NAME

ibv_bind_mw – post a request to bind a type 1 memory window to a memory region

SYNOPSIS

#include <infiniband/verbs.h>

int ibv_bind_mw(struct ibv_qp **qp*, **struct ibv_mw** **mw*,
struct ibv_mw_bind **mw_bind*);

DESCRIPTION

ibv_bind_mw() posts to the queue pair *qp* a request to bind the memory window *mw* according to the details in *mw_bind*.

The argument *mw_bind* is an ibv_mw_bind struct, as defined in <infiniband/verbs.h>.

```
struct ibv_mw_bind {
        uint64_t                wr_id;      /* User defined WR ID */
        int                     send_flags; /* Use ibv_send_flags */
        struct ibv_mw_bind_info bind_info;  /* MW bind information */
}

struct ibv_mw_bind_info {
        struct ibv_mr           *mr;        /* The MR to bind the MW to */
        uint64_t                addr;       /* The address the MW should start at */
        uint64_t                length;     /* The length (in bytes) the MW should span */
        int                     mw_access_flags; /* Access flags to the MW. Use ibv_access_flags */
};
```

The QP Transport Service Type must be either UC, RC or XRC_SEND for bind operations.

The attribute send_flags describes the properties of the WR. It is either 0 or the bitwise OR of one or more of the following flags:

IBV_SEND_FENCE Set the fence indicator.

IBV_SEND_SIGNALED Set the completion notification indicator. Relevant only if QP was created with sq_sig_all=0

The mw_access_flags define the allowed access to the MW after the bind completes successfully. It is either 0 or the bitwise OR of one or more of the following flags:

IBV_ACCESS_REMOTE_WRITE Enable Remote Write Access. Requires local write access to the MR.

IBV_ACCESS_REMOTE_READ Enable Remote Read Access

IBV_ACCESS_REMOTE_ATOMIC Enable Remote Atomic Operation Access (if supported). Requires local write access to the MR.

IBV_ACCESS_ZERO_BASED If set, the address set on the 'remote_addr' field on the WR will be an offset from the MW's start address.

RETURN VALUE

ibv_bind_mw() returns 0 on success, or the value of errno on failure (which indicates the failure reason). In case of a success, the R_key of the memory window after the bind is returned in the mw_bind->mw->rkey field.

NOTES

The bind does not complete when the function return - it is merely posted to the QP. The user should keep a copy of the old R_key, and fix the mw structure if the subsequent CQE for the bind operation indicates a failure. The user may safely send the R_key using a send request on the same QP, (based on QP ordering rules: a send after a bind request on the same QP are always ordered), but must not transfer it to the remote in any other manner before reading a successful CQE.

Note that for type 2 MW, one should directly post bind WR to the QP, using ibv_post_send.

SEE ALSO

ibv_alloc_mw(3), **ibv_post_send**(3), **ibv_poll_cq**(3) **ibv_reg_mr**(3),

AUTHORS

Majd Dibbiny <majd@mellanox.com>

Yishai Hadas <yishaih@mellanox.com>