

Software Product Certification

Model Verified Requirements Consistency Certificate

LaQuSo

This document provides a short summary of the LaQuSo Software Product Certification. It describes the model verified requirements consistency certificate. Details can be found in the LaQuSo SPCM: Software Product Certification Model

November 2007

Copyright © 2007, LaQuSo

Nothing from this report may be duplicated and/or made public (for whatever reason) by print, photocopy, microfilm, electronic or any other way without written permission from LaQuSo.

Contents

1	SOFTWARE PRODUCT CERTIFICATION	5
1.1	The Certificate.....	5
1.2	Input	5
1.3	Conformance	5
1.4	Certification Criteria.....	6
1.5	Certificate Types	6
1.6	LaQuSo Software Product Certification	7
2	MODEL VERIFIED REQUIREMENTS CONSISTENCY.....	8
	REFERENCES	9

I Software Product Certification

If an organization wants certainty about or confidence in a software artifact a LaQuSo certificate can be requested.

Being part of universities, LaQuSo is able to perform the independent evaluator role in many certification projects.

1.1 The Certificate

Certification is a check that the artifact fulfills a well-defined set of requirements. These requirements are defined by the customer or a third party; LaQuSo will do the check. The certificate will always refer to the requirements that were used to check the artifact against. The certificate covers only the product quality, not the management and development processes.

The quality certificate consists of a diagnosis report plus verdict document and will grow from a diagnosis report with a LaQuSo certificate to diagnosis with an official widely recognized certificate.

1.2 Input

Input for the software product certification can be any artifact (documents, models, source files, executables, etc.) from one of the following product areas:

- *Context Description*: the environment and main processes of the system;
- *User Requirements*: the functions the system has to fulfill;
- *High-level Design* (also called software requirements): a translation of the user requirements into a more precise description in terms of system architects;
- *Detailed Design*: several models of the system that describe how the system will be built;
- *Implementation*: the system (source code) and its documentation, built according to the design;
- *Tests*: the tests of the different software components and the whole system.

1.3 Conformance

The software artifacts delivered as input will be checked against each other (type a), against a standard or regulation (type b) or against a certain property (type c). See the figure below for the three types.

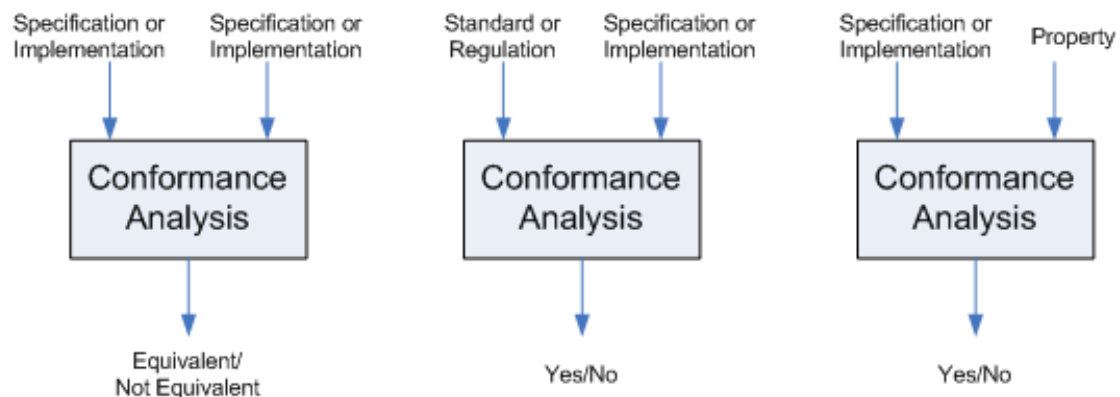


Figure 1: Different Types of Conformance Analysis

The property needed as input in type c conformance analysis can be one of the following:

- *Consistency*: do the different parts of a software artifact conform to each other?
- *Functional*: does input to the system produce the expected output?
- *Behavioral*:
 - *General*: does the system meet general safety and progress properties like absence of deadlocks?
 - *Specific*: are constraints on the specific states of the system met?
- *Quality*: do the artifacts fulfill non-functional requirements in the area of e.g. performance, security, and usability?

Appropriate conformance analysis techniques are chosen depending on the application domain and input artifacts.

1.4 Certification Criteria

The LaQuSo certification model defines three criteria on which the input artifacts are certified. For each criterion three levels of achievement are possible.

1. **Formalness**. All required elements in the Product Area should be present and as much formalized as possible.
2. **Uniformity**. The style of the elements in the Product Area should be standardized.
3. **Conformance**. All elements should conform to the property that is subject of the certification.

Table 1: Certification Criteria Achievement Levels

CC1	Formalness
○	Some required elements are missing
1	All required elements are present
2	Semi-formal elements have been added
3	Formal elements have been added
CC2	Uniformity
○	No standardization
1	Within the artifact
2	Style complies to a company standard
3	Style complies to an industry standard
CC3	Conformance
○	Faults are detected
1	Manual review/testing has not detected any faults
2	Automated testing has not detected any faults
3	Formal verification has not detected any faults

1.5 Certificate Types

Certain certificate types can be defined based on the artifact areas, the type of conformance, the certification criteria and the achievement levels. A certificate

type indicates a predefined “check” LaQuSo can perform on software artifact. For each certificate type the following items are defined:

Product Area	<The type of input artifact: one of the areas, see 1.2>
Properties	<The type of conformance analysis or the property that has to be checked in case of type c, see 1.3>
Level	<The level for the “Conformance” certification criterion, see Table 1>
Description	<A short description of the goal of the certificate type>
Input	<A precise specification of the input needed for this certificate type>
Checks	<For the relevant achievement levels of the different criteria (see 1.4), a list of checks is included that are part of the certificate type; detailed information on the check can be found in the LaQuSo Software Product Certification Model>

A currently defined certificate type is included in the next chapter.

1.6 LaQuSo Software Product Certification

LaQuSo can carry out any software product verification or validation assignment that is based on the above model:

- to verify behavioral properties of the software artifacts;
- to detect errors or inconsistencies in software artifacts;
- to check compliance to standards or regulations;
- to determine quality attributes of software artifacts.

Output of the certification will be a thorough diagnosis report with explanation of the used methods, tools and techniques plus a verdict. In addition to this LaQuSo can deliver products that were created during the verification or validation, such as models or custom-made tools.

2 MODEL VERIFIED REQUIREMENTS CONSISTENCY

For the consistency of requirements a certificate can be handed out on two different levels: manually verified and model verified. The levels and checks (SC1, SC2 and SC3) are explained in [1].

Product Area	User Requirements
Properties	Consistency
Level	Model verified
Description	Check the requirements process and data models for consistency
Input	Models of the requirements

The following checks need to be answered with 'Yes' or 'Not applicable'. It is explicitly specified in the table below when an item may be marked as 'N/A'. Items with '-' in the third column must always be answered with 'Yes' to obtain a certificate.

Check	Description	N/A if:
<i>Required Elements</i>		
SC1.3 a	Relational diagram of data/objects	-
SC1.3 b	Process models	-
SC1.3 c	Behavioral properties specification	-
<i>Formal checks</i>		
SC3.3 a	Correct workflows	-
SC3.3 b	Mutually consistent	-
SC3.3 c	Normal form	-
SC3.3 d	Match behavioral properties	-
SC3.3 e	Match non-functional requirements	non-funct. req. not available
SC3.3 f	Match environment description	no env. descr. available

A certificate will be handed out if all checks in the above table are answered with 'N/A' (if allowed according to the table) or 'Yes'.

REFERENCES

- [1] Software Product Certification Model. LaQuSo.
SoftwareCertificationModel.doc