NAME

PEXGetImpDepConstants - Get Implementation Dependent Constants

SYNTAX

Status PEXGetImpDepConstants(Display **display*, Drawable *drawable*, unsigned long *count*, unsigned short **names*, PEXImpDepConstant ***constants_return*)

PARAMETERS

display A pointer to a display structure returned by a successful **XOpenDisplay** call.

drawable The resource identifier of a drawable.

count The number of implementation-dependent constants.

names An array of names of implementation-dependent constants to be returned. (See the Description for valid values.)

constants_return Returns an array of implementation-dependent constants.

RETURNS

Zero if unsuccessful; non-zero otherwise.

DESCRIPTION

PEXGetImpDepConstants allows an application program to query one or more implementation-dependent constants.

A single integer or floating-point value is returned for each value requested. These values are returned in order, with one return value in constants for each item in *names.* The implementation-dependent constants returned are based on the values used for a drawable having the same root and depth as the drawable specified. PEXlib allocates memory for the returned constants. **XFree** should be called to deallocate the memory.

The following implementation-dependent constants are standard:

PEXIDBestColorApprox (integer) — Either **PEXColorApproxPowersOf2** or **PEXColorApproxAnyValues,** depending on whether there is a significant performance gain if the number of reds/greens/blues in the color approximation table entry is a power of two, so pixels can be composed using shifts and adds.

PEXIDDitheringSupported (integer) — Either **True** if the dithering hint in color approximation lookup tables is used to control dithering, or **False,** if the dithering hint in color approximation lookup tables is not used.

PEXIDDoubleBufferingSupported (integer) — Either **True** if the server supports double-buffering for workstation resources, or **False,** if the server does not supports double-buffering for workstation resources.

PEXIDMaxEdgeWidth (integer) — Width (in pixels) of widest edge that can be drawn.

PEXIDMaxHitsEventSupported (integer) — **True** if the server supports the **PEXMaxHitsReached** event, or **False,** if the server does not support the event.

PEXIDMaxLineWidth (integer) — Width (in pixels) of widest line or curve that can be drawn.

PEXIDMaxMarkerSize (integer) — Largest dimension (either height or width, in pixels) of largest marker that can be drawn. This maximum is exclusive of the marker type **PEXMarkerDot** which is always drawn as the smallest displayable point.

PEXIDMaxModelClipPlanes (integer) — Maximum number of modeling clipping planes may be defined.

PEXIDMaxNameSetNames (integer) — Maximum number of *names* allowed in a name set.

PEXIDMaxNonAmbientLights (integer) — Maximum number of non-ambient light sources that can be enabled at one time.

PEXIDMaxNURBOrder (integer) — Maximum non-uniform rational B-spline order supported.

PEXIDMaxTrimCurveOrder (integer) — Maximum order for trim curves.

PEXIDMinEdgeWidth (integer) — Width (in pixels) of thinnest edge that can be drawn.

PEXIDMinLineWidth (integer) — Width (in pixels) of thinnest line or curve that can be drawn.

PEXIDMinMarkerSize (integer) — Largest dimension (either height or width, in pixels) of smallest marker that can be drawn. This minimum is exclusive of the marker type **PEXMarkerDot** which is always drawn as the smallest displayable point.

PEXIDNominalEdgeWidth (integer) — Width (in pixels) of *standard* edge.

PEXIDNominalLineWidth (integer) — Width (in pixels) of *standard* line or curve.

PEXIDNominalMarkerSize (integer) — Largest dimension (either height or width, in pixels) of *standard* marker.

PEXIDNumSupportedEdgeWidths (integer) — Number of supported edge widths. A value of 0 indicates that all edge widths, including fractional widths, between the minimum and maximum edge width are supported.

PEXIDNumSupportedLineWidths (integer) — Number of supported line or curve widths. A value of 0 indicates that all line widths, including fractional widths, between the minimum and maximum line width are supported.

PEXIDNumSupportedMarkerSizes (integer) — Number of supported marker sizes. A value of 0 indicates that all marker sizes, including fractional values, between the minimum and maximum marker size are supported.

PEXIDTransparencySupported (integer) — Either **True** if the transmission coefficient is utilized in the reflectance calculations, or **False,** if the transmission coefficient is not utilized.

PEXIDChromaticityRedU (flt_point) — CIEYUV u chromaticity coefficient for the red channel of the (properly adjusted) display device.

PEXIDChromaticityRedV (flt_point) — CIEYUV v chromaticity coefficient for the red channel of the (properly adjusted) display device.

PEXIDLuminanceRed (flt_point) — CIEYUV luminance value for the red channel of the (properly adjusted) display device.

PEXIDChromaticityGreenU (flt_point) — CIEYUV u chromaticity coefficient for the green channel of the (properly adjusted) display device.

PEXIDChromaticityGreenV (flt_point) — CIEYUV v chromaticity coefficient for the green channel of the (properly adjusted) display device.

PEXIDLuminanceGreen (flt_point) — CIEYUV luminance value for the green channel of the (properly adjusted) display device.

PEXIDChromaticityBlueU (flt_point) — CIEYUV u chromaticity coefficient for the blue channel of the (properly adjusted) display device.

PEXIDChromaticityBlueV (flt_point) — CIEYUV v chromaticity coefficient for the blue channel of the (properly adjusted) display device.

PEXIDLuminanceBlue (flt_point) — CIEYUV luminance value for the blue channel of the (properly adjusted) display device.

PEXIDChromaticityWhiteU (flt_point) — CIEYUV u chromaticity coefficient for the reference white of the (properly adjusted) display device.

PEXIDChromaticityWhiteV (flt_point) — CIEYUV v chromaticity coefficient for the reference white of the (properly adjusted) display device.

PEXIDLuminanceWhite (flt_point) — CIEYUV luminance value for the reference white of the (properly adjusted) display device.

DATA STRUCTURES

```
typedef union {
        unsigned long        integer;
        float                flt_point;
} PEXImpDepConstant;
```

ERRORS

BadDrawable
The specified drawable resource identifier is invalid.

BadMatch
The specified drawable is unsupported.

BadValue
A specified implementation-dependent constant name is invalid.