

**NAME**

xcb\_glx\_get\_query\_objectiv\_arb -

**SYNOPSIS**

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

**Request function**

```
xcb_glx_get_query_objectiv_arb_cookie_t xcb_glx_get_query_objectiv_arb(xcb_connection_t *conn,
    xcb_glx_context_tag_t context_tag, uint32_t id, uint32_t pname);
```

**Reply datastructure**

```
typedef struct xcb_glx_get_query_objectiv_arb_reply_t {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint8_t pad1[4];
    uint32_t n;
    int32_t datum;
    uint8_t pad2[12];
} xcb_glx_get_query_objectiv_arb_reply_t;
```

**Reply function**

```
xcb_glx_get_query_objectiv_arb_reply_t
xcb_glx_get_query_objectiv_arb_reply(xcb_connection_t *conn,
    xcb_glx_get_query_objectiv_arb_cookie_t cookie, xcb_generic_error_t **e);
```

**Reply accessors**

```
int32_t xcb_glx_get_query_objectiv_arb_data(const xcb_glx_get_query_objectiv_arb_request_t
    *reply);
```

```
int xcb_glx_get_query_objectiv_arb_data_length(const xcb_glx_get_query_objectiv_arb_reply_t
    *reply);
```

```
xcb_generic_iterator_t xcb_glx_get_query_objectiv_arb_data_end(const
    xcb_glx_get_query_objectiv_arb_reply_t *reply);
```

**REQUEST ARGUMENTS**

*conn*           The XCB connection to X11.

*context\_tag*      TODO: NOT YET DOCUMENTED.

*id*                TODO: NOT YET DOCUMENTED.

*pname*            TODO: NOT YET DOCUMENTED.

## REPLY FIELDS

*response\_type*    The type of this reply, in this case *XCB\_GLX\_GET\_QUERY\_OBJECTIV\_ARB*. 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).

*n*                 TODO: NOT YET DOCUMENTED.

*datum*            TODO: NOT YET DOCUMENTED.

## DESCRIPTION

### RETURN VALUE

Returns an *xcb\_glx\_get\_query\_objectiv\_arb\_cookie\_t*. Errors have to be handled when calling the reply function *xcb\_glx\_get\_query\_objectiv\_arb\_reply*.

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

## ERRORS

This request does never generate any errors.

## SEE ALSO

### AUTHOR

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