Inception of Software Validation and Verification Practices within CMMI Level 2

Paula Monteiro
Universidade do Minho
pmonteiro@dsi.uminho.pt
paula.monteiro@i2s.pt

Ricardo J. Machado
Universidade do Minho
rmac@dsi.uminho.pt

Rick Kazman
Software Engineering Institute
kazman@sei.cmu.edu

Abstract - Validation and verification are mandatory activities that software companies must perform when developing software products with a high degree of quality. Currently, more companies become aware that adopting CMMI (the software process maturity model developed by the Software Engineering Institute) can be a way to develop quality software. However, some companies are resistant to adopt CMMI maturity level 2 because they do not considerer this maturity level a benefit since its implementation is expensive and does not cover the validation and verification efforts. The simultaneous adoption of CMMI maturity level 2 with validation and verification process areas (from maturity level 3) lacks some methodological recommendations, since some dependencies exist between those two CMMI maturity levels. This PhD thesis will propose one approach to conciliate validation and verification practices with of CMMI maturity level 2 and by adopting ISO/IEC 29119 standard to fulfill a product lifecycle perspective.

Keywords: Software Testing; Software Quality.

I. INTRODUCTION

CMMI (Capability Maturity Model Integration) [1, 2] is a software development standard from the Software Engineering Institute (SEI). CMMI is composed by a set of software development process guidelines and is used to improve the quality of the software and its delivery. Using CMMI, SEI addresses practices that companies can use as a guideline for process improvement. It can be seen as a collection of best practices that could be followed to improve the quality of products.

However, there are organizations that do not adopt CMMI [3] (in particular the Maturity Level 2 – ML2) and the main reasons they give is that: the company is small, the cost to implement CMMI is high, they use another Software Process Improvement (SPI) and they not see a clear benefit in using it.

Several companies do not want do adopt exclusively CMMI Maturity Level 2 because they are mainly interested in the Engineering processes, which is not the focus of this Maturity Level (ML). Since ML2 Process Areas (PAs) are mainly focused in the project management and support processes, most companies tend to consider that the implementation of ML2 does not bring significant benefits to compensate the corresponding cost and time overheads. Companies recognize benefits only in the implementation of PAs of Maturity Level 3 (ML3). They do not understand that, in order to achieve ML3, they have to achieve ML2 before. We refer to this issue as “Level 2 Syndrome”.

The software testing (or Validation and Verification - V&V) phase is one of the most important in the software development life cycle, consuming around 50% of the total duration [4, 5] and 50% of the total cost [6] of many projects. Additionally, companies are becoming more and more aware of the important role that V&V plays in the production of high quality software [7]. Although other process areas of software engineering are also beginning to catch the attention of the software industry, their increased interest in V&V is our main motivation to dedicate our study to the adoption of the validation and verification PAs simultaneously with the implementation of CMMI ML2 (that do not consider V&V activities) as a way to overcome the Level 2 Syndrome.

The main goal underlying this thesis is not to propose a new version of the CMMI by moving the V&V PAs to the ML2, but to achieve simultaneous implementation of the CMMI ML2 and CMMI ML3 V&V PAs.

V&V are very important for companies that develop large scale software products due to the size of the solutions. So, those companies have to demonstrate that the product or product components accomplish its intended use when placed in its intended environment, as well as ensuring that selected work products meet their specified requirements.

Companies that develop software at a large scale become aware that implementing CMMI can be a good choice for developing and delivering software with a high degree of quality. However, a company that is applying for CMMI ML2 assessment must take into consideration that the validation and verification efforts are not considered for that level. This level of maturity is only concerned if an organization ensures that in their projects the requirements are managed and their processes are planned, performed, measured, and controlled. Therefore, it is important for those companies to have the possibility of simultaneously implementing CMMI ML2 and the V&V PAs.

A way to help those companies accomplishing the V&V PAs at an earlier stage of the process improvement is the creation of a roadmap with guidelines to perform those efforts. Using this roadmap, companies that are changing to a CMMI ML2 and have in its concerns the V&V efforts could use it as a guide to achieve those PAs implementation.

The main motivation for this study was finding a solution to the problem faced by the companies that want implement V&V PAs simultaneously with CMMI ML2. However the scope of this work might be enlarged to other PAs from a ML higher than ML2. This arises from the fact that it is known that the CMMI PAs has some dependencies between

978-0-7695-3777-1/09 $26.00 © 2009 IEEE
DOI 10.1109/ICSEA.2009.84

them which raise the possibility that the anticipation of the V&V implementation could also mean the implementation of other CMMI ML3 PAs.

Currently, there are very few attempts to address this anticipation of V&V PAs within CMMI ML2. We can find several efforts that are concerned with the V&V or with the V&V in some of the existent maturity models. However, V&V efforts within CMMI ML2 contexts have not yet been documented.

The definition of this roadmap to help the simultaneous execution of inclusion of V&V PAs with the CMMI ML2 demands for the verification if the impact of the adoption of those PAs in an earlier stage will become an advantage to the company or not.

In this paper, a description of the state-of-the-art related with the subject of this research is presented in Section 2. Section 3 describes in detail the research objectives and the research methodology. In Section 4, the current work and preliminary results already done in the context of this research are briefly described. Section 5 presents the work plan for the next 2 years of research. Finally, in Section 6 some conclusions are presented.

II. State of the Art

The state of art of this work essentially relates to: CMMI, Validation and Verification area and ISO/IEC 29119 standard. Some trends in V&V are also discussed.

A. CMMI

CMMI [1, 2] is a well known Software Process Improvement (SPI) maturity model developed by the Software Engineering Institute (SEI). It is concerned in helping organizations to improve their processes. This SPI has been implemented by several organizations [8, 9] that report a great improvement in cost reduction, productivity, and performance gains.

CMMI was designed to integrate all the models created by SEI and other organizations through the years. It has two representations: the staged and the continuous representation.

CMMI staged representation is divided into Maturity Levels (ML). An ML is a set of practices for a predefined set of PAs that will improve the performance of a company. An ML shows the level of performance that a company has in a particular discipline or set of disciplines. In each ML, a company improves a set of processes and gets prepared to evolve to the next level. The degree of maturity of a company in each level is measured by assessing the accomplishment of the goals of each predefined set of PAs.

The continuous representation uses Capability Levels (CL) to characterize the improvement of the company in relation of a given PA. In the continuous representation the company can choose the PA or set of PAs that will be improved in the organization, as well the order of the improvements to meet the organization objectives. With the continuous representation the company can improve the PAs at different levels, which can be seen as an advantage since it offers maximum flexibility in the improvement. But at the same time there is a limitation caused by the dependencies between the PAs [1, 2].

In CMMI, the V&V practices are PAs of the CMMI ML3. The term “system testing” or “software test” is not used in CMMI. This comes from the fact that these terms can be interpreted in several ways. Instead of using the term “test”, CMMI uses the terms “verification and validation” because “test” can be part of the verification and validation and is one of the methods used to perform verification and validation efforts [10].

There is an urgent need to implement in CMMI ML2 the PAs of V&V that arise with the companies’ need of having software test covering all the lifecycle of the product. A large scale software development, companies need V&V efforts covering the whole product lifecycle which is not considered within CMMI ML2.

B. Validation and Verification

In 1989, V&V activities where already under study and a work was published introducing a V&V method [11]. This method was supported by a group of standards related with the V&V. At this time, there was already the motivation to apply the V&V through the product lifecycle to improve the quality of the products.

In [12], we can find the definitions of validation and of verification. Validation is “the process of evaluating a system or component during or at the end of the development process to determine whether it satisfies specified requirements” and verification is “the process of evaluating a system or component to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase”. However, validation and verification have also a common definition: “the process of determining whether the requirements for a system or component are complete and correct, the products of each development phase fulfill the requirements or conditions imposed by the previous phase, and the final system or component complies with specified requirements”.

The difference between the validation and verification can be explained by looking at the purpose of the tests performed. The use of prototypes to test if requirements can be addressed is an example of a verification practice, but if the prototype is evaluated by the users to test if the product fulfils their needs we are on the presence of an example of a validation practice. In other words, we can say that the verification should ask “Are you meeting the specified requirements?” and “Are you building the product right?”. In the same way we can say that the validation should ask if “Are you meeting the operational need?”, “Does this product meet its intended use in the intended environment?” and if “Are you building the right product?”. The main goal that leads to this division of the software test within CMMI in two practices was that separating it in two different processes was a way to emphasize both practices [10].

C. Trends in Validation and Verification

Despite some works that describe V&V efforts in several areas, the issue of V&V PAs within CMMI ML2 are rarely addressed. We can find some efforts that are related with CMMI [13-17] or V&V in general [11, 18-22], with V&V in CMMI not specifically within ML2 [7, 23], or with V&V

within others maturity models [24]. However, until now, there are very few attempts to address the simultaneous execution of V&V PAs within CMMI ML2.

There are some authors [7] that consider that V&V assessment in CMMI is not enough when we are dealing with "safety-critical" software. So, they propose a new framework for V&V assessment, focused on the safety-criticality. This framework is defined using some standards related with safety, together with V&V PAs of CMMI and ISO 9001 standard [25].

Other authors use CMMI to evaluate a generic software testing model called CenPRA test process [23]. This test process defines a set of ordered activities and test artifacts. The evaluation done in this work wants to check what are the aspects of CMMI that are taken in account by CenPRA test process. The authors want also to evaluate how this test process can be used as a supplement of the software testing of CMMI. But even though this work presents a test process that can be used by CMMI, the impact of using this test process in CMMI ML 2 was not analyzed.

D. ISO/IEC 29119

In May of 2007, a working group was formed by ISO/IEC to produce a new software testing standard (the ISO/IEC 29119 [26-29]) which could be applicable to all types of software products and software-intensive systems. The purpose of this working group is to propose an international standard that will cover software testing in the development and maintenance of a product or system. The motivation that gives origin to the creation of this group was the inexistence, until now, of standards that cover all aspects of the software test lifecycle.

The existing standards do not cover risk-based testing, static testing, use case testing, non-functional testing, etc. Another motivation issue is the existence of conflicts in the definitions and the difficulty by the practitioners to choose what standard to use to define their software testing. Since it is a main goal to create a standard that covers all aspects of the software test lifecycle, ISO/IEC 29119 will be consistent with the ISO/IEC 12207 [30-32].

ISO/IEC 29119 standard could be a used to study the adaptation/extension of V&V PAs of CMMI to be adopted in CMMI ML 2. The reason to consider ISO/IEC 29119 (even though it is under development) is based on the coverage of the entire lifecycle of the software test, which is crucial to adequately support the usage of V&V PAs within CMMI ML 2.

III. Research Objectives and Approach

This section starts by presenting the research objectives of the thesis. Then some research methodologies are presented, in particular the research methodology chosen to be used in the present work.

A. Research objectives

Companies that develop large scale software systems must comply with V&V practices. But when these companies apply for CMMI ML2 assessment, V&V efforts are not even considered. Therefore, in these contexts it is important to implement CMMI ML 2 and V&V PAs at the same time. CMMI conceives the possibility of adopting the continuous representation. However, dependencies between PAs to support the simultaneous implementation of a particular ML with some PAs in the continuous representation are not document.

This thesis has four main objectives. The first objective is to analyze the impact incorporating V&V activities in CMMI ML2. The anticipation of V&V PAs from ML3 to ML2 must be analyzed. The impact to implement those practices in an earlier level of CMMI must be carefully evaluated to verify whether or not this anticipation will bring advantages. The continuous representation can be, *per si*, considered a solution for anticipating some PAs of CMMI ML3. However, we do not consider this a solution because with this representation it may appear to the companies that they can only appraise V&V PAs in CMMI ML2 and skip the others PAs from ML3. As said before, when implement V&V PAs simultaneously with CMMI ML2, we have to take into account the other PAs that are directly related with V&V. So, we need to study the dependencies between PAs that appear when a company uses the continuous representation to anticipate some PAs in the context of using CMMI in the staged representation. During this analysis, if we detect problems in that anticipation, we are going to review our definition for this extension in order to mitigate its impact.

The second objective is to review V&V PAs of CMMI based on the ISO/IEC 29119 standard. V&V PAs in the current version of CMMI cannot be considered as a lifecycle process. ISO/IEC 29119, as said before, is a standard that covers all the aspects in the software testing lifecycle. To extend the V&V PAs to CMMI ML2 we will take into account ISO/IEC 29119. While this standard is under development, we want to use some of the results of our work as a contribution to its final deliverable.

The third objective is to define a roadmap to guide the implementation of V&V PAs simultaneously with CMMI ML2. Currently, there are no guidelines to help companies to adopt that approach. It is our goal to develop a roadmap to help companies that are applying for CMMI ML2 appraisals and want to formally adopt V&V practices in their software development. This roadmap could be developed taking into consideration existing software engineering processes like, for instance, the Rational Unified Process [33].

Finally, the last objective is to validate the research results in a real case. This is our final goal. We plan to apply our results to a real case to demonstrate and validate our proposal to solve the above described problem.

B. Research methodology

There are several research methodologies that can be used in the software engineering research. Four of those methodologies were identified in 1989 [34]: the scientific method; the engineering method; the empirical method; and the analytical method. The scientific method [34] is based on the observation of the real world, then a model or theory of behavior is proposed, measured and analyzed, the hypotheses of the model or theory are validated, and if possible repeated.

The engineering method [34] is based on the observation of existing solutions, then better solutions are proposed, build or developed, the solutions are measured and analyzed, and this process is repeated until no further improvements are possible. In the empirical method [34] a model is proposed and this model will be applied to case studies, to measure and analyze, and validate the model, and if it is necessary the process can be repeated. Finally, in the analytical method [34] formal theory or set of axioms are proposed, a theory is developed, the results are derive and if possible compared with empirical observations.

Beside the large number of research methodologies existent the research methodology intended to use in order to validate the results obtained in this research is the Action Research [35-39]. Action research is a type of research that involves researchers and practitioners which act together on a particular set of activities.

Action research has been called by many different names like: participatory research, collaborative inquiry, emancipatory research, action learning, and contextual action research [38] all meaning the same. But action research can be defined in a simple way as "learning by doing". In other words, we can say that this kind of research is an iterative process.

First, the problem is defined and, then, we do something to solve it, verify if our solution is successful and if it is not successful we try again to find another solution [38].

The fact that this research could be done in a cyclic way was an important factor to let us choose this as our research methodology. The capability to provide a solution to our problem, analyze the impact of the solution and having the possibility to change this solution was the major factor for our choice.

IV. Current Work and Preliminary Results

Until now, the main research activities were directed towards the analysis of the dependencies between CMMI PAs.

As said before, some small companies suffer from the Level 2 Syndrome. They are undecided in adopting CMMI ML2 because they do not see any advantage in that, when engineering PAs are not considered for implementation. One solution to lead companies overcome this syndrome is by achieving the ML2 at same time that some PAs from ML3 (in particular the V&V PAs) are implemented.

The purpose of this PhD work is to study a way to overcome the CMMI Level 2 Syndrome by anticipating CMMI V&V PA. In this effort, the dependencies between PAs of CMMI ML2 have been study.

This work is particularly interested in studying the simultaneous implementation of V&V PAs (from ML3) and CMMI ML2. Finding the dependencies of V&V implies finding the PAs that must be implemented or at least taking into account during the V&V PA implementation.

Looking to the official CMMI documentation [1, 2] it is not possible to have a global view of the dependencies between the all the CMMI Process Areas. It is only possible to see what are the dependencies of each PA, independently, by reading the Related Process Areas section of each PA. there are other sources of dependency analysis [40, 41] that can be considered to formalize the CMMI PAs dependencies.

To obtain the complete list and a graph representation of all the dependencies between the several CMMI process areas the Related Process Areas section for all the PAs were analyzed. So it was decided to create a matrix (that contains the information of all the dependencies) and a set of graphs (that graphically represents the information stored in the matrix). In Figure 1 it is presented an example of the dependencies graphs achieved, in this case the dependencies for the CMMI ML2 PAs. This graph shows the dependencies that each CMMI ML2 PA has from all the CMMI PAs. As one example, when looking to the graph it is possible to see that PPQA (Project and Product Quality Assurance) has a dependency from VER (Verification) and PP (Project Planning).

The main idea of this dependency study was to analyze if the anticipation of some PAs (in particular the V&V PAs) will be a benefit or not.

V. Work Plan

The first activity is the state-of-the-art. This activity will complement the brief state-of-the-art presented in this manuscript. With this literature review, we intent to acquire knowledge about the efforts made for similar problems. We have in mind to review the following main areas of study:

- Software process improvement approaches, in particular the CMMI;
- The new standard ISO/IEC 29119, as well as the works that have been meanwhile developed and that are related to this software testing standard;
- V&V approaches, both within and without CMMI.

The second activity that we intend to execute is the analysis if the adoption of V&V activities within CMMI ML2 will cause an improvement or a deterioration of the product lifecycle. This analysis of impact will be performed by comparing the state of the product lifecycle before and after the introduction of V&V PAs According to the results achieved, some considerations will be taken. If the result is not satisfactory, some changes in the incorporation of V&V practices into CMMI ML2 will be performed until the best result is achieved.

The third planned activity V&V efforts in CMMI ML2 will be reviewed taking into account ISO/IEC 29119 standard. Since V&V efforts in CMMI are not lifecycle processes, we are going to propose new V&V processes to be extended to the whole product lifecycle.

A final effort to validate the work will be performed through a real case in the I2S Company [42]. Since this company is applying for CMMI ML 2 assessment, we will initiate the incorporation of the V&V practices in their product lifecycle. By doing this, we want to validate the changes that we are proposing for the V&V efforts within CMMI ML2. During the incorporation of those practices in the product lifecycle, we are going to analyze the impact of those efforts in the company. With this impact analysis we are concerned in estimating the ROI of starting the V&V practices in CMMI ML2.

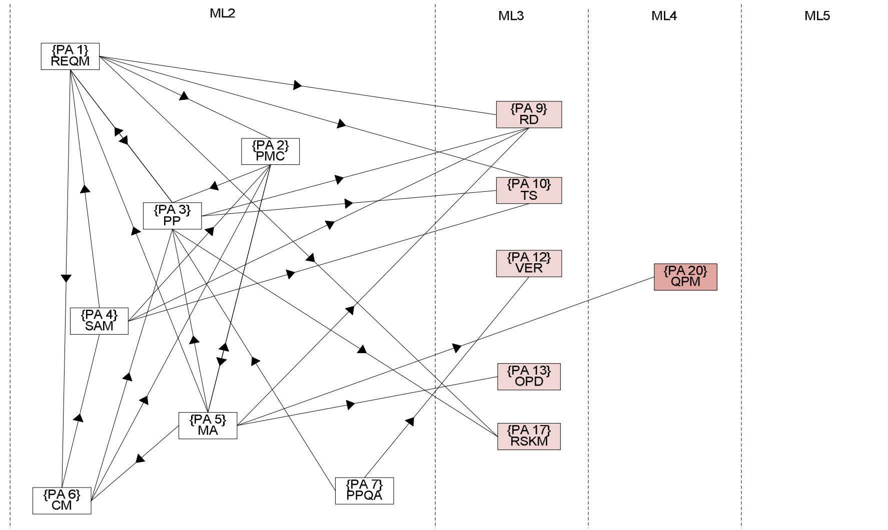

Figure 1. : CMMI ML2 dependencies graph.

Simultaneously with the validation of the research results, we will start the creation of a roadmap to help in the inception of V&V practices within CMMI ML2. While we initiate the implementation of V&V practices in the real case, we will define the roadmap detailing every step executed in this task. This roadmap will be a guideline that could be used by other companies, appraised with CMMI ML2, as a guide to initiate the inclusion of the Validation and Verification practices in their product lifecycle.

The writing of the thesis will be done along the realization of the work.

VI. Conclusion

V&V activities are used to prove if the product fulfils its intended needs and if the product reflects the requirements captured for the product. Companies become aware that CMMI can be helpful to develop and deliver software with a high degree of quality. But at the same time, some found that CMMI only suggests the adoption of V&V efforts when CMMI ML3, or higher, is considered. CMMI ML2 is only concerned if an organization ensures that, in their software development projects, the requirements are managed and that their processes are planned, performed, measured, and controlled. A company that is assessed with this level and needs to perform software test to guarantee the quality of the products cannot be guided in this task by CMMI in the staged representation.

Initiating V&V efforts at CMMI ML2 could be a solution to this situation. Additionally, ISO/IEC 29119 can be used as a strong commitment to assure that the suggested approach to incorporate V&V practices into CMMI ML2 will comply with a full product lifecycle perspective.

The official CMMI documentation [1, 2] lacks of guidelines that must be followed if a company wants to anticipate in CMMI ML2 any PA from the CMMI ML3.

This thesis proposal intends to guide companies to overcome the "Level 2 Syndrome" and this lack of information of the CMMI documentation by studying this problem and propose a roadmap with guidelines that the companies could follow to anticipate PAs from CMMI ML3, in particular the V&V PAs. Although this study is particularly focused in the V&V it could be extended or directed to other PAs from ML3.

References

[1] M. B. Chrissis, M. Konrad and S. Shrum, CMMI(R): Guidelines for Process Integration and Product Improvement (2nd Edition) (The SEI Series in Software Engineering), Addison-Wesley Professional, 2006.

[2] CMMI Product Team, Capability Maturity Model Integration, version 1.2, CMMI for Development, 2006.

[3] M. Staples, M. Niazi, R. Jeffery, A. Abrahams, P. Byatt and R. Murphy, "An exploratory study of why organizations do not adopt CMMI," Journal of Systems and Software, vol. 80, no. 6, 2007, pp. 883-895.

[4] E. Bernard, B. Legeard, X. Luck and F. Peureux, "Generation of test sequences from formal specifications: GSM 11-11 standard case study," Software: Practice and Experience, vol. 34, no. 10, 2004, pp. 915-948.

[5] S. Rayadurgam and M. P. E. Heimdahl, "Test-sequence generation from formal requirement models," Proc. High Assurance Systems Engineering, 2001. Sixth IEEE International Symposium on, 2001, pp. 23-31.

[6] G. Myers, C. Sandler, T. Badgett and T. Thomas, The Art of Software Testing, Second Edition, Wiley, 2004.

[7] A. Y. Kyung, P. Seung-Hun, B. Doo-Hwan, C. Hoon-Seon and J. Jae-Cheon, "A Framework for the V&V Capability Assessment Focused on the Safety-Criticality," Proc. Software Technology and Engineering Practice, 2005. 13th IEEE International Workshop on, 2005, pp. 17-24.

[8] D. L. Gibson, D. R. Goldenson and K. Kost, Performance Results of CMMI®-Based Process Improvement, Technical Report, CMU/SEI-2006-TR-004, ESC-TR-2006-004, Software Engineering Institute, CMU, 2006.

[9] D. R. Goldenson and D. L. Gibson, Demonstrating the Impact and Benefits of CMMI®: An Update and Preliminary Results, Special Report, CMU/SEI-2003-SR-009, Software Engineering Institute, CMU, 2003.

[10] CMMI Frequently Asked Questions; http://www.sei.cmu.edu/cmmi/faq/cov-faq.html#MOD21, 04.06.2009.

[11] D. R. Wallace and R. U. Fujii, "Software verification and validation: an overview," Software, IEEE, vol. 6, no. 3, 1989, pp. 10-17.

[12] IEEE, Standard Glossary of Software Engineering Terminology (IEEE Std 610.12-1990), Software Engineering Technical Committee, IEEE Computer Society, Los Alamitos, CA.

[13] S. Delgado, The Impending Implementation Of CMMI for Test Software, 2007; http://archive.evaluationengineering.com/archive/articles/0907/0907impending_implementation.asp, 04.06.2009.

[14] E. Fredrik and L. Stig, "Experience Report: Using Internal CMMI Appraisals to Institutionalize Software Development Performance Improvement," Proceedings of the 32nd EUROMICRO Conference on Software Engineering and Advanced Applications, IEEE Computer Society, 2006 of Conference, pp. 216-223.

[15] M. Montoni, G. Santos, A. Rocha, S. Figueiredo, R. Cabral, R. Barcellos, A. Barreto, A. Soares, C. Cerdeiral and P. Lupo, "Taba Workstation: Supporting Software Process Deployment Based on CMMI and MR-MPS.BR," Product-Focused Software Process Improvement, 2006, pp. 249-262.

[16] C. Yoo, J. Yoon, B. Lee, C. Lee, J. Lee, S. Hyun and C. Wu, "A unified model for the implementation of both ISO 9001:2000 and CMMI by ISO-certified organizations," Journal of Systems and Software, vol. 79, no. 7, 2006, pp. 954-961.

[17] A. I. F. Ferreira, G. Santos, R. Cerqueira, M. Montoni, A. Barreto, A. O. S. Barreto and A. R. Rocha, "Applying ISO 9001: 2000, MPS.BR and CMMI to Achieve Software Process Maturity: BL Informatica's Pathway," Proc. of the 29th international conference on Software Engineering, IEEE Computer Society, 2007, pp. 642-651.

[18] D. Winkler, B. Riedl and S. Biffl, "Improvement of design specifications with inspection and testing," Proc. Software Engineering and Advanced Applications, 2005. 31st EUROMICRO Conference on, 2005, pp. 222-230.

[19] S. R. Rakitin, Software Verification and Validation: A Practitioner's Guide, Artech House, Inc., 1997, p. 271.

[20] M. S. Fisher and T. Menzies, "Learning IV&V Strategies," Proc. 39th Annual Hawaii International Conference on System Sciences - Volume 09, IEEE Computer Society, 2006.

[21] T. Menzies, M. Benson, K. Costello, C. Moats, M. Northey and J. Richardson, "Learning better IV&V practices," Innovations in Systems and Software Engineering, vol. 4, no. 2, 2008, pp. 169-183.

[22] M. A. Wojcicki and P. Strooper, "An Iterative Empirical Strategy for the Systematic Selection of a Combination of Verification and Validation Technologies," Proceedings of the 5th International Workshop on Software Quality, IEEE Computer Society, 2007 of Conference, pp.

[23] P. Bueno, A. Crespo and M. Jino, "Analysis of an Artifact Oriented Test Process Model and of Testing Aspects of CMMI," Product-Focused Software Process Improvement, 2006, pp. 263-277.

[24] T. Berling and M. Host, "A case study investigating the characteristics of verification and validation activities in the software development process," Proc. Euromicro Conference, 2003. Proceedings. 29th, 2003, pp. 405-408.

[25] "ISO 9001:2000 - Quality management systems - Requirements," 2000.

[26] T. Murnane, ISO/IEC 29119: Software Testing Standard, 2007; http://www.kjross.com.au/content/Document/presentations/ISO%20IEC%2029119%20Software%20Testing%20Standard(1).pdf, 04.06.2009.

[27] "ISO/IEC JTC1/SC7 /N3818, Meeting minutes," 2007.

[28] ISO/IEC JTC1/SC7 /N3820, Specification of the document set for the Testing Standard, 2007.

[29] ISO/IEC 29119 Software Testing; http://www.softwaretestingstandard.org/, 04.06.2009.

[30] R. Singh, "The Software Life Cycle Processes standard," Computer, vol. 28, no. 11, 1995, pp. 89-90.

[31] "ISO/IEC 12207: 1995, Information Technology - Software life cycle processes," 1995.

[32] "ISO/IEC WD, Guide for ISO/IEC 12207 (Software Life Cycle Process)," 1995.

[33] IBM, Rational Unified Process: Best practices for software development teams, 2003; http://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf, 04.06.2009.

[34] W. F. Tichy, N. Habermann and L. Prechelt, "Summary of the Dagstuhl workshop on future directions in software engineering: February 17-21, 1992,Schloß; Dagstuhl," SIGSOFT Softw. Eng. Notes, vol. 18, no. 1, 1993, pp. 35-48.

[35] M. R. d. Villiers, "Three approaches as pillars for interpretive information systems research: development research, action research and grounded theory," Proceedings of the 2005 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries, South African Institute for Computer Scientists and Information Technologists, 2005 of Conference, pp. 142-151.

[36] R. L. Baskerville, "Investigating information systems with action research," Commun. AIS, vol. 2, no. 3es, 1999, pp. 4.

[37] D. E. Avison, F. Lau, M. D. Myers and P. A. Nielsen, "Action research," Commun. ACM, vol. 42, no. 1, 1999, pp. 94-97.

[38] R. O'Brien, An Overview of the Methodological Approach of Action Research, 1998; http://www.web.ca/~robrien/papers/arfinal.html, 04.06.2009.

[39] F. Siv, "Action research on systems development: case study of changing actor roles," SIGCAS Comput. Soc., vol. 18, no. 1, 1988, pp. 22-31.

[40] J. A. C.-M. Villalón, G. C. Agustín, J. Mejía, T. S. F. Gilabert and A. Sánchez, "CMMI-ACQ: A Formal Implementation Sequences of the Processes Areas at Maturity Level 2," Proc. Electronics, Robotics and Automotive Mechanics Conference, 2008. CERMA '08, 2008, pp. 212-217.

[41] X. Chen, M. Staples and P. Bannerman, "Analysis of Dependencies between Specific Practices in CMMI Maturity Level 2," Software Process Improvement, 2008, pp. 94-105.

[42] I2S Company webpage; http://www.i2s.pt, 11.06.2009.