

**NAME**

AllPlanes, BlackPixel, WhitePixel, ConnectionNumber, DefaultColormap, DefaultDepth, XListDepths, DefaultGC, DefaultRootWindow, DefaultScreenOfDisplay, DefaultScreen, DefaultVisual, DisplayCells, DisplayPlanes, DisplayString, XMaxRequestSize, XExtendedMaxRequestSize, LastKnownRequestProcessed, NextRequest, ProtocolVersion, ProtocolRevision, QLength, RootWindow, ScreenCount, ScreenOfDisplay, ServerVendor, VendorRelease - Display macros and functions

**SYNTAX**

```
unsigned long AllPlanes;

unsigned long BlackPixel(Display *display, int screen_number);

unsigned long WhitePixel(Display *display, int screen_number);

int ConnectionNumber(Display *display);

Colormap DefaultColormap(Display *display, int screen_number);

int DefaultDepth(Display *display, int screen_number);

int *XListDepths(Display *display, int screen_number, int *count_return);

GC DefaultGC(Display *display, int screen_number);

Window DefaultRootWindow(Display *display);

Screen *DefaultScreenOfDisplay(Display *display);

int DefaultScreen(Display *display);

Visual *DefaultVisual(Display *display, int screen_number);

int DisplayCells(Display *display, int screen_number);

int DisplayPlanes(Display *display, int screen_number);

char *DisplayString(Display *display);

long XMaxRequestSize(Display *display);
```

```
long XExtendedMaxRequestSize(Display *display);

unsigned long LastKnownRequestProcessed(Display *display);

unsigned long NextRequest(Display *display);

int ProtocolVersion(Display *display);

int ProtocolRevision(Display *display);

int QLength(Display *display);

Window RootWindow(Display *display, int screen_number);

int ScreenCount(Display *display);

Screen *ScreenOfDisplay(Display *display, int screen_number);

char *ServerVendor(Display *display);

int VendorRelease(Display *display);
```

## ARGUMENTS

*display* Specifies the connection to the X server.

*screen\_number* Specifies the appropriate screen number on the host server.

*count\_return* Returns the number of depths.

## DESCRIPTION

The **AllPlanes** macro returns a value with all bits set to 1 suitable for use in a plane argument to a procedure.

The **BlackPixel** macro returns the black pixel value for the specified screen.

The **WhitePixel** macro returns the white pixel value for the specified screen.

The **ConnectionNumber** macro returns a connection number for the specified display.

The **DefaultColormap** macro returns the default colormap ID for allocation on the specified screen.

The **DefaultDepth** macro returns the depth (number of planes) of the default root window for the specified screen.

The **XListDepths** function returns the array of depths that are available on the specified screen. If the specified screen\_number is valid and sufficient memory for the array can be allocated, **XListDepths** sets count\_return to the number of available depths. Otherwise, it does not set count\_return and returns NULL. To release the memory allocated for the array of depths, use **XFree**.

The **DefaultGC** macro returns the default GC for the root window of the specified screen.

The **DefaultRootWindow** macro returns the root window for the default screen.

The **DefaultScreenOfDisplay** macro returns the default screen of the specified display.

The **DefaultScreen** macro returns the default screen number referenced in the **XOpenDisplay** routine.

The **DefaultVisual** macro returns the default visual type for the specified screen.

The **DisplayCells** macro returns the number of entries in the default colormap.

The **DisplayPlanes** macro returns the depth of the root window of the specified screen.

The **DisplayString** macro returns the string that was passed to **XOpenDisplay** when the current display was opened.

The **XMaxRequestSize** function returns the maximum request size (in 4-byte units) supported by the server without using an extended-length protocol encoding. Single protocol requests to the server can be no larger than this size unless an extended-length protocol encoding is supported by the server. The protocol guarantees the size to be no smaller than 4096 units (16384 bytes). Xlib automatically breaks data up into multiple protocol requests as necessary for the following functions: **XDrawPoints**, **XDrawRectangles**, **XDrawSegments**, **XFillArcs**, **XFillRectangles**, and **XPutImage**.

The **XExtendedMaxRequestSize** function returns zero if the specified display does not support an extended-length protocol encoding; otherwise, it returns the maximum request size (in 4-byte units) supported by the server using the extended-length encoding. The Xlib functions **XDrawLines**, **XDrawArcs**, **XFillPolygon**, **XChangeProperty**, **XSetClipRectangles**, and **XSetRegion** will use the extended-length encoding as necessary, if supported by the server. Use of the extended-length encoding in other Xlib functions (for example, **XDrawPoints**, **XDrawRectangles**, **XDrawSegments**, **XFillArcs**, **XFillRectangles**, **XPutImage**) is permitted but not required; an Xlib implementation may choose to split the data across multiple smaller requests instead.

The **LastKnownRequestProcessed** macro extracts the full serial number of the last request known by Xlib to have been processed by the X server.

The **NextRequest** macro extracts the full serial number that is to be used for the next request.

The **ProtocolVersion** macro returns the major version number (11) of the X protocol associated with the connected display.

The **ProtocolRevision** macro returns the minor protocol revision number of the X server.

The **QLength** macro returns the length of the event queue for the connected display.

The **RootWindow** macro returns the root window.

The **ScreenCount** macro returns the number of available screens.

The **ScreenOfDisplay** macro returns a pointer to the screen of the specified display.

The **ServerVendor** macro returns a pointer to a null-terminated string that provides some identification of the owner of the X server implementation.

The **VendorRelease** macro returns a number related to a vendor's release of the X server.

#### **SEE ALSO**

BlackPixelOfScreen(3), ImageByteOrder(3), IsCursorKey(3), XOpenDisplay(3)

*Xlib - C Language X Interface*