

# Is ISO/IEC 15504 Applicable to Agile Methods?

Giuseppe Lami<sup>1</sup>, Fabio Falcini<sup>2</sup>

<sup>1</sup>Consiglio Nazionale delle Ricerche, Istituto di Scienza e Tecnologie dell'Informazione  
via Moruzzi, 1 – I-56124 Pisa, Italy  
[giuseppe.lami@isti.cnr.it](mailto:giuseppe.lami@isti.cnr.it)

<sup>2</sup>Intecs SpA  
via E. Giannessi, 5 – I-56121 Pisa, Italy  
[fabio.falcini@intecs.it](mailto:fabio.falcini@intecs.it)

**Abstract.** In the last two decades several models for evaluating software process capability have been defined and became more and more popular. The application of such models, and in particular the ISO/IEC 15504, determined a general software process improvement in many domains. Nevertheless, the application of the ISO/IEC 15504 standard is still considered by many agile developers as incompatible with agile approaches. Such an attitude is mainly based on common misunderstandings on what the ISO/IEC 15504 is and on what its application involves. This paper aims at showing that this standard, if genuinely applied, can be effectively used also in agile contexts.

**Keywords:** Software Process Capability Determination, ISO/IEC 15504..

## 1 Introduction

After over 30 years of software engineering, software development is today supported by disciplined and formal methodologies, paradigms and tools. Models and standards providing guidelines, processes and evaluation methods for software development exist as well. In particular, in the past two decades, several models for improving software development processes became popular (in particular CMMI and SPICE – ISO/IEC 15504). We witnessed, in our experience, some misuses of such models that determined, sometimes, the paradox to be perceived as a bundle to the development activities in software development teams. Agile methods arose as a reaction to the misuse of firm and disciplined approaches to the software development.

In such a situation two parties rose up in the software community: the “agilers”, challenging the disciplined approach seen as a way to spend the effort most on the documentation than real software (the product), and the “orthodox” believing that the major effort required by a disciplined, rigorous and documented approach pays in terms of quality of the final product. The authors of this paper (being representatives of both the two parties of the barricade) discussed, on the basis of their common experience in leading software process improvement and assessment initiatives, how and if the disciplined and rigorous approach and the agile one can co-exist. In this paper, the applicability of the ISO/IEC 15504 (also known as SPICE) [1] for software

process assessment and improvement to agile contexts is discussed. The SPICE model is generally perceived in the software community as a way to evaluate software process from a formal and document-based perspective. Our aim is to debunk such a myth and to show that the SPICE model, if genuinely applied, can be effectively used also in agile organizations.

This paper is structured as follows: in Section 2 we present the ISO/IEC 15504 standard. In section 3 some typical misunderstandings about what ISO/IEC 15504 is and what it requires to be applied are discussed. In Section 4 the applicability of the ISO/IEC 15504 standard to agile contexts is systematically analyzed. Finally, in Section 5 the final discussion is provided as well as the conclusions.

## 2 ISO/IEC 15504: Key Concepts

It is not the aim of this section to provide a detailed description of the ISO/IEC 15504 standard [1]; what we are interested in is to let the reader able to understand the basic concepts underlying the standard. The purpose of the standard is to provide a scheme for evaluating the capability of the software process and a way to improve it. *Process capability* is defined as a characterization of the ability of a process to meet current or projected business goals.

The three basilar concepts of the standard are: Process Reference Model (PRM), Process Assessment Model (PAM) and Measurement Framework.

**PRM:** it is a model comprising definition of processes in a lifecycle described in terms of process purpose and outcomes, together with an architecture describing the relationships between processes. In other words, the PRM is the set of the descriptions of the processes that will be assessed. While the standard doesn't include any specific PRM (it defines the requirements for defining a PRM), in practice the ISO/IEC 12207 [2] standard is very often used as PRM for ISO/IEC 15504.

**PAM:** it is a model suitable for the purpose of assessing process capability, based on one or more PRMs. A PAM provides a two-dimensional view of process capability. In one dimension, it describes a set of process entities that relate to the processes defined in the specific PRM; in the other dimension the PAM describes capabilities that relate to the process capability levels and process attributes defined in the Measurement Framework that is part of the standard.

**Measurement Framework:** it provides a schema for use in characterizing the capability of an implemented process with respect to the PAM. Capability is defined on a six-value ordinal scale. The scale represents increasing capability of the implemented process. For the description of the meaning of each capability level, refers to [1]. The achievement of a certain capability level is established by the rating of specific Process Attributes (i.e. measurable characteristics of process capability applicable to any process). The extent a Process Attribute is achieved is measured using a four-value rating scale.

From what stated above, it should be clear that ISO/IEC 15504 is a standard having the purpose of providing a structured approach for the assessment of processes. It is neither a software development paradigm, nor a source of practices to be adopted, nor a standard to be compliant with.

### 3 The Perception of SPICE: Myths and Truth

The authors collected, on the basis of their experience as consultants for software process improvement initiatives and as SPICE assessors, common imprecise perceptions about SPICE arising from software developers once it is proposed for process assessment or improvement. In the following typical statements about SPICE developers say, are listed and commented in order to show that they are largely not justified and often based on a misunderstanding of the real nature of SPICE.

“SPICE requires being compliant with the V-model software development”

- False. It is true that the very most common PRM associated to the ISO/IEC 15504 is the ISO/IEC12207 that contains a set of processes that *de-facto* reflects those of the V-model for software development. Nevertheless, the sequence and the importance of the processes defined in the ISO/IEC 12207 are not determined in the standard. Moreover the association between SPICE and a PRM for software development may vary. Every organization can modify the ISO/IEC 12207 PRM or define a new one, without missing the SPICE compliance.

“SPICE requires the performance of a pre-defined set of Base Practices”

- False. The Part 5 of the ISO/IEC 15504 includes some Base Practices but they are not mandatory. In fact, ISO/IEC15504 Part 5 is an exemplar part of the standard. What the standard requires to achieve the Capability Level 1 is the fulfillment of the Process Purpose by means of the achievement of the related Process Outcomes, no matter what practices are used to do that.

“SPICE requires the production of a set of mandatory documents”

- False. The considerations made for the point above, are valid also for documents. SPICE does not require producing any specific document; it only requires producing documents appropriate for the project characteristics and the organization's business goals.

“SPICE – capability level 2 requires the production of a formal project plan document”

- The sentence is partially true. There is no requirements on the degree of formalism of the project plan document (it depends on the specific needs of the organization and projects), but there is the request to have the processes planned in terms of activities and performance to reach the Capability level 2.

“The compliance with SPICE determines an overload of documents”

- False. SPICE does not necessarily produce any overload of documents but only the necessary documents for achieving the target capability level. The ability of the assessors/improvers is to understand, by taking into account the specific business needs, application domain and operational environment, what are the essential documents required, without impose any superfluous effort.

“SPICE – level 3 requires following the same defined process for all the projects independently of their size and complexity”

- False. Having a defined general process is required only at Capability level 3. Having a defined process doesn't mean that such a process shall be replicated for every project independently of its size and complexity. In fact, the process definition shall be accompanied by the capability to deploy processes in a tailored manner according to the specific project characteristics.

## 4 Agile vs. SPICE: Theory and Practice

In this section we analyze the applicability of the SPICE model to Agile organizations. Several different Agile methods exist. Each agile method has its own characteristics and peculiarities [3, 4]. We do not consider any specific method for our analysis, the applicability of SPICE in agile contexts is discussed from a generic point of view considering the agile methods' common aspects and principles.

To perform the analysis we consider the five levels the capability dimension the ISO/IEC 15504 standard PAM is composed of. For each capability level we discuss if, in principle, possible drawbacks in applying SPICE to the agile methods exist.

**Level 1:** as discussed in section 3, to reach the capability level 1, a process needs to achieve its stated purpose. In general, the capability level 1 can be achieved no matter what techniques, methods or tools are applied. The point is to understand if the agile approaches can be able to achieve the purpose of the processes in the PRM in the same way the traditional approaches do. Agile practices are not always new, they are traditional practices applied with a different orientation and with the emphasis given to specific aspects (as the face-to-face communication, the high frequency of releases,...) [5, 6]. In addition, there are several works in literature [7, 8, 9] showing that the agile approaches can co-exist with a typical model based on a traditional and process-based software development: CMMI [10]. Moreover, SPICE is currently and successfully applied to organizations adopting the iterative model for software development [11]. The iterative approaches, in comparison with the waterfall or V models [12], can be considered bringing a certain degree of agility [13]. Agile can also be seen as an extreme application of the iterative approach.

**Level 2:** to be rated at capability level 2, the activities related to specific process shall be planned, monitored and, possibly, adjusted; moreover its work products shall be appropriately established, controlled and maintained. Agile processes are performed paying more attention to answering questions than following a plan as well as to short term planning. Such a principle contained in the Agile Manifesto [14] seems to be in contrast with the requirements of SPICE for level 2. On the contrary, SPICE at capability level 2 focuses on the capability to adapt the plans to changing situations and to establish the right resources allocation to the project; these aspects are fitting well with agile approaches. Moreover, SPICE requires paying attention to the quality requirements and management of Work Products. In the ISO/IEC 15504 standard the term Work Product is used in place of document. The reason is the will to refer to general artifacts produced during the software development (not only documents) and the will to emphasize the working nature of the products. An essential requirement to be satisfied at capability level 2 is the appropriate production, usage and control of work products (i.e. in a way being in line with the project characteristics and needs). Again, such an approach fits well for agile contexts.

**Level 3:** to be rated at level 3 a process shall be managed and implemented using a defined process that is capable of achieving the expected outcomes. Agile processes have the characteristic to be adaptable and flexible in order to be suitable for changing contexts. SPICE, even if it requires a defined process at capability level 3, requires at the same time the capability of tailoring (i.e. adapt the defined process) according to the specific project needs and context. Without such a capability the capability level 3

cannot be achieved. The agile processes can achieve the SPICE capability level 3 without any particular limitation.

**Level 4:** to achieve capability level 4, a process is required to use measurements to ensure that the performance of the process supports the achievement of process objectives in support of defined business goals. Agile processes are, for example, oriented on using the number of customer requirements implemented as the main measure of the progress and achievement of the objectives. SPICE doesn't indicate what the measures to be collected and used are, and then there is not incompatibility also in this case. Moreover, to achieve this capability level it is required to use the measures collected for controlling and possibly adjusting the process performance. That, again, is not in contrast with the agile philosophy.

**Level 5:** to achieve capability level 5, the process shall be continuously improved to meet the current and projected business goal. That is a target for any organization adopting any development methodology, then also for agile organizations.

We showed that ISO/IEC 15504 is, in principle, applicable to agile contexts, nevertheless it doesn't mean that it can be in practice applicable without encountering in practical problems. Below, we identify and discuss some problems that should be solved to make SPICE assessments practically executable in agile organizations.

*Process Reference Model:* we already noted that the SPICE's most used PRM is the ISO/IEC12207. We also discussed in the previous sections that it can, in principle, be applied to agile processes. Nevertheless, the ISO/IEC 12207 may be not suitable for specific contexts. In the past, several initiatives have been undertaken with the aim of personalizing SPICE in order to make it able to fit well in specific application domains. In particular, SPICE has been tailored for the space, automotive, banking and medical domains. All these initiatives arose from the difficulties to apply generic SPICE PRMs to specific domain having some peculiarities that required a tailored model. The most successful initiative (i.e. the today's most widely applied in the specific domain) is the Automotive SPICE [15, 16]. The experiences in defining specific PRMs indicate the way for defining an Agile-specific PRM in order to apply the SPICE assessment mechanism to a set of processes specifically defined and then well focused on the agile methodologies.

*Assessor's competence:* An agile-specific assessors' qualification scheme should be build-up. The assessors should be trained appropriately in order to let them ready to consider the nature of agile approaches and consequently be able to rate the processes objectively and in repeatable way. The assessors of agile processes should be aware of the peculiarities of the agile methodologies and have direct experience of software development in agile contexts.

## 5 Final Discussion and Conclusions

The experience of the last decade shows that the availability of effective models for quantitatively evaluating the capability of software organizations is very important for many reasons. First because it allows to determine the risk of having low-quality software products *a priori* (i.e. before starting a project) on the basis of the evaluation of the software development process. Moreover, it allows the setting up of process

improvement programs by identifying targets in terms of process capability. These advantages can be moved also in the *agile world*. We discussed the applicability of SPICE in agile contexts starting from the objection of some common misunderstanding, and false myths on the real nature of the SPICE approach. Then, in a more systematic way, we highlighted the suitability of the SPICE model for agile contexts as well as the barriers to be overcome in order to let it be practically applied. By choosing an agile approach, an organization should not be prevented to assess its “maturity” using the SPICE model, and to take corresponding improvement steps. The discipline of higher maturity levels over agile projects would combine the best of discipline and agility with an optimal trade-off in many contexts between cost, quality and time-to-market. In conclusion, we are optimistic about the applicability of the SPICE model to agile contexts, and we consider this paper as a contribution to get the *agilers* and the *orthodox* closer in order to mutually take advantages.

## Acknowledgements

Thanks to John Favaro and Paolo Panaroni of Intecs Spa for the precious suggestions and support.

## References

1. International Organization for Standardization. ISO/IEC 15504 International Standard “Information Technology – Software Process Assessment: Part 1–Part 7” 2008.
2. International Organization for Standardization. ISO/IEC 12207 International Standard “System and Software Engineering – Software Life Cycle Processes. 2008.
3. Bohem, B., Turner, R.: *Balancing Agility with Discipline – A Guide for the Perplexed*. Addison Wesley, Boston (2004).
4. Cohen, D., Lindvall, M., Costa, P.: *Agile Software Development*. DACS SOAR Report n. 11. Data & Analysis Center for Software. Rome (NY) (2003).
5. Cockburn, A., Highsmith, J.: *Agile Software Development: The People Factor*. IEEE Computer vol. 34(11): pp. 131-133. IEEE Computer Society (2001).
6. Highsmith, J., Cockburn, A.: *Agile Software Development: The Business of Innovation*. IEEE Computer vol. 34(9): pp. 120-122. IEEE Computer Society (2001).
7. Glazer, H., et al.: *CMMI or Agile: Why Not Embrace Both!*. Technical Note CMU/SEI-2008-TN-003. Carnegie Mellon University - Software Engineering Institute (2008)
8. Paulk, M.C.: *Agile Methodologies and Process Discipline*. CrossTalk: the Journal of defense Software Engineering. Vol. 15(20): pp. 15-18 (2002)
9. Paulk, M.C.: *Extreme Programming from a CMM Perspective*. IEEE Software, vol. 18(6): pp. 19-26. IEEE Computer Society (2001)
10. Chrissis, M.B., Konrad, M., Shrum, S.: *CMMI Guidelines for Process Integration and Product Improvement*. Addison-Wesley (2004)
11. Larman, C., Basili, V.: *Iterative and Incremental Development: A Brief History*. IEEE Computer vol. 36(6): pp. 47-56. IEEE Computer Society (2003)
12. Sommerville, I.: *Software Engineering* 6<sup>th</sup> Ed. Addison Wesley (2001)
13. Larman, C.: *Agile and Iterative Development: A Manager’s Guide*. Addison Wesley Professional (2003)
14. Beck, K., et al.: *The Agile Manifesto* (2001), <http://www.agilemanifesto.org>
15. Automotive SIG: *Automotive SPICE Process Assessment Model (PAM)*, rel. V.2.4 (2008)
16. Automotive SIG: *Automotive SPICE Process Reference Model (PRM)*, rel. V.4.4 (2008)