syn1588 ®

syn1588® Software Suite

Upcoming Changes

Version 1.15.1 – April 14th 2022

Oregano Systems – Design & Consulting GesmbH

Franzosengraben 8, A-1030 Vienna

P: +43 (676) 84 31 04-300

@: contact@oregano.at

W: http://oregano.at

0.1 Legals

Oregano Systems does not assume any liability arising out of the application or use of any product described or shown herein nor does it convey any license under its patents, copyrights, or any rights of others.

Licenses or any other rights such as, but not limited to, patents, utility models, trademarks or tradenames, are neither granted nor conveyed by this document, nor does this document constitute any obligation of the disclosing party to grant or convey such rights to the receiving party.

Oregano Systems reserves the right to make changes, at any time without notice, in order to improve reliability, function or design. Oregano Systems will not assume responsibility for the use of any circuitry described herein.

All trademarks used in this document are the property of their respective owners.

0.2 Contents

1 Intend

This document gives an overview of upcoming changes that may require modifications for certain interactions or interfaces with the syn1588® Software Suite. These typically involve larger feature or infrastructure changes (for example the implementation of the standard logger used by the syn1588® Software Suite PTP Stack and utilities). Smaller changes (e.g., bug fixes) are documented in the release notes.

The changes listed in this document include a description, some examples, and a rough roadmap of how and when changes will replace the current features or infrastructure components and some guidelines for migrating your application. This will by far not cover every application of the syn1588® Software Suite, especially for our source code customers. If the proposed migration will not work for your case and you have an upright maintenance agreement with Oregano Systems, please get in touch with our support team (support@oregano.at), so that we can consider your feedback in our planned implementation of the proposed changes or provide an alternate solution.

2 Upcoming Changes

2.1 PTP logger

2.1.1 Overview

The new logger has been introduced with v1.14 of the syn1588® Software Suite. For backwards compatibility the new logger still supports the old log function syntax so that you do not need to update these right away.

The new PTP logger is a complete re-work of the existing logging class. The new default implementation utilizes the libfmt library (current version: v8.0.1; for more information please visit: https://fmt.dev/8.0.1/).

The old functions used to log messages throughout the syn1588® Software Suite (e.g., m_log->Error(…)) are still supported but will be marked as deprecated and will be removed completely in future releases. Some log messages will be promoted/demoted to the new log severity levels. Look at the respective sub chapters below for more information about these changes and how you can migrate your current setup with little or no impact to your application.

Old log functions start with a capital letter (e.g., Error(…) or Debug(…)). These functions use printf syntax for string formatting (by utilizing libfmt printf support).

New log functions start with a lower case letter (e.g., error(…) or trace(…)). These functions use libfmt syntax for string formatting.

2.1.2 Roadmap

2.1.2.1 Modifications for v1.15

This release completes the overhauling of the PTP logging mechanism. Minor updates to single log messages will be added, when necessary, but the big changes are finalized with this release.

The implementation of the new PTP logger will come with the following modifications in this version:

1. The log message output of the core library will be re-factored according to the rule set described in chapter 2.1.5 New concepts.

2.1.2.2 Modifications for future releases

The old log functions are now marked as deprecated. Please familiarize yourself with the new log syntax as used throughout the PTP source code and source code of the API examples.

We will keep the old functions for Release v1.15 but may remove the support in future releases. When we do this, we will inform you prior to the release in this document.

2.1.3 Source code customers

The PTP_Log interface has been re-designed and will have a direct impact on source code projects that have implemented their own logger. You will have to adapt your own code slightly when updating to this source code version of the syn1588® PTP Stack. Refer to the following chapters for an overview of the new setup as well as recommendations for migrating your application.

2.1.4 Binary customers (syn1588 and shared memory API)

The API example applications have been updated to use the new logger. Refer to the following chapters for an overview of the new setup as well as recommendations for migrating your application.

2.1.5 New concepts

2.1.5.1 Log Sink

The new PTP Logger allows to add different log sinks. A log sink implements an interface to a target for the produced log output.

With v1.14 the following log sinks are implemented by default:

- stdout
  This is similar to the behavior of the old PTP logger and will print the log lines to the console.
- File
  This is similar to the behavior of the old PTP logger and will print the log lines to the provided file.
- System
  This is a new log sink that forwards the log output to a general logger of the operating system.
  This is the Event Logger for Windows and syslog for Linux.

2.1.5.2 Logger hierarchy

The new logger provides a layered architecture that matches the layers of the syn1588 library, PTP Stack and utilities. At the top is a root logger. The next layer of syn1588 software components instantiate their own sub-logger which are linked to this root. Layers further down link their own sub-loggers to these loggers, and so forth.

When building this logging hierarchy during the system initialization, every (sub)logger can be configured with a different severity level or may simply inherit the severity level from its parent. Additional log sinks may be added to a sub-logger. Sub-loggers will always inherit the log sinks of all loggers in direct line to the root logger.

2.1.6 New Log severity classification

The current logger implementation distinguishes between 5 verbosity levels with every log message being associated to a specific level. When the verbosity level was increased to more than “2”, the association of a specific message to the respective severity level was difficult and even ambiguous at times. As a result, monitoring and post-mortem analysis did require in-depth knowledge of the inner workings of the PTP stack when tracking more complex operating conditions and failure modes.

We are re-evaluating all log message assignments with the goal to provide an unambiguous association of every log message to the respective severity level. We increased the granularity and introduced the following seven new classifications: “critical”, “error”, “warning”, “info”, “debug”, “trace” and “off”.

The following functions will be used in subsequent versions (starting with version 1.14) to log messages of different severity:

2.1.6.1 critical

- **Trigger Condition:**
  Something went seriously wrong during the operation of the syn1588® PTP Stack or a syn1588 utility and the state is not recoverable
- **System behaviour:**
  The syn1588® PTP Stack or syn1588® Software Suite utility will shut down its operation in a clean way, if possible.
- **Old equivalence:**
  Some log messages on the “Error” severity level have been promoted to the critical severity level.

2.1.6.2 error

- **Trigger Condition:**
  Something went wrong during the operation of the syn1588® PTP Stack or a syn1588 Software Suite utility and will lead to degraded performance but most likely can be recovered. This is most often a result of an external device experiencing problems and these messages indicate that something must be fixed with or at other, external system components.
- **System behaviour:**

The syn1588® PTP Stack or syn1588 Software Suite utility will commence operation in a degraded state and may switch back to normal operation after the error condition has been handled.

- **Old equivalence:**
  Remaining log messages on the “Error” severity level and some log messages promoted from the “Warning” level.

2.1.6.3 warning

- **Trigger Condition:**
  The internal monitoring of the syn1588® PTP Stack or syn1588 Software Suite utilities has detected a potential problem that may lead to a degraded state or even a termination of the operation in the future.
- **System behaviour:**
  The syn1588® PTP Stack or syn1588 Software Suite utility will continue its operation without an immediate degradation of performance.
- **Old equivalence:**
  Some “Warning” messages have been promoted to “error” severity level (as mentioned above).

2.1.6.4 info

- **Trigger Condition:**
  High-level information about the current operation of the syn1588® PTP Stack or syn1588 Software Suite utility.
- **System behaviour:**
  The syn1588® PTP Stack or syn1588 Software Suite utility will continue its operation.
- **Old equivalence:**
  Some “Info” and “Notice” messages stay on this severity level.

2.1.6.5 debug

- **Trigger Condition:**
  Low-level information about the current operation of the syn1588® PTP Stack or syn1588 Software Suite utility. Typically used for analysing customer applications but not required for normal operation.
- **System behaviour:**
  The syn1588® PTP Stack or syn1588 Software Suite utility will continue its operation.
- **Old equivalence:**
  Most of the “Info” and “Notice” messages have been demoted to this level. Some “Debug” messages stay at this level.

2.1.6.6 trace

- **Trigger Condition:**
  Highly detailed information about the current operation of the syn1588® PTP Stack or syn1588 Software Suite utility.
- **System behaviour:**
  The syn1588® PTP Stack or syn1588 Software Suite utility will continue its operation.
- **Old equivalence:**
  Some Debug messages have been demoted to this level.

2.1.6.7 off

Output to log sinks is completely disabled.

2.1.7 Mapping from old to new severity levels

The syn1588® PTP Stack and syn1588® Software Suite utilities will accept both old and new log severity levels, for now, as command line or configuration file values. The old severity levels are marked as deprecated and may be removed completely in future release.

Old verbosity	New severity level	Message classes to be generated
0	off	“off”
1	error	“critical” + “error”
2	info	“critical” + “error” + “warning” + “info”
3	debug	“critical” + “error” + “warning” + “info” + “debug”
4	trace	“critical” + “error” + “warning” + “info” + “debug” + “trace”

The command line parameter for the log level stays the same:
For example, “-v 2” is equivalent to “-v info”.

The config file parameter stays the same:
For example, “loglevel 4” is equivalent to “loglevel trace” and will contain all possible log output.

2.1.8 Mapping PTP Log output to System Log Levels

As the syn1588 Software Suite supports both Windows and Linux OS, the following table gives an overview for mapping PTP log levels to log levels of these operating systems.

As base for Windows, the definitions given in the Microsoft Documentation for the Event Logger (link) are used.

As base for Linux, the definitions given for the syslog system logger (link) are used.

New PTP Logger	Windows	Linux
“off”	-	-
-	-	LOG_EMERG
-	-	LOG_ALERT
“critical”	Error	LOG_CRIT
“error”	Warning	LOG_ERR
“warning”	Warning	LOG_WARNING
“info”	Information	LOG_NOTICE
-	-	LOG_INFO
“debug”	-	LOG_DEBUG
“trace”	-	LOG_DEBUG
-	Success Audit	-
-	Failure Audit	-

2.1.9 Migration recommendations

2.1.9.1 Migrating source code

For backwards compatibility the new logger still supports the old log function syntax so that you do not need to update these in your code base right away.

2.1.9.2 Migrating PTP stack and utility configuration

If you decide to stay with the old log severity levels, you will get the same and more log output per old severity level.

As the old log level parameter values (command line and config file) will be deprecated in the future, we recommend switching to the proper new severity levels as depicted in chapter 2.1.7 Mapping from old to new severity levels.

For new applications, we recommend using only the new severity levels. For existing applications, we recommend updating the configuration to the new severity levels, when a seamless switch to new versions of the syn1588® Software Suite is planned. v1.14 will support both old and new parameter values.

2.1.10 Configuration

The new logger is fully compatible with the configuration of the old logger. You do not have to change your existing configuration files or command line parameter setup for the syn1588 Software Suite PTP Stack and utilities or your own applications utilizing the syn1588 and/or shared memory API.

The following parameters are applicable for configuring the logger:

- Log severity level – New
  Command line: “-v [off, critical, error, warning, info, debug, trace]”
  Configuration File: “loglevel [off, critical, error, warning, info, debug, trace]”
- Log severity level – Deprecated
  Command line: “-v [0,1,2,3,4]”
  Configuration File: “loglevel [0,1,2,3,4]”
- Enable file logging – no changes
  Command line: “-f [file name]”
  Configuration File: “filelog [file name]”
  This will disable logging to the console.
- Enable/Disable logging to console – New
  Command line: not accessible
  Configuration File: “log_to_stdout [true, false]”
  This will enable logging to the console, even when a file logging has been enabled.
- Enable/Disable system logging - New
  Command line: not accessible
  Configuration File: “log_to_sys [true, false]”
  This will enable logging to the system logger of the operating system.
- Enable/Disable system timestamp output for each log line: Removed
  This parameter will be silently ignored.
- Enable/Disable log severity level output for each log line: Removed
  This parameter will be silently ignored.

2.1.10.1 Multi-port syn1588 PTP Stack logging

The syn1588 PTP Stack can operate as a multi-port PTP Stack, for example, as Boundary Clock. Logging to stdout, a log file and the system logger can be enabled/disabled for the whole log output via the first port (lowest index).

We are currently investigating if further segregation of the log output, e.g., port based and/or general log output, is feasible.

2.1.11 Log output examples and comparison

The old logger implementation had a configurable set of standard output components for each log line. You could enable prepending a system clock timestamp and/or a severity class to each line. The index in round brackets signified the port number to differentiate log output in multi-port PTP Stack configurations.
Old log output example (lines are truncated for clarity):

```
2021-12-12 14:33:35.580602 [INFO]    (1) Found stop clock support
2021-12-12 14:33:35.580615 [INFO]    (1) Using MAC TS Version 3160
2021-12-12 14:33:35.580619 [INFO]    (1) Using programmable 1-step TS
…
2021-12-12 14:33:35.580867 [DEBUG]   (1) creating new shared memory
…
2021-12-12 14:33:35.586763 [NOTICE]  (1) Activated SO_TIMESTAMPING…
```

The new logger will prepend the system clock timestamp and severity level by default. In addition, the application layer that generates the log output is identified in a third field. The port number is part of this third field, in the following example “p1” signals a log output generated by PTP port number 1.

New log output example (lines are truncated for clarity):

```
2021-12-12 16:02:34.766078 [DEBUG   ] [syn1588      ]  Checking build…
…
2021-12-12 16:02:34.766234 [DEBUG   ] [p1.clock     ]  LuckyPacketFilter…
…
2021-12-12 16:02:34.766409 [DEBUG   ] [p1.io.sharedmemory.semaphore]  …
…
2021-12-12 16:02:34.768170 [INFO    ] [p1.engine    ]  Settings: Prio2…
```

2.1.12 Libfmt license and future prospect

Currently, libfmt is licensed according to the MIT license:

→ https://github.com/fmtlib/fmt/blob/master/LICENSE.rst.

libfmt was integrated into the official C++ standard with C++20 (std::format). Hence, libfmt has been well tested and applied in a multitude of projects worldwide.

3 Further Information

You are looking for further information about our syn1588® product line-up? Please contact Oregano Systems support! We will be pleased to provide you all the required information.

Oregano Systems
A Meinberg Company

Franzosengraben 8
A-1030 Vienna
AUSTRIA
https://oreganosystems.at
support@oregano.at