

NAME

xcb_record_query_version -

SYNOPSIS

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

Request function

```
xcb_record_query_version_cookie_t xcb_record_query_version(xcb_connection_t *conn,  
    uint16_t major_version, uint16_t minor_version);
```

Reply datastructure

```
typedef struct xcb_record_query_version_reply_t {  
    uint8_t response_type;  
    uint8_t pad0;  
    uint16_t sequence;  
    uint32_t length;  
    uint16_t major_version;  
    uint16_t minor_version;  
} xcb_record_query_version_reply_t;
```

Reply function

```
xcb_record_query_version_reply_t *xcb_record_query_version_reply(xcb_connection_t *conn,  
    xcb_record_query_version_cookie_t cookie, xcb_generic_error_t **e);
```

REQUEST ARGUMENTS

conn The XCB connection to X11.

major_version TODO: NOT YET DOCUMENTED.

minor_version TODO: NOT YET DOCUMENTED.

REPLY FIELDS

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

major_version TODO: NOT YET DOCUMENTED.

minor_version TODO: NOT YET DOCUMENTED.

DESCRIPTION

RETURN VALUE

Returns an *xcb_record_query_version_cookie_t*. Errors have to be handled when calling the reply function *xcb_record_query_version_reply*.

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

ERRORS

This request does never generate any errors.

SEE ALSO

AUTHOR

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