

Liboverdose API Documentation
Initial Draft

Copyright © 2009 Kim Gill and Mattis Michel

DOCUMENT PURPOSE

The purpose of this document is to outline the application programming interface (API) for liboverdose.

API

The liboverdose API will consist of a number of functions, and several callbacks.

FUNCTION LISTING

bool **overdose_init**()

This function will set up any global storage and other things required by liboverdose.

bool **overdose_connect**(char *server, int port)

This function will connect to Yahoo! via the given server and port.
If either of these arguments are **NULL**, then sane defaults will be selected.

For server, the default is “scs.msg.yahoo.com”, and for port, the default is 5050.

void ***overdose_get_socket**()

This function returns the handle to the connection socket. This is useful for implementing a client that does not use multiple threads.

Note that you should cast this to whatever the local socket data type is on your OS.
For most UNIX systems, this would be an 'int'.

void ***overdose_get_defaults**(int data)

This function returns a number of default operating parameters for liboverdose.

The following macros may be used:

OVERDOSE_DEF_SERVER – Returns a char* containing the server host.

OVERDOSE_DEF_PORT – Returns an int containing the default port.

OVERDOSE_DEF_ROOM – Returns a char* containing the default room name.

int **overdose_login**(char *username, char *password)

This function instructs liboverdose to attempt to login to Yahoo!. If either of username or password are **NULL**, or are zero-length, this function will return -1.

This function returns 0 on successful login, and 1 if there was an error.

int **overdose_join_room**(char *room)

This function, when called, will cause liboverdose to attempt to join the room specified. If room is NULL, the default of “Linux, FreeBSD, Solaris:1” will be assumed.

Please call **overdose_get_captcha_url()** immediately afterwards.

This function will return 0 on success, and -1 on failure.

YEvent ***overdose_get_event()**

This function returns a event from the Yahoo! Chat server. This can be a list of users, a chat message or emote, the CAPTCHA URL, PMs, or any number of things.

This returns a pointer to an event, or NULL if not connected.

Note that this function uses blocking I/O, and will block the thread it is called from. This is useful if you want to implement a multi-threaded interface, however, if you are using a select()-based interface, this will lock your program until an event is received.

DATA TYPES OVERVIEW

YEvent: Specifies an event as received from the Yahoo! Chat server.

The C-language format of this structure is as follows:

```
struct YEvent {
    int type;
    char *user;
    char *message;
    int flags;
}
```