Software Process Improvement

Dr. Mark C. Paulk
SE 4381, Software Project Planning and Management

Management Topics

1. Modern project management

PMBOK

2. Organization strategy and project selection

3. Organization: structure and culture

4. Defining the project

5. Estimating times and costs

6. Developing a project plan

7. Managing risk

8. Scheduling resources and cost

9. Reducing project duration

10. Leadership

11. Teams

12. Outsourcing

13. Monitoring progress

14. Project closure

15. International projects

→ 16. Oversight

17. Agile PM

Critical chain project management

Two Improvement Paradigms

Analytic
- **based on measurement, principles**
  - Deming, Juran, Crosby, Lean, Six Sigma, …

Best Practice Frameworks
- **practices that consistently demonstrate significant improvements to the bottom line**

Important best practice frameworks
- **ISO 9001, ISO 9004 (Quality management systems)**
- **CMMI for Development, Services, Acquisition**
- **ISO/IEC 15504, ISO/IEC 330nn (Process assessment)**
- **Project Management, Software Engineering BOKs**
- **ISO 20000, ITIL, eSCM-SP (Service management)**

The Quagmire…

- **Moore, 1999**

Baldrige
CBA IPI
CMMI
CobiT
DOD-STD-2167A
DOD-STD-2168
DOD-STD-7935A
EIA 632
EIA 731
eSCM
FAA iCMM
IEEE 1220
IEEE/EIA 12207
IPD-CMM
ISO 15939
ISO 9000

ISO/IEC 12207
ISO/IEC 15288
ISO/IEC TR 15504
J-STD-016
MIL-STD-498
MIL-STD-499B
People CMM
PSM
PSP
Q9000
QS 9000
RTCA DO-178B
SA-CMM
SCAMPI
SCE
SDC/CR
SDCE
SECAM
SE-CMM
SW-CMM
TL 9000
TSP
…

What Is a "Best Practice?"

Can be a management or technical practice

Consistently demonstrated to significantly improve the bottom line – one or more of
- **productivity**
- **development and/or maintenance costs**
- **schedule**
- **quality**
- **user satisfaction**
- **predictability of cost and schedule**

Demonstrated high return-on-investment (ROI)

Evidence-Based Software Engineering (& Management)

R.L. Glass, <u>Facts and Fallacies of Software Engineering</u>, 2004.

C. Jones, <u>Software Engineering Best Practices</u>, 2010.

A. Oram and G. Wilson (eds), <u>Making Software: What Really Works, and Why We Believe It</u>, 2011.

Evidence-based management
- **J. Pfeffer and R.I. Sutton, <u>Hard Facts, Dangerous Half-Truths, & Total Nonsense: Profiting from Evidence-Based Management</u>, 2006.**

and so forth…

Pfeffer's One-Eighth Rule

Why aren't "best practices" that are known to be effective implemented by all organizations?

- **One half of all people, in spite of the evidence, don't believe the connection between the practices and the results.**

- **One half of those who see the connection try to make isolated changes rather than using a comprehensive and systemic approach.**

- **Only about half of those who make systemic changes persist long enough to derive the benefits.**

Humphrey's Motivational Speech

Do you believe that management is a good idea that adds value to your organization?

Do you believe that organization learning makes for more effective and efficient work?

Do you believe that using measurement and data to drive decisions is better than intuition?

Do you believe that continual improvement builds business value?

… Then why don't you do any of these things?

Software CMM v1.1 (1987-2005)

Level	Focus	Key Process Areas
5 **Optimizing**	*Continuous process improvement*	**Defect Prevention** **Technology Change Management** **Process Change Management**
4 **Managed**	*Product and process quality*	**Quantitative Process Management** **Software Quality Management**
3 **Defined**	*Engineering processes and organizational support*	**Organization Process Focus** **Organization Process Definition** **Training Program** **Integrated Software Management** **Software Product Engineering** **Intergroup Coordination** **Peer Reviews**
2 **Repeatable**	*Project management processes*	**Requirements Management** **Software Project Planning** **Software Project Tracking & Oversight** **Software Subcontract Management** **Software Quality Assurance** **Software Configuration Management**
1 **Initial**	*Competent people (and heroics)*	

Implications of Maturity

Better predictability… Less variability… Improved performance…

Level	Process Characteristics	Predicted Performance
5 Optimizing	**Process improvement is institutionalized**	
4 Managed	**Product and process are quantitatively controlled**	
3 Defined	**Software engineering and management processes defined and integrated**	
2 Repeatable	**Project management system in place; performance is repeatable**	
1 Initial	**Process is informal and unpredictable**	

What Is An Assessment?

An appraisal of an organization's current software process for self-improvement by a trained team of experienced software professionals.

It is based on
- review of 4 to 6 representative projects
- responses to the maturity questionnaire
- in-depth discussions with project managers and practitioners
- collective knowledge and experience of the assessment team

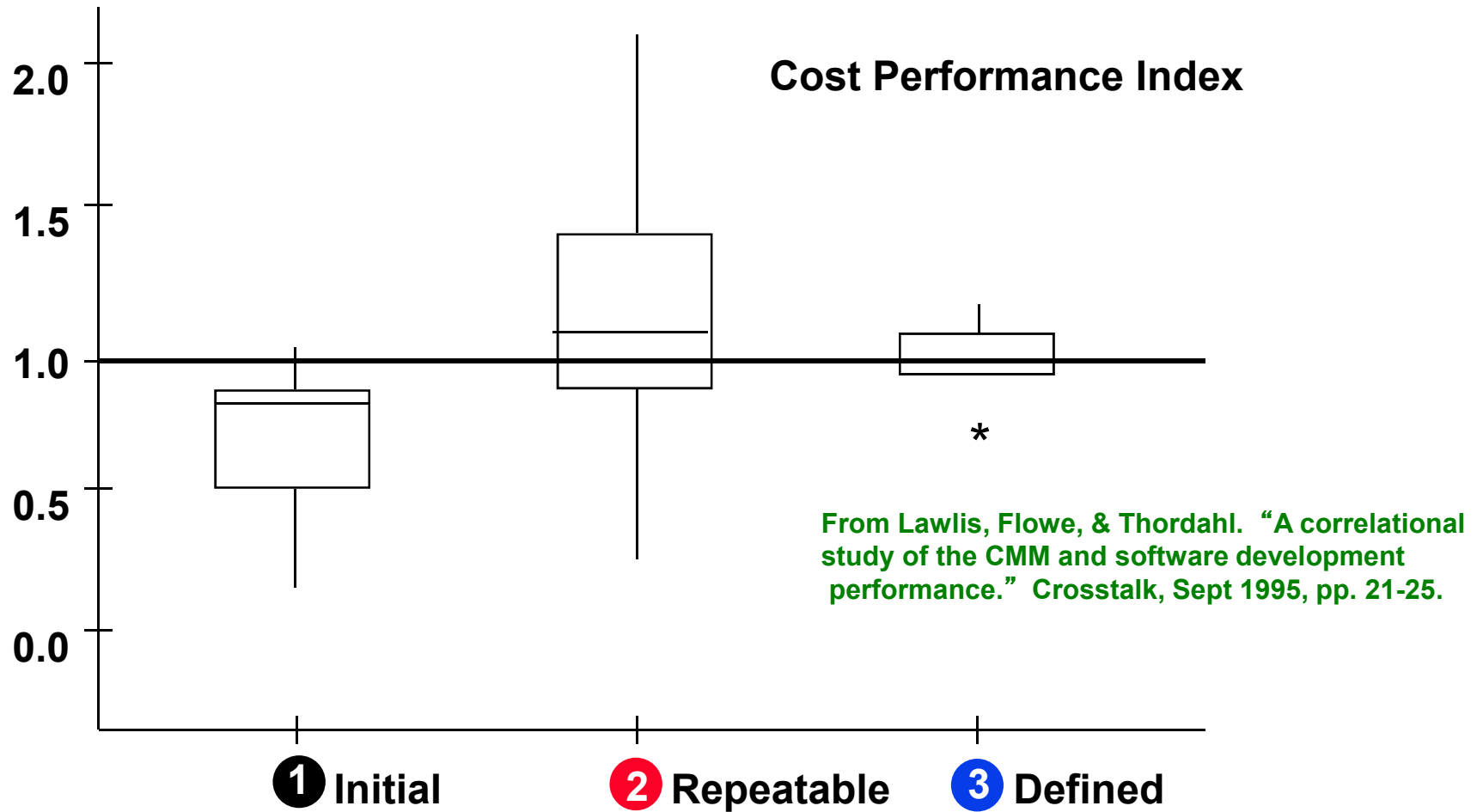# *Using the Software CMM*

Assessments
  • **self improvement**

Evaluations
  • **source selection**
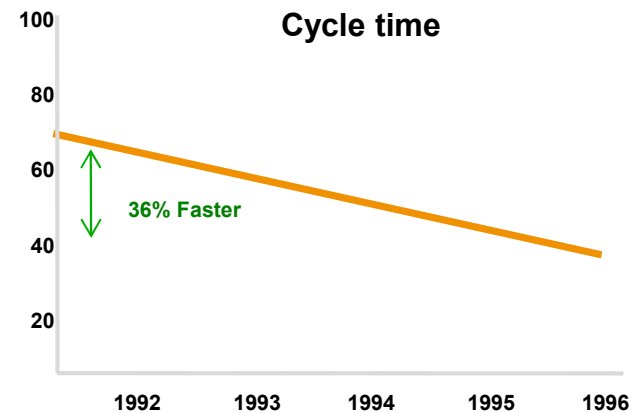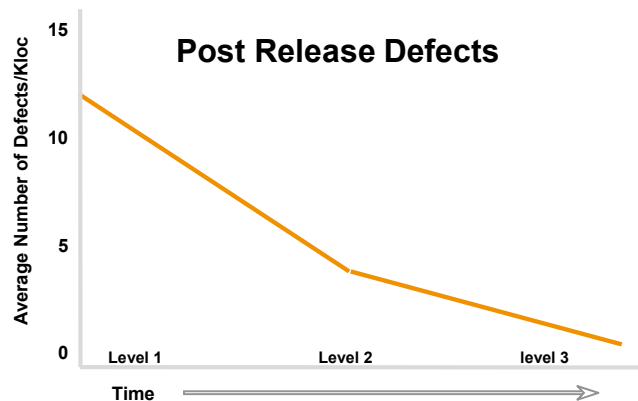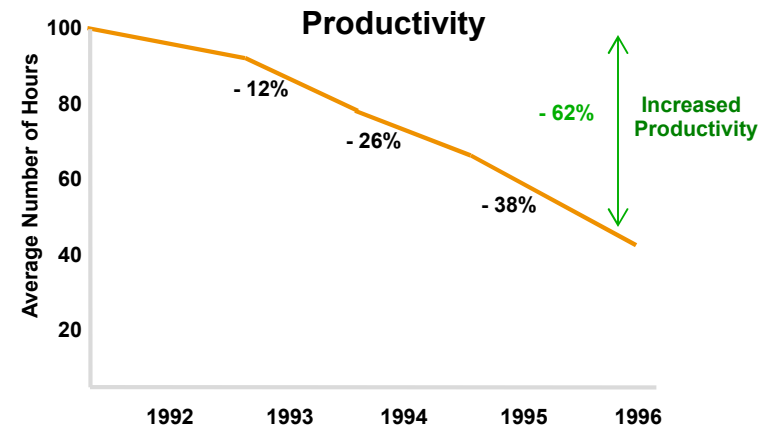 • **contract monitoring**

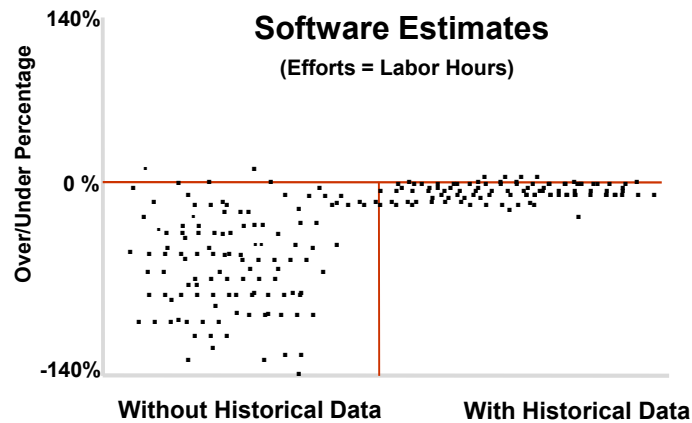**Appraisals = assessments + evaluations**

Other terms used: audit, capability determination

AFIT Study

Cost Performance Index

From Lawlis, Flowe, & Thordahl. "A correlational study of the CMM and software development performance." Crosstalk, Sept 1995, pp. 21-25.

1 Initial   **2** Repeatable   **3** Defined

Boeing Results

Software Estimates
(Efforts = Labor Hours)

Over/Under Percentage: 140%, 0%, -140%

Without Historical Data · With Historical Data

Productivity

Average Number of Hours: 100, 80, 60, 40, 20

- 12%
- 26%
- 38%
- 62%

Increased Productivity

1992 1993 1994 1995 1996

Post Release Defects

Average Number of Defects/Kloc: 15, 10, 5, 0

Level 1   Level 2   level 3

Time

Cycle time

100, 80, 60, 40, 20

36% Faster

1992 1993 1994 1995 1996

John Vu, Boeing, keynote talk at SEPG '97, "Software Process Improvement Journey (From Level 1 to Level 5)"

14

"Trends" in Quality Results

Maturity Level	Design Faults / KSLOC (Keene)	Delivered Defects / FP (Jones)	Shipped Defects / KSLOC (Krasner)	Relative Defect Density (Williams)	Shipped Defects (Rifkin)
5	0.5	0.05	0.5	0.05	1
4	1	0.14	2.5	0.1	5
3	2	0.27	3.5	0.2	7
2	3	0.44	6	0.4	12
1	5-6	0.75	30	1.0	61

Samuel Keene, "Modeling Software R&M Characteristics." Unpublished report.
Capers Jones, "Software Benchmarking," IEEE Computer, October 1995, pp. 102-103.
Herb Krasner, "Self-Assessment Experience at Lockheed," Third Annual SEPG Workshop, 7 November 1990.
Karl D. Williams, "The Value of Software Improvement… Results! Results! Results!" SPIRE97, 4 June 1997.
Stan Rifkin, "The Business Case for Software Process Improvement," Fifth SEPG National Meeting, 26-29 April 1993.

Risks of Model-Based Improvement

Models are simplifications of the real world.

Models are not comprehensive.

Interpretation and tailoring must be aligned to business objectives.

Judgment is necessary to use models correctly and with insight.

Potential for dysfunctional behavior to get a maturity level rating.

Success Depends on People

Success depends on the people, the team, doing the work.
- **design-intensive work, such as software development and maintenance**
- **behavioral-change work, such as process improvement**

Project and organizational success depends on competent people working together effectively.

People as the Leverage Point

People have the greatest impact on performance.

People have the greatest impact on change.

People have the greatest impact on success.

For change to happen, people have to change their behavior.
- **as individuals**
- **as team members**
    - **although an individual may have limited influence on the team**
- **as members of an organization**
    - **through influence more than argument**
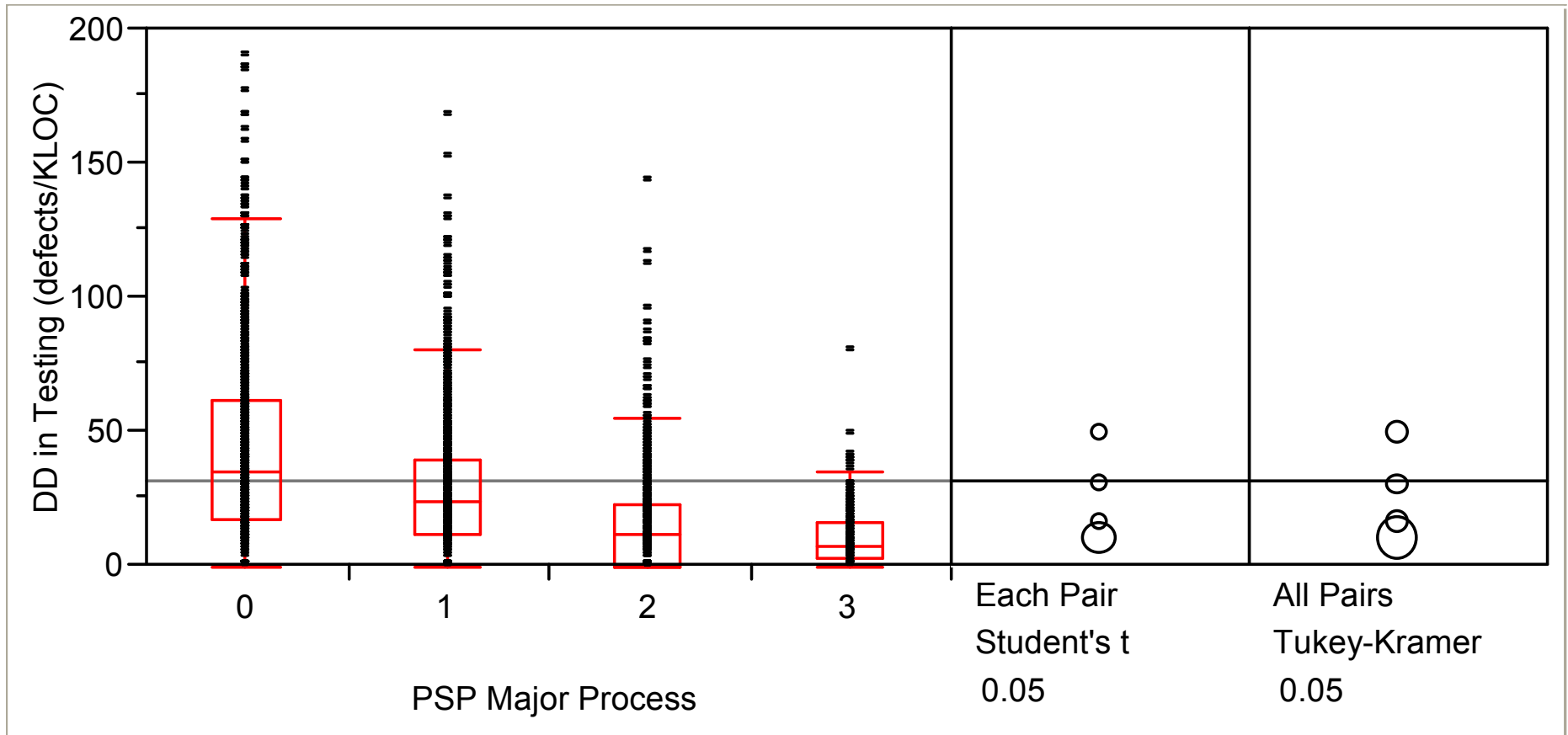
Personal Software Process (PSP)

Applies the CMM concepts of process discipline and quantitative management to the work of the individual software professional in a classroom setting.

Focuses on planning, quality, and productivity.

Typically involves developing programs using increasingly sophisticated processes.

Humphrey, A Discipline for Software Engineering, 1995.

Improvements in PSP Quality

Paulk, "An Empirical Study of Process Discipline and Software Quality," 2005.

Team Software Process (TSP)

Build teams of PSP-trained professionals, launched in an industry project context.

Provide
- **a supportive environment**
- **management standards**
- **peers doing similar work**
- **coaching support**

Note that CMMI tells "what" to do but not how. PSP and TSP tell "how."

Humphrey, Introduction to the Team Software Process, 1999.

Intuit TSP Results

Over 40 TSP teams

Post code-complete effort cut from 33% to 8% of the project

Standard test times cut from four months to one month

Twice the functionality in less time with higher quality

Microsoft TSP Results

Four releases for one team

Delivered on time

Increased productivity by 81%

Reduced system test defects by a factor of 133

Cut acceptance test and production defects to zero

CMMI-DEV v1.3

Level	Process Characteristics	Process Areas	
5 Optimizing	*Focus is on quantitative continuous process improvement*	Causal Analysis & Resolution Organizational Performance Management	
4 Quantitatively Managed	*Process is measured and controlled*	Organizational Process Performance Quantitative Project Management	
3 Defined	*Process is characterized for the organization and is proactive*	Requirements Development Technical Solution Product Integration Verification Validation	Organizational Process Focus Organization Process Definition Organizational Training Integrated Project Management Risk Management Decision Analysis & Resolution
2 Managed	*Process is characterized for projects and is often reactive*	Requirements Management Project Planning Project Monitoring & Control Supplier Agreement Management Product & Process Quality Assurance	Configuration Management Measurement & Analysis
1 Initial	*Process is unpredictable, poorly controlled, and reactive*		

A Hierarchy of Improvement

CMMi →

TSP →

PSP →
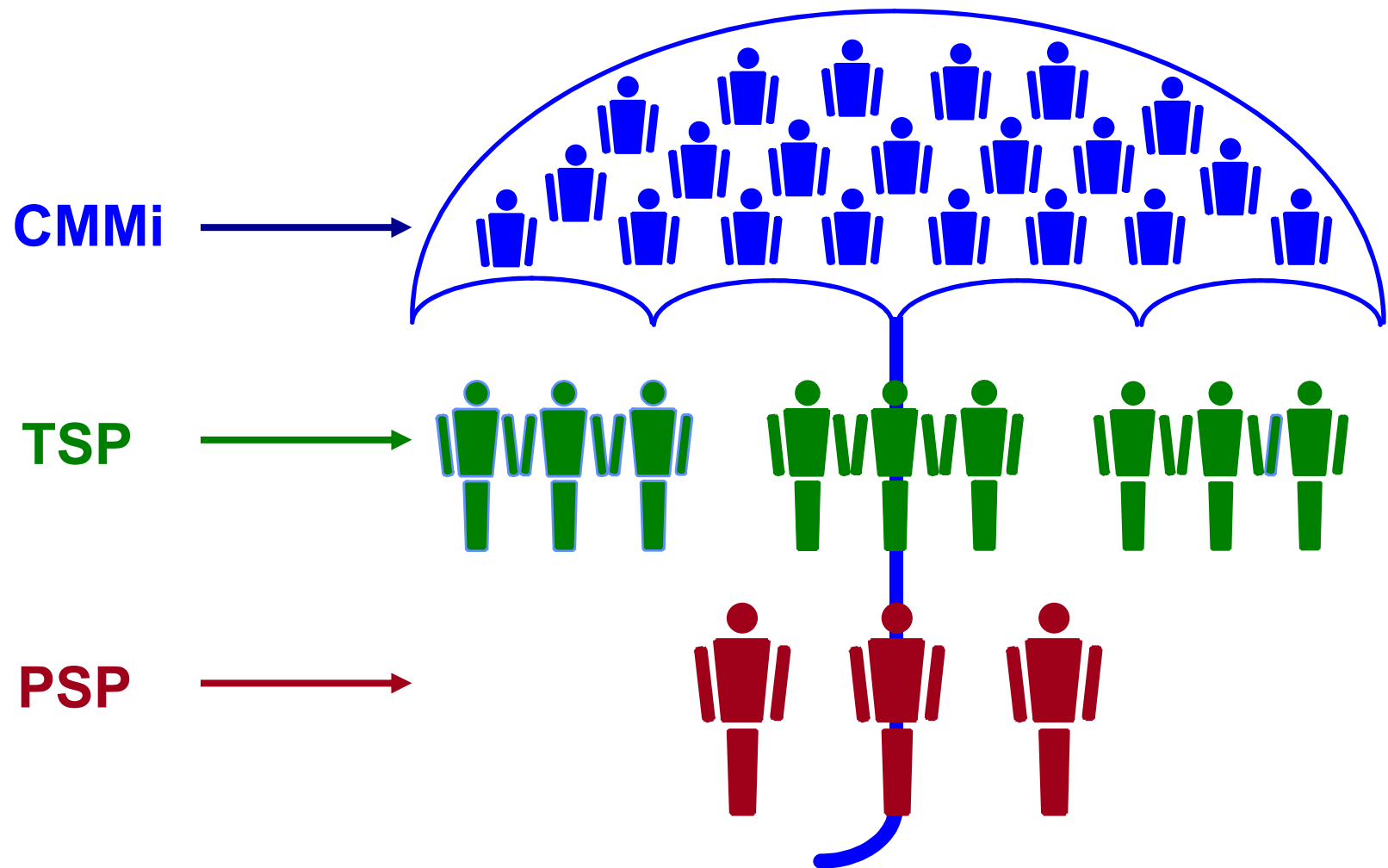
ISO/IEC 15504 aka SPICE aka ISO 33000 (Process Assessment)

Continuous representation
- processes (purpose + outcomes)
- capability dimension (six levels)

Process Reference Models
- ISO/IEC 12207 (Software life cycle processes)
- ISO/IEC 15288 (Systems life cycle processes)

Process Assessment Models
- ISO/IEC 15504-2 (Process assessment – Part 2: Performing an assessment)
- ISO/IEC 15504-5 (Software process assessment – Part 5: An assessment model and indicator guidance)

Rout, "ISO/IEC 15504 – Evolution to an International Standard," Software Process Improvement and Practice, January/March 2003.
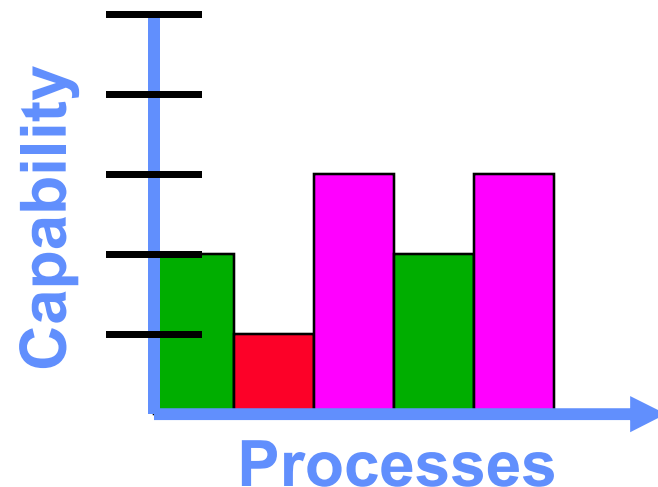
ISO 15504 Capability Dimension

Apply generic process management attributes to rate selected processes
  - **processes from ISO 12207, ISO 15288, ISO 15504-5**
  - **aka "continuous representation"**

Six capability levels
  - **level 0 = incomplete**
  - **level 1 = performed**
  - **level 2 = managed**
  - **level 3 = established**
  - **level 4 = predictable**
  - **level 5 = optimizing**

ISO/IEC 12207
"Software Life Cycle Processes"

A common framework for software life cycle processes
 • with well-defined terminology

Contains processes, activities, and tasks that are to be applied during the acquisition of
 • a system that contains software
 • a stand-alone software product
 • software service
 • during the supply, development, operation, and maintenance of software products

ISO 12207 Development Processes

5.3.1 Process implementation

5.3.2 System requirements analysis

5.3.3 System architectural design

5.3.4 Software requirements analysis

5.3.5 Software architectural design

5.3.6 Software detailed design

5.3.7 Software coding and testing

5.3.8 Software integration

5.3.9 Software qualification testing

5.3.10 System integration

5.3.11 System qualification testing

5.3.12 Software installation

5.3.13 Software acceptance support

ISO 330xx

The ISO/IEC 15504 series of standards is being replaced by the ISO/IEC 33001-99 series of standards.

Core Elements
- 33001 Concepts & Terminology
- 33002 Requirements for Process Assessment
- 33003 Requirements for Process Measurement Frameworks
- 33004 Requirements for Process Reference, Process Assessment and Organizational Maturity Models

Two Architectural Perspectives

A "staged" architecture, e.g., the Software CMM
 • focuses on building organizational capability
 • identifies the vital few issues to focus on
 • describes a <u>roadmap</u> for process improvement

The staged architecture was designed for changing organizational behavior.

A "continuous" architecture, e.g., ISO/IEC 33000
 • focuses on building process capability
 • provides a reference model for rating processes
 • describes the <u>terrain</u> of process management

The continuous architecture was designed for comparing different models.

Note: CMMI integrates both continuous and staged representations.

ISO 9001
(Quality Management Systems)

Demonstrate the ability to provide product that consistently meets customer, regulatory, and statutory requirements.

Enhance customer satisfaction by continual improvement and conformance to requirements.

Main clauses:
- 4 Quality management system
- 5 Management responsibility
- 6 Resource management
- 7 Product realization
- 8 Measurement, analysis and improvement

Quality Management Principles
ISO 9004:2000

Customer focus

Leadership

Involvement of people

Process approach

System approach to management

Continual improvement

Factual approach to decision making

Mutually beneficial supplier relationships

ISO 9004:2009 emphasizes sustainability.

To Be Blunt…

Many – perhaps most – software process improvement efforts fail.

El Emam and Goldenson, "An Empirical Review of Software Process Assessments," November 1999.
- *Approximately two thirds of respondents on SPI surveys do not report marked changes in their organizations as a consequence of their assessments.*
- *Organizations seem to have (unrealistically) high expectations from assessment-based SPI, and these need to be managed better.*

Critical Success Factors (Niazi, 2006)

From the literature:
- Senior management commitment
- Staff involvement
- Training and mentoring
- Staff time and resources
- Creating process action teams/change agents and opinion leaders
- Reviews
- Experienced staff
- Clear and relevant SPI goals

From empirical research:
- Senior management commitment
- Training
- Awareness
- Allocation of resources
- Experienced staff
- Defined SPI implementation methodology
- Staff involvement
- Facilitation

Four Needs Addressed by Process Documentation

Enables the organization to perform and repeat successful work.

Preserves lessons learned on organizational programs.

Facilitates and enhances training.

Evaluates and compares existing and proposed processes.

Gary B. Wigle and George Yamamura, "Practices of an SEI CMM Level 5 SEPG," Crosstalk, November 1997.

Implementation Variations

There is no "one right way" to implement most processes.

- **acceptable implementation does not mean the same to all organizations**
- **acceptable in one environment is not necessarily acceptable in others**
- **less efficient does not mean inadequate**
- **continual improvement means there may be findings even against a satisfied KPA**

Example: we expect configuration management in a 3-person project to be significantly different from CM for a 300-person project...

Adequacy versus Goodness

The objective of process assessment is not to evaluate "goodness" in an absolute sense!

Reference models such as Software CMM and standards such as ISO 9001 say <u>what</u> to do but not <u>how</u>.

Are the practices implemented by an organization <u>adequate</u> in the judgment of a knowledgeable professional relative to the state-of-the-practice / state-of-the-art?

- adequacy and goodness are points on a continuum

<u>Goodness</u> is relative to business objectives!

Keep It Simple…

Define and deploy a useful, usable process quickly

Start with the "as is" process rather than the "should be" process

Iterate rapidly to improve the process – "agile" process improvement

USA principle: <u>U</u>nderstand, <u>S</u>implify, <u>A</u>utomate

Use is determined by usefulness and usability.

Rules of Thumb
for Process Definition

Process descriptions should be 1-2 pages long

Reference subprocesses, procedures, standards, and checklists as needed

Embed contextual information in training materials, templates, tools, etc.

Remember your software design principles!
  - **locality**
  - **low coupling**
  - **high cohesion**
  - **information hiding**
  - **abstraction**
  - **…**

Lean → *Minimizing Waste*

Many steps will be found to unambiguously create value.

Many other steps will be found to create no value but to be unavoidable with current technologies and production assets.

Many additional steps will be found to create no value and to be immediately avoidable.

Remove *muda* – waste – any human activity which absorbs resources but creates no value.

J.P. Womack and D.T. Jones, Lean Thinking, 1996.

Prerequisites for Improvement

Are you unhappy with the status quo with respect to software in your organization?
  - **dissatisfaction can be self-motivated**

Are your customers dissatisfied?
  - **customer expectations change over time**

Is the competition using software superiority to gain competitive advantage?
  - **"quality" is frequently relative to expectations**

If the answers are "No," it is unlikely that
  - *your behavior will change …*
  - *… or that your "sponsorship" will inspire change in others*

Prioritizing Business Objectives

Treacy and Wiersema define three values that a company can focus on to drive business success.

Product leadership **– features, innovation**
  • typical focus of commercial shrinkwrap companies

Customer intimacy **– niche products, relationships**
  • typical focus of IT service providers

Operational excellence **– promised features, on schedule, on budget**
  • typical focus of custom software development

Treacy and Wiersema, The Discipline of Market Leaders, 1997.

The Business of Innovation

If innovation is your business, put innovation in the driver's seat.

The competitive business of innovation requires continual trial, and ongoing program of experimentation.

A. Axelrod, Edison on Innovation, 2008.

Balancing Innovation with the Routine

Driving out variation makes sense when organizations do proven things in proven ways that still work.

- Variance in people, knowledge, activities, and organizational structures is crucial to creativity and innovation.

Famous creators generated a wider range of ideas and completed more products than their contemporaries. They didn't succeed at a higher rate than others. They simply did more.

- Most new ideas are bad and most old ideas are good.
- Doing routine work with proven methods is the right thing to do most of the time.

R.I. Sutton, Weird Ideas That Work: How to Build a Creative Company, 2007.

Adding Business Value

"Business orientation, that is, the extent to which SPI goals and actions are aligned with explicit and implicit business goals and strategies, was identified as one of the factors with the strongest influence on SPI success."

- *T. Dybå, "An Empirical Investigation of the Key Factors for Success in Software Process Improvement," IEEE Transactions on Software Engineering, May 2005*

"Management is mainly concerned with the successful implementation of PI in the sense that it proves useful to the organization. It is also interested in whether the new processes are in fact used homogeneously across the business unit."

- *J. Johansen and J. Pries-Heje, "Success with Improvement – Requires the Right Roles to be Enacted – in Symbiosis," Software Process Improvement and Practice, Nov/Dec 2007*

Questions and Answers