Class CudaStreamHandler

Table of contents

- Defined in File cuda_stream_handler.hpp

Class Documentation

class CudaStreamHandler

This class handles usage of CUDA streams for operators.

When using CUDA operations the default stream '0' synchronizes with all other streams in the same context, see https://docs.nvidia.com/cuda/cuda-runtime-api/stream-sync-behavior.html#stream-sync-behavior. This can reduce performance. The CudaStreamHandler class manages streams across operators and makes sure that CUDA operations are properly chained.

Usage:

- add an instance of CudaStreamHandler to your operator
- call CudaStreamHandler::registerInterface() from the operator registerInterface() function
- in the tick() function call CudaStreamHandler::fromMessage(), this will get the CUDA stream from the message of the previous operator. When the operator receives multiple messages, then call CudaStreamHandler::fromMessages(). This will synchronize with multiple streams.
- when executing CUDA functions CudaStreamHandler::get() to get the CUDA stream which should be used by your CUDA function
- before publishing the output message(s) of your operator call CudaStreamHandler::toMessage() on each message. This will add the CUDA stream used by the CUDA functions in your operator to the output message.

Public Functions

inline ~CudaStreamHandler()

Destroy the CudaStreamHandler object.

inline gxf::Expected<void> registerInterface(gxf::Registrar *registrar, bool required = false)

Register the parameters used by this class.

Parameters

- **registrar** –
- **required** – if set then it's required that the CUDA stream pool is specified

Returns

gxf::Expected<void>

inline gxf_result_t fromMessage(gxf_context_t context, const nvidia::gxf::Expected<nvidia::gxf::Entity> &message)

Get the CUDA stream for the operation from the incoming message

Parameters

- **context** –
- **message** –

Returns

gxf_result_t

inline gxf_result_t fromMessages(gxf_context_t context, const std::vector<nvidia::gxf::Entity> &messages)

Get the CUDA stream for the operation from the incoming messages

Parameters

- **context** –
- **messages** –

Returns

gxf_result_t

inline gxf_result_t toMessage(nvidia::gxf::Expected<nvidia::gxf::Entity> &message)

Add the used CUDA stream to the outgoing message.

Parameters

message –

Returns

gxf_result_t

inline gxf::Handle<gxf::CudaStream> getStreamHandle()

Get the CUDA stream handle which should be used for CUDA commands

Returns

gxf::Handle<gxf::CudaStream>

inline cudaStream_t getCudaStream()

Get the CUDA stream which should be used for CUDA commands.

If no message stream is set and no stream can be allocated, return the default stream.

Returns

cudaStream_t