THE PROCESS GROUP

POST

VOL. 20 NO. 1 HELPING YOU IMPROVE YOUR ENGINEERING PROCESS http://www.processgroup.com/newsletter.html April 2013 V3

ADDING PRACTICES TO SCRUM TO ACHIEVE YOUR GOALS (AND COMPARISON WITH CMMI LEVEL 3)

Introduction

Scrum is a project management framework that helps a team plan, manage and track its work*. A framework is a flexible guide that can be used with a variety of work. Scrum provides details for some practices. For example, Scrum tells you how to:

- write a backlog summarizing user requirements
- estimate relative work magnitude using story points
- break work into increments (Sprints) and scope a release
- manage team task assignments
- track progress and issues daily using burndown charts and standup meetings
- elicit retrospectives (lessons) on how to improve

By definition, a framework does not tell you how to do everything. The framework does not, for example, provide details on:

- eliciting, writing and analyzing user and system requirements
- using existing company practices with defined tailoring guidelines
- managing dependencies between other teams in a large project
- using design practices to minimize design errors (such as interface, performance, and data definition issues)
- planning and executing testing activities that are needed beyond the scope of a Sprint and analyzing the results

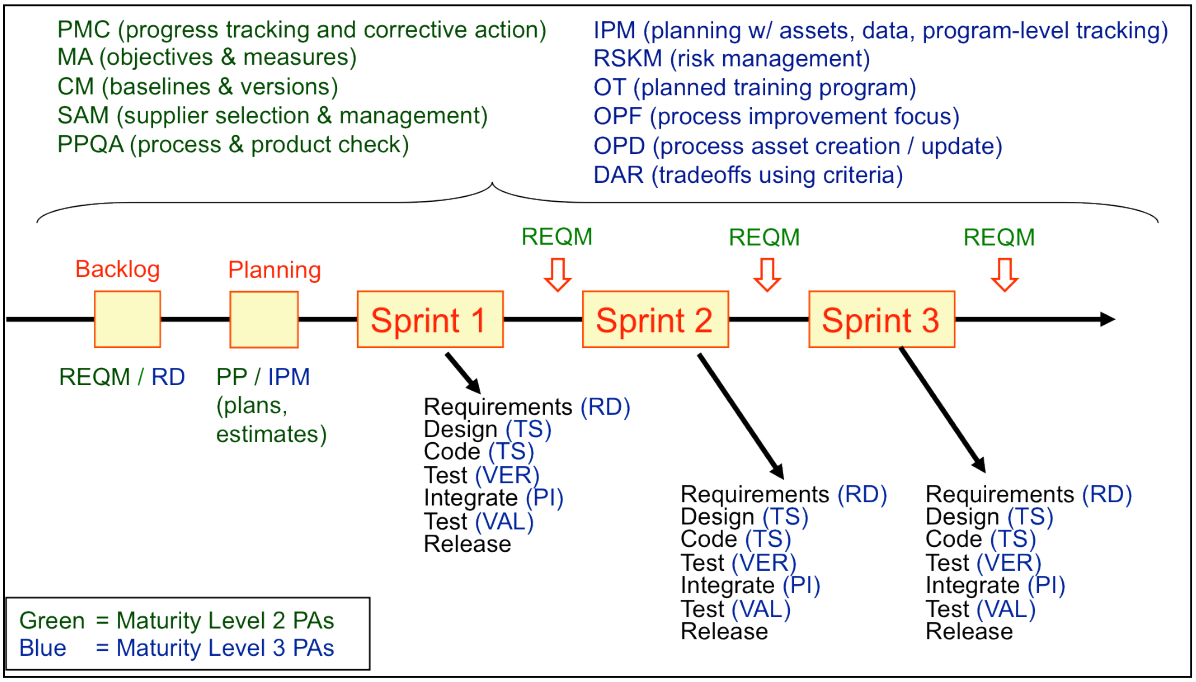

PMC (progress tracking and corrective action)
MA (objectives & measures)
CM (baselines & versions)
SAM (supplier selection & management)
PPQA (process & product check)
IPM (planning w/ assets, data, program-level tracking)
RSKM (risk management)
OT (planned training program)
OPF (process improvement focus)
OPD (process asset creation / update)
DAR (tradeoffs using criteria)
REQM
REQM
REQM
Backlog
Planning
Sprint 1
Sprint 2
Sprint 3
REQM / RD
PP / IPM (plans, estimates)
Requirements (RD)
Design (TS)
Code (TS)
Test (VER)
Integrate (PI)
Test (VAL)
Release
Requirements (RD)
Design (TS)
Code (TS)
Test (VER)
Integrate (PI)
Test (VAL)
Release
Requirements (RD)
Design (TS)
Code (TS)
Test (VER)
Integrate (PI)
Test (VAL)
Release
Green = Maturity Level 2 PAs
Blue = Maturity Level 3 PAs

Figure 1 – Groups of practices overlaid with Scrum (see definitions in ref. [2])

* See Scrum summary on page 2

- collecting, organizing and making available lessons learned and best practices to all projects when they start new work

If you are experiencing requirements, planning, quality or customer problems it might be a good time to add practices to solve these challenges or mitigate the risk of them recurring in the future.

There are many sources available for new practices. It might be from your existing organization or from classes, books and web searches. This article gives examples of adding practices from CMMI.

Adding practices from CMMI Level 2

In a previous article I compared in detail Scrum and CMMI Level 2 [1]. There is a very good match between the two frameworks, Scrum being an example of how to implement approximately half of the practices of Level 2. The remaining practices of Level 2 that don't directly match a Scrum practice can be added to address challenges and risks in those areas (e.g., configuration management, measurement analysis, process assurance and supplier management).

Adding practices from CMMI Level 3

CMMI Level 3 contains project management, engineering and organizational-learning practices that fit well within Scrum (Figure 1). The following sections explain how they can be used. (A list of all the CMMI practices is provided in the references section, item [2].)

Requirements / Backlog

In Scrum, the product owner creates a backlog of user stories, which in their simplest form are one-line statements that describe a user task that a software program is performing. Examples are, “As a user I want to generate a daily report with A, B, C information,” or, “As an operator I want to track all attack aircraft aiming at me within a radius of N miles.”

Although Scrum does not prohibit adding further details to clarify a user story, many teams stop at the one-liner and then discover the details as they write software and perform testing.

Requirements Development (RD) in CMMI Level 3 has three distinct sets of practices to improve backlog creation:

a) elicit requirements
b) develop requirements from the information gathered
c) analyze requirements

Summary of Scrum

Scrum is a process that teams can adopt quickly to plan and manage their work. Each Scrum step has just enough detail to plan, design, build and test code while tracking team progress.

Scrum has three primary roles: Product Owner, Scrum Master, and team member.

The Product Owner communicates the vision of the product to the development team. This includes representing the customer’s interests through requirements and prioritization.

The Scrum Master acts as a liaison between the Product Owner and the team. The Scrum Master does not manage the team but instead works to help the team achieve its Sprint goals by removing obstacles. The Scrum Master verifies that the Scrum process is used.

The team members do the project work. The team typically consists of software engineers, architects, analysts and testers.

The intent of Scrum is to build working components in small iterations, each iteration lasting between one and four weeks. A *typical* Scrum cycle has the following steps:

- Write the user stories (and store in the Backlog)
- Plan the release (which could span more than one 2-4 week Sprint)
- Plan the Sprint
- Conduct the Sprint
  - Analysis
  - Design

    Team-defined tasks
  - Coding
  - Testing
  - Demonstration
- Conduct Sprint retrospective

Daily stand-up meetings are held to track team progress and identify barriers.

These practices improve the skills of a team to gather requirements details from customers and end users, write them down in an organized fashion (which could be an extension to the backlog), and analyze them for errors, ambiguities and feasibility problems. The result is a clearer understanding of the requirements before proceeding.

The practices of RD do not have to be performed at one time or be done perfectly in the beginning. They can be performed to build the initial backlog and then selectively repeated in each sprint as more is learned about the requirements from early sprint demonstrations.

Release Planning, Sprint Planning and Daily Standups

Integrated Project Management (IPM) has distinct sets of practices to plan and manage the project. It includes steps to:

a) plan using company defined best practices and tailoring guidelines
b) use organizational historical data for estimation
c) identify dependencies and stakeholders for coordination, and comprehend this information into a master schedule or overall project plan
d) manage the project with key stakeholders using thresholds to trigger corrective action (such as schedule and effort deviation metrics)

IPM is particularly useful when the project is large and has many dependencies and stakeholders. It provides the structure to make a large project manageable.

Scrum does have three aspects of IPM; the use of historical velocity data that can help a team make better estimates based on previous performance, Scrum-of-Scrums which identifies other teams that need to be coordinated with, and retrospective data to identify lessons learned.

IPM, combined with Level 2 project management practices, provides a more complete set of steps for project coordination and tracking. IPM also includes two practices to take lessons-learned information from a project and make it available for the whole organization.

Risk management

The primary aspect of risk management in Scrum is to assign features with technical risk to earlier sprints so that technical risk can be mitigated early. Technical risk is one source of risk, but not the only source.

The Risk Management (RSKM) process area is used to assess and prioritize all types of risks in a project and develop mitigation actions for the highest priority ones. Example sources of risk are schedule, customers, technology, equipment and suppliers. In RSKM, the project team starts with a standard memory-jogger list derived from previous projects and assesses what could go wrong in the current project. Risk management practices can be easily added to existing planning and tracking activities already performed in Scrum.

Tasks within a sprint

When a team performs sprint planning it develops a task list called a sprint backlog. The content of this list is up to the team. Ideally, it is all the tasks required to complete the work committed for the sprint and results in a successful sprint review demonstration.

Many teams don't develop a task list, and instead, just start coding based on the user stories in the backlog. This approach is haphazard and introduces the risk of defects being delivered to the customer, technical problems that cause overruns and schedule surprises.

The practices defined within RD, Technical Solution (TS), Verification (VER), Product Integration (PI) and Validation (VAL) are used to engineer the product to improve quality, product performance and manage team time. They can be applied to the features being developed in a single sprint, or different selections of features as they are developed across sprints. Here is a summary of how the practices can be used.

RD (requirements elicitation, development and analysis):

- generate, refine and analyze the backlog
- analyze and validate specific requirements in each sprint

TS (design and coding):

- select among design alternatives
- perform design activities (see the short related article on design [1])

- code using organizational coding practices

VER (peer reviews and testing):

- peer-review selected documents and code to find errors early and quickly
- plan and execute component level testing and analyze the results (e.g., defect density, defect pass rate, defect escape rate and root cause)

PI (interface check, integration planning and integration testing):

- plan and execute integration testing of components as they are completed, or when all components are complete
- check that interfaces are correct before spending time in system testing

VAL (end-user environment testing):

- plan and execute testing focused on the end-user's environment and needs
- analyze the results (e.g., defect density, pass rate, escape rate and root cause)

Some Scrum teams do some, or many, of these testing practices as part of their test-driven-development, code reviews and daily builds. The practices contained within the engineering process areas of Level 3 make a team's performance more systematic, organized and based on the best practices of the organization.

Organization learning and refinement

Scrum has a very good built-in retrospectives step to encourage a team to look at what happened in the recent sprint and determine any process and tool changes for future sprints. When performed correctly, a scrum team improves its performance very quickly. This can be measured by less rework, fewer defects and schedule accuracy.

The practices of Organizational Process Focus (OPF), Organizational Process Definition (OPD) and Organizational Training (OT) take lessons-learned to the next step.

OPF (coordinated improvement):

- take what is learned at the team level and organize and deploy this information across the organization. The result is that all teams improve faster from the positive and negative lessons of others

OPD (development of best practices and historical data):

- take what is learned at the team level and develop streamlined templates for the organization to use
- develop tailoring guidelines to determine which practices and options should be used for different project situations

OT (Organizational Training):

- assess, prioritize and deploy training across the organization, including domain-specific, technology and process skills needed to reduce errors and improve team efficiency

Summary

Scrum is a very good framework for planning and managing work. The intent of Scrum is to build systems in increments and obtain early feedback from end users. It is simple to learn and use.

However, Scrum is a framework, and by definition, does not contain all of the practices a project might need for different project situations. Practices from other sources can easily be added, whether they are from existing organizational best practices, published frameworks, books or models. When added with care, the intent of Scrum can be preserved.

References

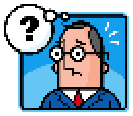
[1] www.processgroup.com/monthlytidbits.html#tidbit10
[2] www.processgroup.com/condensed-cmmi1p3-dev-v1.pdf

Other articles you might like

(www.processgroup.com/newsletter.html#scrum)

- Scrum - Lessons From the Trenches
- Implementing Scrum (Agile) and CMMI Together
- Scrum Sticky Issues
- Using Scrum Wisely - Where does Design Fit?

Need help?



Feel free to email or call with questions. We can:

- Teach Scrum, with or without references to CMMI

  See www.processgroup.com/services18asdcs.html

- Fix your current Scrum challenges

  If you are already using Scrum, but know that it could be better, we can help you determine where it is breaking and provide solutions to fix it.

Practical Solutions for Your Current Challenges

Webinar-style sessions to save on travel, or onsite coaching to save on time.

□ **Run your software development projects faster and incrementally.**
Two-day workshop, AGILE SOFTWARE DEVELOPMENT (SCRUM).

□ **Understand customer needs. Clarify product requirements early.**
Two-day workshop, IN SEARCH OF EXCELLENT REQUIREMENTS.

□ **Manage projects effectively. Meet project deadlines and reduce risks.**
Three-day workshop, PROJECT PLANNING AND MANAGEMENT.

□ **Meet project deadlines. Scope and estimate the project work.**
One-day workshop, PROJECT ESTIMATION.

□ **Avoid schedule delays caused by needless product rework. Find defects rapidly.**
Two-day workshop, INSPECTION (PEER REVIEWS).

□ **Hands-on CMMI. Perform a CMMI gap-analysis.**
The following workshops are available:

- CMMI-DEV: Overview (1/2 day), LEVEL 2 (1 day), LEVEL 3 (2 days), Intro to CMMI-DEV (3 days).
- Intro to CMMI-SVC (3 days), Supplement class (1 day), LEVEL 2 (1 day).

□ **Identify critical changes to improve organizational results. Benchmark against the CMMI.**
A PROCESS APPRAISAL examines your organization's current practices and generates a focused list of strengths and critical areas for improvement. Our certified Lead Appraisers conduct customized CMMI-based appraisals.

□ **Clarify and refine business/project measures and analysis.**
One-day workshop, MEASUREMENT AND ANALYSIS.

□ **Systematically evaluate decision alternatives.**
Half-day workshop, DECISION ANALYSIS AND RESOLUTION.

□ **Goal/problem-based improvement for service and development organizations.**
Two-day workshop, MAKING PROCESS IMPROVEMENT WORK.

□ **Manage your suppliers.**
One and one-half-day workshop, SUPPLIER MANAGEMENT.

□ **Achieve more with your time. Make your staff more productive.**
One-day workshop, TIME MANAGEMENT.

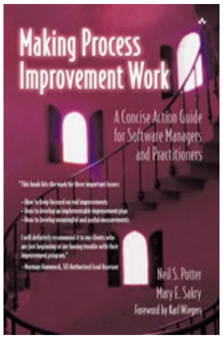
□ **Understand how to save money, produce more and work faster.**
Two-day workshop, DOING MORE FOR LESS.

□ **Tailored assistance. Dedicated phone/web-based assistance.**
This service consists of customized education and coaching on your specific problems (e.g., meeting deadlines, quality and cultural change).

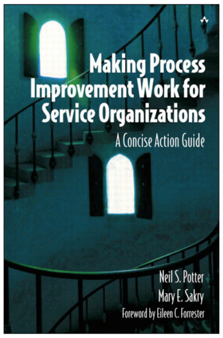
*Detailed information is available at **www.processgroup.com/services.html**. Contact us at **972-418-9541** or **help@processgroup.com** to discuss your needs.*

Read Our Books!

http://www.processgroup.com/book.html


Making Process Improvement Work
A Concise Action Guide for Software Managers and Practitioners
Neil S. Potter
Mary E. Sakry
Foreword by Karl Wiegers

Development audience


Making Process Improvement Work for Service Organizations
A Concise Action Guide
Neil S. Potter
Mary E. Sakry
Foreword by Eileen C. Forrester

Services audience

Chapter 1. Developing a Plan
- Scope the Improvement
- Develop an Action Plan
- Determine Risks and Plan to Mitigate
- Chapter Summary

Chapter 2. Implementing the Plan
- Sell Solutions Based on Need
- Work with the Willing and Needy First
- Keep Focused on the Goals and Problems
- Align the Behaviors of Managers and Practitioners
- Chapter Summary

Chapter 3. Checking Progress
- Are We Making Progress on the Goals?
- Are We Making Progress on our Improvement Plan?
- Are We Making Progress on the Improvement Framework?
- What Lessons Have We Learned So Far?
- Chapter Summary

The Process Group

Telephone: 972-418-9541

Fax: 866-526-4645

E-mail: help@processgroup.com

Web: www.processgroup.com

POST back issues are on line