

Professionalizing the Practice of Software Engineering

*The author's affiliation with
The MITRE Corporation is
provided for identification
purposes only, and is not
intended to convey or
imply MITRE's concurrence
with, or support for, the
positions, opinions, or
viewpoints expressed by
the author.*

James W. Moore, CSDP, F-IEEE
The MITRE Corporation
April 2009

Thanks to Those Contributing Slides

- Don Bagert, Accreditation
- Bob Becnel, SWE Licensing
- Pierre Bourque, SWEBOK Guide
- Dennis Frailey, Accreditation
- Art Pyster, GSwERC
- Chris Ruoff, Certification and Training
- Ann Sobel, Curriculum
- Steve Tockey, Certification
- Janet Wilson, SWEBOK Guide

Software Engineering is a Profession

The Future Of Software Engineering

- **Technical**
 - The dominance of component-based software engineering
 - The codification of reference architectures
 - The evolution of virtual teams
 - The acceptance of well-defined processes
 - The creation of frictionless surfaces
 - The rise of software engineering as a profession
- **Social**
 - The impact of legal issues
 - The scarcity of skilled workers
 - The growth of non-programmers

From Grady Booch, "The Future of Software"

The Growth Of Non-Programmers

"Software Engineering" will have no future unless we accept the fact that only a small fraction of software developers are qualified to be called "Software Engineers" and agree on standards that distinguish those who are qualified from the rest of us.

David Parnas

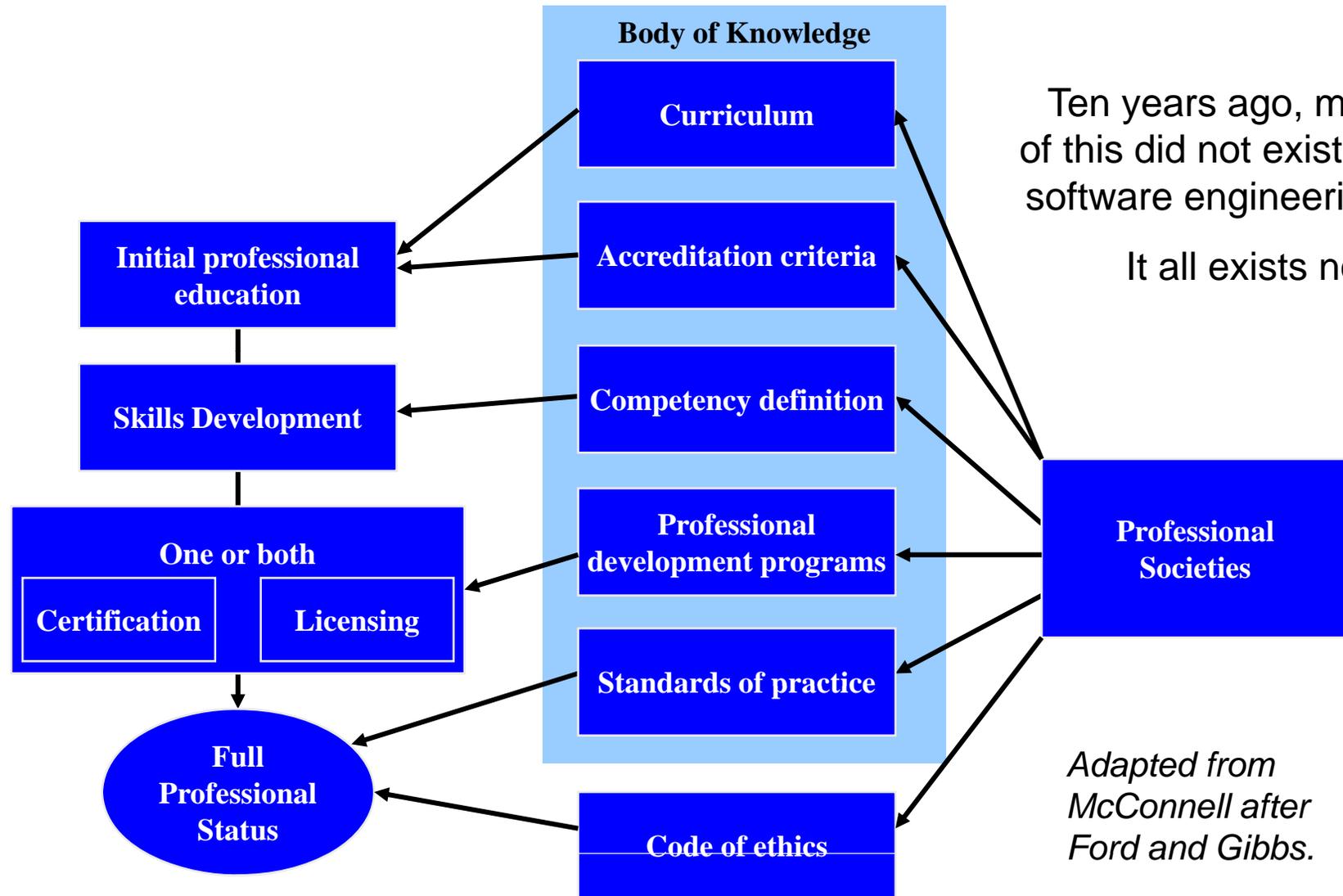
3

Outline

- An “Engineering” Model of a Software Engineering Career
- Body of Knowledge
 - SWEBOK Update
- Professional Education
 - Curriculum
 - Accreditation
 - Graduate Curriculum project
- Professional Development
 - Certification
 - Training
 - Licensing initiative
- Professional Practice
 - Standards
 - Code of Ethics

An “Engineering” Model of a Software Engineering Career

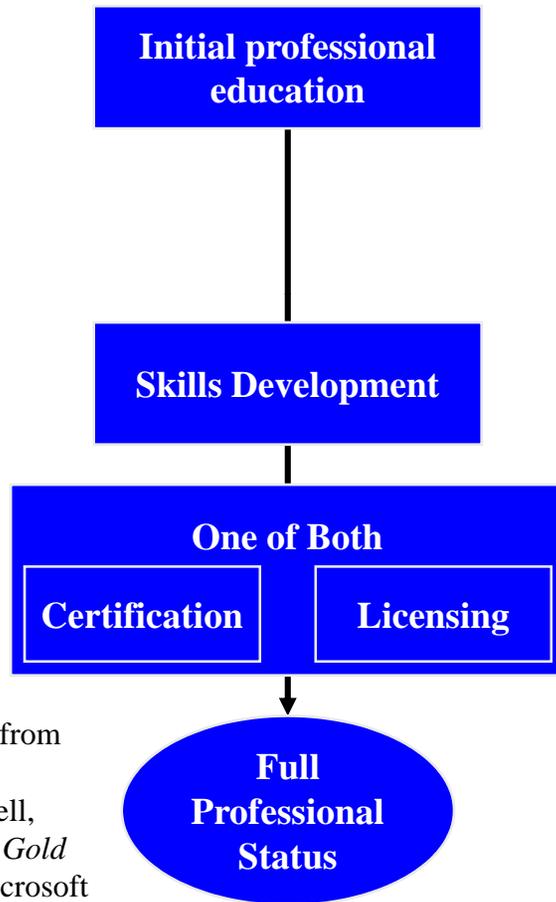
A Model of an Engineering Profession



Ten years ago, most of this did not exist for software engineering.
It all exists now.

*Adapted from
McConnell after
Ford and Gibbs.*

IEEE CS Support of the SWE Professional



Adapted from
Steve
McConnell,
*After the Gold
Rush*, Microsoft
Press

Examples

Body of knowledge

SWEBOK Guide

Professional Education

SE 2004 Curriculum

ABET Accreditation

Graduate Curriculum

Professional Development

CSDA, CSDP

Training

Licensing

Professional Practice

Standards

Code of Ethics

Body of Knowledge

2004 SWEBOK Guide SWEBOK Guide Refresh

Software Engineering Body of Knowledge Project

- Promote a **consistent view of software engineering** worldwide
- Clarify the place of, and **set the boundary of**, software engineering with respect to other disciplines
- **Characterize the contents** of the Software Engineering Body of Knowledge - SWEBOK
- **Provide a topical access** to the SWEBOK
- **Provide a foundation** for curriculum development and individual certification and licensing material

The Body of Knowledge for software engineering already exists in the literature. The mission of the SWEBOK project is to provide an authoritative guide to the portion that is “generally accepted.”

Types of Knowledge

Specialized	Generally Accepted
	Advanced and Research

Focus of the SWEBOK Guide

Generally accepted: “Applies to most projects most of the time and widespread consensus validates its value and effectiveness.” -- PMI

In terms of US education, we target the SWEBOK at bachelor’s degree plus four years of experience.

Ten Knowledge Areas

- **Software Requirements**
- **Software Design**
- **Software Construction**
- **Software Testing**
- **Software Maintenance**
- **Software Configuration Management**
- **Software Eng. Management**
- **Software Eng. Tools & Methods**
- **Software Engineering Process**
- **Software Quality**

Related Disciplines

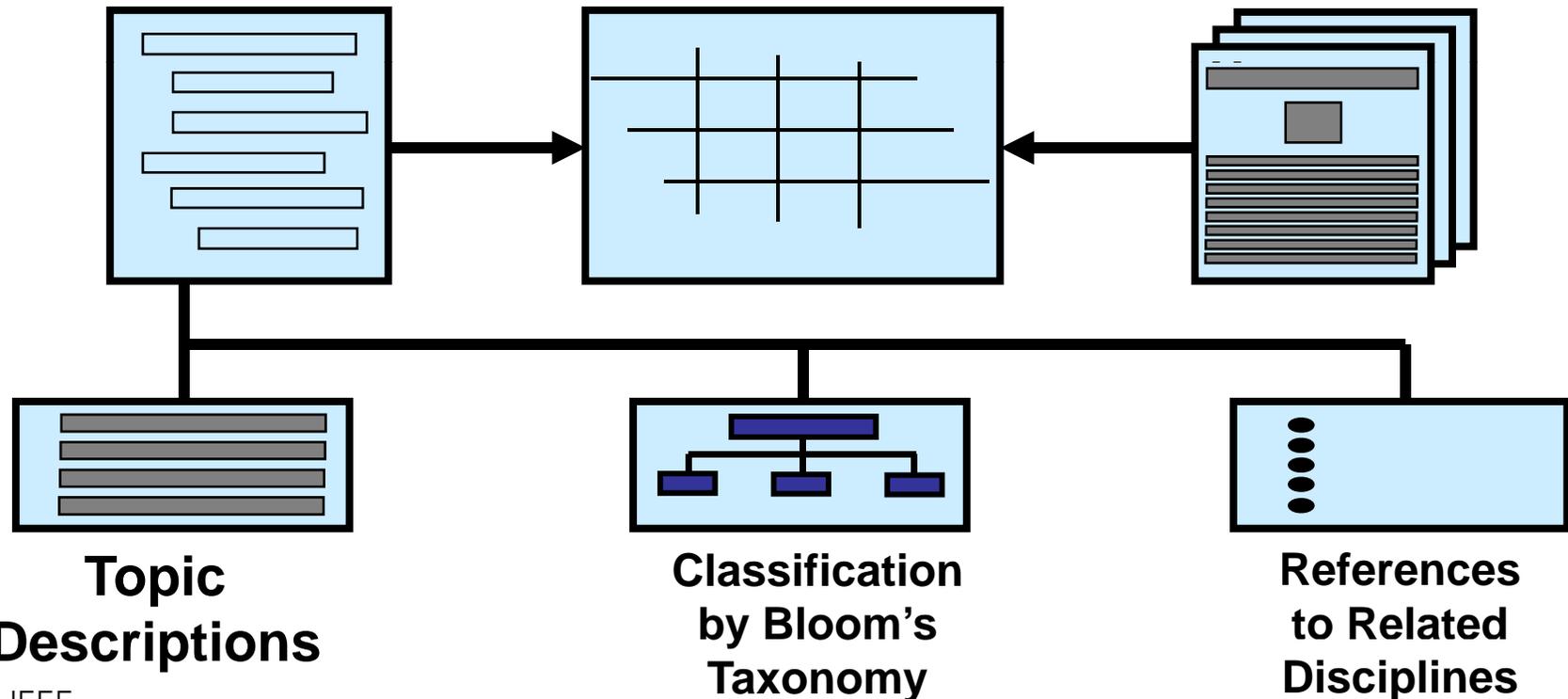
- Computer Science
- Mathematics
- Project Management
- Computer Engineering
- Cognitive Sciences
- Human Factors
- Systems Engineering
- Management
- Management Science

Knowledge Area Description

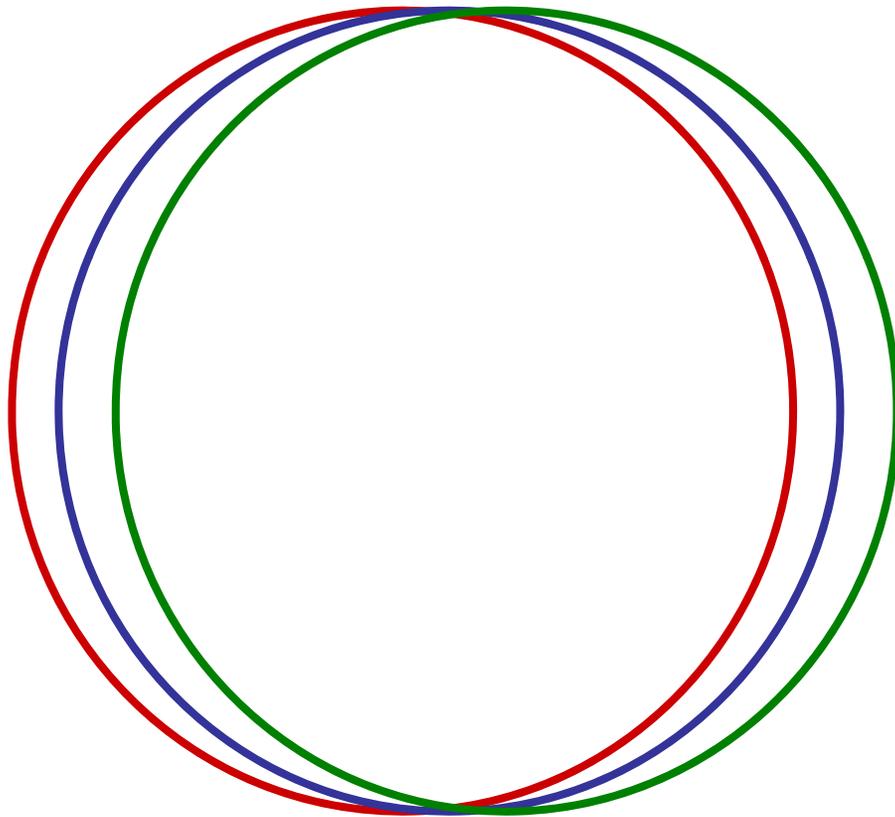
**Classification
of Topics**

**Matrix of Topics
& References**

References



There is substantial agreement on the scope of Software Engineering



SWEBOK, CSDP, and SE2004 each characterized the scope of software engineering.

Each followed its own consensus process.

The extent of agreement among the three is remarkable.

It is a powerful validation of the characterization.

Nevertheless, the minor differences should be resolved.

Plan for Refresh of SWEBOK Guide

- Broaden the contents to span the consensus achieved by:
 - 2004 SWEBOK Guide
 - SE 2004 curriculum
 - Original CSDP test specification
- Move toward a continuous update process
- Provide provisional “supplementary” KAs in emerging and specialty areas for comment and trial usage
 - Security
 - Measurement
- Lead editors
 - Alain Abran, École de technologie supérieure, Canada
 - Pierre Bourque, École de technologie supérieure, Canada
 - Juan Garbajosa, Universidad Politécnica de Madrid, Spain
 - Gargi Keeni, Tata Consultancy Services, India
 - Beijun Shen, Shanghai Jiaotong University, China
- Publication planned for 2010
- Will underlay professional development products of the Society

Planned Content of 2010 SWEBOK Guide

- Characterizing the Practice of Software Engineering
 - SW Requirements
 - SW Design
 - *Human-Computer Interface Design*
 - SW Construction
 - SW Testing
 - *Human-Computer Interface Testing*
 - SW Maintenance
 - SW CM
 - SW Eng Management
 - SW Eng Process
 - SW Eng Methods [changed name]
 - *Cross-KA methodologies and their selection*
 - *[Distribute tools into other KAs]*
 - SW Quality
 - *SW Eng Professional Practice*
- *Required in Educating a Software Engineer*
 - *Computer Science Foundations*
 - *Mathematical Foundations*
 - *Engineering Foundations*
 - *Economic Foundations*
- Related Disciplines
 - Computer Engineering
 - ~~Computer Science~~
 - ~~Mathematics~~
 - Management
 - Project Management
 - Quality Management
 - Software Ergonomics
 - System Engineering
- *Supplementary (“trial-use”)*
 - *Measurement*
 - *Security*

Take Away Items

- There is broad agreement on the body of knowledge for Software Engineering.
- Three different consensus mechanisms have reached nearly identical conclusions.
- The 2010 revision of the SWEBOK Guide will be broadened to deal with the small differences among the three.

Professional Education

Curriculum
Accreditation
Graduate Curriculum

IEEE-CS/ACM Computing Curricula Series

- Computer Science Volume – revised in 2008

- Software Engineering Volume – 2004

- Computer Engineering Volume – 2004
- Information Systems Volume – revised in 2009
- Information Technology Volume – 2008

SE 2004 Components

- Software Engineering Education Knowledge – What every SE graduate should know:
 - Computing Essentials (172 contact hours)
 - Mathematical and Engineering Fundamentals (89)
 - Professional Practice (35)
 - Software Modeling and Analysis (53)
 - Software Design (45)
 - Software Verification and Validation (42)
 - Software Evolution (10)
 - Software Process (13)
 - Software Quality (16)
 - Software Management (19)
- Curriculum – Suggestions for how it should be taught

Specialized Accreditation

- Distinct from “institutional accreditation”
- Focuses on particular aspects of an academic field of study, including engineering, nursing, law, or education, among others
- Some professions that are regulated by and dependent upon a state or national licensing board may require job applicants to have graduated from specific accredited academic programs that have specialized, professional, or programmatic accreditation status.

ABET does Computing Accreditation

- Computing Accreditation Commission
 - Computer Science
 - Software Engineering*
 - Information Science
 - Information Technology
- Engineering Accreditation Commission
 - Computer Engineering*
 - Electrical Engineering
 - Mechanical Engineering
 - Chemical Engineering
 - Civil Engineering ...

* Computer Engineering and Software Engineering are Jointly Accredited.

ABET Accreditation

- Software Engineering and Computer Engineering are evaluated by ABET like any other engineering program
 - EAC General Criteria
 - Software Engineering Program Criteria
 - Curriculum must provide breadth and depth of engineering and computer science topics
 - Graduates must have the ability to:
 - analyze, design, verify, validate, implement, apply, and maintain software systems
 - the ability to appropriately apply discrete mathematics, probability and statistics, and relevant topics in computer science and supporting disciplines to complex software systems
 - and the ability to work in one or more significant application domains.
- There are currently 16 accredited SwE programs

Graduate Software Engineering Reference Curriculum (GSwERC)

- Led by Art Pyster, Stevens Institute
- Sponsored by DoD
- Participation by professional societies
- 3 Reference Curricula at Masters level
 - SW Engineering (with an appropriate amount of SE)
 - Systems Engineering (with an appropriate amount of SwE)
 - An interdisciplinary degree

Outcomes for GSwERC Grads

- Mastery of core body of knowledge
- Mastery of one application domain
- Deep mastery of one knowledge area
- Knowledge of ethics
- SE/SwE relationship
- Teamwork and leadership
- Trade-offs
- “Soft skills”
- Continued professional development
- Mastery of one technology

Take-Away Items

- There is a model curriculum (SE 2004) for an undergraduate degree in Software Engineering
 - Distinct from Computer Science, Computer Engineering, Information Systems, and Information Technology
- There will soon be a model curriculum (GSwERC) for a master's degree in Software Engineering
- ABET accredits Software Engineering programs.
- As of October 2005, in the US, there are...
 - 32 bachelor's level programs (15 accredited)
 - 53 master's level programs
 - 4 PhD programs
 - Another 15 accredited bachelor's programs in Canada.

Professional Development

Certification
Training
Licensing

What is Certification?

- **Formal recognition** of demonstrated proficiency within and comprehension of a specified body of knowledge at a point in time.
- It is **peer recognition** and not registration or licensure.
 - **Registration:** listing by and with a body of individuals or organizations that are certified
 - **Licensure:** authorization granted by government body for an individual or organization to practice a business or occupation
- Certification is **voluntary**.

Three Types of Certification

- **Training-related**

- Completion of a set of courses
- (This is sometimes called a “certificate” rather than a “certification”.)

- **Product-related**

- Knowledge of a particular product or product line
- e.g. MCSE, CNE

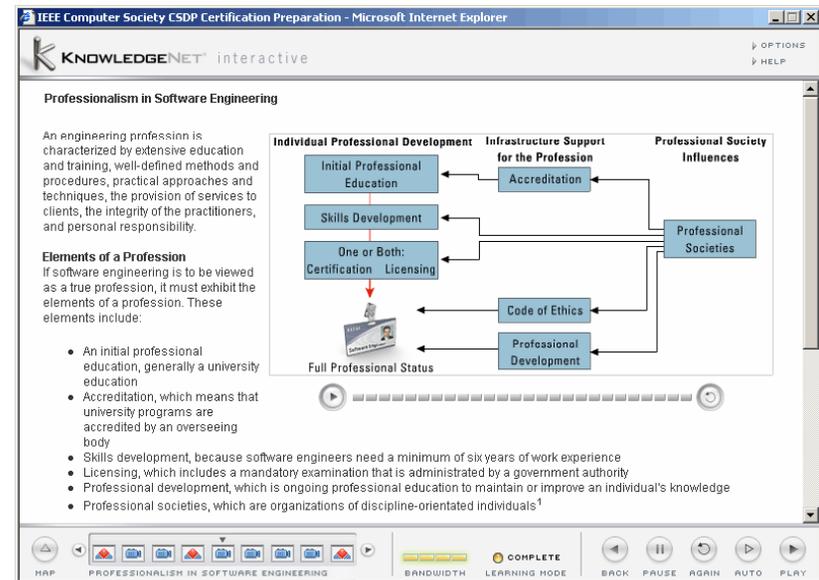
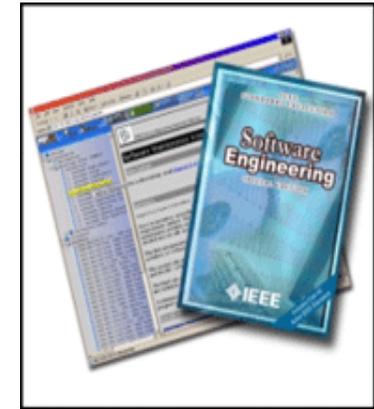
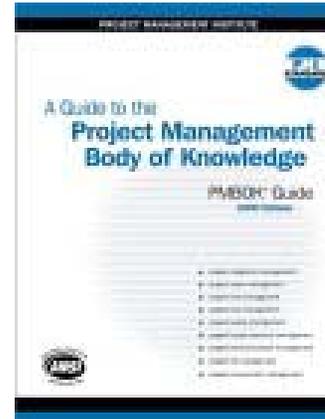
- **Professional**

- Mastery of a long-lived set of principles and ethics fundamental to practice in a professional field
- e.g. PMP, CSDP, CSQE, PHR

Basis for Professional Certifications

- Professional certifications are usually founded on professional society norms such as:
 - A Code of Ethics: *IEEE-CS/ACM Software Engineering Code of Ethics*
 - A specified Body of Knowledge: *IEEE-CS Guide to the Software Engineering Body of Knowledge**
 - A set of professional practice standards: *IEEE and international standards on software engineering*
- ... and supported by appropriate training programs and materials.

* <http://www.swebok.org>

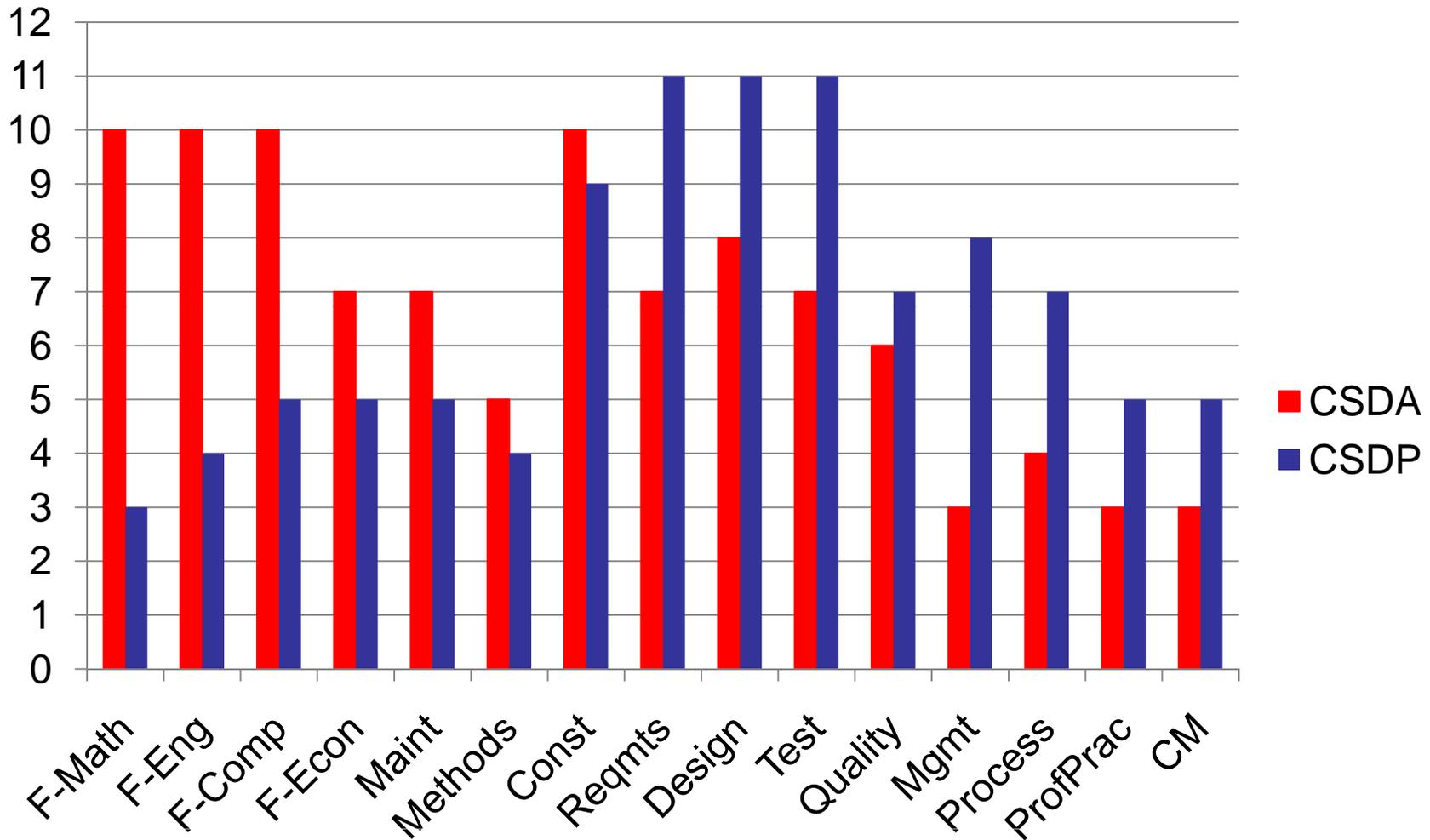


Computer Society Certifications

- **CSDA:** Designed to provide entry-level SW engineer with a baseline knowledge of fundamental development practices and a growth path to the CSDP and beyond
- **CSDP:** Designed for mid-career SW engineers looking to advance in their field and confirm their knowledge of development practices

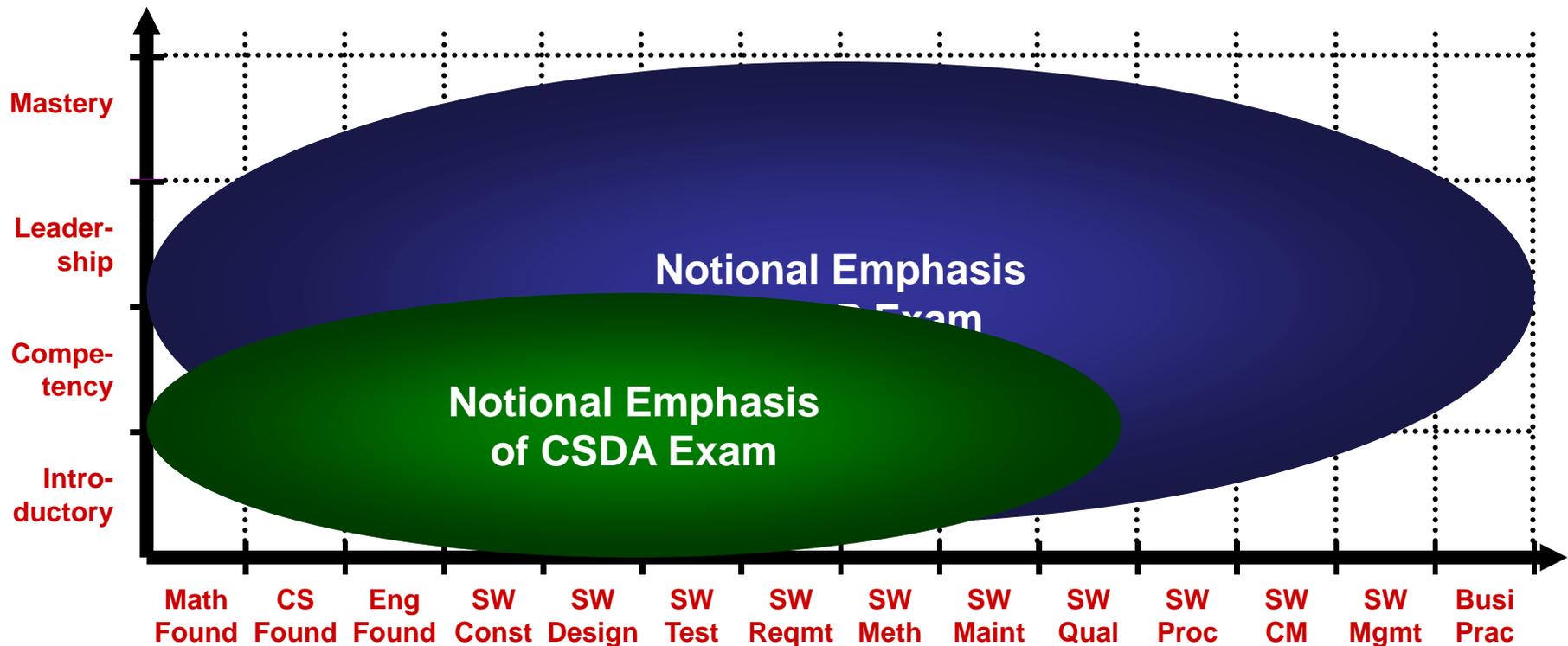


% of Exam Questions in each KA



Depth of Knowledge in Exams

Skill Levels



Knowledge Areas shared by SWEBOK and Certifications
 (roughly ordered with suitability for university teaching toward the left and
 suitability for industrial experience toward the right)

Certification Requirements

- CSDA requirements - none
- CSDP requires both
 - Education qualifications
 - Experience qualifications
- Either requires passing appropriate exam
 - Computer-based
 - About 4 hours long
 - 180 questions
 - Multiple-choice questions selected from a pool of questions
 - Closed book, calculators provided

Exam Preparation Options

- CSDA e-Learning Course
 - 4 on-line modules covering 15 knowledge areas
 - 6 assessments: pre-test, post-test, 4 modules
 - CD or printed
- Sample exam questions
- SWEBOK Guide
- <http://www.computer.org/csda/prep>
- CSDP e-Learning Course
 - Covers 11 CSDP knowledge areas
 - CD or printed
- SWEBOK Guide
- Resource guide
- Book of sample questions
- <http://www.computer.org/csdp/prep>

A Path to Licensing

- A consortium of groups wants to provide a path to licensure for software engineers
 - IEEE subgroups, NSPE subgroups, Texas Board, NCEES
- Endorsements from ten state licensing agencies are required. So far:
 - Eight have endorsed
 - Additional one is rewriting their endorsement
 - Five others are considering

Licensing Software Engineers

- The process for licensing would be similar to any engineering profession:
 - Graduate from ABET-accredited curriculum
 - Pass the Fundamentals of Engineering exam
 - Four years of acceptable, supervised practice
 - Pass the Principles and Practices exam
- The consortium is proposing development of the P&P exam.

Would Everyone Need a License?

- Would all software engineers need to be licensed?
 - No, only those providing their services directly to the public.
- Would all software have to be developed by licensed software engineers?
 - No, only software that has an impact on the lives, property, economy, or security of people or the national defense.

Take-Away Items

- Licensing for software engineers is underway in Texas, British Columbia, Ontario, Canada and other countries.
- It is likely that many states in the US will offer a path to licensing by 2011.
- Currently, IEEE Computer Society offers professional certifications for entry-level and mid-level software professionals.

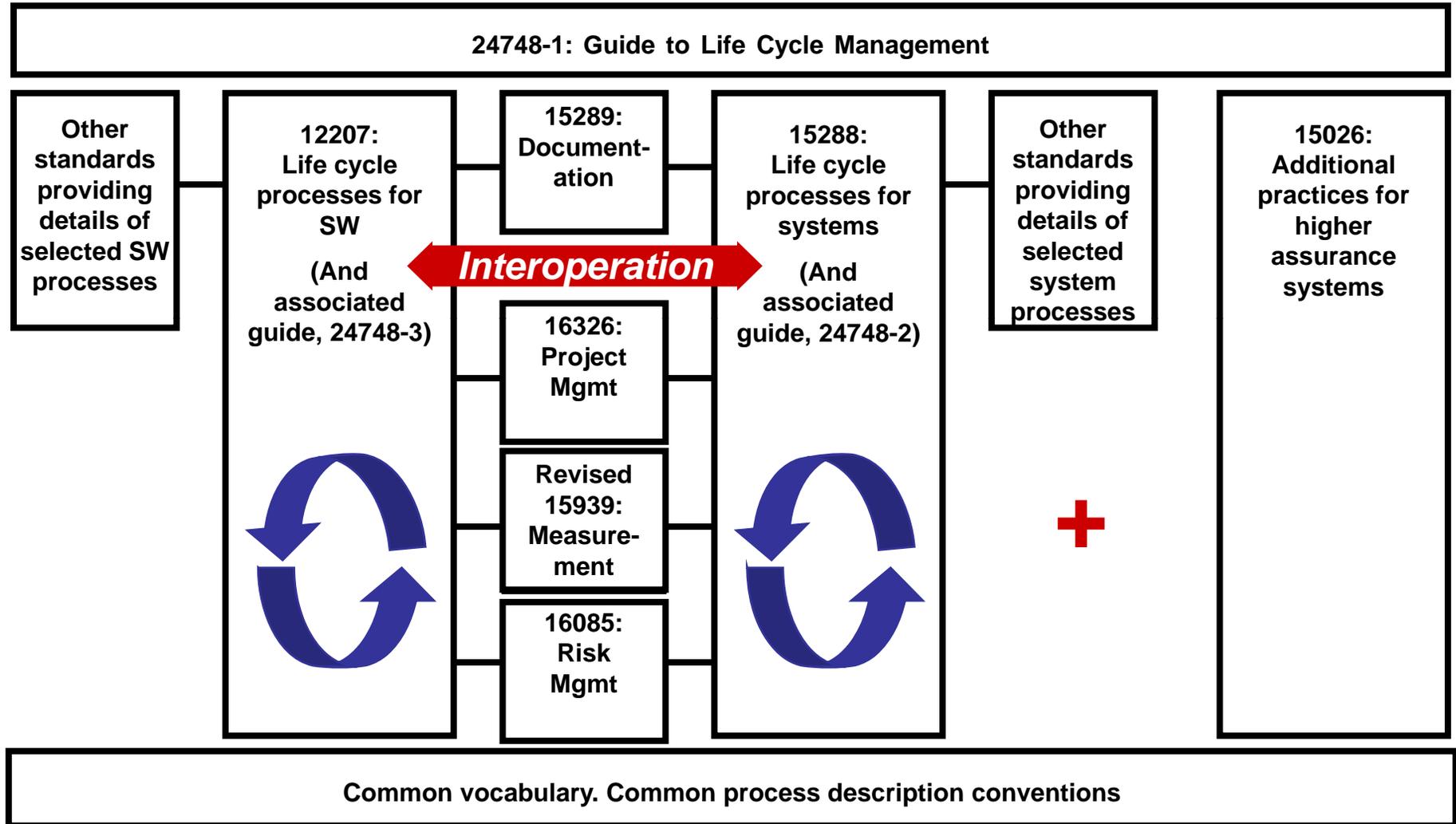
Professional Practice

Standards
Ethics

Harmonization of IEEE and ISO/IEC

- ISO/IEC JTC 1/SC 7 (software and systems engineering) has a large collection of standards.
 - Some of the key standards are difficult to use together.
- IEEE Software and Systems Engineering Standards Committee has a large collection of standards.
 - Some of the key standards are not completely consistent with the ISO/IEC standards
- IEEE Computer Society and ISO/IEC JTC 1/SC 7 have entered into a program to “harmonize” their key standards to provide a shared, common framework, e.g.
 - A single shared set of processes.
 - A single shared vocabulary.

Intended Relationships of Key System and Software Engineering Life Cycle Process Standards



New Vocabulary Standard

- Systems and software engineering vocabulary
- Publicly available website
<http://www.computer.org/sevocab/>
- Provides access to 4100 authoritative definitions of systems and software engineering terms
- Definitions may be reprinted (with attribution).
- Currency of database will be maintained by a vocabulary standards project coordinated between IEEE and ISO/IEC JTC 1/SC 7, and published as ISO/IEC/IEEE 24765.
- Sources:
 - IEEE standards
 - ISO/IEC standards
 - PMBOK® Guide, Third edition
 - Other sources

The State of Harmonization

Topic	Status	Remarks
Terminology & Concepts	Yellow ↑	Shared BOK, joint vocabulary project, potential certification framework
Quality management	Yellow	IEEE is adopting ISO/IEC 90003 approach.
Testing	Orange ↑	Both IEEE and BSI will harmonize with SC7 processes
Architecture description	Green	SC7 adopted IEEE standard and will harmonize with processes.
Product quality	Yellow ↑	ISO/IEC 12119 was revised as 25051. IEEE will withdraw its standard.
Life cycle processes	Green	
Systems engineering	Green	Shared SE process standard; harmonization with other LC processes underway
SW maintenance	Green	Project to merge IEEE and ISO standards is completed
Measurement	Green	Some details remain.
Risk management	Green	SC7 adopted IEEE standard and is now extending it to the systems level.
Project management	Yellow ↑	Project is merging the incompatible standards.
Verification and validation	Red	Fundamentally different approaches. Good intentions, but no action yet.
Configuration management	Yellow	SC7 withdrew its standard; systems issues remain. IEEE is about to revise.
SW process assessment	Yellow ↑	Harmonization with LC process standards is underway
Requirements engineering	Orange ↑	Joint project has been approved; mashup of relevant standards is being prepared.
SW life cycle data	Yellow ↑	IEEE is adopting 15289 to replace 12207.1
User documentation	Yellow ↑	IEEE 1063 has been incorporated into 26514. IEEE will adopt it.
CASE tools	Yellow	Minor incompatibilities
Notations	Harmless	Distinct standards for distinct notations
Internet	Green	Shared standard
IT Services, Management, Governance	Yellow	IEEE will adopt 20000 standards
Specialty Engineering (Safety, Security)	Orange ↑	Unrelated approaches will be addressed in part by coordination revision of 15026
Others	Yellow	Many unrelated standards

Code of Ethics

Software Engineering Code of Ethics and Professional Practices

- Developed by a Joint IEEE-CS/ACM Task Force on Software Engineering Ethics and Professional Practices. Approved in 1998.

“The short version of the code summarizes aspirations at a high level of the abstraction; the clauses that are included in the full version give examples and details of how these aspirations change the way we act as software engineering professionals. Without the aspirations, the details can become legalistic and tedious; without the details, the aspirations can become high sounding but empty; together, the aspirations and the details form a cohesive code.”

Short Version States Eight Principles

- **Public:** Software engineers shall act consistently with the public interest.
- **Client and employer:** Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.
- **Product:** Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.
- **Judgment:** Software engineers shall maintain integrity and independence in their professional judgment.
- **Management:** Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.
- **Profession:** Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.
- **Colleagues:** Software engineers shall be fair and supportive of their colleagues.
- **Self:** Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.

Example from Long Version

Principle 4 JUDGMENT Software engineers shall maintain integrity and independence in their professional judgment. In particular, software engineers shall, as appropriate:

- 4.01. Temper all technical judgments by the need to support and maintain human values.
- 4.02. Only endorse documents either prepared under their supervision or within their areas of competence and with which they are in agreement.
- 4.03. Maintain professional objectivity with respect to any software or related documents they are asked to evaluate.
- 4.04. Not engage in deceptive financial practices such as bribery, double billing, or other improper financial practices.
- 4.05. Disclose to all concerned parties those conflicts of interest that cannot reasonably be avoided or escaped.
- 4.06. Refuse to participate, as members or advisors, in a private, governmental or professional body concerned with software related issues, in which they, their employers or their clients have undisclosed potential conflicts of interest.

Backup

Consensus Process

- Four lead editors
- 21 KA editors (for 10 KAs)
- 10 organizations on Industrial Advisory Board
- Four rounds of review and comment disposition, using a variety of populations
- A total of about 10,000 comments
- Nearly 600 individual reviewers
 - About half were non-US
 - Roughly equal split: BS, MS, PhD
 - Roughly equal split: 0-50 employees, 50-500, 500+

SE 2004 Process

- Objective: Provide guidance to academic institutions and accreditation agencies about what should constitute an undergraduate software engineering education
- Consensus Process:
 - 65 volunteers: 29 US and 36 non-US
 - International representation on the Steering Committee
 - Internal review by international SE experts
 - Review by industrial advisory panel
 - Multiple review cycles with steering committee responses to all issues

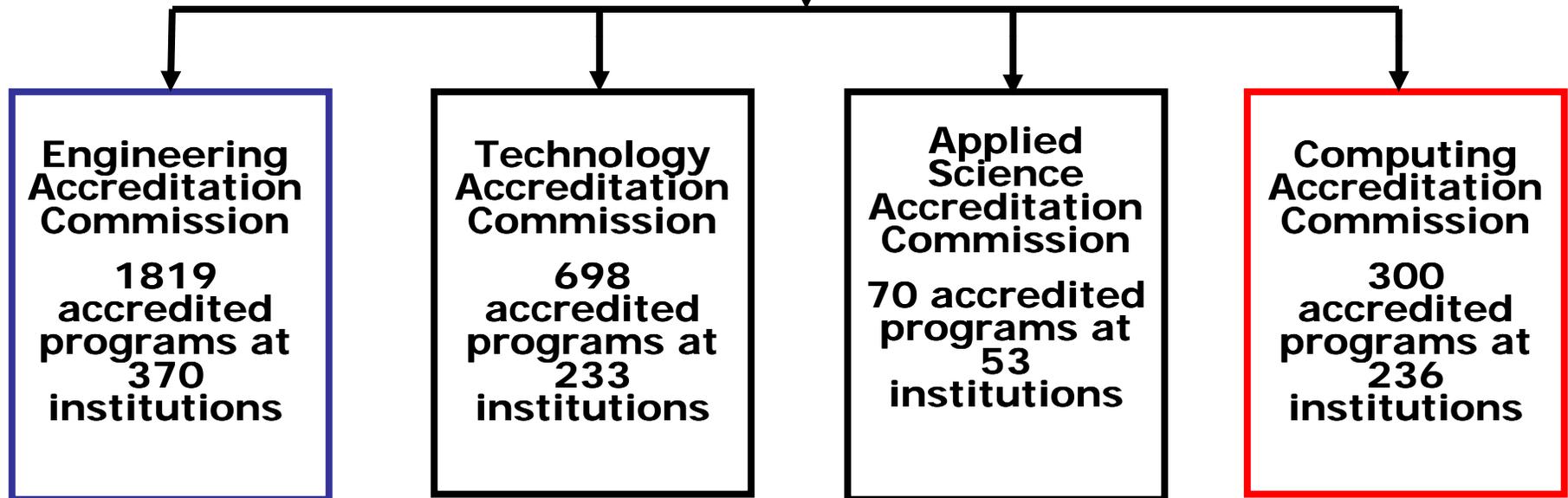
Examples of SWEBOK Uptake

- Available for free on the web (<http://www.swebok.org>) and in book form
- Translated (or translating) into Japanese, Chinese, Spanish, French, Russian, Hungarian, Arabic
- CSDA, CSDP, SE2004, GSwERC, CCPE and IFIP IP3 cite SWEBOK Guide as a source
- SWEBOK Guide was adopted as ISO/IEC TR 19759
- A large defense contractor has experimented with the Guide to calibrate skills descriptions in proposals.
- An FFRDC has rewritten its “Software Systems Engineer” job description in terms of the SWEBOK knowledge areas.
- Construx, Inc. has rewritten its position descriptions in terms of the Guide, structures its professional development around the Guide.
- NTU and SMU have rationalized their software engineering offerings using the SWEBOK.
- SWEBOK provides the taxonomical basis for VISEK, a SWE Portal funded by the German government.
- 155,000 Google hits on “SWEBOK” in May 2007

Professional Societies in Accreditation

- Professional Societies work with accrediting agencies to assure that the technical criteria are appropriate
- For example ABET works with the ACM, IEEE, and other engineering and technical societies to assure that engineering and technical accreditation are based on sound criteria, and on what industry needs.

Accreditation Board for Engineering and Technology

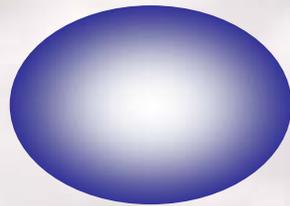


Numbers do not include 2008 cycle

A standard is a name for an otherwise fuzzy concept

In a complex, multidimensional trade space of solutions ...

... a standard gives a name to a bounded region.



It defines some characteristics that a buyer can count on.