

**NAME**

xcb\_randr\_get\_panning -

**SYNOPSIS**

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

**Request function**

```
xcb_randr_get_panning_cookie_t xcb_randr_get_panning(xcb_connection_t *conn,
    xcb_randr_crtc_t crtc);
```

**Reply datastructure**

```
typedef struct xcb_randr_get_panning_reply_t {
    uint8_t    response_type;
    uint8_t    status;
    uint16_t   sequence;
    uint32_t   length;
    xcb_timestamp_t timestamp;
    uint16_t   left;
    uint16_t   top;
    uint16_t   width;
    uint16_t   height;
    uint16_t   track_left;
    uint16_t   track_top;
    uint16_t   track_width;
    uint16_t   track_height;
    int16_t    border_left;
    int16_t    border_top;
    int16_t    border_right;
    int16_t    border_bottom;
} xcb_randr_get_panning_reply_t;
```

**Reply function**

```
xcb_randr_get_panning_reply_t *xcb_randr_get_panning_reply(xcb_connection_t *conn,
    xcb_randr_get_panning_cookie_t cookie, xcb_generic_error_t **e);
```

**REQUEST ARGUMENTS**

*conn*           The XCB connection to X11.

*crtc*            TODO: NOT YET DOCUMENTED.

**REPLY FIELDS**

*response\_type* The type of this reply, in this case *XCB\_RANDR\_GET\_PANNING*. This field is also present in the *xcb\_generic\_reply\_t* and can be used to tell replies apart from each other.

*sequence* The sequence number of the last request processed by the X11 server.

*length* The length of the reply, in words (a word is 4 bytes).

*status* TODO: NOT YET DOCUMENTED.

*timestamp* TODO: NOT YET DOCUMENTED.

*left* TODO: NOT YET DOCUMENTED.

*top* TODO: NOT YET DOCUMENTED.

*width* TODO: NOT YET DOCUMENTED.

*height* TODO: NOT YET DOCUMENTED.

*track\_left* TODO: NOT YET DOCUMENTED.

*track\_top* TODO: NOT YET DOCUMENTED.

*track\_width* TODO: NOT YET DOCUMENTED.

*track\_height* TODO: NOT YET DOCUMENTED.

*border\_left* TODO: NOT YET DOCUMENTED.

*border\_top* TODO: NOT YET DOCUMENTED.

*border\_right* TODO: NOT YET DOCUMENTED.

*border\_bottom* TODO: NOT YET DOCUMENTED.

**DESCRIPTION****RETURN VALUE**

Returns an *xcb\_randr\_get\_panning\_cookie\_t*. Errors have to be handled when calling the reply function

*xcb\_randr\_get\_panning\_reply*.

If you want to handle errors in the event loop instead, use *xcb\_randr\_get\_panning\_unchecked*. See **xcb-requests(3)** for details.

## **ERRORS**

This request does never generate any errors.

## **SEE ALSO**

## **AUTHOR**

Generated from randr.xml. Contact [xcb@lists.freedesktop.org](mailto:xcb@lists.freedesktop.org) for corrections and improvements.