# Towards a Framework to Evaluate and Improve the Quality of Implementation of CMMI® Practices*

Isabel Lopes Margarido[1,**], João Pascoal Faria[1],
Raul Moreira Vidal[1], and Marco Vieira[2]

[1] Faculty of Engineering, University of Porto, Portugal
{isabel.margarido,jpf,rmvidal}@fe.up.pt
[2] Faculty of Sciences and Technology, University of Coimbra, Portugal
mvieira@dei.uc.pt

**Abstract.** CMMI practices can be poorly implemented leading to weak performance gain. SCAMPI verifies model compliance but not performance. Hence, a framework to evaluate the quality of implementation of each practice, based on compliance and performance results, will prevent poor implementation, locate and fix problems, and ultimately achieve better results. In this paper we propose such a framework, based on a combination of leading and lagging indicators measuring compliance, efficiency and efficacy.

**Keywords:** Capability Maturity Model Integration, Measurement, Performance Indicators, Quality of Implementation, Software Process Improvement.

## 1 Introdution

Capability Maturity Model Integration® (CMMI®) is a process improvement model of products and services, composed of 5 maturity levels (ML) achieved via implementation of the specific and generic goals of that ML and all the preceding ones. To satisfy a goal the generic and specific practices or acceptable alternatives to them need to be fulfilled [1]. Organisations that implement CMMI typically improve their performance in terms of predictability, productivity and product quality. Consequently, processes become more predictable and customer satisfaction increases [2]. However, not all organisations have the same performance results; this depends not only on the business context, projects and team but also on the methodologies used in implementation of the model. In a study presented in [3], organisations using the Team Software Process℠ (TSP℠) achieved better product quality performance than the average of organisations appraised as CMM® (Capability Maturity Model®) level 5. There is more variance in performance results when using CMMI, as it is a generic model telling what to do but not how to do it. When using a prescriptive process like TSP, results are more predictable.

* Work partially funded by Fundação para a Ciência e a Tecnologia (FCT): Programa Operacional Potencial Humano (POPH) of QREN, and Fundo Social Europeu (FSE).
** Corresponding author.

O. Dieste, A. Jedlitschka, and N. Juristo (Eds.): PROFES 2012, LNCS 7343, pp. 361–365, 2012.

CMMI Version 1.3 emphasises improvements in organisations performance [1], i.e., it clarifies that organisations need to focus processes on business goals and implement performance improvements to achieve goals that are continuously evolving. The Standard CMMI Appraisal Method for Process Improvement[SM] (SCAMPI[SM]) appraises compliance of organisation processes, activities and outcomes with CMMI, however evaluating performance lies outwith its scope.

CMMI compliance is not a guarantee of good performance *per se*, i.e., there is high variance in performance results within a maturity level [4, 5]. There are several causes for this problem, in particular: **1)** Practices are not used organisation wide [7]; **2)** Poor, or highly varied, implementation of practices leading to multiple solutions results in a lack of clear impact on performance or project improvement [5]; **3)** Baselines quickly erode after achieving a certain maturity level [7]; **4)** Measurement problems, such as metrics uncorrelated and meaningless to upper management, being useless [9]; measures that are unrelated to customer and business objectives [6]; process performance baselines that are not applicable to all projects [7]. In conclusion, as Peterson stated, the big issue is CMMI implementation [5]. To help prevent these problems we propose a framework that provides a catalogue of performance metrics, mapped with CMMI practices and potential organisation goals, used to monitor CMMI implementation across the organisation and through time, to evaluate quality of implementation of CMMI practices and measure effects of process improvements. The framework is inspired by TSP, which is focused on performance results and defines quantitative criteria for process and product quality [8].

## 2 Framework Proposal

We present the proposed framework in the upper left corner of Fig.1, including its components and how organisations can apply it in practice. The framework is composed of a **metamodel**, shaping a **repository** of performance indicators, to evaluate the quality of implementation of CMMI practices, possibly dependent on the methods used to implement those practices. The performance indicators will be tailorable, defined as mandatory or optional, and will be mapped with profiles according to maturity level and methods of the organisation. Additionally, the framework includes **procedures** for setup (tailoring), use in practice and supporting choice of indicators. The framework is developed in two stages: the first is presented in this paper, defining structure, concepts and metamodel; the second is building a repository, calibrated with historical data, which will be object of our future research work.

In general, we propose to characterise the **quality of implementation** of a CMMI practice by a combination of **efficiency** and **efficacy** of implementation, on one hand, and **compliance** of implementation on the other (i.e., alignment with CMMI recommendations or with what is prescribed by the concrete implementation method used), all measured by appropriate **performance indicators** (PI), possibly dependent on the practice and implementation method used. By considering these three quality characteristics, we are looking both at **how** the work is done and **what** its performance results are. For instance, assume we want to evaluate the quality of implementation of specific practice “SP2.2 Conduct Peer Reviews” of the Verification process area. Assume that

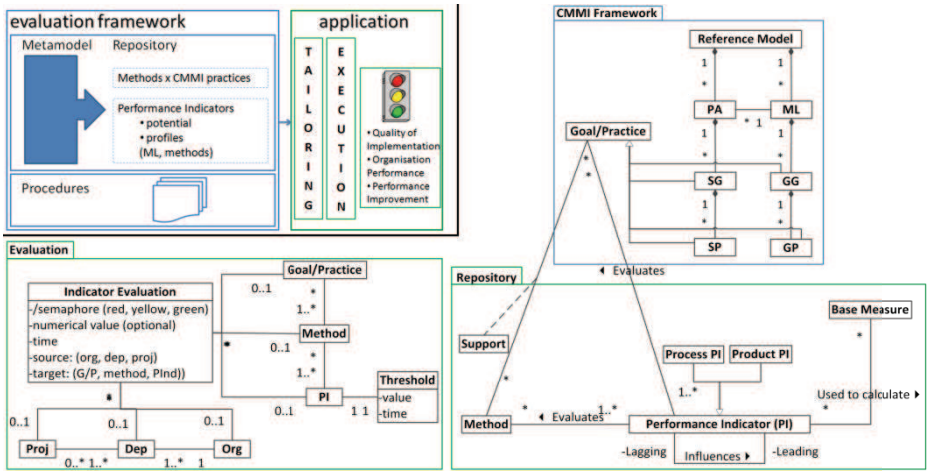


**Fig. 1.** Framework structure and metamodel: org-organisation, dep-department, proj-project [7]

reviewing follows two TSP guidelines: use checklists derived from historical data, and review at a moderate pace. Here, one can measure efficacy by *review yield* (percentage of defects detected), efficiency by *defect detection rate* (defects detected per hour), and compliance by *checklist usage* (a qualitative PI with values, *not used*, *had-hoc checklist*, and *checklist derived from historical data*) and *review rate* (size reviewed per hour), compared with some recommended values.

A rich set of PI usually combines **process** and **product** indicators, and **leading** and **lagging** indicators. In the given example, *review yield* is a **lagging performance indicator**, as the remaining defects can only be known *a posteriori*. Compliance indicators are often **leading indicators**; they influence and can be used to predict and control the values of lagging indicators. In the example, *review rate* is commonly considered a leading indicator of the *review yield* in TSP literature. The *density of defects* found in a review is a **product performance indicator**, whilst the *review rate* is clearly a **process performance indicator**. To evaluate the quality of implementation of methods and CMMI practices, organisation data is analysed through **indicators evaluation**. For that, PI are collected (normally in projects), at a given **time**, and their **numerical value** is compared with a **threshold**. Thresholds have different levels, used to determine the PI **semaphore** colour (red, yellow, green), established according with the organisation quantitative business goals and processes baselines, and define its normal behaviour regarding a PI.

There are three dimensions of aggregation of evaluation results: time, target and source. Aggregation in **time** is done by analysing organisation data in a selected period, given the methods and thresholds at that moment. The **target** of the evaluation can be a PI, a method or a CMMI practice. For each method it may be necessary to monitor one or more PI. A method is evaluated through a semaphore, whose colour is given by the analysis of the PI semaphores. For that reason **numerical value** is optional and **semaphore** is mandatory. Since we map methods with CMMI practices, the aggregation of the evaluation of each method used (mandatory, alternative or

optional) gives the semaphore colour of the practices. The **source** of the evaluation can be: a **project**, evaluated by aggregating PI evaluation; a **department**, evaluated by aggregating its projects' evaluation; or the entire **organisation**, evaluated by aggregating its departments' evaluation. Aggregation at organisation level indicates the degree of institutionalisation of the practices necessary to achieve generic goals and high maturity, and consequently allow their evaluation. A project, department or organisation can also use target aggregation to evaluate a method or a CMMI goal/practice. The evaluation by aggregation of colours is done as follows: green – all green; yellow – at least one yellow and no reds; red – at least one red. We are aware that results aggregation can be more complex.

To find the adequate PI to populate the framework repository, we will undertake bibliographical research and analyse industry data. The value of a PI (e.g. *effort estimation error*) is influenced by two parcels: one is related to process (e.g. *estimate effort*) definition and execution, comprised of **controllable factors** (e.g. *size, historical data*); the other is comprised of **non-controllable factors**, related to project execution and other environment, complexity and context variables (e.g. *change requests*, *complexity*). Controllable factors (leading indicators) can be used to improve PI in advance. By experiment we will analyse organisations data to determine the percentage of each one of these parcels, to know the percentage of the PI value which may be influenced in advance. We will also analyse effects of individual controllable factors, determine recommended values for each one of them and consequently guarantee that implementation of CMMI practices leads to better performance. This step of the research is under development. For calibration we will use different organisations projects data and, when completed, the framework will be tested in an organisation.

## 3 Related Work

There are several frameworks to evaluate success factors in metrics programs [9] and in Software Process Improvements (SPI) [10]. The analysed success factors are related to the way SPI is done, and not to improving processes outputs. There are object-oriented models [11] and metamodels that can be used to develop measurement repositories [12, 13], which can also be aligned with CMMI [11, 14-16] and shape processes [15, 16]. Similar metamodels can be useful to unambiguously define PI. [13] describes a framework to measure processes based on their structure and relations. In our research, when measuring a practice we are focused not only on compliance but also on its efficacy and efficiency. In addition, there are tools to collect and align SCAMPI evidences. [17] proposed a method introducing quality metrics to evaluate SCAMPI, but does not evaluate how CMMI practices are implemented or organisation performance. [18] designed an evidence repository to assess projects activities by number of executions. Nonetheless, it is possible that an evidence is generated but empty, showing that the activity was not performed.

In the case of our framework, the primary evaluation criterion is not the way process improvement implementation is done, but the value, i.e. the outcome for the organisation, of the goal/practice itself. For that we need to understand what the advantage of using it is and whether the organisation benefits from it or not.

## 4 Conclusion

The proposed framework shall support organisations to: 1) implement CMMI by providing a pool of methods, aligned with practices, and performance indicators to monitor them; 2) choose methods for their adequacy and performance in context; 3) evaluate quality of implementation of CMMI practices early; 4) monitor process performance to act before problems occur; 5) anticipate impact of process changes on performance indicators; 6) understand, more accurately, causes of problems; 7) prioritise performance improvements. SEI will be able to assess performance improvements from one appraisal to the next. Aggregation is particularly relevant to evaluate Generic Goals and High Maturity Levels and performance indicators are useful to evaluate quality of implementation.

## References


1. Chrissis, M.B., Konrad, M., Shrum, S.: CMMI for Development: Guidelines for Process Integration and Product Improvement. Addison-Wesley, Massachusetts (2011)
2. Goldenson, D.R., Gibson, D.L., Ferguson, R.W.: Why Make the Switch? Evidence about the Benefits of CMMI. In: SEPG 2004. CMU/SEI (2004)
3. Davis, N., Mullaney, J.: The Team Software Process$^{SM}$ (TSP$^{SM}$) in Practice: A Summary of Recent Results. Technical Report, CMU/SEI, 105 (2003)
4. Radice, R.: Statistical Process Control in Level 4 and Level 5 Software Organizations Worldwide. In: Software Technology Conference. CMU/SEI (2000)
5. CMU/SEI: Accelerated Improvement Method (AIM). Technical Report, CMU/SEI (2010)
6. Leeson, P.: Why the CMMI® does not work. In: SEPG Europe. CMU/SEI, Prague (2009)
7. Lopes Margarido, I., Faria, J.P., Vieira, M., Moreira Vidal, R.: CMMI Practices: Evaluating the Quality of the Implementation. In: SEPG Europe. CMU/SEI, Dublin (2011)
8. Humphrey, W.S.: Introduction to the Team Software Process$^{SM}$. Addison Wesley (2010)
9. Jeffery, R., Berry, M.: A framework for evaluation and prediction of metrics program success. In: Software Metrics Symposium (1993)
10. Niazi, M., Wilson, D., Zowghi, D.: A maturity model for the implementation of software process improvement: an empirical study. J. Syst. Softw. 74(2), 155–172 (2005)
11. Palza, E., Fuhrman, C., Abran, A.: Establishing a generic and multidimensional measurement repository in CMMI context. In: SW Engineering Workshop (2003)
12. Goulão, M.: Component-Based Software Engineering: a Quantitative Approach. Ph.D. Thesis. Departamento de Informática, FCT/UNL (2008)
13. García, F., Ruiz, F., Cruz, J.A., Piattini, M.: Integrated Measurement for the Evaluation and Improvement of Software Processes. In: Oquendo, F., et al. (eds.) EWSPT 2003. LNCS, vol. 2786, pp. 94–111. Springer, Heidelberg (2003)
14. Colombo, A., et al.: The Use of a Meta-Model to Support Multi-Project Process Measurement. In: 15th Asia-Pacific Software Engineering Conference, APSEC (2008)
15. Hsueh, N.-L., et al.: Applying UML and software simulation for process definition, verification, and validation. Inf. Softw. Technol. 50(9-10), 897–911 (2008)
16. Mishra, S., Schlingloff, B.H.: Compliance of CMMI Process Area with Specification Based Development. In: SERA. SERA (2008)
17. Pricope, S., Lichter, H.: Towards a Systematic Metric Based Approach to Evaluate SCAMPI Appraisals. In: Bomarius, F., Oivo, M., Jaring, P., Abrahamsson, P. (eds.) PROFES 2009. LNBIP, vol. 32, pp. 261–274. Springer, Heidelberg (2009)
18. Sunetnanta, T., Nobprapai, N.-O., Gotel, O.: Quantitative CMMI Assessment for Offshoring through the Analysis of Project Management Repositories. In: Gotel, O., Joseph, M., Meyer, B. (eds.) SEAFOOD 2009. LNBIP, vol. 35, pp. 32–44. Springer, Heidelberg (2009)