



## **Class ComponentBase**

# Table of contents

Inheritance Relationships

---

Class Documentation

---

- Defined in [File component.hpp](#)

## Inheritance Relationships

### Derived Types

- `public holoscan::Component` ([Class Component](#))
- `public holoscan::Operator` ([Class Operator](#))

## Class Documentation

class ComponentBase

Base class for all components.

This class is the base class for all components including `holoscan::Operator`, `holoscan::Condition`, and `holoscan::Resource`. It is used to define the common interface for all components.

Subclassed by [holoscan::Component](#), [holoscan::Operator](#)

Public Functions

ComponentBase() = default

```
template<typename ArgT, typename ...ArgsT, typename =  
std::enable_if_t<!std::is_base_of_v<::holoscan::ComponentBase, std::decay_t<ArgT>> &&  
(std::is_same_v<::holoscan::Arg, std::decay_t<ArgT>> || std::is_same_v<::holoscan::ArgList,  
std::decay_t<ArgT>>>>>  
inline explicit ComponentBase(ArgT &&arg, ArgsT&&... args)
```

Construct a new [Component](#) object.

Parameters

**args** – The arguments to be passed to the component.

virtual ~ComponentBase() = default

`inline int64_t id() const`

Get the identifier of the component.

By default, the identifier is set to -1. It is set to a valid value when the component is initialized.

With the default executor (GXFExecutor), the identifier is set to the GXF component ID.

Returns

The identifier of the component.

`inline const std::string &name() const`

Get the name of the component.

Returns

The name of the component.

`inline Fragment *fragment()`

Get a pointer to [Fragment](#) object.

Returns

The Pointer to [Fragment](#) object.

`inline void add_arg(const Arg &arg)`

Add an argument to the component.

Parameters

**arg** – The argument to add.

`inline void add_arg(Arg &&arg)`

Add an argument to the component.

Parameters

**arg** – The argument to add.

```
inline void add_arg(const ArgList &arg)
```

Add a list of arguments to the component.

Parameters

**arg** – The list of arguments to add.

```
inline void add_arg(ArgList &&arg)
```

Add a list of arguments to the component.

Parameters

**arg** – The list of arguments to add.

```
inline std::vector<Arg> &args()
```

Get the list of arguments.

Returns

The vector of arguments.

```
inline virtual void initialize()
```

Initialize the component.

This method is called only once when the component is created for the first time, and use of light-weight initialization.

```
virtual YAML::Node to_yaml_node() const
```

Get a YAML representation of the component.

Returns

YAML node including the id, name, fragment name, and arguments of the component.

```
std::string description() const
```

Get a description of the component.

to\_yaml\_node()

Returns

YAML string.

Protected Functions

void update\_params\_from\_args(std::unordered\_map<std::string, ParameterWrapper> &params)

Update parameters based on the specified arguments.

virtual void reset\_graph\_entities()

Reset the GXF GraphEntity of any arguments that have one.

Protected Attributes

int64\_t id\_ = -1

The ID of the component.

std::string name\_ = ""

Name of the component.

Fragment \*fragment\_ = nullptr

Pointer to the fragment that owns this component.

std::vector<Arg> args\_

List of arguments.

Friends

*friend class* holoscan::Executor

*friend class* holoscan::gfx::GXFExecutor

*friend class* holoscan::Fragment

© Copyright 2022-2024, NVIDIA.. PDF Generated on 06/06/2024