7.8.3.9 `ILint Top ( Box * cube, ILubyte dir, ILint pos, ILint mmt[33][33][33] )`

7.8.3.10 ILfloat Var ( Box \* cube )

7.8.3.11 ILint Vol ( Box \* cube, ILint mmt[33][33][33] )

## 7.8.4 Variable Documentation

7.8.4.1 ILubyte\* buffer

7.8.4.2 ILfloat gm2[33][33][33]

7.8.4.3 ILint i

7.8.4.4 ILint K

7.8.4.5 ILint mb[33][33][33]

7.8.4.6 ILint mg[33][33][33]

7.8.4.7 ILint mr[33][33][33]

7.8.4.8 ILushort\* Qadd

7.8.4.9 ILuint size

7.8.4.10 ILint WindD

7.8.4.11 ILint WindH

7.8.4.12 ILint WindW

7.8.4.13 ILint wt[33][33][33]

## 7.9 src/IL/algo/il\_rle.c File Reference

```
#include "il_internal.h"
#include "il_rle.h"
```

### Macros

- #define IL\_RLE\_C

### Functions

- ILuint ilRleCompress (ILubyte \*Data, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, ILubyte \*Dest, ILenum CompressMode, ILuint \*ScanTable)
- ILboolean ilRleCompressLine (ILubyte \*p, ILuint n, ILubyte bpp, ILubyte \*q, ILuint \*DestWidth, ILenum CompressMode)

### 7.9.1 Macro Definition Documentation

7.9.1.1 #define IL\_RLE\_C

## 7.9.2 Function Documentation

7.9.2.1 `ILuint ilRleCompress ( ILubyte * Data, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, ILubyte * Dest, ILenum CompressMode, ILuint * ScanTable )`

7.9.2.2 `ILboolean ilRleCompressLine ( ILubyte * p, ILuint n, ILubyte bpp, ILubyte * q, ILuint * DestWidth, ILenum CompressMode )`

## 7.10 src/IL/algo/il\_rle.h File Reference

```
#include "il_internal.h"
```

### Macros

- `#define BMP_MAX_RUN 127`
- `#define SGI_MAX_RUN 127`
- `#define TGA_MAX_RUN 128`

### Functions

- `INLINE ILint CountDiffPixels (ILubyte *p, ILuint bpp, ILuint pixCnt)`
- `INLINE ILint CountSamePixels (ILubyte *p, ILuint bpp, ILuint pixCnt)`
- `INLINE ILuint GetPix (ILubyte *p, ILuint bpp)`

### 7.10.1 Macro Definition Documentation

7.10.1.1 `#define BMP_MAX_RUN 127`

7.10.1.2 `#define SGI_MAX_RUN 127`

7.10.1.3 `#define TGA_MAX_RUN 128`

### 7.10.2 Function Documentation

7.10.2.1 `ILint CountDiffPixels ( ILubyte * p, ILuint bpp, ILuint pixCnt )`

7.10.2.2 `ILint CountSamePixels ( ILubyte * p, ILuint bpp, ILuint pixCnt )`

7.10.2.3 `ILuint GetPix ( ILubyte * p, ILuint bpp )`

## 7.11 src/IL/algo/il\_squish.c File Reference

```
#include "il_internal.h"
```

### Functions

- `ILAPI ILubyte *ILAPIENTRY ilSquishCompressDXT (ILubyte *Data, ILuint Width, ILuint Height, ILuint Depth, ILenum DxtFormat, ILuint *DxtSize)`

### 7.11.1 Function Documentation

7.11.1.1 **ILAPI** **ILubyte\*** **ILAPIENTRY** **ilSquishCompressDXT** ( **ILubyte** \* *Data*, **ILuint** *Width*, **ILuint** *Height*, **ILuint** *Depth*, **ILenum** *DxtFormat*, **ILuint** \* *DxtSize* )

## 7.12 src/IL/altivec/common.c File Reference

```
#include <IL/config.h>
```

## 7.13 src/IL/altivec/common.h File Reference

```
#include "il_internal.h"
```

## 7.14 src/IL/altivec/typeconversion.c File Reference

```
#include <IL/config.h>
```

## 7.15 src/IL/altivec/typeconversion.h File Reference

```
#include "altivec_common.h"
```

## 7.16 src/IL/conv/il\_color.h File Reference

### Functions

- **INLINE** **void** **iYCbCr2RGB** (**ILubyte** *Y*, **ILubyte** *Cb*, **ILubyte** *Cr*, **ILubyte** \**r*, **ILubyte** \**g*, **ILubyte** \**b*)

### 7.16.1 Function Documentation

7.16.1.1 **INLINE** **void** **iYCbCr2RGB** ( **ILubyte** *Y*, **ILubyte** *Cb*, **ILubyte** *Cr*, **ILubyte** \* *r*, **ILubyte** \* *g*, **ILubyte** \* *b* )

## 7.17 src/IL/conv/il\_convbuff.c File Reference

```
#include "il_internal.h"  
#include "il_manip.h"  
#include <limits.h>
```

### Macros

- **#define** **CHECK\_ALLOC**()



## Functions

- [ILImage \\*](#) [iConvertPalette](#) ([ILImage \\*](#)Image, [ILenum](#) DestFormat)
- [ILAPI void \\*](#) [ILAPIENTRY ilConvertBuffer](#) ([ILuint](#) SizeOfData, [ILenum](#) SrcFormat, [ILenum](#) DestFormat, [ILenum](#) SrcType, [ILenum](#) DestType, [ILpal \\*](#)SrcPal, [void \\*](#)Buffer)
- [void \\*](#) [ILAPIENTRY iSwitchTypes](#) ([ILuint](#) SizeOfData, [ILenum](#) SrcType, [ILenum](#) DestType, [void \\*](#)Buffer)

### 7.17.1 Macro Definition Documentation

#### 7.17.1.1 #define CHECK\_ALLOC( )

##### Value:

```
if (NewData == NULL) { \
    if (Data != Buffer) \
        ifree(Data); \
    return IL_FALSE; \
}
```

### 7.17.2 Function Documentation

#### 7.17.2.1 [ILImage \\*](#) [iConvertPalette](#) ( [ILImage \\*](#) Image, [ILenum](#) DestFormat )

#### 7.17.2.2 [ILAPI void \\*](#) [ILAPIENTRY ilConvertBuffer](#) ( [ILuint](#) SizeOfData, [ILenum](#) SrcFormat, [ILenum](#) DestFormat, [ILenum](#) SrcType, [ILenum](#) DestType, [ILpal \\*](#) SrcPal, [void \\*](#) Buffer )

#### 7.17.2.3 [void \\*](#) [ILAPIENTRY iSwitchTypes](#) ( [ILuint](#) SizeOfData, [ILenum](#) SrcType, [ILenum](#) DestType, [void \\*](#) Buffer )

## 7.18 src/IL/conv/il\_convert.c File Reference

```
#include "il_internal.h"
#include "il_manip.h"
#include <limits.h>
```

## Functions

- [ILboolean](#) [iAddAlpha](#) ([ILImage \\*](#)Image)
- [ILboolean](#) [iAddAlphaKey](#) ([ILImage \\*](#)Image)
- [ILAPI ILImage \\*](#) [ILAPIENTRY iConvertImage](#) ([ILImage \\*](#)Image, [ILenum](#) DestFormat, [ILenum](#) DestType)
- [ILboolean](#) [ILAPIENTRY iConvertImage\\_](#) ([ILImage \\*](#)BaseImage, [ILenum](#) DestFormat, [ILenum](#) DestType)
- [ILImage \\*](#) [iConvertPalette](#) ([ILImage \\*](#)Image, [ILenum](#) DestFormat)
- [ILboolean](#) [iFixImage](#) ([ILImage \\*](#)Image)
- [ILboolean](#) [iFixImages](#) ([ILImage \\*](#)BaseImage)
- [ILboolean](#) [ilAddAlpha](#) ()
- [ILboolean](#) [ILAPIENTRY ilConvertImage](#) ([ILenum](#) DestFormat, [ILenum](#) DestType)  
*Converts the current image to the DestFormat format.*
- [void](#) [ILAPIENTRY ilKeyColour](#) ([ILclampf](#) Red, [ILclampf](#) Green, [ILclampf](#) Blue, [ILclampf](#) Alpha)
- [ILboolean](#) [ilRemoveAlpha](#) ()
- [ILboolean](#) [ilSwapColours](#) ()
- [ILImage \\*](#) [iNeuQuant](#) ([ILImage \\*](#)Image, [ILuint](#) NumCols)
- [ILImage \\*](#) [iQuantizeImage](#) ([ILImage \\*](#)Image, [ILuint](#) NumCols)
- [ILboolean](#) [iRemoveAlpha](#) ([ILImage \\*](#)Image)
- [ILboolean](#) [iSwapColours](#) ([ILImage \\*](#)Image)

## Variables

- `ILfloat KeyAlpha` = 0
- `ILfloat KeyBlue` = 0
- `ILfloat KeyGreen` = 0
- `ILfloat KeyRed` = 0

## 7.18.1 Function Documentation

7.18.1.1 `ILboolean iAddAlpha ( ILImage * Image )`

7.18.1.2 `ILboolean iAddAlphaKey ( ILImage * Image )`

7.18.1.3 `ILAPI ILImage* ILAPIENTRY iConvertImage ( ILImage * Image, IEnum DestFormat, IEnum DestType )`

7.18.1.4 `ILboolean ILAPIENTRY iConvertImage_ ( ILImage * BaseImage, IEnum DestFormat, IEnum DestType )`

7.18.1.5 `ILImage* iConvertPalette ( ILImage * Image, IEnum DestFormat )`

7.18.1.6 `ILboolean iFixImage ( ILImage * Image )`

7.18.1.7 `ILboolean iFixImages ( ILImage * BaseImage )`

7.18.1.8 `ILboolean ilAddAlpha ( )`

7.18.1.9 `ILboolean ILAPIENTRY ilConvertImage ( IEnum DestFormat, IEnum DestType )`

Converts the current image to the DestFormat format.

### Parameters

<i>DestFormat</i>	An enum of the desired output format. Any format values are accepted.
<i>DestType</i>	An enum of the desired output type. Any type values are accepted.

### Exceptions

<i>IL_ILLEGAL_OPERATION</i>	No currently bound image
<i>IL_INVALID_CONVERSION</i>	DestFormat or DestType was an invalid identifier.
<i>IL_OUT_OF_MEMORY</i>	Could not allocate enough memory.

### Returns

Boolean value of failure or success

7.18.1.10 `void ILAPIENTRY ilKeyColour ( ILclampf Red, ILclampf Green, ILclampf Blue, ILclampf Alpha )`

7.18.1.11 `ILboolean ilRemoveAlpha ( )`

7.18.1.12 `ILboolean ilSwapColours ( )`

7.18.1.13 `ILImage* iNeuQuant ( ILImage * Image, ILuint NumCols )`

7.18.1.14 `ILImage* iQuantizeImage ( ILImage * Image, ILuint NumCols )`

7.18.1.15 `ILboolean iRemoveAlpha ( ILImage * Image )`

7.18.1.16 ILboolean iSwapColours ( ILImage \* *Image* )

## 7.18.2 Variable Documentation

7.18.2.1 ILfloat KeyAlpha = 0

7.18.2.2 ILfloat KeyBlue = 0

7.18.2.3 ILfloat KeyGreen = 0

7.18.2.4 ILfloat KeyRed = 0

## 7.19 src/IL/conv/il\_fastconv.c File Reference

```
#include "il_internal.h"
```

### Functions

- [ILboolean iFastConvert](#) (ILImage \*Image, IEnum DestFormat)

### 7.19.1 Function Documentation

7.19.1.1 ILboolean iFastConvert ( ILImage \* *Image*, IEnum *DestFormat* )

## 7.20 src/IL/formats/il\_blp.c File Reference

```
#include "il_internal.h"
#include "il_dds.h"
```

### Data Structures

- struct [BLP1HEAD](#)
- struct [BLP2HEAD](#)

### Macros

- #define [BLP\\_DXTC](#) 2
- #define [BLP\\_RAW](#) 1
- #define [BLP\\_RAW\\_NO\\_ALPHA](#) 5
- #define [BLP\\_RAW\\_PLUS\\_ALPHA1](#) 3
- #define [BLP\\_RAW\\_PLUS\\_ALPHA2](#) 4
- #define [BLP\\_TYPE\\_DXTC\\_RAW](#) 1
- #define [BLP\\_TYPE\\_JPG](#) 0

### Typedefs

- typedef struct [BLP1HEAD](#) [BLP1HEAD](#)
- typedef struct [BLP2HEAD](#) [BLP2HEAD](#)

## Variables

- [ILformat iFormatBLP](#)
- [ILconst\\_string iFormatExtsBLP \[\]](#)

## 7.20.1 Macro Definition Documentation

7.20.1.1 `#define BLP_DXTC 2`

7.20.1.2 `#define BLP_RAW 1`

7.20.1.3 `#define BLP_RAW_NO_ALPHA 5`

7.20.1.4 `#define BLP_RAW_PLUS_ALPHA1 3`

7.20.1.5 `#define BLP_RAW_PLUS_ALPHA2 4`

7.20.1.6 `#define BLP_TYPE_DXTC_RAW 1`

7.20.1.7 `#define BLP_TYPE_JPG 0`

## 7.20.2 Typedef Documentation

7.20.2.1 `typedef struct BLP1HEAD BLP1HEAD`

7.20.2.2 `typedef struct BLP2HEAD BLP2HEAD`

## 7.20.3 Variable Documentation

7.20.3.1 `ILformat iFormatBLP`

### Initial value:

```
= {
    .Validate = iIsValidBLP,
    .Load     = iLoadBlpInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsBLP
}
```

7.20.3.2 `ILconst_string iFormatExtsBLP[]`

### Initial value:

```
= {
    IL_TEXT("blp"),
    NULL
}
```

## 7.21 src/IL/formats/il\_bmp.c File Reference

```
#include "il_internal.h"
#include "il_bmp.h"
#include "il_endian.h"
#include <stdio.h>
```

## Macros

- `#define IL_BMP_C`

## Typedefs

- `typedef ILuchar BYTE`
- `typedef ILuint DWORD`
- `typedef ILuchar * LPBYTE`
- `typedef ILuint * LPDWORD`
- `typedef ILushort * LPWORD`
- `typedef ILushort WORD`

## Functions

- `void GetShiftFromMask (const ILuint Mask, ILuint *CONST_RESTRICT ShiftLeft, ILuint *CONST_RESTRICT ShiftRight)`
- `ILboolean iCheckBmp (const BMPHEAD *CONST_RESTRICT Header)`
- `ILboolean iCheckOS2 (const OS2_HEAD *CONST_RESTRICT Header)`
- `ILboolean iGetBmpHead (SIO *io, BMPHEAD *const Header)`
- `ILboolean iGetOS2Bmp (ILImage *image, OS2_HEAD *Header)`
- `ILboolean iGetOS2Head (SIO *io, OS2_HEAD *const Header)`
- `ILboolean ilReadRLE4Bmp (ILImage *image, BMPHEAD *Header)`
- `ILboolean ilReadRLE8Bmp (ILImage *image, BMPHEAD *Header)`
- `ILboolean ilReadUncompBmp (ILImage *image, BMPHEAD *header)`
- `ILboolean ilReadUncompBmp1 (ILImage *image, BMPHEAD *Header)`
- `ILboolean ilReadUncompBmp16 (ILImage *image, BMPHEAD *Header)`
- `ILboolean ilReadUncompBmp24 (ILImage *image, BMPHEAD *Header)`
- `ILboolean ilReadUncompBmp32 (ILImage *image, BMPHEAD *Header)`
- `ILboolean ilReadUncompBmp4 (ILImage *image, BMPHEAD *Header)`
- `ILboolean ilReadUncompBmp8 (ILImage *image, BMPHEAD *Header)`
- `ILboolean prepareBMP (ILImage *image, BMPHEAD *Header, ILubyte bpp, ILuint format)`

## Variables

- `ILformat iFormatBMP`
- `ILconst_string iFormatExtsBMP []`

### 7.21.1 Macro Definition Documentation

#### 7.21.1.1 `#define IL_BMP_C`

### 7.21.2 Typedef Documentation

#### 7.21.2.1 `typedef ILuchar BYTE`

#### 7.21.2.2 `typedef ILuint DWORD`

#### 7.21.2.3 `typedef ILuchar * LPBYTE`

#### 7.21.2.4 `typedef ILuint * LPDWORD`

7.21.2.5 `typedef ILushort * LPWORD`

7.21.2.6 `typedef ILushort WORD`

### 7.21.3 Function Documentation

7.21.3.1 `void GetShiftFromMask ( const ILuint Mask, ILuint *CONST_RESTRICT ShiftLeft, ILuint *CONST_RESTRICT ShiftRight ) [inline]`

7.21.3.2 `ILboolean iCheckBmp ( const BMPHEAD *CONST_RESTRICT Header )`

7.21.3.3 `ILboolean iCheckOS2 ( const OS2_HEAD *CONST_RESTRICT Header )`

7.21.3.4 `ILboolean iGetBmpHead ( SIO * io, BMPHEAD *const Header )`

7.21.3.5 `ILboolean iGetOS2Bmp ( ILImage * image, OS2_HEAD * Header )`

7.21.3.6 `ILboolean iGetOS2Head ( SIO * io, OS2_HEAD *const Header )`

7.21.3.7 `ILboolean ilReadRLE4Bmp ( ILImage * image, BMPHEAD * Header )`

7.21.3.8 `ILboolean ilReadRLE8Bmp ( ILImage * image, BMPHEAD * Header )`

7.21.3.9 `ILboolean ilReadUncompBmp ( ILImage * image, BMPHEAD * header )`

7.21.3.10 `ILboolean ilReadUncompBmp1 ( ILImage * image, BMPHEAD * Header )`

7.21.3.11 `ILboolean ilReadUncompBmp16 ( ILImage * image, BMPHEAD * Header )`

7.21.3.12 `ILboolean ilReadUncompBmp24 ( ILImage * image, BMPHEAD * Header )`

7.21.3.13 `ILboolean ilReadUncompBmp32 ( ILImage * image, BMPHEAD * Header )`

7.21.3.14 `ILboolean ilReadUncompBmp4 ( ILImage * image, BMPHEAD * Header )`

7.21.3.15 `ILboolean ilReadUncompBmp8 ( ILImage * image, BMPHEAD * Header )`

7.21.3.16 `ILboolean prepareBMP ( ILImage * image, BMPHEAD * Header, ILubyte bpp, ILuint format )`

### 7.21.4 Variable Documentation

7.21.4.1 `ILformat iFormatBMP`

**Initial value:**

```
= {
    .Validate = iIsValidBmp,
    .Load     = iLoadBitmapInternal,
    .Save     = iSaveBitmapInternal,
    .Exts     = iFormatExtsBMP
}
```

7.21.4.2 `ILconst_string iFormatExtsBMP[]`

**Initial value:**

```
= {
```

```
IL_TEXT("bmp"),  
IL_TEXT("dib"),  
NULL  
}
```

## 7.22 src/IL/formats/il\_bmp.h File Reference

```
#include "il_internal.h"  
#include "pack_push.h"  
#include "pack_pop.h"
```

### Data Structures

- struct [BMPHEAD](#)
- struct [OS2\\_HEAD](#)

### Typedefs

- typedef struct [BMPHEAD](#) [BMPHEAD](#)
- typedef struct [OS2\\_HEAD](#) [OS2\\_HEAD](#)

#### 7.22.1 Typedef Documentation

7.22.1.1 typedef struct [BMPHEAD](#) [BMPHEAD](#)

7.22.1.2 typedef struct [OS2\\_HEAD](#) [OS2\\_HEAD](#)

## 7.23 src/IL/formats/il\_cut.c File Reference

```
#include "il_internal.h"  
#include "il_manip.h"  
#include "il_pal.h"  
#include "pack_push.h"  
#include "pack_pop.h"
```

### Data Structures

- struct [CUT\\_HEAD](#)

### Typedefs

- typedef struct [CUT\\_HEAD](#) [CUT\\_HEAD](#)

### Functions

- [ILboolean](#) [isValidCutHeader](#) (const [CUT\\_HEAD](#) \*header)
- [ILboolean](#) [readScanLine](#) ([ILImage](#) \*image, [ILubyte](#) \*chunk, [ILushort](#) chunkSize, int y)

## Variables

- [ILformat iFormatCUT](#)
- [ILconst\\_string iFormatExtsCUT \[\]](#)

### 7.23.1 Typedef Documentation

#### 7.23.1.1 typedef struct CUT\_HEAD CUT\_HEAD

### 7.23.2 Function Documentation

#### 7.23.2.1 ILboolean isValidCutHeader ( const CUT\_HEAD \* *header* )

#### 7.23.2.2 ILboolean readScanLine ( ILImage \* *image*, ILubyte \* *chunk*, ILushort *chunkSize*, int *y* )

### 7.23.3 Variable Documentation

#### 7.23.3.1 ILformat iFormatCUT

##### Initial value:

```
= {
    .Validate = iIsValidCut,
    .Load     = iLoadCutInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsCUT
}
```

#### 7.23.3.2 ILconst\_string iFormatExtsCUT[]

##### Initial value:

```
= {
    IL_TEXT("cut"),
    NULL
}
```

## 7.24 src/IL/formats/il\_dcx.c File Reference

```
#include "il_internal.h"
#include "il_dcx.h"
#include "il_manip.h"
```

## Variables

- [ILformat iFormatDCX](#)
- [ILconst\\_string iFormatExtsDCX \[\]](#)

### 7.24.1 Variable Documentation

#### 7.24.1.1 ILformat iFormatDCX

##### Initial value:



```
= {
    .Validate = iIsValidDcx,
    .Load     = iLoadDcxInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsDCX
}
```

#### 7.24.1.2 ILconst\_string iFormatExtsDCX[]

**Initial value:**

```
= {
    IL_TEXT("dcx"),
    NULL
}
```

## 7.25 src/IL/formats/il\_dcx.h File Reference

```
#include "il_internal.h"
#include "pack_push.h"
#include "pack_pop.h"
```

### Data Structures

- struct [DCXHEAD](#)

### Typedefs

- typedef struct [DCXHEAD](#) [DCXHEAD](#)

#### 7.25.1 Typedef Documentation

##### 7.25.1.1 typedef struct [DCXHEAD](#) [DCXHEAD](#)

## 7.26 src/IL/formats/il\_dds-save.c File Reference

```
#include "il_internal.h"
#include "il_dds.h"
#include "il_manip.h"
#include "il_stack.h"
#include "il_states.h"
#include <limits.h>
```

### Macros

- #define [NormSquared](#)(c) ((c)->r \* (c)->r + (c)->g \* (c)->g + (c)->b \* (c)->b)
- #define [Sum](#)(c) ((c)->r + (c)->g + (c)->b)

## Functions

- `ILushort As16Bit (ILint r, ILint g, ILint b)`
- `void ChooseAlphaEndpoints (ILubyte *Block, ILubyte *a0, ILubyte *a1)`
- `void ChooseEndpoints (ILushort *Block, ILushort *ex0, ILushort *ex1)`
- `ILushort Color565ToShort (Color565 *Colour)`
- `ILushort Color888ToShort (Color888 *Colour)`
- `ILuint Compress (ILimage *Image, ILenum DXTCFormat)`
- `ILushort * CompressTo565 (ILimage *Image)`
- `ILubyte * CompressTo88 (ILimage *Image)`
- `void CompressToRXGB (ILimage *Image, ILushort **xgb, ILubyte **r)`
- `void CorrectEndDXT1 (ILushort *ex0, ILushort *ex1, ILboolean HasAlpha)`
- `ILuint Distance (Color888 *c1, Color888 *c2)`
- `void GenAlphaBitMask (ILubyte a0, ILubyte a1, ILubyte *In, ILubyte *Mask, ILubyte *Out)`
- `ILuint GenBitMask (ILushort ex0, ILushort ex1, ILuint NumCols, ILushort *In, ILubyte *Alpha, Color888 *OutCol)`
- `ILboolean Get3DcBlock (ILubyte *Block, ILubyte *Data, ILimage *Image, ILuint XPos, ILuint YPos, int channel)`
- `ILboolean GetAlphaBlock (ILubyte *Block, ILubyte *Data, ILimage *Image, ILuint XPos, ILuint YPos)`
- `ILboolean GetBlock (ILushort *Block, ILushort *Data, ILimage *Image, ILuint XPos, ILuint YPos)`
- `ILuint GetCubemapInfo (ILimage *image, ILint *faces)`  
*Checks if an image is a cubemap.*
- `ILuint ILAPIENTRY iGetDXTCData (ILimage *Image, void *Buffer, ILuint BufferSize, ILenum DXTCFormat)`
- `ILAPI ILubyte *ILAPIENTRY iCompressDXT (ILubyte *Data, ILuint Width, ILuint Height, ILuint Depth, ILenum DXTCFormat, ILuint *DXTCSize)`  
*Compresses data to a DXT format using different methods.*
- `ILuint ILAPIENTRY iGetDXTCData (void *Buffer, ILuint BufferSize, ILenum DXTCFormat)`
- `ILboolean iSaveDdsInternal (ILimage *Image)`
- `void PreMult (ILushort *Data, ILubyte *Alpha)`
- `ILuint RMSAlpha (ILubyte *Orig, ILubyte *Test)`
- `void ShortToColor565 (ILushort Pixel, Color565 *Colour)`
- `void ShortToColor888 (ILushort Pixel, Color888 *Colour)`
- `ILboolean WriteHeader (ILimage *Image, ILenum DXTCFormat, ILuint CubeFlags)`

## 7.26.1 Macro Definition Documentation

7.26.1.1 `#define NormSquared( c ) ((c)->r * (c)->r + (c)->g * (c)->g + (c)->b * (c)->b)`

7.26.1.2 `#define Sum( c ) ((c)->r + (c)->g + (c)->b)`

## 7.26.2 Function Documentation

7.26.2.1 `ILushort As16Bit ( ILint r, ILint g, ILint b )`

7.26.2.2 `void ChooseAlphaEndpoints ( ILubyte * Block, ILubyte * a0, ILubyte * a1 )`

7.26.2.3 `void ChooseEndpoints ( ILushort * Block, ILushort * ex0, ILushort * ex1 )`

7.26.2.4 `ILushort Color565ToShort ( Color565 * Colour )`

7.26.2.5 `ILushort Color888ToShort ( Color888 * Colour )`

7.26.2.6 `ILuint Compress ( ILimage * Image, ILenum DXTCFormat )`

7.26.2.7 **ILushort\*** CompressTo565 ( **ILImage \*** *Image* )

7.26.2.8 **ILubyte\*** CompressTo88 ( **ILImage \*** *Image* )

7.26.2.9 **void** CompressToRXGB ( **ILImage \*** *Image*, **ILushort \*\*** *xgb*, **ILubyte \*\*** *r* )

7.26.2.10 **void** CorrectEndDXT1 ( **ILushort \*** *ex0*, **ILushort \*** *ex1*, **ILboolean** *HasAlpha* )

7.26.2.11 **ILuint** Distance ( **Color888 \*** *c1*, **Color888 \*** *c2* )

7.26.2.12 **void** GenAlphaBitMask ( **ILubyte** *a0*, **ILubyte** *a1*, **ILubyte \*** *In*, **ILubyte \*** *Mask*, **ILubyte \*** *Out* )

7.26.2.13 **ILuint** GenBitMask ( **ILushort** *ex0*, **ILushort** *ex1*, **ILuint** *NumCols*, **ILushort \*** *In*, **ILubyte \*** *Alpha*, **Color888 \*** *OutCol* )

7.26.2.14 **ILboolean** Get3DcBlock ( **ILubyte \*** *Block*, **ILubyte \*** *Data*, **ILImage \*** *Image*, **ILuint** *XPos*, **ILuint** *YPos*, **int** *channel* )

7.26.2.15 **ILboolean** GetAlphaBlock ( **ILubyte \*** *Block*, **ILubyte \*** *Data*, **ILImage \*** *Image*, **ILuint** *XPos*, **ILuint** *YPos* )

7.26.2.16 **ILboolean** GetBlock ( **ILushort \*** *Block*, **ILushort \*** *Data*, **ILImage \*** *Image*, **ILuint** *XPos*, **ILuint** *YPos* )

7.26.2.17 **ILuint** GetCubemapInfo ( **ILImage \*** *image*, **ILint \*** *faces* )

Checks if an image is a cubemap.

7.26.2.18 **ILuint** ILAPIENTRY iGetDXTCData ( **ILImage \*** *Image*, **void \*** *Buffer*, **ILuint** *BufferSize*, **ILenum** *DXTCFormat* )

7.26.2.19 **ILAPI ILubyte\*** ILAPIENTRY iCompressDXT ( **ILubyte \*** *Data*, **ILuint** *Width*, **ILuint** *Height*, **ILuint** *Depth*, **ILenum** *DXTCFormat*, **ILuint \*** *DXTCSize* )

Compresses data to a DXT format using different methods.

7.26.2.20 **ILuint** ILAPIENTRY iGetDXTCData ( **void \*** *Buffer*, **ILuint** *BufferSize*, **ILenum** *DXTCFormat* )

7.26.2.21 **ILboolean** iSaveDdsInternal ( **ILImage \*** *Image* )

7.26.2.22 **void** PreMult ( **ILushort \*** *Data*, **ILubyte \*** *Alpha* )

7.26.2.23 **ILuint** RMSAlpha ( **ILubyte \*** *Orig*, **ILubyte \*** *Test* )

7.26.2.24 **void** ShortToColor565 ( **ILushort** *Pixel*, **Color565 \*** *Colour* )

7.26.2.25 **void** ShortToColor888 ( **ILushort** *Pixel*, **Color888 \*** *Colour* )

7.26.2.26 **ILboolean** WriteHeader ( **ILImage \*** *Image*, **ILenum** *DXTCFormat*, **ILuint** *CubeFlags* )

## 7.27 src/IL/formats/il\_dds.c File Reference

```
#include "il_internal.h"
#include "il_dds.h"
```

## Data Structures

- struct [DDS\\_CONTEXT](#)

## Functions

- void [Check16BitComponents](#) ([DDS\\_CONTEXT](#) \*ctx)
- void [CorrectPreMult](#) ([ILImage](#) \*Image)
- [ILuint](#) [CountBitsFromMask](#) ([ILuint](#) Mask)
- [ILuint](#) [DecodePixelFormat](#) ([DDS\\_CONTEXT](#) \*ctx, [ILuint](#) \*CompFormat)
- [ILboolean](#) [Decompress3Dc](#) ([DDS\\_CONTEXT](#) \*ctx)
- [ILboolean](#) [DecompressAti1n](#) ([DDS\\_CONTEXT](#) \*ctx)
- [ILboolean](#) [DecompressDXT1](#) ([ILImage](#) \*Image, [ILubyte](#) \*ICompData)
- [ILboolean](#) [DecompressDXT2](#) ([ILImage](#) \*Image, [ILubyte](#) \*ICompData)
- [ILboolean](#) [DecompressDXT3](#) ([ILImage](#) \*Image, [ILubyte](#) \*ICompData)
- [ILboolean](#) [DecompressDXT4](#) ([ILImage](#) \*Image, [ILubyte](#) \*ICompData)
- [ILboolean](#) [DecompressDXT5](#) ([ILImage](#) \*Image, [ILubyte](#) \*ICompData)
- [ILboolean](#) [DecompressRXGB](#) ([DDS\\_CONTEXT](#) \*ctx)
- void [DxtcReadColor](#) ([ILushort](#) Data, [Color8888](#) \*Out)
- void [DxtcReadColors](#) (const [ILubyte](#) \*Data, [Color8888](#) \*Out)
- void [GetBitsFromMask](#) ([ILuint](#) Mask, [ILuint](#) \*ShiftLeft, [ILuint](#) \*ShiftRight)
- unsigned int [halfToFloat](#) (unsigned short y)
- [ILboolean](#) [iCheckDds](#) ([DDSHEAD](#) \*Head)
- [ILubyte](#) [iCompFormatToChannelCount](#) ([ILenum](#) Format)
- void [iComplexAlphaHelper](#) ([ILubyte](#) \*Data)
- [ILboolean](#) [iConvFloat16ToFloat32](#) ([ILuint](#) \*dest, [ILushort](#) \*src, [ILuint](#) size)
- [ILboolean](#) [iConvG16R16ToFloat32](#) ([ILuint](#) \*dest, [ILushort](#) \*src, [ILuint](#) size)
- [ILboolean](#) [iConvR16ToFloat32](#) ([ILuint](#) \*dest, [ILushort](#) \*src, [ILuint](#) size)
- [ILAPI](#) [ILboolean](#) [ILAPIENTRY](#) [iDxtcDataToImage](#) ([ILImage](#) \*image)
- [ILAPI](#) [ILboolean](#) [ILAPIENTRY](#) [iDxtcDataToSurface](#) ([ILImage](#) \*image)
- void [iFlip3dc](#) ([ILubyte](#) \*data, [ILuint](#) count)
- void [iFlipColorBlock](#) ([ILubyte](#) \*data)
- void [iFlipComplexAlphaBlock](#) ([ILubyte](#) \*Data)
- void [iFlipDxt1](#) ([ILubyte](#) \*data, [ILuint](#) count)
- void [iFlipDxt3](#) ([ILubyte](#) \*data, [ILuint](#) count)
- void [iFlipDxt5](#) ([ILubyte](#) \*data, [ILuint](#) count)
- void [iFlipSimpleAlphaBlock](#) ([ILushort](#) \*data)
- void [iFlipSurfaceDxtcData](#) ([ILImage](#) \*image)
- [ILboolean](#) [iGetDdsHead](#) ([SIO](#) \*io, [DDSHEAD](#) \*Header)
- [ILboolean](#) [iImageToDxtcData](#) ([ILImage](#) \*image, [ILenum](#) Format)
- void [iInvertDxt3Alpha](#) ([ILubyte](#) \*data)
- void [iInvertDxt5Alpha](#) ([ILubyte](#) \*data)
- [ILboolean](#) [iInvertSurfaceDxtcDataAlpha](#) ([ILImage](#) \*image)
- [ILboolean](#) [iIsValidDds](#) ([SIO](#) \*io)
- [ILAPI](#) void [ILAPIENTRY](#) [iFlipSurfaceDxtcData](#) ()
- void [iFreeImageDxtcData](#) ([ILImage](#) \*image)
- void [iFreeSurfaceDxtcData](#) ([ILImage](#) \*image)
- [ILubyte](#) \*[ILAPIENTRY](#) [iGetDxtcData](#) ([ILImage](#) \*image)
- [ILAPI](#) [ILboolean](#) [ILAPIENTRY](#) [iImageToDxtcData](#) ([ILenum](#) Format)
- [ILAPI](#) [ILboolean](#) [ILAPIENTRY](#) [iInvertSurfaceDxtcDataAlpha](#) ()
- [ILboolean](#) [iLoadDdsCubemapInternal](#) ([DDS\\_CONTEXT](#) \*ctx, [ILuint](#) CompFormat)
- [ILboolean](#) [iLoadDdsInternal](#) ([ILImage](#) \*image)
- [ILAPI](#) [ILboolean](#) [ILAPIENTRY](#) [iTexImageDxtc](#) ([ILint](#) w, [ILint](#) h, [ILint](#) d, [ILenum](#) DxtFormat, const [ILubyte](#) \*data)
- [ILAPI](#) [ILboolean](#) [ILAPIENTRY](#) [iSurfaceToDxtcData](#) ([ILImage](#) \*image, [ILenum](#) Format)
- [ILboolean](#) [iTexImageDxtc](#) ([ILImage](#) \*image, [ILint](#) w, [ILint](#) h, [ILint](#) d, [ILenum](#) DxtFormat, const [ILubyte](#) \*data)

## Variables

- [ILformat iFormatDDS](#)
- [ILconst\\_string iFormatExtsDDS \[\]](#)

## 7.27.1 Function Documentation

- 7.27.1.1 void Check16BitComponents ( DDS\_CONTEXT \* ctx )
- 7.27.1.2 void CorrectPreMult ( ILImage \* Image )
- 7.27.1.3 ILuint CountBitsFromMask ( ILuint Mask )
- 7.27.1.4 ILuint DecodePixelFormat ( DDS\_CONTEXT \* ctx, ILuint \* CompFormat )
- 7.27.1.5 ILboolean Decompress3Dc ( DDS\_CONTEXT \* ctx )
- 7.27.1.6 ILboolean DecompressAti1n ( DDS\_CONTEXT \* ctx )
- 7.27.1.7 ILboolean DecompressDXT1 ( ILImage \* Image, ILubyte \* ICompData )
- 7.27.1.8 ILboolean DecompressDXT2 ( ILImage \* Image, ILubyte \* ICompData )
- 7.27.1.9 ILboolean DecompressDXT3 ( ILImage \* Image, ILubyte \* ICompData )
- 7.27.1.10 ILboolean DecompressDXT4 ( ILImage \* Image, ILubyte \* ICompData )
- 7.27.1.11 ILboolean DecompressDXT5 ( ILImage \* Image, ILubyte \* ICompData )
- 7.27.1.12 ILboolean DecompressRXGB ( DDS\_CONTEXT \* ctx )
- 7.27.1.13 void DxtcReadColor ( ILushort Data, Color8888 \* Out )
- 7.27.1.14 void DxtcReadColors ( const ILubyte \* Data, Color8888 \* Out )
- 7.27.1.15 void GetBitsFromMask ( ILuint Mask, ILuint \* ShiftLeft, ILuint \* ShiftRight )
- 7.27.1.16 unsigned int halfToFloat ( unsigned short y )
- 7.27.1.17 ILboolean iCheckDds ( DDSHEAD \* Head )
- 7.27.1.18 ILubyte iCompFormatToChannelCount ( IEnum Format )
- 7.27.1.19 void iComplexAlphaHelper ( ILubyte \* Data )
- 7.27.1.20 ILboolean iConvFloat16ToFloat32 ( ILuint \* dest, ILushort \* src, ILuint size )
- 7.27.1.21 ILboolean iConvG16R16ToFloat32 ( ILuint \* dest, ILushort \* src, ILuint size )
- 7.27.1.22 ILboolean iConvR16ToFloat32 ( ILuint \* dest, ILushort \* src, ILuint size )
- 7.27.1.23 ILAPI ILboolean ILAPIENTRY iDxtcDataToImage ( ILImage \* image )
- 7.27.1.24 ILAPI ILboolean ILAPIENTRY iDxtcDataToSurface ( ILImage \* image )

- 7.27.1.25 void iFlip3dc ( ILubyte \* *data*, ILuint *count* )
- 7.27.1.26 void iFlipColorBlock ( ILubyte \* *data* )
- 7.27.1.27 void iFlipComplexAlphaBlock ( ILubyte \* *Data* )
- 7.27.1.28 void iFlipDxt1 ( ILubyte \* *data*, ILuint *count* )
- 7.27.1.29 void iFlipDxt3 ( ILubyte \* *data*, ILuint *count* )
- 7.27.1.30 void iFlipDxt5 ( ILubyte \* *data*, ILuint *count* )
- 7.27.1.31 void iFlipSimpleAlphaBlock ( ILushort \* *data* )
- 7.27.1.32 void iFlipSurfaceDxtcData ( IImage \* *image* )
- 7.27.1.33 ILboolean iGetDdsHead ( SIO \* *io*, DDSHEAD \* *Header* )
- 7.27.1.34 ILboolean ilmageToDxtcData ( IImage \* *image*, IEnum *Format* )
- 7.27.1.35 void ilinvertDxt3Alpha ( ILubyte \* *data* )
- 7.27.1.36 void ilinvertDxt5Alpha ( ILubyte \* *data* )
- 7.27.1.37 ILboolean ilinvertSurfaceDxtcDataAlpha ( IImage \* *image* )
- 7.27.1.38 ILboolean ilsValidDds ( SIO \* *io* )
- 7.27.1.39 ILAPI void ILAPIENTRY ilFlipSurfaceDxtcData ( void )
- 7.27.1.40 void ilFreeImageDxtcData ( IImage \* *image* )
- 7.27.1.41 void ilFreeSurfaceDxtcData ( IImage \* *image* )
- 7.27.1.42 ILubyte\* ILAPIENTRY ilGetDxtcData ( IImage \* *image* )
- 7.27.1.43 ILAPI ILboolean ILAPIENTRY illmageToDxtcData ( IEnum *Format* )
- 7.27.1.44 ILAPI ILboolean ILAPIENTRY ilinvertSurfaceDxtcDataAlpha ( void )
- 7.27.1.45 ILboolean iLoadDdsCubemapInternal ( DDS\_CONTEXT \* *ctx*, ILuint *CompFormat* )
- 7.27.1.46 ILboolean iLoadDdsInternal ( IImage \* *image* )
- 7.27.1.47 ILAPI ILboolean ILAPIENTRY ilTexImageDxtc ( ILint *w*, ILint *h*, ILint *d*, IEnum *DxtFormat*, const ILubyte \* *data* )
- 7.27.1.48 ILAPI ILboolean ILAPIENTRY iSurfaceToDxtcData ( IImage \* *image*, IEnum *Format* )
- 7.27.1.49 ILboolean iTexImageDxtc ( IImage \* *image*, ILint *w*, ILint *h*, ILint *d*, IEnum *DxtFormat*, const ILubyte \* *data* )

## 7.27.2 Variable Documentation

## 7.27.2.1 ILformat iFormatDDS

**Initial value:**

```
= {
    .Validate = iIsValidDds,
    .Load     = iLoadDdsInternal,
    .Save     = iSaveDdsInternal,
    .Exts     = iFormatExtsDDS
}
```

## 7.27.2.2 ILconst\_string iFormatExtsDDS[]

**Initial value:**

```
= {
    IL_TEXT("dds"),
    NULL
}
```

## 7.28 src/IL/formats/il\_dds.h File Reference

```
#include "il_internal.h"
#include "pack_push.h"
#include "pack_pop.h"
```

**Data Structures**

- struct [Color565](#)
- struct [Color888](#)
- struct [Color8888](#)
- struct [DDSHEAD](#)
- struct [DXTAlphaBlock3BitLinear](#)
- struct [DXTAlphaBlockExplicit](#)
- struct [DXTColBlock](#)

**Macros**

- #define [CUBEMAP\\_SIDES](#) 6
- #define [DDS\\_ALPHA](#) 0x00000002L
- #define [DDS\\_ALPHAPIXELS](#) 0x00000001L
- #define [DDS\\_CAPS](#) 0x00000001L
- #define [DDS\\_COMPLEX](#) 0x00000008L
- #define [DDS\\_CUBEMAP](#) 0x00000200L
- #define [DDS\\_CUBEMAP\\_NEGATIVEX](#) 0x00000800L
- #define [DDS\\_CUBEMAP\\_NEGATIVEY](#) 0x00002000L
- #define [DDS\\_CUBEMAP\\_NEGATIVEZ](#) 0x00008000L
- #define [DDS\\_CUBEMAP\\_POSITIVEX](#) 0x00000400L
- #define [DDS\\_CUBEMAP\\_POSITIVEY](#) 0x00001000L
- #define [DDS\\_CUBEMAP\\_POSITIVEZ](#) 0x00004000L
- #define [DDS\\_DEPTH](#) 0x00800000L
- #define [DDS\\_FOURCC](#) 0x00000004L
- #define [DDS\\_HEIGHT](#) 0x00000002L
- #define [DDS\\_LINEAR\\_SIZE](#) 0x00080000L

- `#define DDS_LUMINANCE 0x00020000L`
- `#define DDS_MIPMAP 0x00400000L`
- `#define DDS_MIPMAPCOUNT 0x00020000L`
- `#define DDS_PITCH 0x00000008L`
- `#define DDS_PIXELFORMAT 0x00001000L`
- `#define DDS_RGB 0x00000040L`
- `#define DDS_TEXTURE 0x00001000L`
- `#define DDS_VOLUME 0x00200000L`
- `#define DDS_WIDTH 0x00000004L`
- `#define IL_MAKEFOURCC(ch0, ch1, ch2, ch3)`

## Typedefs

- `typedef struct Color565 Color565`
- `typedef struct Color888 Color888`
- `typedef struct Color8888 Color8888`
- `typedef struct DDSHEAD DDSHEAD`
- `typedef struct DXTAlphaBlock3BitLinear DXTAlphaBlock3BitLinear`
- `typedef struct DXTAlphaBlockExplicit DXTAlphaBlockExplicit`
- `typedef struct DXTColBlock DXTColBlock`

## Enumerations

- `enum PixFormat {  
PF_ARGB, PF_RGB, PF_DXT1, PF_DXT2,  
PF_DXT3, PF_DXT4, PF_DXT5, PF_3DC,  
PF_ATI1N, PF_LUMINANCE, PF_LUMINANCE_ALPHA, PF_RXGB,  
PF_A16B16G16R16, PF_R16F, PF_G16R16F, PF_A16B16G16R16F,  
PF_R32F, PF_G32R32F, PF_A32B32G32R32F, PF_UNKNOWN = 0xFF }`

## Functions

- `void ChooseAlphaEndpoints (ILubyte *Block, ILubyte *a0, ILubyte *a1)`
- `void ChooseEndpoints (ILushort *Block, ILushort *ex0, ILushort *ex1)`
- `ILushort Color565ToShort (Color565 *Colour)`
- `ILushort Color888ToShort (Color888 *Colour)`
- `ILuint Compress (ILImage *Image, ILenum DXTCFormat)`
- `ILushort * CompressTo565 (ILImage *Image)`
- `ILubyte * CompressTo88 (ILImage *Image)`
- `void CorrectEndDXT1 (ILushort *ex0, ILushort *ex1, ILboolean HasAlpha)`
- `void CorrectPreMult ()`
- `ILuint DecodePixelFormat ()`
- `ILboolean DecompressDXT1 (ILImage *Image, ILubyte *ICompData)`
- `ILboolean DecompressDXT2 (ILImage *Image, ILubyte *ICompData)`
- `ILboolean DecompressDXT3 (ILImage *Image, ILubyte *ICompData)`
- `ILboolean DecompressDXT4 (ILImage *Image, ILubyte *ICompData)`
- `ILboolean DecompressDXT5 (ILImage *Image, ILubyte *ICompData)`
- `ILuint Distance (Color888 *c1, Color888 *c2)`
- `void DxtcReadColor (ILushort Data, Color8888 *Out)`
- `void DxtcReadColors (const ILubyte *Data, Color8888 *Out)`
- `void GenAlphaBitMask (ILubyte a0, ILubyte a1, ILubyte *In, ILubyte *Mask, ILubyte *Out)`



- [ILuint GenBitMask](#) ([ILushort](#) ex0, [ILushort](#) ex1, [ILuint](#) NumCols, [ILushort](#) \*In, [ILubyte](#) \*Alpha, [Color888](#) \*OutCol)
- [ILboolean Get3DcBlock](#) ([ILubyte](#) \*Block, [ILubyte](#) \*Data, [ILimage](#) \*Image, [ILuint](#) XPos, [ILuint](#) YPos, int channel)
- [ILboolean GetAlphaBlock](#) ([ILubyte](#) \*Block, [ILubyte](#) \*Data, [ILimage](#) \*Image, [ILuint](#) XPos, [ILuint](#) YPos)
- [void GetBitsFromMask](#) ([ILuint](#) Mask, [ILuint](#) \*ShiftLeft, [ILuint](#) \*ShiftRight)
- [ILboolean GetBlock](#) ([ILushort](#) \*Block, [ILushort](#) \*Data, [ILimage](#) \*Image, [ILuint](#) XPos, [ILuint](#) YPos)
- [ILboolean iCheckDds](#) ([DDSHEAD](#) \*Head)
- [ILboolean iConvFloat16ToFloat32](#) ([ILuint](#) \*dest, [ILushort](#) \*src, [ILuint](#) size)
- [ILboolean iSaveDdsInternal](#) ([ILimage](#) \*)
- [void PreMult](#) ([ILushort](#) \*Data, [ILubyte](#) \*Alpha)
- [ILuint RMSAlpha](#) ([ILubyte](#) \*Orig, [ILubyte](#) \*Test)
- [void ShortToColor565](#) ([ILushort](#) Pixel, [Color565](#) \*Colour)
- [void ShortToColor888](#) ([ILushort](#) Pixel, [Color888](#) \*Colour)
- [ILboolean WriteHeader](#) ([ILimage](#) \*Image, [ILenum](#) DXTCFormat, [ILuint](#) CubeFlags)

## 7.28.1 Macro Definition Documentation

7.28.1.1 `#define CUBEMAP_SIDES 6`

7.28.1.2 `#define DDS_ALPHA 0x00000002L`

7.28.1.3 `#define DDS_ALPHAPIXELS 0x00000001L`

7.28.1.4 `#define DDS_CAPS 0x00000001L`

7.28.1.5 `#define DDS_COMPLEX 0x00000008L`

7.28.1.6 `#define DDS_CUBEMAP 0x00000200L`

7.28.1.7 `#define DDS_CUBEMAP_NEGATIVEX 0x00000800L`

7.28.1.8 `#define DDS_CUBEMAP_NEGATIVEY 0x00002000L`

7.28.1.9 `#define DDS_CUBEMAP_NEGATIVEZ 0x00008000L`

7.28.1.10 `#define DDS_CUBEMAP_POSITIVEX 0x00000400L`

7.28.1.11 `#define DDS_CUBEMAP_POSITIVEY 0x00001000L`

7.28.1.12 `#define DDS_CUBEMAP_POSITIVEZ 0x00004000L`

7.28.1.13 `#define DDS_DEPTH 0x00800000L`

7.28.1.14 `#define DDS_FOURCC 0x00000004L`

7.28.1.15 `#define DDS_HEIGHT 0x00000002L`

7.28.1.16 `#define DDS_LINEARSIZE 0x00080000L`

7.28.1.17 `#define DDS_LUMINANCE 0x00020000L`

7.28.1.18 `#define DDS_MIPMAP 0x00400000L`

7.28.1.19 `#define DDS_MIPMAPCOUNT 0x00020000L`

7.28.1.20 `#define DDS_PITCH 0x00000008L`

7.28.1.21 `#define DDS_PIXELFORMAT 0x00001000L`

7.28.1.22 `#define DDS_RGB 0x00000040L`

7.28.1.23 `#define DDS_TEXTURE 0x00001000L`

7.28.1.24 `#define DDS_VOLUME 0x00200000L`

7.28.1.25 `#define DDS_WIDTH 0x00000004L`

7.28.1.26 `#define IL_MAKEFOURCC( ch0, ch1, ch2, ch3 )`

**Value:**

```
((ILint)(ILbyte)(ch0) | ((ILint)(ILbyte)(ch1) << 8) | \
  ((ILint)(ILbyte)(ch2) << 16) | ((ILint)(ILbyte)(ch3) << 24 ))
```

## 7.28.2 Typedef Documentation

7.28.2.1 `typedef struct Color565 Color565`

7.28.2.2 `typedef struct Color888 Color888`

7.28.2.3 `typedef struct Color8888 Color8888`

7.28.2.4 `typedef struct DDSHEAD DDSHEAD`

7.28.2.5 `typedef struct DXAlphaBlock3BitLinear DXAlphaBlock3BitLinear`

7.28.2.6 `typedef struct DXAlphaBlockExplicit DXAlphaBlockExplicit`

7.28.2.7 `typedef struct DXTColBlock DXTColBlock`

## 7.28.3 Enumeration Type Documentation

7.28.3.1 `enum PixFormat`

Enumerator

***PF\_ARGB***

***PF\_RGB***

***PF\_DXT1***

***PF\_DXT2***

***PF\_DXT3***

***PF\_DXT4***

***PF\_DXT5***

***PF\_3DC***

***PF\_ATI1N***

***PF\_LUMINANCE***

***PF\_LUMINANCE\_ALPHA***

***PF\_RXGB***

***PF\_A16B16G16R16***

***PF\_R16F******PF\_G16R16F******PF\_A16B16G16R16F******PF\_R32F******PF\_G32R32F******PF\_A32B32G32R32F******PF\_UNKNOWN***

#### 7.28.4 Function Documentation

7.28.4.1 void ChooseAlphaEndpoints ( ILubyte \* *Block*, ILubyte \* *a0*, ILubyte \* *a1* )

7.28.4.2 void ChooseEndpoints ( ILushort \* *Block*, ILushort \* *ex0*, ILushort \* *ex1* )

7.28.4.3 ILushort Color565ToShort ( Color565 \* *Colour* )

7.28.4.4 ILushort Color888ToShort ( Color888 \* *Colour* )

7.28.4.5 ILuint Compress ( IImage \* *Image*, IEnum DXTCFormat )

7.28.4.6 ILushort\* CompressTo565 ( IImage \* *Image* )

7.28.4.7 ILubyte\* CompressTo88 ( IImage \* *Image* )

7.28.4.8 void CorrectEndDXT1 ( ILushort \* *ex0*, ILushort \* *ex1*, ILboolean *HasAlpha* )

7.28.4.9 void CorrectPreMult ( )

7.28.4.10 ILuint DecodePixelFormat ( )

7.28.4.11 ILboolean DecompressDXT1 ( IImage \* *Image*, ILubyte \* *ICompData* )

7.28.4.12 ILboolean DecompressDXT2 ( IImage \* *Image*, ILubyte \* *ICompData* )

7.28.4.13 ILboolean DecompressDXT3 ( IImage \* *Image*, ILubyte \* *ICompData* )

7.28.4.14 ILboolean DecompressDXT4 ( IImage \* *Image*, ILubyte \* *ICompData* )

7.28.4.15 ILboolean DecompressDXT5 ( IImage \* *Image*, ILubyte \* *ICompData* )

7.28.4.16 ILuint Distance ( Color888 \* *c1*, Color888 \* *c2* )

7.28.4.17 void DxtcReadColor ( ILushort *Data*, Color8888 \* *Out* )

7.28.4.18 void DxtcReadColors ( const ILubyte \* *Data*, Color8888 \* *Out* )

7.28.4.19 void GenAlphaBitMask ( ILubyte *a0*, ILubyte *a1*, ILubyte \* *In*, ILubyte \* *Mask*, ILubyte \* *Out* )

7.28.4.20 ILuint GenBitMask ( ILushort *ex0*, ILushort *ex1*, ILuint *NumCols*, ILushort \* *In*, ILubyte \* *Alpha*, Color888 \* *OutCol* )

7.28.4.21 ILboolean Get3DcBlock ( ILubyte \* *Block*, ILubyte \* *Data*, IImage \* *Image*, ILuint *XPos*, ILuint *YPos*, int *channel* )

- 7.28.4.22 `ILboolean GetAlphaBlock ( ILubyte * Block, ILubyte * Data, IImage * Image, ILuint XPos, ILuint YPos )`
- 7.28.4.23 `void GetBitsFromMask ( ILuint Mask, ILuint * ShiftLeft, ILuint * ShiftRight )`
- 7.28.4.24 `ILboolean GetBlock ( ILushort * Block, ILushort * Data, IImage * Image, ILuint XPos, ILuint YPos )`
- 7.28.4.25 `ILboolean iCheckDds ( DDSHEAD * Head )`
- 7.28.4.26 `ILboolean iConvFloat16ToFloat32 ( ILuint * dest, ILushort * src, ILuint size )`
- 7.28.4.27 `ILboolean iSaveDdsInternal ( IImage * )`
- 7.28.4.28 `void PreMult ( ILushort * Data, ILubyte * Alpha )`
- 7.28.4.29 `ILuint RMSAlpha ( ILubyte * Orig, ILubyte * Test )`
- 7.28.4.30 `void ShortToColor565 ( ILushort Pixel, Color565 * Colour )`
- 7.28.4.31 `void ShortToColor888 ( ILushort Pixel, Color888 * Colour )`
- 7.28.4.32 `ILboolean WriteHeader ( IImage * Image, IEnum DXTCFormat, ILuint CubeFlags )`

## 7.29 src/IL/formats/il\_dicom.c File Reference

```
#include "il_internal.h"
```

### Data Structures

- struct [DICOMHEAD](#)

### Typedefs

- typedef struct [DICOMHEAD](#) [DICOMHEAD](#)

### Functions

- [ILfloat](#) [GetFloat](#) ([SIO](#) \*io, [DICOMHEAD](#) \*Header, [ILushort](#) GroupNum)
- [ILuint](#) [GetGroupNum](#) ([SIO](#) \*io, [DICOMHEAD](#) \*Header)
- [ILuint](#) [GetInt](#) ([SIO](#) \*io, [DICOMHEAD](#) \*Header, [ILushort](#) GroupNum)
- [ILboolean](#) [GetNumericValue](#) ([SIO](#) \*io, [DICOMHEAD](#) \*Header, [ILushort](#) GroupNum, [ILuint](#) \*Number)
- [ILuint](#) [GetShort](#) ([SIO](#) \*io, [DICOMHEAD](#) \*Header, [ILushort](#) GroupNum)
- [ILboolean](#) [GetUID](#) ([SIO](#) \*io, [ILubyte](#) \*UID)
- [ILboolean](#) [iCheckDicom](#) ([DICOMHEAD](#) \*Header)
- [ILboolean](#) [iGetDicomHead](#) ([SIO](#) \*io, [DICOMHEAD](#) \*Header)
- [ILboolean](#) [SkipElement](#) ([SIO](#) \*io, [DICOMHEAD](#) \*Header, [ILushort](#) GroupNum, [ILushort](#) ElementNum)

### Variables

- [ILformat](#) [iFormatDICOM](#)
- [ILconst\\_string](#) [iFormatExtsDICOM](#) []

## 7.29.1 Typedef Documentation

### 7.29.1.1 typedef struct DICOMHEAD DICOMHEAD

## 7.29.2 Function Documentation

### 7.29.2.1 ILfloat GetFloat ( SIO \* io, DICOMHEAD \* Header, ILushort GroupNum )

### 7.29.2.2 ILuint GetGroupNum ( SIO \* io, DICOMHEAD \* Header )

### 7.29.2.3 ILuint GetInt ( SIO \* io, DICOMHEAD \* Header, ILushort GroupNum )

### 7.29.2.4 ILboolean GetNumericValue ( SIO \* io, DICOMHEAD \* Header, ILushort GroupNum, ILuint \* Number )

### 7.29.2.5 ILuint GetShort ( SIO \* io, DICOMHEAD \* Header, ILushort GroupNum )

### 7.29.2.6 ILboolean GetUID ( SIO \* io, ILubyte \* UID )

### 7.29.2.7 ILboolean iCheckDicom ( DICOMHEAD \* Header )

### 7.29.2.8 ILboolean iGetDicomHead ( SIO \* io, DICOMHEAD \* Header )

### 7.29.2.9 ILboolean SkipElement ( SIO \* io, DICOMHEAD \* Header, ILushort GroupNum, ILushort ElementNum )

## 7.29.3 Variable Documentation

### 7.29.3.1 ILformat iFormatDICOM

**Initial value:**

```
= {
    .Validate = iIsValidDicom,
    .Load     = iLoadDicomInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsDICOM
}
```

### 7.29.3.2 ILconst\_string iFormatExtsDICOM[]

**Initial value:**

```
= {
    IL_TEXT("dcm"),
    IL_TEXT("dicom"),
    NULL
}
```

## 7.30 src/IL/formats/il\_doom.c File Reference

```
#include "il_internal.h"
#include "il_manip.h"
#include "il_pal.h"
#include "il_doompal.h"
#include "pack_push.h"
#include "pack_pop.h"
```

## Data Structures

- struct [DOOM\\_HEAD](#)

## Variables

- [ILformat iFormatDOOM](#)
- [ILformat iFormatDOOM\\_FLAT](#)
- [ILconst\\_string iFormatExtsDOOM \[\]](#)

### 7.30.1 Variable Documentation

#### 7.30.1.1 [ILformat iFormatDOOM](#)

##### Initial value:

```
= {  
    .Validate = NULL,  
    .Load     = iLoadDoomInternal,  
    .Save     = NULL,  
    .Exts     = iFormatExtsDOOM  
}
```

#### 7.30.1.2 [ILformat iFormatDOOM\\_FLAT](#)

##### Initial value:

```
= {  
    .Validate = NULL,  
    .Load     = iLoadDoomFlatInternal,  
    .Save     = NULL,  
    .Exts     = iFormatExtsDOOM  
}
```

#### 7.30.1.3 [ILconst\\_string iFormatExtsDOOM\[\]](#)

##### Initial value:

```
= {  
    NULL  
}
```

## 7.31 [src/IL/formats/il\\_doompal.h](#) File Reference

### Macros

- [#define IL\\_DOOMPAL\\_SIZE 768](#)

### Variables

- [ILubyte iDefaultDoomPal \[IL\\_DOOMPAL\\_SIZE\]](#)

### 7.31.1 Macro Definition Documentation

#### 7.31.1.1 #define IL\_DOOMPAL\_SIZE 768

### 7.31.2 Variable Documentation

#### 7.31.2.1 ILubyte ilDefaultDoomPal[IL\_DOOMPAL\_SIZE]

## 7.32 src/IL/formats/il\_dpx.c File Reference

```
#include "il_internal.h"
#include "il_dpx.h"
```

### Variables

- [ILformat iFormatDPX](#)
- [ILconst\\_string iFormatExtsDPX \[\]](#)

### 7.32.1 Variable Documentation

#### 7.32.1.1 ILformat iFormatDPX

##### Initial value:

```
= {
    .Validate = iIsValidDpx,
    .Load     = iLoadDpxInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsDPX
}
```

#### 7.32.1.2 ILconst\_string iFormatExtsDPX[]

##### Initial value:

```
= {
    IL_TEXT("dpx"),
    NULL
}
```

## 7.33 src/IL/formats/il\_dpx.h File Reference

```
#include "il_internal.h"
#include "pack_push.h"
#include "pack_pop.h"
```

### Data Structures

- struct [DPX\\_FILE\\_INFO](#)
- struct [DPX\\_IMAGE\\_ELEMENT](#)
- struct [DPX\\_IMAGE\\_INFO](#)
- struct [DPX\\_IMAGE\\_ORIENT](#)

- struct [DPX\\_MOTION\\_PICTURE\\_HEAD](#)
- struct [DPX\\_TELEVISION\\_HEAD](#)
- struct [R32](#)

## Typedefs

- typedef struct [DPX\\_FILE\\_INFO](#) [DPX\\_FILE\\_INFO](#)
- typedef struct [DPX\\_IMAGE\\_ELEMENT](#) [DPX\\_IMAGE\\_ELEMENT](#)
- typedef struct [DPX\\_IMAGE\\_INFO](#) [DPX\\_IMAGE\\_INFO](#)
- typedef struct [DPX\\_IMAGE\\_ORIENT](#) [DPX\\_IMAGE\\_ORIENT](#)
- typedef struct [DPX\\_MOTION\\_PICTURE\\_HEAD](#) [DPX\\_MOTION\\_PICTURE\\_HEAD](#)
- typedef struct [DPX\\_TELEVISION\\_HEAD](#) [DPX\\_TELEVISION\\_HEAD](#)
- typedef struct [R32](#) [R32](#)

### 7.33.1 Typedef Documentation

7.33.1.1 typedef struct [DPX\\_FILE\\_INFO](#) [DPX\\_FILE\\_INFO](#)

7.33.1.2 typedef struct [DPX\\_IMAGE\\_ELEMENT](#) [DPX\\_IMAGE\\_ELEMENT](#)

7.33.1.3 typedef struct [DPX\\_IMAGE\\_INFO](#) [DPX\\_IMAGE\\_INFO](#)

7.33.1.4 typedef struct [DPX\\_IMAGE\\_ORIENT](#) [DPX\\_IMAGE\\_ORIENT](#)

7.33.1.5 typedef struct [DPX\\_MOTION\\_PICTURE\\_HEAD](#) [DPX\\_MOTION\\_PICTURE\\_HEAD](#)

7.33.1.6 typedef struct [DPX\\_TELEVISION\\_HEAD](#) [DPX\\_TELEVISION\\_HEAD](#)

7.33.1.7 typedef struct [R32](#) [R32](#)

## 7.34 src/IL/formats/il\_exr.c File Reference

```
#include "il_internal.h"
#include "il_exr.h"
#include <ImfRgba.h>
#include <ImfArray.h>
#include <ImfRgbaFile.h>
```

## Macros

- #define [HALF\\_EXPORTS](#)

## Functions

- [ILboolean iCheckExr](#) ([EXRHEAD](#) \*Header)
- [ILboolean iGetExrHead](#) ([EXRHEAD](#) \*Header)
- [ILboolean ilsValidExr](#) ()
- [ILboolean illsValidExr](#) ([ILconst\\_string](#) FileName)  
*Checks if the file specified in FileName is a valid EXR file.*
- [ILboolean illsValidExrF](#) ([ILHANDLE](#) File)



*Checks if the ILHANDLE contains a valid EXR file at the current position.*

- [ILboolean](#) [illsValidExrL](#) (const [void](#) \*Lump, [ILuint](#) Size)

*Checks if Lump is a valid EXR lump.*

- [ILboolean](#) [iLoadExrInternal](#) ()
- [ILboolean](#) [iSaveExrInternal](#) ()

### 7.34.1 Macro Definition Documentation

#### 7.34.1.1 #define HALF\_EXPORTS

### 7.34.2 Function Documentation

#### 7.34.2.1 [ILboolean](#) [iCheckExr](#) ( [EXRHEAD](#) \* Header )

#### 7.34.2.2 [ILboolean](#) [iGetExrHead](#) ( [EXRHEAD](#) \* Header )

#### 7.34.2.3 [ILboolean](#) [illsValidExr](#) ( )

#### 7.34.2.4 [ILboolean](#) [illsValidExr](#) ( [ILconst\\_string](#) FileName )

Checks if the file specified in FileName is a valid EXR file.

#### 7.34.2.5 [ILboolean](#) [illsValidExrF](#) ( [ILHANDLE](#) File )

Checks if the ILHANDLE contains a valid EXR file at the current position.

#### 7.34.2.6 [ILboolean](#) [illsValidExrL](#) ( const [void](#) \* Lump, [ILuint](#) Size )

Checks if Lump is a valid EXR lump.

#### 7.34.2.7 [ILboolean](#) [iLoadExrInternal](#) ( )

#### 7.34.2.8 [ILboolean](#) [iSaveExrInternal](#) ( )

## 7.35 src/IL/formats/il\_exr.h File Reference

```
#include "il_internal.h"
#include <ImfIO.h>
```

### Data Structures

- struct [EXRHEAD](#)
- class [ilStream](#)
- class [ilOStream](#)

### Macros

- #define [EXR\\_B44\\_COMPRESSION](#) 6
- #define [EXR\\_B44A\\_COMPRESSION](#) 7
- #define [EXR\\_FLOAT](#) 2

- `#define EXR_HALF 1`
- `#define EXR_NO_COMPRESSION 0`
- `#define EXR_PIZ_COMPRESSION 4`
- `#define EXR_PXR24_COMPRESSION 5`
- `#define EXR_RLE_COMPRESSION 1`
- `#define EXR_UINT 0`
- `#define EXR_ZIP_COMPRESSION 3`
- `#define EXR_ZIPS_COMPRESSION 2`

## Typedefs

- `typedef struct EXRHEAD EXRHEAD`

## Functions

- `ILboolean iCheckExr (EXRHEAD *Header)`
- `ILboolean ilsValidExr ()`
- `ILboolean iLoadExrInternal ()`
- `ILboolean iSaveExrInternal ()`

### 7.35.1 Macro Definition Documentation

7.35.1.1 `#define EXR_B44_COMPRESSION 6`

7.35.1.2 `#define EXR_B44A_COMPRESSION 7`

7.35.1.3 `#define EXR_FLOAT 2`

7.35.1.4 `#define EXR_HALF 1`

7.35.1.5 `#define EXR_NO_COMPRESSION 0`

7.35.1.6 `#define EXR_PIZ_COMPRESSION 4`

7.35.1.7 `#define EXR_PXR24_COMPRESSION 5`

7.35.1.8 `#define EXR_RLE_COMPRESSION 1`

7.35.1.9 `#define EXR_UINT 0`

7.35.1.10 `#define EXR_ZIP_COMPRESSION 3`

7.35.1.11 `#define EXR_ZIPS_COMPRESSION 2`

### 7.35.2 Typedef Documentation

7.35.2.1 `typedef struct EXRHEAD EXRHEAD`

### 7.35.3 Function Documentation

7.35.3.1 `ILboolean iCheckExr ( EXRHEAD * Header )`

7.35.3.2 `ILboolean ilsValidExr ( )`

7.35.3.3 ILboolean iLoadExrInternal ( )

7.35.3.4 ILboolean iSaveExrInternal ( )

## 7.36 src/IL/formats/il\_fits.c File Reference

```
#include "il_internal.h"
```

### Data Structures

- struct [FITSHEAD](#)

### Typedefs

- typedef struct [FITSHEAD](#) [FITSHEAD](#)

### Enumerations

- enum {  
    [CARD\\_READ\\_FAIL](#) = -1, [CARD\\_END](#) = 1, [CARD\\_SIMPLE](#), [CARD\\_NOT\\_SIMPLE](#),  
    [CARD\\_BITPIX](#), [CARD\\_NUMAXES](#), [CARD\\_AXIS](#), [CARD\\_SKIP](#) }

### Variables

- [ILconst\\_string](#) iFormatExtsFITS []
- [ILformat](#) iFormatFITS

### 7.36.1 Typedef Documentation

7.36.1.1 typedef struct [FITSHEAD](#) [FITSHEAD](#)

### 7.36.2 Enumeration Type Documentation

7.36.2.1 anonymous enum

#### Enumerator

***CARD\_READ\_FAIL***

***CARD\_END***

***CARD\_SIMPLE***

***CARD\_NOT\_SIMPLE***

***CARD\_BITPIX***

***CARD\_NUMAXES***

***CARD\_AXIS***

***CARD\_SKIP***

### 7.36.3 Variable Documentation

#### 7.36.3.1 ILconst\_string iFormatExtsFITS[]

**Initial value:**

```
= {  
    IL_TEXT("fit"),  
    IL_TEXT("fits"),  
    NULL  
}
```

#### 7.36.3.2 ILformat iFormatFITS

**Initial value:**

```
= {  
    .Validate = iIsValidFits,  
    .Load     = iLoadFitsInternal,  
    .Save     = NULL,  
    .Exts     = iFormatExtsFITS  
}
```

## 7.37 src/IL/formats/il\_ftx.c File Reference

```
#include "il_internal.h"  
#include "pack_push.h"  
#include "pack_pop.h"
```

### Data Structures

- struct [FTX\\_HEAD](#)

### Variables

- [ILconst\\_string iFormatExtsFTX \[\]](#)
- [ILformat iFormatFTX](#)

### 7.37.1 Variable Documentation

#### 7.37.1.1 ILconst\_string iFormatExtsFTX[]

**Initial value:**

```
= {  
    IL_TEXT("ftx"),  
    NULL  
}
```

#### 7.37.1.2 ILformat iFormatFTX

**Initial value:**

```
= {
    .Validate = NULL,
    .Load     = iLoadFtxInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsFTX
}
```

## 7.38 src/IL/formats/il\_gif.c File Reference

```
#include "il_internal.h"
#include "il_gif.h"
#include <stdio.h>
```

### Functions

- [ILboolean ilsValidGif](#) ([SIO](#) \*io)
- [ILboolean iLoadGifInternal](#) ([ILImage](#) \*TargetImage)

### Variables

- [ILconst\\_string iFormatExtsGIF](#) []
- [ILformat iFormatGIF](#)

#### 7.38.1 Function Documentation

7.38.1.1 [ILboolean ilsValidGif](#) ( [SIO](#) \* *io* )

7.38.1.2 [ILboolean iLoadGifInternal](#) ( [ILImage](#) \* *TargetImage* )

#### 7.38.2 Variable Documentation

7.38.2.1 [ILconst\\_string iFormatExtsGIF](#) []

##### Initial value:

```
= {
    IL\_TEXT("gif"),
    NULL
}
```

##### 7.38.2.2 [ILformat iFormatGIF](#)

##### Initial value:

```
= {
    .Validate = iIsValidGif,
    .Load     = iLoadGifInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsGIF
}
```

## 7.39 src/IL/formats/il\_gif.h File Reference

```
#include "il_internal.h"
```

## Data Structures

- struct [GifGraphicControlExtension](#)
- struct [GifImageDescriptor](#)
- struct [GifLoadingContext](#)
- struct [GifLogicalScreenDescriptor](#)
- struct [GifSignature](#)
- struct [LZWInputStream](#)

## Macros

- `#define` [GIF\\_VERSION87A](#) "GIF87a"
- `#define` [GIF\\_VERSION89A](#) "GIF89a"

## Typedefs

- typedef struct [GifGraphicControlExtension](#) [GifGraphicControlExtension](#)
- typedef struct [GifImageDescriptor](#) [GifImageDescriptor](#)
- typedef struct [GifLoadingContext](#) [GifLoadingContext](#)
- typedef struct [GifLogicalScreenDescriptor](#) [GifLogicalScreenDescriptor](#)
- typedef struct [GifSignature](#) [GifSignature](#)
- typedef struct [LZWInputStream](#) [LZWInputStream](#)

## Enumerations

- enum {  
[GifFlag\\_LSD\\_GlobalColorTableSizeMask](#) = (7<<0), [GifFlag\\_LSD\\_Sort](#) = (1<<3), [GifFlag\\_LSD\\_Color-ResolutionMask](#) = (7<<4), [GifFlag\\_LSD\\_HasGlobalColorTable](#) = (1<<7),  
[GifFlag\\_GCE\\_TransparentColor](#) = (1<<0), [GifFlag\\_GCE\\_UserInput](#) = (1<<1), [GifFlag\\_GCE\\_Disposal-MethodMask](#) = (7<<2), [GifFlag\\_GCE\\_DisposalMethodShift](#) = 2,  
[GifFlag\\_GCE\\_ReservedMask](#) = (3<<5), [GifFlag\\_GCE\\_ReservedShift](#) = 5, [GifFlag\\_IMG\\_LocalColorTable-SizeMask](#) = (7<<0), [GifFlag\\_IMG\\_ReservedMask](#) = (3<<3),  
[GifFlag\\_IMG\\_Sort](#) = (1<<5), [GifFlag\\_IMG\\_Interlaced](#) = (1<<6), [GifFlag\\_IMG\\_HasLocalColorTable](#) = (1<<7), [GifID\\_Terminator](#) = 0x00,  
[GifID\\_Extension](#) = 0x21, [GifID\\_Image](#) = 0x2c, [GifID\\_End](#) = 0x3b, [GifExt\\_PlainText](#) = 0x01,  
[GifExt\\_GraphicControl](#) = 0xf9, [GifExt\\_Comment](#) = 0xfe, [GifExt\\_Application](#) = 0xff, [GifDisposal\\_DontCare](#) = 0,  
[GifDisposal\\_Overlay](#) = 1, [GifDisposal\\_Clear](#) = 2, [GifDisposal\\_Restore](#) = 3 }

### 7.39.1 Macro Definition Documentation

7.39.1.1 `#define` [GIF\\_VERSION87A](#) "GIF87a"

7.39.1.2 `#define` [GIF\\_VERSION89A](#) "GIF89a"

### 7.39.2 Typedef Documentation

7.39.2.1 typedef struct [GifGraphicControlExtension](#) [GifGraphicControlExtension](#)

7.39.2.2 typedef struct [GifImageDescriptor](#) [GifImageDescriptor](#)

7.39.2.3 typedef struct GifLoadingContext GifLoadingContext

7.39.2.4 typedef struct GifLogicalScreenDescriptor GifLogicalScreenDescriptor

7.39.2.5 typedef struct GifSignature GifSignature

7.39.2.6 typedef struct LZWInputStream LZWInputStream

### 7.39.3 Enumeration Type Documentation

7.39.3.1 anonymous enum

Enumerator

*GifFlag\_LSD\_GlobalColorTableSizeMask*

*GifFlag\_LSD\_Sort*

*GifFlag\_LSD\_ColorResolutionMask*

*GifFlag\_LSD\_HasGlobalColorTable*

*GifFlag\_GCE\_TransparentColor*

*GifFlag\_GCE\_UserInput*

*GifFlag\_GCE\_DisposalMethodMask*

*GifFlag\_GCE\_DisposalMethodShift*

*GifFlag\_GCE\_ReservedMask*

*GifFlag\_GCE\_ReservedShift*

*GifFlag\_IMG\_LocalColorTableSizeMask*

*GifFlag\_IMG\_ReservedMask*

*GifFlag\_IMG\_Sort*

*GifFlag\_IMG\_Interlaced*

*GifFlag\_IMG\_HasLocalColorTable*

*GifID\_Terminator*

*GifID\_Extension*

*GifID\_Image*

*GifID\_End*

*GifExt\_PlainText*

*GifExt\_GraphicControl*

*GifExt\_Comment*

*GifExt\_Application*

*GifDisposal\_DontCare*

*GifDisposal\_Overlay*

*GifDisposal\_Clear*

*GifDisposal\_Restore*

## 7.40 src/IL/formats/il\_hdr.c File Reference

```
#include "il_internal.h"
#include "il_endian.h"
#include <string.h>
```

## Data Structures

- struct [rgbe\\_header\\_info](#)

## Macros

- #define [MINRUNLENGTH](#) 4
- #define [RGBE\\_DATA\\_BLUE](#) 2
- #define [RGBE\\_DATA\\_GREEN](#) 1
- #define [RGBE\\_DATA\\_RED](#) 0
- #define [RGBE\\_DATA\\_SIZE](#) 3
- #define [RGBE\\_VALID\\_EXPOSURE](#) 0x04
- #define [RGBE\\_VALID\\_GAMMA](#) 0x02
- #define [RGBE\\_VALID\\_PROGRAMTYPE](#) 0x01

## Functions

- void [ReadScanline](#) (ILImage \*image, ILubyte \*scanline, ILuint w)
- ILboolean [RGBE\\_WriteBytes\\_RLE](#) (ILImage \*image, ILubyte \*data, ILuint numbytes)
- ILboolean [RGBE\\_WriteHeader](#) (SIO \*io, ILuint width, ILuint height, [rgbe\\_header\\_info](#) \*info)
- int [RGBE\\_WritePixels](#) (ILImage \*image, float \*data, int numpixels)
- const char \* [strnstr](#) (const char \*bigstr, const char \*substr, size\_t max)

## Variables

- ILconst\_string [iFormatExtsHDR](#) []
- ILformat [iFormatHDR](#)

### 7.40.1 Macro Definition Documentation

7.40.1.1 #define [MINRUNLENGTH](#) 4

7.40.1.2 #define [RGBE\\_DATA\\_BLUE](#) 2

7.40.1.3 #define [RGBE\\_DATA\\_GREEN](#) 1

7.40.1.4 #define [RGBE\\_DATA\\_RED](#) 0

7.40.1.5 #define [RGBE\\_DATA\\_SIZE](#) 3

7.40.1.6 #define [RGBE\\_VALID\\_EXPOSURE](#) 0x04

7.40.1.7 #define [RGBE\\_VALID\\_GAMMA](#) 0x02

7.40.1.8 #define [RGBE\\_VALID\\_PROGRAMTYPE](#) 0x01

### 7.40.2 Function Documentation

7.40.2.1 void [ReadScanline](#) ( ILImage \* *image*, ILubyte \* *scanline*, ILuint *w* )

7.40.2.2 ILboolean [RGBE\\_WriteBytes\\_RLE](#) ( ILImage \* *image*, ILubyte \* *data*, ILuint *numbytes* )

7.40.2.3 ILboolean [RGBE\\_WriteHeader](#) ( SIO \* *io*, ILuint *width*, ILuint *height*, [rgbe\\_header\\_info](#) \* *info* )



7.40.2.4 `int RGBE_WritePixels ( IImage * image, float * data, int numpixels )`

7.40.2.5 `const char* strnstr ( const char * bigstr, const char * substr, size_t max )`

### 7.40.3 Variable Documentation

#### 7.40.3.1 ILconst\_string iFormatExtsHDR[]

**Initial value:**

```
= {
    IL_TEXT("hdr"),
    NULL
}
```

#### 7.40.3.2 ILformat iFormatHDR

**Initial value:**

```
= {
    .Validate = iIsValidHdr,
    .Load     = iLoadHdrInternal,
    .Save     = iSaveHdrInternal,
    .Exts     = iFormatExtsHDR
}
```

## 7.41 src/IL/formats/il\_header.c File Reference

```
#include "il_internal.h"
```

### Macros

- `#define MAX_LINE_WIDTH 14`

### Variables

- `ILformat iFormatCHEAD`
- `ILconst_string iFormatExtsCHEAD []`

### 7.41.1 Macro Definition Documentation

#### 7.41.1.1 `#define MAX_LINE_WIDTH 14`

### 7.41.2 Variable Documentation

#### 7.41.2.1 ILformat iFormatCHEAD

**Initial value:**

```
= {
    .Validate = NULL,
    .Load     = NULL,
    .Save     = iSaveCHEADInternal,
    .Exts     = iFormatExtsCHEAD
}
```

### 7.41.2.2 ILconst\_string iFormatExtsCHEAD[]

**Initial value:**

```
= {
    IL_TEXT("h"),
    NULL
}
```

## 7.42 src/IL/formats/il\_icns.c File Reference

```
#include "il_internal.h"
#include "il_icns.h"
#include "il_jp2.h"
```

### Functions

- [ILboolean ilcnsReadData](#) ([ILImage](#) \*image, [ILboolean](#) \*BaseCreated, [ILboolean](#) IsAlpha, [ILint](#) Width, [ICNS-DATA](#) \*Entry, [ILImage](#) \*\*Image)

### Variables

- [ILconst\\_string iFormatExtsICNS](#) []
- [ILformat iFormatICNS](#)

### 7.42.1 Function Documentation

**7.42.1.1** [ILboolean ilcnsReadData](#) ( [ILImage](#) \* *image*, [ILboolean](#) \* *BaseCreated*, [ILboolean](#) *IsAlpha*, [ILint](#) *Width*, [ICNSDATA](#) \* *Entry*, [ILImage](#) \*\* *Image* )

### 7.42.2 Variable Documentation

#### 7.42.2.1 ILconst\_string iFormatExtsICNS[]

**Initial value:**

```
= {
    IL_TEXT("icns"),
    NULL
}
```

#### 7.42.2.2 ILformat iFormatICNS

**Initial value:**

```
= {
    .Validate = iIsValidIcns,
    .Load     = iLoadIcnsInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsICNS
}
```

## 7.43 src/IL/formats/il\_icns.h File Reference

```
#include "il_internal.h"
#include "pack_push.h"
#include "pack_pop.h"
```

### Data Structures

- struct [ICNSDATA](#)
- struct [ICNSHEAD](#)

### Typedefs

- typedef struct [ICNSDATA](#) [ICNSDATA](#)
- typedef struct [ICNSHEAD](#) [ICNSHEAD](#)

### Functions

- [ILboolean](#) [ilcnsReadData](#) ([ILImage](#) \*image, [ILboolean](#) \*BaseCreated, [ILboolean](#) IsAlpha, [ILint](#) Width, [ICNSDATA](#) \*Entry, [ILImage](#) \*\*Image)

#### 7.43.1 Typedef Documentation

7.43.1.1 typedef struct [ICNSDATA](#) [ICNSDATA](#)

7.43.1.2 typedef struct [ICNSHEAD](#) [ICNSHEAD](#)

#### 7.43.2 Function Documentation

7.43.2.1 [ILboolean](#) [ilcnsReadData](#) ( [ILImage](#) \* *image*, [ILboolean](#) \* *BaseCreated*, [ILboolean](#) *IsAlpha*, [ILint](#) *Width*, [ICNSDATA](#) \* *Entry*, [ILImage](#) \*\* *Image* )

## 7.44 src/IL/formats/il\_icon.c File Reference

```
#include "il_internal.h"
#include "il_icon.h"
#include <png.h>
```

### Data Structures

- struct [IconData](#)
- struct [ICONDIR](#)
- struct [ICONDIRENTRY](#)

### Typedefs

- typedef [ILuchar](#) [BYTE](#)
- typedef [ILuint](#) [DWORD](#)
- typedef struct [ICONDIR](#) [ICONDIR](#)

- typedef struct [ICONDIRENTRY](#) [ICONDIRENTRY](#)
- typedef [ILuchar](#) \* [LPBYTE](#)
- typedef [ILuint](#) \* [LPDWORD](#)
- typedef [ILushort](#) \* [LPWORD](#)
- typedef [ILushort](#) [WORD](#)

## Variables

- [ILconst\\_string](#) [iFormatExtsICO](#) []
- [ILformat](#) [iFormatICO](#)

### 7.44.1 Typedef Documentation

7.44.1.1 [typedef](#) [ILuchar](#) [BYTE](#)

7.44.1.2 [typedef](#) [ILuint](#) [DWORD](#)

7.44.1.3 [typedef](#) struct [ICONDIR](#) [ICONDIR](#)

7.44.1.4 [typedef](#) struct [ICONDIRENTRY](#) [ICONDIRENTRY](#)

7.44.1.5 [typedef](#) [ILuchar](#) \* [LPBYTE](#)

7.44.1.6 [typedef](#) [ILuint](#) \* [LPDWORD](#)

7.44.1.7 [typedef](#) [ILushort](#) \* [LPWORD](#)

7.44.1.8 [typedef](#) [ILushort](#) [WORD](#)

### 7.44.2 Variable Documentation

7.44.2.1 [ILconst\\_string](#) [iFormatExtsICO](#) []

**Initial value:**

```
= {
    IL\_TEXT("ico"),
    IL\_TEXT("cur"),
    NULL
}
```

7.44.2.2 [ILformat](#) [iFormatICO](#)

**Initial value:**

```
= {
    .Validate = iIsValidIcon,
    .Load     = iLoadIconInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsICO
}
```

## 7.45 src/IL/formats/il\_icon.h File Reference

```
#include "il_internal.h"
#include "pack_push.h"
#include "pack_pop.h"
```

### Data Structures

- struct [ICODIR](#)
- struct [ICODIRENTRY](#)
- struct [ICOIMAGE](#)
- struct [INFOHEAD](#)

### Typedefs

- typedef struct [ICODIR](#) [ICODIR](#)
- typedef struct [ICODIRENTRY](#) [ICODIRENTRY](#)
- typedef struct [ICOIMAGE](#) [ICOIMAGE](#)
- typedef struct [INFOHEAD](#) [INFOHEAD](#)

#### 7.45.1 Typedef Documentation

7.45.1.1 typedef struct [ICODIR](#) [ICODIR](#)

7.45.1.2 typedef struct [ICODIRENTRY](#) [ICODIRENTRY](#)

7.45.1.3 typedef struct [ICOIMAGE](#) [ICOIMAGE](#)

7.45.1.4 typedef struct [INFOHEAD](#) [INFOHEAD](#)

## 7.46 src/IL/formats/il\_iff.c File Reference

```
#include "il_internal.h"
```

### Data Structures

- struct [\\_iff\\_chunk](#)
- struct [iff\\_chunk\\_stack](#)

### Macros

- #define [ALPHA\\_FLAG](#) (2)
- #define [CHUNK\\_STACK\\_SIZE](#) (32)
- #define [RGB\\_FLAG](#) (1)
- #define [ZBUFFER\\_FLAG](#) (4)

### Typedefs

- typedef struct [\\_iff\\_chunk](#) [iff\\_chunk](#)

## Variables

- const ILuint IFF\_TAG\_BLRT = ('B' << 24) | ('L' << 16) | ('R' << 8) | ('T')
- const ILuint IFF\_TAG\_BLUR = ('B' << 24) | ('L' << 16) | ('U' << 8) | ('R')
- const ILuint IFF\_TAG\_CIMG = ('C' << 24) | ('I' << 16) | ('M' << 8) | ('G')
- const ILuint IFF\_TAG\_CLPZ = ('C' << 24) | ('L' << 16) | ('P' << 8) | ('Z')
- const ILuint IFF\_TAG\_ESXY = ('E' << 24) | ('S' << 16) | ('X' << 8) | ('Y')
- const ILuint IFF\_TAG\_FOR4 = ('F' << 24) | ('O' << 16) | ('R' << 8) | ('4')
- const ILuint IFF\_TAG\_HIST = ('H' << 24) | ('I' << 16) | ('S' << 8) | ('T')
- const ILuint IFF\_TAG\_RGBA = ('R' << 24) | ('G' << 16) | ('B' << 8) | ('A')
- const ILuint IFF\_TAG\_TBHD = ('T' << 24) | ('B' << 16) | ('H' << 8) | ('D')
- const ILuint IFF\_TAG\_TBMP = ('T' << 24) | ('B' << 16) | ('M' << 8) | ('P')
- const ILuint IFF\_TAG\_ZBUF = ('Z' << 24) | ('B' << 16) | ('U' << 8) | ('F')
- ILconst\_string iFormatExtsIFF []
- ILformat iFormatIFF

## 7.46.1 Macro Definition Documentation

7.46.1.1 #define ALPHA\_FLAG (2)

7.46.1.2 #define CHUNK\_STACK\_SIZE (32)

7.46.1.3 #define RGB\_FLAG (1)

7.46.1.4 #define ZBUFFER\_FLAG (4)

## 7.46.2 Typedef Documentation

7.46.2.1 typedef struct \_iff\_chunk iff\_chunk

## 7.46.3 Variable Documentation

7.46.3.1 const ILuint IFF\_TAG\_BLRT = ('B' << 24) | ('L' << 16) | ('R' << 8) | ('T')

7.46.3.2 const ILuint IFF\_TAG\_BLUR = ('B' << 24) | ('L' << 16) | ('U' << 8) | ('R')

7.46.3.3 const ILuint IFF\_TAG\_CIMG = ('C' << 24) | ('I' << 16) | ('M' << 8) | ('G')

7.46.3.4 const ILuint IFF\_TAG\_CLPZ = ('C' << 24) | ('L' << 16) | ('P' << 8) | ('Z')

7.46.3.5 const ILuint IFF\_TAG\_ESXY = ('E' << 24) | ('S' << 16) | ('X' << 8) | ('Y')

7.46.3.6 const ILuint IFF\_TAG\_FOR4 = ('F' << 24) | ('O' << 16) | ('R' << 8) | ('4')

7.46.3.7 const ILuint IFF\_TAG\_HIST = ('H' << 24) | ('I' << 16) | ('S' << 8) | ('T')

7.46.3.8 const ILuint IFF\_TAG\_RGBA = ('R' << 24) | ('G' << 16) | ('B' << 8) | ('A')

7.46.3.9 const ILuint IFF\_TAG\_TBHD = ('T' << 24) | ('B' << 16) | ('H' << 8) | ('D')

7.46.3.10 const ILuint IFF\_TAG\_TBMP = ('T' << 24) | ('B' << 16) | ('M' << 8) | ('P')

7.46.3.11 const ILuint IFF\_TAG\_ZBUF = ('Z' << 24) | ('B' << 16) | ('U' << 8) | ('F')

## 7.46.3.12 ILconst\_string iFormatExtsIFF[]

**Initial value:**

```
= {
    IL_TEXT("iff"),
    NULL
}
```

## 7.46.3.13 ILformat iFormatIFF

**Initial value:**

```
= {
    .Validate = iIsValidIff,
    .Load     = iLoadIffInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsIFF
}
```

## 7.47 src/IL/formats/il\_ilbm.c File Reference

```
#include "il_internal.h"
#include <stdlib.h>
#include "pack_push.h"
#include "pack_pop.h"
```

**Data Structures**

- struct [BMHD](#)

**Macros**

- #define [MAXCOLORS](#) 256

**Variables**

- [ILconst\\_string iFormatExtsILBM \[\]](#)
- [ILformat iFormatILBM](#)

## 7.47.1 Macro Definition Documentation

## 7.47.1.1 #define MAXCOLORS 256

## 7.47.2 Variable Documentation

## 7.47.2.1 ILconst\_string iFormatExtsILBM[]

**Initial value:**

```
= {
    IL_TEXT("ilbm"),
    IL_TEXT("lbm"),
    IL_TEXT("ham"),
    NULL
}
```

### 7.47.2.2 ILformat iFormatILBM

#### Initial value:

```
= {  
    .Validate = iIsValidIlbm,  
    .Load     = iLoadIlbmInternal,  
    .Save     = NULL,  
    .Exts     = iFormatExtsILBM  
}
```

## 7.48 src/IL/formats/il\_iwi.c File Reference

```
#include "il_internal.h"  
#include "il_dds.h"  
#include "pack_push.h"  
#include "pack_pop.h"
```

### Data Structures

- struct [IWIHEAD](#)

### Macros

- #define [IWI\\_A8](#) 0x04
- #define [IWI\\_ARGB4](#) 0x03
- #define [IWI\\_ARGB8](#) 0x01
- #define [IWI\\_DXT1](#) 0x0B
- #define [IWI\\_DXT3](#) 0x0C
- #define [IWI\\_DXT5](#) 0x0D
- #define [IWI\\_JPG](#) 0x07
- #define [IWI\\_RGB8](#) 0x02

### Typedefs

- typedef struct [IWIHEAD](#) [IWIHEAD](#)

### Variables

- [ILconst\\_string](#) [iFormatExtsIWI](#) []
- [ILformat](#) [iFormatIWI](#)

## 7.48.1 Macro Definition Documentation

7.48.1.1 #define [IWI\\_A8](#) 0x04

7.48.1.2 #define [IWI\\_ARGB4](#) 0x03

7.48.1.3 #define [IWI\\_ARGB8](#) 0x01

7.48.1.4 #define [IWI\\_DXT1](#) 0x0B



7.48.1.5 `#define IWI_DXT3 0x0C`

7.48.1.6 `#define IWI_DXT5 0x0D`

7.48.1.7 `#define IWI_JPG 0x07`

7.48.1.8 `#define IWI_RGB8 0x02`

## 7.48.2 Typedef Documentation

7.48.2.1 `typedef struct IWIHEAD IWIHEAD`

## 7.48.3 Variable Documentation

7.48.3.1 `ILconst_string iFormatExtsIWI[]`

**Initial value:**

```
= {
    IL_TEXT("iwi"),
    NULL
}
```

7.48.3.2 `ILformat iFormatIWI`

**Initial value:**

```
= {
    .Validate = iIsValidIwi,
    .Load     = iLoadIwiInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsIWI
}
```

## 7.49 src/IL/formats/il\_jp2.c File Reference

```
#include "il_internal.h"
#include <jasper/jasper.h>
#include "il_jp2.h"
```

### Functions

- `jas_stream_t * iJp2ReadStream (SIO *io)`
- `jas_stream_t * iJp2WriteStream ()`
- `ILboolean iLoadJp2LInternal (ILimage *Image, const void *Lump, ILuint Size)`  
*This is separated so that it can be called for other file types, such as .icns.*
- `ILboolean iLoadJp2Internal (ILimage *Image)`  
*This is separated so that it can be called for other file types, such as .icns.*
- `ILint Jp2ConvertData (jas_stream_t *in, jas_image_t *image)`

### Variables

- `ILconst_string iFormatExtsJp2 []`
- `ILformat iFormatJP2`
- `ILboolean JasperInit = IL_FALSE`

## 7.49.1 Function Documentation

### 7.49.1.1 `jas_stream_t* iJp2ReadStream ( SIO * io )`

- If the buffer size specified is nonpositive, then the buffer

### 7.49.1.2 `jas_stream_t* iJp2WriteStream ( )`

### 7.49.1.3 `ILboolean ilLoadJp2LInternal ( ILimage * Image, const void * Lump, ILuint Size )`

This is separated so that it can be called for other file types, such as .icns.

### 7.49.1.4 `ILboolean iLoadJp2Internal ( ILimage * Image )`

This is separated so that it can be called for other file types, such as .icns.

### 7.49.1.5 `ILint Jp2ConvertData ( jas_stream_t * in, jas_image_t * image )`

## 7.49.2 Variable Documentation

### 7.49.2.1 `ILconst_string iFormatExtsJp2[]`

**Initial value:**

```
= {
    IL_TEXT("jp2"),
    IL_TEXT("jpx"),
    IL_TEXT("j2k"),
    IL_TEXT("j2c"),
    NULL
}
```

### 7.49.2.2 `ILformat iFormatJP2`

**Initial value:**

```
= {
    .Validate = iIsValidJp2,
    .Load     = iLoadJp2Internal,
    .Save     = iSaveJp2Internal,
    .Exts     = iFormatExtsJp2
}
```

### 7.49.2.3 `ILboolean JasperInit = IL_FALSE`

## 7.50 `src/IL/formats/il_jp2.h` File Reference

```
#include "il_internal.h"
```

## Functions

- `ILboolean ilLoadJp2LInternal (ILimage *image, const void *Lump, ILuint Size)`

*This is separated so that it can be called for other file types, such as .icns.*

## 7.50.1 Function Documentation

### 7.50.1.1 ILboolean ilLoadJp2LInternal ( IImage \* *image*, const void \* *Lump*, ILuint *Size* )

This is separated so that it can be called for other file types, such as .icns.

## 7.51 src/IL/formats/il\_jpeg.c File Reference

```
#include "il_internal.h"
#include "jpeglib.h"
#include "il_jpeg.h"
#include "il_manip.h"
#include <setjmp.h>
```

### Data Structures

- struct [iread\\_mgr](#)
- struct [iwrite\\_mgr](#)

### Macros

- #define [INPUT\\_BUF\\_SIZE](#) 4096
- #define [OUTPUT\\_BUF\\_SIZE](#) 4096
- #define [RGB\\_BLUE](#) 2
- #define [RGB\\_GREEN](#) 1
- #define [RGB\\_RED](#) 0

### Typedefs

- typedef [iread\\_mgr](#) \* [iread\\_ptr](#)
- typedef [iwrite\\_mgr](#) \* [iwrite\\_ptr](#)

### Functions

- [devil\\_jpeg\\_read\\_init](#) (SIO \*io, j\_decompress\_ptr cinfo)
- [devil\\_jpeg\\_write\\_init](#) (j\_compress\_ptr cinfo)
- [empty\\_output\\_buffer](#) (j\_compress\_ptr cinfo)
- [void ExitErrorHandle](#) (struct jpeg\_common\_struct \*JpegInfo)
- [fill\\_input\\_buffer](#) (j\_decompress\_ptr cinfo)
- [ILboolean iCheckJpg](#) (ILubyte Header[2])
- [ILint iGetJpgHead](#) (SIO \*io, ILubyte \*Header)
- [ILboolean ilLoadFromJpegStruct](#) (IImage \*image, void \*\_JpegInfo)
- [ILboolean ilLoadJpegInternal](#) (IImage \*image)
- [ILboolean ilSaveFromJpegStruct](#) (IImage \*image, void \*\_JpegInfo)
- [init\\_destination](#) (j\_compress\_ptr cinfo)
- [init\\_source](#) (j\_decompress\_ptr cinfo)
- [ILboolean iSaveJpegInternal](#) (IImage \*image)
- [void OutputMsg](#) (struct jpeg\_common\_struct \*JpegInfo)
- [skip\\_input\\_data](#) (j\_decompress\_ptr cinfo, long num\_bytes)
- [term\\_destination](#) (j\_compress\_ptr cinfo)
- [term\\_source](#) (j\_decompress\_ptr cinfo)

## Variables

- [ILconst\\_string iFormatExtsJPG \[\]](#)
- [ILformat iFormatJPG](#)

## 7.51.1 Macro Definition Documentation

7.51.1.1 `#define INPUT_BUF_SIZE 4096`

7.51.1.2 `#define OUTPUT_BUF_SIZE 4096`

7.51.1.3 `#define RGB_BLUE 2`

7.51.1.4 `#define RGB_GREEN 1`

7.51.1.5 `#define RGB_RED 0`

## 7.51.2 Typedef Documentation

7.51.2.1 `typedef iread_mgr* iread_ptr`

7.51.2.2 `typedef iwrite_mgr* iwrite_ptr`

## 7.51.3 Function Documentation

7.51.3.1 `devil_jpeg_read_init ( SIO * io, j_decompress_ptr cinfo )`

7.51.3.2 `devil_jpeg_write_init ( j_compress_ptr cinfo )`

7.51.3.3 `empty_output_buffer ( j_compress_ptr cinfo )`

7.51.3.4 `void ExitErrorHandle ( struct jpeg_common_struct * JpegInfo )`

7.51.3.5 `fill_input_buffer ( j_decompress_ptr cinfo )`

7.51.3.6 `ILboolean iCheckJpg ( ILubyte Header[2] )`

7.51.3.7 `ILint iGetJpgHead ( SIO * io, ILubyte * Header )`

7.51.3.8 `ILboolean ilLoadFromJpegStruct ( ILImage * image, void * _JpegInfo )`

7.51.3.9 `ILboolean iLoadJpegInternal ( ILImage * image )`

7.51.3.10 `ILboolean iSaveFromJpegStruct ( ILImage * image, void * _JpegInfo )`

7.51.3.11 `init_destination ( j_compress_ptr cinfo )`

7.51.3.12 `init_source ( j_decompress_ptr cinfo )`

7.51.3.13 `ILboolean iSaveJpegInternal ( ILImage * image )`

7.51.3.14 `void OutputMsg ( struct jpeg_common_struct * JpegInfo )`

7.51.3.15 `skip_input_data ( j_decompress_ptr cinfo, long num_bytes )`

7.51.3.16 `term_destination ( j_compress_ptr cinfo )`

7.51.3.17 `term_source ( j_decompress_ptr cinfo )`

## 7.51.4 Variable Documentation

### 7.51.4.1 `ILconst_string iFormatExtsJPG[]`

**Initial value:**

```
= {
    IL_TEXT("jffif"),
    IL_TEXT("jif"),
    IL_TEXT("jpe"),
    IL_TEXT("jpeg"),
    IL_TEXT("jpg"),
    NULL
}
```

### 7.51.4.2 `ILformat iFormatJPG`

**Initial value:**

```
= {
    .Validate = iIsValidJpeg,
    .Load      = iLoadJpegInternal,
    .Save      = iSaveJpegInternal,
    .Exts      = iFormatExtsJPG
}
```

## 7.52 src/IL/formats/il\_jpeg.h File Reference

```
#include "il_internal.h"
```

### Functions

- `ILboolean iLoadJpegInternal (ILImage *)`

### 7.52.1 Function Documentation

7.52.1.1 `ILboolean iLoadJpegInternal ( ILImage * )`

## 7.53 src/IL/formats/il\_lif.c File Reference

```
#include "il_internal.h"
#include "il_lif.h"
```

### Variables

- `ILconst_string iFormatExtsLIF []`
- `ILformat iFormatLIF`

### 7.53.1 Variable Documentation

#### 7.53.1.1 ILconst\_string iFormatExtsLIF[]

**Initial value:**

```
= {  
    IL_TEXT("lif"),  
    NULL  
}
```

#### 7.53.1.2 ILformat iFormatLIF

**Initial value:**

```
= {  
    .Validate = iIsValidLif,  
    .Load     = iLoadLifInternal,  
    .Save     = NULL,  
    .Exts     = iFormatExtsLIF  
}
```

## 7.54 src/IL/formats/il\_lif.h File Reference

```
#include "il_internal.h"  
#include "pack_push.h"  
#include "pack_pop.h"
```

### Data Structures

- struct [LIF\\_HEAD](#)

### Typedefs

- typedef struct [LIF\\_HEAD](#) [LIF\\_HEAD](#)

### 7.54.1 Typedef Documentation

#### 7.54.1.1 typedef struct LIF\_HEAD LIF\_HEAD

## 7.55 src/IL/formats/il\_md1.c File Reference

```
#include "il_internal.h"  
#include "il_md1.h"
```

### Variables

- [ILconst\\_string](#) iFormatExtsMDL []
- [ILformat](#) iFormatMDL

### 7.55.1 Variable Documentation

#### 7.55.1.1 ILconst\_string iFormatExtsMDL[]

**Initial value:**

```
= {
    IL_TEXT("mdl"),
    NULL
}
```

#### 7.55.1.2 ILformat iFormatMDL

**Initial value:**

```
= {
    .Validate = iIsValidMdl,
    .Load     = iLoadMdlInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsMDL
}
```

## 7.56 src/IL/formats/il\_mdl.h File Reference

```
#include "il_internal.h"
#include "pack_push.h"
#include "pack_pop.h"
```

### Data Structures

- struct [MDL\\_HEAD](#)
- struct [TEX\\_HEAD](#)
- struct [TEX\\_INFO](#)

### Typedefs

- typedef struct [TEX\\_HEAD](#) [TEX\\_HEAD](#)

### 7.56.1 Typedef Documentation

#### 7.56.1.1 typedef struct TEX\_HEAD TEX\_HEAD

## 7.57 src/IL/formats/il\_mng.c File Reference

```
#include "il_internal.h"
#include <libmng.h>
```

### Macros

- #define [MNG\\_SUPPORT\\_DISPLAY](#)
- #define [MNG\\_SUPPORT\\_READ](#)
- #define [MNG\\_SUPPORT\\_WRITE](#)

## Functions

- [ILboolean iSaveMngInternal \( \)](#)
- `mng_ptr MNG_DECL mymngalloc (mng_size_t size)`
- `mng_bool MNG_DECL mymngclosestream (mng_handle mng)`
- `mng_bool MNG_DECL mymngerror (mng_handle mng, mng_int32 code, mng_int8 severity, mng_chunkid chunktype, mng_uint32 chunkseq, mng_int32 extra1, mng_int32 extra2, mng_pchar text)`
- `void MNG_DECL mymngfree (mng_ptr p, mng_size_t size)`
- `mng_ptr MNG_DECL mymnggetcanvasline (mng_handle mng, mng_uint32 line)`
- `mng_uint32 MNG_DECL mymnggetticks (mng_handle mng)`
- `mng_bool MNG_DECL mymngopenstream (mng_handle mng)`
- `mng_bool MNG_DECL mymngopenstreamwrite (mng_handle mng)`
- `mng_bool MNG_DECL mymngprocessheader (mng_handle mng, mng_uint32 width, mng_uint32 height)`
- `mng_bool MNG_DECL mymngreadstream (mng_handle mng, mng_ptr buffer, mng_size_t size, mng_uint32 *bytesread)`
- `mng_bool MNG_DECL mymngrefresh (mng_handle mng, mng_uint32 x, mng_uint32 y, mng_uint32 w, mng_uint32 h)`
- `mng_bool MNG_DECL mymngsettimer (mng_handle mng, mng_uint32 msecs)`
- `mng_bool MNG_DECL mymngwritedata (mng_handle mng, mng_ptr buffer, mng_size_t size, mng_uint32 *byteswritten)`

## Variables

- [ILconst\\_string iFormatExtsMNG \[\]](#)
- [ILformat iFormatMNG](#)

## 7.57.1 Macro Definition Documentation

7.57.1.1 `#define MNG_SUPPORT_DISPLAY`

7.57.1.2 `#define MNG_SUPPORT_READ`

7.57.1.3 `#define MNG_SUPPORT_WRITE`

## 7.57.2 Function Documentation

7.57.2.1 `ILboolean iSaveMngInternal ( )`

7.57.2.2 `mng_ptr MNG_DECL mymngalloc ( mng_size_t size )`

7.57.2.3 `mng_bool MNG_DECL mymngclosestream ( mng_handle mng )`

7.57.2.4 `mng_bool MNG_DECL mymngerror ( mng_handle mng, mng_int32 code, mng_int8 severity, mng_chunkid chunktype, mng_uint32 chunkseq, mng_int32 extra1, mng_int32 extra2, mng_pchar text )`

7.57.2.5 `void MNG_DECL mymngfree ( mng_ptr p, mng_size_t size )`

7.57.2.6 `mng_ptr MNG_DECL mymnggetcanvasline ( mng_handle mng, mng_uint32 line )`

7.57.2.7 `mng_uint32 MNG_DECL mymnggetticks ( mng_handle mng )`

7.57.2.8 `mng_bool MNG_DECL mymngopenstream ( mng_handle mng )`

7.57.2.9 `mng_bool MNG_DECL mymngopenstreamwrite ( mng_handle mng )`



- 7.57.2.10 `mng_bool MNG_DECL mymngprocessheader ( mng_handle mng, mng_uint32 width, mng_uint32 height )`
- 7.57.2.11 `mng_bool MNG_DECL mymngreadstream ( mng_handle mng, mng_ptr buffer, mng_size_t size, mng_uint32 * bytesread )`
- 7.57.2.12 `mng_bool MNG_DECL mymngrefresh ( mng_handle mng, mng_uint32 x, mng_uint32 y, mng_uint32 w, mng_uint32 h )`
- 7.57.2.13 `mng_bool MNG_DECL mymngsettimer ( mng_handle mng, mng_uint32 msecs )`
- 7.57.2.14 `mng_bool MNG_DECL mymngwritedata ( mng_handle mng, mng_ptr buffer, mng_size_t size, mng_uint32 * byteswritten )`

### 7.57.3 Variable Documentation

#### 7.57.3.1 `ILconst_string iFormatExtsMNG[]`

**Initial value:**

```
= {
    IL_TEXT("mng"),
    IL_TEXT("jpg"),
    NULL
}
```

#### 7.57.3.2 `ILformat iFormatMNG`

**Initial value:**

```
= {
    .Validate = iIsValidMng,
    .Load     = iLoadMngInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsMNG
}
```

## 7.58 src/IL/formats/il\_mp3.c File Reference

```
#include "il_internal.h"
#include "il_jpeg.h"
```

### Data Structures

- struct [MP3HEAD](#)

### Macros

- #define [MP3\\_JPG](#) 1
- #define [MP3\\_NONE](#) 0
- #define [MP3\\_PNG](#) 2

### Typedefs

- typedef struct [MP3HEAD](#) [MP3HEAD](#)

## Functions

- [ILboolean iGetMp3Head](#) ([SIO \\*io](#), [MP3HEAD \\*Header](#))
- [ILboolean iLoadJpegInternal](#) ([ILImage \\*image](#))
- [ILboolean iLoadPngInternal](#) ([ILImage \\*image](#))

## Variables

- [ILconst\\_string iFormatExtsMP3](#) []
- [ILformat iFormatMP3](#)

## 7.58.1 Macro Definition Documentation

7.58.1.1 [#define MP3\\_JPG](#) 1

7.58.1.2 [#define MP3\\_NONE](#) 0

7.58.1.3 [#define MP3\\_PNG](#) 2

## 7.58.2 Typedef Documentation

7.58.2.1 [typedef struct MP3HEAD MP3HEAD](#)

## 7.58.3 Function Documentation

7.58.3.1 [ILboolean iGetMp3Head](#) ( [SIO \\* io](#), [MP3HEAD \\* Header](#) )

7.58.3.2 [ILboolean iLoadJpegInternal](#) ( [ILImage \\* image](#) )

7.58.3.3 [ILboolean iLoadPngInternal](#) ( [ILImage \\* image](#) )

## 7.58.4 Variable Documentation

7.58.4.1 [ILconst\\_string iFormatExtsMP3](#) []

### Initial value:

```
= {
    IL\_TEXT ("mp3"),
    NULL
}
```

7.58.4.2 [ILformat iFormatMP3](#)

### Initial value:

```
= {
    .Validate = iIsValidMp3,
    .Load     = iLoadMp3Internal,
    .Save     = NULL,
    .Exts     = iFormatExtsMP3
}
```

## 7.59 src/IL/formats/il\_pal\_act.c File Reference

```
#include "il_internal.h"
#include "il_pal.h"
#include <string.h>
#include <ctype.h>
#include <limits.h>
```

### Variables

- [ILformat iFormatACT\\_PAL](#)
- [ILconst\\_string iFormatExtsACT\\_PAL \[\]](#)

#### 7.59.1 Variable Documentation

##### 7.59.1.1 ILformat iFormatACT\_PAL

###### Initial value:

```
= {
    .Validate = iIsValidActPal,
    .Load     = iLoadActPal,
    .Save     = NULL,
    .Exts     = iFormatExtsACT_PAL
}
```

##### 7.59.1.2 ILconst\_string iFormatExtsACT\_PAL[]

###### Initial value:

```
= {
    IL_TEXT("act"),
    NULL
}
```

## 7.60 src/IL/formats/il\_pal\_col.c File Reference

```
#include "il_internal.h"
#include "il_pal.h"
#include <string.h>
#include <ctype.h>
#include <limits.h>
```

### Variables

- [ILformat iFormatCOL\\_PAL](#)
- [ILconst\\_string iFormatExtsCOL\\_PAL \[\]](#)

#### 7.60.1 Variable Documentation

##### 7.60.1.1 ILformat iFormatCOL\_PAL

###### Initial value:

```
= {
    .Validate = iIsValidColPal,
    .Load     = iLoadColPal,
    .Save     = NULL,
    .Exts     = iFormatExtsCOL_PAL
}
```

### 7.60.1.2 ILconst\_string iFormatExtsCOL\_PAL[]

**Initial value:**

```
= {
    IL_TEXT("col"),
    NULL
}
```

## 7.61 src/IL/formats/il\_pal\_halo.c File Reference

```
#include "il_internal.h"
#include "il_pal.h"
#include <string.h>
#include <ctype.h>
#include <limits.h>
#include "pack_push.h"
#include "pack_pop.h"
```

### Data Structures

- struct [HALOHEAD](#)

### Typedefs

- typedef struct [HALOHEAD](#) HALOHEAD

### Variables

- [ILconst\\_string](#) iFormatExtsHALO\_PAL []
- [ILformat](#) iFormatHALO\_PAL

### 7.61.1 Typedef Documentation

#### 7.61.1.1 typedef struct HALOHEAD HALOHEAD

### 7.61.2 Variable Documentation

#### 7.61.2.1 ILconst\_string iFormatExtsHALO\_PAL[]

**Initial value:**

```
= {
    IL_TEXT("pal"),
    NULL
}
```

## 7.61.2.2 ILformat iFormatHALO\_PAL

## Initial value:

```
= {
    .Validate = iIsValidHaloPal,
    .Load     = iLoadHaloPal,
    .Save     = NULL,
    .Exts     = iFormatExtsHALO_PAL
}
```

## 7.62 src/IL/formats/il\_pal\_jasc.c File Reference

```
#include "il_internal.h"
#include "il_pal.h"
#include <string.h>
#include <ctype.h>
#include <limits.h>
```

## Macros

- #define `BUFFLEN` 256
- #define `PALBPP` 3

## Variables

- `ILconst_string` `iFormatExtsJASC_PAL` []
- `ILformat` `iFormatJASC_PAL`

## 7.62.1 Macro Definition Documentation

7.62.1.1 #define `BUFFLEN` 2567.62.1.2 #define `PALBPP` 3

## 7.62.2 Variable Documentation

7.62.2.1 `ILconst_string` `iFormatExtsJASC_PAL` []

## Initial value:

```
= {
    IL_TEXT("pal"),
    NULL
}
```

## 7.62.2.2 ILformat iFormatJASC\_PAL

## Initial value:

```
= {
    .Validate = iIsValidJascPal,
    .Load     = iLoadJascPal,
    .Save     = iSaveJascPal,
    .Exts     = iFormatExtsJASC_PAL
}
```

## 7.63 src/IL/formats/il\_pal\_plt.c File Reference

```
#include "il_internal.h"
#include "il_pal.h"
#include <string.h>
#include <ctype.h>
#include <limits.h>
```

### Variables

- [ILconst\\_string iFormatExtsPLT\\_PAL \[\]](#)
- [ILformat iFormatPLT\\_PAL](#)

### 7.63.1 Variable Documentation

#### 7.63.1.1 ILconst\_string iFormatExtsPLT\_PAL[]

##### Initial value:

```
= {
    IL_TEXT("plt"),
    NULL
}
```

#### 7.63.1.2 ILformat iFormatPLT\_PAL

##### Initial value:

```
= {
    .Validate = NULL,
    .Load     = iLoadPltPal,
    .Save     = NULL,
    .Exts     = iFormatExtsPLT_PAL
}
```

## 7.64 src/IL/formats/il\_pcd.c File Reference

```
#include "il_internal.h"
#include "il_manip.h"
#include "il_color.h"
```

### Variables

- [ILconst\\_string iFormatExtsPCD \[\]](#)
- [ILformat iFormatPCD](#)

### 7.64.1 Variable Documentation

#### 7.64.1.1 ILconst\_string iFormatExtsPCD[]

##### Initial value:

```
= {
    IL_TEXT("pcd"),
    NULL
}
```

#### 7.64.1.2 ILformat iFormatPCD

##### Initial value:

```
= {
    .Validate = NULL,
    .Load     = iLoadPcdInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsPCD
}
```

## 7.65 src/IL/formats/il\_pcx.c File Reference

```
#include "il_internal.h"
#include "il_pcx.h"
#include "il_manip.h"
```

### Functions

- [ILuint encput](#) ([SIO \\*io](#), [ILubyte byt](#), [ILubyte cnt](#))

### Variables

- [ILconst\\_string iFormatExtsPCX](#) []
- [ILformat iFormatPCX](#)

### 7.65.1 Function Documentation

7.65.1.1 [ILuint encput](#) ( [SIO \\* io](#), [ILubyte byt](#), [ILubyte cnt](#) )

### 7.65.2 Variable Documentation

7.65.2.1 [ILconst\\_string iFormatExtsPCX](#) []

##### Initial value:

```
= {
    IL_TEXT("pcx"),
    NULL
}
```

#### 7.65.2.2 ILformat iFormatPCX

##### Initial value:

```
= {
    .Validate = iIsValidPcx,
    .Load     = iLoadPcxInternal,
    .Save     = iSavePcxInternal,
    .Exts     = iFormatExtsPCX
}
```

## 7.66 src/IL/formats/il\_pcx.h File Reference

```
#include "il_internal.h"
#include "pack_push.h"
#include "pack_pop.h"
```

### Data Structures

- struct [PCXHEAD](#)

### Typedefs

- typedef struct [PCXHEAD](#) [PCXHEAD](#)

#### 7.66.1 Typedef Documentation

##### 7.66.1.1 typedef struct PCXHEAD PCXHEAD

## 7.67 src/IL/formats/il\_pic.c File Reference

```
#include "il_internal.h"
#include "il_pic.h"
#include "il_manip.h"
#include <string.h>
```

### Variables

- [ILconst\\_string](#) [iFormatExtsPIC](#) []
- [ILformat](#) [iFormatPIC](#)

#### 7.67.1 Variable Documentation

##### 7.67.1.1 [ILconst\\_string](#) [iFormatExtsPIC](#) []

**Initial value:**

```
= {
    IL\_TEXT("pic"),
    NULL
}
```

##### 7.67.1.2 [ILformat](#) [iFormatPIC](#)

**Initial value:**

```
= {
    .Validate = iIsValidPic,
    .Load     = iLoadPicInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsPIC
}
```



## 7.68 src/IL/formats/il\_pic.h File Reference

```
#include "il_internal.h"
#include "pack_push.h"
#include "pack_pop.h"
```

### Data Structures

- struct [CHANNEL](#)
- struct [PIC\\_HEAD](#)

### Macros

- #define [PIC\\_ALPHA\\_CHANNEL](#) 0x10
- #define [PIC\\_AUXILIARY\\_1\\_CHANNEL](#) 0x02
- #define [PIC\\_AUXILIARY\\_2\\_CHANNEL](#) 0x01
- #define [PIC\\_BLUE\\_CHANNEL](#) 0x20
- #define [PIC\\_DEPTH\\_CHANNEL](#) 0x04
- #define [PIC\\_GREEN\\_CHANNEL](#) 0x40
- #define [PIC\\_MIXED\\_RUN\\_LENGTH](#) 0x02
- #define [PIC\\_PURE\\_RUN\\_LENGTH](#) 0x01
- #define [PIC\\_RED\\_CHANNEL](#) 0x80
- #define [PIC\\_SHADOW\\_CHANNEL](#) 0x08
- #define [PIC\\_SIGNED\\_FLOAT](#) 0x20
- #define [PIC\\_SIGNED\\_INTEGER](#) 0x10
- #define [PIC\\_UNCOMPRESSED](#) 0x00
- #define [PIC\\_UNSIGNED\\_INTEGER](#) 0x00

### Typedefs

- typedef struct [CHANNEL](#) [CHANNEL](#)
- typedef struct [PIC\\_HEAD](#) [PIC\\_HEAD](#)

#### 7.68.1 Macro Definition Documentation

7.68.1.1 #define [PIC\\_ALPHA\\_CHANNEL](#) 0x10

7.68.1.2 #define [PIC\\_AUXILIARY\\_1\\_CHANNEL](#) 0x02

7.68.1.3 #define [PIC\\_AUXILIARY\\_2\\_CHANNEL](#) 0x01

7.68.1.4 #define [PIC\\_BLUE\\_CHANNEL](#) 0x20

7.68.1.5 #define [PIC\\_DEPTH\\_CHANNEL](#) 0x04

7.68.1.6 #define [PIC\\_GREEN\\_CHANNEL](#) 0x40

7.68.1.7 #define [PIC\\_MIXED\\_RUN\\_LENGTH](#) 0x02

7.68.1.8 #define [PIC\\_PURE\\_RUN\\_LENGTH](#) 0x01

7.68.1.9 `#define PIC_RED_CHANNEL 0x80`

7.68.1.10 `#define PIC_SHADOW_CHANNEL 0x08`

7.68.1.11 `#define PIC_SIGNED_FLOAT 0x20`

7.68.1.12 `#define PIC_SIGNED_INTEGER 0x10`

7.68.1.13 `#define PIC_UNCOMPRESSED 0x00`

7.68.1.14 `#define PIC_UNSIGNED_INTEGER 0x00`

## 7.68.2 Typedef Documentation

7.68.2.1 `typedef struct CHANNEL CHANNEL`

7.68.2.2 `typedef struct PIC_HEAD PIC_HEAD`

## 7.69 src/IL/formats/il\_pix.c File Reference

```
#include "il_internal.h"
#include "il_manip.h"
#include "il_endian.h"
#include "pack_push.h"
#include "pack_pop.h"
```

### Data Structures

- struct [PIXHEAD](#)

### Typedefs

- typedef struct [PIXHEAD](#) [PIXHEAD](#)

### Variables

- [ILconst\\_string](#) [iFormatExtsPIX](#) []
- [ILformat](#) [iFormatPIX](#)

## 7.69.1 Typedef Documentation

7.69.1.1 `typedef struct PIXHEAD PIXHEAD`

## 7.69.2 Variable Documentation

7.69.2.1 `ILconst_string iFormatExtsPIX[]`

### Initial value:

```
= {
    IL\_TEXT("pix"),
    NULL
}
```

## 7.69.2.2 ILformat iFormatPIX

## Initial value:

```
= {
    .Validate = iIsValidPix,
    .Load     = iLoadPixInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsPIX
}
```

## 7.70 src/IL/formats/il\_png.c File Reference

```
#include "il_internal.h"
#include <png.h>
#include "il_manip.h"
#include <stdlib.h>
```

## Data Structures

- struct [PNGData](#)

## Macros

- #define [GAMMA\\_CORRECTION](#) 1.0

## Functions

- [void flush\\_data](#) (png\_structp png\_ptr)
- [ILboolean iLoadPngInternal](#) (ILImage \*image)
- [ILboolean iSavePngInternal](#) (ILImage \*image)
- [void png\\_write](#) (png\_structp png\_ptr, png\_bytep data, png\_size\_t length)

## Variables

- [ILconst\\_string iFormatExtsPNG](#) []
- [ILformat iFormatPNG](#)

## 7.70.1 Macro Definition Documentation

## 7.70.1.1 #define GAMMA\_CORRECTION 1.0

## 7.70.2 Function Documentation

## 7.70.2.1 void flush\_data ( png\_structp png\_ptr )

## 7.70.2.2 ILboolean iLoadPngInternal ( ILImage \* image )

## 7.70.2.3 ILboolean iSavePngInternal ( ILImage \* image )

## 7.70.2.4 void png\_write ( png\_structp png\_ptr, png\_bytep data, png\_size\_t length )

### 7.70.3 Variable Documentation

#### 7.70.3.1 ILconst\_string iFormatExtsPNG[]

**Initial value:**

```
= {
    IL_TEXT("png"),
    NULL
}
```

#### 7.70.3.2 ILformat iFormatPNG

**Initial value:**

```
= {
    .Validate = iIsValidPng,
    .Load     = iLoadPngInternal,
    .Save     = iSavePngInternal,
    .Exts     = iFormatExtsPNG
}
```

## 7.71 src/IL/formats/il\_pnm.c File Reference

```
#include "il_internal.h"
#include "il_pnm.h"
#include <limits.h>
#include <ctype.h>
#include "il_manip.h"
```

### Macros

- #define `MAX_BUFFER` 180

### Variables

- `ILstring FName` = NULL
- `ILconst_string iFormatExtsPNM` []
- `ILformat iFormatPNM`
- `ILboolean IsLump` = `IL_FALSE`

### 7.71.1 Macro Definition Documentation

#### 7.71.1.1 #define MAX\_BUFFER 180

### 7.71.2 Variable Documentation

#### 7.71.2.1 ILstring FName = NULL

#### 7.71.2.2 ILconst\_string iFormatExtsPNM[]

**Initial value:**

```
= {
    IL_TEXT("pbm"),
    IL_TEXT("pgm"),
    IL_TEXT("pnm"),
    IL_TEXT("ppm"),
    NULL
}
```

### 7.71.2.3 ILformat iFormatPNM

**Initial value:**

```
= {
    .Validate = iIsValidPnm,
    .Load     = iLoadPnmInternal,
    .Save     = iSavePnmInternal,
    .Exts     = iFormatExtsPNM
}
```

### 7.71.2.4 ILboolean IsLump = IL\_FALSE

## 7.72 src/IL/formats/il\_pnm.h File Reference

```
#include "il_internal.h"
```

### Data Structures

- struct [PPMINFO](#)

### Macros

- #define [IL\\_PBM\\_ASCII](#) 0x0001
- #define [IL\\_PBM\\_BINARY](#) 0x0004
- #define [IL\\_PGM\\_ASCII](#) 0x0002
- #define [IL\\_PGM\\_BINARY](#) 0x0005
- #define [IL\\_PPM\\_ASCII](#) 0x0003
- #define [IL\\_PPM\\_BINARY](#) 0x0006

### Typedefs

- typedef struct [PPMINFO](#) [PPMINFO](#)

### 7.72.1 Macro Definition Documentation

7.72.1.1 #define [IL\\_PBM\\_ASCII](#) 0x0001

7.72.1.2 #define [IL\\_PBM\\_BINARY](#) 0x0004

7.72.1.3 #define [IL\\_PGM\\_ASCII](#) 0x0002

7.72.1.4 #define [IL\\_PGM\\_BINARY](#) 0x0005

7.72.1.5 #define [IL\\_PPM\\_ASCII](#) 0x0003

7.72.1.6 `#define IL_PPM_BINARY 0x0006`

## 7.72.2 Typedef Documentation

7.72.2.1 `typedef struct PPMINFO PPMINFO`

## 7.73 `src/IL/formats/il_psd.c` File Reference

```
#include "il_internal.h"
#include "il_psd.h"
```

### Functions

- [ILboolean GetSingleChannel](#) ([ILImage](#) \*image, [PSDHEAD](#) \*Head, [ILubyte](#) \*Buffer, [ILboolean](#) Compressed)
- [ILboolean iSavePsdInternal](#) ([ILImage](#) \*image)
- [ILboolean ParseResources](#) ([ILImage](#) \*image, [ILuint](#) ResourceSize, [ILubyte](#) \*Resources)
- [ILboolean PsdGetData](#) ([ILImage](#) \*image, [PSDHEAD](#) \*Head, [void](#) \*Buffer, [ILboolean](#) Compressed)

### Variables

- [ILushort ChannelNum](#)
- [ILconst\\_string iFormatExtsPSD \[\]](#)
- [ILformat iFormatPSD](#)

### 7.73.1 Function Documentation

7.73.1.1 [ILboolean GetSingleChannel](#) ( [ILImage](#) \* *image*, [PSDHEAD](#) \* *Head*, [ILubyte](#) \* *Buffer*, [ILboolean](#) *Compressed* )

7.73.1.2 [ILboolean iSavePsdInternal](#) ( [ILImage](#) \* *image* )

7.73.1.3 [ILboolean ParseResources](#) ( [ILImage](#) \* *image*, [ILuint](#) *ResourceSize*, [ILubyte](#) \* *Resources* )

7.73.1.4 [ILboolean PsdGetData](#) ( [ILImage](#) \* *image*, [PSDHEAD](#) \* *Head*, [void](#) \* *Buffer*, [ILboolean](#) *Compressed* )

### 7.73.2 Variable Documentation

7.73.2.1 [ILushort ChannelNum](#)

7.73.2.2 [ILconst\\_string iFormatExtsPSD\[\]](#)

#### Initial value:

```
= {
    IL\_TEXT ("psd"),
    IL\_TEXT ("pdd"),
    NULL
}
```

### 7.73.2.3 ILformat iFormatPSD

#### Initial value:

```
= {  
    .Validate = iIsValidPsd,  
    .Load     = iLoadPsdInternal,  
    .Save     = iSavePsdInternal,  
    .Exts     = iFormatExtsPSD  
}
```

## 7.74 src/IL/formats/il\_psd.h File Reference

```
#include "il_internal.h"  
#include "pack_push.h"  
#include "pack_pop.h"
```

### Data Structures

- struct [PSDHEAD](#)

### Typedefs

- typedef struct [PSDHEAD](#) [PSDHEAD](#)

### 7.74.1 Typedef Documentation

#### 7.74.1.1 typedef struct PSDHEAD PSDHEAD

## 7.75 src/IL/formats/il\_psp.c File Reference

```
#include "il_internal.h"  
#include "il_psp.h"
```

### Data Structures

- struct [PSP\\_CTX](#)

### Variables

- [ILconst\\_string](#) [iFormatExtsPSP](#) []
- [ILformat](#) [iFormatPSP](#)

### 7.75.1 Variable Documentation

#### 7.75.1.1 [ILconst\\_string](#) [iFormatExtsPSP](#) []

#### Initial value:

```
= {
    IL_TEXT("psp"),
    NULL
}
```

### 7.75.1.2 ILformat iFormatPSP

#### Initial value:

```
= {
    .Validate = iIsValidPsp,
    .Load      = iLoadPspInternal,
    .Save      = NULL,
    .Exts      = iFormatExtsPSP
}
```

## 7.76 src/IL/formats/il\_psp.h File Reference

```
#include "il_internal.h"
#include "pack_push.h"
#include "pack_pop.h"
```

### Data Structures

- struct [ALPHA\\_CHUNK](#)
- struct [ALPHAINFO\\_CHUNK](#)
- struct [BLOCKHEAD](#)
- struct [CHANNEL\\_CHUNK](#)
- struct [GENATT\\_CHUNK](#)
- struct [LAYERBITMAP\\_CHUNK](#)
- struct [LAYERINFO\\_CHUNK](#)
- struct [PSPHEAD](#)
- struct [PSPRECT](#)

### Typedefs

- typedef struct [ALPHA\\_CHUNK](#) [ALPHA\\_CHUNK](#)
- typedef struct [ALPHAINFO\\_CHUNK](#) [ALPHAINFO\\_CHUNK](#)
- typedef struct [BLOCKHEAD](#) [BLOCKHEAD](#)
- typedef struct [CHANNEL\\_CHUNK](#) [CHANNEL\\_CHUNK](#)
- typedef struct [GENATT\\_CHUNK](#) [GENATT\\_CHUNK](#)
- typedef struct [LAYERBITMAP\\_CHUNK](#) [LAYERBITMAP\\_CHUNK](#)
- typedef struct [LAYERINFO\\_CHUNK](#) [LAYERINFO\\_CHUNK](#)
- typedef struct [PSPHEAD](#) [PSPHEAD](#)
- typedef struct [PSPRECT](#) [PSPRECT](#)

### Enumerations

- enum [PSP\\_METRIC](#) { [PSP\\_METRIC\\_UNDEFINED](#) = 0, [PSP\\_METRIC\\_INCH](#), [PSP\\_METRIC\\_CM](#) }



- enum `PSPBlockID` {  
`PSP_IMAGE_BLOCK` = 0, `PSP_CREATOR_BLOCK`, `PSP_COLOR_BLOCK`, `PSP_LAYER_START_BLOCK`,  
`PSP_LAYER_BLOCK`, `PSP_CHANNEL_BLOCK`, `PSP_SELECTION_BLOCK`, `PSP_ALPHA_BANK_BLOCK`,  
`PSP_ALPHA_CHANNEL_BLOCK`, `PSP_COMPOSITE_IMAGE_BLOCK`, `PSP_EXTENDED_DATA_BLOCK`, `PSP_TUBE_BLOCK`,  
`PSP_ADJUSTMENT_EXTENSION_BLOCK`, `PSP_VECTOR_EXTENSION_BLOCK`, `PSP_SHAPE_BLOCK`,  
`PSP_PAINTSTYLE_BLOCK`,  
`PSP_COMPOSITE_IMAGE_BANK_BLOCK`, `PSP_COMPOSITE_ATTRIBUTES_BLOCK`, `PSP_JPEG_BLOCK`, `PSP_LINestyle_BLOCK`,  
`PSP_TABLE_BANK_BLOCK`, `PSP_TABLE_BLOCK`, `PSP_PAPER_BLOCK`, `PSP_PATTERN_BLOCK` }
- enum `PSPChannelType` { `PSP_CHANNEL_COMPOSITE` = 0, `PSP_CHANNEL_RED`, `PSP_CHANNEL_GREEN`, `PSP_CHANNEL_BLUE` }
- enum `PSPCompression` { `PSP_COMP_NONE` = 0, `PSP_COMP_RLE`, `PSP_COMP_LZ77`, `PSP_COMP_JPEG` }
- enum `PSPCreatorAppID` { `PSP_CREATOR_APP_UNKNOWN` = 0, `PSP_CREATOR_APP_PAINT_SHOP_PRO` }
- enum `PSPCreatorFieldID` {  
`PSP_CRTR_FLD_TITLE` = 0, `PSP_CRTR_FLD_CRT_DATE`, `PSP_CRTR_FLD_MOD_DATE`, `PSP_CRTR_FLD_ARTIST`,  
`PSP_CRTR_FLD_COPYRIGHT`, `PSP_CRTR_FLD_DESC`, `PSP_CRTR_FLD_APP_ID`, `PSP_CRTR_FLD_APP_VER` }
- enum `PSPDIBType` {  
`PSP_DIB_IMAGE` = 0, `PSP_DIB_TRANS_MASK`, `PSP_DIB_USER_MASK`, `PSP_DIB_SELECTION`,  
`PSP_DIB_ALPHA_MASK`, `PSP_DIB_THUMBNAIL` }
- enum `PSPExtendedDataID` { `PSP_XDATA_TRNS_INDEX` = 0 }
- enum `PSPLayerType` { `PSP_LAYER_NORMAL` = 0, `PSP_LAYER_FLOATING_SELECTION` }
- enum `TubePlacementMode` { `tpmRandom`, `tpmConstant` }
- enum `TubeSelectionMode` {  
`tsmRandom`, `tsmIncremental`, `tsmAngular`, `tsmPressure`,  
`tsmVelocity` }

## 7.76.1 Typedef Documentation

7.76.1.1 typedef struct `ALPHA_CHUNK` `ALPHA_CHUNK`

7.76.1.2 typedef struct `ALPHAINFO_CHUNK` `ALPHAINFO_CHUNK`

7.76.1.3 typedef struct `BLOCKHEAD` `BLOCKHEAD`

7.76.1.4 typedef struct `CHANNEL_CHUNK` `CHANNEL_CHUNK`

7.76.1.5 typedef struct `GENATT_CHUNK` `GENATT_CHUNK`

7.76.1.6 typedef struct `LAYERBITMAP_CHUNK` `LAYERBITMAP_CHUNK`

7.76.1.7 typedef struct `LAYERINFO_CHUNK` `LAYERINFO_CHUNK`

7.76.1.8 typedef struct `PSPHEAD` `PSPHEAD`

7.76.1.9 typedef struct `PSPRECT` `PSPRECT`

## 7.76.2 Enumeration Type Documentation

## 7.76.2.1 enum PSP\_METRIC

Enumerator

*PSP\_METRIC\_UNDEFINED*  
*PSP\_METRIC\_INCH*  
*PSP\_METRIC\_CM*

## 7.76.2.2 enum PSPBlockID

Enumerator

*PSP\_IMAGE\_BLOCK*  
*PSP\_CREATOR\_BLOCK*  
*PSP\_COLOR\_BLOCK*  
*PSP\_LAYER\_START\_BLOCK*  
*PSP\_LAYER\_BLOCK*  
*PSP\_CHANNEL\_BLOCK*  
*PSP\_SELECTION\_BLOCK*  
*PSP\_ALPHA\_BANK\_BLOCK*  
*PSP\_ALPHA\_CHANNEL\_BLOCK*  
*PSP\_COMPOSITE\_IMAGE\_BLOCK*  
*PSP\_EXTENDED\_DATA\_BLOCK*  
*PSP\_TUBE\_BLOCK*  
*PSP\_ADJUSTMENT\_EXTENSION\_BLOCK*  
*PSP\_VECTOR\_EXTENSION\_BLOCK*  
*PSP\_SHAPE\_BLOCK*  
*PSP\_PAINTSTYLE\_BLOCK*  
*PSP\_COMPOSITE\_IMAGE\_BANK\_BLOCK*  
*PSP\_COMPOSITE\_ATTRIBUTES\_BLOCK*  
*PSP\_JPEG\_BLOCK*  
*PSP\_LINestyle\_BLOCK*  
*PSP\_TABLE\_BANK\_BLOCK*  
*PSP\_TABLE\_BLOCK*  
*PSP\_PAPER\_BLOCK*  
*PSP\_PATTERN\_BLOCK*

## 7.76.2.3 enum PSPChannelType

Enumerator

*PSP\_CHANNEL\_COMPOSITE*  
*PSP\_CHANNEL\_RED*  
*PSP\_CHANNEL\_GREEN*  
*PSP\_CHANNEL\_BLUE*

## 7.76.2.4 enum PSPCompression

Enumerator

*PSP\_COMP\_NONE*  
*PSP\_COMP\_RLE*  
*PSP\_COMP\_LZ77*  
*PSP\_COMP\_JPEG*

## 7.76.2.5 enum PSPCreatorAppID

Enumerator

*PSP\_CREATOR\_APP\_UNKNOWN*  
*PSP\_CREATOR\_APP\_PAINT\_SHOP\_PRO*

## 7.76.2.6 enum PSPCreatorFieldID

Enumerator

*PSP\_CRTR\_FLD\_TITLE*  
*PSP\_CRTR\_FLD\_CRT\_DATE*  
*PSP\_CRTR\_FLD\_MOD\_DATE*  
*PSP\_CRTR\_FLD\_ARTIST*  
*PSP\_CRTR\_FLD\_CPYRGHT*  
*PSP\_CRTR\_FLD\_DESC*  
*PSP\_CRTR\_FLD\_APP\_ID*  
*PSP\_CRTR\_FLD\_APP\_VER*

## 7.76.2.7 enum PSPDIBType

Enumerator

*PSP\_DIB\_IMAGE*  
*PSP\_DIB\_TRANS\_MASK*  
*PSP\_DIB\_USER\_MASK*  
*PSP\_DIB\_SELECTION*  
*PSP\_DIB\_ALPHA\_MASK*  
*PSP\_DIB\_THUMBNAIL*

## 7.76.2.8 enum PSPExtendedDataID

Enumerator

*PSP\_XDATA\_TRNS\_INDEX*

## 7.76.2.9 enum PSPLayerType

Enumerator

*PSP\_LAYER\_NORMAL*  
*PSP\_LAYER\_FLOATING\_SELECTION*

#### 7.76.2.10 enum TubePlacementMode

Enumerator

*tpmRandom*  
*tpmConstant*

#### 7.76.2.11 enum TubeSelectionMode

Enumerator

*tsmRandom*  
*tsmIncremental*  
*tsmAngular*  
*tsmPressure*  
*tsmVelocity*

### 7.77 src/IL/formats/il\_pxr.c File Reference

```
#include "il_internal.h"  
#include "il_manip.h"  
#include "il_endian.h"  
#include "pack_push.h"  
#include "pack_pop.h"
```

#### Data Structures

- struct [PIXHEAD](#)

#### Typedefs

- typedef struct [PIXHEAD](#) [PIXHEAD](#)

#### Variables

- [ILconst\\_string](#) [iFormatExtsPXR](#) []
- [ILformat](#) [iFormatPXR](#)

#### 7.77.1 Typedef Documentation

##### 7.77.1.1 typedef struct PIXHEAD PIXHEAD

#### 7.77.2 Variable Documentation

##### 7.77.2.1 ILconst\_string iFormatExtsPXR[]

**Initial value:**

```
= {  
    IL\_TEXT("pxr"),  
    NULL  
}
```

## 7.77.2.2 ILformat iFormatPXR

## Initial value:

```
= {
    .Validate = iIsValidPxr,
    .Load     = iLoadPxrInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsPXR
}
```

## 7.78 src/IL/formats/il\_q2pal.h File Reference

## Macros

- `#define IL_Q2PAL_SIZE 768`

## Variables

- `ILubyte ilDefaultQ2Pal [IL_Q2PAL_SIZE]`

## 7.78.1 Macro Definition Documentation

7.78.1.1 `#define IL_Q2PAL_SIZE 768`

## 7.78.2 Variable Documentation

7.78.2.1 `ILubyte ilDefaultQ2Pal[IL_Q2PAL_SIZE]`

## 7.79 src/IL/formats/il\_raw.c File Reference

```
#include "il_internal.h"
#include "pack_push.h"
#include "pack_pop.h"
```

## Data Structures

- struct `RAW_HEAD`

## Variables

- `ILconst_string iFormatExtsRAW []`
- `ILformat iFormatRAW`

## 7.79.1 Variable Documentation

7.79.1.1 `ILconst_string iFormatExtsRAW[]`

## Initial value:

```
= {
    NULL
}
```

### 7.79.1.2 ILformat iFormatRAW

#### Initial value:

```
= {
    .Validate = NULL,
    .Load     = iLoadRawInternal,
    .Save     = iSaveRawInternal,
    .Exts     = iFormatExtsRAW
}
```

## 7.80 src/IL/formats/il\_rawdata.c File Reference

```
#include "il_internal.h"
#include "il_manip.h"
```

### Functions

- **ILboolean ILAPIENTRY iLoadData** (**ILconst\_string** FileName, **ILuint** Width, **ILuint** Height, **ILuint** Depth, **ILubyte** Bpp)  
*Reads a raw data file.*
- **ILboolean ILAPIENTRY iLoadDataF** (**ILHANDLE** File, **ILuint** Width, **ILuint** Height, **ILuint** Depth, **ILubyte** Bpp)  
*Reads an already-opened raw data file.*
- **ILboolean ILAPIENTRY iLoadDataL** (**void** \*Lump, **ILuint** Size, **ILuint** Width, **ILuint** Height, **ILuint** Depth, **ILubyte** Bpp)  
*Reads from a raw data memory "lump".*
- **ILboolean ILAPIENTRY iSaveData** (**ILconst\_string** FileName)  
*Save the current image to FileName as raw data.*

### 7.80.1 Function Documentation

**7.80.1.1 ILboolean ILAPIENTRY iLoadData** ( **ILconst\_string** FileName, **ILuint** Width, **ILuint** Height, **ILuint** Depth, **ILubyte** Bpp )

Reads a raw data file.

**7.80.1.2 ILboolean ILAPIENTRY iLoadDataF** ( **ILHANDLE** File, **ILuint** Width, **ILuint** Height, **ILuint** Depth, **ILubyte** Bpp )

Reads an already-opened raw data file.

**7.80.1.3 ILboolean ILAPIENTRY iLoadDataL** ( **void** \* Lump, **ILuint** Size, **ILuint** Width, **ILuint** Height, **ILuint** Depth, **ILubyte** Bpp )

Reads from a raw data memory "lump".

**7.80.1.4 ILboolean ILAPIENTRY iSaveData** ( **ILconst\_string** FileName )

Save the current image to FileName as raw data.

## 7.81 src/IL/formats/il\_rot.c File Reference

```
#include "il_internal.h"
#include "il_dds.h"
#include "pack_push.h"
#include "pack_pop.h"
```

### Data Structures

- struct [FORM\\_HEAD](#)
- struct [ROT\\_HEAD](#)

### Macros

- #define [ROT\\_DXT1](#) 1028
- #define [ROT\\_DXT3](#) 1029
- #define [ROT\\_DXT5](#) 1030
- #define [ROT\\_RGBA32](#) 1024

### Variables

- [ILconst\\_string](#) [iFormatExtsROT](#) []
- [ILformat](#) [iFormatROT](#)

### 7.81.1 Macro Definition Documentation

7.81.1.1 #define [ROT\\_DXT1](#) 1028

7.81.1.2 #define [ROT\\_DXT3](#) 1029

7.81.1.3 #define [ROT\\_DXT5](#) 1030

7.81.1.4 #define [ROT\\_RGBA32](#) 1024

### 7.81.2 Variable Documentation

7.81.2.1 [ILconst\\_string](#) [iFormatExtsROT](#) []

#### Initial value:

```
= {
    IL\_TEXT("rot"),
    NULL
}
```

7.81.2.2 [ILformat](#) [iFormatROT](#)

#### Initial value:

```
= {
    .Validate = iIsValidRot,
    .Load     = iLoadRotInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsROT
}
```

## 7.82 src/IL/formats/il\_sgi.c File Reference

```
#include "il_internal.h"
#include "il_sgi.h"
#include "il_manip.h"
#include <limits.h>
```

### Variables

- [ILconst\\_string iFormatExtsSGI \[\]](#)
- [ILformat iFormatSGI](#)

### 7.82.1 Variable Documentation

#### 7.82.1.1 [ILconst\\_string iFormatExtsSGI\[\]](#)

**Initial value:**

```
= {
    IL_TEXT("sgi"),
    IL_TEXT("bw"),
    IL_TEXT("rgb"),
    IL_TEXT("rgba"),
    NULL
}
```

#### 7.82.1.2 [ILformat iFormatSGI](#)

**Initial value:**

```
= {
    .Validate = iIsValidSgi,
    .Load     = iLoadSgiInternal,
    .Save     = iSaveSgiInternal,
    .Exts     = iFormatExtsSGI
}
```

## 7.83 src/IL/formats/il\_sgi.h File Reference

```
#include "il_internal.h"
#include "pack_push.h"
#include "pack_pop.h"
```

### Data Structures

- [struct iSgiHeader](#)

### Macros

- [#define SGI\\_COLMAP\\_COLMAP 3](#)
- [#define SGI\\_COLMAP\\_DITHERED 1](#)
- [#define SGI\\_COLMAP\\_NORMAL 0](#)
- [#define SGI\\_COLMAP\\_SCREEN 2](#)



- #define [SGI\\_MAGICNUM](#) 474
- #define [SGI\\_RLE](#) 1
- #define [SGI\\_VERBATIM](#) 0

## Typedefs

- typedef struct [iSgiHeader](#) [iSgiHeader](#)

## 7.83.1 Macro Definition Documentation

7.83.1.1 #define [SGI\\_COLMAP\\_COLMAP](#) 3

7.83.1.2 #define [SGI\\_COLMAP\\_DITHERED](#) 1

7.83.1.3 #define [SGI\\_COLMAP\\_NORMAL](#) 0

7.83.1.4 #define [SGI\\_COLMAP\\_SCREEN](#) 2

7.83.1.5 #define [SGI\\_MAGICNUM](#) 474

7.83.1.6 #define [SGI\\_RLE](#) 1

7.83.1.7 #define [SGI\\_VERBATIM](#) 0

## 7.83.2 Typedef Documentation

7.83.2.1 typedef struct [iSgiHeader](#) [iSgiHeader](#)

## 7.84 src/IL/formats/il\_sun.c File Reference

```
#include "il_internal.h"
#include "il_bits.h"
#include "pack_push.h"
#include "pack_pop.h"
```

## Data Structures

- struct [SUNHEAD](#)

## Macros

- #define [IL\\_SUN\\_BYTE\\_ENC](#) 0x02
- #define [IL\\_SUN\\_EXPER](#) 0xFFFF
- #define [IL\\_SUN\\_IFF](#) 0x05
- #define [IL\\_SUN\\_NO\\_MAP](#) 0x00
- #define [IL\\_SUN\\_OLD](#) 0x00
- #define [IL\\_SUN\\_RAW\\_MAP](#) 0x02
- #define [IL\\_SUN\\_RGB](#) 0x03
- #define [IL\\_SUN\\_RGB\\_MAP](#) 0x01
- #define [IL\\_SUN\\_STANDARD](#) 0x01
- #define [IL\\_SUN\\_TIFF](#) 0x04

## Typedefs

- typedef struct [SUNHEAD](#) [SUNHEAD](#)

## Variables

- [ILconst\\_string](#) [iFormatExtsSUN](#) []
- [ILformat](#) [iFormatSUN](#)

### 7.84.1 Macro Definition Documentation

7.84.1.1 `#define IL_SUN_BYTE_ENC 0x02`

7.84.1.2 `#define IL_SUN_EXPER 0xFFFF`

7.84.1.3 `#define IL_SUN_IFF 0x05`

7.84.1.4 `#define IL_SUN_NO_MAP 0x00`

7.84.1.5 `#define IL_SUN_OLD 0x00`

7.84.1.6 `#define IL_SUN_RAW_MAP 0x02`

7.84.1.7 `#define IL_SUN_RGB 0x03`

7.84.1.8 `#define IL_SUN_RGB_MAP 0x01`

7.84.1.9 `#define IL_SUN_STANDARD 0x01`

7.84.1.10 `#define IL_SUN_TIFF 0x04`

### 7.84.2 Typedef Documentation

7.84.2.1 typedef struct [SUNHEAD](#) [SUNHEAD](#)

### 7.84.3 Variable Documentation

7.84.3.1 [ILconst\\_string](#) [iFormatExtsSUN](#) []

**Initial value:**

```
= {
    IL\_TEXT("sun"),
    IL\_TEXT("ras"),
    IL\_TEXT("im1"),
    IL\_TEXT("im8"),
    IL\_TEXT("im24"),
    IL\_TEXT("im32"),
    NULL
}
```

7.84.3.2 [ILformat](#) [iFormatSUN](#)

**Initial value:**

```
= {  
    .Validate = iIsValidSun,  
    .Load     = iLoadSunInternal,  
    .Save     = NULL,  
    .Exts     = iFormatExtsSUN  
}
```

## 7.85 src/IL/formats/il\_targa.c File Reference

```
#include "il_internal.h"  
#include "il_targa.h"  
#include <time.h>  
#include <string.h>  
#include "il_manip.h"  
#include "il_endian.h"
```

### Variables

- [ILconst\\_string iFormatExtsTGA \[\]](#)
- [ILformat iFormatTGA](#)

### 7.85.1 Variable Documentation

#### 7.85.1.1 [ILconst\\_string iFormatExtsTGA\[\]](#)

##### Initial value:

```
= {  
    IL_TEXT("tga"),  
    IL_TEXT("vda"),  
    IL_TEXT("icb"),  
    IL_TEXT("vst"),  
    NULL  
}
```

#### 7.85.1.2 [ILformat iFormatTGA](#)

##### Initial value:

```
= {  
    .Validate = iIsValidTarga,  
    .Load     = iLoadTargaInternal,  
    .Save     = iSaveTargaInternal,  
    .Exts     = iFormatExtsTGA  
}
```

## 7.86 src/IL/formats/il\_targa.h File Reference

```
#include "il_internal.h"  
#include "pack_push.h"  
#include "pack_pop.h"
```

## Data Structures

- struct [TARGAEXT](#)
- struct [TARGAFOOTER](#)
- struct [TARGAHEAD](#)

## Macros

- #define [IMAGEDESC\\_BOTLEFT](#) 0x00
- #define [IMAGEDESC\\_BOTRIGHT](#) 0x10
- #define [IMAGEDESC\\_ORIGIN\\_MASK](#) 0x30
- #define [IMAGEDESC\\_TOPLEFT](#) 0x20
- #define [IMAGEDESC\\_TOPRIGHT](#) 0x30
- #define [TGA\\_BW\\_COMP](#) 11
- #define [TGA\\_BW\\_UNCOMP](#) 3
- #define [TGA\\_COLMAP\\_COMP](#) 9
- #define [TGA\\_COLMAP\\_UNCOMP](#) 1
- #define [TGA\\_EXT\\_LEN](#) 495
- #define [TGA\\_NO\\_DATA](#) 0
- #define [TGA\\_UNMAP\\_COMP](#) 10
- #define [TGA\\_UNMAP\\_UNCOMP](#) 2

## Typedefs

- typedef struct [TARGAEXT](#) [TARGAEXT](#)
- typedef struct [TARGAFOOTER](#) [TARGAFOOTER](#)
- typedef struct [TARGAHEAD](#) [TARGAHEAD](#)

### 7.86.1 Macro Definition Documentation

7.86.1.1 #define [IMAGEDESC\\_BOTLEFT](#) 0x00

7.86.1.2 #define [IMAGEDESC\\_BOTRIGHT](#) 0x10

7.86.1.3 #define [IMAGEDESC\\_ORIGIN\\_MASK](#) 0x30

7.86.1.4 #define [IMAGEDESC\\_TOPLEFT](#) 0x20

7.86.1.5 #define [IMAGEDESC\\_TOPRIGHT](#) 0x30

7.86.1.6 #define [TGA\\_BW\\_COMP](#) 11

7.86.1.7 #define [TGA\\_BW\\_UNCOMP](#) 3

7.86.1.8 #define [TGA\\_COLMAP\\_COMP](#) 9

7.86.1.9 #define [TGA\\_COLMAP\\_UNCOMP](#) 1

7.86.1.10 #define [TGA\\_EXT\\_LEN](#) 495

7.86.1.11 #define [TGA\\_NO\\_DATA](#) 0

7.86.1.12 #define [TGA\\_UNMAP\\_COMP](#) 10

7.86.1.13 `#define TGA_UNMAP_UNCOMP 2`

## 7.86.2 Typedef Documentation

7.86.2.1 `typedef struct TARGAEXT TARGAEXT`

7.86.2.2 `typedef struct TARGAFOOTER TARGAFOOTER`

7.86.2.3 `typedef struct TARGAHEAD TARGAHEAD`

## 7.87 src/IL/formats/il\_texture.c File Reference

```
#include "il_internal.h"
```

### Functions

- [ILboolean ilsValidDds \(SIO \\*io\)](#)
- [ILboolean iLoadDdsInternal \(ILImage \\*image\)](#)

### Variables

- [ILconst\\_string iFormatExtsTEXTURE \[\]](#)
- [ILformat iFormatTEXTURE](#)

## 7.87.1 Function Documentation

7.87.1.1 `ILboolean ilsValidDds ( SIO * io )`

7.87.1.2 `ILboolean iLoadDdsInternal ( ILImage * image )`

## 7.87.2 Variable Documentation

7.87.2.1 `ILconst_string iFormatExtsTEXTURE[]`

### Initial value:

```
= {
    IL_TEXT("texture"),
    NULL
}
```

## 7.87.2.2 ILformat iFormatTEXTURE

### Initial value:

```
= {
    .Validate = iIsValidTexture,
    .Load      = iLoadTextureInternal,
    .Save      = NULL,
    .Exts      = iFormatExtsTEXTURE
}
```

## 7.88 src/IL/formats/il\_tiff.c File Reference

```
#include "il_internal.h"
#include "tiffio.h"
#include <time.h>
#include "il_manip.h"
```

### Macros

- `#define MAGIC_HEADER1 0x4949`
- `#define MAGIC_HEADER2 0x4D4D`

### Functions

- `void errorHandler` (const char \*mod, const char \*fmt, va\_list ap)
- `ILboolean iSaveTiffInternal` (ILImage \*image)

### Variables

- `ILconst_string iFormatExtsTIF []`
- `ILformat iFormatTIF`

### 7.88.1 Macro Definition Documentation

7.88.1.1 `#define MAGIC_HEADER1 0x4949`

7.88.1.2 `#define MAGIC_HEADER2 0x4D4D`

### 7.88.2 Function Documentation

7.88.2.1 `void errorHandler ( const char * mod, const char * fmt, va_list ap )`

7.88.2.2 `ILboolean iSaveTiffInternal ( ILImage * image )`

### 7.88.3 Variable Documentation

7.88.3.1 `ILconst_string iFormatExtsTIF[]`

#### Initial value:

```
= {
    IL_TEXT("tif"),
    IL_TEXT("tiff"),
    NULL
}
```

7.88.3.2 `ILformat iFormatTIF`

#### Initial value:

```
= {
    .Validate = iIsValidTiff,
    .Load     = iLoadTiffInternal,
    .Save     = iSaveTiffInternal,
    .Exts     = iFormatExtsTIF
}
```

## 7.89 src/IL/formats/il\_tpl.c File Reference

```
#include "il_internal.h"
#include "il_dds.h"
```

### Data Structures

- struct [TPLHEAD](#)

### Macros

- #define [TPL\\_CI14X2](#) 10
- #define [TPL\\_CI4](#) 8
- #define [TPL\\_CI8](#) 9
- #define [TPL\\_CLAMP](#) 0
- #define [TPL\\_CMP](#) 14
- #define [TPL\\_I4](#) 0
- #define [TPL\\_I8](#) 1
- #define [TPL\\_IA4](#) 2
- #define [TPL\\_IA8](#) 3
- #define [TPL\\_MIRROR](#) 2
- #define [TPL\\_PAL\\_IA8](#) 0
- #define [TPL\\_PAL\\_RGB565](#) 1
- #define [TPL\\_PAL\\_RGB5A3](#) 2
- #define [TPL\\_REPEAT](#) 1
- #define [TPL\\_RGB565](#) 4
- #define [TPL\\_RGB5A3](#) 5
- #define [TPL\\_RGBA8](#) 6

### Typedefs

- typedef struct [TPLHEAD](#) [TPLHEAD](#)

### Variables

- [ILconst\\_string](#) [iFormatExtsTPL](#) []
- [ILformat](#) [iFormatTPL](#)

### 7.89.1 Macro Definition Documentation

7.89.1.1 #define [TPL\\_CI14X2](#) 10

7.89.1.2 #define [TPL\\_CI4](#) 8

7.89.1.3 #define [TPL\\_CI8](#) 9

7.89.1.4 #define [TPL\\_CLAMP](#) 0

7.89.1.5 #define [TPL\\_CMP](#) 14

7.89.1.6 #define [TPL\\_I4](#) 0

7.89.1.7 `#define TPL_I8 1`

7.89.1.8 `#define TPL_IA4 2`

7.89.1.9 `#define TPL_IA8 3`

7.89.1.10 `#define TPL_MIRROR 2`

7.89.1.11 `#define TPL_PAL_IA8 0`

7.89.1.12 `#define TPL_PAL_RGB565 1`

7.89.1.13 `#define TPL_PAL_RGB5A3 2`

7.89.1.14 `#define TPL_REPEAT 1`

7.89.1.15 `#define TPL_RGB565 4`

7.89.1.16 `#define TPL_RGB5A3 5`

7.89.1.17 `#define TPL_RGBA8 6`

## 7.89.2 Typedef Documentation

7.89.2.1 `typedef struct TPLHEAD TPLHEAD`

## 7.89.3 Variable Documentation

7.89.3.1 `ILconst_string iFormatExtsTPL[]`

**Initial value:**

```
= {
    IL_TEXT("tpl"),
    NULL
}
```

7.89.3.2 `ILformat iFormatTPL`

**Initial value:**

```
= {
    .Validate = iIsValidTpl,
    .Load     = iLoadTplInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsTPL
}
```

## 7.90 src/IL/formats/il\_utx.c File Reference

```
#include "il_internal.h"
#include "il_utx.h"
```



## Variables

- [ILconst\\_string iFormatExtsUTX \[\]](#)
- [ILformat iFormatUTX](#)

### 7.90.1 Variable Documentation

#### 7.90.1.1 ILconst\_string iFormatExtsUTX[]

##### Initial value:

```
= {
    IL_TEXT("utx"),
    NULL
}
```

#### 7.90.1.2 ILformat iFormatUTX

##### Initial value:

```
= {
    .Validate = iIsValidUtx,
    .Load     = iLoadUtxInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsUTX
}
```

## 7.91 src/IL/formats/il\_utx.h File Reference

```
#include "il_internal.h"
#include "il_dds.h"
#include "pack_push.h"
#include "pack_pop.h"
```

## Data Structures

- struct [UTXENTRYNAME](#)
- struct [UTXEXPORTTABLE](#)
- struct [UTXHEADER](#)
- struct [UTXIMPORTTABLE](#)
- struct [UTXPALETTE](#)

## Macros

- `#define UTX\_DXT1 0x03`
- `#define UTX\_P8 0x00`

## Typedefs

- typedef struct [UTXENTRYNAME](#) [UTXENTRYNAME](#)
- typedef struct [UTXEXPORTTABLE](#) [UTXEXPORTTABLE](#)
- typedef struct [UTXHEADER](#) [UTXHEADER](#)
- typedef struct [UTXIMPORTTABLE](#) [UTXIMPORTTABLE](#)

## 7.91.1 Macro Definition Documentation

7.91.1.1 `#define UTX_DXT1 0x03`

7.91.1.2 `#define UTX_P8 0x00`

## 7.91.2 Typedef Documentation

7.91.2.1 `typedef struct UTXENTRYNAME UTXENTRYNAME`

7.91.2.2 `typedef struct UTXEXPORTTABLE UTXEXPORTTABLE`

7.91.2.3 `typedef struct UTXHEADER UTXHEADER`

7.91.2.4 `typedef struct UTXIMPORTTABLE UTXIMPORTTABLE`

## 7.92 src/IL/formats/il\_vtf.c File Reference

```
#include "il_internal.h"
#include "il_vtf.h"
#include "il_dds.h"
```

### Functions

- [ILuint GetFaceFlag](#) ([ILuint](#) FaceNum)

### Variables

- [ILconst\\_string iFormatExtsVTF](#) []
- [ILformat iFormatVTF](#)

## 7.92.1 Function Documentation

7.92.1.1 [ILuint GetFaceFlag](#) ( [ILuint](#) *FaceNum* )

## 7.92.2 Variable Documentation

7.92.2.1 [ILconst\\_string iFormatExtsVTF](#) []

### Initial value:

```
= {
    IL\_TEXT("vtf"),
    NULL
}
```

7.92.2.2 [ILformat iFormatVTF](#)

### Initial value:

```
= {
    .Validate = iIsValidVtf,
    .Load      = iLoadVtfInternal,
```

```

    .Save      = iSaveVtfInternal,
    .Exts      = iFormatExtsVTF
}

```

## 7.93 src/IL/formats/il\_vtf.h File Reference

```

#include "il_internal.h"
#include "pack_push.h"
#include "pack_pop.h"

```

### Data Structures

- struct [VTFHEAD](#)

### Typedefs

- typedef struct [VTFHEAD](#) [VTFHEAD](#)

### Enumerations

- enum {  
[IMAGE\\_FORMAT\\_NONE](#) = -1, [IMAGE\\_FORMAT\\_RGBA8888](#) = 0, [IMAGE\\_FORMAT\\_ABGR8888](#), [IMAGE\\_FORMAT\\_RGB888](#),  
[IMAGE\\_FORMAT\\_BGR888](#), [IMAGE\\_FORMAT\\_RGB565](#), [IMAGE\\_FORMAT\\_I8](#), [IMAGE\\_FORMAT\\_IA88](#),  
[IMAGE\\_FORMAT\\_P8](#), [IMAGE\\_FORMAT\\_A8](#), [IMAGE\\_FORMAT\\_RGB888\\_BLUESCREEN](#), [IMAGE\\_FORMAT\\_BGR888\\_BLUESCREEN](#),  
[IMAGE\\_FORMAT\\_ARGB8888](#), [IMAGE\\_FORMAT\\_BGRA8888](#), [IMAGE\\_FORMAT\\_DXT1](#), [IMAGE\\_FORMAT\\_DXT3](#),  
[IMAGE\\_FORMAT\\_DXT5](#), [IMAGE\\_FORMAT\\_BGRX8888](#), [IMAGE\\_FORMAT\\_BGR565](#), [IMAGE\\_FORMAT\\_BGRX5551](#),  
[IMAGE\\_FORMAT\\_BGRA4444](#), [IMAGE\\_FORMAT\\_DXT1\\_ONEBITALPHA](#), [IMAGE\\_FORMAT\\_BGRA5551](#),  
[IMAGE\\_FORMAT\\_UV88](#),  
[IMAGE\\_FORMAT\\_UVWQ8888](#), [IMAGE\\_FORMAT\\_RGBA16161616F](#), [IMAGE\\_FORMAT\\_RGBA16161616](#),  
[IMAGE\\_FORMAT\\_UVLX8888](#) }
- enum {  
[TEXTUREFLAGS\\_POINTSAMPLE](#) = 0x00000001, [TEXTUREFLAGS\\_TRILINEAR](#) = 0x00000002, [TEXTUREFLAGS\\_CLAMP](#) = 0x00000004, [TEXTUREFLAGS\\_CLAMP\\_T](#) = 0x00000008,  
[TEXTUREFLAGS\\_ANISOTROPIC](#) = 0x00000010, [TEXTUREFLAGS\\_HINT\\_DXT5](#) = 0x00000020, [TEXTUREFLAGS\\_NOCOMPRESS](#) = 0x00000040, [TEXTUREFLAGS\\_NORMAL](#) = 0x00000080,  
[TEXTUREFLAGS\\_NOMIP](#) = 0x00000100, [TEXTUREFLAGS\\_NOLOD](#) = 0x00000200, [TEXTUREFLAGS\\_MINMIP](#) = 0x00000400, [TEXTUREFLAGS\\_PROCEDURAL](#) = 0x00000800,  
[TEXTUREFLAGS\\_ONEBITALPHA](#) = 0x00001000, [TEXTUREFLAGS\\_EIGHTBITALPHA](#) = 0x00002000, [TEXTUREFLAGS\\_ENVMAP](#) = 0x00004000, [TEXTUREFLAGS\\_RENDERTARGET](#) = 0x00008000,  
[TEXTUREFLAGS\\_DEPTHRENDERTARGET](#) = 0x00010000, [TEXTUREFLAGS\\_NODEBUGOVERRIDE](#) = 0x00020000, [TEXTUREFLAGS\\_SINGLECOPY](#) = 0x00040000, [TEXTUREFLAGS\\_ONEOVERMIPLEVEL](#) = 0x00080000,  
[TEXTUREFLAGS\\_PREMULTCOLORBYONEOVERMIPLEVEL](#) = 0x00100000, [TEXTUREFLAGS\\_NORMALTODUDV](#) = 0x00200000, [TEXTUREFLAGS\\_ALPHATESTMIPGENERATION](#) = 0x00400000, [TEXTUREFLAGS\\_NODEPTHBUFFER](#) = 0x00800000,  
[TEXTUREFLAGS\\_NICEFILTERED](#) = 0x01000000, [TEXTUREFLAGS\\_CLAMP\\_U](#) = 0x02000000 }

#### 7.93.1 Typedef Documentation

7.93.1.1 `typedef struct VTFHEAD VTFHEAD`

## 7.93.2 Enumeration Type Documentation

7.93.2.1 anonymous enum

Enumerator

***IMAGE\_FORMAT\_NONE***  
***IMAGE\_FORMAT\_RGBA8888***  
***IMAGE\_FORMAT\_ABGR8888***  
***IMAGE\_FORMAT\_RGB888***  
***IMAGE\_FORMAT\_BGR888***  
***IMAGE\_FORMAT\_RGB565***  
***IMAGE\_FORMAT\_I8***  
***IMAGE\_FORMAT\_IA88***  
***IMAGE\_FORMAT\_P8***  
***IMAGE\_FORMAT\_A8***  
***IMAGE\_FORMAT\_RGB888\_BLUESCREEN***  
***IMAGE\_FORMAT\_BGR888\_BLUESCREEN***  
***IMAGE\_FORMAT\_ARGB8888***  
***IMAGE\_FORMAT\_BGRA8888***  
***IMAGE\_FORMAT\_DXT1***  
***IMAGE\_FORMAT\_DXT3***  
***IMAGE\_FORMAT\_DXT5***  
***IMAGE\_FORMAT\_BGRX8888***  
***IMAGE\_FORMAT\_BGR565***  
***IMAGE\_FORMAT\_BGRX5551***  
***IMAGE\_FORMAT\_BGRA4444***  
***IMAGE\_FORMAT\_DXT1\_ONEBITALPHA***  
***IMAGE\_FORMAT\_BGRA5551***  
***IMAGE\_FORMAT\_UV88***  
***IMAGE\_FORMAT\_UVWQ8888***  
***IMAGE\_FORMAT\_RGBA16161616F***  
***IMAGE\_FORMAT\_RGBA16161616***  
***IMAGE\_FORMAT\_UVLX8888***

7.93.2.2 anonymous enum

Enumerator

***TEXTUREFLAGS\_POINTSAMPLE***  
***TEXTUREFLAGS\_TRILINEAR***  
***TEXTUREFLAGS\_CLAMP***  
***TEXTUREFLAGS\_CLAMPPT***  
***TEXTUREFLAGS\_ANISOTROPIC***  
***TEXTUREFLAGS\_HINT\_DXT5***  
***TEXTUREFLAGS\_NOCOMPRESS***

***TEXTUREFLAGS\_NORMAL***  
***TEXTUREFLAGS\_NOMIP***  
***TEXTUREFLAGS\_NOLOD***  
***TEXTUREFLAGS\_MINMIP***  
***TEXTUREFLAGS\_PROCEDURAL***  
***TEXTUREFLAGS\_ONEBITALPHA***  
***TEXTUREFLAGS\_EIGHTBITALPHA***  
***TEXTUREFLAGS\_ENVMAP***  
***TEXTUREFLAGS\_RENDERTARGET***  
***TEXTUREFLAGS\_DEPTHRENDERTARGET***  
***TEXTUREFLAGS\_NODEBUGOVERRIDE***  
***TEXTUREFLAGS\_SINGLECOPY***  
***TEXTUREFLAGS\_ONEOVERMIPLEVELINALPHA***  
***TEXTUREFLAGS\_PREMULTCOLORBYONEOVERMIPLEVEL***  
***TEXTUREFLAGS\_NORMALTODUDV***  
***TEXTUREFLAGS\_ALPHATESTMIPGENERATION***  
***TEXTUREFLAGS\_NODEPTHBUFFER***  
***TEXTUREFLAGS\_NICEFILTERED***  
***TEXTUREFLAGS\_CLAMP\_U***

## 7.94 src/IL/formats/il\_wal.c File Reference

```
#include "il_internal.h"  
#include "il_manip.h"  
#include "il_q2pal.h"  
#include "pack_push.h"  
#include "pack_pop.h"
```

### Data Structures

- struct [WALHEAD](#)

### Typedefs

- typedef struct [WALHEAD](#) [WALHEAD](#)

### Variables

- [ILconst\\_string](#) [iFormatExtsWAL](#) []
- [ILformat](#) [iFormatWAL](#)

### 7.94.1 Typedef Documentation

#### 7.94.1.1 typedef struct WALHEAD WALHEAD

### 7.94.2 Variable Documentation

#### 7.94.2.1 ILconst\_string iFormatExtsWAL[]

**Initial value:**

```
= {
    IL_TEXT("wal"),
    NULL
}
```

#### 7.94.2.2 ILformat iFormatWAL

**Initial value:**

```
= {
    .Validate = NULL,
    .Load     = iLoadWalInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsWAL
}
```

## 7.95 src/IL/formats/il\_wbmp.c File Reference

```
#include "il_internal.h"
#include "il_bits.h"
```

### Functions

- [ILimage \\* iNeuQuant \(ILimage \\*Image, ILuint NumCols\)](#)
- [ILimage \\* iQuantizeImage \(ILimage \\*Image, ILuint NumCols\)](#)

### Variables

- [ILconst\\_string iFormatExtsWBMP \[\]](#)
- [ILformat iFormatWBMP](#)

### 7.95.1 Function Documentation

#### 7.95.1.1 ILimage\* iNeuQuant ( ILimage \* *Image*, ILuint *NumCols* )

#### 7.95.1.2 ILimage\* iQuantizeImage ( ILimage \* *Image*, ILuint *NumCols* )

### 7.95.2 Variable Documentation

#### 7.95.2.1 ILconst\_string iFormatExtsWBMP[]

**Initial value:**

```
= {
    IL_TEXT("wbmp"),
    NULL
}
```

### 7.95.2.2 ILformat iFormatWBMP

**Initial value:**

```
= {
    .Validate = iIsValidWbmp,
    .Load     = iLoadWbmpInternal,
    .Save     = iSaveWbmpInternal,
    .Exts     = iFormatExtsWBMP
}
```

## 7.96 src/IL/formats/il\_wdp.c File Reference

```
#include "il_internal.h"
#include <WMPGlue.h>
#include "il_wdp.h"
```

### Functions

- [ERR iCloseWS\\_File](#) (struct WMPStream \*\*ppWS)
- [Bool iEOSWS\\_File](#) (struct WMPStream \*pWS)
- [ERR iGetPosWS\\_File](#) (struct WMPStream \*pWS, size\_t \*poffPos)
- [ERR ilCreateWS\\_File](#) (struct WMPStream \*\*ppWS, const char \*szFilename, const char \*szMode)
- [ILboolean ilLoadWdp](#) (ILconst\_string FileName)  
*Reads a WDP file.*
- [ILboolean ilLoadWdpF](#) (ILHANDLE File)  
*Reads an already-opened WDP file.*
- [ILboolean ilLoadWdpL](#) (const void \*Lump, ILuint Size)  
*Reads from a memory "lump" that contains a WDP.*
- [ILboolean ilLoadWdpInternal](#) ()
- [ERR ilPKCodecFactory\\_CreateDecoderFromFile](#) (PKImageDecode \*\*ppDecoder)
- [ERR ilPKCreateFactory](#) (PKFactory \*\*ppFactory, U32 uVersion)
- [ERR ilPKImageEncode\\_WritePixels\\_DevIL](#) (PKImageEncode \*pIE, U32 cLine, U8 \*pbPixel, U32 cbStride)
- [ERR iReadWS\\_File](#) (struct WMPStream \*pWS, void \*pv, size\_t cb)
- [ERR iSetPosWS\\_File](#) (struct WMPStream \*pWS, size\_t offPos)
- [ERR iWmpDecAppCreateEncoderFromExt](#) (PKCodecFactory \*pCFactory, const char \*szExt, PKImageEncode \*\*ppIE)
- [ERR iWriteWS\\_File](#) (struct WMPStream \*pWS, const void \*pv, size\_t cb)
- [ERR PKImageEncode\\_Create\\_DevIL](#) (PKImageEncode \*\*ppIE)
- [ERR WriteDevILHeader](#) (PKImageEncode \*pIE)

### 7.96.1 Function Documentation

**7.96.1.1** [ERR iCloseWS\\_File](#) ( struct WMPStream \*\* ppWS )

**7.96.1.2** [Bool iEOSWS\\_File](#) ( struct WMPStream \* pWS )

7.96.1.3 `ERR iGetPosWS_File ( struct WMPStream * pWS, size_t * poffPos )`

7.96.1.4 `ERR ilCreateWS_File ( struct WMPStream ** ppWS, const char * szFilename, const char * szMode )`

7.96.1.5 `ILboolean ilLoadWdp ( ILconst_string FileName )`

Reads a WDP file.

7.96.1.6 `ILboolean ilLoadWdpF ( ILHANDLE File )`

Reads an already-opened WDP file.

7.96.1.7 `ILboolean ilLoadWdpL ( const void * Lump, ILuint Size )`

Reads from a memory "lump" that contains a WDP.

7.96.1.8 `ILboolean iLoadWdpInternal ( )`

7.96.1.9 `ERR ilPKCodecFactory_CreateDecoderFromFile ( PKImageDecode ** ppDecoder )`

7.96.1.10 `ERR ilPKCreateFactory ( PKFactory ** ppFactory, U32 uVersion )`

7.96.1.11 `ERR ilPKImageEncode_WritePixels_DevIL ( PKImageEncode * pIE, U32 cLine, U8 * pbPixel, U32 cbStride )`

7.96.1.12 `ERR iReadWS_File ( struct WMPStream * pWS, void * pv, size_t cb )`

7.96.1.13 `ERR iSetPosWS_File ( struct WMPStream * pWS, size_t offPos )`

7.96.1.14 `ERR iWmpDecAppCreateEncoderFromExt ( PKCodecFactory * pCFactory, const char * szExt, PKImageEncode ** ppIE )`

7.96.1.15 `ERR iWriteWS_File ( struct WMPStream * pWS, const void * pv, size_t cb )`

7.96.1.16 `ERR PKImageEncode_Create_DevIL ( PKImageEncode ** ppIE )`

7.96.1.17 `ERR WriteDevILHeader ( PKImageEncode * pIE )`

## 7.97 `src/IL/formats/il_wdp.h` File Reference

```
#include "il_internal.h"
```

### Data Structures

- struct [WDPDCQUANT](#)
- struct [WDPGUID](#)
- struct [WDPHEAD](#)
- struct [WDPIFD](#)
- struct [WDPIMGHEAD](#)
- struct [WDPIMGPLANE](#)
- struct [WDPTILE](#)



## Macros

- #define [WDP\\_ALPHACHANNEL](#) 0x01
- #define [WDP\\_BANDS\\_PRESENT](#) 0x0F
- #define [WDP\\_BAYER](#) 0x05
- #define [WDP\\_BD\\_10](#) 0x09
- #define [WDP\\_BD\\_16](#) 0x02
- #define [WDP\\_BD\\_16F](#) 0x04
- #define [WDP\\_BD\\_16S](#) 0x03
- #define [WDP\\_BD\\_1\\_BLACK](#) 0x0F
- #define [WDP\\_BD\\_1\\_WHITE](#) 0x00
- #define [WDP\\_BD\\_32](#) 0x05
- #define [WDP\\_BD\\_32F](#) 0x07
- #define [WDP\\_BD\\_32S](#) 0x06
- #define [WDP\\_BD\\_5](#) 0x08
- #define [WDP\\_BD\\_565](#) 0x0A
- #define [WDP\\_BD\\_8](#) 0x01
- #define [WDP\\_BITDEPTH](#) 0x0F
- #define [WDP\\_BITSTREAM\\_FMT](#) 0x40
- #define [WDP\\_CH\\_INDEPENDENT](#) 0x02
- #define [WDP\\_CH\\_SEPARATE](#) 0x01
- #define [WDP\\_CH\\_UNIFORM](#) 0x00
- #define [WDP\\_CLR\\_FMT](#) 0xE0
- #define [WDP\\_CMYK](#) 0x04
- #define [WDP\\_CODEC](#) 0xF0
- #define [WDP\\_COLOR\\_INTERP](#) 0x0F
- #define [WDP\\_DC\\_FRAME](#) 0x80
- #define [WDP\\_DC\\_TILE](#) 0x01
- #define [WDP\\_FLEXBITS\\_TILE](#) 0x04
- #define [WDP\\_FORMAT](#) 0xF0
- #define [WDP\\_HIGHPASS\\_TILE](#) 0x03
- #define [WDP\\_INDEXTABLE](#) 0x04
- #define [WDP\\_LONG\\_WORD](#) 0x40
- #define [WDP\\_LOWPASS\\_TILE](#) 0x02
- #define [WDP\\_N\\_CHANNEL](#) 0x06
- #define [WDP\\_NO\\_SCALED](#) 0x10
- #define [WDP\\_NUM\\_CHANS](#) 0xF0
- #define [WDP\\_ORIENTATION](#) 0x38
- #define [WDP\\_OVERLAP](#) 0x03
- #define [WDP\\_RGB](#) 0x07
- #define [WDP\\_RGBE](#) 0x08
- #define [WDP\\_SB\\_ALL](#) 0x00
- #define [WDP\\_SB\\_DC\\_ONLY](#) 0x03
- #define [WDP\\_SB\\_ISOLATED](#) 0x04
- #define [WDP\\_SB\\_NO\\_FLEXBITS](#) 0x01
- #define [WDP\\_SB\\_NO\\_HIGHPASS](#) 0x02
- #define [WDP\\_SHORT\\_HEADER](#) 0x80
- #define [WDP\\_SPATIAL\\_TILE](#) 0x00
- #define [WDP\\_SUBCODEC](#) 0x0F
- #define [WDP\\_TILE\\_HASH](#) 0xF8
- #define [WDP\\_TILE\\_STRETCH](#) 0x08
- #define [WDP\\_TILE\\_TYPE](#) 0x03
- #define [WDP\\_TILING\\_FLAG](#) 0x80
- #define [WDP\\_TRIM\\_FLEXBITS](#) 0x10
- #define [WDP\\_WINDOWING](#) 0x20

- `#define WDP_Y_ONLY 0x00`
- `#define WDP_YUV_420 0x01`
- `#define WDP_YUV_422 0x02`
- `#define WDP_YUV_444 0x03`

## Typedefs

- `typedef struct WDPDCQUANT WDPDCQUANT`
- `typedef struct WDPGUID WDPGUID`
- `typedef struct WDPHEAD WDPHEAD`
- `typedef struct WDPIFD WDPIFD`
- `typedef struct WDPIMGHEAD WDPIMGHEAD`
- `typedef struct WDPIMGPLANE WDPIMGPLANE`
- `typedef struct WDPTILE WDPTILE`

## Functions

- `ILboolean iCheckWdp (WDPHEAD *Header)`
- `ILboolean ilsValidWdp ()`
- `ILboolean iLoadWdpInternal ()`
- `ILuint VLWESC ()`

### 7.97.1 Macro Definition Documentation

7.97.1.1 `#define WDP_ALPHACHANNEL 0x01`

7.97.1.2 `#define WDP_BANDS_PRESENT 0x0F`

7.97.1.3 `#define WDP_BAYER 0x05`

7.97.1.4 `#define WDP_BD_10 0x09`

7.97.1.5 `#define WDP_BD_16 0x02`

7.97.1.6 `#define WDP_BD_16F 0x04`

7.97.1.7 `#define WDP_BD_16S 0x03`

7.97.1.8 `#define WDP_BD_1_BLACK 0x0F`

7.97.1.9 `#define WDP_BD_1_WHITE 0x00`

7.97.1.10 `#define WDP_BD_32 0x05`

7.97.1.11 `#define WDP_BD_32F 0x07`

7.97.1.12 `#define WDP_BD_32S 0x06`

7.97.1.13 `#define WDP_BD_5 0x08`

7.97.1.14 `#define WDP_BD_565 0x0A`

7.97.1.15 `#define WDP_BD_8 0x01`

7.97.1.16 `#define WDP_BITDEPTH 0x0F`

7.97.1.17 `#define WDP_BITSTREAM_FMT 0x40`

7.97.1.18 `#define WDP_CH_INDEPENDENT 0x02`

7.97.1.19 `#define WDP_CH_SEPARATE 0x01`

7.97.1.20 `#define WDP_CH_UNIFORM 0x00`

7.97.1.21 `#define WDP_CLR_FMT 0xE0`

7.97.1.22 `#define WDP_CMYK 0x04`

7.97.1.23 `#define WDP_CODEC 0xF0`

7.97.1.24 `#define WDP_COLOR_INTERP 0x0F`

7.97.1.25 `#define WDP_DC_FRAME 0x80`

7.97.1.26 `#define WDP_DC_TILE 0x01`

7.97.1.27 `#define WDP_FLEXBITS_TILE 0x04`

7.97.1.28 `#define WDP_FORMAT 0xF0`

7.97.1.29 `#define WDP_HIGHPASS_TILE 0x03`

7.97.1.30 `#define WDP_INDEXTABLE 0x04`

7.97.1.31 `#define WDP_LONG_WORD 0x40`

7.97.1.32 `#define WDP_LOWPASS_TILE 0x02`

7.97.1.33 `#define WDP_N_CHANNEL 0x06`

7.97.1.34 `#define WDP_NO_SCALED 0x10`

7.97.1.35 `#define WDP_NUM_CHANS 0xF0`

7.97.1.36 `#define WDP_ORIENTATION 0x38`

7.97.1.37 `#define WDP_OVERLAP 0x03`

7.97.1.38 `#define WDP_RGB 0x07`

7.97.1.39 `#define WDP_RGBE 0x08`

7.97.1.40 `#define WDP_SB_ALL 0x00`

7.97.1.41 `#define WDP_SB_DC_ONLY 0x03`

7.97.1.42 `#define WDP_SB_ISOLATED 0x04`

7.97.1.43 `#define WDP_SB_NO_FLEXBITS 0x01`

7.97.1.44 `#define WDP_SB_NO_HIGHPASS 0x02`

7.97.1.45 `#define WDP_SHORT_HEADER 0x80`

7.97.1.46 `#define WDP_SPATIAL_TILE 0x00`

7.97.1.47 `#define WDP_SUBCODEC 0x0F`

7.97.1.48 `#define WDP_TILE_HASH 0xF8`

7.97.1.49 `#define WDP_TILE_STRETCH 0x08`

7.97.1.50 `#define WDP_TILE_TYPE 0x03`

7.97.1.51 `#define WDP_TILING_FLAG 0x80`

7.97.1.52 `#define WDP_TRIM_FLEXBITS 0x10`

7.97.1.53 `#define WDP_WINDOWING 0x20`

7.97.1.54 `#define WDP_Y_ONLY 0x00`

7.97.1.55 `#define WDP_YUV_420 0x01`

7.97.1.56 `#define WDP_YUV_422 0x02`

7.97.1.57 `#define WDP_YUV_444 0x03`

## 7.97.2 Typedef Documentation

7.97.2.1 `typedef struct WDPDCQUANT WDPDCQUANT`

7.97.2.2 `typedef struct WDPGUID WDPGUID`

7.97.2.3 `typedef struct WDPHEAD WDPHEAD`

7.97.2.4 `typedef struct WDPIFD WDPIFD`

7.97.2.5 `typedef struct WDPIMGHEAD WDPIMGHEAD`

7.97.2.6 `typedef struct WDPIMGPLANE WDPIMGPLANE`

7.97.2.7 `typedef struct WDPTILE WDPTILE`

## 7.97.3 Function Documentation

7.97.3.1 `ILboolean iCheckWdp ( WDPHEAD * Header )`

7.97.3.2 `ILboolean ilsValidWdp ( )`

7.97.3.3 `ILboolean iLoadWdpInternal ( )`

7.97.3.4 `ILuint VLWESC ( )`

## 7.98 src/IL/formats/il\_xpm.c File Reference

```
#include "il_internal.h"
#include <ctype.h>
```

### Data Structures

- struct [XPMHASHENTRY](#)

### Macros

- #define [BUFFER\\_SIZE](#) 2000
- #define [XPM\\_HASH\\_LEN](#) 257
- #define [XPM\\_MAX\\_CHAR\\_PER\\_PIXEL](#) 3

### Typedefs

- typedef struct [XPMHASHENTRY](#) [XPMHASHENTRY](#)
- typedef [ILubyte](#) [XpmPixel](#) [4]

### Functions

- [XPMHASHENTRY](#) \*\* [XpmCreateHashTable](#) ()
- void [XpmDestroyHashTable](#) ([XPMHASHENTRY](#) \*\*Table)
- ILboolean [XpmGetColour](#) ([ILubyte](#) \*Buffer, [ILint](#) Size, int Len, [XPMHASHENTRY](#) \*\*Table)
- void [XpmGetEntry](#) ([XPMHASHENTRY](#) \*\*Table, const [ILubyte](#) \*Name, int Len, [XpmPixel](#) Colour)
- [ILint](#) [XpmGetInt](#) ([ILubyte](#) \*Buffer, [ILint](#) Size, [ILint](#) \*Position)
- [ILint](#) [XpmGets](#) (SIO \*io, [ILubyte](#) \*Buffer, [ILint](#) MaxLen)
- void [XpmInsertEntry](#) ([XPMHASHENTRY](#) \*\*Table, const [ILubyte](#) \*Name, int Len, [XpmPixel](#) Colour)
- ILboolean [XpmPredefCol](#) (char \*Buff, [XpmPixel](#) \*Colour)

### Variables

- [ILconst\\_string](#) [iFormatExtsXPM](#) []
- [ILformat](#) [iFormatXPM](#)

#### 7.98.1 Macro Definition Documentation

7.98.1.1 #define [BUFFER\\_SIZE](#) 2000

7.98.1.2 #define [XPM\\_HASH\\_LEN](#) 257

7.98.1.3 #define [XPM\\_MAX\\_CHAR\\_PER\\_PIXEL](#) 3

#### 7.98.2 Typedef Documentation

7.98.2.1 typedef struct [XPMHASHENTRY](#) [XPMHASHENTRY](#)

7.98.2.2 typedef [ILubyte](#) [XpmPixel](#)[4]

### 7.98.3 Function Documentation

7.98.3.1 **XPMHASHENTRY\*\*** XpmCreateHashTable ( )

7.98.3.2 **void** XpmDestroyHashTable ( **XPMHASHENTRY\*\*** *Table* )

7.98.3.3 **ILboolean** XpmGetColour ( **ILubyte\*** *Buffer*, **ILint** *Size*, **int** *Len*, **XPMHASHENTRY\*\*** *Table* )

7.98.3.4 **void** XpmGetEntry ( **XPMHASHENTRY\*\*** *Table*, **const ILubyte\*** *Name*, **int** *Len*, **XpmPixel** *Colour* )

7.98.3.5 **ILint** XpmGetInt ( **ILubyte\*** *Buffer*, **ILint** *Size*, **ILint\*** *Position* )

7.98.3.6 **ILint** XpmGets ( **SIO\*** *io*, **ILubyte\*** *Buffer*, **ILint** *MaxLen* )

7.98.3.7 **void** XpmInsertEntry ( **XPMHASHENTRY\*\*** *Table*, **const ILubyte\*** *Name*, **int** *Len*, **XpmPixel** *Colour* )

7.98.3.8 **ILboolean** XpmPredefCol ( **char\*** *Buff*, **XpmPixel\*** *Colour* )

### 7.98.4 Variable Documentation

7.98.4.1 **ILconst\_string** iFormatExtsXPM[]

**Initial value:**

```
= {
    IL_TEXT("xpm"),
    NULL
}
```

7.98.4.2 **ILformat** iFormatXPM

**Initial value:**

```
= {
    .Validate = iIsValidXpm,
    .Load     = iLoadXpmInternal,
    .Save     = NULL,
    .Exts     = iFormatExtsXPM
}
```

## 7.99 src/IL/il\_alloc.c File Reference

```
#include "il_internal.h"
#include <stdlib.h>
#include <math.h>
```

### Macros

- **#define** `__ALLOC_C`

### Functions

- **void\*** `ILAPIENTRY ilalloc` (**const ILsizei** *Size*)
- **void\*** `ILAPIENTRY icalloc` (**const ILsizei** *Size*, **const ILsizei** *Num*)

- [void ILAPIENTRY ifree](#) (void \*Ptr)
- [void ILAPIENTRY ilResetMemory](#) ()
- [void iSetMemory](#) (mAlloc AllocFunc, mFree FreeFunc)

## 7.99.1 Macro Definition Documentation

### 7.99.1.1 #define \_\_ALLOC\_C

## 7.99.2 Function Documentation

### 7.99.2.1 void\* ILAPIENTRY ialloc ( const ILsizei Size )

### 7.99.2.2 void\* ILAPIENTRY icalloc ( const ILsizei Size, const ILsizei Num )

### 7.99.2.3 void ILAPIENTRY ifree ( void \* Ptr )

### 7.99.2.4 void ILAPIENTRY ilResetMemory ( )

### 7.99.2.5 void iSetMemory ( mAlloc AllocFunc, mFree FreeFunc )

## 7.100 src/IL/il\_alloc.h File Reference

```
#include <IL/il.h>
```

## Macros

- #define [\\_\\_ALLOC\\_EXTERN](#) extern

## Functions

- [void iSetMemory](#) (mAlloc AllocFunc, mFree FreeFunc)

## Variables

- [\\_\\_ALLOC\\_EXTERN](#) mAlloc ialloc\_ptr
- [\\_\\_ALLOC\\_EXTERN](#) mFree ifree\_ptr

## 7.100.1 Macro Definition Documentation

### 7.100.1.1 #define \_\_ALLOC\_EXTERN extern

## 7.100.2 Function Documentation

### 7.100.2.1 void iSetMemory ( mAlloc AllocFunc, mFree FreeFunc )

## 7.100.3 Variable Documentation

### 7.100.3.1 \_\_ALLOC\_EXTERN mAlloc ialloc\_ptr

### 7.100.3.2 \_\_ALLOC\_EXTERN mFree ifree\_ptr

## 7.101 src/IL/il\_api.c File Reference

Contains public IL entry functions.

```
#include "il_internal.h"
#include "il_stack.h"
#include "il_states.h"
#include "il_alloc.h"
#include "il_manip.h"
```

### Functions

- [void ILAPIENTRY ilBindImage \(ILuint Image\)](#)  
*Makes Image the current active image - similar to glBindTexture().*
- [ILuint ILAPIENTRY ilCloneCurlImage \(\)](#)  
*Creates a duplicate of the currently bound image.*
- [ILboolean ILAPIENTRY ilCopyImage \(ILuint Src\)](#)  
*Copies everything from Src to the current bound image.*
- [ILuint ILAPIENTRY ilCopyPixels \(ILuint XOff, ILuint YOff, ILuint ZOff, ILuint Width, ILuint Height, ILuint Depth, ILenum Format, ILenum Type, void \\*Data\)](#)  
*Copy the pixels of a region of the currently bound image to a buffer.*
- [ILuint ILAPIENTRY ilCreateSubImage \(ILenum Type, ILuint Num\)](#)  
*Creates sub images of the given type for the currently bound image.*
- [ILboolean ILAPIENTRY ilDefaultImage \(\)](#)  
*Creates an ugly 64x64 black and yellow checkerboard image.*
- [ILboolean ILAPIENTRY ilDisable \(ILenum Mode\)](#)  
*Disables a mode.*
- [ILboolean ILAPIENTRY ilEnable \(ILenum Mode\)](#)  
*Enables a mode.*
- [ILboolean ILAPIENTRY ilFormatFunc \(ILenum Mode\)](#)  
*Set the default image format to use.*
- [ILboolean ILAPIENTRY ilGetBoolean \(ILenum Mode\)](#)  
*Returns the current value of the Mode.*
- [void ILAPIENTRY ilGetBooleanv \(ILenum Mode, ILboolean \\*Param\)](#)  
*Sets Param equal to the current value of the Mode.*
- [ILenum ILAPIENTRY ilGetError \(void\)](#)  
*Gets the last error on the error stack.*
- [ILint ILAPIENTRY ilGetInteger \(ILenum Mode\)](#)  
*Returns the current value of the Mode.*
- [ILint ILAPIENTRY ilGetIntegerImage \(ILuint Image, ILenum Mode\)](#)  
*Get a value about a specific image.*
- [void ILAPIENTRY ilGetIntegerv \(ILenum Mode, ILint \\*Param\)](#)  
*Sets Param equal to the current value of the Mode.*
- [ILconst\\_string ILAPIENTRY ilGetString \(ILenum StringName\)](#)  
*Returns a constant string detailing aspects about this library.*
- [void ILAPIENTRY ilHint \(ILenum Target, ILenum Mode\)](#)  
*Specifies implementation-dependent performance hints.*
- [void ILAPIENTRY ilInit \(void\)](#)  
*Initialize the image library.*
- [ILboolean ILAPIENTRY ilIsDisabled \(ILenum Mode\)](#)  
*Checks whether a Mode is not enabled.*



- [ILboolean ILAPIENTRY ilIsEnabled](#) ([ILenum](#) Mode)  
*Checks whether a Mode is enabled.*
- [ILboolean ILAPIENTRY ilIsImage](#) ([ILuint](#) Image)  
*Checks whether a given Image name is valid.*
- [void ILAPIENTRY ilSetInteger](#) ([ILenum](#) Mode, [ILint](#) Param)  
*Sets a parameter value for a Mode.*
- [void ILAPIENTRY ilSetMemory](#) ([mAlloc](#) mallocFunc, [mFree](#) freeFunc)  
*Sets the memory allocation and deallocation functions.*
- [void ILAPIENTRY ilSetString](#) ([ILenum](#) StringName, const char \*String)  
*Sets a string detailing aspects about this library.*
- [void ILAPIENTRY ilShutDown](#) ([void](#))  
*Shuts down the image library.*

### 7.101.1 Detailed Description

Contains public IL entry functions. Just calls the internal versions of the functions for now.

## 7.102 src/IL/il\_bits.c File Reference

```
#include "il_internal.h"
#include "il_bits.h"
```

### Functions

- [ILint bclose](#) ([BITFILE](#) \*BitFile)
- [BITFILE \\* bitfile](#) ([SIO](#) \*io)
- [ILint bread](#) ([void](#) \*Buffer, [ILuint](#) Size, [ILuint](#) Number, [BITFILE](#) \*BitFile)
- [ILuint breadVal](#) ([ILuint](#) NumBits, [BITFILE](#) \*BitFile)
- [ILint bseek](#) ([BITFILE](#) \*BitFile, [ILuint](#) Offset, [ILuint](#) Mode)
- [ILint btell](#) ([BITFILE](#) \*BitFile)

### 7.102.1 Function Documentation

7.102.1.1 [ILint bclose](#) ( [BITFILE](#) \* *BitFile* )

7.102.1.2 [BITFILE\\*](#) *bitfile* ( [SIO](#) \* *io* )

7.102.1.3 [ILint bread](#) ( [void](#) \* *Buffer*, [ILuint](#) *Size*, [ILuint](#) *Number*, [BITFILE](#) \* *BitFile* )

7.102.1.4 [ILuint breadVal](#) ( [ILuint](#) *NumBits*, [BITFILE](#) \* *BitFile* )

7.102.1.5 [ILint bseek](#) ( [BITFILE](#) \* *BitFile*, [ILuint](#) *Offset*, [ILuint](#) *Mode* )

7.102.1.6 [ILint btell](#) ( [BITFILE](#) \* *BitFile* )

## 7.103 src/IL/il\_bits.h File Reference

```
#include "il_internal.h"
```

## Data Structures

- struct [BITFILE](#)

## Macros

- #define [ClearBits](#)(var, bits) (var &= ~(bits))
- #define [SetBits](#)(var, bits) (var |= bits)

## Typedefs

- typedef struct [BITFILE](#) [BITFILE](#)

## Functions

- [ILint bclose](#) ([BITFILE](#) \*BitFile)
- [BITFILE \\* bitfile](#) ([SIO](#) \*io)
- [ILint bread](#) ([void](#) \*Buffer, [ILuint](#) Size, [ILuint](#) Number, [BITFILE](#) \*BitFile)
- [ILint bseek](#) ([BITFILE](#) \*BitFile, [ILuint](#) Offset, [ILuint](#) Mode)
- [ILint btell](#) ([BITFILE](#) \*BitFile)

### 7.103.1 Macro Definition Documentation

7.103.1.1 #define [ClearBits](#)( var, bits ) (var &= ~(bits))

7.103.1.2 #define [SetBits](#)( var, bits ) (var |= bits)

### 7.103.2 Typedef Documentation

7.103.2.1 typedef struct [BITFILE](#) [BITFILE](#)

### 7.103.3 Function Documentation

7.103.3.1 [ILint bclose](#) ( [BITFILE](#) \* *BitFile* )

7.103.3.2 [BITFILE\\*](#) [bitfile](#) ( [SIO](#) \* *io* )

7.103.3.3 [ILint bread](#) ( [void](#) \* *Buffer*, [ILuint](#) *Size*, [ILuint](#) *Number*, [BITFILE](#) \* *BitFile* )

7.103.3.4 [ILint bseek](#) ( [BITFILE](#) \* *BitFile*, [ILuint](#) *Offset*, [ILuint](#) *Mode* )

7.103.3.5 [ILint btell](#) ( [BITFILE](#) \* *BitFile* )

## 7.104 src/IL/il\_endian.c File Reference

```
#include "il_endian.h"
```

## Macros

- #define [IL\\_ENDIAN\\_C](#)

## Functions

- [void iEndianSwapData \(ILImage \\*Image\)](#)

### 7.104.1 Macro Definition Documentation

#### 7.104.1.1 `#define IL_ENDIAN_C`

### 7.104.2 Function Documentation

#### 7.104.2.1 `void iEndianSwapData ( ILImage * Image )`

## 7.105 src/IL/il\_endian.h File Reference

```
#include "il_internal.h"
```

## Macros

- `#define BigDouble\(d\) iSwapDouble\(d\)`
- `#define BigFloat\(f\) iSwapFloat\(f\)`
- `#define BigInt\(i\) iSwapInt\(i\)`
- `#define BigShort\(s\) iSwapShort\(s\)`
- `#define BigUInt\(i\) iSwapUInt\(i\)`
- `#define BigUShort\(s\) iSwapUShort\(s\)`
- `#define Double\(d\)`
- `#define dswap\(x, y\) t=b\[x\]; b\[x\]=b\[y\]; b\[y\]=t;`
- `#define Float\(f\)`
- `#define Int\(i\)`
- `#define Short\(s\)`
- `#define UInt\(i\)`
- `#define UShort\(s\)`

## Functions

- `INLINE ILdouble GetBigDouble \(SIO \*io\)`
- `INLINE ILfloat GetBigFloat \(SIO \*io\)`
- `INLINE ILint GetBigInt \(SIO \*io\)`
- `INLINE ILshort GetBigShort \(SIO \*io\)`
- `INLINE ILuint GetBigUInt \(SIO \*io\)`
- `INLINE ILushort GetBigUShort \(SIO \*io\)`
- `INLINE ILdouble GetLittleDouble \(SIO \*io\)`
- `INLINE ILfloat GetLittleFloat \(SIO \*io\)`
- `INLINE ILint GetLittleInt \(SIO \*io\)`
- `INLINE ILshort GetLittleShort \(SIO \*io\)`
- `INLINE ILuint GetLittleUInt \(SIO \*io\)`
- `INLINE ILushort GetLittleUShort \(SIO \*io\)`
- `void iEndianSwapData \(ILImage \* \_Image\)`
- `INLINE void iSwapDouble \(ILdouble \*d\)`
- `INLINE void iSwapFloat \(ILfloat \*f\)`
- `INLINE void iSwapInt \(ILint \*i\)`
- `INLINE void iSwapShort \(ILshort \*s\)`
- `INLINE void iSwapUInt \(ILuint \*i\)`

- `INLINE void iSwapUShort (ILushort *s)`
- `INLINE ILubyte SaveBigDouble (SIO *io, ILdouble d)`
- `INLINE ILubyte SaveBigFloat (SIO *io, ILfloat f)`
- `INLINE ILubyte SaveBigInt (SIO *io, ILint i)`
- `INLINE ILubyte SaveBigShort (SIO *io, ILshort s)`
- `INLINE ILubyte SaveBigUInt (SIO *io, ILuint i)`
- `INLINE ILubyte SaveBigUShort (SIO *io, ILushort s)`
- `INLINE ILubyte SaveLittleDouble (SIO *io, ILdouble d)`
- `INLINE ILubyte SaveLittleFloat (SIO *io, ILfloat f)`
- `INLINE ILubyte SaveLittleInt (SIO *io, ILint i)`
- `INLINE ILubyte SaveLittleShort (SIO *io, ILshort s)`
- `INLINE ILubyte SaveLittleUInt (SIO *io, ILuint i)`
- `INLINE ILubyte SaveLittleUShort (SIO *io, ILushort s)`

### 7.105.1 Macro Definition Documentation

7.105.1.1 `#define BigDouble( d ) iSwapDouble(d)`

7.105.1.2 `#define BigFloat( f ) iSwapFloat(f)`

7.105.1.3 `#define BigInt( i ) iSwapInt(i)`

7.105.1.4 `#define BigShort( s ) iSwapShort(s)`

7.105.1.5 `#define BigUInt( i ) iSwapUInt(i)`

7.105.1.6 `#define BigUShort( s ) iSwapUShort(s)`

7.105.1.7 `#define Double( d )`

7.105.1.8 `#define dswap( x, y ) t=b[x]; b[x]=b[y]; b[y]=t;`

7.105.1.9 `#define Float( f )`

7.105.1.10 `#define Int( i )`

7.105.1.11 `#define Short( s )`

7.105.1.12 `#define UInt( i )`

7.105.1.13 `#define UShort( s )`

### 7.105.2 Function Documentation

7.105.2.1 `INLINE ILdouble GetBigDouble ( SIO * io )`

7.105.2.2 `INLINE ILfloat GetBigFloat ( SIO * io )`

7.105.2.3 `INLINE ILint GetBigInt ( SIO * io )`

7.105.2.4 `INLINE ILshort GetBigShort ( SIO * io )`

7.105.2.5 `INLINE ILuint GetBigUInt ( SIO * io )`

7.105.2.6 `INLINE ILushort GetBigUShort ( SIO * io )`

- 7.105.2.7 **INLINE** ILdouble GetLittleDouble ( SIO \* *io* )
- 7.105.2.8 **INLINE** ILfloat GetLittleFloat ( SIO \* *io* )
- 7.105.2.9 **INLINE** ILint GetLittleInt ( SIO \* *io* )
- 7.105.2.10 **INLINE** ILshort GetLittleShort ( SIO \* *io* )
- 7.105.2.11 **INLINE** ILuint GetLittleUInt ( SIO \* *io* )
- 7.105.2.12 **INLINE** ILushort GetLittleUShort ( SIO \* *io* )
- 7.105.2.13 **void** iEndianSwapData ( ILimage \* *\_Image* )
- 7.105.2.14 **INLINE void** iSwapDouble ( ILdouble \* *d* )
- 7.105.2.15 **INLINE void** iSwapFloat ( ILfloat \* *f* )
- 7.105.2.16 **INLINE void** iSwapInt ( ILint \* *i* )
- 7.105.2.17 **INLINE void** iSwapShort ( ILshort \* *s* )
- 7.105.2.18 **INLINE void** iSwapUInt ( ILuint \* *i* )
- 7.105.2.19 **INLINE void** iSwapUShort ( ILushort \* *s* )
- 7.105.2.20 **INLINE** ILubyte SaveBigDouble ( SIO \* *io*, ILdouble *d* )
- 7.105.2.21 **INLINE** ILubyte SaveBigFloat ( SIO \* *io*, ILfloat *f* )
- 7.105.2.22 **INLINE** ILubyte SaveBigInt ( SIO \* *io*, ILint *i* )
- 7.105.2.23 **INLINE** ILubyte SaveBigShort ( SIO \* *io*, ILshort *s* )
- 7.105.2.24 **INLINE** ILubyte SaveBigUInt ( SIO \* *io*, ILuint *i* )
- 7.105.2.25 **INLINE** ILubyte SaveBigUShort ( SIO \* *io*, ILushort *s* )
- 7.105.2.26 **INLINE** ILubyte SaveLittleDouble ( SIO \* *io*, ILdouble *d* )
- 7.105.2.27 **INLINE** ILubyte SaveLittleFloat ( SIO \* *io*, ILfloat *f* )
- 7.105.2.28 **INLINE** ILubyte SaveLittleInt ( SIO \* *io*, ILint *i* )
- 7.105.2.29 **INLINE** ILubyte SaveLittleShort ( SIO \* *io*, ILshort *s* )
- 7.105.2.30 **INLINE** ILubyte SaveLittleUInt ( SIO \* *io*, ILuint *i* )
- 7.105.2.31 **INLINE** ILubyte SaveLittleUShort ( SIO \* *io*, ILushort *s* )

## 7.106 src/IL/il\_error.c File Reference

```
#include "il_internal.h"
```

## Macros

- `#define IL_ERROR_STACK_SIZE 32`

## Functions

- `ILenum iGetError (void)`  
*Gets the last error on the error stack.*
- `ILAPI void ILAPIENTRY iSetError (ILenum Error)`

## Variables

- `ILenum ilErrorNum [IL_ERROR_STACK_SIZE]`
- `ILint ilErrorPlace = (-1)`

### 7.106.1 Macro Definition Documentation

7.106.1.1 `#define IL_ERROR_STACK_SIZE 32`

### 7.106.2 Function Documentation

7.106.2.1 `ILenum iGetError ( void )`

Gets the last error on the error stack.

7.106.2.2 `ILAPI void ILAPIENTRY iSetError ( ILenum Error )`

### 7.106.3 Variable Documentation

7.106.3.1 `ILenum ilErrorNum[IL_ERROR_STACK_SIZE]`

7.106.3.2 `ILint ilErrorPlace = (-1)`

## 7.107 src/IL/il\_files.c File Reference

```
#include "il_internal.h"
#include <stdarg.h>
```

## Macros

- `#define __FILES_C`

## Functions

- `void ILAPIENTRY iDefaultClose (ILHANDLE Handle)`
- `void ILAPIENTRY iDefaultCloseW (ILHANDLE Handle)`
- `ILboolean ILAPIENTRY iDefaultEof (ILHANDLE Handle)`
- `ILint ILAPIENTRY iDefaultGetc (ILHANDLE Handle)`
- `ILHANDLE ILAPIENTRY iDefaultOpenR (ILconst_string FileName)`
- `ILHANDLE ILAPIENTRY iDefaultOpenW (ILconst_string FileName)`

- `ILint ILAPIENTRY iDefaultPutc (ILubyte Char, ILHANDLE Handle)`
- `ILuint ILAPIENTRY iDefaultRead (ILHANDLE Handle, void *Buffer, ILuint Size, ILuint Number)`
- `ILint ILAPIENTRY iDefaultSeek (ILHANDLE Handle, ILint Offset, ILuint Mode)`
- `ILuint ILAPIENTRY iDefaultTell (ILHANDLE Handle)`
- `ILint ILAPIENTRY iDefaultWrite (const void *Buffer, ILuint Size, ILuint Number, ILHANDLE Handle)`
- `ILboolean ILAPIENTRY iEofLump (ILHANDLE h)`
- `ILint ILAPIENTRY iGetcLump (ILHANDLE h)`
- `ILuint64 ILAPIENTRY iGetLumpPos ()`
- `void ILAPIENTRY iResetRead ()`
- `void ILAPIENTRY iResetWrite ()`
- `ILboolean ILAPIENTRY iSetRead (fOpenProc aOpen, fCloseProc aClose, fEofProc aEof, fGetcProc aGetc, fReadProc aRead, fSeekProc aSeek, fTellProc aTell)`

*Allows you to override the default file-reading functions.*

- `ILboolean ILAPIENTRY iSetWrite (fOpenProc Open, fCloseProc Close, fPutcProc Putc, fSeekProc Seek, fTellProc Tell, fWriteProc Write)`

*Allows you to override the default file-writing functions.*

- `ILint ILAPIENTRY iPutcLump (ILubyte Char, ILHANDLE h)`
- `ILuint ILAPIENTRY iReadLump (ILHANDLE h, void *Buffer, const ILuint Size, const ILuint Number)`
- `void ILAPIENTRY iResetRead (ILimage *image)`
- `void ILAPIENTRY iResetWrite (ILimage *image)`
- `ILint ILAPIENTRY iSeekLump (ILHANDLE h, ILint Offset, ILuint Mode)`
- `void iSetInputFile (ILimage *image, ILHANDLE File)`
- `void iSetInputLump (ILimage *image, const void *Lump, ILuint Size)`
- `void iSetOutputFake (ILimage *image)`
- `void iSetOutputFile (ILimage *image, ILHANDLE File)`
- `void iSetOutputLump (ILimage *image, void *Lump, ILuint Size)`
- `void ILAPIENTRY iSetRead (ILimage *Image, fOpenProc aOpen, fCloseProc aClose, fEofProc aEof, fGetcProc aGetc, fReadProc aRead, fSeekProc aSeek, fTellProc aTell)`
- `void ILAPIENTRY iSetWrite (ILimage *Image, fOpenProc Open, fCloseProc Close, fPutcProc Putc, fSeekProc Seek, fTellProc Tell, fWriteProc Write)`
- `ILint ILAPIENTRY iSizePutc (ILubyte Char, ILHANDLE h)`
- `ILint ILAPIENTRY iSizeSeek (ILHANDLE h, ILint Offset, ILuint Mode)`

*Fake seek function.*

- `ILuint ILAPIENTRY iSizeTell (ILHANDLE h)`
- `ILint ILAPIENTRY iSizeWrite (const void *Buffer, ILuint Size, ILuint Number, ILHANDLE h)`
- `ILuint ILAPIENTRY iTellLump (ILHANDLE h)`
- `ILint ILAPIENTRY iWriteLump (const void *Buffer, ILuint Size, ILuint Number, ILHANDLE h)`

## 7.107.1 Macro Definition Documentation

### 7.107.1.1 #define \_\_FILES\_C

## 7.107.2 Function Documentation

### 7.107.2.1 void ILAPIENTRY iDefaultClose ( ILHANDLE Handle )

### 7.107.2.2 void ILAPIENTRY iDefaultCloseW ( ILHANDLE Handle )

### 7.107.2.3 ILboolean ILAPIENTRY iDefaultEof ( ILHANDLE Handle )

### 7.107.2.4 ILint ILAPIENTRY iDefaultGetc ( ILHANDLE Handle )

### 7.107.2.5 ILHANDLE ILAPIENTRY iDefaultOpenR ( ILconst\_string FileName )

- 7.107.2.6 **ILHANDLE ILAPIENTRY** iDefaultOpenW ( **ILconst\_string** *FileName* )
- 7.107.2.7 **ILint ILAPIENTRY** iDefaultPutc ( **ILubyte** *Char*, **ILHANDLE** *Handle* )
- 7.107.2.8 **ILuint ILAPIENTRY** iDefaultRead ( **ILHANDLE** *Handle*, **void \*** *Buffer*, **ILuint** *Size*, **ILuint** *Number* )
- 7.107.2.9 **ILint ILAPIENTRY** iDefaultSeek ( **ILHANDLE** *Handle*, **ILint** *Offset*, **ILuint** *Mode* )
- 7.107.2.10 **ILuint ILAPIENTRY** iDefaultTell ( **ILHANDLE** *Handle* )
- 7.107.2.11 **ILint ILAPIENTRY** iDefaultWrite ( **const void \*** *Buffer*, **ILuint** *Size*, **ILuint** *Number*, **ILHANDLE** *Handle* )
- 7.107.2.12 **ILboolean ILAPIENTRY** iEofLump ( **ILHANDLE** *h* )
- 7.107.2.13 **ILint ILAPIENTRY** iGetcLump ( **ILHANDLE** *h* )
- 7.107.2.14 **ILuint64 ILAPIENTRY** iGetLumpPos ( **void** )
- 7.107.2.15 **void ILAPIENTRY** iResetRead ( **void** )
- 7.107.2.16 **void ILAPIENTRY** iResetWrite ( **void** )
- 7.107.2.17 **ILboolean ILAPIENTRY** iSetRead ( **fOpenProc** *aOpen*, **fCloseProc** *aClose*, **fEofProc** *aEof*, **fGetcProc** *aGetc*, **fReadProc** *aRead*, **fSeekProc** *aSeek*, **fTellProc** *aTell* )

Allows you to override the default file-reading functions.

- 7.107.2.18 **ILboolean ILAPIENTRY** iSetWrite ( **fOpenProc** *Open*, **fCloseProc** *Close*, **fPutcProc** *Putc*, **fSeekProc** *Seek*, **fTellProc** *Tell*, **fWriteProc** *Write* )

Allows you to override the default file-writing functions.

- 7.107.2.19 **ILint ILAPIENTRY** iPutcLump ( **ILubyte** *Char*, **ILHANDLE** *h* )
- 7.107.2.20 **ILuint ILAPIENTRY** iReadLump ( **ILHANDLE** *h*, **void \*** *Buffer*, **const ILuint** *Size*, **const ILuint** *Number* )
- 7.107.2.21 **void ILAPIENTRY** iResetRead ( **ILimage \*** *image* )
- 7.107.2.22 **void ILAPIENTRY** iResetWrite ( **ILimage \*** *image* )
- 7.107.2.23 **ILint ILAPIENTRY** iSeekLump ( **ILHANDLE** *h*, **ILint** *Offset*, **ILuint** *Mode* )
- 7.107.2.24 **void iSetInputFile** ( **ILimage \*** *image*, **ILHANDLE** *File* )
- 7.107.2.25 **void iSetInputLump** ( **ILimage \*** *image*, **const void \*** *Lump*, **ILuint** *Size* )
- 7.107.2.26 **void iSetOutputFake** ( **ILimage \*** *image* )
- 7.107.2.27 **void iSetOutputFile** ( **ILimage \*** *image*, **ILHANDLE** *File* )
- 7.107.2.28 **void iSetOutputLump** ( **ILimage \*** *image*, **void \*** *Lump*, **ILuint** *Size* )
- 7.107.2.29 **void ILAPIENTRY** iSetRead ( **ILimage \*** *Image*, **fOpenProc** *aOpen*, **fCloseProc** *aClose*, **fEofProc** *aEof*, **fGetcProc** *aGetc*, **fReadProc** *aRead*, **fSeekProc** *aSeek*, **fTellProc** *aTell* )



7.107.2.30 void ILAPIENTRY iSetWrite ( ILImage \* *Image*, fOpenProc *Open*, fCloseProc *Close*, fPutcProc *Putc*, fSeekProc *Seek*, fTellProc *Tell*, fWriteProc *Write* )

7.107.2.31 ILint ILAPIENTRY iSizePutc ( ILubyte *Char*, ILHANDLE *h* )

7.107.2.32 ILint ILAPIENTRY iSizeSeek ( ILHANDLE *h*, ILint *Offset*, ILuint *Mode* )

Fake seek function.

7.107.2.33 ILuint ILAPIENTRY iSizeTell ( ILHANDLE *h* )

7.107.2.34 ILint ILAPIENTRY iSizeWrite ( const void \* *Buffer*, ILuint *Size*, ILuint *Number*, ILHANDLE *h* )

7.107.2.35 ILuint ILAPIENTRY iTellLump ( ILHANDLE *h* )

7.107.2.36 ILint ILAPIENTRY iWriteLump ( const void \* *Buffer*, ILuint *Size*, ILuint *Number*, ILHANDLE *h* )

## 7.108 src/IL/il\_files.h File Reference

```
#include <IL/il.h>
```

### Macros

- #define `__FILES_EXTERN` extern

### Functions

- `__FILES_EXTERN` void ILAPIENTRY iDefaultClose (ILHANDLE *Handle*)
- `__FILES_EXTERN` ILint ILAPIENTRY iDefaultGetc (ILHANDLE *Handle*)
- `__FILES_EXTERN` ILHANDLE ILAPIENTRY iDefaultOpenR (ILconst\_string *FileName*)
- `__FILES_EXTERN` ILHANDLE ILAPIENTRY iDefaultOpenW (ILconst\_string *FileName*)
- `__FILES_EXTERN` ILint ILAPIENTRY iDefaultPutc (ILubyte *Char*, ILHANDLE *Handle*)
- `__FILES_EXTERN` ILuint ILAPIENTRY iDefaultRead (ILHANDLE *Handle*, void \**Buffer*, ILuint *Size*, ILuint *Number*)
- `__FILES_EXTERN` ILint ILAPIENTRY iDefaultSeek (ILHANDLE *Handle*, ILint *Offset*, ILuint *Mode*)
- `__FILES_EXTERN` ILuint ILAPIENTRY iDefaultTell (ILHANDLE *Handle*)
- `__FILES_EXTERN` ILint ILAPIENTRY iDefaultWrite (const void \**Buffer*, ILuint *Size*, ILuint *Number*, ILHANDLE *Handle*)
- `__FILES_EXTERN` void iSetInputFile (ILImage \*, ILHANDLE *File*)
- `__FILES_EXTERN` void iSetInputLump (ILImage \*, const void \**Lump*, ILuint *Size*)
- `__FILES_EXTERN` void iSetOutputFake (ILImage \*)
- `__FILES_EXTERN` void iSetOutputFile (ILImage \*, ILHANDLE *File*)
- `__FILES_EXTERN` void iSetOutputLump (ILImage \*, void \**Lump*, ILuint *Size*)

### 7.108.1 Macro Definition Documentation

7.108.1.1 #define `__FILES_EXTERN` extern

### 7.108.2 Function Documentation

7.108.2.1 `__FILES_EXTERN` void ILAPIENTRY iDefaultClose ( ILHANDLE *Handle* )

- 7.108.2.2 `__FILES_EXTERN ILint ILAPIENTRY iDefaultGetc ( ILHANDLE Handle )`
- 7.108.2.3 `__FILES_EXTERN ILHANDLE ILAPIENTRY iDefaultOpenR ( ILconst_string FileName )`
- 7.108.2.4 `__FILES_EXTERN ILHANDLE ILAPIENTRY iDefaultOpenW ( ILconst_string FileName )`
- 7.108.2.5 `__FILES_EXTERN ILint ILAPIENTRY iDefaultPutc ( ILubyte Char, ILHANDLE Handle )`
- 7.108.2.6 `__FILES_EXTERN ILuint ILAPIENTRY iDefaultRead ( ILHANDLE Handle, void * Buffer, ILuint Size, ILuint Number )`
- 7.108.2.7 `__FILES_EXTERN ILint ILAPIENTRY iDefaultSeek ( ILHANDLE Handle, ILint Offset, ILuint Mode )`
- 7.108.2.8 `__FILES_EXTERN ILuint ILAPIENTRY iDefaultTell ( ILHANDLE Handle )`
- 7.108.2.9 `__FILES_EXTERN ILint ILAPIENTRY iDefaultWrite ( const void * Buffer, ILuint Size, ILuint Number, ILHANDLE Handle )`
- 7.108.2.10 `__FILES_EXTERN void iSetInputFile ( ILimage *, ILHANDLE File )`
- 7.108.2.11 `__FILES_EXTERN void iSetInputLump ( ILimage *, const void * Lump, ILuint Size )`
- 7.108.2.12 `__FILES_EXTERN void iSetOutputFake ( ILimage * )`
- 7.108.2.13 `__FILES_EXTERN void iSetOutputFile ( ILimage *, ILHANDLE File )`
- 7.108.2.14 `__FILES_EXTERN void iSetOutputLump ( ILimage *, void * Lump, ILuint Size )`

## 7.109 `src/IL/il_formats.c` File Reference

```
#include "il_formats.h"
```

### Data Structures

- struct [ILformatEntry](#)

### Macros

- #define [ADD\\_FORMAT](#)(name)

### Typedefs

- typedef struct [ILformatEntry](#) [ILformatEntry](#)

### Functions

- void [iDeinitFormats](#) ()
- const [ILformat](#) \* [iGetFormat](#) (ILenum id)
- [ILenum](#) [iIdentifyFormat](#) (SIO \*io)
- [ILenum](#) [iIdentifyFormatExt](#) (ILconst\_string Ext)
- void [ilnitFormats](#) ()

## Variables

- `ILchar * _ilLoadExt` = NULL
- `ILchar * _ilSaveExt` = NULL

## 7.109.1 Macro Definition Documentation

### 7.109.1.1 #define ADD\_FORMAT( *name* )

#### Value:

```
extern ILformat iFormat ## name; \
iAddFormat(IL_ ## name, #name, &iFormat ## name);
```

## 7.109.2 Typedef Documentation

### 7.109.2.1 typedef struct ILformatEntry ILformatEntry

## 7.109.3 Function Documentation

### 7.109.3.1 void iDeinitFormats ( )

### 7.109.3.2 const ILformat\* iGetFormat ( ILenum *id* )

### 7.109.3.3 ILenum iIdentifyFormat ( SIO \* *io* )

### 7.109.3.4 ILenum iIdentifyFormatExt ( ILconst\_string *Ext* )

### 7.109.3.5 void iInitFormats ( )

## 7.109.4 Variable Documentation

### 7.109.4.1 ILchar\* \_ilLoadExt = NULL

### 7.109.4.2 ILchar\* \_ilSaveExt = NULL

## 7.110 src/IL/il\_formats.h File Reference

```
#include "il_internal.h"
```

## Data Structures

- struct `ILformat`

## Typedefs

- typedef `ILboolean`(\* `ILformatLoadFunc`)(`ILimage` \*)
- typedef `ILboolean`(\* `ILformatSaveFunc`)(`ILimage` \*)
- typedef `ILboolean`(\* `ILformatValidateFunc`)(`SIO` \*)

## Functions

- [void iDeinitFormats \(\)](#)
- [const ILformat \\* iGetFormat \(ILenum id\)](#)
- [ILenum iIdentifyFormat \(SIO \\*io\)](#)
- [ILenum iIdentifyFormatExt \(ILconst\\_string ext\)](#)
- [void iInitFormats \(\)](#)

### 7.110.1 Typedef Documentation

7.110.1.1 [typedef ILboolean\(\\* ILformatLoadFunc\)\(ILImage \\*\)](#)

7.110.1.2 [typedef ILboolean\(\\* ILformatSaveFunc\)\(ILImage \\*\)](#)

7.110.1.3 [typedef ILboolean\(\\* ILformatValidateFunc\)\(SIO \\*\)](#)

### 7.110.2 Function Documentation

7.110.2.1 [void iDeinitFormats \( \)](#)

7.110.2.2 [const ILformat\\* iGetFormat \( ILenum id \)](#)

7.110.2.3 [ILenum iIdentifyFormat \( SIO \\* io \)](#)

7.110.2.4 [ILenum iIdentifyFormatExt \( ILconst\\_string ext \)](#)

7.110.2.5 [void iInitFormats \( \)](#)

## 7.111 src/IL/il\_internal.c File Reference

```
#include "il_internal.h"
#include <stdlib.h>
#include <ctype.h>
```

## Functions

- [ILboolean iFileExists \(ILconst\\_string FileName\)](#)
- [ILAPI ILboolean ILAPIENTRY iDxtcDataToSurface \(\)](#)
- [ILAPI ILboolean ILAPIENTRY iSurfaceToDxtcData \(ILenum Format\)](#)
- [int iSqrt \(int x\)](#)
- [char \\*ILAPIENTRY SIOgets \(SIO \\*io, char \\*buffer, ILuint MaxLen\)](#)
- [char \\*ILAPIENTRY SIOgetw \(SIO \\*io, char \\*buffer, ILuint MaxLen\)](#)

## Variables

- [FILE \\* iTraceOut = NULL](#)

### 7.111.1 Function Documentation

7.111.1.1 [ILboolean iFileExists \( ILconst\\_string FileName \)](#)

7.111.1.2 ILAPI ILboolean ILAPIENTRY ilDxtcDataToSurface ( void )

7.111.1.3 ILAPI ILboolean ILAPIENTRY ilSurfaceToDxtcData ( ILenum *Format* )

7.111.1.4 int iSqrt ( int *x* )

7.111.1.5 char\* ILAPIENTRY SIOgets ( SIO \* *io*, char \* *buffer*, ILuint *MaxLen* )

7.111.1.6 char\* ILAPIENTRY SIOgetw ( SIO \* *io*, char \* *buffer*, ILuint *MaxLen* )

## 7.111.2 Variable Documentation

7.111.2.1 FILE\* iTraceOut = NULL

## 7.112 src/IL/il\_internal.h File Reference

```
#include <IL/config.h>
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
#include <IL/il.h>
#include <IL/devil_internal_exports.h>
#include "il_files.h"
#include "il_endian.h"
#include "il_string.h"
#include "il_pal.h"
#include "il_formats.h"
```

## Macros

- #define [\\_IL\\_BUILD\\_LIBRARY](#)
- #define [BIT\(n\)](#) (1<<n)
- #define [BIT\\_0](#) 0x00000001
- #define [BIT\\_1](#) 0x00000002
- #define [BIT\\_10](#) 0x00000400
- #define [BIT\\_11](#) 0x00000800
- #define [BIT\\_12](#) 0x00001000
- #define [BIT\\_13](#) 0x00002000
- #define [BIT\\_14](#) 0x00004000
- #define [BIT\\_15](#) 0x00008000
- #define [BIT\\_16](#) 0x00010000
- #define [BIT\\_17](#) 0x00020000
- #define [BIT\\_18](#) 0x00040000
- #define [BIT\\_19](#) 0x00080000
- #define [BIT\\_2](#) 0x00000004
- #define [BIT\\_20](#) 0x00100000
- #define [BIT\\_21](#) 0x00200000
- #define [BIT\\_22](#) 0x00400000
- #define [BIT\\_23](#) 0x00800000
- #define [BIT\\_24](#) 0x01000000
- #define [BIT\\_25](#) 0x02000000
- #define [BIT\\_26](#) 0x04000000
- #define [BIT\\_27](#) 0x08000000

- #define `BIT_28` 0x10000000
- #define `BIT_29` 0x20000000
- #define `BIT_3` 0x00000008
- #define `BIT_30` 0x40000000
- #define `BIT_31` 0x80000000
- #define `BIT_4` 0x00000010
- #define `BIT_5` 0x00000020
- #define `BIT_6` 0x00000040
- #define `BIT_7` 0x00000080
- #define `BIT_8` 0x00000100
- #define `BIT_9` 0x00000200
- #define `iAssert(x)`
- #define `IL_BMPCOMP` 0x04
- #define `IL_PCXCOMP` 0x02
- #define `IL_SGICOMP` 0x03
- #define `IL_TGACOMP` 0x01
- #define `imemclear(x, y)` `memset(x,0,y);`
- #define `iTrace(...)`
- #define `iTraceV(fmt, args)`
- #define `NUL` `'\0'`

## Functions

- `ILboolean iAddAlpha (ILImage *Image)`
- `ILboolean iAddAlphaKey (ILImage *Image)`
- `ILboolean iCopyImage (ILImage *DestImage, ILImage *SrcImage)`
- `ILuint iDuplicateImage (ILuint SrcName)`
- `ILAPI ILboolean ILAPIENTRY iDxtcDataToSurface (ILImage *image)`
- `ILboolean iFastConvert (ILImage *Image, ILenum DestFormat)`
- `ILboolean iFileExists (ILconst_string FileName)`
- `ILboolean iFixImages (ILImage *Image)`
- `ILenum iGetError (void)`  
*Gets the last error on the error stack.*
- `ILenum iGetHint (ILenum Target)`
- `ILconst_string iGetILString (ILenum StringName)`  
*Returns a constant string detailing aspects about this library.*
- `ILint iGetInt (ILenum Mode)`
- `char * iGetString (ILenum StringName)`
- `void iHint (ILenum Target, ILenum Mode)`  
*Specifies implementation-dependent performance hints.*
- `void ilDefaultStates (void)`  
*Set all states to their defaults.*
- `ILboolean ILAPIENTRY ilLoadFuncs2 (ILImage *image, ILenum type)`
- `ILAPI ILubyte *ILAPIENTRY ilNVidiaCompressDXT (ILubyte *Data, ILuint Width, ILuint Height, ILuint Depth, ILenum DxtFormat, ILuint *DxtSize)`
- `ILuint ilNVidiaCompressDXTFile (ILubyte *Data, ILuint Width, ILuint Height, ILuint Depth, ILenum DxtType)`
- `void ilRemoveRegistered (void)`
- `ILuint ilRleCompress (ILubyte *Data, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, ILubyte *Dest, ILenum CompressMode, ILuint *ScanTable)`
- `ILboolean ilRleCompressLine (ILubyte *ScanLine, ILuint Width, ILubyte Bpp, ILubyte *Dest, ILuint *DestWidth, ILenum CompressMode)`
- `ILAPI ILubyte *ILAPIENTRY ilSquishCompressDXT (ILubyte *Data, ILuint Width, ILuint Height, ILuint Depth, ILenum DxtFormat, ILuint *DxtSize)`
- `ILboolean iRemoveAlpha (ILImage *Image)`
- `int iSqrt (int x)`
- `ILAPI ILboolean ILAPIENTRY iSurfaceToDxtcData (ILImage *image, ILenum Format)`
- `ILboolean iSwapColours (ILImage *Image)`

## Variables

- FILE \* [iTraceOut](#)

### 7.112.1 Macro Definition Documentation

7.112.1.1 `#define _IL_BUILD_LIBRARY`

7.112.1.2 `#define BIT( n ) (1<<n)`

7.112.1.3 `#define BIT_0 0x00000001`

7.112.1.4 `#define BIT_1 0x00000002`

7.112.1.5 `#define BIT_10 0x00000400`

7.112.1.6 `#define BIT_11 0x00000800`

7.112.1.7 `#define BIT_12 0x00001000`

7.112.1.8 `#define BIT_13 0x00002000`

7.112.1.9 `#define BIT_14 0x00004000`

7.112.1.10 `#define BIT_15 0x00008000`

7.112.1.11 `#define BIT_16 0x00010000`

7.112.1.12 `#define BIT_17 0x00020000`

7.112.1.13 `#define BIT_18 0x00040000`

7.112.1.14 `#define BIT_19 0x00080000`

7.112.1.15 `#define BIT_2 0x00000004`

7.112.1.16 `#define BIT_20 0x00100000`

7.112.1.17 `#define BIT_21 0x00200000`

7.112.1.18 `#define BIT_22 0x00400000`

7.112.1.19 `#define BIT_23 0x00800000`

7.112.1.20 `#define BIT_24 0x01000000`

7.112.1.21 `#define BIT_25 0x02000000`

7.112.1.22 `#define BIT_26 0x04000000`

7.112.1.23 `#define BIT_27 0x08000000`

7.112.1.24 `#define BIT_28 0x10000000`

7.112.1.25 `#define BIT_29 0x20000000`

7.112.1.26 `#define BIT_3 0x00000008`

7.112.1.27 `#define BIT_30 0x40000000`

7.112.1.28 `#define BIT_31 0x80000000`

7.112.1.29 `#define BIT_4 0x00000010`

7.112.1.30 `#define BIT_5 0x00000020`

7.112.1.31 `#define BIT_6 0x00000040`

7.112.1.32 `#define BIT_7 0x00000080`

7.112.1.33 `#define BIT_8 0x00000100`

7.112.1.34 `#define BIT_9 0x00000200`

7.112.1.35 `#define iAssert( x )`

7.112.1.36 `#define IL_BMPCOMP 0x04`

7.112.1.37 `#define IL_PCXCOMP 0x02`

7.112.1.38 `#define IL_SGICOMP 0x03`

7.112.1.39 `#define IL_TGACOMP 0x01`

7.112.1.40 `#define imemclear( x, y ) memset(x,0,y);`

7.112.1.41 `#define iTrace( ... )`

**Value:**

```
if (iTraceOut) {\
    fprintf(iTraceOut, "%s:%d: ", __FILE__, __LINE__); \
    fprintf(iTraceOut, __VA_ARGS__); \
    fputc('\n', iTraceOut); \
    fflush(iTraceOut); \
}
```

7.112.1.42 `#define iTraceV( fmt, args )`

**Value:**

```
if (iTraceOut) {\
    fprintf(iTraceOut, "%s:%d: ", __FILE__, __LINE__); \
    vfprintf(iTraceOut, fmt, args); \
    fputc('\n', iTraceOut); \
    fflush(iTraceOut); \
}
```

7.112.1.43 `#define NUL '\0'`

**7.112.2 Function Documentation**

7.112.2.1 `ILboolean iAddAlpha ( ILimage * Image )`



7.112.2.2 **ILboolean** iAddAlphaKey ( **ILImage** \* *Image* )

7.112.2.3 **ILboolean** iCopyImage ( **ILImage** \* *DestImage*, **ILImage** \* *SrcImage* )

7.112.2.4 **ILuint** iDuplicatImage ( **ILuint** *SrcName* )

7.112.2.5 **ILAPI ILboolean ILAPIENTRY** iDxtcDataToSurface ( **ILImage** \* *image* )

7.112.2.6 **ILboolean** iFastConvert ( **ILImage** \* *Image*, **ILenum** *DestFormat* )

7.112.2.7 **ILboolean** iFileExists ( **ILconst\_string** *FileName* )

7.112.2.8 **ILboolean** iFixImages ( **ILImage** \* *Image* )

7.112.2.9 **ILenum** iGetError ( **void** )

Gets the last error on the error stack.

7.112.2.10 **ILenum** iGetHint ( **ILenum** *Target* )

7.112.2.11 **ILconst\_string** iGetILString ( **ILenum** *StringName* )

Returns a constant string detailing aspects about this library.

7.112.2.12 **ILint** iGetInt ( **ILenum** *Mode* )

7.112.2.13 **char\*** iGetString ( **ILenum** *StringName* )

7.112.2.14 **void** iHint ( **ILenum** *Target*, **ILenum** *Mode* )

Specifies implementation-dependent performance hints.

7.112.2.15 **void** iDefaultStates ( **void** )

Set all states to their defaults.

7.112.2.16 **ILboolean ILAPIENTRY** iLoadFuncs2 ( **ILImage** \* *image*, **ILenum** *type* )

7.112.2.17 **ILAPI ILubyte\*** **ILAPIENTRY** iNvidiaCompressDXT ( **ILubyte** \* *Data*, **ILuint** *Width*, **ILuint** *Height*, **ILuint** *Depth*, **ILenum** *DxtFormat*, **ILuint** \* *DxtSize* )

7.112.2.18 **ILuint** iNvidiaCompressDXTFile ( **ILubyte** \* *Data*, **ILuint** *Width*, **ILuint** *Height*, **ILuint** *Depth*, **ILenum** *DxtType* )

7.112.2.19 **void** iRemoveRegistered ( **void** )

7.112.2.20 **ILuint** iRleCompress ( **ILubyte** \* *Data*, **ILuint** *Width*, **ILuint** *Height*, **ILuint** *Depth*, **ILubyte** *Bpp*, **ILubyte** \* *Dest*, **ILenum** *CompressMode*, **ILuint** \* *ScanTable* )

7.112.2.21 **ILboolean** iRleCompressLine ( **ILubyte** \* *ScanLine*, **ILuint** *Width*, **ILubyte** *Bpp*, **ILubyte** \* *Dest*, **ILuint** \* *DestWidth*, **ILenum** *CompressMode* )

7.112.2.22 **ILAPI ILubyte\*** **ILAPIENTRY** iSquishCompressDXT ( **ILubyte** \* *Data*, **ILuint** *Width*, **ILuint** *Height*, **ILuint** *Depth*, **ILenum** *DxtFormat*, **ILuint** \* *DxtSize* )

7.112.2.23 **ILboolean** iRemoveAlpha ( **ILImage** \* *Image* )

7.112.2.24 **int** iSqrt ( **int** *x* )

7.112.2.25 **ILAPI ILboolean ILAPIENTRY** iSurfaceToDxtcData ( **ILImage** \* *image*, **ILenum** *Format* )

7.112.2.26 **ILboolean** iSwapColours ( **ILImage** \* *Image* )

### 7.112.3 Variable Documentation

7.112.3.1 **FILE\*** iTraceOut

## 7.113 src/IL/il\_io.c File Reference

```
#include "il_internal.h"
#include "il_register.h"
#include "il_pal.h"
#include <string.h>
#include "il_formats.h"
```

### Functions

- **ILboolean ILAPIENTRY** ilsValid (**ILenum** *Type*, **SIO** \**io*)
- **ILenum ILAPIENTRY** iIDetermineType (**ILconst\_string** *FileName*)
- **ILenum ILAPIENTRY** iIDetermineTypeF (**ILHANDLE** *File*)
- **ILenum ILAPIENTRY** iIDetermineTypeFuncs ()
- **ILenum ILAPIENTRY** iIDetermineTypeL (const **void** \**Lump*, **ILuint** *Size*)
- **ILboolean ILAPIENTRY** iIsValid (**ILenum** *Type*, **ILconst\_string** *FileName*)
- **ILboolean ILAPIENTRY** iIsValidF (**ILenum** *Type*, **ILHANDLE** *File*)
- **ILboolean ILAPIENTRY** iIsValidL (**ILenum** *Type*, **void** \**Lump*, **ILuint** *Size*)
- **ILboolean ILAPIENTRY** iILoad (**ILenum** *Type*, **ILconst\_string** *FileName*)  
*Attempts to load an image from a file. The file format is specified by the user.*
- **ILboolean ILAPIENTRY** iILoadF (**ILenum** *Type*, **ILHANDLE** *File*)  
*Attempts to load an image from a file stream. The file format is specified by the user.*
- **ILboolean ILAPIENTRY** iILoadFuncs (**ILenum** *type*)  
*Attempts to load an image using the currently set IO functions. The file format is specified by the user.*
- **ILboolean ILAPIENTRY** iILoadFuncs2 (**ILImage** \**image*, **ILenum** *type*)
- **ILboolean ILAPIENTRY** iILoadImage (**ILconst\_string** *FileName*)  
*Attempts to load an image from a file with various different methods before failing - very generic.*
- **ILboolean ILAPIENTRY** iILoadL (**ILenum** *Type*, const **void** \**Lump*, **ILuint** *Size*)  
*Attempts to load an image from a memory buffer. The file format is specified by the user.*
- **ILboolean ILAPIENTRY** iISave (**ILenum** *type*, **ILconst\_string** *FileName*)  
*Attempts to save an image to a file. The file format is specified by the user.*
- **ILuint ILAPIENTRY** iISaveF (**ILenum** *type*, **ILHANDLE** *File*)  
*Attempts to save an image to a file stream. The file format is specified by the user.*
- **ILAPI ILboolean ILAPIENTRY** iISaveFuncs (**ILenum** *type*)
- **ILboolean ILAPIENTRY** iISaveFuncs2 (**ILImage** \**image*, **ILenum** *type*)
- **ILboolean ILAPIENTRY** iISaveImage (**ILconst\_string** *FileName*)  
*Saves the current image based on the extension given in FileName.*
- **ILuint ILAPIENTRY** iISaveL (**ILenum** *Type*, **void** \**Lump*, **ILuint** *Size*)  
*Attempts to save an image to a memory buffer. The file format is specified by the user.*
- **ILenum ILAPIENTRY** iTypeFromExt (**ILconst\_string** *FileName*)

- char \*ILAPIENTRY iMultiByteFromWide (const wchar\_t \*Wide)
- wchar\_t \*ILAPIENTRY iWideFromMultiByte (const char \*Multi)

### 7.113.1 Function Documentation

7.113.1.1 ILboolean ILAPIENTRY ilsValid ( IEnum Type, SIO \* io )

7.113.1.2 IEnum ILAPIENTRY ilDetermineType ( ILconst\_string FileName )

7.113.1.3 IEnum ILAPIENTRY ilDetermineTypeF ( ILHANDLE File )

7.113.1.4 IEnum ILAPIENTRY ilDetermineTypeFuncs ( )

7.113.1.5 IEnum ILAPIENTRY ilDetermineTypeL ( const void \* Lump, ILuint Size )

7.113.1.6 ILboolean ILAPIENTRY illsValid ( IEnum Type, ILconst\_string FileName )

7.113.1.7 ILboolean ILAPIENTRY illsValidF ( IEnum Type, ILHANDLE File )

7.113.1.8 ILboolean ILAPIENTRY illsValidL ( IEnum Type, void \* Lump, ILuint Size )

7.113.1.9 ILboolean ILAPIENTRY ilLoad ( IEnum Type, ILconst\_string FileName )

Attempts to load an image from a file. The file format is specified by the user.

#### Parameters

<i>Type</i>	Format of this file. Acceptable values are IL_BLP, IL_BMP, IL_CUT, IL_DCX, IL_DDS, IL_DICOM, IL_DOOM, IL_DOOM_FLAT, IL_DPX, IL_EXR, IL_FITS, IL_FTX, IL_GIF, IL_HDR, IL_ICO, IL_ICNS, IL_IFF, IL_IWI, IL_JP2, IL_JPG, IL_LIF, IL_MDL, IL_MNG, IL_MP3, IL_PCD, IL_PCX, IL_PIX, IL_PNG, IL_PNM, IL_PSD, IL_PSP, IL_PXR, IL_ROT, IL_SGI, IL_SUN, IL_TEXTURE, IL_TGA, IL_TIF, IL_TPL, IL_UTX, IL_VTF, IL_WAL, IL_WBMP, IL_XPM, IL_RAW, IL_JASC_PAL and IL_TYPE_UNKNOWN. If IL_TYPE_UNKNOWN is specified, ilLoad will try to determine the type of the file and load it.
<i>FileName</i>	Ansi or Unicode string, depending on the compiled version of DevIL, that gives the filename of the file to load.

#### Returns

Boolean value of failure or success. Returns IL\_FALSE if all three loading methods have been tried and failed.

7.113.1.10 ILboolean ILAPIENTRY ilLoadF ( IEnum Type, ILHANDLE File )

Attempts to load an image from a file stream. The file format is specified by the user.

#### Parameters

<i>Type</i>	Format of this file. Acceptable values are IL_BLP, IL_BMP, IL_CUT, IL_DCX, IL_DDS, IL_DICOM, IL_DOOM, IL_DOOM_FLAT, IL_DPX, IL_EXR, IL_FITS, IL_FTX, IL_GIF, IL_HDR, IL_ICO, IL_ICNS, IL_IFF, IL_IWI, IL_JP2, IL_JPG, IL_LIF, IL_MDL, IL_MNG, IL_MP3, IL_PCD, IL_PCX, IL_PIX, IL_PNG, IL_PNM, IL_PSD, IL_PSP, IL_PXR, IL_ROT, IL_SGI, IL_SUN, IL_TEXTURE, IL_TGA, IL_TIF, IL_TPL, IL_UTX, IL_VTF, IL_WAL, IL_WBMP, IL_XPM, IL_RAW, IL_JASC_PAL and IL_TYPE_UNKNOWN. If IL_TYPE_UNKNOWN is specified, ilLoadF will try to determine the type of the file and load it.
-------------	--

<i>File</i>	File stream to load from. The caller is responsible for closing the handle.
-------------	---

**Returns**

Boolean value of failure or success. Returns IL\_FALSE if loading fails.

**7.113.1.11 ILboolean ILAPIENTRY ilLoadFuncs ( ILenum type )**

Attempts to load an image using the currently set IO functions. The file format is specified by the user.

**Parameters**

<i>Type</i>	Format of this file. Acceptable values are IL_BLP, IL_BMP, IL_CUT, IL_DCX, IL_DDS, IL_DICOM, IL_DOOM, IL_DOOM_FLAT, IL_DPX, IL_EXR, IL_FITS, IL_FTX, IL_GIF, IL_HDR, IL_ICO, IL_ICNS, IL_IFF, IL_IWI, IL_JP2, IL_JPG, IL_LIF, IL_MDL, IL_MNG, IL_MP3, IL_PCD, IL_PCX, IL_PIX, IL_PNG, IL_PNM, IL_PSD, IL_PSP, IL_PXR, IL_ROT, IL_SGI, IL_SUN, IL_TEXTURE, IL_TGA, IL_TIF, IL_TPL, IL_UTX, IL_VTF, IL_WAL, IL_WBMP, IL_XPM, IL_RAW, IL_JASC_PAL and IL_TYPE_UNKNOWN. If IL_TYPE_UNKNOWN is specified, ilLoadFuncs fails.
<i>File</i>	File stream to load from.

**Returns**

Boolean value of failure or success. Returns IL\_FALSE if loading fails.

**7.113.1.12 ILboolean ILAPIENTRY ilLoadFuncs2 ( IImage \* image, ILenum type )****7.113.1.13 ILboolean ILAPIENTRY ilLoadImage ( ILconst\_string FileName )**

Attempts to load an image from a file with various different methods before failing - very generic.

The ilLoadImage function allows a general interface to the specific internal file-loading routines. First, it finds the extension and checks to see if any user-registered functions (registered through ilRegisterLoad) match the extension. If nothing matches, it takes the extension and determines which function to call based on it. Lastly, it attempts to identify the image based on various image header verification functions, such as illValidPngF. If all this checking fails, IL\_FALSE is returned with no modification to the current bound image.

**Parameters**

<i>FileName</i>	Ansi or Unicode string, depending on the compiled version of DevIL, that gives the filename of the file to load.
-----------------	--

**Returns**

Boolean value of failure or success. Returns IL\_FALSE if all three loading methods have been tried and failed.

**7.113.1.14 ILboolean ILAPIENTRY ilLoadL ( ILenum Type, const void \* Lump, ILuint Size )**

Attempts to load an image from a memory buffer. The file format is specified by the user.

**Parameters**


---

<i>Type</i>	Format of this file. Acceptable values are IL_BLP, IL_BMP, IL_CUT, IL_DCX, IL_DDS, IL_ - DICOM, IL_DOOM, IL_DOOM_FLAT, IL_DPX, IL_EXR, IL_FITS, IL_FTX, IL_GIF, IL_HDR, IL_ICO, IL_ICNS, IL_IFF, IL_IWI, IL_JP2, IL_JPG, IL_LIF, IL_MDL, IL_MNG, IL_MP3, IL_P- CD, IL_PCX, IL_PIX, IL_PNG, IL_PNM, IL_PSD, IL_PSP, IL_PXR, IL_ROT, IL_SGI, IL_SUN, IL_TEXTURE, IL_TGA, IL_TIF, IL_TPL, IL_UTX, IL_VTF, IL_WAL, IL_WBMP, IL_XPM, IL_R- AW, IL_JASC_PAL and IL_TYPE_UNKNOWN. If IL_TYPE_UNKNOWN is specified, ilLoadL will try to determine the type of the file and load it.
<i>Lump</i>	The buffer where the file data is located
<i>Size</i>	Size of the buffer

**Returns**

Boolean value of failure or success. Returns IL\_FALSE if loading fails.

#### 7.113.1.15 ILboolean ILAPIENTRY ilSave ( IEnum type, ILconst\_string FileName )

Attempts to save an image to a file. The file format is specified by the user.

**Parameters**

<i>Type</i>	Format of this file. Acceptable values are IL_BMP, IL_CHEAD, IL_DDS, IL_EXR, IL_HDR, IL_JP2, IL_JPG, IL_PCX, IL_PNG, IL_PNM, IL_PSD, IL_RAW, IL_SGI, IL_TGA, IL_TIF, IL_ - VTF, IL_WBMP and IL_JASC_PAL.
<i>FileName</i>	Ansi or Unicode string, depending on the compiled version of DevIL, that gives the filename to save to.

**Returns**

Boolean value of failure or success. Returns IL\_FALSE if saving failed.

#### 7.113.1.16 ILuint ILAPIENTRY ilSaveF ( IEnum type, ILHANDLE File )

Attempts to save an image to a file stream. The file format is specified by the user.

**Parameters**

<i>Type</i>	Format of this file. Acceptable values are IL_BMP, IL_CHEAD, IL_DDS, IL_EXR, IL_HDR, IL_JP2, IL_JPG, IL_PCX, IL_PNG, IL_PNM, IL_PSD, IL_RAW, IL_SGI, IL_TGA, IL_TIF, IL_ - VTF, IL_WBMP and IL_JASC_PAL.
<i>File</i>	File stream to save to.

**Returns**

Boolean value of failure or success. Returns IL\_FALSE if saving failed.

#### 7.113.1.17 ILAPI ILboolean ILAPIENTRY ilSaveFuncs ( IEnum type )

#### 7.113.1.18 ILboolean ILAPIENTRY ilSaveFuncs2 ( IImage \* image, IEnum type )

#### 7.113.1.19 ILboolean ILAPIENTRY ilSaveImage ( ILconst\_string FileName )

Saves the current image based on the extension given in FileName.

## Parameters

<i>FileName</i>	Ansi or Unicode string, depending on the compiled version of DevIL, that gives the filename to save to.
-----------------	---

## Returns

Boolean value of failure or success. Returns IL\_FALSE if saving failed.

## 7.113.1.20 ILuint ILAPIENTRY ilSaveL ( IEnum Type, void \* Lump, ILuint Size )

Attempts to save an image to a memory buffer. The file format is specified by the user.

## Parameters

<i>Type</i>	Format of this image file. Acceptable values are IL_BMP, IL_CHEAD, IL_DDS, IL_EXR, IL_HDR, IL_JP2, IL_JPG, IL_PCX, IL_PNG, IL_PNM, IL_PSD, IL_RAW, IL_SGI, IL_TGA, IL_TIF, IL_VTF, IL_WBMP and IL_JASC_PAL.
<i>Lump</i>	Memory buffer to save to
<i>Size</i>	Size of the memory buffer

## Returns

The number of bytes written to the lump, or 0 in case of failure

## 7.113.1.21 IEnum ILAPIENTRY ilTypeFromExt ( ILconst\_string FileName )

## 7.113.1.22 char\* ILAPIENTRY iMultiByteFromWide ( const wchar\_t \* Wide )

## 7.113.1.23 wchar\_t\* ILAPIENTRY iWideFromMultiByte ( const char \* Multi )

## 7.114 src/IL/il\_kail.c File Reference

```
#include "il_internal.h"
#include <string.h>
#include <limits.h>
#include "il_manip.h"
```

## Functions

- ILboolean ILAPIENTRY iBlit (ILimage \*Image, ILuint Source, ILint DestX, ILint DestY, ILint DestZ, ILuint SrcX, ILuint SrcY, ILuint SrcZ, ILuint Width, ILuint Height, ILuint Depth)
- ILboolean iCopyImage (ILimage \*DestImage, ILimage \*SrcImage)
- ILboolean iCopySubImage (ILimage \*Dest, ILimage \*Src)
- ILboolean iCopySubImages (ILimage \*Dest, ILimage \*Src)
- ILuint iDuplicatImage (ILuint SrcName)
- ILboolean ILAPIENTRY iBlit (ILuint Source, ILint DestX, ILint DestY, ILint DestZ, ILuint SrcX, ILuint SrcY, ILuint SrcZ, ILuint Width, ILuint Height, ILuint Depth)
- void ILAPIENTRY iClearColour (ILclampf Red, ILclampf Green, ILclampf Blue, ILclampf Alpha)
- ILboolean ILAPIENTRY iClearImage ()

*Clears the current bound image to the values specified in iClearColour.*

- ILAPI ILboolean ILAPIENTRY iClearImage\_ (ILimage \*Image)
- ILAPI ILimage \*ILAPIENTRY iCopyImage\_ (ILimage \*Src)

- **ILAPI ILboolean ILAPIENTRY iCopyImageAttr** (ILImage \*Dest, ILImage \*Src)
- **ILAPI void ILAPIENTRY iGetClear** (void \*Colours, IEnum Format, IEnum Type)
- **ILubyte \*ILAPIENTRY iGetData** (void)  
*Returns a pointer to the current image's data.*
- **ILubyte \*ILAPIENTRY iGetPalette** (void)  
*Returns a pointer to the current image's palette data.*
- **ILAPI ILboolean ILAPIENTRY iInitImage** (ILImage \*Image, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, IEnum Format, IEnum Type, void \*Data)
- **ILAPI ILImage \*ILAPIENTRY iNewImage** (ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, ILubyte Bpc)
- **ILAPI ILImage \*ILAPIENTRY iNewImageFull** (ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, IEnum Format, IEnum Type, void \*Data)
- **ILboolean ILAPIENTRY iOverlayImage** (ILuint Source, ILint XCoord, ILint YCoord, ILint ZCoord)  
*Overlays the image found in Src on top of the current bound image at the coords specified.*
- **ILAPI ILboolean ILAPIENTRY iResizeImage** (ILImage \*Image, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, ILubyte Bpc)
- **ILboolean ILAPIENTRY iSetData** (void \*Data)  
*Uploads Data of the same size to replace the current image's data.*
- **ILboolean ILAPIENTRY iTexImage** (ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, IEnum Format, IEnum Type, void \*Data)  
*Changes the current bound image to use these new dimensions (current data is destroyed).*
- **ILAPI ILboolean ILAPIENTRY iTexImage\_** (ILImage \*Image, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, IEnum Format, IEnum Type, void \*Data)
- **ILAPI ILboolean ILAPIENTRY iTexSubImage\_** (ILImage \*Image, void \*Data)

### 7.114.1 Function Documentation

- 7.114.1.1 **ILboolean ILAPIENTRY iBlit** ( ILImage \* Image, ILuint Source, ILint DestX, ILint DestY, ILint DestZ, ILuint SrcX, ILuint SrcY, ILuint SrcZ, ILuint Width, ILuint Height, ILuint Depth )
- 7.114.1.2 **ILboolean iCopyImage** ( ILImage \* DestImage, ILImage \* SrcImage )
- 7.114.1.3 **ILboolean iCopySubImage** ( ILImage \* Dest, ILImage \* Src )
- 7.114.1.4 **ILboolean iCopySubImages** ( ILImage \* Dest, ILImage \* Src )
- 7.114.1.5 **ILuint iDuplicateImage** ( ILuint SrcName )
- 7.114.1.6 **ILboolean ILAPIENTRY iBlit** ( ILuint Source, ILint DestX, ILint DestY, ILint DestZ, ILuint SrcX, ILuint SrcY, ILuint SrcZ, ILuint Width, ILuint Height, ILuint Depth )
- 7.114.1.7 **void ILAPIENTRY iClearColour** ( ILclampf Red, ILclampf Green, ILclampf Blue, ILclampf Alpha )
- 7.114.1.8 **ILboolean ILAPIENTRY iClearImage** ( void )  
  
Clears the current bound image to the values specified in iClearColour.
- 7.114.1.9 **ILAPI ILboolean ILAPIENTRY iClearImage\_** ( ILImage \* Image )
- 7.114.1.10 **ILAPI ILImage\* ILAPIENTRY iCopyImage\_** ( ILImage \* Src )
- 7.114.1.11 **ILAPI ILboolean ILAPIENTRY iCopyImageAttr** ( ILImage \* Dest, ILImage \* Src )
- 7.114.1.12 **ILAPI void ILAPIENTRY iGetClear** ( void \* Colours, IEnum Format, IEnum Type )

**7.114.1.13 ILubyte\* ILAPIENTRY ilGetData ( void )**

Returns a pointer to the current image's data.

The pointer to the image data returned by this function is only valid until any operations are done on the image. After any operations, this function should be called again. The pointer can be cast to other types for images that have more than one byte per channel for easier access to data.

**Exceptions**

<b><i>IL_ILLEGAL_OPERATION</i></b>	No currently bound image
------------------------------------	--------------------------

**Returns**

ILubyte pointer to image data.

**7.114.1.14 ILubyte\* ILAPIENTRY ilGetPalette ( void )**

Returns a pointer to the current image's palette data.

The pointer to the image palette data returned by this function is only valid until any operations are done on the image. After any operations, this function should be called again.

**Exceptions**

<b><i>IL_ILLEGAL_OPERATION</i></b>	No currently bound image
------------------------------------	--------------------------

**Returns**

ILubyte pointer to image palette data.

**7.114.1.15 ILAPI ILboolean ILAPIENTRY ilInitImage ( IImage \* Image, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, IEnum Format, IEnum Type, void \* Data )****7.114.1.16 ILAPI IImage\* ILAPIENTRY ilNewImage ( ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, ILubyte Bpc )****7.114.1.17 ILAPI IImage\* ILAPIENTRY ilNewImageFull ( ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, IEnum Format, IEnum Type, void \* Data )****7.114.1.18 ILboolean ILAPIENTRY ilOverlayImage ( ILuint Source, ILint XCoord, ILint YCoord, ILint ZCoord )**

Overlays the image found in Src on top of the current bound image at the coords specified.

TODO: move to [il\\_api.c](#)

**7.114.1.19 ILAPI ILboolean ILAPIENTRY ilResizImage ( IImage \* Image, ILuint Width, ILuint Height, ILuint Depth, ILubyte Bpp, ILubyte Bpc )****7.114.1.20 ILboolean ILAPIENTRY ilSetData ( void \* Data )**

Uploads Data of the same size to replace the current image's data.



## Parameters

<i>Data</i>	New image data to update the currently bound image
-------------	--

## Exceptions

<i>IL_ILLEGAL_OPERATION</i>	No currently bound image
<i>IL_INVALID_PARAM</i>	Data was NULL.

## Returns

Boolean value of failure or success

#### 7.114.1.21 ILboolean ILAPIENTRY ilTexImage ( ILuint *Width*, ILuint *Height*, ILuint *Depth*, ILubyte *Bpp*, ILenum *Format*, ILenum *Type*, void \* *Data* )

Changes the current bound image to use these new dimensions (current data is destroyed).

## Parameters

<i>Width</i>	Specifies the new image width. This cannot be 0.
<i>Height</i>	Specifies the new image height. This cannot be 0.
<i>Depth</i>	Specifies the new image depth. This cannot be 0.
<i>Bpp</i>	Number of channels (ex. 3 for RGB)
<i>Format</i>	Enum of the desired format. Any format values are accepted.
<i>Type</i>	Enum of the desired type. Any type values are accepted.
<i>Data</i>	Specifies data that should be copied to the new image. If this parameter is NULL, no data is copied, and the new image data consists of undefined values.

## Exceptions

<i>IL_ILLEGAL_OPERATION</i>	No currently bound image.
<i>IL_INVALID_PARAM</i>	One of the parameters is incorrect, such as one of the dimensions being 0.
<i>IL_OUT_OF_MEMORY</i>	Could not allocate enough memory.

## Returns

Boolean value of failure or success

#### 7.114.1.22 ILAPI ILboolean ILAPIENTRY ilTexImage\_ ( IImage \* *Image*, ILuint *Width*, ILuint *Height*, ILuint *Depth*, ILubyte *Bpp*, ILenum *Format*, ILenum *Type*, void \* *Data* )

#### 7.114.1.23 ILAPI ILboolean ILAPIENTRY ilTexSubImage\_ ( IImage \* *Image*, void \* *Data* )

## 7.115 src/IL/il\_main.c File Reference

```
#include "il_internal.h"
```

## 7.116 src/IL/il\_manip.c File Reference

```
#include "il_internal.h"
#include "il_manip.h"
```

## Functions

- **ILuint ILAPIENTRY iCopyPixels** (ILImage \*Image, ILuint XOff, ILuint YOff, ILuint ZOff, ILuint Width, ILuint Height, ILuint Depth, IEnum Format, IEnum Type, void \*Data)
- **ILboolean iCopyPixels1D** (ILImage \*Image, ILuint XOff, ILuint Width, void \*Data)
- **ILboolean iCopyPixels2D** (ILImage \*Image, ILuint XOff, ILuint YOff, ILuint Width, ILuint Height, void \*Data)
- **ILboolean iCopyPixels3D** (ILImage \*Image, ILuint XOff, ILuint YOff, ILuint ZOff, ILuint Width, ILuint Height, ILuint Depth, void \*Data)
- **ILboolean iDefaultImage** (ILImage \*Image)  
*Creates an ugly 64x64 black and yellow checkerboard image.*
- **ILAPI void ILAPIENTRY iFlipBuffer** (ILubyte \*buff, ILuint depth, ILuint line\_size, ILuint line\_num)
- **ILboolean ILAPIENTRY iFlipImage** (ILImage \*Image)
- **ILubyte \*ILAPIENTRY iGetFlipped** (ILImage \*img)
- **ILboolean ILAPIENTRY iClampNTSC** (void)  
*Clamps data values of unsigned bytes from 16 to 235 for display on an.*
- **ILubyte \*ILAPIENTRY iGetAlpha** (IEnum Type)
- **void ILAPIENTRY iModAlpha** (ILdouble AlphaValue)
- **ILboolean ILAPIENTRY iSetAlpha** (ILdouble AlphaValue)
- **void ILAPIENTRY iSetPixels** (ILint XOff, ILint YOff, ILint ZOff, ILuint Width, ILuint Height, ILuint Depth, IEnum Format, IEnum Type, void \*Data)
- **ILboolean ILAPIENTRY iMirrorImage** (ILImage \*Image)  
*Mirrors an image over its y axis.*
- **void ILAPIENTRY iSetPixels** (ILImage \*Image, ILint XOff, ILint YOff, ILint ZOff, ILuint Width, ILuint Height, ILuint Depth, IEnum Format, IEnum Type, void \*Data)
- **ILboolean iSetPixels1D** (ILImage \*Image, ILint XOff, ILuint Width, void \*Data)
- **ILboolean iSetPixels2D** (ILImage \*Image, ILint XOff, ILint YOff, ILuint Width, ILuint Height, void \*Data)
- **ILboolean iSetPixels3D** (ILImage \*Image, ILint XOff, ILint YOff, ILint ZOff, ILuint Width, ILuint Height, ILuint Depth, void \*Data)

### 7.116.1 Function Documentation

**7.116.1.1 ILuint ILAPIENTRY iCopyPixels** ( ILImage \* Image, ILuint XOff, ILuint YOff, ILuint ZOff, ILuint Width, ILuint Height, ILuint Depth, IEnum Format, IEnum Type, void \* Data )

**7.116.1.2 ILboolean iCopyPixels1D** ( ILImage \* Image, ILuint XOff, ILuint Width, void \* Data )

**7.116.1.3 ILboolean iCopyPixels2D** ( ILImage \* Image, ILuint XOff, ILuint YOff, ILuint Width, ILuint Height, void \* Data )

**7.116.1.4 ILboolean iCopyPixels3D** ( ILImage \* Image, ILuint XOff, ILuint YOff, ILuint ZOff, ILuint Width, ILuint Height, ILuint Depth, void \* Data )

**7.116.1.5 ILboolean iDefaultImage** ( ILImage \* Image )

Creates an ugly 64x64 black and yellow checkerboard image.

**7.116.1.6 ILAPI void ILAPIENTRY iFlipBuffer** ( ILubyte \* buff, ILuint depth, ILuint line\_size, ILuint line\_num )

**7.116.1.7 ILboolean ILAPIENTRY iFlipImage** ( ILImage \* Image )

**7.116.1.8 ILubyte\* ILAPIENTRY iGetFlipped** ( ILImage \* img )

**7.116.1.9 ILboolean ILAPIENTRY iClampNTSC** ( void )

Clamps data values of unsigned bytes from 16 to 235 for display on an.

7.116.1.10 **ILubyte\*** ILAPIENTRY ilGetAlpha ( **ILenum Type** )

7.116.1.11 **void** ILAPIENTRY ilModAlpha ( **ILdouble AlphaValue** )

7.116.1.12 **ILboolean** ILAPIENTRY ilSetAlpha ( **ILdouble AlphaValue** )

7.116.1.13 **void** ILAPIENTRY ilSetPixels ( **ILint XOff, ILint YOff, ILint ZOff, ILuint Width, ILuint Height, ILuint Depth, ILenum Format, ILenum Type, void \* Data** )

7.116.1.14 **ILboolean** ILAPIENTRY iMirrorImage ( **ILimage \* Image** )

Mirrors an image over its y axis.

7.116.1.15 **void** ILAPIENTRY iSetPixels ( **ILimage \* Image, ILint XOff, ILint YOff, ILint ZOff, ILuint Width, ILuint Height, ILuint Depth, ILenum Format, ILenum Type, void \* Data** )

7.116.1.16 **ILboolean** iSetPixels1D ( **ILimage \* Image, ILint XOff, ILuint Width, void \* Data** )

7.116.1.17 **ILboolean** iSetPixels2D ( **ILimage \* Image, ILint XOff, ILint YOff, ILuint Width, ILuint Height, void \* Data** )

7.116.1.18 **ILboolean** iSetPixels3D ( **ILimage \* Image, ILint XOff, ILint YOff, ILint ZOff, ILuint Width, ILuint Height, ILuint Depth, void \* Data** )

## 7.117 src/IL/il\_manip.h File Reference

### Functions

- **ILuint** iCopyPixels ( **ILimage \*Image, ILuint XOff, ILuint YOff, ILuint ZOff, ILuint Width, ILuint Height, ILuint Depth, ILenum Format, ILenum Type, void \*Data** )
- **ILboolean** iDefaultImage ( **ILimage \*Image** )  
Creates an ugly 64x64 black and yellow checkerboard image.
- **INLINE ILushort** ILAPIENTRY ilFloatToHalf ( **ILuint i** )
- **INLINE ILfloat** ilFloatToHalfOverflow ( )
- **INLINE ILuint** ILAPIENTRY ilHalfToFloat ( **ILushort y** )

### 7.117.1 Function Documentation

7.117.1.1 **ILuint** iCopyPixels ( **ILimage \* Image, ILuint XOff, ILuint YOff, ILuint ZOff, ILuint Width, ILuint Height, ILuint Depth, ILenum Format, ILenum Type, void \* Data** )

7.117.1.2 **ILboolean** iDefaultImage ( **ILimage \* Image** )

Creates an ugly 64x64 black and yellow checkerboard image.

7.117.1.3 **INLINE ILushort** ILAPIENTRY ilFloatToHalf ( **ILuint i** )

7.117.1.4 **INLINE ILfloat** ilFloatToHalfOverflow ( )

7.117.1.5 **INLINE ILuint** ILAPIENTRY ilHalfToFloat ( **ILushort y** )

## 7.118 src/IL/il\_pal.c File Reference

```
#include "il_internal.h"
#include "il_pal.h"
#include <string.h>
#include <ctype.h>
#include <limits.h>
```

### Data Structures

- struct [COL\\_CUBE](#)

### Typedefs

- typedef struct [COL\\_CUBE](#) [COL\\_CUBE](#)

### Functions

- [ILboolean](#) [ILAPIENTRY](#) [iConvertImagePal](#) ([ILimage](#) \*Image, [ILenum](#) DestFormat)
- [ILAPI](#) [ILpal](#) \*[ILAPIENTRY](#) [iConvertPal](#) ([ILpal](#) \*Pal, [ILenum](#) DestFormat)
- [ILAPI](#) [ILpal](#) \*[ILAPIENTRY](#) [iCopyPal](#) ([ILimage](#) \*Image)
- [ILboolean](#) [iCopyPalette](#) ([ILpal](#) \*Dest, [ILpal](#) \*Src)
- [ILboolean](#) [ILAPIENTRY](#) [ilApplyPal](#) ([ILconst\\_string](#) FileName)
- [ILboolean](#) [ILAPIENTRY](#) [ilConvertPal](#) ([ILenum](#) DestFormat)  
*Converts the current image to the DestFormat format.*
- [ILboolean](#) [ILAPIENTRY](#) [ilLoadPal](#) ([ILconst\\_string](#) FileName)  
*Loads a palette from FileName into the current image's palette.*
- [ILboolean](#) [ILAPIENTRY](#) [ilSavePal](#) ([ILconst\\_string](#) FileName)
- [ILAPI](#) [void](#) [ILAPIENTRY](#) [iSetPal](#) ([ILimage](#) \*Image, [ILpal](#) \*Pal)
- [int](#) [sort\\_func](#) ([void](#) \*e1, [void](#) \*e2)

### Variables

- [ILuint](#) CurSort = 0

#### 7.118.1 Typedef Documentation

7.118.1.1 typedef struct [COL\\_CUBE](#) [COL\\_CUBE](#)

#### 7.118.2 Function Documentation

7.118.2.1 [ILboolean](#) [ILAPIENTRY](#) [iConvertImagePal](#) ( [ILimage](#) \* *Image*, [ILenum](#) *DestFormat* )

7.118.2.2 [ILAPI](#) [ILpal](#)\* [ILAPIENTRY](#) [iConvertPal](#) ( [ILpal](#) \* *Pal*, [ILenum](#) *DestFormat* )

7.118.2.3 [ILAPI](#) [ILpal](#)\* [ILAPIENTRY](#) [iCopyPal](#) ( [ILimage](#) \* *Image* )

7.118.2.4 [ILboolean](#) [iCopyPalette](#) ( [ILpal](#) \* *Dest*, [ILpal](#) \* *Src* )

7.118.2.5 [ILboolean](#) [ILAPIENTRY](#) [ilApplyPal](#) ( [ILconst\\_string](#) *FileName* )

7.118.2.6 ILboolean ILAPIENTRY ilConvertPal ( ILenum *DestFormat* )

Converts the current image to the DestFormat format.

7.118.2.7 ILboolean ILAPIENTRY ilLoadPal ( ILconst\_string *FileName* )

Loads a palette from FileName into the current image's palette.

7.118.2.8 ILboolean ILAPIENTRY ilSavePal ( ILconst\_string *FileName* )7.118.2.9 ILAPI void ILAPIENTRY iSetPal ( ILimage \* *Image*, ILpal \* *Pal* )7.118.2.10 int sort\_func ( void \* *e1*, void \* *e2* )

## 7.118.3 Variable Documentation

## 7.118.3.1 ILuint CurSort = 0

## 7.119 src/IL/il\_pal.h File Reference

```
#include "il_internal.h"
```

## Functions

- [ILAPI ILboolean ILAPIENTRY iConvertImagePal](#) (ILimage \*Image, ILenum DestFormat)
- [ILboolean iCopyPalette](#) (ILpal \*Dest, ILpal \*Src)

## 7.119.1 Function Documentation

7.119.1.1 ILAPI ILboolean ILAPIENTRY iConvertImagePal ( ILimage \* *Image*, ILenum *DestFormat* )7.119.1.2 ILboolean iCopyPalette ( ILpal \* *Dest*, ILpal \* *Src* )

## 7.120 src/IL/il\_profiles.c File Reference

```
#include "il_internal.h"
#include <lcms/lcms.h>
```

## Macros

- [#define NON\\_WINDOWS](#) 1

## Functions

- [ILboolean ILAPIENTRY ilApplyProfile](#) (ILstring InProfile, ILstring OutProfile)

## 7.120.1 Macro Definition Documentation

### 7.120.1.1 `#define NON_WINDOWS 1`

## 7.120.2 Function Documentation

### 7.120.2.1 `ILboolean ILAPIENTRY ilApplyProfile ( ILstring InProfile, ILstring OutProfile )`

## 7.121 `src/IL/il_register.c` File Reference

```
#include "il_internal.h"
#include "il_register.h"
#include <string.h>
```

### Functions

- `void ILAPIENTRY ilRegisterFormat (ILenum Format)`
- `ILboolean ILAPIENTRY ilRegisterLoad (ILconst_string Ext, IL_LOADPROC Load)`
- `ILboolean ILAPIENTRY ilRegisterMipNum (ILuint Num)`
- `ILboolean ILAPIENTRY ilRegisterNumFaces (ILuint Num)`
- `ILboolean ILAPIENTRY ilRegisterNumImages (ILuint Num)`
- `void ILAPIENTRY ilRegisterOrigin (ILenum Origin)`
- `void ILAPIENTRY ilRegisterPal (void *Pal, ILuint Size, ILenum Type)`
- `ILboolean ILAPIENTRY ilRegisterSave (ILconst_string Ext, IL_SAVEPROC Save)`
- `void ILAPIENTRY ilRegisterType (ILenum Type)`
- `ILboolean ILAPIENTRY ilRemoveLoad (ILconst_string Ext)`  
*Unregisters a load extension - doesn't have to be called.*
- `void ilRemoveRegistered ()`
- `ILboolean ILAPIENTRY ilRemoveSave (ILconst_string Ext)`  
*Unregisters a save extension - doesn't have to be called.*
- `ILboolean ILAPIENTRY ilSetDuration (ILuint Duration)`
- `ILboolean iRegisterLoad (ILconst_string FileName)`
- `ILboolean iRegisterSave (ILconst_string FileName)`

### Variables

- `iFormatL * LoadProcs = NULL`
- `iFormatS * SaveProcs = NULL`

## 7.121.1 Function Documentation

### 7.121.1.1 `void ILAPIENTRY ilRegisterFormat ( ILenum Format )`

### 7.121.1.2 `ILboolean ILAPIENTRY ilRegisterLoad ( ILconst_string Ext, IL_LOADPROC Load )`

### 7.121.1.3 `ILboolean ILAPIENTRY ilRegisterMipNum ( ILuint Num )`

### 7.121.1.4 `ILboolean ILAPIENTRY ilRegisterNumFaces ( ILuint Num )`

### 7.121.1.5 `ILboolean ILAPIENTRY ilRegisterNumImages ( ILuint Num )`

7.121.1.6 void ILAPIENTRY ilRegisterOrigin ( IEnum *Origin* )

7.121.1.7 void ILAPIENTRY ilRegisterPal ( void \* *Pal*, ILuint *Size*, IEnum *Type* )

7.121.1.8 ILboolean ILAPIENTRY ilRegisterSave ( ILconst\_string *Ext*, IL\_SAVEPROC *Save* )

7.121.1.9 void ILAPIENTRY ilRegisterType ( IEnum *Type* )

7.121.1.10 ILboolean ILAPIENTRY ilRemoveLoad ( ILconst\_string *Ext* )

Unregisters a load extension - doesn't have to be called.

7.121.1.11 void ilRemoveRegistered ( void )

7.121.1.12 ILboolean ILAPIENTRY ilRemoveSave ( ILconst\_string *Ext* )

Unregisters a save extension - doesn't have to be called.

7.121.1.13 ILboolean ILAPIENTRY ilSetDuration ( ILuint *Duration* )

7.121.1.14 ILboolean iRegisterLoad ( ILconst\_string *FileName* )

7.121.1.15 ILboolean iRegisterSave ( ILconst\_string *FileName* )

## 7.121.2 Variable Documentation

7.121.2.1 iFormatL\* LoadProcs = NULL

7.121.2.2 iFormatS\* SaveProcs = NULL

## 7.122 src/IL/il\_register.h File Reference

```
#include "il_internal.h"
```

### Data Structures

- struct [iFormatL](#)
- struct [iFormatS](#)

### Macros

- #define [I\\_LOAD\\_FUNC](#) 0
- #define [I\\_SAVE\\_FUNC](#) 1

### Typedefs

- typedef struct [iFormatL](#) [iFormatL](#)
- typedef struct [iFormatS](#) [iFormatS](#)

## Functions

- [ILboolean iRegisterLoad \(ILconst\\_string FileName\)](#)
- [ILboolean iRegisterSave \(ILconst\\_string FileName\)](#)

### 7.122.1 Macro Definition Documentation

7.122.1.1 `#define I_LOAD_FUNC 0`

7.122.1.2 `#define I_SAVE_FUNC 1`

### 7.122.2 Typedef Documentation

7.122.2.1 `typedef struct iFormatL iFormatL`

7.122.2.2 `typedef struct iFormatS iFormatS`

### 7.122.3 Function Documentation

7.122.3.1 `ILboolean iRegisterLoad ( ILconst_string FileName )`

7.122.3.2 `ILboolean iRegisterSave ( ILconst_string FileName )`

## 7.123 src/IL/il\_size.c File Reference

```
#include "il_internal.h"
```

## Functions

- [ILAPI ILuint ILAPIENTRY ilDetermineSize \(ILenum Type\)](#)  
*Returns the size of the memory buffer needed to save the current image into this Type.*
- [ILint ILAPIENTRY iSizePutc \(ILubyte Char, ILHANDLE h\)](#)
- [ILint ILAPIENTRY iSizeSeek \(ILHANDLE h, ILint Offset, ILuint Mode\)](#)  
*Fake seek function.*
- [ILuint ILAPIENTRY iSizeTell \(ILHANDLE h\)](#)
- [ILint ILAPIENTRY iSizeWrite \(const void \\*Buffer, ILuint Size, ILuint Number, ILHANDLE h\)](#)

### 7.123.1 Function Documentation

7.123.1.1 `ILAPI ILuint ILAPIENTRY ilDetermineSize ( ILenum Type )`

Returns the size of the memory buffer needed to save the current image into this Type.

7.123.1.2 `ILint ILAPIENTRY iSizePutc ( ILubyte Char, ILHANDLE h )`

7.123.1.3 `ILint ILAPIENTRY iSizeSeek ( ILHANDLE h, ILint Offset, ILuint Mode )`

Fake seek function.



7.123.1.4 ILuint ILAPIENTRY iSizeTell ( ILHANDLE h )

7.123.1.5 ILint ILAPIENTRY iSizeWrite ( const void \* Buffer, ILuint Size, ILuint Number, ILHANDLE h )

## 7.124 src/IL/il\_skia.cc File Reference

```
#include "il_internal.h"
#include "Windows.h"
#include "D:\\devel\\devil-1.7.8 - 2012-04-19\\include\\IL\\il.h"
#include "core\\SkStream.h"
#include "images\\SkImageDecoder.h"
#include "core\\SkTemplates.h"
```

### Functions

- ILboolean \_\_stdcall skiaLoadImage (void \*buf, size\_t bufSize)

#### 7.124.1 Function Documentation

7.124.1.1 ILboolean \_\_stdcall skiaLoadImage ( void \* buf, size\_t bufSize )

## 7.125 src/IL/il\_stack.c File Reference

```
#include "il_internal.h"
#include "il_stack.h"
```

### Functions

- void iBindImage (ILuint Image)
- ILAPI void ILAPIENTRY iBindImageTemp ()
- ILuint iCreateSubImage (ILimage \*Image, IEnum Type, ILuint Num)
- ILboolean iEnlargeStack ()
- ILAPI ILimage \*ILAPIENTRY iGetBaseImage ()
- ILAPI ILimage \*ILAPIENTRY iGetCurlImage ()
- ILimage \*ILAPIENTRY iGetImage (ILuint Image)
- ILimage \*ILAPIENTRY iGetMipmap (ILimage \*Image, ILuint Number)
- ILimage \*ILAPIENTRY iGetSubImage (ILimage \*Image, ILuint Number)
 

*Used for setting the current image if it is an animation.*
- void iInitIL ()
- ILboolean iIsImage (ILuint Image)
 

*Checks if Image is a valid iGenImages-generated image (like gllsTexture()).*
- ILboolean ILAPIENTRY iActiveFace (ILuint Number)
 

*Used for setting the current face if it is a cubemap.*
- ILboolean ILAPIENTRY iActiveImage (ILuint Number)
 

*Used for setting the current image if it is an animation.*
- ILboolean ILAPIENTRY iActiveLayer (ILuint Number)
 

*Used for setting the current layer if layers exist.*
- ILboolean ILAPIENTRY iActiveMipmap (ILuint Number)
 

*Sets the current mipmap level.*

- **ILAPI void ILAPIENTRY ilCloseImage** (ILImage \*Image)  
*Closes Image and frees all memory associated with it.*
- **ILAPI void ILAPIENTRY ilClosePal** (ILpal \*Palette)  
*Closes Palette and frees all memory associated with it.*
- **void ILAPIENTRY ilDeleteImage** (const ILuint Num)
- **void ILAPIENTRY ilDeleteImages** (ILsizei Num, const ILuint \*Images)  
*Deletes Num images from the image stack - similar to glDeleteTextures().*
- **ILuint ILAPIENTRY ilGenImage** ()
- **void ILAPIENTRY ilGenImages** (ILsizei Num, ILuint \*Images)  
*Creates Num images and puts their index in Images - similar to glGenTextures().*
- **ILAPI ILuint ILAPIENTRY ilGetCurName** ()
- **ILAPI ILboolean ILAPIENTRY ilIsValidPal** (ILpal \*Palette)
- **void \*ILAPIENTRY ilRealloc** (void \*Ptr, ILuint OldSize, ILuint NewSize)
- **ILAPI void ILAPIENTRY ilReplaceCurlImage** (ILImage \*Image)
- **ILAPI void ILAPIENTRY ilSetCurlImage** (ILImage \*Image)
- **void iShutDownIL** ()

## Variables

- **ILuint CurName** = 0
- **iFree \* FreeNames** = NULL
- **ILImage \* iCurlImage** = NULL
- **ILImage \*\* ImageStack** = NULL
- **ILuint LastUsed** = 0
- **ILboolean OnExit** = IL\_FALSE
- **ILboolean ParentImage** = IL\_TRUE
- **ILuint StackSize** = 0

## 7.125.1 Function Documentation

7.125.1.1 **void iBindImage** ( ILuint Image )

7.125.1.2 **ILAPI void ILAPIENTRY iBindImageTemp** ( void )

7.125.1.3 **ILuint iCreateSubImage** ( ILImage \* Image, ILenum Type, ILuint Num )

7.125.1.4 **ILboolean iEnlargeStack** ( void )

7.125.1.5 **ILAPI ILImage\* ILAPIENTRY iGetBaseImage** ( void )

7.125.1.6 **ILAPI ILImage\* ILAPIENTRY iGetCurlImage** ( void )

7.125.1.7 **ILImage\* ILAPIENTRY iGetImage** ( ILuint Image )

7.125.1.8 **ILImage\* ILAPIENTRY iGetMipmap** ( ILImage \* Image, ILuint Number )

7.125.1.9 **ILImage\* ILAPIENTRY iGetSubImage** ( ILImage \* Image, ILuint Number )

Used for setting the current image if it is an animation.

7.125.1.10 **void** `ilInitIL ( )`

7.125.1.11 **ILboolean** `ilIsImage ( ILuint Image )`

Checks if Image is a valid ilGenImages-generated image (like glIsTexture()).

7.125.1.12 **ILboolean ILAPIENTRY** `ilActiveFace ( ILuint Number )`

Used for setting the current face if it is a cubemap.

7.125.1.13 **ILboolean ILAPIENTRY** `ilActiveImage ( ILuint Number )`

Used for setting the current image if it is an animation.

7.125.1.14 **ILboolean ILAPIENTRY** `ilActiveLayer ( ILuint Number )`

Used for setting the current layer if layers exist.

7.125.1.15 **ILboolean ILAPIENTRY** `ilActiveMipmap ( ILuint Number )`

Sets the current mipmap level.

7.125.1.16 **ILAPI void ILAPIENTRY** `ilCloseImage ( IImage * Image )`

Closes Image and frees all memory associated with it.

7.125.1.17 **ILAPI void ILAPIENTRY** `ilClosePal ( ILpal * Palette )`

Closes Palette and frees all memory associated with it.

7.125.1.18 **void ILAPIENTRY** `ilDeleteImage ( const ILuint Num )`

7.125.1.19 **void ILAPIENTRY** `ilDeleteImages ( ILsizei Num, const ILuint * Images )`

Deletes Num images from the image stack - similar to glDeleteTextures().

7.125.1.20 **ILuint ILAPIENTRY** `ilGenImage ( void )`

7.125.1.21 **void ILAPIENTRY** `ilGenImages ( ILsizei Num, ILuint * Images )`

Creates Num images and puts their index in Images - similar to glGenTextures().

7.125.1.22 **ILAPI ILuint ILAPIENTRY** `ilGetCurName ( void )`

7.125.1.23 **ILAPI ILboolean ILAPIENTRY** `ilIsValidPal ( ILpal * Palette )`

7.125.1.24 **void\* ILAPIENTRY** `ilRealloc ( void * Ptr, ILuint OldSize, ILuint NewSize )`

7.125.1.25 **ILAPI void ILAPIENTRY** `ilReplaceCurlImage ( IImage * Image )`

7.125.1.26 **ILAPI** void ILAPIENTRY ilSetCurlImage ( **ILImage** \* *Image* )

7.125.1.27 void iShutDownIL ( )

## 7.125.2 Variable Documentation

7.125.2.1 **ILuint** CurName = 0

7.125.2.2 **iFree**\* FreeNames = NULL

7.125.2.3 **ILImage**\* iCurlImage = NULL

7.125.2.4 **ILImage**\*\* ImageStack = NULL

7.125.2.5 **ILuint** LastUsed = 0

7.125.2.6 **ILboolean** OnExit = **IL\_FALSE**

7.125.2.7 **ILboolean** ParentImage = **IL\_TRUE**

7.125.2.8 **ILuint** StackSize = 0

## 7.126 src/IL/il\_stack.h File Reference

```
#include "il_internal.h"
```

### Data Structures

- struct [iFree](#)

### Macros

- #define [I\\_STACK\\_INCREMENT](#) 1024

### Typedefs

- typedef struct [iFree](#) [iFree](#)

### Functions

- void [iBindImage](#) (**ILuint**)
- **ILuint** [iCreateSubImage](#) (**ILImage** \*Image, **ILenum** Type, **ILuint** Num)
- **ILboolean** [iEnlargeStack](#) (void)
- void [iFreeMem](#) (void)
- void [ilnitIL](#) ()
- **ILboolean** [ilslImage](#) (**ILuint** Image)  
Checks if Image is a valid *ilGenImages*-generated image (like *glIsTexture()*).
- void [iShutDownIL](#) ()

## 7.126.1 Macro Definition Documentation

7.126.1.1 `#define I_STACK_INCREMENT 1024`

## 7.126.2 Typedef Documentation

7.126.2.1 `typedef struct iFree iFree`

## 7.126.3 Function Documentation

7.126.3.1 `void iBindImage ( ILuint )`

7.126.3.2 `ILuint iCreateSubImage ( IImage * Image, IEnum Type, ILuint Num )`

7.126.3.3 `ILboolean iEnlargeStack ( void )`

7.126.3.4 `void iFreeMem ( void )`

7.126.3.5 `void ilnitiL ( )`

7.126.3.6 `ILboolean ilslImage ( ILuint Image )`

Checks if Image is a valid ilGenImages-generated image (like gllsTexture()).

7.126.3.7 `void iShutDownIL ( )`

## 7.127 src/IL/il\_states.c File Reference

```
#include "il_internal.h"
#include "il_states.h"
#include "il_stack.h"
#include <stdlib.h>
```

### Macros

- `#define _strdup strdup`

### Functions

- `ILboolean iAble (IEnum Mode, ILboolean Flag)`
- `char * iClipString (ILconst_string String_, ILuint MaxLen)`
- `ILboolean iFormatFunc (IEnum Mode)`
- `ILuint iGetActiveNum (IEnum Type)`  
*Internal function to figure out where we are in an image chain.*
- `IEnum iGetHint (IEnum Target)`
- `ILconst_string iGetILString (IEnum StringName)`  
*Returns a constant string detailing aspects about this library.*
- `ILint iGetInt (IEnum Mode)`
- `ILint iGetInteger (IEnum Mode)`
- `ILint ILAPIENTRY iGetIntegerImage (IImage *Image, IEnum Mode)`  
*Sets Param equal to the current value of the Mode.*
- `char * iGetString (IEnum StringName)`

- `void iHint (ILenum Target, ILenum Mode)`  
*Specifies implementation-dependent performance hints.*
- `ILboolean ilsEnabled (ILenum Mode)`  
*Checks whether the mode is enabled.*
- `ILboolean ILAPIENTRY ilCompressFunc (ILenum Mode)`
- `void ilDefaultStates ()`  
*Set all states to their defaults.*
- `ILboolean ILAPIENTRY ilOriginFunc (ILenum Mode)`  
*Sets the default origin to be used.*
- `void ILAPIENTRY ilPopAttrib ()`  
*Pops the last entry off the state stack into the current states.*
- `void ILAPIENTRY ilPushAttrib (ILuint Bits)`  
*Pushes the states indicated by Bits onto the state stack.*
- `ILboolean ILAPIENTRY ilTypeFunc (ILenum Mode)`  
*Sets the default type to be used.*
- `void iSetInteger (ILenum Mode, ILint Param)`
- `void iSetString (ILenum Mode, const char *String_)`

## Variables

- `ILchar * _ilLoadExt`
- `ILchar * _ilSaveExt`
- `ILconst_string _ilVendor = IL_TEXT("kolrabi")`
- `ILconst_string _ilVersion = IL_TEXT("kolrabi's another Image Library (kaLL) 1.8.3")`
- `ILuint ilCurrentPos = 0`
- `IL_HINTS ilHints`
- `IL_STATES ilStates [IL_ATTRIB_STACK_MAX]`

## 7.127.1 Macro Definition Documentation

7.127.1.1 `#define _strdup strdup`

## 7.127.2 Function Documentation

7.127.2.1 `ILboolean iAble ( ILenum Mode, ILboolean Flag )`

7.127.2.2 `char* iClipString ( ILconst_string String_, ILuint MaxLen )`

7.127.2.3 `ILboolean iFormatFunc ( ILenum Mode )`

7.127.2.4 `ILuint iGetActiveNum ( ILenum Type )`

Internal function to figure out where we are in an image chain.

7.127.2.5 `ILenum iGetHint ( ILenum Target )`

7.127.2.6 `ILconst_string iGetILString ( ILenum StringName )`

Returns a constant string detailing aspects about this library.

7.127.2.7 **ILint** iGetInt ( *ILenum Mode* )

7.127.2.8 **ILint** iGetInteger ( *ILenum Mode* )

7.127.2.9 **ILint ILAPIENTRY** iGetIntegerImage ( *ILimage \* Image, ILenum Mode* )

Sets Param equal to the current value of the Mode.

7.127.2.10 **char\*** iGetString ( *ILenum StringName* )

7.127.2.11 **void** iHint ( *ILenum Target, ILenum Mode* )

Specifies implementation-dependent performance hints.

7.127.2.12 **ILboolean** ilsEnabled ( *ILenum Mode* )

Checks whether the mode is enabled.

7.127.2.13 **ILboolean ILAPIENTRY** ilCompressFunc ( *ILenum Mode* )

7.127.2.14 **void** ilDefaultStates ( *void* )

Set all states to their defaults.

7.127.2.15 **ILboolean ILAPIENTRY** ilOriginFunc ( *ILenum Mode* )

Sets the default origin to be used.

7.127.2.16 **void ILAPIENTRY** ilPopAttrib ( *void* )

Pops the last entry off the state stack into the current states.

7.127.2.17 **void ILAPIENTRY** ilPushAttrib ( *ILuint Bits* )

Pushes the states indicated by Bits onto the state stack.

7.127.2.18 **ILboolean ILAPIENTRY** ilTypeFunc ( *ILenum Mode* )

Sets the default type to be used.

7.127.2.19 **void** iSetInteger ( *ILenum Mode, ILint Param* )

7.127.2.20 **void** iSetString ( *ILenum Mode, const char \* String\_* )

### 7.127.3 Variable Documentation

7.127.3.1 **ILchar\*** \_ilLoadExt

7.127.3.2 **ILchar\*** \_ilSaveExt

7.127.3.3 **ILconst\_string** \_ilVendor = IL\_TEXT("kolrabi")

7.127.3.4 `ILconst_string ilVersion = IL_TEXT("kolrabi's another Image Library (kal) 1.8.3")`

7.127.3.5 `ILuint ilCurrentPos = 0`

7.127.3.6 `IL_HINTS ilHints`

7.127.3.7 `IL_STATES ilStates[IL_ATTRIB_STACK_MAX]`

## 7.128 `src/IL/il_states.h` File Reference

```
#include "il_internal.h"
```

### Data Structures

- struct [IL\\_HINTS](#)
- struct [IL\\_STATES](#)

### Macros

- `#define IL_ATTRIB_STACK_MAX 32`

### Typedefs

- typedef struct [IL\\_HINTS](#) [IL\\_HINTS](#)
- typedef struct [IL\\_STATES](#) [IL\\_STATES](#)

### Functions

- [ILboolean iAble](#) ([ILenum](#) Mode, [ILboolean](#) Flag)
- [ILboolean iFormatFunc](#) ([ILenum](#) Mode)
- [ILboolean iGetBoolean](#) ([ILenum](#) Mode)
- [ILint iGetInteger](#) ([ILenum](#) Mode)
- [ILboolean ilsEnabled](#) ([ILenum](#) Mode)  
*Checks whether the mode is enabled.*
- [void iSetInteger](#) ([ILenum](#) Mode, [ILint](#) Param)
- [void iSetString](#) ([ILenum](#) Mode, const char \*String\_)

### Variables

- [ILuint ilCurrentPos](#)
- [IL\\_HINTS ilHints](#)
- [IL\\_STATES ilStates](#) [[IL\\_ATTRIB\\_STACK\\_MAX](#)]

## 7.128.1 Macro Definition Documentation

7.128.1.1 `#define IL_ATTRIB_STACK_MAX 32`

## 7.128.2 Typedef Documentation



7.128.2.1 typedef struct IL\_HINTS IL\_HINTS

7.128.2.2 typedef struct IL\_STATES IL\_STATES

### 7.128.3 Function Documentation

7.128.3.1 ILboolean iAble ( IEnum *Mode*, ILboolean *Flag* )

7.128.3.2 ILboolean iFormatFunc ( IEnum *Mode* )

7.128.3.3 ILboolean iGetBoolean ( IEnum *Mode* )

7.128.3.4 ILint iGetInteger ( IEnum *Mode* )

7.128.3.5 ILboolean ilsEnabled ( IEnum *Mode* )

Checks whether the mode is enabled.

7.128.3.6 void iSetInteger ( IEnum *Mode*, ILint *Param* )

7.128.3.7 void iSetString ( IEnum *Mode*, const char \* *String\_* )

### 7.128.4 Variable Documentation

7.128.4.1 ILuint iCurrentPos

7.128.4.2 IL\_HINTS ilHints

7.128.4.3 IL\_STATES ilStates[IL\_ATTRIB\_STACK\_MAX]

## 7.129 src/IL/il\_string.c File Reference

```
#include "il_internal.h"
#include <ctype.h>
```

### Functions

- char \*ILAPIENTRY iCharStrDup (const char \*Str)
- ILboolean ILAPIENTRY iCheckExtension (ILconst\_string Arg, ILconst\_string Ext)
- ILstring ILAPIENTRY iGetExtension (ILconst\_string FileName)
- ILstring ILAPIENTRY iStrDup (ILconst\_string Str)

*Glut's portability.txt says to use this...*

### 7.129.1 Function Documentation

7.129.1.1 char\* ILAPIENTRY iCharStrDup ( const char \* *Str* )

7.129.1.2 ILboolean ILAPIENTRY iCheckExtension ( ILconst\_string *Arg*, ILconst\_string *Ext* )

7.129.1.3 ILstring ILAPIENTRY iGetExtension ( ILconst\_string *FileName* )

#### 7.129.1.4 ILstring ILAPIENTRY iStrDup ( ILconst\_string Str )

Glut's portability.txt says to use this...

### 7.130 src/IL/il\_string.h File Reference

#### 7.131 src/IL/il\_utility.c File Reference

```
#include "il_internal.h"
```

#### Functions

- [ILAPI ILubyte ILAPIENTRY ilGetBpcType \(ILenum Type\)](#)
- [ILAPI ILubyte ILAPIENTRY ilGetBppFormat \(ILenum Format\)](#)
- [ILAPI ILubyte ILAPIENTRY ilGetBppPal \(ILenum PalType\)](#)
- [ILAPI ILenum ILAPIENTRY ilGetFormatBpp \(ILubyte Bpp\)](#)
- [ILAPI ILenum ILAPIENTRY ilGetPalBaseType \(ILenum PalType\)](#)
- [ILAPI ILenum ILAPIENTRY ilGetTypeBpc \(ILubyte Bpc\)](#)
- [ILAPI ILuint ILAPIENTRY ilNextPower2 \(ILuint n\)](#)
- [ILAPI void ILAPIENTRY iMemSwap \(ILubyte \\*s1, ILubyte \\*s2, const ILuint size\)](#)

#### 7.131.1 Function Documentation

7.131.1.1 ILAPI ILubyte ILAPIENTRY ilGetBpcType ( ILenum *Type* )

7.131.1.2 ILAPI ILubyte ILAPIENTRY ilGetBppFormat ( ILenum *Format* )

7.131.1.3 ILAPI ILubyte ILAPIENTRY ilGetBppPal ( ILenum *PalType* )

7.131.1.4 ILAPI ILenum ILAPIENTRY ilGetFormatBpp ( ILubyte *Bpp* )

7.131.1.5 ILAPI ILenum ILAPIENTRY ilGetPalBaseType ( ILenum *PalType* )

7.131.1.6 ILAPI ILenum ILAPIENTRY ilGetTypeBpc ( ILubyte *Bpc* )

7.131.1.7 ILAPI ILuint ILAPIENTRY ilNextPower2 ( ILuint *n* )

7.131.1.8 ILAPI void ILAPIENTRY iMemSwap ( ILubyte \* *s1*, ILubyte \* *s2*, const ILuint *size* )

### 7.132 src/IL/pack\_pop.h File Reference

### 7.133 src/IL/pack\_push.h File Reference

### 7.134 src/ILU/ilu\_alloc.c File Reference

```
#include <stdio.h>
```

## 7.135 src/ILU/ilu\_alloc.h File Reference

## 7.136 src/ILU/ilu\_error.c File Reference

```
#include "ilu_internal.h"
#include "ilu_error/ilu_err-arabic.h"
#include "ilu_error/ilu_err-dutch.h"
#include "ilu_error/ilu_err-english.h"
#include "ilu_error/ilu_err-japanese.h"
#include "ilu_error/ilu_err-spanish.h"
#include "ilu_error/ilu_err-german.h"
#include "ilu_error/ilu_err-french.h"
```

### Macros

- `#define ILU_NUM_LANGUAGES 7`

### Functions

- `ILconst_string ILAPIENTRY iluErrorString (ILenum Error)`
- `ILboolean ILAPIENTRY iluSetLanguage (ILenum Language)`

### Variables

- `ILconst_string * iluErrors`
- `ILconst_string * iluErrorStrings [ILU_NUM_LANGUAGES]`
- `ILconst_string * iluLibErrors`
- `ILconst_string * iluLibErrorStrings [ILU_NUM_LANGUAGES]`
- `ILconst_string * iluMiscErrors`
- `ILconst_string * iluMiscErrorStrings [ILU_NUM_LANGUAGES]`

### 7.136.1 Macro Definition Documentation

7.136.1.1 `#define ILU_NUM_LANGUAGES 7`

### 7.136.2 Function Documentation

7.136.2.1 `ILconst_string ILAPIENTRY iluErrorString ( ILenum Error )`

7.136.2.2 `ILboolean ILAPIENTRY iluSetLanguage ( ILenum Language )`

### 7.136.3 Variable Documentation

7.136.3.1 `ILconst_string* iluErrors`

7.136.3.2 `ILconst_string* iluErrorStrings[ILU_NUM_LANGUAGES]`

#### Initial value:

```
= {
    iluErrorStringsEnglish,
    iluErrorStringsArabic,
```

```

    iluErrorStringsDutch,
    iluErrorStringsFrench,
    iluErrorStringsJapanese,
    iluErrorStringsSpanish,
    iluErrorStringsGerman
}

```

### 7.136.3.3 ILconst\_string\* iluLibErrors

### 7.136.3.4 ILconst\_string\* iluLibErrorStrings[ILU\_NUM\_LANGUAGES]

**Initial value:**

```

= {
    iluLibErrorStringsEnglish,
    iluLibErrorStringsArabic,
    iluLibErrorStringsDutch,
    iluLibErrorStringsFrench,
    iluLibErrorStringsJapanese,
    iluLibErrorStringsSpanish,
    iluLibErrorStringsGerman
}

```

### 7.136.3.5 ILconst\_string\* iluMiscErrors

### 7.136.3.6 ILconst\_string\* iluMiscErrorStrings[ILU\_NUM\_LANGUAGES]

**Initial value:**

```

= {
    iluMiscErrorStringsEnglish,
    iluMiscErrorStringsArabic,
    iluMiscErrorStringsDutch,
    iluMiscErrorStringsFrench,
    iluMiscErrorStringsJapanese,
    iluMiscErrorStringsSpanish,
    iluMiscErrorStringsGerman
}

```

## 7.137 src/ILU/ilu\_error/ilu\_err-arabic.h File Reference

```
#include "ilu_internal.h"
```

### Variables

- [ILconst\\_string iluErrorStringsArabic \[IL\\_FILE\\_READ\\_ERROR-IL\\_INVALID\\_ENUM+1\]](#)
- [ILconst\\_string iluLibErrorStringsArabic \[IL\\_LIB\\_EXR\\_ERROR-IL\\_LIB\\_GIF\\_ERROR+1\]](#)
- [ILconst\\_string iluMiscErrorStringsArabic \[2\]](#)

### 7.137.1 Variable Documentation

#### 7.137.1.1 ILconst\_string iluErrorStringsArabic[IL\_FILE\_READ\_ERROR-IL\_INVALID\_ENUM+1]

**Initial value:**

```

= {
    IL_TEXT("enumerant "),
    IL_TEXT("  "),
    IL_TEXT("  "),

```

```

    IL_TEXT(" "),
    IL_TEXT(" "),
    IL_TEXT(" "),
    IL_TEXT(" ( ) "),
    IL_TEXT(" "),
    IL_TEXT(" ( ) "),
    IL_TEXT(" "),
    IL_TEXT(" "),
    IL_TEXT(" "),
    IL_TEXT("out format equivalent"),
    IL_TEXT(" ( ) "),
    IL_TEXT(" ( ) "),
    IL_TEXT(" "),
    IL_TEXT(" ")
}

```

#### 7.137.1.2 ILconst\_string iluLibErrorStringsArabic[IL\_LIB\_EXR\_ERROR-IL\_LIB\_GIF\_ERROR+1]

**Initial value:**

```

= {
    IL_TEXT(" gif"),
    IL_TEXT(" jpeg"),
    IL_TEXT(" png"),
    IL_TEXT(" tiff"),
    IL_TEXT(" mng"),
    IL_TEXT(" jp2"),
}

```

#### 7.137.1.3 ILconst\_string iluMiscErrorStringsArabic[2]

**Initial value:**

```

= {
    IL_TEXT(" "),
}

```

## 7.138 src/ILU/ilu\_error/ilu\_err-dutch.h File Reference

```
#include "ilu_internal.h"
```

### Variables

- [ILconst\\_string iluErrorStringsDutch \[IL\\_FILE\\_READ\\_ERROR-IL\\_INVALID\\_ENUM+1\]](#)
- [ILconst\\_string iluLibErrorStringsDutch \[IL\\_LIB\\_EXR\\_ERROR-IL\\_LIB\\_GIF\\_ERROR+1\]](#)
- [ILconst\\_string iluMiscErrorStringsDutch \[2\]](#)

### 7.138.1 Variable Documentation

#### 7.138.1.1 ILconst\_string iluErrorStringsDutch[IL\_FILE\_READ\_ERROR-IL\_INVALID\_ENUM+1]

**Initial value:**

```

= {
    IL_TEXT("Ongeldige enumerant"),
    IL_TEXT("Geen vrij geheugen meer"),
    IL_TEXT("Format wordt nog niet ondersteund"),
    IL_TEXT("Interne fout"),
}

```

```

    IL_TEXT("Ongeldige waarde"),
    IL_TEXT("Foute bewerking"),
    IL_TEXT("Foute bestandswaarde"),
    IL_TEXT("Foute bestandsbegin"),
    IL_TEXT("Ongeldige parameter"),
    IL_TEXT("Kan het bestand niet openen"),
    IL_TEXT("ongeldige"),
    IL_TEXT("Bestand bestaat reeds"),
    IL_TEXT("uitgaand formaat equivalent"),
    IL_TEXT("stapel overstroming"),
    IL_TEXT("stapel onderstroming"),
    IL_TEXT("ongeldige omzetting"),
    IL_TEXT("slechte afmetingen"),
}

```

#### 7.138.1.2 ILconst\_string iluLibErrorStringsDutch[IL\_LIB\_EXR\_ERROR-IL\_LIB\_GIF\_ERROR+1]

**Initial value:**

```

= {
    IL_TEXT("fout in gif bibliotheek"),
    IL_TEXT("fout in jpeg bibliotheek"),
    IL_TEXT("fout in png bibliotheek"),
    IL_TEXT("fout in tiff bibliotheek"),
    IL_TEXT("fout in mng bibliotheek"),
    IL_TEXT("fout in jp2 bibliotheek"),
}

```

#### 7.138.1.3 ILconst\_string iluMiscErrorStringsDutch[2]

**Initial value:**

```

= {
    IL_TEXT("geen fout"),
}

```

### 7.139 src/ILU/ilu\_error/ilu\_err-english.h File Reference

```
#include "ilu_internal.h"
```

#### Variables

- [ILconst\\_string iluErrorStringsEnglish \[IL\\_FILE\\_READ\\_ERROR-IL\\_INVALID\\_ENUM+1\]](#)
- [ILconst\\_string iluLibErrorStringsEnglish \[IL\\_LIB\\_EXR\\_ERROR-IL\\_LIB\\_GIF\\_ERROR+1\]](#)
- [ILconst\\_string iluMiscErrorStringsEnglish \[2\]](#)

#### 7.139.1 Variable Documentation

##### 7.139.1.1 ILconst\_string iluErrorStringsEnglish[IL\_FILE\_READ\_ERROR-IL\_INVALID\_ENUM+1]

**Initial value:**

```

= {
    IL_TEXT("invalid enumerant"),
    IL_TEXT("out of memory"),
    IL_TEXT("format not supported yet"),
    IL_TEXT("internal error"),
    IL_TEXT("invalid value"),
}

```

```

    IL_TEXT("illegal operation"),
    IL_TEXT("illegal file value"),
    IL_TEXT("invalid file header"),
    IL_TEXT("invalid parameter"),
    IL_TEXT("could not open file"),
    IL_TEXT("invalid extension"),
    IL_TEXT("file already exists"),
    IL_TEXT("out format equivalent"),
    IL_TEXT("stack overflow"),
    IL_TEXT("stack underflow"),
    IL_TEXT("invalid conversion"),
    IL_TEXT("bad dimensions"),
}

```

#### 7.139.1.2 ILconst\_string iluLibErrorStringsEnglish[IL\_LIB\_EXR\_ERROR-IL\_LIB\_GIF\_ERROR+1]

**Initial value:**

```

= {
    IL_TEXT("gif library error"),
    IL_TEXT("jpeg library error"),
    IL_TEXT("png library error"),
    IL_TEXT("tiff library error"),
    IL_TEXT("mng library error"),
    IL_TEXT("jp2 library error"),
}

```

#### 7.139.1.3 ILconst\_string iluMiscErrorStringsEnglish[2]

**Initial value:**

```

= {
    IL_TEXT("no error"),
}

```

## 7.140 src/ILU/ilu\_error/ilu\_err-french.h File Reference

```
#include "ilu_internal.h"
```

### Variables

- [ILconst\\_string iluErrorStringsFrench \[IL\\_FILE\\_READ\\_ERROR-IL\\_INVALID\\_ENUM+1\]](#)
- [ILconst\\_string iluLibErrorStringsFrench \[IL\\_LIB\\_EXR\\_ERROR-IL\\_LIB\\_GIF\\_ERROR+1\]](#)
- [ILconst\\_string iluMiscErrorStringsFrench \[2\]](#)

### 7.140.1 Variable Documentation

#### 7.140.1.1 ILconst\_string iluErrorStringsFrench[IL\_FILE\_READ\_ERROR-IL\_INVALID\_ENUM+1]

**Initial value:**

```

= {
    IL_TEXT("\u00e9num\u00e9ration invalide"),
    IL_TEXT("d\u00e9passement de m\u00e9moire"),
    IL_TEXT("format non support\u00e9"),
    IL_TEXT("erreur interne"),
    IL_TEXT("valeur ill\u00e9gale"),
    IL_TEXT("op\u00e9ration ill\u00e9gale"),
}

```

```

    IL_TEXT("valeur de fichier ill\u00e9gale"),
    IL_TEXT("en-t\u00eate de fichier invalide"),
    IL_TEXT("param\u00eatre invalide"),
    IL_TEXT("ne peut pas ouvrir le fichier"),
    IL_TEXT("extension invalide"),
    IL_TEXT("fichier d\u00e9j\u00e9 existant"),
    IL_TEXT("\u00e9quivalent hors-format"),
    IL_TEXT("stack overflow"),
    IL_TEXT("stack underflow"),
    IL_TEXT("conversion invalide"),
    IL_TEXT("mauvaises dimensions"),
}

```

#### 7.140.1.2 ILconst\_string iluLibErrorStringsFrench[IL\_LIB\_EXR\_ERROR-IL\_LIB\_GIF\_ERROR+1]

**Initial value:**

```

= {
    IL_TEXT("gif : erreur dans la librairie"),
    IL_TEXT("jpeg : erreur dans la librairie"),
    IL_TEXT("png : erreur dans la librairie"),
    IL_TEXT("tiff : erreur dans la librairie"),
    IL_TEXT("mng : erreur dans la librairie"),
    IL_TEXT("jp2 : erreur dans la librairie"),
}

```

#### 7.140.1.3 ILconst\_string iluMiscErrorStringsFrench[2]

**Initial value:**

```

= {
    IL_TEXT("pas d'erreur"),
}

```

### 7.141 src/ILU/ilu\_error/ilu\_err-german.h File Reference

```
#include "ilu_internal.h"
```

#### Variables

- [ILconst\\_string iluErrorStringsGerman \[IL\\_FILE\\_READ\\_ERROR-IL\\_INVALID\\_ENUM+1\]](#)
- [ILconst\\_string iluLibErrorStringsGerman \[IL\\_LIB\\_EXR\\_ERROR-IL\\_LIB\\_GIF\\_ERROR+1\]](#)
- [ILconst\\_string iluMiscErrorStringsGerman \[2\]](#)

#### 7.141.1 Variable Documentation

##### 7.141.1.1 ILconst\_string iluErrorStringsGerman[IL\_FILE\_READ\_ERROR-IL\_INVALID\_ENUM+1]

**Initial value:**

```

= {
    IL_TEXT("Ung\u00fcltiger Enumerator"),
    IL_TEXT("Kein Speicher verf\u00fcgbar"),
    IL_TEXT("Das Format wird noch nicht unterst\u00fctzt"),
    IL_TEXT("Interner Fehler"),
    IL_TEXT("Ung\u00fcltiger Wert"),
    IL_TEXT("Unzul\u00e4ssige Operation"),
    IL_TEXT("Unzul\u00e4ssiger Datei-Wert"),
}

```



```

    IL_TEXT("Unzul\u00e4ssiger Datei-Header"),
    IL_TEXT("Unzul\u00e4ssiger Parameter"),
    IL_TEXT("Datei konnte nicht ge\u00f6ffnet werden"),
    IL_TEXT("Ung\u00fcltiger Erweiterung"),
    IL_TEXT("Die Datei existiert bereits"),
    IL_TEXT("out format equivalent"),
    IL_TEXT("Stack \u00dcberlauf"),
    IL_TEXT("Stack Unterlauf"),
    IL_TEXT("Ung\u00fcltige Konvertierung"),
    IL_TEXT("Unzul\u00e4ssige Abmessungen"),
}

```

#### 7.141.1.2 ILconst\_string iluLibErrorStringsGerman[IL\_LIB\_EXR\_ERROR-IL\_LIB\_GIF\_ERROR+1]

**Initial value:**

```

= {
    IL_TEXT("Fehler in der gif Bibliothek"),
    IL_TEXT("Fehler in der jpeg Bibliothek"),
    IL_TEXT("Fehler in der png Bibliothek"),
    IL_TEXT("Fehler in der tiff Bibliothek"),
    IL_TEXT("Fehler in der mng Bibliothek"),
    IL_TEXT("Fehler in der jp2 Bibliothek"),
}

```

#### 7.141.1.3 ILconst\_string iluMiscErrorStringsGerman[2]

**Initial value:**

```

= {
    IL_TEXT("Kein Fehler"),
}

```

## 7.142 src/ILU/ilu\_error/ilu\_err-japanese.h File Reference

```
#include "ilu_internal.h"
```

### Variables

- [ILconst\\_string iluErrorStringsJapanese \[IL\\_FILE\\_READ\\_ERROR-IL\\_INVALID\\_ENUM+1\]](#)
- [ILconst\\_string iluLibErrorStringsJapanese \[IL\\_LIB\\_EXR\\_ERROR-IL\\_LIB\\_GIF\\_ERROR+1\]](#)
- [ILconst\\_string iluMiscErrorStringsJapanese \[2\]](#)

### 7.142.1 Variable Documentation

#### 7.142.1.1 ILconst\_string iluErrorStringsJapanese[IL\_FILE\_READ\_ERROR-IL\_INVALID\_ENUM+1]

**Initial value:**

```

= {
    IL_TEXT(""),
    IL_TEXT(""),
    IL_TEXT(""),
    IL_TEXT(""),
    IL_TEXT(""),
    IL_TEXT(""),
    IL_TEXT(""),
    IL_TEXT(""),
}

```

```

    IL_TEXT(""),
    IL_TEXT(""),
    IL_TEXT(""),
    IL_TEXT(""),
    IL_TEXT(""),
    IL_TEXT(""),
    IL_TEXT(""),
    IL_TEXT(""),
    IL_TEXT(""),
    IL_TEXT("")
}

```

#### 7.142.1.2 ILconst\_string iluLibErrorStringsJapanese[IL\_LIB\_EXR\_ERROR-IL\_LIB\_GIF\_ERROR+1]

**Initial value:**

```

= {
    IL_TEXT("gif"),
    IL_TEXT("jpeg"),
    IL_TEXT("png"),
    IL_TEXT("tiff"),
    IL_TEXT("mng"),
    IL_TEXT("jp2"),
    IL_TEXT("exr")
}

```

#### 7.142.1.3 ILconst\_string iluMiscErrorStringsJapanese[2]

**Initial value:**

```

= {
    IL_TEXT(""),
}

```

### 7.143 src/ILU/ilu\_error/ilu\_err-spanish.h File Reference

```
#include "ilu_internal.h"
```

#### Variables

- ILconst\_string iluErrorStringsSpanish [IL\_FILE\_READ\_ERROR-IL\_INVALID\_ENUM+1]
- ILconst\_string iluLibErrorStringsSpanish [IL\_LIB\_EXR\_ERROR-IL\_LIB\_GIF\_ERROR+1]
- ILconst\_string iluMiscErrorStringsSpanish [2]

#### 7.143.1 Variable Documentation

##### 7.143.1.1 ILconst\_string iluErrorStringsSpanish[IL\_FILE\_READ\_ERROR-IL\_INVALID\_ENUM+1]

**Initial value:**

```

= {
    IL_TEXT("enumerador incorrecto"),
    IL_TEXT("no queda memoria disponible"),
    IL_TEXT("formato no soportado todav\u00eda"),
    IL_TEXT("error interno"),
    IL_TEXT("valor incorrecto"),
    IL_TEXT("operaci\u00f3n ilegal"),
    IL_TEXT("valor de fichero ilegal"),
    IL_TEXT("cabecera incorrecta"),
    IL_TEXT("par\u00e1metro incorrecto"),
}

```

```

    IL_TEXT("no se puede abrir el fichero"),
    IL_TEXT("extensi\u00f3n desconocida"),
    IL_TEXT("el fichero ya existe"),
    IL_TEXT("formato de salida equivalente"),
    IL_TEXT("desbordamiento superior de pila"),
    IL_TEXT("desbordamiento inferior de pila"),
    IL_TEXT("conversi\u00f3n incorrecta"),
    IL_TEXT("\n\u00f3fama de dimensiones incorrecto"),
}

```

#### 7.143.1.2 ILconst\_string iluLibErrorStringsSpanish[IL\_LIB\_EXR\_ERROR-IL\_LIB\_GIF\_ERROR+1]

**Initial value:**

```

= {
    IL_TEXT("error en la librer\u00eda gif"),
    IL_TEXT("error en la librer\u00eda jpeg"),
    IL_TEXT("error en la librer\u00eda png"),
    IL_TEXT("error en la librer\u00eda tiff"),
    IL_TEXT("error en la librer\u00eda mng"),
    IL_TEXT("error en la librer\u00eda jp2"),
    IL_TEXT("error en la librer\u00eda exr"),
}

```

#### 7.143.1.3 ILconst\_string iluMiscErrorStringsSpanish[2]

**Initial value:**

```

= {
    IL_TEXT("no error"),
}

```

## 7.144 src/ILU/ilu\_filter.c File Reference

```

#include "ilu_internal.h"
#include "ilu_filter.h"
#include <math.h>
#include <limits.h>

```

### Functions

- [ILubyte \\* Filter](#) ([ILImage](#) \*Image, const [ILint](#) \*matrix, [ILint](#) scale, [ILint](#) bias)
  - [void iApplyMatrix](#) ([ILImage](#) \*Image, [ILfloat](#) Mat[4][4])
  - [void iIdentity](#) ([ILfloat](#) \*Matrix)
  - [void iIntExtImg](#) ([ILImage](#) \*Image1, [ILImage](#) \*Image2, [ILfloat](#) a)
  - [ILboolean ILAPIENTRY iluAlienify](#) ([void](#))
- Funny as hell filter that I stumbled upon accidentally.*
- [ILboolean ILAPIENTRY iluBlurAvg](#) ([ILuint](#) lter)
  - [ILboolean ILAPIENTRY iluBlurGaussian](#) ([ILuint](#) lter)
  - [ILboolean ILAPIENTRY iluContrast](#) ([ILfloat](#) Contrast)
  - [ILAPI ILboolean ILAPIENTRY iluConvolution](#) ([ILint](#) \*matrix, [ILint](#) scale, [ILint](#) bias)
  - [ILboolean ILAPIENTRY iluEdgeDetectE](#) ([void](#))
  - [ILboolean ILAPIENTRY iluEdgeDetectP](#) ([void](#))
  - [ILboolean ILAPIENTRY iluEdgeDetectS](#) ([void](#))
  - [ILboolean ILAPIENTRY iluEmboss](#) ([void](#))

- [ILboolean ILAPIENTRY iluGammaCorrect](#) (ILfloat Gamma)
- [ILboolean ILAPIENTRY iluPixelize](#) (ILuint PixSize)  
*Pixelizes an image.*
- [ILboolean ILAPIENTRY iluSaturate1f](#) (ILfloat Saturation)
- [ILboolean ILAPIENTRY iluSaturate4f](#) (ILfloat r, ILfloat g, ILfloat b, ILfloat Saturation)
- [ILboolean ILAPIENTRY iluScaleAlpha](#) (ILfloat scale)
- [ILboolean ILAPIENTRY iluScaleColours](#) (ILfloat r, ILfloat g, ILfloat b)  
*Scales image colours.*
- [ILboolean ILAPIENTRY iluSharpen](#) (ILfloat Factor, ILuint Iter)

### 7.144.1 Function Documentation

7.144.1.1 [ILubyte\\* Filter](#) ( IImage \* Image, const ILint \* matrix, ILint scale, ILint bias )

7.144.1.2 [void iApplyMatrix](#) ( IImage \* Image, ILfloat Mat[4][4] )

7.144.1.3 [void ilidentity](#) ( ILfloat \* Matrix )

Matrix++ = 0.0; // row 4 Matrix++ = 0.0; Matrix++ = 0.0; Matrix++ = 1.0;

7.144.1.4 [void ilntExtIlg](#) ( IImage \* Image1, IImage \* Image2, ILfloat a )

7.144.1.5 [ILboolean ILAPIENTRY iluAlienify](#) ( void )

Funny as hell filter that I stumbled upon accidentally.

7.144.1.6 [ILboolean ILAPIENTRY iluBlurAvg](#) ( ILuint Iter )

7.144.1.7 [ILboolean ILAPIENTRY iluBlurGaussian](#) ( ILuint Iter )

7.144.1.8 [ILboolean ILAPIENTRY iluContrast](#) ( ILfloat Contrast )

7.144.1.9 [ILAPI ILboolean ILAPIENTRY iluConvolution](#) ( ILint \* matrix, ILint scale, ILint bias )

7.144.1.10 [ILboolean ILAPIENTRY iluEdgeDetectE](#) ( void )

7.144.1.11 [ILboolean ILAPIENTRY iluEdgeDetectP](#) ( void )

7.144.1.12 [ILboolean ILAPIENTRY iluEdgeDetectS](#) ( void )

7.144.1.13 [ILboolean ILAPIENTRY iluEmboss](#) ( void )

7.144.1.14 [ILboolean ILAPIENTRY iluGammaCorrect](#) ( ILfloat Gamma )

7.144.1.15 [ILboolean ILAPIENTRY iluPixelize](#) ( ILuint PixSize )

Pixelizes an image.

7.144.1.16 [ILboolean ILAPIENTRY iluSaturate1f](#) ( ILfloat Saturation )

7.144.1.17 [ILboolean ILAPIENTRY iluSaturate4f](#) ( ILfloat r, ILfloat g, ILfloat b, ILfloat Saturation )

7.144.1.18 [ILboolean ILAPIENTRY iluScaleAlpha](#) ( ILfloat scale )





7.146.2.6 `ILuint iluScaleAdvanced ( ILuint Width, ILuint Height, ILenum Filter )`

7.146.2.7 `double Lanczos3_filter ( double t )`

7.146.2.8 `double Mitchell_filter ( double t )`

7.146.2.9 `int roundcloser ( double d )`

7.146.2.10 `double sinc ( double x )`

7.146.2.11 `double triangle_filter ( double t )`

7.146.2.12 `int wrap_filter_sample ( int i, int size )`

7.146.2.13 `int zoom ( IImage * dst, IImage * src, double*)(double) filterf, double fwidth )`

## 7.146.3 Variable Documentation

7.146.3.1 `CLIST* contrib`

## 7.147 src/ILU/ilu\_internal.c File Reference

```
#include "ilu_internal.h"
```

### Macros

- `#define ILU_INTERNAL_C`

### Variables

- `const ILdouble IL_DEGCONV = 0.0174532925199432957692`
- `const ILdouble IL_PI = 3.1415926535897932384626`
- `IImage * iluCurlImage = NULL`

## 7.147.1 Macro Definition Documentation

7.147.1.1 `#define ILU_INTERNAL_C`

## 7.147.2 Variable Documentation

7.147.2.1 `const ILdouble IL_DEGCONV = 0.0174532925199432957692`

7.147.2.2 `const ILdouble IL_PI = 3.1415926535897932384626`

7.147.2.3 `IImage* iluCurlImage = NULL`

## 7.148 src/ILU/ilu\_internal.h File Reference

```
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
#include <IL/ilu.h>
#include <IL/devil_internal_exports.h>
```

### Macros

- `#define _IL_BUILD_LIBRARY`
- `#define _IL_BUILD_LIBRARY`
- `#define _ILU_BUILD_LIBRARY`
- `#define _ILU_BUILD_LIBRARY`
- `#define imemclear(x, y) memset(x,0,y);`

### Functions

- `INLINE ILfloat ilCos (ILfloat Angle)`
- `INLINE ILint ilRound (ILfloat Num)`
- `INLINE ILfloat ilSin (ILfloat Angle)`
- `ILuint iluScaleAdvanced (ILuint Width, ILuint Height, ILenum Filter)`
- `ILubyte * iScanFill (void)`

### Variables

- `const ILdouble IL_DEGCONV`
- `const ILdouble IL_PI`
- `ILimage * iluCurlImage`

#### 7.148.1 Macro Definition Documentation

7.148.1.1 `#define _IL_BUILD_LIBRARY`

7.148.1.2 `#define _IL_BUILD_LIBRARY`

7.148.1.3 `#define _ILU_BUILD_LIBRARY`

7.148.1.4 `#define _ILU_BUILD_LIBRARY`

7.148.1.5 `#define imemclear( x, y ) memset(x,0,y);`

#### 7.148.2 Function Documentation

7.148.2.1 `INLINE ILfloat ilCos ( ILfloat Angle )`

7.148.2.2 `INLINE ILint ilRound ( ILfloat Num )`

7.148.2.3 `INLINE ILfloat ilSin ( ILfloat Angle )`

7.148.2.4 `ILuint iluScaleAdvanced ( ILuint Width, ILuint Height, ILenum Filter )`



7.148.2.5 `ILubyte* iScanFill ( void )`

### 7.148.3 Variable Documentation

7.148.3.1 `const ILdouble IL_DEGCONV`

7.148.3.2 `const ILdouble IL_PI`

7.148.3.3 `ILImage* iluCurlImage`

## 7.149 src/ILU/ilu\_main.c File Reference

```
#include "ilu_internal.h"
#include "ilu_states.h"
```

### Functions

- `void ILAPIENTRY iluInit ()`
- `ILuint ILAPIENTRY iluLoadImage (ILconst_string FileName)`

### 7.149.1 Function Documentation

7.149.1.1 `void ILAPIENTRY iluInit ( void )`

7.149.1.2 `ILuint ILAPIENTRY iluLoadImage ( ILconst_string FileName )`

## 7.150 src/ILU/ilu\_manip.c File Reference

```
#include "ilu_internal.h"
#include "ilu_states.h"
#include <float.h>
#include <limits.h>
```

### Data Structures

- struct `BUCKET`

### Typedefs

- typedef struct `BUCKET BUCKET`

### Functions

- `ILuint ILAPIENTRY iluColoursUsed ()`
- `ILboolean ILAPIENTRY iluCompareImage (ILuint Comp)`
- `ILboolean ILAPIENTRY iluCrop (ILuint XOff, ILuint YOff, ILuint ZOff, ILuint Width, ILuint Height, ILuint Depth)`
- `ILboolean iluCrop2D (ILuint XOff, ILuint YOff, ILuint Width, ILuint Height)`
- `ILboolean iluCrop3D (ILuint XOff, ILuint YOff, ILuint ZOff, ILuint Width, ILuint Height, ILuint Depth)`
- `ILboolean ILAPIENTRY iluEnlargeCanvas (ILuint Width, ILuint Height, ILuint Depth)`

*Enlarges the canvas.*

- `ILboolean ILAPIENTRY iluEqualize ( )`
- `ILboolean ILAPIENTRY iluFlipImage ( )`

*Flips an image over its x axis.*

- `ILboolean ILAPIENTRY iluInvertAlpha ( )`

*Inverts the alpha in the image.*

- `ILboolean ILAPIENTRY iluMirror ( )`

*Mirrors an image over its y axis.*

- `ILboolean ILAPIENTRY iluNegative ( )`

*Inverts the colours in the image.*

- `ILboolean ILAPIENTRY iluReplaceColour (ILubyte Red, ILubyte Green, ILubyte Blue, ILfloat Tolerance)`
- `ILboolean ILAPIENTRY iluSwapColours ( )`
- `ILboolean ILAPIENTRY iluWave (ILfloat Angle)`

## 7.150.1 Typedef Documentation

### 7.150.1.1 typedef struct BUCKET BUCKET

## 7.150.2 Function Documentation

### 7.150.2.1 ILuint ILAPIENTRY iluColoursUsed ( void )

### 7.150.2.2 ILboolean ILAPIENTRY iluCompareImage ( ILuint Comp )

### 7.150.2.3 ILboolean ILAPIENTRY iluCrop ( ILuint XOff, ILuint YOff, ILuint ZOff, ILuint Width, ILuint Height, ILuint Depth )

### 7.150.2.4 ILboolean iluCrop2D ( ILuint XOff, ILuint YOff, ILuint Width, ILuint Height )

### 7.150.2.5 ILboolean iluCrop3D ( ILuint XOff, ILuint YOff, ILuint ZOff, ILuint Width, ILuint Height, ILuint Depth )

### 7.150.2.6 ILboolean ILAPIENTRY iluEnlargeCanvas ( ILuint Width, ILuint Height, ILuint Depth )

Enlarges the canvas.

### 7.150.2.7 ILboolean ILAPIENTRY iluEqualize ( void )

### 7.150.2.8 ILboolean ILAPIENTRY iluFlipImage ( void )

Flips an image over its x axis.

### 7.150.2.9 ILboolean ILAPIENTRY iluInvertAlpha ( void )

Inverts the alpha in the image.

### 7.150.2.10 ILboolean ILAPIENTRY iluMirror ( void )

Mirrors an image over its y axis.

### 7.150.2.11 ILboolean ILAPIENTRY iluNegative ( void )

Inverts the colours in the image.

7.150.2.12 ILboolean ILAPIENTRY iluReplaceColour ( ILubyte *Red*, ILubyte *Green*, ILubyte *Blue*, ILfloat *Tolerance* )

7.150.2.13 ILboolean ILAPIENTRY iluSwapColours ( void )

7.150.2.14 ILboolean ILAPIENTRY iluWave ( ILfloat *Angle* )

## 7.151 src/ILU/ilu\_mipmap.c File Reference

```
#include "ilu_internal.h"
```

### Functions

- ILboolean iBuildMipmaps (ILImage \*Parent, ILuint Width, ILuint Height, ILuint Depth)
- ILboolean ILAPIENTRY iluBuildMipmaps ()

#### 7.151.1 Function Documentation

7.151.1.1 ILboolean iBuildMipmaps ( ILImage \* *Parent*, ILuint *Width*, ILuint *Height*, ILuint *Depth* )

7.151.1.2 ILboolean ILAPIENTRY iluBuildMipmaps ( void )

## 7.152 src/ILU/ilu\_noise.c File Reference

```
#include "ilu_internal.h"
#include <math.h>
#include <limits.h>
```

### Functions

- ILboolean ILAPIENTRY iluNoisify (ILclampf *Tolerance*)

#### 7.152.1 Function Documentation

7.152.1.1 ILboolean ILAPIENTRY iluNoisify ( ILclampf *Tolerance* )

## 7.153 src/ILU/ilu\_region.c File Reference

```
#include "ilu_internal.h"
#include "ilu_region.h"
```

### Macros

- #define iRegionSetPixel(x, y) (iRegionMask[y \* iluCurlImage->Width + x] = 1 )

## Functions

- `void BuildActiveList (ILint scan, Edge *active, Edge *edges[])`
- `void BuildEdgeList (ILuint cnt, ILpointi *pts, Edge **edges)`
- `void DeleteAfter (Edge *q)`
- `void FillScan (ILint scan, Edge *active)`
- `void ILAPIENTRY iluRegionfv (ILpointf *Points, ILuint n)`
- `void ILAPIENTRY iluRegioniv (ILpointi *Points, ILuint n)`
- `void InsertEdge (Edge *list, Edge *edge)`
- `ILubyte * iScanFill ()`
- `void MakeEdgeRec (ILpointi lower, ILpointi upper, ILint yComp, Edge *edge, Edge *edges[])`
- `void ResortActiveList (Edge *active)`
- `void UpdateActiveList (ILint scan, Edge *active)`
- `ILint yNext (ILint k, ILint cnt, ILpointi *pts)`

## Variables

- `ILubyte * iRegionMask = NULL`
- `ILuint PointNum = 0`
- `ILpointf * RegionPointsf = NULL`
- `ILpointi * RegionPointsi = NULL`

### 7.153.1 Macro Definition Documentation

7.153.1.1 `#define iRegionSetPixel( x, y ) (iRegionMask[y * iluCurlImage->Width + x] = 1 )`

### 7.153.2 Function Documentation

7.153.2.1 `void BuildActiveList ( ILint scan, Edge * active, Edge * edges[] )`

7.153.2.2 `void BuildEdgeList ( ILuint cnt, ILpointi * pts, Edge ** edges )`

7.153.2.3 `void DeleteAfter ( Edge * q )`

7.153.2.4 `void FillScan ( ILint scan, Edge * active )`

7.153.2.5 `void ILAPIENTRY iluRegionfv ( ILpointf * Points, ILuint n )`

7.153.2.6 `void ILAPIENTRY iluRegioniv ( ILpointi * Points, ILuint n )`

7.153.2.7 `void InsertEdge ( Edge * list, Edge * edge )`

7.153.2.8 `ILubyte* iScanFill ( void )`

7.153.2.9 `void MakeEdgeRec ( ILpointi lower, ILpointi upper, ILint yComp, Edge * edge, Edge * edges[] )`

7.153.2.10 `void ResortActiveList ( Edge * active )`

7.153.2.11 `void UpdateActiveList ( ILint scan, Edge * active )`

7.153.2.12 `ILint yNext ( ILint k, ILint cnt, ILpointi * pts )`

### 7.153.3 Variable Documentation

7.153.3.1 ILubyte\* iRegionMask = NULL

7.153.3.2 ILuint PointNum = 0

7.153.3.3 ILpointf\* RegionPointsf = NULL

7.153.3.4 ILpointi\* RegionPointsi = NULL

## 7.154 src/ILU/ilu\_region.h File Reference

```
#include "ilu_internal.h"
```

### Data Structures

- struct [Edge](#)

### Typedefs

- typedef struct [Edge](#) Edge

#### 7.154.1 Typedef Documentation

7.154.1.1 typedef struct Edge Edge

## 7.155 include/IL/ilu\_region.h File Reference

### Data Structures

- struct [Edge](#)

### Typedefs

- typedef struct [Edge](#) Edge

#### 7.155.1 Typedef Documentation

7.155.1.1 typedef struct Edge Edge

## 7.156 src/ILU/ilu\_rotate.c File Reference

```
#include "ilu_internal.h"
#include "ilu_states.h"
```

### Functions

- ILboolean ILAPIENTRY iluRotate (ILfloat Angle)
- ILboolean ILAPIENTRY iluRotate3D (ILfloat x, ILfloat y, ILfloat z, ILfloat Angle)
- ILAPI ILimage \*ILAPIENTRY iluRotate3D\_ (ILimage \*Image, ILfloat x, ILfloat y, ILfloat z, ILfloat Angle)

- [ILAPI IImage \\*ILAPIENTRY iluRotate\\_ \(IImage \\*Image, ILfloat Angle\)](#)

*Rotates a bitmap any angle.*

### 7.156.1 Function Documentation

7.156.1.1 [ILboolean ILAPIENTRY iluRotate \( ILfloat Angle \)](#)

7.156.1.2 [ILboolean ILAPIENTRY iluRotate3D \( ILfloat x, ILfloat y, ILfloat z, ILfloat Angle \)](#)

7.156.1.3 [ILAPI IImage\\* ILAPIENTRY iluRotate3D\\_ \( IImage \\* Image, ILfloat x, ILfloat y, ILfloat z, ILfloat Angle \)](#)

7.156.1.4 [ILAPI IImage\\* ILAPIENTRY iluRotate\\_ \( IImage \\* Image, ILfloat Angle \)](#)

Rotates a bitmap any angle.

## 7.157 src/ILU/ilu\_scale.c File Reference

```
#include "ilu_internal.h"
#include "ilu_states.h"
```

### Functions

- [ILboolean ILAPIENTRY iluEnlargeImage \(ILfloat XDim, ILfloat YDim, ILfloat ZDim\)](#)
- [ILboolean ILAPIENTRY iluScale \(ILuint Width, ILuint Height, ILuint Depth\)](#)
- [IImage \\* iluScale1D\\_ \(IImage \\*Image, IImage \\*Scaled, ILuint Width\)](#)
- [IImage \\* iluScale2D\\_ \(IImage \\*Image, IImage \\*Scaled, ILuint Width, ILuint Height\)](#)
- [IImage \\* iluScale3D\\_ \(IImage \\*Image, IImage \\*Scaled, ILuint Width, ILuint Height, ILuint Depth\)](#)
- [ILAPI IImage \\*ILAPIENTRY iluScale\\_ \(IImage \\*Image, ILuint Width, ILuint Height, ILuint Depth\)](#)

### 7.157.1 Function Documentation

7.157.1.1 [ILboolean ILAPIENTRY iluEnlargeImage \( ILfloat XDim, ILfloat YDim, ILfloat ZDim \)](#)

7.157.1.2 [ILboolean ILAPIENTRY iluScale \( ILuint Width, ILuint Height, ILuint Depth \)](#)

7.157.1.3 [IImage \\* iluScale1D\\_ \( IImage \\* Image, IImage \\* Scaled, ILuint Width \)](#)

7.157.1.4 [IImage\\* iluScale2D\\_ \( IImage \\* Image, IImage \\* Scaled, ILuint Width, ILuint Height \)](#)

7.157.1.5 [IImage\\* iluScale3D\\_ \( IImage \\* Image, IImage \\* Scaled, ILuint Width, ILuint Height, ILuint Depth \)](#)

7.157.1.6 [ILAPI IImage\\* ILAPIENTRY iluScale\\_ \( IImage \\* Image, ILuint Width, ILuint Height, ILuint Depth \)](#)

## 7.158 src/ILU/ilu\_scale2d.c File Reference

```
#include "ilu_internal.h"
#include "ilu_states.h"
```

## Functions

- [ILImage](#) \* [iluScale2D\\_](#) ([ILImage](#) \*Image, [ILImage](#) \*Scaled, [ILuint](#) Width, [ILuint](#) Height)
- [ILImage](#) \* [iluScale2DBilinear\\_](#) ([ILImage](#) \*Image, [ILImage](#) \*Scaled, [ILuint](#) Width, [ILuint](#) Height)
- [ILImage](#) \* [iluScale2DLinear\\_](#) ([ILImage](#) \*Image, [ILImage](#) \*Scaled, [ILuint](#) Width, [ILuint](#) Height)
- [ILImage](#) \* [iluScale2DNear\\_](#) ([ILImage](#) \*Image, [ILImage](#) \*Scaled, [ILuint](#) Width, [ILuint](#) Height)

### 7.158.1 Function Documentation

7.158.1.1 [ILImage](#)\* [iluScale2D\\_](#) ( [ILImage](#) \* *Image*, [ILImage](#) \* *Scaled*, [ILuint](#) *Width*, [ILuint](#) *Height* )

7.158.1.2 [ILImage](#) \* [iluScale2DBilinear\\_](#) ( [ILImage](#) \* *Image*, [ILImage](#) \* *Scaled*, [ILuint](#) *Width*, [ILuint](#) *Height* )

7.158.1.3 [ILImage](#) \* [iluScale2DLinear\\_](#) ( [ILImage](#) \* *Image*, [ILImage](#) \* *Scaled*, [ILuint](#) *Width*, [ILuint](#) *Height* )

7.158.1.4 [ILImage](#) \* [iluScale2DNear\\_](#) ( [ILImage](#) \* *Image*, [ILImage](#) \* *Scaled*, [ILuint](#) *Width*, [ILuint](#) *Height* )

## 7.159 src/ILU/ilu\_scale3d.c File Reference

```
#include "ilu_internal.h"
#include "ilu_states.h"
```

## Functions

- [ILImage](#) \* [iluScale3D\\_](#) ([ILImage](#) \*Image, [ILImage](#) \*Scaled, [ILuint](#) Width, [ILuint](#) Height, [ILuint](#) Depth)
- [ILImage](#) \* [iluScale3DBilinear\\_](#) ([ILImage](#) \*Image, [ILImage](#) \*Scaled, [ILuint](#) Width, [ILuint](#) Height, [ILuint](#) Depth)
- [ILImage](#) \* [iluScale3DLinear\\_](#) ([ILImage](#) \*Image, [ILImage](#) \*Scaled, [ILuint](#) Width, [ILuint](#) Height, [ILuint](#) Depth)
- [ILImage](#) \* [iluScale3DNear\\_](#) ([ILImage](#) \*Image, [ILImage](#) \*Scaled, [ILuint](#) Width, [ILuint](#) Height, [ILuint](#) Depth)

### 7.159.1 Function Documentation

7.159.1.1 [ILImage](#)\* [iluScale3D\\_](#) ( [ILImage](#) \* *Image*, [ILImage](#) \* *Scaled*, [ILuint](#) *Width*, [ILuint](#) *Height*, [ILuint](#) *Depth* )

7.159.1.2 [ILImage](#)\* [iluScale3DBilinear\\_](#) ( [ILImage](#) \* *Image*, [ILImage](#) \* *Scaled*, [ILuint](#) *Width*, [ILuint](#) *Height*, [ILuint](#) *Depth* )

7.159.1.3 [ILImage](#) \* [iluScale3DLinear\\_](#) ( [ILImage](#) \* *Image*, [ILImage](#) \* *Scaled*, [ILuint](#) *Width*, [ILuint](#) *Height*, [ILuint](#) *Depth* )

7.159.1.4 [ILImage](#) \* [iluScale3DNear\\_](#) ( [ILImage](#) \* *Image*, [ILImage](#) \* *Scaled*, [ILuint](#) *Width*, [ILuint](#) *Height*, [ILuint](#) *Depth* )

## 7.160 src/ILU/ilu\_scaling.c File Reference

## 7.161 src/ILU/ilu\_states.c File Reference

```
#include "ilu_internal.h"
#include "ilu_states.h"
```

## Functions

- `ILint ILAPIENTRY iluGetInteger (ILenum Mode)`
- `void ILAPIENTRY iluGetInterv (ILenum Mode, ILint *Param)`
- `ILstring ILAPIENTRY iluGetString (ILenum StringName)`
- `void ILAPIENTRY iluImageParameter (ILenum PName, ILenum Param)`

## Variables

- `ILconst_string _iluVendor = IL_TEXT("kolrabi")`
- `ILconst_string _iluVersion = IL_TEXT("kolrabi's another Image Library Utilities (ILU) 1.8.3")`
- `ILenum iluFilter = ILU_NEAREST`
- `ILenum iluPlacement = ILU_CENTER`

### 7.161.1 Function Documentation

7.161.1.1 `ILint ILAPIENTRY iluGetInteger ( ILenum Mode )`

7.161.1.2 `void ILAPIENTRY iluGetInterv ( ILenum Mode, ILint * Param )`

7.161.1.3 `ILstring ILAPIENTRY iluGetString ( ILenum StringName )`

7.161.1.4 `void ILAPIENTRY iluImageParameter ( ILenum PName, ILenum Param )`

### 7.161.2 Variable Documentation

7.161.2.1 `ILconst_string _iluVendor = IL_TEXT("kolrabi")`

7.161.2.2 `ILconst_string _iluVersion = IL_TEXT("kolrabi's another Image Library Utilities (ILU) 1.8.3")`

7.161.2.3 `ILenum iluFilter = ILU_NEAREST`

7.161.2.4 `ILenum iluPlacement = ILU_CENTER`

## 7.162 src/ILU/ilu\_states.h File Reference

### Variables

- `ILenum iluFilter`
- `ILenum iluPlacement`

### 7.162.1 Variable Documentation

7.162.1.1 `ILenum iluFilter`

7.162.1.2 `ILenum iluPlacement`

## 7.163 src/ILU/ilu\_utilities.c File Reference

```
#include "ilu_internal.h"
```



## Functions

- `void ILAPIENTRY iluDeleteImage (ILuint Id)`
- `ILuint ILAPIENTRY iluGenImage ()`
- `void ILAPIENTRY iluGetImageInfo (ILinfo *Info)`  
*Retrieves information about the current bound image.*

### 7.163.1 Function Documentation

7.163.1.1 `void ILAPIENTRY iluDeleteImage ( ILuint Id )`

7.163.1.2 `ILuint ILAPIENTRY iluGenImage ( )`

7.163.1.3 `void ILAPIENTRY iluGetImageInfo ( ILinfo * Info )`

Retrieves information about the current bound image.

## 7.164 src/ILUT/ilut\_allegro.cc File Reference

```
#include "ilut_internal.h"
```

## 7.165 src/ILUT/ilut\_allegro.h File Reference

## 7.166 src/ILUT/ilut\_directx.c File Reference

```
#include "ilut_internal.h"
```

## 7.167 src/ILUT/ilut\_directx9.c File Reference

```
#include "ilut_internal.h"
```

## 7.168 src/ILUT/ilut\_internal.c File Reference

```
#include "ilut_internal.h"
```

## 7.169 src/ILUT/ilut\_internal.h File Reference

```
#include <IL/ilut.h>
#include <IL/devil_internal_exports.h>
#include <stdlib.h>
#include <string.h>
```

## Macros

- `#define _IL_BUILD_LIBRARY`
- `#define _ILU_BUILD_LIBRARY`
- `#define _ILUT_BUILD_LIBRARY`
- `#define CUBEMAP_SIDES 6`
- `#define SAFE_RELEASE(p) {if((p)!=NULL){(p)->lpVtbl->Release(p);(p)=NULL;}}`

## Functions

- `void ilutDefaultStates (void)`

### 7.169.1 Macro Definition Documentation

7.169.1.1 `#define _IL_BUILD_LIBRARY`

7.169.1.2 `#define _ILU_BUILD_LIBRARY`

7.169.1.3 `#define _ILUT_BUILD_LIBRARY`

7.169.1.4 `#define CUBEMAP_SIDES 6`

7.169.1.5 `#define SAFE_RELEASE( p ) {if((p)!=NULL){(p)->lpVtbl->Release(p);(p)=NULL;}}`

### 7.169.2 Function Documentation

7.169.2.1 `void ilutDefaultStates ( void )`

## 7.170 src/ILUT/ilut\_main.c File Reference

```
#include "ilut_internal.h"
```

## Functions

- `void ILAPIENTRY ilutInit ()`

### 7.170.1 Function Documentation

7.170.1.1 `void ILAPIENTRY ilutInit ( void )`

## 7.171 src/ILUT/ilut\_opengl.c File Reference

```
#include "ilut_opengl.h"
```

## 7.172 src/ILUT/ilut\_opengl.h File Reference

```
#include "ilut_internal.h"
```

## 7.173 src/ILUT/ilut\_sdlsurface.c File Reference

```
#include "ilut_internal.h"
```

## 7.174 src/ILUT/ilut\_states.c File Reference

```
#include "ilut_internal.h"
#include "ilut_states.h"
#include "ilut_opengl.h"
```

### Functions

- [ILboolean ilutAble](#) ([ILenum Mode](#), [ILboolean Flag](#))
- [void ILAPIENTRY ilutD3D8MipFunc](#) ([ILuint NumLevels](#))
- [void ilutDefaultStates](#) ()
- [ILboolean ILAPIENTRY ilutDisable](#) ([ILenum Mode](#))
- [ILboolean ILAPIENTRY ilutEnable](#) ([ILenum Mode](#))
- [ILboolean ILAPIENTRY ilutGetBoolean](#) ([ILenum Mode](#))
- [void ILAPIENTRY ilutGetBooleanv](#) ([ILenum Mode](#), [ILboolean \\*Param](#))
- [ILint ILAPIENTRY ilutGetInteger](#) ([ILenum Mode](#))
- [void ILAPIENTRY ilutGetIntegerv](#) ([ILenum Mode](#), [ILint \\*Param](#))
- [ILstring ILAPIENTRY ilutGetString](#) ([ILenum StringName](#))
- [ILboolean ILAPIENTRY ilutIsDisabled](#) ([ILenum Mode](#))
- [ILboolean ILAPIENTRY ilutIsEnabled](#) ([ILenum Mode](#))
- [void ILAPIENTRY ilutPopAttrib](#) ()
- [void ILAPIENTRY ilutPushAttrib](#) ([ILuint Bits](#))
- [ILboolean ILAPIENTRY ilutRenderer](#) ([ILenum Renderer](#))
- [void ILAPIENTRY ilutSetInteger](#) ([ILenum Mode](#), [ILint Param](#))

### Variables

- [ILconst\\_string \\_ilutVendor](#) = [IL\\_TEXT](#)("kolrabi")
- [ILconst\\_string \\_ilutVersion](#) = [IL\\_TEXT](#)("kolrabi's another Image Library Utility Toolkit (ILUT) 1.8.3")

### 7.174.1 Function Documentation

7.174.1.1 [ILboolean ilutAble](#) ( [ILenum Mode](#), [ILboolean Flag](#) )

7.174.1.2 [void ILAPIENTRY ilutD3D8MipFunc](#) ( [ILuint NumLevels](#) )

7.174.1.3 [void ilutDefaultStates](#) ( [void](#) )

7.174.1.4 [ILboolean ILAPIENTRY ilutDisable](#) ( [ILenum Mode](#) )

7.174.1.5 [ILboolean ILAPIENTRY ilutEnable](#) ( [ILenum Mode](#) )

7.174.1.6 [ILboolean ILAPIENTRY ilutGetBoolean](#) ( [ILenum Mode](#) )

7.174.1.7 [void ILAPIENTRY ilutGetBooleanv](#) ( [ILenum Mode](#), [ILboolean \\* Param](#) )

- 7.174.1.8 **ILint** ILAPIENTRY ilutGetInteger ( *ILenum Mode* )
- 7.174.1.9 **void** ILAPIENTRY ilutGetIntegerv ( *ILenum Mode*, *ILint \* Param* )
- 7.174.1.10 **ILstring** ILAPIENTRY ilutGetString ( *ILenum StringName* )
- 7.174.1.11 **ILboolean** ILAPIENTRY ilutIsDisabled ( *ILenum Mode* )
- 7.174.1.12 **ILboolean** ILAPIENTRY ilutIsEnabled ( *ILenum Mode* )
- 7.174.1.13 **void** ILAPIENTRY ilutPopAttrib ( **void** )
- 7.174.1.14 **void** ILAPIENTRY ilutPushAttrib ( *ILuint Bits* )
- 7.174.1.15 **ILboolean** ILAPIENTRY ilutRenderer ( *ILenum Renderer* )
- 7.174.1.16 **void** ILAPIENTRY ilutSetInteger ( *ILenum Mode*, *ILint Param* )

## 7.174.2 Variable Documentation

- 7.174.2.1 **ILconst\_string** \_ilutVendor = IL\_TEXT("kolrabi")
- 7.174.2.2 **ILconst\_string** \_ilutVersion = IL\_TEXT("kolrabi's another Image Library Utility Toolkit (ILUT) 1.8.3")

## 7.175 src/ILUT/ilut\_states.h File Reference

```
#include "ilut_internal.h"
```

### Data Structures

- struct [ILUT\\_STATES](#)

### Macros

- **#define** [ILUT\\_ATTRIB\\_STACK\\_MAX](#) 32

### Typedefs

- typedef struct [ILUT\\_STATES](#) [ILUT\\_STATES](#)

### Functions

- **ILboolean** [ilutAble](#) (*ILenum Mode*, *ILboolean Flag*)

### Variables

- **ILuint** [ilutCurrentPos](#) = 0
- **ILUT\_STATES** [ilutStates](#) [[ILUT\\_ATTRIB\\_STACK\\_MAX](#)]

### 7.175.1 Macro Definition Documentation

7.175.1.1 `#define ILUT_ATTRIB_STACK_MAX 32`

### 7.175.2 Typedef Documentation

7.175.2.1 `typedef struct ILUT_STATES ILUT_STATES`

### 7.175.3 Function Documentation

7.175.3.1 `ILboolean ilutAble ( ILenum Mode, ILboolean Flag )`

### 7.175.4 Variable Documentation

7.175.4.1 `ILuint ilutCurrentPos = 0`

7.175.4.2 `ILUT_STATES ilutStates[ILUT_ATTRIB_STACK_MAX]`

## 7.176 src/ILUT/ilut\_win32.c File Reference

```
#include "ilut_internal.h"
```

## 7.177 src/ILUT/ilut\_x11.c File Reference

```
#include "ilut_internal.h"
```

## 7.178 src/test/iltest-format-load.c File Reference

```
#include <IL/devil_internal_exports.h>
#include "IL/il_endian.h"
#include <stdlib.h>
#include <string.h>
```

### Macros

- `#define CHECK(x) if (!(x)) { fprintf(stderr, "FAILED in line %d: %s\n", __LINE__, #x); return 1; }`
- `#define CHECK_EQ(x, y) if ((x) != (y)) { fprintf(stderr, "FAILED in line %d: %s (%d) == %s (%d)\n", __LINE__, #x, x, #y, y); return 1; }`

### Functions

- `int main (int argc, char **argv)`

### 7.178.1 Macro Definition Documentation

7.178.1.1 `#define CHECK( x ) if (!(x)) { fprintf(stderr, "FAILED in line %d: %s\n", __LINE__, #x); return 1; }`

7.178.1.2 `#define CHECK_EQ( x, y ) if ((x)!=y)) { fprintf(stderr, "FAILED in line %d: %s (%d) == %s (%d)\n", __LINE__, #x, x, #y, y); return 1; }`

## 7.178.2 Function Documentation

7.178.2.1 `int main ( int argc, char ** argv )`

## 7.179 src/test/iltest-io.c File Reference

```
#include <IL/devil_internal_exports.h>
#include "IL/il_endian.h"
#include "IL/il_internal.h"
#include <stdlib.h>
#include <string.h>
```

### Macros

- `#define CHECK(x) if (!(x)) { fprintf(stderr, "FAILED in line %d: %s\n", __LINE__, #x); return 1; }`
- `#define CHECK_EQ(x, y) if ((x)!=y)) { fprintf(stderr, "FAILED in line %d: %s (%d) == %s (%d)\n", __LINE__, #x, x, #y, y); return 1; }`
- `#define TEST(n) if (!strcmp(*argv, #n)) result = test_##n(argv+1); else`

### Functions

- `int main (int argc, char **argv)`

## 7.179.1 Macro Definition Documentation

7.179.1.1 `#define CHECK( x ) if (!(x)) { fprintf(stderr, "FAILED in line %d: %s\n", __LINE__, #x); return 1; }`

7.179.1.2 `#define CHECK_EQ( x, y ) if ((x)!=y)) { fprintf(stderr, "FAILED in line %d: %s (%d) == %s (%d)\n", __LINE__, #x, x, #y, y); return 1; }`

7.179.1.3 `#define TEST( n ) if (!strcmp(*argv, #n)) result = test_##n(argv+1); else`

## 7.179.2 Function Documentation

7.179.2.1 `int main ( int argc, char ** argv )`

## 7.180 src/test/iltest-memory.c File Reference

```
#include <IL/devil_internal_exports.h>
#include "IL/il_endian.h"
#include <stdlib.h>
#include <string.h>
```

### Macros

- `#define CHECK(x) if (!(x)) { fprintf(stderr, "FAILED in line %d: %s\n", __LINE__, #x); return 1; }`
- `#define TEST(n) if (!strcmp(*argv, #n)) result = test_##n(); else`

## Functions

- int [main](#) (int argc, char \*\*argv)
- int [test\\_endian](#) ()

### 7.180.1 Macro Definition Documentation

7.180.1.1 `#define CHECK( x ) if (!(x)) { fprintf(stderr, "FAILED in line %d: %s\n", __LINE__, #x); return 1; }`

7.180.1.2 `#define TEST( n ) if (!strcmp(*argv, #n)) result = test_##n(); else`

### 7.180.2 Function Documentation

7.180.2.1 int main ( int *argc*, char \*\* *argv* )

7.180.2.2 int test\_endian ( )