



Program Listing for File config.hpp

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

```
/* * SPDX-FileCopyrightText: Copyright (c) 2022 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_CONFIG_HPP #define HOLOSCAN_CORE_CONFIG_HPP #include
<filesystem> #include <iostream> #include <string> #include <vector> #include
"./common.hpp" namespace holoscan { class Config { public: Config() = default;
explicit Config(const std::string& config_file, const std::string& prefix = "") :
config_file_(config_file), prefix_(prefix) { if (std::filesystem::exists(config_file)) {
parse_file(config_file); } else if (config_file != "") { HOLOSCAN_LOG_WARN("Config file
'{}' doesn't exist", config_file); } } virtual ~Config() = default; const std::string&
config_file() const { return config_file_; } const std::string& prefix() const { return
prefix_; } const std::vector<YAML::Node>& yaml_nodes() const { return yaml_nodes_;
} private: void parse_file(const std::string& config_file); std::string config_file_;
std::string prefix_; std::vector<YAML::Node> yaml_nodes_; }; } // namespace holoscan
#endif/* HOLOSCAN_CORE_CONFIG_HPP */
```

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