Introducing Root-Cause Analysis and Orthogonal Defect Classification at Lower CMMI Maturity Levels

Luigi Buglione[1] and Alain Abran[1]

[1] École de Technologie Supérieure - ETS
1100 Notre-Dame Ouest, Montréal, Canada H3C 1K3
Luigi.buglione@computer.org; alain.abran@etsmtl.ca

Abstract. This paper discusses and analyzes possible solutions for achieving an effective process improvement in one specific key process area: measurement, whatever the maturity level and without the constraints of a software process improvement model staged representation. It investigates in particular a Support Process Area, that is, Causal Analysis & Resolution (CAR), together with Orthogonal Defect Classification.

Keywords: Root-Cause Analysis (RCA), CMMI, Capability Level, Orthogonal Defect Classification (ODC), Total Quality Management (TQM).

1 Introduction

Over the past few years in the software engineering community, there has been a growing interest in process improvements and in measurement to support decision-making. Process improvement (PI) can be "measured" and benchmarked against several reference models, and, in the software engineering community, the two best-known benchmarking models are probably:

- **ISO 9001:2000** [28], which is a general domain requirement model that can be instantiated to the software engineering domain using, for instance, the ISO 90003 standard [17];
- **CMMI** (Capability Maturity Model Integration) version 1.2 [11], which is, specifically, a software engineering process model applicable to both the software and systems engineering domains.

While most organizations adopt only one of the two benchmarking models, a few adopt both, and there already exist two-way mappings to compare ISO requirements to CMMI practices for each process.

A few studies have investigated the maturity level equivalence for those organizations already ISO 9001:2000-certified and implementing CMMI or SPICE [22] processes between maturity levels 2 and 3 [24], and they have raised a few issues. For instance, an ISO-certified organization must – to be certified – demonstrate that they have a process in place to identify and eliminate the causes of non conformities (§8.5 – Improvement), implicitly through a Root-Cause Analysis (RCA). This means that, for these ISO 9001-certified organizations, there should be documented evidence of some measurement-intensive process areas (PAs), such as

Causal Analysis & Resolution (CAR), which would correspond to the evolution of the Defect Prevention key process area (KPA) at maturity level 5 in the older Sw-CMM [23]) or *Decision Analysis & Resolution (DAR)* at maturity level 3.

However, in a SCAMPI appraisal for the CMMI model [18] adopting a staged evaluation against maturity level 2, these level 3 and level 5 measurement processes would not even be looked into, thereby possibly underrating the real maturity level of such organizations. By contrast, if these organizations were assessed using the continuous representation, this would not be an issue: all processes must then be assessed one by one against the capability process attributes, from level 0 up to level 5, whether or not all the processes connected by a particular maturity level have been implemented [13], as is the case in the staged representation.

Most organizations cannot implement processes at a higher maturity level all at once and from scratch; in practice, they progressively introduce the elements of a higher-level practice by starting with the easiest fit in their own environment, and gradually learning how to master this process, often initially within the limited scope of a pilot project. Only later would these organizations fully deploy a new process, either for business reasons or for a formal maturity assessment. Whatever the kind of model representation chosen (staged or continuous) [2][5][21], it is important to remember why an organization should adopt one or the other. There is no one answer that is valid for every organization: size, number of employees, business type, time-to-market pressure, the business process model adopted and target certifications requested by clients as prerequisites for participating in bids all are examples of some of the parameters to take into account when selecting either a staged or a continuous representation for assessment and benchmarking purposes.

ISO 9001:2000 requires RCA for achieving certification, and it could reasonably be argued that it be positioned within either level 2 or level 3 of CMMI[1]. Positioning the CAR process (a CMMI-related process) at level 5 can be challenged:

Is the CAR practice indeed observed only in organizations with high-level maturity? Could it be introduced at lower maturity levels, and, if so, in what way?

More specifically, can an earlier use of RCA help an organization achieve higher CMMI maturity and capability levels faster?

This paper discusses and analyzes possible strategies for achieving an effective PI by the application of Total Quality Management (TQM) measurement-based tools, whatever the maturity level and the kind of representation (staged or continuous) chosen. This paper focuses in particular on the CAR process of the CMMI Support PA.

Section 2 presents an overview of the role of the Support Processes in PI initiatives. Section 3 presents related work on CAR process and identifies outstanding issues. Section 4 proposes a quantitative usage of CAR as a foundation for achieving higher maturity levels. Section 5 presents a discussion on suggestions concerning the adoption of this quantitative approach to CAR and the benefits of doing so.

---

[1] According to [24], there is an indicative maturity correspondence between Sw-CMM ML2-3 companies and ISO 9001:1994-certified ones. Taking into account the newer mappings between their respective evolutions (CMMI vs Sw-CMM; ISO 9001:2000 vs 9001:1994), such maturity level equivalence can be assumed.

2 Support Processes in the CMMI

CMMI proposes a classification of process areas (Pas) by typology, grouping them into four classes[2]:

- **Process Management** – includes the cross-project activities related to the defining, planning, deploying, implementing, monitoring, controlling, appraising, measuring and improving processes.
- **Project Management** – includes the project management activities related to planning, monitoring and controlling the project.
- **Engineering** – includes the development and maintenance activities that are shared across engineering disciplines. The engineering PAs were written using general engineering terminology, so that any technical discipline involved in the product development process (e.g. software engineering or mechanical engineering) can use them for process improvement.
- **Support** – includes the activities to support product development and maintenance. The Support PAs address processes that are used in the context of performing other processes. In general, they address processes that are targeted toward the project, and may address processes that apply more generally to the organization.

The Support Processes in the CMMI model are listed in Table 1 – they are divided into Basic and Advanced PAs, and their respective purpose and related General Practices (GPs) are included.

In particular, CMMI states that the Basic Support PAs "*address fundamental support functions that are used by all process areas. Although all Support process areas rely on the other process areas for input, the Basic Support process areas provide support functions that also help implement several generic practices*" while Advanced PAs "*provide the projects and organization with an improved support capability.*"

As can be inferred from Table 1, maturity level 2 Support processes play a dual role in CMMI:

- as **PA**s, and
- as **GP**s

This dual role helps organizations in building the foundations for achieving improvements and contributing to reaching higher maturity levels faster. For instance, a proper Measurement & Analysis (MA) implementation has positive impacts, both on the PAs, PMC and PPQA[3], as well as on the ratings of two GPs, GP3.2 – *Collect Improvement Information* – and GP 4.2 – *Stabilize Subprocess Performance*.

---

[2] ISO 15504 proposes a similar classification, through the use of five groups, adding a Management (MAN) group. See http://www.isospice.com for details about ISO standard parts and status (parts 1-5 have already been published, and parts 6 & 7 are under development).

[3] A list of acronyms is provided in Appendix A.

Table 1. CMMI v1.2 Support Process Areas (PAs).

Maturity Level	Process Area – PA	Title	Process Area Purpose	Related General Process - GP
Basic				
ML2	CM	Configuration Mgmt	Establish and maintain the integrity of work products using configuration identification, configuration control, configuration status accounting and configuration audits	GP2.6
ML2	PPQA	Process & Product Quality Assurance	Provide staff and management with objective insight into processes and associated work products	GP2.9
ML2	MA	Measurement & Analysis	Develop and sustain a measurement capability that is used to support management information needs	GP2.8
Advanced				
ML3	DAR	Decision Analysis & Resolution	Analyze possible decisions using a formal evaluation process that evaluates identified alternatives against established criteria	N.A.
ML5	CAR	Causal Analysis & Resolution	Identify causes of defects and other problems, and take action to prevent them from occurring in the future	GP5.2

3 The Causal Analysis & Resolution (CAR) PA

3.1 CAR Process

CAR is a CMMI process at level 5, and it is expressed by two Specific Goals (SGs) split into five Specific Practices (SPs) – see Table 2.

Table 2. CMMI v1.2 CAR Support PAs

Specific Goals – SG	Specific Practices – SP
SG.1	**Determine Cause of Defects**
	SP1.1 - Select defect data for analysis
	SP1.2 – Analyze causes
SG.2	**Address Cause of Defects**
	SP2.1 – Implement the action proposals
	SP2.2 – Evaluate the effect of changes
	SP2.3 – Record data

3.2 Tools for CAR

In particular, SP1.2 sub-practice #2 asks for the following: "*Analyze selected defects and other problems to determine their root causes.*" A recommended analytical tool from TQM [7][8][9] is the Fishbone diagram (or Ishikawa or Cause-Effect diagram) [6]. This quality tool is useful for detecting the root causes of a defect/problem, and for classifying and prioritizing issues in a well-established and ordered manner. In more general terms, as explained in the introduction, the process for detecting and solving problems is usually referred to as RCA (Root-Cause Analysis) in the CMMI practices within the CAR (Causal Analysis & Resolution) process. In Figure 1, an example is presented, where the defect to analyze and remove was "*Software Defects*".

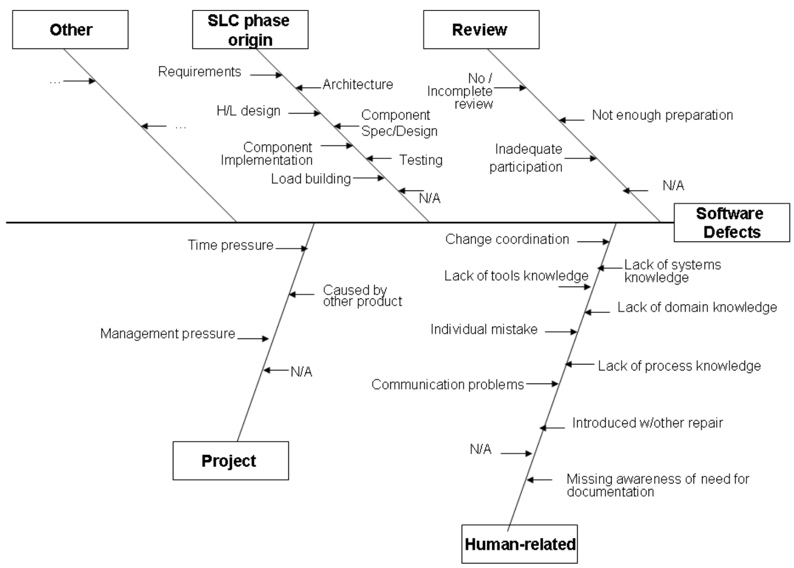

Fig. 1 – *Factors contributing to the high rate of software project failures*[4]

3.3 Related work: Why is CAR positioned at level 5?

The positioning of CAR at level 5 seems to have come about as a result of an assumption by CMMI architects that CAR can be considered as an "evolution" of the "*Defect Prevention*" KPA from the old Sw-CMM, as mentioned in the introduction. On a few occasions, it has been suggested that both RCA and CAR be initially introduced in a qualitative approach at lower maturity levels before including them in a quantitative approach at higher maturity levels:

[4] Elaboration from [15]. Another possible classification for software failures is proposed in [1].

- Williams (2002) [14] mapped the specific CAR goals against Juran's 10 points, suggesting intensive use of qualitative and quantitative TQM tools for each CAR SP, but without providing suggestions about the "how to" on each tool listed in the fourth point of this list (*Identify root causes*)
- Norausky (2003) [10] proposed a "distributed usage" of CAR across the five maturity levels, using a "*hybrid implementation approach*" for CAR, which would constitute a parallel continuous improvement on CAR implementation, while also pursuing an overall staged representation implementation strategy for those organizations adopting this representation. However, no detailed suggestions are provided for individual maturity levels (from maturity level 1 on), but only suggested usage of high-level CAR measurement.

Thus, a possible solution could be to use CAR at lower maturity levels, and to apply it in a quantitative manner.

4 Quantitative CAR as a foundation for use at Higher Maturity Levels

4.1 From RCA to ODC – Related work

To allow comparability among several instances for a certain problem/effect, Ram Chillarege proposed a technique in the early '90s called Orthogonal Defect Classification (ODC) [3][4], as a way of categorizing defects found both during the development process and after customers receive and begin using the product.

In ODC, defects are classified according to key attributes and then data are analyzed to form the basis for action plans and process improvement activities. ODC is a technique mid-way between the traditional RCA (more qualitative and time-consuming) and Statistical Defect Models (more quantitative, but not easily translatable into corrective action). Through the *orthogonal* classification of defects found (*defect type*) and their association with their *trigger*, it is possible to create consistent and meaningful characterizations of the problems that are found across all software development life cycle stages.

4.2 ODC: Strengths and limitations

A list of strengths and possible limitations in applying ODC has been compiled from the literature:

<u>Strengths</u> [20][25][26] [27]:

- It is an evolution of RCA from a qualitative to a quantitative approach.
- It has adopted a standard taxonomy (types; triggers), which allows comparability across time and organizations.

- It helps in gathering defect data over time, enabling an organization to run statistical analysis and – more generally – to look at defect data in a more objective way.

Limitations:

- It is challenging to use Software Defect Removal, since a large part of the SPI activity is focused on the code. Furthermore, the later a generic defect (not only code) is detected, the more difficult and costly it is to remove [12].
- It is typically applied by organizations having a robust measurement system: ODC needs the capability to consistently gather and analyze data over time; a number of organizations are at lower maturity levels and do not have this capability, or the payback period is too remote for its application to be economical.
- The updating of defect *types* and related defect *triggers* makes it difficult to maintain a backward comparability of source defect data over a long period of time.

4.3 Generalizing and customizing ODC

Our proposed approach is to customize the ODC principles to each implemented PA. In particular, the suggestion is to quantify RCA using the GQM-GQ(I)M [19] approach in the following way: to each low-level leaf (or bone) in a Fishbone diagram, each organization can collect its own groups of causes and adopt this tool whatever its maturity level, and start to do so as early as possible.

Figure 3 illustrates how to determine new, useful measures, or use existing ones, in each related process within a leaf/bone. These measures are shown in blue in Figure 3.

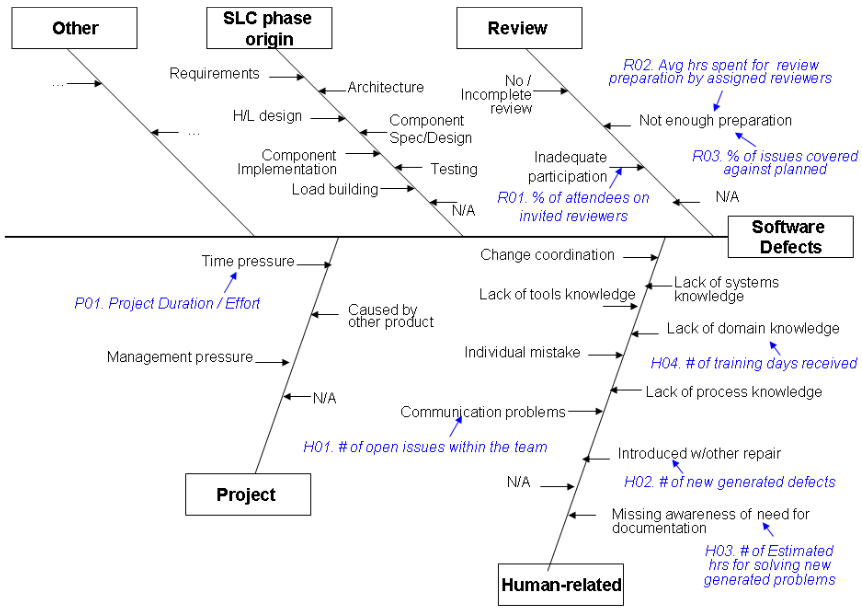

Fig. 2 – *Measures applied to the final bones in a Fishbone diagram*

Some practical guidance is suggested here:

- Build your own defect *types* and *triggers* for each implemented PA – to be refined over time: this will make it possible to define a personalized local standard taxonomy (or start by using RCA classifications such as 3Ms and P (Methods, Machinery, Materials, People) or 4Ps (People, Process, Procedure, Plans).
- Link the measures detected from RCA to their related processes: the suggestion provided by CMMI of gathering only the number and typology of measures[5] seems to be limited to monitoring and controlling CAR. Taking into account more measures derived after the "quantitative" implementation of RCA can help the organization to succeed in the practice of collecting improvement information faster and more easily (GP3.2).

Possible outcomes of such implementation can be:

- Facilitate the adoption of (new) measures needed for removing defects and related causes;
- Facilitate the data collection process in the organization: this is the foundation for statistical analysis later on, at maturity level 4;
- Reduce the cost of non quality (CONQ) in the medium to long term and improve this ratio over the cost of quality (COQ) (e.g. CONQ/COQ ratio): it is reported in [29] that the return on investment from the removal of a cause (these costs are related to CONQ items) is higher than that for removing a defect;

---

[5] See CAR GP2.8, "Elaboration" section.

- Facilitate the proper implementation of other processes (i.e. Project Monitoring & Control (PMC) and Project and Process Quality Assurance (PPQA)) and general practices such as monitoring and controlling the process (GP2.8), collecting improvement information (GP3.2) and stabilizing sub-process performances (GP4.2) by more skilled resources[6].

5 Discussion

An important goal for every organization is to achieve a real and valuable improvement, with the result that it moves up through the maturity levels. Process improvement models constitute a roadmap for doing so, describing the steps to follow and the techniques and tools to implement. Whatever the model and kind of representation chosen, it is fundamental to properly understand the underlying appraisal principles for the rating process and for deriving useful suggestions for improving processes and related outcomes and outputs.

Measurement and Causal Analysis represent two powerful analytical tools for pursuing PI; they are classified as Support processes by the CMMI model, and should be used and managed in the same way ("dual role" and allocation at a certain maturity level).

Two suggestions for improving the CMMI architecture have been proposed:.

- Introduction of the CAR PA area at maturity level 2, as a Basic (rather than an Advanced) Support Process.
  - Supporting rationale: if the software and systems engineering community recognizes Cause-Effect detection and removal ability as a basic process improvement principle, which is also mandatory for ISO 9001 certification and corresponds approximately to CMMI level 2 or 3, then it would be more coherent to classify CAR as a basic process at level 3 than strictly as an advanced process at level 5; this would also improve consistency across both ISO and CMMI benchmarking models.
- Addition of a direct reference to CAR in the general practice related to the capability of adhering to internal processes and policies (GP2.9),: if RCA (and therefore CAR) were to be recognized as a CMMI basic practice, then GP2.9 could be reinforced by introducing a reference to the CAR process[7]. This would help in overcoming the possible risk of maintaining a conservative view of quality (*Quality Assurance*) rather than a proactive one (*Quality Improvement*), which should be at the core of TQM, and therefore of SPI practices.

---

[6] A less visible, but tangible effect from maturity level 3 on will be to enhance a basic organizational culture of RCA [33][34], introducing and applying it in a gradual manner to each performed process, learning to distinguish, increasingly and at all levels, between defects and causes and their economical impact, and allowing organizations to write their *strategic (process) maps* more and more effectively, as, for instance, in a Balanced Scorecard (BSC).

[7] Each process area can refer to related processes at the end of a certain GP. For instance, GP2.8 refers to the Project Monitoring & Control (PMC) and Measurement & Analysis (MA) processes.

In this paper, we proposed a quantitative approach to RCA and Ishikawa (Fishbone) diagrams, overcoming some limitations noted in the ODC technique, but generalizing some lessons learned from that experience. This approach, going back to TQM studies, would help an organization in its measurement ability, as well as in the rating of other processes at maturity level 2, such as PMC and PPQA. Our suggestion to software organizations is, therefore, to make the teaching of TQM tools a priority in their training programs, not only with reference to CAR, but also as cross-knowledge that would have a positive impact on CMMI (or any other SPI model) processes.

Future work will include the identification of defect *types* and *triggers* for single CMMI PAs, possibly from case studies available in the technical literature, to leverage the advantages of the ODC approach and our suggestion for a quantitative approach to RCA.

References

1. Al-Karaghouli W., AlShawi S. & Elstob M., Enhancing the Understanding of Customer Requirements in Business Software Engineering, Conference on Management Science and Operational Research (ABAS 1999), Barcelona (Spain), URL: http://www.sba.muohio.edu/abas/1999/al-karwa.pdf
2. Buglione, L. & Abran A., Creativity and SPI: an exploratory paper on their measurement, in "Current Trends in Software Measurement", Proceedings of the IWSM 2001 (11th International Workshop on Software Measurement), Montréal, Québec (Canada), August 28-29, 2001, Shaker Verlag, ISBN 3-8265-9681-1, pp. 112-126
3. Chillarege, R., Bhandari, I. S., Chaar, J. K., Halliday, M. J., Moebus, D. S., Ray, B. K. & Wong, M., Orthogonal Defect Classification: Concept for In-Process Measurement, IEEE Transaction on Software Engineering, Vol. 18 No. 11, November 1992, URL: http://www.chillarege.com/odc/articles/odcconcept/odc.html
4. Chillarege, R., ODC – Orthogonal Defect Classification, Presentation, January 19 2000, Center for Software Engineering, IBM Thomas J. Watson Research Center, pp. 92
5. Constant, D., Re: CMMI Representations, which one is the better? Yahoo SPI Mailing List, February 10 2004, URL: http://groups.yahoo.com/group/cmmi_process_improvement
6. Ishikawa, K, Guide to Quality Control (Industrial Engineering & Technology), Asian Productivity Organization; 2/e, 1986, ISBN 9283310357
7. Deming, W. E., Out of the Crisis, MIT Press, 1986, ISBN 0-911379-01-0
8. Juran, J. & Gryna Jr., F. M., Quality Planning and Analysis, McGraw-Hill, 1980, ISBN 0-07-033178-2
9. Crosby, P. B., Quality is free, McGraw-Hill, 1979, ISBN 0-451-62585-4
10. Norausky, G. F., Causal Analysis & Resolution: A Business Driver at all Levels, 2nd Annual Technology and User Group CMMI Conference, NDIA, November 14 2002, URL: http://www.dtic.mil/ndia/2002cmmi/norausky3a1.pdf
11. CMMI Product Team, CMMI for Development, Version 1.2, CMMI-DEV v1.2, Continuous Representation, CMU/SEI-2006-TR-008, Technical Report, Software Engineering Institute, August 2006, URL: http://www.sei.cmu.edu/pub/documents/06.reports/pdf/06tr008.pdf
12. Kan, S. H., Metrics and Models in Software Quality Engineering, 2/e, Addison-Wesley, 2002, ISBN 0201729156

13. Software Engineering Institute, Process Maturity Profile – SCAMPI v1.1 Appraisals Results 2006 Year End Update, Software Engineering Measurement and Analysis Team, Carnegie Mellon University, March 2006, URL: http://www.sei.cmu.edu/appraisal-program/profile/pdf/CMMI/2006marCMMI.pdf
14. Williams, R., Causal Analysis & Resolution (CAR) at Level 1, 3rd Annual Technology and User Group CMMI Conference, NDIA, November 19, 2003, URL: http://www.dtic.mil/ndia/2003CMMI/Slo.ppt
15. Leszak, M., Perry, D. E. & Stoll, D., A Case Study in Root Cause Defect Analysis, Proceedings of the 22nd International Conference on Software Engineering (ICSE 2000), Limerick (Ireland), June 4-11 2000, ISBN 1581132069, pp. 428-437
16. Agena Ltd, Bayesian Belief Nets, URL: http://www.agena.co.uk/bbn_article/bbns.html
17. ISO/IEC, IS 90003:2004 -- Software engineering -- Guidelines for the application of ISO 9001:2000 to computer software, International Organization for Standardization, Geneva, February 2004
18. SEI, Standard CMMI Appraisal Method for Process Improvement (SCAMPI), version 1.1: Method Definition Document, Software Engineering Institute, Handbook, CMU/SEI-2001-HB-001, December 2001, URL: http://www.sei.cmu.edu/pub/documents/01.reports/pdf/01hb001.pdf
19. Tian, J., Quality-Evaluation Models and Measurements, IEEE Software, Vol. 21 No. 3, May/June 2004, pp. 84-91
20. El Emam, K. & Wieczorek, I., The Repeatability of Code Defect Classifications, ISERN Technical Report, ISERN-98-09, Fraunhofer Institute for Experimental Software Engineering, 1998
21. Bate, R., Kuhn, D. A., Wells, C., Armitage, J., Clark, G., Cusick, K., Garcia, S., Hanna, M., Jones, R., Malpass, P., Minnich, I., Pierson, H., Powell, T. & Reichner, A., Systems Engineering Capability Maturity Model (SE-CMM) Version 1.1, Software Engineering Institute, CMU/SEI-1995-MM-003, Maturity Model, November 1995, URL: http://www.sei.cmu.edu/pub/documents/95.reports/pdf/mm003.95.pdf
22. ISO/IEC JTC1/SC7/WG10, TR 15504 – Parts 1-9, Software Process Assessment, v.3.03, 1998
23. Paulk, M. C., Weber, C. V., Garcia, S. M., Chrissis, M. B. & Bush, M., Key Practices of the Capability Maturity Model Version 1.1, Software Engineering Institute/Carnegie Mellon University, CMU/SEI-93-TR-025, February 1993, URL: http://www.sei.cmu.edu/pub/documents/93.reports/pdf/tr25.93.pdf
24. Paulk, M. C., A Comparison of ISO 9001 and the Capability Maturity Model for Software, Software Engineering Institute, CMU/SEI-94-TR-012, July 1994, URL: http://www.sei.cmu.edu/pub/documents/94.reports/pdf/tr12.94.pdf
25. Butcher, M., Munro, H. & Kratschmer, T., Improving software testing via ODC: Three case studies, IBM Systems Journal, Vol. 41, No. 1, 2002, pp. 31-44, URL: http://researchweb.watson.ibm.com/journal/sj/411/butcher.html
26. Dalal, S., Hamada, M., Matthews, P. & Patton G., Using Defect Patterns to Uncover Opportunities for Improvement, Applications of Software Measurement (ASM) Conference, San Jose, CA, February 1999; URL: http://www.argreenhouse.com/papers/paul4/edaP4.pdf
27. Hsiao D., Actionable Root Cause Analysis: Measure for Phase Containment, SEPG2004, Software Engineering Process Group Conference, March 8-11 2003, Orlando, FL (USA)
28. ISO, IS 9001:2000, Quality Management Systems. Requirements, International Organization for Standardization, December 2000
29. Ebert, C., Dumke, R., Bundschuh, M. & Schmietendorf, A., Best Practices in Software Measurement. How to use metrics to improve project and process performance, Springer, 1/e, 2005, ISBN 3-540-20867-4

30. Fredericks, M. & Basili, V., Using Defect Tracking and Analysis to Improve Software Quality, Technical Report, DACS-SOAR-98-2, DoD Data & Analysis Center for Software, 1998, URL: http://www.dacs.dtic.mil/techs/defect/defect.pdf
31. Huber, J. T., A Comparison of IBM's Orthogonal Defect Classification to Hewlett Packard's Defect Origins, Types and Modes, SM /ASM 2000 Conference, March 6-10, 2000, San Jose, CA (USA), URL: http://www.stickyminds.com/stickyfile.asp?i=1764291&j=52901&ext=.pdf
32. Rus, I., Combining Process Simulation and Orthogonal Defect Classification for Improving Software Dependability, ISSRE 2002, Fast Abstract, URL: http://www.chillarege.com/fastabstracts/issre2002/27.pdf
33. Buttles-Valdez, P., People Capability Maturity Model and Anthropology, Software Engineering Institute, 2006, URL: http://www.sei.cmu.edu/sepg/2007/files/Buttles%20Valdez.pdf
34. Howley, P. P., Teaching How to Calibrate a Process Using Experimental Design and Analysis: The Ballistat, *Journal of Statistics Education* Volume 11, Number 2 (2003), URL: www.amstat.org/publications/jse/v11n2/howley.html,

Appendix A: List of Acronyms

BBN	Bayesian Belief Net
BPM	Business Process Model
BSC	Balanced Scorecard
CAR	Causal Analysis & Resolution
CL	Capability Level
CMMI	Capability Maturity Model Integration
DAR	Decision Analysis & Resolution
GP	Generic Practice
ISO	International Organization for Standardization (www.iso.ch)
KPA	Key Process Area
ML	Maturity Level
ODC	Orthogonal Defect Classification
PA	• Process Area (in the CMMI model) • Process Attribute (in the SPICE model)
PI	Process Improvement
PMC	Project Monitoring and Control
PPQA	Process & Product Quality Assurance
RCA	Root-Cause Analysis
SCAMPI	Standard CMMI Appraisal Method for Process Improvement
SP	Specific Practice
SPI	Software Process Improvement
Sw-CMM	Software Capability Maturity Model
TQM	Total Quality Management