

NAME

xcb_xf86dri_get_device_info -

SYNOPSIS

```
#include <xcb/xf86dri.h>
```

Request function

```
xcb_xf86dri_get_device_info_cookie_t xcb_xf86dri_get_device_info(xcb_connection_t *conn,
    uint32_t screen);
```

Reply datastructure

```
typedef struct xcb_xf86dri_get_device_info_reply_t {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint32_t framebuffer_handle_low;
    uint32_t framebuffer_handle_high;
    uint32_t framebuffer_origin_offset;
    uint32_t framebuffer_size;
    uint32_t framebuffer_stride;
    uint32_t device_private_size;
} xcb_xf86dri_get_device_info_reply_t;
```

Reply function

```
xcb_xf86dri_get_device_info_reply_t *xcb_xf86dri_get_device_info_reply(xcb_connection_t *conn,
    xcb_xf86dri_get_device_info_cookie_t cookie, xcb_generic_error_t **e);
```

Reply accessors

```
uint32_t *xcb_xf86dri_get_device_info_device_private(const xcb_xf86dri_get_device_info_request_t
    *reply);
```

```
int xcb_xf86dri_get_device_info_device_private_length(const xcb_xf86dri_get_device_info_reply_t
    *reply);
```

```
xcb_generic_iterator_t xcb_xf86dri_get_device_info_device_private_end(const
    xcb_xf86dri_get_device_info_reply_t *reply);
```

REQUEST ARGUMENTS

conn The XCB connection to X11.

screen TODO: NOT YET DOCUMENTED.

REPLY FIELDS

response_type The type of this reply, in this case *XCB_XF86DRI_GET_DEVICE_INFO*. 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).

framebuffer_handle_low
 TODO: NOT YET DOCUMENTED.

framebuffer_handle_high
 TODO: NOT YET DOCUMENTED.

framebuffer_origin_offset
 TODO: NOT YET DOCUMENTED.

framebuffer_size
 TODO: NOT YET DOCUMENTED.

framebuffer_stride
 TODO: NOT YET DOCUMENTED.

device_private_size
 TODO: NOT YET DOCUMENTED.

DESCRIPTION

RETURN VALUE

Returns an *xcb_xf86dri_get_device_info_cookie_t*. Errors have to be handled when calling the reply function *xcb_xf86dri_get_device_info_reply*.

If you want to handle errors in the event loop instead, use *xcb_xf86dri_get_device_info_unchecked*. See **xcb-requests(3)** for details.

ERRORS

This request does never generate any errors.

SEE ALSO**AUTHOR**

Generated from xf86dri.xml. Contact xcb@lists.freedesktop.org for corrections and improvements.