

NAME

`curl_multi_timeout` – how long to wait for action before proceeding

SYNOPSIS

```
#include <curl/curl.h>
```

```
CURLMcode curl_multi_timeout(CURLM *multi_handle, long *timeout);
```

DESCRIPTION

An application using the libcurl multi interface should call **`curl_multi_timeout(3)`** to figure out how long it should wait for socket actions – at most – before proceeding.

Proceeding means either doing the socket-style timeout action: call the **`curl_multi_socket(3)`** function with the **`sockfd`** argument set to `CURL_SOCKET_TIMEOUT` and the **`easy`** argument set to `CURL_EASY_TIMEOUT`, or simply calling **`curl_multi_perform(3)`** if you're using the simpler and older multi interface approach.

The timeout value returned in the long **`timeout`** points to, is in number of milliseconds at this very moment. If 0, it means you should proceed immediately without waiting for anything. If it returns -1, there's no timeout at all set.

RETURN VALUE

The standard CURLMcode for multi interface error codes.

TYPICAL USAGE

Call **`curl_multi_timeout(3)`**, then wait for action on the sockets. You figure out which sockets to wait for by calling **`curl_multi_fdset(3)`** or by a previous call to **`curl_multi_socket(3)`**.

AVAILABILITY

This function was added in libcurl 7.15.4, although not deemed stable yet.

SEE ALSO

`curl_multi_cleanup(3)`, **`curl_multi_init(3)`**, **`curl_multi_fdset(3)`**, **`curl_multi_info_read(3)`**,
`curl_multi_socket(3)`