XDrawLine, XDrawLines, XDrawSegments, XSegment − draw lines, polygons, and line structure

XDrawLine(*display***,** *d***,** *gc***,** *x1***,** *y1***,** *x2***,** *y2***)**
> **Display ****display***;**
> **Drawable** *d***;**
> **GC** *gc***;**
> **int** *x1***,** *y1***,** *x2***,** *y2***;**

XDrawLines(*display*, *d*, *gc*, *points*, *npoints*, *mode*)
> Display *display;
> Drawable *d*;
> GC *gc*;
> XPoint *points;
> int *npoints*;
> int *mode*;

XDrawSegments(*display*, *d*, *gc*, *segments*, *nsegments*)
> Display *display;
> Drawable *d*;
> GC *gc*;
> XSegment *segments;
> int *nsegments*;

d	Specifies the drawable.
display	Specifies the connection to the X server.
gc	Specifies the GC.
mode	Specifies the coordinate mode. You can pass **CoordModeOrigin** or **CoordModePrevious**.
npoints	Specifies the number of points in the array.
nsegments	Specifies the number of segments in the array.
points	Specifies an array of points.
segments	Specifies an array of segments.
x1	
y1	
x2	
y2	Specify the points (x1, y1) and (x2, y2) to be connected.

The XDrawLine function uses the components of the specified GC to draw a line between the specified set of points (x1, y1) and (x2, y2). It does not perform joining at coincident endpoints. For any given line, **XDrawLine** does not draw a pixel more than once. If lines intersect, the intersecting pixels are drawn multiple times.

The **XDrawLines** function uses the components of the specified GC to draw npoints−1 lines between each pair of points (point[i], point[i+1]) in the array of **XPoint** structures. It draws the lines in the order listed in the array. The lines join correctly at all intermediate points, and if the first and last points coincide, the first and last lines also join correctly. For any given line, **XDrawLines** does not draw a pixel more than once. If thin (zero line-width) lines intersect, the intersecting pixels are drawn multiple times. If wide lines intersect, the intersecting pixels are drawn only once, as though the entire **PolyLine** protocol request were a single, filled shape. **CoordModeOrigin** treats all coordinates as relative to the origin, and **CoordModePrevious** treats all coordinates after the first as relative to the previous point.

The **XDrawSegments** function draws multiple, unconnected lines. For each segment, **XDrawSegments** draws a line between (x1, y1) and (x2, y2). It draws the lines in the order listed in the array of **XSegment** structures and does not perform joining at coincident endpoints. For any given line, **XDrawSegments** does not draw a pixel more than once. If lines intersect, the intersecting pixels are drawn multiple times.

All three functions use these GC components: function, plane-mask, line-width, line-style, cap-style, fill-style, subwindow-mode, clip-x-origin, clip-y-origin, and clip-mask. The **XDrawLines** function also uses the join-style GC component. All three functions also use these GC mode-dependent components: foreground, background, tile, stipple, tile-stipple-x-origin, tile-stipple-y-origin, dash-offset, and dash-list.

XDrawLine, **XDrawLines**, and **XDrawSegments** can generate **BadDrawable**, **BadGC**, and **BadMatch** errors. **XDrawLines** can also generate a **BadValue** error.

The XSegment structure contains:

typedef struct {
        short x1, y1, x2, y2;
} XSegment;

All x and y members are signed integers. The width and height members are 16-bit unsigned integers. You should be careful not to generate coordinates and sizes out of the 16-bit ranges, because the protocol only has 16-bit fields for these values.

BadDrawable A value for a Drawable argument does not name a defined Window or Pixmap. **BadGC** A value for a GContext argument does not name a defined GContext. **BadMatch** An **InputOnly** window is used as a Drawable. **BadMatch** Some argument or pair of arguments has the correct type and range but fails to match in some other way required by the request. **BadValue** Some numeric value falls outside the range of values accepted by the request. Unless a specific range is specified for an argument, the full range defined by the argument's type is accepted. Any argument defined as a set of alternatives can generate this error.

XDrawArc(3X11), XDrawPoint(3X11), XDrawRectangle(3X11)
Xlib − C Language X Interface