TESSY®

TEST SYSTEM

automated

unit/integration tests

safety manual

release 08/19 | revision 42.011 | TESSY v4.2

RAZORCAT

Imprint

Razorcat Development GmbH
Witzlebenplatz 4
Germany, 14057 Berlin
tel: +49 (030) 53 63 57 0
fax: +49 (030) 53 63 57 60
e-mail: support@razorcat.com
internet: http://www.razorcat.com

Liability exclusion

Razorcat Development GmbH assumes no liability for damage that is caused by improper installation or improper use of the software or the non-observance of the handling instructions described in this manual.

Thanks

We would like to thank Frank Büchner, Hitex Development Tools GmbH for his valuable contribution and commitment in supporting TESSY and spotlighting functionalities and features.

Contents

Safety Manual

About TESSY

The test system TESSY was developed by the Research and Technology Group of Daimler. The former developers of the method and tool at Daimler were:
Klaus Grimm
Matthias Grochtmann
Roman Pitschinetz
Joachim Wegener

TESSY has been well-tried in practice at Daimler and is since applied successfully. TESSY is commercially available since spring 2000 and is further developed by Razorcat Development GmbH.

TESSY offers an integrated graphic user interface conducting you comfortably through the unit test. There are special tools for every testing activity as well as for all organizational and management tasks.

Dynamic testing is indispensable when testing a software system. Today, up to 80% of the development time and costs go into unit and integration testing. It is therefore of urgent necessity to automate testing processes in order to minimize required time and costs for developing high-quality products. The test system TESSY automates the whole test cycle; unit testing for programs in C/C++ is supported in all test phases. The system also takes care of the complete test organization as well as test management, including requirements coverage measurement and traceability.

Safety Manual

Core workflow and registration for safety information

> **Important:** If you work with TESSY in a safety-relevant environment, please read this chapter carefully and register for our safety customer e-mail-list to be informed about known problems as described below!

TESSY can be used for testing of safety-relevant software. Therefore, the core workflow of TESSY as well as the release and test process of the TESSY product has been certified according to ISO 26262-08:2011 and IEC 61508-3:2010. In the course of the re-certification of TESSY 4.1 by TÜV SÜD Rail GmbH the certification was extended to also cover EN 50128 and IEC 62304. Our quality management system ensures proper handling of all development processes for the TESSY product and constantly improves all procedures concerning quality and safety.

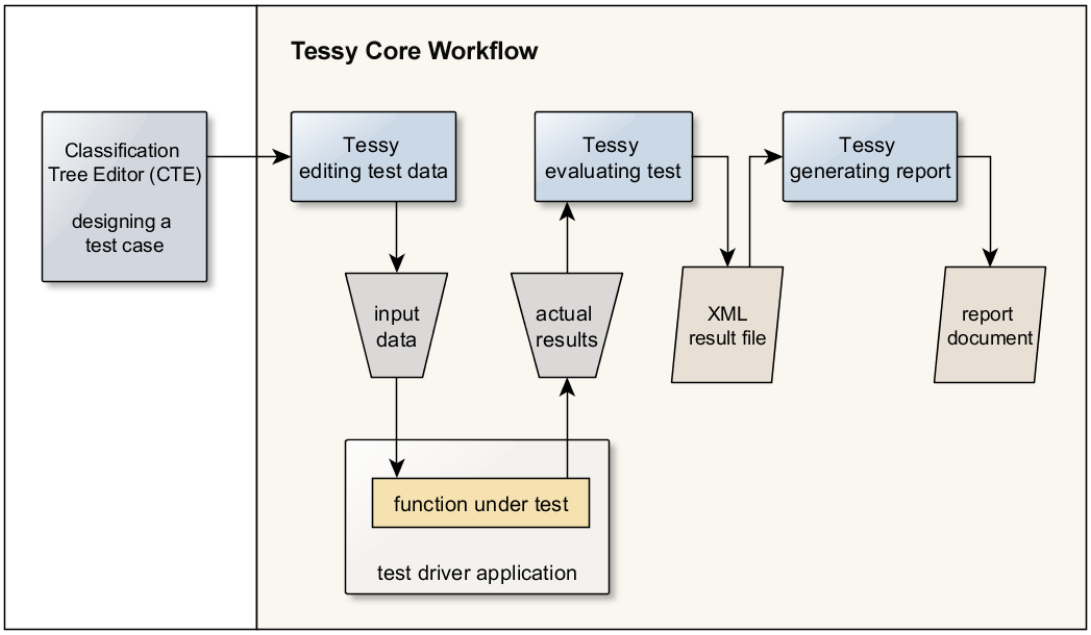

Figure 0.1: Core workflow of TESSY

The figure above shows the core workflow of TESSY that is fully automated and subject to tool qualification. All other tool capabilities like editing or environment and interface settings are additional features out of scope of the tool qualification. The core workflow of TESSY has been certified according to ISO 26262:2011 and IEC 61508:2010 as well as EN 50128

and IEC 62304. Starting from editing of test data, the core workflow covers test execution, evaluation of test results and report generation. Additionally, the coverage measurements have been verified according to our certified safety plan. Please note, that the Classification Tree Editor (CTE) which covers test preparation activities is not part of the certified core workflow of TESSY.

Safety-relevant problems arising in released TESSY versions will be reported (once they are detected) and regarded closely to have them fixed as fast as possible. If you work with TESSY in a safety-related environment, **please register for our safety customer e-mail-list**:

- → Send an e-mail to support@razorcat.com
- → Topic: “Known problems requested”
- → Content: your contract data

You will be informed about current and newly arising “known problems” as well as work-arounds.

Verification and certification of TESSY

The “Tool Qualification Pack” (TQP) is an additional purchase of documents and tests for TESSY, provided as baseline for the certification process in order to qualify TESSY as a software verification tool according to DO-178B/C.
Please contact via support@razorcat.com.

Additionally, TESSY has been qualified by the German certification authority TÜV SÜD Rail GmbH as a testing tool for usage in safety-related software development according to ISO 26262 and IEC 61508. TESSY was also evaluated against IEC 62304 (medical technology) and EN 50128 (railway technology). EN 50128:2011 is an application standard derived from IEC 61508. TESSY was classified as a T2 offline tool in accordance with EN 50128:2011. The TÜV certificate and a certification report is available on http://www.razorcat.com.

The TQPack contains tests for ANSI-C compliant source code using the GNU GCC compiler that is part of the TESSY installation. Using an embedded compiler/debugger for a specific microcontroller requires adaptation of the TQPack for this specific target environment. This can be provided as an engineering service by Razorcat.

Instrumentation for coverage measurement

When executing tests using coverage measurements, it is recommended that all tests are executed once with and once without coverage instrumentation. TESSY uses a copy of the original source file when creating the test application. This copy of the source file will be instrumented for coverage measurements. Usually both test runs yield the same result, indicating that the instrumentation did not change the functional behavior of the test objects.

Please note, that the source code will be instrumented even if no coverage measurement has been selected in the following cases:

- When using the call trace feature
- When using static local variables

Some extra code will be added at the end of the copied source file in the following cases:

- When testing static functions
- When using static global variables

Please keep this behavior in mind when preparing and executing tests with TESSY.

Adaptation to target environment

When running tests on a specific target platform, adaptations of compiler options and target debugger settings may be needed within the respective target environment. The verification of the TESSY core workflow covers tests conducted on a Windows host system using the GNU GCC compiler. In order to verify the transmission of test data and expected results to and from the target device, there are tests available that may be executed using the adapted target environment. These tests check the communication layers of the test driver application.

For details on how to run these tests refer to the application note “048 Using Test Driver Communication Tests.pdf” within the TESSY installation directory.

It is recommended to run these tests with your specific compiler/target environment after initial project setup or after any changes of the environment settings.

Command line interface (CLI)

The command line execution mode of TESSY is designed for usage on continuous integration platforms like e.g. Jenkins. Therefore it is desired that TESSY does an auto-reuse of existing tests on interface changes and tries to execute as many tests as possible with newer versions of the source code being tested when running in CLI mode.

As a result, the tests executed in CLI mode may be run with test data that do not match with the source code being tested (e.g. with uninitialized new variables) which could hide existing or newly introduced errors within that source code. It is recommended to regularly check that the existing tests still match with the interface of the software being tested.

Operating limits

TESSY is constructed for usage as a unit testing tool in order to verify the functional correctness of the function under test. The following restrictions and prerequisites for TESSY apply:

- The source code to be tested shall be compilable without errors and warnings by the compiler of the respective microcontroller target. TESSY may fail analyzing the interface of the module to be tested, if there are syntactical errors within the source code.
- TESSY does not check any runtime behavior or timing constraints of the function under test.
- The test execution on the target system highly depends on the correct configuration of the target device itself, the correct compiler/linker settings within the TESSY environment and other target device related settings within TESSY (if applicable). Any predefined setup of the TESSY tool for the supported devices requires manual review by the user to ensure proper operation of the unit testing execution.
- The usage of compiler specific keywords and compiler command line settings may require additional tests for tool qualification. Correct operation of the TESSY toolset with respect to the Qualification Test Suite (QTS) test results is only provided for ANSI compliant C code.

Since TESSY 4.x the test driver code will be generated and attached at the end of (a copy of) each source file. The following restrictions apply:

- All types used within usercode must be available within the source file of the respective test object.

- When using usercode definitions/declarations on module level, all used types must be available within all source files of the module.

For backward compatibility, you can disable the usage of the new CLANG parser and test driver generation by setting the attribute “Enable CLANG” to “false” within the project configuration using the environment editor TEE.