

NAME

XpmCreateData - create an Data structure

SYNOPSIS

```
int XpmCreateDataFromImage(Display *display, char ***data_return,  
    XImage *image, XImage *shapemask, XpmAttributes *attributes);
```

```
int XpmCreateDataFromPixmap(Display *display, char ***data_return,  
    Pixmap pixmap, Pixmap shapemask, XpmAttributes *attributes);
```

```
int XpmCreateDataFromXpmImage(char ***data_return, XpmImage *image,  
    XpmInfo *info);
```

ARGUMENTS

display

Specifies the connection to the X server.

data_return

Returns the data which is created.

pixmap

Specifies the pixmap.

shapemask

Specifies the shape mask pixmap.

attributes

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

info

Specifies the location of a structure to get information.

image

Specifies the image

DESCRIPTION

XpmCreateDataFromImage

In some cases, one may want to create an XPM data from an XImage, to do so use **XpmCreateDataFromImage()**. The **XpmCreateDataFromImage()** function exactly works as **XpmWriteFileFromImage(3)** does and returns the same way. It just writes to a single block malloc'ed data instead of to a file. It is the caller's responsibility to free the data, using **XpmFree(3)** when finished.

XpmCreateDataFromPixmap

XpmCreateDataFromPixmap() creates an XPM data from a Pixmap. The **XpmCreateDataFromPixmap()** function uses **XGetImage(3)** to get from the given pixmaps the related X images which are passed to **XpmCreateDataFromImage()**. Then it destroys the created images using **XDestroyImage(3)**. **XpmCreateDataFromPixmap()** returns the same errors as **XpmCreateDataFromImage()**.

XpmCreateDataFromXpmImage

XpmCreateDataFromXpmImage() creates an XPM data from an XpmImage. The **XpmCreateDataFromXpmImage()** function writes out the given image to a single block malloc'ed data in XPM format. If insufficient working storage is allocated, it returns **XpmNoMemory**, and returns **XpmSuccess** on success. If the passed XpmInfo structure pointer is not NULL, **XpmCreateDataFromXpmImage()** looks for the following attributes: XpmExtensions, and XpmHotspot, and writes the related information out as well. It is the caller's responsibility to free the data, using **XpmFree(3)** when finished.

SEE ALSO

XpmFree(3), **XpmWriteFileFromImage(3)**