NAME

XpmMisc - xpm misc functions to free used memory and handle errors

SYNOPSIS

int XpmFreeXpmImage(XpmImage ****image*);**

int XpmFreeXpmInfo(XpmInfo ****info*);**

int XpmFreeAttributes(XpmAttributes ****attributes*);**

int XpmAttributesSize(void);

int XpmFreeExtensions(XpmExtension ****extensions*, int** *nextensions***);**

int XpmFree(char ****ptr*);**

char *XpmGetErrorString(int *errorcode***);**

int XpmLibraryVersion(void);

ARGUMENTS

image
Specifies the structure to free

info
Specifies the structure to free

ptr Specifies the data to free

errorcode
Specifies the XPM error

extensions
Specifies the array to free

nextensions
Specifies the number of extensions

attributes

Specifies the structure to free

DESCRIPTION

To free possible data stored into an XpmImage structure use **XpmFreeXpmImage**(). The **XpmFreeXpmImage**() function frees the structure members which are not NULL, but not the structure itself.

To free possible data stored into an XpmInfo structure use **XpmFreeXpmInfo().**

To free data possibly stored into an array of XpmExtension use **XpmFreeExtensions**().

To free any data allocated by an XPM function use the **XpmFree**() function. The current distribution of the XPM library uses the standard memory allocation functions and thus **XpmFree**() is nothing else than a define to the standard **free**(3). However since these functions may be redefined in specific environments it is wise to use **XpmFree**()

To free possible data stored into an XpmAttributes structure use **XpmFreeAttributes**(). The **XpmFreeAttributes**() function frees the structure members which have been malloc'ed such as the pixels list.

To dynamically allocate an XpmAttributes structure use the **XpmAttributesSize**() function. The **XpmAttributesSize**() function provides applications using dynamic libraries with a safe way to allocate and then refer to an XpmAttributes structure, disregarding whether the XpmAttributes structure size has changed or not since compiled.

To get data when building an error message, one can use **XpmGetErrorString**(). **XpmGetErrorString**() returns a string related to the given XPM error code.

The **XpmLibraryVersion**() function can be used when one needs to figure out which version of the library is in use. The value returned by **XpmLibraryVersion**() can be compared to the value of **XpmIncludeVersion** which is defined in the header file "xpm.h". These numbers are computed with the following formula:

(XpmFormat * 100 + XpmVersion) * 100 + XpmRevision

where **XpmFormat** is the version number of the format, **XpmVersion** is the library version number (which changes only if the API changes), and **XpmRevision** is the library minor version number.

The **XpmFreeExtensions**() function frees all data stored in every extension and the array itself. Note

that **XpmFreeAttributes**() calls this function and thus most of the time it should not need to be explicitly called.

SEE ALSO

XpmCreateBuffer(3), **XpmCreateData**(3), **XpmCreateImage**(3), **XpmCreatePixmap**(3), **XpmCreateXpmImage**(3), **XpmRead**(3), **XpmWrite**(3)