|  |  |
| --- | --- |
|  |  |

EPTF CLL Scheduler, User Guide

Contents

[1 Introduction 2](#_Toc437000196)

[1.1 Revision history 2](#_Toc437000197)

[1.2 How to read this document 2](#_Toc437000198)

[1.3 Abbreviations 2](#_Toc437000199)

[1.3.1 Terminology 2](#_Toc437000200)

[1.4 System requirements 3](#_Toc437000201)

[2 Scheduler 3](#_Toc437000202)

[2.1 Overview 3](#_Toc437000203)

[2.2 Red Black Tree Scheduler 3](#_Toc437000204)

[2.3 Description of files in this feature 3](#_Toc437000205)

[2.4 Description of required files from other features 3](#_Toc437000206)

[2.5 Installation 4](#_Toc437000207)

[2.6 Configuration 4](#_Toc437000208)

[2.7 DTE handling 5](#_Toc437000209)

[2.8 Load measurement 5](#_Toc437000210)

[3 Error messages 5](#_Toc437000211)

[4 Warning messages 7](#_Toc437000212)

[5 Examples 7](#_Toc437000213)

[5.1 Demo module 7](#_Toc437000214)

[6 References 7](#_Toc437000215)

# Introduction

## Revision history

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Rev | Characteristics | Prepared |
| 2007-12-01 | PA1 | First draft version | EGBOTAT |
| 2007-12-06 | PA2 | Final version after review | EGBOTAT |
| 2011-09-29 | PB1 | FBQ removed | ETHJGI |
| 2014-10-08 | PC1 | Added section DTE Handling. | EDORNAG |
| 2014-11-21 | PC2 | Added comment on DTE handling | ESZILSZ |
| 2014-12-02 | C | Updated for release | ESZILSZ |
| 2015-08-12 | PD1 | Load measurement added | EZOLZSI |
| 2015-08-17 | PD2 | Reference chapter updated | ESZILSZ |
| 2015-12-03 | D | Updated for release | ESZILSZ |

## How to read this document

This is the User Guide for the Scheduler of the Ericsson Performance Test Framework (EPTF), Core Library (CLL). EPTF CLL is developed for the TTCN-3 [1] Toolset with TITAN [2]. This document should be read together with the Function Description of the Scheduler feature [6]. For more information on the EPTF CLL, please consult the Product Revision Information [3], the User Guide [4] and the Function Specification [5] of the TitanSim.

## Abbreviations

API Application Programming Interface

CLL Core Library

EPTF Ericsson Load Test Framework, formerly TITAN Load Test Framework

TitanSim Ericsson Load Test Framework, formerly TITAN Load Test Framework

TTCN-3 Testing and Test Control Notation version 3 [1]

### Terminology

*EPTF Core Library (CLL)* is that part of the TitanSim software that is totally project independent. (That is, which is not protocol-, or application-dependent). The EPTF CLL is to be supplied and supported by the TCC organization. Any EPTF CLL development is to be funded centrally by Ericsson.

*Red Black Tree* The EPTF Red Black Tree feature is a self-balancing binary search tree. It can be used to store indexes into an associated array.

## System requirements

In order to use the Scheduler feature the system requirements listed in EPTF CLL User Guide [5] should be fulfilled.

# Scheduler

## Overview

The EPTF CLL Scheduler component is a fundamental component providing an implementation for scheduling in a load test environment.

The EPTF Scheduler feature makes it possible to schedule events (actions) that should happen at a given time measured from the start of the component or relative to the schedule of the event. The Scheduler uses only two TTCN-3 timers to implement scheduling. These are the component clock – which is started at initialization – and the next event timer.

## Red Black Tree Scheduler

The EPTF Scheduler feature contains an event queue which is based on Red Black Tree [7].

The Red Black Tree can insert, delete and search elements in O(log n) time.

## Description of files in this feature

The EPTF CLL Scheduler API includes the following files:

* Scheduler
  + EPTF\_CLL\_Scheduler\_Definitions.ttcn: This TTCN-3 module contains common type definitions of the Scheduler Component.
  + EPTF\_CLL\_RBTScheduler\_Definitions.ttcn: This TTCN-3 module contains type definitions for the Red Black Tree based Scheduler
  + EPTF\_CLL\_RBTScheduler\_Functions.ttcn: This TTCN-3 modlue contains function definitions for the Red Black Tree based Scheduler.

## Description of required files from other features

The Scheduler feature is part of the TitanSim EPTF Core Library (CLL). It relies on several features of the CLL. To use the Scheduler the user has to obtain the respective files from the following features:

* Base
* Common
* Logging
* Red Black Tree

## Installation

Since EPTF\_CLL\_Scheduler is used as a part of the TTCN-3 test environment this requires TTCN-3 Test Executor to be installed before any operation of these functions. For more details on the installation of TTCN-3 Test Executor see the relevant section of [2].

If not noted otherwise, in the respective sections, the following are needed to use EPTF\_CLL\_Scheduler:

* Copy the files listed in section [2.3, 2.4] to the directory of the test suite or create symbolic links to them.
* Import the Scheduler demo or write your own application using Scheduler.
* Create Makefile or modify the existing one. For more details see the relevant section of [2].
* Edit the config file according to your needs, see the following section [2.6].

## Configuration

The executable test program behavior is determined via the run-time configuration file. This is a simple text file, which contains various sections. The usual suffix of configuration files is .cfg. For further information on the configuration file see [2].

This Scheduler feature defines TTCN-3 module parameters as defined in [2] clause 4. Actual values of these parameters – when no default value or a different from the default actual value wished to be used – shall be given in the [MODULE\_PARAMETERS] section of the configuration file.

This Scheduler feature defines the following module parameters:

**tsp\_debug\_EPTF\_SchedulerFunctions**

This boolean type module parameter is defined in module EPTF\_CLL\_Scheduler\_Definitions. It is used to enable debug logging. By default, its value is false.

**tsp\_debugVerbose\_EPTF\_SchedulerFunctions**

This boolean type module parameter is defined in module EPTF\_CLL\_Scheduler\_Definitions. It is used to enable debug logging. By default, its value is false.

**tsp\_EPTF\_ELEMENTARY\_TIMESTEP\_PARAM**

This float type module parameter defines the granularity of the time-axis of the event queue.

**tsp\_EPTF\_Scheduler\_enableRounding**

This boolean type module parameter enables/disables rounding the desired time of the scheduled event.

**tsp\_EPTF\_Scheduler\_enableLoadMeasurement**

This boolean type module parameter enables/disables the load measurement functionality.

## DTE handling

When the user wants his own function to be called at the time, when during the execution of his action a DTE happens, then this function can be given in the f\_EPTF\_SchedulerComp\_scheduleAction pl\_dteHandler parameter.

The given function, after the DTE intervention, will be called by the system and the action’s data just as the message of DTE will be transmitted in a parameter.

By default DTE handling is disabled.

## Load measurement

Load measurement functionality can be used to detect/prevent overload situation. It measures the load based on the cumulative sum of lag of events during the measurement period (LoadMeasurementPeriod). When the measured load level exceeds the threshold (MaxLoadThreshold), the events to be scheduled into the past are scheduled in the future. This way the overload situation can be prevented.

By default the load measurement is enabled.

# Error messages

Please note that besides the below described error messages, error messages shown in [2] or those of other used features or product may also appear.

EPTF CLL Scheduler defines the following error messages:

* Error: f\_EPTF\_SchedulerComp\_CancelEvent: clearing invalid\_chain or non-empty event at qidx: <index>
* Error: f\_EPTF\_SchedulerComp\_performActions has failed to freeze event: <event>
* Error: f\_EPTF\_SchedulerComp\_performActions has failed with event: <event>
* Error: cannot remove current event from queue
* ERROR: f\_EPTF\_SchedulerComp\_performActions() has been failed
* [f\_EPTF\_Scheduler\_getNewTimerVal]: Error: getNewTimerVal: Cannot get smallest element. Tree could be empty
* [f\_EPTF\_Scheduler\_schedulerIsNotLate] Error: Cannot get smallest element, maybe tree is empty
* [f\_EPTF\_Scheduler\_eventIsValid] Error: checking non-existent event at pl\_qidx: <index>
* [f\_EPTF\_Scheduler\_removeValidEvent] Error: clearing invalid, or non-empty event at pl\_qidx: <index>
* [f\_EPTF\_Scheduler\_freezeBusyEventHead] Error: Cannot get smalledst element index. Tree might be empty.
* [f\_EPTF\_Scheduler\_freezeBustEventHead] Error: freezing non-existent event.
* Error: f\_EPTF\_Scheduler\_removeInvalidEvent: clearing non-invalid, or non-existent event at qidx: <index>
* Error: getNewTimerVal: list is empty
* Error: f\_EPTF\_Scheduler\_schedulerIsNotLate: list is empty
* Error: f\_EPTF\_Scheduler\_eventIsValid: checking non-existent event at pl\_qidx: <index>, pl\_NextEventQueue.order.itemList[pl\_qidx]: <value>
* Error: f\_EPTF\_Scheduler\_eventIsInvalid: checking non-existent event at pl\_qidx: <index>, pl\_NextEventQueue.order.itemList[pl\_qidx]: <value>
* Error: f\_SetSchedulerActionRoundable: invalid not-busy event during appending to!: <value>
* Error: f\_EPTF\_Scheduler\_removeValidEvent: clearing invalid, or non-empty event at pl\_qidx: <index>, pl\_nextEventQueue.order.itemList[pl\_qidx]: <value>
* Error: f\_EPTF\_Scheduler\_freezeBusyEventHead: freezing non-existent event: <event>, pl\_NextEventQueue.order.itemList[pl\_NextEventQueue.order.status.busyStatus.head]: <value>
* Error: f\_EPTF\_Scheduler\_removeInvalidEvent: clearing not-invalid, or non-existent event at qidx: <index>, pl\_NextEventQueue.order.itemList[qidx]: <value>
* Error: f\_EPTF\_Scheduler\_rescheduleEvent: invalid not-busy event during appending to!: <value>

# Warning messages

Please note that besides the below described warning messages, warning messages shown in [2] or those of other used features or product may also appear.

The EPTF CLL Scheduler defines the following warning messages:

* Warning: getNewTimerVal: scheduler is behind clock, timerval: <value>, event: <event>

# Examples

The “demo” directory of the deliverable contains the following examples:

* EPTF\_Scheduler\_test.prj
* EPTF\_Scheduler\_test.ttcn

## Demo module

The demo module (EPTF\_Scheduler\_test.ttcn) illustrates a typical usage of the Scheduler feature.

# References

1. ETSI ES 201 873-1 v3.2.1 (2007-02)  
   The Testing and Test Control Notation version 3. [Part 1: Core Language](http://www.etsi.org/deliver/etsi_es/201800_201899/20187301/03.02.01_60/es_20187301v030201p.pdf)
2. 1/198 17-CRL 113 200/5 Uen  
   User Guide for the TITAN TTCN-3 Test Executor
3. 109 21-CNL 113 512-22 Uen   
   EPTF CLL for TTCN-3 toolset with TITAN, Product Revision Information
4. 155 17-CNL 113 512 Uen   
   EPTF CLL for TTCN-3 toolset with TITAN, Function Specification
5. 198 17-CNL 113 512 Uen  
   EPTF CLL for TTCN-3 toolset with TITAN, User Guide
6. 14/155 16-CNL 113 512 Uen  
   EPTF CLL Scheduler, Function Description
7. 12/198 17-CNL 113 512 Uen  
   EPTF CLL Red Black Tree, User Guide
8. EPTF CLL for TTCN-3 toolset with TITAN, [Reference Guide](http://ttcn.ericsson.se/TCC_Releases/Libraries/EPTF_Core_Library_CNL113512/doc/apidoc/html/index.html)