Scrum and CMMI: A Magic Potion for Code Warriors

Jeff Sutherland, Ph.D.
Co-Creator of Scrum

Carsten Ruseng Jakobsen
Systematic Software Engineering

Kent Johnson
AgileDigm Inc.
Agenda

- Who are we?
  - Systematic Software Engineering, A/S
  - AgileDigm, Incorporated
- Scrum, agile and Lean
  - Scrum community
  - Scrum method
  - Scrum performance
- Systematic experience
  - What does CMMI 5 imply
  - Lean mindset transferred to S/W
  - Improvement Plans, Pilots, & Results
- Synergy between Scrum and CMMI
- Summary
- Questions
Systematic Software Engineering

- Established in **1985** and now Denmark’s **largest privately-owned** software and systems company
- **400+ employees**; 70% hold a MSc or PhD in software engineering
- High **employee satisfaction** – attractive **workplace** for ambitious software engineers
- Dun & Bradstreet credit rating: **AAA**
  - **High solidity. No bank debt** – fully **self-financing**
  - CMII **Level 5** and ISO 9001:2000 and AQAP 2110 + 150
  - Supplier of products and projects to more than **27 countries**, **export share** is 60%
  - 97% of our **customers would recommend Systematic** to other customers
- For further information – see **www.systematic.dk**

*Established in 1985 and now Denmark’s largest privately-owned software and systems company.*

*400+ employees; 70% hold a MSc or PhD in software engineering.*

*High employee satisfaction – attractive workplace for ambitious software engineers.*

*Dun & Bradstreet credit rating: AAA*

**High solidity. No bank debt** – fully **self-financing**

*CMII **Level 5** and ISO 9001:2000 and AQAP 2110 + 150*

*Supplier of products and projects to more than **27 countries**, **export share** is 60%*

*97% of our **customers would recommend Systematic** to other customers*

*For further information – see **www.systematic.dk***
**Plan driven development**

- High risk, high cost
- Problems with software
  - Fails to fit customer needs
  - High defect rate
- 50% waste
- Delays time to market
- Poor working environment

**Value driven development**

- Low risk, low cost
- Produces software that excites the customer
- Minimal waste
- Accelerates early revenue
- Energized working environment
Lean Strategy: replace waterfall plan with adaptive planning ... while maintaining CMMI Level 5 compliance
Directive from Strategic Planning Session in summer 2005: Future Improvements should be primarily based on Lean
Causal Dependencies

Tools divided into three dimensions

Level\Dimension

<table>
<thead>
<tr>
<th>Level</th>
<th>Dimension</th>
<th>Value</th>
<th>Flow</th>
<th>Pull</th>
<th>Perfection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Production</td>
<td>Value</td>
<td>P6 Build Integrity in</td>
<td>P2 Amplify Learning</td>
<td>P2 Amplify Learning</td>
<td>P6 Build Integrity in</td>
</tr>
<tr>
<td></td>
<td></td>
<td>T19 Refactoring</td>
<td>T5 Synchronization</td>
<td>T3 Feedback</td>
<td>T18 Conceptual integrity</td>
</tr>
<tr>
<td></td>
<td></td>
<td>T20 Test</td>
<td>T4 Iterations</td>
<td>T6 Setbased development</td>
<td></td>
</tr>
<tr>
<td>Management</td>
<td>Value</td>
<td>P1 Create Value</td>
<td>P4 Deliver Fast</td>
<td>P7 See the Whole</td>
<td>P3 Defer Commitment</td>
</tr>
<tr>
<td></td>
<td></td>
<td>T1 Eliminate Waste</td>
<td>T11 Queue theory</td>
<td>T22 Contracts</td>
<td>T7 Options thinking</td>
</tr>
<tr>
<td></td>
<td></td>
<td>T2 Value streams</td>
<td>T12 Cost of delay</td>
<td>T21 Measurement</td>
<td>T8 Defer commitment</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>T10 Pull</td>
<td>T9 Decisionmaking</td>
</tr>
<tr>
<td>People</td>
<td>Value</td>
<td>P5 Empower team</td>
<td>P5 Empower team</td>
<td>P5 Empower team</td>
<td>P5 Empower team</td>
</tr>
<tr>
<td></td>
<td></td>
<td>T16 Expertise</td>
<td>T14 Motivation</td>
<td>T15 Leadership</td>
<td>T13 Self determination</td>
</tr>
</tbody>
</table>

Thinking tools are best transformed by people and projects
Improvement Opportunities

Quantitative Thinking driven by Business Objectives, Process Performance, and Cost Benefit Analysis was used to identify key improvements:

- **Defect Containment**
  Analysis shows that cost of fixing defects in later phases than coding increases significantly.

- **Lean Thinking Tool**
  Refactoring / Test

- **Suggested resolution**
  Early Test

- **Cycle Time**
  Focus on high quality has gradually increased time spent on final test and thereby cycle time.

- **Lean Thinking Tool**
  Synchronization and Iterations

- **Suggested resolution**
  Deliver often in small iterations
Improvement Plan and Pilots - 1

- Four projects pilot in a period of 4-6 months
  - Early testing:
    - Enhanced story-based (early testing) development
    - Colocation of tester and developer
    - Features subdivided into stories
    - Checklist driving implementation of stories
  - Synchronization and iterations:
    - Plan with many small iterations of 2-4 weeks and ensure high communication within team and to customer
Results

- Large projects doubled productivity
- Early testing: Strong indication that it works. Defects in final test reduced by 40%.
- It was realized that what we trying to pilot already had a name: **SCRUM**.

Decision

- Scrum and Story Based Development to be the new default choice for future projects within Systematic
Scrum in a Project’s Lifecycle

CMMI: *Project Planning*
- SG1: Establish Estimates
- SG2: Develop a Project Plan
- SG3: Obtain Commitment to the Plan

CMMI: *Project Monitor and Control*
- SG1: Monitor Project Against Plan
- SG2: Manage Corrective Actions to Closure

**Definition & Planning**

**Implementation**

**Launch & Closeout**

Sprints

Scrum: *Create Product Backlog*
- Define backlog items
- Establish Estimates
- Prioritize backlog items
- Identify dependencies

Scrum: *Create Sprint Backlog*
- Monitor progress against sprint plan
- Remove impediments

© Jeff Sutherland 1993-2007
The CMMI Model says *what to do*, but not *how to do it!*
Systematic CMMI 5 Analysis
First six months of Scrum

- 80% reduction in planning and documentation costs (still under discussion)
- 40% reduction in defects
- 50% reduction in rework
- 100% increase in overall productivity
- Systematic decided to change CMMI Level 5 process to make Scrum the default mode of project management
- When waterfall project management is required, they are now need to be contracted for twice the price of Scrum projects
  - Required by some defense and healthcare agencies
  - Results are lower business value
  - Lower customer satisfaction
  - Lower quality
  - Twice the cost

© Jeff Sutherland 1993-2007
CMMI/Agile Lessons Learned

- Key to achieve more agility with the CMMI is to realize that the practices are primarily advisory or indication only.

- Practices don't need to be the ones described in the CMMI specification. You have the freedom to choose other evidence.

- The core issue is not whether CMMI-like or Agile-like practices are best, the core issue is building an organizational culture with a balance of discipline and adaptability.
Numbers are true and accurate

- We simultaneously achieved
  - Lower cost
  - More features
  - Shorter time
  - Higher customer satisfaction
  - Higher employee satisfaction

- All at once!
Summary

- **Lean Software Development is a successful operational tool to identify improvement opportunities in a CMMI Level 5 company.**

- Using CMMI and Scrum together results in significantly improved performance
  - Scrum pilot projects showed significant gains in productivity and quality over traditional methods.
  - Pilots show reduction in every category of work (defects, rework, total work required, and process overhead) by almost 50%.

- **CMMI Generic Practices can be used to institutionalize and strengthen agile practices.** Implementing these practices can help establish needed discipline for any Agile Method.
Recommendations

- Companies requiring high maturity processes should consider introducing Agile practices
  - The cost of implementing CMMI may be significantly reduced with Scrum.
  - Time to implement CMMI may not be reduced as significant cultural changes are required.

- The Agile community should use the CMMI generic practices to amplify the benefits from Agile methods.
  - The concepts of institutionalization, standardization, training, communication, and management responsibilities embedded in CMMI practice can enhance a Scrum implementation.

- The key is to build an organizational culture with a balance between discipline and change readiness.
Questions?
“Development is an exercise in discovery, while production is an exercise in reducing variation, and for this reason, a lean approach to development results in practices that are quite different than lean production practices.”

“At the core of this book are 22 thinking tools to aid software development leaders as they develop the agile practices that works best in their particular domain. This is not a cookbook of agile practices.”
Toyota Way: Learn by Doing
Fujio Cho, President, 2002

- We place the highest value on actual implementation and taking action. *Agile Principle #1*

- There are many things one doesn’t understand the therefore, we ask them why don’t you just go ahead and take action? *Agile Principle #3, #11*

- You realize how little you know and you face your own failures and redo it again and at the second trial you realize another mistake ... so you can redo it once again. *Agile Principle #11, #12*

- So by constant improvement ... one can rise to the higher level of practice and knowledge. *Agile Principle #3*
Toyota consulting applied to most agile U.S. company (industrial sensor company – 6 month results)

- 93% reduction in lead time to product
- 83% reduction in work-in-progress inventory
- 91% reduction in finished goods inventory
- 50% reduction in overtime
- 83% improvement in productivity

- The mindset is the key to transformation. Consulting help from outside experts is critical in early stage.