

Open BOK Context

Subjects: [Education & Educational Research](#) | [Others](#) | [Computer Science, Information Systems](#)

Contributor: Pablo Alejandro Quezada-Sarmiento , Jon A. Elorriaga , Ana Arruarte , Hironori Washizaki

A Body of Knowledge (BOK) is a concept used to represent concepts, terms, and activities that make up a professional domain. In addition, an Open BOK is necessary because it allows us to develop the abilities and talents of professionals in different Knowledge Areas (KAs).

Body of Knowledge

Education

OpenBok

1. Introduction

The goal of knowledge description is to reach a consensus on the core subsets of the knowledge characterizing engineering disciplines ^[1], and it is a well-known fact that developing an Open BOK is a complex task. This is done by considering the fact that knowledge can often be represented as interconnected BOK, KAs, Knowledge Units (KUs), and Knowledge Topics (KTs) ^[1].

The main guide that is used for the description of the necessary knowledge of technical academic disciplines is Software Engineering Body of Knowledge (SWEBOK V. 3.0) ^[1], which generally describes accepted knowledge about Software Engineering (SE). This guide is taken as a reference for the implementation of SE in industrial contexts ^{[2][3][4]}, in educational contexts ^{[5][6][7][8]}, and in Information Technology (IT), Governance, where the focus is on how the knowledge is being described ^[9].

In accordance with ^[10], from 2000 to 2010, knowledge was represented as knowledge, skills, and attitudes accepted and applied by investment professionals worldwide.

Furthermore, in ^[11] it is mentioned that knowledge was often written in a specific language with rules and algorithms that are not compatible with other Knowledge-Based Information Technology (KBE-IT) frameworks. In short, articulating an Open BOK is of paramount importance, because it is an essential step to develop an academic profession ^[12]. Nevertheless, a set of widely agreed guidelines on how to develop these BOKs and, more specifically, on the way to describe the knowledge, is not yet available.

2. Open BOK Context

First, Open BOK is used by those who are interested in expanding their skills and professional training in different areas of knowledge. For the scientific community, BOK allows for the widening of the spectrum of research fields

based on consensus and highlights similarities between disciplines [13]. For example, BOK highlights techniques used in materials science that are common between chemistry and physics [14].

Regarding the knowledge levels of a BOK, the amount of knowledge that will be offered within an educational program is defined in [15]. BOK has a specific structure according to the area of engineering or science in which they are applied.

Second, according to [14][16], to establish the description of the BOK, it is necessary to consider the Core Book (CB), and Context Domain (CD) of the BOK study area. In the same way, BOK must establish their respective KAs. Each description of KAs should use the structure shown in [1].

Moreover, as part of this second finding, it can be said that KA divided into smaller divisions called KU [14][16][17][18], which represent individual thematic modules within a KA. Each KU is subdivided into a set of topics, which are the lowest level of the hierarchy. The themes depend on the evolution and context of the KAs and the discipline.

Third, in the Open BOK context, it is also necessary to standardize a knowledge updating process according to how advanced the discipline is and the existing needs of the communities. In general, BOK has different committees, organizations, and collaborative groups that develop and update their contexts considering the progress of science and its areas of knowledge.

Fourth, in order to build an Open BOK with a bottom-up approach (Knowledge Sweep), researchers must consider the '*materials*' from which the knowledge is extracted by the discipline-directed. When analyzing these materials, it is assumed that a certain degree of knowledge could be obtained and used to formulate an Open BOK.

The reference materials will be the scientifically agreed information [7][8][19][20], and the matrix of topics is divided into details in order to establish its relationship with the respective materials.

Moreover, a list of readings should be considered to complement the information of the proposed KAs. In the same context, when an Open BOK knowledge is developed, it is necessary to establish the origin of the information.

Fifth, it was found that there exist structures, elements, descriptions, and learned lessons of the BOK evolution. In order to show the evolution of BOK, this entry provides the structure, versions, and learned lessons synthesized in Table 1.

Table 1. Structure and version of relevant bodies of knowledge.

Body of Knowledge	Structure	Versions
Name		

B1: BOK for MPM [21] .	The Structure of the BOK has been outlined to identify the various knowledge, and skills required by today's medical practice executive [21] .	Three versions [22] .
B2: Usability Body of Knowledge [23] .	The Body of Knowledge is organized in an architectural hierarchy [23] .	Conditions, and circumstances that are relevant to an event, fact or knowledge, in the process of being organized.
B3: The Personal Software Process (PSP) Body of Knowledge [24] .	This BOK is organized in an architectural hierarchy in which the concepts and skills of the PSP are described and decomposed into three levels of abstraction [24] .	Unique version [24] .
B4: SLA BOK [25] .	The SLA BOK is organized by competency clusters and knowledge areas. Individual competencies (CMP) include skills, related competencies, examples and high maturity skills [25] .	Unique version [25] .
B5: SWEBOK [1] .	Hierarchical structure using different levels of topics.	Three version [1] .
B6: PMBOK [26] .	Hierarchical structure using different levels of topics.	Six versions [26] .
B7: ITS BOK [27] .	Structured by well-defined competencies, notional security roles, four primary functional perspectives, and an IT Security Role, Competency, and Functional Matrix.	Unique version [27] .
B8: WEBOK [28] .	Hierarchical structure using different levels of topics [28] .	Unique version [29] .

B9: ITBOK.	Hierarchical structure. 13 knowledge areas [28] .	Unique version [28] .
B10: SEBOK [30] .	Sevent parts.	SEBOK has versions 1.0 to 1.4 with very small changes in between. [30] .
B11: BKCASE [31] .	The BKCASE project is of courses structured similarly as the SEBOK itself [31] .	Three versions [31] .
B12: EABoK [32] .	Hierarchical structure.	Unique version [32] .

Sixth, another finding of this article is the establishment of a general structure of an Open BOK in engineering. This structure begins with the set of KAs, continues KUs, and ends with topics according to the research area.

Seventh, Open BOKs provide the foundation for curriculum development and maintenance [\[33\]](#).

Open BOK promotes integrations and connections with other related disciplines [\[34\]](#).

Eighth, at the level of professional education in engineering contexts BOK, should provide the following detail levels [\[34\]](#):

- Know the basic concepts and the main areas of application.
- Know the basic technologies and their relationship with basic concepts.
- Know both authorized and unauthorized sources of information, and how to evaluate the quality of the information.
- Have the ability to work with standards.

3. Open BOK in an Educational Context

Furthermore, as part of this finding, educational programs in engineering and engineering technology have been developed to address many aspects associated with computer science [\[35\]](#). For example, the BOK of Computer Science Technology, the SWEBOK, and the IT BOK are based on inputs provided from various perspectives, including industry demand, previous works in the creation of computer BOKs, and institutional factors.

Knowledge should reflect current best practices, which inevitably change over time. However, updates cannot be carried out in an uncontrolled manner, since associated conferences and other educational materials must be kept

in line with the BOK [36].

Finally, other important factors to consider in BOK are the Stakeholders [7], which are people, groups, companies, and either organizational or governmental entities that have an interest in educational programs.

All interested parties must be identified as well as their responsibilities towards educational programs based on BOK (RaPSEEM) [35]. An Open BOK has an important role in the advancement of an area as a knowledgeable practice [36]. SE is a young field of human experience if compared with others. However, the knowledge in this field has evolved at a very high speed, which is a characteristic of Computer Science in general [37].

SWEBOK provides a consensually validated characterization of the bounds of the software engineering discipline and to provide access to the BOK supporting that discipline [38]. On the other hand, SWEBOK [3] is oriented toward the private and public sector for this reason, the aims of the SWEBOK guide in the process of training, education, and evaluation of professional of software engineering. The SWEBOK knowledge architecture in this report provides a hierarchical description and decomposition of a body of knowledge for software engineering [39].

For the purposes of this article, the term “*knowledge*” is used to describe the whole spectrum of content for the discipline: information, terminology, artifacts, data, roles, methods, models, procedures, techniques, practices, processes, and literature [40].

The GSWEBOK is a good first step in characterizing the contents of the software engineering discipline and in providing topical access to the SWEBOK.

Figure 1 shows the three levels of abstraction and the relationships that were used in modeling SWEBOK v 3.0.

Figure 1. Levels of abstraction of SWEBOK [1].

A hierarchical description of software engineering knowledge that organizes and structures the knowledge into three levels of hierarchy KC, KA, and KU [1]; in the same context, the highest level of the hierarchy is the education KA, representing a particular subdiscipline of SE that is generally recognized as a significant part of the SWEBOK that an undergraduate should know [31]. In particular, the curricular recommendations for an undergraduate degree program as put forward by the Working Group on SEET are considered [41].

4. Elements to Describe Knowledge on Open BOK

The Curriculum of SEE evolved in terms of a new design, revised, minor and major changes [42]. Software engineering curriculum (SEC) implementation and assessment in academia took place in different regions all over the world [43]. The process of building the Open BOK should assist in highlighting similarities across disciplines, for example, techniques used in materials science [44].

Elements needed to describe Open BOK is presented in Table 2. These elements permit an adequate description of Open BOK.

Table 2. BOK elements to describe knowledge on Open BOK.

Elements	Association	Description
C1. Domain	C1.1 Context	Knowledge identified by name, context, and application.
	C1.1.1 BOK Application	
	C1.2 Stakeholders	
	C1.3 Education	
	C1.4 Industry	
C2. Knowledge Organization	C2.1 BOK Content	The conditions, and circumstances that are relevant to an event, fact or knowledge, in the process of being organized.
	C2.1.1 Disciplines	
	C2.2 Structure	
	C2.2.1 Knowledge Categories	
	C2.2.2 Hierarchical Organization	
	C2.2.2.1 Knowledge Areas	
	C2.2.2.1.1 Organization of Knowledge Area	

Elements	Association	Description
	C2.2.2.1.2 Breakdown of Topics	
	C2.2.2.1.2.1 List of Future Readings	
	C2.2.2.1.2.2 List of Acronyms	
	C2.2.2.1.2.3 References	
	C2.2.2.1.2.3.1 Materials	
	C2.2.2.1.2.3.2 Matrix	
	C2.2.2.1.2.3.3 Related Disciplines	
	C2.2.2.1.2.4 Taxonomies	
	C2.2.2.1.2.4.1 Types of Taxonomies	
	C2.2.2.1.2.4.2 Levels of Taxonomies	
	C2.2.2.1.2.4.3 Application of Taxonomies	
	C2.2.2.1.3 Knowledge Organization	
	C2.2.2.1.3.1 Knowledge Unit	
	C2.2.2.1.3.1.1 Knowledge Topic	
	C2.2.2.1.3.1.2 Knowledge Subtopic	
C3. Knowledge Representation	C3.1 Concepts	Characteristics to represent knowledge in an educational context.
	C3.2 Supporting Tools	
	C3.3 Ontology	
	C3.3.1 Models	
	C3.3.2 Vocabulary	
	C3.4. Skills	

Elements	Association	Description
	C3.4.1.1 Instructional Skills	
	C3.4.1.1.1 Types of Skills	
	C3.4.1.1.1.1 Technical	
	C3.4.1.1.1.2 Pedagogical	
	C3.4.1.2 Capacities	
	C3.4.1.3 Capabilities	
C4. Domain Management	C4.1. BOK Areas	Structure of BOKs, where topics are thoroughly detailed.
	C4.2 BOK Details	
	C4.3 BOK Structure	
C5. Knowledge Acquisition	C5.1 Types of Standards	Ways to acquire Knowledge.
	C5.2 Application of Standards	
	C6.1 Consensus	
	C6.2 BOK Objective	
	C6.2.1 Scope of BOK	
C6. Evolution Boks	C6.3 Type of BOK	Any process of formation, growth or development in the BOK context.
	C6.4 Knowledge Acquisition	
	C6.4.1 Lessons Learned	
	C6.4.2 Material	
C7. Knowledge Resource	C7.1 Guides	Resources about the BOK.
	C7.2 Communities	
	C7.3 Standards	
C8. Knowledge Education	C8.1 Education	A set of characteristics that identify a knowledge within the educational context
	C8.1.1 Profile	
	C8.1.2 Guidelines for Profiles	
	C8.1.3 Educational Institution	

Elements	Association	Description
	C8.1.4 Educational Training	
	C8.1.4.1University Curricula	
	C8.1.4.1.1 Curriculum	
	C8.1.4.1.1.1Curriculum Process	
	C8.1.4.1.1.2 Curriculum Develop	
	C8.1.4.1.1.3Curriculum Resource	
	C8.1.4.1.1.4 Curriculum Architecture	
	C8.1.4.1.1.5 Code of Ethics	
	C8.1.4.2 BOK Accreditation	
	C8.1.5 Professional Certification	
	C8.1.5.1 Evaluation Policies	
	C8.1.5.2 Licensing	
	C8.1.5.2.1 Competences	
	C8.1.5.2.2 Certification	
	C8.1.5.3 Professional Standard	
	C8.1.5.4 Professional Practice	
	C8.1.5.5 Professional Development	
	C8.1.6 Educational Objectives	
	C8.1.7 Committees	
	C8.1.8 Innovation	

References

1. Bourque, P.; Dupuis, R. Guide to the Software Engineering Body of Knowledge 2014 Version, SWEBOK. 2014. Available online: <https://www.computer.org/education/bodies-of-knowledge/software-engineering> (accessed on 18 March 2020)
2. Pérez, F.; Marcen, A.C.; Lapena, R.; Cetina, C. Evaluating Low-cost in internal crowdsourcing for software engineering: The case of feature location in an industrial environment. *IEEE Access* 2020, 8, 65745–65757.
3. Napier, N.P.; Mathiassen, L.; Johnson, R.D. Combining perceptions and prescriptions in requirements engineering process assessment: An industrial case study. *IEEE Trans. Softw. Eng.* 2009, 35, 593–606.
4. Mylopoulos, J.; Chaudhri, V.; Plexousakis, D.; Shrufi, A.; Topologlou, T. Building knowledge base management systems. *VLDB J.* 1996, 5, 238–263.
5. Hill, E.; Johnson, P.M.; Port, D. Is an athletic approach the future of software engineering education? *IEEE Softw.* 2016, 33, 97–100.
6. Quezada-Sarmiento, P.A.; Morocho-Quezada, M.; Pacheco-Jara, L.; Garbajosa, J. Evaluation of occupational and professional profiles in ecuadorian context based on guide of knowledge SWEBOK and ontological model. In *Proceedings of the 3rd International Conference on eDemocracy and eGovernment (ICEDEG)*, Sangolqui, Ecuador, 30 March–1 April 2016; pp. 42–47.
7. Fairley, R.E.D.; Bourque, P.; Keppler, J. The impact of SWEBOK version 3 on software engineering education and training. In *Proceedings of the IEEE 27th Conference on Software Engineering Education and Training (CSEE&T)*, Klagenfurt, Austria, 23–25 April 2014; pp. 192–200.
8. Pyster, A.; Turner, R.; Henry, D.; Lasfer, K.; Bernstein, L. Master's degrees in software engineering: An analysis of 28 university programs. *IEEE Softw.* 2009, 26, 94–101.
9. Alain Abran; Juan José Cuadrado; Elena García-Barriocanal; Olavo Mendes; Salvador Sánchez-Alonso; Miguel A. Sicilia; Engineering the Ontology for the SWEBOK: Issues and Techniques. *Ontologies for Software Engineering and Software Technology* **2006**, null, 103-121, 10.1007/3-540-34518-3_3.
10. Patrick Klein; Dante Pugliese; Johannes Lützenberger; Giorgio Colombo; Klaus-Dieter Thoben; Exchange of Knowledge in Customized Product Development Processes. *Procedia CIRP* **2014**, 21, 99-104, 10.1016/j.procir.2014.03.149.
11. Hezhen Yang; J.F. Chen; N. Ma; D.Y. Wang; Implementation of knowledge-based engineering methodology in ship structural design. *Computer-Aided Design* **2012**, 44, 196-202, 10.1016/j.cad.2011.06.012.

12. Alexandra González-Eras; Pablo Quezada S.; Patricia Jeanneth Ludeña González; Carolina Gallardo; Comparing Competences on Academia and Occupational Contexts based on Similarity Measures. *11th International Conference on Web Information Systems and Technologies* **2015**, null, 540-546, 10.5220/0005491405400546.
13. C. Batini; M. Lenzerini; S. B. Navathe; A comparative analysis of methodologies for database schema integration. *ACM Computing Surveys* **1986**, 18, 323-364, 10.1145/27633.27634.
14. Art Pyster; Rick Adcock; Mark Ardis; Robert Cloutier; Devanandham Henry; Linda Laird; Harold 'Bud' Lawson; Michael Pennotti; Kevin Sullivan; Jon Wade; et al. Exploring the Relationship between Systems Engineering and Software Engineering. *Procedia Computer Science* **2015**, 44, 708-717, 10.1016/j.procs.2015.03.016.
15. Pierre Bourque; Vasile Stroian; Alain Abran; Proposed Concepts for a Tool for Multidimensional Performance Modeling in Software Engineering Management. *2006 IEEE International Symposium on Industrial Electronics* **2006**, 4, 3252-3257, 10.1109/isie.2006.296138.
16. Patrick Steyaert; Marco Barzman; Jean-Paul Billaud; Hélène Brives; Bernard Hubert; Guillaume Ollivier; Bénédicte Roche; The role of knowledge and research in facilitating social learning among stakeholders in natural resources management in the French Atlantic coastal wetlands. *Environmental Science & Policy* **2007**, 10, 537-550, 10.1016/j.envsci.2007.01.012.
17. Pierre Bourque; SWEBOK Refresh and Continuous Update: A Call for Feedback and Participation. *2009 22nd Conference on Software Engineering Education and Training* **2009**, null, 288-289, 10.1109/cseet.2009.61.
18. Peter Dolog; Lone Leth Thomsen; Bent Thomsen; Assessing Problem-Based Learning in a Software Engineering Curriculum Using Bloom's Taxonomy and the IEEE Software Engineering Body of Knowledge. *ACM Transactions on Computing Education* **2016**, 16, 1-41, 10.1145/2845091.
19. Ardis, M.; Bourque, P.; Hilburn, T.; Lasfer, K.; Lucero, S.; McDonald, J.; Shaw, M. Advancing Software engineering professional education. *IEEE Softw.* 2011, 28, 58–63.
20. Bourque, P.; Dupuis, R. Guide to the Software Engineering Body of Knowledge 2014 Version, SWEBOK. 2014. Available online: <https://www.computer.org/education/bodies-of-knowledge/software-engineering> (accessed on 18 March 2020).
21. Medical Group Management Association Englewood. Body of Knowledge for Medical Practice Management. 2017. Available online: <https://www.cmgma.org/acmpe/body-of-knowledge/> (accessed on 10 July 2020)
22. Mark Ardis; David Budgen; Gregory W. Hislop; Jeff Offutt; Mark Sebern; Willem Visser; SE 2014: Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering. *Computer* **2015**, 48, 106-109, 10.1109/mc.2015.345.

23. Bevan, N. Usability Body of Knowledge; Usability Professionals' Association: Bloomingdale, IL, USA, 2005.
24. Pomeroy-Huff, M.; Mullaney, J.L.; Cannon, R.; Sebern, M. Personal Software Process (PSP) Body of Knowledge, Version 1.0. Available online: <https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=7317> (accessed on 24 August 2020).
25. Masters, S.; Behrens, S.; Mogilensky, J.; Ryan, C. Scampi Lead Appraiser Body of Knowledge (SLA BOK); Software Engineering Institute, Carnegie Mellon University: Pittsburgh, PA, USA, 2005.
26. Dzimińska, M.; Fijalkowska, J.; Sułkowski, L. A Conceptual model proposal: Universities as culture change agents for sustainable development. *Sustainability* 2020, 12, 4635.
27. Sánchez-Arias, L.F.; Solarte-Pazos, L. The body of knowledge of the project management institute-PMBOK® guide, and the specificities of project management—A critical review. *Innovar* 2010, 20, 89–100.
28. Parnell, G.S.; Hardman, N.; McCarthy, D.J.; Yale, G.E. Using the guide to the systems engineering body of knowledge (sebok version 0.5) for undergraduate system engineering program assessment. *INCOSE Int. Symp.* 2012, 22, 2208–2220.
29. International Council on Systems Engineering. The INCOSE fellow's edition: The technical vision of systems engineering; the intellectual content of systems engineering. *INCOSE Insight* 2006, 8, 1–64.
30. Agresti, W.W. An IT body of knowledge: The key to an emerging profession. *IT Prof.* 2008, 10, 18–22.
31. Olwell, D.H.; Henry, D.; Pyster, A.; Hutchison, N.; Enck, S.; Anthony, J.F., Jr. Analysis of the references from the guide to the systems engineering body of knowledge (SEBoK). *Procedia Comput. Sci.* 2013, 16, 1000–1006.
32. MITRE. What is the Enterprise Architecture Body of Knowledge? Available online: <http://www.eabok.org/> (accessed on 2 July 2020)
33. Pierre Bourque; Robert Dupuis; Alain Abran; James W Moore; Leonard Tripp; Sybille Wolff; Fundamental principles of software engineering – a journey. *Journal of Systems and Software* **2002**, 62, 59-70, 10.1016/s0164-1212(01)00136-4.
34. Bourque, P.; Buglione, L.; Abran, A.; April, A. Bloom's taxonomy levels for three software engineer profiles. In Proceedings of the 11th Annual International Workshop on Software Technology and Engineering Practice (STEP 2003), Amsterdam, The Netherlands, 19–21 September 2003; pp. 123–129.

35. Bull, C.N.; Whittle, J. Supporting reflective practice in software engineering education through a studio-based approach. *IEEE Softw.* 2014, 31, 44–50.
36. Thomas, B.; Hilburn, I.; Hirmanpour, S.; Khajenoori, R.; Turner, R.; Abir, Q. A Software Engineering Body of Knowledge Version 1.0. Technical Report CMU/SEI-99-TR-004, ESC-TR-99-004ACM. 1999. Available online: https://resources.sei.cmu.edu/asset_files/TechnicalReport/1999_005_001_16733.pdf (accessed on 1 July 2020).
37. Abdulrahman Alarifi; Mohammad Zarour; Noura AlOmar; Ziyad AlShaikh; Mansour Alsaleh; SECDEP: Software engineering curricula development and evaluation process using SWEBOK. *Information and Software Technology* **2016**, 74, 114-126, 10.1016/j.infsof.2016.01.013.
38. Vahid Garousi; Gorkem Giray; Eray Tuzun; Understanding the Knowledge Gaps of Software Engineers. *ACM Transactions on Computing Education* **2020**, 20, 1-33, 10.1145/3360497.
39. Voas, J.; Kuhn, R.; Paulsen, C.; Schaffer, K. Computer Science Education in 2018. *IT Prof.* 2018, 20, 9–14.
40. Dieste, O.; Juristo, N.; Moreno, A.M. How higher-education systems influence software engineering degree programs. *IEEE Softw.* 2004, 21, 78–85.
41. Kilicay-Ergin, N.; Laplante, P.A. An online graduate requirement engineering course. *IEEE Trans. Educ.* 2013, 56, 208–216.
42. Maxville, V. eScience: Building our body of knowledge. *Procedia Comput. Sci.* 2011, 4, 1953–1963.
43. Caulfield, C.; Xia, J.C.; Veal, D.; Paul Maj, S. A Systematic survey of games used for software engineering education. *Mod. Appl. Sci.* 2011, 5, 28–43.
44. Ekaputra, F.J.; Serral, E.; Biffl, S. Building an empirical software engineering research knowledge base from heterogeneous data sources. In *Proceeding of the 14th International Conference on Knowledge Technologies and Data-driven Business*, Graz, Austria, 16–19 September 2014.

Retrieved from <https://encyclopedia.pub/entry/history/show/4395>