Scientific Report Organizational Science

Integration Evaluation of Scrum and CMMI
-
Theoretical Survey at the Project-Management Level

Armin Preis
research@arminpeis.at
www.arminpreis.at

2012-01-18
Vienna, Austria

Abstract:

In the software industry the use of quality management systems like ISO 9001, CMMI for Development or SPICE is quite common. These quality management systems define the organizational frame for development and standardize the development processes within the organizational context. The major question is: *Are these strict and formal process specifications compatible with the idea of agile software development?* And if so: *How to realize the integration of these two seemingly different approaches*? The goal of this paper is to analze the feasibility of the CMMI quality management system in conjunction with the agile development and project management methodology Scrum.

Table of Contents

Abstract

In the software industry the use of quality management systems like ISO 9001, CMMI for Development or SPICE is quite common. These quality management systems define the organizational frame for development and standardize the development processes within the organizational context. The major question is: Are these strict and formal process specifications compatible with the idea of agile software development? And if so: How to realize the integration of these two seemingly different approaches? The goal of this paper is to analyze the feasibility of the CMMI quality management system in conjunction with the agile development and project management methodology Scrum.

1 Introduction

In the software industry the use of quality management systems (QMS) like ISO 9001, CMMI for Development or SPICE is quite common as the improvement of development processes has been empirically linked to the improvement of software quality. Quality management systems define the organizational context and standardize different development and management processes in software development enterprises.

One disadvantage of the mentioned quality management and process improvement systems is that the documentation and formalization-overhead involved is enormous. Many companies and especially developers claim to suffer from this overhead and desire more efficiency in development and project management while maintaining the high quality of their software products.

Over the last decade a new software development methodology, that might be an answer to the depicted problem, has gained foothold in the software industry: Agile software development. It is based on iterative and incremental development where requirements and solutions evolve with the progress of the project. It is highly dependent on adaptive planning as well as rapid and flexible response to change.

Both approaches, software improvement frameworks and agile development methods, have proven to be a valuable asset in the toolbox of a software development organization. However, one major question arises: Are these strict and formal process specifications compatible with the idea of agile software development? And if so: How to realize the integration of these two seemingly different practices?

The goal of this paper is to analyze the feasibility of the CMMI process improvement in combination with the agile software development and project management method Scrum. We will start the discussion of this question on the common ground they share, but also analyze apparent differences of these divergent frameworks. The details on the way the theoretical analysis is conducted and how empirical evidences are considered can be found in the following section about research issues.

2 Research Issues

2.1 Research Methodology

The research methodology of choice is a broad literature review and an analysis of Specific Goals and Practices in the Project Management Process Areas of CMMI for Development. Especially for the integration evaluation of Scrum at higher CMMI Maturity Levels empirical evidence like case studies will be provided. The goal of this survey is to find out whether Scrum can fulfill the the CMMI project management requirements and whether Scrum fits to the broader general vision and goal of CMMI Maturity Levels two to five.

2.2 Research Hypothesis

The decision for this research approach bases on the strong project management characteristics of Scrum. The research hypothesis is that CMMI and Scrum are not mutually exclusive approaches for software quality improvement, but rather, combined with one another, a strong basis for software quality improvement. The second part of the hypothesis is that the broadly claimed disadvantages of Scrum (e. g. "just applicable for small projects") and CMMI (e.g. "involves too much overhead for small organizations") can be reduced by the combination of the two primary distinct models.

2.3 Research Questions

1. Does Scrum fulfill the project management requirements of CMMI Maturity Levels two to five?
2. If not, can Scrum be adopted, in order to fit the various CMMI project management requirements?

2.4 Specific CMMI Project Management Goals as Basis for the Evaluation

This survey examines the chances for integration and possible synergy effects on the different Maturity Levels of CMMI. Due to the nature of Scrum as a project management method the focus of the integration evaluation lies on the project management process areas within different CMMI Maturity Levels. As additional connecting factor between CMMI and Scrum the Agile Manifesto is taken in consideration.

Main measures of integrability are Specific Goals of the CMMI Project Management Process Areas and Generic Goals of CMMI Maturity Levels. The review of Specific Goals and Practices of CMMI, that are related to the Scrum project management process, will reveal common ground. The Generic Vision and Goals of CMMI are considered in order to reveal a way for the institutionalization of Scrum practices in organizations. The influence of organizational culture on the integration of the Scrum and CMMI frameworks is examined as second crucial factor of integrability.

3 Integration Evaluation of CMMI and Scrum

On the basis of the depicted research approach the evaluation of a possible integration of CMMI for Development and Scrum is done after a short clarification of necessary prerequisites for the integration.

3.1 Prerequisites for Integration

Boehm describes that the evolution of software development methods tends to swing between extremes like a pendulum over time [47]. In the long run all development-methodologies advance on the basis of second order feedback loops created by the experiences made by the application of distinct models. What is common to most software development frameworks is the shared vision of creating better software sometimes even with less resources or in a shorter period of time. CMMI for Development and Scrum share the goal of improving software quality, yet the approaches differ. CMMI focuses on written communication and documentation to share explicit knowledge within whole companies, whereas Scrum focuses on direct, interpersonal communication to share implicit knowledge mainly within development teams [47]. The subsequent concept for the integration of CMMI and Scrum focuses firstly on common practices or processes and secondly on synergy effects that can arise by the integration of divergent strategies and objectives of the two methodologies.

A serious integration evaluation of CMMI and Scrum requires an accurate definition of both concepts. So, before presenting the evaluation results, some common misunderstandings and misconceptions are put straight in the following in order to keep the concepts and their application as concise as possible.

3.2 The Concept of CMMI and Frequent Misunderstandings

Anderson et al. adduce misuse, lack of accurate information and terminology difficulties as major problems during in the implementation of the CMMI in organizations [48]. They highlight the reduction of unnecessary effort in order to make organizations leaner and to get in respectively keep in contact with actual development progress as the ultimate goal of CMMI. Baker underlines the view of Anderson et al. and explains that CMMI "focuses on the goals that must be achieved but not how to achieve them" [8], what means that development teams and process managers should not only apply all practices listed in the CMMI standard, but rather realize which processes and practices already exist in the company.

On this ground the improvement of processes and the implementation of CMMI is done in a serious and realistic basis, namely the current situation and specific objectives of the particular organization. Baker stresses the importance of such a tailoring of different CMMI practices to prevent the misuse of CMMI as simple cookbook. According to his experiences many companies make the mistake that development teams take the goals described in CMMI too literally [8]. He states that some enterprises even try to gain perfection in each CMMI policy statement and

process flow definition what results in selecting and customizing every template that could be found while the real paradigm of CMMI gets lost [8].

Organizations that decide to use the CMMI framework, must know the distinction between obligatory and optional documentation elements provided in CMMI [1] in order to keep a clear image of what is necessary and what is meant as supportive measure. Another point of interest concerning obligatory and optional elements is the fact that CMMI "does not prescribe specific software development approaches" [31], rather it was created in a way that should support the integration of various development models as Leithiser and Hamilton explain.

This knowledge clarifies that the use of Scrum or other agile development and project management methodologies is not inconsistent with CMMI. In contrast, the use of the Standard CMMI Appraisal Method for Process Improvement (SCAMPI) is the method of choice for the evaluation of CMMI in combination with integrated and customized development and project management methods [53]. An important prerequisite for a SCAMPI Appraisal is that the development methodology organizations use is defined concise and used in the exact, defined way. That involves in the case of the CMMI and Scrum integration evaluation a clear image of Scrum to prevent possible misuse.

3.3 The Vision of Scrum and Possible Misuse

It is crucial to know that Scum does have strict rules and regulations. It may not be wrongly used as pseudo-concept in order to gain freedom in development. Scrum does not allow developers to hack and develop what and however they want, as all people involved in Scrum process have to stick to the Scrum rules [45], [46] as strict as people have to approve CMMI rules. So the fact that Scum fosters creativity and interpersonal communication should not mask the discrete and defined underlying processes.

Again, although Scrum fosters direct interpersonal communication it does not stand in contrast to CMMI. Interpersonal communication and creative meetings can be defined as processes including different sub-practices. A major challenge during the integration of Scrum and CMMI is to keep the clearness of Scrum in mind when dealing with possible ways of integration. The awareness of this situation does already reduce contradictions and facilitate smother integration.

Another enabler for the integration of project management and development methods in the CMMI concept is the Agile Manifesto [50]. The ways in which the manifesto supports the desired combination is described below.

3.4 The Agile Manifesto as Enabler for Integration

The Scrum methodology is a specific manifestation of the agile mindset combined with the tacit knowledge management methods introduced by Nonaka and Takeuchi [49]. In order to keep aware of the whole potential of agile development methods, references to the Agile Manifesto [50] and some other agile methodologies like

Extreme Programming will be adduced later on as supportive elements that facilitate the integration of Scrum and CMMI.

An interesting aspect is that according to Leithiser and Hamilton [31] much of the CMMI basis stems from Deming's work on the Deming Cycle and statistical control theory in management [51]. Additional to this hard facts management methods Deming highlights 14 points "for softer subjective guidance based on systems thinking", [6] what he claims to be extremely beneficial for successful management. Anderson surveyed that seven of these points are central aspects of agile development approaches [6]. He adduces 'supplier trust and loyalty', 'training on the job', leadership', 'drive out fear' and 'break down barriers between departments' as such elements and makes them traceable to the agile mindset and frameworks [31].

The fact that Scrum and CMMI have common roots in Deming's work might be one first silver lining on the way to the integration evaluation of Scrum and CMMI. It seems that Deming already had the idea that both the 'hard fact and soft fact', [31] management elements, that are on the one side part of Scrum and on the other side of CMMI, must be combined to gain maximum business performance. In the evaluation this hypothesis will be considered with a focus on common project management processes, visions and required cultural change for the integration of Scrum on different CMMI Maturity Levels.

3.5 Evaluation Results

The results of the integration evaluation are presented in this section. On the ground of CMMI Project Management Process Areas including Specific Goals and Practices as well as Generic Goals of higher CMMI Maturity Levels the results are presented for each Maturity Level from two to five.

3.5.1 Scrum at CMMI Maturity Level 2

Marcal et al. have examined the Specific Goals (SG) and Specific Practices (SP) of this process area in order to measure the Capability Degree of Scrum processes in project planning [32]. Diaz et al. mapped to Scrum project management practices to CMMI Level 2 practices [54]. The results of both theoretical papers are further analyzed and classified in 'satisfied', 'partially satisfied' and 'unsatisfied' Specific Goals in the subsections below.

3.5.1.1 Project Planning (PP) - Project Management Process Area

The Specific Practices of the three Specific Goals *SG 1 'Establish Estimates', SG 2 'Develop a Project Plan'* and SG 3 *'Obtain Commitment to the Plan'* are compared with the defined practices of the Scrum process. The specific goals are presented in the three categories described above.

Satisfied Specific Goals

SG 1 Establish Estimates

Estimate the Scope of the Project (SP 1.1): This Specific Practice is satisfied by the Pre-Game Planning phase where the Product-Backlog is created [54] and by detailed estimates at the beginning of each Sprint [32].

Define Project Lifecycle (SP 1.3): Scrum defines a concrete lifecycle [45], [46] consisting of planning-, staging, development- and release-phases what fully satisfies this Specific Practice [32], [54].

SG 2 Develop a Project Plan

Establish the Budget and Schedule (SP 2.1): Diaz et al. explain that budged is set up in the Pre-Game phase based on the initial Product Backlog, additional budged can be assigned in later Sprint Plannings and corrections can be done in the Retrospective Meeting what makes SP 2.1 fulfilled from their view [54]. Marcal et al. claim that "Scrum doesn't provide orientations about establishing budget" what makes this specific practice just partially satisfied from their point of view. The exact examination of this issue must follow the conception of Diaz et al. as all necessary sub-practices of SP 2.1 [1] are fully satisfied.

Plan for Project Resources (SP 2.4): The staffing and equipment resources are defined in the Pre-Game phase of Scrum [32] and during the project the Scrum Master is responsible for the management of resources [54].

Plan Stakeholder Involvement (SP 2.6): The involvement of the stakeholders is defined in the Scrum Process during the different lifecycles [54]. The Scrum Master monitors the involvement and registers it in a communication plan [32]. He is responsible for corrective actions concerning the involvement of stakeholders.

Establish the Project Plan (SP 2.7): The Project Vision and Product Backlog satisfy this specific practice [32], [54].

Review Plans That Affect the Project (SP 3.1): Sprint Plannings and Retrospective Meetings satisfy this practice [54], as "the CMMI model doesn't explicit which plans need to be revised, such as the QA plan, the CM plan nor the Test plan amongst others", [32].

SG 3 Obtain Commitment to the Plan

Reconcile Work and Resource Levels (SP 3.2): The reconciliation and iterative re-planning by the Scrum Team, Product Owner and Scrum Master [32] during the Planning and Retrospective Meetings satisfy this practice as the Product Backlog is changed dynamically [54] according to relevant issues.

Obtain Plan Commitment (SP 3.3): The Product Owner is responsible for obtaining plan commitment during meetings with stakeholders [54] and even during sprints by the possibility of removing Backlog Items when the planned workload is too high [32].

Partially satisfied Specific Goals

SG 2 Develop a Project Plan

Identify Project Risks (SP 2.2): Risks are recorded as so called Impediments, but there is no systematic plan on how to identify risks as Diasz et al. describe [54]. Marcal et al. adduce that risk categories and risk sources which would be a serious basis for risk identification are missing in the Scrum definition [32]. So SP 2.2 is at least partially fulfilled.

Plan for Needed Knowledge and Skills (SP 2.5): In the beginning of a Scrum project the required knowledge and skills are analyzed and planned. According to the requirements the Scrum Team is arranged, though Scrum defines no systematic mechanisms for recording skills of developers. So this Practice is partially satisfied.

Unsatisfied Specific Goals

SG 1 Establish Estimates:

Establish Estimates of Work Product and Task Attributes (SP 1.2): Size, service level, connectivity, complexity, availability and structure are major attributes mentioned in CMMI Standard [1]. Scrum does not explicitly include orientations on the attributes themselves or methods for establishment of attributes [32]. So SP 1.2 unsatisfied.

SG 2 Develop a Project Plan:

Plan for Data Management (SP 2.3): Diaz et al. show that Scrum uses public folders and whiteboards for planning and data management, however they reveal that "there is no formal data management plan or procedure to collect this data" [54], as Marcal et al. approve. So SP 2.3 is unsatisfied.

Results-Outline of this Area

In general, Scrum satisfies most of the Specific Goals of the Project Planning Process Area. Some adaptions like the implementation of a more sophisticated and systematic risk- [22], skill-management [14] and data-management [31] would increase the CMMI compliance of Scrum to a very high level.

3.5.1.2 Project Monitoring and Control (PMC) - Project Management Process Area

The fulfillment of the Specific Goals and Specific Practices in the CMMI Project Monitoring Process Area by Scrum are reviewed in the following paragraphs. This CMMI Process Area contains the Specific Goals SG 1 *'Monitor Project against Plans'* and SG 2 *'Manage Corrective Action to Closure'* .

Satisfied Specific Goals

SG 1 Monitor Project against Plans

Monitor Commitments (SG 1.2): Daily Meetings, Sprint Burndown Charts and the Retrospective Meeting fulfill the required monitoring of commitments [32], [54]. The monitored project-state is compared to the current Sprint Backlog.

Monitor Stakeholder Involvement (SP 1.5): Marcal et al. explain that the Scrum Master is responsible for monitoring the stakeholder involvement during Project

Meetings [32]. According to them the indirect evidence of updated Impediment Lists, Product Backlog and Sprint Backlog satisfies the specific monitoring practice. Diaz et al. see the Retrospective Meeting [54] as relevant Scrum practice for the stakeholder monitoring.

Conduct Progress Review (SP 1.6): Progress Review Meetings and frequent inspections [32] allow the conduction for progress reviews via Burndown Graphs [54]. Again, the Scrum Master is responsible for the necessary involvement of relevant stakeholders in the progress review, in order to coordinate corrective and adaptive actions.

Conduct Milestones Review (SP 1.7): Milestone Reviews are conducted in the Sprint-Review Meetings, see [32], [54].

SG 2 Manage Corrective Action to Closure

Analyze Issues (SP 2.1): The gathering and analysis of issues, required to determine the need for corrective action, is done in the Daily Meeting where team members report Impediments "against expected quality or performance levels", [32] and record them on whiteboards or Impediment Lists. The Retrospective Meetings provide a further context where bigger issues are gathered and analyzed [54].

Partially satisfied Specific Goals

SG 1 Monitor Project against Plans

Monitor Project Planning Parameter (SP 1.1): Diaz et al. describe that the monitoring happens in Daily and Retrospective Meetings [54] by the use of Burndown Charts that allow for example the analysis of the release speed [32] as Marcal et al. declare. There are no systematic Scrum rules for monitoring cost, size and effort estimates in Scrum [32] what makes SP 1.1 only partially satisfied.

Monitor Project Risks (SP 1.3): As explained above, risks are gathered with different tools like whiteboards or Impediment Lists, however the monitoring process is an informal one [32] what keeps SP 1.3 partially satisfied.

Unsatisfied Specific Goals

SG 1 Monitor Project against Plans

Monitor Data Management (SP 1.4): Scrum does not include any practices for monitoring data management, see [32], [54].

Results-Outline of this Area

Scrum fulfills most requirements of this Process Area perfectly. Again, the introduction of systematic approaches for monitoring cost, size, effort [13], risks and data would raise the adapted Scrum practice to a very high CMMI compliance.

3.5.1.3 Requirements Management (REQM) - Project Management Process Area

The only Specific Goal of the Requirements Management Process Area is *SG 1: "Requirements are managed and inconsistencies with plans and work products are identified."*, [1] with the following Specific Practices:

Develop an understanding with the requirements provider on the meaning of the requirements (SP 1.1): The intense involvement of stakeholders guarantees an common understanding of the requirements' meanings. User Stories are a further mean that supports the fulfillment of SP 1.1 [54].

Obtain commitment to requirements from project participants (SP 1.2): The general commitment is ensured by Backlogs created in Planning Meetings [54]. The obtaining of commitment is a task of the Scrum Master who can take necessary actions to gain commitment or to reduce requirements in cooperation with stakeholders and the Scrum Team.

Manage changes of requirements (SP 1.3): During the Sprint Planning and Review Meetings the changing requirements are managed [54] by all involved parties. The agile mindset and the frequent meetings in the Scrum methodology embrace changing requirements and fulfill SP 1.3 entirely.

Maintain bidirectional traceability among requirements and work products (SP 1.4): The use of User Stories satisfies SP 1.4 [54]. Important is the precise application of User Stories to enable horizontal and vertical traceability.

Ensure that plans and work products remain aligned with the requirements (SP 1.5): Diaz et al. see the Planning Meetings and the Pre-Game as satisfying activities for this Specific Practice [54]. The Backlog Item-List is created initially in the Pre-Game phase, but the actual guarantee of alignment happens in the Sprint-Planning Meetings and the Daily Meetings.

Results-Outline of this Area

Regarding the Requirements Management Process Area all Specific Practices, Scrum satisfies all required Specific Practices entirely. So the Scrum Practices can be used for requirements management in the CMMI context without any further adjustment.

3.5.1.4 Supplier Agreement Management (SAM) - Project Management Process Area

Marcal et al. explain that there is no Scrum activity addressing the acquisition of products from suppliers what makes all Specific Practices of this Process Area Unsatisfied [32]. Omran explains the situation of SAM under the conditions of agile development (XP) in similar ways and traces it back to the high consume of resources in small teams [35]. So Supplier Agreement Management is the first Project Management Process Area that Scrum does not cope with at all. Wether

there are any contradictions between SAM and the Scrum methodology has to be detected in further research.

Results-Outline of this Area

No Specific Goal of this Process Area is satisfied. Research on the feasibility of SAM in combination with Scrum has to be done in further surveys.

3.5.1.5 Similar Surveys on other Agile Methodologies

Omran [35] compares the agile principles of Extreme Programming (XP) to the CMMI Project Management Process Area and draws similar conclusions as Marcal et al., namely the necessity of more rigidity in some Specific Practices. According to him the focus on the first intensive planning game and the succeeding planning strategy with frequent reconsiderations and small releases fulfills the same practices as Marcal et al. state for Scrum.

3.5.2 Scrum at CMMI Maturity Level 3

Sutherland et al. explain that the progression to CMMI Level 3 is crucial as many development companies have implemented Scrum correctly, but fail due to a missing institutionalization of the Scrum processes [41]. Another problem characterized by them is that enterprises that comply with CMMI often miss optimal performance due to imperfect process-implementations. Both issues can be addressed by the institutional integration of Scrum at CMMI Level 3 organizations and of course later on at higher Maturity levels by stipulated performance optimization.

The fulfillment of the Specific Goals and Practices in the Project Management Process Areas by Scrum activities are again depicted by Marcal et al. [32] and further reviewed with regard to the CMMI definition [1] of Specific Practices and Sub-Practices.

3.5.2.1 Integrated Project Management (IPM) - Project Management Process Area

The Integrated Project Management Process Area contains the Specific Goals SG 1 *'Use the Project's Defined Process'* and SG 2 *'Coordinate and Collaborate with Relevant Stakeholders'* .

Satisfied Specific Practices

SG 2 Coordinate and Collaborate with Relevant Stakeholders

Manage Stakeholder Involvement (SP 2.1): This practice is satisfied by the Scrum Rules and practices concerning stakeholder involvement, as Marcal et al. confirm [32].

Partially satisfied Specific Practices

SG 2 Coordinate and Collaborate with Relevant Stakeholders

Manage Dependencies (SP 2.2): Though there exists general dependency management by the use of Impediment-Lists, a comprehensive dependency management system including planning of tracking strategies or verification and registers of negotiation are missing, as Marcal et al. surveyed [32].

Resolve Coordination Issues (SP 2.3): "This practice is Partially Satisfied, for the same reason presented for SP 2.2", [32].

Unsatisfied Specific Practices

SG 1 Use the Project's Defined Process

The goal is, to conduct projects using "a defined process tailored from the organizations' set of standard processes" [1]. As the Scrum processes and practices are not yet installed in the organizational level, but rather derived from the Scrum definition itself, all practices of the SG 1 are classified as unsatisfied. Marcal et al. came to the same conclusion [32] in their theoretical survey.

Results-Outline of this Area

The Specific Goals of this Process Area are, in contrast to CMMI Maturity Level 2 Project Management Areas, only satisfied partially. The definition of the Scrum processes at organizational Level is necessary and has to be evaluated in the context of cultural change management and General Goals of CMMI Maturity Level 3.

3.5.2.2 Risk Management (RM) - Project Management Process Area

Marcal et al. underline that in the Scrum framework risks are identified, but there exist no explicit rules or practices to define categories and sources of risk what would be necessary for serious risk management [32]. The only specific practice that is partially fulfilled is the identification of risks (SP 2.1), as it is done in an informal way. All other Specific Goals and Practices are rated unsatisfied [32]. Omran illustrates the situation in the, in many of the relevant practices comparable, agile Extreme Programming framework in similar ways [35].

Results-Outline of this Area

With the exception of the partially fulfilled "identification of risks" Specific Practice, Scrum does not satisfy the Specific Goals of this Process Area.

3.5.2.3 Organizational Change Management and Higher Maturity Levels in the Back of Organizations Minds

Sutherland et al. argue that shared common processes within organizations make it easier for developers to share experiences and to move between projects [41], so the institutionalization of Scrum as managed process is a core advantage of the combination between Scrum and CMMI.

Baker reflects deeply on the institutionalization of Scrum: "We believe that sustainable culture change comes from within. Many teams seeking to embrace agile

methods hire an agile coach to mentor them; likewise, we hired an experienced CMMI consultant to guide us through the model and its real-world application.", [7]. What baker tries to make clear is that organizational change management is an enduring and iterative process that arises within an organization and can not be replaced by consultants that introduce new frameworks and leave the company afterwards without any further support.

The way in which organizations handle change management and institutionalize Scrum as Defined and Managed Process as the Generic Goals of this Maturity Level demand depends on individual organizational aspects like culture and objectives as [39|, [43]. Turner and Chain [42] as well as Greening [17] present different ways to install Scrum as enterprise wide defined and managed processes.

As much as change management is an continuous and never ending process within an organization, as much is the CMMI implementation. According to DeMarco and Boehm many organizations stop their maturity development process at ML 3, because many governmental organizations demand exactly this level from producing enterprises [9]. This organizational mindset results in a documentation overhead where nothing ever gets removed from, whereas the higher Maturity Level 5 would explicitly aim for simplification [47]. This situation induces the feeling of producing documents to fulfill CMMI goals instead of delivering business value to customers [18].

Empirical evidence on Scrum at CMMI Maturity Level 4 in combination with supportive enablers is provided in the following subsection.

3.5.2.4 Empirical Evidence - A Case Study Outline of Supportive Facilitators

The CMMI framework is typically used for large and complex projects within rather big organizations that can afford the involved overhead [31]. In order to reduce this overhead and to make CMMI Maturity Level 3 achievable for smaller organizations, integrated tools for configuration management have empirically turned out to be useful.

Hansen and Baggesen describe their positive experiences with the Microsoft Team Foundation Server TFS that provides one place for all source code, documents and builds even for distributed development teams [18]. The TFS is not only a configuration management system, but also addresses 17 of the 21 Specific Goals of Level 3 Process Areas [31]. Within the TFS developers are free to chose which templates to apply according to the development methodology they use, so they can choose for example a agile- or CMMI-compatible templates their project, which involves automated generation of corresponding files and configuration management [31].

The automated generation of files and documentation as well as the integrated configuration management reduce the workload for developers while CMMI

compliance at Maturity Level 3 is maintained. Similar recommendations on configuration management and some further integration facilitators provide Jakobsen and Johnson in their work on the combination of agile methods and CMMI [22].

3.5.3 Scrum at CMMI Maturity Level 4

3.5.3.1 Specific Goals in the Quantitative Project Management (QPM) - Project Management Process Areas

The Specific Goals of the Quantitive Project Management Process Area are *SG 1 "preparation for quantitative management"* and *SG 2 "quantitative management of projects"* and the corresponding Specific Practices are not addressed in Scrum [32] or other classic agile methods [35]. Concerning SG 1, Scrum misses for example a defined use of quantitative or statistical techniques, performance measurement of subprocess by critical attributes and further analytic techniques. Regarding SG 2, a root cause analysis is missing additionally to the mentioned topics [1].

Results-Outline of this Area

The Specific Goals of this Process Area are not addressed by Scrum practices. Especially quantitative control and root cause analysis are missing.

3.5.3.2 Empirical Evidence and Case Studies

One approach for introducing quantitative project management while keeping the agile Scrum process alive is to use integrated development-, configuration- and project management systems that support automated generation and monitoring of statistical project- and development status-data.

As mentioned before, the Microsoft Team Foundation Server (TFS) is one example for such an integrated system. The TFS addresses partially the process areas "Quantitative Project Management" as well as "Causal Analysis and Resolution" through reporting features, while still supporting agile development methods [31]. Systems like the TFS that support the integration of agile processes in formal quality management and process improvement systems might be an essential enabler for the integration of Scrum at advanced Maturity Levels of CMMI. Further examination on similar systems will be necessary in futher surveys.

Cohan and Glazer explain that high-maturity practices of CMMI Maturity Levels 4 and 5 help to quantify the performance of agile practices and to analyze data from agile projects instead of "relying on 'gut feel'", [10]. This is an important aspect, as many organizations do not use methods that are empirically proved to be effective and efficient for them. One thing is evident, the fact that a development method is trendy or used by many companies does not implicate that it fits any organization, culture or objectives.

Cohan and Glazer also illustrate their experiences with agile metrics in the context of CMMI. The describe in the context of their case study each metric they use and the

corresponding perceived value very detailed [10]. They proclaim that the gathering of metrics that satisfy CMMI Maturity Level 4 and 5 requirements is difficult to define for enterprises with little experience in this field, however the continuous occupation with agile measures [19], [20], [43] has showed up to lead in the right direction by an refining, iterative process, according to Cohan and Glazer.

Cohan and Glazer explain that the choice of the right metrics is crucial [10]. First they explain that metrics may not hinder the development-team's ability to work agile. Second they add that metrics must be easy to collect and be useful for predicting future outcomes in order to improve future performance.

The case study of Cohan and Glazer is one of the rare empirical evidences for the implementation of agile practices and measures at CMMI Maturity Level 4. At this state of research it is not sure, whether Scrum and CMMI at Maturity Level 4 fit together without serious contradictions. There exist agile approaches like the ones depicted above that worked for some organizations, nevertheless a general statement about the integrability of Scrum and CMMI at this high level requires further empirical evidence.

3.5.4 Scrum at CMMI Maturity Level 5

3.5.4.1 Organizational Performance Management - Project Management Process Area

The Organizational Performance Management Process Area includes three Specific Goals:

SG 1 Manage Business Performance: "The organization’s business performance is managed using statistical and other quantitative techniques to understand process and to identify areas for process improvement. ", [1]

SG 2 Select Improvements: "Improvements are proactively identified, evaluated using statistical and other quantitative techniques, and selected for deployment based on their contribution to meeting quality and process performance objectives.", [1]

SG 2 Deploy Improvements: "Measurable improvements to the organization’s processes and technologies are deployed and evaluated using statistical and other quantitative techniques." [1]

None of the Specific Goals of Organizational Performance Management at CMMI Maturity Level 5 are satisfied by Scrum practices, as statistical and quantitative measurement, management as well as corresponding actions and deployments are missing in the definition of the Scrum method.

Results-Outline of this Area

Scrum satisfies no Specific Goals of this Process Area.

3.5.4.2 Empirical Evidence - The Systematic Case Study

Boehm explains that CMMI Level 5 advocates to screen where organizations are overdoing formal documentation and to reduce it [47] according to the experiences and knowledge gained during the maturity process that led to Level 5.

The Systematic case study by Jakobsen and Sutherland provides empirical evidence for this synergetic maturity process leading to strong efficiency. Jakobsen and Sutherland describe the introduction of Scrum at the CMMI Level 5 compliant company Systematic. Systematic has about 500 employees in Denmark and worldwide and focuses on complex and critical IT solutions with high demands on reliability and safety [23]. They explain: "From a CMMI perspective Scrum is one process out of a set of processes used to execute a project. In a CMMI context all processes for development are monitored for effectiveness and efficiency. Therefore measures were also established on the Scrum process. [...] We wanted a measure to help establish focus on a 'Stop the line' mindset to defects, to ensure defects are addressed immediately after they are identified." [23].

This Philosophy led, in the case of Systematic, to measures like "fix time after failed builds", to find out whether problems are handled proactively and "flow in implementation of story", to find out whether an implementation is done without brakes.

Later on, they introduced the objectives "reduce average fix-time after failed build to less than a working day" and "increase flow of implementation of story to greater than 60%" for special projects [23]. "As a consequence the productivity of those projects became 140% and 360% better than the average one in the company. "The two projects participated in piloting of the use of cosmic function points (CFP) as a measure for size [...] and used these measures to systematically identify impediments to meet the overall objective to be able to deliver high quality working code to the customer every month. Both measures are established using the disciplines from CMMI and analyzed using statistical process control techniques. [...] The causes were addressed and resolved with an attitude based on lean and agile values, where management in a respectful way supported the projects by eliminating impediments. [...] A Lean mindset suggests that you address a defect immediately after it is identified as opposed to a mindset where defects are stored to be fixed later." [23].

"Using CMMI and Scrum together results in significantly improved performance while maintaining CMMI compliance. Scrum reduces every category of work (defects, rework, total work required, and process overhead) by almost 50%. We now have a clearly defined strategy to reduce all categories of work by 75% and have achieved that goal with a small number of teams. That success needs to be institutionalized in the company." [23]; Sutherland et al. explain further that the combination of CMMI and agile practices "assures 92% of all milestones are delivered early or on time", [41].

Considering the General Goal of Organizational Performance Management at CMMI Maturity Level 5, in the Systematic company it was observed that in small projects (less that 4000 hours) "the productivity is 181% the productivity of large projects", [41]. By the introduction of Scrum at the CMMI Level 5 in this company it was monitored by Sutherland et al. that the productivity of small projects did not change significantly, but the productivity of large projects showed an increase of 201% [41].

They belief the missing change in small projects is due to the facts that this kind of projects had already been managed in an very agile way before. Nevertheless the strong performance improvement in large projects is especially interesting at CMMI Level 5. Companies that move to CMMI Level 5 must have a lot of experience in their development field as well as with the improvement, control and institutionalization of their processes. Usually, companies that fulfill these demands have the necessary competence and drive to conduct and manage large projects.

If we transfer the goal of reducing unnecessary processes to the customers of CMMI Level 5 organizations it is interesting to see that Sutherland et al. kept track of the fact that the introduction Scrum practices reduced requirements of customers and the price of the desired product by 50%, [41]. This can be traced back to a strong and frank involvement of customers at early project phases and aligns the results of the Standish Group Study which reports that "64% of features in a fixed price contract are never or rarely use by end-users", quoted in [41].

Lavallée and Robillard show a further interesting empirical evidence for the necessity of combining CMMI with other methods. According to them, CMMI can reduce the number of software defects, but not strongly increase efficiency [30], so that methods like Scrum seem to be even necessary to fulfill the General CMMI Goal of increased efficiency.

3.6 Discussion of the Evaluation Results

The results of the evaluation at different CMMI Maturity Levels are clearly depicted in the different results subsections above. In the following, the two research questions and possible answers are discussed. An summary over the whole evaluation can be found at the next section 'summary'.

3.6.1 Answers to the Research Questions

The results of the theoretical integration evaluation in combination with several case study evaluations are sufficient for answering the two research questions concerning the integration of CMMI for Development and Scrum.

1. Does Scrum fulfill the CMMI project management requirements of CMMI Maturity Levels (ML) two to five?

ML 2: The Scrum processes satisfy all Project Management Specific Goals but "data management" and "risk management".

ML 3: Scrum processes and activities satisfy the the Project Management Specific Goals at this level "partially" at large. The missing definition of the Scrum processes at organizational Level is the main deficiency.

ML 4: Scrum does not address or satisfy the Specific Project Management Goals of this Level.

ML 5: Scrum does not address or satisfy the Specific Project Management Goals of this Level.

2. If not, can Scrum be adapted, in order to satisfy the CMMI project management requirements?

Yes, according to the theoretical evaluation, empirical case studies and current state of research, the Scrum processes can be adapted in a way that satisfies all CMMI Project Management Process Areas.

At CMMI Maturity Levels 2 and 3 many organizations have reached this goal by minor changes like the introduction of systematic risk management, data management and the institutionalization of the Scrum processes.

At CMMI Maturity Levels 4 and 5 the adaption of Scrum requires sophisticated knowledge in agile measurement and strong organizational change management experience. Nevertheless, there exist first empirical evidences, like case studies, showing that the implementation of Scrum even at Maturity Level 5 is feasible. However, ML 4 and 5 case studies are rare, so that further empirical evaluation and proof is necessary.

4 Summary

Summary of the Evaluation

The seemingly opposite mindsets of CMMI and Scrum dissolve by a deeper look at the background and vision of both concepts that reveals common visions and objectives like producing high quality software in an effective and efficient way.

The evaluation of the Scrum and CMMI integration approach shows that, as Deming proposed decades ago, hard fact and soft fact management must be combined to gain optimal business performance.

The Scrum project management processes and practices satisfy the CMMI project management requirements at Maturity Level 2 fully and at Level 3 at least partially. The evaluation shows that the integration and the adaption of Scrum, in order to satisfy all CMMI project management goals, is feasible by introducing systematic data-, risk-, and knowledge-management. Further, the institutionalization of Scrum as managed process is no problem, or even a natural step in the maturity process of organizations.

The project management requirements of CMMI Maturity Levels 4, namely quantitative and statistical evaluation and control, is reachable by the introduction of agile metrics. This requires strong self-reflexion and sustainable organizational change management. CMMI Maturity Level 5 and Scrum seem to be ultimate partners, as the goal of Maturity Level 5 is to reduce complexity and increase organizational and process efficiency. After having reached ML 4 the transition to ML 5 in combination with Scrum practices showed strong synergy effects resulting in clear increase of efficiency and defect reduction, especially in large projects

On the whole, Scrum and CMMI match in large parts and the integration of both concepts can involve strong synergetic effects, like reducing complexity in CMMI and and expanding Scrum's processes towards higher process quality. Nevertheless, organizations must be aware that it usually takes long, to combine both concepts and to improve them over the lifecycle of the maturity process. Nevertheless, it can be worth it, when the whole organization is aware of the effort that is necessary to change the organizational system in order to reach the common vision of better software and more efficiency.

Future Prospects for Software Engineers

On the level of organizational culture it became obvious that the introduction of CMMI in enterprises tends to isolate teams from each other and builds distrust, whereas agile practices engage collaboration and build trust [18]. So perhaps the main question of interest should not be, whether CMMI and Scrum can work together, but quite the contrary, namely, if CMMI and Scrum can exist in way without one another in a sustainable, large organization.

Perhaps a shift of paradigm is necessary to become aware, that both, CMMI and Scrum, are nothing more than abstract maps of reality [44], created by human minds. The fact that they were created autonomously and existed for long time independently creates categories in the minds of developers. One task for developers in the near future is to escape from those constricting patterns of thought, in order to increase the own creativity and create new integrative paradigms. Just as in software development, the development of the human mind is an iterative process, resulting in breaking out of mental cages that are no longer necessary.

Future Work for Researchers

The major part of the integration evaluation grounds on the theoretical analysis of Scrum project management practices in the context of CMMI project management requirements. The theoretical analysis brings clear results that comply with many empirical evidences on Maturity Levels two and three and at least some first evidence at higher Maturity Levels.

Nevertheless, further Research on the CMMI Process Areas that were not included in this survey is necessary to ensure that *Support, Engineering,* and *Process Management* CMMI Areas are not contradictory to the Scrum processes.

5 Bibliography

[1] M.B. Chrissis, M. Konrad, S. Shrum. CMMI for Development (3rd Edition). Boston, MA: Addison-Wesley, 2011, Print.

[2] M.K. Kulpa, K.A.Johnson. Interpreting the CMMI (2nd Edition). Auerbach Publications, 2008, Print.

[3] P. E. McMahon. Integrating CMMI and Agile Development. Boston, MA: Addison-Wesley, 2010, Print.

[4] Succeeding with Agile, Addison Wesley

[5] Agile Testing, Addison Wesley

[6] D. J. Anderson, “Stretching agile to fit CMMI level 3 - the story of creating MSF for CMMI/spl reg/ process improvement at Microsoft corporation,” in Agile Development Conference (ADC’05), 2005, pp. 193-201.

[7] S. W. Baker, “Formalizing Agility, Part 2: How an Agile Organization Embraced the CMMI,” in AGILE 2006 (AGILE’06), 2006, pp. 147-154.

[8] S. W. Baker, “Formalizing agility: an agile organization’s journey toward CMMI accreditation,” in Agile Development Conference (ADC’05), 2005, pp. 185-192.

[9] B. Boehm, “Value-based software engineering,” ACM SIGSOFT Software Engineering Notes, vol. 28, no. 2, p. 3, Mar. 2003.

[10] S. Cohan and H. Glazer, “An Agile Development Team’s Quest for CMMI® Maturity Level 5,” in 2009 Agile Conference, 2009, pp. 201-206.

[11] M. Cohn, “Agile Estimating and Planning,” North, 2007.

[12] E. Damiani, A. Colombo, F. Frati, and C. Bellettini, A Metamodel for Modeling and Measuring Scrum Development Process, vol. 4536. Springer Berlin Heidelberg, 2007, pp. 74-83.

[13] K. Dinakar, “Agile development: overcoming a short-term focus in implementing best practices,” in Proceeding of the 24th ACM SIGPLAN conference companion on Object oriented programming systems languages and applications - OOPSLA ’09, 2009, pp. 579-588.

[14] I. E. Espinosa-Curiel, J. Rodríguez-Jacobo, and J. A. Fernández-Zepeda, “A competency framework for the stakeholders of a software process improvement initiative,” in Proceeding of the 2nd workshop on Software engineering for sensor network applications - SESENA ’11, 2011, pp. 139-148.

[15] M. Foegen, M. Solbach, and C. Raak, Der Weg zur professionellen IT -Eine praktische Anleitung für das Management von Veränderungen mit CMMI, ITIL oder SPICE. Heidelberg: Springer Verlag, 212AD, p. 2008.

[16] B. Gloger, Scrum - Produkte zuverlässig und schnell entwickeln. München: Carl Hanser Verlag München Wien, 2008, p. 381.

[17] D. R. Greening, “Enterprise Scrum: Scaling Scrum to the Executive Level,” in 2010 43rd Hawaii International Conference on System Sciences, 2010, pp. 1-10.

[18] M. T. Hansen and H. Baggesen, “From CMMI and Isolation to Scrum, Agile, Lean and Collaboration,” in 2009 Agile Conference, 2009, pp. 283-288.

[19] D. Hartmann and R. Dymond, Appropriate Agile Measurement: Using Metrics and Diagnostics to Deliver Business Value. IEEE, 2006, pp. 126-134.

[20] D. Hartmann and R. Dymond, “Using Metrics and Diagnostics to Deliver Business Value,” Business.

[21] E. Hossain, P. L. Bannerman, and R. Jeffery, “Towards an understanding of tailoring scrum in global software development,” in Proceeding of the 2nd workshop on Software engineering for sensor network applications - SESENA ’11, 2011, p. 110.

[22] C. R. Jakobsen and K. A. Johnson, “Mature Agile with a Twist of CMMI,” in Agile 2008 Conference, 2008, pp. 212-217.

[23] C. R. Jakobsen and J. Sutherland, “Scrum and CMMI Going from Good to Great,” in 2009 Agile Conference, 2009, pp. 333-337.

[24] L. Jiang and A. Eberlein, “Towards a framework for understanding the relationships between classical software engineering and agile methodologies,” in Proceedings of the 2008 international workshop on Scrutinizing agile practices or shoot-out at the agile corral, 2008, pp. 9-14.

[25] K. Kautz, H. W. Hansen, and K. Thaysen, “Applying and adjusting a software process improvement model in practice,” in Proceedings of the 22nd international conference on Software engineering - ICSE ’00, 2000, pp. 626-633.

[26] F. Keenan, “Agile process tailoring and problem analysis (APTLY),” in Proceedings. 26th International Conference on Software Engineering, 2004, pp. 45-47.

[27] R. Kneuper, CMMI Verbesserung von Software- und Systementwicklungsprozessen mit Capability Maturity Model Integration (CMMI-DEV), 3rd ed. Heidelberg: dpunkt.verlag., 2007, p. 284.

[28] M. K. Kulpa and K. A. Johnson, Interpreting the CMMI - A Process Improvement Approach. Taylor & Francis, 2008.

[29] R. E. Landaeta, S. Viscardi, and A. Tolk, “Strategic management of scrum projects: An organizational learning perspective,” in First International Technology Management Conference, 2011, pp. 651-656.

[30] M. Lavallée and P. N. Robillard, “Do software process improvements lead to ISO 9126 architectural quality factor improvement,” in Proceedings of the 8th international workshop on Software quality - WoSQ ’11, 2011, p. 11.

[31] R. Leithiser and D. Hamilton, “Agile versus CMMI - process template selection and integration with team foundation server,” in Proceedings of the 46th Annual Southeast Regional Conference on XX - ACM-SE 46, 2008, p. 186.

[32] A. S. C. Marcal, F. S. F. Soares, and A. D. Belchior, “Mapping CMMI Project Management Process Areas to Scrum Practices,” in 31st IEEE Software Engineering Workshop (SEW 2007), 2007, pp. 13-22.

[33] P. E. McMahon, Integrating CMMI and Agile Development - Case Studies and Proven Techniques for Faster Performance Improvement, 2nd ed. Addison-Wesley, 2011, p. 325.

[34] R. Morris, “Agile Estimation and Planning,” October.

[35] A. Omran, “AGILE CMMI from SMEs perspective,” in 2008 3rd International Conference on Information and Communication Technologies: From Theory to Applications, 2008, pp. 1-8.

[36] L. Rising and N. Janoff, “The Scrum Software Development Process for Small Teams,” IEEE Software, vol. 17, no. August, pp. 26-32, 2000.

[37] M. Sharma and M. Chandwani, “Maturing capability in unified paradigm,” in Proceedings of the International Conference on Advances in Computing, Communication and Control - ICAC3 ’09, 2009, p. 737.

[38] S. Shrum, Mi. Konrad, and M. B. Chrissis, CMMI - Richtlinien für Prozess-Integration und Podukt-Verbesserung. München: Addison-Wesley, 2009, p. 813.

[39] M. Staples and A. T. Park, “Two Case Studies on Small Enterprise Motivation and Readiness for CMMI,” in Proceedings of the 11th International Conference on Product Focused Software, 2010, pp. 63-66.

[40] J. Sutherland and I. Altman, “Organizational Transformation with Scrum: How a Venture Capital Group Gets Twice as Much Done with Half the Work,” in 2010 43rd Hawaii International Conference on System Sciences, 2010, pp. 1-9.

[41] J. Sutherland, C. R. Jakobsen, and K. Johnson, “Scrum and CMMI Level 5: The Magic Potion for Code Warriors,” in AGILE 2007 (AGILE 2007), 2007, pp. 272-278.

[42] R. Turner and A. Jain, “Agile Methods - CMMI-SW: Linking The Two Extremes,” in Proceedings of the Second XP Universe and First Agile Universe Conference on Extreme Programming and Agile Methods - XP/Agile Universe 2002, 2002, pp. 153-165.

[43] B. Wong and S. Hasan, “Cultural influences and differences in software process improvement programs,” in Proceedings of the 6th international workshop on Systems development in SOA environments - WoSQ ’08, 2008, pp. 3-10.

[44] SEI (Software Engineering Institute), CMMI® for Development, Version 1.3 CMMI-DEV, V1.3, no. November. 2010.

[45] P. Deemer, G. Benefield, and C. Larman, The Scrum Primer. 2010.

[46] K. Schwaber and J. Sutherland, “The Scrum Guide,” Scrum Alliance, vol. 19, no. 6, pp. 1-21, 2009.

[47] T. Demarco and B. Boehm, “The Agile Methods Fray,” Computer, vol. 35, no. 6, pp. 90-92, 2002

[48] H. Glazer, D. Anderson, D. J. Anderson, M. Konrad, and S. Shrum, CMMI ® or Agile?: Why Not Embrace Both?!, no. November. Carnegie Mellon University, Software Engineering Institute, 2008, p. 48.

[49] I. Nonaka and H. Takeuchi, The Knowledge Creating Company. Oxford University Press, 1995, p. 284.

[50] M. Fowler and J. Highsmith, “The Agile Manifesto,” Software Development, vol. 9, no. August, pp. 28–35, 2001.

[51] W. E. Deming, Out of the Crisis, vol. 1, no. 2. MIT Press, 2000, p. 507.

[52] DeGrace, Peter; Stahl, Leslie Hulet (1990-10-01). Wicked problems, righteous solutions. Prentice Hall. ISBN 978-0-135-90126-7.

[53] “Standard CMMI Appraisal Method for Process Improvement (SCAMPI) A , Version 1 . 3 : Method Definition Document Standard CMMI ® Appraisal Method for Process Improvement (SCAMPI SM) A ,” *Management*, 2011.

[54] J. Diaz, J. Garbajosa, and J. A. Calvo-Manzano, *Mapping CMMI Level 2 to Scrum Practices: An Experience Report*, vol. 42, no. 42. Springer Berlin Heidelberg, 2009, pp. 93-104.