Development of An International Consensus on the Software Engineering Body of Knowledge

Alain Abran
P. Bourque, R. Dupuis, J. W. Moore, L. Tripp

Universidad de Alcalá,
Alcalá de Henares (Spain)
June 2005
Guide to the Software Engineering Body of Knowledge (SWEBOK®)

- Began as a collaboration among IEEE CS, ACM and the Université du Québec à Montréal
- International participation from industry, professional societies, standards bodies, academia, authors
- Over 500 hundred software engineering professionals have touched the document
- Release of Trial Version in 2001

® Registered in U.S. Patent Office
Trial Version (2001)
2004 SWEBOK Guide

- Available on www.swebok.org
- The 2004 Version was endorsed by the project’s Industrial Advisory Board in January 2004 and approved by the IEEE Computer Society Board of Governors in February 2004
- To be published as ISO Technical Report 19759
List of Knowledge Areas

- Software Requirements
- Software Design
- Software Construction
- Software Testing
- Software Maintenance
- Software Configuration Management
- Software Quality
- Software Engineering Tools & Methods
- Software Engineering Process
- Software Engineering Management
Presentation Objectives

- Give an overview of the emergience process for this international consensus on the “core body of knowledge” of software engineering
- Present the development process used to reach this consensus
- Illustrate some uses of the SWEBOK Guide
Presentation Plan

- Project background
  - Project scope, objectives, audience and development process
  - Contents of the Guide
  - Some uses of the Guide in organizations
  - Evolution of the Guide
  - Conclusion
  - Appendix: Breakdown of topics
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

Development of a Profession

Initial professional education

Skills Development

One or both
Certification Licensing

Full Professional Status

Accreditation

Professional development

Code of ethics

Professional societies

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

- Project background
- **Project scope, objectives, audience and development process**
  - Contents of the Guide
  - How can you leverage the Guide within your organization?
  - Evolution of the Guide
- Conclusion
- Breakdown of topics
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 trainers
What Was Out of Scope?

- 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>

Target of the SWEBOK Guide

«Applicable to most projects, most of the time, and widespread consensus about their value and usefulness»

Project Management Institute - PMI

○ North American Bachelor’s degree + 4 years of experience
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 of the Guide
- Industrial Advisory Board
- Associate Editors of the Knowledge Areas
- Reviewers
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
Formal resolutions

- Industrial Advisory Board (2001)
- IEEE CS Board of Governors (2001)
  - "The Board of Governors of the IEEE Computer Society accepts the Guide to the Software Engineering Body of Knowledge (Trial Version) as fulfilling its development requirements and is ready for field trials for a period of two years"
- IEEE CS Board of Governors (Feb. 2004)
  - Officially approved the 2004 Version
- ISO Technical Report 19759 (upcoming)
A Three-Phase Approach for Developing the Guide

Straw Man Phase

Stone Man Phase

Iron Man Phase (Sub-phase 1)

Iron Man Phase (Sub-phase 2)

Trial Version

2004 Version

1998 1999 2000 2001 2002 2003
Process - Strawman

Undergrad. SE curricula

Graduate Admission Criteria

Graduate SE Curricula

SE Textbooks

Preliminary list of related disciplines

ISO/IEC 12207

List of related disciplines

Proposed SE Knowledge Areas

Proposed related disciplines and SE KAs

Body of SE Standards

Must be discussed in general SE textbooks

Must be specifically adapted to SE
Stoneman- Trial Version 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
Trial Version 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
Stone Man Version 0.5
Review Results

Option 1
Choose one or more from the following lists:

- Choose a Knowledge Area
- Choose a Review Viewpoint
- Choose a Question

See Detailed Questionnaire

Search

Option 2
View all responses for a reviewer:

- Choose a Reviewer

Search

Option 3
Enter the Unique Identifier of the Response:
Comment Resolution

Guide to the SWEBOK - Stone Man Version 0.5
Review Results Report

Knowledge Area: Software design
Review Viewpoint: Researchers

**Question 1:**
Do you find that the breakdowns of topics comply with the requirement of being sound and reasonable?

<table>
<thead>
<tr>
<th>Unique Reviewer Response Identifier: 280</th>
<th>Response Disposition: No disposition yet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reviewer Response:</td>
<td>Disposition Rationale:</td>
</tr>
<tr>
<td>Yes</td>
<td></td>
</tr>
</tbody>
</table>

Reviewers:
Du, Weichang | Marcos, Esperanza | Rodeiro Iglesias, Javier

<table>
<thead>
<tr>
<th>Unique Reviewer Response Identifier: 281</th>
<th>Response Disposition: No disposition yet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reviewer Response:</td>
<td>Disposition Rationale:</td>
</tr>
<tr>
<td>The distinction between architectural and detailed design is traditional but perhaps becoming unmanageable as the size of a typical program/system grows</td>
<td></td>
</tr>
</tbody>
</table>

Reviewers:
Sander, Bo

<table>
<thead>
<tr>
<th>Unique Reviewer Response Identifier: 282</th>
<th>Response Disposition: No disposition yet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reviewer Response:</td>
<td>Disposition Rationale:</td>
</tr>
<tr>
<td>The inclusion of structure charts under architectural design suggests that we are</td>
<td></td>
</tr>
</tbody>
</table>

Document Done
Data on reviewers

Trial Version

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

- USA: 55%
- Europe: 18%
  - 90 reviewers from 25 countries
- Canada: 10%
- Australia: 5%
- Asia: 5%
- Latin America: 4%
Number of employees at reviewer location (Version 0,7)

- 31% of employees are in 0-50 employees
- 37% are in 50-500 employees
- 32% are in Over 500 employees
Number of years of practical experience (Version 0,7)
A Three-Phase Approach for Developing the Guide

Straw Man Phase

Stone Man Phase

Iron Man Phase (Sub-phase 1)

Experimentation and Trial Usage

Iron Man Phase (Sub-phase 2)

Trial Version

Revision

1998 1999 2000 2001 2002 2003 2004

Version
Reviewers (2004 Version)

- Registered reviewers: 573
- Number of countries: 55
- Number of comments: 1020
- Number of reviewers submitting comments: 124
- Number of represented countries: 21

Years in the field

Years in industry
Project Overview
Presentation Plan

- Project background
- Project scope, objectives, audience and development process

Contents of the Guide
- Uses of the Guide in organizations
- Evolution of the Guide
- Conclusion
- Appendix: Breakdown of topics
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
## Knowledge Areas and Related Disciplines

- 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 Engineering
- Computer Science
- Mathematics
- Project Management
- Management
- Quality Management
- Software Ergonomics
- Systems Engineering
Guide to the Software Engineering Body of Knowledge
(2004 Version)
Summary of changes in 2004 Version

- Standardization of the contents of the chapters in terms of table of contents, topic breakdown, terminology, reference citations and writing style

- Structural improvements in the breakdown of topics:
  - Software Construction,
  - Software Engineering Management,
  - Software Quality,
  - Software Engineering Process

- Better representation of text in topic breakdown:
  - Software Requirements,
  - Software Testing,
  - Software Maintenance
Summary of changes in 2004 Version

- New chapter on Related Disciplines (instead of an appendix)
- Better representation of standards in chapters and a new Appendix devoted to standards
- Updating of reference material
- Handling of trial usage feedback
- Handling of reviewer comments
Presentation Plan

- Project background
- Project scope, objectives, audience and development process
- Contents of the Guide
- Uses of the Guide in organizations
  - Evolution of the Guide
  - Conclusion
- Appendix: Breakdown of topics
Applications of the Guide

- Industry & Government
  - Career planning
    - Construx
  - Inter-company benchmarking
  - Hiring
  - Job and role descriptions
    - Bombardier Transportation
  - Policy making
    - Turkish Industry Survey, Alberta Software Testing Survey
Applications of the Guide

- Professional development
  - Security Industry Automation Corporation
  - SAP, Boeing Australia
  - http://www.software-kompetenz.de
Applications of the Guide

- Education
  - Course Design/Assessment:
    - École de technologie supérieure
  - Program Design/Assessment:
    - CRISTEL project
    - National Technological University
Applications of the Guide

- Licensing & Certification
  - IEEE CS CSDP
  - Ordre des ingénieurs du Québec
Presentation Plan

- Project background
- Project scope, objectives, audience and development process
- Contents of the Guide
- Uses of the Guide in organizations
- Evolution of the Guide
- Conclusion
- Appendix: Breakdown of topics
Evolution of the Guide

- Copyright belongs to the IEEE
  - They must decide the evolution of the Guide
- Transition to self-supporting, volunteer-led process—i.e. self-funded.
- Coordination with related projects (internal and external)
- Time-boxed block updates
- Involvement with stakeholder groups
- Openness and transparency
- Technical excellence
Presentation Plan

- Project background
- Project scope, objectives, audience and development process
- Contents of the Guide
- How you can leverage the Guide within your organization
- Evolution of the Guide
- Conclusion
- Appendix: Breakdown of topics
Concluding Remarks

- Consensus on the core body of knowledge is key in all disciplines and pivotal for the evolution toward a professional status
www.swebok.org
Presentation Plan

- Project background
- Project scope, objectives, audience and development process
- Contents of the Guide
- How you can leverage the Guide within your organization
- Evolution of the Guide
- Conclusion
- Appendix: Breakdown of topics
Figure 1 Breakdown of topics for the Software Design KA
Figure 1. Breakdown of topics for the Software Construction KA.
Software Maintenance

Fundamentals
- Definitions and Terminology
- Nature of Maintenance
- Need for Maintenance
- Majority of Maintenance Costs
- Evolution of Software
- Categories of Maintenance

Key Issues in Software Maintenance
- Technical
- Management

Maintenance Process
- Maintenance Process Models
- Maintenance Activities

Techniques for Maintenance
- Program Comprehension
- Re-engineering
- Reverse Engineering
- Impact Analysis
Software Engineering Process

Process Implementation and Change
- Process Infrastructure
- Activities
- Models for Process Implementation and Change
- Practical Considerations

Process Definition
- Life Cycle Models
- Software Life Cycle Processes
- Notations for Process Definitions
- Process Adaptation
- Automation

Process Assessment
- Process Assessment Models
- Process Assessment Methods

Process and Product Measurements
- Process Measurement
- Software Products Measurement
- Quality of Measurement Results
- Software Information Models
- Process Measurement Techniques
Software Engineering Tools and Methods

I. Software Tools

- Software Requirements Tools
  - Requirements modeling
  - Traceability
- Software Design Tools
- Software Construction Tools
  - Program editors
  - Compilers
  - Interpreters
  - Debuggers
- Software Testing Tools
  - Test generators
  - Test execution frameworks
  - Test evaluation
  - Test management
  - Performance analysis
- Software Maintenance Tools
  - Comprehension
  - Re-engineering
- Software Engineering Process Tools
  - Process modeling
  - Process management
  - Integrated CASE environments
  - Process-centered software engineering environments
- Software Quality Tools
  - Inspection
  - Static analysis
- Software Configuration Management Tools
  - Defect, enhancement, issue and problem tracking
  - Version management
  - Release and build
- Software Engineering Management Tools
  - Project planning and tracking
  - Risk management
  - Measurement
- Infrastructure Support Tools
  - Interpersonal communication
  - Information retrieval
  - System administrative and support
- Miscellaneous Tools Issues
  - Tool integration techniques
  - Meta tools
  - Tool evaluation

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