

NAME

xcb_grab_key - Grab keyboard key(s)

SYNOPSIS

```
#include <xcb/xproto.h>
```

Request function

```
xcb_void_cookie_t xcb_grab_key(xcb_connection_t *conn, uint8_t owner_events,  
    xcb_window_t grab_window, uint16_t modifiers, xcb_keycode_t key, uint8_t pointer_mode,  
    uint8_t keyboard_mode);
```

REQUEST ARGUMENTS

conn The XCB connection to X11.

owner_events If 1, the *grab_window* will still get the key events. If 0, events are not reported to the *grab_window*.

grab_window Specifies the window on which the key should be grabbed.

modifiers The modifiers to grab.

Using the special value *XCB_MOD_MASK_ANY* means grab the key with all possible modifier combinations.

key The keycode of the key to grab.

The special value *XCB_GRAB_ANY* means grab any key.

pointer_mode One of the following values:

XCB_GRAB_MODE_SYNC

The state of the keyboard appears to freeze: No further keyboard events are generated by the server until the grabbing client issues a releasing *AllowEvents* request or until the keyboard grab is released.

XCB_GRAB_MODE_ASYNC

Keyboard event processing continues normally.

keyboard_mode

One of the following values:

XCB_GRAB_MODE_SYNC

The state of the keyboard appears to freeze: No further keyboard events are generated by the server until the grabbing client issues a releasing *AllowEvents* request or until the keyboard grab is released.

XCB_GRAB_MODE_ASYNC

Keyboard event processing continues normally.

DESCRIPTION

Establishes a passive grab on the keyboard. In the future, the keyboard is actively grabbed (as for *GrabKeyboard*), the last-keyboard-grab time is set to the time at which the key was pressed (as transmitted in the *KeyPress* event), and the *KeyPress* event is reported if all of the following conditions are true:

The keyboard is not grabbed and the specified key (which can itself be a modifier key) is logically pressed when the specified modifier keys are logically down, and no other modifier keys are logically down.

Either the *grab_window* is an ancestor of (or is) the focus window, or the *grab_window* is a descendant of the focus window and contains the pointer.

A passive grab on the same key combination does not exist on any ancestor of *grab_window*.

The interpretation of the remaining arguments is as for *XGrabKeyboard*. The active grab is terminated automatically when the logical state of the keyboard has the specified key released (independent of the logical state of the modifier keys), at which point a *KeyRelease* event is reported to the grabbing window.

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

A *modifiers* argument of *AnyModifier* is equivalent to issuing the 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 *keycode* argument of *AnyKey* is equivalent to issuing the request for all possible *KeyCodes*. Otherwise, the specified *keycode* must be in the range specified

by `min_keycode` and `max_keycode` in the connection setup, or a `BadValue` error results.

If some other client has issued a `XGrabKey` with the same key combination on the same window, a `BadAccess` error results. When using `AnyModifier` or `AnyKey`, the request fails completely, and a `BadAccess` error results (no grabs are established) if there is a conflicting grab for any combination.

RETURN VALUE

Returns an `xcb_void_cookie_t`. Errors (if any) have to be handled in the event loop.

If you want to handle errors directly with `xcb_request_check` instead, use `xcb_grab_key_checked`. See [xcb-requests\(3\)](#) for details.

ERRORS

`xcb_access_error_t`

Another client has already issued a `GrabKey` with the same button/key combination on the same window.

`xcb_value_error_t`

The key is not `XCB_GRAB_ANY` and not in the range specified by `min_keycode` and `max_keycode` in the connection setup.

`xcb_window_error_t`

The specified `window` does not exist.

SEE ALSO

[xcb-requests\(3\)](#), [xcb_grab_keyboard\(3\)](#)

AUTHOR

Generated from `xproto.xml`. Contact xcb@lists.freedesktop.org for corrections and improvements.