XdbeGetVisualInfo

#include <X11/extensions/Xdbe.h>

XdbeScreenVisualInfo *XdbeGetVisualInfo(
    Display *dpy,
    Drawable *screen_specifiers,
    int *num_screens)

This function returns information about which visuals support double buffering. The argument `num_screens` specifies how many elements there are in the `screen_specifiers` list. Each drawable in `screen_specifiers` designates a screen for which the supported visuals are being requested. If `num_screens` is zero, information for all screens is requested. In this case, upon return from this function, `num_screens` will be set to the number of screens that were found. If an error occurs, this function returns NULL, else it returns a pointer to a list of `XdbeScreenVisualInfo` structures of length `num_screens`. The nth element in the returned list corresponds to the nth drawable in the `screen_specifiers` list, unless `num_screens` was passed in with the value zero, in which case the nth element in the returned list corresponds to the nth screen of the server, starting with screen zero. The `XdbeScreenVisualInfo` structure has the following fields:

```
    int count
    XdbeVisualInfo *visinfo
```

count specifies the number of items in visinfo. visinfo specifies a list of visuals, depths, and performance hints for this screen.

The `XdbeVisualInfo` structure has the following fields:

```
    VisualID visual
    int  depth
    int  perflevel
```

visual specifies one visual ID that supports double-buffering. depth specifies the depth of the visual. perflevel is a performance hint.

The only operation defined on a perflevel is comparison to a perflevel of another visual on the same screen. The visual having the higher perflevel is likely to have better double-buffering graphics performance than the visual having the lower perflevel. Nothing can be deduced from the following: the magnitude of the difference of two perflevels, a perflevel value in isolation, or comparing perflevels from different servers.

BadDrawable

One or more values passed in `screen_specifiers` is not a valid drawable.

DBE, XdbeAllocateBackBufferName(), XdbeBeginIdiom(), XdbeDeallocateBackBufferName(), XdbeEndIdiom(), XdbeFreeVisualInfo(), XdbeGetBackBufferAttributes(), XdbeQueryExtension(), XdbeSwapBuffers().