Class GXFInputContext

Table of contents

- Defined in File gxf_io_context.hpp

Inheritance Relationships

Base Type

- `public holoscan::InputContext` (Class InputContext)

Class Documentation

class GXFInputContext : public holoscan::InputContext

> Class to hold the input context for a GXF Operator.
>
> This class provides the interface to receive the input data from the operator using GXF.
>
> Public Functions
>
> GXFInputContext(ExecutionContext *execution_context, Operator *op)
>
>> Construct a new GXFInputContext object.
>>
>> Parameters
>>
>> - **execution_context** – The pointer to the execution context.
>>
>> - **op** – The pointer to the GXFOperator object.
>
> GXFInputContext(ExecutionContext *execution_context, Operator *op, std::unordered_map<std::string, std::shared_ptr<IOSpec>> &inputs)
>
>> Construct a new GXFInputContext object.
>>
>> Parameters
>>
>> - **execution_context** – The pointer to the execution context.
>>
>> - **op** – The pointer to the GXFOperator object.

- **inputs** – inputs The references to the map of the input specs.

gxf_context_t gxf_context() const

> Get a pointer to the GXF execution runtime.

> Returns

> The pointer to the GXF context.

Protected Functions

virtual bool empty_impl(const char *name = nullptr) override

> The implementation of the `empty` method.

> Parameters

> **name** – The name of the input port

> Returns

> True if the input port is empty or by default. Otherwise, false.

virtual std::any receive_impl(const char *name = nullptr, bool no_error_message = false) override

> The implementation of the `receive` method.

> Depending on the type of the data, this method receives a message from the input port with the given name.

> Parameters

> - **name** – The name of the input port.

> - **no_error_message** – Whether to print an error message when the input port is not found.

> Returns

The data received from the input port.