



Program Listing for File `std_entity_serializer.hpp`

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

`include/holoscan/core/resources/gxf/std_entity_serializer.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_RESOURCES_GXF_STD_ENTITY_SERIALIZER_HPP #define
HOLOSCAN_CORE_RESOURCES_GXF_STD_ENTITY_SERIALIZER_HPP #include
<memory> #include <vector> #include <gxf/serialization/std_entity_serializer.hpp>
#include "../gxf/gxf_resource.hpp" namespace holoscan { class StdEntitySerializer :
public gxf::GXFRResource { public:
HOLOSCAN_RESOURCE_FORWARD_ARGS_SUPER(StdEntitySerializer, GXFRResource)
StdEntitySerializer() = default; const char* gxf_typename() const override { return
"nvidia::gxf::StdEntitySerializer"; } void setup(ComponentSpec& spec) override; void
initialize() override; nvidia::gxf::StdEntitySerializer* get() const; private:
Parameter<std::vector<std::shared_ptr<holoscan::Resource>>>
component_serializers_; Parameter<bool> verbose_warning_; }; } // namespace
holoscan #endif/* HOLOSCAN_CORE_RESOURCES_GXF_STD_ENTITY_SERIALIZER_HPP
*/
```

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