Measuring Software Product Quality: the ISO 25000 Series and CMMI

European SEPG
June 14, 2004

Dave Zubrow
Objectives

Provide status on a new Software Product Quality Measurement standard and its connection to CMMI

Provide ideas on how to get started with Software Product Quality Measurement today
Outline

Background and Overview

Concepts and Models

Software Product Quality Measurement

Summary
Achieving Quality Software

Requires planning and intentional design

More than achieving the desired functionality

Must explicitly attend to both functional and non-functional requirements

Need to verify all requirements are being met throughout the life cycle
CMMI Definition for Quality Requirements

Requirements Development

This process area describes three types of requirements:
• customer requirements (quality in use)
• product requirements (external quality attributes)
• product-component requirements (internal quality attributes)

Taken together, these requirements address the needs of relevant stakeholders, including those pertinent to various product life-cycle phases (e.g., acceptance testing criteria) and product attributes (e.g., safety, reliability, maintainability).

Requirements also address constraints caused by the selection of design solutions (e.g., integration of commercial off-the-shelf products).
Requirements Development Goals

SG 1  Develop Customer Requirements
Stakeholder needs, expectations, constraints, and interfaces are collected and translated into customer requirements.

SG 2  Develop Product Requirements
Customer requirements are refined and elaborated to develop product and product-component requirements.

SG 3  Analyze and Validate Requirements
The requirements are analyzed and validated, and a definition of required functionality is developed.
Process Management and Performance

The organization’s process needs and objectives cover aspects that include the following:
- characteristics of the processes
- process performance objectives, such as time to market and **product quality**
- process effectiveness

A quantitatively managed process is institutionalized by doing the following:
- controlling the process using statistical and other quantitative techniques such that **product quality**, service quality, and process performance attributes are measurable and controlled throughout the project (**internal and external quality measures and criteria**)
Key Points in Relationship of CMMI and ISO 9126/25000 - 1

CMMI takes a total life cycle view and is inclusive in its approach to requirements development.

Requirements development explicitly seeks to have the developer consider quality requirements.

Project and Process Management processes explicitly consider product quality as process performance objectives.

Neither the standard nor CMMI endorses a uni-dimensional view of quality.
Key Points in Relationship of CMMI and ISO 9126/25000 - 2

Product Quality Requirements are transformed into designs and implemented via the Technical Solution and Product Integration process areas.

The implementation of Product Quality Requirements are monitored and confirmed via the Project Management, Verification, and Validation process areas.

CMMI acknowledges the need for interaction and perhaps iteration among the related process areas to satisfactorily identify, specify, and address Product Quality Requirements.
Relating Requirements, Evaluation, and Measurement

Customer
User
Development Organization

provide guidance

Product Quality
Requirements

Product Quality
Evaluation

determines

Product Quality
Measurement

supports

Acquirer
Evaluator
Developer

guidance perform

guidance supports
Outline

Background and Overview

Concepts and Models

Software Product Quality Measurement

Summary
Internal and External Quality Requirements may be stated in coding standards, project quality goal statements, process descriptions (e.g., exit criteria), test case descriptions, etc. They need not be explicitly identified as requirements.
The Product Quality Measurement Reference Model
Quality In Use Model (ISO/IEC 9126)

- Quality In Use
  - Effectiveness
  - Productivity
  - Safety
  - Satisfaction
Internal and External Software Quality Model (ISO/IEC 9126)

Quality Characteristics
- Functionality
  - Suitability
  - Accuracy
  - Interoperability
  - Security
  - Compliance
- Reliability
  - Maturity
  - Fault tolerance
  - Recoverability
  - Compliance
- Usability
  - Understandability
  - Learnability
  - Operability
  - Comp
  - Attractiveness
- Efficiency
  - Time behavior
  - Resource utilization
  - Compliance
- Maintainability
  - Analyzability
  - Changeability
  - Stability
  - Testability
  - Compliance
- Portability
  - Adaptability
  - Installability
  - Co-existence
  - Replaceability
  - Comp

© 2004 by Carnegie Mellon University
Outline

Background and Overview

Concepts and Models

Software Product Quality Measurement

Summary
Quality Model Elements and Measurement Model Elements

Conceptual Model

- Characteristic
  - Subcharacteristic
    - Attribute

Operationalization

- Quality Measure
  - Measurement Primitive
  - Quality Measure

represents

comprise

One or more produce

One or more produce

represents

represents
Relating the Quality and Measurement Models

- Software Product Quality
  - Quality Characteristics
    - Quality Sub-characteristics
      - Quality Attributes
  - Quality Evaluation Report
    - Assessment Analysis Rating
      - Quality Measures
        - Function (Formula)
          - Measurement Primitives
          - Measurement Method
            - Quality Attribute
Relating the Quality Measurement Model to the ISO Software Measurement Process (15939)

Quality Evaluation Report
- Assessment
- Analysis Rating

Quality Measures
- Function (Formula)
- Measurement Primitives
- Measurement Method
- Quality Attribute

Information Product
- Interpretation
- Indicator
- Analysis Model
- Derived Measures
- Measurement Function
- Base Measures
- Measurement Method
- Attribute

Analysis 25030 25040
Definition & Collection 2502n
CMMI Measurement & Analysis
Process Area Goals

Align Measurement and Analysis Activities

Provide Measurement Results

Institutionalize a Managed Process
Activities for Goal 1

Align Measurement and Analysis Activities
• Establish Measurement Objectives
• Specify Measures
• Specify Data Collection and Storage Procedures
• Specify Analysis Procedures

Note: The first two practices directly address the need to translate from the conceptual to the operational.
Activities for Goal 2

Provide Measurement Results
  • Collect Measurement Data
  • Analyze Measurement Data
  • Store Data and Results
  • Communicate Results
Mapping of M&A Practices to Indicator Template

INDICATOR TEMPLATE

Measurement Goal #____:

Objective
Questions
Visual Display

Input(s)
Data Elements
Responsibility for Reporting
Form(s)

Algorithm
Assumptions
Interpretation

X-reference
Probing Questions
Evolution

Establish Measurement Objectives
Specify Measures
Specify Data Collection Procedures
Specify Analysis Procedures
Collect Data
Analyze Data
Communicate Results
Store Data & Results
Measuring External Quality to Manage Software Development

Quality Characteristic/Subcharacteristic: Efficiency/Time Behavior
Operational Measure: Response Time

Objective: Track satisfaction of user requirement for system response time.

Questions: What is the system response time with respect to common transaction? What is the variability in response time?
Outline

Background and Overview

Concepts and Models

Software Product Quality Measurement

Summary
Summary

Measurement links the specification of requirements to acceptance criteria

Quality is conceptual; measurement is operational.

GQ(I)M provides a means for moving from the conceptual to the operational.

The ISO 25000 series and the GQ(I)M Indicator Template together can help with your implementation of CMMI Requirements Development, Verification, and Validation.
Contact Information

Dave Zubrow
3118 SEI
4500 Fifth Ave
Pittsburgh, Pa 15213
USA

+1-412-268-5243 (v)
+1-412-268-5758 (f)

dz@sei.cmu.edu