NAME

XtAppAddSignal, XtRemoveSignal, XtNoticeSignal − register and remove a signal source

SYNTAX

XtSignalId XtAppAddSignal(XtAppContext *app_context*, XtSignalCallbackProc *proc*, XtPointer *client_data*);

void XtRemoveSignal(XtSignalId *id*);

void XtNoticeSignal(XtSignalId *id*);

ARGUMENTS

app_context	Specifies the application context.
client_data	Specifies the argument that is to be passed to the specified procedure when a signal has been raised.
id	Specifies the ID returned from the corresponding **XtAppAddSignal** call.
proc	Specifies the procedure that is to be called when the signal has been raised.

DESCRIPTION

The **XtAppAddSignal** function initiates a mechanism for handling signals within the context of the Intrinsics. Prior to establishing an operating system dependent signal handler the application may call **XtAppAddSignal** and store the returned *id* in a place accessible to the signal handler.

Upon receipt of a signal from the operating system, the application may call **XtNoticeSignal** passing the *id* returned by the call to **XtAppAddSignal**.

XtNoticeSignal is the only Intrinsics function that can safely be called from a signal handler. If **XtNoticeSignal** is called multiple times before the Intrinsics are able to invoke the registered callback, the callback is only called once. Logically the Intrinsics maintain ‘‘pending’’ for each registered callback. This flag is initially **False** and is set to **True** by **XtNoticeSignal**; the Intrinsics invoke the callback whenever the flag is **True**, and the flag is set to **False** just before the callback is invoked.

The **XtRemoveSignal** function is called to remove the specified Intrinsics signal handler. The client should disable the source of the signal before calling **XtRemoveSignal**.

SEE ALSO

XtAppAddTimeOut(3),XtAppAddInput(3),XtAppAddWorkProc(3)
X Toolkit Intrinsics − C Language Interface
Xlib − C Language X Interface