Lessons Learned in Seamless Integration of CMMI, TSP, and PSP
Why All Three Are Needed

10th Annual Systems Engineering Conference
October 24, 2007
Advanced Information Services Inc.

Winner IEEE Software Process Achievement Award
http://www.sei.cmu.edu/managing/ieee-award/ieee.award.html
Topics

- Issues
  - Quality and Schedule
  - Rational Management and Commitment
  - Insanity and Malpractice

- Three Improvement Perspectives
  - Organization - CMM/CMMI
  - Individual – PSP
  - Team – TSP

- Seamless Integration of CMMI, PSP, TSP
  - The glue – Process Improvement Proposal
  - AIS Experience

- Lessons Learned
Quality Is More Important Than Schedule

“In today’s software marketplace, the principal focus is on cost, schedule, and function; quality is lost in the noise. This is unfortunate since poor quality performance is the root cause of most software cost and schedule problems.”

Watts Humphrey
Rational Management - Developers

- When pressed for early deliveries, the responsible team members say

  “I understand your requirements, I will do my utmost to meet it, but until I make a plan, I cannot responsibly commit to a date”
Rational Management - Managers

- When pressed for early deliveries, the responsible managers say

  “I trust you to create an aggressive and realistic plan, I will review the plan, but I will not commit you to a date that you can not meet”
Rational Management - Principles

- Set challenging goals
- Get the facts
- Use facts and data
- Anticipate and address problems
Insanity or Malpractice?

Insanity
Doing the same thing over and over and expecting a different result

Malpractice
An organization which does not have a top-management-sponsored continuous improvement initiative in place
<table>
<thead>
<tr>
<th>Level</th>
<th>Focus</th>
<th>Key Process Areas (KPA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Optimizing</td>
<td>Defect prevention</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Technology change management</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Process change management</td>
</tr>
<tr>
<td>4</td>
<td>Managed</td>
<td>Product and process quality</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Quantitative process management</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Software quality management</td>
</tr>
<tr>
<td>3</td>
<td>Defined</td>
<td>Engineering process</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Organization process focus</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Organization process definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Training program</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Integrated software management</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Software product engineering</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Intergroup coordination</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Peer reviews</td>
</tr>
<tr>
<td>2</td>
<td>Repeatable</td>
<td>Project management</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Requirements management</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Software project planning</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Software project tracking</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Software quality assurance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Software configuration management</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Software subcontract management</td>
</tr>
</tbody>
</table>
Comparing SW-CMM to CMMI

**SW-CMM key process areas**

- Level 5: Optimizing
  - Defect Prevention
  - Technology Change Management
  - Process Change Management

- Level 4: Managed
  - Quantitative Process Management
  - Software Quality Management

- Level 3: Defined
  - Organization Process Focus
  - Organization Process Definition
  - Training Program
  - Integrated Software Management
  - Software Product Engineering
  - Intergroup Coordination
  - Peer Reviews

- Level 2: Repeatable
  - Requirements Mgmt
  - Software Project Planning
  - Software Project Tracking & Oversight
  - Software Subcontractor Management
  - Software Quality Assurance
  - Software Configuration Management

**CMMI Process Areas**

- Causal Analysis and Resolution
- Organizational Innovation and Deployment
- Organizational Process Performance
- Quantitative Project Management
- Risk Management
- Requirements Development
- Technical Solution
- Product Integration
- Verification
- Validation
- Decision Analysis and Resolution

Advanced Information Services Inc.

© AIS 2007
Issues Addressed by CMM

- Getting management attention
- Maintaining long-term improvement focus
- Guiding the improvement work
CMM Results – Schedule
GM

- Average number of days late in meeting milestones declined from over 50 days to fewer than 10 following organization focus on CMMI
CMM Results – Defects

<table>
<thead>
<tr>
<th>CMM Level</th>
<th>Defects/KLOC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7.5</td>
</tr>
<tr>
<td>2</td>
<td>6.24</td>
</tr>
<tr>
<td>3</td>
<td>4.73</td>
</tr>
<tr>
<td>4</td>
<td>2.28</td>
</tr>
<tr>
<td>5</td>
<td>1.05</td>
</tr>
</tbody>
</table>

The TSP in Practice, SEI Technical Report, September 2003
CMM Problems

- No simple model could precisely measure process maturity and complex models are not useful in guiding improvement.
- CMM consciously focused on what organization should do, not on how they should do it.
- The teamwork practices and personal disciplines required for quality software work are almost entirely issues of how, and not just what.
- Because engineers will not change the way they work without very specific guidance, the CMM does not change engineering behavior.
The Real Need

- The need is not for lots of process data but for engineers who gather and use that data.
- What would happen if software professionals used sound engineering practices?
  - made and followed detailed plans
  - gathered and used historical data
  - measured and managed quality
  - analyzed and improved their processes
- The need is for a Level 5 Process at the individual level.
Self Improvement
From Project To Project

“You can not stand still, so you should treat every project as a way to build talent rather than merely treating your talent as a way to build projects”

Watts Humphrey
Self Improvement
Personal Software Process - 1

- PSP3: Cyclic development
- PSP2: Code reviews, Design reviews
- PSP2.1: Design templates
- PSP1: Size estimating, Test report
- PSP1.1: Task planning, Schedule planning
- PSP0: Current process, Time recording, Defect recording, Defect type standard
- PSP0.1: Coding standard, Size measurement, Process improvement proposal (PIP)

Source: Software Engineering Institute
Self Improvement
Personal Software Process - 2

At the end of the PSP training, developers know how to:

- Consistently gather size, time, and defect data
- Make commitments based on historical data
- Analyze personal data to answer questions
  - Where am I spending my time?
  - What are my common defects?
  - Where do I inject the defects?
  - What goals do I need to set to improve?
PSP Results – Schedule
AIS

Schedule Deviation Individual Value Control Chart - Commercial Systems

Date of Project Phase Start

Individual Data Points
Mean
Upper Natural Process Limit
Lower Natural Process Limit
One Standard Deviation
PSP Problems

- To do quality work, engineers need a detailed plan and a defined process.
- Without the process, they cannot make detailed plans, take consistent measurements, or track their work against the plan.
- However, when engineers have a project to deliver, they are rarely willing to take the time to define a complex process, even when they know how.
The Real Need

- Need a mechanism to guide teams through defining their processes and making complete, precise, and detailed plans.

- Need a vehicle to help organizations capitalize on the potential benefits of disciplined teamwork.
Team Improvement
Jelled Teams

“The speed with which organizations form and deploy teams is the single most important factor in determining their competitive success”

“Jelled teams are the most powerful tool ever devised for doing challenging work”

Watts Humphrey
Team Improvement
Self-directed Teams

- Characteristics of self-directed teams
  - Sense of membership and belonging
  - Commitment to a common team goal
  - Ownership of the process and plan
  - The skill to make a plan, the conviction to defend it, and the discipline to follow it
  - Dedication to excellence
Building Self-directed Teams
The TSP Launch Process

Day 1
1. Establish product and business goals
2. Assign roles and define team goals
3. Produce development strategy and process

Day 2
4. Build overall and near-term plans
5. Develop the quality plan
6. Build individual and consolidated plans

Day 3
7. Conduct risk assessment
8. Prepare management briefing and launch report

Day 4
9. Hold management review
Launch postmortem

A qualified TSP team coach guides the team through a defined process to develop its plan and to negotiate that plan with management.
Self-directed Teams
Project Tracking Issues - 1

- With PSP training, developers know how to plan, schedule, and track their work
- TSP teams use these PSP-learned methods to make detailed plans
  - Tasks are no more than 10 task hours each
  - Task time is recorded daily
  - EV is measured weekly
- You can tell project status to within 10 task hours
- TSP teams regularly report their status
Self-directed Teams
Project Tracking Issues - 2

- Project schedules slip a day at a time
- If you cannot precisely measure project status, you will not know where projects stand
- Without such knowledge, you cannot address schedule problems in time to fix them
- With the TSP, you can
  - closely monitor team performance
  - address problems in time
  - consistently meet schedules
TSP Results – Task Hours

Average Task Hours Per Team Member

Source: Allied Signal
Source: “From MCC to CMM”, Dr. Bill Curtis, DC SPIN, April 2006
Process Improvement Principles

- It takes time, skill, and money to improve the software process
- To improve the software process, someone must work on it
- Unplanned process improvement is wishful thinking
- Automation of a poorly defined process will produce poorly defined results
- **Improvements should be made in small steps**
- Train, train, train!

Source: *Managing the Software Process*, Watts Humphrey
Empowered Culture
Process Improvement Proposals (PIPS)

PROCESS IMPROVEMENT PROPOSAL (PIP)

PIP# :
Written By:
Date :
Author(s) : Project :

Process Name : Key Process Area :

Improvement Description :

Improvement Benefits (Check One):
- Document Improvement
- Reduced Cycle Time
- Improved Quality
- Reduced Risk

Benefits Description (Quantify Where Possible):
(Attach files if needed)

Attach the PIP Pilot Report here (if applicable):

Submit

▼ SEPG Evaluation
The AIS PIP Process

Process Change Management

- Create New PIP
- Open PIP Queue
- About PIP Database

Initiating

Diagnosing

Establishing

Assign Resources

Implement & Integrate

Feedback

Leveraging

Acting

Advancing PIP
AIS PIPS Summary

Jan 22, 1992 – To date

No. of PIPS submitted: 1502
No. of PIPS implemented: 972
No. of PIPS by improvement category:
  • Improved quality: 232
  • Reduced cycle time: 86
  • Reduced risk: 63
  • Improved documentation: 161
  • Not categorized: 410
Sample PIPs – Organization Process

- Incorporate the TSP into the AIS CPIW as suggested by the attached work products (ProjectCommitmentProcess.zip) which reflect the current practice.
- Change Launch meeting 9A so that review is held, not only by management, but also peer Project Managers. Accordingly, these same individuals may need to be present in meeting 1B.
Sample PIPs – Team Process

- For UI component enhancements, change process to do Design Inspection, Test Case Inspections and Code Inspections after Compile

- For components where performance requirement is critical, execute two rounds of unit test
  - Unit test of performance test cases before code inspection
  - Unit test of features after code inspection
Sample PIPs – Personal Process

- Reduce phase distribution % for Design Review for UI Components
- Update Personal Review Checklist
- Batch process E Mail three times a day
- Move end of day post mortem to start of day to process and analyze previous day’s data
Lessons Learned - 1

- While models are useful to indicate where improvements are needed, only committed people can make the improvements.
- A supportive management environment that rewards disciplined behavior is absolutely essential.
- Timely feedback on the status and disposition of the PIPs is important to sustain the PIP mechanism and feeling of empowerment.
- Do not need to wait till level 5 to start implementing process change management.
Lessons Learned - 2

- While CMM is necessary as an organizational capability improvement model, it is not sufficient to change engineering behavior; the PSP provides the detailed “how to” for improvement at the individual level.

- The TSP provides the management framework for continuously improving self directed teams. The PIP mechanism is key for team ownership of the project’s process and commitment to improve.

- CMM, TSP, and PSP all three are needed for an integrated approach to model based improvement at the organization, team, and individual levels without the risk of sub-optimization.
Trademarks and Service Marks

- The following are service marks of Carnegie Mellon University.
  - CMMI<sup>SM</sup>
  - Team Software Process<sup>SM</sup>
  - TSP<sup>SM</sup>
  - Personal Software Process<sup>SM</sup>
  - PSP<sup>SM</sup>

- The following are registered trademarks of Carnegie Mellon University.
  - Capability Maturity Model®
  - CMM®
  - Capability Maturity Model® Integration
  - CMMI®
  - CERT®
Contact Information

Girish Seshagiri
Advanced Information Services Inc.
(703) 286 0781
Email: girishs@advinfo.net
Website: www.advinfo.net