

NAME

`ibv_reg_mr`, `ibv_dereg_mr` - register or deregister a memory region (MR)

SYNOPSIS

```
#include <infiniband/verbs.h>
```

```
struct ibv_mr *ibv_reg_mr(struct ibv_pd *pd, void *addr,  
                          size_t length, int access);
```

```
int ibv_dereg_mr(struct ibv_mr *mr);
```

DESCRIPTION

`ibv_reg_mr()` registers a memory region (MR) associated with the protection domain *pd*. The MR's starting address is *addr* and its size is *length*. The argument *access* describes the desired memory protection attributes; it is either 0 or the bitwise OR of one or more of the following flags:

IBV_ACCESS_LOCAL_WRITE Enable Local Write Access

IBV_ACCESS_REMOTE_WRITE Enable Remote Write Access

IBV_ACCESS_REMOTE_READ Enable Remote Read Access

IBV_ACCESS_REMOTE_ATOMIC Enable Remote Atomic Operation Access (if supported)

IBV_ACCESS_MW_BIND Enable Memory Window Binding

IBV_ACCESS_ON_DEMAND Create an on-demand paging MR

If **IBV_ACCESS_REMOTE_WRITE** or **IBV_ACCESS_REMOTE_ATOMIC** is set, then **IBV_ACCESS_LOCAL_WRITE** must be set too.

Local read access is always enabled for the MR.

`ibv_dereg_mr()` deregisters the MR *mr*.

RETURN VALUE

`ibv_reg_mr()` returns a pointer to the registered MR, or NULL if the request fails. The local key (**L_Key**) field **lkey** is used as the lkey field of struct `ibv_sge` when posting buffers with `ibv_post_*` verbs, and the the remote key (**R_Key**) field **rkey** is used by remote processes to perform Atomic and RDMA operations. The remote process places this **rkey** as the rkey field of struct `ibv_send_wr` passed

to the `ibv_post_send` function.

`ibv_dereg_mr()` returns 0 on success, or the value of `errno` on failure (which indicates the failure reason).

NOTES

`ibv_dereg_mr()` fails if any memory window is still bound to this MR.

SEE ALSO

`ibv_alloc_pd(3)`, **`ibv_post_send(3)`**, **`ibv_post_recv(3)`**, **`ibv_post_srq_recv(3)`**

AUTHORS

Dotan Barak <dotanba@gmail.com>