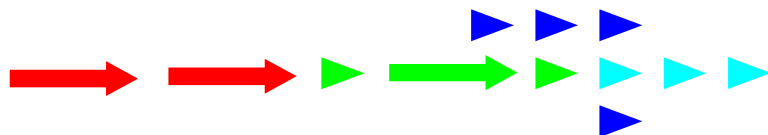# 4180: Defined Processes, Evidence, and Rescuing Corporate Knowledge: Achieving Standards Compliance in Agile and Lean Environments

SEPG Conference—March 2012

Dr. Richard Bechtold

Abridge Technology: www.abridge-tech.com

Overview

- Problem Statement
- Overview: Agile Software Development
- Overview: Lean Software Development
- Problem Statement, Part 2
- 7 Core Principles for Rescuing Corporate and Mission Knowledge in Agile and Lean Environments (actually 12)
  - The basic premise is that although all 12 principles may not be applicable in your environment, ideally at least 7 will be useful

Problem Statement

- According to a recent Gartner Report, an average of 10,000 'Baby Boomers' will retire every single day for the next 20 years

- The replacement workforce, often fresh from college, can have a radically different sense of loyalty, focusing on loyalty to:
  - Their resume
  - Their peers
  - Their career growth

Problem Statement

- In Agile or Lean environments, the emphasis is on software, not documentation

- The extensive exit of senior, experienced personnel can potentially result in a huge loss of essential corporate or mission knowledge

- How do you retain critical corporate knowledge in Agile and Lean environments (and leverage world-class standards)?

Overview:
Agile Software Development

Source: Wikipedia

- Agile development is a group of software development methodologies based on iterative, adaptive, and incremental development

- Solutions emerge from self-organizing, cross-functional teams

- Teams rapidly adapt to change

- Typically, time determines how much work is attempted and accomplished

Overview:
Agile Software Development

Source: Wikipedia

- Agile Manifesto: "We are uncovering better ways of developing software by doing it, and helping others do it. Through this work we have come to value:
  - Individuals and interactions over processes and tools
  - Working software over comprehensive documentation
  - Customer collaboration over contract negotiation
  - Responding to change over following a plan
- That is, while there is value in the items on the right, we value the items on the left more."

Overview:
Agile Software Development

- 12 principles of Agile Development
  - 1. Customer satisfaction by rapid delivery of useful software
  - 2. Welcome changing requirements, even late in development
  - 3. Working software is delivered frequently (weeks, rather than months)
  - 4. Working software is the principal measure of progress

Source: Wikipedia

Overview:
Agile Software Development

- 12 principles of Agile (cont.)

  - 5. Sustainable development, able to maintain a constant pace

  - 6. Close, daily co-operation between business people and developers

  - 7. Face to face conversation is the best form of communication (co-location)

  - 8. Projects are built around motivated individuals who should be trusted

Source: Wikipedia

Overview:
Agile Software Development

Source: Wikipedia

- 12 principles of Agile (cont.)

  - 9. Continuous attention to technical excellence and good design

  - 10. Simplicity

  - 11. Self-organizing teams

  - 12. Regular adaptation to changing circumstances

Overview:
Lean Software Development

- Lean software development is based upon concepts from
    - Agile software development
    - Lean Manufacturing
    - Lean Information Technology
- Premise: "Think big, act small, fail fast, learn rapidly."

Source: Wikipedia

Overview:
Lean Software Development

- 7 basic principles of Lean software development
  - 1. Eliminate waste
  - 2. Amplify learning
  - 3. Decide as late as possible
  - 4. Deliver as fast as possible
  - 5. Empower the team
  - 6. Build integrity in
  - 7. See the whole

Source: Wikipedia

Abridge

▶ ▶ ▶
▶ ▶ ▶
Technology ▶

Slide #: 11

rbechtold@abridge-tech.com

Overview:
Lean Software Development

- If it doesn't add value for the customer, it's considered to be waste, including

  - 1. Unnecessary code and functionality

  - 2. Delay in the software development process

  - 3. Unclear requirements

  - 4. Insufficient testing, leading to avoidable process repetition

  - 5. Bureaucracy

  - 6. Slow internal communications

Source: Wikipedia

Problem Statement, Part 2

- In the context of Agile and Lean development environments, how do you
  - Leverage experience gained during business or mission performance?
  - Reduce the likelihood of corporate or mission knowledge from being lost every time someone leaves the company?
  - Avoid repeated failures and capture lessons learned?

Core Principle #1: Extremely Brief Policies

- Key corporate and mission principles and best practices should be captured as extremely brief—bordering on terse—policies

- Visible (documented) policies clearly communicate what the organization requires (but typically not what it only expects)

Core Principle #1: Extremely Brief Policies

- Policy documents may be as brief as half a page (or less), containing only
  - Title
  - Purpose Statement
  - Policy Statement
  - Exemption Criteria
  - Conflict Resolution

Abridge
▶ ▶ ▶
➡ ➡ ▶ ➡ ▶ ▶
Technology ▶

Slide #: 15

rbechtold@abridge-tech.com

Core Principle #2: Minimalist Procedures

- Even procedures can/should be kept minimalist

- Initially target the material to senior, experienced professionals

- Wait for people to ask for details, before providing such details

- Strive for "demand driven" process migration or improvement to Lean and CMMI

Core Principle #3: Meeting Agendas

- Ineffective meetings are a commonly reported source of wasted time, effort, money, and motivation

- Typically, most every meeting will be more effective when driven by an agenda

- When people know the agenda, they may ask to be excused from the meeting, or may ask to be invited; either way, you can potentially improve the value of the meeting to all attendees

Core Principle #4: Key Meeting Minutes

- Even in a stand-up 15 minute meeting, if 8 people are attending, you just invested two staff hours

- Someone typically can take a moment to
  - Take a picture of the white board
  - Document key decisions
  - Check off (on the agenda) who attended
  - Document action items
  - Distribute recorded results

Core Principle #5: Self-Paced Training

- Corporate and mission knowledge should be captured primarily as training material

- Generally, when you conduct a post-increment or post-project lessons learned session, you may want to update your training material

- Make training material available to people to use on their own time, and many people will do exactly that

Core Principle #6: After Hours Training

- Consider holding training classes as an after-hours (unpaid) event

- Possibly ask developers to create and present at least 2 classes per year on any topic they want

- Open up attendance to anyone interested
  - If someone from the mailroom wants to volunteer his/her time to attend a website design class, why not allow them?

Core Principle #7: Optimal Templates

- Many of the artifacts you are repeatedly building do not need a unique structure and style every time you build them

- Tailorable templates can be created that actually accelerate the product creation process
  - Simple project plans
  - High-level requirements specifications
  - Standard interface conventions

Core Principle #8: Supplemental Material

- All organizations, due in part to experience, typically develop an extensive set of really good ideas
- These ideas originate in peoples' heads, and need to be captured, at least, within supplemental material
  - Training material
  - Standard templates
  - Guidelines
  - Checklists
  - FAQ databases
- You may eventually be audited against your policies, procedures, and plans, but virtually no one will audit you against supplemental material

rbechtold@abridge-tech.com

Core Principle #9: Independent Quality

- Someone attempting to do quality reviews of their own work tends to see what they expect to see

- Everything is supposed to go perfectly, but that rarely happens

- The amount of time that your organization should spend on independent quality assurance depends on your context and constraints, but "no time at all" is always the wrong approach

Core Principle #10: Avoid Perfection

- In the world of process transition and improvement, striving for perfection is a disaster

- The basic truth is you'll never get it perfect, regardless of how much money, time, and effort you pour into it

- For example, when developing plans sometimes you just know the plan is probably wrong due to
  - Key information you don't yet have
  - Key factors (and future events) you are not aware of

Core Principle #10: Avoid Perfection

- One option is to wait until you know everything (but this typically won't happen until well after the increment or project is completed)

- The other option is simply to ask yourself, is this initial version of the plan barely adequate?

- If the answer is yes, then go ahead and release it as version 1.0.

- By definition, if it's adequate, it's usable

- Use the barely adequate version to get started and, as with everything else, improve it over time

Core Principle #11: Leverage Resistance

- As a rule, people resist your attempted changes, transitions, and process improvements for a reason

- Sometimes, it's just reluctance to change

- However, as the expression goes, "Sometimes resistance is just process improvement in disguise"

- When you encounter resistance, always investigate carefully whether or not those resisting are doing so for a very good reason

Core Principle #11: Leverage Resistance

- Additionally, resistance typically comes in two categories
    - Active, overt, obvious, and confrontational
    - Passive, stealthy, subversive, and sometimes even disguised as support
- Which of the above do you want to encourage?

Core Principle #12: Leverage Standards

- Standards typically are built based upon world-wide, industry specific, best practices
  - CMMI-DEV
  - CMMI-SVC
  - ISO 9000
  - AS 9100
  - ISO 20000
- But you *always* have to interpret standards and best practices in relation to your specific context and constraints

Summary

- Agile and Lean development can potentially increase project efficiency and effectiveness

- However, an under-reliance on documentation and evidence may indicate that extensive— and highly valuable—corporate or mission knowledge exists only in peoples' heads, and therefore is too easily lost to the organization

- A careful blend of Agile principles and world-class standards compliance will likely be your best path to reliable agility

Contact Information

Dr. Richard Bechtold

President; Senior Consultant

Abridge Technology; Ashburn, VA, USA

(USA)703.729.6085

rbechtold@rbechtold.com

www.rbechtold.com

Biographical Highlights

Dr. Bechtold is a senior consultant for Abridge Technology, a Virginia-based company he founded in 1996. **Abridge Technology is an SEI Partner** and is authorized to provide SEI licensed training and appraisal services. Dr. Bechtold is an SEI Certified Lead Appraiser for both CMMI-DEV and CMMI-SVC. He is also a Certified Instructor for both. Dr. Bechtold provides consulting, training, and support services in the areas of project management, process improvement, process definition, measurement, and risk management. Dr. Bechtold has assisted government and industry with implementing the Software CMM since 1992, the Acquisition CMM since 1996, and the CMMI since 2000. Dr. Bechtold's expertise spans organizations of all types and sizes, from multi-billion dollar companies and agencies to organizations with less than 10 personnel.