The SWEBOK Initiative and Software Measurement Intentions

ALAIN ABRAN
Executive Co-editor, SWEBOK Project

Pierre Bourque, Robert Dupuis (Co-editors)

Abstract

Articulating a body of knowledge is an essential step toward developing a profession because it represents a broad consensus regarding the contents of the discipline. The IEEE Computer Society, with the support of a consortium of industrial sponsors, has recently published the Guide to the Software Engineering Body of Knowledge (SWEBOK). Throughout this Guide, measurement is pervasive as a fundamental engineering tool. In addition, ISO is at present in the process of adopting this Guide as an ISO Technical Report.

This presentation will provide overviews of the development process that was followed and of the current version of this Guide. In addition, the topic of measurement will be highlighted, both in terms of its presence throughout the ten SWEBOK knowledge areas and of its depth of treatment.

Keywords: *SWEBOK, Software Engineering, Measurement, Metrics, Body of Knowledge*

1. Software Engineering as a Profession?

1.1. What is Software Engineering?

The IEEE Computer Society defines software engineering as

“(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)” [IEE90].

1.2. What is a Recognized Profession?

In spite of the millions of software professionals worldwide and the ubiquitous presence of software in our society, software engineering has not yet reached the status of a legitimate engineering discipline and a recognized profession.

For software engineering to be known as a legitimate engineering discipline and a recognized profession, consensus on a core body of knowledge is imperative. This is well illustrated by Starr when he defines what can be considered a legitimate discipline and a recognized profession. In his Pulitzer-prize-winning book on the history of the medical profession in the USA, he states that: "the legitimization of professional authority involves three distinctive claims: first, that the knowledge and competence of the professional have been validated by a community of his or her peers; second, that this consensually validated knowledge rests on rational, scientific grounds; and third, that the professional’s judgment and advice are oriented toward a set of substantive values, such as health".

1.3. What are the Characteristics of a Profession?

Gary Ford and Norman Gibbs [FOR96] studied several recognized professions including medicine, law, engineering and accounting. They concluded that an engineering profession is characterized by several components:

- An initial *professional education* in a curriculum validated by society through *accreditation;*
- Registration of fitness to practice via voluntary *certification* or mandatory *licensing*;
- Specialized *skill development* and *continuing professional education*;
- Communal support via a *professional society;*
- A commitment to norms of conduct often prescribed in a *code of ethics.*

This Guide contributes to the first three of these components.

Articulating a Body of Knowledge is an essential step toward developing a profession because it represents a broad consensus regarding what a software engineering professional should know. Without such a consensus, no licensing examination can be validated, no curriculum can prepare an individual for an examination, and no criteria can be formulated for accrediting a curriculum. The development of the consensus is also prerequisite to the adoption of coherent skill development and continuing professional education programs in organizations.

2. SWEBOK

The Body of Knowledge is subdivided into ten Knowledge Areas (KAs), and the descriptions of the KAs are designed to discriminate among the various important concepts, permitting readers to find their way quickly to subjects of interest. Upon finding a subject, readers are referred to key papers or book chapters selected because they present the knowledge succinctly.

2.1. Intended Audiences

The Guide is oriented toward a variety of audiences, all over the world. It aims to serve public and private organizations in need of a consistent view of software engineering for defining education and training requirements, classifying jobs, developing performance evaluation policies or specifying development tasks. It also addresses practicing, or managing, software engineers and the officials responsible for making public policy regarding licensing and professional guidelines. In addition, professional societies and educators defining the certification rules, accreditation policies for university curricula, and guidelines for professional practice will benefit from SWEBOK, as well as students learning the software engineering profession and educators and trainers engaged in defining curricula and course content.

2.2. Depth of Treatment

From the outset, the question has arisen as to the depth of treatment the Guide should provide. We adopted an approach providing a foundation for curriculum development, certification and licensing. We applied a criterion of *generally accepted* knowledge, which we had to distinguish from advanced and research knowledge (on the grounds of maturity) and from specialized knowledge (on the grounds of generality of application). A second definition of *generally accepted knowledge* comes from the Project Management Institute: “Generally accepted knowledge applies to most projects most of the time, and widespread consensus validates its value and effectiveness” [PMI96].

However, generally accepted knowledge does not imply that one should apply the designated knowledge uniformly to all software engineering endeavors—each project's needs determine that—but it does imply that competent, capable software engineers should be equipped with this knowledge for potential application. More precisely, generally accepted knowledge should be included in the study material for a software engineering licensing examination that graduates would take after gaining four years of work experience. Although this criterion is specific to the U.S. style of education and does not necessarily apply to other countries, it was deemed useful.

Additionally, the KA descriptions of software engineering are also forward-looking—considering not only what is generally accepted today, but also what could be generally accepted in three to five years.

Close to five hundred software engineering professionals from 41 countries and representing various viewpoints participated in the project. Professional and learned societies and public agencies involved in software engineering were officially contacted, made aware of this project and invited to participate in the review process. KA Specialists or chapter authors were recruited from North America, the Pacific Rim and Europe.

2.3. SWEBOK sponsors

The SWEBOK project has received support from the following organizations: Boeing, Raytheon, MITRE Corporation, USA National Institute of Standards & Technology, Construx Software, Rational Software, SAP Lab. Canada, NRC, Canadian Council of Professional Engineers.

3. Objectives and deliverables

3.1. A Guide

The Guide should not be confused with the Body of Knowledge itself. The Body of Knowledge already exists in the published literature. The purpose of the Guide is to describe what portion of the Body of Knowledge is generally accepted, to organize that portion, and to provide a topical access to it.

The Guide to the Software Engineering Body of Knowledge (SWEBOK) was established with the following five objectives:

1. Promote a consistent view of software engineering worldwide.
2. Clarify the place—and set the boundary—of software engineering with respect to other disciplines, such as computer science, project management, computer engineering and mathematics.
3. Characterize the contents of the software engineering discipline.
4. Provide a topical access to the Software Engineering Body of Knowledge.
5. Provide a foundation for curriculum development and individual certification and licensing material.

3.2. Consistent view of software engineering

The first of these objectives, a consistent worldwide view of software engineering, was supported by a development process that has engaged approximately 500 reviewers from 41 countries. (More information regarding the development process can be found in the Preface and on the Web site.) Professional and learned societies and public agencies involved in software engineering were officially contacted, made aware of this project and invited to participate in the review process. KA Specialists or chapter authors were recruited from North America, the Pacific Rim and Europe. Presentations on the project were made to various international venues and more are scheduled for the upcoming year.

3.3. Boundaries

The second of the objectives, the desire to set a boundary, motivates the fundamental organization of the Guide. The material that is recognized as being within software engineering is organized into the ten KAs listed in Table 1. Each of them is treated as a chapter in this Guide.

Table 1: The SWEBOK knowledge areas (KAs).

Software requirements
Software design
Software construction
Software testing
Software maintenance
Software configuration management
Software engineering management
Software engineering process
Software engineering tools and methods
Software quality

In establishing a boundary, it is also important to identify what disciplines share a boundary, and often a common intersection, with software engineering. To this end, the guide also recognizes seven related disciplines, as listed in Table 2 (See also Appendix B). Software engineers should, of course, know material from these fields (and the KA descriptions may make references to the fields). It is not, however, an objective of the SWEBOK Guide to characterize the knowledge of the related disciplines, but rather what is viewed as specific to software engineering.

Table 2: Related disciplines.

Cognitive sciences and human factors
Computer engineering
Computer science
Management and management science
Mathematics
Project management
Systems engineering

3.4. Hierarchical Organization

The organization of the KA descriptions or chapters, shown in Figure 1, supports the third of the project's objectives—a characterization of the contents of software engineering.

The Guide uses a hierarchical organization to decompose each KA into a set of topics with recognizable labels. A two- or three-level breakdown provides a reasonable way to find topics of interest. The Guide treats the selected topics in a manner compatible with major schools of thought and with breakdowns generally found in industry and in software engineering literature and standards. The breakdowns of topics do not presuppose particular application domains, business uses, management philosophies, development methods, and so forth. The extent of each topic description is only what is needed to understand the generally accepted nature of the topics and for the reader to successfully find reference material. After all, the Body of Knowledge is found in the reference materials, not in the Guide itself.

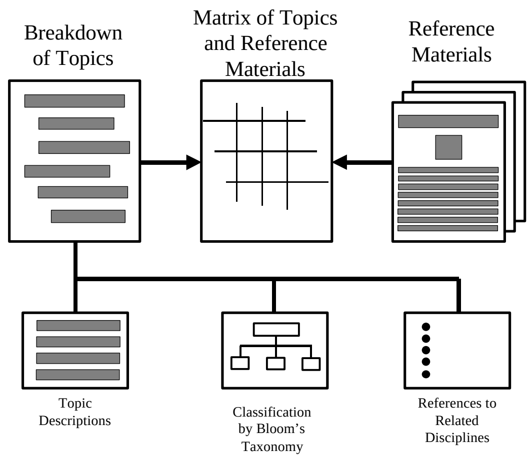

Figure 1: The organization of a Knowledge Area (KA) description

4. Measurement within SWEBOK

4.1. Editorial criteria

The topic of measurement within SWEBOK was one of the editorial criteria for the initial write-up. The KA Specialists were expected to adopt the position that, even though the quality (in general) and measurement "themes" are common across all Knowledge Areas, they are also an integral part of all KAs and therefore have to be incorporated into the proposed breakdown of topics in each KA. Since the acceptance criterion for inclusion in this Guide to the SWEBOK was 'generally accepted', as previously defined, it is important to ask what did in fact gain approval on a consensual basis with respect to measurement, and what can be learned from this consensus, or the lack of it.

For this paper, I have elected to look at this issue from three perspectives:

- Vincenti's [VIN01] classification of engineering knowledge

- ISO vocabulary of basic and general terms in metrology [ISO93]

- an analysis of a specific type of measurement method, that is, functional sizing.

4.2. Vincenti's Classification of Engineering Knowledge

Vincenti [VIN90], on the basis of his analysis of the evolution of aerospace engineering knowledge, identified different types of engineering knowledge, and classified them into six categories – Table 3.

Table 3. Classification of Engineering Knowledge – Vincenti [VIN90]

1. Fundamental design concepts
2. Criteria and specifications
3. Theoretical tools
4. Quantitative data
5. Practical considerations
6. Design *instrumentalities*

Vincenti [VIN90] postulated that this classification was not specific to aerospace engineering, but more generic and applicable to engineering in the broad sense. It had been suggested to the KA Specialists that they use this classification for their initial draft of each KA; this was, of course, a challenging assignment: the domain was not mature enough and the classification could not be directly implemented in most of the KA taxonomy and descriptions.

This classification of Vincenti's is, however, very useful from an analytical perspective to understand the depth of coverage of some engineering topics within each of the KAs. For instance, in a 2001 fall session seminar with a group of graduate students in software engineering, it was observed that, while the term "measurement" was present throughout all the KAs (by design, that is, it was a required editorial criterion), neither the KA Specialists nor the set of reviewers had pointed to key references providing generally accepted quantitative data for any of the topics identified in each KA. Similarly, an analysis of the analytical research methods used for all the references in some chapters, such as Construction, indicated that the vast majority of the statements were based on 'assertions' and 'expert judgments' rather than on the basis on experimental methods and rigorously documented experiments which could be replicated.

4.3. ISO Vocabulary of basic and general terms in metrology [ISO93]

The ISO document that represents the official national, international and legal consensus is the ISO vocabulary of basic and general terms used in metrology [ISO93]. While this key ISO document is widely known in the field of metrology, it is almost unknown in the 'software metrics' community. This vocabulary contains six sets of terms, divided into the categories listed in Table 4.

Table 4. Categories of terms in the ISO Metrology Vocabulary [ISO93]

1. Quantities and units
2. Measurements
3. Measurement results
4. Measurement instruments
5. Characteristics of the measurement instruments
6. Measurement standards – Etalons

Based on an analysis of their textual definitions and relationships defined by their semantics, the model of the set of metrology term categories listed in Table 4 is presented in Figure 2 [ABR02]

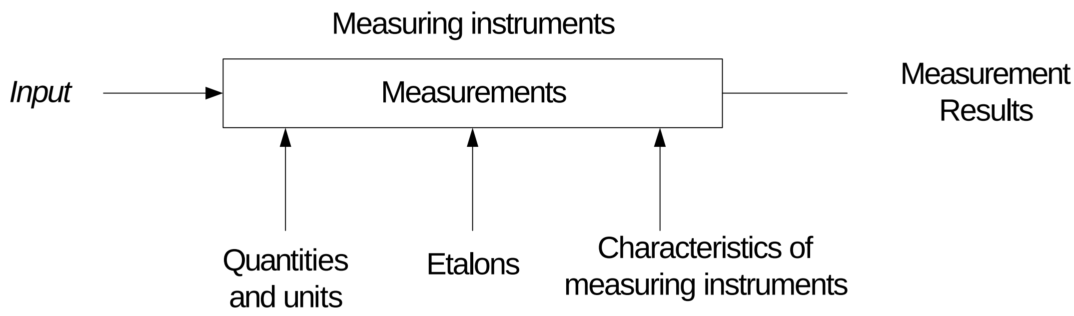

Figure 2: Model of ISO metrology term categories [ABR02]

The results of the analysis of the presence of metrology concepts within each KA are presented in Appendix A. They highlight the fact that, even though the use of measurement results is quoted in every single KA, there is a lack of discussion – or of references – on most of the other measurement concepts of the recognized body of knowledge on metrology. This is, of course, a clear indication that measurement in software engineering is far from being mature in and of itself, and that it constitutes a fairly weak engineering foundation for the field of software engineering.

4.4. Functional sizing

It can be observed that functional sizing is referred to a few of the ten KAs. This confirms the international consensus on the usefulness of such a measurement concept within the generally accepted knowledge. It is to be noted that, at the time the SWEBOK was published, only the ISO meta-standard on functional size measurement, ISO 14143-1, was referenced in the SWEBOK chapters. Other related meta-standards have since progressed significantly (e.g. parts 2 to 5 of ISO 14143 dealing with conformity assessment, verification, references and functional domains respectively). In addition, four specific methods were advanced in the process of their becoming officially recognized by ISO, that is: COSMIC [ISO 19761], IFPUG, NESMA and MKII [ISO 20968].

5. Summary

Measurement is recognized as a key element of engineering and, because of design criteria in the Guide to SWEBOK, it is pervasive in the Guide. However, it fails to cover the full spectrum of metrology concepts, and, in no KA, is there any significant reference to highly credible and documented data numbers and relevant repositories of quantitative references. The analysis based on the Vincenti classification and the metrology perspective also suggests that the field of software measurement has not yet been fully addressed by current research, and that much work remains to be done to support software engineering as an engineering discipline based on quantitative data.

6. References

[ABR02] A. Abran, A. Sellami, 'Initial Modeling of the Measurement Concepts in the ISO Vocabulary of Terms in Metrology', IWSM 2002, Magdeburg (Germany), Oct. 2002.

[DUM01] R. Dumke, M. Lother, A. Abran, 'An Approach for Integrated Software Measurement Processes in the IT Area', 4th European Conference on Software Measurement and ICT Control in co-operation with DASMA (FESMA-DASMA). Heidelberg, Germany, 2001.

[IEE90] "IEEE Standard Glossary of Software Engineering Terminology", IEEE, Piscataway, NJ, IEEE Standard 610.12-1990, 1990.

[ISO14143], ISO/IEC 14143-1:1998 Software engineering - Software measurement - Functional size measurement - Definition of concepts, International Organization for Standardization /International Electrotechnical Commission, Geneva, 1998

[ISO19761], DIS ISO/IEC 19761 Software Engineering – COSMIC FFP: A functional sizing method, International Organization for Standardization/International Electrotechnical Commission., Geneva, 2002

[ISO20968], DIS ISO/IEC 20968 MkII Function Point Analysis, International Organization for Standardization/International Electrotechnical Commission., Geneva, 2002.

[FOR96] G. Ford and N. E. Gibbs, "*A Mature Profession of Software Engineering*," Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, Technical CMU/SEI-96-TR-004, January 1996.

[PMI96] Project Management Institute, A Guide to the Project Management Body of Knowledge, Upper Darby, PA, 1996, http://www.pmi.org/publictn/pmboktoc.htm/.

[ISO93] ISO, *ISO: International Vocabulary of Basic and General Terms in Metrology,* International Organization for Standardization, Geneva, 1993.

[VIN90] Vincenti, G. W. "What Engineers Know and How They Know It – Analytical Studies from Aeronautical History," Johns Hopkins University Press, 1990.

[ZUS97] Zuse, H. *A Framework of Software Measurement*. New York, 1997.

Appendix A

Categories of metrology concepts within SWEBOK KA

SWEBOK KA and topics	**ISO Categories of Terms in Metrology**		
	Values & Units	**Measurement Instruments + Characteristics + Standards**	**Measurement Results**
Software Requirements			
Requirement Process			×
Requirements Analysis			×
Specifications			×
Validation			×
Requirements Management			
Software Design			
Design Size Evaluation	×		
Software Testing			
Program Evaluation			×
Test Evaluation			
Software Maintenance			
Cost Estimation	×		
Measurement of Maintenance			×
Software Configuration Management (SCM)			
Measurement of SCM			×
Audits of SCM			×
Software Engineering Management			
Measurement Objectives			
Measurement Selection			×
Software Measures			×
Data Collection		×	
Measurement model			×
Software Engineering Process			
Measurement Process Methodology			×
Measurement Paradigms			×
Software Quality			
Measurement Foundation	×		
Measure			×
Analysis Techniques of Measures			×
Defaults Characterization			×
Further use of SQA & V&V data			