Suryn-Abran Consolidated Quality Lifecycle (CQL) Model - the Applicative Evolution

Witold Suryn[1],
[1] Department of Software and IT Engineering, École de technologie supérieure, Montreal, Canada
witold.suryn@etsmtl.ca

Daniel Girard[2]
[2] Department of Software Engineering and Information Technology, École de technologie supérieure
dagirard@cisco.com

Abstract

In 2003 Witold Suryn and Alain Abran have defined and later published the consolidated model for integrating quality engineering into software development processes. The model in its practical nature is meant to be a tool to help the quality specialist in his efforts in implementing and maintaining quality throughout the lifecycle of a software product. This paper presents the results of the analysis and the applicative verification of this model in order to allow for its evolution while maintaining applicability and usability objectives of the initial work.

1 Introduction

The primary objective of this research was to provide a detailed analysis and evolution recommendations of the Suryn-Abran Consolidated Lifecycle Quality (CLQ) model (Fig.1) after its academic and industrial verification. The Suryn-Abran CQL model is the tool that has been developed in order to help a specialist responsible for software quality to analyse, define, implement, measure and evaluate (or so to speak – engineer) the quality of the software product throughout its entire lifecycle. The model and the recommendations for its application were published in [Suryn 2004].

The general approach used when conducting the research was based on the analysis of related recent developments in the domain of standardization of both software engineering and software quality engineering and on the results obtained during several workshop sessions conducted in industry and the academic environment.

The part of the analysis related to standardization focused on the adherence of the CLQ model to internationally known and accepted standards as well as to recently published domain developments. The standards taken into consideration were those published by the International Organisation for Standards (ISO) and by the Institute of Electrical and Electronics Engineers (IEEE) due to their worldwide application and acceptance.

Workshop sessions conducted in the academic environment were executed in the framework of master and Ph.D. specialised course on software quality engineering (SYS 867) given at the École de technologie supérieure in Montreal, Canada and through a dedicated research program realised by the international software quality engineering research group IQUAL (Ingénierie de la Qualité du Logiciel) existing at the École de technologie supérieure and gathering researchers from Europe, USA and Canada.

Workshop sessions that took place at the industry were executed with the participation of specialists responsible for software product quality and software quality assurance in one of the biggest international networking companies.

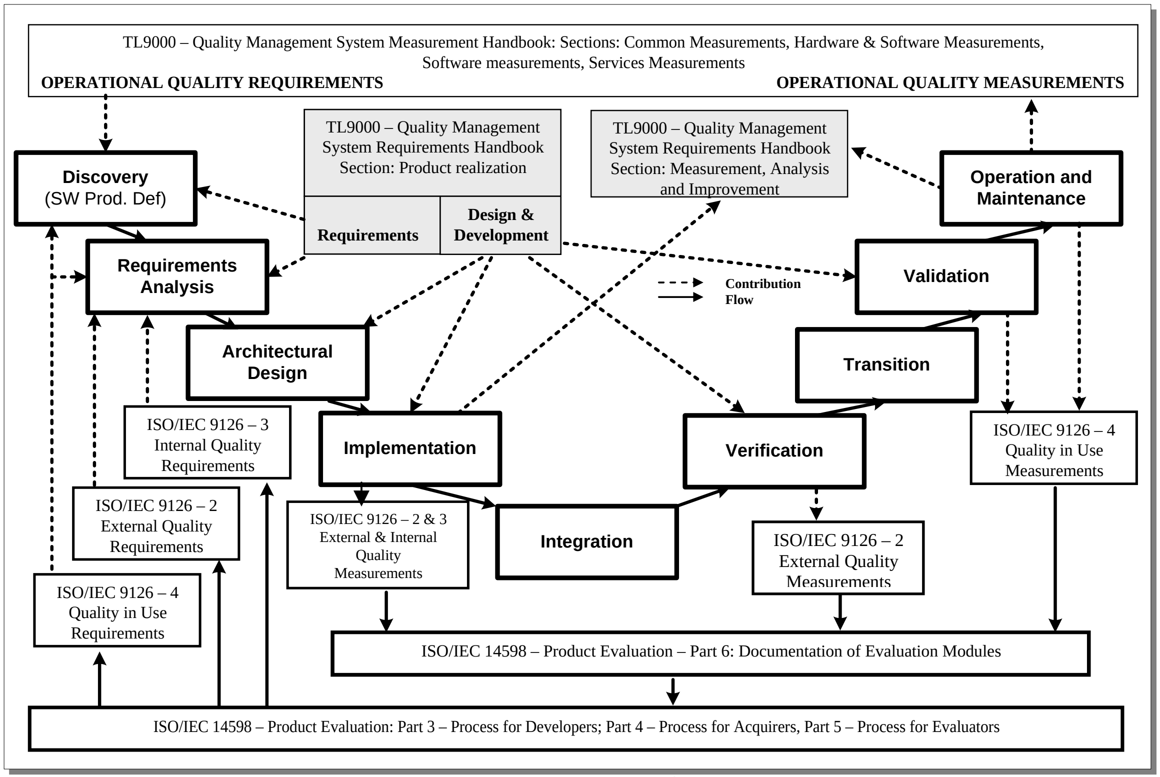
TL9000 – Quality Management System Measurement Handbook: Sections: Common Measurements, Hardware & Software Measurements, Software measurements, Services Measurements
OPERATIONAL QUALITY REQUIREMENTS
OPERATIONAL QUALITY MEASUREMENTS
TL9000 – Quality Management System Requirements Handbook Section: Product realization
Requirements
Design & Development
TL9000 – Quality Management System Requirements Handbook Section: Measurement, Analysis and Improvement
Discovery (SW Prod. Def)
Operation and Maintenance
Requirements Analysis
Validation
Contribution
Flow
Architectural Design
Transition
ISO/IEC 9126 – 3 Internal Quality Requirements
Implementation
Verification
ISO/IEC 9126 – 4 Quality in Use Measurements
ISO/IEC 9126 – 2 External Quality Requirements
ISO/IEC 9126 – 2 & 3 External & Internal Quality Measurements
Integration
ISO/IEC 9126 – 2 External Quality Measurements
ISO/IEC 9126 – 4 Quality in Use Requirements
ISO/IEC 14598 – Product Evaluation – Part 6: Documentation of Evaluation Modules
ISO/IEC 14598 – Product Evaluation: Part 3 – Process for Developers; Part 4 – Process for Acquirers, Part 5 – Process for Evaluators

Figure 1: Suryn-Abran Consolidated Quality Lifecycle (CQL) model version 1.0

2 Analysis of the adherence of CQL model to standards

The first part of the analysis has been conducted through a review of standards making part of the CQL model in its actual version. The following standards have been studied:

- ISO/IEC 9126 - Software Engineering – Product Quality [IEEE 04, ISO 05, ISO 06, ISO 07];
- TL9000 – Measurements Handbook [TL 2001a];
- TL9000 – Requirements Handbook [TL 2001b];
- ISO/IEC 14598 - Information Technology – Software Product Evaluation [ISO 10, ISO 11, ISO 12, ISO 13, ISO 14, ISO 15].

As it can easily be found the major contribution to the CQL model comes from the ISO/IEC 9126 and ISO/IEC TL9000 standards (Fig.2 below presents the common quality model for ISO/IEC 9126 internal and external software product quality, Fig.3 presents the quality in use model and Fig.4 illustrates the TL9000 model). All of those standards are published for several years with some of them being adopted as national standards (for example ISO/IEC 9126 was adopted as the national standard in Japan) what guarantees the appropriate base for the application and normative support for CQL model. As both ISO/IEC 9126 and ISO/IEC 14598 are being actually updated and modernised (SQuaRE project of ISO/IEC JTC1 SC7 Working Group 6) the CQL model will also - in some sense automatically - be updated keeping thus its applicative value intact.

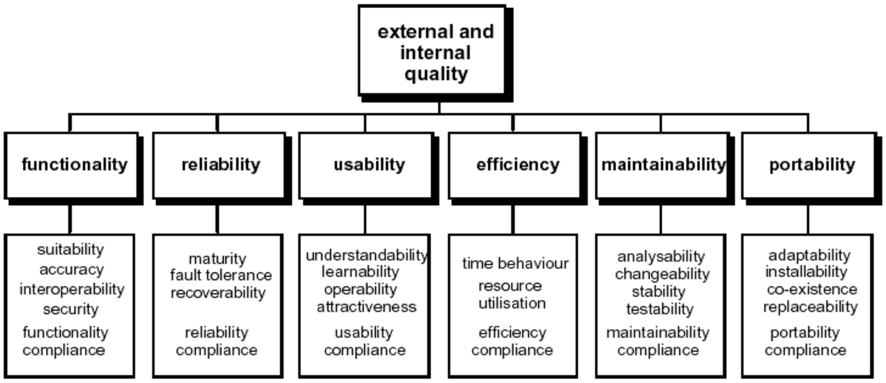

Figure 2: ISO/IEC 9126 Quality Model - External and Internal Quality

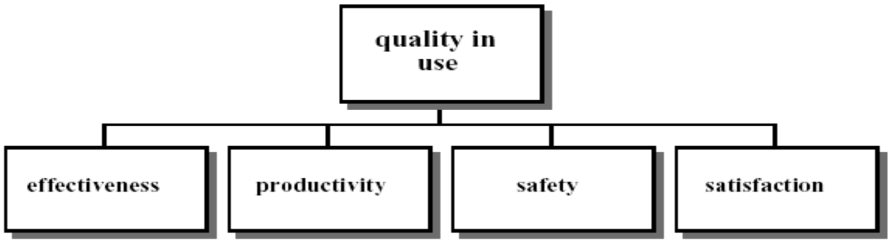

Figure 3: ISO/IEC 9126 Quality Model - Quality in Use

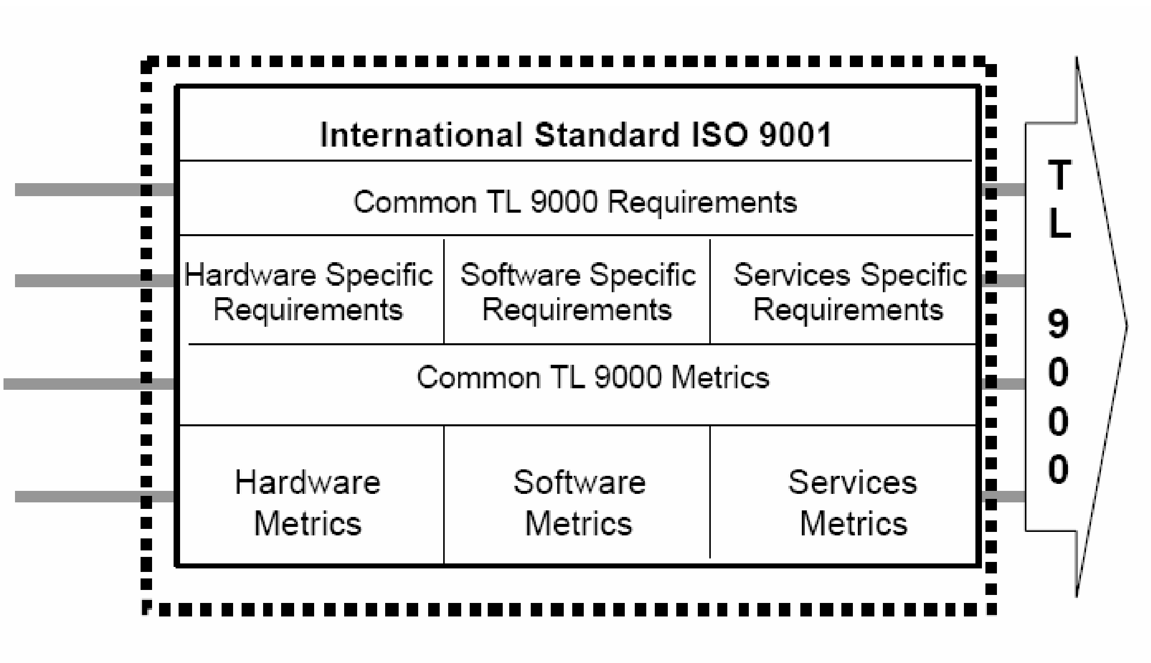
International Standard ISO 9001
Common TL 9000 Requirements
Hardware Specific Requirements
Software Specific Requirements
Services Specific Requirements
Common TL 9000 Metrics
Hardware Metrics
Software Metrics
Services Metrics
TL 9000

Figure 4: ISO/IEC TL9000 Model

Due to the nature of the CQL model and its broad applicability it was decided to also consider other standards for their additional contribution. A special attention has been paid to standards associated with the system lifecycle process:

- ISO/IEC 15288 - System and Software Engineering – System Lifecycle Processes [ISO 17];

- ISO/IEC 12207 - Information Technology – Software Lifecycle Processes [ISO 09];

ISO/IEC 12207 standard [ISO 09] defines the overall activities and tasks necessary to control software projects. It identifies three process categories i.e. primary processes, supporting processes, and organizational processes (Fig.5).

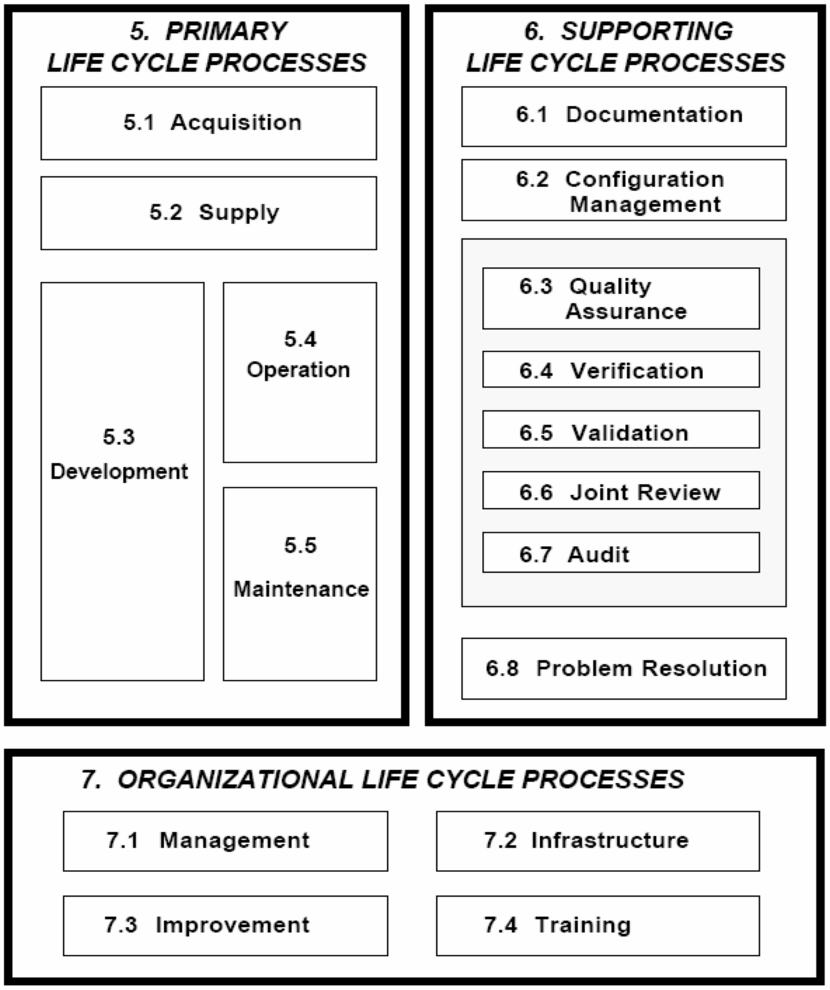

Figure 5: ISO/IEC 12207, Software Lifecycle Process

Analogically to the software lifecycle, ISO/IEC 15288 standard applies to systems. It defines four process categories of which three are related to project control, and the fourth to the organization of the enterprise (Fig.6).

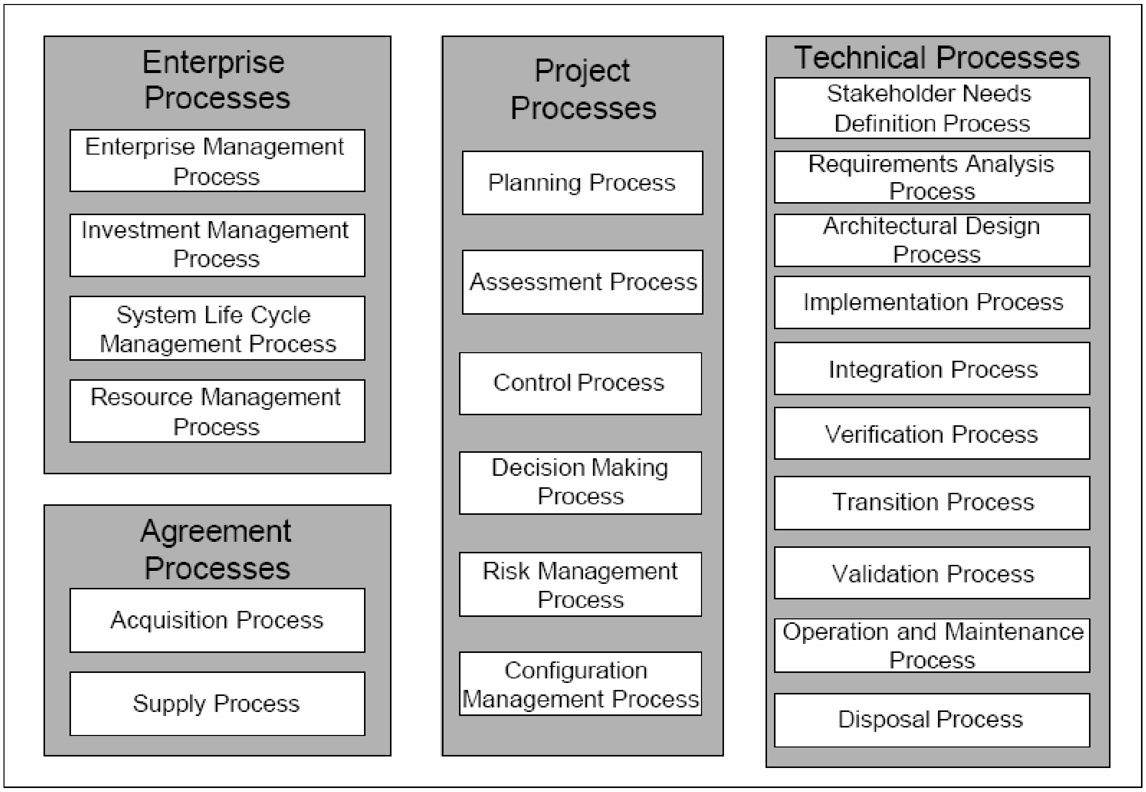

Figure 6: ISO/IEC 15288, Software Lifecycle Process

In both cases it was found crucial for CQL model to adhere to these standards as they represent most recognised and stable modelling of the generic processes within software engineering domain. One of the additional important arguments for applying the ISO/IEC 12207 and ISO/IEC 15288 models in CQL model is the harmonisation project being conducted actually within ISO/IEC JTC1 SC7 having as the objective to align, synchronise and modernise both standards so they reflect the latest developments in software engineering area. Analogically to the case of the adherence to ISO/IEC 9126 and ISO/IEC 14598 the adherence to these above standards should guarantee CQL model its continuous update and modernisation with the usability aspects firmly conserved.

Finally, the following references were also considered:

- ISO/IEC 15939, Software Engineering – Software Measurement Process [ISO 18];
- Project Management Book of Knowledge 2000 [PMI 19];
- ISO 10006, Quality management - Guidelines to Quality in Project Management [ISO 08];
- Guide to the Software Engineering Body of Knowledge [Abran 2004];
- Architecture Tradeoff Analysis Initiative [SEI 2004].

All the above positions represent important references to CQL model auxiliary areas like software measurement processes (helpful in quality measurement and evaluation), project management (with the possibility to add software quality engineering) or full alignment to SWEBOK that would allow CQL model to become an integrated part of this document. The enhancements required to address these areas are planned to be taken into consideration in the continuation of this research.

3 Model Analysis

The analysis conducted during this research proved that the phases of the CQL model having actually little or no quality engineering standards support could, and for the sake of fuller usability and applicability should have such a support. Figure 7 shows both phases where normative support is nonexistent ("Transition" and "Integration") and phases being poor in terms of quality engineering standards that could be securely applied ("Architectural Design").

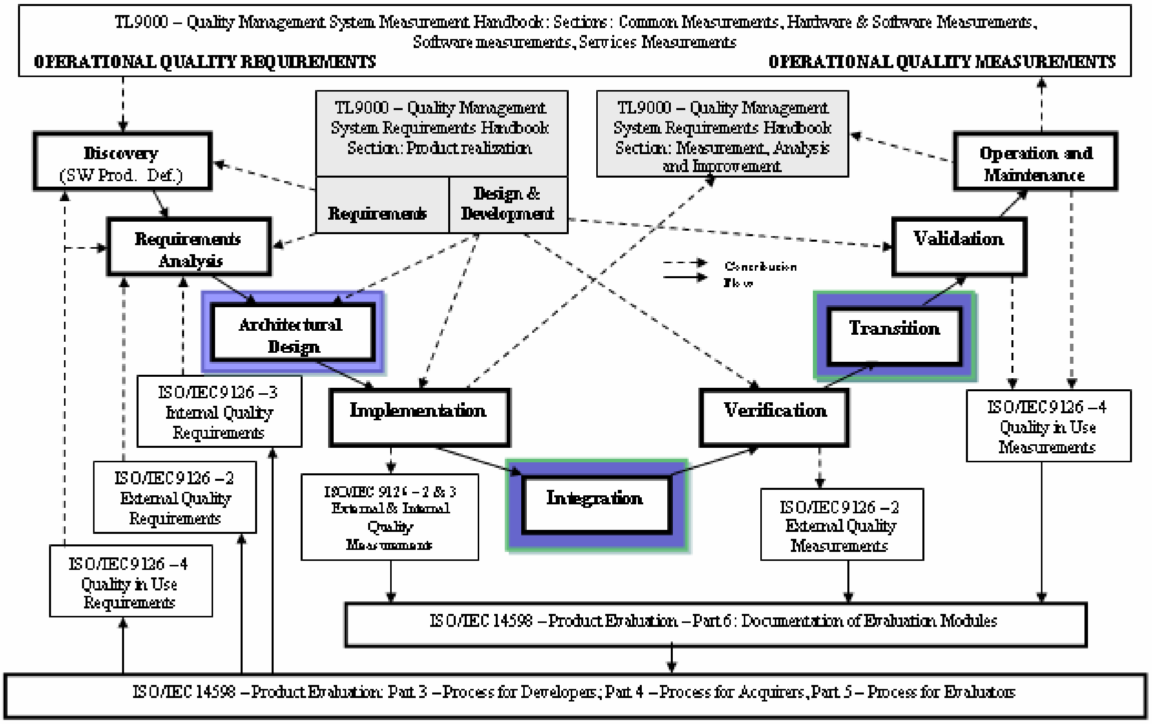
TL9000 – Quality Management System Measurement Handbook: Sections: Common Measurements, Hardware & Software Measurements, Software measurements, Services Measurements
OPERATIONAL QUALITY REQUIREMENTS
OPERATIONAL QUALITY MEASUREMENTS
TL9000 – Quality Management System Requirements Handbook Section: Product realization
Requirements
Design & Development
TL9000 – Quality Management System Requirements Handbook Section: Measurement, Analysis and Improvement
Discovery (SW Prod. Def.)
Operation and Maintenance
Requirements Analysis
Validation
Contribution
Flow
Architectural Design
Transition
ISO/IEC 9126 – 3 Internal Quality Requirements
Implementation
Verification
ISO/IEC 9126 – 4 Quality in Use Measurements
ISO/IEC 9126 – 2 External Quality Requirements
ISO/IEC 9126 – 2 & 3 External & Internal Quality Measurements
Integration
ISO/IEC 9126 – 2 External Quality Measurements
ISO/IEC 9126 – 4 Quality in Use Requirements
ISO/IEC 14598 – Product Evaluation – Part 6: Documentation of Evaluation Modules
ISO/IEC 14598 – Product Evaluation: Part 3 – Process for Developers; Part 4 – Process for Acquirers, Part 5 – Process for Evaluators

Figure 7: Phases of CQL model with poor or no support from quality engineering standards

It can also be noted that some parts of applied standards such as ISO/IEC 9126-1, ISO/IEC 14598-1 and ISO/IEC 14598-2 describing quality models and management practices are not present in CQL model even though they are intrinsic to the generic standards.

The proposed additions to the improved CQL model (named further CQL version 1.1) resulting from the above observations are presented in Fig.8. It is to be noted that the new support to the "Integration" phase in the form of ISO/IEC 9126-2 (External Quality Measurements) has been identified and added.

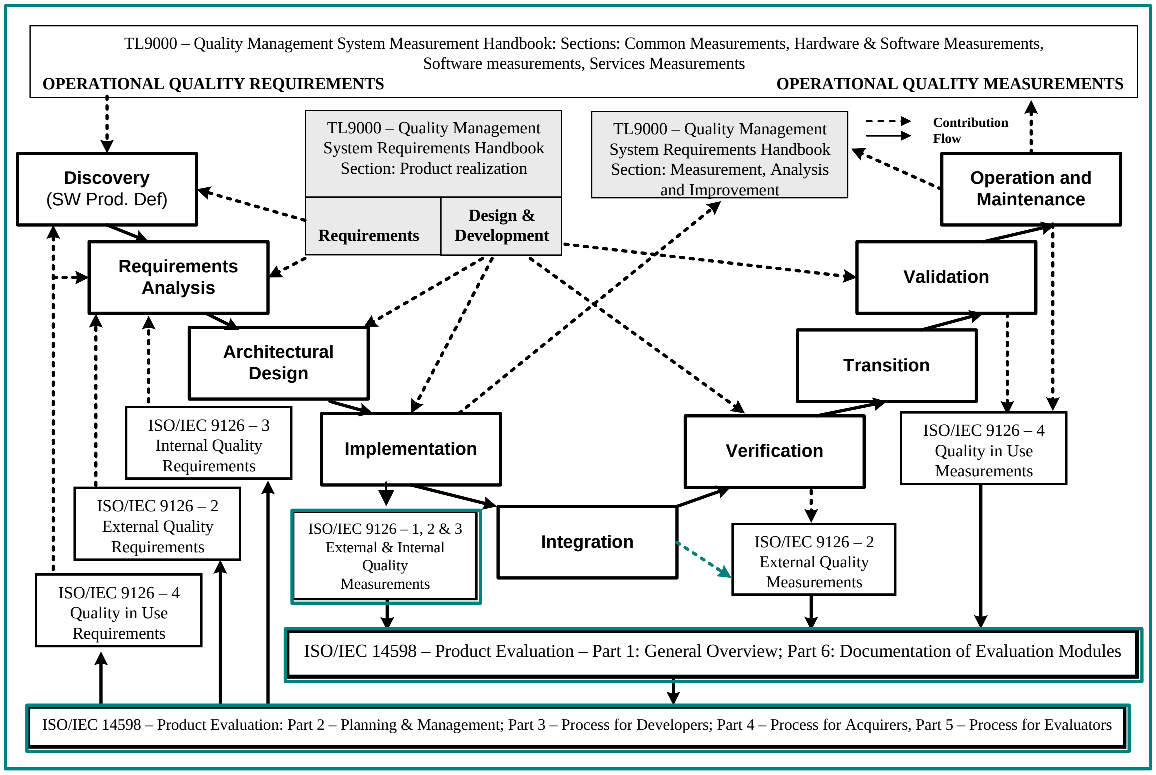
TL9000 – Quality Management System Measurement Handbook: Sections: Common Measurements, Hardware & Software Measurements, Software measurements, Services Measurements
OPERATIONAL QUALITY REQUIREMENTS
OPERATIONAL QUALITY MEASUREMENTS
TL9000 – Quality Management System Requirements Handbook Section: Product realization
Requirements
Design & Development
TL9000 – Quality Management System Requirements Handbook Section: Measurement, Analysis and Improvement
Contribution
Flow
Discovery (SW Prod. Def)
Operation and Maintenance
Requirements Analysis
Validation
Architectural Design
Transition
ISO/IEC 9126 – 3 Internal Quality Requirements
Implementation
Verification
ISO/IEC 9126 – 4 Quality in Use Measurements
ISO/IEC 9126 – 2 External Quality Requirements
ISO/IEC 9126 – 1, 2 & 3 External & Internal Quality Measurements
Integration
ISO/IEC 9126 – 2 External Quality Measurements
ISO/IEC 9126 – 4 Quality in Use Requirements
ISO/IEC 14598 – Product Evaluation – Part 1: General Overview; Part 6: Documentation of Evaluation Modules
ISO/IEC 14598 – Product Evaluation: Part 2 – Planning & Management; Part 3 – Process for Developers; Part 4 – Process for Acquirers, Part 5 – Process for Evaluators

Figure 8: Proposed improvements to CQL model. Boxes with double bold borders contain new normative references, the dashed line arrow from "Integration" phase indicates added normative reference specific for this phase

In course of this research it has also been found that while for the professionals CQL model is relatively simple and easy to follow for the novices it may still remain difficult. In order to further support and simplify the work of industry specialists responsible for software quality engineering a CQL-ISO/IEC 15288 mapping model has been created during this research. The model shown in Fig.9 allows for easier referencing of development models and their related technical processes to these applied in CQL model.

The details of proposed modifications of CQL model are discussed further in phase-by-phase manner.

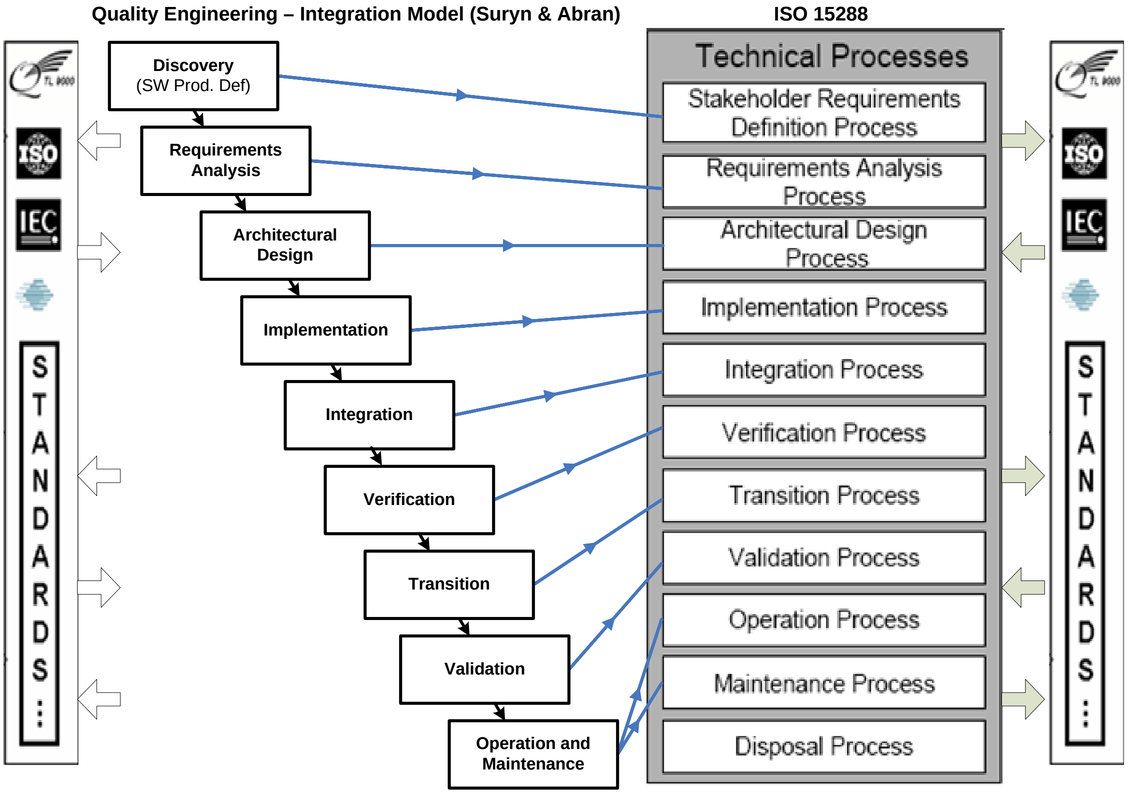

Figure 9: CQL-ISO/IEC 15288 mapping model

4 CQL model version 1.1

The new proposed Suryn-Abran CQL model version 1.1 is presented in Fig.10. For the sake of clarity and ease of application the changes, modifications and required additions are presented only for those CQL model phases where the major modifications appear.

The detailed recommendations for the application of CQL model in each phase during the development process can be found in [Suryn 2004] and so remain out of the scope of this paper.

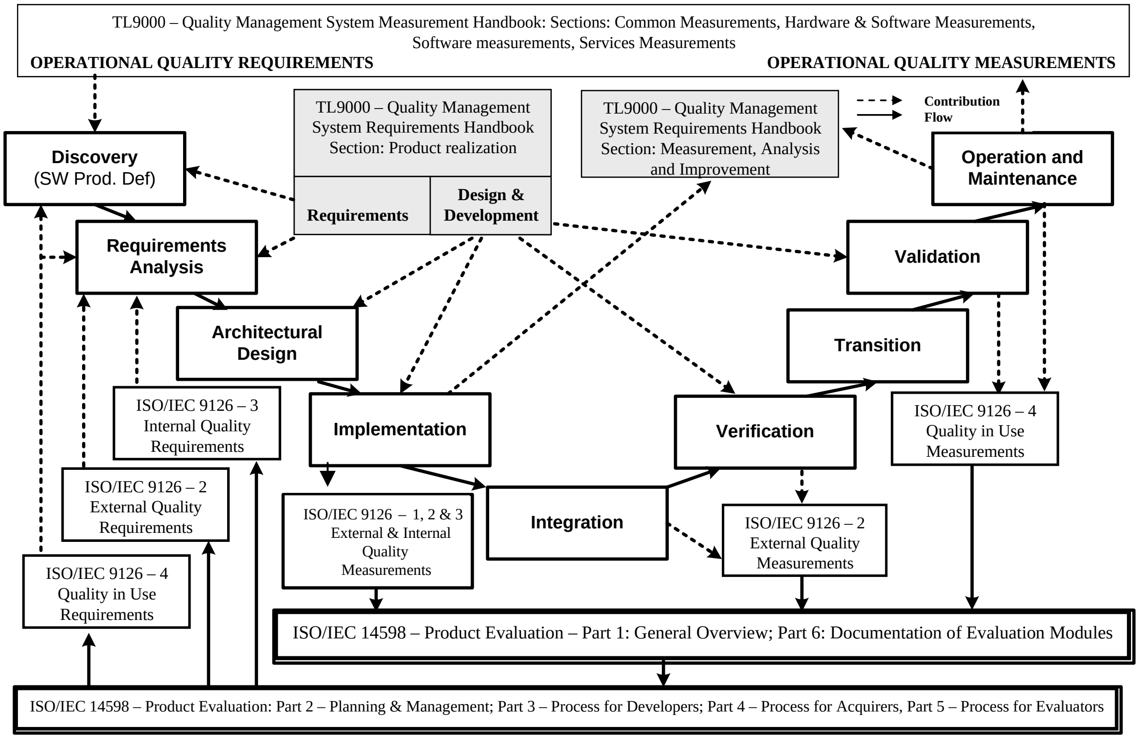
TL9000 – Quality Management System Measurement Handbook: Sections: Common Measurements, Hardware & Software Measurements, Software measurements, Services Measurements
OPERATIONAL QUALITY REQUIREMENTS
OPERATIONAL QUALITY MEASUREMENTS
TL9000 – Quality Management System Requirements Handbook Section: Product realization
Requirements
Design & Development
TL9000 – Quality Management System Requirements Handbook Section: Measurement, Analysis and Improvement
Contribution
Flow
Discovery (SW Prod. Def)
Requirements Analysis
Architectural Design
Implementation
Integration
Verification
Transition
Validation
Operation and Maintenance
ISO/IEC 9126 – 3 Internal Quality Requirements
ISO/IEC 9126 – 2 External Quality Requirements
ISO/IEC 9126 – 4 Quality in Use Requirements
ISO/IEC 9126 – 1, 2 & 3 External & Internal Quality Measurements
ISO/IEC 9126 – 2 External Quality Measurements
ISO/IEC 9126 – 4 Quality in Use Measurements
ISO/IEC 14598 – Product Evaluation – Part 1: General Overview; Part 6: Documentation of Evaluation Modules
ISO/IEC 14598 – Product Evaluation: Part 2 – Planning & Management; Part 3 – Process for Developers; Part 4 – Process for Acquirers, Part 5 – Process for Evaluators

Figure 10: The new Suryn-Abran CQL model version 1.1

4.1 Discovery and Requirements Analysis Phases

The improvement proposed for these phases is the addition of the reference to ISO/IEC 14598-2 – Planning and management [ISO 11]. The rationale for such an addition comes from the nature of these phases where a software quality specialist may have to carefully plan the engineering, measurement and evaluation of the quality of the product (or system) to be developed. ISO/IEC 14598-2 provides details about the planning and management requirements which are associated with software product quality evaluation. While this standard is mainly concerned with product quality evaluation, wherever it is relevant the corresponding process evaluation activities are also discussed

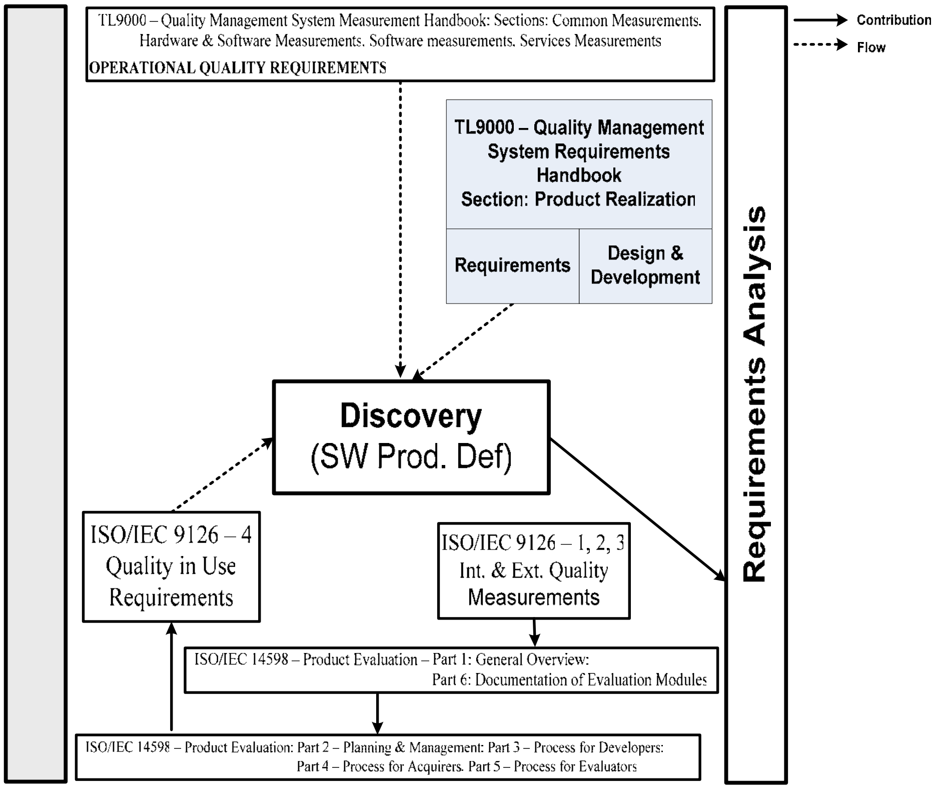
TL9000 – Quality Management System Measurement Handbook: Sections: Common Measurements. Hardware & Software Measurements. Software measurements. Services Measurements
OPERATIONAL QUALITY REQUIREMENTS
TL9000 – Quality Management System Requirements Handbook Section: Product Realization
Requirements
Design & Development
Discovery (SW Prod. Def)
ISO/IEC 9126 – 4 Quality in Use Requirements
ISO/IEC 9126 – 1, 2, 3 Int. & Ext. Quality Measurements
ISO/IEC 14598 – Product Evaluation – Part 1: General Overview: Part 6: Documentation of Evaluation Modules
ISO/IEC 14598 – Product Evaluation: Part 2 – Planning & Management: Part 3 – Process for Developers: Part 4 – Process for Acquirers. Part 5 – Process for Evaluators
Requirements Analysis
Contribution
Flow

Figure 11: Improvements of CQL model in Discovery Phase

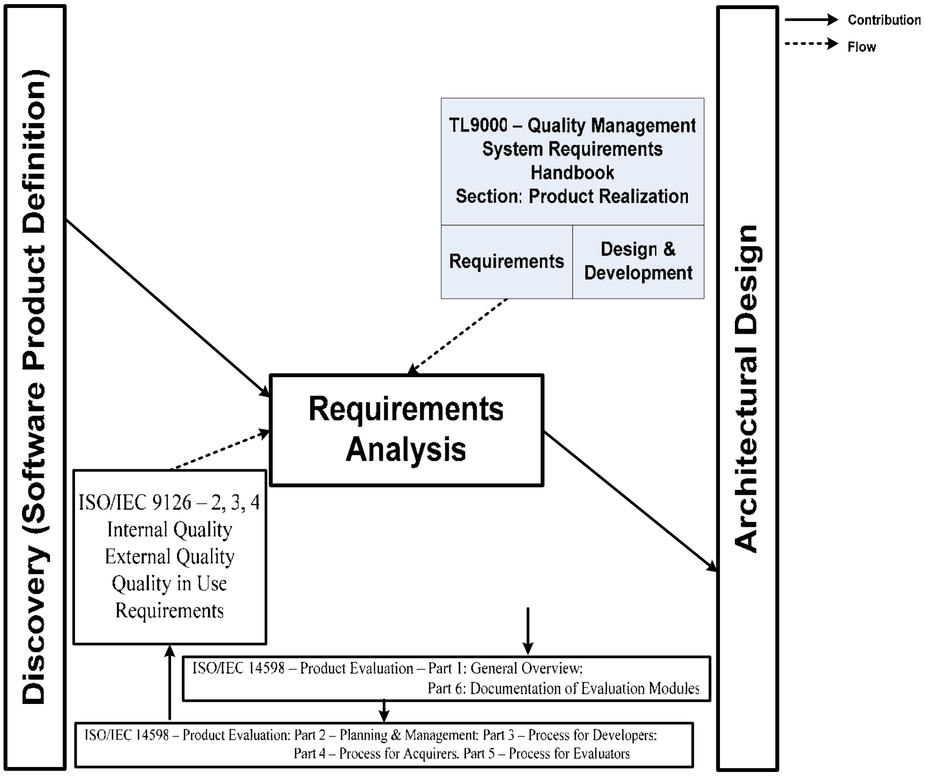
Discovery (Software Product Definition)
TL9000 – Quality Management System Requirements Handbook Section: Product Realization
Requirements
Design & Development
Requirements Analysis
ISO/IEC 9126 – 2, 3, 4 Internal Quality External Quality Quality in Use Requirements
ISO/IEC 14598 – Product Evaluation – Part 1: General Overview: Part 6: Documentation of Evaluation Modules
ISO/IEC 14598 – Product Evaluation: Part 2 – Planning & Management: Part 3 – Process for Developers: Part 4 – Process for Acquirers. Part 5 – Process for Evaluators
Architectural Design
Contribution
Flow

Figure 12: Improvements of CQL model in Analysis Phase

4.2 Architectural Design Phase

As indicated in the Model Analysis section, the normative support for Architectural Design phase is very limited. Figure 13 shows the existing normative support in CQL model reduced to TL9000 System Requirements [TL 2001b] in its Design & Development part.

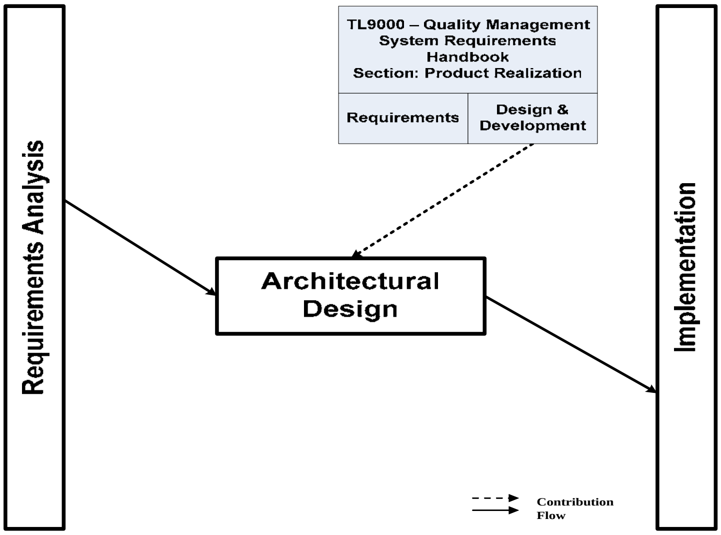

Figure 13: Architectural Design Phase – existing normative support

During this research it was agreed that in the absence of pertinent information on quality, known best practices or processes could be of help in the deficient phases. Following this decision, several broadly recognised processes, practices or industry standards that were found useful for this phase are pointed out as possible solutions allowing for quality engineering provisions.

Fig.14 shows proposed applicable normative support within the phase of Architectural Design of CQL model version 1.1. As it can be observed four categories of support that could influence the quality *in* and *of* the phase were identified:

- related to the specific processes in the phase within the lifecycle (ex: ISO/IEC 15288 [18]);
- related to best practices available for the phase (ex: SWEBOK 2004, Chapter 3 [Abran 2004]);
- related to phase specific techniques (ex: IEEE STD 1016 IEEE Recommended Practice for Software Design Descriptions [IEEE 01]);
- related to phase specific models (ex: ATAM [SEI 2004]).

It can also be observed that all proposed additions are of process-related nature thus remaining of rather indirect influence on the quality of the software product itself. To improve this situation a dedicated research has been undertaken and recently concluded in the Department of Software Engineering and Information Technology at the École de technologie supérieure, Montreal, Canada. The obtained results will be available in pending publication of *Suryn W., Djilive F., "Quality Design Process for Software System Architecture"*.

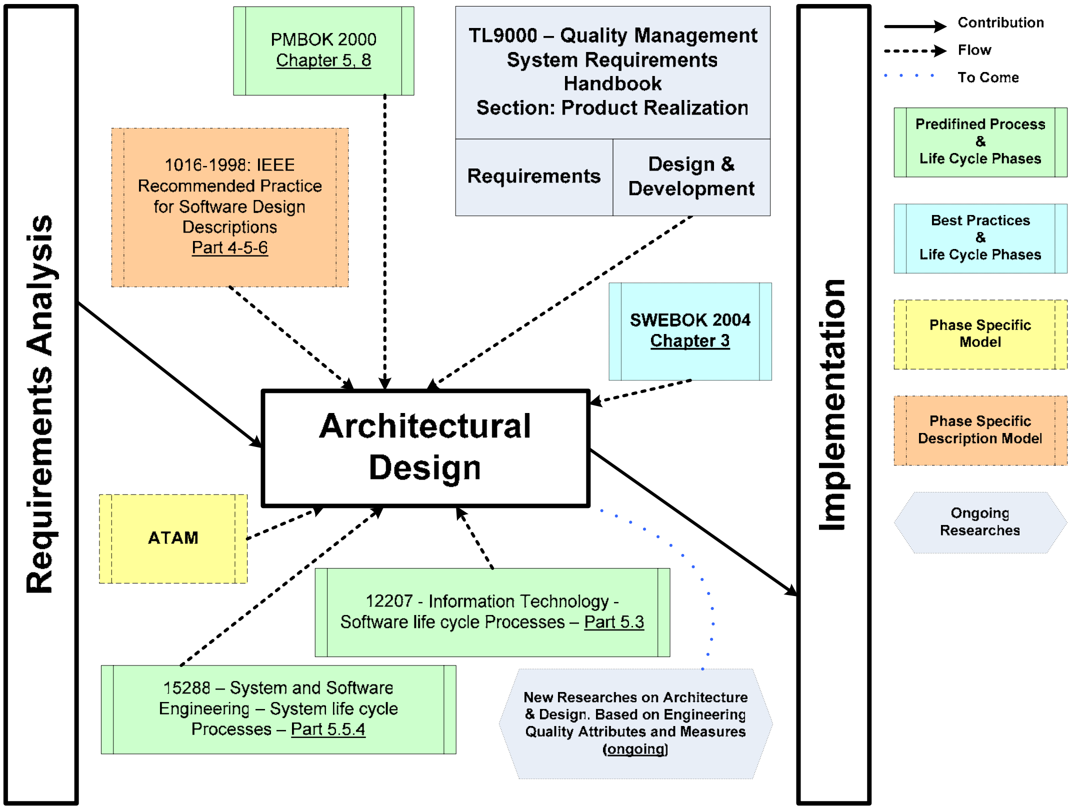
Requirements Analysis
PMBOK 2000
Chapter 5, 8
TL9000 – Quality Management System Requirements Handbook
Section: Product Realization
Requirements
Design & Development
1016-1998: IEEE Recommended Practice for Software Design Descriptions
Part 4-5-6
SWEBOK 2004
Chapter 3
Architectural Design
ATAM
12207 - Information Technology - Software life cycle Processes – Part 5.3
15288 – System and Software Engineering – System life cycle Processes – Part 5.5.4
New Researches on Architecture & Design. Based on Engineering Quality Attributes and Measures
(ongoing)
Implementation
Contribution
Flow
To Come
Predifined Process & Life Cycle Phases
Best Practices & Life Cycle Phases
Phase Specific Model
Phase Specific Description Model
Ongoing Researches

Figure 14: Improvements of CQL model in Architectural Design Phase

4.3 Integration Phase

The Integration phase consists of assembling all the modules produced during the Implementation phase into a linked group of co-operating modules. The new sub-system, as a collection of integrated modules, will need to be measured as a new entity. The recommended procedure is as follows:

.

- Documents to be used: ISO/IEC 9126 – Part 2 (*ISO/IEC 9126-2*, 2001)
- Documentation of measurements. Document to be used: ISO/IEC 14598 – Part 1 (*ISO/IEC 14598-1*, 1996) and ISO/IEC 14598 – Part 6 (*ISO/IEC 14598-6*, 1999)
- Evaluation of the quality of the new intermediate products. Documents to be used, depending on the position of the evaluating entity: ISO/IEC 14598 – Part 3: Process for Developers (*ISO/IEC 14598-3*, 1996), Part 4: Process for Acquirers (*ISO/IEC 14598-4*, 1996) or Part 5: Process for Evaluators (*ISO/IEC 14598-5*, 1996)

It should be stressed that due to the classification of the product created in the phase of integration (a collection of modules that can co-operate but still not a functional system) the measurements and evaluation will rather be focused on the inter-module level than on the system level.

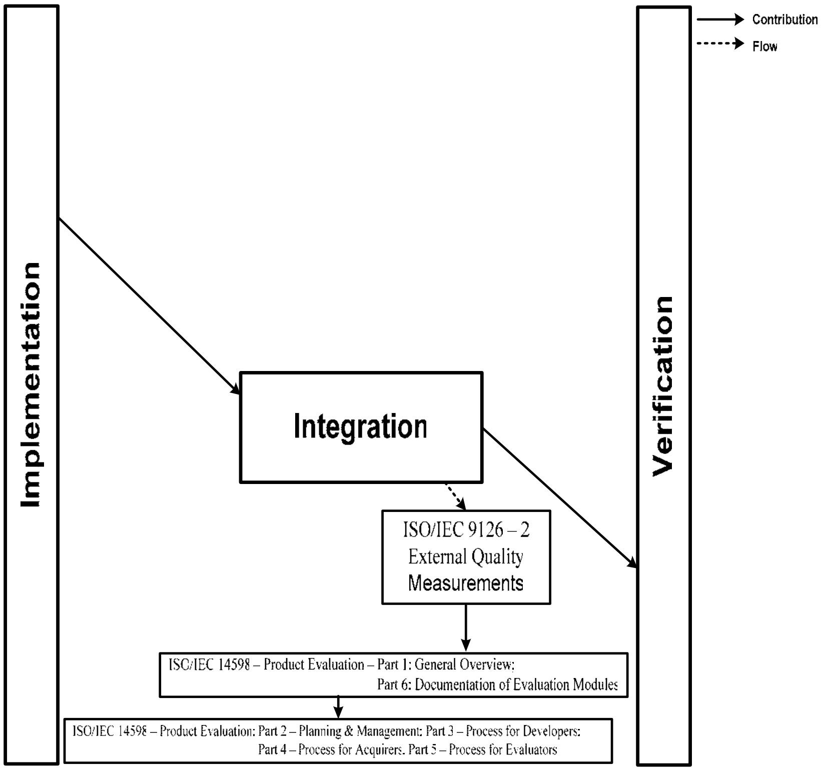
Contribution
Flow
Implementation
Verification
Integration
ISO/IEC 9126 – 2
External Quality
Measurements
ISC/IEC 14598 – Product Evaluation – Part 1: General Overview;
Part 6: Documentation of Evaluation Modules
ISO/IEC 14598 – Product Evaluation: Part 2 – Planning & Management: Part 3 – Process for Developers:
Part 4 – Process for Acquirers. Part 5 – Process for Evaluators

Figure 15: Improvements of CQL model in Integration Phase

4.4 Transition Phase

This phase covers all activities required for implementing the system in its production environment in order to allow the end user to validate and accept the new system to be deployed. Similarly to Architectural Design phase it was agreed that in the absence of pertinent information on quality, known best practices or processes could be of help in this phase of CQL model; however the analysis performed during this research indicated that even procedural or process-oriented quality-related normative support for this phase is severely limited (SWEBOK, ISO/IEC 15288 and ISO/IEC 12207 mostly). Such a state-of-the-art gave the impulse for launching the new research program within IQUAL research group entitled “Software Quality Engineering Processes and Models for Software Transition Lifecycle Phase”. The results are expected to be published in 2005.

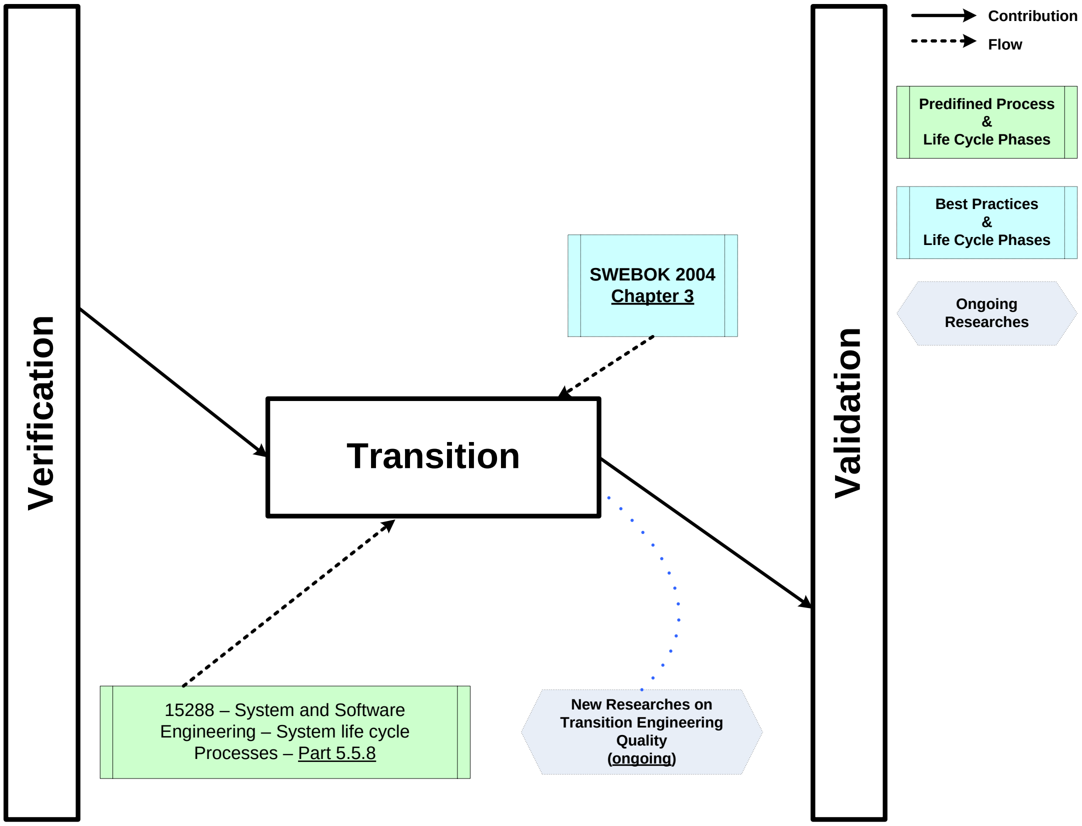
Contribution
Flow
Predifined Process & Life Cycle Phases
Best Practices & Life Cycle Phases
Ongoing Researches
Verification
Validation
Transition
SWEBOK 2004 Chapter 3
15288 – System and Software Engineering – System life cycle Processes – Part 5.5.8
New Researches on Transition Engineering Quality (ongoing)

Figure 16: Improvements of CQL model in Transition Phase

5 Conclusions

During the research program presented in this paper the analysis of Suryn-Abran Consolidated Quality Lifecycle (CQL) model has been thoroughly executed. The analysis was performed in both academic and industrial environments rendering several observations allowing for improvements and the evolution of CQL model.

As the result of the analysis the improvements proposed to phases of Architectural Design, Integration and Transition were implemented in version 1.1 of the CQL model. Two additional specialised research programs resulted from the one presented in this paper with the results to be publicly available in 2005. Other research studies are also underway to support CQL model with separate layers of project management, risk analysis and predefined processes. A handbook of industrial practices for quality engineering is being prepared with a plan to be published in 2006.

6 Acknowledgements

The authors would like to express their deep gratitude to Mr. Hassan Ait El Cadi and Mr. Denis Beauchemin for their considerable contributions in this research.

7 Bibliography

[Abran 2004] Abran, A., Moore, J. W., Bourque, P., Dupuis, R., & Tripp, L. L. (2004). Guide to the Software Engineering Body of Knowledge (2004 Version Ed.). Los Alamitos, California: IEEE Computer Society.

[IEEE 01] IEEE STD 1016 IEEE Recommended Practice for Software Design Descriptions. IEEE 1998.

[IEEE 02] IEEE STD 1016.1 IEEE Guide to Software Design Descriptions. IEEE 1993.

[IEEE 03] IEEE STD 1471: IEEE Recommended Practice for Architectural Description of Software-Intensive Systems. IEEE 2000.

[IEEE 04] ISO/IEC 9126-1: Software Engineering - Product Quality - Part 1: Quality Model. (1999).

[ISO 05] ISO/IEC 9126-2: Software Engineering - Product Quality Part 2 - External Metrics. (2001).

[ISO 06] ISO/IEC 9126-3: Software Engineering - Product Quality Part 3 - Internal Metrics. (2001).

[ISO 07] ISO/IEC 9126-4: Software Engineering – Software Product Quality - Part 4: Quality In Use Metrics. (2001).

[ISO 08] ISO/IEC 10006: Quality Management - Guidelines to Quality in Project Management. (1997).

[ISO 09] ISO/IEC 12207, Information Technology - Software Lifecycle Processes. (1995).

[ISO 10] ISO/IEC 14598-1: Information Technology - Software Product Evaluation - Part 1: General Overview. (No. 6/N376)(1996).

[ISO 11] ISO/IEC 14598-2: Information Technology - Evaluation of Software Products - Part 2: Planning and Management. (No. 6/N376)(1995).

[ISO 12] ISO/IEC 14598-3.2: Information Technology - Software Product Evaluation Part 3: Process for Developers. (No. 6/N376)(1996).

[ISO 13] ISO/IEC 14598-4: Information Technology - Software Product Evaluation - Part 4: Process for Acquirers. (No. 6/N376)(1996).

[ISO 14] ISO/IEC 14598-5: Information technology - Evaluation of Software Product - Part 5: Process for Evaluators. (No. 6/N376)(1996).

[ISO 15] ISO/IEC 14598-6: Software Engineering - Product Evaluation - Part 6: Documentation of Evaluation Modules. (No. 6/N376)(1999).

[ISO 16] ISO/IEC 14598, Information Technology - Software Product Evaluation. (No. 6/N376)(1999).

[ISO 17] ISO/IEC 15288: System and Software Engineering - System Lifecycle Processes. (2002).

[ISO 18] ISO/IEC 15939: Software Engineering - Software Measurement Process. (2002).

[PMI 19] Project Management Institute. (2000). A Guide to the Project Management Body of Knowledge (PMBOK) (2000 Ed.). Pennsylvania.

[SEI 2004] SEI. (2004). Architecture Tradeoff Analysis Initiative Model, from www.sei.cmu.edu

[Suryn 2004] Suryn, W., Abran, A., & Laporte, C. An integrated lifecycle quality model for general public market software products. Proceedings of the international conference SQM & INSPIRE 2004, Canterbury, Kent, UK. 5-7 April 2004

[TL 2001a] TL9000 - Quality Management System Measurements Handbook. (2001a).

[TL 2001b] TL9000 - Quality Management System Requirements Handbook. (2001b).