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

EPTF CLL Variable, User Guide

Contents

[1 Introduction 2](#_Toc453592996)

[1.1 Revision history 2](#_Toc453592997)

[1.2 About this Document 2](#_Toc453592998)

[1.2.1 How to Read this Document 2](#_Toc453592999)

[1.2.2 Abbreviations 2](#_Toc453593000)

[1.2.3 Terminology 2](#_Toc453593001)

[1.3 System Requirements 3](#_Toc453593002)

[2 Variable 3](#_Toc453593003)

[2.1 Overview 3](#_Toc453593004)

[2.2 Description of files in this feature 3](#_Toc453593005)

[2.3 Description of required files from other features 3](#_Toc453593006)

[2.4 Installation 4](#_Toc453593007)

[2.5 Configuration 4](#_Toc453593008)

[2.6 Usage 5](#_Toc453593009)

[3 Error messages 6](#_Toc453593010)

[4 Warning messages 8](#_Toc453593011)

[5 Examples 9](#_Toc453593012)

[5.1 Configuration file 9](#_Toc453593013)

[5.2 Demo Module 9](#_Toc453593014)

[6 References 10](#_Toc453593015)

# Introduction

## Revision history

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Rev | Characteristics | Prepared |
| 2007-11-30 | PA1 | First draft version | <ETHJGI> |
| 2007-12-05 | PA2 | Updated after review | <ETHJGI> |
| 2008-02-14 | PA3 | NewVar, FloatList added | ETHJGI |
| 2008-11-07 | PB1 | T\_Data-3\_Variable | ETHJGI |
| 2008-11-07 | PB2 | Updated after review | ETHJGI |
| 2010-06-28 | PC1 | Custom refresh added | ETHJGI |
| 2012-01-18 | PD1 | Settable MaxWaitTime for ByeAck | ETHJGI |
| 2015-12-03 | E | Updated for release | ESZILSZ |
| 2015-12-16 | PF1 | tsp\_EPTF\_Var\_maxWaitTimeForByeAck default value changed | ESZILSZ |
| 2016-06-09 | F | Updated for release | ESZILSZ |

## About this Document

### How to Read this Document

This is the User Guide for the Variable feature of the Ericsson Performance Test Framework (TitanSim), 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 Variable feature [5]. For more information on the EPTF CLL please consult the Product Revision Information, the User’s Guide [3] and the Function Specification [4] of the TitanSim.

### Abbreviations

CLL Core Library

EPTF Ericsson Performance Test Framework

TitanSim New synonym for the EPTF Framework

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

### Terminology

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

*Variables* contain values, that can be accessed from remote components.

## System Requirements

In order to use the Variable feature the system requirements listed in EPTF CLL User Guide [4] should be fulfilled.

# Variable

## Overview

The EPTF CLL Variable component is a fundamental component providing an enhanced implementation for variables in a TTCN-3 test environment. It has been developed specifically for load test requirements, but can be used in other application areas, as well.

## Description of files in this feature

The EPTF CLL Variable API includes the following files:

* Variable
  + EPTF\_CLL\_Variable\_Definitions.ttcn: This TTCN-3 module contains common type definitions that should be used in all Variable Components.
  + EPTF\_CLL\_Variable\_Functions.ttcn: This TTCN-3 module contains the implementation of Variable functions.
  + EPTF\_CLL\_Variable\_ExternalFunctions.cc: This TTCN-3 module contains the c++ implementation of external functions of Variable.

## Description of required files from other features

The Variable feature is part of the TitanSim EPTF Core Load Library (CLL). It relies on several features of the CLL. The user has to obtain the products/files to be found under the respective feature names:

* Base
* Common
* Hashmap
* FreeBusyQueue
* TCCUsefulFunctions

## Installation

Since EPTF CLL Variable 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.

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

* Copy the files listed in section [‎2.2, ‎2.3] to the directory of the test suite or create symbolic links to them.
* Import the Variable demo or write your own application using Variable.
* Create Makefile or modify the existing one.
* Edit the config file according to your needs.

## Configuration

The executable test program behaviour 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.

Actual values of these parameters – when there is no default value or a different from the default actual value to be used – shall be given in the [MODULE\_PARAMETERS] section of the configuration file.

The EPTF Variable feature defines the following module parameters:

boolean **EPTF\_CLL\_Variable\_Definitions. tsp\_debugVerbose\_PoolMgmt**

This module parameter can be used to enable debugging of the pool of booleans used for handling blocking situations. Default: false.

float **tsp\_EPTF\_Var\_SyncInterval**

This module parameter sets the period in seconds for refreshing the buffered subscribers (timeLine, sampled, sampledAtSync subscription modes). Default: 10.0

boolean **tsp\_EPTF\_Var\_enableName2Id**

Enables the hashMap used for finding variables by name. Default: false

boolean **tsp\_EPTF\_Var\_enableCompRef2Id**

Enables the hashMap used for finding variables by component reference. Default: false

boolean **tsp\_EPTF\_Var\_enableEmptyTimeLine**

By default the heartbeat message that is sent to all remote components where there are subscribers is disabled. This module parameter can be used to enable this heartbeat. It is not recommended to enable this module parameter because the heartbeat messages can generate a lot of internal messages. Default: false

charstring **tsp\_EPTF\_Var\_snapshotFilename**

The default name of the snapshot file without the timestamp used by the function f\_EPTF\_Var\_save. Its default value is “EPTF\_Var\_SnapshotFile”.

charstring **tsp\_EPTF\_Var\_snapshotTimestamp**

The default timestamp of the snapshot file used by the function f\_EPTF\_Var\_save. Its default value is “\_%Y-%m-%d\_%H.%M.%S.cfg”.

float **tsp\_EPTF\_Var\_maxWaitTimeForByeAck**

The maximum waiting time for ByeAck messages during cleanup. Its default value is 240.0 sec.

## Usage

To use the EPTF Variable feature

* extend your component with the EPTF\_Var\_CT component.
* call the init function of the Variable to initialize the feature
* use its public functions to create/adjust/subscribe for variables

Note, that the init function activates the main altstep of the Variable, so it is not necessary to call it explicitly in any of your altsteps.

Do not access the component variables in EPTF\_Var\_CT directly! Use the API functions instead.

To create EPTF Variables use the f\_EPTF\_Var\_new\* functions. To modify their values use f\_EPTF\_Var\_adjust\*. To set a new value without invoking the refresh mechanism (notifying the subscribers) call f\_EPTF\_Var\_set\*. For subscribing to variables in remote PTC-s use f\_EPTF\_var\_subscribe\*.

Please note, that when you create referenced EPTF Variables do not modify the referred variables afterwards (do not add new elements to arrays, record of-s). This is especially true if you are assigning referenced EPTF Variables to elements of **record of** types. Adding new elements may result in memory reallocation and that will make the stored references in the EPTF Variables unusable.

Guard functions can be used to validate new values before setting them into the EPTF Variables.

CalcFn function can be used to automatically calculate the value of an EPTF Variable from other EPTF Variables.

PostProc functions can be used to perform some tasks after the value of the EPTF Variable was set, but before the subscribers were notified.

The unsubscribe notify functions are useful to get call-back if the variable was unsubscribed from its provider.

The adjust handler (argument of the adjust function) can be used to perform some task after the adjust response is received, but without blocking the execution of the process calling the adjust function.

It is possible to register callback functions for the sync-event for any refresh rate (the event when the update of timelined, sampled etc. subscribers takes place for a given refresh rate) by the f\_EPTF\_Var\_addSyncCallBackFn function. All registered callbacks are called in order of registration before/after the updates are sent to the subscribers.

It is also possible to snapshot the value of all EPTF Variables registered to save by the function f\_EPTF\_Var\_registerVarToSave with the help of the function f\_EPTF\_Var\_save. It saves the value of EPTF Variables in a module parameter format. To load the saved data call f\_EPTF\_Var\_loadVarsFromCfg. Note, that the saved file should be included into the runtime configuration file and the module parameter in it shall be defined in a TTCN-3 module.

# 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\_Var[<idx>] content is not and <type>Type reference**

The f\_EPTF\_Var\_get<type>Value function is called for a referenced content EPTF Variable with a different type.

**EPTF\_Var[<idx>] has invalid content**

One of the f\_EPTF\_Var\_get<some type>Value functions was called for a variable that has invalid content

**EPTF\_Var[<idx>] is not a referenced variable**

The f\_EPTF\_Var\_getValueRef function was called for a variable which is not a referenced content variable

**Cannot set local provider for EPTF\_Var[<idx>]. It has already a provider**

The function f\_EPTF\_Var\_subscribeLocal was called for a variable which already has a provider

**Cannot adjust the content of the EPTF\_Var[<idx>]. It has a local provider! <the data of the variable>**

The f\_EPTF\_Var\_adjustContent function was called for a variable that has a local provider. These kinds of variables cannot be adjusted.

**Cannot adjust the content of the EPTF\_Var[<idx>]. It has a local top-provider! <the data of the variable>**

The f\_EPTF\_Var\_adjustContent function was called for a variable that is a subscriber of a remote variable, but the top level provider is a calculated variable.

**New EPTF Variable name is not unique: <name> Already defined by EPTF\_Var[<idx>]: <the data of the variable>**

When a new EPTF Variable was created the name specified is already in use by an existing variable

**Cannot convert invalid content to direct content <directContent>.**

The content of the EPTF Variable to be converted is invalid

**Cannot convert direct content <directContent> to direct content <content> Types are incompatible**

Cannot set the given directContent into the EPTF Variable. It has a different content type.

**Cannot convert direct content <directContent> to string.**

The given content cannot be converted to string.

**Variable <subscriberVarName> on component < provider.compRef> does not exist. Subscribe failed by the subscriber: <subscriber>**

Subscribe failed. The variable with the given name does not exist on the specified component.

**Auto-generated name <autoName> for subscriber EPTF\_Var[<subscriber.idx>] is not unique. Already defined by EPTF\_Var[<idx>]: <Variable data>**

The local name for the subscriber variable was not specified, so it was auto-generated. But the auto-generated name already exists.

**Callback function for AdjustRemoteContentResp handler not found. Database error!**

This is an internal error. Internal database of call-back functions is corrupt.

**SyncBuffer does not contain data for component <remoteCompRef>**

The syncBuffer database is corrupt.

**Cannot adjust the content of the EPTF\_Var[<idx>]. It has a local provider! <variable data> EPTF\_Var\_Msg\_AdjustContent message dropped.**

An adjust request is received for a variable that cannot be adjusted. This is an internal error. This message should not have been sent.

**Callback function not found. Database error!**

Internal error.

**Unexpected message received from <senderComp>: <msg data>**

Unexpected message received on the Variable internal interface.

**<some text> Assertion failed! <some text>**

Internal error.

# 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.

**Already subscribed for <remoteCompRef>:<remoteProviderVarName> by v\_EPTF\_Vars[<idx>]: <variable data>**

The subscription already exists.

**No response received for subscribe data: <idx> <remoteCompRef> <remoteProviderVarName> <subscriptionMode> <blockingFlagIdx>**

The provider component does not respond for subscribe. The subscribe function timeouts, and the created variable is not properly subscribed. This may cause errors later.

**The guard function #<i> (<guardFn>]) of EPTF\_Var[<idx>] does not accept the new value: <newContent>**

There was a failed attempt to modify the value of the EPTF Variable

**Unexpected message received from <senderComp> during cleanup. Message dropped: <msg data>**

Cleanup is in progress. Most of the messages are dropped (e.g. no more subscriptions allowed, etc.).

**--VAR CLEANUP TIMEOUT-- Not all responses received for bye messages.**

The cleanup is not properly performed. One ore more connected Variable components did not respond in time. The execution of the current component is forced to terminate. This may result in errors in other components afterwards. The maximum waiting time can be adjusted via the module parameter tsp\_EPTF\_Var\_maxWaitTimeForByeAck or by calling the function f\_EPTF\_Var\_setMaxWaitTimeForByeAck.

**Cannot forward adjustcontentNack to variable <variable>. Not a subscriber of EPTF\_Var[<idx>].**

The subscriber variable that was adjusted unsubscribed from its provider before the response to the adjust request is received. The response is dropped.

**EPTF\_Var\_Msg\_GetRemoteContent is received for a nonexistent EPTF Variable: <name>**

The component which sent the getRemote message exited before it was answered.

**EPTF\_Var\_Msg\_AdjustRemoteContent is received for a nonexistent EPTF Variable: <name>**

The component which sent the adjustRemote message exited before it was answered.

**EPTF\_Var\_Msg\_AdjustRemoteContent received for an EPTF Variable with local top-provider: <idx>**

There was a failed attempt to adjust the variable without subscription.

**Cannot load Variable <name> from config: Variable does not exist.**

The variable that should be loaded from the configuration data does not exist.

**f\_EPTF\_Var\_save: cannot open file <filename>: <reason>**

The specified file could not be opened.

**f\_EPTF\_Var\_save: cannot save EPTF\_Variable data to file <filename>: <reason>**

A write error encountered. Saving data to the specified file failed.

# Examples

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

* main.cfg
* EPTF\_Variable\_test.ttcn

## Configuration file

The used configuration file (main.cfg) is for the Variable example is placed in the demo directory.

## Demo Module

The demo module EPTF\_Variable\_test.ttcn illustrates a typical usage of the Variable feature.

There are two main demo configurations. The first configuration uses one component and the second uses several components. The basic behaviour of the Variable feature is demonstrated on these test configurations.

Figure 1. Basic Configuration

EPTF Variable

names

* myVar
* intVar
* intSum
* *x*.intSum (auto)
* remoteIntRef
* *x*.remoteIntRef (auto)

Component

Variables

v\_myVar

v\_myVar2

Referenced var

Local subscr

Remote subscr

Indexes

0 - intRef

1 - intVar

2 – intSum

3 – remoteInt

4 – remoteintRef

5 - remoteRemoteInt

The left box shows the component variables which are used to create a referenced EPTF Variable. The left column in the right box shows the EPTF Variable names created (auto refers to auto-generated names), and the subscription relation between them. On the right the index of the variables are shown together with the name of the component variable that stores the index.

# 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. 155 17-CNL 113 512 Uen   
   EPTF CLL for TTCN-3 toolset with TITAN, Function Specification
4. 198 17-CNL 113 512 Uen  
   EPTF CLL for TTCN-3 toolset with TITAN, User Guide
5. 18/155 16-CNL 113 512  
   EPTF CLL Variable Function Description
6. TitanSim 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)