



**Program Listing for File  
annotated\_double\_buffer\_receiver.hpp**

[Return to documentation for file \(](#)

`include/holoscan/core/resources/gxf/annotated_double_buffer_receiver.hpp` )

```
/* * SPDX-FileCopyrightText: Copyright (c) 2023 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_RESOURCES_GXF_ANNOTATED_DOUBLE_BUFFER_RECEIVER_HPP
#define
HOLOSCAN_CORE_RESOURCES_GXF_ANNOTATED_DOUBLE_BUFFER_RECEIVER_HPP
#include <gxf/std/double_buffer_receiver.hpp> #include
<gxf/core/component.hpp> #include <gxf/core/entity.hpp> #include
<gxf/core/handle.hpp> namespace holoscan { // Forward declarations class
Operator; class AnnotatedDoubleBufferReceiver : public
nvidia::gxf::DoubleBufferReceiver { public: AnnotatedDoubleBufferReceiver() =
default; gxf_result_t receive_abi(gxf_uid_t* uid); holoscan::Operator* op() { return
op_; } void op(holoscan::Operator* op) { this->op_ = op; } private:
holoscan::Operator* op_ = nullptr; }; } // namespace holoscan #endif/*
HOLOSCAN_CORE_RESOURCES_GXF_ANNOTATED_DOUBLE_BUFFER_RECEIVER_HPP
*/
```

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