

**NAME**

curl\_multi\_fdset - extracts file descriptor information from a multi handle

**SYNOPSIS**

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

```
CURLMcode curl_multi_fdset(CURLM *multi_handle,  
                           fd_set *read_fd_set,  
                           fd_set *write_fd_set,  
                           fd_set *exc_fd_set,  
                           int *max_fd);
```

**DESCRIPTION**

This function extracts file descriptor information from a given multi\_handle. libcurl returns its fd\_set sets. The application can use these to select() on, but be sure to FD\_ZERO them before calling this function as *curl\_multi\_fdset(3)* only adds its own descriptors, it doesn't zero or otherwise remove any others. The *curl\_multi\_perform(3)* function should be called as soon as one of them is ready to be read from or written to.

If no file descriptors are set by libcurl, *max\_fd* will contain -1 when this function returns. Otherwise it will contain the higher descriptor number libcurl set.

You should also be aware that when doing select(), you should consider using a rather small (single-digit number of seconds) timeout and call *curl\_multi\_perform* regularly - even if no activity has been seen on the fd\_sets - as otherwise libcurl-internal retries and timeouts may not work as you'd think and want.

Starting with libcurl 7.16.0, you should use **curl\_multi\_timeout** to figure out how long to wait for action.

**RETURN VALUE**

CURLMcode type, general libcurl multi interface error code. See *libcurl-errors(3)*

**SEE ALSO**

**curl\_multi\_cleanup(3)**, **curl\_multi\_init(3)**, **curl\_multi\_timeout(3)**