

Proceedings of the First International Research Workshop for Process Improvement in Small Settings, 2005

Suzanne Garcia
Caroline Graettinger
Keith Kost

January 2006

SPECIAL REPORT
CMU/SEI-2006-SR-001



**Carnegie Mellon
Software Engineering Institute**

Pittsburgh, PA 15213-3890

Proceedings of the First International Research Workshop for Process Improvement in Small Settings, 2005

CMU/SEI-2006-SR-001

Suzanne Garcia
Caroline Graettinger
Keith Kost

January 2006

Software Engineering Process Management

Unlimited distribution subject to the copyright.

This report was prepared for the

SEI Administrative Agent
ESC/XPK
5 Eglin Street
Hanscom AFB, MA 01731-2100

The ideas and findings in this report should not be construed as an official DoD position. It is published in the interest of scientific and technical information exchange.

This work is sponsored by the U.S. Department of Defense. The Software Engineering Institute is a federally funded research and development center sponsored by the U.S. Department of Defense.

Copyright 2006 Carnegie Mellon University.

NO WARRANTY

THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

Use of any trademarks in this report is not intended in any way to infringe on the rights of the trademark holder.

Internal use. Permission to reproduce this document and to prepare derivative works from this document for internal use is granted, provided the copyright and "No Warranty" statements are included with all reproductions and derivative works.

External use. Requests for permission to reproduce this document or prepare derivative works of this document for external and commercial use should be addressed to the SEI Licensing Agent.

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.

For information about purchasing paper copies of SEI reports, please visit the publications portion of our Web site (<http://www.sei.cmu.edu/publications/pubweb.html>).

SM CMM Integration, IDEAL, Personal Software Process, PSP, SCAMPI, SCAMPI Lead Appraiser, SEPG, Team Software Process, and TSP are service marks of Carnegie Mellon University.

[®] Capability Maturity Model, Carnegie Mellon, CMM, and CMMI are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

Table of Contents

Abstract	xi
Acknowledgements	xiii
1 Introduction	1
1.1 Goals/Approach of the Workshop.....	1
1.2 Organization of the Proceedings	2
2 Research Directions	5
2.1 Addressing Infrastructure Issues in Very Small Settings	5
2.2 A Multi-Method Evaluation of the Practices of Small Software Projects ...	12
2.3 Barriers to Adoption of the CMMI Process Model in Small Settings	18
2.4 Measuring Performance Results in Small Settings: How do you do it and what matters most?	23
2.5 Results of a Field Study of CMMI for Small Settings Using Rapid Applied Ethnography	27
2.6 Applying Software Engineering Standards in Small Settings: Recent Historical Perspectives and Initial Achievements.....	39
2.7 Defect Reduction Through Objectivity in Small Settings	52
2.8 Team Innovation Management (TIM): Research Into Practice	60
2.9 Critical Success Factors (CSF) in SPI Bibliography	72
3 Process Improvement Approaches and Models	81
3.1 An Experience on Implementing the CMMI in a Small Organization Using the Team Software Process.....	81
3.2 MoProSoft®: A Software Process Model for Small Enterprises.....	93
3.3 Using Agile Practices and the CMMI to Achieve High Project Management Capability in Small Settings	102
3.4 Business Benefits from Successful Process Measurements.....	108
3.5 Empowered Engineers are key players in process improvements	115
3.6 Accelerated Process Improvements for Small Settings.....	117
4 Process Improvement Tools and Techniques	127

4.1	Process Improvement as a Real Option to Extract Value from Project Failure in Context of Small Business.....	127
4.2	Process Performance Models - Lessons Learned from 27 SCAMPI Appraisals for Small Companies	135
4.3	A Pattern-Based Approach to Deploy Process Improvements in Small Settings	157
4.4	Value-Centric Process Improvement for Small Organizations by Using QFD and CMMI	166
5	Regional Approaches.....	175
5.1	The Capability Maturity Model (SW and Integrated) Tailored in Small Indigenous Software Industries	175
5.2	A Software Process Improvement Solution for Small and Medium-Size Enterprises	182
5.3	RAMALA: A SPI Service Provider for SMEs.....	193
5.4	The ACTI PROFO Process Improvement Initiatives.....	202
5.5	International Research Workshop for Process Improvement in Small Settings Position Paper CMM Fast Track (CMMFT) Programme for Hong Kong SME Software Companies	208
5.6	Quality Software Map of South America.....	216
5.7	The Implementation of SW-CMM Level 2—A Successful Case of a Brazilian Company’s Organizational Competence and Commitment.....	228
6	Selected Case Studies	233
6.1	A Giant Taking Small Steps	233
6.2	Small Steps, Giant Leap	236
6.3	Software Process Improvement at Schweitzer Engineering Laboratories, Inc.....	240
6.4	Two Case Studies in Implementing Model Based Process Improvement in Small Organizations	245
6.5	Process Improvement in a Small Company	251
7	Workshop Highlights.....	267
7.1	Keynote	267
7.2	Breakout Sessions Summary	267
7.2.1	Day 1	267
7.2.2	Day 2	269
7.3	Discussant Reflections	270
8	Suggested Next Steps and Summary	273
8.1	Suggested Next Steps.....	273
8.1.1	Community Support.....	273

8.1.2	Online Training/Distance Learning.....	274
8.1.3	Finding Research Sponsors.....	274
8.1.4	Publications.....	274
8.1.5	Sponsoring/Hosting Future Workshops	274
8.1.6	What's Happening Now/Near term	275
8.2	Summary	275
Appendix	Workshop Breakout Session Results.....	277

List of Figures

Figure 1: Research Design.....	31
Figure 2: Evolution of Published ISO/IEC Software and Systems Engineering Standards (SC7 2005).....	41
Figure 3: Relationship Between Key SC7 Standards [Coallier 03].....	42
Figure 4: Priorities and Concern Differences Based on Organization Size [McFall 03]	44
Figure 5: The New Working Group’s Proposed Work Schedule.....	45
Figure 6: Referenced Documents.....	48
Figure 7: Measurement of Objectivity in a QA Evaluation	53
Figure 8: The Intersection.....	64
Figure 9: Smart Pipe Process Flow.....	68
Figure 10: Smart Pipe ETVX Diagram.....	69
Figure 11: CSF from Bibliography	73
Figure 12: Critical Success Factors.....	74
Figure 13: Self-Assessment Results	85
Figure 14: Process Categories	96
Figure 15: Control Chart Plotted for Weekly Effort Burn-Down	105
Figure 16: Size Tracking.....	110
Figure 17: Productivity Tracking	111
Figure 18: Rapid-Q: Off The Shelf Quality System.....	120

Figure 19: Rapid-Q Process View	121
Figure 20: Rapid-Q Accelerates Process Implementation.....	122
Figure 21: Waterfall Lifecycle Model	136
Figure 22: The Purpose for Each Phase	138
Figure 23: An Example of a Resource Model.....	147
Figure 24: Model-driven Process	155
Figure 25: Structure of Project Patterns	159
Figure 26: Project Categorization.....	160
Figure 27: Project Pattern Search Criteria	161
Figure 28: Project Pattern Instantiation	162
Figure 29: Quantitative Postmortem Analysis	163
Figure 30: Research Validation	178
Figure 31: Action Package Architecture	183
Figure 32: Requirements Development Process.....	185
Figure 33: Requirements Management Process.....	185
Figure 34: Structure of the Requirements Engineering Action Package “Products and Documents”	186
Figure 35: RAMALA Knowledge Base Structure.....	195
Figure 36: RAMALA Software Architecture	196
Figure 37: Software Reference Model Elements Stored in RAMALA	197
Figure 38: Selecting Processes from the PMBOK Process Framework for Assessment.....	197
Figure 39: Process Element Assessment Questionnaire.....	198
Figure 40: The Organization’s Software Process Elements Capacity.....	199

Figure 41: An Organization's Standard Software Process	200
Figure 42: Major forces behind improvement initiatives	221
Figure 43: Typical Engineering Life Cycle Model	252
Figure 44: CMMI in a Nutshell	254
Figure 45: CMMI Model Components.....	255
Figure 46: CMMI Staged Representation Model	255
Figure 47: CMMI Model Structure.....	256
Figure 48: SSAI CMMI Model Structure, Maturity Level 2 (CMMI-SE/SW, V1.1, Staged)	259
Figure 49: Problem Solving Process	260
Figure 50: Process Improvement Life Cycle Model.....	262

List of Tables

Table 1:	IRW Breakout Topics.....	3
Table 2:	Size of IT Companies in the Montreal Area [Laporte 05].....	40
Table 3:	Interspersing Quality Gates Within SDLC	56
Table 4:	Roles for Engineering Activities.....	57
Table 5:	Results When Each Role is Performed by Discrete Personnel.....	57
Table 6:	Results When Each Role is not Performed by Discrete Personnel	58
Table 7:	Innovation Recording Form	67
Table 8:	Activities and Time Estimation for the SPI Implementation	86
Table 9:	Model Comparison	94
Table 10:	Capacity Level Increase by Category.....	98
Table 11:	Effort and Improvement Data by Company	99
Table 12:	Large Company vs. Small Company Pain Areas.....	118
Table 13:	Basic Attributes of Resource Model	144
Table 14:	An Example—Metrics for Design Phase	148
Table 15:	Indirect Metrics Transform Computing Table	152
Table 16:	Voice of Customer Matrix	169
Table 17:	House of Quality	170
Table 18:	Practice Evaluation Matrix.....	171
Table 19:	Action Evaluation Matrix.....	171

Table 20:	IDEAL Model with QFD	172
Table 21:	Industrial Domains of the Enterprises Involved in MESOPYME Experimentation	188
Table 22:	Employees Distribution — Totals and Those Involved in Software Development Unit.....	188
Table 23:	MESOPYME's Obtained Results	189
Table 24:	Hour Fee per Consultant.....	217
Table 25:	Quality Map on South America.....	218
Table 26:	Asociación Latinoamericana y del Caribe de Entidades de Tecnología de Información (ALETI)	219
Table 27:	Active SPINs [SEI 05]	222
Table 28:	Emergent SPINs [SEI 05].....	222
Table 29:	Time and Effort for Mini and Small Projects	236
Table 30:	Case Attribute Profile	246
Table 31:	Contents of Case 1 and Case 2 Process Improvement Implementation Plans	247
Table 32:	SCAMPI Appraisals and Model Scopes for Case 1 and Case 2	248
Table 33:	Process Performance Gains for Case 1 and Case 2	248
Table 34:	Case 1 and Case 2 Lessons Learned in Implementing Formal Process Improvement	249
Table 35:	Practice Implementation Indicators (PII)	257
Table 36:	SSAI Bodies of Knowledge Mapping	258

Abstract

The first International Research Workshop for Process Improvement in Small Settings was held October 19-20, 2005 at the Software Engineering Institute in Pittsburgh, Pennsylvania. Attendees from Australia, Canada, Chile, China, Germany, Ireland, India, Japan, Malaysia, Mexico, Spain, and the United States discussed the challenges of process improvement in small and medium size enterprises, small organizations within large companies, and small projects. The presentations addressed starting and sustaining process improvement, qualitative and quantitative studies, and using Capability Maturity Model Integration (CMMI), Agile, Modelo de Procesos para la Industria de Software (MoProSoft), International Organization for Standardization (ISO), Quality Function Deployment (QFD), and Team Software Process (TSP) in small settings. The workshop also had working groups that discussed issues unique to small settings, such as regional support centers and process improvement “on a shoestring.”

This report includes the papers from this workshop and presents conclusions and next steps for process improvement in small settings. This report also contains the workshop breakout session results.

Acknowledgements

The authors would like to thank the following individuals for their hard work, dedication, and commitment to making this workshop a success.

Program Committee

Nidhi Srivastava (Tata Consultancy Services)

Hanna Oktaba (National Autonomous University of Mexico)

Gonzalo Cuevas (Polytechnic University of Madrid)

Mary Anne Herndon (Transdyne Corporation)

Group Facilitators

Agapi Svolou (Software Engineering Institute)

Miguel Serrano (Mexico Center for Mathematical Research)

Event Coordinators

Linda Canon (Software Engineering Institute)

Shane McGraw (Software Engineering Institute)

Stacy Mitchell (Software Engineering Institute)

David Gregg (Software Engineering Institute)

Special thanks go to Dr. Paul Nielsen, director and CEO of the SEI, and Mr. William Peterson, director of the SEI's Program on Software Engineering Process Management. These individuals provided strong and visible support to this endeavor and the process needs of small settings.

1 Introduction

The first International Research Workshop (IRW) for Process Improvement in Small Settings was held in Pittsburgh, Pennsylvania, USA, on October 19 and 20, 2005. The workshop was the result of two synergistic forces:

- One of the goals of the SEI's Applying CMMI in Small Settings (ACSS) project was to foster communication and collaboration among worldwide researchers to leverage learning related to applying CMMI and other process improvement techniques in small settings (projects, organizations, and companies).
- The International Process Research Consortium (IPRC) identified implementing process improvement in small settings as one of the early high-priority topics needing primarily transition research rather than technology research.

The workshop was conceived as a joint effort between these two projects and the solicitation for position papers was met with a much greater response than we anticipated. We selected participants from a truly international pool of candidates—both university and industry researchers—who provided the papers included in this report and made presentations at the workshop.

These two projects and the University of Pittsburgh Medical Center (UPMC) were the sponsors for the workshop. UPMC is one of the largest nonprofit integrated health care systems in the United States. Its strong clinical and research reputation draws patients from throughout the United States and dozens of countries across the globe. With many small projects making up UPMC's IT strategy, UPMC is a strong supporter of process improvement in small settings.

It is the hope of the project team that this will be the first of many such workshops. The participants were enthusiastic about hosting future workshops on this topic, and the SEI is actively considering where and how to make this happen. To stay up to date with future plans in this area, visit the IPRC website: <http://www.sei.cmu.edu/iprc>.

1.1 Goals/Approach of the Workshop

The workshop focused on research from the world-wide community addressing the unique issues of process improvement in small settings, including small teams, small projects, small organizations, and small businesses. Researchers were asked to submit papers that addressed the following topics:

- their activities in small settings

- their environment (small projects, small organizations, etc.)
- the models, techniques, and approaches for process improvement they are using, recommending, and researching
- the strengths and weaknesses of their models, techniques, or approaches

The approach in designing the workshop was to invite and solicit position papers from researchers working in small settings and then to select the submitted papers that represented diverse regional, conceptual, domain, and business sector viewpoints. In addition to the position paper that each author submitted for inclusion in this report, we invited participants to highlight, in a “10-minute madness” session, the key points of their position that they would like other participants to consider in the breakout/discussion sessions. The time-constrained presentation format forced presenters to distill their presentations down to one or two ideas. All attendees participated in this session, which resulted in 22 presentations over two days.

The workshop format allowed for presentation sessions in the morning and the breakout/discussion groups in the afternoon. We grouped the presentations that addressed similar topics and then proposed discussion topics that used these topics as starter ideas. Participants were also invited to suggest additional topics for breakouts.

1.2 Organization of the Proceedings

This report contains four major sections: the papers submitted by participants, organized into five major topics; workshop highlights; next steps and summary; and raw notes from the breakout sessions.

The papers submitted by participants are organized into the following categories:

- Research directions: these papers focus on new concepts and ideas for supporting process improvement in small settings that would benefit from further research by the community.
- Process improvement approaches and models: these papers focus on experiences with different models of improvement and different approaches for motivating and supporting process improvement in small settings.
- Process improvement tools and techniques: these papers focus on particular tools or techniques used in some aspect of an improvement effort in a small setting.
- Regional approaches: these papers focus on ways that different geographical regions are motivating and supporting process improvement in small settings.
- Selected case studies: these papers are experience reports that provide insight into the issues, challenges, and solutions common in small settings.

In the Workshop Highlights section of this report there are summaries of the breakout sessions that were held on both days of the workshop. The raw notes are included in the appendix of this report in Mind Map format. On each day, the SEI suggested one topic and

solicited additional topics from the participants. In the case of one topic, return on investment for process improvement in small settings, a subset of the participants from the first session decided to continue that session on the second day to explore additional ideas.

Table 1 shows the breakout topics that were covered each day. Each breakout session was facilitated and notes were taken on flip charts. At the end of each day, a report was given for each session to the larger group.

Table 1: IRW Breakout Topics

Day 1	Day 2
A: Cost/Benefit Models for Small Settings	A: Economical Process Improvement Infrastructure
B: Success Stories of SPI in Small Settings	B: Cost/Benefit Models for Small Settings 2
C: Regional SPI Support Initiatives–Plans & Experiences	C: Further Research
	D: ISO SC7/Working Group 24 Inputs from Participants

2 Research Directions

2.1 Addressing Infrastructure Issues in Very Small Settings

Author

Oscar A. Mondragon

Abstract

A process improvement (PI) project based on a comprehensive reference model such as the Capability Maturity Model Integration (CMMI) requires additional effort and time to interpret the model. Because it is common that small companies have budget and schedule constraints, the challenges to successfully carry out a PI project based on CMMI are considerable. This paper addresses one of these challenges, building the infrastructure for the PI project in very small settings, and discusses how these challenges are greater in very small settings (i.e., relevant aspects such as the cash flow of the company, people skills, and project sizes). The IDEAL model from the SEI recommends at least one full-time employee as project leader (PL). The PL should carry out activities from the Management Steering Group, Engineering Process Group, and Technical Working Group. As a result, a question should be answered: From what part of the organization shall the PL be drawn? There are three scenarios: (1) a top executive, (2) a respected project manager or technical expert, or (3) a practitioner as leader of the PI effort. Finally, the author would like the research community to address the problem of biased judgment in very small settings.

Introduction

Process improvement (PI) efforts require investments such as budget, time, and organizational resources similar to any other project. Activities such as planning, task assignment, training, and developing schedules are also needed. A PI project further requires sponsorship from top executives and a good communication scheme to motivate the individuals involved in this continuous endeavor. Furthermore, when the PI project is based on a comprehensive reference model such as the CMMI [SEI 01], the effort and time required to interpret the model increases significantly. Because it is common that small companies have budget and schedule constraints, the challenges to successfully carry out a PI project based on CMMI are considerable. This paper addresses one of these challenges: how to build the infrastructure for the PI project? In addition, the paper points out how these challenges

are greater in companies or organizations smaller than the size suggested by the IDEAL model [McFeeley 96]. These settings can be defined as *very small settings*.

The paper also discusses the activities I performed, a description of the environment in which the activities take place, the approaches to build the PI infrastructure, the strengths and weaknesses of the approaches, and my view on the most important topics for the research community to address in the future.

My Activities in Small Settings

As a consultant for process improvement projects, I am helping several very small companies with training in different knowledge areas related to CMMI's Process Areas (PA), and I am conducting gap analyses. I am also helping executives defining the scope, effort, and infrastructure of the PI project and setting the priority for PA's implementation. Additionally, I am assisting technical working groups (TWG) members in process definition and process implementation in pilot projects.

The Environment

The term *small setting* has been defined as an organization or company of fewer than approximately 100 people, and a project of fewer than approximately 20 people [SEI 04]. However, the companies I am working with average between 10-20 people, and the project size ranges from 1-6 people. There is a huge difference between a company with 10 employees and a company with 100 employees. For this reason, we will refer to organizations or companies with less than 25 people and a project of fewer than approximately 6 people as a *very small setting*. The resources in these companies are typically very limited. As mentioned in the Software Engineering Institute Web site for small settings, a major aspect to be considered in these environments is that the amount of resources used to support a process improvement effort would be a large percentage of an organization's operating budget, [SEI 04].

In addition to the above aspect, there are three more aspects about the environment that I consider relevant in very small settings:

- **The cash flow of the company:** The cash flow is essential because it will allow removing people from the operation either partially or fully. In very small settings, more than 100% of the technical expert's time may already be assigned. Many very small companies are level 1, and these companies are not proficient at performing scope and effort estimations. Therefore, if people are scheduled for 50% of their time, they may end up working in their "free" time on the PI tasks, or, even worse; these tasks may not get any attention.
- **The people skills:** People skills and knowledge directly affect the speed at which technology adoption takes place. In this environment, there are not many individuals with a university degree. Because the majority of the employees do not have a degree, they may not have developed strong analytical thinking skills. These individuals frequently

encounter more difficulties when adopting new technology. A more complete solution, such as process guides, training, a change request system, and tool configuration support is required to alleviate the resistance to change.

- **The project size:** The project size directly affects the amount of communication, documentation, and management skills that are needed. On one hand, there are big projects where software engineering (SE) methods and techniques are essential to deliver a quality working product. On the other hand, there are small projects where a working product can be delivered without documentation, and with poor communication, and a lack of management skills, or with all of the above. Individuals who have only had the opportunity to work in very small settings may have problems visualizing the need and advantages of good SE practices. This lack of understanding hinders motivation and increases the resistance for change. Unfortunately, these individuals may only feel the effect of the workload and regret the good practices' overhead.

Models, Techniques, and Approaches for PI

The information discussed in this section is based on my experience of assisting very small companies that have adopted the staged representation of the CMMI [SEI 01] and are pursuing level 2 of the model. The approaches discussed in this paper are the way in which the PI infrastructure may be built. First, the approach suggested by the IDEAL model [McFeeley 02] is briefly described and then three other approaches are described.

The PI infrastructure has three groups [McFeeley 02]: the management steering group (MSG), the engineering process group (EPG) and, the technical working group (TWG). The membership for the MSG is drawn from senior executives. MSG responsibilities include aligning the PI goals to the business needs, demonstrating sponsorship to the PI project, allocating resources, and providing guidance by monitoring and offering corrective actions to the PI program. IDEAL does not provide time estimation for the MSG members. The EPG membership is drawn from respected project managers and practitioners. The recommended size of the group is 3% of the organization's developers. The EPG ensures coordination of the PI effort throughout the organization by providing process consultation, theme expertise, training, and PI assessments. The TWG membership is drawn from line managers and practitioners from the organization areas affected by the PI project. TWG members assign approximately 20% of their time. For small settings, IDEAL recommends at least one full-time employee.

Satisfying the one full-time employee requirement recommended by IDEAL in very small companies is a challenge. The first question to be answered is from what part of the organization should the PI leader be drawn? The PI leader may either be a top executive, manager, or a practitioner. The following points provide a case for each of the above options:

- **A top executive as leader of the PI effort:** A top executive is a natural candidate for the MSG membership. The strengths of this approach are that the PI effort is sponsored, guided, and controlled from the top of the organization. A top executive with an engineering background has good insight into the company's processes. Top executive

involvement alleviates the selection of pilot projects, training, and process implementation. There are some weaknesses to this approach. The top executive is probably also doing EPG and TWG activities. The company loses direction because the top executive is involved with the operation of the PI project, for instance, documenting processes' current and desired state, training personnel, developing guides, selecting and setting up tools, and implementing pilots. Important activities such as retailing projects and services may be diminished, putting at risk a healthy cash flow for the company. If the cash flow of the company gets affected, the top executive will be forced to get back to his company-leading responsibilities.

- **A respected project manager or technical expert as leader of the PI effort:** It is common in very small settings that the technical expert is the respected project manager. This person may participate as an analyst, architect, or project manager. The strengths of this approach are many. This person is a natural candidate for the EPG because he or she knows and understands current processes. This approach facilitates documentation of processes' current and desired state. There are some disadvantages to this approach as well. It requires that the PI leader continuously monitors high management's commitment. The project manager may not have the authority to create new roles, make organizational changes, and commit other projects to follow pilot implementation. The technical expert may already be overloaded; therefore, this person may consider the PI effort only in his or her spare time. If the technical expert commits 100% to the PI effort, operational performance may be affected and project schedules may be delayed. If the cash flow of the company gets affected, the technical expert will be forced to get back to his or her responsibilities. A major problem in very small settings is that it is not viable to substitute the technical expert.
- **A practitioner as leader of the PI effort:** If the practitioner is assigned the PI leader, there is a high possibility of allowing the assigned practitioner to commit 100% of his or her time to the PI effort. This approach is promising with the appropriate authority for the PI leader and the commitment and support of the top management for the PI project. The disadvantages to this approach are that the practitioner may only know a few process areas or part of the process areas and may lack management skills. The practitioner may not be respected by all in the organization, especially by managers who may be overloaded and reluctant to change. The practitioner may have problems getting: a) process area experts to participate in process documentation, b) managers to participate in pilot implementations, and c) personnel to adhere to new processes. In this approach, top management must provide an extensive monitoring of the PI leader's activities.

Based on my experience among several companies, the third approach (a practitioner as a leader of the PI effort) is currently providing the best results. This approach, however, requires full commitment from top management, support and respect from managers to the PI leader, support from experts to document current and desired processes, written and oral communication skills from the PI leader, and the commitment from managers and practitioners to implement pilots. The consultant also helps them set up the PI infrastructure, assists them with model interpretation, and provides them with good practices expertise.

Author's View of Important Topics

Because of the nature of very small settings, I would like the research community to address the following issues:

1. the problem of biased judgment
2. providing resources for new practitioners in the PI effort (e.g., developing guidelines in model interpretation in very small settings, templates for self assessment, staffing the PI infrastructure, and affordable training)

The Problem of Biased Judgment

Because of the reduced number of personnel, multiple roles may be played by the same person. This condition will also be present for the process improvement project. One person may do several or all of the following activities: requirements gathering, requirements analysis, project planning, project monitoring and controlling, design, programming, testing, and quality assurance. Some compromises may be forced to ignore or diminish some of the activities mentioned above due to the problem of biased judgment. The objectivity of performing reviews, testing, and quality assurance activities may be compromised in this situation. Similarly, the PI leader may play MSG, EPG, and TWG roles while he or she also has to meet management responsibilities or business goals with strict deadlines. When the top executive is the leader of the PI, priorities and guidelines to provide status effort must be established.

The objectivity of the reviews, the monitoring of the activities, and the status with high management must be guaranteed. Guidelines to avoid or mitigate a biased judgment must be developed.

Providing Resources for New Practitioners in the PI effort

Carrying out a process improvement project is a challenging venture, especially in very small settings. There is a need to motivate very small companies to do process improvement, and, more specifically, there is a need to keep them committed to a process improvement project. Some major problems are the limited skills and resources and the perception that the reference model, such as CMMI, entails a substantial overhead. A common guideline to interpret CMMI is "make what makes sense." The problem is that, for many new practitioners in very small settings, CMMI is only for large settings. Guidelines to adopt and implement the essence of CMMI process areas should be tailored for very small settings. Although very small settings may not be able to get their own CMMI consultant, they still need to perform self-assessments. Templates and online help should be developed for self-assessments.

Acknowledgements

I would like to thank my colleagues Jose E. Perez and Mario Rodriguez at the ESICenter Mexico for their valuable comments. I also want to thank Nelly Delgado at Microsoft for her time and insightful comments.

References

- [McFeeley 96] McFeeley, Robert. *IDEAL: A User's Guide for Software Process Improvement* (CMU/SEI-96-HB-001, ADA305472). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1996.
<http://www.sei.cmu.edu/publications/documents/96.reports/96.hb.001.html>
- [SEI 01] CMMI Product Team. *CMMI for Systems Engineering/Software Engineering, Version 1.1, Staged Representation* (CMU/SEI-2002-TR-002, ADA339224). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2002.
<http://www.sei.cmu.edu/publications/documents/02.reports/02tr002.html>
- [SEI 05] Software Engineering Institute, Carnegie Mellon University. *CMMI in Small Settings*.
<http://www.sei.cmu.edu/cmmi/adoption/cmmismall.html> (2005).

Biography

Oscar A. Mondragon

ESICenter Mexico

oscar.mondragon@itesm.mx

Dr. Oscar A. Mondragon has been at ESICenter Mexico since May 2004, working as a consultant for software quality models and process improvement methods. He worked as a research assistant at the System and Software Engineering Affinity Laboratory at the University of Texas at El Paso (UTEP) from 1996 to 1999. He had two research appointments at the Real-time System Lab at the University of Pennsylvania and one at AT&T labs at Middletown, NJ. Mondragon worked at Instituto Tecnológico Estudios Superiores Monterrey (ITESM) as a research professor leading the Mexican part of the project “Modularity Capability Attached Processor” founded by NSF and CONACYT. He has customized seminars for Mexican companies in quality assurance, independent verification and validation, and software testing and taught undergraduate courses in software engineering. Mondragon has a PhD in computer engineering from UTEP (May 2004) and a master’s degree in computer science with specialization in software engineering from California State University at Sacramento (July 1996). He did his master’s and doctoral work as a Fulbright and CONACYT grantee, respectively. Mondragon is an IEEE member and has been a

Certified Software Developer Professional (CSDP) since January 2002. His areas of interest are formal specification languages and requirements engineering.

2.2 A Multi-Method Evaluation of the Practices of Small Software Projects

Author

Khaled El Emam

Abstract

Benchmark models for process improvement (such as the SW-CMM and CMMI) are based on structuring and documenting the best practices of contemporary organizations. They have had a profound impact on transitioning good software engineering practices into industry and government. However, these models did not always address the realities of small projects. To have a similar impact on small projects, we need to first document and evaluate the practices of these small projects. This paper presents the results of two descriptive studies whose aim was to understand the practices that small projects follow and which ones are successful. The first study was a qualitative interview study of Canadian companies. The second was an international survey of information systems projects. The findings indicate which agile and iterative practices these projects used and how the projects customized them. The findings also quantify the success rates of small projects and identify some contrasts between small and large projects.

Research Methods

The following are brief overviews of the methods for the two studies.

Qualitative Study

We followed an exploratory, descriptive research method, namely *grounded theory* [Glaser 67]. This involved an iterative process of data collection and analysis to identify practices, and the integration of these findings with the literature. The result is a summary of practices that is grounded in the realities of small projects.

Project managers, CEOs, VPs of engineering, and developers were interviewed for one to two hours each. Some of the interviewees discussed the practices in more than one project in their organization.

Survey

The survey was conducted in 2005 with a convenience sample of 232 clients of the Cutter Consortium. These clients were contacted and asked to complete a web questionnaire. Respondents were mostly from the US (37%), followed by Australia (11%), India (10%), and the UK (8%). The respondents worked mostly in the financial services sector (22%) and the computer consulting sector (21%). Approximately half of the projects had their first release

within 9 months of starting and approximately half had 10 people or less (developers and technical staff).

Results

The samples of small projects in both studies were from (1) small projects in large companies, and (2) small projects in small organizations. In some of the latter cases, the project studied was all that the company did. Therefore, we expect that the issues identified in our studies to span the two scenarios above.

The following is a summary of the findings across both studies. These are organized by theme:

Project cancellation. We found that 15% of projects overall were cancelled. There was no difference in cancellation rate between small and large projects.

Success criteria. We used five criteria for evaluating project success: user satisfaction, ability to meet schedule targets, ability to meet budget targets, quality, and productivity. These are similar to the criteria used in previous studies [El Emam 00a, El Emam 00b, Goldenson 95]. Approximately one third of the projects that were not cancelled failed on the following success criteria: user satisfaction, product quality, and staff productivity.¹ Approximately half of the projects that were not cancelled were not able to meet budget or schedule commitments. It would therefore seem that meeting budget and schedule commitments are the bigger problems for software projects.

Successful projects. We defined a successful project as one that did not fail on any of the five success criteria or that failed on only one. Almost half of the projects were considered successful.

Failed projects. Approximately one fifth of the projects were considered failures, in that they failed on 4 or more of the 5 success criteria.

Small project advantage. Smaller projects tended to perform better in their ability to meet schedule and budget targets and their productivity. There was no evidence that they performed better in terms of product quality and user satisfaction.

Adoption of agile practices. Many of the small projects covered by the interviews adopted some form of agile methods, but none adopted them in a pure form. For example, none adopted all of the practices of XP (which is the most popular agile method). The common pattern is for projects to adopt some practices and modify others to get them to work in their contexts.

¹ Failure is defined as a response of “poor” or “fair” on the success criterion 4-point Likert response scale.

High reliability requirements. Projects that had high reliability requirements did not use agile practices. These project managers did not believe that agile practices were capable of delivering high quality software.

Training. None of the projects that adopted agile practices received formal training or coaching. They usually learned about the practices from colleagues, articles, books, the web, and attending conferences.

Architecture. All projects, including those that adopted agile practices, emphasized the need for a documented and well understood architecture for the system. Those that tried to deploy agile practices without an architecture had rapid deterioration of the system structure over time (this may be related to the lack of refactoring—see below).

Refactoring. Few projects were able to refactor on a regular basis. They were either not able to do it or said that they did not have time. Refactoring was perceived as inefficient without an architecture because functions, services, and interfaces were constantly changing. Refactoring of data was seen as a difficult problem because it also involved live data migration.

User participation. We found a direct association between user participation and user satisfaction, ability to meet budget commitments, and productivity (after controlling for the project size effect). No relationship with the ability to meet schedule targets and product quality was found. A number of projects had difficulties with user participation. Sometimes it was difficult to get the right users to participate at the frequency that was needed. In other cases, there was no single user, so proxies (real or virtual) were created to compensate. Hostile users were mentioned, and these were individuals who refused to prioritize requirements or constantly shifted priorities. Some users got frustrated with refactoring iterations because they perceived they were not getting any value.

Pair programming. In the interviews we found that no small projects were using pair programming for all of their developers. Some had it as an optional practice. Difficulties were encountered with personality conflicts and awkward furniture arrangements (e.g., because the project did not have appropriate furniture, the pairs ended up sitting on one regular desk which was perceived as “too close” to be comfortable). In the survey we found no relationship between the adoption of pair programming and project size. Overall, the survey results indicated that sixteen percent of projects used pair programming “frequently” or “always.”

Test driven development (TDD). While it is recognized that TDD is advantageous, few small projects were able to implement TDD successfully. We found no quantitative relationship between the implementation of TDD and project success. Overall, 23% of projects responded that they implemented TDD at least “frequently.”

Peer reviews. None of the interviewed projects performed formal inspections, but a majority performed some kind of peer review. In the survey we found that 33% of the projects perform

inspections frequently or always (this included small and large projects). Although, it is reasonable to assume that the rigor of implementation of inspections in these projects would vary (i.e., not all were rigorous inspection processes with well-defined roles and data collection). Larger projects were more likely to perform inspections than smaller projects.

Influence of executives. In small companies, the executives had strong influence on the practices, and there is constant pressure to just deliver functionality. This is particularly true for executives who lack a software engineering background. In one organization the executives noted to the development manager that “if you don’t develop this feature now we will go bankrupt,” which was a pressure tactic to get the development manager to over-commit.

Planning. The schedules were determined by the clients, so the development team had little control over these. The budgets were fixed. Therefore, the main characteristic that the development team had control over was scope. Most projects used an iterative development process to enable them to manage scope. Negotiating variable scope with clients is not easy and unless there was good user participation, this did not work well.

Duration of iterations. We did not find quantitative evidence of a relationship between the duration of iterations and project success. However, from the interviews it was found that short iterations (e.g., 3-4 weeks) exhausted the development team and did not give them enough time to do proper analysis and design of the features in the next iteration. Projects with an iteration cycle of 3 months or more did not have that problem. The actual duration of the iterations was a business decision driven largely by the frequency of changes.

Open source. Many of the development projects used open source development tools (such as IDEs and scripting tools). The main driver for that decision was to minimize development costs.

Investors and software engineering.² Investors in small companies rarely paid attention to software engineering practices as a decision making criterion. The most important factors were the perceived capability of the executive team, the market, and the technology. Whether the software engineering practices were strong did not, historically, differentiate between successful companies and those that fail. Therefore, this was not considered a critical issue (the exception being when it was a requirement for doing business—such as in regulated industries).

Limitations

The main limitation of these studies was that neither of them collected data from a random sample of small projects, which limits the *generalizability* of the results. In both cases convenience samples were used. However, the findings of both studies were consistent, and the qualitative study was grounded in the literature (which reflects the experiences of a much

² This point was based on a series of discussions the author had with partners in VC firms in 2005.

wider community). We would argue that the samples in both studies would have to be biased similarly, and in concordance with a publication bias, to threaten the validity of the findings.

A second limitation to the findings is that they are based on perceptions rather than objective measurements of process and outcomes. While we did not witness any explicit attempts at systematic bias, it is plausible that respondents oriented their answers to make themselves look good (for example, respondents may inflate user participation if the project failed and deflate it if the project succeeded in order to take less/more of the credit respectively [Hawk 90]).

Acknowledgements

I would like to thank Hakan Erdogmus of the National Research Council of Canada, and Terry Rout of Griffith University in Brisbane for reviewing an earlier version of this position paper and providing comments.

References/Bibliography

- [Brewer 89]** Brewer, John & Hunter, Albert. *Multimethod Research: A Synthesis of Styles*. Newbury Park, CA: Sage Publications, 1989.
- [EI Emam 00a]** El Emam, Khaled & Birk, Andreas. "Validating the ISO/IEC 15504 Measures of Software Development Process Capability." *Journal of Systems and Software* 51 (2000): 119-149.
- [EI Emam 00b]** El Emam, Khaled & Birk, Andreas. "Validating the ISO/IEC 15504 Measures of Software Requirements Analysis Process Capability," *IEEE Transactions on Software Engineering* 26 (2000) 541-566.
- [EI Emam 03]** El Emam, Khaled. "Finding Success in Small Software Projects." *Cutter Consortium* 4, 11 (2003).
- [EI Emam 05a]** El Emam, Khaled. "Does Project Size Matter?" *Cutter Consortium* 6, 14 (2005).
- [EI Emam 05b]** El Emam, Khaled. "User Participation in Software Projects," *Cutter Consortium* 6, 19 (2005).
- [EI Emam 05c]** El Emam, Khaled. "Software Practices and Project Success," *Cutter Consortium* 6, 17 (2005).
- [Glaser 67]** Glaser, Barney & Strauss, Anselm. *The Discovery of Grounded Theory: Strategies for Qualitative Research*. New York: Aldine de Gruyter, 1967.

- [Goldenson 95]** Goldenson, Dennis & Herbsleb, James. *After the Appraisal: A Systematic Survey of Process Improvement, Its Benefits, and Factors that Influence Success* (CMU/SEI-95-TR-009, ADA302225). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1995.
<http://www.sei.cmu.edu/publications/documents/95.reports/95.tr.009.html>
- [Hawk 90]** Hawk, S. and Aldag, R. "Measurement biases in user involvement research," *OMEGA: International Journal of Management Science* 18 (1990): 605-613.

Biography

Khaled El Emam

University of Ottawa

kelemam@uottawa.ca

Dr. Khaled El Emam is a senior scientist at the Children's Hospital of Eastern Ontario Research Institute, where he is leading the eHealth research program. He is also an associate professor at the University of Ottawa, Faculty of Medicine, and a Canada Research Chair in Electronic Health Information. In addition, Khaled is the chief scientist at TrialStat Corporation, a software company that develops applications for clinical research and drug development, and a senior consultant with Cutter Consortium's Agile Software Development & Project Management Practice. Previously, El Emam was a senior research officer at the National Research Council of Canada, where he was the technical lead of the Software Quality Laboratory, and prior to that he was head of the Quantitative Methods Group at the Fraunhofer Institute for Experimental Software Engineering in Kaiserslautern, Germany. In 2003 and 2004, El Emam was ranked as the top systems and software engineering scholar worldwide by the Journal of Systems and Software based on his research on measurement and quality evaluation and improvement, and ranked second in 2002 and 2005. Currently, he is a visiting professor at the Center for Global eHealth Innovation at the University of Toronto (University Health Network) and at the School of Business at Korea University in Seoul. He holds a PhD from the Department of Electrical and Electronics Engineering, King's College, at the University of London (UK).

2.3 Barriers to Adoption of the CMMI Process Model in Small Settings

Author

Gene Kelly

Abstract

The purpose of this paper is to examine the barriers to adoption of the CMMI process model in small settings. The paper is focused on a case study analysis that describes the experience of a small business in process development and certification to the ISO 9001:2000 standard. The evaluation process for implementation of the CMMI model, and the primary barriers to implementation, will be described in a manner that can be generalized to other small settings. The relevance of this research serves three key purposes. First, this work supports the objectives of the IPRC by contributing key perspectives to the R&D roadmap from small work groups and businesses. Second, it provides applied research that can be generalized to a variety of small settings. The third key purpose of this research centers on understanding why the CMMI model would not be utilized by small businesses or work groups who have invested in ISO 9001 certification. If the barriers to the implementation of CMMI can be evaluated and eliminated, the impact for these organizations would be reduced time and cost of implementation, leading to improved maturity of their software processes and enhanced prospects for growth.

Introduction

The purpose of this paper is to examine the barriers to adoption of the CMMI model in small businesses. The paper will be focused on a case study that describes the experience of CertTech, L.L.C., a 20 person company located in the suburban Kansas City area. The company's primary clients are in the aerospace industry; CertTech provides software verification and validation services on FAA-regulated avionics systems used on commercial aircraft. Other customers of the company are in the ground-based avionics and rail transportation industries.

The primary objective of the case study analysis will be to describe the CertTech experience in process development and discuss the preparation and subsequent certification to the ISO 9001:2000 standard. The evaluation process for implementation of the CMMI model, and the primary barriers to implementation, will be described in a manner that can be generalized to other small business settings.

The relevance of this research serves three key purposes. First, this work supports the objectives of the IPRC by contributing key perspectives to the R&D roadmap from the perspective of industry, particularly small business. This input is critical if the IPRC is to serve as a technology scout to develop a comprehensive view of the future landscape of

process needs in industry and to generate an accurate forecast of technology possibilities and an agenda for future research directions for the next 5-10 years. Second, it provides applied research that can be generalized to a variety of small business settings, which will help improve the maturity of the software processes used in industry.

Adoption of ISO 9001 as a process model has been widespread, with over 40,000 companies certified in the US. The subset of those companies that are software-related small businesses need to leverage investments where possible. If they have invested in ISO 9001 compliance, which is estimated at \$25,000 in training, external audits, and registration costs, it would be of significant benefit to small businesses to take the baseline provided by ISO 9001 and improve the maturity of their processes through application of the CMMI model.

The third key purpose of this research centers on understanding why the CMMI model would not be utilized by these businesses. If the barriers to implementation of CMMI can be evaluated and eliminated, the impact for these businesses would be reduced time and cost of implementation and improved prospects for growth in new customers, who increasingly prefer suppliers who are CMMI certified.

Case Study: Activities in Small Settings

CertTech was started in 1999 as a supplier to Honeywell Aerospace. The company has matured from a one-person operation to one that employs 20 people. The level of process maturity has also increased, along with the size of the company. As a software V&V contractor to Honeywell, the quality system requirements are significant. In the beginning, CertTech relied on the Honeywell documentation and developed test methodologies based on those requirements. However, that approach proved insufficient if the company was to grow and attract additional customers.

ISO 9001 compliance was seen as an achievable goal, and quality manual and key system processes were generated. After the establishment of the basics of the quality system, CertTech decided to pursue ISO registration. The processes were reevaluated through a gap analysis audit, and gap closure plans were implemented. The ISO registration audit occurred in October 2004, and registration occurred in December 2004.

CMMI certification was evaluated after completion of ISO registration. CertTech held a teleconference with a CMMI consulting company through a referral. After reviewing the state of the QMS, it was estimated that the gap analysis would be five days at \$20,000 and the SCAMPI certification audit would be eight days at \$30,000 for a total of \$50,000. In addition, there were concerns that CMMI implementation would be not be aligned with the CertTech business model and would create extra infrastructure and activities. CertTech decided not to pursue CMMI certification unless there was a clear customer requirement.

Models, Techniques, and Approaches

The primary models to be used are the ISO 9001:2000 standard and CMMI, Version 1.1. In addition, CMMI in Small Settings Toolkit Repository from AMRDEC SED Pilot Sites; DRAFT 14 was evaluated for use.

The adoption of these models is critical to establishing a process infrastructure and baseline. Both the ISO 9001 and CMMI models have strengths and weaknesses when applied in small businesses.

Strengths and Weaknesses

ISO 9001:2000

The key strength is its applicability to a wide variety of industries, which leads to another strength, which is its worldwide adoption. However, the ability to be generalized is a key weakness because it is difficult to apply the standard to a software company and especially a service-related company that does not produce software. Specifically, the ISO standard appears to be geared more toward manufacturing than service-related businesses. This required CertTech to develop process-tailoring processes that would enable it to serve multiple customers in the quality system.

CMMI

The key advantage of CMMI is that it was developed for use in the software industry. However, its ability to be used in small businesses is hampered by additional requirements that are not in ISO. For example, GP 3.1 Establish a Defined Process has no ISO equivalent and would have to be addressed in the process documentation. Other project management elements that appear in the CMMI model appear to be very burdensome for a small business due to the numerous reviews and reporting requirements that are contained in the model.

Future Research Topics

Improvement in the software process maturity of small businesses has significant potential impact. Many of the new programs in the defense and aerospace industries are increasingly reliant on software to provide functionality. For example, synthetic instrumentation, which replaces traditional hardware stimulus and measurement functions with a mix of hardware and software representations, is emerging as a key technology with both civilian and military applications. In addition, large programs, such as the US Army Future Combat Systems, rely on software-defined communication and data analysis to provide command and control capabilities to warfighters. Given this environment, the following questions warrant further research:

- How many small businesses are involved in software engineering activities such as development and/or testing? A search of *NAICS: 541511 Custom Computer Programming Services* could provide an initial data source.

- Of those businesses, how many are ISO 9001:2000 compliant and/or certified?
- Following ISO certification, have the businesses evaluated CMMI? If not, why?

This is a research area that could use mixed research design approach. A survey instrument could be developed and distributed to test hypotheses around the barriers to CMMI. Is it lack of awareness, cost, time, or other factors? In addition, follow-up interviews or focus groups could be held to provide qualitative information in addition to the quantitative data from the survey.

Summary

The future US defense establishment will increase its reliance on safety critical software. The IPRC can play a key role in helping develop the process infrastructure to enable small businesses to contribute to the effort. However, understanding why small businesses do not utilize the CMMI framework is important if the SEI and IPRC are to improve the adoption of the CMMI model among small businesses. Obtaining this information should be a key part of the roadmap for future process needs.

Bibliography

- [Garcia 04]** Garcia, Suzanne; Cepeda, Sandra; Miluk, Gene; & Staley, Mary Jo. *CMMI in Small Settings Toolkit Repository from AMRDEC SED Pilot Sites; DRAFT 14*. <http://www.sei.cmu.edu/tp/publications/toolkit/> (2004).
- [Granieri 04]** Granieri, Mike. "Synthetic instrumentation: An emerging technology (Part I)." *RF Design* (Feb. 2004).
- [ISO 05]** ISO 2000. *ISO 9001:2000, Quality management systems—requirements*. <http://www.iso.org/iso/en/ISOOnline.frontpage> (2005).
- [SEI 02]** CMMI Product Team. CMMI for Systems Engineering/Software Engineering/Integrated Product and Process Development, Version 1.1, Staged Representation (CMMI-SE/SW/IPPD, V1.1, Staged) (CMU/SEI-2002-TR-004, ADA339221). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2002. <http://www.sei.cmu.edu/publications/documents/02.reports/02tr004.html>.
- [Shirreffs 03]** Shirreffs, Allison. "Making sense and cents of ISO certification," *Atlanta Business Chronicle* (Nov. 28, 2003).

Biography

Gene Kelly

CertTech, LLC

gkelly@certtech.com

Gene Kelly is the director of New Business Development for CertTech, LLC. He is also responsible for compliance of the CertTech quality management system, and he played a key role in the ISO 9001 certification effort. He has held site quality leadership roles in both the aerospace and medical device industries, in addition to management positions in program management and business development. Kelly is a member of the American Society for Quality and is a Certified Quality Manager. He is also a member of the Joint RTCA Special Committee 205 and EUROCAE Working Group 71 for Software Considerations in Aeronautical Systems and serves as the US Secretary for SG3–Tool Qualification. In addition, he is a member of the AAMI Medical Device Software Committee. His degrees include a BS in electrical engineering from the University of Missouri-Columbia and an MBA from the University of Missouri-Kansas City, where he is currently a PhD student in the Bloch School of Business and Public Administration.

2.4 Measuring Performance Results in Small Settings: How do you do it and what matters most?

Authors

Dennis Goldenson, Terry Rout, and Angela Tuffley

Abstract

This paper addresses three related issues. First of all, how can small organizations and projects effectively monitor their progress and evaluate their performance in meeting their business goals without busting the bank? Second, what factors account for the degree of success that they can accomplish? And third, are there aspects of CMMI that are difficult or inappropriate to apply in small organizational settings? We briefly describe our thoughts about these three issues along with a synopsis of our ongoing and proposed work in this area.

With Respect to Performance Measurement

Small organizational units are just as likely to be confronted by demands for credible evidence about their ability to deliver quality products on time and on budget as are large, multinational organizations. Similarly, managers in small settings are equally or even more likely than their counterparts in larger organizational units to have to make well-founded business decisions about process improvement and technology adoption, and have the wisdom of taking new business opportunities.

Where the small and large organizations differ is in the resources they have available to devote to a serious measurement program. It is a good idea to start small in any measurement program. Measurement objectives always should be closely aligned with high priority business goals and technical problems if the measurement results are to be pertinent, useful, or used to inform technical and business decisions. This is even more important in small organizational settings.

But, from where do the necessary resources come? Consortia of small organizations are being developed in a number of areas throughout the world, and consulting organizations are beginning to offer products and services designed expressly for small organizational settings. Our task is to better understand the measurement needs of small projects and organizations and to provide guidance for them.

The Conditions of Failure and Success

Of course, a major purpose of process improvement is to increase the likelihood that organizational units will achieve or exceed their commitments to deliver quality products on time and on budget. Yet, there still are few broadly based studies that compare the

experiences among large numbers of organizational units with respect to the successes and failures of their process improvement efforts. Even fewer focus on small organizational settings.

While there are notable case studies that address organizational and cultural barriers to successful process improvement, we also need more broadly based samples to be able to attribute our results with confidence to process performance *versus* other factors or unintended measurement effects. In addition to measures of process performance that are pertinent to small organizational settings, we need measures of organizational, project, and product context (e.g., sector, domain, technologies, criticality, or *precedentedness*). Our task is to define and capture such measures and link them with independent measures of quality and other aspects of organizational unit performance.

The Pertinence of CMMI

Ever since the development of the SW-CMM, people have questioned the applicability of such models in small organizational settings. The same is so for CMMI. Similar to much of the advocacy for agile methods, and to our own admonition above to start small with respect to measurement, the arguments against relying too heavily on CMMI in small settings often focus on limited resources.

Perhaps more to the point, others have argued that certain processes advocated by CMMI do not scale well to small organizational settings. In particular, work in Australia has raised questions about Process and Product Quality Assurance (PPQA) in that regard. One can argue that the practices followed by capable organizational units in small settings do in fact map well to the process areas, even if practitioners fail to recognize the one-to-many linkages between their own practices and CMMI. Better interpretive guidance for process appraisal and improvement may be the solution. Regardless, our task also is to critically analyze the applicability of the process areas as currently written.

Our Study

As part of a proposal for funding by the Australian Research Council and our ongoing work at the SEI on benchmarking the performance results of CMMI-based process improvement, we are gathering and analyzing data with respect to all three issues that we raised above. Beginning with focus groups and survey methods, we are querying practitioners about the successes and challenges that they have encountered in measuring their performance results, their adherence to CMMI based process improvements, and their insights about other factors that may account for their varying successes and difficulties. We also are working with selected industry partners in Australia, the United States, and elsewhere as they develop and enhance their enterprise performance measurement programs. Our end goals include the development of measures and measurement procedures that are appropriate for use in small projects and organizations, both for internal process improvement and for wider community benchmarking.

Biographies

Dennis R. Goldenson
Software Engineering Institute
Carnegie Mellon University
dg@sei.cmu.edu

Dennis R. Goldenson is a senior member of the technical staff at the Software Engineering Institute (SEI) in Pittsburgh, Pennsylvania, USA. A principal author of the Measurement and Analysis Process Area for Capability Maturity Model Integrated (CMMI) models, Dr. Goldenson is the technical lead for the SEI's empirical investigations about the impact of CMMI-based process improvement. He previously served as co-lead of test and evaluation for CMMI, and he was the international trials coordinator for the SPICE project in support of ISO/IEC 15504.

Goldenson came to the SEI in 1990 after teaching at Carnegie Mellon University since 1982. He has published numerous papers and made many presentations. His work focuses on improving the practice of measurement and analysis, the improvement of process models and appraisal methods, and the impact and transition of process improvement and other engineering technologies. His related interests are in survey research, experimental design, the visual display of quantitative information, and tools to support collaborative processes.

Terence P. Rout
Software Quality Institute
Griffith University
t.rout@griffith.edu.au

Terence P. Rout is an associate professor in the School of Computing and Information Technology and has served as acting director of the Software Quality Institute at Griffith University in Brisbane, Queensland, Australia. He has contributed numerous publications in the field of process assessment and improvement, and he has been an invited speaker on frequent occasions. Rout has broad experience as a leader of process assessments and has advised many companies on the implementation of effective assessment-based improvement programs.

The overall project editor for ISO 15504—Software Process Assessment, Rout has led the Australian effort in the SPICE Project since its inception, developing an extensive network of industry participants in the evaluation of approaches to process assessment. His contributions have been acknowledged in a series of achievement awards from the international Software Engineering Standards Committee. Rout led the development of the leading international training course for assessors using ISO 15504, and he has defined a new assessment method for the rapid evaluation of software process capability. In recent years, he has led research into improvement based upon the Capability Maturity Model Integration models, and he serves as a charter member of the International Process Research Consortium (IPRC).

Angela Tuffley
Software Quality Institute
Griffith University
a.tuffley@griffith.edu.au

Angela Tuffley is a senior consultant in the Software Quality Institute at Griffith University in Brisbane, Queensland, Australia. An early participant in the SPICE project in support of the development of ISO/IEC 15504, the international standard for process assessment and improvement, she became an Experience Qualified SPICE Assessor and co-developed and delivered the SPICE Assessor Training Course. She also assisted in the development of the RAPID Assessment model and method, a one day, ISO/IEC 15504 conformant, low cost assessment of limited scope suitable for small to medium enterprises.

Tuffley is an SEI-authorized SCAMPI Lead Appraiser and instructor for the Introduction to CMMI, Intermediate Concepts of CMMI, and CMMI Instructor Training courses. She participated as an appraisal team member in three important CMMI pilot appraisals sponsored by the Australian Defence Materiel Organisation. In addition, she has provided technical expertise for detailed mapping of CMMI to ISO/IEC 15504 and ISO/IEC 12207, and she participated in a detailed mapping of ISO9001 to CMMI with the Australian Department of Defence. Ms. Tuffley is the co-author of publications covering process improvement in small organizations, benchmarking, cost effective assessment, and the mapping exercise.

2.5 Results of a Field Study of CMMI for Small Settings Using Rapid Applied Ethnography³

Author

Gene Miluk

Abstract

Disciplined development methods and the CMMI have not generally been applied to small to medium enterprises (SME). Little is known about the need for, and efficacy of, applying this technology to this community. This research is an exploratory investigation designed to gather and analyze data indicating whether disciplined development methods and the CMMI for small to medium enterprises (CMMI-SME) are compatible with the business needs, culture, and environment of SMEs, as well as their technical and organizational culture. The research questions are divided into three categories: need, technical feasibility, and organizational compatibility.

While traditional ethnography would be too time consuming and expensive for this research, an approach combining elements of two successful ethnographically based methodologies was used. Rapid Applied Ethnography (RAE) combines elements of Rapid Appraisal, devised by anthropologists for expedited ethnography, and Applied Ethnography, a specialized form of ethnography utilized in new product development.

The data for this study was provided from a multi-year effort working with small enterprises by the Software Engineering Institute at Carnegie Mellon University. The study results focus on need, feasibility, and organizational compatibility.

Purpose of the Study

The problems surrounding the development of large scale systems are well documented. There has been an extensive amount of research, and a broad body of knowledge is extant on best system development practices for large software projects and organizations. The associated technical and organizational issues are clear to the large system development community. Accordingly, the purpose of this research is to gather and analyze relevant data in order to assess the need and applicability of disciplined system development methods within the SME (small to medium enterprise) community. The SME community's culture, as well as its business, technical and, organizational environment, were investigated and analyzed as it relates to (a) the need for disciplined system development processes, (b) the suitability and applicability of the CMMI to SMEs, and (c) the likely impacts the technology will have on the organization.

³ Special permission to use data, notes, interviews, lessons learned, and workshop output from TIDE[®] 2004 by Carnegie Mellon University, Software Engineering Institute.

Research Questions

Disciplined development methods and the CMMI have not generally been applied to the SME environment. Little is known about the need for, and efficacy of, applying this technology to this community. This research is an exploratory investigation designed to gather and analyze data on whether disciplined development methods and the CMMI for small enterprises (CMMI-SME) are compatible with the SME community. SME business needs, culture, competitive environment, and technical and organizational culture was investigated. This research builds on the experience large organizations have had with the adoption of disciplined system development methods.

The research questions are divided into three categories: need, technical feasibility, and organizational compatibility. The set of research questions include the following:

- What are the needs (perceived and actual) of SMEs who incorporate systems and software in their products and services? Does the CMMI address problems and opportunities important to the SME community?
- Is it feasible to implement the CMMI in the SME environment? How will the CMMI need to be adapted, developed, and packaged to effectively address SMEs?
- For organizational compatibility, what are the likely impacts to the existing managerial and technical social order within the organization? How compatible is the CMMI technology with the SME culture?

Methodology

This research is an exploratory qualitative study of the applicability of a specific technology (disciplined systems development practices, CMMI-SME) in a target population (the SME community). Qualitative research was chosen for this study for two primary reasons: 1) the data available from the TIDE program lends itself to this approach (the TIDE program worked with a small number of SMEs over an extended period of time and gained valuable insights); 2) the research questions for the study require a deep level of investigation and understanding to get at the core issues. This would best be accomplished using qualitative methods.

Qualitative research has a long and rich history with multiple adaptations and approaches evolving over time. John Creswell indicates that there are five traditions in qualitative research: biography, phenomenology, grounded theory, ethnography, and case study [Creswell 98]. While each of these approaches share the fundamentals of qualitative research, they do differ in their perspective and the type of issue they were designed to address.

The tradition chosen for this research is ethnography that is rooted in cultural anthropology. This tradition has an accomplished history: “early 20th-century anthropologists such as Boas, Malinowski, Radclif-Brown, and Meade and their studies of comparative cultures. Although they took the natural sciences as a model for research, they differed from traditional scientific

approaches through the firsthand collection of data of existing “primitive” cultures” [Creswell 98].

Ethnography studies a group’s behaviors, beliefs, and social system. Although ethnographic methods were originally developed by anthropologists, they have found application in the design, development, and adoption of computerized systems [Rogers 97, Klein 99, Myers 99, Millen 00, Spinuzzi 00]. In the process of applying ethnography to systems development and adoption, ethnography was found to be very valuable in supporting organizational change.

A related area that has benefited from the application of ethnographic methods has been new product development (NPD). Elizabeth Sanders, writing in Visions Magazine, the online magazine of the Product Development and Management Association (PDMA), advocates the use of “Applied Ethnography” as a valuable contributor to the new product development process:

Ethnography today is recognized as a new form of consumer research that is useful in uncovering and identifying emerging and unmet customer needs. Although it is not really a new scientific technique, the application of ethnography to the new product development (NPD) process is relatively recent” ... Applied ethnography can be useful throughout the new product development process. But it is probably most useful in the earliest stages of the NPD process especially the Fuzzy Front End. It can be used to explore the emerging and unmet needs for a particular target group [Sanders 02].

According to Sanders, applied ethnography is characterized by the following [Sanders 02]:

1. takes place in natural surroundings
2. is open to change and refinement throughout the process as new learning shapes future observations
3. combines a range of research methods, including observation and open-ended forms of inquiry
4. has a goal that is more likely exploratory than evaluative
5. aims at discovering the local person or native’s point of view, wherein the native may be a consumer or end-user

This study of the CMMI and the SME community can be viewed in the context of new product development, where the answers to the research questions will be used to formulate and package the CMMI to be desirable and easily adopted by the target SME population.

Traditional ethnography can be too time consuming and expensive to be practical for system design or new product development [Millen 00, Sanders 02]. Anthropologists have developed a non-traditional approach to expedited ethnography called “Rapid Appraisal” [Beebe 95, Driscoll 98]. Rapid Appraisal is built on three basic anthropological principles: 1) a systems

approach, 2) triangulation and 3) iterative data collection and analysis [Beebe 95]. This approach to rapid ethnography has been successfully used in system design [Millen 00].

Based on the research questions, the available data, and the likely audience for the results, applied ethnography incorporating elements of rapid appraisal was used for this research. This researcher has termed this combined approach Rapid Applied Ethnography (RAE). This methodological approach combines ethnography (looking at the beliefs, behaviors, and social system of the SME community) with the goals of new product development, technology adoption, and organizational change.

Research Design

The SEI normally works with large organizations to improve their software and systems development capability. The SEI through a project called “Technology Insertion, Demonstration, and Evaluation” (TIDE) Program had the opportunity to work with SMEs. The TIDE program was founded to encourage and assist small manufacturers in the adoption of commercially available software and information technology. TIDE is specifically focused on small manufacturers that supply goods and services to the national defense; however, much of the work of the TIDE program is broadly applicable to all small businesses. A complete description of the TIDE project can be accessed at <http://www.sei.cmu.edu/tide/>.

As a direct result of this project, information regarding the SME community and the issues associated with systems and software development were generated. This information, however, needed to be extracted, organized, and analyzed in order to be useful in addressing the above research questions. The raw data resided in multiple files in the TIDE project repository located at the SEI in Pittsburgh. The data consists of artifacts generated as a result of the SEI consultants working with the TIDE SMEs. The data is in the form of meeting minutes, interview data, workshop data, and the personal notes and observations of the TIDE consultants. A high level overview of the design of this research is presented in the following figure.

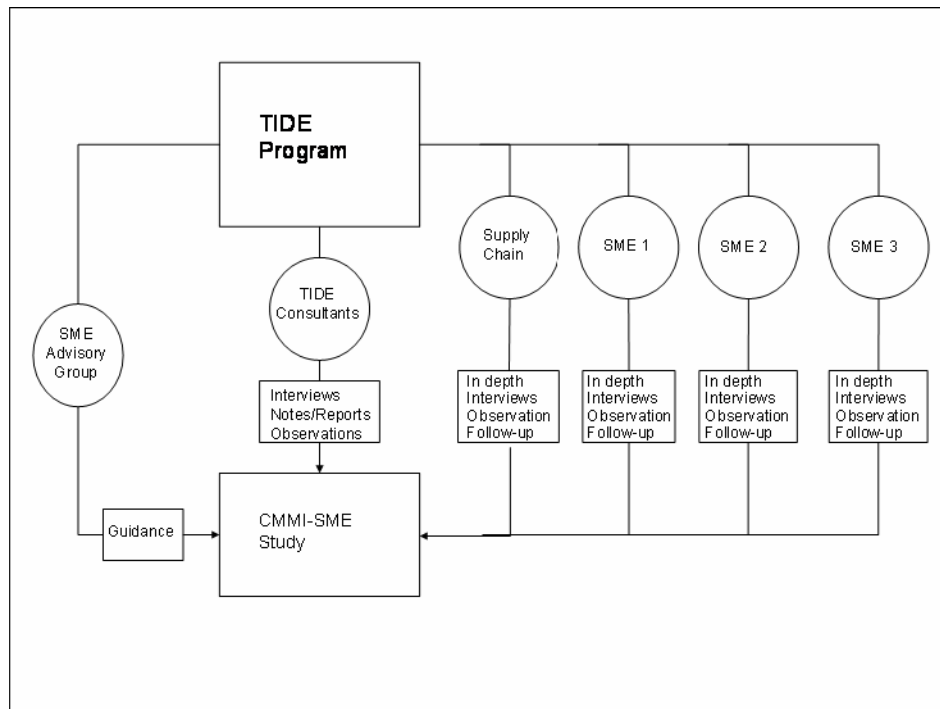


Figure 1: Research Design

The TIDE program provides the environment and structure for this research. As part of this study this researcher created an SME advisory group to provide guidance for the project and insight into the SME environment. Members of the advisory group were composed of SEI consultants and engineers on the TIDE program that had extensive first hand experience working for or with SMEs. This group included several engineers with SME experience and one member who spent several years in an executive position at a prominent SME. This group provided the systems perspective important for the effectiveness of rapid assessment. “Rapid appraisal should be based on what the participants in the system believe to be the critical elements, their relative importance, and how they relate to each other” [Beebe 95]. In addition to providing this critical systems perspective, the CMMI-SME advisory group was actively involved in the generation of the semi-structured interview scripts as well as helping to select SMEs for inclusion in this study.

Sources of Data

The goal of multiple sources of data from multiple perspectives was achieved by including data from five separate sources:

- TIDE consultants working with SMEs in the program were interviewed by the principal investigator about their experiences working with the SMEs. Notes were taken for later analysis. In addition, the principal investigator collected the output from their work with the SMEs. These included workshop outputs, meeting minutes, action plans, etc.
- The supply chain management and development arm of a major U.S. defense contractor was included for in-depth interviews and analysis by the principal investigator.

- Three representative SMEs were selected by the principal investigator and the SME advisory group for in-depth interviews, analysis, and follow up by the principal investigator.

These five sources of data provide a diverse set of perspectives. Each of the SMEs interviewed provide their unique experience. The supply chain management and development interviews provide a comprehensive perspective that summarizes the situation and experience gained from working with many SMEs over many years. The TIDE consultants bring a separate viewpoint from working closely with the TIDE participating SMEs as they worked with them to adopt new technology.

The sampling method used was purposeful sampling. This principal investigator worked with the advisory group to select appropriate SMEs. The target SME organizations were chosen because they are expected to represent the population of SMEs appropriate for CMMI-SME. Each of the SMEs is involved with some form of systems and software development.

Three companies were selected by the principal investigator and the SME advisory group based on the following criteria:

- they are typical of SMEs that might benefit from CMMI-SME.
- the SMEs are known to the expert group
- local (South West Pennsylvania) participants are preferred
- the SMEs are willing to cooperate with this investigation

Many SMEs work in supply chains for both the commercial and defense industries. These sometimes sophisticated arrangements create performance and coordination expectations that must be achieved by each partner in the chain in order for the overall project to be successful. The supply chain management and development office of a major U.S. defense contractor was selected by the principal investigator and the SME advisory group, and it agreed to be included in this study.

Data Collection Strategies

The data, relative to the TIDE consultants, is scattered and fragmented throughout the TIDE program files. Data appropriate for this investigation was located, de-identified, and moved from the TIDE program servers by this researcher as the principal investigator. A qualitative analysis tool was acquired and used to store, organize, and analyze the TIDE de-identified data. The software tool chosen for this analysis is QSR NVivo 2 from QSR International Pty. Limited, 651 Doncaster Road, Doncaster, Victoria 3108 Australia (www.qsrinternational.com). The QSR tool along with the de-identified data resides on a separate computer located in a separate facility and controlled by this researcher as the principal investigator.

This investigator, along with appropriate TIDE program personnel, reviewed the available data, decided on its suitability, de-identified the data, and moved it from the TIDE program

computer system to the separate QSR machine. Information that could be used to identify individuals directly or indirectly was not selected and moved to the QSR machine.

This researcher, as the principal investigator, conducted in-depth semi-structured interviews which were audio taped and transcribed. This transcribed text was entered into NVivo and analyzed. The semi-structured script was developed with the advisory group and piloted using members of the group acting as interviewees. As a result of this piloting, it was decided to support two interviews for each selected organization. One interview would be conducted with an individual familiar with the technical issues, and a separate interview was conducted with an executive familiar with the business and organizational issues. It was determined that with these two perspectives from each organization, the research questions would be more fully addressed.

Analysis

The data for this study was de-identified textual data from the TIDE project. This data consisted of artifacts developed as a result of SEI working closely with the TIDE SMEs. Some example sources of data came from lessons learned sessions, workshop results, and the results of the SME interview sessions.

The de-identified data on the separate computer system was entered into NVivo, the QSR qualitative analysis tool. The analysis follows the steps for qualitative analysis as described by Morse and Richards [Morse 02] and supported by the NVivo tool. While the methodology chosen, Rapid Applied Ethnography (RAE), will influence how these steps are executed; the steps are consistent for qualitative analysis [Morse 02]. These steps are

1. Descriptive coding: This step records information about the data. This information consisted of attributes about the data that provide context for the study. In this case, it includes general demographics about the SME organization, along with other information useful for providing context to the analysis.
2. Topic coding: This form of coding involves categorization and tagging of data. The researchers develop or find in the data subjects of interest. The subjects of interest are labeled (categorized), and the source of the data is linked or tagged to the label/category. This allows for consistency of notation and retrievability of these subjects of interest or categories.
3. Analytic coding: According to Morse and Richards, as the researchers continue topic coding the process becomes more “analytic.” It is labeled *analytic* because in creating categories you go *on*, not just linking them to data but also questioning the data about new ideas developing new codes. The purposes of analytic coding include the following:
 - to alert you to new messages or themes
 - to allow you to explore and develop new categories or concepts
 - to allow you to pursue comparisons [Morse 02, p. 119]

4. Theme-ing: Theme-ing involves abstracting up from the data. The researcher immersed in the data and coding begins to see patterns and relationships that lead to some form of conceptualization. The nature of the conceptualization can be simple or complex and is determined by the nature of the data and context. In some cases this abstraction and conceptualization leads to theory.

Conclusion and Recommendations for CMMI-SME

This study supports two fundamental conclusions: 1) Based on the data collected in this study, there is strong evidence supporting the need for disciplined systems development methods within the SME community; 2) The CMMI in its current format and packaging is not feasible for SMEs to adopt and implement.

The following recommendations for CMMI-SME are based on the data and insights gained from this investigation. Extrapolating the needs and expectations of the typical SME from the available data leads to the following requirements statement for CMMI-SME: “The typical SME is looking for short-term, point solutions to known problems with minimum investment, minimal disruption, and quick demonstrable results.”

The requirements statement for the typical SME is particularly important in helping to envision and shape a successful CMMI-SME. This is the initial set of SME expectations. Giving customers what they want is an important marketing principle and should be tempered with the reality that what is provided to them should also meet their needs. In this case the CMMI-SME must provide solutions and results as well as being quick and efficient. The following recommendations hold promise that many of the benefits CMMI provides can be addressed by CMMI-SME quickly and efficiently. If and when these concepts are piloted in the SME environment, the reality may be that given current technology, some problems may not have quick or inexpensive SME solutions.

The recommendations are provided in three areas: packaging, supporting infrastructure, and community transition.

Packaging Recommendations for CMMI-SME

The data collected in this study indicates that the SMEs view the CMMI to be a large bureaucratic process model that is appropriate for large organizations like the Department of Defense (DoD) but inappropriate for smaller organizations. SMEs look at the current packaging of the CMMI and see a model that is too large and complicated to be practical for a small organization to implement. The problem seems to be that SMEs are immediately put off by the size of the model and the fact that they cannot easily relate their business problems to the model contents. This situation is compounded by the lack of case studies of SMEs successfully using the CMMI.

The CMMI is a well-ordered, comprehensive collection of best system development practices. The CMMI-SME packaging recommendations presented here build on the

continuous representation of the CMMI model. The continuous representations allows the CMMI to be viewed as a collection of independent solutions from which an SME can chose to implement certain pieces based on their needs. The ideal packaging for CMMI-SME would present a clear mapping of SME problems to the relevant CMMI-SME model components and from there, to easily implemented solutions, which would provide complete traceability from need to solution. This strategy implies that the CMMI needs to be augmented with a front-end component that links the model to the SME's problem areas and a back-end that provides effective and efficient solutions.

The CMMI is a process framework; it alone does not provide the required methods, tools, or executable processes necessary for successful implementation by the SME. In order to provide the SME with a solution, the CMMI needs to be augmented with appropriate additional components, such as methods, tools, and processes (back-end). SMEs do not have the time, resources, or expertise to convert the CMMI process framework into their own SME-specific solutions. It is very clear that for the CMMI-SME to be successful in the SME community, it will have to be packaged as an organized collection of easily implemented "whole product" solutions. Whole product is defined by Geoffrey Moore as "the minimum set of product and services necessary to ensure that the target customer will achieve his or her compelling reason to buy" [Moore 95, p. 21]. The optimum packaging for the CMMI-SME would provide line of sight connectivity from SME problem, to model components, and to implementation solutions.

Supporting Infrastructure

The recommendations for the CMMI-SME supporting infrastructure are based on the SME data collected, as well as the supply chain development group's extensive experience working with SMEs to improve their performance.

The first infrastructure recommendation is to develop a set of system development performance measures for SMEs and a database of actual performance measures. Large organizations have the resources to benchmark their system development performance in the marketplace to help identify problem areas and to identify best practices for adoption. SMEs normally will not have this capability. Understanding an organization's relative performance and comparing itself to best practices in the industry are very powerful motivators.

The CMMI performance measures should be designed to provide comprehensive and reliable benchmark data on the efficacy of the organization's systems development capability. Once developed, the recommendation is to collect these measures across the SME community and make them available for analysis and comparison. This would allow individual SMEs to compare their system development performance to others in the community thereby providing a reliable benchmark.

The second CMMI infrastructure recommendation is to develop a reliable diagnostic tool to analyze an SME's system development operations. The primary output of this diagnostic would be a determination of the key problem areas the SME will need to address based on its

performance and comparison to best practices. This will include a relative performance rating of the SME to peers in the community.

The third infrastructure recommendation is to develop and provide a collection of whole product solutions for as many of the likely problem areas as possible. These solutions should link to the practices in the CMMI and include as much of the total solution as possible, including training, tools, methods, processes, etc. A very important recommendation, as part of providing these whole product solutions, is to exploit the synergies of existing, successful approaches such as Agile and Lean. Where possible, the infrastructure and artifacts that help make these approaches successful as individual technologies should be incorporated into the CMMI-SME solutions space.

The effect of this would be to use the CMMI to provide a comprehensive systems development architecture from which the SME could chose to plug in their own appropriately packaged solutions to address particular needs and situations. This approach retains the thoroughness and rigor of the CMMI while exploiting the efficacy and cost effectiveness of existing solutions.

SME Community Transition

This exploratory study provides data supporting the need for CMMI in the SME community. The study also indicates that the CMMI in its current form would most likely be too difficult and expensive for SMEs to adopt without significant investment and support. This gives rise to an interesting question: Could the packaging and the infrastructure recommendations be developed once by a commercial or government entity and reused across the SME population? The TIDE consultants believe this is possible. If so, a relatively small investment could make an enormous positive impact to this important economic sector.

Everett Rogers advocates the use of “change agency” in supporting the diffusion and adoption of an innovation across a target community [Rogers 95]. This concept could be applied to CMMI-SME. By introducing a common structure for supporting adoption of this technology by the SME community, investment costs for packaging and infrastructure could be shared and thus greatly reduced for any one SME. In addition, specialized CMMI-SME expertise, such as the change agents advocated by Everett, could be developed and shared across the community. The focal point for initiating this transition support could come from a number of sources: 1) a professional society or association; 2) a commercial venture; 3) government entity interested in economic or industrial development.

Future Research Recommendations

This research is an exploratory study. Its purpose was to investigate feasibility and to influence the direction of future research. One limitation of this study is the generalizability of the study’s findings based on the limited amount of data collected and analyzed relative to the size of the SME population. This suggests that this qualitative study be augmented by

quantitative studies to strengthen the data supporting the need and applicability of CMMI-SME to the SME community.

An intriguing part of this research has been the potential benefit of repackaging the CMMI into value added whole product solutions easily identified and implemented by the SMEs. This approach holds promise and should be piloted with applicable SMEs to establish viability.

The SME community is vital to the safety and economic prosperity of the nation. Supporting this vital sector as it transitions successfully into the digital age will bring great benefits. Based on this research, the CMMI-SME has the potential to be a key enabling technology for SMEs.

References

- [Beebe 95] Beebe, James. "Basic Concepts and Techniques of Rapid Appraisal." *Human Organizations* 54, 1 (1995).
- [Creswell 98] Creswell, John W. *Qualitative Inquiry and Research Design: Choosing Among Five Traditions*. Thousand Oaks, CA: Sage Publications, 1998.
- [Driscoll 98] Driscoll, David L. *Contamination and Communication*. <http://www.sfaa.net/eap/driscoll/driscolla.html> (1998).
- [Klein 99] Klein, Heinz K. & Michael D. Myers. "A set of Principles for Conducting and Evaluating Interpretive Field Studies in Information Systems." *MIS Quarterly* 23, 1 (1999).
- [Millen 00] Millen, David R. "Rapid Ethnography: Time Deepening Strategies for HCI Field Research." *Proceedings of the conference on Designing interactive systems: processes, practices, methods, and techniques*. New York, NY: ACM Press, 2000.
- [Moore 95] Moore, Geoffrey A. *Inside the Tornado: Marketing Strategies from Silicon Valley's Cutting Edge*. New York, NY: HarperCollins Publishers Inc., 1995.
- [Morse 02] Morse, Janice M. & Lyn Richards. *Readme First for a User's Guide to Qualitative Research*. Thousand Oaks, CA: Sage Publications, 2002.
- [Myers 99] Myers, Michael D. "Investigating Information Systems with Ethnographic Research." *Communications of the Association for Information Systems* 2, 23 (1999).
- [Rogers 95] Rogers, Everett. M. *Diffusion of Innovations*. New York, NY: Free Press, 1995.

- [Rogers 97]** Rogers, Yvonne & Victoria. Bellotti. "Grounding Blue-Sky Research: How can Ethnography Help." *ACM Interactions Magazine* (May-June 1997).
- [Sanders 02]** Sanders, Elizabeth. "Ethnography in NPD Research How 'applied ethnography' can improve your NPD research." *Visions Magazine* (April 2002).
- [Spinuzzi 00]** Spinuzzi, Clay. "Investigating the technology-work relationship: a critical comparison of three qualitative field methods." *Proceedings of 2000 Joint IEEE International and 18th Annual Conference on Computer Documentation (IPCC/SIGDOC 2000)*. Cambridge, MA, 2000.

Biography

Gene Miluk

Software Engineering Institute

gem@sei.cmu.ed

Gene Miluk is currently a senior member of the technical staff at the Software Engineering Institute (SEI). During his 13 years at the SEI, Miluk has been active consulting and supporting process improvement efforts for both government and commercial clients. He was part of the CMMI development team, coordinating and leading the CMMI pilots, and a member of the SCAMPISM method development team. Miluk was project lead for the recently released SCAMPI B and C development effort.

Miluk has a BS in systems engineering from the Polytechnic University of New York and an MS in information systems from the University of Colorado, and is completing his doctorate in organizational change from Pepperdine University.

2.6 Applying Software Engineering Standards in Small Settings: Recent Historical Perspectives and Initial Achievements

Authors

Claude Y. Laporte and Alain April

Abstract

The software industry recognizes the value of very small enterprises (VSE) in contributing valuable products and services. ISO International standards were not written for and are hard to apply in small projects, small development organizations, and companies that have between 1 and 25 employees. The current international Life Cycle Standards, ISO/IEC 12207 and ISO/IEC 15288 and their associated guides, do not explicitly address the needs of VSEs. This new international standardization project proposal will attempt to address some of those difficulties by developing profiles and guidance to comply with ISO software engineering standards such as ISO/IEC 12207 and ISO 9001.

Introduction

This paper presents a new project which proposes to facilitate access to, and utilization of, the International Organization for Standardization (ISO) and software engineering standards in very small enterprises (VSEs). The term VSE includes small software development departments and small projects within larger organizations. VSEs are typically organizations and projects with between 1 and 25 employees. In Europe, for instance, 85% of the Information Technology (IT) sector's companies have between 1 and 10 employees [ESI 05]. In the Montreal area, as illustrated in Table 2, close to 80% of IT companies have between 1 and 25 employees [Laporte 05]. Another study conducted by the Technology Assessment Group (CITA) of Wallonia [CITA 97] has published similar data: about 60% of IT companies there have fewer than 5 employees. In Brazil, small IT companies⁴ represent about 70% of the total number of companies [Anacleto 04]. Finally, in Northern Ireland, a survey reports that 66% of IT organizations within companies employ fewer than 20 employees [McFall 03]. There is a need to help these organizations understand and use the ISO software engineering international standards.

⁴ Brazil classifies companies with fewer than 10 employees as micro companies, and those with 10-49 employees as small companies.

Table 2: Size of IT Companies in the Montreal Area [Laporte 05]

Size (employees)	IT Companies		Jobs	
	Number	%	Number	%
1 to 25	540	78%	5,105	29%
26 to 100	127	18%	6,221	36%
over 100	26	4%	6,056	35%
TOTAL	693	100%	17,382	100%

The current ISO software engineering life cycle standards, ISO/IEC 12207 and ISO/IEC 15288, and their associated guides, are not easily applied in small settings. Compliance with those standards is difficult, if not impossible, for small settings to achieve.

This paper is divided into four sections. In the first section, we briefly describe the ISO/IEC SC7 organization mandate in software engineering standards development. In the second section, we relate the recent history that led to an ISO/IEC SC7 Project Proposal for small settings. In the third section, we present the text of the final proposal tabled at the last ISO/IEC SC7 meeting. Finally, in the last section, we report on what was accomplished at a recent meeting in Thailand.

Overview of the ISO/IEC SC7 Mandate

During 1987, the ISO and the IEC (the International Electrotechnical Commission) joined forces and put in place a joint technical committee, named Joint Technical Committee 1 (ISO/IEC JTC1) with the following mandate: “Standardization in the Field of Information Technology: Information technology includes the specification, design, and development of systems and tools dealing with the capture, representation, processing, security, transfer, interchange, presentation, management, organization, storage, and retrieval of information” [ISO/IEC 98]. The mandate of sub-committee SC7, within JTC1, is to standardize processes, supporting tools, and supporting technologies for the engineering of software products and systems.

Figure 2 illustrates the evolution of the over 90 published ISO/IEC standards that are the responsibility of SC7.

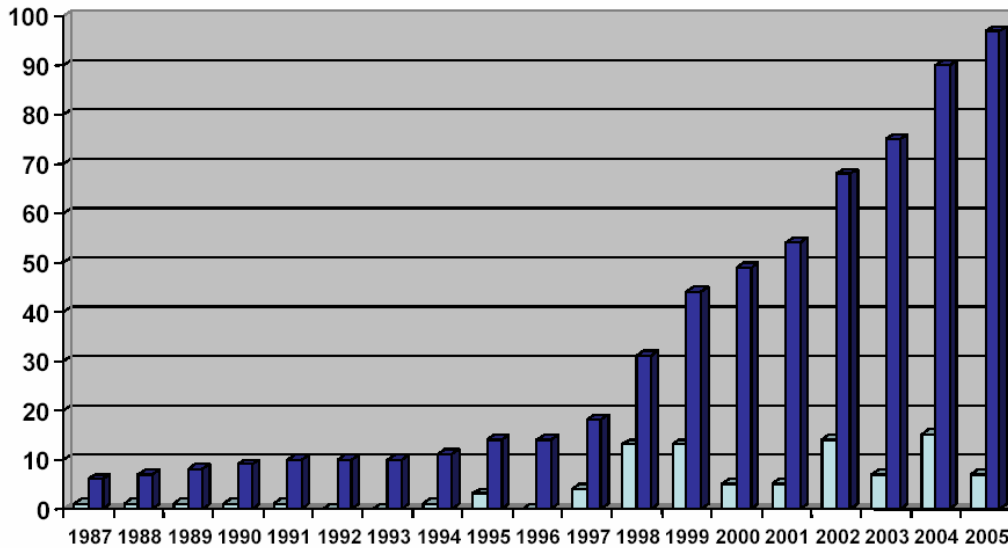


Figure 2: Evolution of Published ISO/IEC Software and Systems Engineering Standards (SC7 2005)

Within the portfolio of SC7 standards, a number are grouped together in a category called “Software and Systems Engineering Processes.” These are standards describing good software and systems engineering practices, as well as standards assessing software and systems engineering practices. In this group, there are four key ISO/IEC standards:

- ISO/IEC 12207 Software Life Cycle Processes
- ISO/IEC 15288 Systems Life Cycle Processes
 - developed with the strong participation of the International Council on Systems Engineering (INCOSE)
- ISO/IEC 15504 Software Process Assessment series
 - CMMI is compatible with ISO/IEC 15504.
- ISO/IEC 90003 Guidelines for the Application of ISO 9001 to computer software

The relationship between these standards is illustrated in Figure 3.

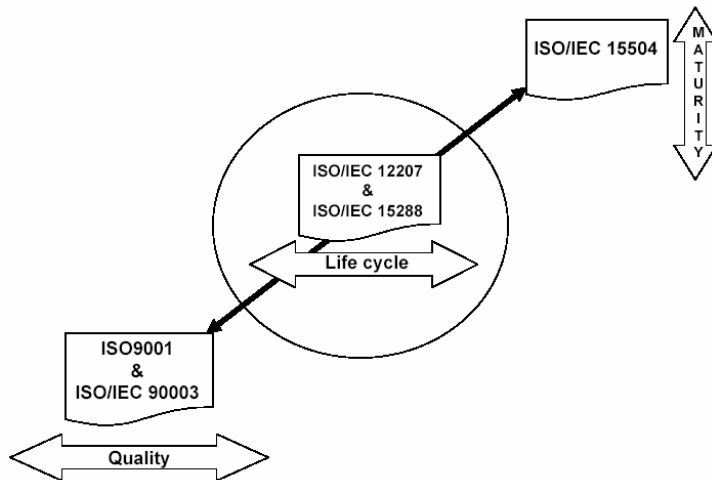


Figure 3: Relationship Between Key SC7 Standards [Coallier 03]

These key standards are well known in the software and systems engineering community. Harmonization of those standards is always a topic of discussion and is included among the newest ISO/IEC SC7 work items. Although ISO 9001 and maturity model usage in small settings is the subject of a debate which has already been initiated, life cycles need to be addressed.

Recent History Leading to an ISO/IEC SC7 Project Proposal for Small Settings

In this section, we describe the history behind the creation of a new ISO/IEC SC7 Working Group (WG).

First Meeting of ISO/IEC SC7 in Australia

At the Brisbane meeting of the SC7 (SC7-04) in 2004, Canada raised the issue of small enterprises requiring standards adapted to their size and maturity level. The current software engineering standards target (or are perceived as targeting) large organizations. Australia supported Canada's position in this regard, and the two national bodies took action to investigate possible ways to forward this issue. A meeting of interested parties was held with delegates from five national bodies (Australia, Canada, the Czech Republic, South Africa and Thailand) where a consensus was reached on the general objectives:

- to make the current software engineering standards more accessible to VSEs
- to provide documentation requiring minimal tailoring and adaptation effort
- to provide harmonized documentation integrating available standards:
 - process standards
 - work product and deliverables
 - assessment and quality
 - modeling and tools

- to generate multiple profiles from the standards mentioned above
 - a profile is a set of one or more base standards or ISPs, or both, and, where applicable, the identification of chosen classes, conforming subsets, options and parameters of those base standards, or the ISPs necessary to accomplish a particular function [ISO/IEC 98]
- to align, if desirable, profiles with the notions of maturity levels presented in ISO/IEC 15504

It was also decided that a special interest group (SIG) be created to explore these objectives to better articulate the priorities and the project plan. The participants felt that it would be possible, during 2004, to achieve the following:

- a set of requirements
- an outline of key deliverables and the associated processes to create them (e.g., how to create profiles)
- a Terms of Reference document for the working group
- an example of a simple profile

First Special Working Group Meeting in Thailand

In March 2005, the Thailand Industrial Standards Institute (TASI) invited a Special Working Group (SWG) to advance the work items defined at the Brisbane meeting. The meeting was attended by delegates from the following countries: Australia, Belgium, Canada, Czechoslovakia, Finland, South Africa, South Korea, USA, and Thailand.

A key topic of discussion was to clearly define the size of VSE that would be targeted by the working group. The working group used a paper published by the Centre for Software Process Technologies to help define the size of small organizations [McFall 03]. McFall presents the various perceived priorities and concern areas for different organization sizes. As illustrated in Figure 4, the priorities and concerns for organizations with fewer than 20 employees are different from those of larger organizations. The consensus reached was that a VSE for IT services, organizations, and projects would have between 1 and 25 employees.

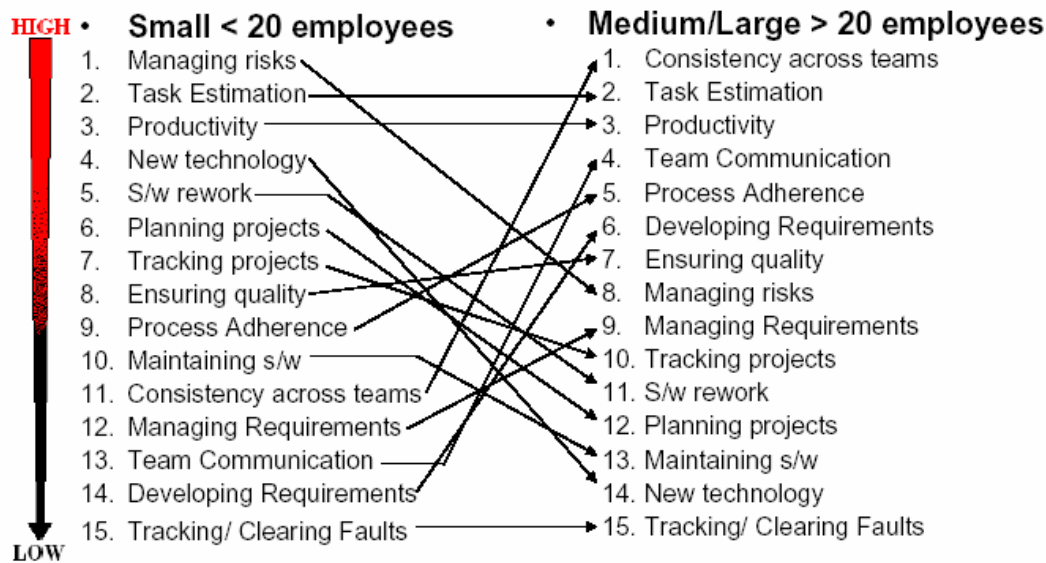


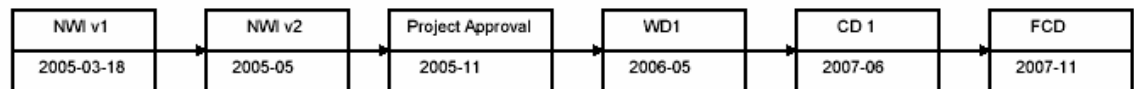
Figure 4: *Priorities and Concern Differences Based on Organization Size [McFall 03]*

A list of actions that could be undertaken by a future ISO/IEC SC7 Working Group was developed at this meeting. The proposed action items are

1. to validate the Work Products produced by the working group
2. to prepare, conduct, analyze, and communicate survey results
3. to search for other centers/organizations focused on SMEs and VSEs
4. to assemble a complete list of characteristics of VSEs and projects
5. to prepare communication material to inform VSEs about the work performed by the WG
6. to develop business cases for the adoption/deployment of work products developed by the WG
7. to develop ISO 12207 Roadmap(s)
8. to pilot Roadmaps

A schedule was also developed for the new working group. As illustrated in Figure 5, the top row shows the standard steps for the development and approval of an ISO standard. There are typically six stages that lead to the publication of a standard [Coallier 03]. After a study period, a new work item (NWI) is developed and sent for balloting. If the ballot is successful, a new project is established with the support of experts from different countries. Then a working draft (WD) is prepared, followed by a committee draft (CD), and lastly a final committee draft (FCD) that is sent for approval.

ISO JTC1 Process



Shadow Process

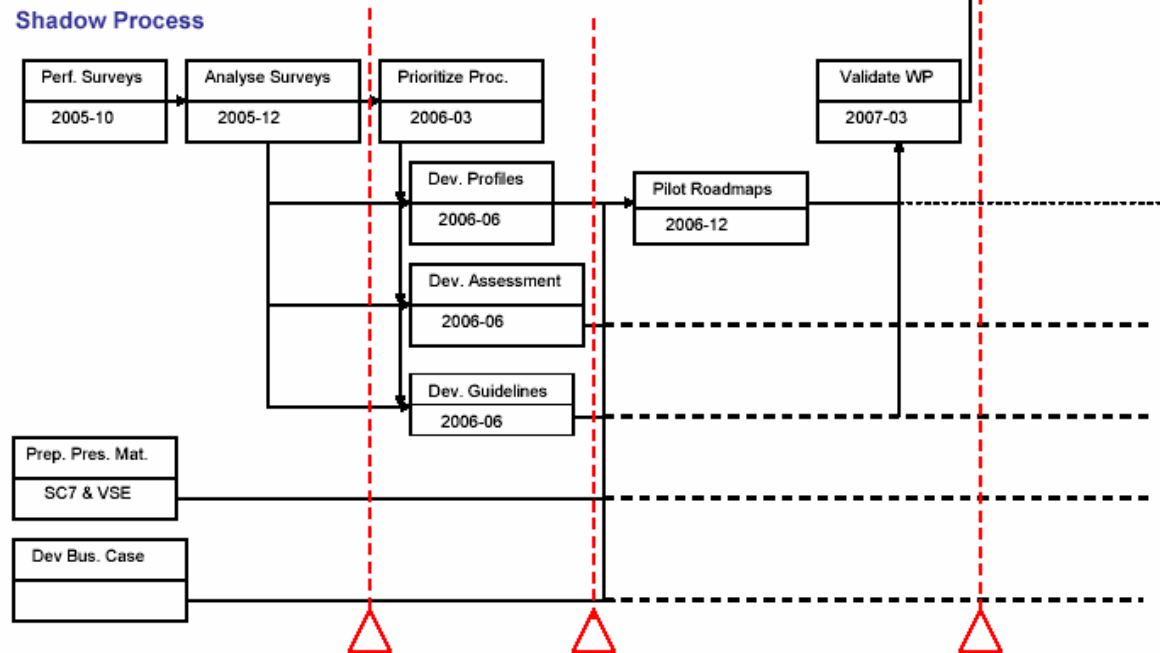


Figure 5: The New Working Group’s Proposed Work Schedule

The major output of this one-week meeting was a draft of the New Work Item described in the “Proposed project tabled at ISO/IEC SC7” section.

SC7 Meeting in Finland

The document developed in Thailand was reviewed during a meeting of one of the WGs at the 2005 SC7 meeting in Helsinki. A resolution was approved as follows: *JTC1/SC7 instructs its Secretariat to distribute for letter ballot an updated version of New Work Item Proposal (ISO-05B) for the development of Software Life Cycle Profiles and Guidelines for use in Very Small Enterprises (VSE) by 20 June 2005.*

- This document was balloted until 21 September 2005. Over twelve countries voted in favor of the NWI Proposal and the following countries indicated a commitment to participate to the new working group: Belgium, Canada, Czech Republic, Ireland, Italy, Japan, Korea, Luxemburg, South Africa, Thailand, UK, and USA.

As a result of this balloting, the Project was approved and the new working group, WG 24, was established as follows:

- Mr. Tanin Uthayanaka (Thailand) was appointed Convener.
- Mr. Jean Bérubé (Canada) was appointed Secretary.

- Mr. Claude Y. Laporte (IEEE CS) was appointed Project Editor.

Proposed Project Tabled at ISO/IEC SC7

The document tabled at the ISO/IEC SC7 Helsinki plenary describes the scope, purpose and justification, and vision statement of a proposed working group. In the following paragraphs, each element of that project is presented. The text below has been extracted from the document balloted by the ISO.⁵

Project Scope

- organizations and projects with fewer than 25 employees
- the current scope of ISO/IEC 12207 and its amendments, the associated guidance document, and other relevant SC7 Standards (e.g., ISO/IEC 15504, ISO/IEC 90003)
- production of technical reports (guides), establishing a common framework for describing assessable life cycle profiles used in VSEs, including small software systems development departments and projects within larger organizations
 - guides to be based on International Standardized Profiles (ISP), identifying which parts of the existing standards are applicable to VSEs, at a specific level and for a specific domain.
 - guides which can be applied throughout the life cycle for managing and performing software development activities; the ultimate goal being to improve the competitiveness and competence of VSEs

Purpose and Justification

The software systems industry as a whole recognizes the value of VSEs in terms of their contribution of valuable products and services. The majority of software organizations fall within the VSE size category. From the various surveys conducted by some of the national bodies that initially contributed to the development of this NWI, it is clear that the current SC7 Life Cycle Standards (ISO/IEC 12207 and ISO/IEC 15288 and their associated guides) are a challenge to use in these organizations because compliance with them is difficult (if not impossible) to achieve. Consequently, VSEs have few, or very limited, ways to be recognized as organizations that produce quality software systems, and, therefore, they do not have access to some markets. Currently, conformity with software engineering standards requires a critical mass in terms of number of employees, cost, and effort, which VSEs cannot provide.

This project will attempt to ease the use of ISO/IEC 12207 processes and ISO9001:2000 and reduce the conformance obligations by providing VSE profiles. The project will develop guidance for each process profile and provide a road map for compliance with ISO/IEC 12207 and ISO 9001:2000.

⁵ The complete text is available free on the SC7 Web site at: <http://www.jtc1-sc7.org/> as item number N3288.

It has been reported that VSEs find it difficult to relate ISO/IEC 12207 to their business needs and to justify the application of the international standards in their operations. Most VSEs cannot afford the resources for, or see a net benefit in, establishing software processes as defined by current standards (e.g., ISO/IEC 12207/15288) and maturity models (e.g. ISO/IEC 15504). The proposed work will liaise with other work in SC7; specifically, it will track the progress of the ISO/IEC 12207 and ISO/IEC 15288 harmonization projects.

Vision Statement

This project will

- provide VSEs with a way to be recognized as producing quality software systems without the initial expense of implementing and maintaining use of an entire suite of systems and software engineering standards or performing comprehensive assessments
- produce guides which are easy to understand, affordable, and usable by VSEs
- produce a set of profiles which builds on or improves a VSE's existing processes or provides a guide to establishing those processes.
- address the market needs of VSEs by allowing domain-specific profiles and levels
- provide examples in order to encourage VSEs to adopt and follow processes that lead to quality software, matching the needs, issues and risks of their domain
- provide a baseline for how multiple VSEs can work together or be assessed as a project team on projects that may be more complex than can be performed by any one VSE
- develop scalable Profiles and Guides so that compliance with ISO/IEC 12207 and/or ISO 9001:2000 and assessment becomes possible with a minimum of redesign of the VSE's processes

Referenced Documents

As illustrated in Figure 6, the following documents have been identified as pertinent to this project: ISO 90003, ISO/IEC 12207, ISO/IEC 15288, ISO/IEC 15504, CMMI and SW-CMM.

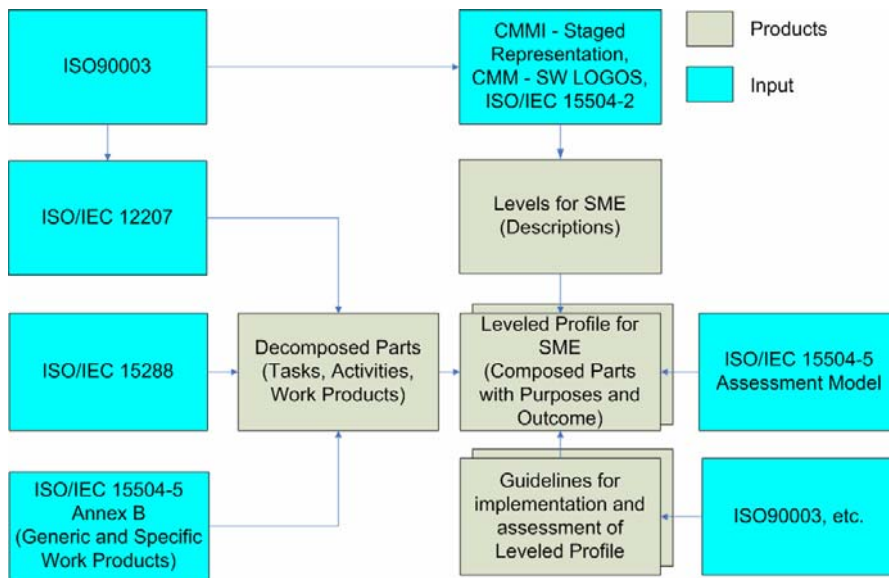


Figure 6: Referenced Documents

Second Special Working Group Meeting in Thailand

In July 2005, the Thailand Industrial Standards Institute (TASI) sent out a second invitation to participate in the Special Working Group (SWG) to be held in September 2005 in Bangkok. The main objective of the meeting was to prepare material that would be presented to WG 24 in order to facilitate the start-up of the working group. The main outputs of the meeting were

- proposed requirements for International Standard Profiles (ISPs) based on Technical Report ISO/IEC TR10000-1
- a proposed survey on VSEs' exposure and needs for software development life cycles
- proposed approaches to profile development and profile architecture
 - proposed business models
- proposed agenda for the first WG 24 meeting
- proposed draft strategic plan for WG 24
- proposed goals of the standard

Acknowledgments

The authors are grateful to the following people who participated in the First Special Working Group meeting in Bangkok and drafted the document that was presented to the SC7 Plenary in May 2005: Jean Bérubé (Canada), Brent Cahill (Australia), Sunthree Charoenngam (Thailand), Melanie Cheong (South Africa), Robert Jonhson (USA), Anatol Kark (Canada), Gil Jo Kim (South Korea), Michael Krueger (USA), Ota Novotny (Czechoslovakia), Alain Renault (Belgium), Joan Stredler (USA), Anukul Tamprasit (Thailand), Varkoi Timo Kalervo (Finland), Tanin Uthayanaka (Thailand), Pipat Yodprudtikan (Thailand), and Patcharada Youyoun (Thailand).

Additional Information

The SC7 Web site www.jtc1-sc7.org provides more information. All JTC 1/SC7 standards can be purchased directly from the ISO (www.iso.ch) or from the national standards bodies.

References/Bibliography

- [Anacleto 04]** Anacleto, Alessandra; von Wangenheim, Christiane Gresse; Salviano, Clenio F.; & Savi, Rafael. "Experiences gained from applying ISO/IEC 15504 to small software companies in Brazil," 33-37. *Proceedings of 4th International SPICE Conference on Process Assessment and Improvement*. Lisbon, Portugal, April 2004.
- [CITA 97]** Cellule Interfacultaire de Technology Assessment (CITA). *Utilisation des Systèmes d'Information Inter-Organisationnels [SIO] par les PME Belges* (SIO Research Report). Cita-Computer Sciences Dept., University of Namur, 1997.
- [Coallier 03]** Coallier, Francois. "International Standardization in Software and Systems Engineering." *CrossTalk* 16, 2 (February 2003): 21.
- [ESI 05]** European Software Institute. *Welcome to ESI*. <http://www.esi.es/> (2005).
- [ISO/IEC 98]** ISO/IEC. *Information technology–Framework and taxonomy of International Standardized Profiles–Part 1: General principles and documentation framework*, 4th ed. (TR 10000-1). 1998.
http://ftp2.ansi.org/download/free_download.asp?document=ISO/IEC%20TR%2010000-1:1998
- [ISO-05A 05]** ISO-05A. *New Work Item Proposal–Software Life Cycles for Very Small Enterprises* (ISO/IEC JTC1/SC7 N3288). May 2005. <http://www.jtc1-sc7.org/>
- [ISO-05B 05]** ISO-05B. *Resolutions–JTC1/SC7 Plenary Meeting* (ISO/IEC JTC1/SC7 N3274). Helsinki, Finland, May 2005.
- [Laporte 05]** Laporte, Claude .Y. et al. "Initiating Software Process Improvement in Small Enterprises: Experiment with Micro-Evaluation Framework, SWDC-REK," 153-163. *Proceedings of International Conference on Software Development*. University of Iceland, Reykjavik, Iceland May 27-June 1, 2005.

- [McFall 03]** McFall, Donald. et al. "Software Processes and Process Improvement in Northern Ireland," 1–10. *Proceedings of 16th International Conference on Software & Systems Engineering and their Applications*. Paris, France, December 2003.
- [SC7 05]** SC7. "Secretariat Presentation." *ISO/IEC Advisory Group Planning Meeting*. Helsinki, Finland, May 23, 2005.
- [SC7-04 04]** SC7-04. *Ad Hoc Report—Software Engineering Standards for Small and Medium Enterprises* (ISO/IEC JTC1/SC7 /N3060). May 2004. <http://www.jtc1-sc7.org/>

Biographies

Claude Y Laporte
École de Technologie Supérieure
Department of Software and IT Engineering
claporte@ele.etsmtl.ca

Claude Y Laporte is a professor of software engineering at the École de Technologie Supérieure. His interests include the development and deployment of software engineering processes, process assessment, software quality, and the management of technological change.

Laporte was a professor at the Canadian Department of Defence when he instigated the development of a Canadian software engineering center. He worked at Oerlikon Contraves, where he coordinated the development and deployment of engineering and management processes. He was the co-project editor of ISO/IEC TR 19759 Software Engineering—Guide to the Software Engineering Body of Knowledge (SWEBOK). He is the editor of a new ISO/IEC working group tasked to develop software life cycle profiles and guidelines for use in very small enterprises.

Laporte co-founded and chaired the Montréal Software Process Improvement Network (SPIN). He is a member of the IEEE Computer Society, the International Council on Systems Engineering, and the Project Management Institute.

Alain April
École de Technologie Supérieure
Department of Software and IT Engineering

Alain April is an associate professor of software engineering at the École de Technologie Supérieure—ÉTS, Université du Québec, Montréal, Canada. He is pursuing a doctorate at the Otto von Guericke University of Magdeburg, Germany. His research interests are software maintenance, software quality, and multimedia database management systems. He has worked in the IT field for more than 20 years in the telecommunications industry. April has contributed to the internal measurement of software of ISO9126 (part 3) and is the associate

editor of the SWEBOK (Software Engineering Body of Knowledge) software maintenance and quality chapters that have recently been accepted as ISO/IEC Technical Report #19759.

2.7 Defect Reduction Through Objectivity in Small Settings

Author

Anoop R Madhavan

Abstract

This paper captures the approach of CODE Plus, Inc. (CODEplus) in reducing defect escape rates by realigning organizational product quality assurance assets per the guidelines of the CMMI. Significant issues that affect the defect escape rates of software products such as requirements volatility, objectivity/coverage of quality assurance (QA) activities, etc. are examined. Being a parameter that can be controlled internally in an organization, objectivity of QA has been modeled and studied.

Techniques that can be used to improve objectivity merely by realigning existing organizational assets have been examined. These techniques have been specifically devised for small business environments where the sensitivity to any negative impacts on project budgets is higher than in resource rich environments. The economies of scale resulting from realignment of organizational assets bring in significant improvement to the objectivity and significant reduction to the correlated defect escapes in the product.

Background

This paper captures the approach of CODE Plus, Inc. (CODEplus) in reducing defect escape rates within the constraints of organizational resources simply by realigning organizational product quality assurance assets using the guidelines of Capability Maturity Model Integration (CMMI), SE/SW/IPPD/SS Version 1.1, Continuous Representation.

In November 2004, CODEplus commenced process improvement work to support our strategic vision of pursuing maturity profiles with the CMMI. One of the major challenges identified during the effort was to implement the additional practices of the model to the existing ISO 9001:2000-based quality management system. The company's product monitoring, verification, and validation procedures have carried official ISO 9001 registration since the year 2000. Reengineering these deep-seated procedures required substantial change management acumen on part of our process improvement team.

Following SEI's IDEAL model for adoption of the CMMI, initial data analysis performed under the guidelines of the Causal Analysis and Resolution (CAR) process area of the CMMI, the organization's defect escape rates (DER) were discovered to have significant room for improvement. DER is defined as the percentage of defects in a product that ended up in the hands of the customer compared to the total number of defects in the product. In

short, DER can be thought of as the defects in the product that escaped the scrutiny of CODEplus and were noticed by the customer.

Causal Analysis and Establishment of Objectives

Mathematical defect-cause (DC) models built on multivariate analysis of historic data revealed that the most prominent parameters affecting the DER were objectivity of quality assurance, verification coverage and requirements volatility. Verification coverage is the number of requirements and/or specifications and the nodes of operational decision trees of a product that have been checked through a formal verification activity. This essentially becomes a function of the resources allocated for verification and cannot be increased without increase in resources. Requirements volatility is the average number of times a requirement has changed through external or internal stimulus. This again is a strong function of the nature of the customer, technology, and product being developed, and little improvements can be effected on this parameter from within the organization.

This left CODEplus with only one parameter that can be improved to cause positive effects on the DER. Although objectivity is a requirement in internal audits according to the ISO 9001: 2000 standards, it is not mandatory in the case of verification, validation or measurement, and monitoring of products. Contrastingly, CMMI's Process and Product Quality Assurance (PPQA) process area places objectivity in product quality assurance practices (Specific Practice SP 1.2-1) as a paramount requirement. CODEplus established the following guidelines for measurement of objectivity in a QA evaluation. For 100% objectivity, the creator of the product, the creator of the criteria against which the product is evaluated, and the evaluator must be three distinct personnel.

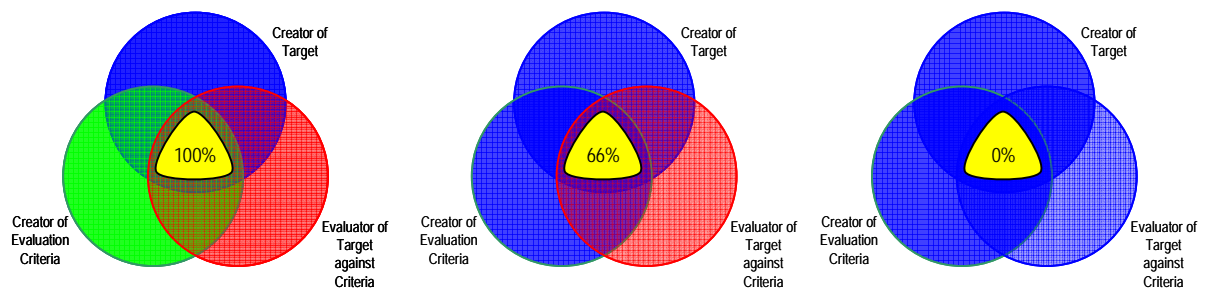


Figure 7: Measurement of Objectivity in a QA Evaluation

Other combinations would yield objectivity percentages less than 100%. If one person creates the product and criteria and evaluates the product against the criteria, the objectivity would be 0%. Data analysis of historic data yielded that a bug was five times more likely to be uncovered during a 100% objective QA evaluation than on a non-objective evaluation. The DC model was also used to establish that achieving CODEplus' desired/target DER would require an average objectivity level greater than 80%.

Challenges

The major challenge of obtaining 80% objectivity in a small business setting is most often the lack of dedicated quality assurance engineers. This essentially means that cross team-based QA evaluations become necessary to achieve the level of objectivity required. The challenge that cross team activities introduces in operations is significant in that schedules of projects become heavily dependent on each other. Naturally, this approach needed to slide over opposition and resistance to change from the operations and project management communities due to its potential to impact schedules across multiple projects.

In addition, another point of contention raised by the project management community and supported by data was that objective quality assurance increased the QA cost ratio (ratio of cost of quality assurance to total cost of product development), and this resulted in more expensive products. Intangibles (such as loss of customer goodwill) apart, this observation was deemed valid through data analysis. Engineers who had no part in creating the product would invariably need to study and familiarize themselves with the product against a learning curve to effectively evaluate the product or develop criteria for evaluation of the product. In such situations, over 70% of the costs of quality assurance were incurred in developing evaluation criteria.

Cost and schedule are the most prominent performance monitoring elements of a project management community. Therefore any changes impacting cost or schedule needed adequate buy in from operations and project managers to be implemented successfully. The approach adopted in addressing this sensitivity and balancing the organizational requirement of achieving the target objectivity level is elaborated in the subsequent sections.

Approach

The Seven Elements of Quality Assurance Evaluations

Software quality assurance (SQA) evaluation is the process of objectively evaluating a software product or an intermediary work product (henceforth referred to as target) against a set of documented criteria in the established environment, identifying and documenting the results of the activity, and disposing the evaluated work product as a pass or a fail depending on how well it meets the criteria. The elements of SQA evaluations include the following:

Evaluator: Evaluator conducts or leads the evaluation procedures. Depending on the involvement of the evaluator in other activities such as the creation of the product or creation of the criteria, the objectivity levels of the evaluation varies from 0 to 100%.

Target: Target is the work product that is being evaluated. It may be a document, a record, or another intangible or tangible work result. Often intermediary work products that are not deliverables to the customer may also undergo quality assurance evaluations since they are the basis for further development work and any unidentified mistakes in them have the potential to impact work products that have to be delivered to the customer. The target should

always be under configuration control, in that the version undergoing quality assurance should clearly be identified and demarcated to ensure that if quality assurance evaluation fails, the team will not inadvertently use that version of the target further.

Evaluation Method: Methods for SQA evaluation can include peer reviews, sign-offs, and testing. Peer review as suggested by the name is a review of a product by the peers of the creator of the product. It can be conducted as walk-throughs, offline assessments, informal reviews, or product inspections. Sign-offs are formal contractual reviews that are conducted on a product in the presence of project management by the customer. Testing involves traversing a work product through as many operational paths as possible to ensure that the work product performs with the required functionality and meets the specifications.

Criteria: Each target will have a distinct set of criteria that are used to objectively verify its quality. Criteria are always measurable and heavily dependant on the characteristics of the target. For example, a good quality class diagram will have classes that have high cohesion and low coupling. So if the target of an SQA evaluation is a class diagram, the criteria will be objective measures of the level of cohesion and the level of coupling of the individual classes in the class diagram.

Environment: Each SQA evaluation is conducted in an environment that has the maximum potential to uncover inherent defects of the target. In functional testing, it is preferable to conduct the testing in the actual operational environment of the product or in one that simulates this environment. In stress testing, the environment will be compromised in a pre-decided manner that can be used to measure the performance of the product during infrastructure degradations.

Observations: Each observation during the SQA evaluation whether positive or negative is noted down for purposes of data collection. Observations may include success, partial success, minor non-conformances, or critical non-conformances in the ability of the target to meet the criteria.

Disposition: Using the observation data collected, a decision regarding the status of the SQA evaluation is made. Decisions are usually pass, conditional pass, or fail.

It is important that all seven of these parameters are well planned and organized to maximize the defect discovery potential of each SQA evaluation.

Quality Assurance Evaluations in the Life-Cycle

In addition to the above seven parameters, timing of SQA evaluations deserves special attention. Conventionally quality assurance activities in the life-cycle are “back-end heavy.” Many software development methodologies in fact have a testing phase at the end of the life-cycle that performs all verification and validation procedures. Through trial and error, CODEplus early on discovered that productivity can be maximized and rework can be

minimized by interspersing quality assurance activities within the various phases of the software development life-cycle (SDLC).

Table 3: Interspersing Quality Gates Within SDLC

PHASE	Gate #	SQA Evaluation
ANALYSIS	1	Requirements Peer Review
		Requirements Sign Off
HIGH LEVEL DESIGN	2	Architecture Review
DETAILED DESIGN	3	Component Design Review
		Design Sign Off
IMPLEMENTATION	4	Code Review
	5	Unit Test
INTEGRATION	6	Integration Test
	7	Regression Test
	8	Installation Test
	9	Functional Test

At CODEplus, each SQA evaluation performed internally during the SDLC is called a *quality gate*. A quality gate essentially has a gate keeper who is the engineer who performs the evaluation. Based on the results, the evaluation can be dispositioned as a pass or a fail. If the disposition is a pass, the gate is considered opened by the gate keeper. In short, for the gate to open the engineer performing quality assurance evaluation must certify that the target of the evaluation meets all the evaluation criteria. By interspersing quality gates throughout the SDLC as shown in Table 3, the quality of intermediary work products such as requirements, architecture, and component designs that are critical to the final product’s quality are also ensured in a structured and formal manner.

Definition of Roles and Policies

Monitoring objectivity in each quality gate requires clear delineation of roles performing the various engineering activities. Based on broad functions performed, the following eight roles were defined and separated into four groups:

Table 4: Roles for Engineering Activities

Group A	Group B	Group C	Group D
Requirements Analyst	Project Manager	Build Engineer	Quality Engineer
System Architect	Technical Manager		
Design Engineer			
Implementation Engineer			

In addition, organizational policies were established restricting personnel in the same project to play roles outside their groups. This would mean that a system architect can play the role of a design engineer but not of a quality engineer.

Allocation of Responsibilities

Each activity in the SDLC was assigned to one of the above eight roles to ensure that the objectivity was 100% and that the QA cost ratio was minimized by assigning creation of criteria to roles that require the minimal learning curve. For example, creation of the criteria for functional testing (functional test cases) was assigned to the requirements analyst since that role is responsible for creating the functional requirements that are used as the basis for functional test case creation.

Results

The resulting objectivity in projects, where each role is performed by discrete personnel, is shown in the following table:

Table 5: Results When Each Role is Performed by Discrete Personnel

QA Activity	Creator of Criteria	Creator of Target	Evaluator	Objectivity
Requirements Review	TM	RA	QE	100%
Architecture Review	RA	SA	QE	100%
Component Design Review	SA	DE	QE	100%
Code Review	TM	IE	QE	100%
Unit Test	DE	IE	QE	100%
Integration Test	SA	BE	QE	100%

Regression Test	RA	BE	QE	100%
Installation Test	RA	BE	QE	100%
Functional Test	RA	BE	QE	100%
Average Objectivity				100%

If one engineer plays multiple roles in the same group, as permissible by the policies, the objectivity levels in the project would be less than 100%, as shown in the following table:

Table 6: Results When Each Role is not Performed by Discrete Personnel

QA Activity	Creator of Criteria	Creator of Target	Evaluator	Objectivity
Requirements Review	TM	RA	QE	100%
Architecture Review	RA	SA	QE	66.66%
Component Design Review	SA	DE	QE	66.66%
Code Review	TM	IE	QE	100%
Unit Test	DE	IE	QE	66.66%
Integration Test	SA	BE	QE	100%
Regression Test	RA	BE	QE	100%
Installation Test	RA	BE	QE	100%
Functional Test	RA	BE	QE	100%
Average Objectivity				88.89%

The objective established was to improve the objectivity levels to greater than 80%, which is achieved in both of the above cases. After implementing this approach, the objectivity levels on CODEplus projects continuously rose and now are stable between 88 and 100%. The DER of products developed in these projects has reduced to one third and is well within CODEplus' desired level.

In summary, this exercise served as a clear demonstration to CODEplus of how CMMI-benchmarked process reengineering results in measurable improvements without the necessity of any additional organizational or project-related material or human resource.

Biography

Anoop R Madhavan

CODE Plus, Inc.

amadhavan@code-plus.com

Anoop Madhavan is a quality manager at CODEplus. He has several years of experience in process management, ISO 9001, and CMMI. He is responsible for all quality, configuration, and release management activities at CODEplus. He has completed training as a Lead Auditor of ISO 9001-based quality management systems and has undergone the introductory training programs to the CMMI.

Prior to CODEplus, Madhavan successfully led the ISO 9001 registration and CMMI Maturity Level 2 accomplishments of Newtek International, where he served as an analyst. Madhavan has an MBA from the University of Alabama and is a member in good standing of the Project Management Institute (PMI), American Society for Quality (ASQ), and the American Mensa, Ltd.

2.8 Team Innovation Management (TIM): Research Into Practice

Author

Don O’Neill

Abstract

Innovation enables an enterprise to elevate its offerings in the software stack. Team Innovation Management (TIM) is organized to encourage innovation within the U.S. software industry and to advance the competitive development of the enterprise by renovating functional tasks and activities and accelerating the innovation management capability and capacity needed to substantially increase innovation in both the production and use of systems and software. It is specifically focused on the systems engineering and software engineering roles and capabilities needed to systematically collaborate in the cross discipline intersection between producer and consumer.

Background

The Center for National Software Studies (CNSS) has recently released the “Software 2015: A National Software Strategy to Ensure Security and Competitiveness,” which is a report from the Second National Software Summit [CNSS 05]. The Software 2015 report identifies four programs and eleven initiatives to carry out the national software strategy. One of these programs focuses on encouraging innovation. As the executive vice president for the CNSS, I have directed the global software competitiveness studies and authored the section of the Software 2015 report on encouraging innovation. Team Innovation Management is organized to transform research into practice and is being conducted to renovate and accelerate innovation management capability and capacity in the production and use of systems and software.

Macroeconomic Positioning

Let’s begin by looking at the view from the top from the perspective of macroeconomic positioning. Michael Porter of Harvard University has identified the macroeconomic stages that drive national competitive development including cheap labor, investment in infrastructure, innovation, and economic advantage [Porter 90]. The U.S. is now transitioning from the investment driven stage where the infrastructure is organized to improve productivity and quality to the knowledge-based, innovation driven stage where software and information technology intersect with application domains in every industry sector to produce novel and useful results that extend the state of the art [Council of Competitiveness 04]. In fact the creative sector of the U.S. economy accounts for 47% of the wealth generated with 30% of the workforce [Florida 05].

Innovation enables an enterprise to elevate its offerings in the software stack. To ignore innovation is to risk falling into commodity status and offshoring. Innovation in the industrial age was achieved through individual genius; in the knowledge age, it is achieved from collaborative activity.

The U.S. has been pushed out of the cheap labor stage and can't seem to complete the infrastructure stage, but necessity has forced upon us the dual challenges of innovation and offshore outsourcing. The outcomes envisioned in dealing with these dual challenges include increasing U.S. innovation in both the production and use of software products and systems, aligning global software participants and functional tasks according to an innovation-driven value hierarchy, and retaining high value, knowledge-based service innovation onshore while pushing the highest skill work to the lowest cost of performance whether onshore or offshore [CNSS 05].

R&D and Innovation

The first step in seeding the future and dealing with the challenge of innovation is to distinguish R&D and Innovation. The government view and the industry view present us with a dual focus.

Government sponsored R&D is aimed at a limited number of large innovations. This innovation in the large results in public goods not appropriable by a single enterprise and open to all both onshore and offshore. The artifacts of Government R&D are roadmaps and agendas that facilitate dissemination and foster collaboration.

The business enterprise focuses on an uncountable large number of small innovations. This innovation in the small yields private goods fully appropriable unless the enterprise chooses to make them open, an emerging business practice. The artifacts of enterprise innovation are the distinguishing products that deliver business success and boost competitiveness and the patents that insure future success. Here inventors are the point of the spear in the struggle for global competitiveness. The Team Innovation Management (TIM) research is seeking a systematic approach to achieving innovation in the small.

Business Context

The difficulty in completing the infrastructure stage in seeking national competitive advantage is illustrated by the plodding struggle to advance the nation's software process maturity. Many organizations breeze past the even numbered process maturity levels (2, 4) with ease only to struggle with the odd levels (3, 5) [Chrissis 03]. This is especially true of level 5 whose very foundation is less well defined and understood. Organizational Innovation and Deployment (OID) is often overlooked, viewed more as an obstacle to a successful assessment and less as a value producing mechanism. The bigger concern here is that the CMMI Organizational Innovation and Deployment (OID) process area is too narrowly and inwardly focused on process innovation and fails to capture product innovation. One reviewer of the article sensibly observed that the purpose of OID states "that measurably improve the

organization's process and technologies," and it would be very easy for the SEI to change this wording in the next release to "that measurably improve the organization's process, products, and technologies."

Like the SEI and its CMMI, the Air Force also overlooked innovation in listing its minimum software focus areas in its memo (04A-003) on "Revitalizing the Software Aspects of System Engineering."

While achieving innovation is usually sporadic, management prefers something more systematic and predictable. TIM bridges the gap between the realities of uncertainty and experimentation associated with creativity and invention and the more focused goals and objectives environment of the enterprise and its managers. IBM's "Global CEO Study 2004" reported that today's CEO's are intent on increasing revenue through new and differentiated products and services and on containing cost through strategic offshore outsource partnering. The choice for business is boiling down to innovate or outsource.

In the Balanced Scorecard, the choices are operational excellence, customer intimacy, and product innovation [Treacy 95]. TIM addresses product innovation.

Becoming Innovative

Let's look at the right mind set and what it takes to become innovative on the factory floor. Essential behaviors found in an innovative work environment include listening to stakeholders, valuing diversity of thought, giving way to superior knowledge not management power, brainstorming ideas where judgment is deferred, encouraging a high volume of ideas, and breaking old habits that limit thinking.

Value Point

In pursuing innovation it pays to focus on value. In software these are value points. "software: the infrastructure within the critical infrastructure" [CNSS 05] disciplines the TIM focus on software usage where the value points of critical industries are identified. A value point is a computer program or software system within an enterprise product line that is strategically essential to the competitiveness of the enterprise. Once identified, value points are tagged as strategic assets subject to the rigors of the enterprise strategic planning process. This ensures that allocated resources are committed to achieve the best industry practice in their project management, product engineering, and process management.

Research Direction

Let's turn now to a concept and plan for the research direction. TIM research is committed to identifying, applying, and verifying the practice, knowledge, skills, and behaviors needed to substantially increase innovation in both the production and use of systems and software. It is specifically focused on the systems engineering and software engineering roles and

capabilities needed to systematically collaborate in the cross discipline intersection between producer and consumer.

The goals of TIM are to

- encourage innovation within the U.S. software industry in accordance with the “Software 2015: A National Software Strategy to Ensure U.S. Security and Competitiveness,” report of the Second National Software Summit [CNSS 05]
- advance the competitive development of the enterprise by renovating functional tasks and activities and accelerating the innovation management capability and capacity needed to substantially increase innovation in both the production and use of systems and software
- provide systems engineers and software engineers with the essential knowledge, skills, behaviors, and motivation needed to substantially improve team innovation management in the enterprise and on the project

The Paradigm Shift

Past is not prologue. Achieving innovation demands an important paradigm shift. While the pursuit of innovation may be systematic, achieving innovation is more chaotic. In some ways innovation management resembles quality process improvement, but the paradigm is essentially different. While the infrastructure-based quality process demands conformance, standards compliance, and risk adversity with the hope for perfection, the innovation management process demands creativity, experimentation, and risk taking with the hope for success but the possibility of failure. Consequently, the enterprise faces a competency destroying change management challenge for both staff and management.

The enterprise beginning the transition from an infrastructure-based quality process to an innovation management operation may tend to depend too much on getting lucky and not enough on being good.

- In getting lucky, success is measured in terms of return on technology where gains too easily labeled as innovative are commoditized at the outset, and directional changes originate from the producer that tend to promote efficiency and better-cheaper-faster, all of which draw upon existing enterprise staff skills and old visions from the infrastructure stage.
- In being good, success is measured in terms of return on innovation where truly innovative gains are more strategic, and intersectional changes originate in the cross discipline collaboration and culture clash between producer and consumer where changes are deep seated and transformational, all of which require the renovation of enterprise staff skills and new visions.

The Intersection

The intersection is an energizing model and the place where innovation occurs. Innovative ideas exist in the minds of practitioners; they simply must be harvested. Innovation lies at the

intersection of invention and insight dependent on ideas, collaboration, and expertise [Johansson 04]. At the intersection there are many ideas and many combinations, and there are many forces including culture, science, and high performance computing.

The intersection is modeled in Figure 8. Software engineers and information technology specialists enter the intersection from the top. Systems engineers and industry specialists enter the intersection from the left. Once a systems engineer and a software engineer pair step into the intersection there is a clash of disciplines and culture. The initial reaction of participants may be to recoil and repel the onslaught of new ideas and concepts and retreat into the comfort zone of ones profession with its protective myths, biases, and old habits. Simply stated by a reviewer of the article, “Software and system engineers when confronted with new ideas may hide behind their disciplines traditional concepts and opinions.” Where the outcome of these encounters is one sided, where either the systems engineer or the software engineering perspective dominates, the changes are considered directional. Where there is intense interaction and give and take, where neither systems engineer nor software engineer dominates, the changes are considered intersectional. Intersectional changes are usually the most valuable.

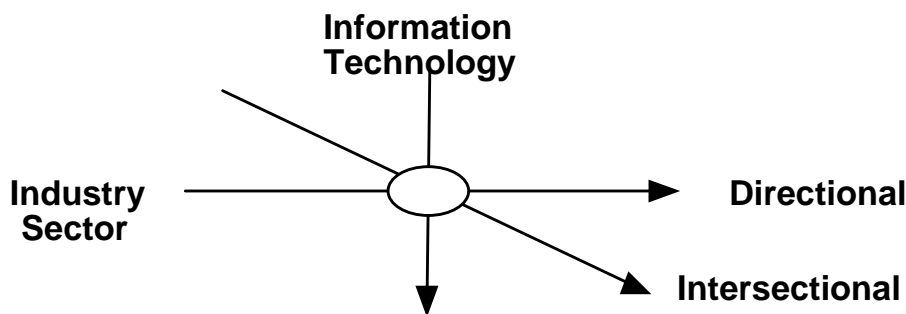


Figure 8: The Intersection

Once at the intersection, it is the role and responsibility of the systems engineer and the software engineer to generate as many ideas as possible.

- Some ideas will be simply directional, rules-based work force reducing efficiencies [Levy 04]. Directional ideas spawn new features and capabilities, are often customer driven, and can be implemented by planned and predictable steps. These new features may extend the dwell time of the product line within the niche.
- Others will be intersectional, process pattern transformations [Levy 04] that involve radically new directions driven by the cross discipline clash with new directions not based on detailed knowledge and ideas originated from people least expected. These new directions may open new niches.

When applying one of the essential behaviors, there is no substitute for superior knowledge. The systems engineer brings an understanding of the state of technology and ongoing

collaborative research as well as application domain patterns, system requirements, customer needs, and sustaining operations. The software engineer brings an understanding of available COTS solutions and open source resources as well as the software engineering practices that foster the structure and modularity needed for trustworthiness and usability and the computer science foundations that enable numerical analysis needed for performance and reliability.

The systems engineer and software engineer attempt to discover the deep needs of the project and to draw out the customer desires and operational considerations that have not received sufficient attention. These engineers look for process patterns within the application domain that can be exploited. Process patterns are ways of organizing inputs and outputs, performing transformations, and managing the information and control flow of the application. In designing transformations on data, they identify the essential algorithms within the application and seek to engineer them in the best possible way. Throughout this process, these engineers work to discover nonobvious ideas that are novel and useful. The following Intersection Script can be used to stimulate dialogue between systems engineers and software engineers:

1. What are the deep needs of the project? Are they being met? If not, what should be done?
2. Are there some customer desires that have not received attention? What are they?
3. Have operational usability considerations been overlooked? What are they?
4. Have any stakeholders been overlooked? If so, who are they?
5. What are the process patterns of the application domain? Are they being applied? If not, what should be done?
6. What are the essential algorithms of the solution? Are they being implemented in the best possible way? If not, what should be done?
7. Are there novel ideas that should be considered? If so, what should be done?
8. Are there other useful ideas that should be considered? What are they?
9. Are there some nonobvious ideas that should be considered? What are they?

The Lab

The centerpiece of the TIM is the TIM Lab, which is the operational model for the intersection structured to accept teams of five systems engineers and five software engineers. Here systems engineers and software engineers are paired-up for their appearance in the intersection of innovation. This is when the application domain and information technology clash and when each pair generates as many good ideas as possible and then presents the results to the group. The group then ranks and orders the most promising ideas. Participants engage each other in seeking out the deep needs in the application domain, identifying process transforming innovations, and pinpointing rules-based innovations. Participant pairs with the most promising ideas are invited to record their innovations in the form of Innovation Value Statements and to present these to an Enterprise Innovation Committee.

TIM Labs are held periodically. During the specification and design activities of the software life cycle, TIM Labs may be conducted monthly. During the maintenance and sustaining engineering activities, scheduling TIM Labs on a quarterly or semiannually basis may be best. Each systems engineer enters the intersection with each software engineer yielding a total of 25 intersection appearances. With five intersection appearances occurring concurrently and each appearance scheduled for 30 minutes, a TIM Lab for five systems engineers and five software engineers can be kicked off, conducted, and wrapped up in a half day.

The Lab Results

As systems engineers and software engineers enter the intersection in pairs to discuss needs and capabilities, each pair generates as many innovative ideas as possible. If each appearance generated just one innovative idea, the TIM Lab session would harvest 25 innovative ideas. If four ideas were generated per appearance, then 100 innovative ideas is the result. These ideas are explicitly recorded and categorized according to directional or intersectional, rules-based or process pattern, deep-seated need or nice-to-have capability, and producer or consumer innovation. Ideas are categorized in this way to provide metrics for the operational analysis and improvement of innovation management activities.

The TIM Lab results are recorded on the Innovation Recording Form shown in table 1. It may be useful to discuss the recording of one entry in detail. On Innovative Idea #1, the computation of true interest cost is quite important in the conduct of ecommerce online auctions of fixed income instruments. Sellers want to show the highest possible true interest cost to attract buyers. Controlling the finite word effects of underflow and the loss of low order bits achieves this. Therefore, this idea addresses a deep need. It is directional because the idea is simply the application of good computer science practice. The implementation is rules based in that it involves the method and rules of calculation and not the process. The idea originates with the producer. In the field of business and finance the idea is nonobvious and novel.

Table 7: Innovation Recording Form

Innovative Idea	Deep Need (Y/N)	Directional or Intersectional (D/I)	Rules-based or Process Pattern (R/P)	Novel (Y/N)	Useful (Y/N)	Nonobvious (Y/N)	Producer or Consumer (P/C)	Description of the Innovative Idea
1	Y	D	R	Y	Y	Y	P	True interest cost computed with controlled finite word effects for increased accuracy
2	Y	D	R	Y	Y	Y	P	Account for gravitational affect of Andes Mountains on GPS satellite rotation time
3	Y	I	P	N	Y	N	P	Adopt Ultra Thin Client to improve security by reducing dependence on Microsoft products

Innovation Value Statement

The Innovation Value Statement contains the following topics:

1. Background
2. Need for Change
3. Description
4. Benefit
5. Impact
6. Value to Customer
7. Value to Project
8. Value to Organization

9. Actions to Implement

Smart Pipe of Innovation Management Tiers

It is not enough to simply generate ideas. Ideas must systematically feed business and economic development. This involves harvesting ideas as intellectual property from knowledge workers on projects, selecting the most promising ideas and refining them into well framed statements of value, and impacting the business strategy and development with the best ideas. The smart pipe process flow is shown as a state machine in Figure 9.

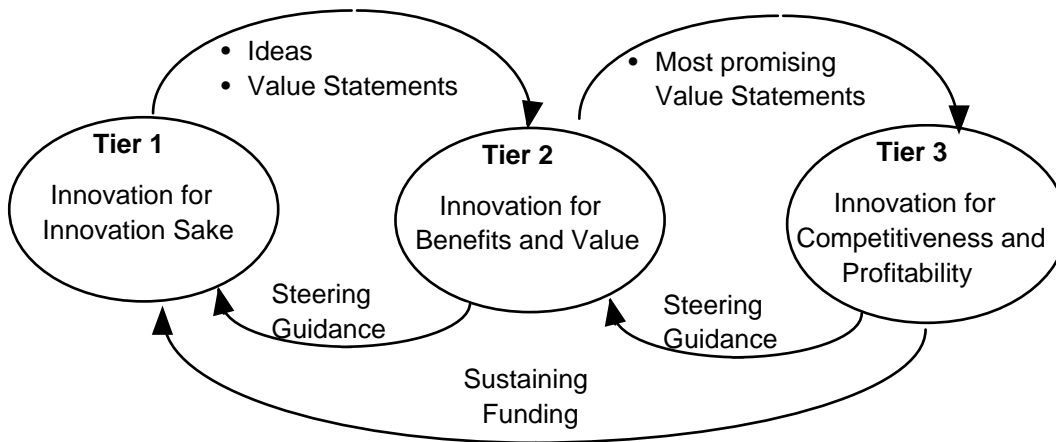


Figure 9: Smart Pipe Process Flow

The smart pipe, tiered process will synthesize the interconnected layers or contexts formed by innovators who generate ideas, brokers who manage ideas and idea development, and buyers who use ideas for the benefit of the enterprise. The three tiers operate to identify innovative ideas and specify their value in multiple dimensions using an Innovation Value Statement, to judge and select the most promising value statements and to identify refinements intended to increase their benefit, and incorporate the best ideas in the business strategy and ongoing development in order to improve the competitiveness and profitability of the business and the value of the enterprise. The three tiers are synchronized through steering guidance from buyers to brokers and from brokers to innovators. Figure 10 illustrates the smart pipe process shown as an ETVX diagram.

The three processes span onshore and offshore operations. The innovators in tier 1 include both global enterprise personnel and outsourced personnel who might be all onshore, all offshore, or some onshore and some onshore. The committee of brokers in tier 2 includes global enterprise personnel who are expected to be onshore. The buyers in the marketplace in tier 3 include global enterprise personnel who are expected to be onshore.

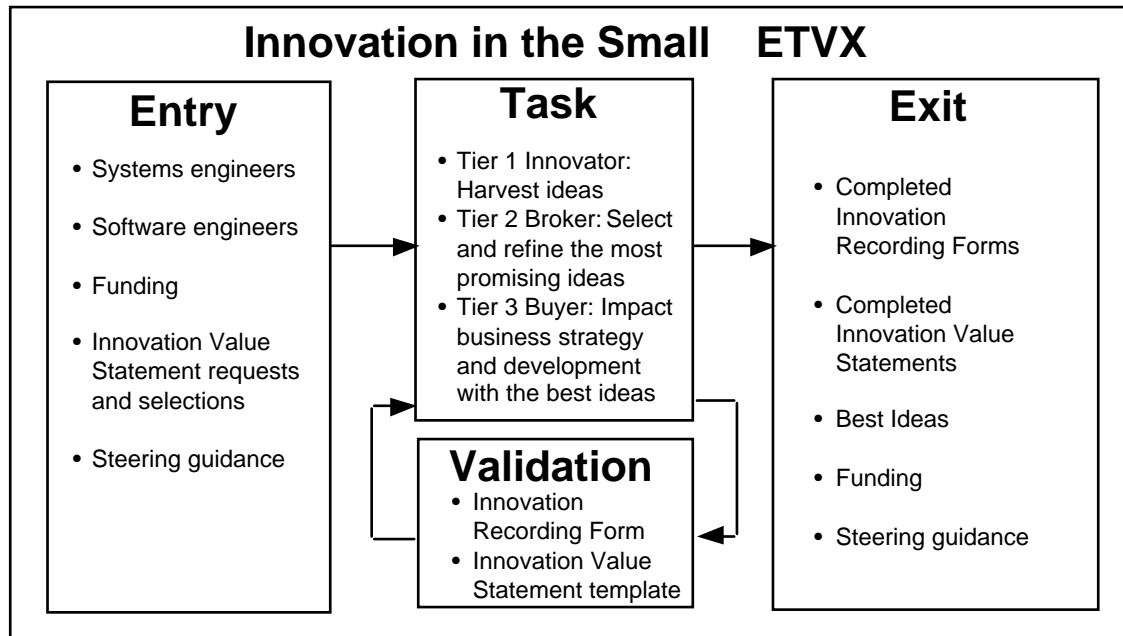


Figure 10: Smart Pipe ETVX Diagram

New Insights and New Benefits

Looking forward, there will be new insights and new benefits. The persistent application of the TIM Lab will reveal valuable insights. These insights are enabled by TIM, which is characterized by a short, intense, and repetitious exercise whose results are persistently retained. It features the TIM Lab process, which is a mode of team collaboration among systems engineers and software engineers. TIM also presents the model of the intersection with its distinctions for directional and intersectional innovations that are rules based or process patterned and producer or consumer sourced and may reveal valuable insights with respect to achieving highly-valued novel, useful, and nonobvious innovations.

The benefits of TIM include the following:

- improve systems and software engineering team capability to systematically collaborate in the cross discipline intersection between producers and consumers
- improve enterprise compliance with the Organizational Innovation and Deployment process areas of the CMMI aimed at selecting and deploying improvements and institutionalizing the defined process associated with innovation and its management
- improve the capability to guide producers and consumers towards intersectional, process transforming innovations that address deep needs
- improve the capability to guide producers and consumers towards directional, rules-based innovations that improve efficiency and productivity
- improve organizational readiness to deploy strategic offshore outsourcing

References

- [Chrissis 03]** Chrissis, Mary Beth; Konrad, Mike; & Shrum, Sandy. *CMMI: Guidelines for Process Integration and Product Improvement*. Boston, MA: Addison-Wesley, 2003.
<http://www.sei.cmu.edu/cmml/publications/cmml-book.html>
- [CNSS 05]** Center for National Software Studies. "Software 2015: A National Software Strategy to Ensure U.S. Security and Competitiveness," *Report of the Second National Software Summit* (Apr. 25, 2005).
<http://www.cnsoftware.org/nss2report/NSS2FinalReport04-29-05PDF.pdf>
- [Council of Competitiveness 04]** Council on Competitiveness. *Innovate America: National Innovation Initiative Report*. Washington, DC: Council on Competitiveness, 2004.
- [Florida 05]** Florida, Richard L. *The Flight of the Creative Class: The New Global Competition for Talent*. New York, NY: Harper-Collins, 2005.
- [Johansson 04]** Johansson, Frans, *The Medici Effect: Breakthrough Insights at the Intersection of Ideas, Concepts, and Cultures*. Cambridge, MA: Harvard Business School Press, 2004.
- [Levy 04]** Levy, Frank & Murnane, Richard J. *The New Division of Labor: How Computers are Creating the Next Job Market*. Princeton, NJ: Princeton University Press, 2004.
- [Porter 90]** Porter, Michael E. *The Competitive Advantage of Nations*. New York, NY: The Free Press, 1990.
- [Treacy 95]** Treacy, Michael & Wiersema, Fred. *The Discipline of Market Leaders: Choose Your Customers, Narrow Your Focus, Dominate Your Market*. Reading, MA: Addison-Wesley, 1995.

Biography

Don O'Neill

Center for National Software Studies

ONeillDon@aol.com

Don O'Neill is a seasoned software engineering manager and technologist currently serving as an independent consultant. Following his twenty-seven year career with IBM's Federal Systems Division, O'Neill completed a three-year residency at Carnegie Mellon University's Software Engineering Institute (SEI) under IBM's Technical Academic Career Program. There he developed a blueprint for charting software engineering evolution in the

organization, including the training architecture and change management strategy needed to transition skills into practice.

As an independent consultant, O'Neill conducts defined programs for managing strategic software improvement. These include implementing an organizational software inspections process, directing the national software quality experiment, implementing software risk management on the project, conducting the project suite key process area defined program, and conducting global software competitiveness assessments.

In his IBM career, O'Neill completed assignments in management, technical performance, and marketing in a broad range of applications including space systems, submarine systems, military command and control systems, communications systems, and management decision support systems. He was awarded IBM's Outstanding Contribution Award three times.

O'Neill served on the Executive Board of the IEEE Software Engineering Technical Committee and as a Distinguished Visitor of the IEEE. He is a founding member of the Washington, DC Software Process Improvement Network (SPIN) and the National Software Council (NSC) and serves as the President of the Center for National Software Studies (CNSS). O'Neill has a Bachelor of Science degree in mathematics from Dickinson College in Carlisle, Pennsylvania.

2.9 Critical Success Factors (CSF) in SPI Bibliography

Authors

Tomás San Feliu, Suzanne Garcia, and Caroline Graettinger

Reason to Study

How to Implement SPI

- Problem with SPI is not a lack of models
- It's a lack of an effective strategy to successfully implement these models
- We have studied in the SPI literature experiences (12)
- We have collected 155 factors and categorized them

CSF from Bibliography



Figure 11: CSF from Bibliography

Critical Success Factors

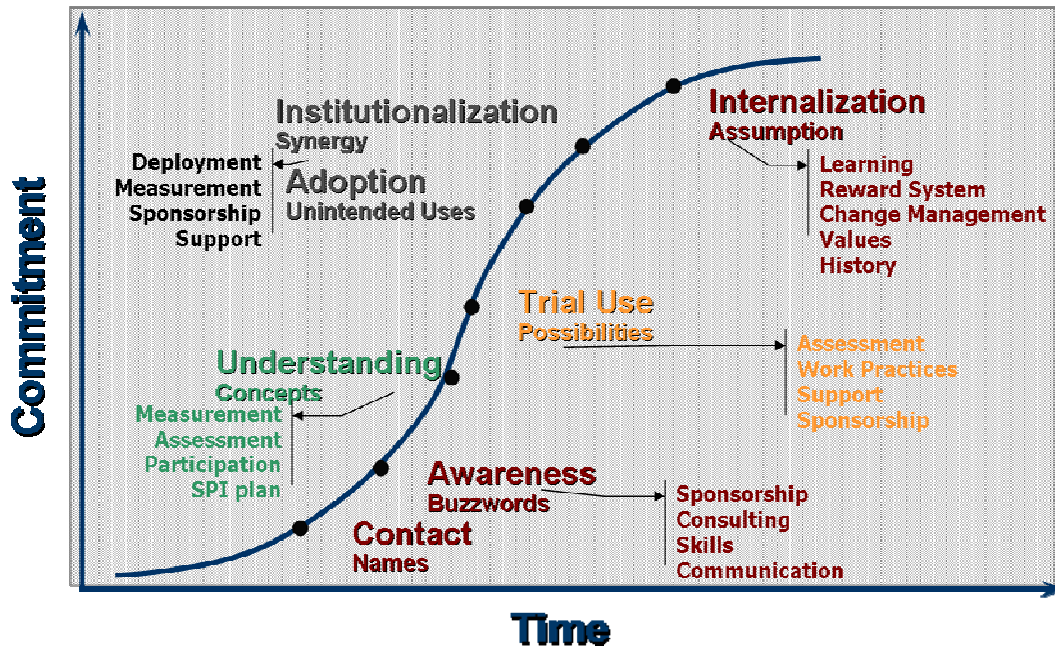


Figure 12: Critical Success Factors

Conclusions

Critical Success Factors

- Show the current experience on SPI implementation, it is focus on Top-risks
- Note that Reward system is only showing as a CSF since 2000
- Need further evaluation and improvement
- Help to improve the SPI implementation

Sponsorship

Problems

- Lack of Management commitment

Factors

- Company's commitment to SPI activities
- Top Level
 - Senior management commitment
 - Senior management spending their time in participating, monitoring and resolving issues

- Big changes start from the top and managerial support is required throughout process change

SPI Action Plan

Problems

- Achieving a Maturity Level become the Goal
- Unrealistic Management Expectations
- Lack of guidelines

Factors

- Goals
 - Clear and relevant SPI goals
 - Unanimity of the business goals
 - Challenging achievable and measurable goals
 - Business orientation
 - Measurable targets set to SPI work
 - Scope definition
- Strategy
 - Formal methodology
 - Flexible and tailored approach to assessment and improvement
 - Improvement should be conducted in small and tested steps
 - Start simple
 - A real project
 - Tailoring improvement initiatives
- Roles and responsibilities
 - Assignment of responsibilities of SPI
 - Creating process action team
 - Responsibilities set for process areas
- Setup
 - Estimating tools
 - Managing the SPI project
 - Stalling on Action Plan Implementation
- Measurement
 - Use the data
 - A goal and knowledge of the current process
 - Measurable targets set to SPI work
 - Set measurable targets to improvement
 - Concern for measurement
 - Metrics

Assessment

Problem

- Have not a baseline reference

Factors

- Assessment is needed before improvement can be conducted
- Assessment planning
- Selection of an appropriate process model
- Reliable assessment method
- Unanimity of the situation
- Providing enhanced understanding
- Frequency of process assessment

Deployment

Problem

- Failing to scale formal process to projects

Factors

- Risk Assessment
- Having a suitable delivery system in place for deploying process
- Keeping the defined process under configuration control
- Synchronization of software parts
- Process ownership
- Formal audit system
- Consulting in use of processes
- Inspections and reviews
- Project post-mortem

Support

Problems

- Lack of support
- Staff turnover
- Lack of resources
- Time pressure

Factors

- Staff
 - Staff time and resources dedicated to SPI
 - Time-Stingy Project Leaders
- Infrastructure
 - Group focus of SPI, Creating process action teams/Change agents and opinion leaders
 - Having a dedicated group responsible for SPI initiative with some full-time members
 - SPI environment support for a long period of time
- Funding
 - Adequate funding
 - Change requires investment and resources
 - External financial support at the beginning
- Resources
 - Providing the resources
 - Availability of company's resources
- Management
 - Executive support
 - Management's support for SPI

Consulting

Problems

- Lack of guidelines in the action plan
- Lack of knowledge on the focus area

Factors

- External guidance of the SPI effort
- External guidance and mentoring

Skills

Problems

- Inadequate Training

Factors

- Awareness
- Training and mentoring
- SPI
 - Understanding essential SPI concepts

- SPI related training
- Ensuring adequate knowledge of the model
- Process-related training

Work Practices

Problems

- Expecting defined procedures to make people interchangeable

Factors

- Defined processes
- Exploitation of existing knowledge
- Baselining the current way of working
- Standards and procedures
- Documentation of the process

Reward System

Problem

- Gradual change

Factors

- Establish incentives
- Reward schemes

Participation

Problems

- Inexperienced staff
- Lack of knowledge

Factors

- Everyone must be involved
- Involvement of the key personnel
 - Involved leadership
 - User involvement
 - Experienced staff
 - Developer's involvement
- SPI people highly/well respected
- Presence of champions

- Internal leadership

Communicate

Problems

- Lack of visibility into SPI process

Factors

- Publish widely
- Facilitate debate
- Communication to entire personnel
- Encouraging communication

Change Management

Factors

- Conscious effort and periodic reinforcement
- Process change needs to be continuous
- Maintenance of momentum
- Organizational politics

Learning

Factor

- Create learning environment
 - Continual learning and growth
 - Exploration of new knowledge

Values

Factors

- Changing the mindset of management and technical staff
- Inertia
- Cultural awareness

History

Factor

- Negative/Bad experience

3 Process Improvement Approaches and Models

3.1 An Experience on Implementing the CMMI in a Small Organization Using the Team Software Process

Authors

Miguel A. Serrano, Carlos Montes de Oca, and Karina Cedillo

Abstract

In this paper we describe an experience of how the Team Software Process (TSP) was used as the base for implementing an SPI initiative in a small software development company. Initially, the SW-CMM was chosen as the reference model for process improvement, and the IDEAL model was chosen as a guide and lifecycle model for the organizational improvement initiative for planning and implementing the improvement actions. The TSP to SW-CMM gap analysis and the Capability Maturity Model-Based Assessment for Internal Process Improvement (CBA IPI) were used to self diagnose and guide the SPI implementation efforts. Later, when the CMMI was released in 2002, the organization used it as the new reference process model. Since then, some gap analyses and internal evaluations using the Standard CMMI Appraisal Method for Process Improvement (SCAMPI) have been performed to help guide the SPI activities.

Introduction

In the last two decades, software process improvement (SPI) has become an important topic in theoretical software engineering and for software companies who are trying to be competitive in this challenging global market. One of the main SPI goals is to produce quality software on time, under budget, and with the desired functionality. Quality in SPI is based on the premise that mature and capable processes generate quality products. The theory of software quality is based mainly on the work developed in the last decades by Shewhart, Deming, Crosby, Juran, and Humphrey [Shewhart 31, Deming 86, Crosby 79, Juran 88, Humphrey 95]. Towards this end, in the last few years different efforts, initiatives, methodologies, models, and standards such as Six-Sigma Software, Agile Methodologies,

EIA/IS 731, ISO/IEC 15504 (SPICE), SW-CMM and CMMI have been developed [Tayntor 03, Cockburn 02, Sheard 01].

SW-CMM was proposed by the SEI at Carnegie Mellon University to address and solve some of the problems of the so called “software crisis”[Paulk 98]. SW-CMM is a reference model for process improvement. SW-CMM also is considered a roadmap for SPI, and it has had a major influence in the software community around the world. In 2002, the SEI released CMMI V1.1 as the new reference model replacing the SW-CMM. CMMI is now the de-facto standard model of reference for process improvement around the world.

The Team Software Process (TSP), also developed at the SEI, was designed to facilitate superior performance of software development teams. The TSP is a methodology that helps organizations implement processes and best practices, included in the SW-CMM and CMMI models, at the team and project level.

The CMMI, as well as many other reference models and methodologies for SPI, have been successfully implemented in many large software development and maintenance companies. However, around the world many small software organizations are struggling to implement those reference models. Among other factors, the reason is the lack of guidance for implementing an SPI initiative in small environments, as well as the cost and time that it takes. The TSP has been used as a methodology for SPI in small settings and several success stories have been reported [Webb 99, McAndrews 00].

In this paper, we describe an experience on how TSP was used as the base for implementing a SPI initiative in a small software development company. Initially, the SW-CMM was chosen as the reference model for process improvement, and the IDEAL model [Gremba 97, McFeeley 96] was chosen as a guide and lifecycle model for the organizational improvement initiative for planning and implementing the improvement actions. The TSP to SW-CMM gap analysis [Davis 02b] and the Capability Maturity Model Based Assessment for Internal Process Improvement (CBA IPI) [Dunaway 96] were used to self-diagnose and guide the SPI efforts. When the CMMI was released in 2002, the organization used it as the new reference process model. Since then, some gap analyses [McHale 04] and internal evaluations using the Standard CMMI Appraisal Method for Process Improvement (SCAMPI) [SEI 01] have been performed to help guide the SPI activities.

The paper is organized as follows: section 2 presents company environment and strategy for implementing the SPI; section 3 describes the SW-CMM and CMMI implementations using TSP; finally, section 4 presents the lessons learned and conclusions.

The Company Environment and Strategy for Implementing the SPI

QuarkSoft[®] is a small software development company with about 50 Software Engineers, 10 administrative staff, and 2 SEPG and QA staff. All projects in this company follow the Personal Software Process (PSP) and TSP methodologies, and all engineers are trained in

PSP and TSP as part of the induction process when they are hired. The size of the projects that QuarkSoft has developed in the past are up to 76,158 lines of code (LOC) with a duration in the range of a couple of weeks to 8,300 hours. The number of people participating in the teams has ranged from 3 to 14 people. Some of the programming languages and environments in which the projects have been developed include Progress, .Net, C++, Java, and J2EE.

From the beginning, the company was created with the goal and commitment to develop quality software. To reach this goal, the company decided to launch an SPI initiative and use the SW-CMM model as guidance and the TSP as the instance to implement SW-CMM. In June 2002, the SEI released a gap analysis between TSP and the SW-CMM [Davis 02b], where it was argued that an organization could build a SW-CMM-based SPI effort based on TSP. Consequently, QuarkSoft decided to start a software process initiative to consolidate the implementation of SW-CMM.

The IDEAL model was selected to guide the implementation of the SPI. To start the SPI effort three questions were addressed [Humphrey 98]:

1. Why should we improve? The company executives and stakeholders had commitment to quality since the creation of the company. There was a special interest for improvement and not for advertising or getting compliance with a specific model or rating system. The compelling business reason for improving the software processes was to maintain a strong competitive position based on the ability to follow a mature process and produce a quality product (e.g., low defects, deliver under schedule and under budget, higher productivity, reduced cycle time, and estimation accuracy).
2. What must we do to achieve a superior software capability? The way to answer this question in QuarkSoft was to look at the available models and international standards such as the ISO 9001 (SPICE 15504) and SW-CMM [Sheard 01] and compared the actual practices used at QuarkSoft to the recommendations provided by those models. At QuarkSoft, it was decided to use SW-CMM [Paulk 93, Paulk 94] as the reference descriptive model and to do an internal assessment (CBA IPI-like) to determine what needed to be improved.
3. How do we make the improvements? To perform the SPI initiative, PSP and TSP [Humphrey 00] were used as the prescriptive processes for implementing the SW-CMM at the individual and team level respectively. An SEPG was created at the organization level to help establish the definition, control, and improvement tasks needed to launch the improvement program.

When the SPI initiative started it was clear that the implementation would be anything but easy. It is well documented in the literature the difficulties and challenges to implement an SPI initiative in large and small organizations [Paulk 98, Davis 02a, Kelly 99, McGarry 02, Ward 00]. Therefore, it was decided that a good planning of the strategy, as well as a good tracking and monitoring methodology for the execution of the plan at the time of implementation was necessary. Small organizations implementing a process improvement

initiative have some differences and challenges when compared to large organizations [Kelly 99]. The ratio of effort and investment in small organizations is significant compared to that of large organizations, but generating cultural change is faster organization-wide for small organizations.

By the time the SPI initiative started, TSP was already widely used in the company by all teams working in the development and maintenance of software. In addition, the management team of the company had been using TSP as a support methodology for running the company [Montes de Oca 02, Serrano 02].

One more motivation to implement the SW-CMM using TSP was the SEI release of the TSP and SW-CMM gap analysis [Davis 02b] that promised a quick implementation of high SW-CMM levels of maturity in a short period of time [Davis 02a].

When the CMMI was released in 2002, the organization used it as the new reference process model. Since then, several gap analyses [McHale 04] and internal evaluations using SCAMPI [SEI 01] were performed to help guide the SPI activities.

To help deal with the SPI implementation complexity, an SEPG team of two people was created. The TSP offers flexibility to be used by teams others than software teams. Therefore, it was natural to think of TSP as the supporting methodology for the management, planning, and tracking activities of the SEPG team. The SPI initiative would be treated like a project and the team would be the SEPG team. Since the TSP was designed for software development teams, the SEPG team customized it for their specific requirements.

SW-CMM and CMMI Implementation Using TSP

In this section, we describe the SW-CMM and CMMI implementations that followed the IDEAL model as the SPI life cycle model and used the TSP as the process foundation. We describe the implementation throughout all five phases of the IDEAL model. In this section, we focus more on the SW-CMM implementation (first improvements cycles), but the same approached has been used for the CMMI implementation (later improvements cycles).

Initiating

In QuarkSoft, the initiating phase (i.e., reason to improve and secure executive commitment) was clearly covered. The SPI initiative came from the executives of the company; therefore, it was not necessary to do any buy-in or work to get executive support and commitment. The company from the beginning was created with the goal and commitment to develop quality software with emphasis on improvement and not for advertising or getting compliance with a specific model or rating system. Having a mature organization was seen as the principal factor for maintaining a strong competitive position. The executives of the company also had an idea of the amount of work, effort, and investment necessary to implement the SPI program and were committed to support the SPI initiative.

Diagnostic

In order to characterize the initial state of the organization software processes as well as to assess the improvements done through the SPI cycles, a reduced version of a CBA IPI assessment was applied, and later when the CMMI was released, SCAMPI Class B and ARC C were performed.

In the diagnostic phase and in each improvement cycle, the idea was to characterize the real current state of the implemented processes and determine the real gap of the TSP implemented in the organization and the SW-CMM since some process deviation from the TSP could exist. As an example, the following figure shows the results of the initial self-assessment. The charts show a summary of the percentage of processes areas already addressed in the existing processes in the organization, and the results are grouped by SW-CMM level.

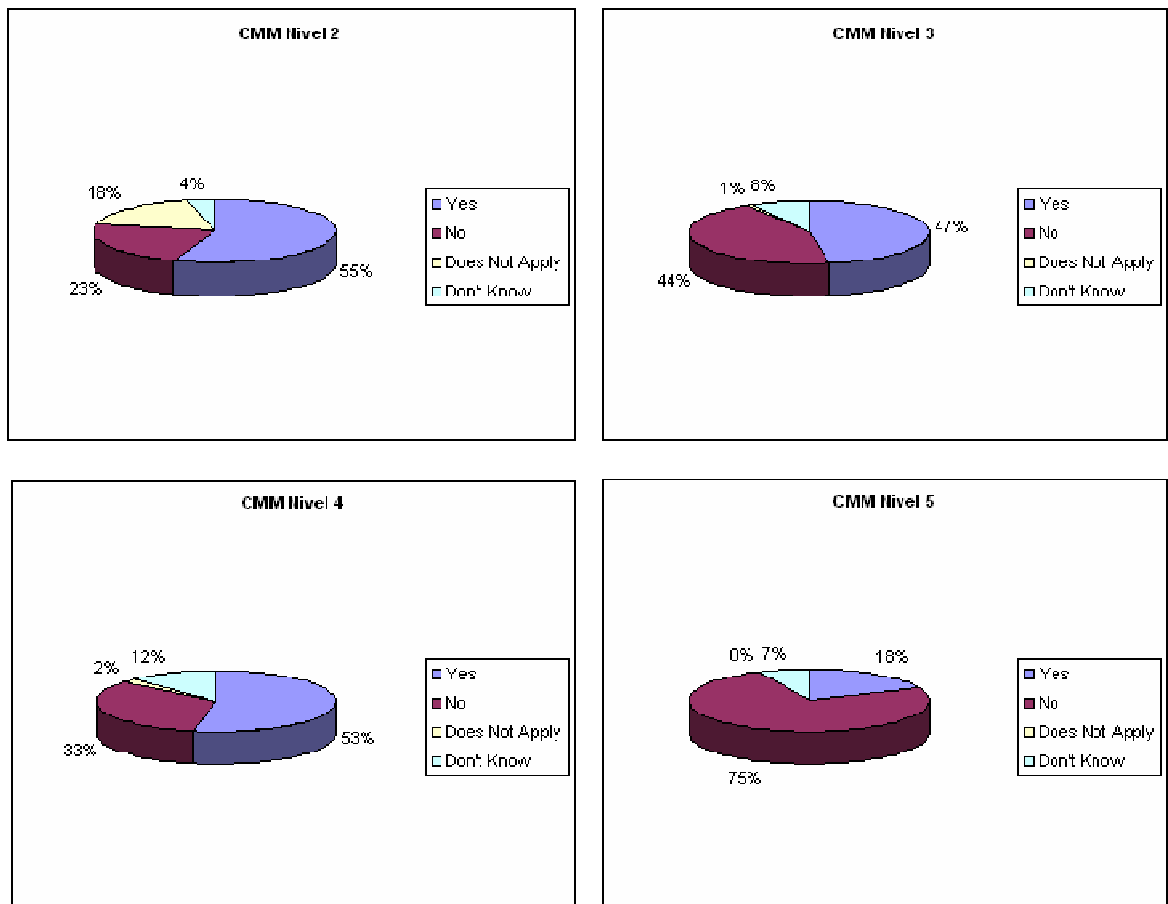


Figure 13: Self-Assessment Results

From the initial diagnostic and findings, it was clear that a lot had to be done in several process areas especially at the organizational level. One of the recommendations was to address the KPAs of SW-CMM level 2, then level 3, then level 4, and then Level 5. One more

recommendation of the diagnostic was to follow and build up on the existing processes dictated by PSP and TSP and the ones developed in the organization. From the diagnostic and the TSP-SW-CMM gap analysis it was discovered that there were some practices from the TSP process that were not followed faithfully; therefore, one of the recommendations was to make sure the TSP was closely followed by all engineers.

Establishing

TSP was used as a tool to help the SEPG team to make the planning and tracking of the SPI implementation project. The IDEAL model recommends to set the priorities, then develop the approach, and finally plan actions. The outcome of the establishing phase is a detailed implementation plan that includes several items, such as specific actions, schedule, milestones, deliverables, decision points, resources, responsibilities, measurements, tracking mechanisms, and risk and mitigation strategies. Clearly, the TSP Launch, when applied to software projects, produces a detailed plan with very similar features. Therefore, given the organizational experience with using TSP, it was natural to think of the TSP as the supporting process for planning, tracking, and controlling the SPI implementation activities of the SEPG team. A TSP launch was performed to define the detailed plan for the establishing phase. Table 8 shows a summary of the number of major activities and estimated effort (time) for the implantation of each level of SW-CMM according to the plan developed in the TSP Launch.

In Table 8, the row called Set infrastructure means the set of activities necessary as prerequisites before starting the implementation of all the activities related with SW-CMM levels; some of the activities included SW-CMM training to the SEPG team, development of the Organization Standard Software Development process, and definition of processes for the operation of the SEPG for the SPI implementation. The initial effort estimation for level 4 of SW-CMM was surprisingly short. The reason was that TSP covers most of the requirements of the QPM and SQM KPAs of the SW-CMM level 4.

Table 8: Activities and Time Estimation for the SPI Implementation

	# Major Activities	Estimated Time (in Hours)
Set Infrastructure	9	300
Level 2	55	668
Level 3	59	1274
Level 4	8	127
Level 5	41	609

Acting

Since the TSP Launch was used in the establishing phase as the supporting tool to get the detailed plan to implement the SPI initiative, during the acting phase it was just a matter of executing the plan following the strategy of the TSP Launch. The strategy set the SW-CMM and CMMI implementation as a project with multiple cycles. Each cycle would last roughly 4 weeks. At the end of the cycle, a post mortem would be conducted and a relaunch held at the start of each new cycle. In addition, weekly status meetings would be held to control and manage the plan.

There were a variety of activities to perform for implementing the KPAs and PAs, such as the definition of processes and policies, identification of the set of tools to support the implementation of some KPAs and PAs (e.g., the Software Configuration Management tool), and definition of the database for the organization's process library.

The resources devoted to implement the plan were the SEPG team and all the software engineers. The software engineers were grouped in team working groups (TWGs). The TWGs performed activities such as researching, creating the first version of processes, developing standards, and piloting the new processes. The SEPG group was mainly responsible for activities such as coordinating the initiative, defining infrastructure, defining the process, reviewing the process, and training.

The steps recommended by the IDEAL acting phase (i.e., Create Solution, Pilot/Test Solution, Implement Solution) were followed. In addition, the SEPG team developed a process to define processes and policies (PDP). The PDP included elements such as process templates, standards templates, checklists, and inspections. This process was established to assure consistency since not only the SEPG team but also the TWGs would be defining processes and policies. The PDP also helped to collect measurements on the time spent developing processes. These measurements were used to adjust the time estimated for implementing each KPA and each PA.

One of the challenges for the piloting step was to find the right time for any ongoing project to pilot a new artifact/asset. Since the artifact/asset could belong to some specific phase of the software life cycle and it was possible that no project was in that phase, it was necessary to wait until a project was in the corresponding phase for testing the artifact/asset.

During the execution of the plan, following the TSP multi-cycle structure with relaunches and weekly meetings it was possible to detect estimation errors based on the historical data and then apply corrective actions. Therefore, it was possible to keep the SW-CMM and CMMI implementation project under control.

Learning

Measurements and indicators were being collected and analyzed during post mortems in an incremental way as learning experiences as to how things were working and how things could have been done more efficiently and effectively. Additionally, in the weekly meetings

and relaunches, the business needs identified during the initiating phase were revisited to see if they had been addressed.

Lessons Learned and Conclusions

This early experience in using the TSP as a foundation for implementing the SW-CMM and CMMI in a small organization has shown that TSP makes the SW-CMM and CMMI implementation easier. TSP has good coverage of the SW-CMM and CMMI at a project level. However, it is important to assure that the TSP and the PSP are followed faithfully by all the teams and software engineers in the organization. To this end, the CBA IPI self mini-assessment and SCAMPI B appraisals proved to be a convenient and useful diagnosis tool. In addition, the SEI release of the SW-CMM-to-TSP gap analysis as well as the TSP-to-CMMI mapping were invaluable tools to define the strategy and to do the detailed planning of this TSP-based SW-CMM and CMMI SPI initiative.

As mentioned above, TSP has good coverage of SW-CMM and CMMI at the team level. Therefore, most of the SW-CMM and CMMI implementation has to be at the organizational level. Even if all the teams are following the TSP by the book, there are still many organizational aspects of the SW-CMM and CMMI that have to be covered.

It is important to note that careful planning for piloting the new processes is critical, especially in small settings. Without this planning, it is very likely that a process will not be piloted for a long time because no suitable projects will be available. In addition, not all projects are good candidates to pilot new processes and this adds an additional constraint for pilot planning. Therefore, good pilot planning is essential for implementing the SPI initiative on time.

Implementing a SPI initiative based on the SEI-sponsored family of products was a good decision. This family of products includes models (such as SW-CMM and CMMI), an implementation lifecycle model (IDEAL), assessment and appraisal methods (CBA-IPI and SCAMPI), and specific processes (TSP and PSP). These products provide a consistent and complementary set of tools that facilitates the implementation of a SPI initiative. In particular, TSP has good coverage of the SW-CMM and CMMI models at project level, which represents a huge shortcut for implementing such models.

References

[Cockburn 02]

Cockburn, Alistair. *Agile Software Development*. Boston, MA: Addison-Wesley, 2002.

[Crosby 79]

Crosby, Phillip B. *Quality Is Free*, New York, NY: McGraw-Hill, 1979.

- [Davis 02a]** Davis, Noopur. "Using the TSP to Implement the CMM," *CROSSTALK: The Journal of Defense Software Engineering* 16, 9 (2002): 30-38.
- [Davis 02b]** Davis, Noopur & McHale, James. *Relating the Team Software Process (TSP) to the Capability Maturity Model for Software (SW-CMM)* (CMU/SEI-2002-TR-008, ADA404970). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2002.
<http://www.sei.cmu.edu/publications/documents/02.reports/02tr008.html>
- [Deming 86]** Deming, W. Edwards. *Out of the Crisis*. Cambridge, MA: MIT Press, 1986.
- [Dunaway 96]** Dunaway, Donna K. & Masters, Steve. *CMM-Based Appraisal for Internal Process Improvement (CBA IPI): Method Description* (CMU/SEI-96-TR-007, ADA307934). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1996.
<http://www.sei.cmu.edu/publications/documents/96.reports/96.tr.007.html>
- [Gremba 97]** Gremba, Jennifer & Myers, Chuck. "The IDEAL Model: A Practical Guide for Improvement." *SEI Bridge* 1, 3 (1997): 1-6.
- [Humphrey 95]** Humphrey, Watts. *A Discipline for Software Engineering*. Boston, MA: Addison-Wesley, 1995.
- [Humphrey 98]** Humphrey, Watts. "Three Dimensions of Process Improvement. Part I: Process Maturity." *CROSSTALK: The Journal of Defense Software Engineering* 11, 2 (1998): 14-17.
- [Humphrey 00]** Humphrey, Watts. *Introduction to the Team Software Process*. Boston, MA: Addison-Wesley, 2000.
- [Juran 88]** Juran, Joseph M. *Juran on Planning for Quality*. New York, NY: MacMillan, 1988.
- [Kelly 99]** Kelly, Declan & Culleton, Bill. "Process Improvement for Small Organizations." *IEEE Computer* 32, 10 (1999): 41-47.

- [McAndrews 00]** McAndrews, Donald. *The Team Software Process: An Overview and Preliminary Results of Using Disciplined Practices* (CMU/SEI-2000-TR-015, ADA387260). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2000.
<http://www.sei.cmu.edu/publications/documents/00.reports/00tr015.html>
- [McFeeley 96]** McFeeley, Robert. *IDEAL: A User's Guide for Software Process Improvement* (CMU/SEI-96-HB-001, ADA305472). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1996.
<http://www.sei.cmu.edu/publications/documents/96.reports/96.hb.001.html>
- [McGarry 02]** McGarry, Frank & Decker, Bill. "Attaining Level 5 in CMM Process Maturity." *IEEE Software* 19, 6 (2002): 87-96.
- [McHale 04]** McHale, James & Wall, Daniel S. *Mapping TSP to CMMI*. (CMU/SEI-2004-TR-014). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2004.
<http://www.sei.cmu.edu/publications/documents/04.reports/04tr014.html>
- [Montes de Oca 02]** Montes de Oca, Carlos & Serrano, Miguel. "Managing a Company Using TSP Techniques." *CROSSTALK: The Journal of Defense Software Engineering* 15, 9 (2002): 17-21.
- [Paulk 93]** Paulk Mark C. et al. *Capability Maturity Model, Version 1.1*, (CMU/SEI-93-TR-24, ADA263403). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1993.
- [Paulk 95]** Paulk Mark C. et al. *The Capability Maturity Model Guidelines for Improving the Software Process*. Reading, MA: Addison-Wesley, 1995.
- [Paulk 98]** Paulk, Mark C. "Using the Software CMM in Small Organizations," 350-361. *Proceedings of the Eighth International Conference on Software Quality*, Portland, Oregon, Oct. 13-14, 1998,
- [SEI 01]** Software Engineering Institute, Assessment Method Integrated Team. *Standard CMMI Appraisal Method for Process Improvement Version 1.1: Method Definition Document* (CMU/SEI-2001-HB-001, ADA3399204). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2001.
<http://www.sei.cmu.edu/publications/documents/01.reports/01hb001.html>

- [Serrano 02] Serrano, Miguel & Montes De Oca, Carlos. "An Experience Report on Using The Team Software Process." *7th International Symposium on Future Software Technology (ISFST 02)*. Wuhan, China, SEA, Japan, Oct 20-25, 2002.
- [Sheard 01] Sheard, Sarah A. "Evolution of the Frameworks Quagmire." *IEEE Computer* 34, 7 (2001): 96-98.
- [Shewhart 31] Shewhart, Walter A. *Economical Control of Quality of Manufactured Product*. New York, NY: D Van Nostrand Co., 1931.
- [Tayntor 03] Tayntor, Christine B. *Six Sigma Development Software*. Boca Raton, FL: CRC Press, 2003.
- [Ward 00] Ward, Robert P.; Fayad, Mohamed E.; & Laitinen, Mauri. "Thinking Objectively: Software Process Improvement in the Small." *Communications of the ACM* 43, 4 (2000): 105-107.
- [Webb 99] Webb, David & Humphrey, Watts. "Using the TSP on the TaskView Project." *CROSSTALK: The Journal of Defense Software Engineering* 12, 2 (Feb. 1999): 3-10.

Biographies

Miguel A. Serrano

Department of Computer Science at the Center for Mathematical Research (CIMAT), Mexico

masv@cimat.mx

Miguel Serrano is a researcher and consultant in software engineering in the Department of Computer Science at the Center for Mathematical Research (CIMAT), Mexico. His research interests include software quality and software process improvement, software measurements and statistical process control, software reliability, and reengineering of legacy systems.

Miguel Serrano is an SEI-authorized PSP Instructor, SEI-trained TSP Coach, and CMMI/SCAMPI Lead Appraiser. He also is an authorized PSM Instructor by the PSMSC. Serrano received a master of science in business information systems and decision sciences, a master's in system science, and a PhD in computer science from Louisiana State University.

Carlos Montes de Oca

Department of Computer Science at the Center for Mathematical Research (CIMAT), Mexico

moca@cimat.mx

Carlos Montes de Oca is a researcher in the Department of Computer Science at the Center for Mathematical Research (CIMAT). He is an SEI-authorized PSP Instructor and an SEI-trained TSP Coach. He has been involved in several PSP and TSP projects in both academia and industry. He has published several articles in SPI. His current research and professional

interests include software process improvement and software quality. Montes de Oca holds a PhD in computer science from Louisiana State University.

Karina Cedillo

QuarkSoft S.C.

kcedillo@quarksoft.net

Karina Cedillo is the SEPG lead for QuarkSoft S.C., where she has been working on the SW-CMM and CMMI implementation effort since 2001. She is an SEI-authorized PSP instructor and has trained most of the software engineers at QuarkSoft. She has more than nine years of experience in software development and SEPG-related activities. She participated in the early trials of TSP while she was enrolled in graduate school and has experience using TSP for software projects. Cedillo has a master's degree in software engineering from Carnegie Mellon and a BS in computing systems from ITESM (Mexico).

3.2 MoProSoft[®]: A Software Process Model for Small Enterprises

Author

Hanna Oktaba

Abstract

The high cost of SW-CMM and CMMI adoption in small enterprises and the need for a national standard were the basic reasons to develop a new software process model for the Mexican software industry. The model, which is in Spanish, is called Modelo de Procesos para la Industria de Software (MoProSoft). This model builds on the well-known practices of SW-CMM, ISO 9000:2000, PMBoK and others, and offers a new process structure, some new process documentation elements, a more precise process relationship, and an explicit process improvement mechanism. The model is complemented with the process assessment method EvalProSoft, which is based on the recommendations of ISO IEC 15504 Part 2. The process model and the assessment method were applied to four small enterprises that had a typical Mexican software industry company profile. The results of those trials are presented in this paper.

Introduction

In 2002, the Mexican government started a program to promote the software industry. Of the IT companies in Mexico 92% are small and medium-sized (with less than 100 people) [Industria Mexicana de Software 05]. The average process capability level of the software development companies is 0.9 (in 0 to 5 ISO/IEC 15504 [ISO/IEC 03] scale) [Estudio del nivel 05]. To increase competitiveness the industry needs to adopt a massive software process improvement program, but their resources, capital, and trained people are limited. Providing low cost software process improvement mechanisms is a government strategy for reaching this goal.

The selection of the software process reference model and the assessment method, accessible by local industry, was the first problem to resolve. The government and the industry defined the selection criteria (see the section Evaluation of Selected Standards and Reference Models) which were applied to evaluate the suitability of the most popular standards and reference models: SW-CMM [Paulk 95], CMMI [Chrissis 03], ISO/IEC 12207 [ISO/IEC 95], ISO 9000:2000 [ISO 00], ISO/IEC 15504. The conclusion of this analysis was: None of those models fulfills all the criteria. The section Evaluation of Selected Standards and Reference Models contains the results of the model evaluation.

As a consequence, the Mexican government decided to develop the national standard of MoProSoft[®] software process reference model [MoProSoft 03] and the EvalProSoft process assessment method [EvalProSoft 04]. The basic requirements for their definition were

meeting the criteria C1-C5 (defined in the next section of this paper) and conforming to the ISO/IEC 15504 Part 2.

Evaluation of Selected Standards and Reference Models

The suitability criteria for the software process reference model and assessment method for the Mexican industry were defined as follows:

- C1. Proper for small and medium-sized enterprises (SME) with low maturity levels.
- C2. Not expensive to adopt and assess.
- C3. Permissible as a national standard.
- C4. Specific for software development and maintenance organizations.
- C5. Defined as a set of processes based on internationally recognized practices.

Those criteria were applied to evaluate the suitability of the selected standards and models. Table 9 summarizes the evaluation results. A “Yes/No” value means that the standard or model fulfills/does not fulfill the criteria. The question mark (?) means that there is no evidence for the decision being made.

Table 9: Model Comparison

Model	C1	C2	C3	C4	C5
ISO 9000: 2000	Yes	Yes	Yes	No	No
CMM /CMMI	Yes	No	No	Yes	Yes
ISO/IEC 12207	?	?	Yes	Yes	Yes
ISO/IEC 15504	?	?	Yes	Yes	No

The ISO 9000:2000 is not specific for software development organizations (C4) and it is not defined as a set of processes (C5). On the other hand, examples exist of its being adopted by Mexican SME at reasonable costs (C1, C2) and it is already a national standard (C3).

SW-CMM or CMMI models fulfill the C1 criteria because they apply to organizations of any size, independent of the organization’s maturity level starting point. Also they are specific for software development entities (C4) and defined as a set of (key) process areas (C5). Nevertheless, the cost of its adoption and assessments is one of its drawbacks (C2). Finally, due to the Mexican law, those models cannot be accepted as national standards (C3).

The problem with ISO/IEC 12207 and ISO/IEC 15504 is that both have suffered significant modifications during the last three years. In particular, ISO/IEC 15504 Part 5: A Process Assessment Model Sample has not yet been delivered. There are few experiences of its adoption and assessment [El Emam 97], so its suitability for SME (C1) and its adoption costs (C2) are unknown. Both could be national standards (C3) and are specific for software

development organizations (C4). ISO/IEC 12207 is defined as a set of processes (C5), but in the case of ISO/IEC 15504 a specific software process reference model is not addressed (C5).

The high cost of adopting SW-CMM and CMMI⁶ [Konrad 02] and the need for a national standard were the basic reasons for developing a new software process model. The model, in Spanish, **Modelo de Procesos para la Industria de Software** (MoProSoft), summarizes the well-known practices of SW-CMM, ISO9000:2000, PMBoK and others, offering a new process structure, some new process documentation elements, a more precise process relationship, and explicit process improvement mechanisms. Multiple sections of this paper briefly describe those characteristics of MoProSoft.

The Process Assessment section introduces a very short description of the EvalProSoft process assessment method, based on the recommendations of ISO/IEC 15504 Part 2, and the Trial Experience section summarizes the results of the MoProSoft and EvalProSoft tests on four small enterprises with a typical profile of the Mexican software industry.

Finally, the Conclusions and Further Work section presents some conclusions about the use of MoProSoft and EvalProSoft and plans for future work.

Process Structure

To define the structure of the process model, first we analyzed the structure of the software development enterprise. Even the micro-enterprise (having less than 10 people) has a top management group that makes decisions concerning the direction of the business. Also, it has a middle management group that is responsible for project and resource procurement and control. Finally, there exists an operations group that develops projects using assigned resources. The members of those groups acknowledge their responsibilities through their assigned roles. Roles have vertical authority alignment and horizontal collaboration relationships between them.

Based on those observations, we decided to group our processes into three categories: Top Management, Management, and Operations. The purpose of this categorization is to provide specialized processes for each functional group. Figure 14 presents the MoProSoft category and process structure in the form of a Unified Modeling Language (UML) package diagram.

⁶ From the SEI *Report on the May 2002 CMMI Workshop* (CMU/SEI-2002-SR-005). This report was only made available to workshop participants.

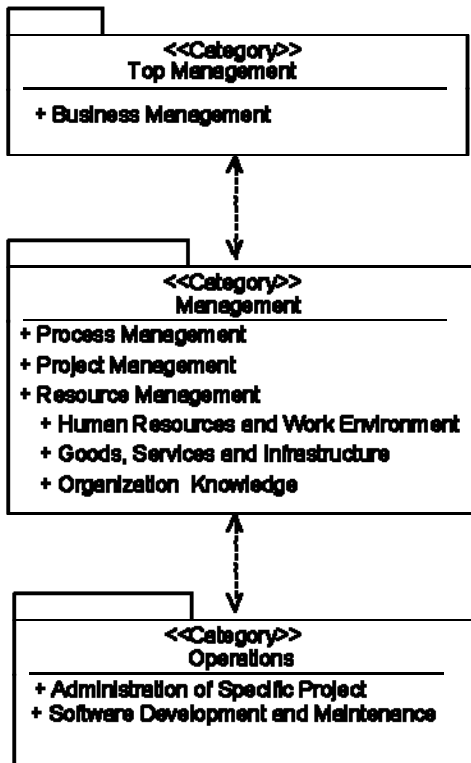


Figure 14: Process Categories

The Top Management category (DIR) includes practices related to business management. It provides the directions for the processes of the Management category, and receives reports from them. This category includes the Business Management process.

The Management category (MAN) includes process, project, and resource management practices, which are in line with the business goals of the Top Management category. The MAN category provides the elements for the performance of the Operations category processes, receives and evaluates the information generated by those processes, and informs the Top Management category of the results. Process Management, Project Management, and Resource Management are the processes that comprise the MAN category. The Resource Management process includes three sub-processes: Human Resources and Work Environment, Goods, Services and Infrastructure and Knowledge of the Organization.

The Operations category (OPE) addresses the practices of software development and maintenance projects. This category performs the activities using the elements provided by the Management category, and delivers the reports and the generated software products. The OPE category includes the Administration of Specific Project process and the Software Development and Maintenance process.

Process Documentation Pattern

The process pattern is a framework of elements needed to document a process. It contains three sections: *General Process Definition*, *Practices*, and *Tailoring Guidelines*.

General Process Definition includes: process name, category, purpose, abstract of process activities, goals, goals indicators, quantitative objectives for goals, roles of responsibility and authority, sub-processes (if any), related processes, inputs, outputs, internal products, and bibliographical references.

The *Practices* section (*a*) identifies the roles involved in the process and the training required, (*b*) describes details of the activities associating them to the process goals and to the roles involved, (*c*) includes the UML activity diagram, (*d*) describes the product's verifications and validations required, (*e*) lists the products that should be incorporated into the organization knowledge base, (*f*) identifies the infrastructure resources necessary to support the activities, (*g*) exemplifies the process measurements for each goal indicator, as well as (*i*) recommended training practices, (*j*) exceptional situation management, and (*k*) the use of lessons learned.

Tailoring Guidelines suggest possible process modifications that should not affect the achievement of the process goals.

Process Relationship

The relationship between the processes is based on the exchange of products and the participation of roles. Each output product generated by the process is explicitly identified as the input product in one or more processes. The internal products are “consumed” by the same process that has generated them.

The process relationship based on role participation means that some roles of one process participate in the activities of other processes. For example, the role of Responsible for the Business Management participates in the validation of the *Process Plan* activity of the Process Management process. This participation is defined in the description of the process activity.

Process Improvement

The process improvement is explicitly included in the model by means of the Process Management process. The purpose of this process is to establish the organization processes based on the *Required Processes* identified in the *Strategic Plan*, as well as to define, plan, and implement the corresponding improvement activities. The *Strategic Plan* is developed by the Business Management process which assures that the process improvement program is in line with the organization business goals.

Each process has defined goal indicators and sets up the measurements practices for them. Periodic results of these measurements and suggested improvements for the process are reported to the Process Management process. Based on the reports of all the processes, the

quantitative and qualitative process performance report is elaborated and delivered to the Business Management process.

Process Assessment

The EvalProSoft process assessment method, based on the recommendations of ISO/IEC 15504 Part 2, was developed. The process reference model for the assessment is MoProSoft. It allows for the capacity level of each process to be evaluated. Additionally, we defined the organization's maturity level in terms of the maximum capacity level achieved by all processes.

Trial Experience

In 2004 we have run four tests on small enterprises with a typical profile of the Mexican software industry.

The purpose of the tests was to evaluate the ease and usefulness of MoProSoft as a software process model for small companies and the cost of the EvalProSoft assessment method.

We conducted the initial assessments to establish the base line capabilities of the enterprise processes. The result was “classic”—between 0 and 1. During the next 6 months our consultants coached the companies on MoProSoft tailoring and adoption. Finally, we applied the second assessment to each company.

All enterprises achieved a 1.08 average increase of the capacity level of all their processes. Table 10 contains the average capacity level increase by process category.

Table 10: Capacity Level Increase by Category

Category	Increase
DIR	1.00
GES	1.20
OPE	0.75
Global	1.08

Table 11 shows the number of employees, total improvement effort in hours and effort per person for each company. The last column includes the average capacity improvement per process. It is interesting to observe the relationship between the effort per person and the average process improvement. The C company has invested the largest number of hours per person and achieved the greatest increment in its process improvement. The average number of employees was 18 and the average effort per person was 21.28 hours during six months.

Table 11: Effort and Improvement Data by Company

Company	Employees	Total effort in hours	Effort per person	Average improvement
A	17	479	28.18	1.00
B	8	199	24.88	1.00
C	17	628	36.94	1.56
D	29	221	7.62	0.78
Average	18	383	21.28	1.08

Conclusions and Future Work

The tests of MoProSoft and EvalProSoft confirmed that both fulfill the C1-C5 criteria mentioned in section 2. Due to the results of this experiment the Secretary of the Economy decided to formally make MoProSoft and EvalProSoft a Mexican standard.

The future work is to monitor the test companies so as to observe their process capability improvement and its relationship to the capability improvement of its internal Process Management process.

Another direction is to increase the number of companies using MoProSoft, and compare their results with the first trial group.

Acknowledgements

I would like to acknowledge the contributions of Claudia Alquicira Esquivel, Angélica Su Ramos, Gloria Quintanilla Osorio, Mara Ruvalcaba López, Francisco López Lira Hinojo, Maria Julia Orozco, Alfonso Martínez Martínez, Maria Elena Rivera López, Yolanda Fernández Ordoñez, Miguel Ángel Flores Lemus, Carlos Pérez Escobar, Jorge Palacios Elizalde, Cecilia Montero Mejía, and Alfredo Calvo to the MoProSoft and EvalProsoft definition.

Also, I want to extend very grateful thanks to Ana Isabel Vázquez Urbina and Claudia Gutiérrez for their excellent management of the trial project, and to the SGA, ARTEC, MAGNABYTE and E-Genium companies for their enthusiasm and dedication during the experiment.

Finally, I would like to thank Mexico's Secretary of the Economy for the confidence in and the support for this project.

References

- [Chrissis 03]** Chrissis, Mary Beth; Konrad, Mike; & Shrum, Sandy. *CMMI: Guidelines for Process Integration and Product Improvement*. Boston, MA: Addison-Wesley, 2003.
<http://www.sei.cmu.edu/cmmi/publications/cmmi-book.html>
- [El Emam 97]** El Emam, Khaled & Briand, Lionel. "Costs and Benefits of Software Process Improvement." *International Software Engineering Network Technical Report 97, 12* (1997).
- [Estudio del nivel 05]** *Estudio del nivel de madurez y capacidad de procesos de la industria de tecnologías de información*.
<http://www.software.net.mx> (2004).
- [EvalProSoft 04]** *EvalProSoft Process Assessment Method*. Internal Report, 2004.
- [Industria Mexicana de Software 04]** Industria Mexicana de Software. *Estudio sobre la industria de software*. <http://www.software.net.mx> (2004).
- [ISO 00]** *ISO 9000:2000 Quality Management Systems – Requirements*. 2000.
- [ISO/IEC 03]** ISO/IEC 15504. *Software Process Assessment, part 2*.
<http://isospice.com/standard/tr15504.htm> (2003).
- [ISO/IEC 95]** *ISO/IEC 12207/IEC 12207:1995 Information Technology - Software Life Cycle Processes. Amd.1*, 2002.
- [MoProSoft 04]** *MoProSoft v. 1.1*. <http://www.software.net.mx> (2004).
- [Paulk 95]** Paulk Mark C. et al. *The Capability Maturity Model Guidelines for Improving the Software Process*. Reading, MA: Addison-Wesley, 1995.

Biography

Hanna Oktaba

Facultad de Ciencias, Universidad Nacional Autónoma de México (UNAM)

ho@fciencias.unam.mx

Dr. Hanna Oktaba received a PhD in computer science from the University of Warsaw, Poland, in 1982. Since 1983, she has been a professor of Computer Science at the Universidad Nacional Autónoma de México (UNAM). Her area of interests include software engineering, object-oriented technology, and software process models. She was a member of the group that founded the Mexican Society of Quality for Software Engineering (Asociación Mexicana para la Calidad en Ingeniería de Software, AMCIS). Currently, she is vice president of this society.

Since 2002, she has been in charge of the MoProSoft and EvalProSoft projects for the Mexican government program PROSOFT. MoProSoft is a software process model for micro and small software development organizations, and EvalProSoft is a process assessment method based on ISO/IEC 15504-2. In August 2005, the model and the method were accepted as the Mexican national standards for the software industry.

3.3 Using Agile Practices and the CMMI to Achieve High Project Management Capability in Small Settings

Author

John Gómez

Abstract

Organizations in small settings may find improvement efforts overwhelming. Agile methods can solve this but also may limit an organization's growth or the compliance of the requisites of a recognized model like the CMMI. Also, many small organizations' main weakness is the lack of sound project management practices. This paper presents research into the proposal that an organization may define and implement a set of agile project management practices that can be strengthened using the CMMI model as a reference. When an organization does this, it may be able to achieve high capability levels in all project management process areas while keeping agile, value-added, and focused processes that may be linked even with quantitative organizational goals.

Introduction

It's recognized that for small companies, projects or teams, process improvement efforts are difficult and frequently overwhelming; however, the need for improvement remains and cannot be ignored. This situation presents a challenge to those dedicated to promoting quality and improvement. Agile methods have been a practicable (and affordable) solution for many small settings since they are easier to adopt than other more rigorous approaches.⁷

A company may be interested in not only improving but also in demonstrating the achievement of a recognized status in the implementation of quality standards or models (e.g., ISO 9001 or CMMI). The likelihood of achieving these statuses using an as is agile method is low. We also are aware that for many small companies the main weakness is the lack of sound project management practices; improvements in this area may result in relevant benefits.

The results of the application of agile practices for project management in improvement projects for small settings have lead us to think that is feasible to use the CMMI model to strengthen defined agile practices. These practices allow an organization to improve its project management activities significantly and, at the same time, be compliant with the requirements of a high capability level (using the continuous representation) for project

⁷ This affirmation needs more discussion but is not in the scope of this paper. Examples of the variety of factors that can be analyzed are average adoption time and quantity and the presentation of the practices described in the method.

management activities. Based on these initial results, we defined more formal research on this subject. This paper is intended to present those initial findings, a formal research scope, challenges, trends, and future developments.

Environment Identification

The research is based on the information gathered up to date from two SPI projects that we've been conducting independently in two different software development organizations. Both organizations have an average of less than 10 people on their project teams and an average project duration that is less than six months. One organization is using SCRUM, and the second one is using Crystal Clear. This last company is also in a project to get ISO 9001:2000 certification and achieve SW-CMM maturity level 2.

Research Objectives and Scope

The main objective is to demonstrate that a company may use the CMMI model as a framework to strengthen defined agile project management practices, improve the project performance, and achieve high CMMI capability levels.

We think that this thesis may be applicable to other process categories, but the data gathered so far corresponds to project management activities, so we decided to limit the current scope research to the project management (PM) category of the CMMI Model.

Our first goal was to articulate capability level 2 compliance with the defined agile practices discussed in this paper. This work is being performed, and current findings will be presented here. Next we are going to articulate capability level 3. Theoretical capability level 2 and capability level 3 compliance will be documented by third quarter 2005, and actual capability level 2 compliance is expected to be confirmed during the official SCAMPI A appraisal planned for one of the organizations in the first quarter of 2006. Compliance for capability level 4 deserves a special discussion (and it is addressed further in this paper).

Project Planning (PP) and Project Monitoring and Control (PMC) were already analyzed for capability level 2 compliance. We are going to continue gathering data for the implementations of these process areas and are going to articulate Risk Management (RSKM) and then Integrated Project Management (IPM) development to prepare for capability level 3 compliance. From the basic PM process areas only Supplier Agreement Management (SAM) is out of the scope of the research. In the advanced process areas, Integrated Supplier Management (ISM) is also out of scope. The special role of Quantitative Project Management (QPM) will be discussed later.⁸

⁸ It is appropriate to clarify that the only discipline in scope of this research is Software Development. This is especially important to the goals for IPM.

Identified Agile Practices for Project Management

The selected agile practices correspond to documented practices as they appear in the agile literature. The common feature of all these practices is that they are defined as non-prescriptive processes in the sense that they're not rigorously defined or documented. They all propose simple and generative rules to perform the activities constrained mainly by an explicit goal and set of values and principles. The application of the agile values and principles is also considered a part of the agile method implementation.⁹

As stated before, the current environments are using SCRUM and Crystal Clear, but the performed practices are very common within most of the agile methods. We think the results of this research are applicable for teams using another agile approach or even a mixture of methods as long as the following practices are defined and implemented:

- iterative and incremental product development life cycle¹⁰
- agile planning as defined by the XP planning game or the Sprint planning practice of SCRUM [Beck 99, Schwaber 04]
- daily inspections as defined by daily stand-ups in XP or Crystal, and daily SCRUM in SCRUM [Beck 99, Schwaber 04, Cockburn 04]
- effort tracking and burn-down as articulated by the Sprint backlog and effort burn-down charts work products in SCRUM [Schwaber 04]
- information radiators as defined in Crystal Clear and Sprint reviews as defined by SCRUM [Schwaber 04, Cockburn 04]

Findings Articulating Capability Level 2 and Capability Level 3 Compliance

As stated before, this work is almost done for capability level 2 for PP and PMC. Planning parameters (e.g., requirements, business value or priorities, available effort, resources needed, estimated effort dedicated to the process, and estimated effort dedicated to support activities) are used and documented in simple spreadsheet-based tools. Estimation is informal and adaptation was required, especially for estimating size, so an estimation method was developed and integrated with the tools in a way that may be consistent with the planning practice as a whole.

Monitoring and control is definitively a strength because the status of tasks, constraints, issues, and risks are tracked daily. Remaining effort also is re-estimated daily and tracked to plot the burn-down charts. At the end of the iterations, formal reviews are conducted and project progress is easily linked to requirements realization.

⁹ For more information on the agile Principles and values see the agile manifesto.

¹⁰ Iterative development is not exclusive nor is the creation of “agilists,” but every agile method promotes and practically requires it [Boehm 88, Highsmith 02, Cockburn 02].

Stakeholder involvement and commitment are also strengths. The agile philosophy promotes customer and team collaboration and involvement in planning and tracking activities. Support practices were needed to provide generic goals fulfillment. CM was not a problem since agile methods promotes continuous integration and that indirectly leads to the existence of a robust CM system. PPQA had to be developed from scratch. MA deserves special comments and will be addressed later.

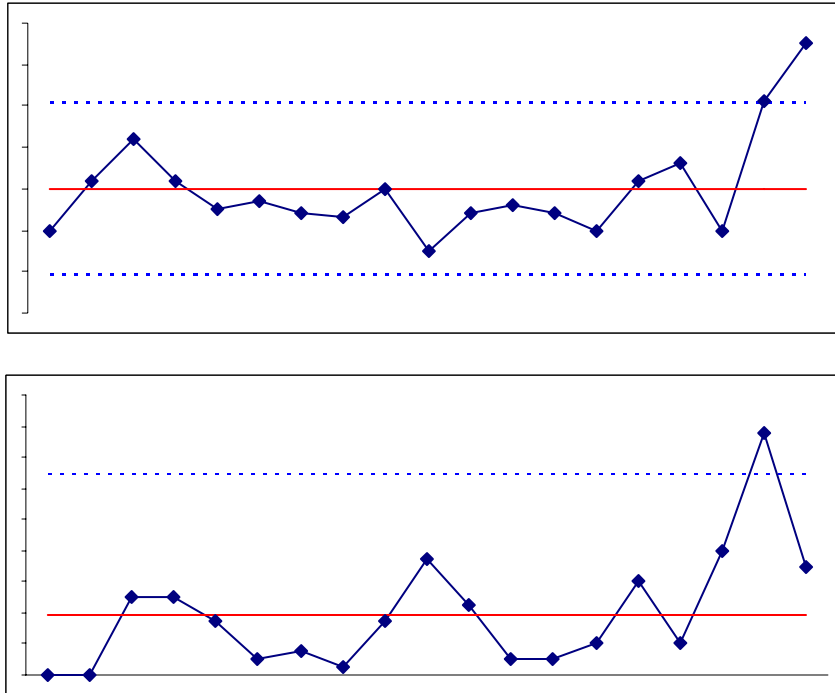


Figure 15: Control Chart Plotted for Weekly Effort Burn-Down

RSKM as a formal practice is another weakness. Risk identification and monitoring were articulated as part of the planning and monitoring activities, but formal definitions required in RSKM are our next step. The challenge here is to develop the process in a way that keeps the project environment agile.

IPM articulation that enables capability level 3 compliance is not considered a significant challenge, at least theoretically. The practices are simple but are documented with tailoring guidelines, training guides, and materials. Tools are considered a critical point and are also documented. Specific practices for process tailoring are analyzed but may be articulated based on the retrospective practice of SCRUM or the reflection in Crystal¹¹.

A Word About Capability Level 4 Compliance

Previously we commented about the special role of MA and QPM. One surprising thing during this work was the amount of data that were collected from the very first iteration due

¹¹ See previous footnote for references on this topic.

the use of tools to track burn-down effort and planned versus actual parameters. The simplicity of the tools used makes it easy to use the gathered historical data for the following planning activities. Metrics were defined and trends are being analyzed in order to determine if the processes may be statistically controlled. A sample plotted control chart for the weekly effort burn-down is presented here (see Figure 15). Still much more analysis is needed to prove the validity of these trends.

Conclusions and Future Developments

Some recognized authors have declared that a balance between agile methods and the CMMI are feasible [Paulk 02, Boehm 03]. We believe that declaration is the basis for this work. We find the current results promising. The work performed so far with the PP and PMC process areas lets us state that these organizations are able to achieve CMMI capability level 2 for these process areas and keep the process agile and adequate for the organizations. The next step is to try to articulate RSKM and IPM development to support capability level 3 for PP, PMC, and RSKM. We want to keep gathering quantitative data and analyze it to verify if the agile project management process can be statistically controlled and linked to quantitative organizational goals (e.g., in estimation or productivity). We think this also may lead to a better scaling of the methods that allow healthy company growth.

References

- [Agile Alliance 01]** Agile Alliance. *Manifesto for Agile Software Development*. <http://www.agilealliance.org> (2001).
- [Beck 99]** Beck, Kent. *Extreme Programming Explained*. Boston, MA: Addison-Wesley, 1999.
- [Boehm 03]** Boehm Barry W. & Turner Richard. *Balancing Agility and Discipline: A Guide for the Perplexed*. Boston, MA: Addison-Wesley, 2003.
- [Boehm 88]** Boehm, Barry W. "A Spiral Model for Software Development and Enhancement." *Computer 21* (May 1988): 61-72.
- [Cockburn 02]** Cockburn, Alistair. *Agile Software Development*. Boston, MA: Addison-Wesley, 2002.
- [Cockburn 04]** Cockburn, Alistair. *Crystal Clear: A Human-Powered Methodology for Small Teams, including The Seven Properties of Effective Software Projects*. Boston, MA: Addison-Wesley, 2004.
- [Highsmith 02]** Highsmith, Jim. *Agile Software Development Ecosystems*. Boston, MA: Addison-Wesley, 2002.

- [Paulk 02]** Paulk, Mark C. "Agile Methodologies and Process Discipline." *CrossTalk: The Journal of Defense Software Engineering* 15, 10 (October 2002): 15-18.
- [Schwaber 02]** Schwaber, Ken, & Beedle, Mike. *Agile Software Development with Scrum*. Upper Saddle River, NJ: Prentice-Hall, 2002
- [SEI 02]** CMMI Product Team. *CMMI for Systems Engineering/Software Engineering/Integrated Product and Process Development/Supplier Sourcing, Version 1.1, Continuous Representation (CMMI-SE/SW/IPPD/SS, V1.1, Continuous)* (CMU/SEI-2002-TR-011, ADA339818). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2002.
<http://www.sei.cmu.edu/publications/documents/02.reports/02tr011.html>

Biography

John Gómez

Practia Consulting

jgomez@pragmaconsultores.com

John Gómez is a senior consultant for SPI projects in Practia Consulting. He is a systems engineer with 15+ years of work in software projects as a programmer, analyst, architect, and project manager. For the last three years, Gómez's main interests have been methodology development and process improvement.

3.4 Business Benefits from Successful Process Measurements

Authors

Rajeev Lal and Niranjana Kumar

Abstract

Infotech achieved CMMI maturity level 5 for software services in three disciplines during 2003. Infotech's Retail Applications division is re-engineering a supply chain logistics application project using Java technologies for one of the top retail companies in the world. The customer, a professional information technology (IT) services organization, accepted Infotech's organizational capability and variances obtained from quantitative process measurements. This was a big benefit, as error level targets set by the customer were beyond the measured organizational capabilities of Infotech.

At the end of the system requirement specifications (SRS) phase, a re-estimation was performed and the size of the development work, measured in function points, was found to be larger than the original estimate by 20.5% on account of hidden functionality not fully visible to an external agency (i.e., Infotech) in the business specification. The productivity, error levels, etc. were measured right from the beginning. As the project progressed, continuous measurement and transparency helped the customer to accept variations that occurred.

Introduction

Infotech achieved CMM maturity level 5 for software services in April 2002 and was appraised at CMMI maturity level 5 for three disciplines in April 2003. The Retail Applications division in Infotech employs about 60 software personnel. Of these, a team of 45 is working on a re-engineering project for a supply chain logistics application for one of the top three retail companies in the world. The project was awarded by the Information Services Company (i.e., the customer), which is a part of the group, and the end user is the group's logistics company (i.e., the user).

The customer, a professional IT services organization, chose to work with Infotech with the objective of getting improved results and measurements in a project to be developed using Java technologies. The customer accepted Infotech's organizational capability and variances as obtained from quantitative process measurements. This was Infotech's biggest benefit.

Project Overview

The project consists of four functional modules and a fifth module that implements common specifications and a framework for the application. The customer supplied a fairly

comprehensive business specification document for all the modules. A framework was supplied by the customer for Java 2 Platform, Enterprise Edition (J2EE) based development. The project is being executed in the offshore mode in India with on-site visits by Infotech personnel at defined milestones. Customer and user personnel also visit the development team in India as needed.

Challenges

The first challenge of the project was that organizational quality data in Infotech was calibrated against lines of code (LOC) during estimation. At the start of this project the unit of size chosen was function points (FPs). Direct translation from LOC to function points for this technology from past projects within Infotech was not possible as only one comparable project was in progress. As development was to be done using a framework, standard data available from the Internet could not be used. Some approximations were made to the best of the knowledge of process experts at Infotech and organizational data was recalibrated for the project in FPs on the basis of the formula, $50 \text{ LOC} = 1 \text{ FP}^{12}$. These approximations were used for tracking productivity and defects and became the basis of a commercial quotation in which the assumptions were stated.

Size Tracking

Work on the project was started in August 2003 with a team of 11 people, which gradually ramped up to 30 people in six months and to 45 people in eleven months. The first activity was the creation of a system requirement specifications (SRS) for each module. After the SRS had been prepared offshore, it was sent to the customer and one iteration of feedback and correction was performed. After the corrections were incorporated, the module leaders from the user and customer companies came to India for discussions and agreed on a final draft of the SRS document.

From this stage of the project, the finalization of the SRS from “final draft” to a “sign off” took much longer than expected. At the end of the SRS phase, which included 4 subsequent rounds of change requests (CR), a re-estimation was performed and the size in FPs was found to be larger than the estimate by 20.5% (see Figure 16). The complete detail of this estimate was shared with the customer, who accepted it in total.

¹² The project is now at an acceptance testing stage. Based on the data available currently, $1\text{FP} = 110 \text{ LOC}$ (approximately).

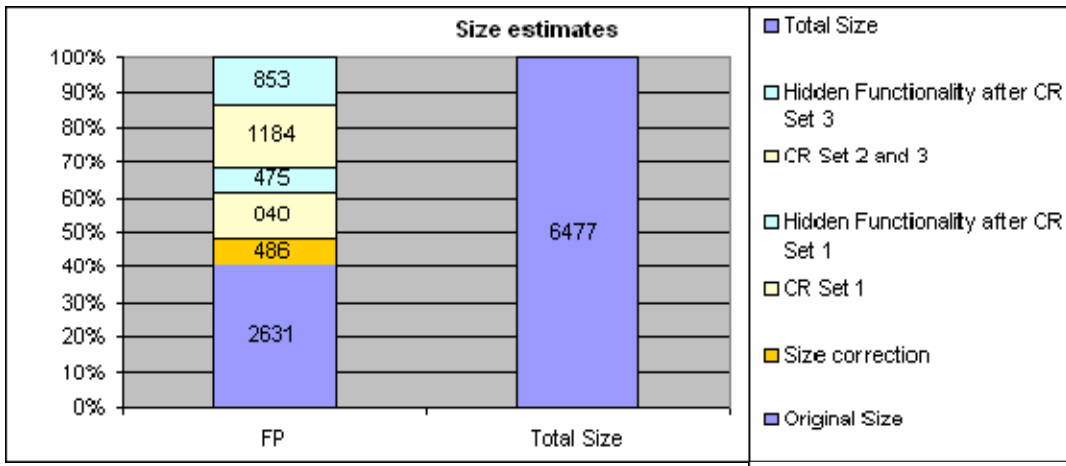


Figure 16: Size Tracking

Analysis showed that a major reason for the increase in size was the “hidden functionality” that was not fully visible to an external agency (i.e., Infotech) in the business specification. Infotech’s organizational norm for estimate divergence was based on LOC. This norm allows for a 18.5% variation, which is calculated as a flat band across all the phases of the development lifecycle. This norm was used as-is for FPs also and the 18.5% variation on the original size helped to account for some part of the “hidden functionality.” The customer accepted this analysis. Of the remaining 20.5% variation, the customer graciously accepted to pay for about 10%. Infotech absorbed the remaining 10%.¹³

Development Process

A modified waterfall process was used on this project. Iterations of reduced scope were planned as early releases right through the project to reduce risks. Requirements management and analysis and design were essentially tool based and centered on the use of Rational Requisite Pro and Rational Rose. Object Oriented Analysis and Design (OOAD) with Unified Modeling Language (UML) was fully employed and the analysis and design methodology was considered successful by the customer too. The requirements traceability process deserves special mention. Requirements were captured in enormous detail, which laid the foundation for subsequent strong change management processes. Traceability of the requirements to design and test cases was another strong process implementation, which was recognized by the client. This traceability served as valuable documentation for the maintenance phase.

Effort and Productivity Tracking

As Infotech had handled a few projects of this size using Java technology, the account manager decided to use international norms obtained from the Web as the basis of delivery

¹³ The entire discussion of FP is based on the assumption that the scope was available at the start of the project. In practice, rework was encountered and the effective FP was higher. However, this has not been considered in calculations.

rate—15 function points per person per month¹⁴. The productivity (FP per person per month) was measured for each module from the beginning of development. In the first three months, average productivity was 10 FPs per person. Most of the members of the team were experienced Java programmers and fresh graduate engineers trained in Java were added to the team after the first 6 months. Productivity dipped to as low as 8 FP per month on one occasion in one module. The latest projections indicate that productivity for the project is expected to vary between just under 10 and up to 12 FP per person month, depending on the module. The most significant reason for lower productivity has been the long ramp up times required for each resource to become fully productive for development. Attrition and efforts required to make up for it were also factors that lowered productivity at the start of the project.

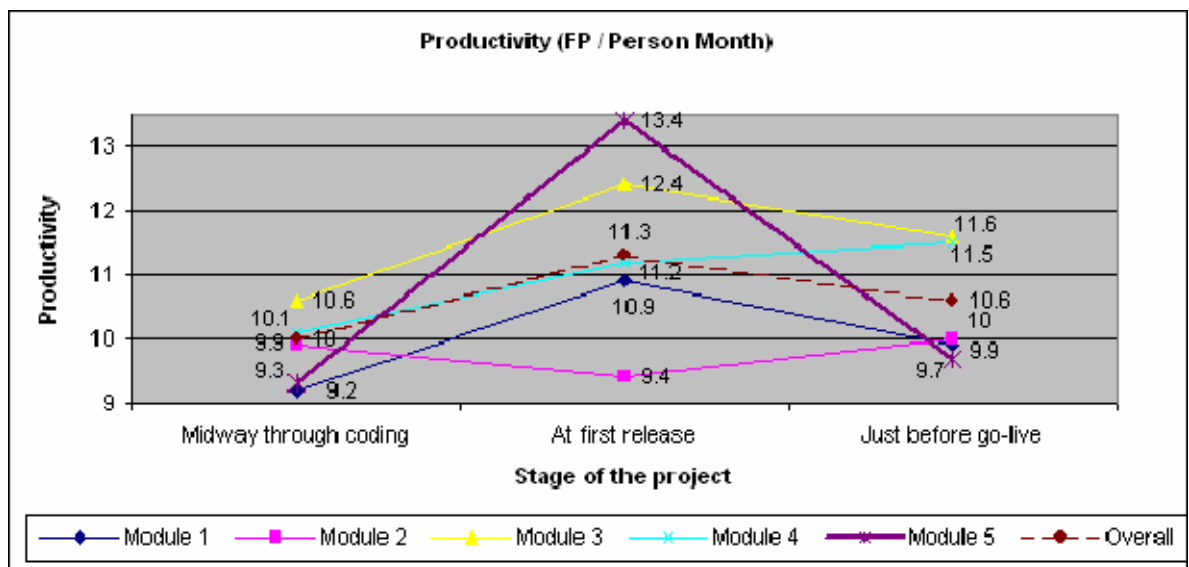


Figure 17: Productivity Tracking

The assumption that development productivity will rise close to the level taken for initial estimates was not realized. This fact, along with a delayed resource ramp-up, resulted in an overall delay in meeting milestones. The customer judged one of the reasons for this result as weakness in project management within Infotech. However, as total transparency had been maintained and the customer’s intent was to have a successful experience with offshore development, the customer agreed to revise schedules taking into account this delay.

While estimating effort on this project, data from past projects was used and an 11% variation on a mean of 101% of the base estimates was applied. The effort calculated was reduced by 20% to account for a J2EE framework provided by the customer.

¹⁴ While computing productivity, a development FP count was used and not an enhancement FP count even though the CRs came in at varying points of time during the lifecycle of the project.

Schedule Tracking

During the tender process, the customer requested an unrealistic 6-month development schedule. In order to respond to this request, since there was a very simplistic mechanism in Infotech's organizational process for arriving at the schedule, a heuristic formula was used¹⁵. A schedule estimate of 12.5 months was arrived at using the productivity and size baseline at the start. Based on current data it is expected that the actual duration will be 28 months, which is closer to the heuristically arrived at schedule estimate of 24 months.

Defect Tracking

Based on past experience of its own high costs if defect levels were high, the customer insisted on certain limits for the number of errors. These limits came from the need for acceptance on a certain scope to be accomplished within two cycles of testing. Four levels of errors were specified in the contract and a maximum number of errors allowed per level for each version and module were also specified.¹⁶ At the start of the project, Infotech's quality capability of post-release defects (based on data recalibrated for FP) was pegged at 1 bug per 20 FP (for all severities).¹⁷ Infotech's organizational data did not provide information on post-release defects based on severity. The error-level target requested was beyond the measured organizational capabilities and Infotech had to look at means to meet these limits and avoid penalty.

As the project progressed, detailed error levels were measured. Errors were logged at different severities and for each use case, and further classified based on the solution expected. As testing progressed, these metrics were continuously updated.¹⁸ The customer commented that this is the first time that they were able to get a measure of errors at various levels from the software developer along with software delivery. These measurements helped project management teams to make decisions pertaining to releases.

Attempts were also made to make further improvements in the testing process both within Infotech and in the customer's environment, including the following:

1. Use of additional tools was considered to see if the customer could reduce the cost of acceptance efforts. The customer paid for pilot projects involving the use of Mercury Quick Test Professional (QTP) and the development of test scripts that were based on system test cases. After some development and much financial analysis of the efforts

¹⁵ This formula is $\text{schedule_in_months} = 3.0 * (\text{man_months})^{1/3}$. The formula is from RAPID DEVELOPMENT by Steve McConnell.

¹⁶ These defect levels were frozen at the estimated size of the application when quoted. As the real size became known through subsequent re-estimation and change requests, the baseline of acceptable defect levels was redefined in a manner that was directly proportional to the size.

¹⁷ The error level requested was 1 bug per 30 FP. These bugs referred to only severity 2 bugs in the first cycle of testing and severity 3 defects in the second cycle of testing. Zero bugs of higher levels of severity and an unspecified number of bugs of lower levels of severity were possible.

¹⁸ Based on current data, post-release defect density is projected to be around 1 defect per 13 FP for all severities.

required, it was adjudged that this approach could not be justified financially. Instead, the budget was used in the creation of a regression test suite for the project, which could be enhanced and used during software maintenance phase.

2. Extensive use of JUnit was made in this project as a unit testing strategy. Experienced developers would know about the amount of repetitive programming required to use JUnit. With a view to optimize these efforts, Infotech spent some effort on evaluating a third-party tool called Agitator. It was finally concluded that though the tool was good, benefits would not be seen in the project environment. Further, proprietary and Oracle frameworks used in the project were not supported by Agitator

Conclusion

CMMI processes helped Infotech right from the beginning. Organizational variances were included in the commercial contract. As the project progressed, continuous measurement and transparency helped in the customer accepting variations that occurred. The project experience has added a lot of valuable data to Infotech's repository. A final validation of this data would be done during the project closure. The experience has reaffirmed our faith in quality processes and their direct use for commercial decisions.

Important Topics for the Research Community to Address

1. Is hidden functionality a common occurrence and should a parameter of measure be put for this in the estimation process?
2. What should be the model of quantifying additional effort for commitments beyond measured organizational capabilities?
3. What is the range of improvement seen in productivity in projects? Can there be a model to estimate the benefit of the learning curve?
4. How can organizations best calculate the saving of effort from a framework (re-usable design or code) supplied from outside the project?

Biographies

Rajeev Lal

Infotech Enterprises Limited

rajeevl@infotechsw.com

Rajeev Lal is the president of Strategic & New Business Initiatives at Infotech Enterprises Limited, Hyderabad, India. Founded in 1991, Infotech (www.infotechsw.com) is a leading provider of engineering, software, and geospatial services to the manufacturing, utilities, telecom, transportation, and government verticals worldwide. Infotech currently employs over 3,000 resources across 17 global locations. A public company, Infotech has equity participation by United Technologies Corporation and Tele Atlas.

Lal has been in information services for over 33 years and worked at a number of renowned companies before joining Infotech. In Infotech, he led the initiative for successful

certification of the software development business to ISO 9001 in 1999, SW-CMM Level 5 in 2002, and CMMI Level 5 in 2003. He is chairman of the company's Security Forum, and the company is certified to BS 7799 standard. Infotech has participated in benchmarking for software measurements with peer companies in India.

Lal is a Fellow of The Institution of Engineers (India). He was Chairman of the Computer Society of India's Lucknow Chapter. He has been a member of the American Management Association. He also has been a visiting faculty at institutes and writes occasionally.

K. Niranjan Kumar

Knowledge Management and .NET

niranjank@infotechsw.com

K. Niranjan Kumar has over 15 years of work experience in the IT industry. He has experience spanning varied roles from hands-on development to business analysis and program and project management in various technologies ranging from mainframe applications to systems software development to Web applications. He has worked in both onsite and offshore environments.

Over the last six years, Kumar has spearheaded the development of Web application software at InfoTech. He has been involved in many business requirements analysis activities and employs formal requirements documentation and management techniques that have helped him manage scope, schedule, and cost well in past projects. He has also been a pioneer in Infotech in using formal OOAD processes for design and development and has consistently explored new project management and software engineering techniques.

In his current role, he is the practice manager for two technology practices, viz. knowledge management and .NET. His interests have been in the field of knowledge management (KM) and the ability to use KM techniques, such as search technologies, to solve pressing industrial problems. He has already architected solutions in KM for two Fortune 500 organizations. He has set up projects for students in engineering so that both education and industrial sectors can benefit from the synergy.

His current task is to create a competence area for the retail industry vertical, focusing on supply chain management.

Niranjan holds a bachelor's degree in electronics and telecommunication engineering from the National Institute of Technology, Tiruchirapally, India.

3.5 Empowered Engineers are key players in process improvements

Author

Katsutoshi Shintani

Abstract

Process improvement will be continuously pursued when development engineers are empowered with ownership of roles and associated processes instead of relying on an outside SEPG. At least one person within a development group needs to be identified and assigned to act as a catalyst for process definition, assessment, and improvement. This person must help development engineers recognize their own roles and associated processes. The definition of role is vague in the current ISO/IEC 12207 process model, but the concept of role can be easily understood by the development engineers associated with activities. If process can be defined as a set of activities, then it is easily associated with roles. It also may be beneficial to have each development engineer come up with improvement items.

Paper

Whether for small or large companies, process improvement needs to be initiated by those who own software development processes and are in essence development engineers. Small organizations do not usually have an independent SEPG that supports the development group; therefore, each development group has to understand the full spectrum of the processes under which they work. Each person in a development group needs to be assigned to cover a particular process (or processes) related to his or her assigned activities.

The following points describe how to make sure the entire process is covered by the development group:

- The first approach is to make sure each member in the development group has a thorough idea what role each individual is playing and how these roles link to the pre- and post-processes. Here, a *role* represents what is assigned and may not be mapped one-to-one with a particular process.
- The second approach is to assign an individual to document the processes with roles and to make sure the development members cover the entire process from end to end. Here, the key point is to associate roles and processes. A two-dimensional chart would make the roles and processes more visible. Often development engineers can easily associate their activities or responsible areas with roles rather than processes.
- The third approach is to find any missing links in the first two approaches and notify the development members in a group meeting in which they will adjust the roles each individual is assigned with possible new processes.

- The fourth approach is to identify any items to be improved for each role with associated processes. There may be a need to discuss the adjacent roles to be determined by the associated processes or the adjacent processes to be determined by the associated roles.
- The fifth approach is to identify improvement items that each individual can own for each assigned role or process and define metric items. Identifying metric items may not be so difficult, but to measure and collect them may require some training or support a group leader. Then, each development engineer must record measures. By doing this, metrics can be associated with each role that is owned by each engineer and with the processes associated with roles.

As mentioned in the second approach, even in a small team, there needs to be a person who coordinates the above approaches and has a responsibility to do the following:

- encourage process owners to update process descriptions based on those identified during the development efforts
- to let all in the development be aware of the entire process spectrum and changes in any processes. To do this, a common library needs to be set up, and the person assigned to coordinate these efforts must initiate a team meeting to share identified improvement items with the measurements. It is advisable to set up the team objectives for the improvement so that at the team meeting, all of the team members will be able to relate the improvement efforts to the objectives. It may not be easy to set up the team objectives from the beginning as the scope of possible improvement items will be learned after the project starts. However, once a team meeting starts and reports by team members will be shared, it will become evident that the team needs to have common objectives; this is exactly the point when process improvements become a way of life for the team.

Biography

Katsutoshi Shintani

Software Engineering Center,

Information-technology Promotion Agency, Japan

k-shinta@ipa.go.jp

Katsu Shintani joined IBM Japan in 1969 as a customer education instructor; he was a systems engineer for 10 years, and a software development engineer with various management positions for another 10 years. In his last 10 years at IBM, Shintani was part of a software development group working with the corporate level reengineering representing software development groups at Yamato Laboratory of IBM Japan. After leaving IBM in 2004, he became a researcher at a newly founded Software Engineering Center that is an affiliate of the Ministry of Economy, Trade and Industry. His current interests are in the areas of process assessment and improvement and software engineering dissemination. Shintani has a bachelor's of engineering from Kyoto University.

3.6 Accelerated Process Improvements for Small Settings

Authors

Dr. Anil Revankar, Raghavendra Mithare, and Venkata Muralidhar Nallagonda

Abstract

Small organizations, and especially organizations with small IT settings, have their own issues and advantages associated with process improvements. Process improvements in small organizations typically have constraints because these organizations cannot attract the best-of-the-breed process experts, have limited resource availability, take a long time to define processes, and must deal with the comparatively high process improvements costs. However, small organizations also have a strong focus and commitment of top management and can easily spread process knowledge because the number of practitioners working in these organizations is comparatively small.

This paper discusses the need, methodology, and experiences with the implementation of Rapid-Q, which is adopted to accelerate the process improvement initiatives in small organizations. Quality frameworks such as ISO 9000, CMMI, and BS 7799 preach what is best and not how to achieve results. The framework incorporates the “common-how” part and accelerates the process improvements. This framework has been successfully deployed by inducting the processes of Rapid-Q: a customizable set of processes that can be directly deployed into organizations with minimal effort. The usage of this framework has reduced the cycle time for process induction to 1-2 months (compared to 6-9 months for the conventional in-house process development and deployment approach).

Need For Accelerated Process Improvements

Large organizations undertake process improvement initiatives, due to the magnitude of impact it has, through an elaborate process of model-based gap analysis, definition; pilot and organization-wide deployment, and sustenance of processes. This is accomplished by forming a processes taskforce, piloting the new processes on projects, and fine tuning the process elements based on the lessons learned in the pilot phase. The fine tuned processes are then implemented across the organization.

The traditional approach, though it has good buy-in from practitioners, lacks the speed in implementation since it involves a good share of bandwidth from the practitioners in evolving and piloting the process. The cycle time for a new process model implementation at the organization level has been in the range of 8 to 14 months. In view of this, we looked at alternate approaches to reduce the cycle time without losing the essence of process components. This is akin to the commercially off the shelf (COTS) concept of software, extended to process improvement projects.

Comparative Analysis of Organizational Pain Areas

Any process improvement initiative needs to consider the organizational nuances for effective deployment of quality and process frameworks. Typical pain areas associated with process improvement in IT projects in small and large businesses are shown in the following table.

Table 12: Large Company vs. Small Company Pain Areas

Large Company	Small Company
Typically, large organizations have a big budget for process improvements.	Small organizations have budgetary constraints on process improvement; <i>hence, look for low cost solutions without losing process essence.</i>
Typically, there are multiple filters between the customer and the development staff for requirements.	Typically, there are few filters between the customer and the development staff; this makes for a more challenging environment for expectations management and <i>requires agile processes.</i>
Typically, blends multiple quality frameworks to meet its diverse process improvement needs	The focus is always on immediate problems and issues; hence, the quality system <i>should be flexible to address point solutions.</i>
If a team member is absent, there is often someone to take his or her place for a meeting or other process improvement (PI) related event.	If a team member is absent, it is likely that there will be no substitute and, in fact, the event may have to be canceled, which <i>creates a need to ease the stress of traditional quality system development structures.</i>
A one day meeting with the project team for process gap analysis isn't usually too difficult to schedule (although more than one often is).	A one day meeting with consultants is often quite challenging to schedule, which leads to the <i>need to develop rapid quality system development.</i>
Getting budget approval for a PI support tool, or even books, often goes through a complex administrative network	If the money is available (usually determined with one interaction), getting approval/making purchases for PI support materials can be quite trivial. This <i>enables rapid process improvements.</i>

Projects have independent resources to manage different roles such as project manager, project coordinator, architects, designers.	Individuals subsume multiple IT roles in the projects and the <i>quality system requires flexibility to subsume multiple roles.</i>
Deployment of PI proposals need several rounds of negotiation with stakeholders through the pilot and refinement process, and the deployment will have a long timeline	Deployment is instantaneous once the stakeholders buy-in is obtained; the <i>ability to deliver as agreed is the essence.</i>
By sheer size of the organization, ROI of process improvements may require a long time to realize	In small organizations, the <i>impact is felt immediately, so expect quick ROI.</i>
Typically, large organizations may have buffers associated with scheduling timelines on process improvements.	Small organizations do not have the luxury of a relaxed timeline for implementation, <i>so they require quick solutions.</i>

Rapid-Q Process Framework

Based on the analysis of the above pain areas, small organizations need a different approach for process improvement compared to large organizations. To cater to these needs, Wipro has developed its own product *Rapid-Q*, which is depicted in Figure 18. Rapid-Q is a customizable, off-the-shelf quality system that incorporates years of experience and best practices. It is flexible, modular, and cuts down the implementation lead time. It has a library of Processes, Templates, Procedures, and Guidelines. The deployment of Rapid-Q delivers significant savings on schedule and cost for implementation of quality processes.



Figure 18: Rapid-Q: Off The Shelf Quality System

At the core of Rapid-Q process framework are Policies, Procedures, Templates, Checklist and Lifecycle models. The processes are primarily designed to cater to global requirements and different organizations needs. Before inducting such processes, one needs to fine tune the Rapid-Q process components to meet the organization's internal requirements. This paper focuses on how the Rapid-Q framework is used for accelerating the induction of processes into small organizations. The Rapid-Q is similar to any COTS products that are currently available in the IT world (e.g., RUP, SAP, and MRP). This section attempts to look at the parameters and components examined while evolving the framework.

Compliance to the Quality Standards/Frameworks

Most of the organizations evolve their quality processes based on quality frameworks such as ISO 9000, SW-CMM, CMMI, BSI - BS 7799, and IEEE Software Engineering Life Cycle Standards. The Rapid-Q has process assets that meet the requirements of these quality models. Rapid-Q supports continuous process improvement and helps address specific pain areas of customers. In addition, the tool provides a repository of Six Sigma tools and techniques.

Structured Methodology

Considering the user base of process models, it is important that the application of Rapid-Q processes are consistent. This can be achieved only if the implementation follows a structured methodology, like ETVX, with clearly defined roles and approval levels. The structured methodology also provides a uniform understanding of process through templates, guidelines, checklists, etc. The structured approach enables faster deployment of Rapid-Q processes. A high level process view of Rapid-Q is shown in Figure 19.



Figure 19: Rapid-Q Process View

Seamless Integration

One of the major aspects of a process improvement framework is consistency and seamless integration. When a new process is introduced, it should blend seamlessly with the existing practices and best practices derived from SW-CMM, CMMI, ISO 9001, BSI-BS 7799, etc. Blending happens at different levels: in terminology, the work flow, reporting system, delivery mechanisms on training, hand holding, etc. This blending requires organizations to engineer new processes and also customize existing processes.

Visual Realization

Rapid-Q has benchmarked practices from the best of breed models mentioned above and addresses the common *how* part with clearly defined roles and approval authority that are followed across the industry. It is easy to for organization to visualize processes that align business goals without spending much time trying to understand various models and how to use them.

Flexibility

The Rapid-Q framework allows the flexibility to address point solutions that cater to client-specific needs, such as configuration management, testing, release management, and portfolio management, without implementing the entire processes of a quality model. This gives organizations the flexibility to incorporate only selected maturity levels of CMMI or chosen quality frameworks, models, or standards. Customizable processes help organizations align process goals with immediate business needs.

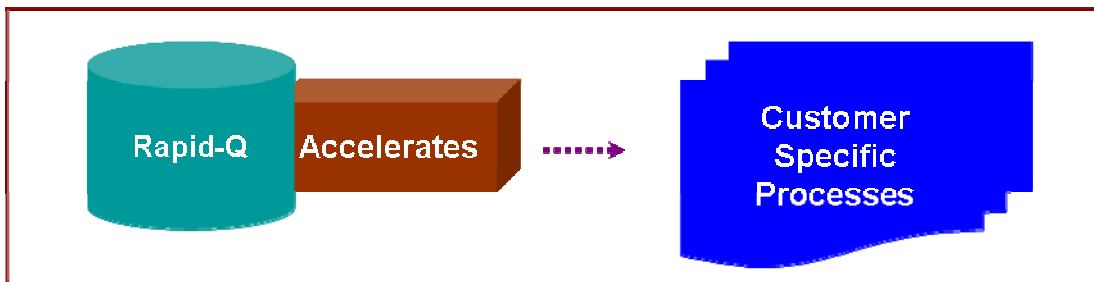


Figure 20: Rapid-Q Accelerates Process Implementation

Case Study

Initial Scenario

The client started initiatives for improving their current software process capabilities by establishing a methodology consisting of metrics, artifacts, and processes for project execution. This methodology had few templates and checklists but was lacking in procedures and guidelines. Additionally, it had no processes defined for different project scenarios, such as sustenance, small projects, and global delivery model. Projects used only a few parts of the methodology, resulting in discrepant project management and inconsistent SDLC practices across different projects. The effectiveness of project execution depended on the people and their past experiences. Practitioners focused on quality during the testing phase and not any earlier. There were no project levels measurements to track project performance through different phases of the lifecycle. A lack of a senior management mandate for process compliance made quality negotiable.

Implementation

The consultants conducted a comprehensive gap analysis to understand existing the business model, process, and practices and to establish a common process framework. The team identified opportunities for improvement and designed a road map for process framework development. It was critical to define processes quickly, and the consultants customized Rapid-Q. The Rapid-Q process framework contained well-defined process structure using the ETVX Model (Entry Task, Verification and Validation) for project management, engineering, support, and organization activities. This aided in faster process customization. The customization of Rapid-Q was completed in 12 weeks, compare to the 24-26 weeks that was planned. Process customization was mostly related to project management and engineering process areas. In order to develop a process framework that caters to various project types and scenarios, 24 processes were developed. These covered some of the important process areas like business requirements, software requirement specification, project planning for small and enhancement projects, onsite and offshore engagement, vendor evaluation and selection, build and release management, and measurement and configuration management. The project teams were trained on the new processes.

Business Benefits

By defining a standardized set of processes, time for process improvement was reduced by more than 50% and the work was less complex because artifacts were reused. This ensured a reduced interdependency between project teams for the projects developed jointly across two geographic locations. The teams used consistent requirements, architecture, and design process.

Lessons Learned and Conclusions

The following lessons were learned from the experiences of the Wipro Technologies consultants for the small organizations process improvement initiatives.

1. Small organizations need help implementing quality systems at an accelerated pace for process improvement results to be realized. A good way to start this is to facilitate the small organizations with the common part of *how* to achieve process improvement, which is not addressed by quality models.
2. Our experience shows that more research and facilities need to be made available to small organizations, especially in the areas of metrics. These metrics and facilities include small industry-specific categorized data for various types and technologies, cost effective training, and the expertise needed for interpreting and internalizing the models.
3. As small organizations start making progress in process improvements, their success stories set the tone for other small organizations to adopt quality models and internalize the best practices. The adoption of process improvement by small organizations can be expected to rise with availability of tools to meet their requirements and environmental support by the SEI, other industry bodies, and consulting firms.

Bibliography

- [Garcia 04]** Garcia, Suzanne; Cepeda, Sandra; Miluk, Gene; & Staley, Mary Jo. *CMMI in Small Settings Toolkit Repository from AMRDEC SED Pilot Sites; DRAFT 14*. <http://www.sei.cmu.edu/ttp/publications/toolkit/> (2004).
- [SEI 02]** CMMI Product Team. CMMI for Systems Engineering/Software Engineering/Integrated Product and Process Development, Version 1.1, Staged Representation (CMMI-SE/SW/IPPD, V1.1, Staged) (CMU/SEI-2002-TR-004, ADA339221). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2002. <http://www.sei.cmu.edu/publications/documents/02.reports/02tr004.html>.
- [Wipro 05a]** Wipro Technologies Ltd. *Rapid-Q–Quality System Tool*. http://www.wipro.com/consulting/quality/rapid_q.php (2005).
- [Wipro 05b]** Wipro Technologies Ltd. *SPEED–Quality Implementation Methodology*, White Paper. <http://www.wipro.com/index.htm> (2005).

Biographies

Anil Revankar

Wipro Technologies

Anil.revankar@wipro.com

Anil Revankar is based in Detroit, MI and leads a team of quality and process consultants that enables Fortune 100 companies to achieve higher process maturity. He was the Head of Tools Group at Wipro Technologies. He has more than 20 years of experience spanning engagements in academics and the corporate world. He was involved in CMM and CMMI Level 5 assessment as a core Assessment Team Member and has contributed significantly in establishing several high maturity processes. He was responsible for development and deployment of a workflow-based tool–iPAT (Integrated Process Automation Tool) across 20,000 software developers. He has trained several hundred project managers on various software engineering topics. Before joining Wipro, he was the associate professor of Systems at Post Graduate Management School. His core competencies include high maturity process definition and deployment, productivity and product quality enhancements through tools, process modeling, process automation, Six Sigma techniques, expert systems for decision making, risk management, metrics and top management dashboards. His qualifications include a Wipro certified Six Sigma Black Belt, graduation in mechanical engineering, post graduation in systems engineering and management, and a PhD in expert systems and risk assessment.

Raghavendra Mithare**Wipro Technologies****Raghavendra.mithare@wipro.com**

Raghavendra Mithare is at present Senior Process Consultant at Wipro technologies. He has extensive experience in process definition and implementation. He was instrumental in implementing the RUP within Wipro and developing supplementary processes to meet the requirements of CMMI level 5. He has handled various consulting engagements in Process Improvements, Software Metrics and Process Automation. Currently he is working on Business Process and Engineering Processes Integration and Automation, Member of Rapid-Q development team. Helped in developing processes to meet requirements of CMMI, ISO 9000, ITIL and CoBIT. Qualifications: B.E., M.Tech., PMP (PMI USA), Certified RUP Professional (IBM Rational).

Venkata Muralidhar Nallagonda**Wipro Technologies****Venkata.Muralidhar@wipro.com**

Venkata Muralidhar Nallagonda is at present senior process consultant at Wipro Technologies, providing consultancy to Fortune 100 companies. He has about 13 years of experience in the IT arena. In his current role, he is responsible for defining and deploying new processes into organizations' Quality Management Systems. He played a critical role in designing and deploying Quality Management Systems to various organizations involving complex processes, including multimedia processes. He participated in assessments as a core Assessment Team Member and Field Area Representative for CMM and CMMI. He was also instrumental in implementing ISO 9000, BS7799, ITIL, CoBIT, and Six Sigma in large software organizations. He extensively worked on various quality functions, including Corporate SEPG. Qualifications: Graduate Engineer (Electronics & Communication Engineering), Post Graduate Program (ME) in Quality Management, CSQA (QAI, USA), and PMP (PMI, USA).

4 Process Improvement Tools and Techniques

4.1 Process Improvement as a Real Option to Extract Value from Project Failure in Context of Small Business

Authors

Mohan Sharma and Rajneesh Sharma

Abstract

This paper analyzes the importance of process improvement for small and medium businesses to extract value from failed projects. Matured processes employed during development of a project not only increase the probability of success, but can also extract additional value from it if the project fails. The ability to exploit Real Options in case of failure is possible in an environment of matured processes and makes a compelling case for process improvement in small and medium size organizations.

IT project failure is commonplace due to a variety of reasons. Strong software engineering processes increase the probability of project success but may not always lead to a successful project if, for example, a business change makes the product useless. But strong processes reduce the financial risk for small and medium size organizations by providing visibility and flexibility into the software product and the project making that product. This flexibility is necessary to change the product features and change the project direction. The visibility, flexibility, and control emanating from strong processes provide a handle on the project and product of the project; this handle can be effectively used to salvage value out of the failed project, which may not be otherwise possible if the processes are weak.

Real Options

The Real Options approach identifies strategic choices and tries to determine the value of the choices using option valuation theory. Strategic choices faced by a firm are Real Options. To choose the timing of a project, spend money on expansion, and abandon a project are all examples of Real Options [Mun 02]. These options are valuable and their valuation should be incorporated in the analysis of any project. Standard capital budgeting utilizing Discounted

Cash Flow analysis may underestimate the value of the project to the firm if the Real Options embedded in the project are not accounted properly [Myers 87]. Management by the virtue of creating, identifying, and optimally exercising these options can increase the value of the firms. The greater the flexibility afforded to a project, the higher the value of these options. The value of these options also increases with the increase in the uncertainty of the project.

The nature of small businesses propels the argument that project risk contribution to the overall risk of the company is higher for small business than for a big business. Given a project of the same size, the return to a small company will have a higher value compared to the return to a big company. The risk value of a small company's project makes the process deal with risk differently. This paper identifies the processes where small companies should put extra resources in comparison to big companies. We also show that these resources and changes in the process should bring in an increase in the expected value of the project to the company.

For a small business, the affect any particular project may have on the overall profitability is proportionally larger than the same project would have on large firm. If a similar project fails, it may have little impact on the expected profitability of a big firm due to diversification benefits and size effects. However, for a small firm, the failure of the same project may significantly affect the profitability of the firm. Though suffering from this drawback, the relative strength of a small business is its higher flexibility. The flexibility incorporates faster starts and stops in a project and keeps a dynamic environment with constant feedback between all parties involved. This flexibility increases the relative value of Real Options for a small firm.

IT Project Failures and Real Options

IT projects are notorious for their failure rate. In a 2003 article, Julia King reports, "At companies that aren't among the top 25% of technology users, three out of 10 IT projects fail on average" [King 03]. Per Standish Group CHAOS Report, "average project cost of a development project for a small company in US is \$434,000. A great many of these projects will fail" [Standish Group 94]. Scope creep, budget overruns, business restructuring, schedule slippage, and the inability of the final product to be able to stand up to expectation of quality, performance, stability, scalability, etc. are the common malaises which afflict IT projects in varying degrees. The need to contain these risks, and sail through them in case they occur, makes a strong business case for process improvement in small companies.

Consider two firms: one large and one small developing projects. Assume that the large firm has \$10 and can take 10 projects costing \$1 per project at time. The small firm has \$1 and can take only take 1 project at time. Let the payoff of the project if successful be \$2. Let the probability of success be 80% and independent over projects. Thus the expected profit from each project is $.8*2-1=\$0.60$. Given this situation, the probability of breaking even (having a payoff of at least the original investment) is 99.36% for the large firm and 80% for small firm. The risk to the small firm is excessively high. Assume there is process improvement

that when implemented would change the payoff of the project in case of a failure to \$1. For example, imagine the firm spending extra to make the project flexible enough so that good portion of the project can be salvaged in case of a failure and used in another project. If such an improvement were free, both the small and big company would implement such a process. If the process improvement were to cost \$0.30, then the improvement may not be suitable for the large firm, but it would still be suitable for the small firm. The process improvement is a Real Option that is identified and created at extra costs by the small firm only.

Process Improvement and the Small Enterprise

With the growing acceptance of Real Options analysis (ROA) as a modern approach to investment analysis several information systems (IS) researchers have tried to apply ROA to IT investment decision-making [Benaroch 02]. Early proposals were put forth by Dos Santos [Dos Santos 91] and Kambil et al. [Kambil 93].

Processes Maturity: A Real Option for Success

Software process maturity in the organizations give high visibility into the software development life cycle. Project managers in particular and top management in general will have a much better handle on the project in an environment of matured software process than in the absence of it. It does not need emphasis that the project has a better chance for success when there is a better handle on the budget, scope, requirements, quality, etc. When the product of the project, intended for a specific single customer or toward a market segment, is right on the target business need, there is greater receptiveness for that product from the targeted customers. Matured processes increase this probability.

Processes Maturity: A Real Option in Failure

While the increased success rate of the IT project in itself is a Real Option, we are aware of the fact that matured software processes only increase the probability of success; they *do not* guarantee success. We live in dynamic business environments where businesses change, restructuring forces certain projects under development to be shelved or changed, business relationships break, and markets expand, contract and shift forcing us to think of many more products, applications, and IT Solutions than was previously envisaged. These developments carry potential to affect the success of the software project under development.

Even when the business case of a project is strong, it is still possible (though less likely under matured process environment) that certain requirements may be missed, particularly some non-functional requirements. The application may be unique, not clearly understood by anyone, and involve new technology. The expertise required for the unique technology skills may be scarce. IT projects, by their very nature, have large number of technical and business risks involved and occasionally lead to project failure. Ironically, for a small enterprise, while a project can have significant impact on the bottom line, the window of failure is relatively shorter when compared to a large enterprise. The window for scope creep, budget overruns, schedule slippage, etc. is generally small for a smaller company compared to its bigger

counterparts. The purse of a smaller organization is small, and when it is recognized that a project starts going astray, management is unwilling to throw any more good money after the bad and votes for discontinuing such a project.

Amidst project failure or project crisis scenarios, the small organization is better prepared to use Real Options *if* it has high maturity processes. The ability of a small organization to execute Real Options has a direct relation with the process maturity of the organization

The business side of a company may identify one or more potential Real Options for a crisis project. This paper does not intend to focus on whether Real Options exist or not for a given project or how business will identify Real Options in case of failure. For the sake of focus, the idea presented here is that the success of a potential Real Option will depend to a great extent on the software process maturity of the organization.

The very confidence of invoking a Real Option will come out of the organizational process maturity. Process maturity brings greater accuracy to estimates, project plans, schedules, budgets, and quality targets. The confidence arising from this accuracy sets the ground for invoking Real Options.

Having a Real Option and Executing a Real Option

Given that Real Options exist for a project (or the product of the project) does not automatically mean that they can be executed easily. For example, the ability to change the usage model of the project mid-stream is one Real Option that requires very high visibility into the requirements of the product and the ability of the project team to make changes to those requirements with minimal cost within a short time frame. It is also necessary that a Real Option should be measurable on all parameters like business value, ROI, budget, schedule, quality, etc. If the schedule, budget, and quality of the product cannot be defined, executed, and measured with a high degree of confidence, then this Real Option is not really a good option to pursue. The ability to measure these things and achieve it per plan is tightly linked to the organizational software process maturity and is important for executing the chosen Real Option successfully.

Process Maturity Helps Bring Changes Necessary for Implementing Real Options

Let's take the case of the vital Requirements process area. Real Options are about change, and change starts with identifying the requirements according to the new business model that the project will try to address. Field states that "projects fail too often because the project scope was not fully appreciated and/or user needs not fully understood" [Field 97].

Below are some points which endorse the importance of Requirements process area in implementing Real Options. Similar arguments can be presented for other process areas.

- To change the usage model of the product, by contraction, expansion or to switch to a different path, there needs to be a strong visibility into the requirements so that it can be easily decided what to keep, what to shed, and what to change.
- A review of SRS by the business representatives and business analysts to make changes in functional and non-functional requirements at the granular level goes a long way in the success of the product. For change management to work effectively, it is necessary to have bidirectional requirement traceability down the stream, which comes only in an environment of matured processes.
- Where quality factors are critical to success, requirement process maturity plays an important role. It has been observed that since non-functional requirements require a greater degree of rigor and requirement process maturity. This is because quality is the biggest casualty in the product in the absence of requirement process maturity as many non-functional requirements are missed in an environment of immature requirement process.
- Efficient requirement engineering helps in an efficient modular-based application development. It gives leverage to development team that allows them to make quick changes by plugging in new modules, removing redundant ones, and modify the existing ones. Such an approach is also useful in alleviating, to a great extent, integration challenges of complex projects.
- For estimation of the effort (and budget) corresponding to the changes necessitated by invoking the Real Option, matured requirement process is inevitable. The same argument holds for the development of a realistic project plan and schedule corresponding to the change.

The above arguments focused only on the Requirement process area as an example to emphasize the importance of software processes in the ability to execute real options. Similar arguments exist for other process areas if Real Options are to be used.

A software processes are like a system, where each process is a subsystem in the whole system. As can be seen from CMMI documents, all the software processes are connected with many other processes [SEI 02]. Software processes are interconnected like an organic whole just like the software components they help to make. So the Requirements process area mentioned above, though very crucial in realizing the Real Option, will need to have other supporting matured software practices.

Overall maturity that will come from the organizational set of processes increases the confidence level in

- project plans, schedules, budget and resource planning, risk assessment and management, testing, release, and constant visibility into the product or project. All of these are important *before* executing the Real Option and *while* executing the Real Option.

While following a CMMI-based approach to process improvement is certainly very useful because of its comprehensiveness, not all small organizations may need the rigor of all the

process areas mentioned in the CMMI staged or continuous representations. At the end, it is up to the organization to decide if it wants to implement CMMI by taking into account various factors prevailing in that organization. Kulpa and Johnson give a set of arguments that can help an organization in this decision making exercise. [Kulpa 03]

Tracking the Real Option: Executing and Tracking the Change

Constant visibility into the project development or project change is necessary to execute and track the Real Option. Matured processes bring *sufficient* and *trustworthy* information necessary for this exercise. In the absence of mature practices, the project management data is incomplete and occasionally not very accurate. In a low process maturity environment, the project manager (or a project office) will be making reports based on data that is low in reliability and validity. This shows the importance of metrics which are available, reliable, and valid only in an environment of matured processes.

For example, in the absence of matured software practices, the measurement of efforts and budget will be weak and, consequently, the confidence level in the schedule outcome will be low (e.g., the estimates, one of the weak areas in IT, is even weaker in low process maturity organizations and is occasionally quite off the mark). Project plans and schedules based on weak estimations will be weak. Hence strong processes play a key role in *invoking* a Real Option and then *executing* a Real Option with confidence.

Strengths

- Strength of using the Real Options approach lies in the fact that a project failure has significant repercussions for a small business and could sound the death knell for a small organization if the failed project happens to be one of the key component of the overall business strategy of that organization. Using Real Options via matured software processes in an organization can extract value from the failed project and minimize the risk of project failure.
- A small organization is by nature more flexible compared to a bigger organization and thus better placed to exploit the potential of Real Options.
- Because a strong process oriented software development environment creates a strong platform for swift changes to the project, a quick change of path with creative business strategy can reap dividends in a short period of time.

Weaknesses

- Usage of Real Options has to gain more ground where IT projects are involved. The actual calculation of the value generated by a Real Option is complex. Some of the variables needed as inputs cannot always be measured with satisfactorily high accuracy.

A Research Opportunity

Process improvement is not without effort and cost. Small businesses have smaller wallets and need a sound business case for process improvement before it is bought by management. The research community can give a high focus on the business value of software processes in small settings instead of just seeing it from a general management, project, or technology perspective. Real options for IT projects looks at the value of the process improvement from a business perspective. In this case, it emphasizes that matured processes play a role not only in the success but also in the failure of a project as seen in the following:

- Matured software processes *increase* the probability of success of a project.
- Matured software processes can *minimize* the damage from a failed software project.

Because the usage of Real Options involving IT projects is still an under-explored territory and there is little empirical data or case studies to gauge the extent of benefits, research in this direction will help better understand the utility of matured software processes not only in success but also in failure.

References

- [Benaroch 02]** Benaroch, Michel. "Managing Information Technology Investment Risk: A Real Options Perspective" *Journal of Management Information Systems* 19, 2 (2002).
- [Dos Santos 91]** Dos Santos Brian L. "Justifying Investment in New Information Technologies," *Journal of Management Information Systems* 7, 4 (1991): 71-89.
- [Field 97]** Field, Tom. "When bad things happen to good projects." *CIO Magazine* 11, 2 (Oct. 15, 1997): 54-60.
- [Kambil 93]** Kambil, Ajit; Henderson, John; & Mohsenzadeh, Hossein. "Strategic Management of Information Technology: An Options Perspective," 161-178. *Strategic Information Technology Management: Perspectives on Organizational Growth and Competitive Advantage*. Middletown, PA: Idea Group Publishing, 1993.
- [King 03]** King, Julia. "Survey shows common IT woes." *Computerworld* (June 23, 2003).
<http://www.computerworld.com/managementtopics/management/story/0,10801,82404,00.html>
- [Kulpa 03]** Kulpa, Margaret & Johnson, Kent. Ch. 10. "Is CMMI Right for Small Organizations?" 119-129. *Interpreting the CMMI: A Process Improvement Approach*. Boca Raton, FL: Auerbach Publications, 2003.

- [Mun 02]** Mun, Johnathan. *Real Options Analysis: Tools and Techniques for Valuing Strategic Investments and Decisions*. Hoboken, NJ: John Wiley & Sons, 2002.
- [Myers 87]** Myers, Stewart C. "Finance Theory and Financial Strategy." *Midland Corporate Finance Journal* 5, 5 (1987).
- [SEI 02]** CMMI Product Team. CMMI for Systems Engineering/Software Engineering/Integrated Product and Process Development, Version 1.1, Staged Representation (CMMI-SE/SW/IPPD/SS, V1.1, Staged) (CMU/SEI-2002-TR-012, ADA339907). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2002.
<http://www.sei.cmu.edu/publications/documents/02.reports/02tr012.html>
- [Standish Group 94]** The Standish Group. *The CHAOS Report*.
http://www.standishgroup.com/sample_research/chaos_1994_1.php (1994).

Biographies

Mohan Sharma
PMP, CFPS, CSQE, CQA
mohansharma@ups.com

Mohan Sharma has 11 years of experience in informational technology in the areas of software processes, quality, project management and PMO. He has extensive experience in implementing software process improvement initiatives, developing and deploying projects, integrating software processes with project management processes, and implementing PMO processes for the portfolio.

One of his current areas of interest is the integration of finance and business aspects with software, project, and program management processes.

He is a certified Project Management Professional (PMP) from PMI, Certified Function Point Specialist (CFPS) from International Function Point users group, Certified Quality Auditor (CQA) from ASQ and Certified Software Quality Engineer (CSQE) from ASQ.

Rajneesh Sharma
Department of Finance
Haub School of Business
Saint Joseph's University
rsharma@sju.edu

Rajneesh Sharma is an assistant professor of finance at Saint Joseph's University.

4.2 Process Performance Models - Lessons Learned from 27 SCAMPI Appraisals for Small Companies

Authors

Bosheng Zhou and Pu Bai

Abstract

Five process performance models are presented and described one by one in this technical report according to the lessons learned from 27 SCAMPI appraisal cases. The emphasis of these descriptions is the inherent relationship among the five models.

Introduction

Cyber Keji has sponsored to train two CMM CBA IPI Lead Assessors, three SCAMPI Lead Appraisers, three candidate SCAMPLI Lead Appraisers, two instructors of Introduction to CMMI, and one PSP instructor.

During the past five years, Cyber Keji has conducted 45 formal appraisals using the CMM/CMMI model at different maturity levels, distributed from CMM/CMMI level 2 to CMM/CMMI level 4. More than half (27 cases, 60%) of the 45 appraisal cases use the SCAMPI method for appraisals especially during 2004 and 2005. Within the 27 appraisal cases, there are 7 (25.9%) at CMM level 2, 5 (18.5%) at CMM level, 4 (14.8%) at CMM level 4, and 11 (40.8%) at CMMI level 2. Only one organization of the 27 cases is of middle size (in total there are 1800 employees); all others are small companies (the largest one has less than 300 employees).

During the 27 appraisals, we have found that different process performance models can be used to estimate or predict the values of each process performance (such as the project size, cost, schedule, and quality) from the values of other subprocesses, projects, and work product measurements. According to our lessons learned from conducting 26 SCAMPI appraisals for small companies, the following five performance models are of special significant value to process improvement from the view points of both efficiency and accuracy.

These process performance models typically use process and product measurements collected throughout the lifecycle of the project to estimate progress towards achieving objectives. Whether the final practical objectives can be achieved cannot be known until later in the project's life. Whether it is possible to achieve the final objectives, however, can be estimated or predicted during the project progress by using the practical values measured, such as the values of project size, cost, schedule, and quality.

The five significant process performance models are the following:

- lifecycle model
- quality model
- resource model
- measurement model
- control model

Lifecycle Model

The lifecycle model is defined as the description of a lifecycle of a project. There are many different types of lifecycles, such as waterfall model, incremental model, spiral model, prototyping model, and iterative-incremental model (so called USDP). No matter how many types of lifecycles there are, the waterfall model can be viewed as the core of all other models. Therefore, the following describes the characteristics of the waterfall model as an example.

In general, a typical waterfall model is composed of several phases (as seen in Figure 21), such as requirements analysis phase, general/detailed design phase, implementation phase, system testing, and acceptance testing phase.

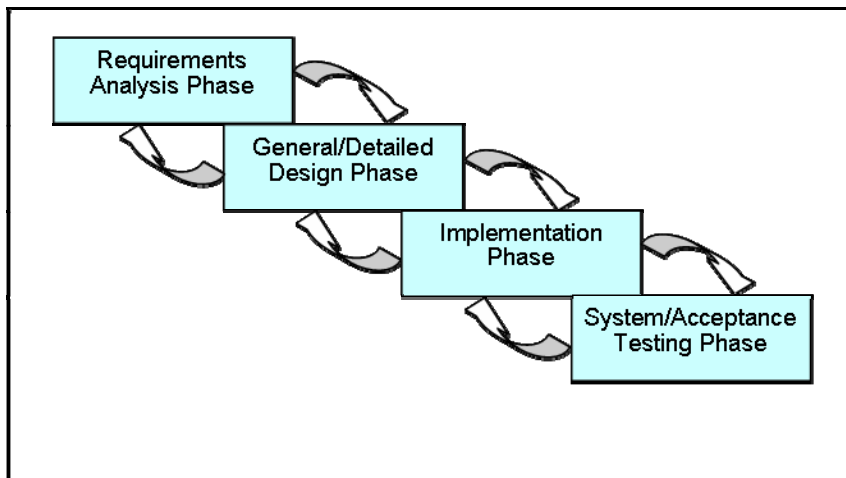


Figure 21: Waterfall Lifecycle Model

In describing the characteristics model, each phase of the waterfall model should be specified with the following 11 elements:

1. Purpose: The purpose of the phase. For example, for the general/detailed phase, the purpose is to design the architecture first based on the requirements specification, which is the output of requirements analysis phase, and then perform a detailed design from the architecture.
2. Roles and their Responsibilities: The roles and responsibilities involved during this phase. For example, for the general/detailed phase, the roles included are architect,

detailed designer, requirements analyzer, customer needs representative, tester, QA engineer, and configuration engineer.

3. Inputs/References: Inputs are defined as those which are consumed and/or modified during processing; references are defined as those which are used but not consumed and/or modified. Examples for the former are artifacts and stimulus from the outputs of the previous phase, other systems outputs, or some needed control signals. For example, for the general/detailed phase, the inputted baseline products include requirements specification, basic bi-directed requirements tracking matrix, and partial system test cases. The references include different design document checklists, different design document templates, etc., which generally are developed by the project team.
4. Entry Criteria: The benchmarks or requirements for each input artifact. An example of entry criteria for requirements specification is a document must have passed peer review and the defects found must have been fixed. The basic bi-directed requirements tracking matrix should include two matrixes and should be consistent between customer requirements and requirements specification and between requirements specification and partial system test cases.
5. Activities (Tasks): It is performed to produce output artifacts from input artifacts, which can be represented in partial ordered diagrams. For example, for the general/detailed phase, the activities typically are those described in Technical Resolution process area, including both design activities (such as architecture design, buy/make/reuse decision, and interface design) and design peer review activities (such as formal evaluation to alternative designs and design documents peer reviews).
6. Outputs: The work products produced at this phase that will be used as the inputs for the next phase. The outputted baseline products include the architecture/detailed design document, refined/supplemented bi-directed requirements tracking matrix (two more matrixes added are the matrix between requirements specification and design document, and between design document and integrated test cases.), refined/supplemented system test cases, and integrated test cases.
7. Exit Criteria: The benchmark for the output products used during verification and validation activities. Checklists for each of the design documents, the bi-directed tracking matrixes, and the test cases are the fundamental verification and validation methods. All outputs to be outputted should have passed the corresponding exit criteria.
8. Resource: The necessary human resource, funding, facilities, and office rooms, etc. to be used at this phase. The actual cost will be equal to the product of the number of each human resource times the corresponding effort plus the cost for funding, facilities, and rent expenditure of office room.
9. Constraints: The constraints (the values and their corresponding thresholds) on schedule, resource, costs, quality, etc. provided and required by management. In practice, the sum of the needed expenditure mentioned above (item 8: Resource) is limited by the actual resource available, including human resource, funding, office room, etc. Most of the

management activities include performing a trade-off study for balancing the required and the available resource and effectively using the resource available.

10. Measurement and Analysis: Measurement and analysis is to define the needed product and process metrics during each phase of the lifecycle. These metrics should be collected and analyzed, and can be used to understand the current status, make corrections if significant deviation is found, and confirm whether the expected objectives have been achieved in proper cost and rational schedule.
11. Applicable Standards and Disciplines: ISO, National, Industry, and Enterprise standards, processes, procedures, templates, and disciplines to be used at this phase. Language references, design document standards, and coding standards are typical examples that are generally developed by respected communities.

The characteristics for each phase of a lifecycle model is depicted in Figure 22. In this figure, only phase 1 of a lifecycle is described.

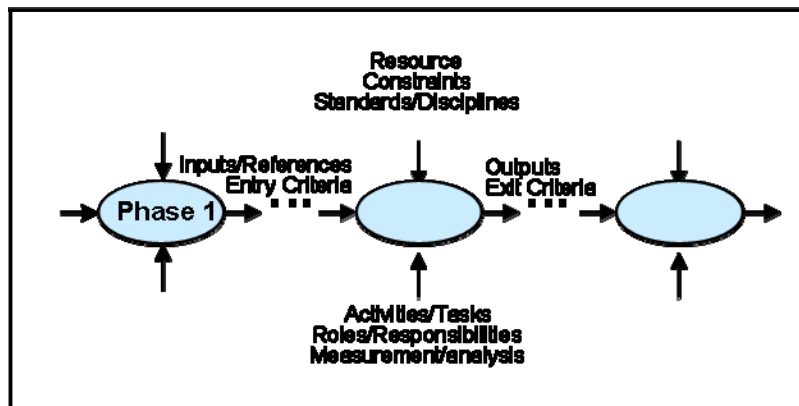


Figure 22: The Purpose for Each Phase

The lifecycle model description can be used as the guideline for developing configuration management library. For example, assume that outputs in each phase should be classified into three categories (based on customers' requirements):

- Class A: baseline products, directly related the final products that will be delivered to the customer or kept in organizational asset library
- Class B: non-baseline but to be controlled products, such as different plans
- Class C: working products, such as timesheets, a variety of reports, meeting minutes; only to be referenced as history in the later phase

Assume that, the inputs of Phase 1 and the outputs of each Phase i , $i=1, \dots, n$, can be presented as follows:

- $(\text{inputs})_{1A_j}$: The Class A inputs of Phase 1, $j=1, \dots, m_{1A}$, here m_{1A} is the number of Class A inputs of Phase 1.

- $(inputs)_{1Bj}$: The Class B inputs of Phase 1, $j=1, \dots, miB$, here miB is the number of Class B inputs of Phase 1.
- $(inputs)_{1Cj}$: The Class C inputs of Phase 1, $j=1, \dots, miC$, here miC is the number of Class C inputs of Phase 1.
- $(outputs)_{iAj}$: The Class A outputs produced at Phase i . $i=1, \dots, n$, $j=1, \dots, miA$, here miA is the number of Class A outputs produced during phase i .
- $(outputs)_{iBj}$: The Class B outputs produced at Phase i . $i=1, \dots, n$, $j=1, \dots, miB$, here miB is the number of Class B outputs produced during phase i .
- $(outputs)_{iCj}$: The Class C outputs produced at Phase i . $i=1, \dots, n$, $j=1, \dots, miC$, here miC is the number of Class C outputs produced during phase i .

As described above, all the products which should be managed are: $(inputs)_{1Aj}$, $(inputs)_{1Bj}$, and $(inputs)_{1Cj}$; $(outputs)_{iAj}$, $(outputs)_{iBj}$, and $(outputs)_{iCj}$, $i=1, \dots, n$. And the CM Library will be classified as A, B, and C as follows:

- CM Library A = $\sum[(inputs)_{1Aj} + (outputs)_{iAj}]$, $i=1, \dots, n$.
- CM Library B = $\sum[(inputs)_{1Bj} + (outputs)_{iBj}]$, $i=1, \dots, n$.
- CM Library C = $\sum[(inputs)_{1Cj} + (outputs)_{iCj}]$, $i=1, \dots, n$.

Here n is the number of the phase that were described above. The three formulas can be used as a benchmark for detecting whether all necessary work products have been managed and whether they already have been put in the right place.

In other words, it should be possible and necessary to extract information from the input products of the first phase (Phase 1) and the output products of all other phases of a lifecycle model to establish a configuration management plan. If we really do so, the consistencies among different documents are naturally reserved.

It should be mentioned that the following are some of the requirements for the description of lifecycle:

- to classify the inputs and outputs into three (the number can be defined by customer) classes
- to build the structure of CM Library from the description
- to specify the input products should meet the input criteria
- to specify the output products should meet the output criteria

The following data and the corresponding points should be given in the description of the lifecycle model:

- the number of resource type needed for each phase
- the value of constraint, such as the duration, efforts, and quality (expected value and their threshold)

- in human resource effort, the amount of effort times duration should accommodate the effort for each phase
- in giving the expected value and their threshold of quality (described in number of product and process defects), the severity level should be also considered

Quality Model

On one hand, the ISO 9126 defines software quality as six attributes: functionality, reliability, time efficiency, resource efficiency, maintainability, and portability. In recent years, security, safety, and flexibility are given more attention. Each of the attributes are further defined by sub-attributes. These are only requirements for software/system. They do not mention how to reach each of the expected quality attributes.

On the other hand, quality assurance, quality control, and total quality management are three quality technologies that addressed the techniques for quality assurance for process, quality control for products, and quality efforts from both process and product views and from technical and social views. However, none of them discuss the correlations among different methods.

Based on our experience and systematic thinking during 27 SCAMPI appraisals, we present a brand new idea about software/system product quality, which is inspired from the manufacturing industry, where each part is specified and produced under certain requirements or constraints. The requirements insist that the manufacturing of the parts must be precise and that the surface of the parts must be smooth and part's material must have the correct hardness. If each part satisfies the specified requirements, during product integration the product is obtained by assembling parts. The quality model is based on the similar concepts.

Therefore, quality model (also called quality design), in both theory and practice, is defined as the quantitative description of the distribution of quality objectives using three dimensions that are orthogonal between any two.

1. The vertical view of quality objectives distribution: specifying the allowed number of injected defects within each phase of the lifecycle and specifying the required defect removal rate for each phase of the lifecycle. During specifying, one should distinguish between product defects and process defects.

Let $(\text{Defects-injected})_i$ represent the allowed number of injected defects within each phase of the lifecycle, let $(\text{Defects-fixed})_i$ represent the allowed number of removed defects within each phase of the lifecycle, then the process benefit of phase i of the lifecycle is equal to:

$$(\text{Process Benefit})_i = (\text{Defects-fixed})_i / [(\text{Defects-escaped})_j + (\text{Defects-injected})_i],$$

Here $j=i-1, i=1, \dots, n, (\text{Defects-escaped})_0 = 0$.

The total process benefit of the lifecycle is equal to the product of process benefit of each lifecycle phase, that is:

$$(\text{Process Benefit})_{\text{total}} = (\text{Process Benefit})_1 \times \dots \times (\text{Process Benefit})_n$$

2. The horizontal view of quality objectives distribution: the allowed number of defects is assigned to product components according to their size and importance (a WBS diagram may be useful). That is:

$$(\text{Defects})_{\text{total}} = (\text{Defects})_1 + \dots + (\text{Defects})_n,$$

Here n is the total number of components of the system. However, some correcting coefficient should be considered during defects assignment:

- considering a variety of correcting coefficients, such as size, type, complexity, technique accepted, and human resource status
- considering the interface defects when performing product integration and the defects arisen from transfer parameters

Therefore, the allowed defects number of each component can be computed according to the following formula:

$$(\text{Defects})_{\text{total}} = \left\{ \sum_{i=1, \dots, n} k_{1i} k_{2i} k_{3i} k_{4i} k_{5i} (\text{Defects})_i \right\} + \left\{ \sum_{j=1, \dots, m} (\text{Defects})_j \right\}$$

Here n is the number of total components of the software/system and m is the number of total interfaces of the software/system.

3. The finding method view of quality objectives distribution, which is to specify that, during each phase of the lifecycle, each defect finding method should detect a certain number of defects.

$$(\text{Defects})_{\text{total}} = \left\{ \sum_{i=1, \dots, 6} \left[\sum_{j=1, \dots, m} (\text{Defects})_{ij} \right] \right\}$$

Here m is the number of phases of the lifecycle defined.

Here the finding method view of quality objectives distribution can be classified into six categories; both development activities and process activities are included.

- noncompliance issues found during review especial by PPQA activities
- peer review at each phase of the lifecycle, especially emphasizing on requirements/design reviews and by team organized by project/technique lead
- different types of testing
- milestone reviews
- problem report/improvement proposal by different kinds of people, including, but not limited to, stakeholders and customers
- process appraisal findings by the appraisal team against at predefined representation and its levels

It should be emphasized that, in quality model, the defects collected should be categorized into different levels according to their severity (three levels are most common and are used as our example):

- Class 1 errors: causing basic system functionality not achieved, system abnormally stopped or crashed, or the contents of database destroyed
- Class 2 errors: tiny functionality is not achieved; efficiency is not as expected, etc.
- Class 3 errors: not easy to use, not pretty enough, etc.

Based on case studies, proper quality model use can ease the difficulty of process improvement. Our lessons learned can be classified into the following two aspects:

- Quality criteria: quality criteria shall be depicted as exit criteria of each phase in the lifecycle description. In other words, the set of exit criteria for each phase plus the entry criteria for the first phase of the lifecycle is the quality criteria for the project. That is:

$$(\text{Quality criteria})_{\text{total}} = (\text{Entry criteria})_1 + \sum_{i=1, \dots, n} (\text{Exit criteria})_i$$

- Quality method: in order to achieve expected performance objectives, during the development of the product, different defects finding methods are used. That is:

$$(\text{Defects})_{\text{total}} = \sum_{i=1, \dots, n} \{ \sum_{j=1, \dots, m} (\text{Defects})_{ij} + \sum_{j=1, \dots, m} (\text{Defects})_{kj} \}, k=1, \dots, 6 =$$

Here n is the number of total components of the software/system, m is the number of total interfaces of the software/system, k means to different defects finding methods. However, the emphasis should be put on testing and different kinds of reviews (especially the peer reviews of requirements and design and process appraisals).

For the first point, the exit criteria of implementation (coding and unit testing) phase are presented here as an example.

- Code will cover 100% design.
- The transfer statement will be used properly to keep the structured feature.
- The mapping relationship (tracking matrix) between code and unit test cases and between code and design will be established.
- The previous mapping relationship (tracking matrix) between design and requirements specification and between requirements specification and customer needs, etc. will have been elaborated or modified.
- All the test cases prepared in a unit test plan will be successfully performed. All the test cases that have not passed, if any, will have been properly analyzed.
- 100% statement coverage will be required for unit testing.
- Peer review will have been performed for newly coded code and core code of the system.
- The amount of LOC conducted peer review will be no less than 25%-35% of the total code of the system.
- The effort consumed on peer review will be no less than 20%-30 of the total effort of coding and unit phase.
- Different severity level of defects found in this phase will be less than a assigned value (such as 1.0-2.0/KLOC).
- Latent defects will be less than a assigned value (such as 0.2%-0.5/KLOC).
- Severity class A defects found will have been thoroughly fixed, and severity class B defects that have not been fixed will be less than assigned value (such as 20%).

During each phase of lifecycle, defect data found will be recorded, accumulated, and analyzed for the following aspects based on our experience, and then the specific course and the root course for these defects will be removed at different capability maturity levels.

- different distributions of defects based on the following:

- finding method
- finding phase
- fixing phase
- severity level of defects
- module of the system
- process benefit of each phase of the lifecycle
- efficiency of different method of finding defects
- 20%/80% theorem to find the weak part of the system
- recording and accounting of the occurring number of the same defect using tool
- 20%/80% theorem to analysis the rate/effectiveness for each test case
- computation of the resident time from finding date and fixing date of the defect and the ordering of them from longer time to less time
- prediction of the latent defects when delivering products

Resource Model

The resource model is the consuming model of human beings' efforts in person-week (or person-month, or person-year, etc.), which can be used to determine the following:

- percentage of each phase time period to the total time period of the lifecycle
- percentage of each phase effort to the total effort of the lifecycle
- percentage of each task type effort to the total effort of each phase

It should be mentioned, in developing cost model, in addition to considering human beings efforts, other costs should also be taken into account, such as, hardware, other facilities, office room, and different communication and transportation expenditures.

All of the organizations need to estimate, analyze, and predict the process performance, to assess the potential ROI for process improvement activities. Establishing a resource model can help organization achieve these goals.

Any high maturity organizations should use their own projects' data to build their own resource model at project level first and at organizational level next.

Obviously the quality of the resource model is heavily depended on the quality of data. Therefore, the following provisions shall be followed to ensure the quality of the resource model, which can be viewed as the prerequisite of building the resource model:

- Pay special attention to distinguishing the size, software/system type and lifecycle category of projects because the costs for different sizes and for different software/system types are very different and because different categories of the lifecycle cannot be synchronized and compared.
- With a focus on the record and measure, accumulate and analyze the data both by process and product. Four principles are presented for this issue:

- Usefulness: all the data collected should meet the requirements for information needs which are specified as item 10 of the lifecycle description (see the Lifecycle Model section of this report).
- Trueness: employees are encouraged to fill in real data consumed by them during daily activities no matter how many effective work hours they have.
- Timeness: employees should record the data when it is still in their memory. The slogan “fill-in timesheet daily, accumulate data weekly, and analyze data monthly” is presented to warn every employee.
- Reasonableness: data should be kept as three digit in recording, accumulating, and analyzing.
- Prepare an organizational document for specifying roles (task types) within the whole organization. Each role will have a work effort data flow. There are a certain number of roles, and they will have the same number of data flows.

From the discussion above, it is obvious that we need to pay special attention to fill in daily timesheets. The project size, software/system type, lifecycle category, and the current phase should be filled in by the data collector. The discussion is summarized in the Table 13:

Table 13: Basic Attributes of Resource Model

	Basic Characteristics	Descriptions of the characteristics
1	System common characteristics	System name
		Working language (such as C++, Java, etc.)
		Successful level of the project
2	System type	Simple, Moderate, Embedded
		New development, Assembling from components, Maintenance
		Operating system, Compiler, Tool development, Applications
	Product size	Requirements number/Function point/Loc/Use case number, etc.
		Micro, small, meddle, large, huge
Lifecycle type	Such as Waterfall, USDP, etc.	
Project team size	Micro, small, meddle, large, huge (in person number)	
3	Total effort	Micro, small, meddle, large, huge (in person week/person month)

The percentage of each phase effort to the total effort of the

	Basic Characteristics	Descriptions of the characteristics
		lifecycle
		The percentage of each task type effort to the total effort of each phase
	Project duration	The time period of the whole project (in week/month/year)
		The percentage of each phase time period to the total time period of the lifecycle
4	System complexity	Program complexity MaCabe/HalStead
		Computing complexity
	Technology novelty level selected	Team member's expertise in selected working language
	Team member status	Expertise level, devotion, spirit, morale, healthy status, etc.

In order to improve estimating accuracy, each project should use their own resource model (history data) to conduct estimations for schedule, effort, cost, etc. When creating project resource model, the following issues should be taken into account:

- Each project should have an independent resource model. If people play more than one role in different projects, they should build more than one resource model to ensure each project has its own resource model.
- At an organizational level, task types should be defined and used within the whole organization.
- A unified timesheet template should be defined and used within the whole organization.
- Each project member should fill in the timesheet daily, accumulate data weekly, and analyze data monthly.
- Each project automatically accumulates personal timesheet data weekly to form the project team weekly report.
- Software/system category, product size, and lifecycle type should be uniformly defined at the organizational level.
- The project resource model should be partitioned by software/system category, product size, and lifecycle type.
- It should also be determined if the project is very successful, successful, or failed after the project has been terminated.

During each development phase, the project resource model should be formed from the weekly extract effort information (person-day) from the timesheets of employees with the same role; this action should be repeated for every role. The concrete procedure can be characterized as follows:

- Step 1: form the resource expenditure curve for one role type.
- Step 2: form resource expenditure curve for other roles. There should be the same number of roles in the project as there are resource expenditure curves for the project.
- Step 3: form the resource expenditure curve for the project one phase after another. The curve is the resource model of the project.

Based on the project resource model, the organization resource model can be built up. According to statistics control theory, in order to form the organization resource model, there first must be eight effective project resource models. The concrete procedure can be characterized as follows:

- Step 1: transform the above project resource model into a non-unit equivalent project resource model one by one. Within a project resource model, one curve corresponds to each role. It is obvious that the last points of these curves, which are at the last milestone, are equal to 100%. All the different milestone points for each role resource model are represented in percentages.
- Step 2: compute the mean value of the time period at each milestone from those effective project resource models. As mentioned before, there are at least eight effective project resource models. It is obvious that the mean value of time period at the last milestone is still equal to 100%. All the mean values of other milestone points for the organization resource model are represented in percentages.
- Step 3: compute the mean value of effort for each role one phase after another from those effective project resource models. It is obvious that the sum of different role efforts for each phase is equal to 100%. Each role's effort in the sum for the same phase is represented as a percentage.
- Step 4 is to compute the mean value of effort for the whole lifecycle from the curves. Assume that the total effort for the whole lifecycle is equal to 100%. The effort of each development phase is represented as a percentage.
- Note that for all of the percentages mentioned in Step 1 through Step 4, no matter the time period or effort, the absolute value for time period (day or month) and for effort (man-day) must be mentioned.

An example of resource model diagram is shown in Figure 23. When creating a resource model, the following points should be taken into account:

- The collection density should be increased (in general, one data each week, and so a slogan "daily filling in, weekly reporting and monthly analyzing" is presented); actually, for the resource model, the denser the collection, the more precise it will be.

- The depiction density of the resource model should be also increased. In general, the depiction density should be equal to the data collection density mentioned above (i.e., one data point for each equivalent week).
- Pay special attention to abnormal points, such as discontinuous points, and abnormally high or low points. From the example curves in Figure 23, the value of coding and unit testing is too high (>42.00%) and the value of review is too low (<5.00%) and are non-expected values.

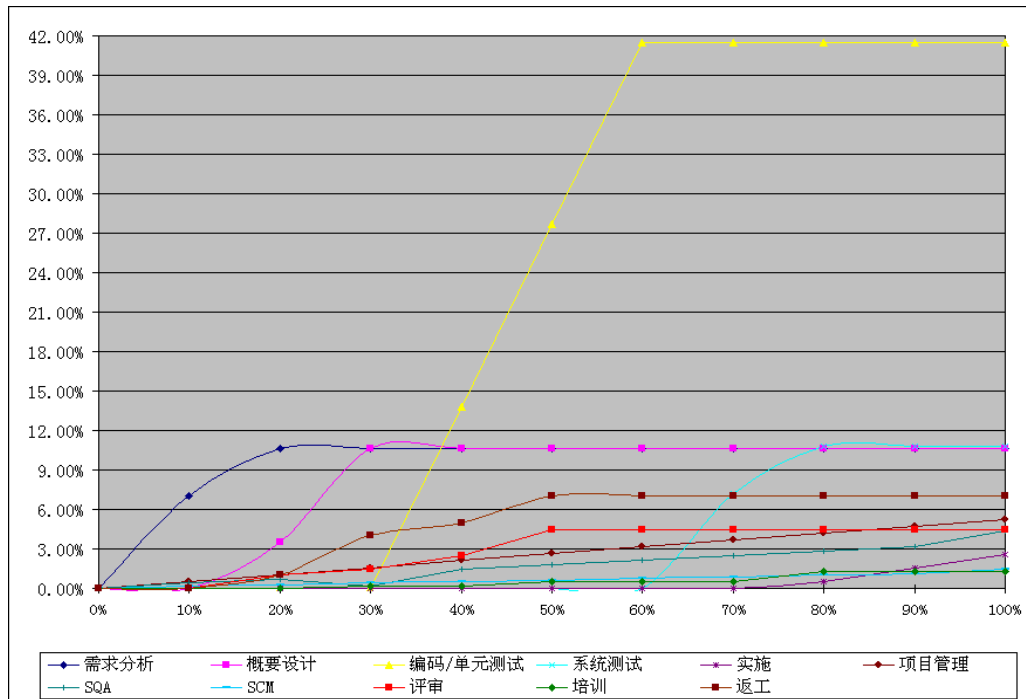


Figure 23: An Example of a Resource Model

Resource models can be used to detect non-expected and abnormal issues. There may be some specific sources, and there may be generic sources. Then proper correction action may be taken to fix these issues. If the situation is not improved or improved enough, an organization may need to try again to find the specific sources or generic sources until the expected goals are satisfied.

Measurement Model

As mentioned in the discussion above, the resource model and quality model can be built by extracting, collecting, and analyzing a variety of quantitative data that should have already been specified in the lifecycle model. The measurement model is built in the same manner. Here the measurement model is defined as the metrics specification during each phase of the lifecycle model. Of course, those metrics should be significant to meet the information needs of an organization.

Therefore, a measurement model should directly, clearly, and quantitatively answer the following two questions:

- What are the direct metrics and what is their use in each phase of a selected lifecycle of a project, and how can they be measured?
- What are the indirect metrics in each phase of a selected lifecycle of a project, and how can they be computed from direct metrics?

For example, in CMM/CMMI, the key metrics of each project include the following:

- size, cost, effort, schedule, and milestone review situations
- changing rate of different requirements (new added, deleted, and changed) and the changing number of the same requirements
- changing rate of different configuration items (new added, deleted, and changed) and the changing number of the same configuration items
- expected value and the threshold value of important metrics (such as size, effort, cost, schedule, and quality)
- the number of defects during different modes of finding defects

When data are collected, the following points should be given special attention:

- usability (meet business goals)
- true data (not for performance review)
- timely record (e.g., a timesheet that should be filled in daily and reported weekly)
- scienceness (effective number of data)
- semi-automation/automation to increase the efficiency

In item 10 of the lifecycle model, it is stated that measurement and analysis define the needed product and process metrics during each phase of the lifecycle. These metrics should be collected, and analyzed, and can be used to understand the current status, to make corrections if significant deviation is found, and to confirm whether the expected objectives have been achieved in proper cost and rational schedule. In other words, the measurement model should be formed by extracting the information mentioned in the description of each lifecycle phase. Table 14 and Table 15 are examples of the design phase of the lifecycle.

Table 14: An Example—Metrics for Design Phase

Category	Metrics Name	Direct/ Indirect
size	the number of design diagram during system analysis	Direct
	the number of tables needed in database	Direct

Category	Metrics Name	Direct/ Indirect
	the number of word segment for each database	Direct
	the number of subsystems	Direct
	the number of interfaces	Direct
	the number of components	Direct
	the number of classes	Direct
	the complexity of classes	Direct
	the number of modules	Direct
	the number of fan-in modules	Direct
	the number of fan-out modules	Direct
	the cohesion of two modules/the number of parameters transformed	Direct
	the number of pages of design document	Direct
	the number of integration test cases	Direct
quality	the number of defects found by PPQA	Direct
	the number of defects found during review	Direct
	the number of defects injected in requirements phase	Direct
	the number of defects found in preparation time	Direct
	the number of L1 defects	Direct
	the number of L2 defects	Direct
	the number of L3 defects	Direct
	the number of L4 defects	Direct
	the number of L5 defects	Direct
	the total number of defects	Direct

Category	Metrics Name	Direct/ Indirect
	the number of defects found	Direct
	the number of latent defects	Direct
	the number of defects based on type	Direct
	the number of defects opened	Direct
	the number of defects closed	Direct
	defects found date	Direct
	defects clear date	Direct
	the rate of defects escaped	Indirect
	the rate of defects cleared	Indirect
	the resident time of defects	Indirect
	deviation rate of quality goal	Indirect
	review preparation time	Direct
	review time	Direct
schedule	design planned start date	Direct
	design planned terminate date	Direct
	design actual start date	Direct
	design actual terminate date	Direct
	design start deviation	Indirect
	design terminate deviation	Indirect
	planned duration	Indirect
	actual duration	Indirect
	duration deviation	Indirect
	duration deviation efficiency	Indirect

Category	Metrics Name	Direct/ Indirect
	the number of problems described	Direct
	the number of problems closed	Direct
	the efficiency of problem solving	Indirect
resource and cost	planned needed design facilities	Direct
	actual needed design facilities	Direct
	planned design environment components	Direct
	actual design environment components	Direct
	assigned resource deviation	Indirect
	the total planned design effort	Direct
	the total effort of actual design	Direct
	the effort for create integration testing	Direct
	the effort for modifying defects	Direct
	the effort for verifying defects	Direct
	review effort	Direct
	the effort consumed by PPQA	Direct
	design effort deviation	Indirect
	review efficiency	Indirect
	rework efficiency	Indirect
	PPQA efficiency	Indirect
	the effort rate of task type in this phase	Indirect
	design efficiency	Indirect
	planned cost	Direct
	actual cost	Direct

Category	Metrics Name	Direct/ Indirect
	the number of planned assignment	Direct
	the number of actually assigned	Direct
	cost deviation	Indirect
	cost deviation efficiency	Indirect

Table 15: Indirect Metrics Transform Computing Table

Metrics Name	Direct/ Indirect	Indirect Metrics Computing
the rate of defects escaped	Indirect	the number of latent defects/the total number of defects
the rate of defects cleared	Indirect	the number of defects fixed/the total number of defects
the resident time of defects	Indirect	defects cleared date - defects found date
deviation rate of quality goal	Indirect	the number of defects actually found/the number of defects planned found
design start deviation	Indirect	actual design start date - planned design start date
design terminate deviation	Indirect	design actual terminate date - design planned terminate date
planned duration	Indirect	design planned start date - design planned terminate date
actual duration	Indirect	design actual start date - design actual terminate date
duration deviation	Indirect	actual duration/planned duration
duration deviation efficiency	Indirect	duration deviation/duration deviation rate
the efficiency of problem solving	Indirect	the number of problems closed/the number of problems described

Metrics Name	Direct/ Indirect	Indirect Metrics Computing
resource injection deviation	Indirect	actual injected resource - planned injected resource
design effort deviation	Indirect	the total number of actual design effort - the total number of planned design
review efficiency	Indirect	the number of defects found/review effort
rework efficiency	Indirect	the number of defects fixed during rework/the rework effort
PPQA efficiency	Indirect	the number of defects found by PPQA/the effort of PPQA
the phase effort rate of task type	Indirect	the effort of different task type/the total effort of this phase
design efficiency	Indirect	the number of pages of design document/effort
cost deviation efficiency	Indirect	cost deviation/planned cost

Control Model

Control Model is the specification of control mode. It can be classified into three categories:

- The control mode of process improvement, which is the macro-adjust control mode. For the CMM and CMMI models, this control model should follow the IDEAL model. That is to say, the macro-adjust mode includes five steps: initiating, diagnosing, establishing, acting, and learning.
 - The control mode of process enacting. Process improvement itself should be viewed as a project, it is also necessary to plan, implement, monitor, and control, through a variety of meetings and reports during process improvement.
 - The control mode of process control threshold, which is the micro-adjust control mode. For the CMM and CMMI model, this control model is distinguished into three different modes: reactive control, proactive control, and statistics control.
1. Reactive control mode. For example, at CMM/CMMI level 2, control mode is reactive, an expected value, and its corresponding threshold value should be assigned to each control value. However, in addition to the control value itself, the following issues should also be taken into account, the requirements for reactive control are as follows:
 - to assign an expected value to each task, including size, effort and cost, schedule, and quality

- to assign a control threshold value to each of the expected value mentioned above. For example, the threshold value could be assigned to 5%-25% for schedule
- to understand the business goal as the instruction of assigning the control threshold value. For example, if the schedule is much more of a concern, then the control threshold value of schedule should be less than others; if the cost is much more of a concern, then the control threshold value of cost should be narrower than others.
- to distinguish critical path and non-critical path. It is project manager's duty to identify critical path and then pay much more attention to it. It should be noticed that, of course, with the progress of project the critical path could be changed.
- to take correct action according to relative variation (for each phase of the lifecycle) and to report based on relative and absolute variation
- to set a warning value based on the control value (threshold) according to system dynamics. That is to say, we need to distinguish control threshold value and warning value of the control threshold value.

The control mode for Reactive Model at CMM/CMMI level 2 can be depicted as follows:

(Reactive Model)_{at CMM/CMMI L2} = {Control value, their threshold, Business goal, Critical path/Non-critical path are qualitatively considered, Relative variation/absolute variation, Warning value}

2. Proactive mode. At CMM/CMMI level 3, most of the control should use proactive mode. At CMM/CMMI level 4 or 5, all of the control should use proactive mode. In addition to the six points mentioned in the reactive mode that should be obeyed, the following two issues should be taken into account:
 - dynamically and proportionally assign lack of time for each non-critical path, according to their duration, to each task on the non-critical path, and to keep each task with the same threshold.
 - use the non-extrapolation trick to compute the trends, according to the current and the previous two to four points' values to get the upcoming two to three points' values, and to depict them in a graph.

Reactive Model_{at CMM/CMMI L3/L4/L5} = {{Reactive Model_{at CMM/CMMI L3}}, quantitatively consider critical path/non-critical path (lack time), non-extrapolation}

3. Statistics control. For a high maturity organization, for example, one that has already reached CMM/CMMI level 4 or 5, it is still useful to detect whether a subprocess is abnormal. Based on the principle of economics, the following four detection rules should be used:
 - Rule 1: there is one measurement point outside the control limit “3σ”
 - Rule 2: in continuous 3 measurement points, at the least, there are 2 falling in [2σ,3σ]
 - Rule 3: in continuous 5 measurement points, at the least, there are 4 falling in [σ,2σ]
 - Rule 4: in continuous 8 measurement points, all of them fall on one side of the mean line

That is to say, in a general sense, the proper value of an expected value is limited by [-3σ, +3σ], here the σ is the variation of the expected value. In a statistics control sense, the four rules should be used as detect criteria.

Remarks

- The relationship among the five models is shown in Figure 24. It implies that, as mentioned above, the quality model, resource model, measurement model, and control model all can be derived from the lifecycle model.
- Based on our experience and lessons learned, the five models can significantly ease the pain of CMMI appraisals for all sizes of companies, including small companies (reduce effort, increase quality).
- A case study is still needed and is still being developing. Process improvement effect especially should be measured and quantitatively analyzed during a designated time period, and the following points should be examined in that time period:
 - How much productivity is increased
 - How much defect rate is decreased
 - How much estimating accuracy is increased
- Three needs especially need to be considered and conquered:
 - using the model at the enterprise level
 - considering culture and tradition when developing practice
 - thinking how to motivate people factors
- A model-driven process supporting environment is absolutely needed and is in development at Cyber Keji.

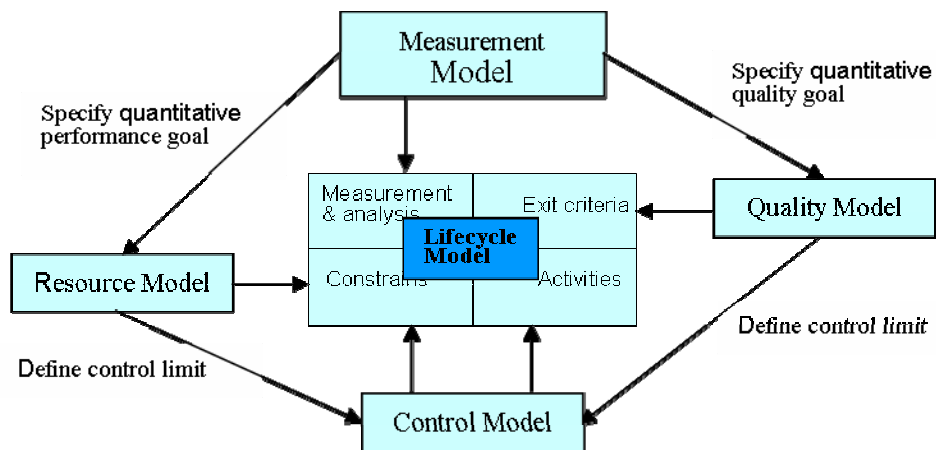


Figure 24: Model-driven Process

Biographies

Bosheng Zhou

Cyber Keji Park, Inc.

bszhou@cyberspi.com.cn

Bosheng Zhou, the CTO of Cyber Keji, is a professor and PhD student's supervisor at the Software Engineering Institute of Beihang University. He is an SEI-authorized CMM CBA IPI Lead Assessor, SCAMPI Lead Appraiser, and Introduction to CMMI instructor. He has 47 years experience in the computer science field and specializes in software engineering and process engineering. Zhou has conducted more than 100 training courses, consultations, and appraisals for more than 40 organizations.

Pu Bai

Cyber Keji Park, Inc.

baipu@cyberspi.com.cn

Pu Bai, a senior consultant at Cyber Keji, is a visiting professor at the Software College of Beifang Jiaotong University. She has more than ten years of experience in software requirements analysis, design, and implementation, and has more than four years experience in CMM/CMMI training, consulting, and appraising.

4.3 A Pattern-Based Approach to Deploy Process Improvements in Small Settings

Authors

Antonio de Amescua Seco, Gonzalo Cuevas Agustín, Javier García Guzmán, Juan Llorens, Paloma Martínez, and Diego Martín

Abstract

PM-CAKE is a processes and projects knowledge management framework for software intensive organizations to be used by project managers, software engineers, process management groups, and managers of information technologies units.

This framework allows transferring the know-how of performing the organization's projects from individuals to the company. So, when the knowledge is conveniently indexed, classified, and available to the organization's members, it will be available to different types of operational, tactical, and strategic utilities.

Description of the Problem Related to Process Improvement Deployment in Small Settings

The implementation of a software process improvement program is very expensive, especially for SMEs and those organizations that first undertake an initiative of this type. During our experience in twelve software process improvement programs, we have concluded that the activities related to process improvement deployment and training needs account for almost a third of the effort in the improvement project effort. These activities also are the most difficult to achieve due to human resistance to change.

In addition, most software organizations do not have the technology and the appropriate tools for process improvement deployment because many of the tools are completely new to the software engineers and do not integrate seamlessly with the tools already being used.

Many SPI researchers and related organizations are in accordance with this point of view, so the tools used to deploy and use process assets comprise an intensive area of R&D work related to software process improvement. In this sense, it is important to highlight the agreements between SEI and Microsoft and Teraquest Company and Borland, which intend to enrich the software development tools commercialized by Microsoft and Borland. The tools have functionalities that provide software engineers with a seamless access to process assets defined during a software process improvement initiative.

The functionalities that will be provided by the next generation software development tools will allow the software engineer to access seamlessly the functionalities related to project management, application modeling and design, components integration, software testing,

quality assurance, configuration management, requirements management, and e-collaboration tools to implement distributed software engineering.

This approach will allow software engineers to use a process that is already adapted to a concrete project, but if we want to deploy a process efficiently, the project managers and software engineers need to know how to adapt and use the process for each software project. In this sense, Rational, Microsoft, and Borland software development tools provide the possibility for performing this adaptation, but they do not provide any help related on how to do it. The software project managers need to be coached continuously in process adaptation. This is one of the most important success factors of SPI programs.

In our experience in small software development units, the most efficient way to coach project managers is to give them access to an expert in project planning. This expert should have knowledge about the organization's software process definition, efficient practice, and project characteristics. Use of this knowledge will result in efficient and high quality software project management. So, our R&D initiatives focus on an efficient deployment of software process improvements and are centered on effective knowledge management for an organization.

PM-CAKE fundamentals

The Project "A platform to model, reuse and measure software process management (GPS-TIN-2004-0783)," funded by Spanish government, has the main goal to making a tool that helps software engineers satisfy effective practices that are defined by most well-known international institutions such us ISO, SEI, and Project Management Institute. In this context, the research has defined a knowledge management environment called PM-CAKE¹⁹.

PM-CAKE (Process Management Computer Aided Knowledge Environment) is a process and project knowledge management framework for software intensive organizations that can be used by project managers, software engineers, process management groups, and managers of information technology units. This framework will allow an organization to transfer the know how of performing projects from the organization's experts to the whole company. So, when the knowledge is conveniently indexed, classified, and available to the organization's project managers, to the organization can efficiently adapt software process into projects.

So, in relation to a project's knowledge management area, PM-CAKE is a knowledge management tool that helps project managers plan new software projects that comply with the organization's process definition and include the organization's previous experience gathered by the organization's personnel. The main benefits of PM-CAKE are related to the following:

- cost savings because it is not necessary to make strong investments in more tools and training

¹⁹ During conference presentation, we will show a demo of PM-CAKE environment

- organization know how is transferred to the whole company, allowing a repeatable and controlled project management
- efficient support to SPI programs through an analytic study of software projects

Software Project Management Data Sources

The first problem to solve for this knowledge institutionalization program consists of identifying and processing software project management data sources. For this issue, through the analysis of many software intensive organizations, we have identified two situations:

1. Some project managers document the project management activities using MS-Project or structured MS-Excel files. Moreover, they store the managerial and development artifacts (doc, calculation sheets, presentations, software models, follow up reports) obtained through the project execution.
2. Other project managers store the same information, but in an unstructured manner, using text files or presentation slides, so these data sources are not able to be processed in an automatic way with reasonable costs. This information could be processed if it were translated to structured formats.

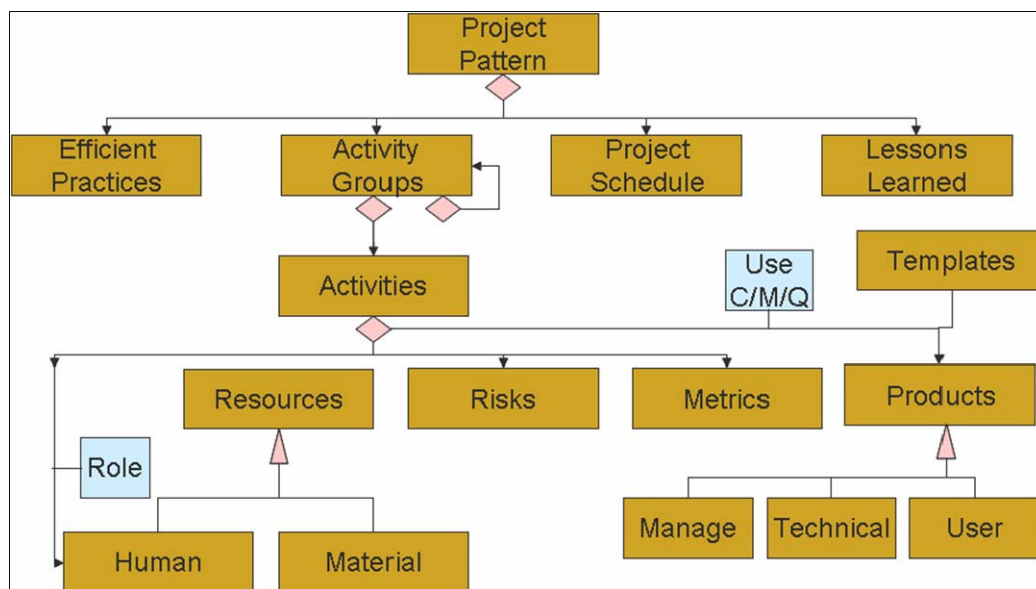


Figure 25: Structure of Project Patterns

On the other hand, in order to guarantee that the projects fulfill quality requirements, the new projects should include some practices coming from software process reference models (e.g., CMMI, continuous representation).

So in this sense, the organization will also store software project patterns that integrate into the organization's projects activities and some other practices that will ensure the quality and productivity of the projects. These patterns will be adapted to the typology and characteristics of the projects and services provided, such as types of functionality, lifecycle, processes

considered, complexity factors, project team characteristics, and development environments. These patterns will be composed of knowledge items related to projects management and templates of the artifacts that should be generated during the project execution. The templates of the artifacts included in the project patterns will be related to project estimation, data collection of project execution, software requirements specification, tests plan, configuration management plans, quality plans, etc. These templates will be provided in files with rtf, xls, ppt, or mpp formats or others if it is considered necessary. Some product examples will be included in the project pattern.

Project Planning

Once PM-CAKE repository contains the organization's knowledge related to project management, the project managers will be able to plan new projects based on previous experiences. The procedure to plan a new project with PM-CAKE is described in the following three points:

1. Project categorization

The first step will allow the creation of the project; during this creation; the new project will be categorized. The qualitative specification consists of the selection of estimated values related to project typology: types of functionality, lifecycle, processes considered complexity factors, project team characteristics, and development environments.

Afterward, the project manager should be provided with the quantitative estimation of the project. This information will be derived from the estimation of some metrics related to project size, effort, duration, cost, and quality.

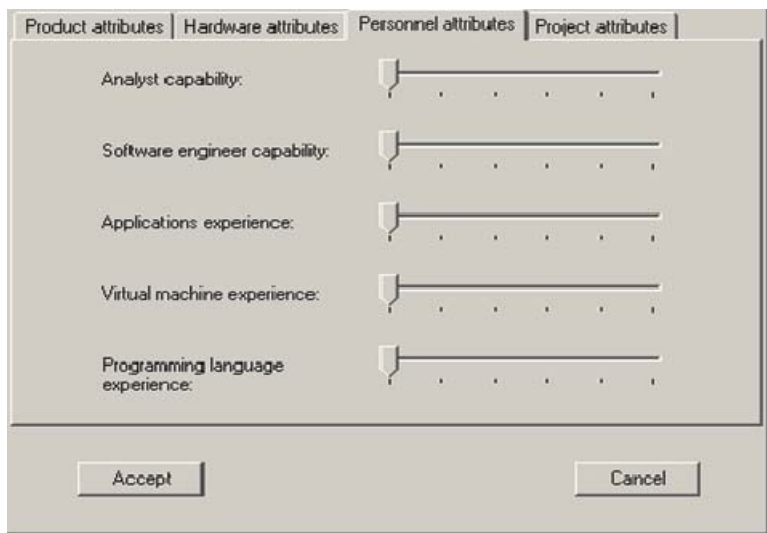


Figure 26: Project Categorization

2. Identification of estimated project activities

The project manager will use the organization's previous experience to determine the activities and practices to use during the project. The activities specification could be

manual, but it is not recommended because the plan may not comply with the quality criteria of the organization.

When the project managers begin to define the new project activities, they will search in the PM-CAKE repository, introducing values for the search criteria (project type, business domain, team characteristics, development environment, technical factors, etc.). The system will perform the search and provide a list of results that are composed of an organization's project patterns and previous projects. Each result will be qualified by its degree of coincidence.

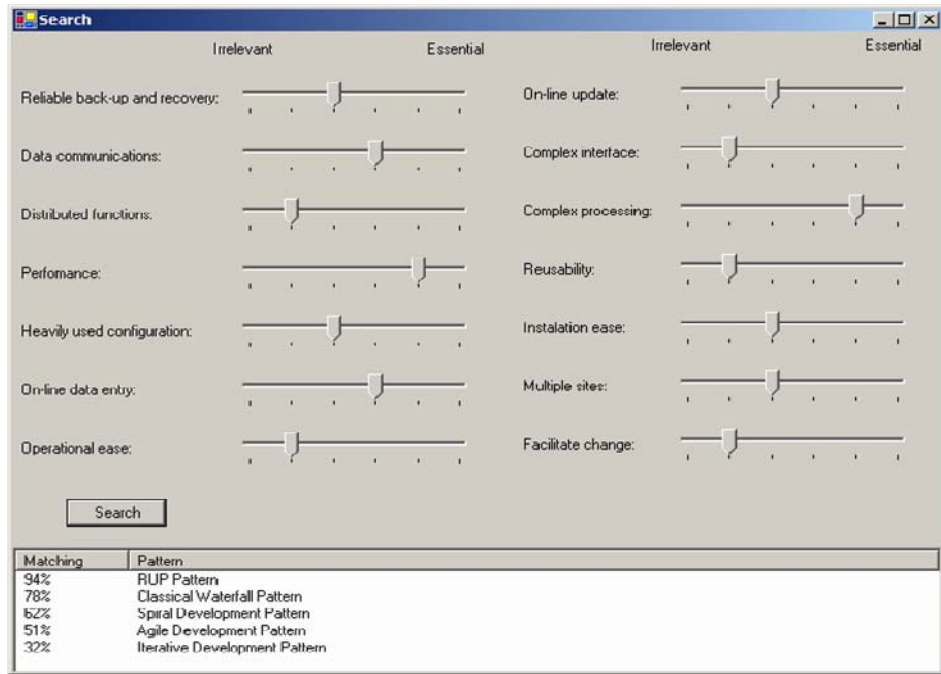


Figure 27: Project Pattern Search Criteria

To define the work breakdown structure, the project manager will be able to copy (all or some activities) patterns and paste them into the new project. The results from previous projects will not be available to be copied because all the project plans should have a pattern as a base. The reason for this restriction resides in the need to assure that all projects comply with the organization's quality criteria. If the project will be based in other projects, some inefficient practices and absences will be transmitted between them.

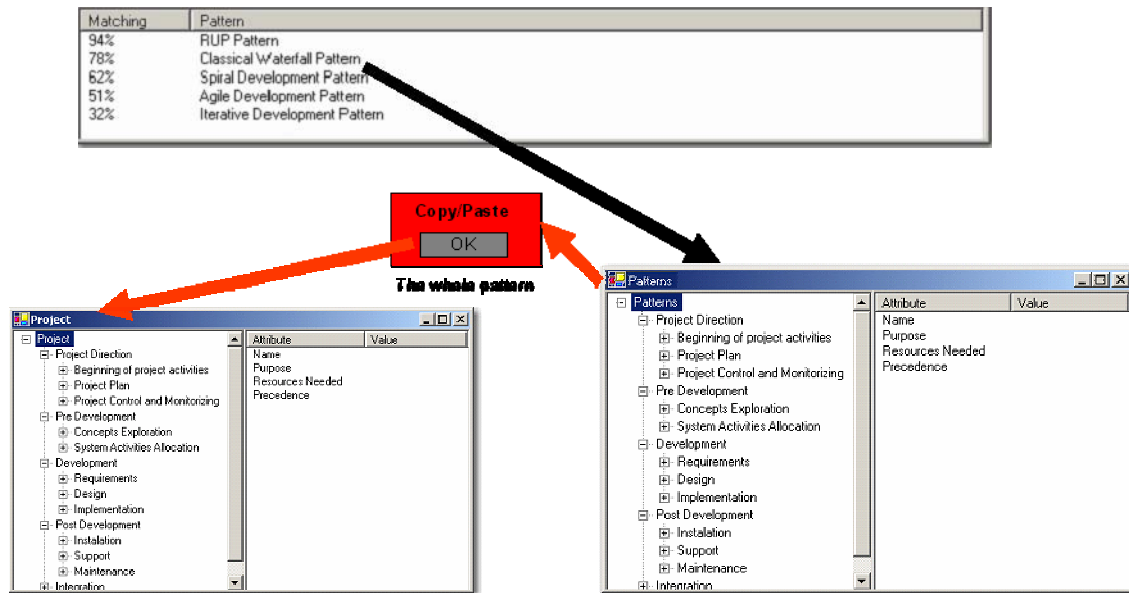


Figure 28: Project Pattern Instantiation

3. Identification of estimated project products

When the project WBS is validated, the project manager should create the product breakdown structure. The base of this structure is determined by the project pattern copied. This information should be completed by the project manager by attaching new templates or examples of the product coming from previous projects. The project manager will be able to specify the product type, the activity or activities related and the characteristics of the project in which it is elaborated.

Finally, the project manager will be able to export the project plan to another project management tool (e.g., MS-Project) to complete and tune the plan.

Postmortem Review

Once the project execution is finished, the project manager should perform the postmortem review and the classification of its information in a knowledge base. This procedure is described in the following four points:

1. Selection of project files

The project manager will determine the files with management information and those with technical (requirements, design, tests, source code, etc.) artifacts.

2. Project categorization

This step will allow the qualitative and quantitative specification of the project. The qualitative specification consists of the selection of real values related to project typology: types of functionality, lifecycle, processes considered complexity factors, project team characteristics, and development environments. Afterward, the project manager should be provided with the quantitative assessment of the project. This information will be offered through the specification of real values for the metrics presented by the tool. The base metrics package will be composed of project size, effort,

duration, cost, and quality. So, the project manager will be responsible for value accounting. It is only necessary to fill the metrics that are useful for the organization and not all of the metrics proposed.

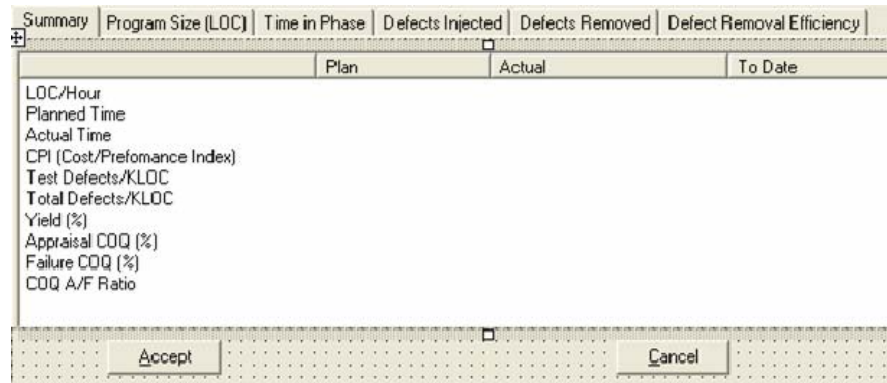


Figure 29: Quantitative Postmortem Analysis

3. Identification of actual project activities

Once the general information of the project is gathered, PM-CAKE will analyze, automatically, the information of the final project plan file. The result will be a hierarchical list of the project activity. The WBS information will be complemented with specialized information for each activity: name, purpose, tasks, effort, duration, resources, and precedence. This information should be validated by the project manager, who will be enabled to update some information if it is considered necessary.

4. Identification of actual project products

When the project WBS is validated, the project manager should validate the product breakdown structure that will be proposed by PM-CAKE systems through the analysis of the project plan and the technical products files. The information provided will be shown using a hierarchical list of activities and, for each activity, there will be three sub-lists with the artifacts used, created, and modified during the activity execution. The sub-lists also will provide a hyperlink to the file with the content of the product.

Statistical Analysis

When a project is going to be qualitatively characterized, PM-CAKE offers a set of measures that could be used objectively to qualify the project performances. This set of measures is structured according to the GQM (Goal-Question-Measuring) philosophy defined by Basili and includes size, cost, effort, schedule, and quality. The set of measures, according to business objectives, supplies questions and measures to be used at the beginning of the project, and then throughout the project they supply data for goal fulfillment. PM-CAKE could export data to a spreadsheet to obtain graphics when necessary. Lastly, when measurement data exceeds an established threshold, PM-CAKE supplies a warning.

Biographies

Antonio de Amescua Seco

Computer Science Department, Carlos III University of Madrid

amescua@inf.uc3m.es

Antonio de Amescua has been a full professor of the Computer Science Department at the Carlos III Technical University of Madrid since 1991. Before that he was a researcher at the Polytechnic University of Madrid from 1983 to 1991. de Amescua holds a BS in computer science and a PhD in computer science from the Polytechnic University of Madrid. Also, he has done software engineering work in a public company (Iberia Airlines) and in a private company (Novotec Consultores) as a software engineering consultant.

de Amescua has been researching new software engineering methods and has shared his results in papers and at conferences. He was the research project leader for the development of the Information System Development Methodology for the Spanish Administration and participated in other projects sponsored by the European Union.

Gonzalo Cuevas Agustín

Faculty of Computer Science, Polytechnic University of Madrid, Spain

gcuevas@fi.upm.es

Gonzalo Cuevas Agustín has a PhD in Telecommunications and an MS in computer science. He is a full professor in the Computer Science School of Polytechnic University of Madrid, Spain. He has occupied senior management positions during more than 35 years of university and industry experience. Cuevas Agustín is a founder and director of the Master of Software Engineering at Polytechnic University of Madrid in association with Carnegie Mellon University. He is a reviewer of the European Commission, consultant in software process improvement, IPRC member, senior member of IEEE, and member of ACM. He has written numerous publications and books on software engineering and has been a speaker at conferences, seminars, and symposia.

Javier García Guzmán

Computer Science Department, Carlos III University of Madrid

jgarcia@inf.uc3m.es

Javier García received an engineering degree and PhD in computer science from the Carlos III University of Madrid. He has seven years of experience as a software engineer and consultant in public and private companies. He has participated in numerous research projects financed with public (European and national) and private funds. These projects were related to software process improvement and its integration with organizational business processes. He has published books and international scientific papers related to software engineering and collaborative working environments. His current research interest is formal measurement of processes improvement, ISO 15504 assessments, software capacity rapid audits, evaluations for pre-diagnosis according to ISO 15504 and CMMI, and management of knowledge related to software engineering.

Juan Llorens
Computer Science Department, Carlos III University of Madrid

Paloma Martínez
Computer Science Department, Carlos III University of Madrid

Diego Martín
Computer Science Department, Carlos III University of Madrid

4.4 Value-Centric Process Improvement for Small Organizations by Using QFD and CMMI

Author

Zaijun Hu

Abstract

Small organizations are subject to the changing business environment, including their organizations, their customers, and their market. They are usually short- or middle-term oriented. Their present value changes from time to time and from project to project. Different organizations also have different needs, constraints, and expectations. To capture, evaluate, and monitor the customer requirements correctly and efficiently and to transform them to the concrete technical implementations consistently, we propose a QFD (Quality Function Deployment) and CMMI-based process improvement tool that is integrated into the IDEAL model. The tool makes full use of the QFD and CMMI features. The voice of the customer, the house of quality, the practice evaluation matrix, the implementation evaluation matrix, and CMMI process areas are the key elements of the defined tool.

Introduction

ABB is a leading power and automation company with strong market positions in its core business areas—the power and automation technology. It has an annual revenue of over 20 billion USD. ABB has a number of business units and subsidiaries located throughout the world and has established a distributed development culture where joint development of solutions is often required and many cooperating partners from different ABB business units participate the development project.

ABB business units are in many cases relatively small, ranging from a dozen to a few hundred people. Each organization can have some organization-specific development process components, such as its own guidelines, standards, and knowledge databases, and is subject to its surrounding country's culture. These organizations can also have their own business domains (manufacturing, power generation, and chemical industry) with different development tasks (software, hardware, and firmware) and project types (system engineering and research and development). All of these factors create the context that has a big impact on the value shaping of an organization; a few examples of the value spectrum include quality, time-to-market, performance, efficiency, and conformance to standards (such as IEC 61508, MISRA).

The diverse value understandings and the continuous change of the business context create one of the difficult challenges for process improvement in these small organizations. First, no common value definition can apply for all small organizations. For each process improvement initiative the value interpretation of a small organization has to be analyzed to

determine what is the essential impact on the process improvement activities. A proper analysis model or approach is needed to understand the voice of customers correctly and identify the most important issues for process improvement. Second, small organizations adapt quickly to their environments. That means the environmental changes also lead to the change in the value interpretation or weighting. Continuously monitoring the value composition of an organization during process improvement is extremely critical for deciding if the improvement plan needs to be changed. In addition, for the distributed or collaborative development where many project stakeholders are involved, it is also very important to identify who provides key voices in the value definition. The change in the personnel also leads to the change in the value weighting. Answering the following questions would help address the challenges:

- how to capture, analyze, and manage the requirements of customers
- how to associate the planned work to customers' requirements
- how to monitor the change in customers' requirements

Current Process Improvement Activities

For the purpose of process improvement, ABB has established a team called ASPI (ABB Software Process Initiative) that helps ABB business units to identify the process improvement potential and suggest the appropriate approaches for improving the process. The main activities of ASPI include

- internal CMMI-based appraisal for ABB business units
- organization of workshop for the creation of process improvement plan
- continuous coaching of improvement activities

Current Process Improvement Model

ASPI uses a process improvement model based on IDEAL. The IDEAL improvement model consists of four phases:

Initializing

In this phase the focus is put on the understanding of business goals and objectives. Identifying and analyzing drivers for change, analyzing the impact and related relationships, obtaining commitment, and setting up charter infrastructure to ensure enough resources are the major activities of this phase.

Diagnosing

During the diagnosing phase two characterizations of the organization are developed: the current state of the organization and the desired future state.

ASPI uses CMMI as reference model for the diagnosing phase. The process areas defined in CMMI are selected for the appraisal activities.

Establishing

The purpose of the phase is to deliver a detailed work plan for process improvement. The main activities include: setting priority, developing approaching, and planning action.

Acting

The purpose of the phase is to implement the work that is conceptualized and planned in the previous three phases. The main activities include creating the solution, piloting/testing the solution, refining the solution, and implementing the solution.

Learning

In the learning phase, the entire IDEAL experience is reviewed to determine what was accomplished, whether the effort accomplished the intended goals, and how the organization can implement change more effectively and/or efficiently in the future.

Challenges for the Current Model

Using IDEAL as basis for process improvement and CMMI as reference model for appraisal of the development process is a good way to implement improvement and to capture the current state and define the desired state. However, the IDEAL model doesn't provide proper approaches for how to capture, analyze, and manage the customers' requirements. It doesn't tell how to associate the conceptualized and planned actions to the customers' requirements, and doesn't define a way to continuously monitor how well the customers' requirements are satisfied. This creates the motivation to extend the current process improvement model by using Quality Function Deployment (QFD) methodology.

Combination of QFD and CMMI

QFD, developed by Akao et al., is a structured approach for defining customer needs or requirements and translating them into specific plans to produce solutions to meet those needs. The voice of the customer and the house of quality matrices are two key elements of QFD. The voice of the customer is the term used to describe the stated and unstated customer needs or requirements while the house of quality matrices are used to translate higher level "what's" or needs into lower level "how's" - technical characteristics to satisfy these needs.

For the CMMI-based process improvement combined with QFD, the following steps are defined:

1. Identify the stakeholders or sponsors for process improvement and assign a weighting factor of a scale (e.g., 1-3) to each stakeholder. The stakeholders having more influence on the process improvement activities get greater weighting factor (in this paper, 1: weak, 2: medium; 3: strong).
2. Capture the customer requirements for process improvement.
3. Prioritize the customer requirements on the scale from 0 to 5 (0: not necessary; 1: nice to have; 3: important; 5: very important; 9: critical).

4. Create the voices of customer matrix and calculate the normalized overall weighting, e.g., for the requirement complexity, the normalized overall weighting = $(3 \times 3 + 3 \times 3 + 3 \times 2 + 3 \times 1) / (3 + 3 + 2 + 1) = 3$. The normalized overall weighting will be used as the customer priority in the house of quality matrix later.

Table 16 shows an example of the voice of the customer matrix. The normalized overall weighting is used to measure how strong the voice of the customer is and later for the priority calculation in the house of quality matrix.

Table 16: Voice of Customer Matrix

Stakeholder weighting	3	3	2	1	
Req. \ Stakeholder	A	B	C	D	Normalized Overall Weighting
Complexity	3	3	3	3	3
Fewer Defects	9	9	9	9	9
Better communication	3	3	3	3	3
Reusability	3	3	5	5	3.7
Common Understanding	5	3	3	5	3.8

5. After the voice of customer matrix is created, it is possible to build the house of quality. Typical house of quality is a matrix consisting of Whats, Hows, correlation of Whats vs. Hows, How Muches, and Customer Priority.

Whats—this is a list of the customer requirements that are to be achieved.

Hows—this is a list of technical implementations (also called product requirements, technical requirements/characteristics/features, technical specification, technical solutions) that will help to achieve the customer requirements. For process improvement based on the CMMI (used by ASPI currently), it is quite practical and effective way to structure the technical implementations or solutions in the CMMI process areas. Table 17 shows five selected process areas (RD, TS, VER, VAL, PMC) as an example to address the customer requirements. In our method the design of the technical implementations means selection, configuration, and use of CMMI process areas. The defined structure in CMMI process areas with their specific, generic goals, and practices simplifies the design of the more detailed technical implementations on the low level.

Correlation of Whats vs. Hows—this is a relationship matrix that correlates the customer requirements and the technical implementations. The correlation will be weighed to measure

how well the technical implementations can meet the customer requirements. A scale (e.g., 9-5-3-1, Very High-High-Medium-Low) can be used for the rating.

Customer Priority—this shows how important a requirement is for the customer

How Muches—this covers the technical priority of each technical implementation, the target that is set to be achieved, the customer effort and supplier/consultant effort (man days). The customer effort is important for the customer for taking easibility into account. The technical priority can be calculated by summing all products of correlation weighting with the customer priority, e.g., for RD in the Table 17 the technical priority = $5 \times 5 + 3 \times 9 + 1 \times 3 + 3 \times 3.7 + 5 \times 3.8 = 75.1$. From Table 17, we can see that VER and VAL are most important in the all selected process areas. We use CMMI capability levels for the target setting. 100 stands for capability level 5. In Table 17, we use 60 for capability level 3.

QFD sometimes also requires the analysis of relationship between technical implementations. In our method, CMMI process areas provides the technical implementations on the high level. Due to the orthogonality of the CMMI process areas such an analysis may not be necessarily needed.

Table 17: House of Quality

PA \ Req.	RD	TS	VER	VAL	PMC	CP
Complexity	5	5	3	3	1	3
Fewer Defects	3	3	9	9	1	9
Better Communication	1	1	1	1	5	3
Reusability	3	5	3	3	1	3.7
Common Understanding	5	5	1	1	1	3.8
Technical Priority	75.1	82.5	107.9	107.9	34.5	
Target	60	60	60	60	60	
Custom. Effort	3	10	8	4	2	
Supplier Effort	18	30	18	9	6	
RD: Requirement Development; TS: Technical Solution; VER: Verification; VAL: Validation; PMC: Project Monitoring and Control; CP: Customer Priority						

6. After the house of quality is created, the next step is to perform more detailed design of technical implementations. In our context, the detailed design means selection and configuration of the specific and generic practices of the selected process areas, including subpractices. To make the selection or configuration easier, the so-called practice evaluation matrix is used, and it is built on the same principle as the house of quality, as shown in Table 18. For the example of requirement development (RD), we list a few specific and generic practices on the table for demonstration.

Table 18: Practice Evaluation Matrix

Req. \ PA	SP2.3.1	SP3.3.1	GP2.3	CP
Complexity	5	1	1	5
Fewer Defects	5	9	9	9
Better Communication	1	5	1	3
Reusability	5	1	1	3.7
Common Understanding	5	1	1	3.8
Tech. Priority	100.5	106.5	94.5	
SP 2.3.1 Identify Interface Requirement, SP 3.3.1 Validate Requirement GP 2.3 Provide Resource, CP: Customer Priority				

7. The next step is to create the so-called action evaluation matrix that contains the concrete actions to implement the selected practices (and subpractices). The weighing of correlation between the implementation approach and the process areas is performed in the same way as it is for the house of quality. Table 19 is an example of the matrix.

Table 19: Action Evaluation Matrix

PA \ Part of PA	AD	MD	TLS	T	C	I
RD	9	9	5	5	5	9
TU	9	9	9	9	5	9
VER	9	9	9	5	3	9
VAL	1	3	3	1	1	9

PMC	1	3	3	1	1	9
AD: Activity Definition; MD: Method Definition; TLS: Tool Suggestion; T: Training; C: Coaching; I: Implementation						

8. Monitor the change in customer requirements and adapt the matrices to reflect the new value of the customer if necessary.

Relation to IDEAL

In our QFD and CMMI-based process improvement we still follow the IDEAL model. Table 20 demonstrates the relationship.

Table 20: IDEAL Model with QFD

IDEAL	Initialize	Diagnose	Establish	Act	Learn
QFD	Capture & evaluate stakeholder	Capture and evaluate voice of customer	Create house of quality Evaluate practices Develop impl.	Perform & monitor impl. Control quality	

Strength, weakness and future

The proposed tool is easy to use to capture, analyze, and monitor the customer requirements and changes. It ensures the consistent technical implementations of them and, therefore, a good complement to the CMMI and IDEAL models. But misuse of the tool could also generate a lot of unnecessary matrices that need would so much effort to treat that the tool would become inapplicable. Another issue is the rating. The subjective estimate of a person can lead to unreasonable rating; therefore, teamwork is always required. From the research point of view, the combination of CMMI and QFD opens a new perspectives for the use of CMMI and at same time, provides new challenges such as how to integrate other development process models (V-model, RUP, etc.) and how to measure the execution of the practices defined in CMMI.

Conclusion

In this paper, we propose a tool for process improvement that can be used to capture, evaluate, and monitor the customer requirements and their implementations efficiently. The positive experience shows that this tool is easy to use and provides a good integration of CMMI, IDEAL and QFD so that the full use of each model's features is possible.

Bibliography

- [Akao 90] Akao, Yoji. *Quality Function Deployment: Integrating Customer Requirements into Product Design*. New York, NY: Productivity Press, 1990.
- [Chrissis 03] Chrissis, Mary Beth; Konrad, Mike; & Shrum, Sandy. *CMMI: Guidelines for Process Integration and Product Improvement*. Boston, MA: Addison-Wesley, 2003.
<http://www.sei.cmu.edu/cmmi/publications/cmmi-book.html>
- [Gremba 97] Gremba, Jennifer & Myers, Chuck. *The IDEAL Model: A Practical Guide for Improvement*.
<http://www.sei.cmu.edu/ideal/ideal.bridge.html> (1997).

Biography

Zaijun Hu

ABB Corporate Research

zaijun.hu@de.abb.com

As project leader and principal scientist at ABB Corporate Research in Germany, Zaijun Hu is currently leading the software process improvement projects based on CMMI® in the area of embedded system development. The projects cover different process areas such as project management, requirement engineering, technical development, verification and validation, and configuration management. Hu has published a number of papers in the areas of software architecture, XML, ontology, Internet, and software applications for the automation industry. After receiving BS and MS degrees at Tsinghua University (China) in process control and automation, he received a PhD at Stuttgart University in 1993. His current interest focuses on software process improvement based on CMMI, system development process including hardware and software, and the product development life cycle for embedded systems, development.

5 Regional Approaches

5.1 The Capability Maturity Model (SW and Integrated) Tailored in Small Indigenous Software Industries

Authors

Rosario Blowers and Ita Richardson

Abstract

The Irish Software Industry is undergoing rapid change due to increased competition from low cost global software service providers. Prior to this, Ireland had emerged as one of the leading low cost software exporters in the world. Then came the downturn in the global economy, the burst of the dot com bubble, and an increasing local cost base. Ireland now faces competition in the form of developing third world economies. The Irish software industry will struggle to compete with the vast workforce of cheap but skillful labor that these economies can offer in abundance. Can the Irish software industry compete in this changing environment? Software process improvement is recognized by the Irish government as a key differentiator in this competitive environment for the future. Quality improvement in Ireland had traditionally been the preserve of large software multi-nationals and the manufacturing industry. However, since the continued development of the local Irish software industry, this community is beginning to take software quality seriously. Research into the availability of software process models and best practices and how they can be effectively applied to small software industries in the Irish mid-west region is the main topic of this paper.

Research Environment (Small Organizations)

Research into software process in small to medium sized enterprises (SME) has grown within the University of Limerick over the past ten years. In 1996, there was one researcher, there are now eight people involved in SME process research at various levels, and this number continues to grow. As the economic environment within Ireland is supported by the presence of many SMEs, it is important that we focus this research within local industry. Our research to date has concentrated on the improvement of software processes within small companies, regardless of the model used. In some cases, companies are interested in implementing SW-CMM/CMMI, but due to market conditions, ISO9000-2000 is particularly important to the software industry in Ireland.

We collaborate with other research groups nationally and internationally, particularly with researchers in Finland and Wales, who face similar problems to ourselves. To develop and implement techniques, we endeavor to understand current processes and process improvements within SMEs and other companies. Using qualitative research methods, we interview, observe and analyze documentation within small companies to understand the conditions under which they work. Output is analyzed using, for example, content analysis. The next stage is to develop techniques and use action research methods to implement and evaluate what we have done. Supported by funding received from Science Foundation Ireland, our recent focus is researching how SMEs' software processes are operating within the global software development environment. We are developing collaboration with researchers from management to ensure that organization and change management are inherent in our output. This funding has also given us the opportunity to present our research to SMEs through various workshops and seminars.

Research Approach, Models, and Techniques for Process Improvement

The objective of CMMI version 1.1 was to provide a cleaner and more stable CMM. CMMI version 1.1 was released in January 2002. A significant number of organizations have committed to adoption of the CMMI.

However, the following questions still remain for organizations, especially small to medium organizations, that want to improve their processes:

- Which representation makes sense?
- What are the organization's business goals?
- What product/service does the organization develop/maintain?
- What is the product life cycle and development/maintenance organization?
- How much process improvement experience does the organization have? [Menezes 02]

The research project described in this paper is an investigation of how software process improvement (SPI), change management, and industry best practice can be applied in small software industries. The definition of SME, which is a term used in the Irish market, is companies that have less than 50 employees, have less than 3,000,000 (Ir)/4,800,000(euro) turnover, were founded in Ireland, have no parent company, and produce software products²⁰. The main focus area of this research is the SEI's software process improvement Capability Maturity Models (SW-CMM and CMMI) and investigation of other process models utilizing 9000-3 guidance for software (e.g., ISO9000, Tick IT). The IDEAL change model is investigated for implementation of SPI.

The following are the relevant research questions:

²⁰ Richardson, Ita. *Improving the Software Process in Small Indigenous Software Companies Using a Model Based on Quality Function Deployment*. University of Limerick, Ireland.

- Can the SW-CMM's and CMMI's tools and techniques be tailored for use in small indigenous software organizations?
- Can SPI be effectively achieved, utilizing the tailored CMMs and change management techniques, in this environment?

Two small companies were selected to perform an assessment that was to be tailored to best suit the organization. A literature review of SPI industry best practice was conducted, and the CMM was identified as the basis of an SPI program. We will quantitatively verify any process improvement through the application of the CMMI. We will use a triangulation strategy and will quantitatively verify any perceived improvements via

- CMM assessments tailored to the specific environment
- independent measurement utilizing questionnaire and interviews specifically focused on the business, organization, and customer perceived benefits (or lack of benefits) obtained from the improvement effort.
- questionnaire and interview feedback from authorized lead assessor networks based on their use of the latest CMM release, CMMI. A focus will be taken on their experiences in the small project environment across the software industry in a range of organization sizes.

The following topics detail the research phases.

Phase I–Company 1

The SW-CMM assessment process was tailored to suit the size of the organization and its business objectives. A tailored mentored self-assessment (MSA) was carried out against the SW-CMM. Its objectives were to identify software process improvements which were prioritized by the organization. The assessment process generated findings based on data gathered at the goal level for each key process area (KPA). Improvements identified and changes implemented over the next period were done based on this data, which was also used to validate assessment tailoring decisions made initially. The assessment process generated some global findings, which, at this stage, allowed some initial research conclusions to be drawn as answers to the questions posed. Phase I output acts as input to further assessment and validation in Phase II.

Phase II–Company 2

Prior to commencement of Phase II, the CMMI was published and consequently a tailored SCAMPI type C assessment was conducted with company 2. This assessment was focused on specific high priority process areas (PAs) identified in Phase I, which were aligned to CMMI (i.e., Requirements Management and Project Planning). A decision to apply the continuous model introduced by the CMMI was taken in Phase II based on the outcome of Phase I. Findings data at goal level will be quantitatively analyzed at KPA/PA goal level to contrast finding across the 2 organizations.

Further data will be obtained via independent business measures (e.g., customer surveys to support the validation process after the SPI programs have been completed in both companies). Feedback from lead assessors of the CMMI and SCAMPI assessment process will also be gathered as further input to research conclusions.

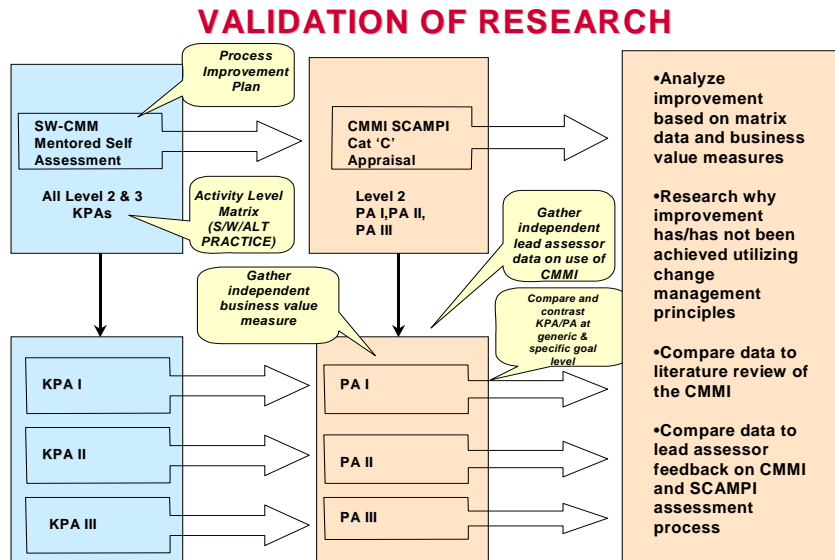


Figure 30: Research Validation

Strengths and Weaknesses of the Models, Techniques, and Approaches Used for Process Improvement

Phase I Validation

The validation phase of this research program has benefited from use of a formal assessment following a structured industry recognized standard regardless of the size of the organization. It may even be considered a major benefit, as small organizations do not have the resources to develop their own version of improvement programs. Significant improvement opportunities have been identified, as the organization and participants approved the tailored assessment final findings. Phase I was a success, but the opportunity existed to further tailor the assessment process and SPI plans as part of Phase II by applying the lessons learned in Phase I.

The closing assessment executive meeting with the Company 1 leadership identified the areas of Requirements Management and Project Planning as high priority areas for improvement in line with the organizations business goals. Further research into these areas formed the basis for Phase II of the research.

Organizations with less than 20 staff practitioners may have the greatest difficulty addressing specialized roles for process areas (e.g., Software Quality Assurance, Configuration Management, Measurement and Analysis). Key technical staff hold the major burden of these functions. Some sharing of specialized resources across small to medium enterprises may address these needs with distributed costs.

Evidence in the form of documentation is limited in a small environment, especially at the early stages of maturity, and knowledge of KPAs/PAs is necessary to ensure that valuable informal practices are not missed.

There is a considerable challenge to involve customers more actively in CMM and process improvement initiatives and changes. Small organizations may depend on one or two *big* customers and need to be perceived to be doing productive work all the time. CMM and the use of formal change methods (e.g., IDEAL) give a structured approach to building customer sponsorship in a competitive market.

The continuous representation and tailored assessments approach was seen to be more appropriate to small organizations. They can maximize use of the limited resources available for SPI activities by focusing on the PAs identified and prioritized by the organization. Use of tailored assessment processes (e.g., mentored self-assessments or SCAMPI type C assessments) give a good return on investment where budget is limited.

Phase II Validation

Company 2 has completed a tailored CMMI SCAMPI type C assessment using the continuous model on two high priority PAs, Requirements Management and Project Planning, identified in Phase I. Data has been collected via the assessment process in both these PAs at generic and specific goal level. The SPI program for Company 2 is in progress with several critical processes identified and under development. The next steps are to compare and analyze the data gathered via assessment against the corresponding KPAs in Company 1 to identify trends and support research conclusions.

The Most Important Topics for the Research Community to Address in the Future

Research into efficient tools and techniques, which give cost effective return on Investment, is a critical success factor in the small software industry. One of the most significant changes from SW-CMM to CMMI is the emphasis on measurement as a level 2 PA. However, this PA can still be overlooked if the continuous model is applied, especially in small organization where the resources required for an effective measurement program are sometimes perceived to be an overhead.

Process systems such as Six Sigma align with the quantitative process management, product quality management, and process optimization practices associated with levels 4 and 5. Research into how such systems could be used with the level 2 Measurement and Analysis

process area to address the requirements for levels 4 and 5 could lead to further improvement of the model.

Further data will be obtained via independent business measures (e.g., customer surveys to support the validation process after the SPI programs have been completed in both companies). Feedback from lead assessors on the CMMI and SCAMPI assessment process will also be gathered as further input to research conclusions.

References/Bibliography

- [Dunaway 96]** Dunaway, Donna & Masters, Steve. CMM-Based Appraisal for Internal Process Improvement (CBA IPI): Method Description (CMU/SEI-96-TR-007, ADA307934). Pittsburgh, Pa.: Software Engineering Institute, Carnegie Mellon University, 1996.
<http://www.sei.cmu.edu/publications/documents/96.reports/96.tr.007.html>
- [Dymond 98]** Dymond, Kenneth M. *A Guide to the CMM: Understanding the Capability Maturity Model for Software*. Wexford, PA: Process Transition International, 1998.
- [Florac 99]** Florac, William A. & Carleton, Anita D. *Measuring the Software Process Statistical Process Control for Software Process Improvement*. Boston, MA: Addison-Wesley, 1999.
- [Menezes 02]** Menezes, Winifred. "To CMMI or Not to CMMI: Issues to Think About." *CROSSTALK, The journal of Defense Software Engineering* 15, 2 (Feb. 2002).
- [Paulk 93a]** Paulk Mark C. et al. *Capability Maturity Model, Version 1.1*, (CMU/SEI-93-TR-24, ADA263403). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1993.
- [Paulk 93b]** Paulk, Mark C.; Weber, Charlie; Garcia, Suzanne; Chrissis, Mary Beth; & Bush, Marilyn. *Key Practices of the Capability Maturity Model, Version 1. 1* (CMU/SEI-93-TR-025, ADA263432). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, February 1993.
<http://www.sei.cmu.edu/publications/documents/93.reports/93.tr.025.html>
- [Paulk 95]** Paulk Mark C. et al. *The Capability Maturity Model Guidelines for Improving the Software Process*. Reading, MA: Addison-Wesley, 1995.
- [PMI]** *Project Management Institute Web site*.
<http://www.pmi.org/info/default.asp>

[Pressman 97]

Pressman, Roger S. *Software Engineering. A Practitioner's Approach*, 4th ed. New York, NY: McGraw-Hill, 1997.

Biographies

Rosario Blowers

**Computer Science and Information Systems Department, University of Limerick
0202819@STUDENT.ul.ie**

Rosario Blowers BSc (UL) has participated in postgraduate studies and research into software process improvement in the Irish industry for the past four years as part of the Irish Software Engineering Research Consortium (ISERC) at the University of Limerick, Ireland. She previously worked for 20 years in the multi-national software industry at Electronic Data System (EDS), EMEA, where she was largely responsible for strategic initiatives including CMM, ISO 9001, and TickIT, and was an auditor and assessor across a matrixed network of 12 software centres. Her experience in organizational change management and project and software life cycle methodologies has been gained in the Irish financial services sector.

Ita Richardson

Computer Science and Information Systems Department, University of Limerick

Dr. Ita Richardson, PhD(UL), MSc (UL), BSc (UL), Ceng C dipAF, CPIM, MBCS, is a senior lecturer in software quality at the University of Limerick, Ireland. She is the project leader of research into Global Software Development under the ISERC program and researcher on B4-STEP. These projects are funded by the Irish government via the Science Foundation of Ireland(SFI). Previously, she held a position as systems analyst with Wang Laboratories. Her research areas include the following: software process improvement and its implementation in industry, especially in small companies (less than 50 employees); global software development; and the use of quality function deployment (QFD) in the software industry.

5.2 A Software Process Improvement Solution for Small and Medium-Size Enterprises²¹

Authors

Jose A. Calvo-Manzano, Gonzalo Cuevas Agustín, Iván García Pacheco, Tomás San Feliu Gilabert, and Ariel Serrano

Abstract

The focus of this paper is to outline the main structure of an alternative software process improvement method for small- and medium-size enterprises. This method is based on the action package concept, which helps to institutionalize the effective practices with affordable implementation costs. This paper also presents the results and lessons learned when this method was applied to three enterprises in the requirements engineering domain.

Introduction

During the last few years, several software process improvement (SPI) models have been developed to increase the quality and productivity of software. Models, like IDEAL [McFeeley 96] or ISO/IEC 15504 [ISO/IEC 05], have been useful to initiate a software process improvement effort in many organizations. However, such models are oriented to large enterprises and their implementation frequently implies a high cost that is not affordable to small and medium-size enterprises [Cuevas 98].

This paper presents an alternative SPI solution for small and medium-size enterprises. This solution is called MESOPYME and is based on the Action Package concept [Fowler 97]. The experimentation of this package has been carried out on the Requirements Engineering domain and specifically on the process areas of Requirements Management and Requirements Development. These process areas have been selected because our investigations have found that most of the Spanish enterprises focus their improvement priorities on these areas.

The MESOPYME Method

MESOPYME has been developed as a SPI method that is focused on small and medium-size enterprises. Its structure has been divided in two parts. The first part is focused on process assessment and is based on a two-staged questionnaire that is a tool to determine the current state of the process [Cuevas 04]. The second part of the MESOPYME structure is focused on process improvement and uses the Action Package concept to establish and maintain a new

²¹ This work is sponsored by Endesa and Fundación DMR Consulting through the “Cátedra de Mejora de Procesos de Software aplicados a los Sistemas de Información en el Espacio Iberoamericano.”

process [Calvo-Manzano 00]. This paper only addresses the second part of the MESOPYME method and presents the Action Package main structure. Also, this paper presents the results obtained from method experimentation on the Requirements Engineering domain. This experimentation was carried out in three enterprises.

Action Package

The Action Package is structured as a set of components (organizational, management, and technical) that help get a solution for a specific software process domain. The Action Package is the mechanism that assists faster and affordable SPI program implementation for small and medium-size enterprises (SMEs). The main architecture of the MESOPYME Action Package is shown in Figure 31.

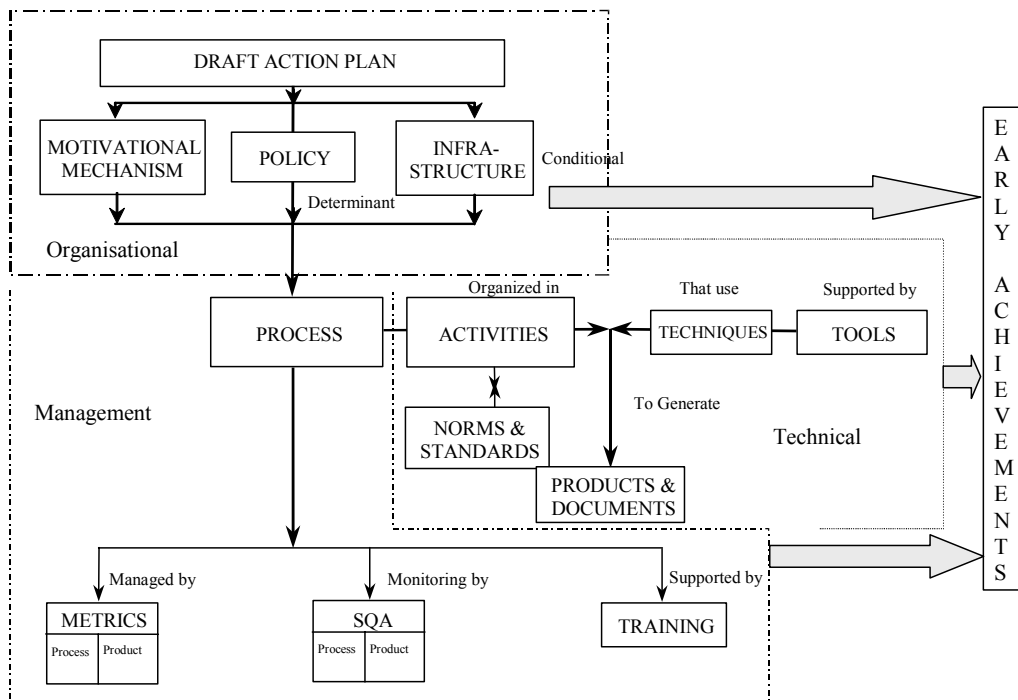


Figure 31: Action Package Architecture

- **Draft Action Plan:** This component contains a generic plan for the improvement project. The structure of the improvement plan helps to define with precision the context for the improvement action, its objectives, scope, and the specific tasks. The generic activities of the improvement plan include the following:

1. define organizational structure
2. establish the improvement context, with initial training in process, teamwork, and change management; conduct global and technical training in the action package
3. define a short-term action plan
4. review and adapt the action package

5. select the pilot project
 6. conduct training to implement the techniques
 7. run and collect measures
 8. assess the pilot of the new processes
 9. refine processes by having the pilot results in mind
- **Motivational Mechanism:** This component is used to get commitments from relevant stakeholders and to achieve process institutionalization. Human and organizational aspects are considered.
 - **Policy:** Principles that drive the improvement strategy of the enterprise are stated explicitly. The package describes a guide of the policy for each process area. Also, some general rules, policy content, document format, practical cases, and measures to support the policy are provided.
 - **Infrastructure:** For each process included at the package, an organizational infrastructure must be implemented. The roles and the responsibilities needed are established. A software process improvement structure based on the roles that people have to perform is proposed, independently of their position in the organizational structure.
 - **Process:** For each process area included, the process activities are established. These activities are represented in a graphical way and are complemented by a detailed description. Any additional artifact (documents, templates, techniques and tools) are described in a matrix relationship with, if needed, the tasks involved in their execution, documents and/or the inputs used and the outputs generated, participant roles for each activity, and techniques used to carry out each stage of the process.

Examples of graphical representation corresponding to Requirements Engineering domain are shown in Figure 32 and Figure 33.

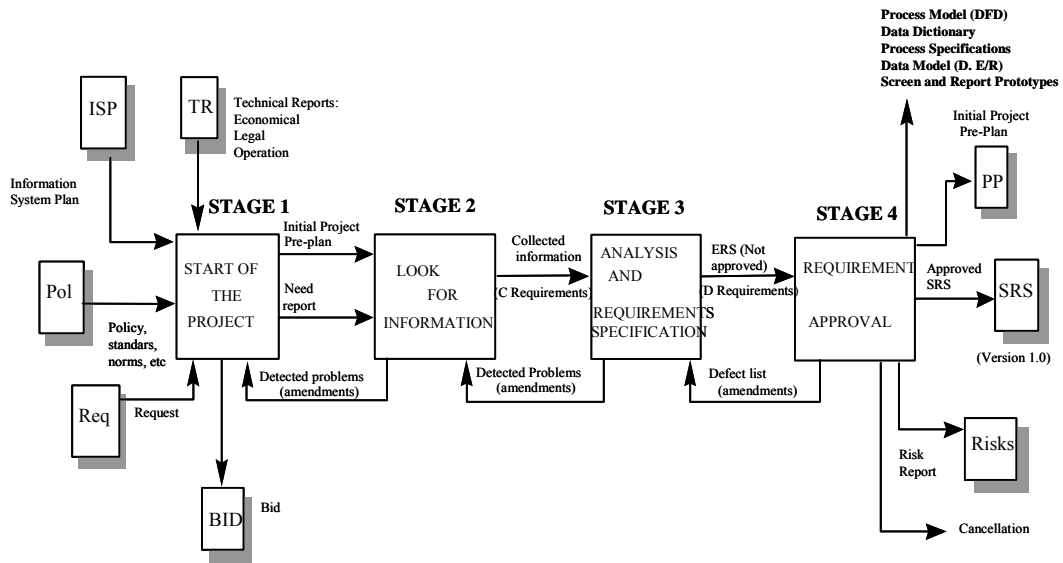


Figure 32: Requirements Development Process

In the case of the Requirements Development process area four stages have been defined. In each stage, activities involved, input documents and products, roles involved, organizational structures, output documents, and products and techniques to be used are specified.

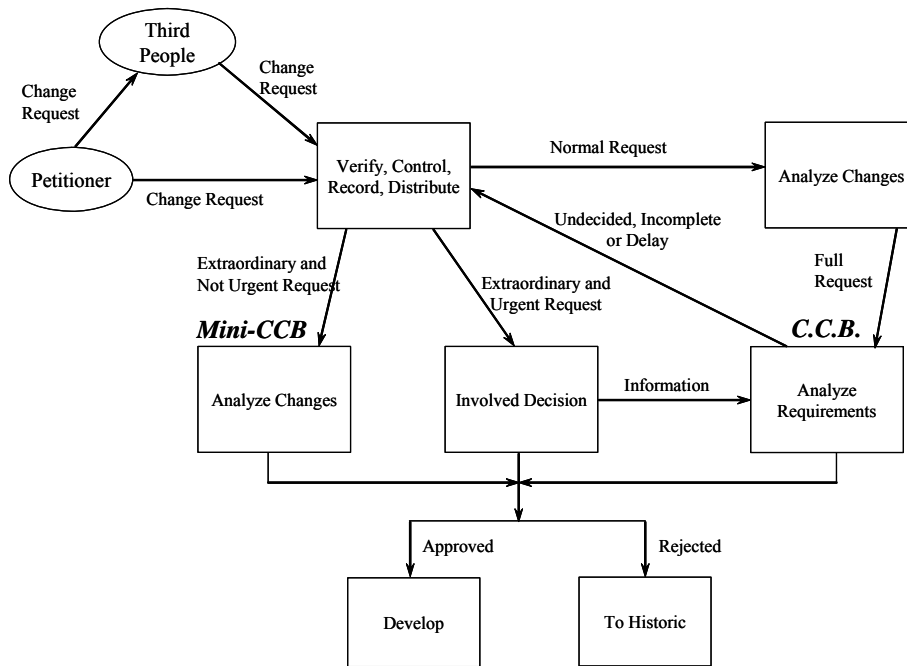


Figure 33: Requirements Management Process

In the case of the Requirements Engineering domain, roles defined are functional analyst, technical analyst, customer, change control board (CCB), project leader, quality expert, operation expert, methods and processes expert, system expert, and user.

- Products and documents:** Global document architecture is defined to describe the detailed products formats, which will be obtained when applying the above processes. It is important to highlight that this architecture has common parts, such as general management documents for all action packages. The detailed designed formats include an agenda model that will be used in the meetings call and a change requests template that will be used to register and treat the change requests. An example of the Requirements Engineering document structure is shown in Figure 34.

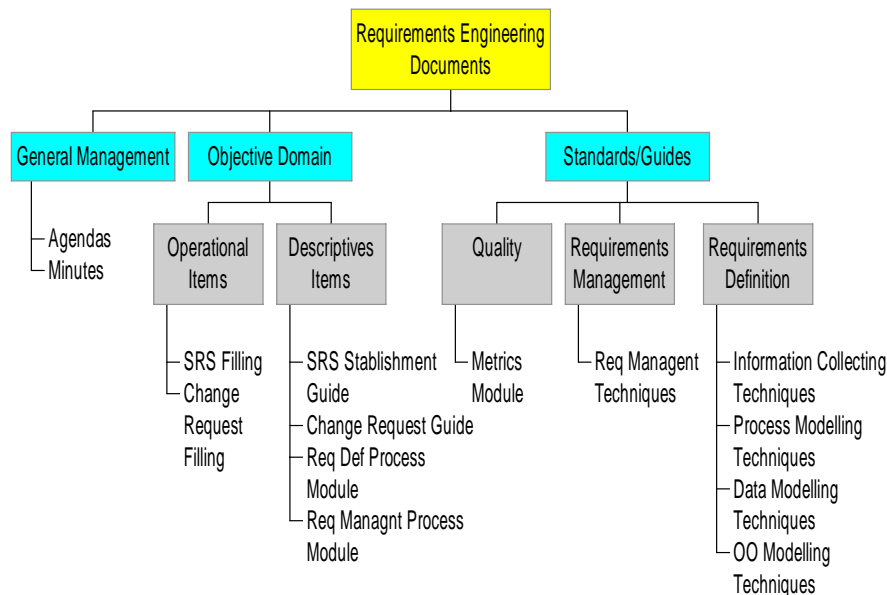


Figure 34: Structure of the Requirements Engineering Action Package “Products and Documents”

- Techniques:** The techniques to be applied in the activities described in the process are defined. For example, in the Requirements Development process area, techniques to extract information such as interviews and JAD (joint application development) are established. Moreover, techniques to model functional behaviors following a structured or object-oriented approach are presented. Finally, inspections techniques are also defined.
- Tools:** Due to continuous evolution and changes of the tools and the wide variety of the platforms that can be found in enterprises, the Action Package does not provide a tool list. However, the Action Package provides tool taxonomy to guide the selection of an adequate tool. In this way, when the process corresponding to an action package has been established, enterprises can select quickly and easily to get the more convenient tool for them.
- Metrics:** To begin a successful software process improvement program, it is important to quantify it, based on the attributes which help us to understand issues that affect quality, opportunity, cost, process implementation rate. All of these issues are key in the decision making process. The action package contains a set of basic and calculated metrics. The

metrics are selected according the business goals. There are tables that relate business goals with specific metrics. There are defined procedures to register, using data collect templates, and to analyze data of the measures.

- **Software Quality Assurance:** Reduced software quality assurance requirements to be used for processes and products are described. Guides in which objectives, organizational structure, and functions of the quality assurance structure applied to software development are described. Also, in order to assess objectively the process and standards, a set of metrics is defined.
- **Training:** Training consists of courses on SPI, team building, CMMI, and the specific software process area to be improved. Usually a general course on the process area will be provided, and the course will present an overview of the current technology and standards that are applicable to the process area. Also, specific courses to introduce the action package and courses related to the implementation of the techniques and specific tools selected will be defined.
- **Early Achievements:** There are improvement goals to be achieved in a short period of time from the beginning of the improvement project. Early achievements aim to obtain visible results for the whole organization, so they maintain the commitment of the organization to the improvement project. For example the early achievements defined in the enterprises where the requirements engineering action package have been tested include the following:
 - establish a requirement engineering policy
 - define roles and responsibilities
 - define roles and responsibilities of the Change Control Board
 - define a change request template
 - start the revision of the current requirement change process with users
 - define a checklist of potential impact areas, which can be analyzed with requirements
 - define a template to define requirement in new developments
- **Glossary:** A glossary is included to provide a common vocabulary to assure intelligibility and utility. The glossary contains the more common terms of the process improvement technology, as well as specific terms used.

The action package architecture helps to do the following:

- separate information into useful components for different purposes
- select the parts of the component that have interest
- carry out change management and improvements easily due to the established architecture in components
- provide access to required information in an easy way

Experimentation Results

The MESOPYME method has been applied on three SMEs on the Requirements Engineering domain [ESSI 97a, ESSI 97b, ESSI 97c]. Also, two external consulting companies carried out

the implementation of the method. Table 21 shows the industrial domains in which MESOPYME has been applied.

Table 21: Industrial Domains of the Enterprises Involved in MESOPYME Experimentation

Enterprise Sector →	IT	Production	Service	Others
Enterprise ↓				
E1				X
E2	X			
E3			X	

Table 22 shows the sizes of software development units and enterprises in which MESOPYME has been applied.

Table 22: Employees Distribution — Totals and Those Involved in Software Development Unit

Employees →	Enterprise				Software Development Unit			
Enterprises ↓	1-10	11-50	51-500	+500	1-5	6-20	21-50	+50
E1				10.000			45	
E2			257					83
E3		25				12		

The results obtained in these experiences have been promising and provide evidence that MESOPYME method could be adequate in small and medium enterprises. Also these results have demonstrated that MESOPYME method could accelerate the implementation of software processes improvement. The results show that the smaller the enterprise, the less effort to adapt the package, and the more profit made.

With the MESOPYME action package, the external help could be reduced to a half or third, according to previous experiences obtained without the use of MESOPYME. The experience confirms that a typical improvement life cycle is implemented around six months, with an average external effort of 2.5 month-people. The obtained results are shown in Table 23.

Table 23: MESOPYME's Obtained Results

Phase→	Total duration per phase (months)				Total Time (months)	Effort and cost (month-people y K\$ USD)			
	Method↓					Internal		External	
	Initial	Diagnostic	Establishing	Acting	Total months	M-P	K\$USD	M-P	K\$USD
MESOPYME	0.5	0.5	0.5	4.5	6	8.5	28.3	2.5	32.5

Conclusions and Lessons Learned

The lessons learned from the MESOPYME method experimentation on three enterprises include the following:

- First, the action plan that must be done by the working team. The final action plan is obtained with the customization of MESOPYME action plan.
- Second, the process definition procedures have examples and prototypes of process areas. The procedures helped reduce the time spent in defining the processes. The experiences in these enterprises have demonstrated that the procedures were useful and easy to adapt to the enterprise needs. Having the components in advance helped the enterprises focus their work and avoid spending time deciding to develop or not develop a procedure. They also helped the enterprises avoid spending a great deal of time that is produced in the initial phases of an improvement project due to the uncertainties associated with the new way of working.
- Third, the effort to understand the possible solutions proposed and its adequacy to the enterprise has been reduced. Managers, project leaders, and technicians have understood their activities and responsibilities easily by the use of action packages. With other methods, this activity could be done spending more time because the working team has to design a whole solution.
- And finally, MESOPYME quality assurance activities have helped to create a SQA group that was identified as a gap in these enterprises.

The principal contribution of MESOPYME method is that the modular architecture is a good solution to software process improvement in small enterprises.

The action package has a modular structure that defines different elements to be considered in the implementation. This structure is common to all process areas and its implementation, is accessible to SMEs, allowing a fast technology transfer. The results show that it is possible to implement a few process areas in five to seven months at an affordable cost.

The implementation of MESOPYME has achieved good results in the enterprises where was applied it and minimized their change resistance. It is important to highlight that

MESOPYME includes the coordination of three SPI relevant aspects: human, process, and technology. Finally, with the obtained results, it has been found that the Action Package might be a satisfactory software process improvement solution for SMEs at an affordable cost.

References

- [Calvo-Manzano 00]** Calvo-Manzano, Jose A. "Método de Mejora del Proceso de Desarrollo de Sistemas de Información en la Pequeña y Mediana Empresa," Doctoral Thesis. Universidad de Vigo, España, July 2000.
- [Cuevas 98]** Cuevas, Gonzalo & Gil, Miguel Ángel. "Mejora rápida en los procesos de desarrollo de software," *II Congreso Nacional de Ingeniería de Telecomunicación*. Madrid: June 1998.
- [Cuevas 04]** Cuevas, Gonzalo; Serrano, Alan; & Serrano, Ariel. "Assessment of the Requirement Management Process Using a Two-Stage Questionnaire," 110-116. *Proceedings of the Fourth International Conference on Quality Software (QSIC2004)*. Braunschweig, Germany, September 8-9, 2004.
- [ESSI 97a]** ESSI, Erosky. "SCOOP. SYMQUAD: System Requirements and Quality Assurance in a Distribution Company," ESSI Project: No. 10344, March 1997.
- [ESSI 97b]** ESSI, Lantik S.A. "LANHOBEEK: Client Relationship and Requirement Management Improvement," ESSI Project No. 24155, March 1997.
- [ESSI 97c]** ESSI, G.O.C. "HOSPUR: User Requirements and Reuse in Health Management Systems," ESSI Project: No. 24181, March 1997.
- [Fowler 97]** Fowler, Priscilla & Patrick, Mac. "Requirements Management Transition Packages." *Proceedings of the Introducing Requirements Management Into Organizations Workshop*. CMU/SEI Transition Enabling Program, May 1997.
- [ISO/IEC 05]** ISO/IEC 15504. "Information Technology — Process Assessment," International Organization for Standardization, 2005.

[McFeeley 96]

McFeeley, Robert. IDEAL: A User's Guide for Software Process Improvement (CMU/SEI-96-HB-001, ADA305472). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1996.
<http://www.sei.cmu.edu/publications/documents/96.reports/96.hb.001.html>

Biographies**Jose A. Calvo-Manzano Villalón**

Faculty of Computer Science, Polytechnic University of Madrid, Spain

jacalvo@fi.upm.es

Jose A. Calvo-Manzano Villalón has a PhD in computer science and is assistant professor at the Computer Science School of Polytechnic University of Madrid, Spain. He is teaching in the area of Software Engineering. He is an author of several international papers related to software engineering and has written books about software process improvement and software engineering. Calvo-Manzano Villalón also is a reviewer of a number of international journals and conferences, and he has participated in several European projects related to software risk management and software process improvement in requirements management.

Gonzalo Cuevas Agustín

Faculty of Computer Science, Polytechnic University of Madrid, Spain

gcuevas@fi.upm.es

Gonzalo Cuevas Agustín has a PhD in Telecommunications and an MS in computer science. He is a full professor in the Computer Science School of Polytechnic University of Madrid, Spain. He has occupied senior management positions during more than 35 years of university and industry experience. Cuevas Agustín is a founder and director of the Master of Software Engineering at Polytechnic University of Madrid in association with Carnegie Mellon University. He is a reviewer of the European Commission, consultant in software process improvement, IPRC member, senior member of IEEE, and member of ACM. He has written numerous publications and books on software engineering and has been a speaker at conferences, seminars, and symposia.

Iván A. García Pacheco

Faculty of Computer Science, Polytechnic University of Madrid, Spain

igarcia@zipi.fi.upm.es

Iván A. García Pacheco is a doctoral candidate in computer science at the Polytechnic University of Madrid, Spain. He is a co-author of several international papers related to software engineering and, more specifically, software process improvement. García Pacheco has participated in several institutional research projects in industry (IBM Global Services) and in academia. At present, he is part of the OPTIMA project to improve processes in a Spanish enterprise.

Tomás San Feliu Gilabert

Faculty of Computer Science, Polytechnic University of Madrid, Spain

tsanfe@fi.upm.es

Tomás San Feliu Gilabert has a PhD in computer science and is an assistant professor in the Computer Science School of Polytechnic University of Madrid, Spain. He teaches software engineering courses. San Feliu Gilabert is an author of several international papers related to software engineering and has written books about software process improvement and software engineering. He has participated in several European projects related to software risk management and software process improvement in requirements management.

Ariel E. Serrano Rico

Faculty of Computer Science, Polytechnic University of Madrid, Spain

aserrano@zipi.fi.upm.es

Ariel E. Serrano Rico has a BS in computer science from the Universidad Iberoamericana, México and was an assistant professor at the same university. He has occupied many management positions during his 15 years of industry experience. Serrano Rico has participated in several research projects in academia and the industrial domain. Currently, he is a doctorate candidate in the Computer Science School of Polytechnic University of Madrid, Spain.

5.3 RAMALA: A SPI Service Provider for SMEs

Authors

Antonio de Amescua Seco, Javier García Guzmán, Yaser Rimawi, and Gonzalo Cuevas Agustín

Abstract

RAMALA is the name of an SPI service providing infrastructure that permits SMEs to access specialized SPI services, which is provided by experts, in a cheaper way.

RAMALA knowledge base, supported by a software tool also called RAMALA, gathers the software engineering knowledge needed to deploy a software process improvement program within a software organization. RAMALA knowledge base contains a software process framework that is mainly based on the PMBOK process framework, is detailed by software engineering experts using the best practices of the main software reference models like CMMI and ISO 15504, and is enriched with process assets of the most outstanding software development methodologies.

RAMALA knowledge base fulfils three main functionalities in a software process improvement deployment program: process assessment, process definition, and process improvement tracking.

Description of the SPI Problem in Spanish SMEs

Software organizations are aware that having the best professionals is not everything for project success. Unless they understand the software processes of an organization, these professionals cannot be part of a productive and high quality software development project.

The SEI carried out a study in response to a demand for information on the results of software process improvement efforts. This study covered 13 organizations that represent a variety of maturity levels. The results showed that the average yearly cost of software process improvement was \$245,000 and the average number of years engaged in software process improvement was 3.5. This means that implementing a software process improvement program is very expensive, especially for SMEs.

In addition, most software organizations do not have the technology and the appropriate tools to interpret the reference models and to define their processes. The following are most important weaknesses of these tools:

- The software process evaluation tools are not connected to process definition tools.
- Current tools do not allow an organization's knowledge and classification into process assets.

- Current tools do not allow information gathering to carry out improvement plans.
- the gathering and classification of instances of the organization's assets, and their instantiation into project's assets

To add to this, the current reference models do not provide a detailed process definition, and the organization needs a great deal of resources and time to define, in an integrated way, the processes mentioned in the standards and the reference models.

The majority of SMEs cannot dedicate the resources and funds needed for SPI programs. Moreover, in many cases, they do not have personnel with the capabilities and skills required.

So, as some SMEs externalize their IT services to other companies, we propose the externalization of SPI services as a good strategy for initiating a software process improvement program in SMEs. In this sense, the SEPG team is composed of company personnel (1-5 people, depending on the SME size) that is supported by the services of the SPI provider. These services are supported by a Web-based software tool.

The main benefit consists of the reduction of the effort and the cost due to a SPI program. The following benefits are also important for SMEs:

- The experts' knowledge is stored in a repository that is also used to store the company's knowledge, merging them to provide efficient SPI results. Obviously, the access to the company's knowledge is restricted to other companies that are also users of the SPI services.
- The SPI provider services permit SMEs to share the logic and physical infrastructure necessary to support SPI, suppressing its costs of creation and maintenance.
- Finally, the SPI provider services permit SMEs access to SPI solutions customized to their business areas.

RAMALA Infrastructure for SPI Service Providing

RAMALA is the name of a SPI service providing infrastructure that permits SMEs access to specialized SPI services that are provided by experts in a cheaper way.

RAMALA knowledge base, supported by a software tool also called RAMALA, gathers the software engineering knowledge needed to deploy a software process improvement program within a software organization. RAMALA knowledge base contains a software process framework, which is mainly based on the PMBOK process framework, detailed by software engineering experts using the best practices of the main software reference models like CMMI and ISO 15504, and enriched with process assets of the most outstanding software development methodologies.

RAMALA knowledge base fulfills three main functionalities in a software process improvement deployment program: process assessment, process definition, and process improvement tracking.

RAMALA Knowledge Base

RAMALA knowledge base is the result of research work developed in the Computer Science Department at Carlos III University of Madrid. Its main scope and goal was to model and develop a software engineering knowledge base for software process improvement supported by a software tool that enables the definition, assessment, and improvement tracking of an organization's software processes.

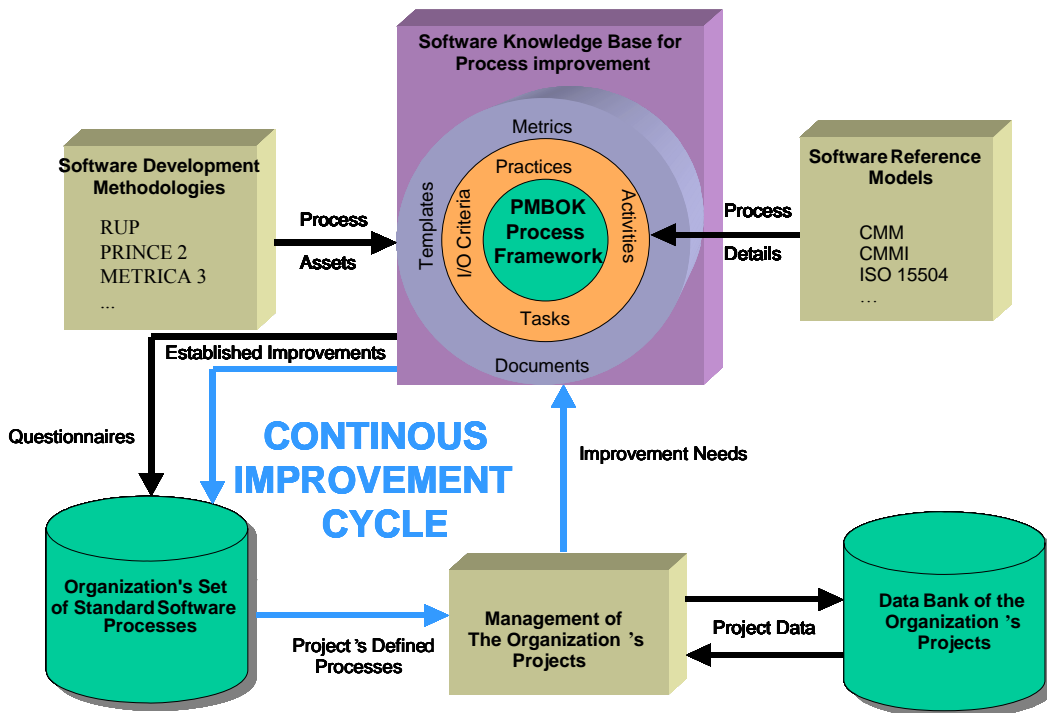


Figure 35: RAMALA Knowledge Base Structure

RAMALA knowledge base structure is shown in Figure 35. As we can see, the process definition functionality is covered by the software knowledge base for process improvement component, where the PMBOK Guide Process Framework is its core. Software engineering experts using the best practices of the software reference models and process assets of the most outstanding software development methodologies detail the process framework.

Implementing a formal assessment method valid for any software reference model covers the process assessment functionality. During the assessment, RAMALA gathers and classifies all process assets in the organization and links them to the related software process elements. Along with the assessment result, which is a color snapshot of the knowledge base, RAMALA provides the organization's set of standard software processes.

The improvement tracking functionality is covered by providing a mechanism to establish the project's defined processes, managing the project's process assets instances, and gathering measure data to verify the fulfillment of the improvements.

How to Use RAMALA for SPI Service Providing

The most important features in using RAMALA will be described in this section.

RAMALA software applies the Application Service Provider (ASP) concept that only requires software organizations to need an Internet browser and an Internet connection. Figure 36 shows RAMALA software architecture.

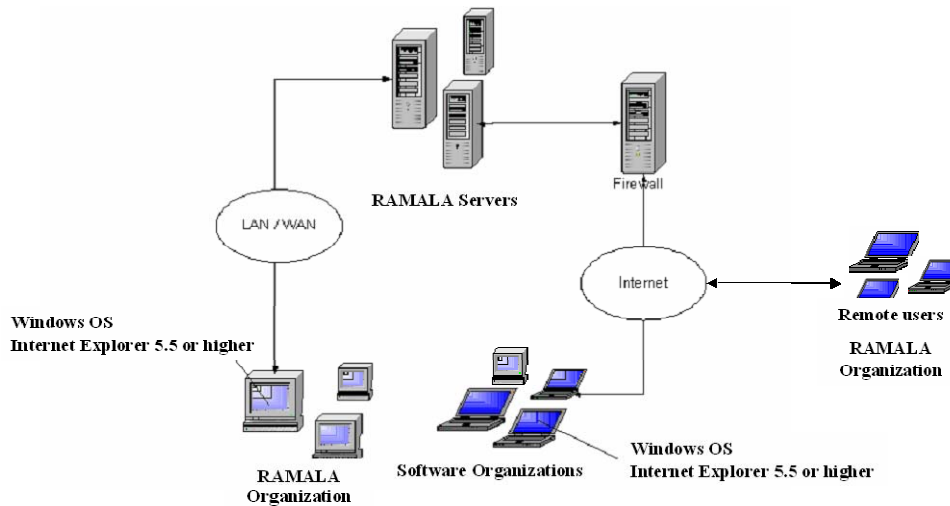


Figure 36: RAMALA Software Architecture

Software organizations, before signing on RAMALA, can take a tour within the knowledge base. Once, they sign on, the first thing that a software organization has to do is select a software reference model that it wants to follow. The RAMALA knowledge base has CMMI and ISO 15504 models stored already. Figure 37 shows elements of the CMMI as a selected software reference model.

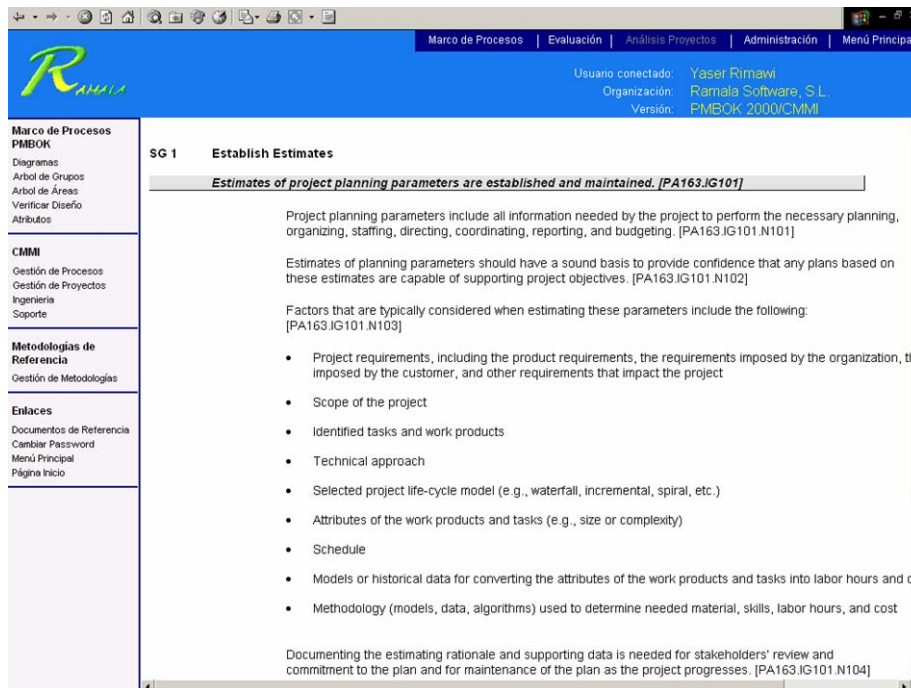


Figure 37: Software Reference Model Elements Stored in RAMALA

RAMALA, as described before, has stored, for each software reference model, a meta software process definition based on the PMBOK process framework. The relevant next step that the software organization should do is select a set of processes that it wants to assess. Figure 38 shows how processes are selected for assessment in RAMALA.

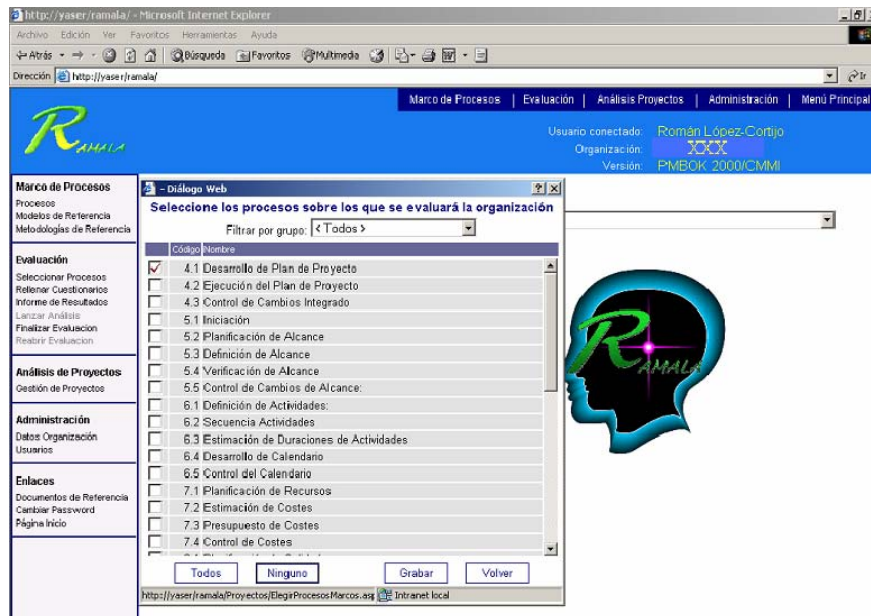


Figure 38: Selecting Processes from the PMBOK Process Framework for Assessment

In order to carry out the assessment, special members of the organization had to fulfill a detailed questionnaire for each process selected and its elements. During the assessment, direct evidences (organization's process assets) that indicate that the organization is satisfying the software reference model practices are collected, classified, associated to the corresponding software process elements, and stored within the organization's particular knowledge base in RAMALA. Figure 39 shows how the organization members had to complete a questionnaire for each process element, and how organization process assets are collected and associated to process elements.

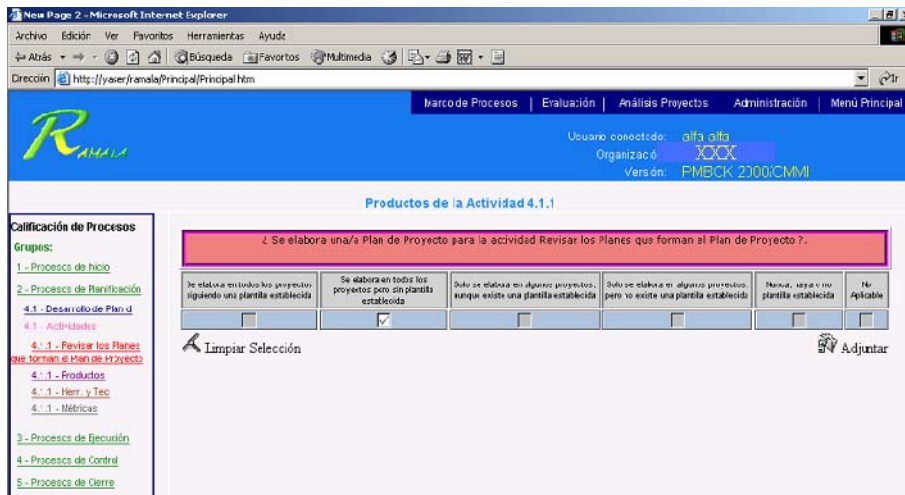


Figure 39: Process Element Assessment Questionnaire

Once the organization completed the questionnaires, an automatic algorithm is executed, which calculates the capacity of each process and its elements. Figure 40 shows a report with the process elements capacity.

Organización: XXX

Procesos			
Número	Nombre Proceso	Estado	Cobertura
4.1	Desarrollo de Plan de Proyecto		50,3472 %
Actividades			
Número	Nombre Actividad	Estado	Cobertura
4.1.1	Revisar los Planes que forman el Plan de Proyecto		62,5 %
Prácticas Genéricas del Proceso 4.1 - Desarrollo de Plan de Proyecto			
Número	Nombre Práctica Genérica	Estado	Cobertura
04.1.1	Realizar las Prácticas Base		81,25 %
04.2.1	Establecer una política Organizativa		75 %
04.2.2	Planificar el Proceso		75 %
04.2.3	Preparar Recursos		75 %
04.2.4	Asignar Responsabilidades		75 %
04.2.5	Formar a la Gente		75 %
04.2.6	Definir las Configuraciones		50 %
04.2.7	Identificar e involucrar a los Stakeholders Relevantes		43,75 %
04.2.8	Controlar el Proceso		37,5 %
04.2.9	Evaluar la Adherencia Objetivamente		37,5 %
04.2.10	Revisar el Estado con la Alta Dirección		43,75 %
04.3.1	Establecer un Proceso Definido		40,75 %
04.3.2	Colectar Información de Mejora		31,25 %
04.4.1	Establecer los Objetivos Cuantitativos para el Proceso		16,75 %
04.4.2	Estabilizar el Rendimiento de los Subprocesos		16,75 %
04.5.1	Asegurar la Mejora Continua del Proceso		37,5 %
04.5.2	Comprender las Causas Principales de los Problemas		25 %
Prácticas Específicas de la Actividad 4.1.1 - Revisar los Planes que forman el Plan de Proyecto			
Número	Nombre Práctica Específica	Estado	Cobertura
04.3.1.1	Revisar los Planes que forman el Proyecto		62,5 %
Productos de la Actividad 4.1.1 - Revisar los Planes que forman el Plan de Proyecto			
Número	Nombre del Producto (Atributo Salida)	Estado	Cobertura
0	Plan de Proyecto		41,00937 %
Herramientas y Técnicas de la Actividad 4.1.1 - Revisar los Planes que forman el Plan de Proyecto			
Número	Nombre de la Herramienta/Técnica	Estado	Cobertura
1	Metodología de planificación de proyectos		56,25 %
2	Conocimientos/habilidades de los stakeholders		62,5 %
3	Sistema de Información de Gestión de Proyecto (PMIS)		62,5 %

indica que el Proceso, Actividad, Práctica Genérica, Práctica Específica, Producto (Atributo Salida), Herramienta/Técnica o Métrica se ha conseguido.
 indica que el Proceso, Actividad, Práctica Genérica, Práctica Específica, Producto (Atributo Salida), Herramienta/Técnica o Métrica NO se ha conseguido en su totalidad.
 indica que el Proceso, Actividad, Práctica Genérica, Práctica Específica, Producto (Atributo Salida), Herramienta/Técnica o Métrica NO se ha conseguido.

Figure 40: The Organization's Software Process Elements Capacity

Along with the assessment results, the organization will obtain its own software engineering knowledge base where the definition of its set of standard software processes is stored as a color snapshot of the meta software process.

Later, the organization can manage its own knowledge base by adapting its process assets. RAMALA offers process assets of the most outstanding software development methodologies that the organization can use to adapt their own process assets. Figure 41 shows an organization's process description stored within its knowledge base.

Marco de procesos PMBOK del PMI			
Proceso 4.1 Desarrollo de Plan de Proyecto (Procesos de Planificación) (Cobertura: 50.34722%)			
Proceso(s) Anterior(es) <ul style="list-style-type: none"> 6.4 Desarrollo de Calendario 7.3 Presupuesto de Costes 	Propósito El proceso desarrollo de plan de proyecto utiliza salidas de otros procesos de planificación incluyendo la planificación estratégica para crear un documento coherente y consistente que puede ser utilizado como guía de la ejecución y control del proyecto. Este		Proceso(s) Sucesor(es) <ul style="list-style-type: none"> 4.2 Ejecución del Plan de Proyecto
Criterios de Entrada <ul style="list-style-type: none"> Terminar el proceso 6.4 Terminar el proceso 7.3 	Actividades <ul style="list-style-type: none"> 4.1.1 Revisar los Planes que forman el Plan de Proyecto 4.1.2 Desarrollo del Plan de Proyecto 	Prácticas Genéricas Todas las prácticas genéricas del área de procesos Planificación de Proyecto. <ul style="list-style-type: none"> GP 1.1 Realizar las Prácticas Básicas GP 2.1 Establecer una Política Organizativa GP 2.2 Planificar el Proceso 	Criterios de Salida <ul style="list-style-type: none"> Tener el Plan de Proyecto Aprobado aprobado por la Alta dirección
	Proceso(s) de Soporte <ul style="list-style-type: none"> 8.1 Planificación de Calidad 9.1 Planificación Organizativa 10.1 Planificación de 		
Entradas <ul style="list-style-type: none"> Otras salidas de planificación Información histórica Políticas organizativas 	Roles Clave <ul style="list-style-type: none"> Jefe de Proyecto Responsable de Gestión de Riesgos Responsable de 	Herramientas y Técnicas <ul style="list-style-type: none"> Metodología de planificación de proyectos Conocimientos y habilidades de los stakeholders 	Salidas <ul style="list-style-type: none"> Plan de Proyecto Detalles de soporte
Guías y Estándares <ul style="list-style-type: none"> Guía PMBOK Modelo de Referencia CMMI 		Notas	
Activos Comunes <ul style="list-style-type: none"> Plan de Proyecto 	Activos Propios <ul style="list-style-type: none"> plantilla Plan de Proyecto 	Métricas Comunes <ul style="list-style-type: none"> Tiempo de Elaboración de Plan 	Métricas Propias

Figure 41: An Organization's Standard Software Process

Once the organization implements a software process improvement plan based on the assessment results, RAMALA helps organizations assure the institutionalizing of the new processes by acting as a historical database of software process asset instances. An organization that uses RAMALA can do the following:

- create projects
- establish the project's defined processes for each project
- gather project results (process asset instances) and associate them to the corresponding project's defined process elements
- analyze project results
- determine the fulfillment degree of newly implemented processes

Biographies

Antonio de Amescua Seco

Computer Science Department, Carlos III University of Madrid

amescua@inf.uc3m.es

Antonio de Amescua has been a full professor of the Computer Science Department at the Carlos III Technical University of Madrid since 1991. Before that he was a researcher at the Polytechnic University of Madrid from 1983 to 1991. de Amescua holds a BS in computer science and a PhD in computer science from the Polytechnic University of Madrid. Also, he has done software engineering work in a public company (Iberia Airlines) and in a private company (Novotec Consultores) as a software engineering consultant.

de Amescua has been researching new software engineering methods and has shared his results in papers and at conferences. He was the research project leader for the development of the Information System Development Methodology for the Spanish Administration and participated in other projects sponsored by the European Union.

Javier García Guzmán

Computer Science Department, Carlos III University of Madrid

jgarciag@inf.uc3m.es

Javier García received an engineering degree and PhD in computer science from the Carlos III University of Madrid. He has seven years of experience as a software engineer and consultant in public and private companies. He has participated in numerous research projects financed with public (European and national) and private funds. These projects were related to software process improvement and its integration with organizational business processes. He has published books and international scientific papers related to software engineering and collaborative working environments. His current research interest is formal measurement of processes improvement, ISO 15504 assessments, software capacity rapid audits, evaluations for pre-diagnosis according to ISO 15504 and CMMI, and management of knowledge related to software engineering.

Yaser Rimawi

Computer Science Department, Carlos III University of Madrid

Gonzalo Cuevas Agustín

Faculty of Computer Science, Polytechnic University of Madrid, Spain

gcuevas@fi.upm.es

Gonzalo Cuevas Agustín has a PhD in Telecommunications and an MS in computer science. He is a full professor in the Computer Science School of Polytechnic University of Madrid, Spain. He has occupied senior management positions during more than 35 years of university and industry experience. Cuevas Agustín is a founder and director of the Master of Software Engineering at Polytechnic University of Madrid in association with Carnegie Mellon University. He is a reviewer of the European Commission, consultant in software process improvement, IPRC member, senior member of IEEE, and member of ACM. He has written numerous publications and books on software engineering and has been a speaker at conferences, seminars, and symposia.

5.4 The ACTI PROFO Process Improvement Initiatives

Authors

Karin Steembecker and Miroslav Pavlovic

Abstract

From our knowledge, we describe in this paper a unique SPI initiative which aims to implement two different improvement models, namely ISO 9000:2000 and CMMI level 2, at six small IT companies that must work together in order to obtain financial support for this project from the Chilean government. These subsidies, and the companies' sizes, introduce restrictions of time, resources and budget that have to be followed, as well as many challenges that we have had to address.

Introduction

The Chilean government is interested in attracting transnational IT companies to set up their regional platforms in their country. To that end, the government has offered financial aid to small companies for obtaining ISO 9000:2000 certifications. However, the small IT industry was keener on adopting CMMI. The result was the creation of an SPI initiative that combines both models in a pioneering approach that is now known as the ACTI²² PROFO projects.

In this paper, we focus on the strategies to implement CMMI level 2 at small, and very small, IT organizations. In particular, we explain how the definitions of process areas (PAs) are being managed in a specific project named PROFO-C. In the Project Description section, we describe the scope and goals of a general ACTI PROFO project. The methodology to carry out the dual application is explained in the Project Methodology section and is focused on the CMMI side. How the methodology has been applied within the PROFO-C project is also covered in this section. In the Lessons Learned section, we comment on the strengths and weaknesses of our methodology that we have identified so far. Finally, in the Conclusions section, we sum up the problems we have run into when implementing CMMI level 2 at small organizations and discuss our approaches to overcome these difficulties. We consider these approaches to be contributions to future extensions of the model to address small settings.

Project Description

Many characteristics make ACTI PROFO projects unique:

²² Asociación Chilena de Empresas de Telecomunicaciones e Informática (Chilean IT Companies Association)

- It is a SPI project in which at least five companies, whose revenues do not exceed US\$3M per year²³, must work together to implement the improvements in parallel.
- Each participant must work towards achieving two goals: the main, compulsory goal is obtaining an ISO9000:2000 certification; the secondary, optional goal is passing a CMMI level 2 evaluation.
- Participants are entitled to obtain governmental economic aid, which is provided in three annual subsidies.
- Due to these subsidies, projects have hard deadlines that must be met.

Each one of these characteristics has introduced challenges for our consultancy, which are discussed later.

In this paper, we mainly present our work in a particular project, named PROFO-C, which started in November 2004 and is currently in the models implementation phase. Six companies of different sizes were gathered to participate in this project. Considering that these companies must work in a cooperative environment, they were carefully selected so that no direct competitors met. Therefore their niche markets are different: some are mostly oriented to business intelligence; others are oriented to mobile communications; while others are focused on retail. Customers are mainly Chilean, though one participant has recently started businesses in Ecuador, and others will soon be doing the same.

Most of these companies depend largely on maintenance work for one or two key clients. Accordingly, they have a simple structure based on the maintenance business area; a few others include a development area, and one includes a software factory. This is in direct relation to the kinds and sizes of projects they develop, which ranged from few weeks to several months, with project groups from two to twenty people.

Project Methodology

The fact that there are no competitors in PROFO-C group does not automatically encourage teamwork. The strategy has been to establish two project levels: there is an inner SPI project within each participating company and an outer project that extends to all of them.

The Outer Project

A general structure has been defined for managing the dual application of SPI initiatives. This has been named the outer project, which allots 12 months to implement ISO 9000:2000 and an extension of five extra months for those companies adopting CMMI level 2, whose adaptation begins on the second month of the general project. The outer project's life cycle, adapted from the PDCA cycle [Shewhart 1939; Deming 1989], includes six phases: planning, definition, piloting, training, institutionalization and, optionally, evaluation. There is a steering team, which assembles in monthly follow-up meetings, responsible for monitoring

²³ This figure is significant for the Chilean market and should not be directly compared to American market's standards.

the achievement of the milestones defined in each phase and taking corrective actions if deviations occur. This team is composed of one or two representatives from each company and two consultants and is led by a project manager who was proposed by the members but assigned by ACTI. By including members of the inner SPI projects in the outer project's steering team, we seek to encourage cooperation and synergy, which we expect will boost motivation within the inner SPI projects.

PROFO-C's outer project is currently closing the second phase of its life's cycle. The definition phase, on which this paper focuses, corresponds to the mapping of six CMMI level 2 PAs²⁴ to the current activities in each participating company. The milestones of this phase are the validation of documented descriptions of each PA. The companies make the definitions at the same time, sequentially, with tight deadlines of one month per PA.

The Inner SPI Project

The methodology employed for inner projects aims to address the reluctance of small companies to spare precious resources for improvement-related tasks, which include a dynamic organization of roles, recommendations of the amount of time dedicated to the project, and the close monitoring, guidance, and support by our consultants.

Each company assigns an inner team to work on its SPI project, which is managed by a project leader. This leader is the same person throughout the project and normally participates in the monthly meetings of the outer project. As such, he or she is responsible for maintaining the team's enthusiasm and being vigilant about the deadlines. The other members of the group may vary from PA to PA because we prefer to have the people who work in direct relation to a PA's activities participate in its adaptation. This dynamic assignment of roles allows the companies to not sacrifice resources of each area of the software development process to large permanent teams dedicated to the SPI project.

The inner SPI project incorporates three meetings per PA. The first one is held shortly before the adaptation of the PA begins with the participation of all inner teams. This meeting has a workshop structure, in which we explain the scope of the area in improvement. We also seek to make people aware of the relevance of the PA on their daily work so that motivation rises.

The second meeting takes place soon after the workshop and is between the consultant and each inner team separately. At this time, the company's current activities are discussed and modeled, and then suggestions to fill the gaps, in respect to the CMMI model, are proposed and evaluated. One of the main considerations when building this model is to keep activities as simple as possible so that they can be continued.

In order to complete the workload set up at this second meeting, we recommend that inner team members dedicate 30% of their daily time to the PA adaptation. Likewise, project leaders are expected to invest 75% of their time on the project. In addition, it is suggested that

²⁴ The SAM PA was not considered relevant to most of the companies in the project.

other people from the organization participate on the validation and training of the process improvement, using between 4 and 20 working hours. These recommendations were estimated from experience on a previous PROFO project and consider that a consultant will be doing supporting tasks. However, the participants are free to organize their resources accordingly to their needs as long as they comply with the outer project's constraints.

The third, and last, meeting with the consultant is held a week before the follow-up meeting of the outer project team. On this occasion, we discuss the PA adaptation drafts, analyze the difficulties that could have arisen, and validate the proposal against the model.

It should be clarified that between these meetings, inner team members can resort to remote consultancy by contacting us by email or telephone.

This mechanism spurs a closer relationship with, and trust in, the consultant and maintains a trade-off between the quality and the financial viability of the consultations.

Lessons Learned

PROFO projects set fixed deadlines that only consider the time necessary to pass on the concepts of CMMI level 2. On the other hand, small companies that usually work on tight budgets do not invest greatly in training their personnel [Broadman 94]. We have seen the effects of this lack of updating and had to spend a significant amount of time introducing the latest software engineering paradigms. Consequently, progress is dependent on both people's competencies and the interest they put on covering their conceptual gaps during working hours.

CMMI assumes the existence of a software development methodology that is the cornerstone on which improvements are built. However, we have come across organizations unaware of their own working procedures. Moreover, projects are carried out independently according to the views and background of the particular project manager in charge. Without having a common way of developing software, it is difficult to find instances of improvement. Our solution has been to assemble a methodology first, by seeking and collecting the most common practices. Once an agreement has been reached on the methodology, the search for target processes begins.

One of the achievements of the methodology has been the underpinning of synergy among the participating companies. They are encouraged by the success of their peers to catch up with their progress. Furthermore, they are sharing experiences and tools in order to maintain the well being of the overall project.

The milestones established by the outer project, and the fact that all share a common market, also urge them to advance the work to accomplish the targets in order to avoid losing competitiveness and credibility.

Another important aspect of a PROFO project, though not discussed in detail here due to space issues, is that the parallel implementation of ISO 9000:2000 helps to provide a better support for a CMMI initiative by improving the areas of the organization that are not directly related to software development.

One of the key factors that the literature identifies for a successful SPI implementation is the high-level management support. In PROFO-C, we have learned that having motivated people is equally important. We have had cases where high-level sponsorship of the commitment of team to the project falters because of business opportunities that represent attractive short-term benefits.

Conclusions

Most of the difficulties we have had to deal with came from the suppositions that the CMMI model makes. It presumes that the target practices exist within the organization implementing it, at least in a not-well-organized form. Unfortunately, these assumptions seem to be quite ambitious for very small settings. Somehow, this happens because CMMI tries to cover the characteristics of most IT organizations, neglecting to consider the special needs of the smallest ones. Perhaps it would be convenient to study a type of pre-processing with a clear checklist of the minimum requirements, which could prepare the way for implementing CMMI level 2. Among the considerations that should be taken into account are

- Small organizations do not have their core competences fully independent from their main clients, who might not always understand that their requests will take longer because there are new commitments with an internal SPI project. Our approach has been to keep the members of the inner teams motivated. They respond by making all the efforts to do their SPI-related work without affecting their normal business-related tasks.
- Small settings, in our experience, have rather flat organizational hierarchies, in which one person might fill more than one position. Consequently, roles and responsibilities are not clearly established. Our approach has been to apply parts of ISO 9000:2000 before starting the CMMI SPI. This way, by the time the first PA adaptation begins, the whole organization is much more structured.
- Small companies cannot deviate their valuable resources from the business activities for long nor afford new entrances. Our approach has been to establish dynamic teams with unambiguous goals, clear schedules, and effective guidance by the consultants.

We are conscious that we have yet to face more challenges in PROFO-C. We will do it, as we have up to now, with clever ideas and an open mind. Despite the past and future difficulties, we believe CMMI can be an important asset to small settings provided that their special needs are properly addressed.

Acknowledgements

We would like to thank to everybody who gave us their precious comments and encouraged us to write this work. We are grateful to Practia Consulting for their support and ACTI for making this project possible.

References

- [Broadman 94]** Broadman, Judith G. & Johnson, Donna L. *What small business and small organizations say about the CMM* (Technical Report 0270-5257/94). LOGOS International Inc., Needham, MA, 1994.
- [Deming 86]** Deming, W. Edwards. *Out of the Crisis*. Cambridge, MA: MIT Press, 1986.
- [Shewhart 31]** Shewhart, Walter A. *Economical Control of Quality of Manufactured Product*. New York, NY: D Van Nostrand Co., 1931.

Biographies

Karin Steembecker

Practia Consulting S.A.

ksteembe@pragmaconsultores.com

Karin Steembecker received her first degree in computer science from the Universidad de Santiago de Chile, in 1996. She worked as a software engineer for several companies in the IT area. In 2004, she obtained her master's degree in information systems from the Leeds Metropolitan University, England. She is a consultant of Practia Consulting in Chile, helping organizations to be appraised against SW-CMM and improving their processes. She is an invited lecturer of the University of Santiago de Chile. Her main areas of interest are software process improvement and information systems strategy.

Miroslav Pavlovic

Practia Consulting S.A.

mpavlovi@pragmaconsultores.com

Miroslav Pavlovic received his first degree from the University of Belgrado. He worked as leader of engineering and development in Yugoslavia and Germany. He is an active member of the IEEE-Standard Association and participates in the development of the IEEE P1074 (12207) - Standard for Software Development Process Lifecycle. He is the consultancy manager of Practia Consulting in Chile and helps companies to be appraised against SW-CMM.

5.5 International Research Workshop for Process Improvement in Small Settings Position Paper CMM Fast Track (CMMFT) Programme for Hong Kong SME Software Companies

Authors

Dr. Hareton Leung, Mr. John Goh, Ms. Yvette Lui

Abstract

This paper presents the CMMFT Programme, funded by the Hong Kong Government, under the SME (small and medium enterprises) Development Fund. Our project aims at developing a Toolkit that can expedite understanding and adoption of CMMI by a model group of small and medium software organizations. The objectives and benefits of the project will be explained first, followed by a high level project plan. Before commencement of Toolkit development, we collected Toolkit requirements by surveying a sample group of 21 SMEs followed by conducting mini-assessments on seven of them. This enabled us to develop a “generic” model of an SME that best represents the characteristics and requirements of a typical software organization in Hong Kong. We fully understand that no one model will fit all organizations. Thus, our aim is not to develop a Toolkit that can be applied to all software organizations; rather, we are focusing on a subset of companies that best fits our generic model by reviewing the results of the survey and mini-assessments. Finally, some key challenges facing our projects and key lessons learned will also be presented.

Background

The CMMFT Programme was initiated and is currently structured as follows:

- It is funded by the Hong Kong Government, under the SME (Small and Medium Enterprises) Development Fund.
- It is being managed and developed jointly by a local non-profit professional IT body, Hong Kong Computer Society (HKCS), and a local tertiary institute, the Hong Kong Polytechnic University.

Objectives

The main objectives of the CMMFT Programme are to

- produce a solution by way of a Toolkit to enable SMEs to reach software development capability while complying to the CMMI assessed to levels 2 (Managed) and 3 (Defined) in the most expedient and effective manner
- address the need for the Hong Kong SMEs to readily accept and use CMMI as the de facto standard for software development

Benefits

The major benefits that are expected to be obtained from using the Toolkit include the following:

- faster and cheaper method of achieving CMMI capability by using a Toolkit that helps speed up understanding, application, and compliance
- significant improvements in the quality of software and systems developed by local SMEs with increased understanding, application, and compliance to CMMI
- increased competitive advantage and recognition for the Hong Kong software industry in the local and international circles with increased adoption of CMMI and related software quality improvement standards

Small Settings and Hong Kong Software SMEs Environment and Activities

With less staff numbers, the communication channels in small settings are likely to be on a smaller scale.

There are approximately 1,000 software SMEs in Hong Kong. A software SME is defined as one that is likely to employ less than 50 full and/or part-time staff with an approximate annual gross turnover of less than \$2 million (USD). Other characteristics that typify a Hong Kong software SME include the following:

- key business area is focused on developing, customizing, and/or supporting a specific line of software products or systems
- organization structure could include a development office or offices in mainland China where labor costs are inexpensive, keeping the managing and controlling functions in Hong Kong
- main thrust tends to be on remaining competitive by delivering products and services in a timely manner with less focus on management, control, documentation, and other quality-related processes

To remain competitive on a local, regional, and global scale, Hong Kong software SMEs need to gain recognition for the work they do by complying to an internationally recognized quality management standard like CMMI in the most cost-effective and timely fashion.

The CMMFT Programme was started with all of the above in mind.

Models, Techniques, Approaches Under Development

The CMMFT Programme consists of 4 phases of development:

Phase I: Initiate and develop programme and product, i.e., CMMFT Toolkit for Hong Kong SMEs

Phase 2: Implement CMMI Fast Track Programme using the Toolkit for Hong Kong SMEs

Phase 3: Establish local software quality certification scheme leveraging on CMMI Fast Track Programme and Toolkit

Phase 4: Promote software quality/development excellence leveraging on success stories on all of the above

We are currently working on Phase 1 of the CMMFT Programme and focused on developing the Toolkit. When ready, the Toolkit will consist of the following:

- an **Introductory Guide** to serve as an introduction to the Toolkit and a reference to other supporting pieces of documentation in the overall delivery package
- a **Procedure Manual** that details the procedures that are required to be followed to reach levels 2 and 3 in the most expedient manner
- a **Technical Guidebook** that provides guidelines, templates, and techniques that should be used in adhering to the procedures in the Procedure Manual
- An **Implementation and Improvement Guide** that describes the steps that should be followed and critical success factors in implementing and using the overall Toolkit

The CMMFT Toolkit will be written in an easy to access, understand, follow, and adopt approach/method, employing relevant work-flows, diagrams, templates, and icons to illustrate and accompany the written text.

Before starting the development of the Toolkit, we first collected the SME requirements by conducting a survey and mini-assessment of seven companies.

In the development of the Toolkit, we will go through three rounds of review by experts and CMMI consultants. Then the whole Toolkit will be reviewed again by six external reviewers. Finally, the Toolkit will be trialed by seven SMEs. Their feedback will be used to improve the Toolkit.

Over time, the intention is to translate the Toolkit into Chinese to alleviate language difficulties in the majority of Hong Kong and regional IT staff whose first language is Chinese.

Project Status

Survey

From January to May 2005, we conducted a survey of local software SMEs.

Objectives

- study the development practices of these SMEs

- classify SMEs into a list of common characteristics and practices for business operations and requirements
- assist in the construction of a *generic* or *model* software organization that best represents the characteristics of a typical Hong Kong Software SME in business nature, company organization, product type, mode of operation, and requirements for CMMI implementation

Raw Data

- successfully contacted over 200 local software companies
- invited those who were interested to participate in the survey
- 21 questionnaires were returned
- seven companies selected to be in the Pilot Group

Analysis

- Majority agreed that software quality improvement should be part of the organization's business strategy or objectives.
- Hong Kong Software Quality Assurance (SQA) practice of the industry is low when compared with other countries in the region, such as India.
- Majority have not instituted process and provided sufficient resources for SQA activities.
- Mainland China provides low-cost labor intensive computer workforce.

Characteristics

- Clients of surveyed companies are not confined to a specific industry although they lean more towards trading and manufacturing software applications.
- Internet-based applications, enterprise management systems, and database applications are in big demand.
- For a typical project, the project duration is 3 to 6 months with 5 staff, 18 person-months, and project value of about \$81,000 (USD).
- More than 40% of the companies have set up offices in Mainland China.
- Workforce size of the Mainland China office is larger than the Hong Kong office.
- In terms of development practices, most companies lack measurement, contractor management, and SQA process.

Mini Assessment

The first mini assessment exercise was conducted during June and July 2005. This was *not* a formal SEI assessment, thus, the true and precise ratings of the company will need to be independently verified by a qualified SEI assessor

Objectives

- solicit requirements from participating SMEs in order to prioritize the development of the procedure documents
- obtain quantifiable benchmarks on which to compare results before and after using the CMMFT Toolkit
- serve as gap analysis to allow SMEs to know how much effort they would need to close the gap in reaching a certain level of CMMI maturity

Methodology and Coverage

- at SMEs' premises
- brief presentation (about 30 minutes) explaining the necessity and scope of the mini assessment
- short questionnaire (about 20 questions) posed to different personnel within each SME
- covered only levels 2 and 3 of CMMI
- not all PAs within CMMI level 3 were assessed

Results

- scores were given later using the long questionnaire
- summaries of strengths and improvement opportunities were prepared for each SME, and de-briefings were arranged
- presented in terms of *categories*, and ratings were given as *High (H)*, *Medium (M)* or *Low (L)*
- ranged from mediocre to “marginally qualified”

Analysis

- A couple of SMEs did fairly well since they were into ISO 9001, and thus their documented procedures and evidences were quite adequate.
- Yet a couple more followed IBM[®]'s RUP[®] (Rational Unified Process[®]) religiously, and although some of the required documentation may not be sufficient, generally speaking they did alright.
- A few companies need more effort into building the process and quality mindset. Although the CIO may be very enthusiastic about the Programme, the staff may not be aware of senior management expectation.
- Overall, all of the companies scored lowest in the *Process Category*, as most do not have the meta-level procedural documentation of guiding the staff how to write the procedure for procedure.
- In addition, a few companies expressed concern in *Risk Management*, saying they would require most help in this area.

Next Step

- orientation of tailoring guidelines and pilot procedure
- solicit feedback from SMEs

Challenge and Solution

For SMEs

- lack of resources (people and time) to implement CMMI
- long distance management of staff in another location/country (company outsourced its labor-intensive jobs to China)
- dissemination of CMMI concepts and importance to Chinese staff
 - need initial hand holding and frequent travel between Hong Kong and China
 - use of tools (e.g., MSN) to facilitate constant communication between dispersed offices
- languages/cultural differences between Hong Kong and China
- very steep learning curve for people who do not fully appreciate the meaning of procedures
 - attend free-of-charge process/quality briefings hosted by government sub-vented body (HKPC)

For CMMFT Implementation Team

- design/creation of web-based Toolkit
- design of different facets of looking at processes (SDLC, Phases, Levels, etc.)
- additional effort to “train” SMEs in understanding the procedural documents—to ensure that the prepared documentation is well received and understood by SMEs, it was decided to have an orientation for all SMEs.

Lessons Learned

For SMEs

- Most of the SMEs have staff in China. Hong Kong offices evolve to be Project Management/Sales offices rather than software development.
- Having a person with a quality role in China would facilitate the dissemination and collaboration of process improvement-related initiatives.

For CMMFT Implementation Team

- The idea of “Fast Track” meant well, but in practice, it is not in the best interest to just throw a piece of procedure to SMEs to pilot without going through (at least initially) some of the contents and expectations of CMMFT Implementation Team.

What Are the Most Important Topics for the Research Community to Address in the Future?

1. How can we reduce the cost of implementing CMMI?
2. Is there a “reduced” CMMI model for SME?
3. What are the cultural issues that affect the successful implementation of CMMI?

Acknowledgement

The CMMFT Programme is organized by the Hong Kong Computer Society and funded by the SME Development Fund of the Trade and Industry Department, the Government of the Hong Kong Special Administrative Region.

Biographies

Hareton Leung

Dept of Computing, The Hong Kong Polytechnic University

cshleung@inet.polyu.edu.hk

Dr. Hareton Leung is an associate professor and director of the Software Development and Management Laboratory, Department of Computing, Hong Kong Polytechnic University. He is currently managing the CMM Fast Track (CMMFT) Programme, funded by the Hong Kong Government. Leung serves on many international and local conferences on software quality, software metrics, and maintenance. In 1995, he co-founded the Hong Kong Software Process Improvement Network (HKSPIN) to promote software process improvement and quality improvement in Hong Kong.

John Goh

Dept of Computing, The Hong Kong Polytechnic University

john.johngoh@gmail.com

John Goh is a project and technical consultant on the CMMFT Programme. He has more than 30 years of experience that includes systems, project, and people management/consulting on an international scale. In 1997, he managed the HKSQA Model development, which was successfully launched in August 1999 and is now being used by local software organizations in Hong Kong. He teaches general, project, and quality management courses for both the Learning Tree International and the Quality Assurance Institute (QAI). He is a founding committee member of HKSPIN; the president of QFA (QAI Federation Asia) headquartered in Hong Kong; a project management professional; and a Certified Software Test Engineer (CSTE).

Yvette Lui

Dept of Computing, The Hong Kong Polytechnic University

csyvette@comp.polyu.edu.hk

Yvette Lui is a technical consultant on the CMMFT Programme. She is a Certified Software Quality Analyst (CSQA), was trained in the SEI Challenge Program (Motorola, based on CMM v.1.1), and has led more than 20 internal IT quality assessments for Motorola

Semiconductors (Hong Kong). She headed Motorola Asia IT quality, handling PQT (process, quality, and tools) and PMO (program management office) activities. She set up the Motorola Asia IT SEI Advisory Board in 2001 and led consultative meetings to go through assessment questions in KPA Levels 2 and 3, producing guidelines to assist project leaders and practitioners in handling self-assessments.

5.6 Quality Software Map of South America

Authors

Alejandro Bedini G, Adriana Llamosa Ardila, Miroslav Pavlovic, and Karin Steembecker

Abstract

Processes are improved in accordance to the needs of the organization. This condition is particularly strong for small organizations in South America, where their main concern is on operative issues rather than strategic ones. This situation is especially exacerbated by the surrounding environment, where the shortage of human and economical resources occurs on an everyday basis.

In this study, we present a map of the main improvement frameworks used across South America such as CMMI, SPICE, and ISO9000:2000 and associate them to the improvements initiatives that have permitted them to spread in the region. We also point out the success factors that have allowed these models to be adopted. The discussion goes further, analyzing the role of the SPIN (Software Process Improvement Network) organization as a gathering entity of improving initiatives and its challenges.

Introduction

It has been twenty years since the term software process improvement (SPI) became part of the software engineering field. Every year, the number of companies applying either maturity models or process improvement frameworks in their organization grows. On the other hand, it is well known that frameworks like CMM, CMMI, IDEAL, and ISO 9001 were initially conceived for larger institutions, so then what about small to medium size companies? Is it possible to use these improvement frameworks despite a company's limited resources?

All these misgivings are pretty common among small and medium size companies and at congresses and discussion forums that we have attended. They have turned out to be the main excuse for not committing to an improvement initiative in many cases.

The present study aims to show the current software process improvement situation with respect to the number of certified organizations, the success and failure facts, the costs associated to SPI, and the SPINs' effect as a gathering organization of improvement initiatives in South America.

This study involved 19 software improvements projects, ranging from those on their way to obtaining a certification to those that already have it. Additionally, there were five projects which never finished, and some that had just started.

To complete the study, thirty five official documents and reports from SPINs were analyzed. A series of conversations were held with people from the 25 software engineering congresses that the authors attended during the last few years.

The study focuses on the four frameworks most frequently used in the region: SW-CMM, CMMI, SPICE, and ISO 9000.

Quality YES, to pay for it NO!

For companies to survive on the XXI market, they must implement two key elements: quality and technological supported business (eBusiness)²⁵.

On average, small and medium size companies represent between 65% and 90% of the work force in each country of South America. They are tilted to have an operative vision of their business rather than a strategic one. A common factor spotted during this study was the lack of alignment of the organization to the improvement project based on a technical decision.

This situation prompted companies to cancel their improvement projects as soon as they faced economic troubles, diminishing the importance of the project for the survival of the company.

The fact that organizations in South America last for only about five years, it is possible to get the higher ROI (Return of Investment) under ISO 9000:2000 certifications [Rico 04]. In spite of this, a successful CMMI evaluation has a major impact on the market, allowing organizations to expand business. A well known case is Boeing Corporation where executives claim that moving from level 1 to level 2 brought about a 145% of reduction on being over schedule [Gartner 05]. Furthermore, to get to level 3 meant having less rework on their projects.

When deciding whether or not to invest on an improvement initiative, the primary cost weighed is the consultant's fees. Table 24 shows the fee range of consultants for a group of South American countries.

Table 24: Hour Fee per Consultant

Country	US\$ per Hour
Argentina	60
Bolivia	25
Brazil	70
Chile	95

²⁵ From Bedini G, Alejandro. Calidad y conocimiento, Cartagena de Indias Colombia.

Colombia	53
Ecuador	55
Paraguay	43
Uruguay	40
Venezuela	35

The quality map shown on Table 25, exhibits the number of CMMI-SW/SPICE and ISO9000:2000 appraisals performed on local companies. The first column indicates the country, the second and third columns show the number of companies that performed SW-CMM and CMMI evaluations, the third column shows SPICE trials, and the last column shows ISO900:2000 certifications.

Table 25: Quality Map on South America

Country	CMM	CMMI	SPICE (ISO/IEC TR 15504)	ISO 9000
Argentina	A <10 1 L2 1 L3			4
Bolivia	A <10 1 L3			
Brasil	A 28 24 L2 6 L3 1L4 1 L5	A <10	2	85
Chile	A 19 7 L2	A <10 L2 1 L3		

	5 L3 1 L5			
Colombia	A <10 1 L2 1 L5	A <10		63
Ecuador			2	1
Paraguay	NO records			
Perú	A <10 1 L2			
Uruguay	A <10 1 L5			
Venezuela	A <10 1 L2	A		2

Table 26: Asociación Latinoamericana y del Caribe de Entidades de Tecnología de Información (ALETI)

Argentina	CESSI - Cámara de Empresas de Tecnologías de Información de Argentina
Bolivia	CICOMBOL - Cámara de Informática y Telecomunicaciones de Bolivia
Brasil	ASSESPRO - Associação das Empresas Brasileiras de Tecnologia da Informação
Chile	ACTI - Asociación Chilena de Empresas de Tecnología de Información
Colombia	FEDESOFTE - Federación Colombiana de la Industria del Software y Tecnologías Informáticas Relacionadas
Ecuador	AESOFTE - Asociación Ecuatoriana de Software
Paraguay	APUDI - Cámara Paraguaya de la Informática y las Telecomunicaciones
Perú	APESOFTE - La Asociación Peruana de Productores de Software
Uruguay	CUTI - Cámara Uruguaya de Tecnologías de la Información

Venezuela	CAVEDATOS - Cámara Venezolana de Empresas de Tecnologías de la Información
------------------	----------------------------------------------------------------------------

One critical factor to consider when analyzing a framework's applicability on the region is the fact that many of them (SW-CMMI and SPICE) assume that all those involved on the improvement project have strong software engineering knowledge. Unfortunately, this is not the reality in South America where many of these practices have been left out at many higher study centers.

Regardless of this, the local industry requires skillful people trained on software engineering practices that are able to deal with every day problems on a professional basis. This promise shows exactly why improvements models are most needed in this region.

IT companies need to reach an excellence on software engineering and management practices. Nowadays, organizations are more concerned with on-time deliveries, robust products, projects that run on schedule, and cost. Due to these concerns, it is not enough to apply testing to the final product; an organization must have a development process. One way to get a process is to adopt one of the named frameworks.

Even though competitiveness is the most importance driving force for improvement initiatives, it should not be the only one. Consciousness of the implications of bad quality software must encourage an organization to seek better quality. Cases of software failure and its disastrous consequences are found in many magazines and books.

Forces Behind Improvements Initiatives.

A typical phrase that we have heard over and over on improvement projects with which we have helped is "*we want to be better.*" This is good, but when it is time to evaluate the expected benefits, the focus is on economical benefits. A common trigger for many companies was another company's certification. This made competitors look for a certification so that they would not loose clients. After analyzing these statements case by case we obtained the graphic shown in Figure 42.

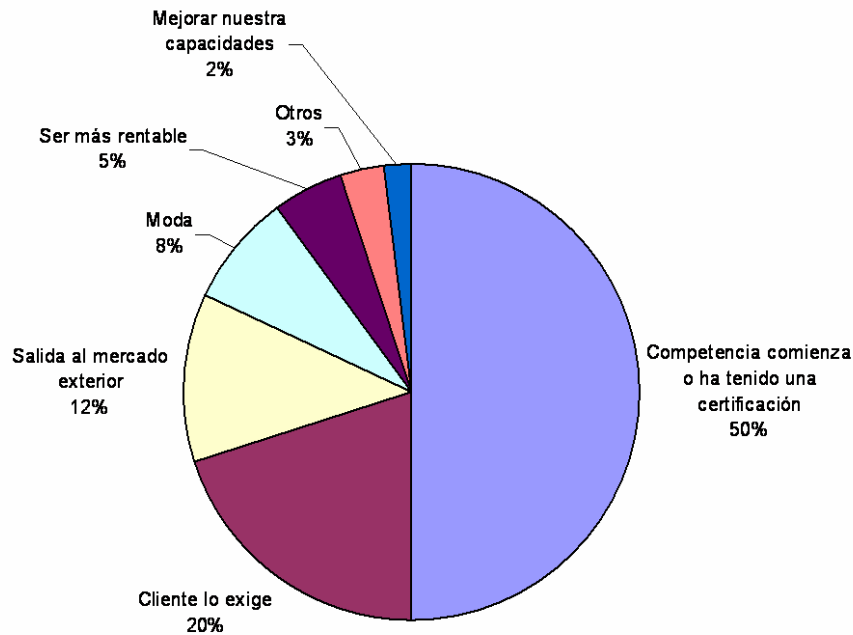


Figure 42: Major forces behind improvement initiatives

The following factors influence an improvement project the most:

- the external niche market to which the company is oriented. South America is largely influenced by US quality frameworks whereas in Europe, the ISO standards are more appreciated.
- previous understanding of the framework to be implemented
- number of consultancy firms in the area
- amount of personnel required on the project and the scope of it
- the cost of the final evaluation or certification
- subsidies granted to these initiatives by the government

SPIN in South America

The aim of SPIN organizations is to encourage improvement initiatives in the local IT industry. Table 27 and Table 28 show the active SPINs in the region, how many per country there are, and their level of activity categorized in high, medium, and low. To see the level of activity for each SPIN, the authors held phone interviews with members and chiefs participants of each SPIN. Web sites also were reviewed for number of popular links using the *Popularidad Web Estable* technique. Additionally, it is worthy to mention that one of the authors is founder and co-founder of four of the SPINs.

Table 27: Active SPINs [SEI 05]

Country	Number of SPINs	Level of Activity
Bolivia	1	Baja
Brasil	10	Alta
Chile	1	Media
Colombia	1	Baja
Ecuador	2	Baja

Table 28: Emergent SPINs [SEI 05]

Country	Number of SPINs	Situation
Argentina	2	Lleva más de 3 años y no ha pasado a emergente
Bolivia	1	La iniciativa se canceló
Brasil	2	Alta probabilidad de convertirse en activa según respuestas de sus presidentes
Colombia	1	La iniciativa se canceló.
Perú	1	Lleva más de 1 año y no ha pasado a emergente

We consider SPIN as a motivating element for improvement projects among the local IT industry, a source of knowledge in regard to the latest paradigms and shared experiences, and a negotiation channel with local governments that support certifications or evaluations on the frameworks under study on this paper.

As mentioned before, to share experiences is one of the cornerstones of the SPINs organizations. To that end, LA (Latin America) SPIN has been established (www.latinspin.org) and its primary concern is to translate CMMI and ISO/IEC 15504 into a Spanish version so that the idiom barriers are not an obstacle for improving. This initiative is being sponsored by SEI (Software Engineering Institute) and it is on conversation with ESI (European Software Institute).

Success Factors on Improvement

It is not easy to categorize the analyzed projects under a common set of factors that influence successful initiatives the most. However, the authors have come across the following factors which appear in many successful initiatives:

Success Factors by Project

- the team profile chose to undertake the project: they tend to be proactive, persevering, and committed.
- to carry out the improvement initiative as if it was a normal project with deadlines, milestones, and controlling points
- to follow an improvement cycle with clear and simple phases of elaboration, review, and implementation
- to use simple, easy-to-be-obtained, and meaningful metrics
- to choose indicators in alignment with business's objectives
- to drive the project as an essential strategic asset of the whole organization
- to look for quality and not for a useless piece of paper

Success Factors by Country

Chile and Brazil have the highest rate of companies that are certified or evaluated on CMMI, SPICE, or ISO9000:2000. The reasons for this are government support to boost the local IT industry and their vision of software as an exportable product.

Brazil is one of the pioneering countries in the region that adopts quality standards in the local industry.²⁶ This achievement is mainly due to the implementation of the *productivity and quality software program* (PBQP-Software), which was subsidized by the Brazilian government in 1997. It involves around 3,600 IT organization and companies that produced close to USD \$100 million in just four years since the initiative was launched.

As an anecdotal note, the *Associação das Empresas Brasileiras de Tecnologia da Informação* (ASSESPRO) (1996) organized around 1,200 IT companies. The association has its own *Nacional de Software para Exportación* (SOFTEX) program, which has a goal to export USD \$2,000 million by 2006.

Similarly, Chile has its software improvement programs sponsored by the Chilean government and implemented under a mechanism called PROFO. In this way, seven IT companies are on their way to be evaluated with SW-CMM by 2005. There is also an association called *Asociación Chilena de Empresas de Tecnología de Información* (ACTI) that gathers small and medium size companies that generated near USD \$850 million in 2004. In addition, there is another association called *Sociedad Chilena de Software y*

²⁶ From Qualidade de no sector de software Brasileiro. Ministério da ciencia e tecnologia, Secretaria de política de informática e automaÇao 2001.

Servicios (GECHS) that includes around 60 IT organizations. GECHS is currently working on a standard based on CMM but oriented to small and medium size companies in the local market.

Likewise, Ecuador has an exportation rate of software of around USD \$30 million and Venezuela reached USD \$206 million in 1999.

In Colombia, the government is sponsoring a program that encourages IT companies to improve their management skills of quality by subsidizing 50 percent of the total cost of the project and 50 percent of the final evaluation. On the other hand, Proexport, a profitless institution, gives an economical aid of 50 percent subsidy for those that want to be evaluated or certified on international standards.

Peru has developed a program, sponsored by *Asociación Peruana de Productores de Software* (APESOFT), that has about 90 companies that will evaluate CMMI within the next year. To do this, a budget of USD \$1 million was set aside to be spent in 40 months. The association gathers around 150 IT companies, and it was founded in 2000. The rate of sales in software is approximately USD \$69 million.

All of the experiences mentioned in this paper show that benefits multiply when the rules of competitiveness and cooperation are united toward a common objective. Consequently, the best thing that companies can do in South America is concentrate forces toward the goal of improving their software process. In other words, when organizations work together, it is easier to obtain reduced costs and government support.

Common Mistakes

According to ESI, the following mistakes are the most common when applying improvement frameworks:

- to impose defined processes on people without considering them as users
- do not listen to the problems of the organization
- do not understand the reality of the organization
- have not been able to have a proper interpretation of the model that is being applied on the organization
- to follow a framework as a path rather than a guide for improvement
- to force the organization to tailor to the model instead of tailoring the model to the organization
- to make the external consultant responsible for motivating people when the consultant is just a methodological tool
- to choose cumbersome metrics that do not provide useful information for the assessment of the initiative

Conclusions

India is often compared to South America in terms of the quality of its software products, but the South American work force is cheaper than it is in India, United States, or Europe. This condition has persuaded the strong economies to invest in the region and prompted local governments to launch improvement programs for the IT industry. SPI is playing an important part in these initiatives by promoting the use of the best practices of software engineering and sharing experiences of improvement projects.

Based on our experience, we recommend the following measures for boosting the IT sector:

- educate and reorient the IT industry by promoting software as an exportable product
- empower SPIN organizations by acknowledging their role as a center of improvement initiatives and a communication channel between centers of studies (e.g., government and the IT industry)
- incorporate quality topics on the pre- and post-graduate studies.

Finally, what we have found in this study is that improvement is feasible for small and medium size companies and CMMI and ISO are the most adopted frameworks in the region.

References/Bibliography

- [CAF 01] Corporación Andina de Fomento. *Cluster de Software en Venezuela de Software en Venezuela Diagnóstico, Benchmarking y Principales Áreas de Acción*. <http://conindustria.org/CLUSTERSDESFTWARESEPT2001.pdf> (2001).
- [Gartner 05] Gartner. *The Benefits of the Capability Maturity Model for Application Development*. http://www.gartner.com/4_decision_tools/measurement/measure_it_articles/2003_0424/ben_cmm.jsp (2005).
- [IBM 05] IBM. *Bolivia Software Factory*. http://www.ibm.com/bo/services/ams/sw_factory.phtml (2005).
- [Rico 04] Rico, David F. *ROI of Software Process Improvement Metrics for Project Managers and Software Engineers*. Boca Raton, FL: J. Ross Publishing, 2004.
- [SEI 2005] Software Engineering Institute, Carnegie Mellon University. *Software Process Improvement Network*. <http://www.sei.cmu.edu/collaborating/spins/spins.html> (2005).
- [SPIN-Chile 05] SPIN-Chile A.G. *¿Qué es un SPIN?* <http://www.spin-chile.cl> (2004).

[TATA 05]

TATA Consultancy Services. *Iberoamerica*.
<http://www.tcs.com/iberoamerica/index.htm> (2005).

**[Tecnologia da
Informação 05]**

Tecnologia da Informação. *Qualificação*.
<http://www.mct.gov.br/sepim/Dsi/qualidad/CMM.htm> (2005).

Biographies

Alejandro Bedini G

SPIN Latinoamérica, Chile

spin@latinspin.org

Alejandro Bedini G received a master's degree in software engineering from the Universidad Santa María, Chile. He possesses a post-degree in financial management from the University de Buenos Aires, Argentina. He is a local coordinator for the SPICE project (ISO/IEC 15504) in South America. He is the author of the Condor model, which allows companies with limited resources to improve their management processes. He is the founder of many Latin SPINs (Software Process Improvement Network) organizations such as SPIN-Ecuador, SPIN-Peru, SPIN Mendoza-Argentina, SPIN-Bolivia, and SPIN-Colombia. He is the author of the "Gestión de proyectos de software" published in Chile, and he is currently working as a consultant at Practia Consulting S.A., Chile.

Adriana Llamosa Ardila

CIDLIS UIS y SPIN, Colombia

adriana@cidlisuis.org

Adriana Llamosa A. is an industrial engineer and internal quality auditor for the Industrial University of Santander (UIS), Colombia. She has trained many internal quality auditors on ISO 9000:2000. She has worked as an ISO 9001:2000 consultant for several IT companies. At the moment, she works as security quality chief for the Latin American Technological Institute of Computer science, IT Colombia, and in the Center of Innovation and Development Investigation in Engineering of Software.

Miroslav Pavlovic

Practia Consulting S.A.

mpavlovi@pragmaconsultores.com

Miroslav Pavlovic received his first degree from the University of Belgrado. He worked as leader of engineering and development in Yugoslavia and Germany. He is an active member of the IEEE-Standard Association and participates in the development of the IEEE P1074 (12207) - Standard for Software Development Process Lifecycle. He is the consultancy manager of Practia Consulting in Chile and helps companies to be appraised against SW-CMM.

Karin Steembecker

Practia Consulting S.A.

kstembe@pragmaconsultores.com

Karin Steembecker received her first degree in computer science from the Universidad de

Santiago de Chile, in 1996. She worked as a software engineer for several companies in the IT area. In 2004, she obtained her master's degree in information systems from the Leeds Metropolitan University, England. She is a consultant of Practia Consulting in Chile, helping organizations to be appraised against SW-CMM and improving their processes. She is an invited lecturer of the University of Santiago de Chile. Her main areas of interest are software process improvement and information systems strategy.

5.7 The Implementation of SW-CMM Level 2—A Successful Case of a Brazilian Company’s Organizational Competence and Commitment

Authors

Ana Patrícia Silveira Viotti, Lilian Corrêa Fontana, and Gabriela Elisa da Cunha

Abstract

Spres Informática is currently located in the city of Belo Horizonte (MG), Brazil, and maintains a team of 150 employees, half of them residing in other main cities of Brazil, acting as commercial and supporting points for the company. Spres achieved SW-CMM level 2 in April of 2005. The company operates as a software house in an extremely competitive market that is constantly changing, making it fundamental to offer high quality products. Because of this factor, in September of 2002, Spres joined a group of seven companies in the project “Towards CMM with Fumsoft (State of Minas Gerais Society of Software).”

The improvement group, SEPG, was formed by integrators of the development staff itself. Commitment, initiative, and responsibility were factors that strongly contributed to the excellent performance of the SEPG group. Spres always concerned itself with communicating the importance of this project not only to the members of the SEPG, but to the entire team involved with the program. Resources, training, incentives, and rewards were provided, creating the commitment and determination to achieve the final objective: SW-CMM level 2.

Paper

Spres Informática is currently located in the city of Belo Horizonte (MG), Brazil, and maintains a team of 150 employees, half of them residing in other main cities of Brazil, acting as commercial and supporting points for the company. The company’s income for 2004 was \$5.88 million, and the forecast for 2005 is have a 20% increase. Spres achieved SW-CMM level 2 in April of 2005. The company operates as a software house in an extremely competitive market that is constantly changing, making it fundamental to offer high quality products, promote continuous innovation in the company’s systems, hastily adjust itself to changes, and quickly and appropriately respond to opportunities. Because of these needs, the company required an effective software process that allowed the development of new software products and the improvement of already existing products. The company had to accomplish this within the timeframe forecasted, according to the previously budgeted cost, quality standards, and satisfactory functionality.

The company wanted to maintain the principle of being desirable to work for, so it always aimed to offer enjoyable working conditions and stability for its employees. In September of 2002, with this objective in mind, Spress joined a group of seven companies in the project “Towards CMM with Fumsoft (State of Minas Gerais Society of Software).” This project was coordinated by UNISINOS (Vale do Rio dos Sinos University) with the purpose of helping these companies implement the level 2 practices of the SW-CMM by providing them with training and consultation.

In the initial phase of the processes improvement project, Spress was using a few internal processes and one of its own tools that had few options such as registration of requirements, meetings, and a basic control of tasks. Even during this initial phase, all the employees were already trained and used this tool during daily activities. Another good practice already being performed was the weekly presentation of technical workshops coordinated by the development team. Spress was always concerned with the implementation of good practices, but more improvement was needed. The solution was to look for support by using a recognized guide in the software community that would act as guidance for software process improvement and also as means for formalizing and legitimizing an improvement program at Spress. That’s why the company decided to improve its software development process by using the SW-CMM model.

The organization was constituted of the Department of Research and Development (R&D) and the Department of Product (DP), which are the areas of the company that are directly involved with the development of software processes. In the period of implementation, the technical staff was composed of only 32 employees, but currently there are 48.

The improvement group, SEPG, was formed by integrants of the development staff itself, discarding the need to hire outside professionals. It was important to know the strengths and weaknesses of current processes. Key people with credibility inside the company, with consistent technical opinions, that already displayed an interest and will to learn new techniques, that showed motivation in acquiring new knowledge in the area of quality, and that were eager to apply these new concepts to the organizational environment were invited to join this group. Group members also had to have basic abilities such as listening and flexibility. A big effort was put into divulging the importance of being part of this group, which also brought status to its integrants. The managers of the areas involved with this project were also invited to take part in this group.

The task of managing the SEPG was designated to a first-line manager who had great influence in the company and had the autonomy to take decisions regarding the improvement project. The competency, professionalism, and determination of the SEPG in conducting the improvement project were a determinant for its success.

None of the integrants of the SEPG staff dedicated all of their time to the improvement project; they continued to work on the software projects but with a reduced time allocation. This fact, which at first seems like a negative, ended up contributing a lot to the results because those that were responsible for defining and approving the new processes were

already implementing these new practices in their software projects. This situation contributed to a fast and consistent creation of adherent and stable processes once the professionals were able to identify many of the difficulties and suggest improvement opportunities.

Commitment, initiative, and responsibility were factors that strongly contributed to the excellent performance of the SEPG in this work format. Spress always concerned itself with communicating the importance of this project, not only to the members of the SEPG, but to the entire team involved with the program. Resources, training, incentives, and rewards were provided. This brought the commitment and determination to achieve the final objective: the SW-CMM level 2. There was also a combination of “knowing how to do it,” experiences, and behavior exercised in the context of improvement.

The SEPG, composed of seven professionals, met systematically every 15 days, and performed monthly consultancies together with UNISINOS with full attendance. Each integrant was responsible for a level 2 KPA. In the initial phase, the technical lectures were used to inform the entire staff of the concepts of the model. A schedule of the project was elaborated, and all the tasks were performed under the strict supervision of the SEPG manager.

Specific trainings based on the new processes and procedures proposed were being conducted along the project of improvement. During these trainings, conducted by the SEPG, fun activities were performed, and prizes were distributed to the participants in order to make this learning process more enjoyable and productive. The participants had to fill up a detailed evaluation form of the result and identify the problems and the possible improvements to the procedures involved.

This dynamism was possible due to a management model used by the project coordinator, which was supported by the board of directors. This model provided the creation of a stable and friendly environment for the development of competencies that allowed an adequate relationship with the instabilities of the environment, which ended up being fundamental to the survival and the success of the implementation of SW-CMM level 2. This innovation environment allowed the participants to develop, besides the technical competencies required by the process of implementation of the SW-CMM itself, competencies such as creativity, innovation, flexibility, adaptation to change, perception, and improvisation without overlooking the strategic implications. This participative management model was essential for a relationship environment based on respect and trust, which was reflected throughout the entire organization.

The selection and the definition of strategies, along with the adopted actions for the implementation of SW-CMM level 2 into the company, was not an easy task. Many obstacles had to be overcome, not only on the technical and relationship sides, but also in the supply of financial resources and staff. This is the reality of small and medium Brazilian companies. The success of the improvement project depended on the ability and the involvement of its professionals and the relation of co-responsibility between the staff and the company. We

conclude that if a team is well prepared, motivated, and well coordinated and uses a tool that supports the implementation of the model, implementing process improvement, and achieving the desired maturity level, can be quick and successful.

Biographies

Ana Patrícia Silveira Viotti

anapaty@spress.com.br

Ana Patrícia Silveira Viotti graduated as a civil engineer and holds a master's degree in computer science. She is a technical director and SEPGroup leader at Spress Informática—a CMM Level 2 software house in southeast Brazil. She has experience with technical presentations on the applications and technology used by Spress and has represented the company in several local and international congresses.

Lilian Corrêa Fontana

lfontana@unisinós.br

Lilian Corrêa Fontana is currently acting as a project manager and is responsible for the definition and implementation of software process improvement programs for SMEs and large companies, using mainly CMMI-SE/SW. She teaches supplier agreement management, testing, and quality assurance courses. She has published articles and is a speaker at local and national conferences on software quality. She holds an MBA in strategy and competitiveness and a BS in computer science.

Gabriela Elisa da Cunha

gcunha@unisinós.br

Gabriela is currently acting as a project manager and is responsible for the definition and implementation of software process improvement programs for SMEs and large companies, using mainly CMMI-SE/SW. She teaches metrics and quality assurance courses. She has published articles and is a speaker at local and national conferences on software quality. She holds a master's degree in administration and a BS in computer science.

6 Selected Case Studies

6.1 A Giant Taking Small Steps

Author

Christian Carmody

Abstract

This paper focuses on the adopted approach of University of Pittsburgh Medical Center (UPMC) for internalizing the CMMI model to reduce organizational risks and costs, to meet requirements more effectively, and to become more consistent and efficient when delivering information technology (IT) solutions and services. The approach is allowing for process focus and improvement to become consumable and sustainable by IT staff and departments. It allows the freedom of each unique group to address process improvement activities that are most sensible as determined by the group's business objectives while meeting other major initiatives such as Sarbanes-Oxley compliance and Information Technology Infrastructure Library/Information Technology Service Management (ITIL/ITSM) best practices.

Paper

UPMC is the premier health system in western Pennsylvania and one of the most renowned academic medical centers in the United States. During the past decade, UPMC has reshaped the health care landscape in western Pennsylvania. As a \$5 billion organization and the region's largest employer, it has transformed the economic landscape as well.

Today, with 40,000 employees, UPMC comprises 19 hospitals and a network of other care sites across a 29-county service area that include doctors' offices, cancer centers, outpatient treatment centers, specialized imaging and surgery facilities, in-home care, rehabilitation sites, behavioral health care, and nursing homes.

Even with these impressive statistics, most of the organization operates in disparate and unique settings, which simulate a conglomeration of small businesses. Each business has its own tactical objectives which are supported by both centralized and non-centralized IT staff. IT systems and solutions are mostly acquired, yet modifications are applied to meet the internal requirements of UPMC. Only about 25% of the systems used at UPMC are internally developed.

Multiple simultaneous initiatives at UPMC are altering the way the information technology and services are delivered. UPMC's Board of Directors mandated compliance with the financial reporting guidelines established in the Sarbanes-Oxley Act of 2002 (SOX), as well as implementation of the CMMI model for software process improvement. In addition, IT operations are being aligned with the best practices set forth by Information Technology Infrastructure Library/Information Technology Service Management (ITIL/ITSM). ITIL/ITSM is guidance developed by the United Kingdom's Office of Government Commerce (OGC) describing an integrated, process based, best practices framework for managing IT services. The common goal across each initiative is to focus on improving processes to reduce organizational risk, effectively meet customer requirements, and become more efficient at delivering quality applications, systems, and services.

The uniqueness of UPMC from other organizations attempting to accomplish the same objectives is distinguishable by two facts: (1) Both board initiatives are self-imposed at UPMC rather than by external circumstances or regulations, and (2) deriving the benefits of implementing all three initiatives through one collaborative effort. The process improvement focus provided by the CMMI and ITIL/ITSM frameworks will at a minimum address any process and control weakness identified through SOX preparation and certification.

UPMC's tactical application of implementing the CMMI model started with the popular focus on the staged representation of the model. A group of IT management and staff members within UPMC worked part-time to attempt to implement the model with a goal of achieving maturity level 2. Even with the external support of some of the SEI's finest, the process improvement application of CMMI appeared to be overwhelming and undesirable for the majority of the staff who were exposed to it. After months of effort, very little tangible progress was made.

A revised approach was derived through the understanding of the model by UPMC, and the SEI's understanding of UPMC's environment and culture. The revised approach included switching to the continuous representation of the CMMI model and selecting a subset of process areas to focus on through a phased approach. The criteria for selecting the process areas, included mapping to SOX compliance and UPMC's understanding of its own internal areas in need of improvement. This resulted in the selection of a base set of process areas which included:

- Project Planning
- Project Monitoring and Control
- Requirements Development
- Requirements Management
- Configuration Management

This more focused approach has allowed for a process focus and for improvement to become consumable and sustainable by IT staff and departments. It allows the freedom of each unique group to address process improvement activities which are most sensible as

determined by the group's business objectives. Additional process areas are introduced and applied where it has been determined to be applicable and feasible. Management also has more visibility into the actual achievement of process improvement goals by conducting frequent internal Class B SCAMPI appraisals.

Additionally, UPMC allocated full-time resources to oversee the process improvement implementation. A Software Engineering Process Group (SEPGSM) was formed to train and consult with IT staff and departments to continually improve processes and achieve and sustain high levels of quality. UPMC's SEPG also provides direct feedback to projects, departments, and management on improvement progress (through Class B SCAMPI Appraisals) and maintains a repository of guidance, tools, and other process assets available for use throughout the organization.

The focus and success of continuous process improvement is not new to UPMC, or healthcare. The innovation of evolving clinical processes, procedures, and treatment lie at the heart of UPMC's success and has become (in many cases) what distinguishes UPMC from its competitors. IT process improvement focus is now being positioned to follow suit by supporting UPMC's commitment to continuous process and quality improvement.

Biography

Christian Carmody

University of Pittsburgh Medical Center

carmodyc@upmc.edu

Christian Carmody is an associate director of the Software Engineering Process Group for the University of Pittsburgh Medical Center (UPMC), which is one of the nation's largest integrated healthcare providers. He is responsible for continuous process improvement activities in the information services department. His responsibilities include the successful implementation of Sarbanes-Oxley compliance and Capability Maturity Model Integration (CMMI). Carmody, a certified information systems auditor (CISA), previously served as IT audit manager within UPMC. Before joining the audit profession, Carmody worked as a system/local area network (LAN) administrator at Mellon Financial. Carmody obtained a BA degree in accounting with a minor in business administration from Westminister College (PA) and a Master of Business Administration (MBA) and Master of Information Systems Management (MS) from Robert Morris University (PA). In addition, he serves as an adjunct associate professor at Point Park University and Duquesne University, teaching information systems and business related courses.

6.2 Small Steps, Giant Leap

Authors

Nidhi Srivastava and V. Sathya Murthy

Abstract

TATA Consultancy Services (TCS) is a global IT consulting organization with development centers spread across 5 continents. The size of these development centers varies from about 70-100 associates depending on the business requirements of the center. In the small centers, approximately 5 to 10 projects are executed. These projects are classified in the mini and small categories.

TCS has implemented a quality management framework, called Integrated Quality Management System (iQMS), that is a key component of the organization's process driven culture. This framework provides a common ground that can be tailored when the settings are defined.

This paper presents the tailoring elements required to implement a high maturity framework in a small setting environment. In the small development centers, it was discovered that tailoring is required for three key focal areas:

1. staffing and managing the SEPG and SQA function with optimal investment
2. proving process and product improvements statistically, given the small size of the center
3. tapping process improvement ideas and cross pollinating

Introduction

TATA Consultancy Services (TCS) is a global IT consulting organization with development centers spread across 5 continents. The size of these development centers varies from about 70-100 associates to over 1000+ associates, depending on the business requirements of the center.

In the small centers, approximately 5 to 10 projects are executed. These projects are classified in the mini and small categories.

Table 29: Time and Effort for Mini and Small Projects

Project Type	Elapsed Time	Effort
Mini	Up to 8 weeks	More than 3 person months and up to 15 person months

Small	More than 2 months up to 6 months	More than 15 person months and up to 24 person months
-------	-----------------------------------	-------------------------------------------------------

TCS has implemented a quality management framework, called Integrated Quality Management System (iQMS), that is a key component of the organization's process driven culture. This framework provides a common ground that can be tailored when the settings are defined.

This paper presents the tailoring elements required to implement a high maturity framework in a small setting environment. In the small development centers, it was discovered that tailoring is required for three key focal areas:

1. Staffing and Managing the SEPG and SQA Function with Optimal Investment

The roles in the CMMI model were studied for interdependencies, responsibilities, and activities to find conflicts that will prevent a person from taking a certain combination of roles. The process group in these settings was equipped with one quality lead to oversee definition and deployment activities along with an auditor to perform independent verification. As small projects have short schedules and limited resources, the process group activities were mainly focused on building organization level infrastructure providing easy, simple templates, tools, methods, and aids

The presentation will focus on the role and responsibilities of the quality lead and the audit function in the small center environment. The interface with the corporate process groups, such as Software Engineering Process Group (SEPG), Quality Assurance Group (QAG) & Audit Group (AG), will also be articulated.

2. Proving Process and Product Improvements Statistically, Given the Small Size of the Center

CMMI level 4 practices posed a major challenge in the small settings environment as it was quite difficult to prove process and product improvements statistically due to the small number of projects and, consequently, limited availability of data points. However this challenge was handled by commencing data collection from the project start-up stage and by appropriately tailoring the metrics program to ensure that process and product quality attributes were planned and collected at a granular level in each project. The key tailoring elements of the process and product metrics collection approach includes the following:

- usage of threshold limits to monitor a particular metric until Process Capability Baselines were set
- process metrics collection for internal and external milestones to ensure detailed planning

3. Tapping Process Improvement Ideas and Cross Pollinating

Buy in of the new process and subsequent process improvements was ensured by involving the project leaders as *process owners* and by assigning lead roles on a deployment task force that was constituted for effective deployment. Formal mentoring programs, CMMI forums, and brown bags were initiated to establish and sustain new practices and procedures. Approaches such as one-on-one sessions on process facilitation were also done because they were feasible in these settings. Early training for key people in the model and appraisal methods, continuous mentoring from an appraisal expert within the organization, and excellent management support were key factors for the successful deployment.

The learning objectives for this session were

- business case of high maturity practices in small settings–The Why
- high maturity framework components required in small settings–The What
- tailoring needed for the small setting environment–The How
- results of process improvement–Is it worth the investment?

Biographies

Nidhi Srivastava

TATA Consultancy Services

n.srivastava@usa-tcs.com

Nidhi Srivastava heads the Quality Consulting practice of Tata Consultancy Services in North America. In this capacity, Srivastava oversees TCS's relationship with the Software Engineering Institute (SEI) at Carnegie Mellon University and has worked with several Fortune 500 clients on quality consulting. Srivastava was also instrumental in supporting the North America geography for the most recent appraisal of TCS. In August of 2004, TCS became the world's first organization to achieve an integrated, enterprise-wide Maturity Level 5 on both the Capability Maturity Model Integration (CMMI) and the People CMM.

Srivastava has over 15 years of experience in software development, software project management, and software process improvement. Prior to becoming the practice head for Quality Consulting, Srivastava was the Software Engineering Process Group head at the TCS Delivery Center in New Delhi and was part of TCS's journey to achieving CMM Level 5 there.

Srivastava holds a bachelor's degree in computer science and engineering from the University of Lucknow, India. She is also a Software CMM Assessor and Certified Software Quality Analyst and is trained on the People CMM model. Srivastava has submitted papers to international conferences and has presented papers in several conferences in the US.

V Sathya Murthy

TATA Consultancy Services

m.sathya@usa-tcs.com

Sathya Murthy is a quality manager at TATA Consultancy Services and is currently based in

San Jose, California. He has worked in the small center setting as quality lead and was part of the CMMI journey in the TCS Development Centers in Uruguay and Brazil.

Murthy has five years of experience in software development, software project management, and software process improvement. Prior to becoming the quality manager for the US west coast region, Murthy was the quality reviewer at the TCS Delivery Center in Chennai, India. Subsequently, he had the role of quality lead in the Uruguay and Brazil Global Development centers and was part of TCS' journey to achieving CMM/CMMI level 5 there.

Murthy holds a bachelor's degree in mechanical engineering from the University of Chennai, India.

He is also a Software CMM Assessor and Certified Software Quality Analyst and is trained on the People CMM model.

6.3 Software Process Improvement at Schweitzer Engineering Laboratories, Inc.

Author

Stephen Rupp

Abstract

Schweitzer Engineering Laboratories (SEL) introduced the world's first digital protective relay in 1984, revolutionizing the electric power protection industry by offering fault locating and other features for a fraction of the cost of earlier systems.

SEL's products go into very demanding and critical applications protecting the security and integrity of the world's electric power systems. Consequences of failures of SEL's products in the field can be significant. Schweitzer Engineering has initiated a software process improvement initiative. The most significant business driver for software process improvement is achieving a significant reduction in defects in products delivered to customers. Initiatives consist of a software process group, process improvement teams, and a software quality engineering function.

Challenges to the success of these initiatives exist in commitment, resources, and development projects that span the range of sizes from very small to large. The initiatives and challenges are detailed in this paper as well as suggestions for future research projects.

Schweitzer Engineering Laboratories, Inc. (SEL)

Schweitzer Engineering Laboratories introduced the world's first digital protective relay in 1984, revolutionizing the electric power protection industry by offering fault locating and other features for a fraction of the cost of earlier systems. In the years since, SEL has grown and developed a complete line of products for the protection, monitoring, control, automation, and metering of electric power systems. SEL relays, communications processors, meters, fiber optics, and software products are the roots of complete integrated solutions for protection and communications for the electric power industry.

The company was founded by Edmund O. Schweitzer III, PhD, then a professor at Washington State University in Pullman, WA. SEL has its headquarters in Pullman.

Business Environment

Schweitzer Engineering now sells products and services in more than one hundred countries worldwide. These products go into very demanding and critical applications protecting the security and integrity of the world's electric power systems. Consequences of failures of

SEL's products in the field can be significant, causing outages of electric power and misoperations of the control systems controlling power flow and generation.

These products often operate in unattended, remote locations. They are expected to perform flawlessly for years of operation with no scheduled maintenance or downtime.

SEL takes its commitment to quality very seriously. SEL's worldwide, ten-year warranty not only reflects this commitment but also demonstrates the quality and value SEL delivers. SEL is certified to the International Standards Organization (ISO) 9001:2000 Quality System Standard. This certification is evidence that critical design, manufacturing, and business processes meet the exacting requirements of the internationally recognized ISO program. Our products meet or exceed both national and international testing standards.

Organization

SEL has a workforce of approximately one thousand employees worldwide. There is a small software engineering staff that is involved in software or firmware development.

Projects vary in size from very small maintenance projects to large, new product development projects.

All projects use a common software development process and employ common toolsets.

Business Goals and Objectives for Software Process Improvement (SPI)

The most significant business driver for software process improvement is achieving a significant reduction in defects in products delivered to customers. Actual defect density has improved significantly (2x) but the size of the software and firmware products has grown much faster (5x-7x). Recognition of this fact has initiated a software process improvement program at SEL. The stated goal is a significant reduction in defects.

SEL has also been concerned about the costs of defects to our customers. The costs of upgrading software and firmware to fix potential defects can be a significant operating cost to our customers.

Another business goal is improving performance in project planning and execution. Project overruns cause slips in new product introductions as well as unplanned expenses. The goal is to improve on-time project performance.

Current Software Process Improvement Activities

Software Process Group (SPG)

Firmware engineers, software engineers, power engineers, engineering managers, software quality engineers, and project managers populate the SPG. The group is chartered with the

conduct of software process definition, creation, and improvement activities and with helping the development community achieve SPI goals. As areas for improvement are identified, small process action teams called IMI teams (for identify, measure, and improve) are formed to address those issues.

Software Quality Engineering (SQE) Team

One full-time ASQ-certified software quality engineer is leading a team of nine part-time SQEs, two of whom are also certified. The function and duties of the SQE team are clearly defined and include the mentoring of SPI techniques, process compliance reviews of project teams, assistance with project data collection and metrics, assistance with project planning, customer advocacy, change control board participation, peer review observation, and reporting organizational metrics to senior management. The lead SQE reports independently through the Quality department, while those who participate part-time are prohibited from functioning as an SQE on any project in which they are actively involved. All SQE assets, reports, metrics, etc. are maintained in an exclusive database with access control.

ISO Certification

SEL maintains ISO 9001:2000 certification using a team of internal auditors as well as annual, external certification audits.

Software Process Improvement Models

SEL is using the SW-CMM as its software process improvement guide and is actively working to achieve level 2 compliance by the end of 2005. We are considering a transition to CMMI when SW-CMM level 2 is achieved.

Other Strengths and Advantages

Quality Management Systems (QMS) Organization

The QMS organization conducts internal ISO audits using trained ISO auditors. It also conducts qualification of suppliers and tracks to closure field defect reports.

Defined Processes

Processes are well defined and documented with monitoring and tailoring for very small projects.

Emphasis on Metrics

On a monthly basis, quality indicators are reported to senior management from around the company.

Quality Focus

The following focus is demonstrated in the SEL Corporate Quality Policy: to identify, measure, and improve (IMI) processes. Further, the vice president of Marketing, Research, and Development is also the vice president of Quality. All employees receive training on our quality policy as well as control chart training. Software engineering issues are being communicated via newsletters, luncheons, and presentations. We have a horizontal organization structure that lends itself to bring issues to and getting support from senior management. Funding for improvement investment is generally not an issue.

Process Improvement Challenges

One challenge is that project priorities and work engage engineering managers and limit their involvement in improvement activities. The software and firmware managers are part of the SPG and are a key component for improvement.

Another challenge to the SEL development organization are leftover stovepipes in the organization. Separate functions in the organization make it difficult to achieve a truly integrated development team.

SEL has a stable engineering workforce and thus has a more limited exposure to transfer of ideas from other companies and work environments. Participation by our software engineers in engineering organizations and industry forums is limited.

There is also the challenge of adapting good development processes to the needs of very small maintenance projects. There is always a struggle to balance good process with efficiency.

Rapid growth of the business causes a regular reevaluation of development processes to ensure that they are consistent with the emerging needs of the business. Processes that were appropriate three years ago may no longer be adequate for the business need.

The small size of the development organization requires that we staff software process improvement initiatives part-time. Process improvement projects are sometimes delayed due to the pressure of project work.

Desired Future Research Topics

Because many of our projects are short in duration and have small teams, we could make good use of guidelines for tailoring the process models such that the effort spent in process compliance and confirmation does not exceed that of the development itself, without sacrificing the effectiveness of the process.

Also, a continuing forum to share experiences with those who implement SPI processes or programs from smaller and high growth environments would be welcome.

Biography

Stephen Rupp

Schweitzer Engineering Laboratories, Inc.

steve_rupp@selinc.com

Stephen Rupp is a software engineering manager for Schweitzer Engineering Labs in Pullman, WA. He manages a group of firmware engineers as well as having responsibility for the software development process, training, and engineering tools.

He previously worked for Cummins Engine Co. in Columbus, Indiana. Rupp was the director of Software Technology and the leader of Electronics Functional Excellence for Cummins R&D. In this role, he was responsible for electronics process and tools development for the corporation.

His prior roles included leadership of the Cummins Core Product team, which has responsibility for development of reusable systems and components that are the central part of all engine controls. This included the development of software product lines for Cummins.

Rupp's other experience consists of design of substation communications products for electric power line communications and load management, gas turbine controls, plant automation, missile tracking, and publishing systems.

His special emphasis is on development of embedded systems for severe environments. He is a graduate of the University of Illinois, Urbana with a degree in computer science.

6.4 Two Case Studies in Implementing Model Based Process Improvement in Small Organizations

Authors

Dr. Mary Anne Herndon and Sandra Salars

Abstract

Over the last three years, the applicability of practices in formal process improvement models, such as CMMI and Six Sigma, has often been perceived as more suitable for larger organizations. This perception is fostered by the projected cost impacts to smaller organizations, lack of staff, and information describing the benefits. This paper provides rare insights into the CMMI journeys of two small organizations and their quantitative process performance achievements. Both of these organizations were distinct in attributes such as customer type, geographical location, and management style. Each organization lacked a trained staff, financial resources, and awareness of the potential gains to their businesses.

Although these two organizations had different business goals and consequently different process performance objectives, both organizations shared two key lessons learned from their successes.

The two common key lessons learned were that it was important to focus on business goals during measurement development and it was crucial to have the continued advocacy of higher level management to maintain project continuity.

Background

This paper provides rare insights into the CMMI journeys of two small organizations and their quantitative process performance achievements. The profit and loss organizational structure of these two cases is similar. Each organization started their formal process improvement journey with very limited investment margins. Each of these small organizations started their CMMI journeys three years ago with a target of optimizing key business processes. These key business processes were based upon unique customer requirements and business goals. Case 1 started their formal process improvement journey with little experience in formal process improvement. The starting point in the Case 2 journey was an inconsistent SW-CMM maturity level 2 heritage.

Table 30 provides a profile of attributes for these two cases.

Table 30: Case Attribute Profile

Case Number	S/W Development Team Size	Applications
1	7	Web-based development and maintenance (live applications)
2	10	Web-based development and maintenance (live applications)

Description of Formal Process Improvement Journeys

Each of the organizations operated within critical profit margins. Neither one of these organizations could justify the support of a full-time process improvement champion due to cost constraints and small staffs. At the beginning of each journey, each organization assigned a staff member process improvement as a collateral function. The main customer's endorsement in Case 1 was very strong and favorable, and direct customer support was used to defray the costs of implementation. In Case 2, however, the customer did not advocate or support the implementation activities, and the organization provided all of the investment funds.

While each journey was totally independent, the starting point for each case was to invest in obtaining training in the formal process improvement model by attending the three-day Introduction to CMMI course. The organization in Case 1 arranged to have training at the on-site location with customer participation. The organization in Case 2 also arranged to have on-site delivery of the course but without customer participation. After attending the Introduction to the CMMI course, the organization in Case 1 was able to procure additional training by sending one project team member to the Intermediate CMMI course. The organization in Case 2 did not provide additional training to their staff members due to cost factors.

Immediately after the training, both organizations formed process improvement teams that included higher management participation. Additionally, in Case 2, the organization facilitated institutionalization by using rotating membership from the members of project team. In Case 1 and Case 2, each organization obtained the services of a SCAMPI lead appraiser and developed a formal project implementation plan and an appraisal input statement. The information included in each of the implementation plans is listed in Table 31. Both process improvement implementation plans contained regularly scheduled progress reviews with upper level managers. Each appraisal input statement (AIS) contained the required information.

Table 31: Contents of Case 1 and Case 2 Process Improvement Implementation Plans

Case	Organizational Scope	Cost	Schedule	Key Risk to Project
1	S/w development for live databases and applications	Estimated as % LOE	36 months	Impact to customer development cycles of 1-2 months
2	S/w development for live databases and applications	Fixed price	18 months	Same as Case 1

The business case motivating the organizations in both cases was the current level of customer satisfaction. Each organization faced the challenges of latent defects and delivery schedules. The organization in Case 1 was equally concerned with latent defects due to the impact of rework on a small staff and schedule forecasting. The organization in Case 2 was very concerned with their lack of ability to provide their customer with an accurate delivery date. Both of the organizations were advised, during meetings with their SCAMPI lead appraisers, to focus their investments by defining realistic business goals and measurements that would serve to gauge achievement of these goals. Additionally, each organization was asked to develop estimates, based upon historical data, for acceptable intervals for both latent defects and schedule intervals goals. In each case, the achievement of a SCAMPI benchmark was not considered the focus of the investment.

Formal Process Improvement Journey Benchmarks

The process improvement implementation plans for both cases included a series of Class C, B, and A benchmarks led by an authorized SCAMPI lead appraiser. Both organizations implemented the continuous representation to achieve the SCAMPI benchmarks.

Table 32 provides the SCAMPI classes and the capability levels for each organization during their journey. Both organizations used equivalent staging to obtain maturity levels. Since neither organization had a subcontractor, the ISM Process Area was not applicable; however, SAM was implemented to mitigate risks in implementing new development tools and training from third parties.

Table 32: SCAMPI Appraisals and Model Scopes for Case 1 and Case 2

Case	Class C	Class B	Class A
1	CL3	CL3	CL3
	(ML 2 PAs)	(ML 2 PAs)	(ML 2 PAs)
	CL 3	CL 3	CL 3
	(ML 2 & 3 PAs)	(ML 2 & 3 PAs)	(ML 2 & 3 PAs)
2	CL 4 & 5	CL 4 & 5	CL 4 & 5
	(VER, VAL, PPQA, PP, PMC, IPM)	(VER, VAL, PPQA, PP, PMC, IPM)	(VER, VAL, PPQA, PP, PMC, IPM)
	CL3	CL3	CL3
	(ML 2 PAs)	(ML 2 PAs)	(ML 2 PAs)
1	CL 3	CL 3	CL 3
	(ML 2 & 3 PAs)	(ML 2 & 3 PAs)	(ML 2 & 3 PAs)
	CL 4 & 5	CL 4 & 5	CL 4 & 5
	(PP, PMC, VER, PPQA)	(PP, PMC, VER, PPQA)	(PP, PMC, VER, PPQA)

Results of Implementing the CMMI

The results of the process performance gains for Case 1 and Case 2 are provided in Table 33.

Table 33: Process Performance Gains for Case 1 and Case 2

Case	Performance Gains
1	4.5 < customer satisfaction < 5.0 0 < latent defects < 3
2	-8 < scheduling accuracy < 8 days 0 < latent defects < 1

The lessons learned in Case 1 and Case 2 had many similarities. However, there were differences attributable to the customer environment and the management of the process improvement team.

Table 34 provides a summary of the lessons learned for Case 1 and Case 2.

Table 34: Case 1 and Case 2 Lessons Learned in Implementing Formal Process Improvement

Case	Lessons Learned
1	<ol style="list-style-type: none"> 1. Focus on business goal–measurement development 2. Continued advocacy of higher level management crucial to maintain project continuity 3. Care in managing gaps in customer Maturity levels
2	<ol style="list-style-type: none"> 1. Same as Case 1 except for 3 2. Use appropriate project management tools 3. Rotating participation from staff members important for institutionalization <p>SW-CMM measurements need refocusing for the CMM</p>

Biographies

Mary Anne Herndon
Transdyne Corporation
San Diego, CA
maherndon@san.rr.com

Dr. Mary Anne Herndon is the process improvement director of Transdyne Corporation, an SEI Partner. She specializes in process improvement technology insertions in small settings and engineering and scientific organizations. She is an authorized SCAMPI Lead Appraiser and an instructor for the Introduction to the CMMI course.

Specifically, Herndon provides small settings technology transfers. These transfers include Web-based and legacy software applications performance management, development and applications of risk management processes, enterprise dashboard development, and statistical process control (SPC) applications to balance cost and technical performance.

Before joining Transdyne Corporation, she was a vice president of quality at SAIC and served as the chair of a process improvement team for an organization of over 9,000 employees across six geographically distributed business units—a virtual collaborative process improvement environment.

As a member of the International Research Consortium (IPRC), she is focusing her efforts on the impact to the global economy of technology development in small settings and on

implementing formal process improvement models to develop teaming alliances, as described on <http://transdynecorp.com>.

Sandra Salars

MEI Technologies

ssalars@houston.rr.com

Sandra Salars is CMMI project manager for MEI Technologies in Houston, Texas. She has successfully led two SCAMPI Maturity Level 5 appraisals in small settings. One appraisal was for a software development organization and the other was for a finance organization. The software development organization's SCAMPI included customer participation on the appraisal team.

Process improvement has been an area of special interest to Salars for the past fifteen years. It began with her participation in the Malcolm Baldrige assessment while working at IBM and then continued in her work as a software quality assurance engineer for Lockheed Martin and as the process improvement manager at SAIC. Sandra is currently supporting a Boeing CMMI process improvement initiative for the International Space Station Vehicle Organization.

6.5 Process Improvement in a Small Company

Author

James E. Jones

Abstract

Studies have shown that the Capability Maturity Model Integration (CMMI) based process improvement enables organizations to more consistently deliver better products and services on time, with higher quality, and for the predicted costs. Process improvement initiatives are impacted by many factors (e.g., financial resources, human resources, business objectives, culture of the organization, and vision). When starting a process improvement initiative, it is imperative to determine the appropriate scope and tasking.

Understanding the advantages of using CMMI when working with the Department of Defense, as well as understanding the advantages over competitors, associate contractors, and subcontractors, Support Systems Associates, Incorporated (SSAI) has established a roadmap for implementing a Process Improvement Program (PIP) to improve its processes for better products and services using the International Organization for Standardization (ISO) 9001:2000 and the CMMI frameworks. This paper will discuss process improvement activities at the SSAI-Warner Robins office.

Background

The SEI, for the sake of process improvement, defines small settings as a company of fewer than approximately 100 people, an organization of fewer than approximately 50 people, and a project of fewer than approximately 20 people.

Three major investment elements are involved in CMMI-based improvement:

1. appraisal
2. definition/infrastructure support
3. deployment

Typically, small businesses have a disadvantage with resources for appraisal and definition but have a distinct advantage in deployment. A frequent misconception about implementing CMMI is that it works only for large organizations, i.e., its cost and complexity appear to make it impractical for small businesses to implement.

Increasingly, organizations are obtaining ISO 9001:2000 certifications and CMMI ratings to help set process improvement objectives and priorities, improve current processes, and provide guidance for ensuring stable, capable, and mature business practices. A study conducted by the SEI found that for every dollar invested in process improvement,

organizations reaped benefits averaging from four to eight dollars in savings or reduced costs primarily attributable to the elimination of rework.

SSAI Profile

SSAI is a privately owned small business, which specializes in engineering, logistics, and management services to both the government and industry. Headquartered in Melbourne, Florida, SSAI has offices across the United States and employs over 300 people. The Certification Body of TUV American Inc. Management Service Division has certified that SSAI (Melbourne, Warner Robins, and Mary Ester offices) has implemented a quality management system in accordance with ISO 9001:2000 for engineering and logistics services associated with the design, development, and sustainment of aerospace and electronic systems for government and commercial customers.

The SSAI-Warner Robins office provides professional engineering and logistics support services to Air Force, Special Operations Forces (SOF), and Army customers and is a small business prime contractor. Figure 43 depicts a typical engineering life cycle model.

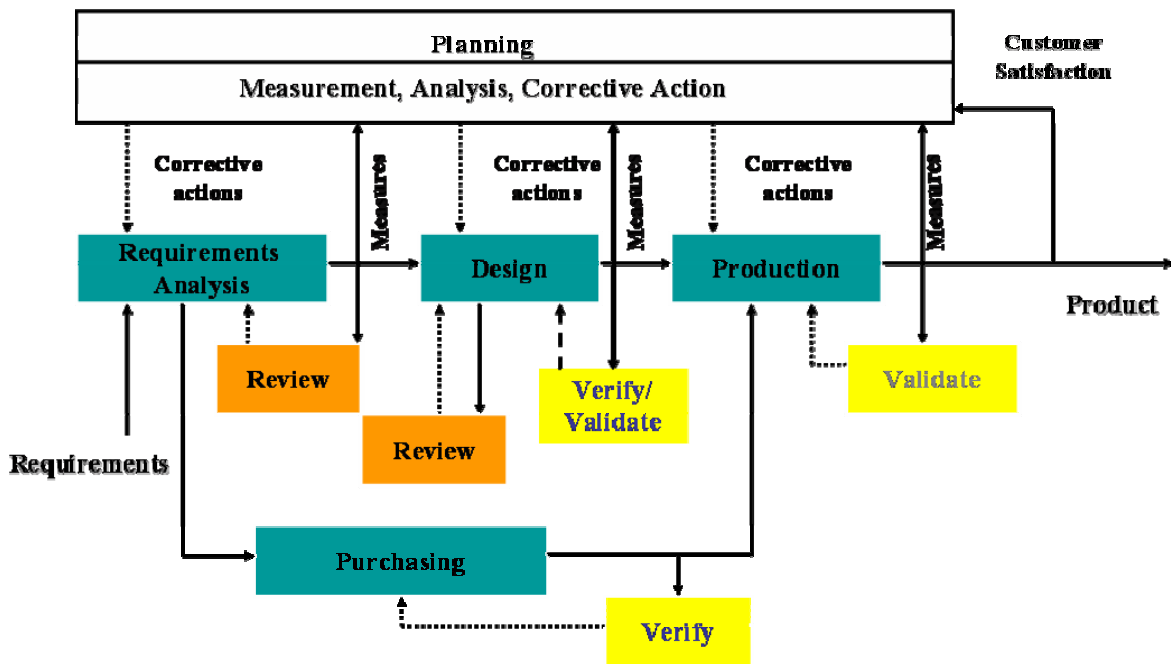


Figure 43: Typical Engineering Life Cycle Model

Examples of contract vehicles include

- flexible acquisition and sustainment tool (FAST)
- SOF support services contract II (SSSC II)
- design and engineering support program II (DESP II)
- aircraft systems engineering support (ASES)

SSAI-Warner Robins also designs and develops aerospace systems for government and commercial customers. Some specific engineering product examples include the following: AC-130H Gunship Trainer, MC-130P Electronic Load Analysis, AC-130H Digital Map Interface (DMIS), and MC-130E Radar Boresight Alignment.

As a small business prime contractor, SSAI-Warner Robins performs engineering services and delivers products under contract vehicles that require small integrated product teams (5-10 people) for 6 to 18 month durations. These tasks involve program management, project management, systems engineering, software engineering, quality assurance, configuration management, and administrative support.

CMMI Overview

CMMI places proven approaches into a structure that help organizations examine the effectiveness of their processes, establish priorities for improvement, and help implement these improvements. Figure 44 illustrates CMMI in a nutshell.

There are two CMMI model representations and one appraisal method. An organization needs to decide which CMMI model best fits the organization's process improvement needs. The organization then selects a model representation and determines the bodies of knowledge to include in the model. The SEI provides the initiating, diagnosing, establishing, acting, and learning (IDEAL) model to guide development of a long-range, integrated plan for initiating and managing a process improvement program [McFeeley 96].

CMMI In A Nutshell

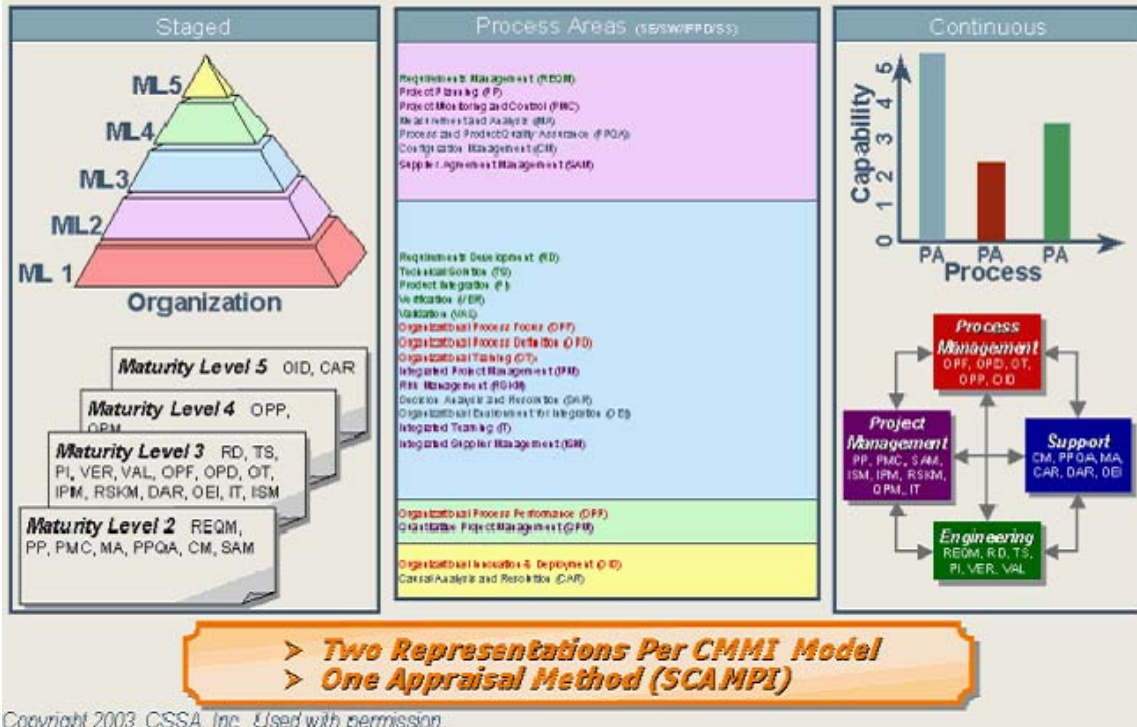


Figure 44: CMMI in a Nutshell

CMMI Model Representation

When selecting a CMMI model, there are two CMMI model representations available: continuous or staged. The CMMI model components of both the staged and continuous representations are process areas, specific goals, specific practices, generic goals, and generic practices. CMMI model components with a staged representation are illustrated in Figure 45.

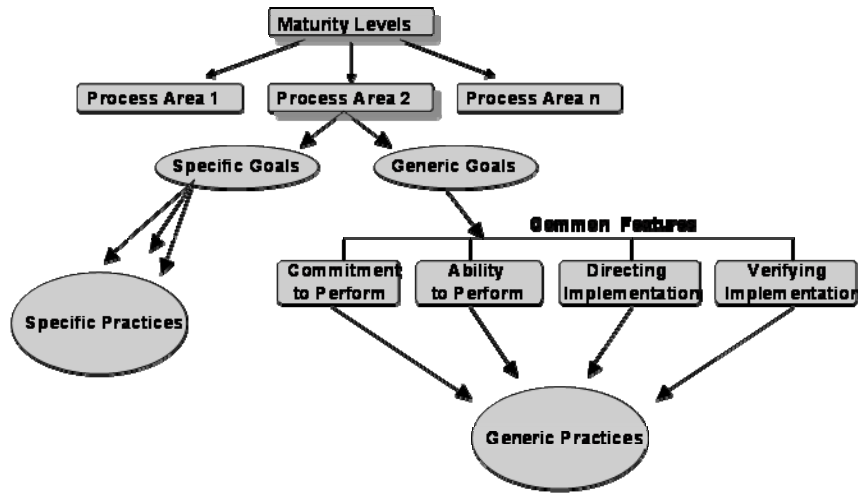


Figure 45: CMMI Model Components

Model Structure

As shown in Figure 46, the CMMI staged representation model has a number of process areas for each of the five maturity levels—initial, managed, defined, quantitatively managed, and optimizing. To attain a level of maturity, the specific and generic goals of all process areas in the maturity level must be attained.

Maturity Level	Focus	Process Areas
5: Optimizing	Continuous Process Improvement	Organizational Innovation and Deployment (OID) Causal Analysis and Resolution (CAR)
4: Quantitatively Managed	Quantitative Management	Organizational Process Performance (OPP) Quantitative Project Management (QPM)
3: Defined	Process Standardization	Requirements Development (RD) Technical Solution (TS) Product Integration (PI) Verification (VER) Validation (VAL) Organizational Process Focus (OPF) Organizational Process Definition (OPD) Organizational Training (OT) Integrated Project Management (IPM) Risk Management (RSKM) Integrated Teaming (IT) Integrated Supplier Management (ISM) Decision Analysis and Resolution (DAR) Organizational Environments for Integration (DEI)
2: Managed	Basic Project Management	Requirements Management (REQM) Project Planning (PP) Project Monitoring and Control (PMC) Supplier Agreement Management (SAM) Measurement and Analysis (MA) Process and Product Quality Assurance (PPQA) Configuration Management (CM)
1: Initial		

- Quality Productivity
- Reduction in time-to-market
- Reduction in post delivery defects

Risk Rework

Figure 46: CMMI Staged Representation Model

The CMMI staged representation model also permits comparisons across and among organizations by using maturity levels. It will also provide a single rating that summarizes appraisal results and allows comparisons among organizations.

Bodies of Knowledge

There are four Bodies of Knowledge available to include in the model: Systems Engineering (SE), Software Engineering (SW), Integrated Product and Process Development (IPPD), and Supplier Sourcing (SS). When selecting bodies of knowledge (disciplines) for the model, the model will contain a set of process areas. The process areas are grouped into four categories: Process Management, Project Management, Engineering, and Support. Figure 47 depicts the CMMI model structure. When selecting the SE and SW disciplines, the model will contain the Process Management, Project Management, Engineering, and Support process areas.

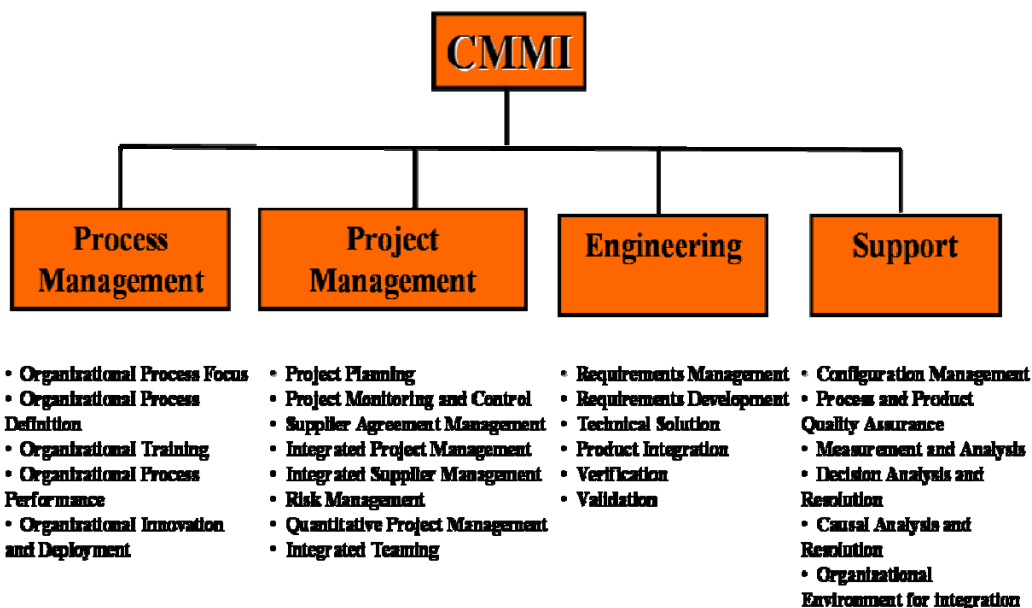


Figure 47: CMMI Model Structure

Appraisal

A number of organizations are using the Standard CMMI Appraisal Method for Process Improvement (SCAMPI) [SEI 01a] to focus on identifying improvement opportunities based upon the CMMI model. These organizations are using the SCAMPI appraisal method in accordance with the Appraisal Requirements for CMMI (ARC) [SEI 01b] to identify opportunities for improvement and to either achieve a specific maturity level or the satisfaction of a process area.

The ARC specifies the following requirements for CMMI appraisal method:

- responsibilities
- appraisal method documentation

- planning and preparing for the appraisal
- appraisal data collection
- data consolidation and validation
- rating
- reporting results

The ARC identifies three CMMI appraisal method classes (A, B, and C). The ARC provides the characteristics for each class (i.e., amount of objective evidence, rating generated, resource needs, team size, and appraisal team leader requirements).

During a SCAMPI Class A appraisal, projects must provide data sources (at least one direct plus one indirect artifact/affirmation) for each CMMI specific/generic practice within scope. Table 35 defines the indicator type.

Table 35: Practice Implementation Indicators (PII)

Indicator Type	Description	Examples
Direct artifacts	Tangible outputs resulting directly from a practice	Documents Deliverable products
Indirect artifacts	Artifacts that are a consequence of performing a practice	Meeting minutes Review results
Affirmations	Oral or written statements conforming or supporting implementation of a practice	Questionnaire response Interviews Presentations

The SCAMPI appraisal method is verification-based, requiring extensive objective evidence. Decisions on practice implementation and ratings are made based upon aggregated objective evidence. In accordance with the ARC, a CMMI appraisal rating is generated only when a SCAMPI Class A appraisal is lead by an SEI-authorized SCAMPI Lead Appraiser.

One large organization reported that a total of 2, 486 artifacts were provided for four projects to achieve a CMMI maturity level 3 rating. An Air Force organization reported using five SEI-authorized SCAMPI Lead Appraisers with nine appraisers for eight days and 80 participants for four projects to achieve a CMMI level 5 rating. Several large organizations have reported 24–36 months preparation time to achieve a SCAMPI Class A CMMI Level 3 rating.

Achieving a SCAMPI Class A CMMI rating is a labor-intensive effort, extensive objective evidence, and sources of evidence are required.

Process Improvement Approach

Quality, product technology, requirements instability, and complexity are the critical engineering process issues. Process improvement is a significant undertaking that involves three major investment elements. The initiative is impacted by many factors such as resources, business objective, and vision. When starting a process improvement initiative, it is imperative to determine the appropriate scope and tasking.

This section addresses SSAI’s approach to achieve mission success through model selection, appraisal selection, and problem solving process selection

CMMI Model Selection

Based upon process improvement objectives and priorities and its current ISO processes, SSAI first decided which CMMI model best suited the organization’s process-improvement needs and determined the bodies of knowledge to include in the model. The following paragraphs discuss the CMMI model representation, bodies of knowledge, and problem solving process selection.

Representation Selection

SSAI selected the CMMI Staged Representation model to provide a sequence of improvements, beginning with basic management practices and progressing through a predefined and proven path of successive levels, with each serving as a foundation for the next. This representation focuses on best practices that SSAI can use to improve in the process areas that are within the maturity level it chooses to achieve.

Bodies of Knowledge Selection

Before an organization can begin using the CMMI staged representation model for improving processes, the organization must determine which integrated model to select based upon the CMMI bodies of knowledge. As shown in Table 36, SSAI mapped the prime contracts–programs (FAST, SSSC II, DESP II, and ASES)–to the CMMI four bodies of knowledge to determine which integrated CMMI model to choose to enable the organization to control process improvement.

Table 36: SSAI Bodies of Knowledge Mapping

Programs	CMMI Bodies of Knowledge			
	SE	SW	IPPD	SS
<i>FAST</i>	X	X		X
<i>SSSC II</i>	X	X		

DESP II	X	X		X
ASES	X	X		X

Based upon priorities, business and process-improvement objectives, SSAI selected the *CMMI SE and SW, Staged Representation Model (CMMI-SE/SW, V1.1, Staged)* to use to focus process improvement opportunities. Figure 48 illustrates SSAI CMMI Model Structure.

To identify candidate projects to participate in the CMMI baseline appraisal (gap analysis) activity and to help SSAI track the organization’s level of conformance to the CMMI model, SSAI mapped the projects for the programs that were mapped to the SE and SW bodies of knowledge to the CMMI process areas for maturity level 2–Managed. SSAI selected three key projects to give a broad cross-section of the systems engineering and software engineering bodies of knowledge that span across the engineering life cycle model shown in Figure 43.

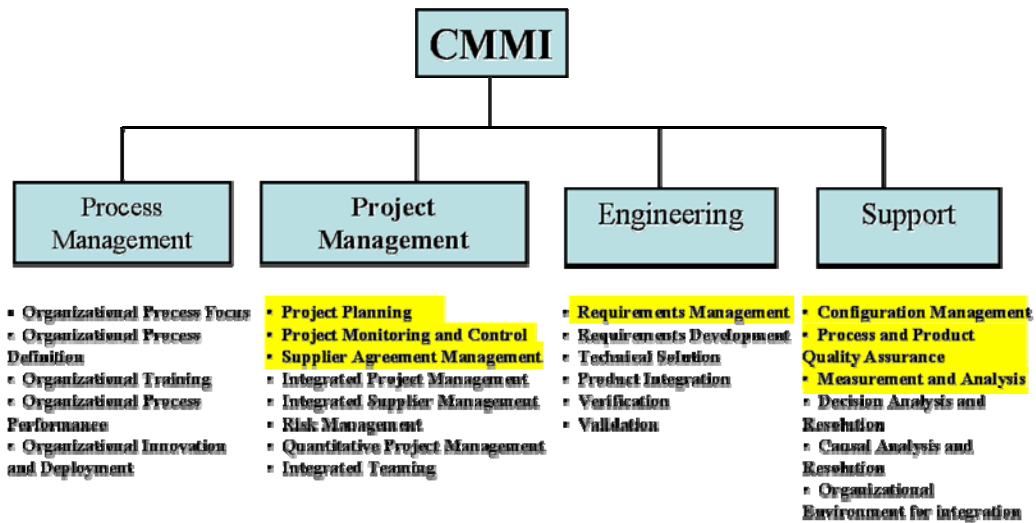


Figure 48: SSAI CMMI Model Structure, Maturity Level 2 (CMMI-SE/SW, V1.1, Staged)

Appraisal Method Selection

SSAI selected the SCAMPI Class A appraisal method to obtain a benchmark quality rating relative to the selected CMMI model, (*CMMI-SE/SW, V1.1, Staged*), to focus on identifying improvement opportunities. SSAI selected the SCAMPI method based upon the following essential characteristics:

- accuracy
- repeatability
- cost/resource effectiveness

- meaningfulness of results
- ARC compliance

SSAI will use the SCAMPI Class A appraisal to gain insight into its project management, engineering, and support capabilities by identifying the strengths and improvement opportunities of its current ISO 9001: 2000 processes relative to the CMMI model. The SCAMPI Class A appraisal will enable SSAI to prioritize improvement plans, focus on improvements that are most beneficial, derive a maturity level rating, and identify risks relative to maturity determinations.

Problem Solving Process Selection

Organizations are using several problem solving processes such as Plan-Do-Check-Act (PDCA), ISO 9004:2000, and IDEAL to implement CMMI. SSAI process improvement approach is driven by business objectives (e.g., customer satisfaction, productivity, revenue growth, etc.) using the IDEAL problem solving process with ISO 9001:2000 and CMMI frameworks. As shown in Figure 49, SSAI selected the IDEAL model as a framework to guide development of a long-range, integrated plan for initiating and managing the PIP.

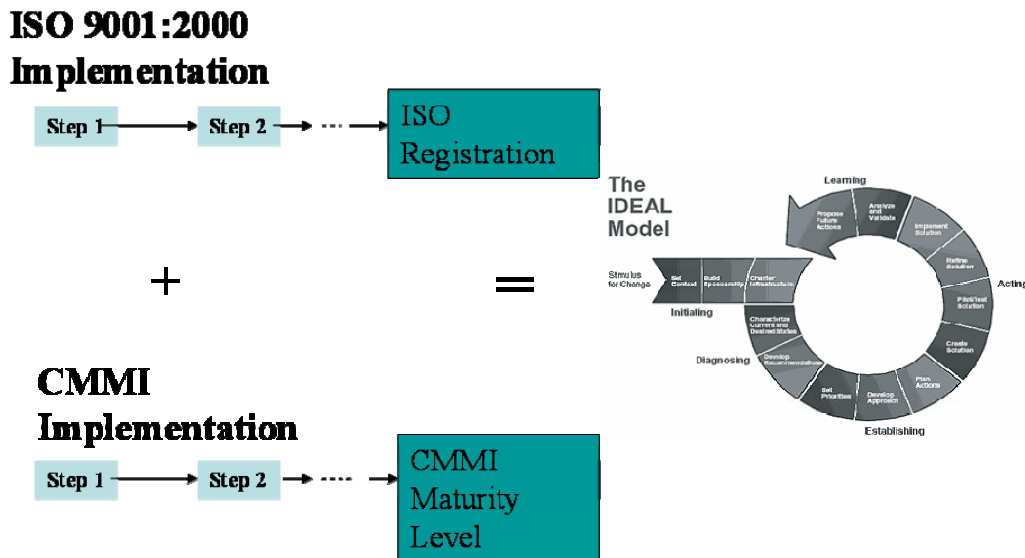


Figure 49: Problem Solving Process

Organization for Implementing CMMI

SSAI understands that process improvement is a significant undertaking that requires senior level management sponsorship and firm commitment of resources to be successful. To achieve that goal, SSAI established a three-level organizational infrastructure. Level 1– Management Steering Group (MSG) is chaired by the vice president of Operations and co-chaired by the vice president–Air Force Administration. Members consist of vice president - Programs, director of Engineering, and program managers. Senior-management sponsorship

is a key factor in the appraisal process. At Level 2, SSAI established a technically competent process group to guide process improvement efforts—an Engineering Process Group (EPG) whose members consist of project management, systems engineering, software engineering, quality assurance, and configuration management. An experienced process improvement change agent also functions as a process improvement mentor and chairs the EPG. The Level 3 - Technical Working Groups (TWG) are established as needed to get practitioners involved in developing new processes and procedures. The TWG will exist for specific process areas and may have three to five members and is chaired by an EPG member.

It is important to note that the infrastructure set up to accomplish the PIP will play a significant role in the success or failure of the PIP. The value that the infrastructure brings to the PIP, along with the understanding of its roles and responsibilities, cannot be underestimated.

Process Improvement Life Cycle Model

Using the IDEAL model as a guide, SSAI established the Process Improvement Life Cycle (PILC) model as depicted in Figure 50. This model outlines a series of seven phases to improve products and services quality, productivity, and process maturity. Planning, measurement, analysis, and corrective action activities will be implemented throughout the entire life cycle. As shown in Figure 50, to ensure success, a series of activities are performed prior to the SCAMPI Class A appraisal.

During each phase, the EPG will communicate the status to the organization and will review progress with the MSG. After our initiating phase to establish the organization infrastructure, understand the CMMI implementation requirements, and attend training (SEI-authorized *Introduction to the CMMI*), SSAI will perform a CMMI baseline appraisal (gap analysis). A gap analysis phase will be performed to identify specific deficiencies in implementing practices relative to the CMMI model. During the gap analysis phase, SSAI will perform the following activities:

- map programs to the CMMI bodies of knowledge
- select candidate projects
- prepare a process improvement indicator (PII) tool
- prepare the projects for the gap analysis
- execute the gap analysis

SSAI will use an SEI-authorized SCAMPI Lead Appraiser to execute the gap analysis. For the selected projects, prior to executing the gap analysis, the PII tool will be populated with practice artifacts (direct artifacts, indirect artifacts, and affirmations) for the selected process areas. These artifacts will provide evidence that gives a basis for verification of implementing the CMMI practice.

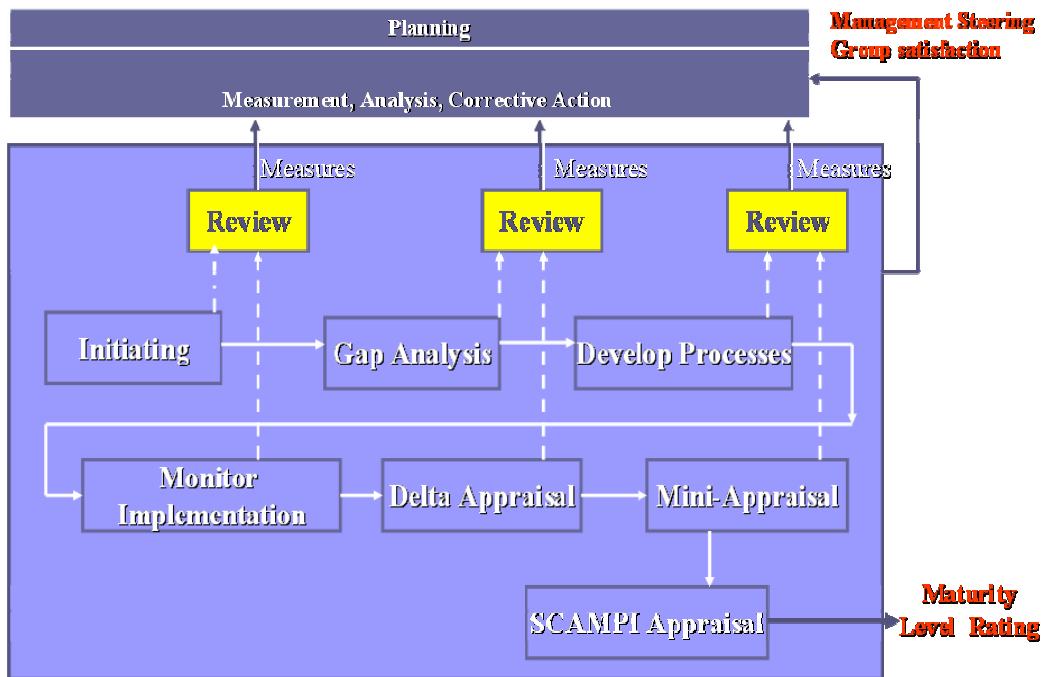


Figure 50: Process Improvement Life Cycle Model

During the gap analysis appraisal execution, the PII tool will be used to characterize the practices. Based upon the practice implementation data for a process instantiation, the appraisal team will use the PII tool to assign values to characterize the extent to which the CMMI model practice is implemented. Each practice will be characterized as fully implemented (FI), largely implemented (LI), partially implemented (PI), or not implemented (NI). Observations that can later be exported as strengths or improvement opportunities are captured. Upon assigning characterization values for a given model practice for each instantiation, the PII tool will be used to capture aggregated characterization values for the organization using the appraisal team consensus. The PII tool also will be used to establish the process area generic and specific goals rating and provide the appraisal results.

After the gap analysis, SSAI will commence the develop processes phase to establish an action plan in accordance with the Project Management ISO process to implement the opportunities for improvement. TWGs will be established to implement the action plan. After the opportunities for improvement are developed and baselined, SSAI will enter the monitor implementation phase. During this phase, SSAI will conduct the following activities: establish the process assists library, conduct process training, monitor the TWG activities, review commitments and changes to commitments with the management steering group, track implementation progress, collect and analysis metrics, and determine readiness for delta appraisals.

As shown in Figure 50, SSAI will conduct a series of informal appraisals prior to executing a SCAMPI Class A appraisal to achieve a CMMI maturity level 2 rating. During each informal appraisal, SSAI will use an SEI-authorized SCAMPI Lead Appraiser.

Strengths and Weakness

Strengths (SSAI)

SSAI has the following strengths:

- currently ISO 9001:2000 certified
- organization ownership of ISO 9001:2000
- process improvement expert as change agent
- understands the advantage of CMMI
- understands CMMI requirements
- understands SCAMPI Class A appraisal
- process improvement life cycle model selected
- use of gap analysis/PII and mini-appraisals prior to the formal SCAMPI Class A appraisal.

Weaknesses (SSAI)

- full understanding of CMMI commitment (e.g., resources, funding, schedule, etc.)
- understanding activities to perform process definition

Strengths (CMMI Products)

- CMMI implementation well documented
- integrated model (bodies of knowledge)
- CMMI models adaptable for small settings
- standards for conducting CMMI appraisal established
- CMMI model provides typical work products

Weaknesses (CMMI Model)

- CMMI model multiple representations: continuous and staged with multiple ratings: capability level and maturity level

Important Topics for the Research Community

The following are the most important topics for the research community to address in the future:

- How to implement process improvement using CMMI in small settings. Addressing the following topics:
 - commitment to perform–policy for implementing CMMI
 - ability to perform–responsibilities for implementing CMMI, adequate resources and funding, and participants trained in implementing CMMI

- measurement and analysis
- activities performed
- verifying implementation
- Identify critical factors affecting process improvement initiatives in small settings. Addressing the following topics:
 - financial resources
 - human resources
 - corporate political pressure
 - business objectives
- Establishing *one* CMMI representation model and rating.
- Develop a guidebook for implementing CMMI in small settings
- Develop a guidebook for a successful SCAMPI appraisal (i.e., organization roles and responsibilities, its intended use, tailoring options, and project management requirements) in small settings.

References/Bibliography

- [McFeeley 96]** McFeeley, Robert. *IDEAL: A User's Guide for Software Process Improvement* (CMU/SEI-96-HB-001, ADA305472). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1996.
<http://www.sei.cmu.edu/publications/documents/96.reports/96.hb.001.html>
- [SEI 01a]** CMMI Product Development Team. *SCAMPI V1.1, Standard CMMI Assessment Method for Process Improvement: Method Definition Document* (CMU/SEI-2001-HB-001, ADA3399204). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, December 2001.
<http://www.sei.cmu.edu/publications/documents/00.reports/00tr009.html>
- [SEI 01b]** CMMI Product Team. *Appraisal Requirements for CMMI, Version 1.1 (ARC, V1.1)* (CMU/SEI-2001-TR-034, ADA3399208). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, December 2001.
<http://www.sei.cmu.edu/publications/documents/01.reports/01tr034.html>
- [SEI 02]** CMMI Product Team. *CMMI for Systems Engineering/Software Engineering/Integrated Product and Process Development, Version 1.1, Staged Representation (CMMI-SE/SW/IPPD/SS, V1.1, Staged)* (CMU/SEI-2002-TR-012, ADA339907). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2002.
<http://www.sei.cmu.edu/publications/documents/02.reports/02tr012.html>

Biography

James E. Jones

Support Systems Associates, Incorporated

Warner Robins, Georgia

jejones1@alltel.net

James E. Jones is the project manager at SSAI-Warner Robins for the ASES C-130 Avionics Modernization Program (AMP) Engineering Support Task. He provides independent validation and verification (IV&V) engineering support to the United States Air Force at Warner Robins Air Logistics Center for the acquisition of the C-130 aircraft modernization. Jones has 37 years of software system experience in process improvement, project management, systems integration, software acquisition management, independent validation and verification, software development, and software sustainment. With more than 20 years process improvement experience, Jones was designated the engineering process group (EPG) lead at SSAI-Warner Robins. Prior to SSAI, Jones' process improvement experience was with several major defense contractors and organizations. He served as the Software Engineering Process Group (SEPG) manager at Lockheed Martin Communications Systems and as a Capability Maturity Model consultant for several Lockheed Martin organizations, as well as the Federal Aviation Administration (FAA). Jones served as an SEPG member for United Defense, LP Armament Systems Division, developing and training personnel on organization-wide processes.

Jones provided software acquisition management support for Lockheed Martin and the FAA. He served as the software subject matter expert for contract terminations. He is the co-inventor of the Stromberg-Carson's Community Office Switching System (United States Patent: 4,451,702; 4,479,034).

7 Workshop Highlights

This section provides high level summaries of several of the events of the workshop, including the keynote speaker's presentation, summaries of the individual breakout sessions, and the discussant's reflections on the workshop content.

7.1 Keynote

Claude Laporte, professor at University of Quebec's Ecole de Technologie Superieure and Project Editor for the new ISO SC7/Working Group 24, was the keynote speaker for the workshop. He provided an overview (see his paper in this report titled "Applying Software Engineering Standards in Small Settings: Recent Historical Perspectives and Initial Achievements") of the background and current plans for ISO SC7's new Working Group 24, which is currently called "Life Cycles for Very Small Enterprises."

In particular, Claude cited the need for the contents for the eventual guidebook to be extensively reviewed by the Very Small Enterprise (VSE) community. Given the expense of sponsoring ISO working group members, many developing nations do not participate as official members. Yet, these very countries are where software engineering is becoming an anchor for future economies. To address this, he is planning to host a Web site outside the formal ISO structure to preview work going on in the Working Group with interested reviewers around the world.

7.2 Breakout Sessions Summary

7.2.1 Day 1

7.2.1.1 Group A: Cost/Benefit Models

Group 1-A focused on types of business cases needed to support three different categories of process improvement (PI) motivation:

- Before the crisis: elements of a business case that would be useful for a small enterprise that is not in a crisis mode with relation to their business
- After the crisis: elements of a business case that would be useful for a small enterprise that is experiencing (or has recently experienced) a crisis that threatens the business

- Market/regulatory gate: elements of a business case that would be useful for small enterprises operating in a market that requires certification of good practices of some sort or in a regulated market that demands certification against one or more sets of practices

See the appendix for more details on what this group came up with to support the PI business case in these three contexts.

7.2.1.2 Group B: Success Stories

Group 1-B had three objectives:

- draft a taxonomy of success factors for PI in small settings
- capture success stories on how PI was “sold” in small settings
- discuss what is different about PI in large/small settings

Details of the results of this group's effort are found in the appendix.

One of the things proposed by this group was that a template could be provided to the community that would allow a comparison of case study examples. The following are suggested elements to include in the case study:

- Context: domain, size, experience with PI, and any other context factors that would help an organization understand its situation as it begins its improvement effort
- Initial reaction: initial reaction of the organization to proposals related to implementing process improvement
- Approach to overcome initial negative reaction: assuming that the first reaction to PI proposals is negative, an explanation of what was done (pilots, bringing in other executives, etc.) to overcome this reaction
- Actions: what actions were actually taken to implement improvements
- Results: qualitative and/or quantitative results from the effort, preferably from the viewpoint of the executive sponsor
- Critical success factors: the factors that the participants believe contributed most concretely to the organization’s adoption success

7.2.1.3 Group C: Regional Support Centers

Group 1-C focused on capturing the experiences of regional support (usually provided by governments) represented by the attendees of the workshop. The regions represented/discussed included the following:

- Malaysia: the model here is to provide extra funding to cover consulting, appraisal, and training costs for companies that embark on a CMM-based PI effort; 40 companies have been supported by this project so far.

- Ireland: Enterprise Ireland, a government economic agency, is extending techniques it has used to stimulate improvement activities in other industries into the software/IT space, primarily focusing on providing three different levels of extra funding depending on the state of the company. Over 1200 companies (out of a potential 3500) have participated in this program to date.
- India: rather than providing funding up front, the Indian government provides tax incentives for companies that demonstrate improvement against a number of accepted models. They are finding this approach more successful than providing extra funding for training, etc.
- Hong Kong: the Chinese government funded 15 companies (out of 600 who applied) and provided expertise and training to support them in pursuing model-based improvement. Five of these companies achieved CMMI maturity level 2 during the course of the project (May 2003-Dec 2004).
- USA/Georgia: the state of Georgia has recently established an Aerospace Innovation Center that is intended to, among other things, incentivize small enterprises to engage in process improvement. The representative from this organization was thrilled to get such diverse perspectives to help them get going!

Details of several of these initiatives are found in the presented papers. There also are papers in the Regional Support section that are not represented here because participants in those regional activities (e.g., Chile) participated in other breakout sessions.

Similar to Group 1-B, this group, after sharing their experiences informally, thought that a template that allowed government groups to share their context and approach would be useful; some of them may work together after the workshop to suggest such a template.

7.2.2 Day 2

7.2.2.1 Group A: Economical PI Infrastructure

Group 2-A differentiated infrastructure into the groups of internal (to the company) and external (e.g., government initiatives and supports). They primarily identified issues in establishing and sustaining internal infrastructure in small settings. Details on their thinking are in the appendix.

7.2.2.2 Group B: Cost/Benefit Models 2

In group 2-B, a subset of the team from group 1-A met to continue discussions about the three categories of business cases needed and potential elements to include. See the appendix for details of their ideas.

7.2.2.3 Group C: Current/Future Research Topics

Group 2-C focused primarily on future research topics that the group suggested are needed in this area, citing the body of work reflected in this and other conferences as evidence that current research is taking place.

They brainstormed several areas they thought would benefit from additional research:

- behavioral science studies on starting and sustaining improvement efforts
- intercultural studies
- establishing business cases at the process topic or process area level
- accelerators and inhibitors to PI within a supply chain

The following items became initial inputs into the Next Steps section of this report:

- current areas
- future areas
- behavioral science studies, starting and sustaining improvements
- intercultural studies
- business cases at the process area (PA) level
- accelerators and inhibitors in the supply chain
- volunteers needed to expand on next step topics

7.2.2.4 Group D: ISO/SC 7 Working Group 24 Input

See the appendix for notes about this session.

7.3 Discussant Reflections

Claude Laporte was asked to act in the role of discussant for the workshop, which committed him to summarize and/or challenge what he heard on the first day and a half of the workshop. This was done to help the participants identify possible gaps in the content being discussed and to provide a meta view of what has gone on in the workshop.

The following are a summary of his points:

- What about pre-defined frameworks? Could there be enough commonality in where people get started (i.e., the similarity in PAs attempted first in the presentations) to be beneficial in codifying?
- The summary of what SMEs want or need from Gene Miluk's applied ethnography study seems to be a good set of criteria to pay attention to when working in this sector
- Although there are differences in approach and techniques exhibited in the presentations, there are also many similarities—we need to work to exploit them.

- Why should small businesses have to understand our jargon? (This goes back to some of Gene Miluk's findings.)
- How do we communicate the value proposition of PI? ROI seems like it goes too far? Experience here indicates that business impact that is perceived as positive, even if qualitative in nature, is useful.
- It still appears to be the team or individual that makes the most difference in the performance of a small setting. How do we take advantage of that?
- The open sharing of this workshop has energized all the participants. How do we keep this kind of open exchange going?

8 Suggested Next Steps and Summary

8.1 Suggested Next Steps

Some of the next steps that were suggested at the end of the workshop include the following:

- create an online community support repository that would support researchers, consultants, and small settings considering process improvement
- create and support distance learning facilities and assets that would enable those in small settings to get high-quality process improvement training in ways that do not demand travel and large chunks of participants' dedicated time away from their primary worksite
- put together a list of potential sponsors for researchers in this area
- write a paper intended for publication that would define small settings from economic, practitioner, and academic points of view
- sponsor future workshops in this area of research

More detail on each of these is included in the following points.

8.1.1 Community Support

Actions suggested in this section include the creation of a publicly available area that would, primarily by asynchronous methods, allow interested parties and topics to connect with each other. The following are the three primary interest groups and their needs that were identified:

- **Researchers:** needs expressed here included linking to the WG24 community area, special interest group support areas, connections to SEI and other publications of interest to the research community, and a place for collecting and discussing regional and national level best practices, ROI, etc.
- **Consultants/PI Practitioners:** needs expressed here are support for identifying optimal effort to apply PI in small settings, guidance on effective approaches for appraisal in small settings, and guidance for implementing high maturity concepts in small settings
- **"New to PI":** essentially a potential PI sponsors area, the needs expressed here were primarily around linking business value to process improvement in the small settings context and identifying the business case for adopting different practices or processes

8.1.2 Online Training/Distance Learning

One of the perennial issues for process improvement in small settings is that people working in those settings rarely have resources (either money or time) to spend large chunks of dedicated time away from their primary work site. Many of the PI approaches that are successful in larger settings cause a tremendous burden on small settings and are one of the oft-cited reasons for not getting involved in process improvement.

A solution for this particular issue that is starting to be adapted for this environment is distance learning. The SEI is currently experimenting with the application of blended learning technologies and awareness level training for process improvement. Individuals in small settings would be natural users of such training.

8.1.3 Finding Research Sponsors

In some regions (e.g., Europe) there are well-known funding approaches (e.g., research projects sponsored by the European Union) that could be applied to research in small settings. Mechanisms for other regions are not as well known. One possible solution would be to develop a contacts database for different countries/regions.

8.1.4 Publications

Some of the ideas expressed in the publications topic included the following:

- Identify particular journals (e.g., SPIP) that would be amenable to publishing special issues on PI small settings.
- Create a paper intended for publication to stimulate discussion on useful definitions of small settings from at least the economic, practitioner, and academic settings. During the workshop, a brief discussion of what different countries considered *small* or *medium* identified very different mental models that could affect the utility of proposed solutions if definitions are not commonly understood or at least explicitly articulated!
- Create some kind of technical report (possibly as an input to ISO SC7/WG 24) on the diversity of regional support models for process improvement in small settings. This report would cover different models of economic motivation as well as different approaches to government involvement.

8.1.5 Sponsoring/Hosting Future Workshops

Some ideas for topics to feature at future workshop included the following:

- contact someone from the Microsoft Solutions Framework team to discuss their experience with users in small settings
- select one or two large customers to talk about their issues and solutions in working with small suppliers

- include themes as an explicit part of the solicitation; one in particular that was suggested was social and cultural aspects and differences in implementing PI in small settings

8.1.6 What's Happening Now/Near term

One next step that the SEI has taken, primarily in response to this workshop, was to reframe the *Applying CMMI in Small Settings* project to *Improving Processes in Small Settings*. This reframing recognizes that the model used in small settings does not seem to be as big an issue as some of the infrastructure and other issues that are common in these settings.

Another internal step that the SEI performed was to create a publicly accessible area in the SEI's collaboration workspace, BSCW. This workspace allows participants of the workshop and other interested parties to start sharing assets and develop publications, as well as to post and engage in discussions on relevant topics. To gain access to this workspace, please send an email to Keith Kost at kkost@sei.cmu.edu.

A next step that took place outside of the SEI was covered in the keynote for the workshop. This step was the formation of Working Group 24 of Subcommittee 7 of the International Standards Organization (ISO). The title of the working group is "Life Cycles for Very Small Enterprises," and its intent at this point (some of this could change as the committee actually forms and starts work) is to leverage existing ISO standards to provide guidance for small enterprises on how to leverage their work in the standards area to give them the greatest benefit. Claude Laporte, the Project Editor for Working Group 24 and the workshop keynote speaker, is planning a Web site that will allow anyone to follow the work of the Working Group and provide informal feedback. This approach is being taken to encourage people actively working in small settings to provide feedback on the work without incurring the expense typical of participation in face-to-face ISO meetings.

8.2 Summary

Both the format and content of the workshop were generally well received by the participants. The breadth and depth of papers submitted indicates a high level of interest and activity both in the research community and within industry across a broad spectrum of geographic regions and domains.

Much of the literature (see the paper "Critical Success Factors (CSF) in SPI Bibliography" in this report) up to this point has focused on issues and barriers to process improvement in small settings. It was encouraging to see that a significant portion of the papers presented here showed some aspect of successful implementation of process improvement in small settings although those successes faced many of the same issues and challenges recorded in earlier literature.

This group had several classes of ideas for future work in this area. Although the SEI is the primary sponsor of the workshop, it was not assumed that all the next steps were meant to be

for the SEI, and the participants were enthusiastic about creating a community that would collaborate actively to leverage the scarce resources that are available to research this area.

We hope that the results of this workshop will be considered seminal to a far-reaching, well-supported worldwide research focus for the sector of business that, regardless of region, is the largest in most economies, and the hardest to reach when addressing many issues, including process improvement.

We would once again like to thank all of the participants, reviewers, event staff, writers, and facilitators who made this event a success. We look forward to eclipsing this success in future workshops!

Appendix Workshop Breakout Session Results

Day 1 Sessions

Group A: Cost/Benefit Models

Facilitator

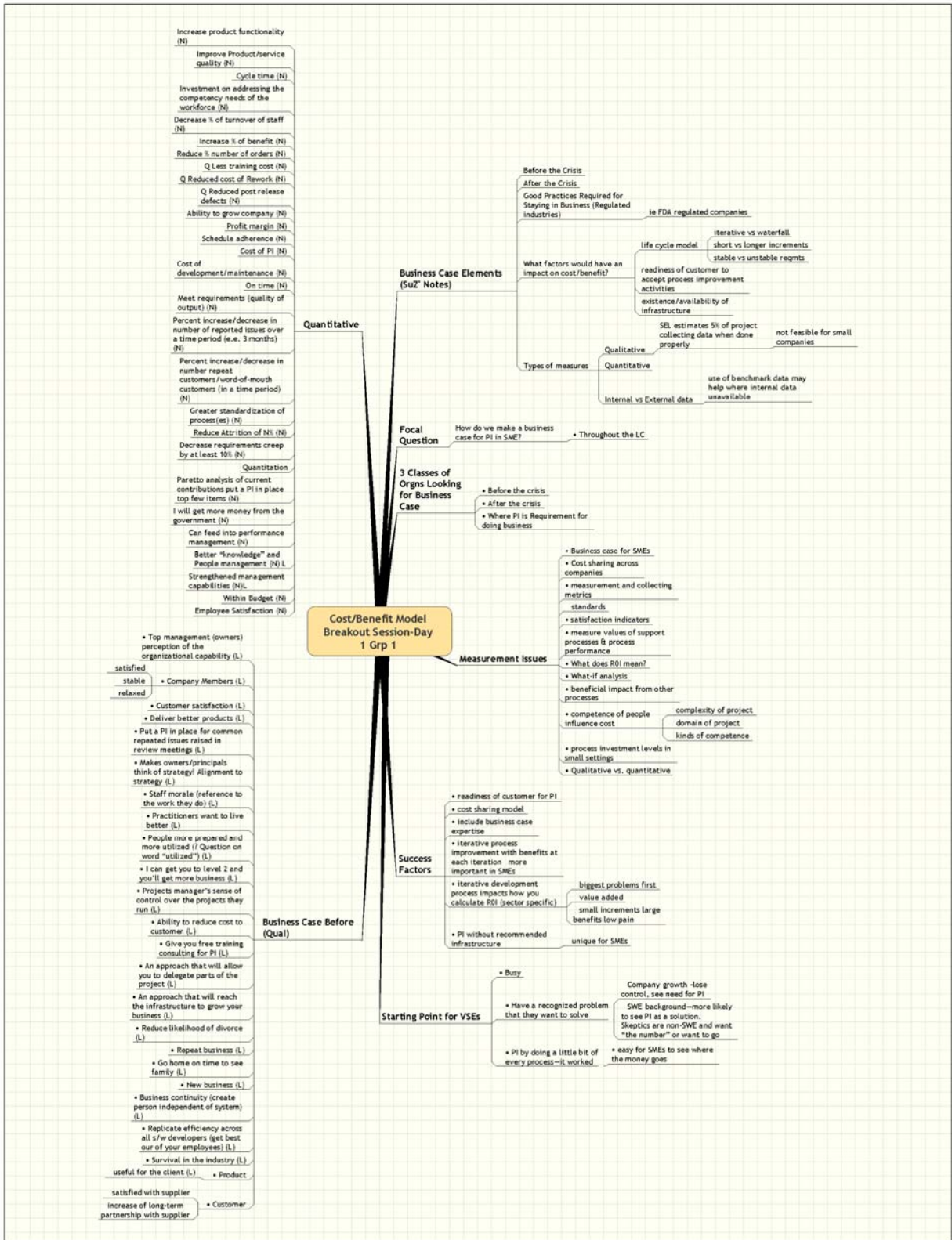
Caroline Graettinger

Participants

Jose A. Calvo-Manzano, Christian Carmody, Jose Antonio Cerrade, Gonzalo Cuevas Agustín, Khaled El Emam, Javier García, Diane Gibson, Dennis Goldenson, K. Niranjan Kumar, Oscar A. Mondragon, Anil Revankar, Tomás San Feliu, Rajneesh Sharma, Katsutoshi Shintani, Dorai Sinna, Karin Steembecker, and Ng Wan Peng

Flip Chart Notes

See the following Mind Map.



Group B: Success Stories

Facilitator

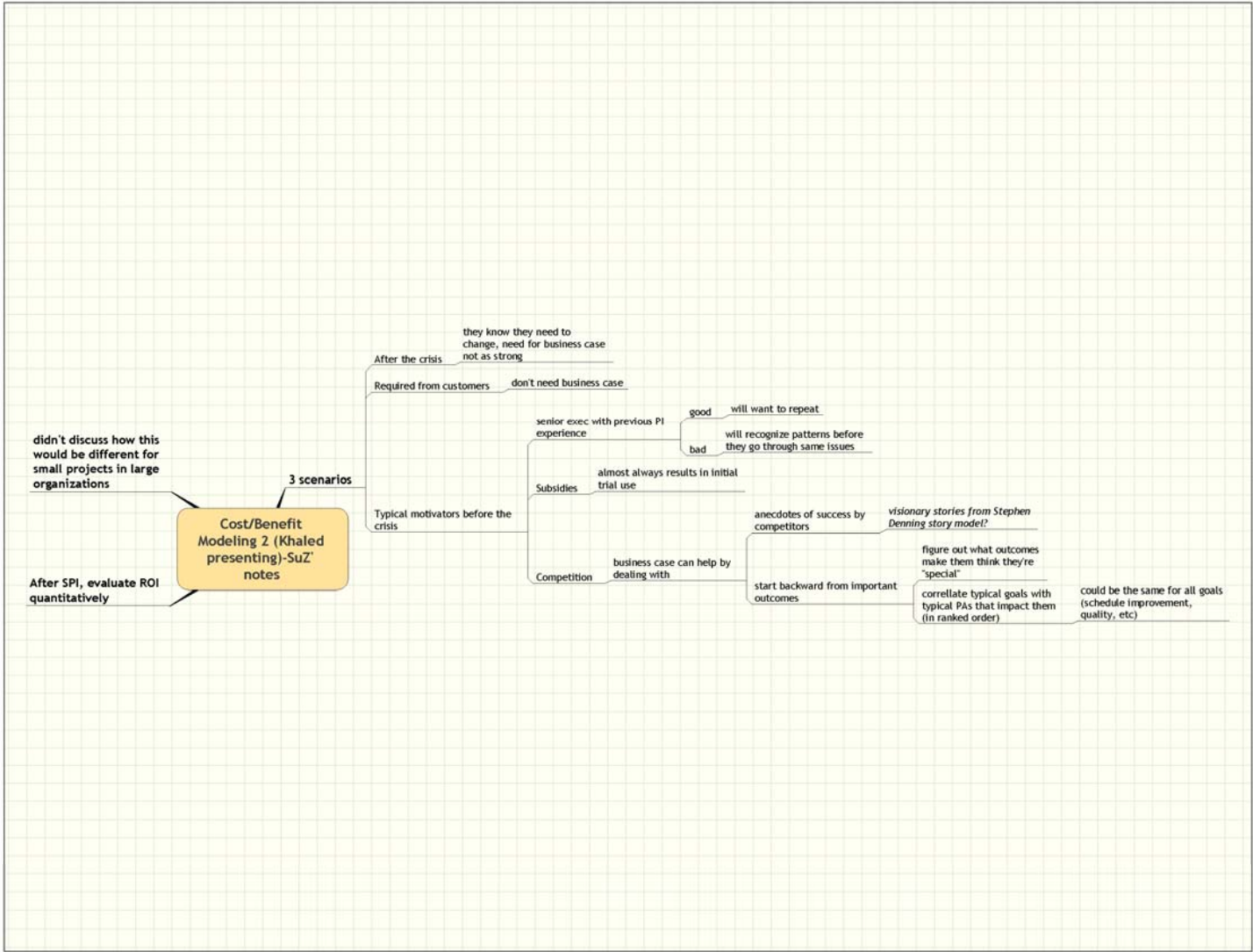
Agapi Svolou

Participants

Rosario Blowers, Nazrina Khurshid Mohamed, V. Sathya Murthy, Stephen Rupp, Angela Tuffley, and Bosheng Zhou

Flip Chart Notes

See the following Mind Map.



Group C: Regional Support Centers

Facilitator

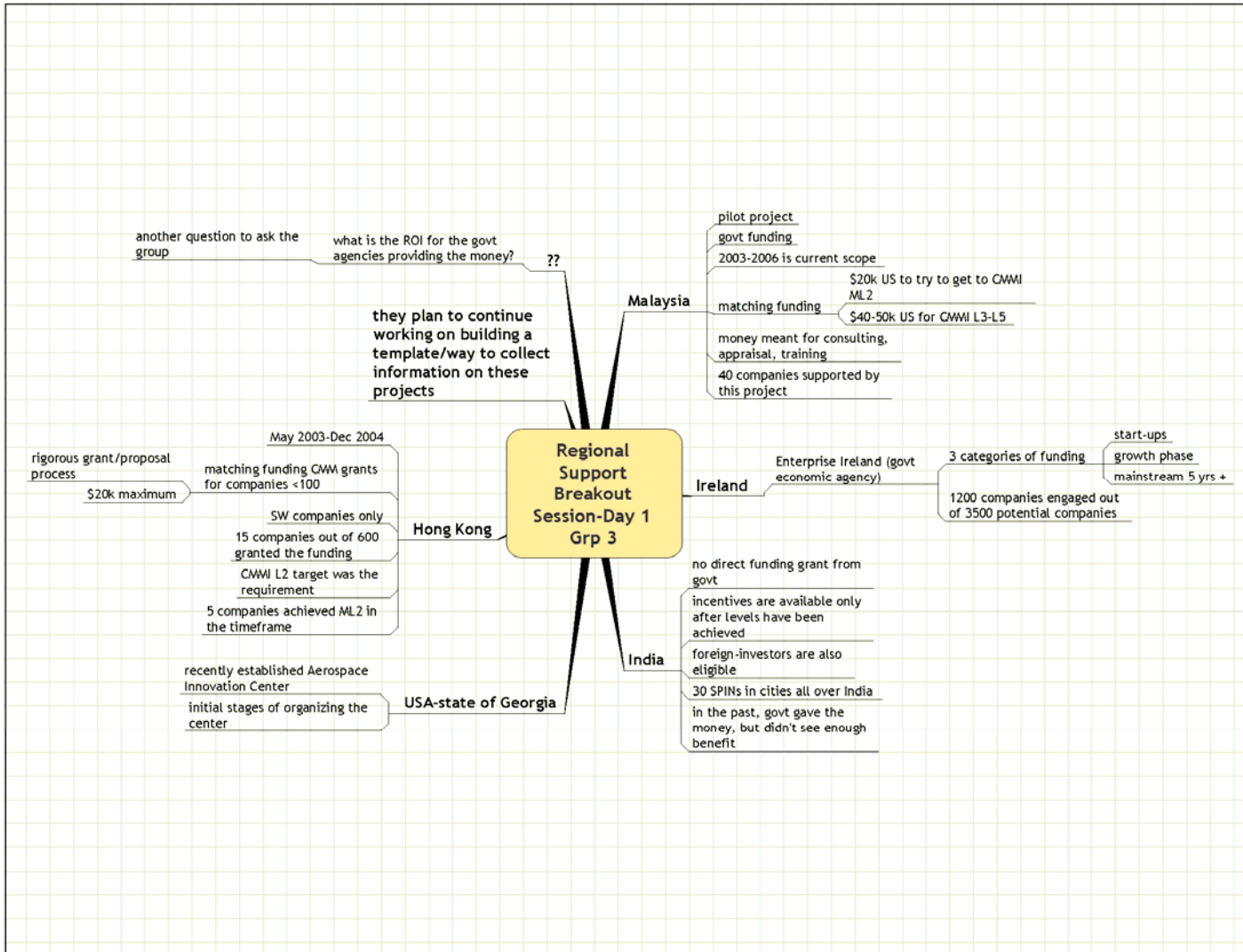
Miguel A. Serrano

Participants

Zaijun Hu, James E. Jones, Yvette Lui, Venkata Muralidhar Nallagonda, and Karl Ng Kah Hou

Flip Chart Notes

See the following Mind Map.



Day 2 Sessions

Group A: Economical PI Infrastructure

Facilitator

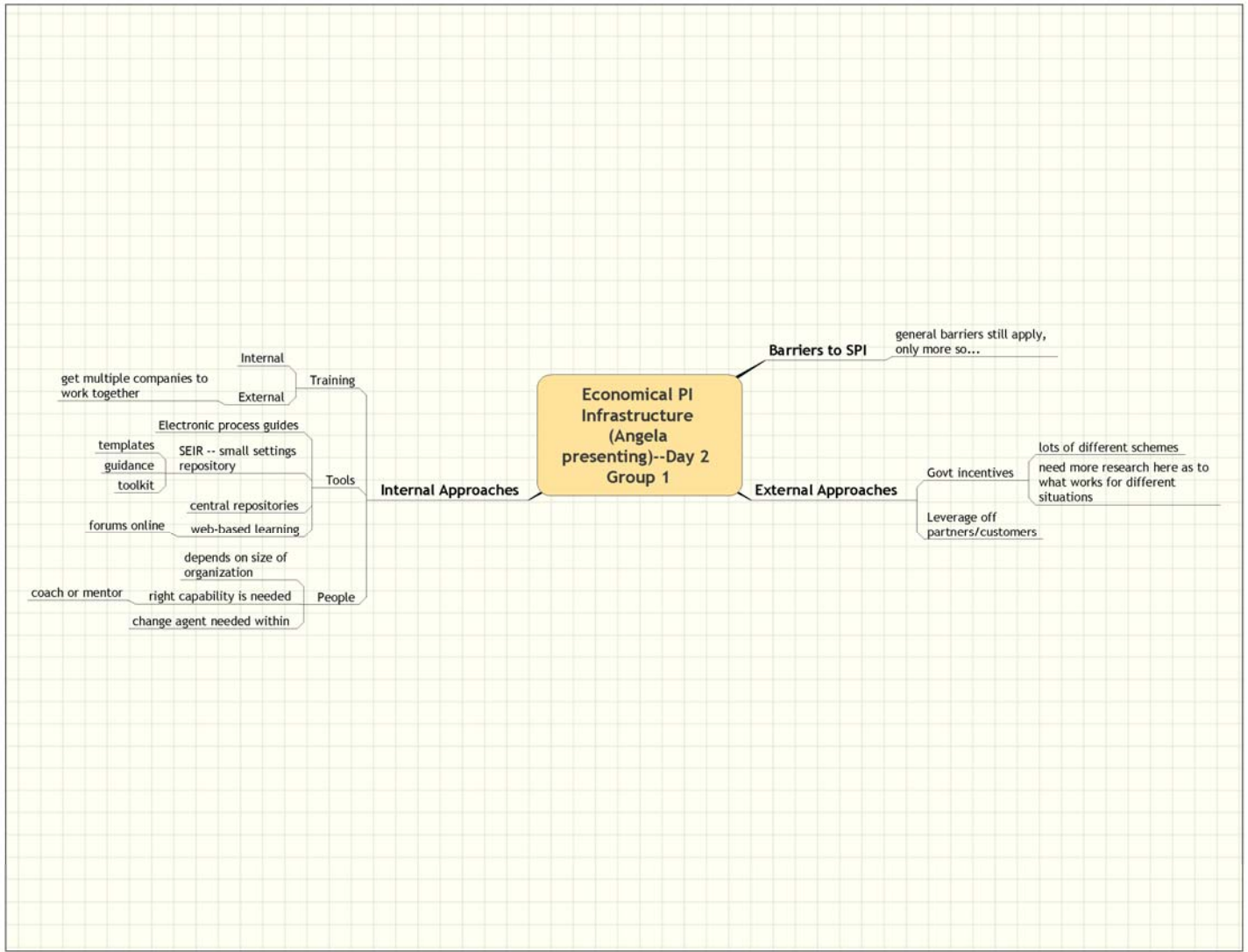
Agapi Svolou

Participants

Jose A. Calvo-Manzano, Khaled El Emam, Bob Ferguson, Dennis Goldenson, Rajneesh Sharma, and Dorai Sinna

Flip Chart Notes

See the following Mind Map.



Group B: Cost/Benefit Models 2

Facilitator

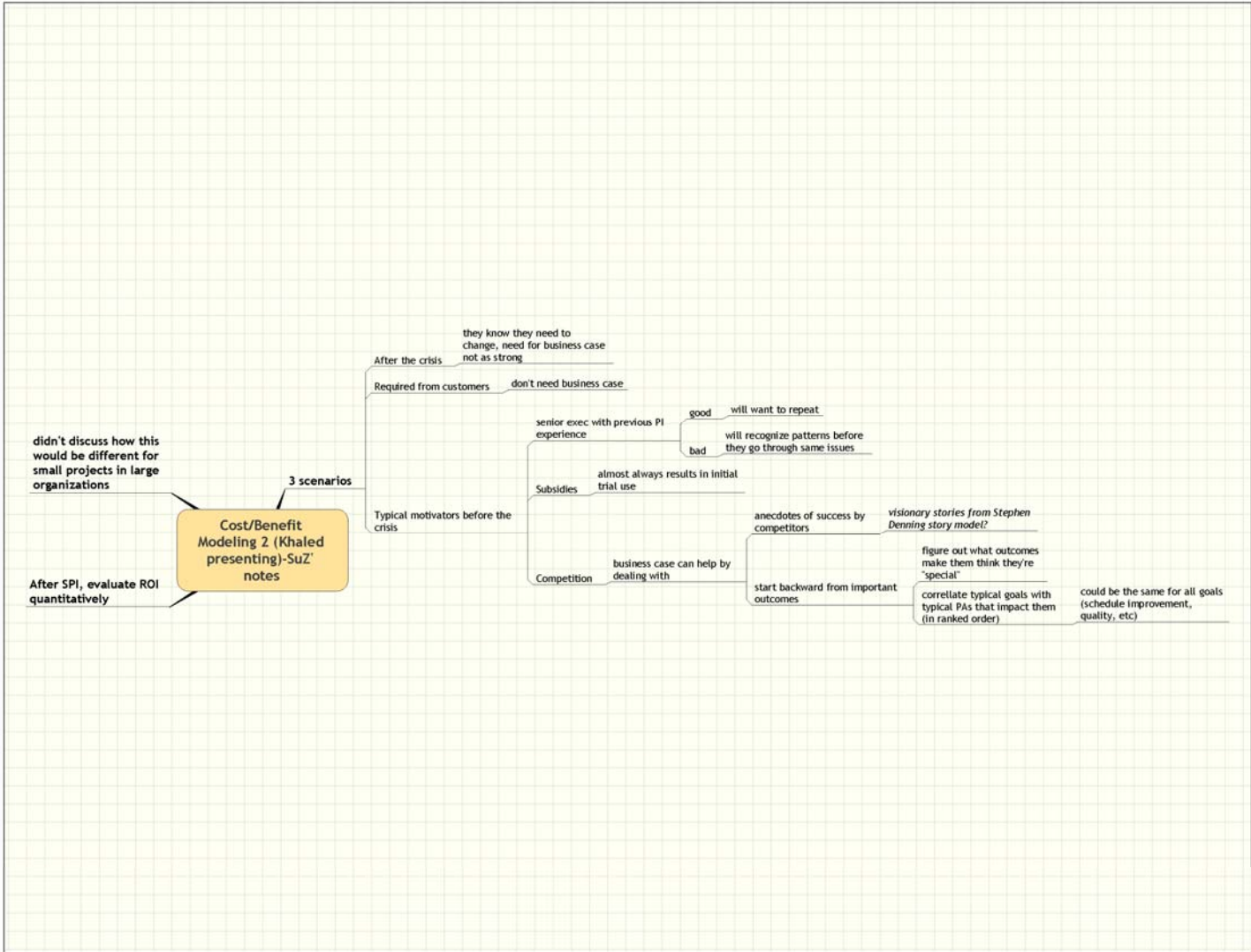
Miguel A. Serrano

Participants

Christian Carmody, Jose Antonio Cerrade, Diane Gibson, Mary Anne Herndon, Zaijun Hu, James E. Jones, Nazrina Khurshid Mohamed, Yvette Lui, Oscar A. Mondragon, Anil Revankar, Tomás San Feliu, Katsutoshi Shintani, Karin Steembecker, Angela Tuffley, and Ng Wan Peng

Flip Chart Notes

See the following Mind Map.



Group C: Current/Future Research Topics

Facilitator

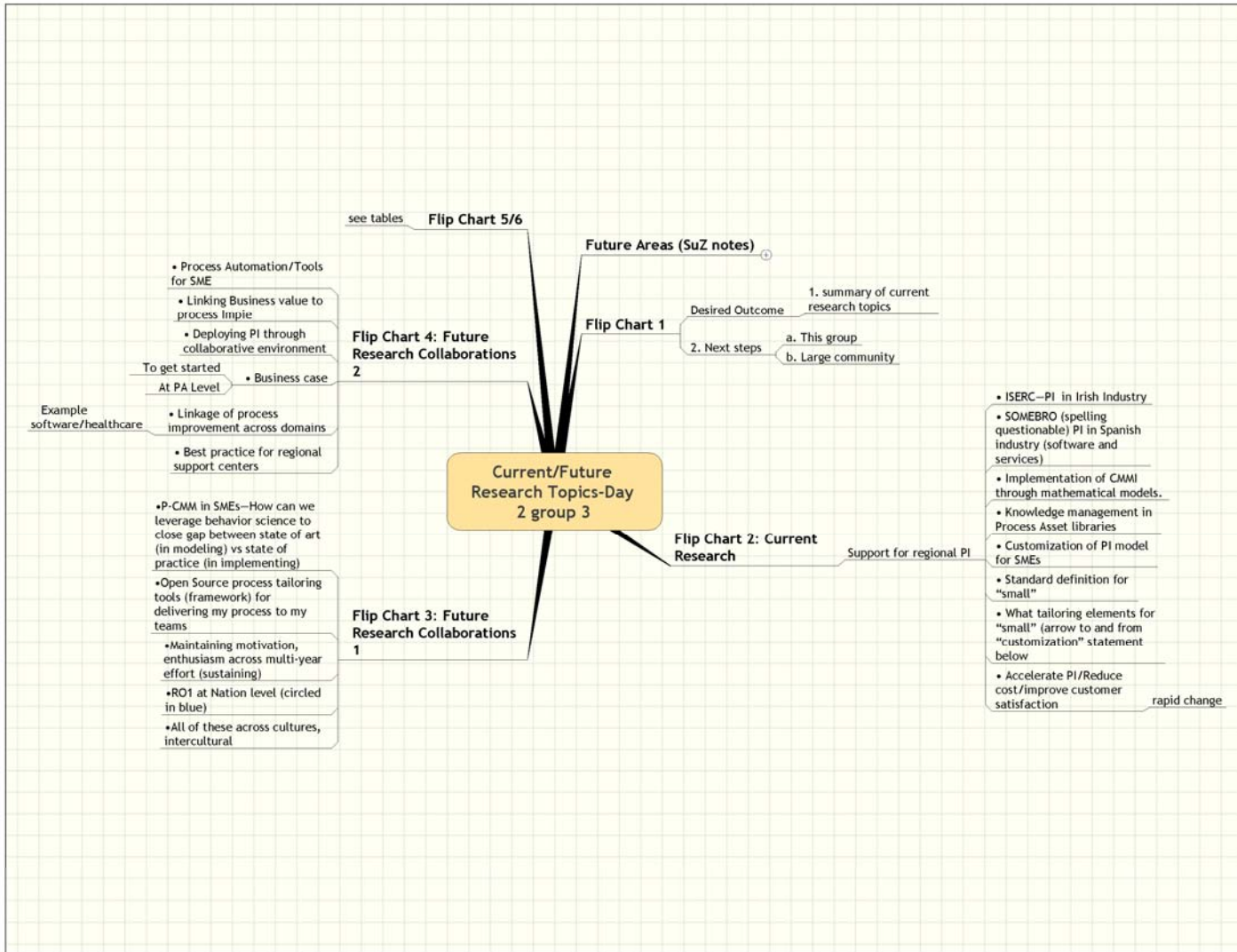
Caroline Graettinger

Participants

Rosario Blowers, Gonzalo Cuevas Agustín, Javier García, Alan Lawson, Venkata Muralidhar Nallagonda, V. Sathya Murthy, Karl Ng Kah Hou, K. Niranjana Kumar, Stephen Rupp, and Bosheng Zhou

Flip Chart Notes

See the following Mind Map and tables.



Flip Chart 5

Current	Future Research Collaboration (1)
<ul style="list-style-type: none"> • Support for regional PI 	<ul style="list-style-type: none"> • Best Practice for Regional Support Centers
<ul style="list-style-type: none"> • CMMI implementation VP Mathematical Models 	<ul style="list-style-type: none"> • ROI at National Level • CMMI Implementation via Mathematical models
<ul style="list-style-type: none"> • Definition and Tailoring of “Small” 	
<ul style="list-style-type: none"> • Faster, Better, Cheaper (arrow to Process automation and tools) 	<ul style="list-style-type: none"> • Linking Business value to process improvement
<ul style="list-style-type: none"> • Knowledge management in process asset libraries 	<ul style="list-style-type: none"> • Process automation and tools • Open Source
<ul style="list-style-type: none"> • CMMI continuous vs. staged 	<ul style="list-style-type: none"> • Collaborative environment

Flip Chart 6

Current	Future Research Collaboration (2)
<ol style="list-style-type: none"> 1. Client specifics level 2. does not specify, but vendor prepares of future 3. Client carries out supplier improvement program. 	Behavioral science, starting and sustaining
	Intercultural
	Business case to get started at PA level
	Accelerators and inhibitor in the supply chain
	Volunteers to expand on next step topics

Group D: ISO/SC 7 Working Group 24 Input

Facilitator

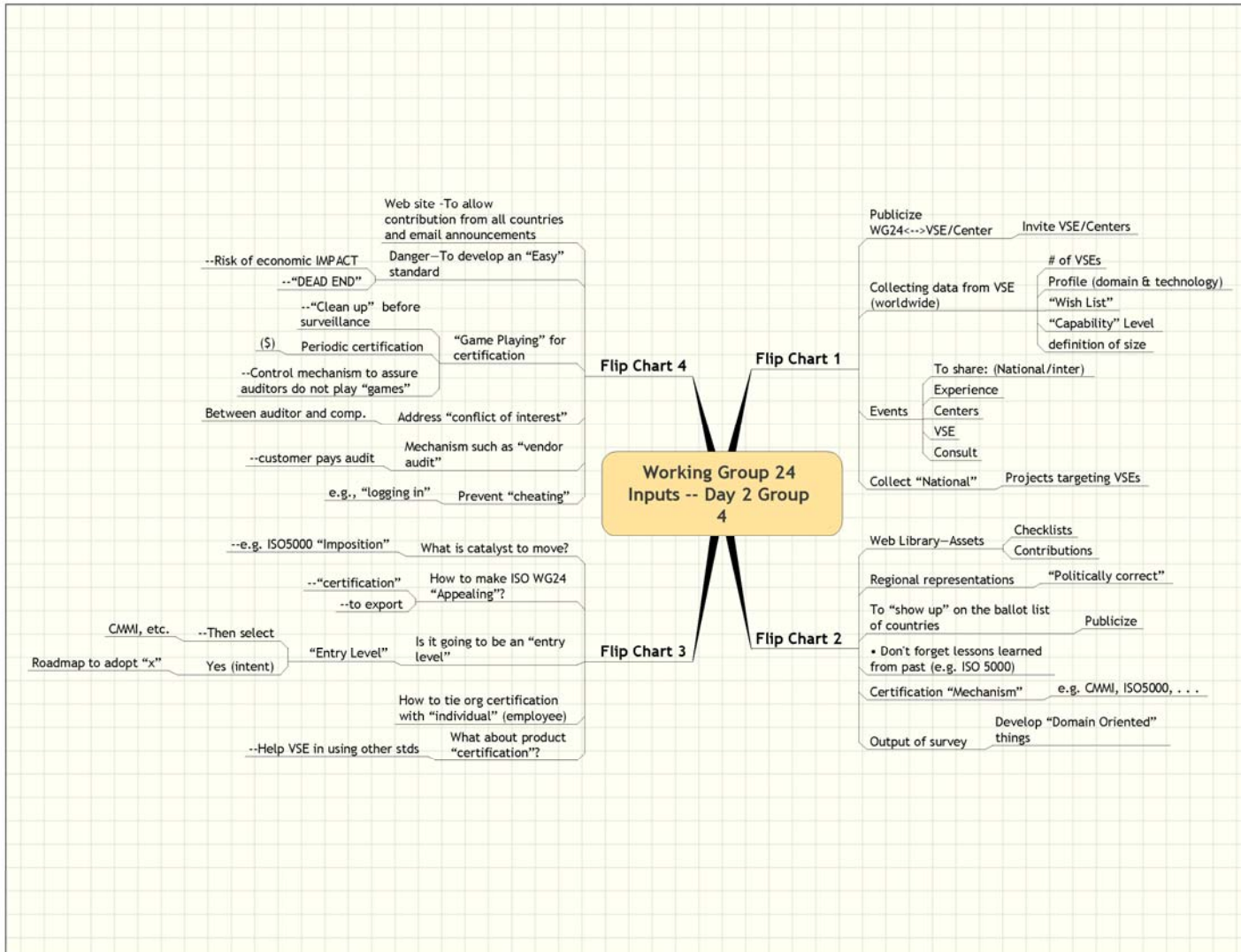
Miguel A. Serrano

Participants

Hanna Oktaba

Flip Chart Notes

See the following Mind Map.



REPORT DOCUMENTATION PAGE			<i>Form Approved</i> <i>OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE January 2006	3. REPORT TYPE AND DATES COVERED Final		
4. TITLE AND SUBTITLE Proceedings of the First International Research Workshop for Process Improvement in Small Settings, 2005		5. FUNDING NUMBERS FA8721-05-C-0003		
6. AUTHOR(S) Suzanne Garcia, Caroline Graettinger, and Keith Kost				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Software Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213		8. PERFORMING ORGANIZATION REPORT NUMBER CMU/SEI-2006-SR-001		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) HQ ESC/XPK 5 Eglin Street Hanscom AFB, MA 01731-2116		10. SPONSORING/MONITORING AGENCY REPORT NUMBER		
11. SUPPLEMENTARY NOTES				
12A DISTRIBUTION/AVAILABILITY STATEMENT Unclassified/Unlimited, DTIC, NTIS		12B DISTRIBUTION CODE		
13. ABSTRACT (MAXIMUM 200 WORDS) The first International Research Workshop for Process Improvement in Small Settings was held October 19-20, 2005 at the Software Engineering Institute in Pittsburgh, Pennsylvania. Attendees from Australia, Canada, Chile, China, Germany, Ireland, India, Japan, Malaysia, Mexico, Spain, and the United States discussed the challenges of process improvement in small and medium size enterprises, small organizations within large companies, and small projects. The presentations addressed starting and sustaining process improvement, qualitative and quantitative studies, and using Capability Maturity Model Integration (CMMI), Agile, Modelo de Procesos para la Industria de Software (MoProSoft), International Organization for Standardization (ISO), Quality Function Deployment (QFD), and Team Software Process (TSP) in small settings. The workshop also had working groups that discussed issues unique to small settings, such as regional support centers and process improvement "on a shoestring." This report includes the papers from this workshop and presents conclusions and next steps for process improvement in small settings. This report also contains the workshop breakout session results.				
14. SUBJECT TERMS small organization, small project, small setting, CMMI, CMMI adoption, TSP, ISO, process improvement guidance		15. NUMBER OF PAGES 309		
16. PRICE CODE				
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	