Software Engineering and Big Data Software

Nykolay Sydorov[1], *Nika Sydorova*[2]

[1] National Technical University of Ukraine Igor Sikorsky Kyiv Polytechnic Institute, street Politechnichna, 41. Kyiv, 02000, Ukraine
[2] VTI Private Joint Stock Company, street F.Pushynoi, 30/32, Kyiv, 03115, Ukraine

Abstract

The software engineering is a mature industry of human activity focused on the creation, deployment, marketing and maintenance of software. The life cycle model, three components of life cycle phases - products, processes and resources and engineering and methodologies for creating, deployment and maintaining software are the fundamental concepts of the software engineering. A software is the foundation of technological advances that lead to new high performance products. In addition, the new products of technology are usually the software-intensive products. The functionality of the products determines the scale and complexity of the software. As the functionality of products grows, so does the need to efficiently and correctly create and maintain the complex software that corresponds to this growth. Therefore, in addition to solving its own problems, software engineering serves the solution of the problems of creating and maintaining software in other domains, which are called application domain. The information technology and the computer science are a well-known application domains for the software engineering. The basis of the information technology domain is data. The information technology are being implemented in an organization to improve its effectiveness and efficiency. The base component of the information technology is an information system. The functionality of information systems for has grown dramatically when big data began to be used. This growth has led to the emergence of a wide variety of software-intensive big data information systems. At the same time, the role and importance of software engineering for solving the problems of this application domain has only intensified. Modern possibilities of the software engineering are shown. The aspects of interaction between the software engineering and the big data systems are analyzed. The topics for the study of a software in the big data ecosystems and the software of big data system of systems are outlined.

Keywords

Big data, software engineering, big data software, big data ecosystem, big data system of systems

1. Introduction

The software engineering is a mature industry of human activity focused on the creation, deployment, marketing and maintenance of software. The life cycle model, three components of life cycle phases - products, processes and resources and engineering and methodologies for creating, deployment and maintaining software are the fundamental concepts of the software engineering. A software is the foundation of technological advances that lead to new high performance products. In addition, the new products of technology are usually the software-intensive products. The functionality of the products determines the scale and complexity of the software. As the functionality of products grows, so does the need to efficiently and correctly create and maintain the complex software that corresponds to this growth. Therefore,

13th International Scientific and Practical Conference from Programming UkrPROGP'2022, October 11-12, 2022, Kyiv, Ukraine
EMAIL: NykSydorov@gmail.com (A. 1); nika.sidorova@gmail.com (A. 2)
ORCID: 0000-0002-3794-780X (A. 1); 0000-0002-2989-3637 (A. 2);

in addition to solving its own problems, software engineering serves the solution of the problems of creating and maintaining software in other domains, which are called application domain. The information technology and the computer science are a well-known application domains for the software engineering. The basis of the information technology domain is data. The information technology are being implemented in an organization to improve its effectiveness and efficiency. The base component of the information technology is an information system. The functionality of information systems for has grown dramatically when big data began to be used. This growth has led to the emergence of a wide variety of software-intensive big data information systems. At the same time, the role and importance of software engineering for solving the problems of this application domain has only intensified.

In paper, modern possibilities of the software engineering are shown. The aspects of interaction between the software engineering and the big data software are analyzed. The topics for the study of a software in the big data ecosystems and the software of big data system of systems are outlined.

2. The software engineering

The software engineering, having appeared more than fifty years ago, is now rather mature industry of human activity aimed at creating, deployment, marketing and maintaining software [1]. The credo of software engineers is expressed in the view of software products as reality objects, which are created, act, are changed and destroyed by the implementation of appropriate processes. And, these processes can be controlled and they can be managed. This, successively, led to the concept of modular software design, starting with the method of structured programming, continuing in block-oriented, modular and object-oriented programming languages, and ending with a component approach to software development. This also led to modern the software life cycle models, when two phases of the cod-and-fix model were replaced in the waterfall model on real-life processes: vertical processes - domain analysis, requirements specification, and design, construction, testing and debugging; horizontal processes - management, quality assurance, verification and validation, documentation [2]. This contributed to the emergence of new forms of software products, methods, tools and professions. The result of solving a software engineering problem is always a product that has a customer. The software engineering product is created in the context of a life cycle and must meet the requirements specific to any engineering (product construction, quality, standards, documentation, maintenance, environmental impact and economics). In this regard, in the software engineering there is a number of types of the engineering (Table 1).

Table 1
The types of engineering

№	Engineering	Goals
1	The forward engineering	Creating software products
2	The backward (reverse) engineering	Extract design and implementation information about legacy products
3	The reengineering	Changing software products (repairs, reworking, reusing)
4	The search-based engineering	Finding COTS, the reusable components and the other parts of software
5	The empirical engineering	Researching software products
6	The value based engineering	Integrating value considerations into the full range of existing and emerging software engineering principles and practices.

The software culture and related tools, for example, CMMI, P-CMM, etic code are also important [3]. The impact on the environment is studied in the context of the concept of the sustainable development in the form of lean and green software development, as well as the software reusable development. The evolution of products is explored in the form of a product line. The interaction and mutual influence of products, as well as software assets and artefacts, is explored in the form of software ecosystems and software artefacts ecosystems [4]. In the context of globalization, the software engineering products are implemented and researched as a system of systems. The software engineering products and the processes for creating and maintaining them are costly. There is software economics with effective methods, models and tools for the cost estimating of software development [5, 6].

The Software Engineering Body of Knowledge (SWEBOK) and the Curriculum Guidelines have been developed for undergraduate and graduate education of the software engineers [7]. Training of the software engineers for the application domains is carried out at master's postgraduate education. For example, postgraduate education for big data software development [8, 9].

1. The software engineering domain and the big data domain – decision of problems

Consider interrelation between the software engineering and big data application domain in aspect decision of problems in big data software development. Usually, the interaction of the software engineering with an application domain is carried out by the "problem-solution" principle [10]. In the application domain, a problem is formulated to be solved, and in the software engineering domain, a solution is sought. If there is none solution, a new one is created (Figure 1).

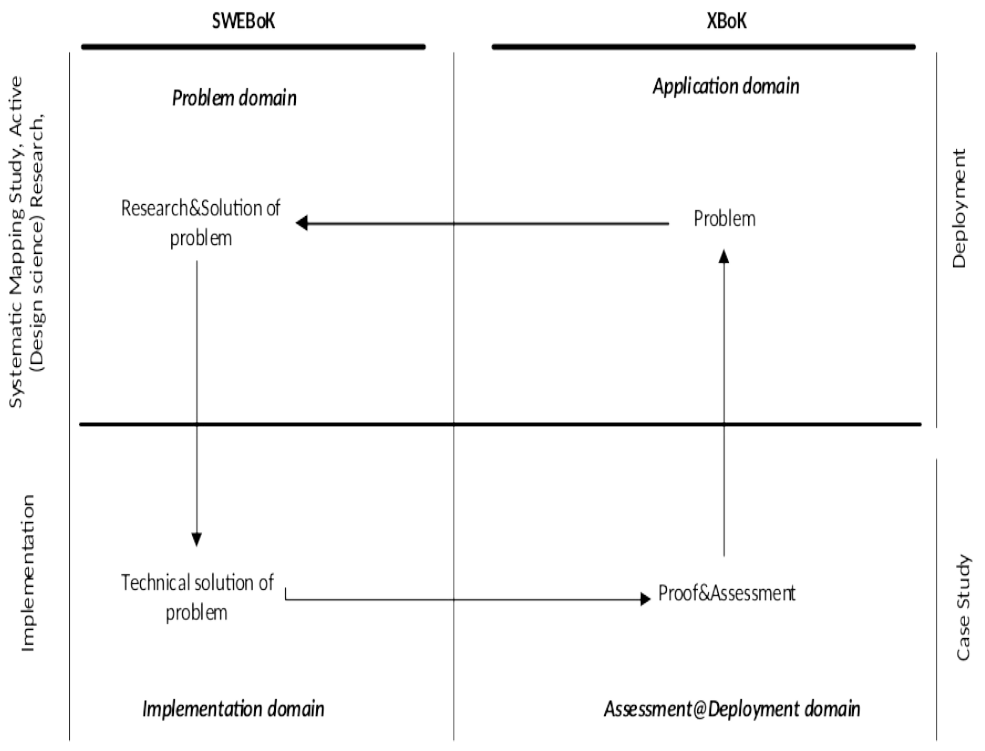

Figure 1. The "problem-solution" principle implementation

In more detail, the solution of the application domain problem should be considered in the context of the following four domains:

- Application domain, for which the problem is solved and (or) software (software product) is created - for example, this is the application domain "information technology" (represented by the relevant XBoK knowledge, Figure 1). For example, for information systems and technologies it is Core Body of Knowledge for ICT Professionals [11];

- Problem domain, in that new solutions are created or existing solutions are improved. This is the scientific part of the software engineering domain (represented by SWEBoK [7]);

- Implementation domain, in that the software (software product) is created according to the solution. This is the engineering part of the software engineering domain (represented by SWEBoK);

- Domain of evaluation and implementation of the results, in which individual properties of the results obtained are checked and the issues of their implementation are examined. This is the practical part of the software engineering domain.

Nowadays, there is no separate Body of Knowledge for big data domain (XBoK, Figure 1). Therefore, knowledge regarding big data is located in several documents, for example, in the following [12 - 15]. With respect to this, the following can be said, either the big data is not considered as a separate domain as a result of its immaturity, or it is inappropriate to consider it as a separate domain. At the same time, in [16] proposes the task of creating big data body knowledge as a basis for developing a core curriculum. This is the accepted practice. If the application domain is software-intensive, then postgraduate education programs for application domain specialists can be created to interact with specialists in the software engineering domain. Usually, these are master's programs of universities. If the request for education of specialists in the applied domain is sustainable in demand, then the Common Body of Knowledge can be created as a basis for a consistent curriculum development.

Thus, the big data domain, for software engineering is similar to other application domains, nothing more.

2. The software engineering and big data software - interaction

However, from the point of view of the software engineering, it is impossible not to pay attention to the diversity of the big data technologies and, consequently, the diversity of software of big data systems. This diversity is created due to the presence of characteristics of big data such as Variety, Volatility, and Value. The diversity of the big data systems provides the presence of the big data software ecosystems. In addition, a variety of the big data systems, which are used to achieve an emergent goal, can be a form of the big data system of systems. Ecosystems of this type and big data system of systems will be interesting for software engineering researchers.

The article analyzes aspects of the interaction between the software engineering and the big data software.

2.1The software engineering and challenges of big data software

Guided by the “problem-solution” principle (Figure 1), the big data researchers formulate the problem of a big data software. The problem (Figure 1) can be formulated for different domains, such as those represented by the following Body of Knowledge - DMBOK, BABOK, PMI-BOK, SWEBOK, ACM BoK. The big data challenges are associated with big characteristics (10V) [17]. At the same time, each of the 10V puts different requirements for different the big data systems. Experts and research indicate, that not all big characteristics are inherent in every big data software [18]. On the contrary, only some require support, and that this depends on the type of domain. According to the study [19], in the largest number (7 V's) are need of Retail & Oil and Gas domains. Other domains are distributed the following: (5 V's) - Agriculture, Manufacturing, Health Care, Law enforcement & Education; (4 V's) - Waste Management; (3 V's) - Telecommunication. If the subject it is from the software engineering (SWEBOK), then the problem is formulated for the software engineering problem domain (Figure 1). When the problem is solving, the challenges of big data software should be taken into account.

For software engineering, the followings problems arising in connection with this, require additional study. The processes the software architecture design, the software quality assurance are less studied [20]. In [21], the problems of implementing the project and requirements management, deployment and operations processes are added to these problems. In [10], the problems of using data-centric software development to implement the big data analytics software and the problems of testing and debugging for the big data analytics software are formulated. The paper [22] analyzes the possibilities of the software engineering for the implementation of the big data systems. In [23, 24], the problems of the developing systems in the application domains with big data, for example, military, aviation industry, banking and financial industry are considered. The main challenges the big data for software engineering in context of the software life cycle phases look in Table 1.

Table 1

The main Challenges the big data for software engineering

№	SLCF	Challenges	Paper
1	Requirements	Integrating Big Data characteristics in requirements (Big Data scenarios, Data specific requirements), Highly Interdisciplinary Teams	[25], [26]
2	Architecture design	Providing a scalable storage and processing of massive datasets by the distribution and parallel processing of data using cloud-based Big Data technologies, automatic resiliency, computation intensive analytics.	[25], [18], [27]
3.	Quality assurance	It impossible to comprehensively test every aspect of Big Data Software at scale before deployment to production. The assessment of data quality, complex data processing and different notions of correctness.	[25]
4.	Testing	Data quality testing, testing of machine learning algorithms	[21]
5.	Deployment & Operations	Deployed to Cloud environment, complex monitoring of the operations, Long-term operation.	[21]

4.2 The big data and challenges of software engineering

The methods and tools used in the big data domain can be used in solving software engineering challenges and main challenges already are studying (Table 2). Using the big data analytics for the growing software projects, the data mining for studying software repositories, and the visual analytics for knowledge discovery are considered [28]. The application of the machine learning algorithms for software analysis [29] and the artificial intelligence in difference phases of the software life cycle are considered [30].

Table 2

The main Challenges of the software engineering

№	Challenges	Decisions	Paper
1.	Software projects grow in size and complexity	The big data analysis methods	[31]
2.	Reuse of components (recommendation systems)	Datamining methods	[32]
3.	The knowledge discovery (software understanding for maintenance end evolution, software visualization, decision-making on human resources)	Visual analytics methods	[33]

5. The challenges for software engineering researchers

For software engineering researchers, new themes from big data application domains are likely to be software of big data ecosystems and software of big data system of systems.

5.1 The big data system of systems

The system of systems is set of systems networked together in a large-scale integrated system. The systems are heterogeneous and independently operable on its own, but is networked together for a common goal. [34]. Modern system of systems is characterized in terms of five main properties, sometimes referred to as the acronym "OMGEE" [35]: Operational Independence, Managerial Independence, Geographic Distribution, Evolutionary Development, and Emergent Behavior.

The system of systems concept is used for big data system [36]. If we consider that each stage of pipeline is implemented by the system, then the implementation of the entire data processing pipeline will be the system of systems (Figure 2).

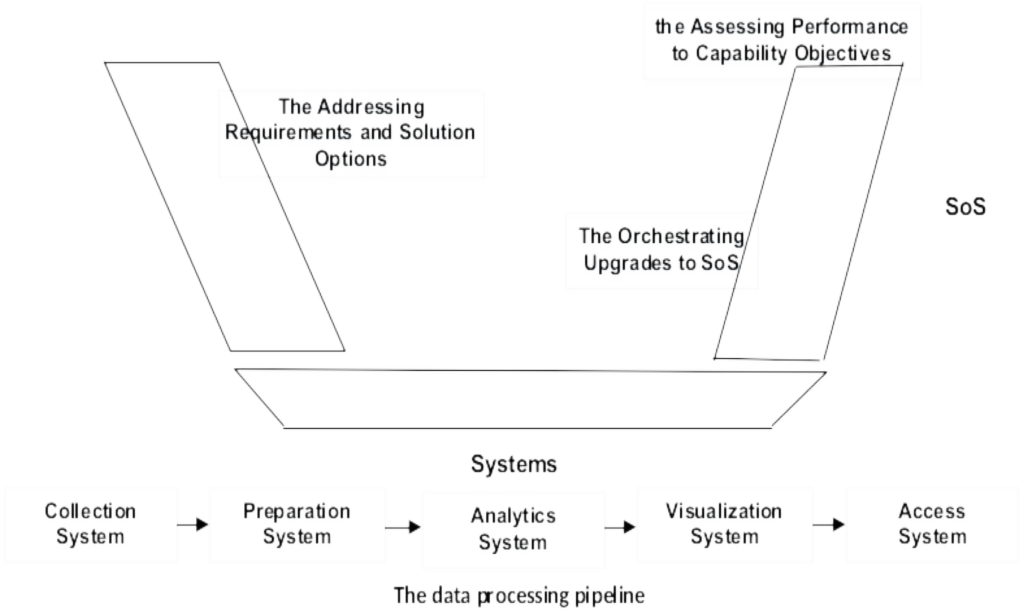

Figure 2. The big data system of systems SE with a focus on upgrade

We will distinguish between the system of systems core and the system of systems environment (context [37]).

The system of systems core can be consists of the seven core elements of the system of systems SE and provides the context for applying systems engineering processes. On Figure 2 shows three core elements from the V model that focus on system of systems upgrades [38].

The system of systems environment consists of a set of functional systems. On Figure 2, the system of systems environment is represented by systems of big data processing pipeline.

The composition of functional systems of the big data system of systems environment may vary and depends on the lead system integrator (systems engineer) – "Orchestrating Upgrades to system of systems" element (Figure 2).

One functional system will definitely be included in the big data system of systems environment - big Data Management System (Platform). It provides the implementation of the metadata at the physical and conceptual levels [39]. The big Data Management System, creates a common big data space for functional systems included in the big data system of systems (data collection, classification, indexing, storage, search, transform functions, representing big data).

The big data system of systems environment may include an analytics (modeling, prediction) system that performs analytics operations of two categories of analysis - particularly direct analysis and exploratory analysis which requires real time response [40]; presentation system performing data delivery, visualization; situation awareness system, which involves collecting, aggregating, and interpreting information in order to know what is happening in the environment. In addition, the system of systems environment may include big data systems of application domains, for example, banking and financial services, energy, healthcare, media, and education among others. In this case, the platform must provide an application layer of data presentation.

With a focus on system of systems upgrade the "Orchestrating Upgrades to system of systems» element is used. This element is used with additional elements from the core system of systems. This are the "Addressing Requirements and Solution Options" element and the "Assessing Performance to Capability Objectives" element (Figure 2). These three elements together provide the application of systems engineering processes for developing of new systems and continuation to support system of systems.

Inputs to "Addressing Requirements and Solution Options" element include [38]:

- Current system of systems architecture and associated constraints, opportunities for changes and options;
- Expected impacts of changes on system of systems, including planned individual system changes;

- Problems/issues associated with implementation of previous system of systems updates.

Outputs of this core element to other system of systems SE core elements are identification of requirements together with an approach for implementing those requirements. The implementing new requirements may be:

- Adding new systems;
- Updating or extending functionality of existing systems;

The result of "Addressing Requirements and Solution Options" element is a technical plan that triggers orchestration of new system of systems upgrades [38].

The “Assessing Performance to Capability Objectives” element receives inputs on goals and objectives, and on system of systems changes that to affect the performance. In the assessment is used metrics, based on receives inputs. The output of the assessments provides feedback to the systems engineer of system of systems on the accomplishment and feasibility of the capability objectives.

The “Orchestrating Upgrades to system of systems” element coordinates the changes being implemented in the system of systems to effect performance improvements and added capability.

5.1.2 The software of big data systems as software of system of systems

All systems in the system of systems environment of big data system of systems are software-intensive systems. Therefore, software plays an important role in big data system of systems and is often the most complex part of the system. Software system engineering and software-intensive systems engineering are used to create and maintain the software of system of systems environment [41]. Ignoring the systems aspects of software of system of systems environment can result in software that will not integrate with other software systems of system of systems environment.

To create and maintain a system of systems, system engineering is used, which is different from the traditional one [41]. Doing the system engineering processes does not lead to software, but obviously requires the use of software-aided systems.

5.2 The big data ecosystems

The data ecosystems are considered in [42]. The transition from a data ecosystem to the big data ecosystems is considered in [39] and the big data architecture framework and big data lifecycle in big data ecosystem are proposed. In [43], the actors of such ecosystem are introduced, and the modules of the big data ecosystems are introduced in [40].

The modules that formed the big data ecosystems are software-intensive systems and must be considered as software of big data ecosystems.

5.2.1 The big data software ecosystems

The big data software ecosystems will be heterogeneous both in terms of actors, roles and strategies, as well as products and platforms. This is a consequence of the presence of such in the big data software ecosystem is involved making use of a wide variety of technologies and tools, such as data management - acquisition, storage, management and retrieval of data; data processing - text analytics, business intelligence, data visualization and statistical analysis, predictive analysis. This will entail the presence of different actors, data products and platforms in the ecosystems. In software ecosystems, the variety of these entities is usually limited of the software engineering territory [44]. In the big data software ecosystem, there is no such restriction. Conversely, diversity is a consequence of using solutions from different domains. Problems of big data system of systems arise with owners of a big data can also influence on big data software ecosystems.

The data processing pipeline in a typical big data system of systems consists of activities united in stages and is fulfilled by own systems of the stages. (Figure 2). For example, Collection stage, consists of Data ingestion, Data loading and preprocessing activities; Preparation includes Information extraction, Data cleaning, big data integration; Analytics consists of Data analysis, Data loading and transformation; Visualization includes Data visualization [25]. Taking into account the concept information (data) value chain it can be argued that each stage forms a new version of the data product, which can be sold or used as input for the next stage. The activities of stage are fulfilled on own landscape, the boundaries of which can be considered as the boundaries of the corresponding software ecosystem (Figure 3) [44].

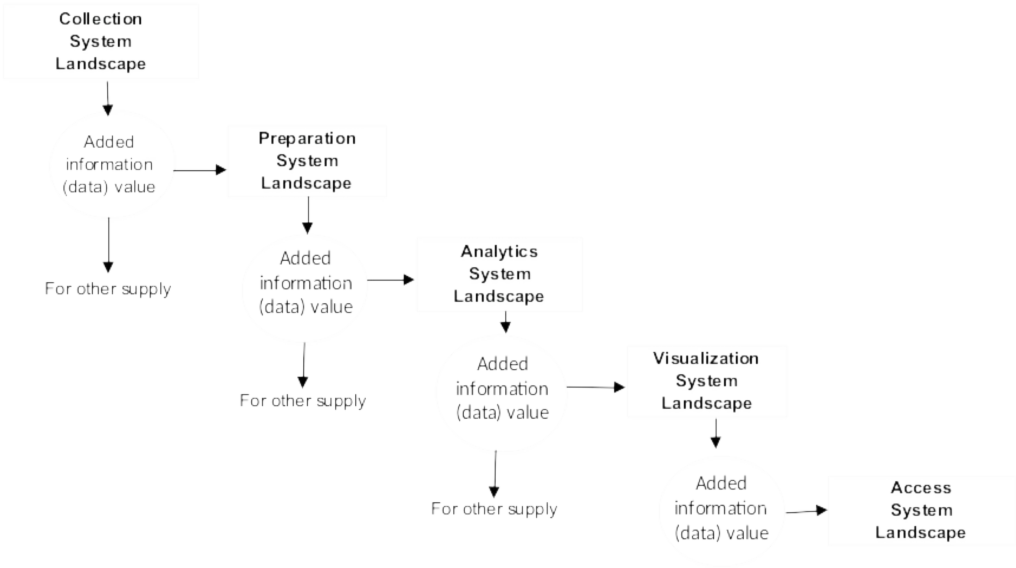

Figure 3. The data (information) value chain

Individual stages may be skipped or repeated, if necessary. They are independent set of activities, each with the goal to form data product and sell it or as input in next stage. At the same time, each stage is filled with a wide range of different supporting functionalities, and as a result different software-intensive tools and actors. Different actors (e.g. data producers, data consumers) can participate by executing one or more activities (e.g. Data Discovery, Data Exploitation), and each activity can consist of a number of actions or value creation techniques, (e.g. Gathering, Visualization, Service Creation). In turn, each action can consist of one or more data value chains, since they might need a series of processes to be executed in order (e.g. visualization requires identifying the data to visualize, then deciding on a visualization method, then rendering the visualization). This leads to a clearly defined landscape and allows to research each stages of the data processing pipeline in the big data system of systems as an ecosystem. At the same time, the role orchestrator – ensure the exchange and sharing of data in the context of ecosystems in order to achieve additional value of data. The data exchange focuses on cooperation between the systems along value chain. For this, standards are used. Data sharing is focusing on generating additional value out of data along value chain. For this, a mode of collaboration is used.

Big data can significantly change both approaches to research and the architecture of software ecosystems by applying the solutions given in Table 2.

3. Conclusion

The big data systems are software-intensive systems. Therefore, to create and maintain software of big data systems, it is advisable to use the methods and tools of software engineering. The goal of paper is to show, that software of big data systems for software engineering are the application domain, nothing more. The article shows the modern possibilities of software engineering and the interconnection of software engineering and big data software. On the one hand, it is considered which challenges of big data software are already being solved by applying the methods of software engineering, and on the other hand, which challenges of software engineering can be solved by applying the methods of processing of big data. This will be useful for both application domain researchers and software engineers. Finally, we have indicated new topics for software engineering researchers. These are the software of big data system as a software of system of systems and the big data software ecosystems. Those and others may be objects of future research.

6. References

[1] Report on a conference sponsored by the NATO science committee, Editors: P. Naur and B. Randell, Garmisch, Germany, 7th to 11th October 1968

[2] M.O. Sidorov Software engineering, lecture curse. - Kiev.: National Aviation University, 2007.-140p

[3] K Wiegers, Creating a Software Engineering Culture, Dorset House, 2011, 384 p

[4] N Sydorov, Programming Style as an Artefact of a Software Artefacts Ecosystem, in: Proceedings of the Advances in Computer Science for Engineering and Education IV, 2021, 244–255

[5] B. Boehm, Software Cost Estimation with COCOMO II, Prentice Hall PTR, 2000.

[6] B. Royce, The Economics of Software Development, Addison-Wesley, 2009.

[7] P. Bourque, R.E. Fairley, eds., Guide to the Software Engineering Body of Knowledge, ver. 3.0, IEEE CS, 2014; URL: www.swebok.org.

[8] URL: https://www.ul.ie/gps/course/software-engineering-msc

[9] URL: https://www.city.ac.uk/prospective-students/courses/postgraduate/software-engineering

[10] M.O. Sydorov Master's Thesis in Software Engineering - Object, Subject, Contents of Research, J.: Problems in programming, №2: 2022; 22-36

[11] The ACS Core Body of Knowledge for ICT Professionals (CBOK), Australian Computer Society Inc, Sydney NSW 2000

[12] NIST Big Data Interoperability Framework: Volume 1, Definitions Final Version 1, NIST Big Data Public Working Group (NBD-PWG) Definitions and Taxonomies Subgroup Information Technology Laboratory, September 2015

[13] A Guide to the Data Science Body of Knowledge – Version 2, IABAC™, 2019

[14] The European Foundational ICT Body of Knowledge Version 1.0 22 February 2015

[15] ISO/IEC JTC 1, Big data, Preliminary Report 2014

[16] A, Hamid, A Functional View of Big Data Ecosystem, International Journal of Computer Trends and Technology (IJCTT) – Volume 68, Issue 4 – April 2020, 233-237

[17] S. Zhaohao, S. Kenneth, L. Rongping, 10 Bigs: Big Data and Its Ten Big Characteristics, BAIS No. 17010, PNG University of Technology, 2018, P15

[18] Three Experts on Big Data Engineering, IEEE Software, March/April 2016, 68-72

[19] S. Nagham, H. Laden, Big Data Characteristics (V's) in Industry,
Iraqi Journal of Industrial Research, Vol. 8, No. 1 (2021)

[20] A Davoudian, M. Liu, Big Data Systems: A Software Engineering Perspective, ACM Computing Surveys, Vol. 1, No. 1, 2019, P 1-43

[21] O. Hummel, H Eichelberger, A Collection of Software Engineering Challenges for Big Data System Development, in: Proceedings of Euromicro SEAA 2018, Prague, 2018

[22] M Kim, Software Engineering for Data Analytics, IEEE Software, July/august 2020, 36-42

[23] S. Punia, M. Kumar, Domains and methodologies for big data project in software engineering, International Journal of Scientific & Engineering Research, Volume 8, Issue 8-2017, 1525-1535

[24] V.D. Kumar, P. Alencar, Software Engineering for Big Data Projects: Domains, Methodologies and Gaps, in: Proceedings of IEEE International Conference on Big Data, 5-8 Dec. 2016

[25] A. Davoudian, M. Liu, Big Data Systems: A Software Engineering Perspective, ACM Computing Surveys, Vol. 1, No. 1, January 2019.

[26] M. Kim, Software Engineering for Data Analytics, July/august, IEEE Software, 2020, 36-42
[27] I. Gorton, A. Bener, A. Mockus, Software Engineering for Big Data Systems, march/april IEEE Software, 2016, 32-35
[28] T. Arndt, Big Data and software engineering: prospects for mutual enrichment, Iran Journal of Computer Science, Springer International Publishing AG, 2017
[29] K Meinke, A. Bennaceur, Machine Learning for Software Engineering, in: Proceedings of ICSE '18 Companion, May 27-June 3, Gothenburg, Sweden, 2018
[30] M. Barenkamp, J Rebstadt, O. Thomas, Applications of AI in classical software engineering, https://doi.org/10.1186/s42467-020-00005-4
[31] P.L. Raymond, T. Zimmermann, Information Needs for Software Development Analytics, Microsoft Research. Technical Report MSR-TR-2011-8. 2011, 16
[32] F. Ricci, L. Rokach, B. Shapira. (eds.) Recommender Systems Handbook, pp. 1–35. Springer (2011). DOI 10.1007/978-0-387-85820-3 1
[33] R. Pressman,. Software engineering: a practitioner's approach / Roger S. Pressman. — 7th ed., Published by McGraw-Hill, 2010, 930p
[34] Systems of systems engineering: innovations for the 21st century / edited by Mo. Jamshidi. John Wiley & Sons, 2009, p 588
[35] M. W. Maier, Architecting Principles for Systems-of-Systems. in: Proceedings of INCOSE 1996 Sixth annual International Symposium of the International Council on Systems Engineering. INCOSE, 1996
[36] K. Barnabas. M. Jamshidi, System of Systems and Big Data analytics – Bridging the gap, Computers and Electrical Engineering 40 (2014) 2–15
[37] R.L. Ackoff, "Towards a System of Systems Concepts", Management Science, v.17:11, July 1971
[38] Systems Engineering Guide for Systems of Systems, Version 1.0. Washington, DC: ODUSD (A&T) SSE, 2008.
[39] Y. Demchenko, P. Membrey, Defining Architecture Components of the Big Data Ecosystem, The proceedings of International Symposium on Collaborative Technologies and Systems, 19-23 May 2014, 1-3
[40] A. Hamid, A Functional View of Big Data Ecosystem, International Journal of Computer Trends and Technology (IJCTT) – Volume 68 Issue 4 – April 2020, P 233-237
[41] R. H. Thayer, Software System Engineering: A Tutorial, Computer, April 2002, 68-73
[42] Data Ecosystems, ISST-Report, Editors Prof. Dr.-Ing. B. Otto, Prof. Dr. J. Rehof, Germany, October 2019
[43] B. Rad, P. Ataei, The big data Ecosystem and its Environs, IJCSNS International Journal of Computer Science and Network Security, VOL.17 No.3, March 2017
[44] N. Sydorov, Toward Software Engineering Ecosystems Definition, J.: Problems in programming, №3, 4: 2022