With some changes for the PSM Conference

Increasing the Functionality of Metrics through Standardization

James W. Moore
The MITRE Corporation
James.W.Moore@ieee.org

The views and opinions expressed in this presentation are those of the author and do not reflect MITRE's current work program.

originally presented to the conference on
Developing Strategic I/T Metrics
Washington, D.C, May 12-14, 1998

Outline

- Introduction to software engineering standards
- Problems in applying SWE standards
- *... and some solutions*
- Integration of 9001/9000-3/12207/IEEE Collection
- Standards for software measurement

Why are Standards Important?

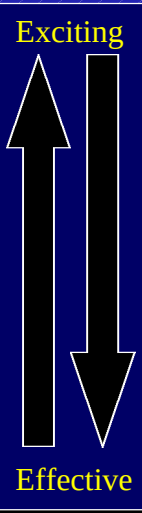
- They consolidate existing technology into a firm basis for introducing newer technology
- They increase professional discipline
- They protect the business
- They protect the buyer
- They improve the product

Software Engineering Standards

- These standards concern the *practices* that should be followed by software practitioners in applying a systematic, disciplined, quantifiable approach to software development, operation and maintenance.
- They are generally *process* standards, hence different than the more familiar *product* standards that describe the interfaces or behavior of products.
- They are playing a larger role:
  - Government shift to commercial standards, e.g.:
    - DoD: IEEE/EIA 12207 instead of DoD 2167A
    - FDA: regulation of medical devices
  - Protection from charges of “recklessness” in SW development
  - Benchmarking, e.g. FP&L won Deming Award
  - ABBET is considering accreditation of a software engineering curriculum

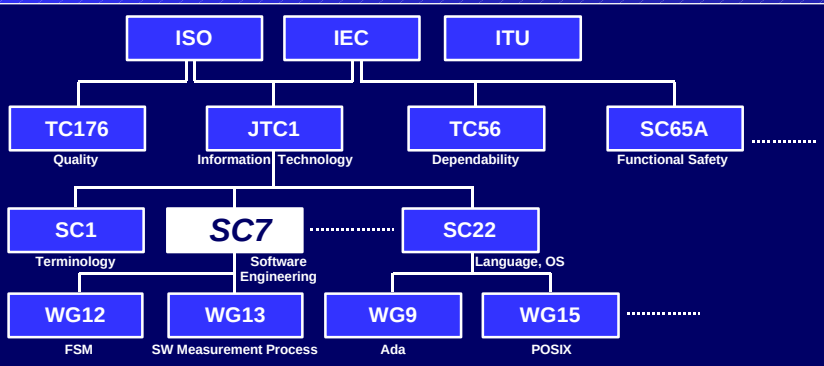
What do Standards Do?

- Specify techniques to develop software faster, cheaper, better, IEEE 982.1 (Measures for Reliable SW)
- Describe "best practices", IEEE 1008 (Unit Testing)
- Provide a systematic treatment of "ilities", IEEE 730 (SW Quality Assurance)
- Provide consensus validity for techniques that cannot be scientifically validated, IEEE 1061 (SW Quality Metrics)
- Provide uniformity in cases where agreement is more important than marginal improvement, IEEE P1320.1 (IDEF0)
- Provide a framework for communication between buyer and seller, IEEE/EIA 12207 (SW Life Cycle Processes)
- Give succinct, precise names to concepts that are otherwise fuzzy, complex, detailed and multi-dimensional, IEEE 1028 (SW Reviews)



7/29/98 15:12 MITRE 5

International Standards Organizations



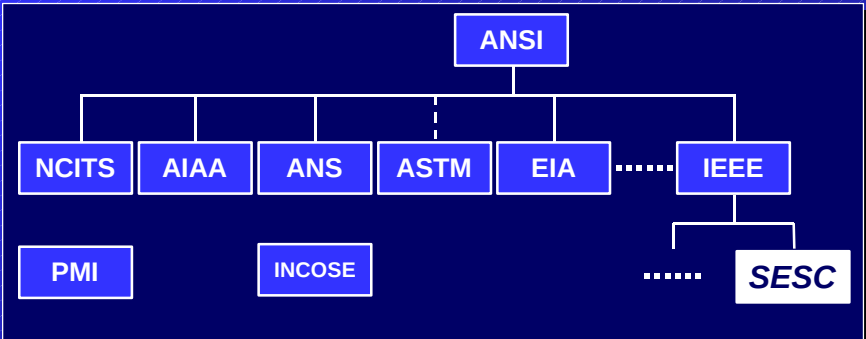
The focal point in international standards is ISO/IEC JTC1/SC7.

Other committees, though, deal with related work.

Members of these committees are "national bodies," i.e. countries, represented by "national delegations."

7/29/98 15:12 MITRE 6

Relevant U.S. Standards Organizations



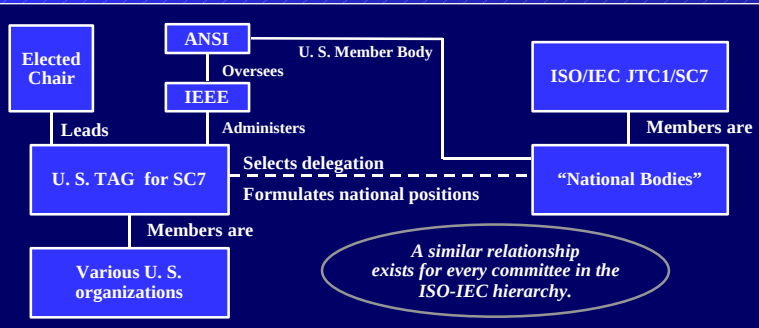
About 550 organizations in the U. S. make standards.

About half of them are accredited by ANSI, allowing them to participate in international standardization activity.

The focal point (in the U. S.) is the Software Engineering Standards Committee of the IEEE Computer Society

7/29/98 15:12 MITRE 7

U. S. Participates in SC7 via a Technical Advisory Group (TAG)



- ANSI oversees that procedures are observed.
- TAG administrator (IEEE in this case) enforces procedures.
- Elected chair provides leadership in developing US positions.
- About 35 organizations are TAG members, e.g. ASQC, IEEE SESC, ICCP, IFPUG, AT&T, IBM, Lockheed-Martin, TRW, DoD, DoE, FDA, NIST, MITRE, SEI.

7/29/98 15:12 MITRE 8

Problems in Applying Software Engineering Standards

- There are too many SWE standards
  - 315 of them, according to *[Magee97]*
  - ***How do you choose?***
- There are too many providers, with uncoordinated offerings
  - 46 different organizations, according to *[Magee97]*
  - ***How do you select appropriate ones?***
- Even from a single provider, the standards may be inconsistent
  - They are written by individually with little overall strategy
  - ***How do you assemble groups of consistent standards?***
- Different disciplines impose uncoordinated requirements
  - QM, PM, and SWE standards are capriciously different
  - ***How do you do multi-disciplinary process improvement?***

7/29/98 15:12 MITRE 9

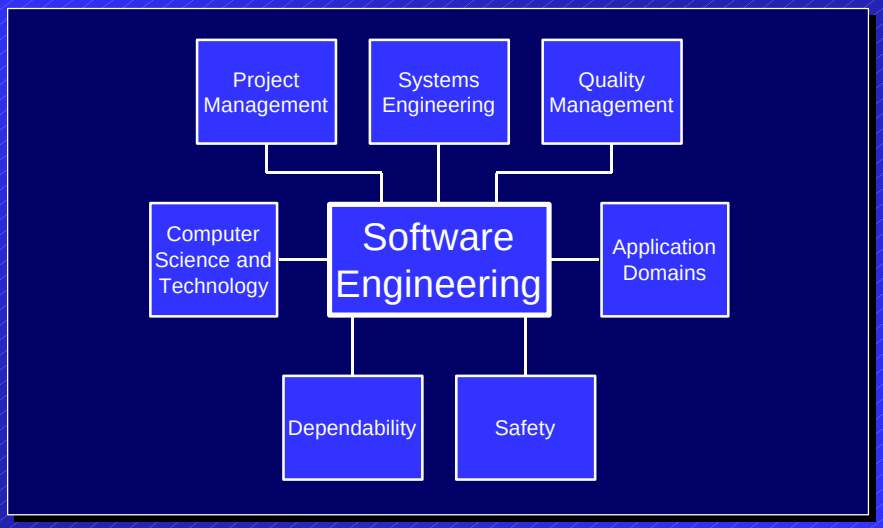
Solutions

- ***IEEE SESC wants to be a provider of an integrated set of software engineering standards***

- Attention to the context of software engineering
- An overall, integrating framework
- An overall guide to the collection
- Adoption of key, integrating standards -- even from other providers
- Strategically driven revisions to the individual standards

7/29/98 15:12 MITRE 10

Integrating Framework: Context of Software Engineering

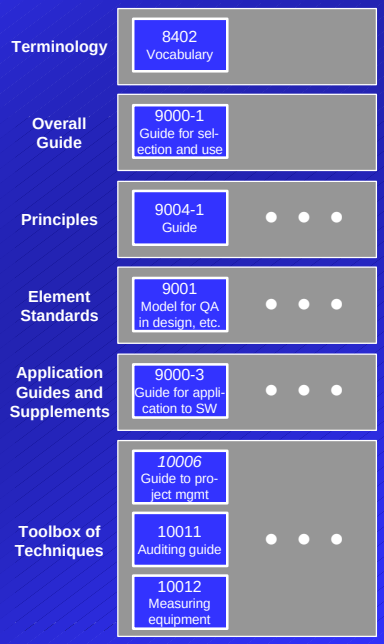


7/29/98 15:12 MITRE 11

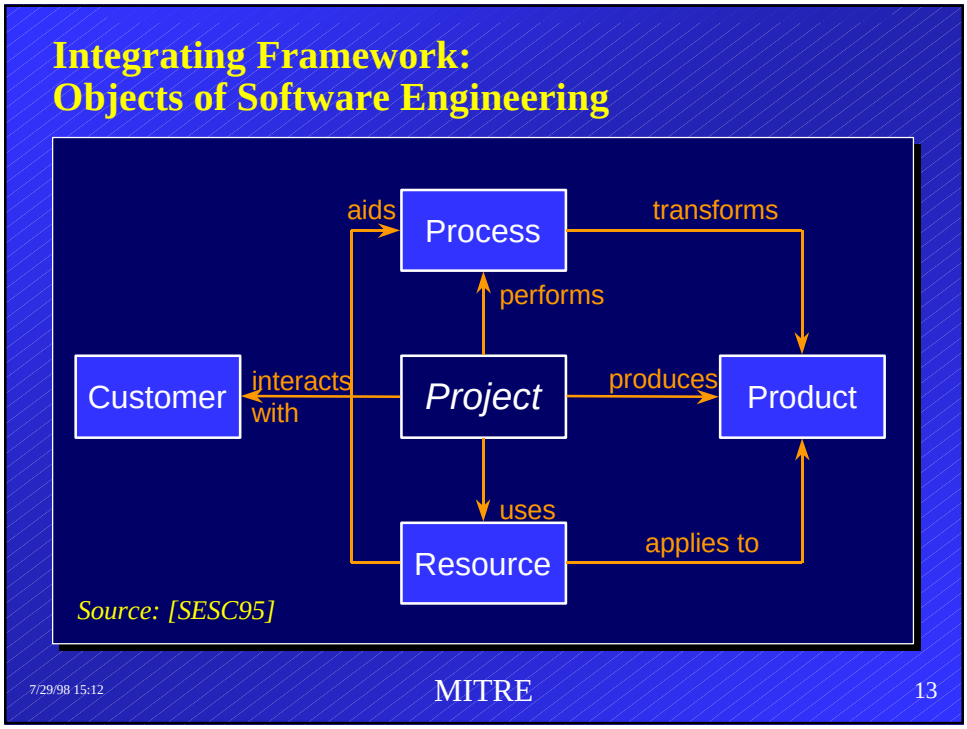
Integrating Framework: Normative Intent

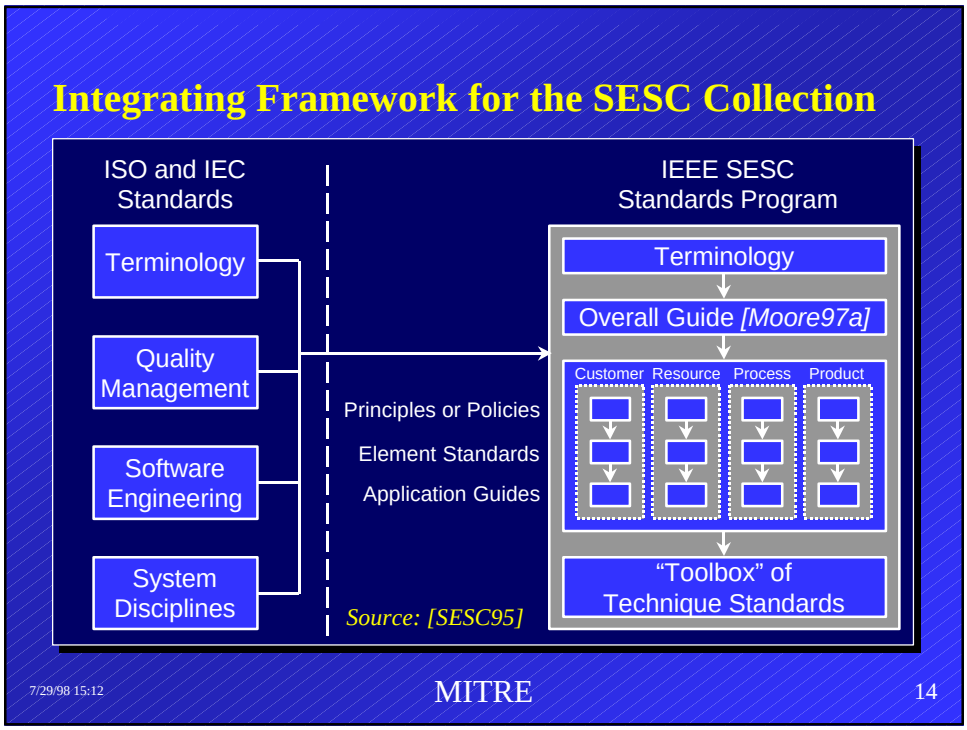
- Model originally developed by IEC TC56
- Example shown in taken from the "ISO 9000" quality management standards

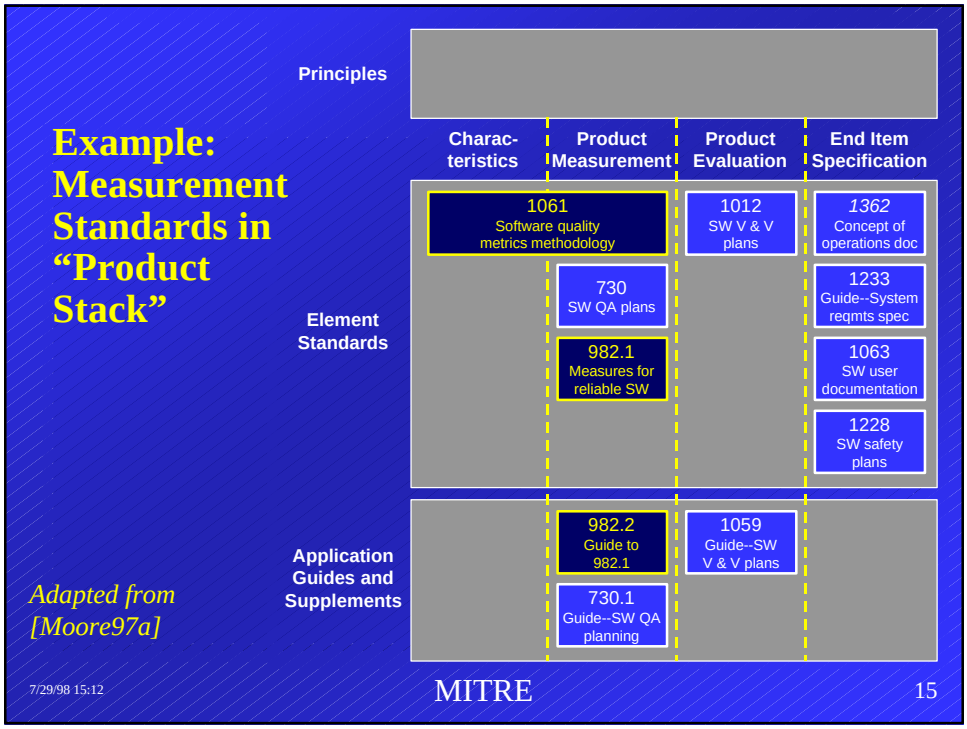
Adapted from [Moore97a]

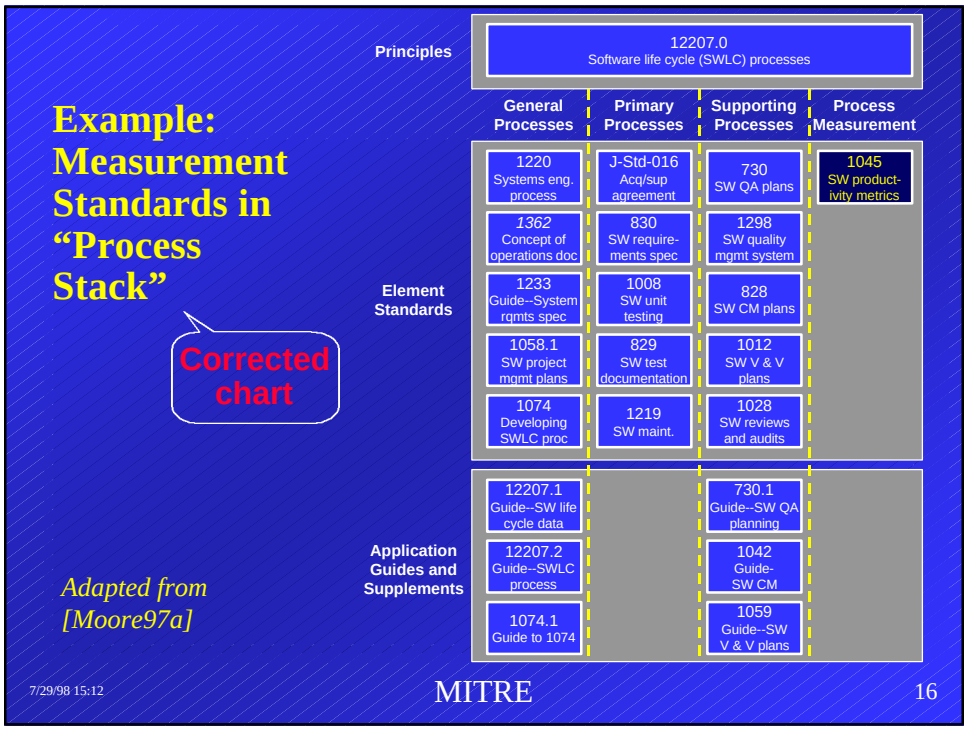


7/29/98 15:12 MITRE 12


Integrating Framework:
Objects of Software Engineering
aids
Process
transforms
performs
Customer
interacts with
Project
produces
Product
uses
Resource
applies to
Source: [SESC95]
7/29/98 15:12
MITRE
13

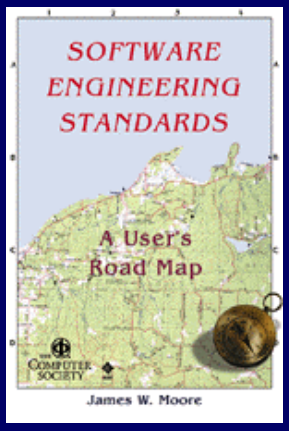

Integrating Framework for the SESC Collection
ISO and IEC Standards
Terminology
Quality Management
Software Engineering
System Disciplines
Principles or Policies
Element Standards
Application Guides
Source: [SESC95]
IEEE SESC Standards Program
Terminology
Overall Guide [Moore97a]
Customer
Resource
Process
Product
"Toolbox" of Technique Standards
7/29/98 15:12
MITRE
14


Example: Measurement Standards in "Product Stack"
Principles
Charac-teristics
Product Measurement
Product Evaluation
End Item Specification
1061 Software quality metrics methodology
1012 SW V & V plans
1362 Concept of operations doc
730 SW QA plans
1233 Guide--System reqmts spec
982.1 Measures for reliable SW
1063 SW user documentation
1228 SW safety plans
Element Standards
982.2 Guide to 982.1
1059 Guide--SW V & V plans
730.1 Guide--SW QA planning
Application Guides and Supplements
Adapted from [Moore97a]
7/29/98 15:12
MITRE
15


Example: Measurement Standards in "Process Stack"
Corrected chart
Principles
12207.0 Software life cycle (SWLC) processes
General Processes
Primary Processes
Supporting Processes
Process Measurement
1220 Systems eng. process
J-Std-016 Acq/sup agreement
730 SW QA plans
1045 SW product-ivity metrics
1362 Concept of operations doc
830 SW require-ments spec
1298 SW quality mgmt system
1233 Guide--System rqmts spec
1008 SW unit testing
828 SW CM plans
1058.1 SW project mgmt plans
829 SW test documentation
1012 SW V & V plans
1074 Developing SWLC proc
1219 SW maint.
1028 SW reviews and audits
Element Standards
12207.1 Guide--SW life cycle data
730.1 Guide--SW QA planning
12207.2 Guide--SWLC process
1042 Guide-SW CM
1074.1 Guide to 1074
1059 Guide--SW V & V plans
Application Guides and Supplements
Adapted from [Moore97a]
7/29/98 15:12
MITRE
16

Overall Guide to the IEEE Collection

- Rather than developing a consensus document, IEEE SESC "commissioned" the writing of an overall guide to the collection.
- Much of this presentation is adapted from that book.
- Relevant standards are "sliced and diced" by:
  - Context
  - Object
  - Normative intent
  - Provider and subject
- Relationships are explained, even between standards from different providers.



7/29/98 15:12 MITRE 17

Adoption of Key, Integrating Standards

- IEEE SESC has adopted standards as key, integrating standards for the overall collection
  - ISO 9001 / 9000-3, Quality Management
  - ISO/IEC 12207, Software Life Cycle Processes
  - Project Management Institute, Guide to the PM Body of Knowledge
- Other IEEE standards are being revised to fit neatly within the structure:
  - For example, all of the IEEE standards describing plans are being revised to correspond to the requirements of 12207.
  - For example, IEEE 1058, SW Project Management Plan, will be a specialization of the principles of the PMBOK.

This chart was changed.

7/29/98 15:12 MITRE 18

IEEE/EIA 12207 Integrates Software Process Life Cycle Standards

Adapted from IEEE/EIA 12207. Used by permission.

Process Class	12207 Process	Relevant IEEE Standards
Primary	Acquisition	*982.1*, 1062, 1228
	Supply	
	Development	829, 830, 1008, 1016, 1028, 1074, 1228
	Operation	
	Maintenance	1219
Supporting	Documentation	
	CM	828, 1012
	QA	730, *1061*
	Verification	1012
	Validation	1012
	Joint Review	1028
	Audit	1028
	Problem Resolution	1044,
Organizational	Management	*1045*, 1058
	Infrastructure	1209. 1348, 1420
	Improvement	
	Training	

7/29/98 15:12 MITRE 19

ISO 9000-3: 1997

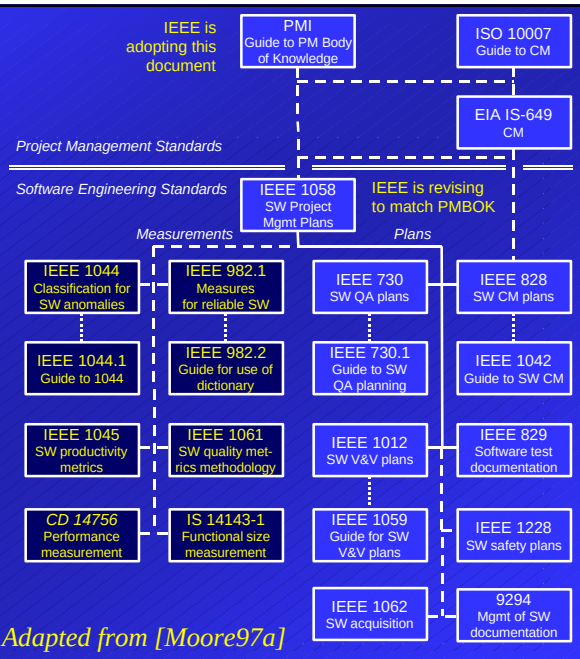
The Bridge between QM and Software Engineering

Clause	Title	Cited ISO standards	Cited Clauses of 12207	Useful IEEE Standards
4.1.2	Organization		6.3.1.6, 7.2	730
4.1.3	Management review		7.1.4	730, 1058, 1028
4.2.1	Quality system—General	10013		730
4.2.3	Quality planning	10005, 10007	6.2 - 6.5	730, 828, 1012,*1045*, 1028,*1061*
4.3.2	Contract review		5.2.1, 5.2.6, 6.4.2.1	1012, 1028
4.3.3	Amendment to a contract		5.1.3.5, 5.2.3.2	
4.4.1	Design control—General			1074
4.4.2	Design and development planning		5.2.4	730, 1058, *1045*, *1061*
4.4.3	Organizational and technical interfaces		5.2.6.1, 6.6.2	1028
4.4.4	Design input	9126	5.3.2 - 5.3.4	830, 1228, 1233
4.4.5	Design output		5.3.5 - 5.3.7	829, 1008, 1016, 1471, 1063
4.4.6	Design review		5.3.4.2, 5.3.5.6, 5.3.6.7, 6.6.3	1028
4.4.7	Design verification		5.3.4.2, 5.3.5.6, 5.3.5.7, 5.3.7.5, 5.3.9, 6.4	1008, 1012, 1028
4.4.8	Design validation		5.3.1, 6.5	1012, 1028
4.4.9	Design changes		5.5.2, 5.5.3, 6.2.3	828
4.5.1	Document and data control—General		6.1	828
4.6.1	Purchasing—General		5.1	1062
4.7	Control of customer-supplied product		6.1	
4.8	Product identification and traceability	10007	6.1, 6.2	828
4.9	Process control		5.3.12, 6.3.3	
4.10.1	Inspection and testing—General		5.1.5, 5.3.5.5, 5.3.6.5, 5.3.6.6, 5.3.7, 5.3.8, 5.3.9, 5.3.10, 5.3.11, 5.3.13	829, 1008
4.11.1	Control of inspection, measuring and test equipment—General		7.2	
4.11.2	Control procedure	10012		
4.12	Inspection and test status		6.2	828
4.13.1	Control of non-conforming product—General		6.2, 6.8	828, 1044
4.14.1	Corrective and preventive action—General		6.2, 6.8, 7.3	828, 1044
4.15.1	Handling, storage, packaging, preservation and delivery—General		5.2.7.1, 5.3.13.2, 6.2.6	828
4.16	Control of quality records		6.1.6.2	730
4.17	Internal quality audits	10011	6.7, 6.8, 7.3.2	1028, 1044
4.18	Training		7.4	
4.19	Servicing	9000-2	5.4.4, 5.5, 6.8	1219, 1044
4.20	Statistical techniques	9126		

7/29/98 15:12 MITRE 20

PMBOK

The Bridge from General Project Management to SW Project Management



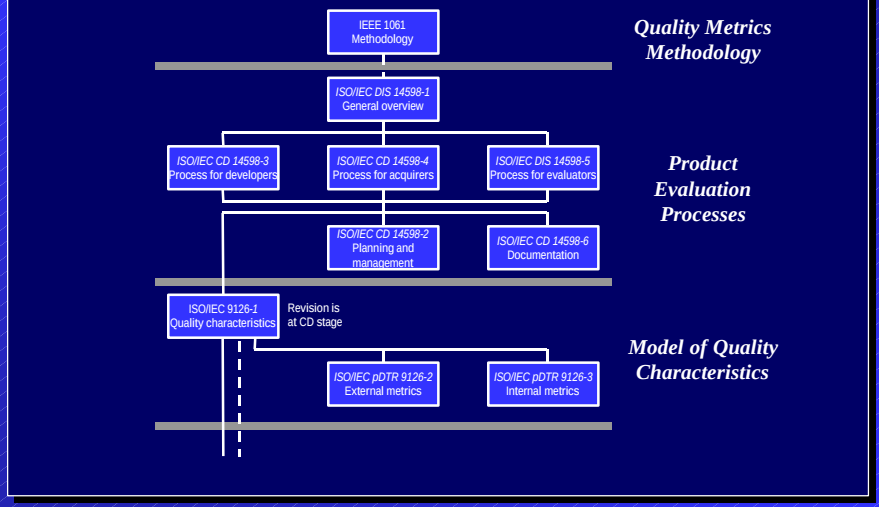
Adapted from [Moore97a]

7/29/98 15:12 MITRE 21

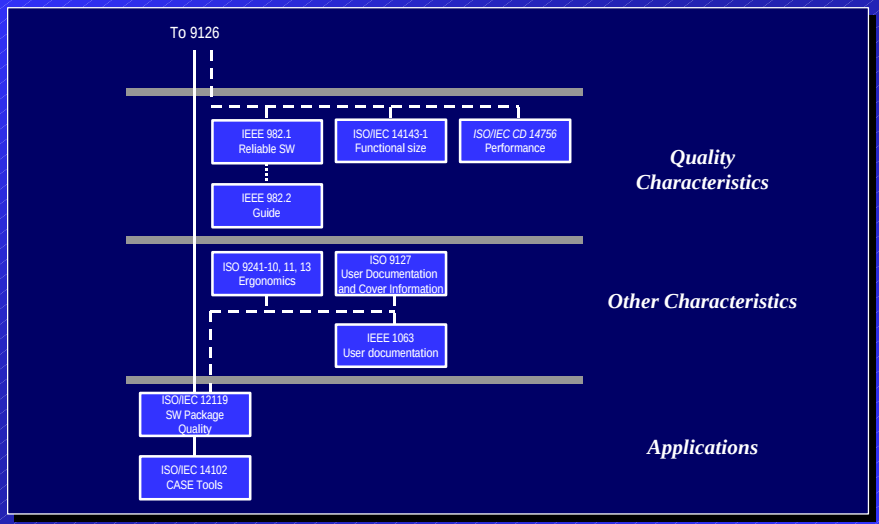
Software Measurement Standards -- in Context

7/29/98 15:12 MITRE 22

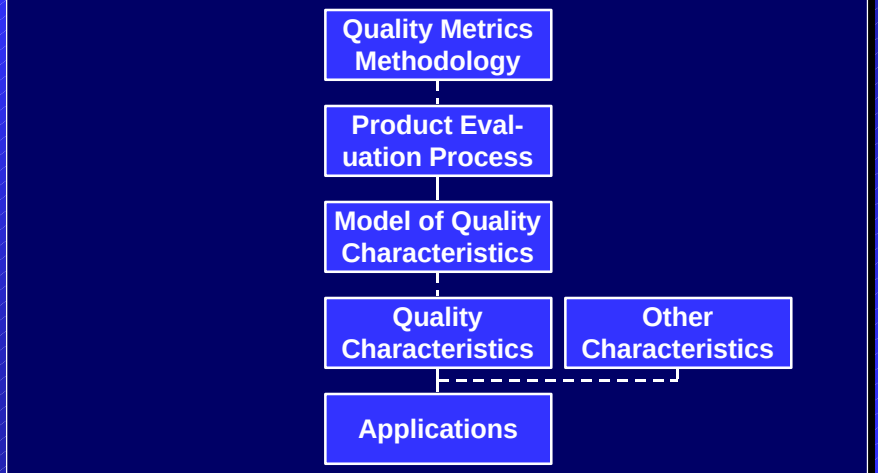
Overall Map of Standards Relevant to Software Product Evaluation (Part 1 of 2)



Overall Map of Standards Relevant to Software Product Evaluation (Part 2 of 2)



A Summary Road Map



IEEE 1061, Quality Metrics Methodology

- IEEE Std 1061-1992, Standard for a Software Quality Metrics Methodology, 88 pages. Will be revised this year.
- "Quality" is intended broadly.
- May be applied by acquirers, developers, maintainers, or independent assessors.
- Does not prescribe metrics, but gives some examples.
- Three-level hierarchy:
  1. Desired external characteristics of software product
     - = Quality requirements
     - = Quality factors representing the requirements
     - = "Direct" metrics associated with the factors
  2. Derived, developer-oriented subfactors
  3. "Indirect" metrics to measure the subfactors
- Indirect metrics are validated against the direct metrics and are used to make early forecasts of the quality factors.

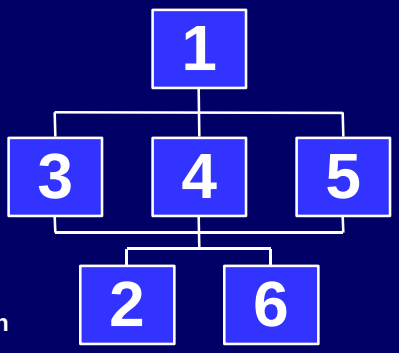
ISO/IEC (Draft) 14598, Software Product Evaluation

- Essential parts of software quality evaluation are:
  - Quality model
  - Method of evaluation
  - Software measurement
  - Supporting tools
- "Metric" is defined as a quantitative scale and method which can be used for measurement. "Measure" is the result of a measurement.
- Requires the use of
  - Quality model
  - Valid, accurate, objective, reproducible measurements

7/29/98 15:12 MITRE 27

ISO/IEC (Draft) 14598, Software Product Evaluation

- Part 1 - Overview
- Part 3 - Process for developers
- Part 4 - Process for acquirers
- Part 5 - Process for (third-party) evaluators
- Part 2 - Management of product evaluation
- Part 6 - Documentation of "evaluation modules" -- instructions and data for evaluation



Requires the use of a model of quality characteristics, such as 9126.

All of these standards are only drafts.

7/29/98 15:12 MITRE 28

ISO/IEC (Revision) 9126, SW Quality Characteristics and Sub-Characteristics

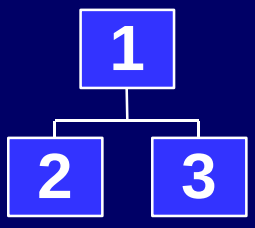
- Three parts will replace the 1991 standard, ISO/IEC 9126.
- Defines, with "minimal overlap," six characteristics (and a set of subcharacteristics) of software quality that are intended to cover all aspects inferred from ISO 9000.
- Characteristics (all externally manifested)
  - Functionality: suitability, accuracy, interoperability, security
  - Reliability: maturity, fault tolerance, recoverability
  - Usability: understandability, learnability, operability, attractiveness
  - Efficiency: time behavior, resource behavior
  - Maintainability: analyzability, changeability, stability, testability
  - Portability: adaptability, installability, co-existence, replaceability

7/29/98 15:12 MITRE 29

ISO/IEC 9126, Software Product Quality Characteristics

- Part 1 - Quality characteristics and subcharacteristics
- Part 2 - External metrics
- Part 3 - Internal metrics

This is the quality model intended for use with ISO/IEC 14598.

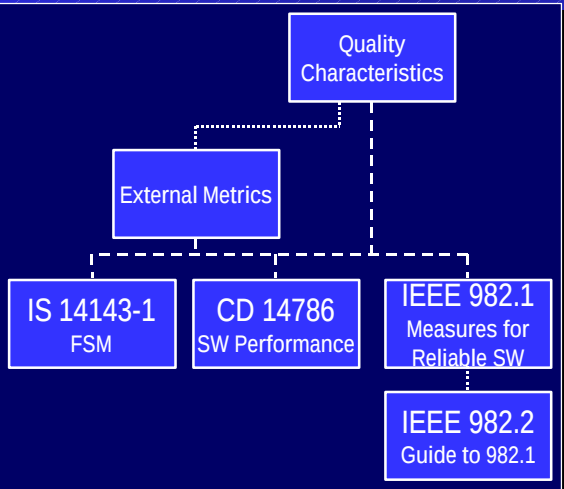


All of these standards are only drafts, except for an existing 1991 predecessor of Part 1.

7/29/98 15:12 MITRE 30

Individual Quality Characteristics

- SC7 and SESC both have standards describing characteristics that map neatly into the quality model of 9126
- They also provide external metrics, in some cases:
  - Functional size measurement
  - Performance
  - Reliability



7/29/98 15:12 MITRE 31

IEEE Software Reliability Standards

- IEEE Std 982.1, Measures to Produce Reliable Software
  - 36 pages, approved in 1988, currently being revised
  - A “catalog” or a “dictionary” of measures that can be applied to produce more reliable software
    - Definitions
    - Taxonomy
  - Emphasis on indicators that can be used early
  - Six categories of product measures
  - Three categories of process measures
  - Proper conditions for use and method of computation
  - Measures of errors, faults, and failures are primitives
- IEEE 982.2 is a guide for applying 982.1
  - 96 pages, approved in 1989

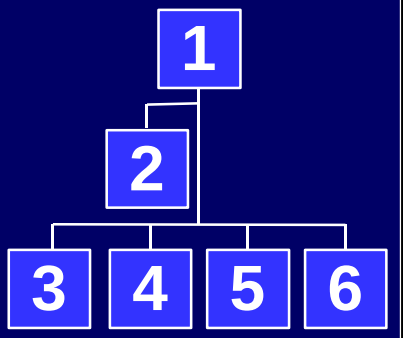
7/29/98 15:12 MITRE 32

ISO/IEC (Draft) 14786, Measurement and Rating of Performance

- Describes how to make user-oriented performance measurements using a “remote terminal emulator” to interact with the system
- A number of metrics are described

ISO/IEC 14143, Functional Size Measurement

- IFPUG representatives participate in this SC7 Working Group.
- Part 1 - Definition of functional size measurement
- Part 2 - Assessing compliance of a method
- Part 3 - Verification of a method’s claims
- Part 4 - Reference model
- Part 5 - Functional domains
- Proposed Part 6 - Process for selecting a method



Part 1 is an IS. The others are only drafts.

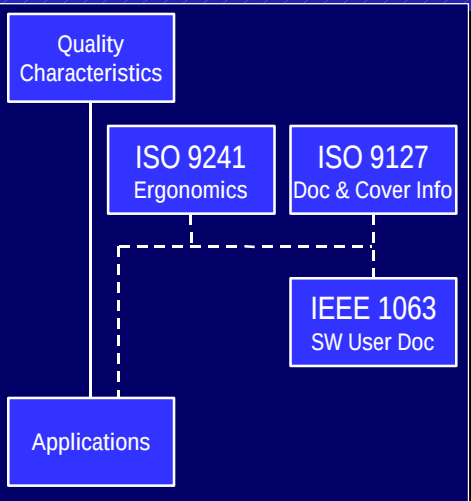
ISO/IEC 14143-1, FSM -- Definition of Concepts

- Premise: Inconsistencies have occurred in the various methods based upon Albrecht's original approach
- Purpose: Promote consistent interpretation of FSM
- Scope:
  - In:
    - Definition of fundamental concepts
    - Description of general principles for application
  - Out:
    - How to measure with any particular method
    - How to use results of measurement
    - How to select a particular method
- Audience: Persons associated with acquisition, development, use, support, maintenance and audit of SW

7/29/98 15:12 MITRE 35

Other Characteristics

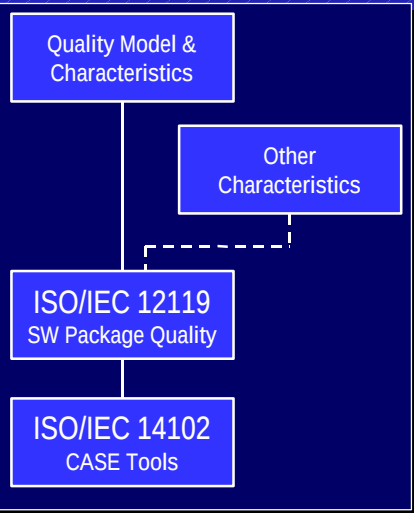
- Standards prepared by groups other than SC7/WG6 don't fit neatly under the WG6 quality model.
- ISO TC159 is preparing a multi-part standard, 9241, on ergonomics.
- When it was formed, SC7 inherited a legacy standard, 9127, for documentation of consumer software.
- IEEE 1063 prescribes requirements for a user manual.



7/29/98 15:12 MITRE 36

Applications of the Model

- ISO/IEC 12119 uses the quality model of ISO/IEC 9126, as well as some other standards, to set requirements for software "packages"
- ISO/IEC 14102 applies the requirements of 12119 to the specific category of CASE tools



Summary (1 of 2)

This chart was changed.

- ***IEEE SESC is overcoming many of the obstacles to applying software engineering standards by:***
  - Describing the context of software engineering
  - Developing an overall framework for software engineering standards
  - Structuring a collection around key, integrating standards
  - Providing an overall guide to the collection
  - ***By acting as the integrator of software engineering standards***

Summary (2 of 2)

This chart was added.

- *There are international and IEEE standards related to software measurement.*
  - They can be rationalized into a framework
  - But, they have not been engineered into a consistent set
  - They are not well-represented in the higher-level process standards.

References

[Magee97] Stan Magee and Leonard L. Tripp, *Guide to Software Engineering Standards and Specifications,* Artech House, Boston, MA, 1997.

[Moore97a] James W. Moore, *Software Engineering: A User's Road Map,* IEEE Computer Society Press, Los Alamitos, CA, 1997.

[SESC95] SESC Business Planning Group, *Vision 2000 Strategy Statement* (Final Draft), v0.9, SESC/BPG-002, August 20, 1995.

Biography

James W. Moore is a twenty-eight-year veteran of software engineering and an ten-year veteran of software engineering standardization. With degrees from the University of North Carolina and Syracuse, he has worked in both the commercial and defense sectors for IBM and, now, The MITRE Corporation, where he is the corporate focal point for standardization activities. Currently, he serves as the chairman of the international standards committee for the Ada language, the chairman of the ACM Technical Standards Committee, a member of the Management Board of IEEE Software Engineering Standards Committee (SESC), and as the Vice-Chair of the U.S. delegation to the international committee responsible for software engineering standards. He was the founder of the Reuse Library Interoperability Group (RIG) and chaired the Reuse Planning Group for SESC. He served for four years as a member of the Defense Department's Federal Advisory Board on Ada. The IEEE Computer Society has recognized him as a Charter Member of their Golden Core. His new book, *Software Engineering Standards: A User's Road Map,* was published this year by the IEEE Computer Society Press.

7/29/98 15:12 MITRE 41

Contact Information

- **Address for postal mail:**
  - **The MITRE Corporation**
    Attn: James W. Moore, M/S W534
    1820 Dolley Madison Blvd.
    McLean, VA 22102
- **Address for express delivery services:**
  - **The MITRE Corporation**
    Attn: James W. Moore, (703) 883-7396
    11493 Sunset Hills Road
    Reston, VA 22090
- **Electronic mail: James.W.Moore@ieee.org**
- **Phone numbers:**
  - **Office: +1.703.883.7396**
  - **Fax: +1.703.883.5432**

7/29/98 15:12 MITRE 42