

NAME

XChangeDeviceDontPropagateList, XGetDeviceDontPropagateList - query or change the dont-propagate-list for extension devices

SYNOPSIS

```
#include <X11/extensions/XInput.h>
```

```
int XChangeDeviceDontPropagateList( Display *display,  
                                   Window window,  
                                   int count,  
                                   XEventClass *event_list,  
                                   int mode);
```

```
XEventClass* XGetDeviceDontPropagateList( Display *display,  
                                           Window window,  
                                           int *count);
```

display

Specifies the connection to the X server.

window

Specifies the window whose dont-propagate-list is to be queried or modified.

event_list

Specifies a pointer to a list of event classes.

mode

Specifies the mode. You can pass AddToList, or DeleteFromList.

count

Specifies the number of event classes in the list.

DESCRIPTION

The XChangeDeviceDontPropagateList request modifies the list of events that should not be propagated to ancestors of the event window. This request allows extension events to be added to or deleted from that list. By default, all events are propagated to ancestor windows. Once modified, the list remains modified

for the life of the window. Events are not removed from the list because the client that added them has terminated.

Suppression of event propagation is not allowed for all input extension events. If a specified event class is one that cannot be suppressed, a `BadClass` error will result. Events that can be suppressed include `DeviceKeyPress`, `DeviceKeyRelease`, `DeviceButtonPress`, `DeviceButtonRelease`, `DeviceMotionNotify`, `ProximityIn`, and `ProximityOut`.

`XChangeDeviceDontPropagateList` can generate a `BadDevice`, `BadClass`, or `BadValue` error.

The `XGetDeviceDontPropagateList` request queries the list of events that should not be propagated to ancestors of the event window.

`XGetDeviceDontPropagateList` can generate a `BadClass` or `BadWindow` error.

DIAGNOSTICS

`BadDevice`

An invalid device was specified. The specified device does not exist or has not been opened by this client via `XOpenInputDevice`. This error may also occur if some other client has caused the specified device to become the X keyboard or X pointer device via the `XChangeKeyboardDevice` or `XChangePointerDevice` requests.

`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`

An invalid window id was specified.

`BadClass`

An invalid event class was specified.