NAME
     XIGrabButton, XIUngrabButton, XIGrabKeycode, XIUngrabKeycode, XIGrabTouchBegin,
     XIUngrabTouchBegin, XIGrabPinchGestureBegin, XIUngrabPinchGestureBegin,
     XIGrabSwipeGestureBegin - grab/ungrab buttons or keys

SYNOPSIS
     #include <X11/extensions/XInput2.h>

     int XIGrabButton(Display *display,
                    int deviceid,
                    int button,
                    Window grab_window,
                    Cursor cursor,
                    int grab_mode,
                    int paired_device_mode,
                    Bool owner_events,
                    XIEventMask *mask,
                    int num_modifiers,
                    XIGrabModifiers *modifiers_inout);

     int XIUngrabButton(Display *display,
                     int deviceid,
                     int button,
                     Window grab_window,
                     int num_modifiers,
                     XIGrabModifiers *modifiers);

     int XIGrabKeycode(Display *display,
                    int deviceid,
                    int keycode,
                    Window grab_window,
                    int grab_mode,
                    int paired_device_mode,
                    Bool owner_events,
                    XIEventMask *mask,
                    int num_modifiers,
                    XIGrabModifiers *modifiers_inout);

     int XIUngrabKeycode(Display *display,
                     int deviceid,

```
                    int keycode,
                    Window grab_window,
                    int num_modifiers,
                    XIGrabModifiers *modifiers);

    int XIGrabTouchBegin( Display *display,
                    int deviceid,
                    Window grab_window,
                    Bool owner_events,
                    XIEventMask *mask,
                    int num_modifiers,
                    XIGrabModifiers *modifiers_inout);

    int XIUngrabTouchBegin( Display *display,
                    int deviceid,
                    Window grab_window,
                    int num_modifiers,
                    XIGrabModifiers *modifiers);

    int XIGrabPinchGestureBegin( Display* display,
                    int deviceid,
                    Window grab_window,
                    int grab_mode,
                    int paired_device_mode,
                    int owner_events,
                    XIEventMask *mask,
                    int num_modifiers,
                    XIGrabModifiers *modifiers_inout);

    int XIUngrabPinchGestureBegin( Display* display,
                    int deviceid,
                    Window grab_window,
                    int num_modifiers,
                    XIGrabModifiers *modifiers);

    int XIGrabSwipeGestureBegin( Display* display,
                    int deviceid,
                    Window grab_window,
                    int grab_mode,
                    int paired_device_mode,
```

```
                        int owner_events,
                        XIEventMask *mask,
                        int num_modifiers,
                        XIGrabModifiers *modifiers_inout);

    int XIUngrabSwipeGestureBegin( Display* display,
                        int deviceid,
                        Window grab_window,
                        int num_modifiers,
                        XIGrabModifiers *modifiers);
```

display
    Specifies the connection to the X server.

device
    Specifies the device that is to be grabbed or released

button
    Specifies the device button that is to be grabbed or
    released or XIAnyButton.

keycode
    Specifies the keycode that is to be grabbed or released
    or XIAnyKeycode.

num_modifiers
    Number of elements in modifiers or modifiers_inout.

modifiers
    Specifies the set of latched and base modifiers or
    XIAnyModifier to ungrab. The data type is for
    consistency with the respective grab request and the
    status code of the XIGrabModifiers struct is ignored.

modifiers_inout
    Specifies the set of latched and base modifiers or
    XIAnyModifier to grab. Returns the modifiers that could
    not be grabbed and their error code.

grab_window

Specifies the grab window.

owner_events
    Specifies a Boolean value that indicates whether the are
    to be reported as usual or reported with respect to the
    grab window.

mask
    Specifies the event mask.

grab_mode
    Specifies further processing of events from this device.
    You can pass XIGrabModeSync or XIGrabModeAsync.

paired_device_mode
    Specifies further processing of events from the paired
    master device. You can pass XIGrabModeSync or
    XIGrabModeAsync. If deviceid specifies a floating slave
    device, this parameter is ignored.

DESCRIPTION
    XIGrabButton, XIGrabKeycode, XIGrabTouchBegin, XIGrabPinchGestureBegin,
    XIGrabSwipeTouchBegin establish a passive grab.
    The modifier device for a button grab is the paired master device
    if deviceid specifies a master pointer. Otherwise, the modifier
    device is the device specified with deviceid. In the future,
    the device is actively grabbed (as for XIGrabDevice, the
    last-grab time is set to the time at which the button or keycode
    was pressed and the XI_ButtonPress or XI_KeyPress event is
    reported if all of the following conditions are true:
      * The device is not grabbed, and the specified button or
        keycode is logically pressed, a touch or a gesture event occurs when the
        specified modifier keys are logically down on the modifier device
        and no other buttons or modifier keys are logically down.
      * Either the grab window is an ancestor of (or is) the focus
        window, OR the grab window is a descendent of the focus
        window and contains the device.
      * A passive grab on the same button/modifier combination does
        not exist on any ancestor of grab_window.

The interpretation of the remaining arguments is as for
XIGrabDevice. The active grab is terminated automatically when
the logical state of the device has all buttons or keys
released (independent of the logical state of the modifier
keys).

If the device is an attached slave device, the device is
automatically detached from the master device when the grab
activates and reattached to the same master device when the
grab deactivates. If the master device is removed while the
device is floating as a result of a grab, the device remains
floating once the grab deactivates.

Note that the logical state of a device (as seen by client
applications) may lag the physical state if device event
processing is frozen.

This request overrides all previous grabs by the same client on
the same button/modifier or keycode/modifier combinations on the
same window. A modifiers of XIAnyModifier is equivalent to
issuing the grab request for all possible modifier combinations
(including the combination of no modifiers). It is not required
that all modifiers specified have currently assigned KeyCodes.
A button of XIAnyButton is equivalent to issuing the request
for all possible buttons. Otherwise, it is not required that
the specified button currently be assigned to a physical
button.

If some other client has already issued a XIGrabButton or
XIGrabKeycode with the same button/modifier or keycode/modifier
combination on the same window, a BadAccess error results. When
using XIAnyModifier or XIAnyButton , the request fails
completely, and a XIBadAccess error results (no grabs are
established) if there is a conflicting grab for any
combination. XIGrabButton and XIGrabKeycode have no effect on an
active grab.

On success, XIGrabButton, XIGrabKeycode, XIGrabTouchBegin,
XIGrabPinchGestureBegin and XIGrabSwipeGestureBegin return 0;
If one or more modifier combinations could not be grabbed,

XIGrabButton, XIGrabKeycode, XIGrabTouchBegin, XIGrabPinchGestureBegin
and XIGrabSwipeGestureBegin return the number of
failed combinations and modifiers_inout contains the failed combinations
and their respective error codes.

XIGrabButton, XIGrabKeycode, XIGrabTouchBegin, XIGrabPinchGestureBegin
and XIGrabSwipeGestureBegin can generate BadClass,
BadDevice, BadMatch, BadValue, and BadWindow errors.

XIUngrabButton, XIUngrabKeycode, XIUngrabTouchBegin, XIUngrabPinchGestureBegin
and XIUngrabSwipeGestureBegin release the
passive grab for a button/modifier, keycode/modifier or touch/modifier
combination on the specified window if it was grabbed by this client. A
modifier of XIAnyModifier is equivalent to issuing the ungrab request
for all possible modifier combinations, including the
combination of no modifiers. A button of XIAnyButton is
equivalent to issuing the request for all possible buttons.
XIUngrabButton and XIUngrabKeycode have no effect on an active
grab.

XIUngrabButton, XIUngrabKeycode, XIUngrabTouchBegin,
XIUngrabPinchGestureBegin and XIUngrabSwipeGestureBegin can generate
BadDevice, BadMatch, BadValue and BadWindow errors.

RETURN VALUE

XIGrabButton, XIGrabKeycode, XIGrabTouchBegin and
XIGrabPinchGestureBegin and XIGrabSwipeGestureBegin return the number of
modifier combination that could not establish a passive grab. The
modifiers are returned in modifiers_inout, along with the respective
error for this modifier combination. If XIGrabButton, XIGrabKeycode
or XIGrabTouchBegin return zero, passive grabs with all requested
modifier combinations were established successfully.

DIAGNOSTICS

BadDevice

An invalid deviceid was specified.

BadMatch

This error may occur if XIGrabButton specified a device
that has no buttons, or XIGrabKeycode specified a device

that has no keys,
or XIGrabTouchBegin specified a device that is not touch-capable,
or XIGrabPinchGestureBegin specified a device that is not gesture-capable,
or XIGrabSwipeGestureBegin specified a device that is not gesture-capable.

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.

BUGS

The protocol headers for XI 2.0 did not provide
XIGrabModeAsync or XIGrabModeSync. Use GrabModeSync and
GrabModeAsync instead, respectively.

SEE ALSO

XIAllowEvents(3)