Agile Quality Management with CMMI-DEV and Scrum

Authors

Andreas Schliep
SPRiNT iT Gesellschaft für Lösungen mbH
andreas.schliep@sprint-it.com

Andreas Schliep is a Certified Scrum Practitioner. He works as a passionate Scrum Coach and retrospective facilitator for SPRiNT iT, Ettlingen/Germany. He has 8 years of leadership experience as a team and group leader. His development experience is based on projects in the field of video conferencing, VoIP and instant messaging, focusing on C++, Java, databases and network protocols.

Boris Gloger
SPRiNT iT Gesellschaft für Lösungen mbH
boris.gloger@sprint-it.com

Boris Gloger has been implementing Scrum since 2003. He became a Certified Scrum Trainer in 2004. He organized the first Scrum Gathering in Vienna in 2004 and the Retrospective Facilitator Gathering 2006, in Baden-Baden. Boris Gloger leads Scrum implementations and is a SPRiNT-iT partner.

Abstract

Agile processes are becoming increasingly popular as they promise frequent deliveries of working code. Traditional Quality Management methods appear to be old fashioned and not suitable for this new era of self-organizing teams, emergent systems and empirical project management. Engineering practices like Test Driven Development propose high quality code without the need of extensive test phases between coding and release.

Yet Quality Management is a field of high expertise and specialization just like system development or database administration. Its practices and measures have been developed to cope with a variety of failure reasons and conditions. Some of them have been incorporated into internationally accepted standards and best practices collections like ISO 9001, Capability Maturity Model Integrated, or ISO 15504 (Software Process Improvement and Capability Determination / SPICE.

There are many opportunities being explored today. More and more companies start using agile processes to cope with stagnant projects and missed deadlines. Many of them are asking about a way to keep or gain a high organizational maturity level in this new situation. Some already managed to implement process frameworks like Scrum in organizations on CMMI Level 3 and higher. We will demonstrate that Scrum fits well into a corporate Quality Management strategy based on systematic process improvement.

Keywords

Agile, Quality Management, Process Improvement, Scrum, XP, CMMI

Copyrights

Contents

Agile Quality Management with CMMI-DEV and Scrum 1

Contents 2

1 Introduction 3

2 Quality Management and Agile Methods 3

3 Scenario: Scrum Enterprise Transition Project 4

3.1 Staging 5

3.2 The Process Framework: Scrum and XP for Software Development 6

3.3 Enterprise Scrum Implementation and Appraisal Roadmap 7

4 Scrum CMMI-DEV Process Area Coverage 8

4.1 Generic Goals 8

4.2 Process Areas 9

5 Conclusion 10

6 Bibliography 11

Appendix A: Exemplary Mapping of CMMI-DEV 1.2 Practices to Scrum/XP 12

6.1 CMMI-DEV 1.2 Maturity Level 2 Process Areas 12

6.1.1 Measurement and Analysis (MA) 12

6.1.2 Project Monitoring and Control (PMC) 13

6.1.3 Project Planning (PP) 14

6.1.4 Process and Product Quality Assurance (PPQA) 15

6.1.5 Requirements Management (REQM) 16

6.1.6 Supplier Agreement Management (SAM) 17

6.2 CMMI-DEV 1.2 Maturity Level 3 Process Areas 18

6.2.1 Integrated Project Management +IPPD (IPM+IPPD) 18

6.2.2 Product Integration (PI) 19

6.2.3 Requirements Development (RD) 20

6.2.4 Risk Management (RSKM) 21

6.2.5 Technical Solution (TS) 22

6.2.6 Validation (VAL) 23

6.2.7 Verification (VER) 23

1 Introduction

In God we trust – all others bring data.
W. Edwards Deming

Agile process management methodologies are often considered too immature and not suitable to comply with process improvement and quality management goals. However, we and other Scrum (1) trainers and coaches have worked with a couple of Fortune 500 companies. They have started using Scrum and related processes to improve the quality of their products and the reliability of their deliveries.

We have discussed the top issues of project management with process improvement experts, QA engineers, testers and consultants. They all identified the same problem areas:

- Requirements
- Communication
- Resources
- Quality

Late completion of the requirements specification, late involvement of quality assurance and the poor execution of quality control contribute to the difficulties in these areas. Agile approaches promise to handle the different problem areas through a new set of values, principles and roles. They promise more flexibility and customer interaction. In this scenario requirements should emerge together with the system in the course of the project. Teams are given room to organize themselves and improve both their process and practices.

Nevertheless, we have experienced so called agile implementations that have resulted in more trouble and quality issues than working code. After we performed retrospectives with the affected teams, we found out that they shared a huge misconception about the implementation of agile methods. They focused on the delivery of features and neglected the necessary practices to ensure the quality of the output. They did not understand that agility is not an excuse for a lack of discipline. What was missing?

In fact, we discovered a lack of knowledge about quality management basics and elementary engineering practices in small companies – actually the opposite of the misconceptions about agile we found in organizations familiar with heavy-weight processes. So we decided to look for approaches that would help both sides to understand the power of an agile mindset together with a core understanding of process improvement. We took a closer look at Capability Maturity Management Integrated for Development (CMMI-DEV) (3) and Software Process Improvement and Capability Determination (SPICE) (5). This led us to rediscover quality management and capability and maturity models, to see them in the light of the agile goals. And we were surprised by their compatibility. The Scrum process is powerful enough to comply with most of the goals and practices stated in CMMI-DEV.

2 Quality Management and Agile Methods

Quality is such an attractive banner that sometimes we think
we can get away with just waving it, without doing the hard work necessary to achieve it.
Miles Maguire

The international quality management core standard ISO 9001:2000 (4) states 8 quality management principles:

1) Customer Satisfaction

2) Leadership

3) Involvement of People

4) Process Approach

5) System Approach to Management

6) Continual Improvement

7) Factual Approach to Decision Making

8) Mutually Beneficial Supplier Relationships

Capability Maturity Model™ Integration for Development (CMMI-DEV) (3) and the exemplary process assessment model ISO-15504-5 (1) –referred to as Software Process Improvement and Capability Determination (SPICE) - were designed with these intentions in mind. A quality management (QM) department can compare the current process implementations with the goals and practices in the model. Shortcomings can be identified, appropriate measures planned, executed and reviewed. The best practice collection would give suggestions on how the processes could be enhanced in order to achieve a higher capability level.

The presence of best practices in CMMI™ and ISO 15504-5 helps internal and independent appraisers or assessors to get an overview about a particular organization's process landscape. Yet they might distract the observer from the goals behind them. This is the main reason for the difficulties CMMI™ appraisers or SPICE assessors are facing when it comes to agile processes.

- ISO 15504-5 is based on the process reference model (PRM) ISO 12207 – which describes a number of practices and work products that might not be present in agile processes. SPICE assessments allow for no separation between goals and practices: "*Evidence of performance of the base practices, and the presence of work products with their expected work product characteristics, provide objective evidence of the achievement of the purpose of the process.*" (1)
- CMMI appears to be more flexible than SPICE regarding the fulfillment of specific practices. The practices listed can be interpreted as *expectations*, not prescriptions : "*Before goals can be considered satisfied, either the practices as described, or acceptable alternatives to them, are present in the planned and implemented processes of the organization.*" (3)

While it is certainly not impossible to figure out an agile SPICE implementation, we have heard about the difficulties. Luckily, we do not need a new model for Scrum and XP to be able to incorporate it into our quality management strategy with CMMI. We found out that we basically have to take a closer look at the goals. Are they fulfilled by Scrum? How are they fulfilled? A mapping of Scrum practices to CMMI practices can provide a deeper understanding – while there still might be gaps to fill.

Agility and quality management are no contradictions but necessary companions to improve an organization's processes and lead it towards higher maturity. We want to demonstrate this through an example based on the experiences with some of the companies we've worked with recently.

3 Scenario: Scrum Enterprise Transition Project

A journey of thousand miles begins with a single step.
Confucius

A Web portal company - SAMPLE Co. - wants to speed up their development processes without suffering quality tradeoffs. SAMPLE Co. has grown from a dozen to several hundred employees in just a few years. The software development process is phase oriented, yet there are numerous shortcuts and workarounds to deal with the growing quality problems. SAMPLE Co. wants to change this situation.

The development managers pick Scrum as a promising process framework. But how can they determine whether the new processes are an improvement compared to their current situation? The Capability Maturity Model Integrated CMMI-DEV 1.2 offers a catalog of capability goals that could be held against the results of the Scrum implementation. The challenge is to figure out how it is possible to implement Scrum in a controlled way. CMMI appraisers should be able to evaluate the actual process capability levels.

3.1 Staging

The initial project phase consists of an Agile Readiness Assessment and a subsequent Scrum pilot implementation. A couple of software teams at SAMPLE Co. are introduced into the Scrum methodology. External Scrum Masters work with business people and developers to gather experiences with the new approach. Appraisers trained in the Standard CMMI Appraisal Method for Process Improvement (SCAMPI) are brought into the project.

The Rollout Plan for Agile Software Process Improvement is demonstrated at SAMPLE Co. Based on dialogues with the SCAMPI appraisers, an enterprise transition project roadmap with combined CMMI and Scrum approaches is produced.

This project resembles the Scrum process model for enterprise environments (9). It is based on Scrum with the addition of processes to meet special organizational needs. Based on the model and following the process, a project roadmap and a project structure are built.

The chart (Agile Transition Project Structure) describes the setup of the process improvement steering, the process implementation team and the scrum / development teams.

The **Process Improvement Steering Committee** is responsible to guide the process implementation team by providing a clear vision and the support through the organization.

The **Process Implementation Team** is responsible to implement the change. It will be led by a change manager. The change manager will organize the change throughout the organization by setting up the process implementation team. He/she will develop the overall project plan in alignment with the process improvement steering. The process implementation team will guide the work of the project teams.

The **Development Scrum Teams** will work according to the standard process that the process implementation team will develop - based on the experiences of the development teams. They will have the responsibility to give feedback to the process implementation team in order to guide the development of standard process.

The overall implementation approach will be to run this project as a Scrum project. The project leader of this project is supposed to have experience with Scrum in a CMMI context or vice versa.

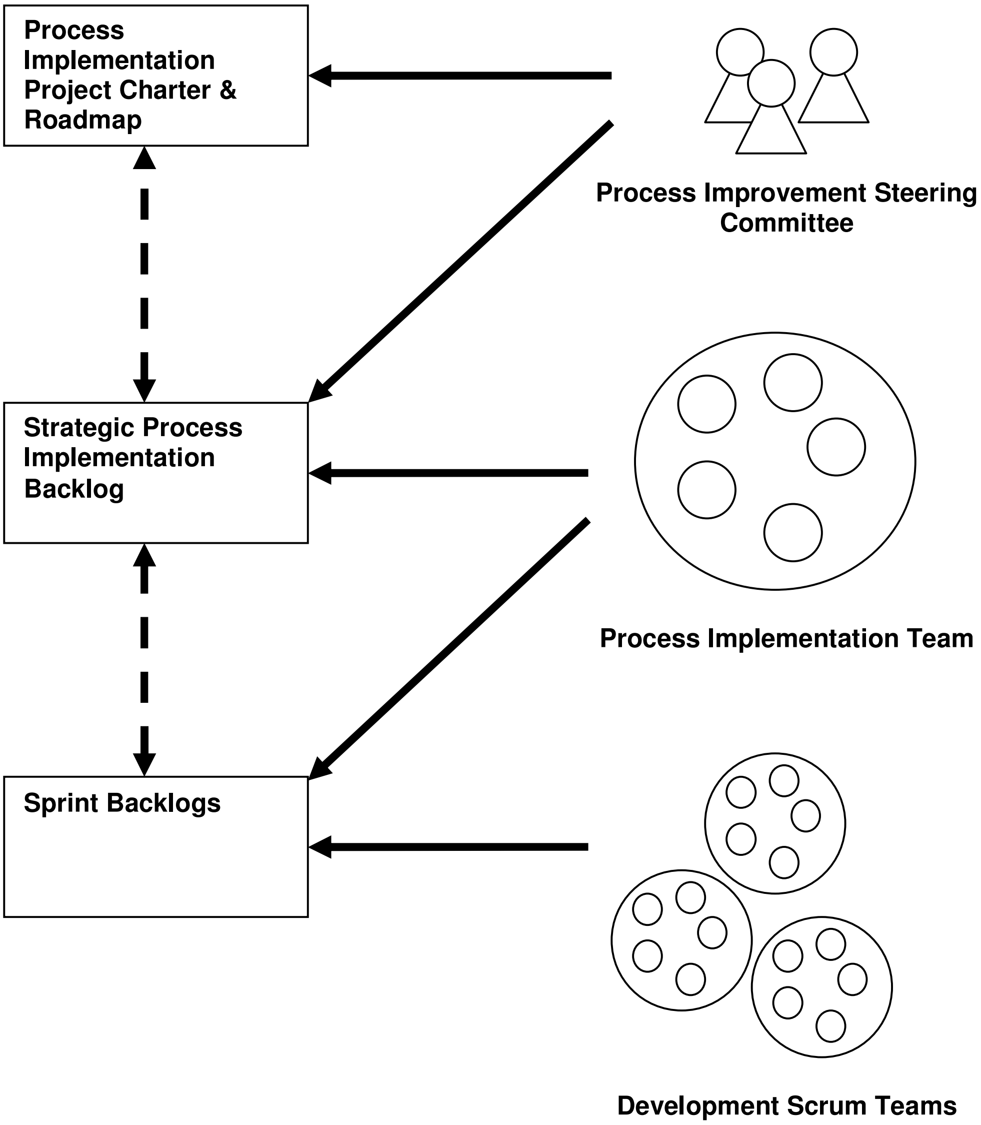

Process Implementation Project Charter & Roadmap
Strategic Process Implementation Backlog
Sprint Backlogs
Process Improvement Steering Committee
Process Implementation Team
Development Scrum Teams

Figure 1 : Agile Transition Project Structure

3.2 The Process Framework: Scrum and XP for Software Development

The main two practices I introduce to a new team are Continuous Integration and Testing. And Testing.
Bas Vodde, Certified Scrum Trainer

Unlike heavyweight process methodologies, the Scrum framework itself does not mention any specific engineering practices. It deals with the management of projects, not with the particular steps of development or production. Unlike eXtreme Programming (9), Scrum embraces any practice that is suitable to reach the goals of the project. This circumstance might lead to the misconception that the explicit introduction of engineering and quality management practices is not necessary. The lack of suitable engineering practices – whether agile or traditional – is one of the major problems in new Scrum implementations.

So, how is it possible for an organization to comply to CMMI if it uses Scrum? Most of the current CMM / CMMI Level 2 or higher appraised organizations using Scrum have accomplished their goal with the integration of XP engineering practices. In fact, Scrum *and* XP – sometimes called XP@Scrum™ – is a powerful and widely used combination.

The XP engineering practices are an agile interpretation of commonly accepted best practices for software engineering. There are several versions of the set available. We will refer to the most widely accepted version of the XP core practices as published in the 2nd edition of "eXtreme Programming Explained" (9).

Category / XP Practice	Explicit Coverage by Scrum Practice	Process Improvement
Fine Scale Feedback Test		Retrospectives
Pair Programming		
Planning Game	Release and Sprint planning	
Test Driven Development		
Whole Team	Cross-functional teams	
Continuous Process		
Continuous Integration		
Design Improvement		
Small Releases	Fixed length Sprints	
Shared Understanding		
Coding Standard		
Collective Code Ownership		
Simple Design		
System Metaphor		
Programmer Welfare		
Sustainable Pace	Team Estimates	

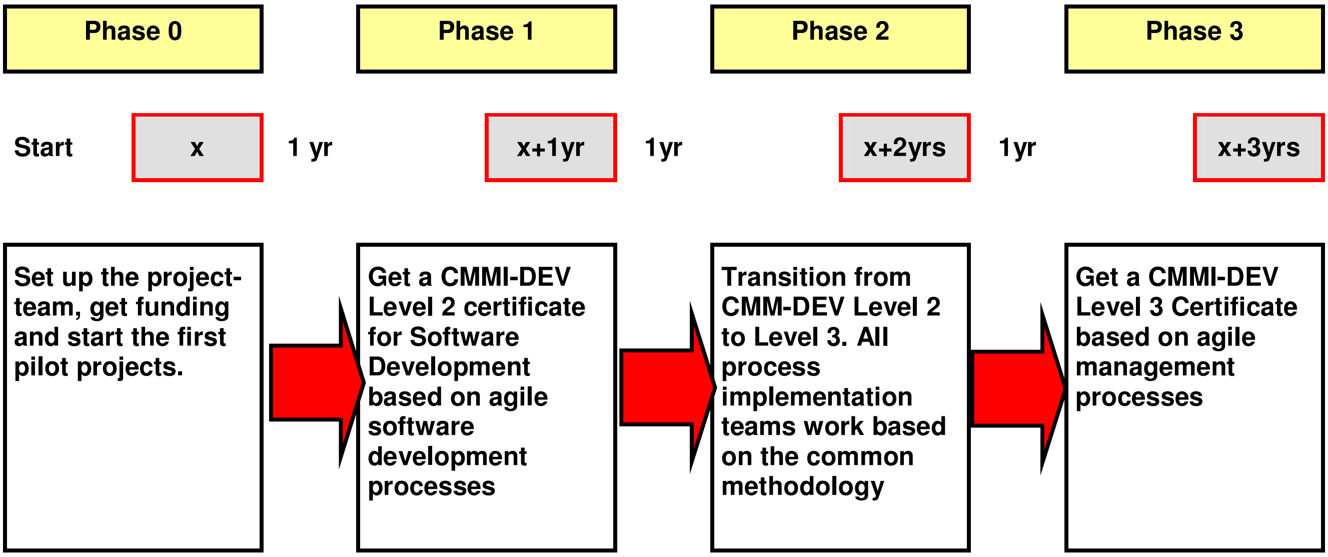
Figure 2 : XP Practice Coverage by Scrum

While other applicable engineering practices probably fit into the Scrum framework, the software engineering practices of XP completely comply with the values and principles of Scrum.

3.3 Enterprise Scrum Implementation and Appraisal Roadmap

Scrum implementation activities and SCAMPI appraisals are brought together in a common roadmap. Ideally, it should be possible to reach CMMI-DEV Maturity Level 3 with Scrum in about three years, possibly even less – provided the entire company is not too resistant to follow the new ways. This approach could be considerably faster than typical CMMI implementations that take roughly two years per level.

The following chart illustrates such a high level combined roadmap with initial steps.



Phase 0

Phase 1

Phase 2

Phase 3

Start x 1 yr x+1yr 1yr x+2yrs 1yr x+3yrs

Set up the project-team, get funding and start the first pilot projects.

Get a CMMI-DEV Level 2 certificate for Software Development based on agile software development processes

Transition from CMM-DEV Level 2 to Level 3. All process implementation teams work based on the common methodology

Get a CMMI-DEV Level 3 Certificate based on agile management processes

Figure 3 : Scrum and CMMI Implementation Roadmap (Extract)

The roadmap contains two SCAMPI appraisals. It should be possible to perform the first SCAMPI appraisal after the first year of the Scrum implementation. The organization should be able to reach Maturity Level 2 with Scrum then. A second appraisal is planned for the end of the third year. The transition from CMMI-DEV Level 2 to Level 3 is considerably harder than the transition up to Level 2. We have to deal with more process areas that are not explicitly covered by Scrum/XP. The next section - Scrum CMMI-DEV Process Area Coverage - provides an overview of the process areas and their possible coverage. The Appendix A: Exemplary Mapping of CMMI-DEV 1.2 Practices to Scrum/XP – shows a possible detailed mapping of CMMI-DEV practices by Scrum.

4 Scrum CMMI-DEV Process Area Coverage

Scrum is no Silver Bullet
Ken Schwaber

Scrum is suitable for a lot of project management and planning activities. It is focused on the delivery of business value and quality. Through the simplicity of Scrum, it can be introduced very quickly. Yet there are several process areas that need to be treated individually. CMMI-DEV helps to fill the gaps, enabling organizations in transition to define adequate agile process implementations.

4.1 Generic Goals

A process framework that does not meet CMMI-DEV Generic Goals will surely not be able to withstand an examination regarding its compliance to the Specific Goals (SG) and Practices (SP) within the different process areas. Yet a Scrum process implementation is capable of meeting even the Optimizing Level through its inspect and adapt cycles. Scrum and custom metrics can be introduced to manage the process quantitatively over the time and identify shortcomings in the process implementation. Scrum project outputs can be measured in a couple of ways. One actual measure for process performance could be the acceptance percentage – the ratio of deliverables that passed the acceptance criteria versus the team commitment. Scrum incorporates empirical process adjustment, using frequent retrospectives to adopt lacking practices and identifying issues that block the particular team. The effects of any adjustment to the process implementation can be compared to previous adjustments.

4.2 Process Areas

The Scrum process framework is suitable for almost any activity related to creative work, including software development, system design and organizational development. However, there are some process areas of CMMI-DEV Version 1.2 which are not explicitly covered by Scrum. The following table helps to identify those areas, assuming a certain interpretation of Scrum Enterprise implementation practices in addition to the Scrum Framework. CMMI-DEV 1.2 process areas are listed with their groups and maturity levels. The Scrum Framework Coverage indicates how much Scrum as a defined process framework can be applied to the specific process area.

The meanings of the coverage attributes for Scrum are:

- A *defined* coverage means that the guidelines about how to apply Scrum to reach the goals of a specific process area are explicitly defined in Scrum.
- An *applicable* coverage indicates that Scrum can be used to introduce and manage practices which are not explicitly defined in Scrum.
- A *partly applicable* coverage is assumed for process areas where Scrum can be used to cover part of the required goals.
- *Demanded by Scrum* means that the practices of this process area are a necessary prerequisite for Scrum processes.

CMMI-DEV Process Area	CMMI-DEV Group	CMMI Maturity Level	Scrum Framework Coverage
Configuration Management (CM)	Support	2	demanded by Scrum
Measurement and Analysis (MA)	Support	2	defined
Process and Product Quality Assurance (PPQA)	Support	2	applicable
Project Monitoring and Control (PMC)	Project Management	2	defined
Project Planning (PP)	Project Management	2	defined
Requirements Management (REQM)	Engineering	2	defined
Supplier Agreement Management (SAM)	Project Management	2	partly applicable

Decision Analysis and Resolution (DAR)	Support	3	partly applicable
Integrated Project Management +IPPD (IPM+IPPD)[1]	Project Management	3	defined
Organizational Process Definition +IPPD (OPD+IPPD)[6]	Process Management	3	applicable - see (9)
Organizational Process Focus (OPF)	Process Management	3	partly applicable
Organizational Training (OT)	Process Management	3	required by Scrum
Product Integration (PI)	Engineering	3	required by Scrum
Requirements Development (RD)	Engineering	3	applicable
Risk Management (RSKM)	Project Management	3	applicable
Technical Solution (TS)	Engineering	3	defined – combined with XP Engineering Practices (9)
Validation (VAL)	Engineering	3	applicable
Verification (VER)	Engineering	3	applicable
Organizational Process Performance (OPP)	Process Management	4	defined
Quantitative Project Management (QPM)	Project Management	4	applicable
Causal Analysis and Resolution (CAR)	Support	5	partly applicable
Organizational Innovation and Deployment (OID)	Process Management	5	partly applicable

Figure 4 : Applicability of Scrum for CMMI-DEV 1.2 Process Areas

For a closer look at any particular process area, see Appendix A.

5 Conclusion

I have made this letter longer than usual, only because I have not had the time to make it shorter.
Blaise Pascal

CMMI-DEV is a very good means to determine whether the new process implementations really brought improvements – measured in a higher capability or maturity. It is possible to cover most of the CMMI-DEV 1.2 process areas with processes based on the Scrum framework. CMMI-DEV Appraisals for Scrum companies might be easier or harder than typical appraisals. They are definitely different.

Moreover, the agile approach can help organizations to quickly find a path out of their daily chaos, without a big portion of the massive overhead brought by heavy-weight processes. Scrum is a suitable method for process improvement itself. Any enterprise can take the challenge and reengineer their process landscape, using a process implementation backlog and a Scrum process.

[1] This process area has "+IPPD" after its name because it contains a goal and practices that are specific to IPPD. The material specific to IPPD is called an "IPPD addition." All process areas with IPPD additions have "+IPPD" after their name.

Scrum /XP Benefits for CMMI Implementations	CMMI Benefits for Scrum Implementations
Focus on important values	Helps to produce a complete agile practice set for ENG
Focus on team	Address organizational learning and support
Result driven	Address leadership
Business value driven	Institutionalization
Explicit promotion of values for every single participant	Address organizational support

Figure 5 : Mutual Benefits for CMMI-DEV and Scrum

We must admit that the described approach to achieve a sound quality management through the transition to mature agile processes appears to be quite ambitious and challenging. Whether an organization wants to prove or improve their maturity using an agile approach in combination with CMMI – they will need to be ready for the dialogue and courageous enough to face the challenges. Scrum is very suitable for the creation of a communicative, creative and constructive environment.

The mapping tables in Appendix A are meant to work as examples for possible process introduction or appraisal considerations.

6 Bibliography

(1) Agile Alliance Website, http://www.agilealliance.org

(2) Certifying for CMMI Level 2 and ISO 9001 with XP@Scrum, http://ieeexplore.ieee.org/iel5/8714/27587/01231461.pdf?arnumber=1231461

(3) CMMI® for Development, Version 1.2, CMU/SEI-2006-TR-008, 2006

(4) ISO/IEC 9001:2000 Quality Management Systems – Requirements, International Standardization Organization, ISO/IEC 9001:2000, Edition 3, May 2005

(5) ISO/IEC 15504-5 Information technology — Process Assessment Part 5: An exemplar Process Assessment Model, International Standardization Organization, ISO/IEC 15504-5:2006(E), March 2006

(6) Jeffrey Liker, The Toyota Way: Fourteen Management Principles from the World's Greatest Manufacturer, McGraw-Hill Companies, ISBN 0071392319, January 2004

(7) Dr Jeff Sutherland, Is CMMI worth doing?, http://jeffsutherland.com/scrum/2006/11/is-cmmi-worth-doing.html

(8) Ken Schwaber and Mike Beedle, Agile Software Development with Scrum, Prentice Hall, ISBN 0130676349, October 2001

(9) Ken Schwaber, The Enterprise and Scrum, Microsoft Press Corp., ISBN 9780735623378, June 2007

(10) Kent Beck, Extreme Programming Explained: Embrace Change (2nd edition), Addison-Wesley, ISBN 0201616416, October 1999

(11) Vadivelu Vinayagamurthy, Challenges in assessing Agile Development projects for CMMI Level 3 based on SCAMPI methodology, Proceedings of the International SPICE Days 2007 in Frankfurt/Main, Germany, ISQI, June 2007

Appendix A: Exemplary Mapping of CMMI-DEV 1.2 Practices to Scrum/XP

Essentially, all models are wrong, but some are useful.

George Box

The mappings include all suitable CMMI-DEV 1.2 process areas for Maturity Level 3 and some of the process areas of Maturity Level 3. The purpose of these mappings is not to define a fixed rule about the application of Scrum for CMMI. They are rather intended to work as an orientation point.

We assume that a Scrum/XP process definition is available throughout the organization. The references (1) are supplemented by organization specific additions.

We used these abbreviations in the mapping tables:

- SG : CMMI-DEV 1.2 Specific Goal
- SP : CMMI-DEV 1.2 Specific Practice

6.1 CMMI-DEV 1.2 Maturity Level 2 Process Areas

Not explicitly covered:

- Configuration Management

6.1.1 Measurement and Analysis (MA)

The purpose of Measurement and Analysis (MA) is to develop and sustain a measurement capability that is used to support management information needs.

CMMI Goals and Practices	**Scrum/XP fulfills this practice if you...**
Measurement objectives and activities are aligned with identified information needs and objectives. SG 1	
Establish and maintain measurement objectives that are derived from identified information needs and objectives. SP 1.1	Identify useful statistics for your information needs, i.e. Product and Sprint Backlog statistics for project management purposes.
Specify measures to address the measurement objectives. SP 1.2	Ensure that the measures are known to all and followed in the process.
Specify how measurement data will be obtained and stored. SP 1.3	Add any measurement data to the information in the Product Backlog and Scrum reports.
Specify how measurement data will be analyzed and reported. SP 1.4	Specify in the dept handbook, that Backlog Analysis and statistics on Total Effort, Team Velocity, Defect Count and any other required data are performed and maintained.
Measurement results that address identified information needs and objectives are provided. SG 2	

Obtain specified measurement data. SP 2.1	Add any other specified information need to the Product Backlog.
Analyze and interpret measurement data. SP 2.2	Add measurement interpretation tasks to the Product Backlog.
Manage and store measurement data, measurement specifications, and analysis results. SP 2.3	Specify the management and storage of data, specifications and results in the Product Backlog.
Report results of measurement and analysis activities to all relevant stakeholders. SP 2.4	Ensure that the Product Backlog and any analysis are visible to everybody.

6.1.2 Project Monitoring and Control (PMC)

The purpose of Project Monitoring and Control (PMC) is to provide an understanding of the project's progress so that appropriate corrective actions can be taken when the project's performance deviates significantly from the plan.

CMMI Goals and Practices	**Scrum/XP fulfills this practice if you...**
Actual performance and progress of the project are monitored against the project plan. SG 1	
Monitor the actual values of the project planning parameters against the project plan. SP 1.1	Use Sprint-, Release- and Product Backlog Burndown Charts to illustrate the progress of the development against scheduled Review-, Release- or Completion dates.
Monitor commitments against those identified in the project plan. SP 1.2	Keep the original Release Backlog and compare it to the Selected Backlogs for each Sprint.
Monitor risks against those identified in the project plan. SP 1.3	Make the progress of risk evaluation as put into relation to highest level Backlog Items visible. An Impediment Backlog is maintained to make additional risks transparent.
Monitor the management of project data against the project plan. SP 1.4	As Scrum demands, integrate and document everything each Sprint, including project data.
Monitor stakeholder involvement against the project plan. SP 1.5	Record each Backlog Item with its originator. Thus any changes in the project can be documented with their origin.
Periodically review the project's progress, performance, and issues. SP 1.6	Use Scrum reporting. The progress is made visible. The team meets daily to exchange the status. Product Owners perform their own Daily Scrum to keep track of the project.
Review the accomplishments and results of the project at selected project milestones. SP 1.7	Perform a Sprint Review after each Sprint in the project. The progress is made visible. A retrospective is performed to gather additional impediments or ways to improve.

Corrective actions are managed to closure when the project's performance or results deviate significantly from the plan. SG 2	
Collect and analyze the issues and determine the corrective actions necessary to address the issues. SP 2.1	Use an Impediment Backlog to collect team related and other issues. The team is responsible to identify organizational issues which are communicated by the Scrum Master.
Take corrective action on identified issues. SP 2.2	Ensure that any Impediment is prioritized and worked on by team and management.
Manage corrective actions to closure. SP 2.3	Ensure that any issue or impediment is only removed from the Impediment Backlog when it is completely resolved.

6.1.3 Project Planning (PP)

The purpose of Project Planning (PP) is to establish and maintain plans that define project activities.

CMMI Goals and Practices	**Scrum/XP fulfills this practice if you...**
Estimates of project planning parameters are established and maintained. SG 1	
Establish a top-level work breakdown structure (WBS) to estimate the scope of the project. SP 1.1	Categorize Product Backlog Items. Equivalent to a WBS, the Product Backlog can be grouped by features and teams. (There is no task level planning in Scrum before the Sprint Planning)
Establish and maintain estimates of the attributes of the work products and tasks. SP 1.2	Perform Estimation Meetings. Backlog Items can be estimated using relative sizes, called Story Points. While Story Points have no absolute meaning, they indicate the relative complexity of each Backlog Item.
Define the project life-cycle phases upon which to scope the planning effort. SP 1.3	Order features in the Product Backlog by their business importance. The top of the Product Backlog is estimated in detail, the remainder more roughly.
Estimate the project effort and cost for the work products and tasks based on estimation rationale. SP 1.4	Summarize the Product Backlog estimates for a given release and compare it to the team velocity to calculate the project effort.
A project plan is established and maintained as the basis for managing the project. SG 2	
Establish and maintain the project's budget and schedule. SP 2.1	Ensure that the Product Owner establishes and maintains the budget and schedule for the project.
Identify and analyze project risks. SP 2.2	Analyze the risk of Backlog Items. Some of the most risky Backlog Items are prioritized very high so they can be evaluated in the

	beginning of the project.
Plan for the management of project data. SP 2.3	Store the Product Backlog in a central place which is accessible for all project participants and stakeholders. Each comment, task or test associated to a Backlog Item is made visible.
Plan for necessary resources to perform the project. SP 2.4	Allocate Scrum teams based on the project budget and provide them with their working environment. The team may determine to extend the working environment or add team members itself constrained by the given budget.
Plan for knowledge and skills needed to perform the project. SP 2.5	Start the project with a cross functional team which includes all necessary skills. Scaling mechanisms will take care of the spreading of knowledge. Teams may acquire additional training at any time.
Plan the involvement of identified stakeholders. SP 2.6	Invite stakeholders to participate in the planning and review sessions. They are allowed to watch the Daily Scrum meetings.
Establish and maintain the overall project plan content. SP 2.7	Use the Product Backlog as mentioned above.
Commitments to the project plan are established and maintained. SG 3	
Review all plans that affect the project to understand project commitments. SP 3.1	List any deliverables and constraints on the Product Backlog that possibly affect the work of any team.
Reconcile the project plan to reflect available and estimated resources. SP 3.2	Adjust the estimations and prioritizations on the Product Backlog on demand. The Team Velocity is used to calculate the validity of planning.
Obtain commitment from relevant stakeholders responsible for performing and supporting plan execution. SP 3.3	Ensure that the budget has to be approved by stakeholders in order to start any Sprint.

6.1.4 Process and Product Quality Assurance (PPQA)

The purpose of Process and Product Quality Assurance (PPQA) is to provide staff and management with objective insight into processes and associated work products.

CMMI Goals and Practices	**Scrum/XP fulfills this practice if you...**
Adherence of the performed process and associated work products and services to applicable process descriptions, standards, and procedures is objectively evaluated. SG 1	
Objectively evaluate the designated performed processes against the applicable process	Ensure that the team sticks to the Scrum Rules. The ScrumMaster is responsible to

descriptions, standards, and procedures. SP 1.1	confront the team with deviations from the original Scrum process or other agreements.
Objectively evaluate the designated work products and services against the applicable process descriptions, standards, and procedures. SP 1.2	Ensure that the acceptance criteria for each work product are added to the Backlog Item description, that acceptance tests are performed and their results documented.
Noncompliance issues are objectively tracked and communicated, and resolution is ensured. SG 2	
Communicate quality issues and ensure resolution of noncompliance issues with the staff and managers. SP 2.1	Ensure the objectivity of the ScrumMaster (by being a QA staff member; for instance), Be honest and transparent in the Sprint Review meetings. Perform retrospectives to address the reasons behind quality issues and resolve them using the Impediment Backlog.
Establish and maintain records of the quality assurance activities. SP 2.2	Verify the completion of Backlog Items on base of their test fulfillment. An incomplete Backlog Item can only be removed from the Product Backlog if the according requirement is no longer needed in the Sprint, release or product.

6.1.5 Requirements Management (REQM)

The purpose of Requirements Management (REQM) is to manage the requirements of the project's products and product components and to identify inconsistencies between those requirements and the project's plans and work products.

CMMI Goals and Practices	**Scrum/XP fulfills this practice if you...**
Requirements are managed and inconsistencies with project plans and work products are identified. SG 1	
Develop an understanding with the requirements providers on the meaning of the requirements. SP 1.1	Discuss Vision, Goals and Product Backlog Items in the Release and Sprint Planning meetings.
Obtain commitment to the requirements from the project participants. SP 1.2	Ensure that Product Owner and Scrum Team agree at least about the stability of the Selected Product Backlog for the duration of a Sprint. Only the Product Owner is entitled to prioritize the remaining Product Backlog.
Manage changes to the requirements as they evolve during the project. SP 1.3	Add Backlog Items to the Product Backlog that is not currently selected for a Sprint. Priorities, size or effort estimates, Business Value of the Product Backlog may be adjusted.
Maintain bidirectional traceability among the requirements and the project plans and work products. SP 1.4	Make the Product Backlog visible for everyone. Each Backlog Item contains a full list of references to other information sources and vice versa. A Sprint Backlog contains

	only tasks that can be mapped to Product Backlog Items.
Identify inconsistencies between the project plans and work products and the requirements. SP 1.5	Split the Product Backlog into Sprints and Releases to support the ongoing project planning. The actual work is noted in the Product Backlog after each Sprint. Any changes to Backlog Items are noted in the Product Backlog.

6.1.6 Supplier Agreement Management (SAM)

The purpose of Supplier Agreement Management (SAM) is to manage the acquisition of products from suppliers.

CMMI Goals and Practices	**Scrum/XP fulfills this practice if you...**
Agreements with the suppliers are established and maintained. SG 1	
Determine the type of acquisition for each product or product component to be acquired. SP 1.1	Perform a Backlog Planning. Identify Product Backlog that should be implemented by external suppliers.
Select suppliers based on an evaluation of their ability to meet the specified requirements and established criteria. SP 1.2	Incorporate the supplier evaluation and selection - in the initial Sprints. If necessary, create and refine selection criteria.
Establish and maintain formal agreements with the supplier. SP 1.3	Ensure that the supplier commits to the Selected Product Backlog. Each Sprint will be followed by a review. Tests are added to Backlog Items to provide specific acceptance criteria.
Agreements with the suppliers are satisfied by both the project and the supplier. SG 2	
Perform activities with the supplier as specified in the supplier agreement. SP 2.1	Ensure that Sprint Review and Retrospective are scheduled and performed.
Select, monitor, and analyze processes used by the supplier. SP 2.2	Ensure that the supplier processes fit into the Scrum process. Ideally, the supplier has incorporated agile engineering practices as well.
Select and evaluate work products from the supplier of custom-made products. SP 2.3	Incorporate the evaluation and selection of custom-made products into so called Staging- (or Zero-) Sprints
Ensure that the supplier agreement is satisfied before accepting the acquired product. SP 2.4	Accept any Backlog Item only if it fulfills the defined acceptance criteria, including tests.
Transition the acquired products from the supplier to the project. SP 2.4	Establish Continuous Integration to ensure the smooth transition. Integration work is included into the Product Backlog

6.2 CMMI-DEV 1.2 Maturity Level 3 Process Areas

Not explicitly covered:

- Decision Analysis and Resolution (DAR)
- Organizational Process Definition +IPPD (OPD+IPPD)
- Organizational Process Focus (OPF)
- Organizational Training (OT)

6.2.1 Integrated Project Management +IPPD (IPM+IPPD)

The purpose of Integrated Project Management (IPM) is to establish and manage the project and the involvement of the relevant stakeholders according to an integrated and defined process that is tailored from the organization's set of standard processes.

For IPPD, Integrated Project Management +IPPD also covers the establishment of a shared vision for the project and the establishment of integrated teams that will carry out objectives of the project.

CMMI Goals and Practices	**Scrum/XP fulfills this practice if you...**
The project is conducted using a defined process that is tailored from the organization's set of standard processes. SG 1	
Establish and maintain the project's defined process from project startup through the life of the project. SP 1.1	Select a main ScrumMaster. The ScrumMaster holds the responsibility to maintain the process.
Use the organizational process assets and measurement repository for estimating and planning the project's activities. SP 1.2	Incorporate Scrum into the corporate process framework portfolio.
Establish and maintain the project's work environment based on the organization's work environment standards. SP 1.3	Establish the project work environment in an initial Sprint and improve it as required.
Integrate the project plan and the other plans that affect the project to describe the project's defined process. SP 1.4	Ensure that the Product Backlog contains the entire plan. References to the process description can be added.
Manage the project using the project plan, the other plans that affect the project, and the project's defined process. SP 1.5	Enable the ScrumMaster to ensure that all participants follow the plan according to the Scrum process.
Contribute work products, measures, and documented experiences to the organizational process assets. SP 1.6	Maintain a Scrum process knowledge base. The ScrumMaster is responsible for the contribution of all necessary artifacts, retrospective results or conventions.
Coordination and collaboration of the project with relevant stakeholders is conducted. SG 2	
Manage the involvement of the relevant stakeholders in the project. SP 2.1	Ensure that stakeholder involvement is well defined for the team. The Product Owner takes the responsibility to regard the stakeholders' needs in the Product Backlog
Participate with relevant stakeholders to identify,	Let multiple Product Owners form a Product

negotiate, and track critical dependencies. SP 2.2	Owner team that interfaces with other stakeholders.
Resolve issues with relevant stakeholders. SP 2.3	Ensure that ScrumMaster and Product Owner work together to remove impediments.
The project is managed using IPPD principles. SG 3	
Establish and maintain a shared vision for the project. SP 3.1	Create and communicate the shared vision as the heart of each Scrum project.
Establish and maintain the integrated team structure for the project. SP 3.2	Build Cross-functional teams. They are not to be changed – at least for the duration of one Sprint.
Allocate requirements, responsibilities, tasks, and interfaces to teams in the integrated team structure. SP 3.3	Perform a scaled planning (9). Allocate Backlog Items to suitable teams.
Establish and maintain integrated teams in the structure. SP 3.4	Ensure that everyone required to deliver the product increment is in the particular team.
Ensure collaboration among interfacing teams. SP 3.5	Establish Scrum of Scrums meetings, where team members exchange information.

6.2.2 Product Integration (PI)

The purpose of Product Integration (PI) is to assemble the product from the product components, ensure that the product, as integrated, functions properly, and deliver the product.

CMMI Goals and Practices	**Scrum/XP fulfills this practice if you...**
Preparation for product integration is conducted. SG 1	
Determine the product-component integration sequence. SP 1.1	Let the team determine and implement the product-component integration sequence.
Establish and maintain the environment needed to support the integration of the product components. SP 1.2	Let the team set up and maintain a continuous integration environment.
Establish and maintain procedures and criteria for integration of the product components. SP 1.3	Define integration and system tests early in order to provide common criteria for component integration.
The product-component interfaces, both internal and external, are compatible. SG 2	
Review interface descriptions for coverage and completeness. SP 2.1	Review Interfaces reviewed through their integration. Completeness of the description is not necessary. At the contrary, an interface is considered complete as it fulfills the necessary integration tests.
Manage internal and external interface definitions, designs, and changes for products and product components. SP 2.2	Perform Continuous Integration which ensures that changes at the interface level do not break the system.

Verified product components are assembled and the integrated, verified, and validated product is delivered. SG 3	
Confirm, prior to assembly, that each product component required to assemble the product has been properly identified, functions according to its description, and that the product component interfaces comply with the interface descriptions. SP 3.1	Perform integration tests on various levels to ensure component compliance.
Assemble product components according to the product integration sequence and available procedures. SP 3.2	Ensure that the assembly is being conducted automatically.
Evaluate assembled product components for interface compatibility. SP 3.3	Perform the required tests in the course of Continuous Integration.
Package the assembled product or product component and deliver it to the appropriate customer. SP 3.4	Let the team establish deployment routines. Ideally, the Continuous Integration environment performs the packaging and delivery. Each Sprint delivers a potentially shippable product increment.

6.2.3 Requirements Development (RD)

The purpose of Requirements Development (RD) is to produce and analyze customer, product, and product component requirements.

CMMI Goals and Practices	**Scrum/XP fulfills this practice if you...**
Stakeholder needs, expectations, constraints, and interfaces are collected and translated into customer requirements. SG 1	
Elicit stakeholder needs, expectations, constraints, and interfaces for all phases of the product life cycle. SP 1.1	Add input from stakeholders to the Product Backlog at any time. Only the Selected Product Backlog must not be changed.
Transform stakeholder needs, expectations, constraints, and interfaces into customer requirements. SP 1.2	Discuss requirements and transform them into Backlog Items / Stories and tests.
Customer requirements are refined and elaborated to develop product and product-component requirements. SG 2	
Establish and maintain product and product-component requirements, which are based on the customer requirements. SP 2.1	Break down or detail Backlog Items to be small, measurable, achievable, relevant and timed.
Allocate the requirements for each product component. SP 2.2	Refine the Product Backlog into feature groups or teams if multiple teams have to work on multiple product components.
Identify interface requirements. SP 2.3	Let the team identify interface requirements.

	They are documented in the code.
The requirements are analyzed and validated, and a definition of required functionality is developed. SG 3	
Establish and maintain operational concepts and associated scenarios. SP 3.1	Collect operational concepts and associated scenarios. They are maintained by the Product Owner and communicated to the team in the Sprint Planning meeting.
Establish and maintain a definition of required functionality. SP 3.2	Prioritize the Product Backlog by business value. Required functionality is always visible.
Analyze requirements to ensure that they are necessary and sufficient. SP 3.3	Create a shared understanding in the Planning meetings. Ensure that the Product Backlog is maintained well.
Analyze requirements to balance stakeholder needs and constraints. SP 3.4	Use the Scrum planning, review and reporting techniques. The transparency of Scrum uncovers any conflicts between needs and constraints. The Sprint Review meeting is used to demonstrate the reached state and gather insight for the next step.
Validate requirements to ensure the resulting product will perform as intended in the user's environment using multiple techniques as appropriate. SP 3.5	Perform Frequent Deliveries and Reviews to ensure that the product meets its requirements.

6.2.4 Risk Management (RSKM)

The purpose of Risk Management (RSKM) is to identify potential problems before they occur so that risk-handling activities can be planned and invoked as needed across the life of the product or project to mitigate adverse impacts on achieving objectives.

CMMI Goals and Practices	**Scrum/XP fulfills this practice if you...**
Preparation for risk management is conducted. SG 1	
Determine risk sources and categories. SP 1.1	Add the project risk source and category determination to the Product Backlog. Add corporate risk source and category determination to the Company Backlog.
Define the parameters used to analyze and categorize risks, and the parameters used to control the risk management effort. SP 1.2	Add the project risk parameter definition to the Product Backlog. Add corporate risk parameter definition to the Company Backlog.
Establish and maintain the strategy to be used for risk management. SP 1.3	Use the Product Backlog to manage known project risks and the Impediment Backlog to manage risks with organizational impact.
Risks are identified and analyzed to determine their relative importance. SG 2	
Identify and document the risks. SP 2.1	Add the risk to the Product or Impediment Backlog.
Evaluate and categorize each identified risk using the defined risk categories and parameters, and	Evaluate the risk and prioritize it according to

determine its relative priority. SP 2.2	its impact.
Risks are handled and mitigated, where appropriate, to reduce adverse impacts on achieving objectives. SG 3	
Develop a risk mitigation plan for the most important risks to the project as defined by the risk management strategy. SP 3.1	Put the most important risks on top of the Product Backlog.
Monitor the status of each risk periodically and implement the risk mitigation plan as appropriate. SP 3.2	Perform Sprint Reviews and retrospectives and include the mitigation of risks.

6.2.5 Technical Solution (TS)

The purpose of Technical Solution (TS) is to design, develop, and implement solutions to requirements. Solutions, designs, and implementations encompass products, product components, and product-related lifecycle processes either singly or in combination as appropriate.

CMMI Goals and Practices	**Scrum/XP fulfills this practice if you...**
Product or product-component solutions are selected from alternative solutions. SG 1	
Develop alternative solutions and selection criteria. SP 1.1	Use initial Sprints to select appropriate basic approaches. The team always demonstrates the best solution from its perspective at the next Sprint Review.
Select the product-component solutions that best satisfy the criteria established. SP 1.2	Use initial and subsequent Sprints to narrow on the best solutions.
Product or product-component designs are developed. SG 2	
Develop a design for the product or product component. SP 2.1	Establish a Simple Design that fits to the current needs.
Establish and maintain a technical data package. SP 2.2	Define Coding Standards to define technical packages.
Design product component interfaces using established criteria. SP 2.3	Use the Coding Standards practice to decide upon interface criteria.
Evaluate whether the product components should be developed, purchased, or reused based on established criteria. SP 2.4	Ensure that decisions about this are made by the development team together with their Product Owner
Product components, and associated support documentation, are implemented from their designs. SG 3	
Implement the designs of the product components. SP 3.1	Acknowledge that Analysis, Design and Implementation are no longer separate phases.
Develop and maintain the end-use documentation.	Ensure that every deliverable has to be

SP 3.2	complete, including documentation.

6.2.6 Validation (VAL)

The purpose of Validation (VAL) is to demonstrate that a product or product component fulfills its intended use when placed in its intended environment.

CMMI Goals and Practices	**Scrum/XP fulfills this practice if you...**
Preparation for validation is conducted. SG 1	
Select products and product components to be validated and the validation methods that will be used for each. SP 1.1	Let the Product Owner determine the selection in the Product Backlog. The methods are determined and implemented by the team.
Establish and maintain the environment needed to support validation. SP 1.2	Ensure that the team is responsible for creating the acceptance environment.
Establish and maintain procedures and criteria for validation. SP 1.3	The team designs Acceptance methods to validate the functionality delivered at the Sprint Review.
The product or product components are validated to ensure that they are suitable for use in their intended operating environment. SG 2	
Perform validation on the selected products and product components. SP 2.1	Perform Acceptance tests.
Analyze the results of the validation activities and identify issues. SP 2.2	Compare the actual results of the acceptance tests to the expected results.

6.2.7 Verification (VER)

The purpose of Verification (VER) is to ensure that selected work products meet their specified requirements.

CMMI Goals and Practices	**Scrum/XP fulfills this practice if you...**
Preparation for verification is conducted. SG 1	
Select the work products to be verified and the verification methods that will be used for each. SP 1.1	Define appropriate integration tests for each component and performed before and at the Sprint Review to verify the delivered functionality.
Establish and maintain the environment needed to support verification. SP 1.2	Ensure that the test environment is an integral part of the Continuous Build environment.
Establish and maintain verification procedures and criteria for the selected work products. SP 1.3	Collect test criteria to complete the Backlog Item descriptions.
Peer reviews are performed on selected work	

products. SG 2	
Prepare for peer reviews of selected work products. SP 2.1	Establish frequent peer reviews via Pair Programming and / or as separate sessions.
Conduct peer reviews on selected work products and identify issues resulting from the peer review. SP 2.2	Perform Code Reviews in addition.
Analyze data about preparation, conduct, and results of the peer reviews. SP 2.3	Let the team define and keep bug counts, hint data and other useful information in order to measure the success of collaborative reviews.
Selected work products are verified against their specified requirements. SG 3	
Perform verification on the selected work products. SP 3.1	Verify separate work products using Unit Tests or other appropriate measures.
Analyze the results of all verification activities. SP 3.2	Communicate test results immediately in order to identify and perform corrective action as soon as possible.