



Program Listing for File gxf_wrapper.hpp

[Return to documentation for file \(include/holoscan/core/gxf/gxf_wrapper.hpp\)](#)

```
/* * SPDX-FileCopyrightText: Copyright (c) 2022-2024 NVIDIA CORPORATION &
AFFILIATES. All rights reserved. * SPDX-License-Identifier: Apache-2.0 * * Licensed
under the Apache License, Version 2.0 (the "License"); * you may not use this file
except in compliance with the License. * You may obtain a copy of the License at * *
http://www.apache.org/licenses/LICENSE-2.0 * * Unless required by applicable law
or agreed to in writing, software * distributed under the License is distributed on an
"AS IS" BASIS, * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express or implied. * See the License for the specific language governing
permissions and * limitations under the License. */ #ifndef
HOLOSCAN_CORE_GXF_GXF_WRAPPER_HPP #define
HOLOSCAN_CORE_GXF_GXF_WRAPPER_HPP #include
"holoscan/core/gxf/gxf_operator.hpp" #include "gxf/std/codelet.hpp" #include
"gxf/core/parameter_parser_std.hpp" namespace holoscan::gxf { class GXFWrapper
: public nvidia::gxf::Codelet { public: virtual ~GXFWrapper() = default; gxf_result_t
initialize() override; gxf_result_t deinitialize() override; gxf_result_t
registerInterface(nvidia::gxf::Registrar* registrar) override; gxf_result_t start()
override; gxf_result_t tick() override; gxf_result_t stop() override; void
set_operator(Operator* op) { op_ = op; } private: void store_exception(); Operator*
op_ = nullptr; }; } // namespace holoscan::gxf #endif/*
HOLOSCAN_CORE_GXF_GXF_WRAPPER_HPP */
```

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