TPT – QUALIFICATION

in accordance with ISO 26262

Version 2.1

March 2019

PIKETEC
software engineering

Version	Date	Editor(s)	Comments
2.1	March 11, 2019	R. Grönberg	Update regarding SGS TÜV-Saar Certificate

TABLE OF CONTENTS

1 Introduction

TPT is a model-based tool environment for the test of embedded systems, in particular for testing control and regulation systems. TPT supports all important areas of the test process. In detail, these are the areas of test management, test case modelling, performing tests, test assessment and test documentation.

TPT does not only test MATLAB/Simulink, TargetLink or ASCET models efficiently and automatically. TPT also enables consistent testing from Model-in-the-Loop, Software-in-the-Loop, Hardware-in-the-Loop, CAN, LIN and other main test platforms.

A tool qualification according to [ISO 26262] is required to use TPT in safety-related development in the automotive industry.

2 ISO 26262

The [ISO 26262] "Functional Safety – Road vehicles" is an international standard that applies for all automobile manufacturers (OEMs) and their suppliers worldwide.

[ISO 26262] is worldwide mandatory for the development of road transport vehicles since November 2011.

3 Confidence in use of software tools

[ISO 26262] requires the qualification of the tools that are used in the development of software in safety related systems. [ISO 26262-8] Chapter 11 - “Confidence in the use of software tools” - specifies the conditions, which have to be met by a tool to reach the „required level of confidence“. In addition – if needed - methods for qualification are defined.

The goal of showing the confidence in the use of a tool is to prevent errors that can get into the product by using this tool. This also includes "the possibility that the malfunctioning software tool and its corresponding erroneous output can … fail to detect errors in a safety-related item or element being developed" ([ISO 26262-8], 11.2]. Thus, each test tool must undergo an examination and if necessary, qualification.

Qualification is done in two steps.
First an analysis and classification of the tool must be done and the tool confidence level (TCL) must be determined. Here the use cases and potential error cases are considered. A malfunction of a particular software tool can introduce - or fail to detect - errors in a safety-related item or element being developed. Based on Tool Impact (TI) and the Tool Error Detection (TD), the required software tool confidence level (TCL) can be determined. The methods needed for the qualification of software tools are derived from the TCL.

Secondly, if necessary, the required qualification measures will be derived and performed.

A qualification is performed during the concrete development process of a concrete system. However, to minimize the qualification effort, the assumptions and methods are described in the following that apply or are required in all cases for the qualification of TPT.

4 Analysis of TPT

The first step of a qualification is an analysis (cf. [ISO 26262-8], 11.4.5] and begins with the description of the use of the tool.

4.1 Using TPT

Section 11.4.5.1 of the [ISO 26262-8] states:

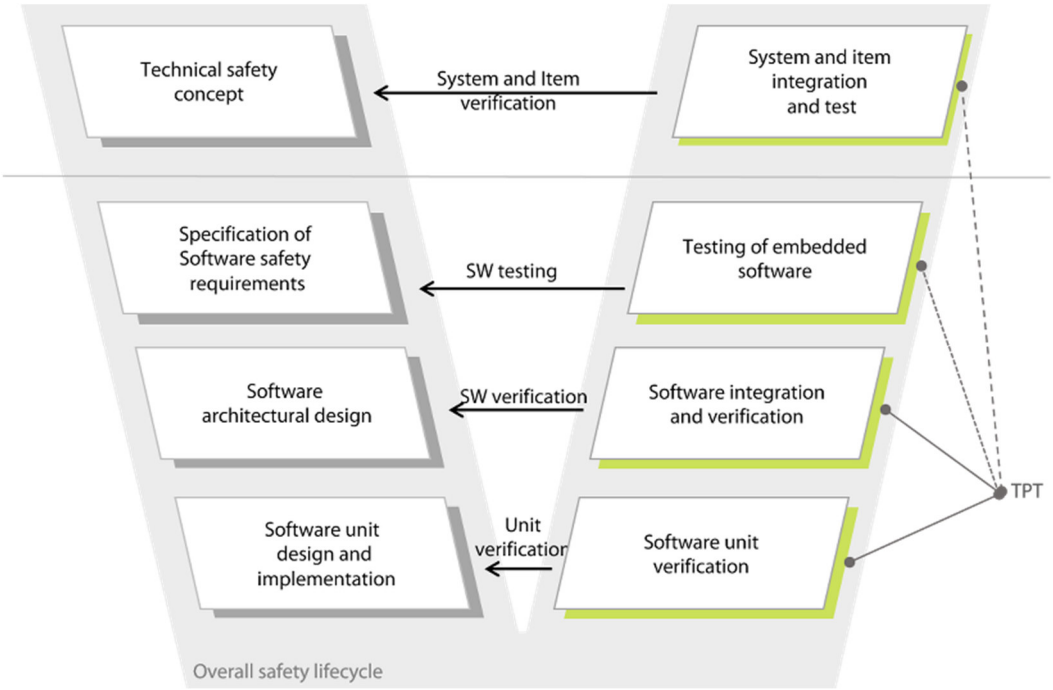
"The description of the usage of a software tool shall contain the following information:

a) the intended purpose,
b) the inputs and expected outputs, and
c) the usage procedure, environmental and functional constraints, if applicable.."

For TPT the following applies:

re a) "Intended purpose":
Using TPT as a model-based tool environment for testing <*a concrete system*>.

The possible use of TPT results from the phase model for software development as defined in Part 6 of the standard [ISO 26262-6]. The options for using TPT are marked in green in the following graphic.



[Fig. 1] Software development - „Reference phase model" ([ISO 26262-6] 5.2, Figure 2)

re b) Inputs and outputs:
Input for TPT are functional requirements, safety requirements and a test specification (test procedures, test scenarios, test cases, test datasets), including a description of the interfaces (signals) between TPT and the system to be tested.
Output from TPT is an automatically generated test report which is based on the generated test data and documents the entire test process.

re c) Functional and environmental conditions – if applicable:
This is to be supplemented for the concrete case, e.g. "usage of shared data by different software tools". For TPT, this could be e.g. data used together with INCA.

4.2 Tool Impact

[ISO 26262-8], Section 11.4.5.2 specifies:

"The intended usage of the software tool shall be analysed and evaluated to determine:

a) the possibility that a malfunction of a particular software tool can introduce or fail to detect errors in a safety-related item or element being developed.
This is expressed by the classes of Tool Impact (TI):

1) TI1 shall be selected when there is an argument that there is no such possibility;
2) TI2 shall be selected in all other cases."

TPT, as a test tool, has no direct influence on the software to be developed, i.e. TPT cannot generate bugged code on its own. Anyhow, a malfunction of TPT could lead to a violation of a safety requirement not being detected.

Thus for TPT, as for each test tool, the existence of a tool impact must be assumed (TI 2). This also applies if the aforementioned use cases are supplemented in a concrete development process.

4.3 Tool error detection

For a given tool impact (TI2), the next step is to determine the tool error detection (TD) ([ISO 26262-8], 11.4.5.2):

b) "the confidence in measures that prevent the software tool from malfunctioning and producing corresponding erroneous output, or in measures that detect that the software tool has mal-functioned and has produced corresponding erroneous output. This is expressed by the classes of Tool error Detection (TD)"

The option of detecting or preventing errors is divided into three classes.

Tool error detection	
High degree of confidence	TD1
Medium degree of confidence	TD2
shall be chosen in all other cases	TD3

"Prevention or detection can be accomplished through process steps, redundancy in tasks or software tools or by rationality checks within the software tool itself." (cf. [ISO 26262-8]11.4.5.2.b Note1).

To prevent errors from TPT, "usage guidelines" can also be used (cf. [ISO 26262-8], 11.4.5.2.b Example 2).

A determination of the tool error detection (TD) is based on a detailed analysis of possible errors of TPT for given use cases. It also depends on the tool chain in which TPT is used. Without wanting to replace the analysis, one can assume in general a classification class TD2 or TD3. A classification in class TD1 is possible.

Explanation:

TD1: TD1 is issued for software, with a high degree of confidence that incorrect outputs are prevented. Example: TD1 is selected for a code generator when the generated source code is verified in accordance with ISO 26262, TD3 if the generated source code is not verified. (cf. [ISO 26262-8], 11.4.5.2 Example 1).

The output of TPT is verified, however, not completely in accordance with ISO 26262 because development of TPT in accordance with a safety standard is not required.

Nevertheless, an assignment to TD1 can be performed - but only by a concrete analysis and considering the downstream processes.

TD3: " TD3 typically applies if there are no systematic measures in the development process available, and therefore malfunctions of the software tool and their corresponding erroneous outputs can only be detected randomly." ([ISO 26262-8], 11.4.5.2.b, Note 2)

There are systematic measures like:
- Performance a self-test in TPT. To do this, an observation function must be modelled that checks whether a certain value is achieved during the runtime. After the test run, you can retest whether the value to be achieved during the test run was actually achieved.
- Development of a "Usage guideline" with which malfunctions can be prevented (cf. [ISO 26262-8], 11.4.5.2.b, Example 2) or error detection measures (self test, see above) can be implemented.

Although there are such measures, only a concrete analysis can determine whether TD3 or TD2 is given.

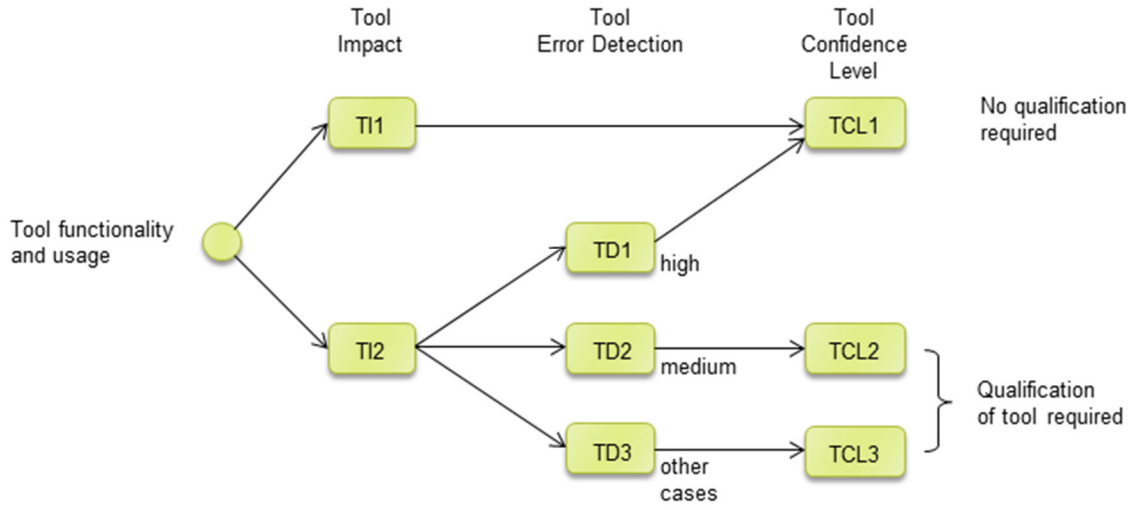
TD2: „shall be selected if there is a medium degree of confidence that a malfunction and is corresponding erroneous output will be prevented or detected" (cf. [ISO 26262-8], 11.4.5.2.b).

The determination of the Tool Error Detection cannot only be done by the tool user. Detailed information about the tool-chain and use cases are only available by the tool user. PikeTec as tool manufacturer can run a preliminary classification which has to be completed for concrete usage. This is usually done by the tool user. If wanted, PikeTec can do this either if all information is given.

Experience has shown that TPT gets a TD2 or TD3, also because tool users want to get detailed information about tool malfunctioning or measures that detect or prevent corresponding erroneous output (usage guidelines).

4.4 Tool Confidence Level

Based on the estimation for Tool Impact TI2 and Tool Error Detection TD2 (TD3), the "Tool Confidence Level" (TCL) is determined [ISO 26262, 11.4.5.5].



[Fig. 2] Determination of the tool confidence level (TCL)

If a tool confidence level TC1 is determined, no additional qualification measures are required. If TCL2 or TCL3 are determined, a qualification must be performed.

It is assumed that usually a TCL2/TCL3 is given for TPT, therefore a qualification has to be done.

5 Qualification methods

If the assumption is that a qualification is required for TPT, there are four methods from which to choose in accordance with [ISO 26262-8] Section 11.4.6:

a. "Increased Confidence from Use",

b. "Evaluation of the Development Process",

c. "Validation of the Software Tool" and

d. "Development in accordance with a safety standard".

Which of the alternative measures for a qualification are possible is determined by the ASIL classification of the product to be developed and by the result of the tool classification.

For TCL3, the methods are generically illustrated in the following table.

Table 5 – Qualification of software tools classified TCL 3

Methods		ASIL			
		A	B	C	D
1a	Increased confidence from use in accordance with 11.4.7	++	++	+	+
1b	Evaluation of the tool development process in accordance with 11.4.8	++	++	+	+
1c	Validation of the software tool in accordance with 11.4.9	+	+	++	++
1d	Development in accordance with a safety standard [a]	+	+	++	++

a No safety standard is fully applicable to the development of software tools. Instead, a relevant subset of requirements of the safety standard can be selected.

EXAMPLE Development of the software tool in accordance with ISO 26262, IEC 61508 or RTCA DO-178.

(+ …recommended; ++ …highly recommended)

For TPT, two methods are not possible:

re a.: "Increased Confidence from Use" is excluded since this method can only be used if the specification of the tool has not been modified (cf. [ISO 26262-8], 11.4.7.2). For TPT, in principle, three releases per year are planned. For every new release, due to the modified specification, the use of the previous releases cannot be considered for an "increased confidence for use".

re d.: "Development in accordance with a safety standard", must be excluded as a qualification method, since TPT is a developed systematically but not/will not be developed according on a safety standard.

"Evaluation of the tool development process" (b) and "Validation of the software tool" (c) can be used as qualification measures for TPT.

Depending on which ASIL of the project, the following picture results:

(1a) Increased confidence from use	ASIL A	ASIL B	ASIL C	ASIL D
TCL2	++	++	++	+
TCL3	++	++	+	+

(1b) Evaluation of the development process	ASIL A	ASIL B	ASIL C	ASIL D
TCL2	++	++	++	+
TCL3	++	++	+	+

(1c) Validation of the software tool	ASIL A	ASIL B	ASIL C	ASIL D
TCL2	+	+	+	++
TCL3	+	+	++	++

(1d) Dev. in acc. to a safety standard	ASIL A	ASIL B	ASIL C	ASIL D
TCL2	+	+	++	++
TCL3	+	+	+	+

(+ …recommended; ++ …highly recommended)

[Fig. 3] Methods for qualifying TPT

5.1 Evaluation of the Development Process

The standard requires:

11.4.8.2 "The development process applied for the development of the software tool shall comply with an appropriate standard."

11.4.8.3 "The evaluation of the development process applied for the development of the software tool shall be based on an appropriate national or international standard and provide evidence that a suitable software development process has been applied."

NOTE This assessment covers the development of an adequate and relevant subset of the features of the software tool.

EXAMPLE Using an assessment method based on e.g. Automotive SPICE or CMMI.

The development process of TPT is systematic, documented, an assessment is possible. A CMMI Level 3 assessment of the development process of TPT by a customer has been passed successfully.

Method 1b can be used for ASIL A and B, but so far, our customers do not ask for this method because there is no need to report known errors for example. In qualification practice, the method has not been used for TPT since 2012. Furthermore for PikeTec the effort measured in terms of benefits is not justified.

As most of our customers use TPT in safety relevant projects with ASIL C or ASIL D and for reasons, mentioned above, method 1b is no longer offered.

5.2 Validation of the software tool

The standard requires [ISO 26262-8, 11.4.9]:

11.4.9.2 a) "the validation measures shall provide evidence that the software tool complies with specified requirements to its purpose as specified in the classification"

NOTE "The validation can be performed either by using a customized test suite developed by the user or by the tool vendor (if the test suite of the vendor includes the tool use cases of the user)."

To ensure the greatest possible application for TPT, method 1c is offered for all ASILs.

Validation can be done by tests (cf. Chap. 6.3).

6 TPT Qualification

As shown in Chapter 4 the analysis of TPT can be summarized as follows:

- Tool Impact is given (TI2) for each test tool
- The Tool confidence level for TPT is expected to be TCL2, if required TCL3 as well
- Therefore, a qualification must be done
- TPT will be qualified according to method (1c) "Validation of the software tool". (cf. [ISO 26262-8], 11.4.9).

A tool can only be qualified by embedding it into a specific development process, which also means integrating the tool into a specific safety lifecycle (see ([ISO 26262-6] 5.2, Figure 2).

To ease the qualification process and to reuse knowledge, a preliminary tool qualification can be reasonable for standard use cases of the tool, if the use of the tool can be derived from a standard configuration. This facilitates the qualification process according to [ISO 26262-8] noticeable.

TPT can be qualified preliminary for standard (TPT) use cases, the use cases given in every TPT usage can be generically determined.

To qualify TPT with additional specific use cases, the generic use cases must be consolidated with the project specific use cases. A gap analysis is done, to clarify whether the generic use cases already cover this requirements or supplement is needed. If applicable, detailed use cases must be specified.

Summarized: Most of the qualification is performed by PikeTec GmbH while the tool user must support with its project specifics. This makes it very simple for the user.

TPT can be qualified for TCL2 to TCL3 and for systems from ASIL A to ASIL D.

6.1 Generic TPT Use Cases

For using TPT in a development process that conforms with ISO 26262, the use cases must be defined during the planning phase [cf. [ISO 26262-8], 11.4.4.1 c].

Strictly speaking, this can only happen for a concrete system with specified safety requirements. Still, use cases can be defined for TPT that apply in all use cases and for every system to be tested. These generic use cases are specified below.

6.1.1 TPT Use Cases

TPT is a model-based tool environment for systematic, functional testing of embedded systems, in particular for testing control and regulation systems.

Thus, the use case that results is:

Performance of systematic, functional tests

Testing is always an empirical process that can never fully test the system quality, therefore a systematic and structured selection and modelling of the test cases is especially important. As a result, in TPT, test cases are not viewed in isolation from each other. Instead, all test cases are created in a common model with the goal of clearly determining commonalities and differences between the test cases and thus gaining a precise overview of which aspects were tested and which, if applicable, were disregarded.

The modelled test cases can then be automatically executed and assessed, then TPT generates a test report.

The aforementioned general use case can be differentiated as follows:

[Fig. 4] Generic use cases in TPT

To be able to stimulate and observe the system to be tested during the test run, there must be a connection between TPT and the "System under test" (platform connection).

Thus, overall, the following use cases result:

TPT Use Cases:

5. Performance of a systematic, functional test

   5.1. Test modelling
   The tester models concrete test cases (test procedures, test variants, test cases, test data sheets) based on functional requirements, safety requirements and a test specification and describes them with the TPT modelling editor.

   5.2. Test execution
   The modelled test cases are run automatically with TPT. TPT generates a scenario description from the modelled test cases that are then automatically run. The test results are recorded in a data base.

   5.3. Definition of assessment rules
   The test assessment is performed automatically. The tester must define assessment rules for this and describe them using the TPT assessment editor.

   5.4. Test assessment and test documentation
   The test assessment is performed automatically with TPT – based on the assessment rules – after the test has been executed. The results are documented, and a summary test report is generated.

5.5. Platform connection
To stimulate and observe the system to be tested, there must be a connection of the "System under- test" to an execution platform from TPT. The connection option to different platforms are made available by TPT and can be configured with the execution editor. Possible platforms are: MATLAB/Simulink/Stateflow/Targetlink, ASCET, C-Code etc.

In a concrete use case, the generic use cases can be supplemented by additional application specific use cases if necessary. E.g. a safety requirement "Reaction xy must not occur later than n ms" can be entered. For each of such additional requirements a gap analysis is done, to clarify whether the generic use cases already cover this requirement or supplement ist needed.

6.1.2 TPT Error Cases

Based on the use cases the following error cases result:

TPT Error Cases:

1. The execution of the test is incorrect,
   i.e. a violation of a safety requirement in the system to be tested is not detected/documented or this is done incorrectly. This can occur on the TPT side if errors arise in the individual test steps.
   1.1. Test modelling is incorrect
   1.2. Test execution is incorrect.
   1.3. Definition of assessment rules is incorrect
   1.4. Test assessment and test documentation is incorrect
   1.5. Platform connection is incorrect.

These generic use and error cases can be qualified generically by using method “(1c) Validation”.

6.2 User specific Use / Error Cases

The generic TPT use cases include a wide range of possible TPT use cases and features. Although it might be, that additional user specific use cases must be considered, e.g., a TPT user wants to use a platform, which is rarely used. The additional requirements are specified by the tool user or derived together with PikeTec.

To qualify TPT with additional use cases, the generic use cases must be consolidated with the project specific use cases. A gap analysis is done, to clarify whether the generic use cases already cover this requirements or supplement is needed. The gap analysis is done by mapping user specific use / error cases and feature on the generic use / error cases and feature of TPT.

If applicable, detailed use cases / features must be specified. This is done by PikeTec.

The mapping, the analysis of a possibly given gap and the specification of additional tests are included in the TPT-Qualification.

With this approach, the requirement “The validation can be performed … by using a customized test suite developed … by the tool vendor (if the test suite of the vendor includes the tool use cases of the user)” ([ISO 26262-8] 11.4.9.2) is met.

6.3 Validation

Validation can be done by tests.

To derive the needed tests, the generic use cases und the additional user specific use cases are considered and all features are specified which realize the use cases. In addition, the possible error cases are determined.

Example:

Use Case Tool user	**TPT feature**	**pot. Error**	**Name/description**	**Validation**	**Comment**
UC 1	F 1.1		Open a TPT-file		
		pE 1.1	TPT-file cannot be opened	no TD=1	Failure is obvious Tool Error Detection =1
		pE 1.2	TPT-file is opened with unwanted changes	yes	

[Fig. 5] TPT feature and error case

For error case with a given tool impact (TI) tests are provided.

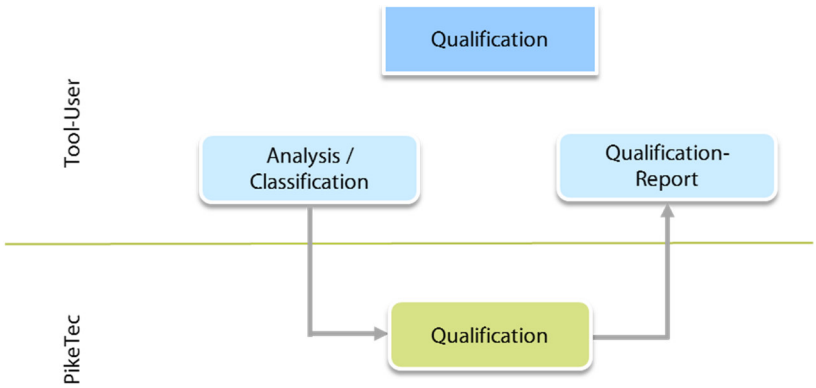
Now about 2.200 functional tests are done manually for each TPT release. A subset of these are analyzed for the qualification relevant use and error cases.

In addition, tests of our Validation Suite are run. The Validation Suite consists of 23 TPT-models with about 12.000 tests for different platforms. The results of the Validation suite are verified by back-to-back tests.

All tests and additional Qualification services are done by PikeTec.

6.4 Effort for the user

A qualification requires effort, the effort for the user is minimal.



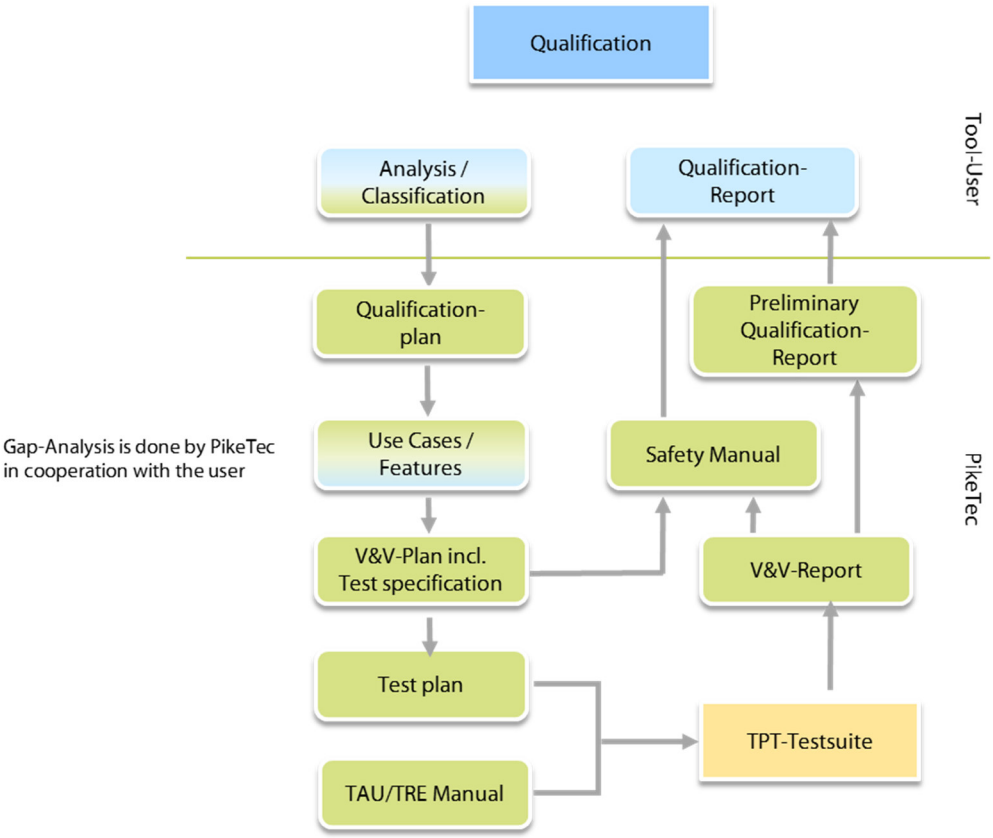
[Fig. 6] Qualification effort I

The TPT-user is responsible for the whole qualification, the tool analysis to determine the tool confidence level and the qualification report.
PikeTec can deliver a preliminary analysis and/or qualification report, respectively offers support to carry out the analysis and to write the report

PikeTec offers a TPT Qualification-Package, to support the tool user in the best and most efficient way.

After determination the user specific use- and error cases together with the tool user no addition work has to be done by the user.

PikeTec performs all needed further work and delivers all needed documents according to [ISO 26262-8].



[Fig. 7] Qualification effort II

6.5 TPT Qualification-Package – scope of work

PikeTec TPT Qualification consist of two parts:

1. **Support**[1]
   - Support regarding the qualification process
   - Support to determine the Tool Confidence Level
   - Support to determine the use cases, if the use cases go beyond the generic scope

2. **Scope of delivery**
   - Determination of all user specific use / error cases
     including a Gap-Analysis, to analyze whether additional use / error cases and tests within the specific project must be taken into account beyond the generic scope
   - Preliminary results of a Tool Analysis – if required
   - Test execution for generic and additional use /error cases
   - Test documentation
     - Results manual tests, analysis of failed tests
     - Results of Validation Suite tests, analysis of failed tests
   - All needed documents e.g.
     - Tool Qualification plan
     - Validation- and Verification-Plan
     - Test plan
     - Validation- and Verification report
     - Guidelines (Safety manual) to recognize or avoid possible errors e.g.
     - Preliminary Qualification report, which can be used for the final Qualification report, which the tool user is responsible for

6.6 TPT Qualification-Package - certification

TPT is used in safety related development for systems up to ASIL D, which addresses the highest criticality and the highest requirements for a tool usage. PikeTec offers a TPT Qualification-Package, to meet these requirements in the best and most efficient way.

The TPT Qualification-Package is certified by SGS-TÜV Saar (Certificate No FS/71/220/19/0370).

TPT Qualification-Package is applicable ASIL A – ASIL D and Tool Confidence Level TCL 2 and TCL 3.

It supports the TPT tool user on qualification of the TPT tool according to ISO 26262-8, clause 11. The TPT Qualification Package defines project specific tool-chain restrictions and process measures. If they are applied ty the tool user, the related TPT tool version can be classified as TCL1.

[1] if required by the tool user

7 References

[ISO 26262] ISO 26262:2018: Road vehicles - Functional safety. International Standard, Part 1-12, December 2018

[ISO 26262-6] ISO 26262-6:2018: Road vehicles - Functional safety. Part 8: Product development at the software level. International Standard, December 2018

[ISO 26262-8] ISO 26262-8:2018: Road vehicles - Functional safety. Part 8: Supporting processes. International Standard, December 2018

8 History

Version	Date	Editor(s)	Comments
1.0	August 23, 2012	R. Grönberg	Document drafted
1.1	September 01, 2012	R. Grönberg	Modification after review on 25.08.12 by J. Lüdemann
1.1	December 13, 2012	R. Grönberg	Layout update
1.2	December 19, 2013	R. Grönberg	Update chapter “Literature”
1.3	November 14, 2014	R. Grönberg	Update chapter “Literature”
1.4	February 02, .2016	R. Grönberg	Update of content
1.5	March 09, .2017	R. Grönberg	Update of content
1.6	November 02, 2017	R. Grönberg	Update of content
1.7	October 29, 2018	R. Grönberg	Update of content
2.0	February 04, 2019	R. Grönberg	Update for ISO26262:2018