



Select Computing
I N C O R P O R A T E D

CMMI and Agile: Apply Both for Project Success

Presented to:

CMS Emerging Edge Forum (CEEF)

Date: June 14, 2017

Select Computing, Inc.

9841 Broken Land Parkway
Suite 209
Columbia, MD 21046

www.selectcomputing.com

© 2017, Select Computing, Inc. All rights reserved.

CMMI and Agile: Apply Both for Project Success

Background

A common concern with Agile software development at the enterprise level is that it does not scale well. There are different approaches to scaling Agile methods to address enterprise concerns. Many of the concepts such as using small, self-managing teams, work well for small projects. However, challenges increase when coordinating multiple teams working on a large-scale, complex project with numerous iterations to deliver releases that provide stakeholder value. The combination of several Agile teams concurrently executing distinct iterations can easily lead to chaos. To address scaling Scrum and Agile for large enterprises, a number of approaches have gained attention, including:

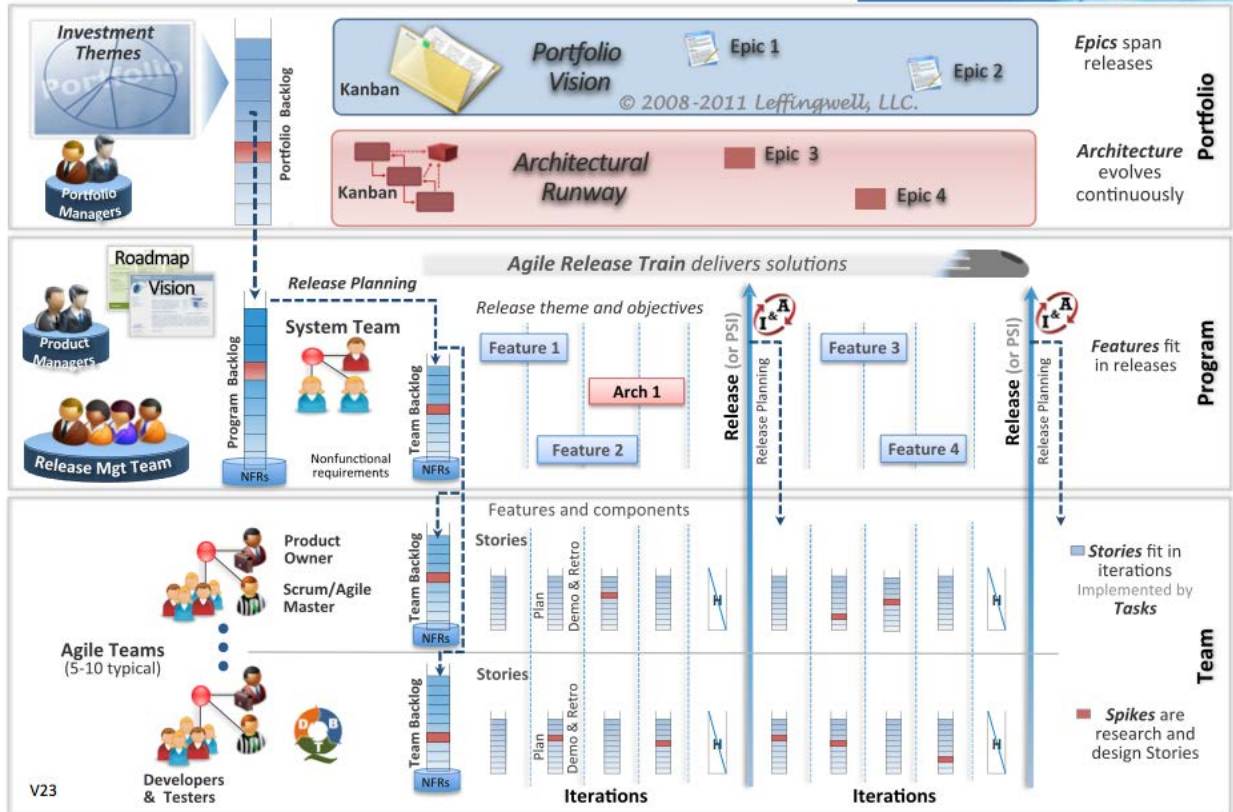
- Scaled Agile Framework (SAFe) (<http://scaledagileframework.com/>) by Dean Leffingwell;
- Disciplined Agile framework (DAD) (<http://www.disciplinedagiledelivery.com/start-here/>) by Scott Ambler; and
- Large Scale Scrum framework (LeSS) (<http://static1.1.sqspcdn.com/static/f/702523/22609354/1367558447003/201305-Larman.pdf>) by Craig Larman and Bas Vodde.

SAFe is an interactive knowledge base for implementing Agile practices at enterprise scale. It has become recognized by its “Big Picture” graphic (see **Figure 1**); SAFe provides a model for enterprise agility. The big picture addresses the enterprise at three levels: Team, Program, and Portfolio.

SAFe utilizes Scrum at the team level and scales Agile and Lean across teams at the program and portfolio management level. Portfolio management helps drive Epics from enterprise investment strategies. Program management coordinates team activities to enact shared business direction and architectural vision, determine related groups of work items for cross-team dependencies and coordinate with external team representatives.

Scaled Agile Framework™ Big Picture

Leffingwell, LLC.



See also www.scalingsoftwareagility.wordpress.com and Leffingwell, D. *Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise*, Addison-Wesley (Pub. 2011)

Figure 1. Scaled Agile Framework Big Picture™ (Leffingwell, LLC)

The DAD process framework is a people-first, learning-oriented hybrid Agile approach to IT solution delivery (see **Figure 2**). It has a risk-value life cycle, is goal-driven, and facilitates enterprise awareness.

The top four priorities in DAD are: (1) People first, (2) Learning-oriented, (3) Agile, and (4) Hybrid. Hybrid means that DAD also draws on other, more traditional sources, especially the versions of Unified Process for governance and life-cycle management. Projects are divided into three phases: Inception, Construction, and Transition.

DAD scales Agile within a delivery team to address inception and transition concerns. The inception phase helps prepare the project backlog needed to guide the project’s iterations. The transition phase includes additional testing and other activities required to move a release into production.

For continuous delivery, the inception and transition phases of DAD may be short and potentially largely automated.

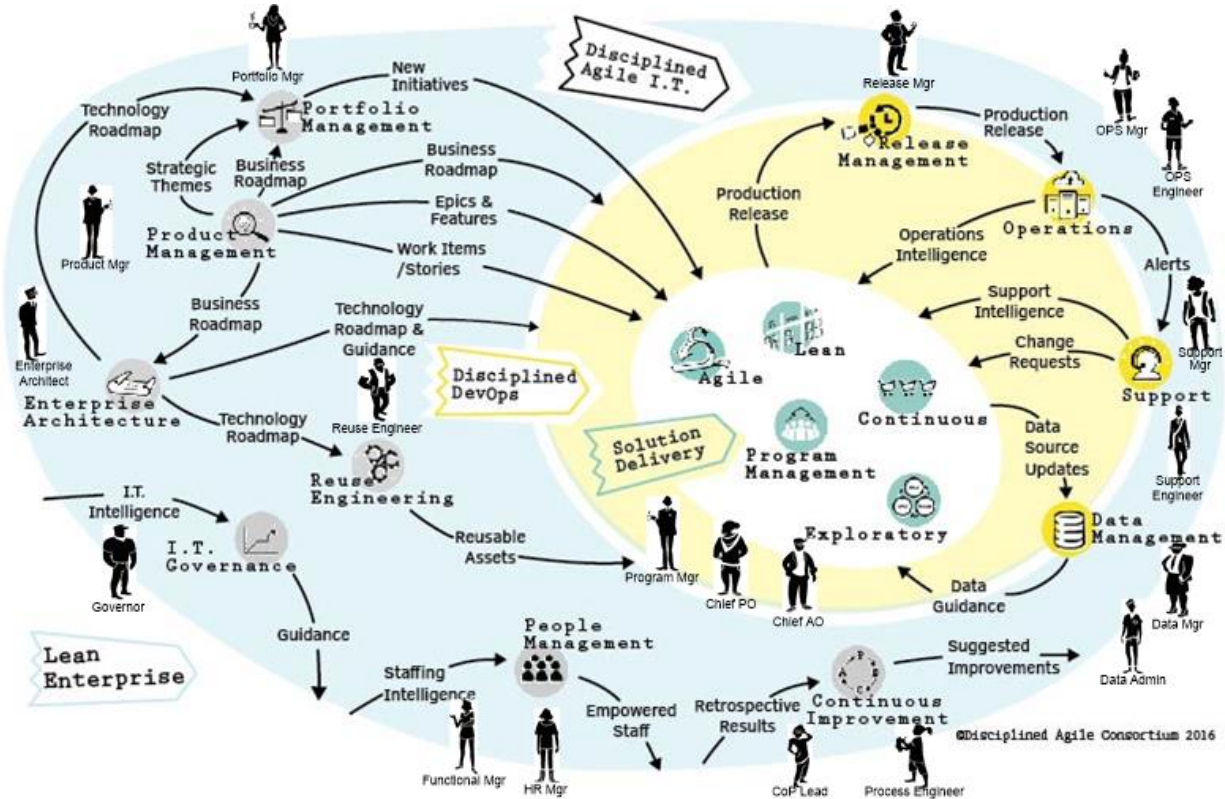


Figure 2. Disciplined Agile Delivery (Disciplined Agile Consortium 2016)

The LeSS framework (see Figure 3) scales Scrum from one team to several teams for large projects (<http://less.works/>). It consists of two process frameworks: Framework-1 and Framework-2. Framework-1 is for projects with 10 or fewer concurrent Agile teams while Framework-2 is for larger than 10.

LeSS adopts Scrum while maintaining the same ceremonies and roles. For example, in Framework-1, the Sprint Planning ceremony involves a representative from each team instead of all team members. In Framework-2, a new role, the Area Product Owner, allows for several product owners to perform this role.

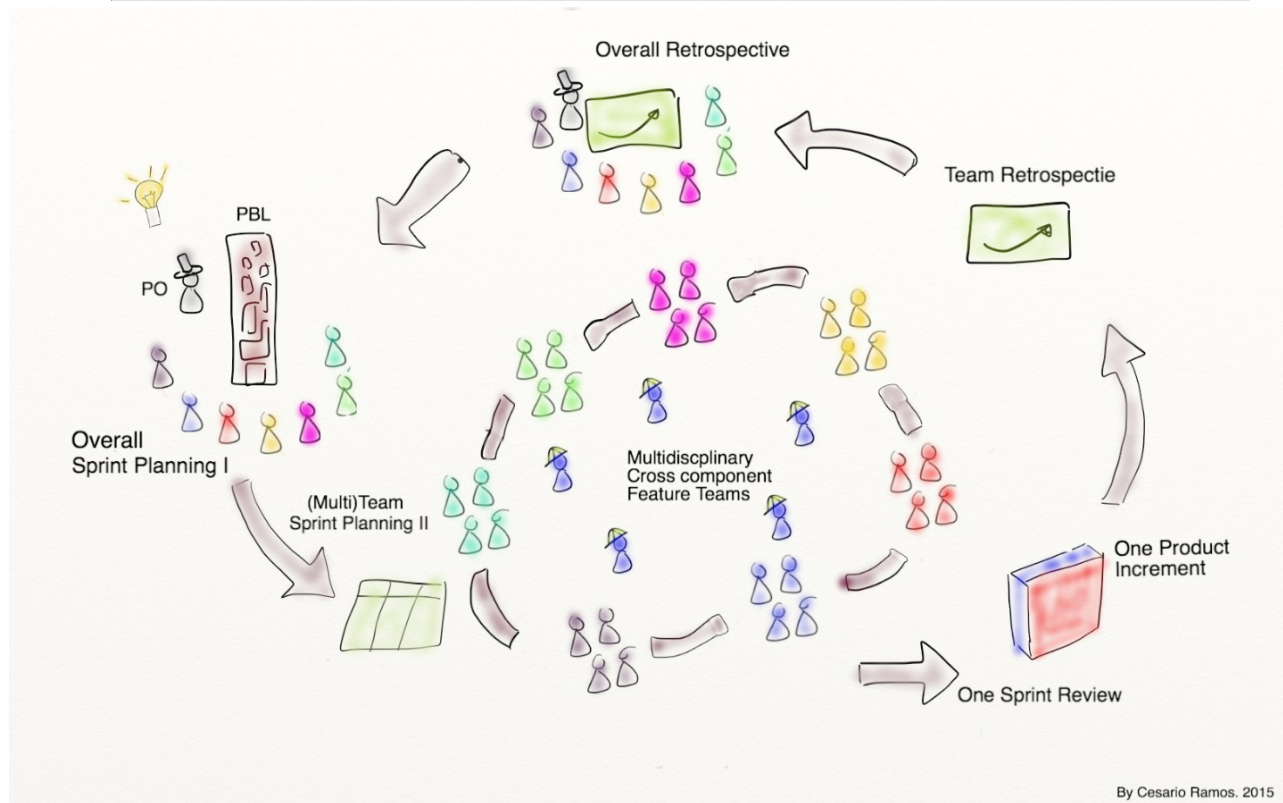


Figure 3. LeSS Framework (Cesario Ramos 2015)

The Challenge

The three leading frameworks, SAFe, DAD, and LeSS, seek to scale one of the most popular Agile methodologies, Scrum, to enable it for large projects. Scrum and Agile methodologies have increased the effectiveness of software development teams. However, Agile methodologies often have to be augmented by other measures to scale up properly to large projects, where an organization has up to 100 or more software developers, analysts, and testers. Team size increases communication and organizational risks on Agile delivery teams. All of the Agile frameworks require high maturity processes to control the complexities of working with a large program.

CMMI-Agile Approach

The **Capability Maturity Model Integration (CMMI)** is a process improvement approach that provides organizations with the essential elements of effective processes. It can be used to guide process improvement across a project, a division, or an entire organization. CMMI helps integrate traditionally

separate organizational functions, set process improvement goals and priorities, guide quality processes, and provide a point of reference for appraising current processes.¹

Both Agile and CMMI are based on the Deming (Plan-Do-Check [or Study]-Act) Cycle and therefore share several values. In the simplest terms, Agile methodologies specify **HOW** things should be done while CMMI specifies **WHAT** should be done². The apparent conflict between CMMI and Agile stems from Agile's focus on deliverables that contribute directly to the product. Deliverables that do not contribute directly to the product are suspected waste. The best perspective from which to address the apparent conflict is risk management. Projects and programs should strive to deliver the product in a manner that minimizes risk to the project, program, and organization.

Managing risk at the organizational level is critical and more difficult when different components use different Agile scaling frameworks. CMMI provides the answer through quantitative project management.

The Information Systems Audit and Control Association (ISACA) defines the Risk IT management framework as shown in **Figure 4**³.

The focus is on managing risks to the business (organizational) objectives. The Risk IT Principles state that effective enterprise governance and management of IT risk:

- Always connects to business objectives;
- Aligns the management of IT-related business risk with overall enterprise risk management (ERM);
- Balances the costs and benefits of managing IT risk;
- Promotes fair and open communication of IT risk;
- Establishes the right tone from the top while defining and enforcing personal accountability for operating within acceptable and well-defined tolerance levels; and
- Is a continuous process and part of daily activities.

¹ VIZTEAMSTM, "Top Reasons Why Agile and CMMI Work Better Together," 2013, <http://www.vizteams.com/blog/top-reasons-agile-and-cmmi-better-together/> (accessed May 10, 2017).

² "CMMI or Agile: Why Not Embrace Both!" 2008, <http://www.sei.cmu.edu/reports/08tn003.pdf> (accessed May 10, 2017).

³ ISACA, "Risk IT Framework for Management of IT Related Business Risks," <http://www.isaca.org/knowledge-center/risk-it-it-risk-management/pages/default.aspx> (accessed May 10, 2017).

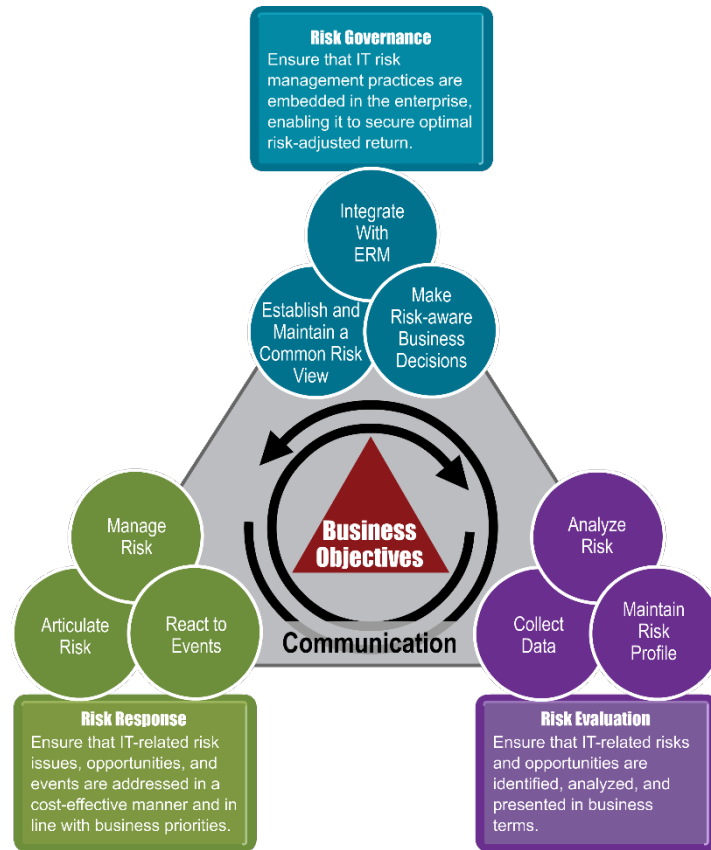


Figure 4. Risk IT Framework for Management of IT Related Business Risks (ISACA)

Below are the key CMMI process areas that appear to be waste but enable effective risk management (the CMMI maturity level, *ML*, is in parentheses):

Quantitative Project Management (QPM) (ML 4). Statistically demonstrate that the project will achieve its established quality and performance objectives. How can we believe the velocity metric of a given project? How comparable is it to the velocity of other projects? All projects operate with the constraints of the organization. Each organization has a normal velocity range. Burndown and velocity reflect process performance of the particular team. However, they do not reflect activity cost performance. Mitigating project risks requires that we compare project performance against the norms of the organization to determine whether the project is on track. Earned value is quite useful. Even for Agile projects, the questions remain: *Has the value the project has produced so far commensurate with the costs to date? What will be the relationship between costs and value in the future?* When implementing Agile at a large scale, managing the risk depends on solid measurements. While the measurements do

not contribute to the end product directly, they enable the organization to predict and detect when the project or program may veer off course.

Organizational Process Performance (OPP) (ML 4). Analyze organization performance and improve to close performance gaps. Use measurements to improve the organization's capability to execute the projects. From an Agile perspective, OPP is an organization-wide retrospective that should be focused on removing process waste. Having an organization-wide retrospective is as useful at the organizational level as it is at the team level. However, its effectiveness is a function of shared language and benchmarks across the organization. The implementation of Quantitative Project Management enables the necessary foundation for process improvement.

Risk Management (ML 3). Identify potential issues that can be prevented or mitigated. Risk identification, classification, and mitigation is built into most Agile methodologies. The focus tends to be on risks emanating from change since the risk of not meeting the requirements is low. Otherwise, there is not a focus on managing risk to organization objectives, such as financial performance for example. Agile risk management is inwardly focused as if the project is outside the context of the organization. Organizational governance is still necessary to protect organizational objectives. This focus on organizational objectives enables ML-4 Organization Process Performance.

Decision Analysis and Resolution (ML 3). Use a formal process for analyzing alternatives where selection criteria are defined and understood. The analysis of alternatives is important in Agile, for example, in estimating story points. In fact, decision making in Agile, by its collaborative nature, is much more complex than in waterfall projects. It is important not to leave the definition of selection criteria to the project team to make sure organizational objectives are represented within the criteria. For example, certain criteria or their definition can be made standard. Beginning at ML-3, the whole organization benefits. Formal criteria in decision making provide the foundation for effective ML-3 Risk Management.

Measurement and Analysis (ML 2). Ensure that sufficient measurement capability exists to make informed management decisions. Crucially, one of these management decisions is the prioritization of the backlog which relies on consistent point valuation of stories, and on the prioritization of the impediments list. This capability is leveraged in ML-3 Decision Analysis and Resolution.

Project Monitoring and Control (ML 2). Understand the actual project progress and correct deviations from objectives and schedules. In Scrum, the Scrum stand-ups and the burndown chart clearly exist for this purpose. The deviations at the project level begin risk identification that rolls up to organizational risk evaluation.

Process and Product Quality Assurance (ML 2). Evaluate processes and associated work products to gain insight in their quality. CMMI’s emphasis is on processes and intermediate work products, more so than on the end product. In Agile, the Scrum Master or the eXtreme Programming (XP) Coach are responsible for ensuring the team follow the methodology and are producing the intermediate deliverables the methodology requires.

When we examine the CMMI process areas on large-scale Agile implementation, we see that CMMI and Agile have much in common, due to their common foundation of the Deming Cycle. From the perspective of the Risk IT framework, we see that ML-2 process areas formalize Risk Response, ML-3 formalizes Risk Evaluation, and ML-4 formalizes Risk Governance. Agile frameworks rely on the Product Owners or Area Owners to reflect and represent Risk Governance within the project or program. As a result, it is critical that Product Owners follow a common Risk Governance approach that pervades throughout projects, programs, and the organization as a whole. By applying CMMI, the organization can manage risk.

Improving Agile Performance with CMMI

Agile and CMMI: Working Together to Help Produce Better Software⁴

While we have focused on Risk Management across the portfolio of a project, the CMMI process areas improve the project team’s ability to execute Agile techniques in response to project challenges. The following table, adapted from CMMIInstitute.com, maps some common business problems where Agile and CMMI provide guidance for improvement. In the first column, the table identifies types of problems organizations face running projects. In the second column, the table identifies how Agile Scrum addresses the problem. The third column cross-references the CMMI process area that addresses the necessary capabilities the organization needs to apply the Scrum techniques effectively. An organization uses the CMMI Process Areas to strengthen the ability to perform the Agile ceremonies/ techniques to ensure scalability and resilience.

Problems	Agile Ceremonies/ Techniques	CMMI Process Areas (Maturity Level)
<i>Project estimates are unrealistic or unknown.</i>	<ul style="list-style-type: none"> ▪ Team Estimating ▪ Planning Poker ▪ Sprint Planning ▪ Backlog Grooming 	<ul style="list-style-type: none"> ▪ Requirements Development ▪ Task Estimation ▪ Release Planning ▪ Sprint Backlog
<i>Projects do not get delivered on schedule.</i>	<ul style="list-style-type: none"> ▪ Daily Standup/ Daily Scrum ▪ Release Burndown ▪ Sprint Burndown 	<ul style="list-style-type: none"> ▪ Project Planning (ML 2) ▪ Integrated Project Management (ML 3) ▪ Requirements Management & Development (ML 2 and ML 3) ▪ Measurement and Analysis (ML 2) ▪ Quantitative Project Management (ML 4)

⁴ “A Guide to Scrum and CMMI®: Improving Agile Performance with CMMI,” 2016 CMMI® Institute, <http://cmmiinstitute.com/cmml-and-agile> (accessed May 10, 2017).

Problems	Agile Ceremonies/ Techniques	CMMI Process Areas (Maturity Level)
Requirements are always changing.	<ul style="list-style-type: none"> Backlog Grooming Sprint Planning Product Backlog User Stories/ Epics Definition of Done (for User Stories) Top “10” Features Release Planning 	<ul style="list-style-type: none"> Requirements Management (ML 2) Project Management & Control (ML 2) Requirements Development (ML 3) Measurement and Analysis (ML 2) Verification (ML 3)
Customer won’t commit to project.	<ul style="list-style-type: none"> Release Planning Sprint Planning 	<ul style="list-style-type: none"> Requirements Management (ML 2) Project Planning (ML 2)
Customers are frustrated with progress.	<ul style="list-style-type: none"> Release Planning Sprint Planning Sprint Demo Sprint Retrospective 	<ul style="list-style-type: none"> Integrated Project Management (ML 3) Validation (ML 3) Project Planning (ML 2) Project Monitoring and Control (ML 2) Measurement and Analysis (ML 2)
Customers are not satisfied with the end product.	<ul style="list-style-type: none"> Definition of Done User Stories/ Epics Test Driven Development Sprint Demo 	<ul style="list-style-type: none"> Validation (ML 3) Verification (ML 3) Requirements Development (ML 3) Requirements Management (ML 2)
There are too many bugs during a project or post project.	<ul style="list-style-type: none"> Test Driver Development Continuous Build/ Continuous Integration Refactoring 	<ul style="list-style-type: none"> Validation (ML 3) Verification (ML 3) Product Integration (ML 3) Technical Solution (ML 3)
Unable to secure or retain project resources.	<ul style="list-style-type: none"> Team Agreements Release Planning Daily Standup Vision 	<ul style="list-style-type: none"> Integrated Project Management (ML 3) Project Planning (ML 2) Risk Management (ML 3) Project Monitoring and Control (ML 2) Organizational Training (ML 3)
Lack of internal and external support for project.	<ul style="list-style-type: none"> Team Agreements Release Planning Sprint Planning Daily Standup Product Owner Meeting with Customer 	<ul style="list-style-type: none"> Integrated Project Management (ML 3) Risk Management (ML 3) Project Planning (ML 2) Project Monitoring and Control (ML 2)
Poor project communication.	<ul style="list-style-type: none"> Daily Standup Release Planning Sprint Planning Sprint Demo Sprint Retrospective Product Owner Meeting with Customer 	<ul style="list-style-type: none"> Project Monitoring and Control (ML 2) Project Planning (ML 2) Integrated Project Management (ML 3)
Requirements are vague or open-ended.	<ul style="list-style-type: none"> User Story/ Epic Definition of Done Backlog Grooming 	<ul style="list-style-type: none"> Requirements Management (ML 2) Requirements Development (ML 3)
Project team responsibilities are not clear.	<ul style="list-style-type: none"> Team Agreements Sprint Planning Release Planning Incremental Release 	<ul style="list-style-type: none"> Integrated Project Planning (ML 2) Project Planning (ML 2)
Project team member training is inadequate for the task.	<ul style="list-style-type: none"> Release Planning 	<ul style="list-style-type: none"> Organizational Training (ML 3) Project Planning (ML 2)
Failure to successfully plan the project.	<ul style="list-style-type: none"> Release Planning Sprint Planning Backlog Grooming 	<ul style="list-style-type: none"> Project Planning (ML 2) Integrated Project Planning (ML 2) Requirements Management (ML 2)
Failure to foresee potential problems.	<ul style="list-style-type: none"> Daily Standup Release Planning Sprint Retrospectives 	<ul style="list-style-type: none"> Project Monitoring and Control (ML 2) Risk Management (ML 3) Integrated Project Management (ML 3)

Problems	Agile Ceremonies/ Techniques	CMMI Process Areas (Maturity Level)
		<ul style="list-style-type: none"> Quantitative Project Management (ML 4)
<i>Project information isn't available when needed.</i>	<ul style="list-style-type: none"> User Stories/ Epics Sprint Planning Release Planning Daily Standup Release Burndown Sprint Burndown Sprint Demo Vision 	<ul style="list-style-type: none"> Project Monitoring (ML 2) Integrated Project Management (ML 3) Risk Management (ML 3) Measurement and Analysis (ML 2) Verification (ML 3)
<i>Code quality is poor.</i>	<ul style="list-style-type: none"> Pair Programming Test Driven Development Continuous Build/ Integration Retrospectives Definition of Done 	<ul style="list-style-type: none"> Technical Solution (ML 3) Verification (ML 3) Validation (ML 3) Integrated Project Management (ML 3) Process & Product Quality Assurance (ML 2)
<i>Organizational performance isn't improving.</i>	<ul style="list-style-type: none"> Retrospectives 	<ul style="list-style-type: none"> Integrated Project Management (ML 3) Organizational Process Focus (ML 3) Organizational Process Definition (ML 3) Organizational Training (ML 3) Process & Product Quality Assurance (ML 2) Organizational Process Performance (ML 4) Quantitative Project Management (ML 4) Causal Analysis and Resolution (ML 5)

Table 1. Common Business Problems Mapped to Agile and CMMI (CMMIInstitute.com)

Conclusion

At **CMMI maturity level 4**, the organization and projects establish quantitative objectives for quality and process performance and use them as criteria for managing projects. Quantitative objectives are based on the needs of the customer, end users, organization, and process implementers. Quality and process performance is understood in statistical terms and is managed throughout the life of projects. A critical distinction between maturity levels 3 and 4 is the predictability of process performance. At maturity level 4, the performance of projects and selected subprocesses is controlled using statistical and other quantitative techniques, and predictions are based, in part, on a statistical analysis of fine-grained process data. ⁵

CMMI level 4 processes can provide the systems engineering practices that help enable an Agile approach on large projects. Regardless of whether an organization chooses to work with SAFe, DAD, LeSS, or another approach, CMMI level 4 can provide a robust, mature framework. CMMI level 4 can provide the process management and support practices that help deploy, sustain, and continuously improve the deployment of an Agile approach in any organization. CMMI level 4 Quantitative Project Management (QPM) processes quantitatively manage a project to achieve the project's established quality and process performance objectives.

⁵ wibas, "Maturity Level 4 Quantitatively Managed (CMMI-DEV)," 2015, <https://www.wibas.com/cmmi/maturity-level-4-quantitatively-managed-cmmi-dev> (accessed May 10, 2017).

Small- and mid-sized companies have demonstrated success using Agile. As discussed previously, many large enterprises are also seeking to adopt the Agile, modern framework, using different approaches to scaling Agile processes, such as SAFe, DAD, and LeSS, to address enterprise concerns. When a contractor is at CMMI level 4, processes are adaptive to the team and environment with a focus on delivering working software. In addition, the standards, process descriptions, and procedures for a project are tailored from the organization's set of standard processes to suit a particular project. At maturity level 4, projects are quantitatively managed. Blending CMMI 4 and Agile to take advantage of both methodologies, will result in delivery of successful software projects.

Using the approach we have described in this paper, the organization is free to apply different Agile scaling frameworks based on the program's circumstances. Different components of the organization applying different scaling frameworks share a common ground. Also, it becomes possible to vary the required CMMI maturity level of projects based on a risk classification. For example, a project with a low classification can require CMMI maturity level 2, medium classification can require level 3, and high classification would require level 4. The important benefit is to match projects to the capabilities of the team while increasing competition. This approach is possible because enterprise governance is operating at maturity level 4.

Alignment and coordination activities, necessary for larger, complex projects, are described in the systems engineering practices found in the Engineering, Risk Management, and Integrated Project Management process areas of CMMI. Therefore, CMMI provides a "safety net" for large projects that helps reduce the risk of something going wrong. An organization seeking to adopt Agile processes to develop high-quality software should strongly consider **CMMI level 4** as a means to achieving excellence.

About Select Computing, Inc.

Select Computing, Inc. (SCi) is an innovative CMMI level 4 appraised company, for both the development and services models. SCi specializes in blending CMMI level 4 process maturity and Agile methodology to produce successful outcomes. SCi can help an organization make the transition to Agile. Implementing Agile, using a contractor that is at CMMI level 4, can result in less rework and provide significant benefits. Implementing a CMMI compliant software development process that is also Agile will bring the repeatability and predictability offered by CMMI.

With direct, practical, and proven experience integrating CMMI level 4 and Agile, SCi brings innovative approaches to our customers and facilitates their initiatives while relying on quantitative management and continuous process improvement to improve efficiency and realize significant cost savings.



Contact Us

Arnold Huff

Sr. Capture/Business Development Manager
Mobile: 301-509-6969 | Office: 410-381-0082 x127
BD@selectcomputing.com

Select Computing, Inc.
9841 Broken Land Pkwy, Suite 209
Columbia, MD. 21046
www.selectcomputing.com