NAME

Peer Context Memory Access −

Functions

CUresult cuCtxDisablePeerAccess (**CUcontext** peerContext)
Disables direct access to memory allocations in a peer context and unregisters any registered allocations.

CUresult cuCtxEnablePeerAccess (**CUcontext** peerContext, unsigned int Flags)
Enables direct access to memory allocations in a peer context.

CUresult cuDeviceCanAccessPeer (int *canAccessPeer, **CUdevice** dev, **CUdevice** peerDev)
Queries if a device may directly access a peer device's memory.

Detailed Description

\brief direct peer context memory access functions of the low-level CUDA driver API (**cuda.h**)

This section describes the direct peer context memory access functions of the low-level CUDA driver application programming interface.

Function Documentation

CUresult cuCtxDisablePeerAccess (CUcontext peerContext)

Returns **CUDA_ERROR_PEER_ACCESS_NOT_ENABLED** if direct peer access has not yet been enabled from `peerContext` to the current context.

Returns **CUDA_ERROR_INVALID_CONTEXT** if there is no current context, or if `peerContext` is not a valid context.

Parameters:
peerContext - Peer context to disable direct access to

Returns:
CUDA_SUCCESS, **CUDA_ERROR_DEINITIALIZED**,
CUDA_ERROR_NOT_INITIALIZED,
CUDA_ERROR_PEER_ACCESS_NOT_ENABLED,
CUDA_ERROR_INVALID_CONTEXT,

Note:
Note that this function may also return error codes from previous, asynchronous launches.

See also:
cuDeviceCanAccessPeer, **cuCtxEnablePeerAccess**

CUresult cuCtxEnablePeerAccess (CUcontext peerContext, unsigned int Flags)

If both the current context and `peerContext` are on devices which support unified addressing (as may be queried using **CU_DEVICE_ATTRIBUTE_UNIFIED_ADDRESSING**) and same major compute capability, then on success all allocations from `peerContext` will immediately be accessible by the current context. See **Unified Addressing** for additional details.

Note that access granted by this call is unidirectional and that in order to access memory from the current context in `peerContext`, a separate symmetric call to **cuCtxEnablePeerAccess()** is required.

Returns **CUDA_ERROR_PEER_ACCESS_UNSUPPORTED** if **cuDeviceCanAccessPeer()** indicates that the **CUdevice** of the current context cannot directly access memory from the **CUdevice** of `peerContext`.

Returns **CUDA_ERROR_PEER_ACCESS_ALREADY_ENABLED** if direct access of `peerContext` from the current context has already been enabled.

Returns **CUDA_ERROR_TOO_MANY_PEERS** if direct peer access is not possible because hardware resources required for peer access have been exhausted.

Returns **CUDA_ERROR_INVALID_CONTEXT** if there is no current context, `peerContext` is not a valid context, or if the current context is `peerContext`.

Returns **CUDA_ERROR_INVALID_VALUE** if `Flags` is not 0.

Parameters:
peerContext - Peer context to enable direct access to from the current context
Flags - Reserved for future use and must be set to 0

Returns:

CUDA_SUCCESS, **CUDA_ERROR_DEINITIALIZED**, **CUDA_ERROR_NOT_INITIALIZED**, **CUDA_ERROR_PEER_ACCESS_ALREADY_ENABLED**, **CUDA_ERROR_TOO_MANY_PEERS**, **CUDA_ERROR_INVALID_CONTEXT**, **CUDA_ERROR_PEER_ACCESS_UNSUPPORTED**, **CUDA_ERROR_INVALID_VALUE**

Note:

Note that this function may also return error codes from previous, asynchronous launches.

See also:

cuDeviceCanAccessPeer, **cuCtxDisablePeerAccess**

CUresult cuDeviceCanAccessPeer (int * canAccessPeer, CUdevice dev, CUdevice peerDev)

Returns in `*canAccessPeer` a value of 1 if contexts on `dev` are capable of directly accessing memory from contexts on `peerDev` and 0 otherwise. If direct access of `peerDev` from `dev` is possible, then access may be enabled on two specific contexts by calling **cuCtxEnablePeerAccess()**.

Parameters:

canAccessPeer - Returned access capability
dev - Device from which allocations on `peerDev` are to be directly accessed.
peerDev - Device on which the allocations to be directly accessed by `dev` reside.

Returns:

CUDA_SUCCESS, **CUDA_ERROR_DEINITIALIZED**, **CUDA_ERROR_NOT_INITIALIZED**, **CUDA_ERROR_INVALID_DEVICE**

Note:

Note that this function may also return error codes from previous, asynchronous launches.

See also:

cuCtxEnablePeerAccess, **cuCtxDisablePeerAccess**

Author

Generated automatically by Doxygen from the source code.