

NAME

XAddConnectionWatch, XRemoveConnectionWatch, XProcessInternalConnection, XInternalConnectionNumbers - handle Xlib internal connections

SYNTAX

```
typedef void (*XConnectionWatchProc)(Display *display, XPointer client_data, int fd, Bool opening, XPointer *watch_data);
```

```
Status XAddConnectionWatch(Display *display, XConnectionWatchProc procedure, XPointer client_data);
```

```
void XRemoveConnectionWatch(Display *display, XConnectionWatchProc procedure, XPointer client_data);
```

```
void XProcessInternalConnection(Display *display, int fd);
```

```
Status XInternalConnectionNumbers(Display *display, int **fd_return, int *count_return);
```

ARGUMENTS

- client_data* Specifies the additional client data.
- count_return* Returns the number of file descriptors.
- display* Specifies the connection to the X server.
- fd* Specifies the file descriptor.
- fd_return* Returns the file descriptors.
- procedure* Specifies the procedure to be called.

DESCRIPTION

The **XAddConnectionWatch** function registers a procedure to be called each time Xlib opens or closes an internal connection for the specified display. The procedure is passed the display, the specified *client_data*, the file descriptor for the connection, a Boolean indicating whether the connection is being opened or closed, and a pointer to a location for private watch data. If opening is **True**, the procedure can store a pointer to private data in the location pointed to by *watch_data*; when the procedure is later called for this same connection and opening is **False**, the location pointed to by *watch_data* will hold this same private data pointer.

This function can be called at any time after a display is opened. If internal connections already exist, the registered procedure will immediately be called for each of them, before **XAddConnectionWatch** returns. **XAddConnectionWatch** returns a nonzero status if the procedure is successfully registered; otherwise, it returns zero.

The registered procedure should not call any Xlib functions. If the procedure directly or indirectly causes the state of internal connections or watch procedures to change, the result is not defined. If Xlib has been initialized for threads, the procedure is called with the display locked and the result of a call by the procedure to any Xlib function that locks the display is not defined unless the executing thread has externally locked the display using **XLockDisplay**.

The **XRemoveConnectionWatch** function removes a previously registered connection watch procedure. The `client_data` must match the `client_data` used when the procedure was initially registered.

The **XProcessInternalConnection** function processes input available on an internal connection. This function should be called for an internal connection only after an operating system facility (for example, **select** or **poll**) has indicated that input is available; otherwise, the effect is not defined.

The **XInternalConnectionNumbers** function returns a list of the file descriptors for all internal connections currently open for the specified display. When the allocated list is no longer needed, free it by using **XFree**. This function returns a nonzero status if the list is successfully allocated; otherwise, it returns zero.

SEE ALSO

Xlib - C Language X Interface