ISSN 1407-7493 COMPUTER SCIENCE 2008-34
DATORZINĀTNE

APPLIED COMPUTER SYSTEMS
LIETIŠĶĀS DATORSISTĒMAS

AN OPEN WORK ON RESEARCH METHOD IN THE FIELD OF SYSTEM ENGINEERING: THE BACHELOR LEVEL

Oksana Nikiforova, *Riga Technical University, Kaļķu 1, Rīga, LV-1658, Latvia, asoc.prof., Dr.sc.ing., oksana.nikiforova@cs.rtu.lv*

Mārīte Kirikova, *Riga Technical University, Kaļķu 1, Rīga, LV-1658, Latvia, asoc.prof., Dr.sc.ing., marite.kirikova@cs.rtu.lv*

Renāte Strazdiņa, *Riga Technical University, Kaļķu 1, Rīga, LV-1658, Latvia, researcher, Dr.sc.ing., renate.strazdina@ lv.pwc.com*

Research methodology, software engineering versus scientific method, bachelor level research

1. Introduction

Strong traditions and requirements for the process of scientific research exist in the fundamental fields of science such as physics, mathematics, chemistry. The same applies to social and economic fields of science. Since the emergence of science and first scientists' research in the ancient days [1] different guidelines and methodologies have been developed for application of scientific methods as well as for their selection for certain types of scientific solutions development. However, this does not apply fully to the research in the area of system engineering where systems are understood as software or information systems.

Software engineering is an engineering discipline concerned with the problem of system development, where *system* is understood as a software system. Software engineering is nor just programming nor just computer science [2]. From research perspective software engineering is a relatively new and heterogeneous field of science, where generally accepted norms for application of scientific methods are yet not established. It is quite enough complicatedly to define a boundary between the required level of problem formalization and simplicity of problem solution. In addition to that, research in the field of system engineering in general and in software engineering in particular requires to take into consideration not only technological aspects of problem solutions, but also to be concerned with economical and social aspects of application domain. There are many and varied classifications of sciences [3], based on different criteria. Depending on the kind of science, different research methods are used. None of these methods, however, seems to be fully suited for scientific research in the field of Software Engineering. As it is proposed in [4], the branches of engineering do not fit perfectly into the classifications proposed in the literature, although they are related with most of the disciplines appearing in them. For this reason, the search for the method appropriate to research in the field of software engineering is becoming a research field itself [4], [5], [6], [7].

The problem stated above is particularly relevant for students first scientific research, which in the most general case is the development of a bachelor thesis. During the bachelor thesis development problems often arise an understanding whether the scope of the research, its level of detail and suggestions on formal foundations for problem solutions are adequate to the requirements for a scientific thesis. The following problems are identified in bachelor thesis development:

1) disorientation in the collection of scientific methods and inability to select the required method for certain task solution;

2) weak skills in usage of bibliography references and appropriate literature search in scientific data bases as well as further analysis and application of such data;

3) lack of understanding of what is scientific research and how to contribute to it.

Normative acts of the Riga Technical University define bachelor thesis as “student research presenting his / her abilities to use acquired knowledge, to make analysis of a problem and its solutions presented in related bibliography, and to propose a solution to the problem based on analytical results”. But it is not enough to understand the process of scientific research especially in such an extraordinary field of science as system engineering.

The paper discusses several issues relevant to development of bachelor theses. The first issue we discuss is the similarity between the process of research as such and system development process in system engineering. We deal with this issue in section 2. The second issue concerns a variety of options in choosing bachelor thesis topics and their positioning in the map of possible topics, which is introduced in Section 3. In section 3 we also discuss how different software development paradigms and methods for system development can shape the research done by bachelor level students in their bachelor thesis development. Section 4 consists of some concluding statements concerning the topic discussed in the paper.

2. Mapping of Software Development Activities into Research Life Cycle

Research methodology based on scientific method application is a process for making observations, recording data, and analyzing data that can be duplicated by other scientists. In 1637 René Descartes published his "Discours de la Méthode" in which he described systematic rules for determining what is true, thereby establishing the principles of the scientific method [8]. The simple version of it looks as shown in Figure 1a):

1) observe an aspect of the universe;

2) invent a tentative description, called a hypothesis, that is consistent with what has been observed;

3) use the hypothesis to make predictions;

4) test predictions by experiments or further observations and modify the hypothesis in light of the results;

5) repeat steps 3 and 4 until there are no discrepancies between theory and experiment and/or observation. [8].

When consistency is obtained the hypothesis becomes a theory, which is then a framework within which observations are explained and predictions are made [9].

For the first scientific research usually it is difficult to understand what is a hypothesis, how to advance it, how to predict its consistency, and so on. But taking in to consideration students high enough skills in system lifecycle implementation for system development it is possible to describe steps of scientific methods in terms understandable to students.

Marcos in [10] assumes that the object of study of the branches of engineering (and software engineering in particular) differs from the object of study of the formal and empirical science (be they sociological or natural). While these subjects deal with the study of existing phenomena or objects, engineering sciences deal with the study of the methods and techniques necessary for the creation of new objects and even with the creation of such methods and techniques. Therefore it has important similarities with the application of software engineering, which is also concerned with creation (of software products). Thus, it is possible to establish an analogy between the process of research in the field of software engineering and process of software development [10].

Every software development effort goes through a lifecycle, a process that includes all activities in the development cycle that take place up to initial release. The main function of a lifecycle model is to establish the order in which a project specifies, implements, tests, and performs its activities.

Software development is a process for organized production of software, using a collection of predefined techniques and notational conventions [11]. In general, software development life cycle defines activities for software development and in spite of difference between life cycle models, that can be presented as waterfall [12], spiral [13], fountain [14], as well as its advanced combinations in Rational Unified Process [15] and Microsoft Solution Framework [16], it is possible to identify the following stages of software development: analysis, design, implementation and testing [17] (shown in Figure 1) [18].

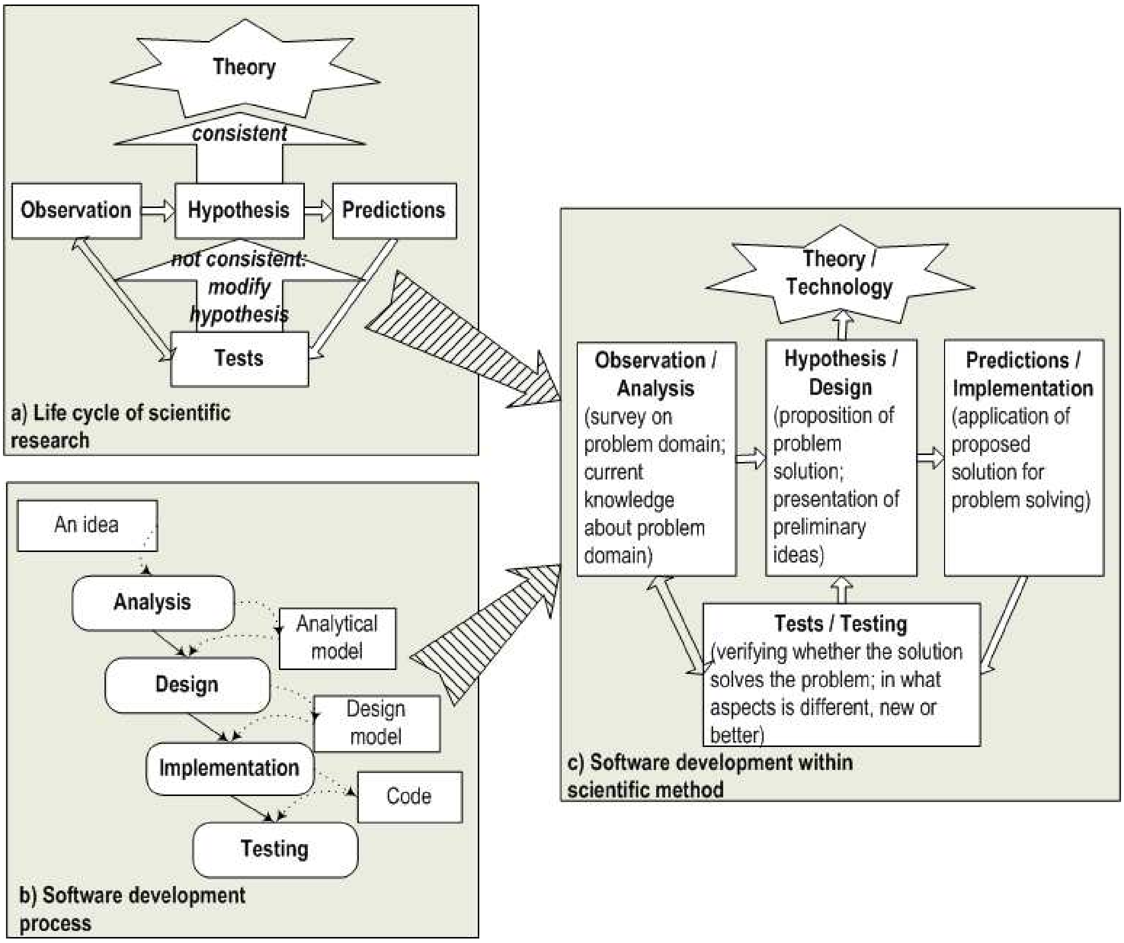


Figure 1. Mapping of stages for system development into schema of scientific research method

Usually, system development starts with an idea. System analysis includes careful acquisition and examination of the requirements for a system with the intent of understanding them, exploring their implications, and removing inconsistencies and omissions. System design presents overall system architecture. During system design the target system is organized into components based on both the analysis structure and the oncoming architecture. The end product of analysis and design is a system representation that corresponds to the requirements and is used for further system implementation. Testing is applied for implemented system verification and validation according the preliminary requirements. Therefore from one side taking as a basis the schema of scientific method presented in Figure 1b) and applying general phases of system development shown in Figure 1a) from the other side a composite schema for research methodology can be presented as shown in Figure 1c).

Aspect observation can be understood as problem domain analysis and knowledge acquisition about the “science needs” pertinent at the moment. Proposition of problem solution, which is better understood as solution architecture development, can serve as the setting of the hypothesis. Acquirements for application of the proposed solution or realization of the developed architecture can help in understanding what should be done during scientific research predictions.

Verifying whether the solution solves the problem can be based to some extent on methods applied for system testing as far as mathematical data analysis methods can be applied for discussing of what aspects are better, different or new in the proposed problem solution.

3. Variety of Options in Choosing Bachelor Thesis Scope

The result of bachelor level research is a Bachelor thesis. Analysis of bachelor theses topics at the Institute of Applied Computer Systems of the Riga Technical University has revealed a wide variety of topics considerably differing in depth and breadth of subjects with respect to both software development phases and research life cycle. However, there was a clear tendency to keep a balance between the student's contribution to research as such and his/her contribution to systems engineering. A variety of bachelor thesis topics is illustrated in Table 1 where a cross screening of systems development and scientific research processes is given. Rows of Table 1 correspond to the phases of the simplified research process [19].

Table 1: Cross screening of software development phases via scientific research life cycle.

Software development phases / ***Scientific research life cycle***	**1. Analysis**	**2. Design**	**3. Implementation**	**4. Testing**
1. Observation	1.1.	1.2.	1.3.	1.4.
2. Hypothesis	2.1.	2.2.	2.3.	2.4.
3. Predictions	3.1.	3.2.	3.3.	3.4.
4. Tests	4.1.	4.2.	4.3.	4.4.

Columns of Table 1 correspond to the phases of the simplified software engineering life cycle. Despite both processes having considerable similarities (Figure 1) there are also differences with respect to the products to be delivered by activities in each life cycle phase [10].

It is also important to take into consideration that a bachelor thesis is not a PhD thesis and is defined as "analytical research with scientific elements", therefore the student is not requested to handle the entire life cycle of scientific research. While such an option is possible for simple systems engineering problems, in more complex problems the student will most probably perform his / her research work only according to the part of the research life cycle. In these complex tasks, however, it is necessary for the student to understand how his / her activities are placed in the research life cycle and what his / her contribution to the research is. The same principle applies to the systems engineering dimension, i.e. systems development phases.

The diagonal of Table 1 actually corresponds to Figure 1 and is an option where performance with respect to both life cycles would be nearly identical. It should be noted here that in this situation every systems engineering task will not qualify as scientific research because the solution of the problem should contribute to the body of scientific knowledge, not just to the needs of a particular business organisation. Thus contribution to the scientific body of knowledge is the main indicator of presence of the scientific method (or a part of it) in a bachelor thesis. In other words students should satisfy science needs instead of software requirements which are defined by user in software engineering life cycle [19].

Therefore we propose that in at least one numbered cell in Table 1 student's work should correspond to the state of the art in scientific context and give a specific small contribution to the scientific knowledge body. Similarly the work of the student should clearly contribute to at least one general phase of software development. At the same time the style of bachelor thesis should resemble the style of doctoral thesis, i.e., it has to have a structure similar to the one depicted in Figure 1a.

Taking into consideration that bachelor theses are the simplest scientific research at the university, the minimum requirements for the scope of bachelor thesis should not exceed one software development phase and one research phase still keeping the requirement that the bachelor student must be able to position clearly his work in both scientific and software development life cycles (see Fig. 2) [19].

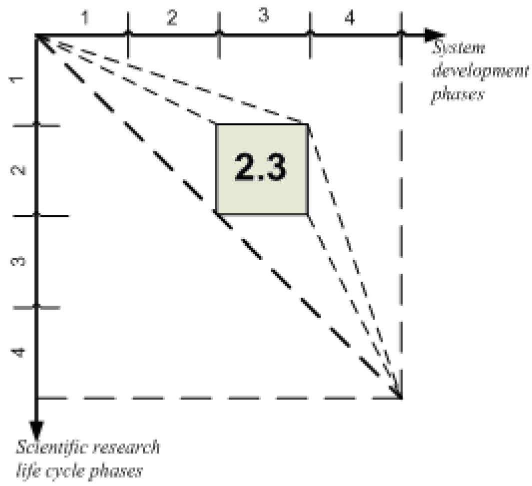


Figure 2. Mapping of stages for software development into schema of scientific method.

Figure 2 illustrates our proposition in more detail. Each numbered cell in Table 1 characterises the research domain of the student. The cell directly below the numbered cell gives practical illustration of student's research work in this domain. In Figure 2 this cell is depicted by the grey square. It is important that student's work be properly positioned with respect to the scientific and systems development life cycles. The positioning is reflected by dotted lines in Figure 2. This means that student shows his/her scientific and practical contribution on a large scale.

On the other hand, each grey square includes all four general stages of research cycle with respect to the problem domain under the consideration, i.e. scope of the bachelor thesis and, thus, reflects scientific research on a small scale. This is reflected by small arrowed links in the rectangle. Figures inside the rectangle correspond to the numbers of cells in Table 1. In terms of generalisation of systems engineering and scientific research lifecycles in four basic phases and according to the positioning principle described above, all themes of bachelor thesis can be represented by 16 basic patterns.

Types of bachelor thesis are briefly described in Tables 2-5 in the remainder of this section. Each type thesis is defined by positioning of a shaded cell (grey square – see also Fig. 2) according to the software development phases and according to the scientific research phases. The point of main emphasis of the bachelor thesis, in terms of phases of scientific research and software development, is denoted as a scope. Each type of thesis is illustrated by at least one instance, which is typical for the author's research institution.

As it has already been stated before, the work of the student should correspond to the state of the art in scientific context and give a particular contribution to scientific knowledge body, and, on the other hand, the work of the student should clearly contribute to at least one general phase of software development, so both the scientific methods and the software development methods could be employed in the development of a bachelor thesis.

Table 2-5 also describes the methods that could be used in the development of a bachelor thesis. As it can be seen from the results described in the table the scientific methods are the same in all development phases whereas the software development methods are different.

Table 2. Mapping of Software Development Phases and Projection of Software Development Methods into Scientific Method Life-Cycle Phase "1. Observation"

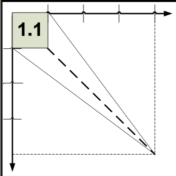
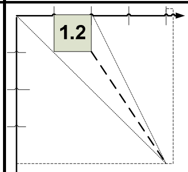
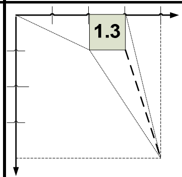
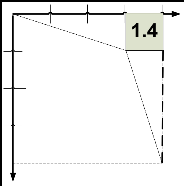
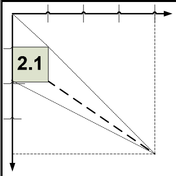
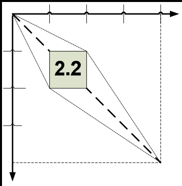
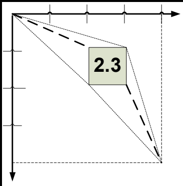
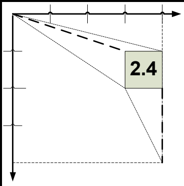
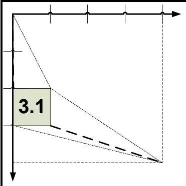
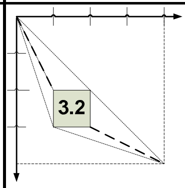
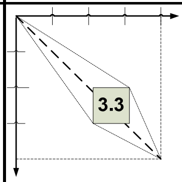
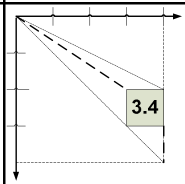
	1.Analysis	2. Design	3. Implementation	4. Testing
Positioning:				
Scope:	1.1. Observation of issues relevant to the analysis phase of SD	1.2. Observation of issues relevant to the design phase of SD	1.3. Observation of issues relevant to the implementation phase of SD	1.4. Observation of issues relevant to the testing phase of SD
Scientific methods:	Literature analysis; Interview; Questionnaire; Case study			
Software development methods:	Interview; Use case; Conceptual modelling; Process modelling; Cause-effect analysis	Component design; Object-based design; Aspect-based design;	PM methods (Prince2); DSDM; Embedded methods (SAP – ARIS; BAN – DEM) Object Process Methodology; Aspect-oriented implementation method; Iterative implementation method	White box; Black box; Agile testing; Exploratory testing; Fuzz testing; Risk-based testing; Scenario-based testing
Example of bachelor thesis essence:	Analysis of a new approach, statement, concept documents, field of interest, problem	Analysis of one or different design solutions	Analysis of one or more systems	Analysis of different sources of scientific research describing the same problem

Table 3. Mapping of Software Development Phases and Projection of Software Development Methods into Scientific Method Life-Cycle Phase "2. Hypothesis"

	1.Analysis	**2. Design**	**3. Implementation**	**4. Testing**
Positioning:				
Scope:	2.1. Finding a solution to a particular problem relevant to the analysis phase of	2.2. Finding a solution to a particular problem relevant to the design phase of SD.	2.3. Finding a solution to a particular problem relevant to the implementation phase of SD.	2.4. Finding a solution to a particular problem relevant to the implementation phase of SD.

	SD.			
Scientific methods:	Inductive reasoning; Deductive reasoning; Bayesian inference			
Software development methods:	Brain Storming; Iterative design;	Brain Storming; Iterative design; Component design; Object-based design; Aspect-based design; Process/service oriented design	Project management methods – Prince2; DSDM; Embedded methods (SAP – ARIS; BAN – DEM); Object Process Methodology; Aspect-oriented implementation method; Iterative implementation method	Brain Storming; Iterative design; Case study
Example of bachelor thesis essence:	Analysis or modification of existing hypotheses.	Formulation of new hypotheses.	Evaluation or implementation of defined hypotheses	Testing of a new or modified hypothesis; testing of an existing hypothesis in different fields.

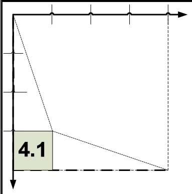
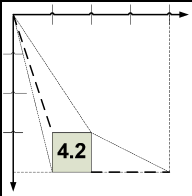
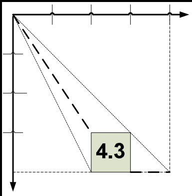
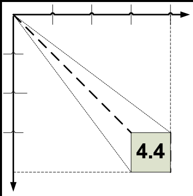
Table 4. Mapping of Software Development Phases and Projection of Software Development Methods into Scientific Method Life-Cycle Phase "3. Predictions"

	1.Analysis	**2. Design**	**3. Implementation**	**4. Testing**
Positioning:				
Scope:	3.1. Application of a particular solution relevant to the analysis phase of SD in different real, experimental or simulated situations.	3.2. Application of a particular solution relevant to the design phase of SD in different real life, experimental or simulated situations.	3.3. Application of a particular solution relevant to the implementation phase of SD in different real life, experimental or simulated situations.	3.4. Application of a particular solution relevant to the testing phase of SD in different real, experimental or simulated situations.
Scientific methods:	Controlled experiment; Natural experiments; Observational studies; Field Experiments; Laboratory experiments; Statistical methods; Case study; Simulation			

	1.Analysis	**2. Design**	**3. Implementation**	**4. Testing**
Software development methods:	By applying an appropriate standard	Iterative design; Component design; Object-based design; Aspect-based design; Process/service oriented design	Project management methods – Prince2; DSDM; Embedded methods (SAP – ARIS; BAN – DEM); Object Process Methodology; Aspect-oriented implementation method; Iterative implementation method	Application of an appropriate standard

Example of bachelor thesis essence:	Development of a concept document for a system.	Design of an agent, system, technology.	Combining different methods; implementation of a system, agent, or technology.	Development of a testing document for a system

Table 5. Mapping of Software Development Phases and Projection of Software Development Methods into Scientific Method Life-Cycle Phase "4. Tests"

	1.Analysis	**2. Design**	**3. Implementation**	**4. Testing**
Positioning:				
Scope:	4.1. Scientific evaluation of a particular solution relevant to the analysis phase of SD.	4.2. Scientific evaluation of a particular solution relevant to the design phase of SD.	4.3. Scientific evaluation of a particular solution relevant to the implemen-tation phase of SD.	4.4. Scientific evaluation of a particular solution relevant to the testing phase of SD.
Scientific methods:	Expert queries; Delphi method; Controlled experiment; Natural experiments; Observational studies; Field Experiments; Laboratory experiments; Statistical methods; Case study; Simulation			
Software development methods:	Classification	Iterative design; Component design; Object-based design; Aspect-based design; Process/service oriented design	Scoring models; Analytic hierarchy process; Balanced scorecard Application; Development Project Estimation; Total cost of ownership; IEEE Recommended Practice for Software Acquisition (IEEE Std 1062, 1998); Value analysis	White box; Black box; Agile testing; Exploratory testing; Fuzz testing; Risk-based testing; Scenario-based testing
Example of bachelor thesis essence:	Evaluation of different approaches, methods, technologies.	Evaluation of different designs (solutions not implemented yet).	Evaluation of different systems.	Approbation and evaluation of a testing document of a system.

The above represented collection describes the basic thesis types only, but the research can not be strictly positioned in phases of software engineering and can not to be rigidly framed in scientific research phases, Because we have to take into consideration that the phases of software engineering are often overlapping and the research can be positioned as a frontier for several phases. As for selection of implemented steps of scientific research the only requirement is that at least one phase, - observation, hypothesis, implementation or scientific testing, would be accomplished for bachelor level of the research.

4. Conclusions

It is often difficult to understand the appropriate level of details or depth of the research, which is required for bachelor thesis. Especially ambiguous this is for research in the field of software engineering, where it is enough complicated to state the boundary between the necessary level of formalization in the research and the ability to present results clearly and perceptibly enough.

The paper presents the discussion about ability to simplify the scientific research life cycle and to try to map it into the schema of software development life cycle for better understanding of the research performance for bachelor level students. The cross screening of scientific research through the phases of software development life cycle give the possibility not only to describe the scientific research process in terms of software engineering, which are easier for students to understand, but also to define the scope and positioning of the main types of bachelor thesis in the field of software engineering.

The division between sub areas in the field of software engineering depends also on the subdivision of the exact phase of software development. One more aspect of scientific research analysis according the software development stages can be investigation of possible sequences of stages, so called models of life-cycles of software development. Analysis of different paradigms existing in software development let to assume that similar paradigms could exist in scientific research too, like agile versus disciplined, structural versus object-oriented, sequential versus iterative and so on by analogy with the software development.

The collection of basic types for bachelor thesis dimensions can serve as a framework for research initialising and positioning it in both life cycles – the scientific and engineering ones. Also the discussion about bachelor thesis types can be useful for students in selection of appropriate methods and strategy of research, as well as for bachelor thesis advisors in definition of recommendations for research performance.

Acknowledgement

The research reflected in the paper is supported by the grant No. ZP/2006-05 of Riga Technical University "Development of the Method for the First Level Scientific Research in the Area of Computer Science and Information Technology" and by the European Social Fund within the National Programme "Support for the carrying out doctoral study program's and post-doctoral researches".

References

1. Kuhn, T., 1962. The structure of scientific revolutions, University of Chicago Press.
2. Sommerville, I., 1992. Software engineering. Addison-Wesley.
3. Bunge, M., 1967. Scientific research. Springer, Berlín, 2 vols.
4. Marcos, E., Marcos, A., 1998. An aristotelian approach to the methodological research: a method for data models construction. In Information Systems- The Next Generation. L. Brooks and C. Kimble (Eds.). Mc Graw-Hill, pp. 532-543.
5. Dobson, P. J., 2001. The Philosophy of Critical Realism-An Opportunity for Information Systems Research. In Information Systems Frontiers, 3:2, pp. 199-210.

6. Gregg, D. G., Kulkarni, U. R., Vinzé, A. S., 2001. Understanding the Philosophical Underpinnings of Software Engineering Research in Information Systems. In Information Systems Frontiers, 3:2, pp. 169-183.
7. Glass, R.L., Vessey, I., Ramesh, V., 2002. Research in software engineering: an analysis of the literature. In Information and Software Technology, Elsevier Science B.V. N.44, pp. 491-506.
8. Wilson, E. B., 1952. An introduction to scientific research, McGraw-Hill.
9. Barrow, J., 1991 . Theories of everything, Oxford University Press.
10. Marcos, E., 2005. Software engineering research versus software development. In ACM SIGSOFT Software Engineering Notes, July 2005, Vol. 30, No. 4, pp. 1-7.
11. Nikiforova, O. *Comparison Methodology of Software Development Means*, PhD Thesis, Riga Technical University, Latvia, 2001, p.136
12. Royce, W.W. 1970. Managing the development of large software systems. In Proceedings of WESTCON, San Francisco.
13. Boehm, B. W., 1986. A spiral model of development and enhancement. In Software Engineering Notes, Vol. 11, No. 4, August.
14. Henderson-Sellers, B., Edwards, J.M., 1993. The object-oriented systems lifecycle. In Communications of the ACM, 33(9).
15. Jacobson, I., Booch, G., Rumbaugh, J., 1999. The Unified Software Development Process, Addison-Wesley.
16. MSF, 2006. Microsoft Solution Framework – available at http://www.microsoft.com/technet/solutionaccelerators/msf/default.mspx
17. Nikiforova, O.. General Framework for Object-oriented Software Development Process. In *Scientific Proceedings of Riga Technical University, Computer Science, Applied Computer Systems,* 3rd thematic issue, 2002, pp.132-144.
18. Nikiforova O., Kirikova M., Strazdina R. „Some Issues on Research Essentials in the Field of Software Engineering. Simplified Look on Scientific Method for Bachelor Level Research", Proceedings of the 2nd International Conference on Evaluation of Novel Approaches to Software Engineering, Barcelona, Spain, July 23-25, 2007, pp. 48-55
19. Nikiforova O., Strazdina R., Kirikova M. „Bachelor Level Research Methodology in the Field of System Engineering", Joining Forces in Engineering Education Towards Excellence Proceedings, Szentirmai, Szarka T.G. Eds., SEFI and IGIP Joint Annual Conference, Miskolc, Hungary, CD, p. 10, 2007

Ņikiforova O., Kirikova M., Strazdiņa R. Izvilkuma darbs par pētījuma metodi sistēmas inženierijas jomā: Bakalaura līmenis

Nebūt ne katrs sistēmas izstrādes projekts ir uzskatāms par zinātnisku darbu, jo zinātniskam darbam ir jādod noteikts zinātnisks ieguldījums, ko varētu identificēt kā jaunas zināšanas attiecībā pret jau eksistējošajām „zinātniskajām" zināšanām projektam atbilstošā sfērā. Tas nozīmē, ka inženierzinātnēs studenta pētījumam, no vienas puses, ir jāsniedz kaut neliels zinātnisks ieguldījums (jaunas "zinātniskās" zināšanas atbilstošajā sfērā) un, no otras puses, studenta darbam ir jāsniedz skaidri identificējams rezultāts vismaz vienai no vispārīgajām sistēmas izstrādes fāzēm. Tā kā bakalaura darbs ir pirmais zinātniskais darbs studiju laikā, tad minimālās prasības pret bakalaura darbu inženierzinātnēs nepārsniedz prasību pēc skaidri identificējama ieguldījumā attiecībā pret vienu zinātnisko pētījumu fāzi un skaidri identificējama ieguldījumu attiecībā pret vienu sistēmas izstrādes fāzi. Lai veicinātu šādas identifikācijas iespēju, zinātnisko pētījumu dzīves cikls tika analizēts no programmatūras inženierijas dzīves cikla viedokļa un abu ciklu aktivitātes kartētas vienotā shēmā. Kartēšanas rezultātā tika iegūti 16 bakalaura darbu pamattipi programmatūras inženierijā. Informācija par šiem pamattipiem ir noderīga gan bakalaura darbu izstrādātājiem, gan viņu vadītājiem centienos panākt bakalaura darbu atbilstību augstāk izvirzītajām prasībām.

Nikiforova O., Kirikova M., Strazdina R. An open work on research method in the field of system engineering: The bachelor level

Not every system engineering task will qualify as scientific research, because the solution of the problem should contribute to the body of scientific knowledge. This means that, on one hand, the research work should correspond to the state of the art of scientific body of knowledge and give a particular, small, contribution to it,

and, on the other hand, the work of the student shall clearly contribute to at least one general phase of software development. Taking into consideration that bachelor thesis are an initial research and the simplest scientific research at the university, the minimum requirements for the scope of bachelor thesis, thus, should not exceed one software development phase and one research phase, still keeping the requirement that the bachelor student must be able to position clearly his work in both, namely, scientific and software development life cycles. Therefore the life cycle of scientific research is analysed from the perspective of software engineering life cycle and the main activities of both are mapped into single schema. 16 types of bachelor thesis in software engineering proposed in the paper are a helpful tool for bachelor thesis developers and advisers to meet the above mentioned requirements.

Никифорова О., Кирикова М., Страздиня Р. Открытая работа о методе исследования в области системной инженерии: Уровень бакалавра

Не каждый проект по разработке систем в области программной инженерии может быть квалифицирован как научное исследование, так как научное исследование должно давать вклад в науку, который можно определить как новые знания по отношению к уже существующим знаниям в соответствующей научной области. Это означает, что в области программной инженерии исследования студентов, посвященные разработке дипломных работ, должны с одной стороны давать хотя бы небольшой научный вклад в соответствующую область, с другой стороны у работы должен быть ощутимый результат для одной из фаз разработки системы программного обеспечения. Так как разработка бакалаврской работы является первым опытом ведения научного исследования, то можно предположить, что минимальным требованием к бакалаврской работе может быть требование продемонстрировать в работе навыки ведения исследования хотя бы для одной из фаз разработки систем, и проведение научного исследования, углубляясь хотя бы только в одну из стадий проведения исследования как такового. Чтобы подтвердить возможность такой классификации бакалаврских работ в статье были проанализированы фазы ведения научного исследования с точки зрения их аналогии с фазами разработки систем программного обеспечения, а также были предложены на рассмотрение 16 типов возможной классификации бакалаврских работ относительно фокуса детализации проведения исследования. Предложенная классификация бакалаврских работ может быть полезна как студентам, готовящимся к проведению исследования в рамках бакалаврской работы, так и их научным руководителям, чтобы определиться со степенью детализации ведения исследования и углубления в область научного исследования.