NAME

curl_multi_remove_handle - remove an easy handle from a multi session

SYNOPSIS

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

CURLMcode curl_multi_remove_handle(CURLM *multi_handle, CURL *easy_handle);
```

DESCRIPTION

Removes a given *easy_handle* from the *multi_handle*. This makes the specified easy handle be removed from this multi handle's control.

When the easy handle has been removed from a multi stack, it is again perfectly legal to invoke *curl_easy_perform(3)* on this easy handle.

Removing an easy handle while being in use is perfectly legal and effectively halts the transfer in progress involving that easy handle. All other easy handles and transfers remain unaffected.

It is fine to remove a handle at any time during a transfer, just not from within any libcurl callback function.

Removing an easy handle from the multi handle before the corresponding transfer is complete might cause libcurl to close the connection - if the state of it and the internal protocol handler deem it necessary. Otherwise libcurl keeps the connection alive in the connection pool associated with the multi handle, ready to get reused for a future transfer using this multi handle.

PROTOCOLS

This functionality affects all supported protocols

EXAMPLE

```
int main(void)
{
 CURLM *multi = curl_multi_init();
 int queued = 0;

 /* when an easy handle has completed, remove it */
 CURLMsg *msg = curl_multi_info_read(multi, &queued);
 if(msg) {
  if(msg->msg == CURLMSG_DONE) {
   /* a transfer ended */
```

```
      fprintf(stderr, "Transfer completed\n");
      curl_multi_remove_handle(multi, msg->easy_handle);
    }
  }
}
```

AVAILABILITY

Added in curl 7.9.6

RETURN VALUE

CURLMcode type, general libcurl multi interface error code.

SEE ALSO

curl_multi_add_handle(3), **curl_multi_cleanup**(3), **curl_multi_init**(3)