

Practical Software Measurement

Practical Software Measurement

A foundation for objective project management



PSM SPE Workshop

Dave Card / SPC

Ed Dudash / NSWCCD

***Office of the Under Secretary of Defense
Acquisition and Technology***

***Joint Logistics Commanders
Joint Group on Systems Engineering***

PSM 1

22 Jul 98

Practical Software Measurement

Outline

- *Workshop Objectives and Intended Output*
- *SPE Background*
- *SPE Measurement Objectives*
- *SPE Definitions*
- *SPE Scope*
- *SPE Measurement Principles*
- *Issue Categorization*
- *Recommended SPE Products*
- *SPE PSM Framework Impacts*
- *Proposed SPE Activities*

PSM 2

22 Jul 98

Practical Software Measurement

Objectives of the Workshop

- *Review SPE Work to Date*
- *Develop Plan for Going Forward*
- *Identify Significant Obstacles or Issues*

PSM 3

22 Jul 98

Practical Software Measurement

Intended Output

- *Assessment of Existing Material*
- *SPE Development Plan*
- *List of Issues to Resolve*

PSM 4

22 Jul 98

Practical Software Measurement

SPE Background

- *Initial SPE workshop held in August 1996*
 - *initial issue set developed*
- *SPE WG meetings in October 1996 and February 1997*
 - *SPE study group formed*
- *SPE study group meeting in June 1997*
- *July 1997 SPE Vail Workshop*
 - *SPE survey*
- *SPE study group meeting in July 1997*
- *February 1998 PSM TWG meeting*
 - *study group recommendations presented*

PSM 5

22 Jul 98

Practical Software Measurement

PSM-SPE Study Group

*Sean Arthur
Will Brimberry
Ed Dudash
Fred Hall
Ken Kelly
Ron Larson*

*Dick Nance
Butch Neal
Sharon Rhode
Linda Rosenberg
Lynn Simms*



PSM 6

22 Jul 98

Practical Software Measurement

1997 SPE Workshop Participants

- *Sean Arthur*
- *Ed Dudash*
- *Ken Kelly*
- *Linda Rosenberg*
- *Jeff Antoniewski*
- *Marie White*
- *James Schottenloher*
- *Robert Vienneau*
- *Michael Roberts*
- *Forrest Brown*
- *Robert Laws*
- *Wendell Mullison*
- *Bonnie Bollinger*
- *Lynn Knight*
- *Dick Newhall*



PSM 7

22 Jul 98

Practical Software Measurement

SPE Measurement Objectives

- To bring software product measurement into practice.
- To develop and promote the understanding of the relationships among the four elements: Cost, Schedule, Functionality, and Quality. All contribute to the consequent determination of performance.
- To develop a measurement approach that complements related measurement efforts in project/program management, process enhancement and systems engineering guidance, and demonstrates an awareness of on-going software measurement policies and initiatives.

PSM 8

22 Jul 98

Practical Software Measurement

SPE Definitions

- Software Product Engineering: Within a project, Software Product Engineering comprises those software engineering activities used to produce products that meet identified customer needs.
- The Customer: An SPE Customer is any person (as an individual or part of an organization) that interacts with, depends on, or relates to the software products. Customers can be categorized as follows: Users, Developers and Maintainers.
- Software Functionality: [Software] Functionality is the ability of the software system products to meet the needs of the customers as specified in the system requirements.
- Software Quality: [Software] Quality is determined by the degree to which the product functionality is met and how well that functionality meets the customers' expectations at a point in time.

PSM 9

22 Jul 98

Practical Software Measurement

SPE Definitions

Users are those individuals that need to interact with, require the services of, or employ the products from the software system to perform their primary job. The category of "users" includes: operational phase users, sponsors, and independent testers. This group of customers interact with the software and/or use other products, such as: user manuals and system outputs.

Developers are those individuals that perform developmental services on the software system during the concept, definition, design, construction, testing and implementation (pre-operational) phases of the software system's life cycle. The category of developers includes anyone associated with the development, such as: program managers, project managers, designers, COTS selectors, system integrators, programmers, development testers, and SQA groups. This group of customers produce the software and/or related products, such as: requirements, specifications, manuals, and test routines.

Maintainers are those individuals that perform sustaining or corrective services on the software system during the operational phase of the software system's life cycle. The category of maintainers includes: operational managers, systems operators, maintenance programmers and other service support agents. This group of customers maintain (or sustain) the software and/or other related products, such as: manuals, SOPs (standard operating procedures), and system logs.

PSM 10

22 Jul 98

Practical Software Measurement

SPE Scope

- All issues, concerns and measures considered during the SPE effort should focus on assessing only product functionality and quality.
- Defined measures should focus on assessing functionality and quality pertaining to a single project. Inter-project comparison is not an objective.
- The SPE effort should investigate and address the issues and concerns from the perspective of three distinct customer categories: the user, the developer and the maintainer.
- Only those issues that fall within the Software Engineering domain should be considered within the SPE effort
- Excluded from consideration are support services and training-related activities and material (provided they have not been stipulated as deliverables).

PSM 11

22 Jul 98

Practical Software Measurement

SPE Scope

- In defining measures that address product functionality and quality, both process artifacts and product deliverables should be subject to measurement.
- The effort should investigate and incorporate measures that enable the customer to both assess and predict functionality and quality.
- The effort should investigate and incorporate measures that enable one to assess the beneficial (as well as detrimental) impact on the product.
- SPE quality measures should include an examination of both internal and external characteristics.
- The effort should investigate and include (if possible) measures that help to determine the completeness with which user expectations are captured. This includes accuracy in interpreting customer needs and accuracy in recording that interpretation.

PSM 12

22 Jul 98

Practical Software Measurement

SPE Measurement Principles

A principle is a fundamental truth, law, doctrine, or motivating force upon which others are based

- All measures should be issue-driven.
- SPE measures are neither absolute nor individually sufficient. Each measure is interpreted according to an established measurement range (yet to be determined), and multiple measures are generally required to provide a comprehensive characterization of product quality.
- SPE measures support two purposes in product evaluation: assessing the quality of a product (or product component), and predicting the quality of the delivered product. Measures for both assessment and prediction should be applied to product components and process artifacts.

PSM 13

22 Jul 98

Practical Software Measurement

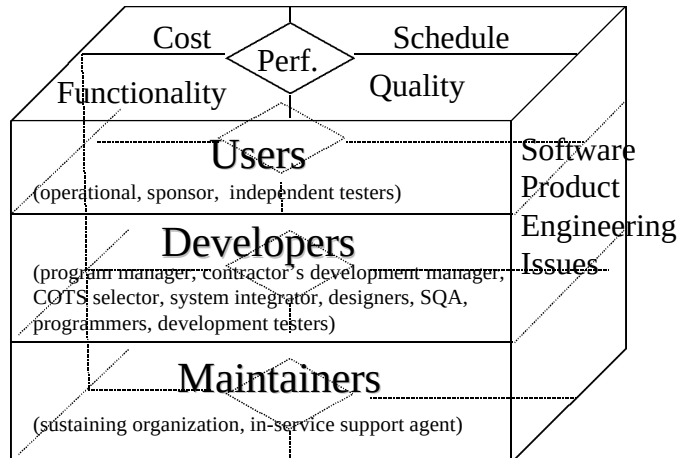
SPE Measurement Principles

- Activities can be performed during development and maintenance in ways that enhance or reduce product quality. SPE measurement must seek to reveal both positive and negative influences.
- The SPE measurement approach should be applicable irrespective of life-cycle model or methodology; however some tailoring is both expected and desirable.
- SPE measures need to support all customers, i.e. users, developers, & maintainers. Such measures must accommodate different perspectives and informational needs of each category of customer.
- SPE measures should focus on an examination of product functionality and quality.

PSM 14

22 Jul 98

Practical Software Measurement PSM Framework Extension



Issue-driven measurement key categories: cost, schedule, functionality, quality, performance
Customer base: users, developers, maintainers
SPE issues: related to functionality and quality

PSM 15

22 Jul 98

Practical Software Measurement The Original Set of "Low-Level" Issues (Partitioned Along Functionality and Quality)

Issue	Fct	Qual	Both	Other	Out	E/O
Definition of quality requirements	1	4		1	1	Def
Definition of quality	1	5		1	1	Def
Acceptance criteria	2	4	1		2	Def
Definition of a failure	2	4		1	1	Def
Standard product quality ranges	1	4	1	2		Def
Product Size	2	1		1	2	OoS
Efficiency of product	3	3				?
Rework	2	3		1		OoS
Predictability of attainment of quality objectives	1	5		1	1	Princi
Adequacy of documentation [2]	1	4	1			Qual
Test coverage						Sean
Defect recognition		6			1	Qual
Safety	1	3	2			Qual
Ease of use	2	3	1			Qual
Supportability	1	4	1			Qual
Usability (same as ease of use)	2	3			1	Qual
Quality Flow Through (not used)	1	6	1			Qual
Reliability	1	5	1		1	Qual
Maintainability	1	5	1			Qual
Product Correctness	2	3	1			Qual
Readiness to deliver [1] / [3F]	1	2	2		1	Qual
Reusability [2]	1	4		1		Qual
Traceability [3]	1	4		1		Qual
Complexity [2]	2	3		2		Qual
Degree of software reuse	1	3	2			Qual
Stability of requirements (used in functionality)	2	3	1		1	Qual
Prediction of defects	1	4			1	Qual
User satisfaction	2	3		1	1	Qual
Quality of NDI (COTS & GOTS)	1	5		1		Qual
Clarity	1	3	1	1		Qual
Testability	4	1	1			Fct
completeness	4	1	1			Fct
Hard vs. soft/fuzzy requirements	1	1				Fct
Conflicting requirements	4	1	1	1		Fct
Requirements understandability	4	2				Fct
Real-time performance	4	1	1			Fct
Are evolved requirements what user intended?	4	1	1		1	Fct
Portability	3	2		1		Fct
Scalability	4	2				Fct
Security	5	1				Fct
Precision	4	2				Fct
Degree of COTS	3	1	2			Fct
Requirements volatility	3	2	1		1	Fct
MTTR	3	2		2		Fct
Fault tolerance	3	1	1	1		Fct
Conformance to standards	5	2				Fct
Consistency (used in quality)	3	2		1		Fct
Interfaces	5				1	Fct
Interoperability	5	1				Fct
Common operating environment (not used)	4			1	1	Fct

PSM 16

22 Jul 98

Practical Software Measurement Issue Categorization

Functionality Measurement Category	Quality Measurement Category	Questions Addressed	Measures (lower level measure issues)
Operational Performance		Given its current functional capabilities, does the software perform adequately?	(real-time performance, security, precision, fault tolerance, MTTR, interfaces, readiness to deliver)
Operational Adaptability		How easily can we include new technology? How easily can functionality be added?	(portability, scalability, interoperability)
Requirements Applicability		Are the requirements well-understood? How accurately has that understanding been captured? Are the requirements changing?	(requirements understandability, conflicting requirements, requirements volatility, hard vs. soft requirements)
Requirements Validity		Is the expected functionality present? Is the software testable and maintainable?	(completeness, testability, customer intent met, readiness to deliver)
Structural Integrity		Does the software adhere to the documented standards? Do NDI (COTS & GOTS) components possess structural characteristics similar to components of developed software?	(degree of NDI [COTS & GOTS] conformance to standards, readiness to deliver)

PSM 17

22 Jul 98

Practical Software Measurement Issue Categorization

Functionality Measurement Category	Quality Measurement Category	Questions Addressed	Measures (lower level measure issues)
	System Software Functionality	To what extent does the software system as a whole meet functional requirements? How difficult will it be to develop the software?	(correctness, consistency, readiness to deliver)
	Software Component Functionality	To what extent do individual software components contribute to or adversely impact functional quality? Can the NDI (COTS & GOTS) products be relied on?	(test coverage, defect recognition, requirements stability, traceability, defect prediction, NDI [COTS & GOTS] quality)
	User Needs	Does the software meet its intended use? Is the software safe to use? Is the software user friendly?	(safety, ease of use, satisfaction, reliability, adequacy of documentation)
	Developer Needs	Are the current requirements sufficiently well-defined so that development can begin/continue? Is the functionality changing?	(reusability, clarity, traceability, complexity, degree of SW reuse, reliability, adequacy of documentation)
	Maintainer Needs	How difficult will it be to maintain the software? Will the computer program documentation support maintenance activities? How much rework for development defects is required?	(supportability, reusability, traceability, complexity, maintainability, reliability, adequacy of documentation)

PSM 18

22 Jul 98

Practical Software Measurement

Recommended SPE Products

- **Guidebook:** The intent of the Guidebook is similar to the existing PSM Handbook in that it outlines the framework within which product measurement should occur and the philosophy underlying the measures and measurement approach.
- **Tool:** Like the PSM Insight Tool, a similar software product needs to be developed that assists the customer in tailoring a measurement program to his/her specific needs.
- **Expert System:** A second tool that links to the "SPE Insight" tool and guides the customer through one or more interpretation scenarios.
 - Guidance should be interactive
 - Guidance should provide timely assistance in interpreting *actual* measures, i.e., those related to an actual software development effort.
 - "Expert System" would be invoked through the Insight tool after it has determined the necessary measures.

PSM 19

22 Jul 98

Practical Software Measurement

SPE PSM Framework Impacts

Occurrences

- | | |
|---|----|
| • SPE Principles | 5 |
| • SPE Issues/Categories/Questions | 8 |
| • Audience change from Manager to Customer | 6 |
| • Life Cycle change from Development to Development <i>and</i> Maintenance | 5 |
| • Expanded focus of Product examination to include life cycle artifacts | 1 |
| • Analysis to include <ul style="list-style-type: none"> – assessment/prediction concepts – guidance as to what to do or NOT to do to impact product quality or functionality | 6 |
| • Defined PSM processes and document updates need to include SPE | 15 |
| • Links to & Conflicts with SYS-ENGR | 5 |

PSM 20

22 Jul 98

Practical Software Measurement Proposed SPE Activities

Review, revise, adopt items recommended by the Study Group: Objectives, Definitions, Scope, Principles, Issue Categorization, SPE Products.

Establish SPE Team and define level of effort.

Examine and resolve SPE impact on PSM-PE and PSM-SYS; begin integration process

Investigate existing measurement initiatives to determine how the SPE measurement framework can utilize/incorporate them in its development/measurement framework, or accommodate them in a complementary fashion

Adopt or evolve a framework to guide measure: customer needs => issues, issues => questions
data items and metric addressing issue/question
justification for data item selection and metric formulation

Develop Measures: Identify/formulate customer needs
Relate needs to issues
Identify critical data items related to issues
Formulate metrics
Justify : Linkage between needs/issues, Linkage between issues/data items, Relationship among data items as defined by the metric, Metric as representative measure of question/issue

Formulate interpretation guidelines for each measure

Develop case study

Develop Products: Guidebook, "Insight" Tool, Expert System

Identify and implement technology transfer program

PSM 21

22 Jul 98

Practical Software Measurement Recommended SPE Development Plan

Review, revise, adopt recommendations

Establish team and define level of effort

Examine and resolve SPE impact on PSM

Investigate, incorporate existing measurement approaches

Adopt/evolve framework to guide measure development

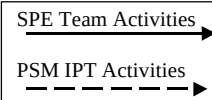
Develop measures

Formulate interpretation guidelines

Develop case study

Develop products

Identify and implement technology transfer program



PSM 22

22 Jul 98