NAME

XpmWrite - write an XPM file

SYNOPSIS

int XpmWriteFileFromPixmap(Display ***_display_, char ***_filename_**,**
Pixmap _pixmap_**, Pixmap** _shapemask_**, XpmAttributes ***_attributes_**);**

int XpmWriteFileFromImage(Display ***_display_ **, char ***_filename_,**
XImage ***_image_, XImage ***_shapeimage_**, XpmAttributes ***_attributes_**);**

int XpmWriteFileFromData(char ***_filename_, char ****_data_**);**

int XpmWriteFileFromXpmImage(char ***_filename_, XpmImage ***_image_**,**
XImage ***_shapeimage_, XpmInfo ***_info_**);**

int XpmWriteFileFromBuffer(char ***_filename_, char ***_buffer_**);**

ARGUMENTS

display
Specifies the connection to the X server.

filename
Specifies the file name to use.

pixmap
Specifies the pixmap.

shapemask
Specifies the shape mask pixmap.

attributes
Specifies the location of a structure containing information (or NULL).

data
Specifies the data array to read.

image

Specifies the image.

info
Specifies the location of a structure to get information from (or NULL).

buffer
Specifies the buffer to read.

DESCRIPTION

XpmWriteFileFromImage

The **XpmWriteFileFromImage**() function writes an *image and its possible shapeimage* out to a file in the XPM format. If the file cannot be opened, it returns **XpmOpenFailed**. If insufficient working storage is allocated, it returns **XpmNoMemory**. If no error occurs then it returns **XpmSuccess**. If the passed XpmAttributes structure pointer is not NULL, **XpmWriteFileFromImage**() looks for the following attributes: XpmColormap, XpmHotspot, XpmCharsPerPixel, XpmRgbFilename, and XpmExtensions. As a backward compatibility feature, **XpmWriteFileFromImage**() also looks for the XpmInfos attributes. If the filename contains an extension such as ".xpm", in order to get a valid C variable name, the dot character is replaced by an underscore '_' when writing out. As a backward compatibility feature, if the XpmInfos attributes are defined it writes out possible stored information such as comments, color defaults and symbol. Finally, if the XpmRgbFilenameattribute is defined, **XpmWriteFileFromImage**() searches for color names in this file and if found writes them out instead of the rgb values.

In addition on systems which support such features if the given file name ends by '.Z' or '.gz' it is assumed to be a compressed file. Then, **XpmWriteFileFromImage**() writes to a piped compress or gzip process. And if instead of a file name, NULL is passed to **XpmWriteFileFromImage**(), it writes to the standard output.

XpmWriteFileFromPixmap

To write out a Pixmap to an XPM file, use **XpmWriteFileFromPixmap**().

If the passed XpmAttributes structure pointer is not NULL, **XpmWriteFileFromPixmap**() looks for the following attributes: XpmSize. If they are not defined it performs an XGetGeometry operation. Then it uses XGetImage to get from the given pixmaps the related X images which are passed to **XpmWriteFileFromImage**(). Finally **XpmWriteFileFromPixmap**() destroys the created images using XDestroyImage. The **XpmWriteFileFromPixmap**() function returns the same errors as **XpmWriteFileFromImage**().

XpmWriteFileFromData

XpmWriteFileFromData() writes an XPM data array to an XPM file.

XpmWriteFileToData() returns **XpmOpenFailed** if it cannot open the file, **XpmFileInvalid** if this is not a valid XPM data, and **XpmSuccess** otherwise.

XpmWriteFileFromXpmImage

To write out an XpmImage to an XPM file, use **XpmWriteFileFromXpmImage**(). The **XpmWriteFileFromXpmImage**() function writes an image out to a file in the XPM format. If the file cannot be opened, it returns **XpmOpenFailed**. If insufficient working storage is allocated, it returns **XpmNoMemory**. If no error occurs then it returns **XpmSuccess**. If the passed XpmInfo structure pointer is not NULL, **XpmWriteFileFromXpmImage()** looks for the following attributes: XpmComments, XpmExtensions, and XpmHotspot, and writes the related information out as well. In addition on systems which support such features if the given file name ends by '.Z' or '.gz' it is assumed to be a compressed file. Then, **XpmWriteFileFromXpmImage**() writes to a piped compress or gzip process. And if instead of a file name, NULL is passed to **XpmWriteFileFromXpmImage**(), it writes to the standard output.

XpmWriteFileFromBuffer

XpmWriteFileFromBuffer() writes a XPM buffer to a file. **XpmWriteFileFromBuffer**() returns **XpmOpenFailed** if it cannot open the file, and **XpmSuccess** otherwise.

As a convenience, the **XpmReadFileToBuffer**() and **XpmWriteFileFromBuffer**() functions are provided to copy a file to a buffer and to write a file from a buffer. Thus for instance one may decide to use **XpmCreateBufferFromPixmap**(), **XpmWriteFileFromBuffer**(), and **XpmFree**() instead of **XpmWriteFileFromPixmap**(). On some systems this may lead to a performance improvement, since the parsing will be performed in memory, but it uses more memory.

SEE ALSO

XpmRead(3)