The Emerging Consensus on the Software Engineering Body of Knowledge

P. Bourque, École de technologie supérieure
R. Dupuis, Université du Québec à Montréal
Alain Abran, ÉTS

CASCON 2001
Toronto
November 6, 2001

www.swebok.org
Presentation Plan

- **Project background**
  - Project scope, objectives, audience and plan
  - Contents of the Guide
  - How you can leverage the Guide
  - Discussion
  - Conclusions
What is Software Engineering?

- IEEE 610.12:
  - “(1) The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software.
  - (2) The study of approaches as in (1).”
Recognized Profession?

- Starr*: Knowledge and competence validated by the community of peers
- Consensually validated knowledge rests on rational, scientific grounds
- Judgment and advice oriented toward a set of substantive values

Professional Development

- Initial professional education
- Skills Development
- One or both: Certification, Licensing
- Accreditation
- Professional societies
- Professional development
- Code of ethics
- Full Professional Status

Adapted from Steve McConnell, *After the Gold Rush*, Microsoft Press, 1999, p. 93
Presentation Plan

- Project background
- **Project scope, objectives, audience and plan**
  - Contents of the Guide
  - How you can leverage the Guide
  - Discussion
  - Conclusion
Project Objectives

- Characterize the contents of the Software Engineering Body of Knowledge
- Provide a topical access to the Software Engineering Body of Knowledge
- Promote a consistent view of software engineering worldwide
Project Objectives

- Clarify the place of, and set the boundary of, software engineering with respect to other disciplines (computer science, project management, computer engineering, mathematics, etc.)

- Provide a foundation for curriculum development and individual certification and licensing material
Intended Audience

- Public and private organizations
- Practicing software engineers
- Makers of public policy
- Professional societies
- Software engineering students
- Educators and and trainers
What Are we Not Trying to Accomplish?

- Not a curriculum development effort!
- Not an all-inclusive description of the sum of knowledge in the field
- Not all categories of knowledge
Categories of Knowledge in the SWEBOK

<table>
<thead>
<tr>
<th>Specialized</th>
<th>Generally Accepted</th>
<th>Advanced and Research</th>
</tr>
</thead>
</table>

Focus of the SWEBOK Guide
Knowledge of a Software Engineer

- Application domain knowledge
- Advanced SE Knowledge
- Specialized SE Knowledge
- Guide to the SWEBOK Stoneman
- Maths
- C.S.
- ...

Maths

Application domain knowledge

Advanced SE Knowledge

Guide to the SWEBOK Stoneman

C.S.

Specialized SE Knowledge

Maths

...
Three Underlying Principles of the Project

- **Transparency**: the development process is itself published and fully documented

- **Consensus-building**: the development process is designed to build, over time, consensus in industry, among professional societies and standards-setting bodies and in academia

- Available **free** on the web
Project Team

- Editorial team
- Industrial Advisory Board
- Knowledge Area Specialists
- Reviewers
Editorial Team

- **Project “Champion”:**
  - Leonard Tripp, 1999 President, IEEE Computer Society
  - President, Professional Practices Committee
- **Executive Editors:**
  - Alain Abran, ETS
  - James W. Moore, The MITRE Corp.
- **Editors:**
  - Pierre Bourque, ETS
  - Robert Dupuis, UQAM
Roles of the Industrial Advisory Board

- Provide input to ensure relevance to various audiences
- Review and approve strategy and deliverables
- Oversee development process
- Assist in promoting the Guide to the Software Engineering Body of Knowledge
- Lend credibility to the project
A Three-Phase Approach for Developing the Guide to the SWEBOK

Straw Man Version

Stone Man Phase (Trial Version)

Iron Man Version (Sub-phase 1)
A Three-Phase Approach for Developing the Guide to the SWEBOK
Knowledge Area Specialists

- Bertolino, Istituto Elaborazione Informazione, CNR, Italy
- Bollinger, MITRE, USA, Martin & Gabrini, UQAM
- Carrington, Queensland University, Australia
- El Emam, National Research Council, Canada
- MacDonell, University of Otago, New-Zealand
- Sawyer & Kotonya, Lancaster University, UK
- Scott, Lawrence Livermore National Lab., USA
- Tremblay, UQAM, Canada
- Pigoski, USA
- Wallace & Reeker, NIST, USA
Phase 2: Stone Man Review Process

Version 0.1
Limited number of domain experts

Review Cycle 1

Version 0.5
Selected users

Review cycle 2

Version 0.7
Community

Review Cycle 3

Version 0.9
Stone Man Review Process

- Transparency and consensus-building
  - All intermediate versions of documents are published and archived on www.swebok.org
  - All comments are made public as well as the identity of the reviewers
  - Detailed comment disposition reports are produced for Review Cycle 2 and 3
Data on reviewers

- Version 0.1: 33
- Version 0.5: 195
- Version 0.7: 378
  - + ISO reviews from 5 countries
Geographic Distribution of Reviewers

- USA: 55%
- Europe: 18%
  - 90 reviewers from 25 countries
- Canada: 10%
- Australia: 5%
- Asia: 5%
- Latin America: 4%
Project Overview
Presentation Plan

- Project background
- Project scope, objectives, audience and plan

Contents of the Guide
- How you can leverage the Guide
- Discussion
- Conclusion
Stone Man Deliverables:

- **Consensus** on a list of Knowledge Areas
- **Consensus** on a list of **topics and relevant reference materials** for each Knowledge Area
- **Consensus** on a list of Related Disciplines
List of 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 (CC2001)
- Mathematics (CC2001)
- Project Management (PMBOK)
- Computer Engineering
- Cognitive Sciences and Human Factors
- Systems Engineering
- Management and Management Science
Knowledge Area Description

Classification of Topics

Matrix of Topics & References

References

Topic Descriptions

Classification by Vincenti’s Taxonomy

Classification by Bloom’s Taxonomy

References to Related Disciplines

Not implemented in Stoneman
Software Construction

Reduction in Complexity
- Linguistic Construction Methods
  - Formal Construction Methods
  - Visual Construction Methods

Anticipation of Diversity
- Linguistic Construction Methods
  - Formal Construction Methods
  - Visual Construction Methods

Structuring for Validation
- Linguistic Construction Methods
  - Formal Construction Methods
  - Visual Construction Methods

Use of External Standards
- Linguistic Construction Methods
  - Formal Construction Methods
  - Visual Construction Methods

© IEEE

www.swebok.org
Software Testing

A. Testing Basic Concepts and Definitions
   A1. Testing-Related Terminology
   A2. Theoretical Foundations
   A3. Relationships of Testing to Other Activities

B. Test Levels
   B1. The Target of the Test
   B2. Objectives of Testing

C. Test Techniques
   C1.1 Based on Tester's intuition and experience
   C1.2 Specification-based
   C1.3 Code-Based
   C1.4 Fault-Based
   C1.5 Usage-Based
   C1.6 Based on Nature of Application
   C2.1 Black-Box Techniques
   C2.1 White-Box Techniques
   C3. Selecting and Combining Techniques

D. Test Related Measures
   D1. Evaluation of the Program Under Test
   D2. Evaluation of the Tests Performed

E. Managing the Test Process
   E1. Management Concerns
   E2. Test Activities
Software Maintenance

- Basic Concepts
  - Definitions and Terminology
  - Majority of Maintenance Costs
  - The Nature of Maintenance
  - Evolution of Software
  - Need for Maintenance
  - Categories of Maintenance

- Maintenance Process
  - Process Models
  - Maintenance Activities

- Key Issues in Software Maintenance
  - Technical
  - Management
  - Cost and Estimation
  - Measures

- Techniques for Maintenance
  - Program comprehension
  - Re-engineering
  - Reverse Engineering
  - Impact Analysis
Software Configuration Management

- Organizational Context for SCM
- Constraints and Guidance for SCM
- Planning for SCM
- Software Configuration Management Plan

- Software Configuration Identification
  - Identifying Items to be Controlled
  - Software Configuration
  - Software Configuration Items
  - Software Configuration Item Relationships
  - Software Versions
  - Baseline
  - Acquiring Software Configuration Items
  - Software Library

- Software Configuration Control
  - Requesting, Evaluating and Approving Software Changes
  - Implementing Software Changes
  - Deviations and Waivers

- Software Configuration Status Accounting
  - Software Configuration Status Information
  - Software Configuration Status Reporting

- Software Configuration Auditing
  - Software Functional Configuration Audit
  - Software Physical Configuration Audit
  - In-Process Audits of a Software Baseline

- Software Release Management and Delivery
  - Software Building
  - Software Release Management

© IEEE  www.swebok.org
Software Engineering Management

Organizational Management

- Policy Management
- Personnel Management
- Communication Management
- Portfolio Management
- Procurement Management

Process/Project Management

- Initiation and scope definition
- Planning
- Enactment
- Review and Evaluation
- Closure

Software Engineering Measurement

- Goals
- Measurement Selection
- Measuring Software and its Development
- Collection of data
- Software Measurement Models
Software Engineering Tools and Methods

I. Software Tools
- Software Requirements Tools
- Software Design Tools
- Software Construction Tools
- Software Testing Tools
- Software Maintenance Tools
- Software Engineering Process Tools
- Software Quality Tools
  - Software Configuration Management Tools
- Software Engineering Management Tools
- Infrastructure Support Tools
- Miscellaneous Tools Issues

II. Software Methods
- Heuristic Methods
  - Structured methods
  - Data-oriented methods
  - Object-oriented methods
  - Domain specific methods
- Formal Methods
  - Specification languages
  - Refinement
  - Verification
- Prototyping Methods
  - Styles
  - Prototyping target
  - Evaluation techniques
- Miscellaneous Method Issues
  - Method evaluation
Software Quality

Software Quality Concepts

- Measuring the Value of Quality
- ISO 9126 Quality Description
- Dependability
- Special Types of Systems and Quality Needs

Purpose and Planning of SQA and V&V

- Common Planning Activities
- The SQA Plan
- The V&V Plan

Activities and Techniques for SQA and V&V

- Static Techniques
- Dynamic Techniques
- Other SQA and V&V Testing

Measurement Applied to SQA and V&V

- Fundamentals of Measurement
- Measures
- Measurement Analysis Techniques
- Defect Characterization
- Additional Uses of SQA and V&V data
Project Overview
Presentation Plan

- Project background
- Project scope, objectives, audience and plan
- Contents of the Guide

How you can leverage the Guide
- Discussion
- Conclusion
A Three-Phase Approach for Developing the Guide to the SWEBOK

- Straw Man Version
- Stone Man Version
- Iron Man Version (Sub-phase 1)
- Iron Man Version (Sub-phase 2)

Applications of the Guide

- Industry and Government
- Education
- Research
- Licensing and Certification?
Applications of the Guide

- Industry & Government
  - HR: job description, hiring, staffing of projects, career planning, contracting:
    - Lockheed-Martin
    - Large Brazilian bio-medical software company
  - Process models, policy: Construx, Brazilian company
  - Makers of public policy: Turkish Society for Quality
Applications of the Guide

- Professional development
  - internal training, corporate universities
  - self-assessment
  - individual training
  - Examples: Construx, Financial Software Company
Applications of the Guide

- Education:
  - Course design/evaluation: Arizona St.
  - Curriculum design/evaluation: NTU, U. of Iceland, SMU, Stevens Institute of Technology (NJ), Musahi U. Japan, etc.
  - Program accreditation: Japan is evaluating...
# Categories of Knowledge in the SWEBOK

<table>
<thead>
<tr>
<th>Specialized</th>
<th>Generally Accepted</th>
<th>Focus of the SWEBOK Guide</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Advanced and Research</td>
<td></td>
</tr>
</tbody>
</table>
Research: Advanced and Research Topics

- What topics should be monitored as the most likely to become *generally accepted* in the near future?
- What mechanisms should be used to monitor these and other topics?
Research: Specialized Domains

- What are the most important domains for which the knowledge should be included in extended versions of the Guide?
- What characteristics make each of these domains different from the core of Software Engineering?
- Do we need additional criteria for recognizing the generally accepted knowledge in each of these domains?
Project Overview

Presentation Plan

- Project background
- Project scope, objectives, audience and plan
- Contents of the Guide
- How you can leverage the Guide
- Discussion

Conclusion
A Three-Phase Approach for Developing the Guide to the SWEBOK

- Straw Man Version
- Stone Man Version
- Iron Man Version (Sub-phase 1)
- Iron Man Version (Sub-phase 2)
Concluding Remarks

- Consensus on the core body of knowledge is key in all disciplines
- Participation of all communities is important
www.swebok.org