NAME

mlx5dv_create_cq − creates a completion queue (CQ)

SYNOPSIS

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

struct ibv_cq_ex *mlx5dv_create_cq(struct ibv_context *context,
                                   struct ibv_cq_init_attr_ex *cq_attr,
                                   struct mlx5dv_cq_init_attr *mlx5_cq_attr);
```

DESCRIPTION

mlx5dv_create_cq() creates a completion queue (CQ) with specific driver properties.

ARGUMENTS

Please see **ibv_create_cq_ex(3)** man page for **context** and **cq_attr**

mlx5_cq_attr

```
struct mlx5dv_cq_init_attr {
    uint64_t comp_mask;
    uint8_t  cqe_comp_res_format;
    uint32_t flags;
    uint16_t cqe_size;
};
```

comp_mask

Bitmask specifying what fields in the structure are valid:

MLX5DV_CQ_INIT_ATTR_MASK_COMPRESSED_CQE enables creating a CQ in a mode that few CQEs may be compressed into a single CQE, valid values in *cqe_comp_res_format*

MLX5DV_CQ_INIT_ATTR_MASK_FLAGS valid values in *flags*

MLX5DV_CQ_INIT_ATTR_MASK_CQE_SIZE valid values in *cqe_size*

cqe_comp_res_format

A bitwise OR of the various CQE response formats of the responder side:

MLX5DV_CQE_RES_FORMAT_HASH CQE compression with hash

MLX5DV_CQE_RES_FORMAT_CSUM CQE compression with RX checksum

MLX5DV_CQE_RES_FORMAT_CSUM_STRIDX CQE compression with stride index

flags A bitwise OR of the various values described below:

MLX5DV_CQ_INIT_ATTR_FLAGS_CQE_PAD create a padded 128B CQE

cqe_size

configure the CQE size to be 64 or 128 bytes other values will fail mlx5dv_create_cq.

RETURN VALUE

mlx5dv_create_cq() returns a pointer to the created CQ, or NULL if the request fails and errno will be set.

SEE ALSO

ibv_create_cq_ex(3),

AUTHOR

Yonatan Cohen <yonatanc@mellanox.com>