Metrology, Measurement and Metrics in Software Engineering

Alain Abran
École de Technologie Supérieure
aabran@ele.etsmtl.ca

Asma Sellami
École de Technologie Supérieure
asma.sellami.1@ens.etsmtl.ca

Witold Suryn
École de Technologie Supérieure
wsuryn@ele.etsmtl.ca

Abstract

Up until recently software 'metrics' have been most often proposed as the quantitative tools of choice in software engineering, and the analysis of these had been most often discussed from the perspective referred to as 'measurement theory'. However, in other disciplines, it is the domain of knowledge referred to as 'metrology' that is the foundation for the development and use of measurement instruments and measurement processes.

The IEEE-Computer Society, with the support of a consortium of industrial sponsors, has recently published a Guide to the Software Engineering Body of Knowledge (SWEBOK) and, throughout this Guide, measurement is pervasive as a fundamental engineering tool. In this paper, we use our initial modelling of the sets of measurement concepts documented in the ISO International Vocabulary of Basic and General Terms in Metrology to investigate and position the measurement concepts within this body of knowledge, and to identify gaps where further research on software measurement is required.

1. Introduction

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)" [14].

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.

In software engineering, the 'software metrics' approach has been up until fairly recently the dominant approach to measurement in this new engineering discipline. Over recent decades, hundreds of 'software metrics' have been proposed by researchers and practitioners alike, in both theoretical and empirical studies, for measuring software products and software processes [1, 8-13, 25]. Most of these metrics have been designed based either on the intuition of the researchers or on an empirical basis, or both. In their analysis of some of these metrics, researchers have most often used the concepts of 'measurement theory' as the foundation for their analytical investigation. However, while relevant, 'measurement theory' deals with only a subset of the classical set of concepts of measurement; 'software metrics' researchers, by focusing solely on 'measurement theory', have investigated mainly the representation conditions, the mathematical properties of the manipulation of numbers and the proper conditions for such manipulations [11, 13, 25]. Our survey of the literature on software metrics has not, however, come up with references to the classical concepts of metrology in these investigations into the quality of the metrics proposed to the software engineering community. Only recently has some of the metrology related concepts been introduced in the ISO software engineering standards community [16-19]; it is to be noted also the selection of the ISO vocabulary on metrology [15] as the basis for measurement terminology for all future ISO standards on software measurement.

Each profession is based on a body of knowledge and recommended practices, although they are not always defined in a precise manner. In many cases these are formally documented, usually in a form that permits them to be used for such purposes as accreditation of academic programs, development of education and training programs, certification of specialists, or professional licensing. Generally a professional society or related body maintains custody of such a formal definition. In cases where no such formality is used, the body of knowledge and recommended practices are "generally recognized" by practitioners and may be codified in a variety of ways for different uses.

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 [22] 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".

Up until recently, there were no such generally accepted body of knowledge in this new field of software engineering. The IEEE-Computer Society therefore initiated in the 1990's various tasks forces to tackle this issue, including the SWEBOK (Software Engineering Body of Knowledge) project[1] [5] to develop an international consensus on a compendium and guide to the body of knowledge that has been developing and evolving over the past four decades. Furthermore, it is to be noted that this body of knowledge is not static — the Guide must, necessarily, develop and evolve as software engineering matures. Nevertheless, the Guide is a valuable element of the software engineering infrastructure.

Articulating a Body of Knowledge, and gaining a large consensus on it, 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.

But, is software measurement itself a mature tool set within this domain, and what are the analytical tools available to investigate this research topic?

This paper presents in Section 2 an overview of the SWEBOK Guide and, in Section 3, the rationale for including measurement within this Guide. Then two sets of analytical tools available are identified to investigate the state of the art of measurement in software engineering: metrology in Section 4, on the basis of the ISO International Vocabulary in Metrology and, in Section 5, a Measurement Process Model.

Then in Section 6 both our initial modeling [3] of the sets of measurement concepts documented in the ISO International Vocabulary of Basic and General Terms in Metrology and our measurement process model [2] are used to survey, and position, the measurement-related statements in the SWEBOK Guide. Metrology-related work in progress at ISO on functional size methods is presented in Section 7 and some concluding observations are presented in Section 8.

2. SWEBOK

The SWEBOK Guide [5] is subdivided into ten Knowledge Areas (KA), and the descriptions of the KA 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.

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.

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.

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[2].

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 KA listed in Table 1. Each of them is treated as a chapter in this Guide.

Table 1: The SWEBOK knowledge areas (KA)

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

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.
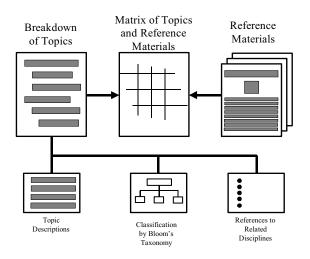
Breakdown of Topics — Matrix of Topics and Reference Materials — Reference Materials

Topic Descriptions — Classification by Bloom's Taxonomy — References to Related Disciplines

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

3. Measurement within SWEBOK

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

it. It is worth reminding that the 'generally accepted' definition adopted in SWEBOK originates from the Project Management Institute (PMI), that is: 'applies to most of the projects, most of the time, and widespread consensus validates its value and effectiveness'.

Another tool used for the development of SWEBOK, from an engineering viewpoint, is the Vincenti [24] classification of engineering knowledge. Vincenti, 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, including of course quantitative data as a category of engineering knowledge. Table 2 presents on the left-hand side, the six categories of engineering knowledge and, in the next three columns, related sub-concepts we identified subsequently for classification purposes in specific engineering disciplines.

Vincenti 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 description[3].

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 KA. 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 KA (by design, that is, it was a required editorial criterion), neither the KA editors nor the set of reviewers had pointed to key references providing generally accepted quantitative data for any of the topics identified in each KA. To be noted that in engineering 'quantitative data' does not mean 'raw data', but rather descriptive or prescriptive data derived usually from controlled experiments using widely recognized measurement concepts, verified measurement instruments, documented measurement protocols, and extensive testing and replication procedures to ensure both verification of data inputs and an in-depth understanding of the phenomena under study to identify both the range of operations, and limitations.

In summary, in no KA, is there any significant reference to highly credible and documented data numbers and relevant repositories of quantitative references. This means for instance, that while there is

---

[3] Software being inherently different from other physical artifacts, there was no attempt to applying general engineering principles to software. The interest was rather in mapping the software engineering knowledge to the categories of engineering knowledge type.

in the software engineering management literature a very large number of papers on estimation, the raw data sets available for study often lack engineering rigor in the data collection procedures, on the one hand, and analytical results have usually both poor explanatory powers and significant limitations in generalization powers, on the other hand.

As another illustration, in both versions of the COCOMO model [6, 7], a large number of parameters are described by linguistic values and their influence determined by experts opinions rather than on the basis of information from descriptive and prescriptive engineering repositories; of course significant effort is currently being invested by the models builders to build engineering strength into such models.

Similarly, even after twenty years of data collection and research, the function points community is still struggling to come up with 'a posteriori' models with error ranges of less than 25% for 75% of the data in the available samples (typically in engineering, error ranges are expected to be much less than 25%).

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 then on the basis on experimental methods, quantitative data and rigorously documented experiments which could be replicated [23].

The next two sections present two other sets of analytical tools to analyze the measurement coverage within SWEBOK.

Table 2: Classification of Engineering Knowledge, and related sub-concepts

Category [24]	Related sub-concepts		
Fundamental Principles	• Operational principles • How its characteristic parts fulfill their special functions in combining to an overall operation which achieves the purpose	Normal configuration–common arrangement of the constituent parts	Vincenti: 'to be learned deliberately and forms an essential part of the *'design'* knowledge'
Criteria & Specifications	• Specific requirements (of operational principles) • Limits (across an entire technology)	• To translate general, qualitative goals couched in concrete technical terms • Note: the '…ilities'	Key knowledge: selection of the appropriate set of criteria
Theoretical Tools	• Concepts about 'design' • Intellectual tools for thinking about 'design'	• Mathematical methods & theories for design calculations • Models (combinations of measures/parameters)	Methods of value are micro-methods, closely tailored to the tasks of developing particular well understood parts of particular well understood products
Quantitative Data	Represented in tables and graphs	Obtained empirically or calculated theoretically	Descriptive or Prescriptive
Practical Considerations	• Theory often not sufficient –considerations from experience and practice • Trade-off which are results of general knowledge about the device, its use and context	• Structured procedures • Practically derived knowledge learned from on-the-job experience • From an expert's skills set	• Ad-hoc assumptions about a phenomenon • Not formally codified but represented by Rules of thumb
Design Instrumentalities	• Knowing How – ways of thinking • Procedural Knowledge	Can seek solutions where some element of novelty is required	• Judgmental Skills • Knowledge on how to carry out tasks • Repeatable and documented

4. Metrology

In engineering as well as in other fields such as business administration and a significant number of the social sciences, measurement is one of a number of analytical tools. Measurement in these other sciences is based on a large body of knowledge; such a body of knowledge, built up over centuries and millennia, is commonly referred to as the field of 'metrology'. This domain is supported by government metrology agencies, which are to be found in most industrially advanced countries.

The ISO document that represents the official international and legal consensus is the ISO vocabulary of basic and general terms used in metrology [15]. While this key ISO document is widely known in the field of metrology, it is almost unknown in the 'software metrics' community.

This ISO Vocabulary follows some of the concepts of the traditional presentation of vocabularies, with 120 terms described individually in textual descriptions. However, this mode of representation is challenging in terms of assembling the full set of interrelated terms; to improve the presentation and the understanding of this complex set of interrelated concepts, we presented in [3] an initial set of models for the various levels of metrology concepts within the ISO Vocabulary.

The high-level model of the set of categories of terms is presented in Figure 2. This model, together with some sub-models presented later on, corresponds to our current understanding of the topology integrated into the vocabulary of this specialized area of the body of knowledge relating to metrology. To represent the relationships across the terms, the classical representation of a production process was selected: e.g. input, output and control variables, as well as the process itself inside the box. In Figure 2, the output is represented by the 'measurement results' and the process itself by the 'measurement' in the sense of measurement operations, while the control variables are the 'étalons'[4] (official yardsticks) and the 'quantities and units'. This set of concepts represents the 'measuring instrument'. It is to be noted that the measurement operations, and, of course, the measurement results, are influenced by the 'characteristics' of the measuring instruments.

In the Vocabulary, the term 'measurements' used as a single term corresponds to the 'set of operations' used for measuring. Also, in all figures and tables in this paper, a term taken directly from the ISO Vocabulary will appear in roman type, while terms representing

---

[4] Étalon: for instance, an internationally recognized material yardstick: the physical 'meter' etalon in length measurement recognized as the official 'étalon' for the meter. Étalons are also refined over time: for instance, the official definition of the meter has changed in 1983: it was then defined as the distance performed by the light, in an empty medium, in 1/299 792 458 second.

---

concepts not specifically listed will appear in italics; for instance, in Figure 2, we have added the term 'Input', which is not included in any of the six categories of the ISO Vocabulary. Models of each of these six categories of metrology terms are presented in [3].
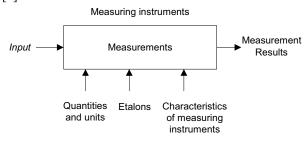

Figure 2: Model of the categories of metrology terms [3]

The term 'metrology' includes all aspects of measurement (theoretical and practical), collectively referred to in the metrology literature as the science of measurement (Figure 3). Metrology encompasses the 'principles of measurement', which represent the scientific basis for measurement. From the principles of measurement, the 'method of measurement' in the general sense is then instantiated by a measurement as a set of operations. Figure 3 depicts this hierarchy of concepts.
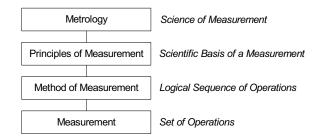

Figure 3: Measurement foundations [3]

The detailed topology of the measurement process is instantiated next in a 'measurement procedure' (Figure 4), again as a process model having the 'measurand' as its inputs, control variables and an output representing the 'measurement results'.

To carry out a measurement exercise, an operator should design and follow a 'measurement procedure' which consists of a set of operations, specifically described, for the performance of a particular measurement according to a given measurement method. The instantiation of a measurement procedure handles a 'measurement signal' and produces a transformed value, which represents a given measurand. The results of the measurement can be influenced by an 'influence quantity' during the measurement process: for example, the temperature of a micrometer during the measurement of the length of a particular object.
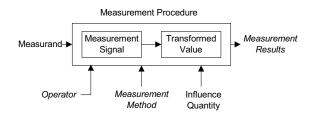
Measurement Procedure

Figure 4: Measurement Procedure [3]

The category 'measurement results' is presented next in the form of a structured table according to the types of measurement results, the modes of verification of the measurement results and information about the uncertainty of measurement – Table 3. Again, this structure is our own.

Table 3: Classification of terms in the category of 'Measurement Results' [3]

Types of measurement results	*Modes of verification of measurement results*	**Uncertainty of measurement**
Indication (of a measuring instrument) Uncorrected result Corrected result	Accuracy of measurement Repeatability (of results of measurements) Reproducibility (of results of measurements)	Experimental standard deviation Error (of measurement) Deviation Relative error Random error Systematic error Correction Correction factor

5. A measurement process model

In their work as ISO editors for the Guide to the Verification of Functional Size Measurement Methods (ISO 14143-3) [17], Abran and Jacquet studied the various software engineering authors dealing with 'metrics validation' [2, 20, 21]. Significant variations were found in the authors' approaches as well as the use of similar terms by these authors, but with very significant differences in the related concepts.

To clarify the confusion due to the inconsistent terminology used by these authors, a broader measurement process model was proposed (Figure 5) identifying 4 distinct steps, from the design of a measurement method to the exploitation of the measurement results [2]. Then, the approaches of the various authors, as well as the validation concepts that were being addressed differently by these authors, were sorted out depending on whether or not they were addressing validation issues related to Steps 1 to 4 of the process model in Figure 5.
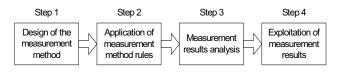

Figure 5: Measurement Process – High-level Model [2, 20, 21]

It is to be noted that very few of the measurement concepts present in the ISO Vocabulary on Metrology address the first step (design of a measurement method) and none address the last step (exploitation of the measurements results) of Figure 5.

This is illustrated in Table 4, which depicts a partial mapping between Figures 2 and 5: for instance, in [2, 20], for the design of a measurement method, the Abran and Jacquet model includes more concepts than simply 'quantities and units'.

Table 4: Alignment of metrology concepts with the measurement process model

Measurement process model [2, 20, 21]	**Step 1** Design of Measurement Methods	**Step 2** Application of measurement method rules	**Step 3** Measurement results analysis	**Step 4** Exploitation of measurement results
ISO metrology model [3, 15]	• Quantities and units	• Measuring instruments • Characteristics of measuring instruments	• Measurement results	

6. Measurement steps and metrology concepts within SWEBOK

Using both the ISO set of metrology concepts model [3, 4] and the measurement process model [2, 20, 21], we can analyze the current status of the field of software measurements as documented in the SWEBOK Guide.

The results of the analysis of the presence of metrology concepts within each KA are presented in Table 5. Using a detailed inventory of the measurement-related statements appearing in the ten SWEBOK chapters, these statements were analyzed in terms of measurement concepts, and then mapped into both the set of metrology concepts presented in Section

4 and to the measurement process model presented in Section 5.

Table 5 lists, for each of the ten chapters of SWEBOK, which metrology concepts and measurement steps are addressed whenever a measurement-related statement appears in the SWEBOK Guide [4]. From Table 5, it can be observed that a large majority of the measurement-related concepts mentioned in SWEBOK are listed in the category of concepts related to the exploitation of the measurement results. Very few SWEBOK statements directly address the measuring instrument or the quality of the direct measurement results (prior to their use in quantitative analytical models (assessment models or predictive models)). And only one measurement related statement in the Software Quality chapter addresses a single aspect of the design of measurement instrument, and only through a subset of the metrology concepts of quantities and units.

Table 5: Measurement steps and metrology category of concepts within SWEBOK [4]

SWEBOK Knowledge Area	Measurement Steps — Topics	Step 1 Design of measurement methods (Quantities and units)	Step 2 Application of measurement method rules (Measuring instruments)	Step 3 Measurement results analysis	Step 4 Exploitation of measurement results
Software engineering requirements	Process support and management				×
	Requirements negotiation				×
	Document quality				×
	Acceptance tests				×
	Requirements tracing				×
Software engineering design	Measures			×	
Software engineering testing	Evaluation of the program under test				×
	Evaluation of the tests performed				×
Software engineering maintenance	Software Maintenance Measurement				×
Software configuration management	Surveillance of software configuration management				×
Software engineering management	Goals				×
	Measurement Selection				×
	Measuring Software and its Development				×
	Collection of data		×		
	Software Measurement Models			×	
Software engineering process	Methodology in process measurement		×		
	Process Measurement Paradigms				×
Software engineering quality	Measuring the value of quality				×
	Fundamentals of Measurement	×			
	Measures			×	
	Measurement analysis techniques				×
	Defect characterization				×
	Additional Uses of SQA and V&V data				×

This highlights the fact that, even though the use of measurement results is quoted in most KA, both the KA editors and the extensive number of reviewers have not been able to come up and agreed on the availability of knowledge on measurement concepts which met the SWEBOK and PMI criteria of generally accepted, that is of 'applies to most of the projects, most of the time, and widespread consensus validates its value and effectiveness'. This does not mean that such other types of measurement knowledge do not exist in the literature, but rather that there is not yet a wide consensus on their value and effectiveness and their generalization power outside of the initial context of operations. It also points out to a significant lack software measurement methods with enough strengths as measurement instruments and meeting the metrology criteria for quality (of measuring instruments). Table 5 also points out to a lack of widely recognized and validated quantitative data to support yet the quality expected from an engineering viewpoint for the software engineering topics described.

This, of course, corresponds to a lack of recognized references to other measurement concepts from the recognized body of knowledge on metrology. This is a clear indication that, when looked at from an engineering perspective, measurement in software engineering is far from being mature and that it constitutes a fairly weak engineering foundation for the field of software engineering.

7. Functional size measurement methods and metrology

In the ISO software engineering community, there is on the other hand work in progress to investigate and apply the metrology concepts to software measurement standards, including within specific measurement methods. The first type of measurement methods tackled at the ISO level were the functional size methods which are referred to in a few of the ten SWEBOK KA, indicating some international consensus on the usefulness of such a measurement concept. 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 [16], dealt with some of the design issues of measurement methods, and was referenced in the SWEBOK chapters. Other related ISO 14143-i meta-standards have since progressed significantly and are expected to be published in 2003:

1. Part 2 of 14143: dealing with conformity assessment of the design of proposed functional size methods.
2. Part 3 of ISO 14143: dealing with the verification criteria of a functional size method to assist

measurements users in selecting the methods most appropriate to their needs.
3. Part 4 of ISO 14143: providing a large set of functional user requirements against which candidate measurement methods can be tested.
4. Part 5 of ISO 14143: providing users with the information for analyzing which measurement method is most appropriate to the functional domain of the software to be measured.

In addition, four specific methods have received final ISO approval and are currently at various stages of publication, that is: IFPUG, NESMA, MKII and COSMIC [19], a second generation functional size measurement method. Many of the metrology related concepts have already been integrated into the design of the COSMIC method (ISO 19761), with particular attention paid to the characterization of the concept being measured, to the selected meta-model of the functionality, and to the units and quantities in the definition of the numerical assignment rules.

8. Concluding observations

While 'software metrics' are most often proposed as the measurement tools of choice in empirical studies in software engineering, this field of 'software metrics' has most often been discussed from the perspective referred to as 'measurement theory'. However, in other disciplines, it is the domain of knowledge referred to as 'metrology' that is the foundation for the development and use of measurement instruments and measurement processes.

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. But, is measurement already a mature tool set within this domain, and what were the analytical tools available to investigate this research topic?

In this paper, we have first identified three sets of analytical tools to investigate the state of the art of measurement in software engineering:

- Vincenti's classification of engineering knowledge [24]
- ISO International Vocabulary of Basic and General Terms in Metrology [15]
- Measurement Process Model (Abran & Jacquet, 1999 [2])

We then next used both our initial modeling of the sets of measurement concepts documented in the ISO International Vocabulary of Basic and General Terms in Metrology and our measurement process model to survey, and position, the measurement-related

statements in the Guide to the Software Engineering Body of Knowledge.

This has revealed that, even though measurement-related statements appear throughout the SWEBOK document, they overwhelmingly concern the use of measurement results in assessment and predictive models. By contrast, there is in the document very little widely recognized validated knowledge from an engineering perspective, little on the quality of the quantitative inputs to these models, and almost nothing on supporting measuring instruments necessary to obtain these inputs.

Similarly, in the software engineering literature, even though there is a large number of 'metrics' proposed, there is still very little discussion on the topic of measuring instruments so overwhelmingly present in the traditional engineering disciplines.

This also illustrates that most of the metrology concepts, and sub-concepts, have not yet been discussed or addressed to a significant extent in the 'software metrics' literature. In the context where measuring instruments are necessary key elements of empirical studies, this points to a potentially significant weakness in current empirical studies in software engineering, while at the same time providing an indication of where metrology-related improvements in software measurement could contribute significantly to strengthening future empirical studies in software engineering.

This analysis based on various analytical techniques such as the Vincenti's classification and the metrology perspective 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 and adequate measurement methods meeting the classic set of criteria for measuring instruments as described by the metrology body of knowledge in large use in the engineering disciplines.

Further work is in progress aimed at a more in-depth study of each measurement-related statement in all SWEBOK chapters, which also includes an analysis of the seminal references quoted in each chapter dealing with measurement-related concepts.

9. Acknowledgments

10. References

[1] A. Abran and R. Dumke, "A Framework of the Software Measurement Domain: Current Situation and Future Directions", International Workshop on Software Measurement - IWSM, Montreal (Canada), 1997.

[2] A. Abran and J.P. Jacquet, "A Structured Analysis of the New ISO Standard on Functional Size Measurement-Definition of Concepts", *Fourth IEEE International Symposium and Forum on Software Engineering Standards*, 1999, pp. 230-241.

[3] A. Abran and A. Sellami, "Initial Modeling of the Measurement Concepts in the ISO Vocabulary of Terms in Metrology", in Software Measurement and Estimation - Proceedings of the 12th International Workshop on Software Measurement - IWSM, Magdeburg (Germany) Oct. 7-9 2002, Shaker-Verlag, Aachen, 2002, ISBN 3-8322-0765-1, pp. 315.

[4] A. Abran and A. Sellami, "Measurement and Metrology Requirements for Empirical Studies in Software Engineering", IEEE Software Technology and Engineering Practice Workshop (STEP), Montreal (Canada), 2002.

[5] A. Abran, J. Moore, P. Bourque, R. Dupuis, and L. Tripp, *Guide to the Software Engineering Body of Knowledge - SWEBOK*, IEEE-Computer Society Press: www.swebok.org, Los Alamitos (USA), 2001, pp. 206.

[6] B.W. Boehm, *Software Engineering Economics*, New York, Prentice Hall, 1981.

[7] B.W. Boehm and C. Abst, *Software Cost Estimation with COCOMO II*, Prentice Hall, 2000, pp. 502.

[8] S.R. Chidamber and C.F. Kemerer, "A Metrics Suite for Object Oriented Design", *IEEE Transactions on Software Engineering*, vol. 20, no. 6, 1994, pp. 476-493.

[9] E. Dimitrov, R. Dumke, E. Foltin, and M. Wipprecht, "Conception and Experience of Metrics Based Software Reuse in Practice", International Workshop on Software Measurement (IWSM), Lac Supérieur, Québec, 1999.

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

[11] N.E. Fenton and S.L. Pleeger, *Software Metrics: A Rigorous and Practical Approach*,

second edition, PWS Publishing Co., 1997, pp. 638.

[12] T. Fetcke, A. Abran, and R. Dumke, "A Generalized Representation for Selected Functional Size Measurement Methods", in *Current Trends in Software Measurement - Proceedings of the 11th International Workshop on Software Measurement*. Montréal: IWSM, Shaker-Verlag, Aachen; 2001, 3-8265-9681-1, 2001, pp. 1-25.

[13] M. Hitz and B. Montazeri, "Chidamber and Kemerer's Metrics Suite: A Measurement Theory Perspective", *IEEE Transactions on Software Engineering*, vol. 22, no. 4, 1996, pp. 267 -271.

[14] IEEE IEEE 610.12: 1990, *IEEE Standard Glossary for Software Engineering Terminology*, IEEE Computer Society, USA, 1990.

[15] ISO, *International Vocabulary of Basic and General Terms in Metrology*, International Organization for Standardization - ISO, Geneva, 1993, pp. 49.

[16] ISO ISO/IEC 14143-1: 1998, *Information technology -- Software measurement -- Functional size measurement -- Part 1: Definition of concepts*, International Organization for Standardization - ISO, Geneva, 1998.

[17] ISO ISO/IEC 14143-3: 2003, *Software Engineering- Functional Size Measurement- Part 3: Verification of Functional Size Measurement Methods*, International Organization for Standardization - ISO, Geneva, 2003.

[18] ISO ISO/IEC 15939: 2002, *Information Technology - Software Engineering - Software Measurement Process*, International Organization for Standardization - ISO, Geneva, 2002.

[19] ISO ISO/IEC 19761: 2003, *Software Engineering -- COSMIC-FFP -- A Functional Size Measurement Method*, International Organization for Standardization - ISO, Geneva, 2003.

[20] J.P. Jacquet and A. Abran, "From Software Metrics to Software Measurement Methods: A Process Model", in *International Software Engineering Standards Symposium and Forum, ISESS 97*: IEEE-Computer Society Press, 1997, pp. 128-135.

[21] J.P. Jacquet and A. Abran, "Metric Validation Proposals : A Structured Analysis", in *Software Measurement - Current Trends in Research and Practice, Proceedings of the 8th International Workshop on Software Measurement - IWSM 98*: Deutscher Universität Verlag, 1999, pp. 43-59.

[22] P. Starr, *The Social transformation of American Medicine*, Basic Books, 1982, pp. 15.

[23] W. Suryn, F. Robert, A. Abran, P. Bourque, and R. Champagne, "A Technical Review of the Software Experimental Support Analysis of the Software Construction Knowledge Area in The SWEBOK Guide (Trial Version 1.0)", in IEEE Software Technology and Engineering Practice Workshop (STEP), Montreal (Canada), 2002.

[24] G.W. Vincenti, *What Engineers Know and How They Know It - Analytical Studies from Aeronautical History*, Johns Hopkins University Press, 1990, pp. 326.

[25] Zuse H., *A Framework for Software Measurement*, Walter de Gruyter, Germany, Berlin, 1997, pp. 755.

IEEE
COMPUTER
SOCIETY