

`mlx5dv_devx_create_cmd_comp, mlx5dv_devx_destroy_cmd_comp, mlx5dv_devx_get_async(3) destroy_cmd_comp, get_async(3)`

NAME

`mlx5dv_devx_create_cmd_comp` – Create a command completion to be used for DEVX asynchronous commands.

`mlx5dv_devx_destroy_cmd_comp` – Destroy a devx command completion.

`mlx5dv_devx_get_async_cmd_comp` – Get an asynchronous command completion. # SYNOPSIS

```
#include <infiniband/mlx5dv.h>

struct mlx5dv_devx_cmd_comp {
    int fd;
};

struct mlx5dv_devx_cmd_comp *
mlx5dv_devx_create_cmd_comp(struct ibv_context *context)

void mlx5dv_devx_destroy_cmd_comp(struct mlx5dv_devx_cmd_comp *cmd_comp)

struct mlx5dv_devx_async_cmd_hdr {
    uint64_t    wr_id;
    uint8_t     out_data[];
};

int mlx5dv_devx_get_async_cmd_comp(struct mlx5dv_devx_cmd_comp *cmd_comp,
                                   struct mlx5dv_devx_async_cmd_hdr *cmd_resp,
                                   size_t cmd_resp_len)
```

DESCRIPTION

Create or destroy a command completion to be used for DEVX asynchronous commands.

The create verb exposes an `mlx5dv_devx_cmd_comp` object that can be used as part of asynchronous DEVX commands. This lets an application run asynchronously without blocking and once the response is ready read it from this object.

The response can be read by the `mlx5dv_devx_get_async_cmd_comp()` API, upon response the `wr_id` that was supplied upon the asynchronous command is returned and the `out_data` includes the data itself. The application must supply a large enough buffer to match any command that was issued on the `cmd_comp`, its size is given by the input `cmd_resp_len` parameter.

ARGUMENTS

context

RDMA device context to create the action on.

cmd_comp

The command completion object.

cmd_resp

The output data from the asynchronous command.

cmd_resp_len

The output buffer size to hold the response.

RETURN VALUE

Upon success `mlx5dv_devx_create_cmd_comp` will return a new `struct mlx5dv_devx_cmd_comp` object, on error NULL will be returned and `errno` will be set.

Upon success `mlx5dv_devx_get_async_cmd_comp` will return 0, otherwise `errno` will be returned.

SEE ALSO

`mlx5dv_open_device(3)`, `mlx5dv_devx_obj_create(3)`

```
mlx5dv_devx_create_cmd_comp, mlx5dv_devx_create_cmd_comp, get_async(3) destroy_cmd_comp, get_async(3)
```

#AUTHOR

Yishai Hadas <yishaih@mellanox.com>