

NAME

XSetTile, XSetStipple, XSetTSTOrigin - GC convenience routines

SYNTAX

```
int XSetTile(Display *display, GC gc, Pixmap tile);
```

```
int XSetStipple(Display *display, GC gc, Pixmap stipple);
```

```
int XSetTSTOrigin(Display *display, GC gc, int ts_x_origin, int ts_y_origin);
```

ARGUMENTS

display Specifies the connection to the X server.

gc Specifies the GC.

stipple Specifies the stipple you want to set for the specified GC.

tile Specifies the fill tile you want to set for the specified GC.

ts_x_origin

ts_y_origin Specify the x and y coordinates of the tile and stipple origin.

DESCRIPTION

The **XSetTile** function sets the fill tile in the specified GC. The tile and GC must have the same depth, or a **BadMatch** error results.

XSetTile can generate **BadAlloc**, **BadGC**, **BadMatch**, and **BadPixmap** errors.

The **XSetStipple** function sets the stipple in the specified GC. The stipple must have a depth of one, or a **BadMatch** error results.

XSetStipple can generate **BadAlloc**, **BadGC**, **BadMatch**, and **BadPixmap** errors.

The **XSetTSTOrigin** function sets the tile/stipple origin in the specified GC. When graphics requests call for tiling or stippling, the parent's origin will be interpreted relative to whatever destination drawable is specified in the graphics request.

XSetTSTOrigin can generate **BadAlloc** and **BadGC** errors.

DIAGNOSTICS

- BadAlloc** The server failed to allocate the requested resource or server memory.
- BadGC** A value for a GContext argument does not name a defined GContext.
- 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.
- BadPixmap** A value for a Pixmap argument does not name a defined Pixmap.

SEE ALSO

XCreateGC(3), XQueryBestSize(3), XSetArcMode(3), XSetClipOrigin(3), XSetFillStyle(3), XSetFont(3), XSetLineAttributes(3), XSetState(3)

Xlib - C Language X Interface