

## **XIconifyWindow, XWithdrawWindow, XReconfigureWMWindow – manipulate top-level windows**

Status **XIconifyWindow**(*display*, *w*, *screen\_number*)

```
Display *display;  
Window w;  
int screen_number;
```

Status **XWithdrawWindow**(*display*, *w*, *screen\_number*)

```
Display *display;  
Window w;  
int screen_number;
```

Status **XReconfigureWMWindow**(*display*, *w*, *screen\_number*, *value\_mask*, *values*)

```
Display *display;  
Window w;  
int screen_number;  
unsigned int value_mask;  
XWindowChanges *values;
```

<i>display</i>	Specifies the connection to the X server.
<i>screen_number</i>	Specifies the appropriate screen number on the host server.
<i>value_mask</i>	Specifies which values are to be set using information in the values structure. This mask is the bitwise inclusive OR of the valid configure window values bits.
<i>values</i>	Specifies the <b>XWindowChanges</b> structure.
<i>w</i>	Specifies the window.

The **XIconifyWindow** function sends a WM\_CHANGE\_STATE **ClientMessage** event with a format of 32 and a first data element of **IconicState** (as described in section 4.1.4 of the *Inter-Client Communication Conventions Manual*) and a window of *w* to the root window of the specified screen with an event mask set to **SubstructureNotifyMask|SubstructureRedirectMask**. Window managers may elect to receive this message and if the window is in its normal state, may treat it as a request to change the window's state from normal to iconic. If the WM\_CHANGE\_STATE property cannot be interned, **XIconifyWindow** does not send a message and returns a zero status. It returns a nonzero status if the client message is sent successfully; otherwise, it returns a zero status.

The **XWithdrawWindow** function unmaps the specified window and sends a synthetic **UnmapNotify** event to the root window of the specified screen. Window managers may elect to receive this message and may treat it as a request to change the window's state to withdrawn. When a window is in the withdrawn state, neither its normal nor its iconic representations is visible. It returns a nonzero status if the **UnmapNotify** event is successfully sent; otherwise, it returns a zero status.

**XWithdrawWindow** can generate a **BadWindow** error.

The **XReconfigureWMWindow** function issues a **ConfigureWindow** request on the specified top-level window. If the stacking mode is changed and the request fails with a **BadMatch** error, the error is trapped by Xlib and a synthetic **ConfigureRequestEvent** containing the same configuration parameters is sent to the root of the specified window. Window managers may elect to receive this event and treat it as a request to reconfigure the indicated window. It returns a nonzero status if the request or event is successfully sent; otherwise, it returns a zero status.

**XReconfigureWMWindow** can generate **BadValue** and **BadWindow** errors.

**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. **BadWindow** A value for a Window argument does not name a defined Window.

**XChangeWindowAttributes(3X11), XConfigureWindow(3X11), XCreateWindow(3X11),  
XDestroyWindow(3X11), XRaiseWindow(3X11), XMapWindow(3X11), XUnmapWindow(3X11)**  
*Xlib – C Language X Interface*