

NAME

xcb_intern_atom - Get atom identifier by name

SYNOPSIS

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

Request function

```
xcb_intern_atom_cookie_t xcb_intern_atom(xcb_connection_t *conn, uint8_t only_if_exists,  
uint16_t name_len, const char *name);
```

Reply datastructure

```
typedef struct xcb_intern_atom_reply_t {  
    uint8_t response_type;  
    uint8_t pad0;  
    uint16_t sequence;  
    uint32_t length;  
    xcb_atom_t atom;  
} xcb_intern_atom_reply_t;
```

Reply function

```
xcb_intern_atom_reply_t *xcb_intern_atom_reply(xcb_connection_t *conn,  
xcb_intern_atom_cookie_t cookie, xcb_generic_error_t **e);
```

REQUEST ARGUMENTS

conn The XCB connection to X11.

only_if_exists Return a valid atom id only if the atom already exists.

name_len The length of the following *name*.

name The name of the atom.

REPLY FIELDS

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

atom TODO: NOT YET DOCUMENTED.

DESCRIPTION

Retrieves the identifier (`xcb_atom_t` TODO) for the atom with the specified name. Atoms are used in protocols like EWMH, for example to store window titles (`_NET_WM_NAME` atom) as property of a window.

If *only_if_exists* is 0, the atom will be created if it does not already exist. If *only_if_exists* is 1, `XCB_ATOM_NONE` will be returned if the atom does not yet exist.

RETURN VALUE

Returns an `xcb_intern_atom_cookie_t`. Errors have to be handled when calling the reply function `xcb_intern_atom_reply`.

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

ERRORS

`xcb_alloc_error_t`

TODO: reasons?

`xcb_value_error_t`

A value other than 0 or 1 was specified for *only_if_exists*.

EXAMPLE

```
/*
 * Resolves the _NET_WM_NAME atom.
 */
void my_example(xcb_connection_t *c) {
    xcb_intern_atom_cookie_t cookie;
    xcb_intern_atom_reply_t *reply;

    cookie = xcb_intern_atom(c, 0, strlen("_NET_WM_NAME"), "_NET_WM_NAME");
    /* ... do other work here if possible ... */
    if ((reply = xcb_intern_atom_reply(c, cookie, NULL))) {
        printf("The _NET_WM_NAME atom has ID %u0, reply->atom);
        free(reply);
    }
}
```

```
}
```

SEE ALSO

xcb-requests(3), **xcb-examples(3)**, **xcb_get_atom_name(3)**, **xlsatoms(1)**

AUTHOR

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