



Agile Development and the CMMI Anti-Matter and Matter or Reconcilable Differences?

Presented by:

Steve Ornburn – GBC Group – sbo@gbc-group.com

David Kane – SRA International – david_kane@sra.com

Background	Introduction	Issues	Conclu
------------	--------------	--------	--------

Proble

The Department of Defense and many other government agencies have made investments in the Capability Maturity Model (CMM)

- In many cases these investments have returned significant value
- There is also investment and interest from the private sector

There's a surge in interest in Agile Development (AD) methodologies such as eXtreme Programming, Adaptive Programming, SCRUM and Crystal.

- Many experience reports indicate impressive results

AD is often described as being at odds with the goals and approaches of the CMMI.

Is there a role for AD in major government software development activities?

- Especially those in which the CMMI plays a significant role

Background	Introduction	Issues	Conclu
------------	--------------	--------	--------

Our Position

AD is different in practice and philosophy than the methods often deployed to achieve the goals of the CMMI

However, there is room for AD in a high-maturity shop

Nevertheless, introducing AD requires overcoming obstacles and entails risk.

Background

Introduction to AD and CMMI

The Issues

- Is AD a mature development practice?
- Can AD fit into in a shop that uses CMMI?
- Is AD worth the trouble?

Conclusions

Metamodel for the software life cycle

IEEE Std 1074

Project Mgmt

Project initiation, Project Monitoring and Control, Quality Mgmt

Definition

Concept Exploration
Systems Allocation

Development

Requirements Analysis
Design
Implementation

Post Development

Installation
Operations & Support
Maintenance
Retirement

Integrated processes

V&V, Configuration Management, Documentation, Training

standard to test the completeness of a software life cycle definition. Process models should map to it, but it does not imply particular sequencing of activities

Background	Introduction	Issues	Conclu
-------------------	---------------------	---------------	---------------

Risk-Driven Systems Development

A Diversity of Process Models

Matching process to project based on risk profile

–Many process models

- Waterfall Model
- Iterative Models
- Evolutionary Models

–Other process attributes

- Document Driven
- Architecture Driven
- Requirements Driven
- Quality Driven

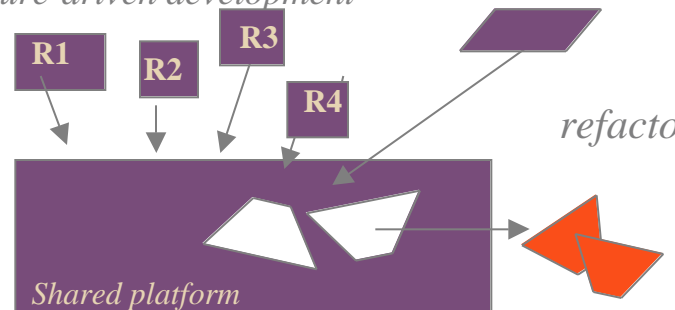
Requirements-driven development



Stand along, loosely coupled features

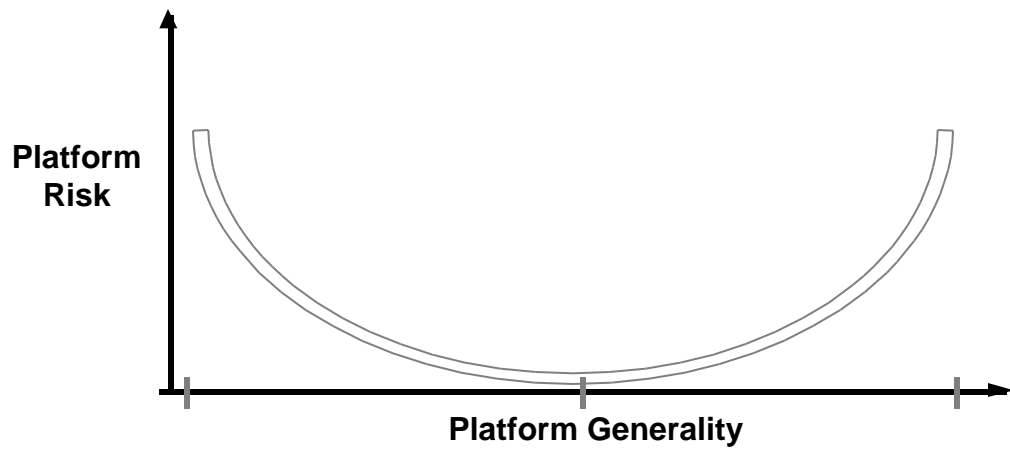
Architecture-driven development

New features



What you do depends on what you're trying to accomplish

Balance and Rhythm



Background	Introduction	Issues	Conclu
------------	--------------	--------	--------

Difference in the Nature of Complexity and

What are the risk-drivers for each type of environment?

- Safety critical with managed change in requirements and schedule
 - CMMI: Complicated, but stable environments
 - Highsmith cites the space shuttle as typical of this domain
- Not safety critical, but subject to unpredictable change
 - Agile Methods: Suitable for complex, chaotic environments
 - Internet-centric application are cited as typical here
 - Speed and changing environments

Background	Introduction	Issues	Conclu
------------	--------------	--------	--------

Agile Methods and Organizational Boundaries

Another difference in risk driver

Risky intergroup communications: low bandwidth, error prone

– For large systems, the key participants reside in different organizations

Reliable intergroup communication: streamline, eliminate redundancy

– The rapid change of Agile techniques is not well-suited for crossing organizational boundaries

- E.g. It is one thing to refactor an interface that you control, it is another to alter an interface shared with another organization

Agile Development Summariz

- Agile Development (AD) refers to a collection of development approaches inspired by the Agile Manifesto
 - Key Manifesto points
 - Individuals and interactions over processes and tools
 - Working software over comprehensive documentation
 - Customer collaboration over contract negotiation
 - Responding to change over following a plan
- Many different published techniques
 - eXtreme Programming (XP)
 - Scrum
 - Adaptive Software Development
 - Crystal
- We'll tend to focus on XP because it is the approach with the most published, but we'll draw on the others over the course of the presentation as well

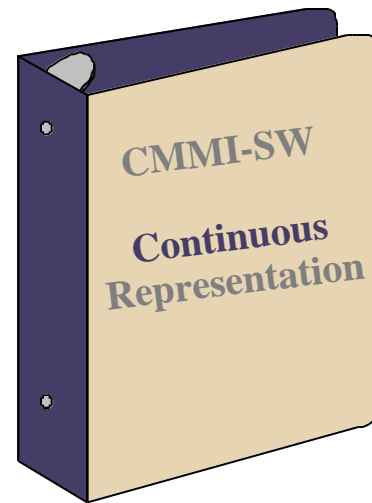
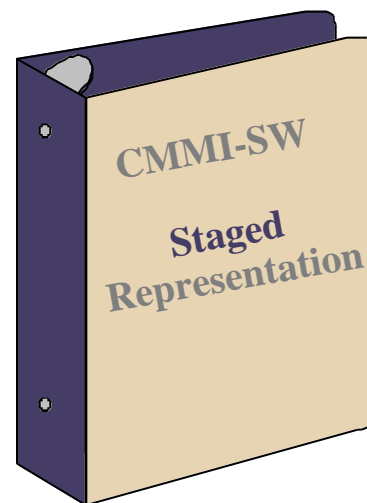
CMMI Model

Source Models

Capability Maturity Model for Software V2, draft C (SW-CMM V2C)

EIA 731, System Engineering Capability Model (SECM)

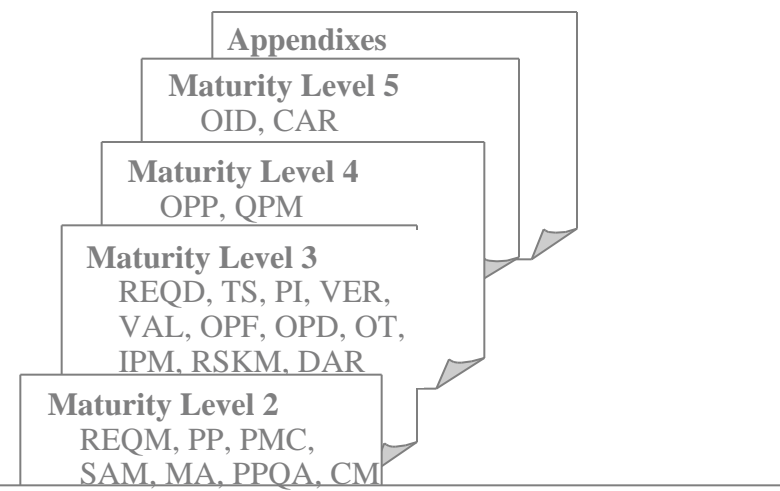
Integrated Product Development Capability Maturity Model, draft V0.98 (IPD-CMM)



CMMI

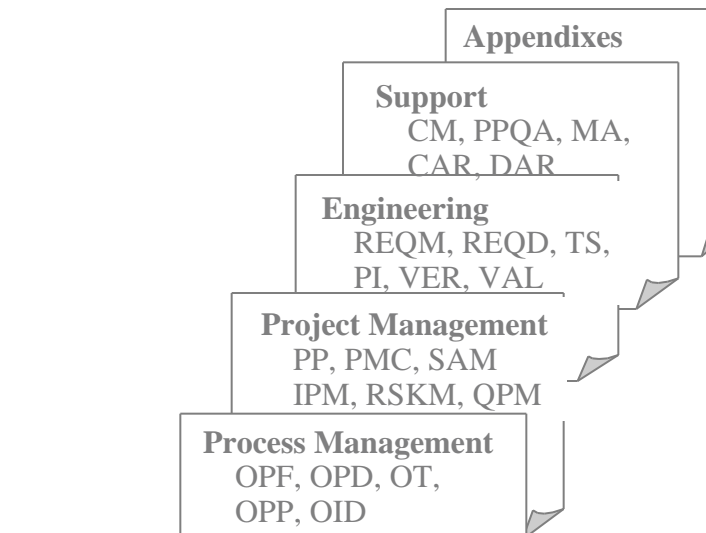
- Meets the needs of software organizations
- Is an upgrade of SW-CMM
- Benefits from best practices contributed from all three source models

One Model, Two Representations



- Overview**
- Introduction
 - Structure of the Model
 - Model Terminology
 - Maturity Levels, Common Features, and Generic Practices
 - Understanding the Model
 - Using the Model

**CMMI-SE/SW
Staged**



- Overview**
- Introduction
 - Structure of the Model
 - Model Terminology
 - Capability Levels and Generic Model Components
 - Understanding the Model
 - Using the Model

**CMMI-SE/SW
Continuous**

The Staged Representation of the CM

Level 5: Optimizing

Organizational innovation & deployment
Risk Analysis and Resolution

Level 4: Quantitatively Managed

Organizational Innovation and Deployment
Quantitative Project Management

Level 3: Defining

Level 2: Managed

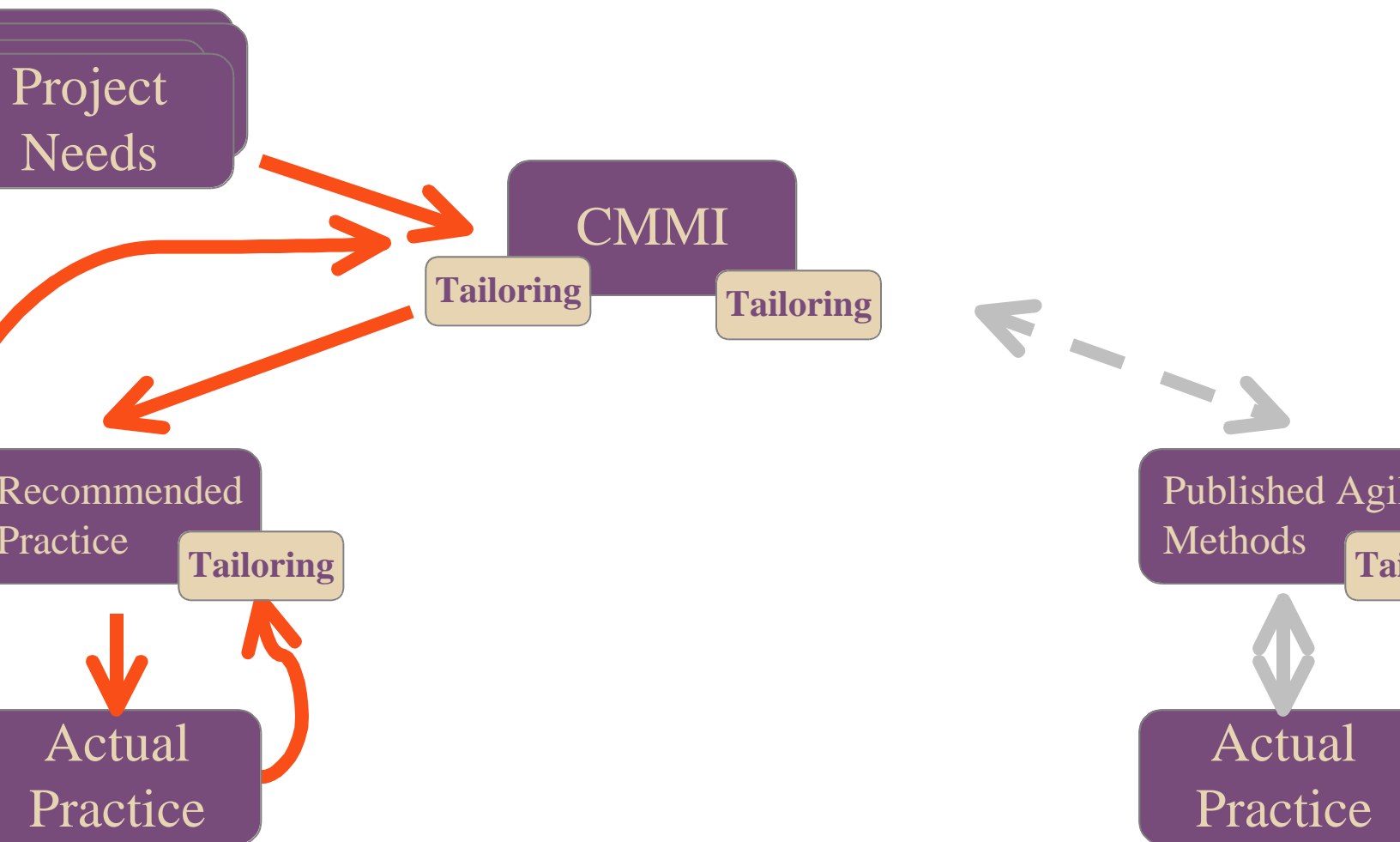
Requirements Management
Project Planning
Project Monitoring & Control
Supplier Agreement Management
Measurement & Analysis
Process & Product QA
Configuration Management

Requirements Development
Technical Solution, Product Integration
Verification, Validation
Organizational Process Focus,
Process Definition, & Training
Integrated Project Management
Risk Management
Decision Analysis and Resolution

Level 1: Initial



Reference Model



			Background	Introduction	Issues	Conclu
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence
						Co

The Issue

Is AD a mature development practice?

- Coverage
- Process
- Work Products

Can AD fit into in a shop that uses CMMI?

- Additional Practices
- Tailoring
- Change Management

Is AD worth the trouble?

- Evidence
- Context

			Background	Introduction	Issues	Conclu	
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence	Co

Is AD a mature development practice

Coverage: How do AD approaches map to the CMMI's Key Process Areas?

Process: Can AD practices satisfy the CMMI's requirements for a process?

Work Products: Will there be enough documentation for an assessment?

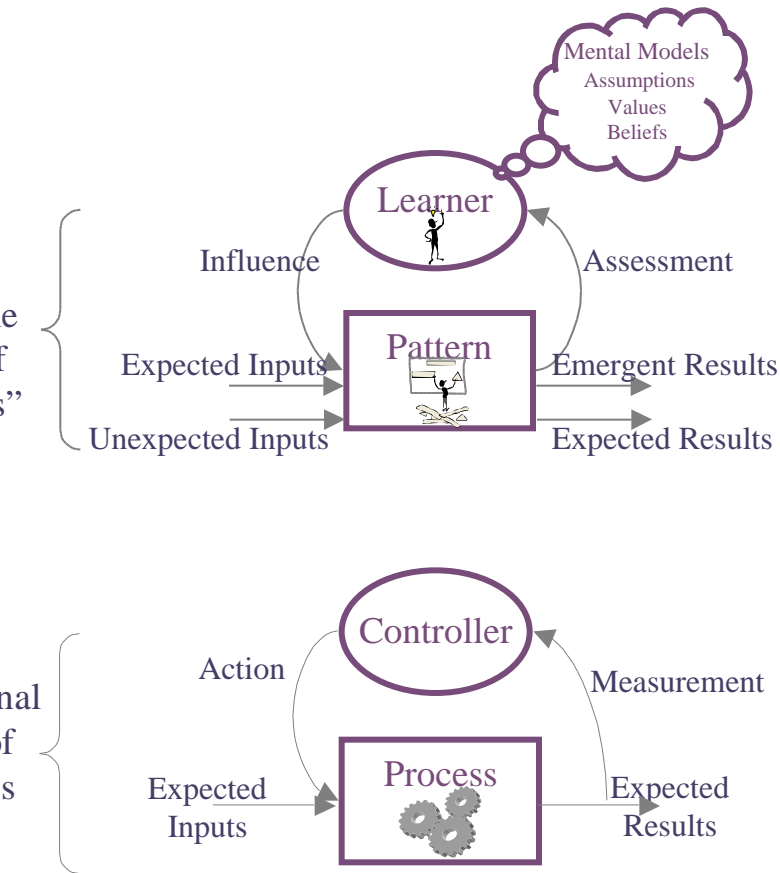
			Background	Introduction	Issues	Conclu	
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence	Co

Mapping of the SW- CMM KPA

Level	KPA	XP	Scrum	ASD
2	Requirements management	++	++	++
2	Software project planning	++	++	++
2	Software project tracking and oversight	++	++	++
2	Software subcontract management	-	-	-
2	Software quality assurance	+	-	++
2	Software configuration management	+	-	-
3	Organization process focus	+	+	+
3	Organization process definition	+	+	+
3	Training program	+	+	+
3	Integrated software management	-	-	-
3	Software product engineering	++	+	++
3	Intergroup coordination	++	++	++
3	Peer reviews	++	-	++
4	Quantitative process management	-	++	-
4	Software quality management	-	-	+
5	Defect prevention	+	-	+
5	Technology change management	-	-	-
5	Process change management	-	-	-

			Background	Introduction	Issues	Conclu
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence
						Co

Can AD practices satisfy the CM requirements for a pro



			Background	Introduction	Issues	Conclu	
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence	Co

Can this outline be used to analyze Agile methods
construct acceptable process descriptions

At maturity level 3, the required characteristics of a process definition are clearly outlined:

- **Purpose:** to iteratively enhance system
- **Inputs:** code, test cases
- **Entry criteria:** previous iteration completed
- **Activities:** nanoincrements of (code, review, test)
- **Roles:** program in pairs, reviews, everyone familiar with all parts of system
- **Measures:** Completed test cases, actual verses estimated effort
- **Outputs:** working code
- **Exit criteria:** all test cases successfully implemented
- **Verification steps:** running tests, informal peer reviews of design, conformance to coding standards

Possible process outline for iteration within XP

			Background	Introduction	Issues	Conclu	
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence	Co

Resolving the Process Iss

Published descriptions of AD approaches often omit explicit ordering of processes, careful documentation of inputs and outputs. But, these omissions can be fixed.

For AD

- Process engineer can diagram the flow of activity
- work products being code, and lists of action items, sketches and other working materials
- Team members have means of defining and assigning tasks, identifying and communicating issues and agreements, and judging whether work has been completed satisfactorily—the core of good process.

For such an approach to make sense, capture the parts of the AD practice that fit the model

- There may be valuable pieces of the AD practice that are not a good match for formal process descriptions
- Avoid over specifying the work instructions
 - E.g. Do not try to document the interaction of pair programmers in such a process model

			Background	Introduction	Issues	Conclu	
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence	Co

Will there be enough documentation to satisfy an assessor?

AD is not without artifacts

- But there is no effort to create artifacts just to verify the process
- E.g. XP uses “working code” in lieu of documentation

Agile techniques often do not generate many formal artifacts, though a number generate informal work products

- E.g. Several focus on whiteboard use, which is an ephemeral medium

			Background	Introduction	Issues	Conclu	
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence	Co

Considerations for the Assessment Approach

Assessment team's findings address whether the goals of the CMMI are met for the process being observed

Assessment team base their findings on observations that, in turn, are based on data gathered from one or more data sources. Data sources include

- Appraisal Questionnaires and Surveys
- Interviews with project leaders, managers, practitioners
- Reviews of work products, plans, process documents, and policies

Observations must be corroborated from multiple data sources

- the data collected must be sufficient to understand
- the extent of the implementation of practices
- whether they are representative of the life-cycle phase and the organization

			Background	Introduction	Issues	Conclu	
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence	Co

Can AD fit into in a high-process-maturity s

Additional practices: augmenting the Agile methods

Tailoring: adapting the CMMI to work with AD

Change management: AD for the Mature and the Bol

			Background	Introduction	Issues	Conclu	
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence	Co

Augmenting A

There are several KPA's that address areas typically out of scope for AD

- Software Subcontract Management
- Integrated Software Management
- Quantitative Process Management
- Software quality management
- Technology change management
- Process change management

For example, XP has 12 practices.

- There are no provisions for systematically identifying and incorporating new practices with in an organization
 - Ironically, for many in the AD movement, introducing change is viewed an industry, rather than an organizational perspective
- We are already seeing changes to the processes
 - E.g. Xbreed, a Scrum/XP variant

			Background	Introduction	Issues	Conclu	
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence	Co

Guidance for Augmenting

The CMMI addresses a broader scope than AD

– Particularly in the areas of

- Change Management
- Executive Support

Organizations still need to address these areas, even if specific practices are not identified in AD

Additional practices can be put in place to augment AD, and satisfy the CMMI

			Background	Introduction	Issues	Conclu	
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence	Co

Can the CMMI be appropriately tailored to work with

The burden of proof is on the organization to satisfy an SEP, SQ or assessment group that their agile approach satisfies the CMMI

- "When you use a CMMI model as a guide, you plan and implement processes that conform to the required and expected components of process areas. Conformance with a process area means that in the planned and implemented processes there is an associated process (or processes) that carries out either the specific and generic practices of the process area, or **ALTERNATIVES THAT CLEARLY AND UNEQUIVOCALLY ACCOMPLISH A RESULT THAT MEETS THE GOAL** associated with that specific or generic practice."

CMMI SE/SW v1.02, Staged Representation, Pa

			Background	Introduction	Issues	Conclu	
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence	Co

Approaches to Tailoring

Conduct pilots and collect data to satisfy the burden-of-proof

Publication of successful AD tailorings could reduce the effort required for others to tailor in a similar fashion

			Background	Introduction	Issues	Conclu	
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence	Co

The Mature (Level 5) sho

Improvements are selected based on a quantitative understanding of their expected contribution to achieving the organization's process improvement objectives versus the cost and impact to the organization.

Optimizing processes that are agile and innovative depend on the participation of an empowered workforce aligned with the business values and objectives of the organization.

The organization's ability to rapidly respond to change and opportunities is enhanced by finding ways to accelerate and share learning.

			Background	Introduction	Issues	Conclu	
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence	Co

In Less Mature Sho

Be prepared for resistance to change

- SEPG and QA groups may feel threatened by the grassroots approach that may accompany the introduction of AD
 - There is no direct role for QA in many AD approaches
- AD teams may feel threatened by the SEPG

Failure to resolve the tension can create a lose-lose scenario

			Background	Introduction	Issues	Conclu	
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence	Co

Is AD worth the trouble?

Evidence: In what circumstances are Agile Methods appropriate?

Context: Are AD and CMMI suitable in any of the same environments?

			Background	Introduction	Issues	Conclu	
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence	Co

Are the extraordinary results claimed for agile methods compelling?

Compelling testimonials are generating considerable interest and activity

But, how much of the extraordinary gains reported by Agile developers are a result of natural differences in programmer productivity?

- Fred Brooks has observed an order of magnitude difference in performance between average programmers and the best programmers

The answer to this should be clearer as more groups put Agile methods into practice.

- In our own experience, not every team is suited to Agile Methods, but empowering those who are generates significant value for the company
- We also find that care must be exercised to spot and manage potential runaways among projects applying Agile methods.

			Background	Introduction	Issues	Conclu
Coverage	Process	Work Products	Additional Practices	Tailoring	Change Management	Evidence
						Co

Are AD and CMMI suitable in any of the s environme

Agile methods

- Focused on small and medium-sized projects
 - Some experience reports of larger-projects are beginning to surface
- Assume a fairly simple model of customer validation
 - Lacks processes for reconciling complex diverse user and customer communities
- Assume a high-level of customer trust
- Note: These assumption tend to break down when software and interfaces cross organizational boundaries.

CMMI

- Frequently applied to large, complicated projects
 - Although it has been adapted for smaller projects
- Compliant processes often require detailed conceptual models before cutting code
- Often used as part of a formal acquisition process



Our Prediction

Agile development represents a paradigm shift for software development

Agile development is unlikely to go away

- Grass roots support
- Availability of inexpensive (or free) tools

Agile techniques will get shaped by current practices as the technology is diffused

- By analogy, the introduction of the OO paradigm did not invalidate earlier lessons about data structures

Our Recommendations

Be aware of the Agile Software Development movement

- It may gain a foothold in your organization, if not through your SEPG, then through your developer underground

Look for opportunities to pilot the methods

- Be aware of the context of your selected pilots

Engage your SEPG

- If you have CMMI requirements to meet, and if you want to promote Agile methods across your organization, their participation is essential.

Reference

- [01] D. Ahern, A. Clouse, R. Turner, *CMMI Distilled: A Practical Introduction to Integrated Process Improvement*. Addison-Wesley, 2001.
- [] ARC, V1.0 Assessment Requirements for CMMI, Version 1.0, CMU/SEI-2000-TR-011.
- [] K. Beck, *Extreme Programming Explained: Embrace Change*. Addison-Wesley, 2000.
- [97] L. Belady, M. Lehman, "A model of large program development", IBM Systems Journal, 1976 vol. 15, no. 1, 1976.
- [88] B.. Boehm. "A spiral model of software development and enhancement," *IEEE Computer*, May 1988 v 21 n2.
- [6] G. Booch, *Object Solutions: Managing the Object-Oriented Project*. Addison-Wesley, 1996.
- G. Box, "Some Problems of Statistics and Everyday Life," *Journal of the American Statistical Association*, v74 n365, pp1-10, 1969.
- G. Box, Personal communication with David Kane, November 25th, 1999.
- [SE/SW00] CMMI for Systems Engineering/Software Engineering, Version 1.02, CMU/SEI-2000-TR-028.
- [rn02] A. Cockburn, *Agile Software Development*, Addison-Wesley, 2002.
- [1] *Cutter IT Journal*, The Great Methodologies Debate: Part 1, December 2001, v14 no12.
- [2] *Cutter IT Journal*, The Great Methodologies Debate: Part 2, January 2001, v15 no1.
- [] D. Dikel, D.Kane, J. Wilson, *Software Architecture: Organizational Principles and Patterns*. Prentice-Hall: 2001.
- [] D. Dikel, D. Kane, S. Ornburn, W. Loftus, J.Wilson, "Applying Software Product-Line Architecture," *Computer*, August 1999, v18, n6.
- [ng01] J. Grenning, "Launching Extreme Programming at a Process-Intensive Company," *IEEE Software*, November/December 2001, v18, n6.
- [ith00] J. Highsmith, *Adaptive Software Development: A Collaborative Approach to Managing Complex Systems*. Dorset House Publishing, 2000.
- [x01] T. Lemieux, M. Moore, "XP and the CMM" presentation to the DC Chapter of the the Society for Software Quality on December 13th, 2001.
- [5] M. Paulk, C. Weber, B. Curtis, M. Chrissis, *The Capability Maturity Model: Guidelines for Improving the Software Process*. Addison-Wesley, 1995.
- [] M.Paulk, "Extreme Programming from a CMM Perspective," *IEEE Software*, November/December 2001, v18, n6.
- [er02] K. Schwaber, M. Beedle, *Agile Software Development with Scrum*. Prentice-Hall, 2002.