

NAME

CURLMOPT_TIMERDATA – custom pointer to pass to timer callback

SYNOPSIS

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

```
CURLMcode curl_multi_setopt(CURLM *handle, CURLMOPT_TIMERDATA, void *pointer);
```

DESCRIPTION

A data **pointer** to pass to the timer callback set with the *CURLMOPT_TIMERFUNCTION(3)* option.

This pointer is not touched by libcurl but is only be passed in to the timer callback's **clientp** argument.

DEFAULT

NULL

PROTOCOLS

This functionality affects all supported protocols

EXAMPLE

```
struct priv {
    void *custom;
};

static int timerfunc(CURLM *multi, long timeout_ms, void *clientp)
{
    struct priv *mydata = clientp;
    printf("our ptr: %p\n", mydata->custom);

    if(timeout_ms >= 0) {
        /* this is the new single timeout to wait for */
    }
    else {
        /* delete the timeout, nothing to wait for now */
    }
    return 0;
}

int main(void)
{
    struct priv mydata;
    CURLM *multi = curl_multi_init();
    curl_multi_setopt(multi, CURLMOPT_TIMERFUNCTION, timerfunc);
    curl_multi_setopt(multi, CURLMOPT_TIMERDATA, &mydata);
}
```

AVAILABILITY

Added in curl 7.16.0

RETURN VALUE

curl_multi_setopt(3) returns a CURLMcode indicating success or error.

CURLM_OK (0) means everything was OK, non-zero means an error occurred, see *libcurl-errors(3)*.

SEE ALSO

CURLMOPT_SOCKETFUNCTION(3), CURLMOPT_TIMERFUNCTION(3)