

# Investigation of Ontologies in Software-Engineering-Meta-Research

Dmitrii Seletkov

Institute for Program Structures and Data Organization (IPD)

Advisor: Dipl.-Inform. Angelika Kaplan

Abstract.

## 1 Introduction

- The definition of Ontology in Philosophy and Computer Science
- The definition of Meta Research
- Above-named terms Ontology and Meta-research in Software Engineering(SE)
- Help of ontology-based systems in scientific search

Generally, the content (what it will be about) of the following proseminar work is summed up.

## 2 Foundations

In this section the required knowledge foundations are introduced and set up.

### 2.1 Ontology in Computer Science

#### 2.1.1 Ontology languages

The short historical survey of ontology languages approaches, based on lectures notes of course “Ontology and knowledge representation from Prof. Boris Konev Head of Department of Computer Science, the University of Liverpool, UK and member of the Knowledge Representation Research Group [13]. In this sequence

- Resource Description Framework Schema: the first standard of W3C for ontologies [6], its semantic power and drawbacks
- Description Logic: introduction in Description Logic, EL language, its architecture and semantic
- Web Ontology Language: the newest standard of W3L based on Description Logic [4]

### 2.1.2 Expediency and reasons for using of Ontologies

Here is discussed, when and where the ontologies should be used and when it is superfluous

### 2.1.3 Examples of Ontologies

Here come the examples of ontologies in medicine, then in Meta-research and SE.

## 2.2 Meta-research

### 2.2.1 Motivation

A statement «Science is one of the main driver of human progress» is indisputable and proved by growing number of published papers and publishing authors [9]. Naturally people face the problems such as data sharing, replications of experiments, its ownership and many others. Moreover, the research practices suffer from lack of systematization and inefficiency. For example, according to [14] 85% of resources in biomedical research are wasted because of above-called reasons. Therefore, there exists the urgent need of Meta-research, which aims to include the evaluation of diverse researches with succeeding suggestions of improvement for research practices.

### 2.2.2 Areas of Meta-research

Discussing the existing problems about contemporary research the logical question arises: is it possible to provide a map of ongoing efforts in the field Meta-research and connect the multiple already made, but still fragmented attempts across science. These goals were set by J. Ioannidis in its work «Meta-research: Evaluation and Improvement of Research Methods and practices» [10].

His suggestions can be summed up in the following categorization. There are five major areas of interests in meta-research: methods, reporting, reproducibility, evaluation and incentives. Each of them was not only explicitly defined and with examples illustrated, but also for each area the existing initiatives were found. These mentioned features of the work can be summarized as following categorizations:

1. **Methods:** practices for performing research (**e.g.** study design, methods, statistics). With **specific interests** in biases and questionable practices in conducting research,

methods to reduce such biases, metaanalysis. Existing **Initiatives** such as Cochrane Collaboration for systematic reviews of health care or Campbell Collaboration for the same ones but in social science.

2. **Reporting:** publications of standards and study registrations (e.g. study registration, information to patients, public and policy-makers). With **specific interests** in biases and questionable practices in reporting, explaining, disseminating and popularizing research. Existing **Initiatives** such as ClinicalTrials.gov for clinical trials registrations or EQUATOR network for reporting standards for research.
3. **Reproducibility:** methods for verifying research (e.g. sharing data and methods, replicability). With **specific interests** in overcoming of obstacles to sharing data, methods and replications. Existing **Initiatives** such as YODA for sharing data in clinical research or BITSS for transparency in social science.
4. **Evaluation:** improvements for scientific quality (e.g. pre- and postpublication peer reviews, research funding criteria). With **specific interests** in effectiveness, costs, and benefits of old and new approaches to peer review. Existing **Initiatives** such as Peer Review Congress for evidence on peer review or ArXiv for preprinting articles.
5. **Incentives:** rewards and penalties for research (e.g. promotion criteria, penalties in research evaluation). With **specific interests** in Accuracy, effectiveness and benefits of old and new approaches to ranking and evaluating the performance. Existing **Initiatives** such as REWARD for reducing waste and rewarding diligence in research or AAAS for science policy.

Certainly, it has to be said that this is an «nonexhaustive list». Also, it is worth of remarking that neither in this paper nor in referenced and citing papers the initiatives in SE-Meta-Research are considered. Therefore, strictly based on definitions and comparing of initiatives in other fields, the mentioned in this work approaches in Software Engineering will be categorized, if it is possible and if not, discussed limitations of such classification.

### 3 **Ontology in Software-Engineering-Meta-Research**

This section aims to show different types of ontologies, which are used by scientists for Meta-research in Software Engineering, and also to compare them.

Firstly, in order to show all this diversity, the Biolchini's[1] classification of empirical studies in SE has been chosen. It shows «the concepts of Primary and Secondary Studies on Software Engineering at a high level[8, p. 1].». According to this classification any experimental study comprises two types of investigations: Primary and Secondary[1, p. 134]. Primary studies are used for evaluation of the researcher's hypothesis and represented above all by Controlled Experiments. In contrast, Secondary studies serve for comparisons between individual investigations to generalize the results and are represented by systematic reviews.

Secondly, in order to make the introduced ontologies comparable, the template «Problem, Objectives, Suggested method and Future works» that was used by me during the reading of papers, will be applied. Although this segmentation cannot transfer the whole information contained in papers, it should be sufficient for the set intentions.

### 3.1 Ontology to support systematic reviews in Software Engineering

Before we go to systematic reviews, let me introduce Evidence-based Software Engineering(EBSE), whose main instrument they are. EBSE was evolved by Kitchenham[12]. The author supposes that SE might benefit from an evidence-based methods, as it was done in medicine with appearance of Evidence-based Medicine(EBM). The goal of EBSE is «to provide the means by which current best evidence from research can be integrated with practical experience and human values in the decision making process regarding the development and maintenance of software»[12, p. 2]. In other words, what SE practice works, when, where and which tools and standards are needed for that. This all can and should be proven by experiments using Systematic reviews(SRs), where SRs are «form of secondary study that uses a well-defined methodology to identify, analyse and interpret all available evidence related to a specific research question in a way that is unbiased and (to a degree) repeatable. »[Link to guidelines kitchenham] or simplified a tool to obtain accurate knowledge by analyzing the primary studies to eliminate possible biases.

Having so detailed introduction of origin SRs, Biolchini has set a **Problem**: produce knowledge that can be based on scientific methodology. Besides, according to his work this problem implies the following **Objectives**:

- Discussing the significance of experimental studies, particularly SRs and their use in supporting software processes
- Present a template designed to support systematic reviews in SE
- Introduce development of ontologies to describe knowledge regarding such experimental studies

So as to fulfill the determined objectives, the author introduces, firstly, a template for systematic review protocol(Figure 1) that was created based corresponding Guidelines[11] and own experience. It is worth to notice the later relevant parts of them, such as Problem (SR target, describing the research context), Intervention(observation target in SR), control(initial data already possessed by researcher), outcome measure (metrics to measure effects), experimental design(which statistical analysis method will be applied on the collected data).

Secondly, the author **suggest** «Scientific research ontology» that is organized in level-structure (entities of levels can be seen as concepts) and the levels posses taxonomic and meronymic hierarchies (can be seen as roles), namely is\_a and has relations. The following

|   |   |
|---|---|
| 1. Question Formularization<br>1.1. Question Focus<br>1.2. Question Quality and Amplitude<br>- Problem<br>- Question.<br>- Keywords and Synonyms<br>- Intervention<br>- Control<br>- Effect<br>- Outcome Measure<br>- Population.<br>- Application<br>- Experimental Design<br>2. Sources Selection<br>2.1. Sources Selection Criteria Definition<br>2.2. Studies Languages<br>2.3. Sources Identification<br>- Sources Search Methods<br>- Search String<br>- Sources List<br>2.4. Sources Selection after Evaluation<br>2.5. References Checking<br>3. Studies Selection<br>3.1. Studies Definition<br>- Studies Inclusion and Exclusion Criteria Definition<br>- Studies Types Definition<br>3.2. Procedures for Studies Selection | 3.3. Selection Execution<br>- Initial Studies Selection<br>- Studies Quality Evaluation<br>- Selection Review<br>4. Information Extraction<br>4.1. Information Inclusion and Exclusion Criteria Definition<br>4.2. Data Extraction Forms<br>4.3. Extraction Execution<br>- Objective Results Extraction<br>i) Study Identification<br>ii) Study Methodology<br>iii) Study Results<br>iv) Study Problems<br>- Subjective Results Extraction<br>i) Information through Authors<br>ii) General Impressions and Abstractions<br>4.4. Resolution of divergences among reviewers<br>5. Results Summarization<br>5.1. Results Statistical Calculus<br>5.2. Results Presentation in Tables<br>5.3. Sensitivity Analysis<br>5.4. Plotting<br>5.5. Final Comments<br>- Number of Studies<br>- Search, Selection and Extraction Bias<br>- Publication Bias<br>- Inter-Reviewers Variation.<br>- Results Application<br>- Recommendations |
|---|---|

Figure 1: Systematic review protocol template[1, p. 142]

paragraph summarizes and analyzes the main components and features of the suggested ontology level by level with pertaining relations between them.

**Level 0:** Different knowledge of domains that are involved in the conduction of SRs in SE, represented by *Experimental Method*, *Primary Research* and *Research Synthesis*

For the further discussion the domain *Primary Research* is selected, but the analog statements and conclusions can be derived and provided by remaining ones.

**Level 1:** The conceptual entity, represented by *Primary Study Element* is the highest level of hypernym of the *Primary Research* and subsumes the concepts in the lower levels of hierarchy.

**Level 2:** The main concepts, represented by *Structure of Study* and *Quality of Study*

**Level 3:** The subcategories of one of the main concept Structure of study, represented by *Problem*, *Hypothesis*, *Intervention*, *Control*, *Measurement*, *Outcome* and *Unit of Study*

**Level 4:** The entities of the subcategory *Outcome* that is demonstrating the ontological hybridism, having not only taxonomic relations, represented by *Target Outcome* and *Surrogate Outcome*, but also meronymic relations, represented by *Endpoint*, *Incidence*, *Prevalence*, *Effect Modification* and *Effect Modifier*

As we can observe, the above described part of the ontology **results** in an object that directly linked with Systematic review protocol template. The full ontology concepts and roles can be read in the paper, but the given example in depth of levels depicts the power comprehensive cover of SR needs.

Though the presented ontology was at the moment of publication only in development, it can be esteemed as robust for entire diversity of SRs in different types of Studies. Nevertheless, the author points on possible **future works**, which are mainly related to merging of the presented Scientific Research Ontology with Software Engineering Ontology and successive integrating them into eSEE (experimental Software Engineering Environment). Moreover, this could lead towards a wider Experimental Software Engineering Ontology that theoretically will combine all received evidence-based knowledge in Software Engineering.

//TODO RESULT!

### 3.2 Ontologies for Controlled Experiments on Software Engineering

In contrast to the previous subsection, this one is about a part of Primary Studies, where one of the main subjects are Controlled Experiments. They serve Experimental Software Engineering as an instrument to build a body of knowledge for diverse and exceeding software practices to support making successful decision in SE.

Obviously, there exist many problems in this field and one of them is trying to be solved by R. E. Garcia[8]. This **Problem** is sharing of knowledge among research groups. It requires replication of Controlled Experiments. The generated Knowledge during these experiments is registered in so-called Lab Packages (procedures, the results and conclusions). However, researchers face difficulties reviewing the lab packages and suffer from the lack of standardization, what leads to obstacles in sharing knowledge among research groups.

According to the problem there were set the following **Objectives**:

- Explore ontologies to support knowledge transfer, helping to elucidate the associated concepts of controlled experiments and their relationships.
- Present an Ontology to experimental studies, named EXPEROntology - tool for knowledge transfer, assisting researchers, reviewers, and meta-analysts in designing, conducting, and evaluating controlled experiments.
- Validate the ontology, whilst instantiating it to a controlled experiment.

Before we come to the suggested ontology, it is noteworthy to talk about the main object of the current research, in particular five Controlled Experiments phases and its components[19]:

1. Definition: hypotheses and experiment goals

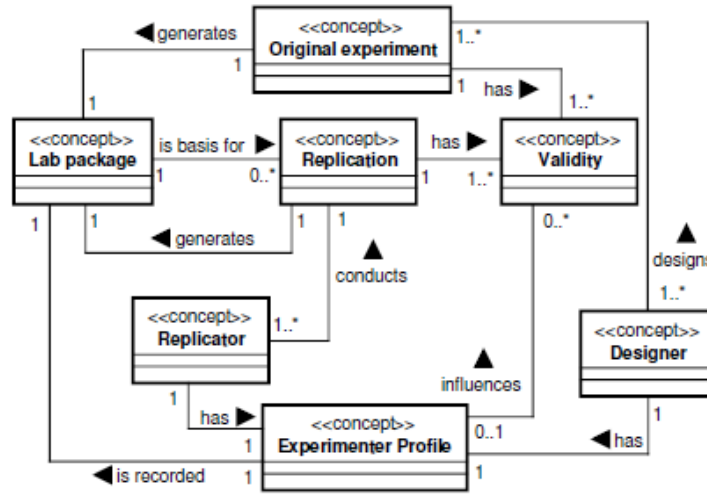


Figure 2: Ontology for Controlled Experiment[8, p. 3]

2. Planning: execution plan and environment; subjects and their profiles; dependent and independent variables; validity
3. Operation: preparation, execution, data validation
4. Analysis: analysis of collected data
5. Packaging : artifacts, procedures, results into Lab Packages (is recommended executing parallel with each phase)

Some above named components show up in the **suggested** ontology, which can be seen on Figure 2 and has the following ordered workflow, consisting as usual of concepts and roles (relations) between them:

1. *Lab Package* from *Original Experiment* (created by *Designer*) is used for *Replication* (by *Replicator*) and generation of a new *Lab Package*.
2. *Designer* and *Replicator* have *Experimenter Profile*: negative influence attests a lack of experience, positive - high experience
3. *Original Experiment* and *Replication* evaluated regarding to *Validity* with four types: conclusion (relationship between the treatment and outcome), internal (relationship between the factors and the outcome), construct (relation theory and observation and external(generalization)).

But so superficial-designed is apparently not be able to sophisticate all needs of researchers. Therefore, Garcia goes deeper and refine the above described ontology, presenting the EX-PEROntology for Lab Package that can be observed on Figure 3. Besides, four years later





the author publishes the guidelines for this ontology[17], extending the earlier work and providing more circumstantial description of that. Let us focus below on the main structures and features of the presented ontology, corresponding to it workflow and how it refers to earlier set up phases of Controlled Experiments:

1. Definition: establishing of *Initial hypothesis*, composed by *Object of study*, *Purpose*, *Quality focus* in a specific *Context*.
2. Planning: generating *Hypothesis formalized* from Definition phase. Experimenter defines *Dependent*, *Independent Variables* and *Experimental Object* that contains *Technology* and *Artifacts* to be used in controlled experiment. Furthermore, in this phase the *Experimental design* is created and refined by *Subject* and its *Profile*.
3. Operation: elaborating *Execution plan*, which is obtained by *Tasks*
4. Analysis: gathering *Results* from Operation phase are to be used in *Analysis* of different types, such as *Confirmatory*(testing *Hypothesis formalized*) or *Exploratory*(investigating new relationships)

Since it is desired to make all these concepts working, we are missing the connection axioms for the entire ontology. It is enough to define four predicates: TODO

1.

$$Design(subject, SetOfTreatment)$$

2.

$$\begin{aligned} & Factor(f_1, \dots, f_n) \\ & \forall f \in Factor, \exists Treatment(f) = v_1, \dots, v_n, n \geq 2 \\ & dom(Treatment) = Artifact \cup Technology \\ & SetOfTreatment = (vf_1, \dots, vf_n) | \forall f, vf_n \in Treatment(f_n) \end{aligned}$$

3.

$$\begin{aligned} & \forall subject, SetOfTreatment \\ & Execution(subject, SetOfTreatment) \rightarrow Task(ta_1) \wedge \dots \wedge Task(ta_n) \end{aligned}$$

4.

$$Task(ta_n) \rightarrow Training(subject, t_r, a, p) \vee Applying(subject, t_e, a, p)$$

The last, but not least is validating of presented ontology by obtaining the **Results** on existing data. Thus, in order to illustrate the power of the suggested ontology, it has been chosen an experiment that was originally conducted by Basili and Selby[2]. The objective of the study was «compare three state-of-the-practice software testing techniques: a) code reading by stepwise abstraction, b) functional testing using equivalence partitioning and boundary value analysis, and c) structural testing using 100 percent statement coverage criteria.»[2, abstract]. So, the Experimental Design of the study: 32 Subjects were divided in 3 groups(advanced, intermediate and junior) and each of them applied 3 software testing techniques(code reading, functional testing and structural testing) on 3 different pieces of software(text processor, numeric abstract data type and database maintainer). According to this description the EXPEROntology was instantiated as following(for instance only for one subject S1).

1.

*Design(S1, Advanced, Code Reading, P3)*  
*Design(S1, Advanced, Functional Testing, P2)*  
*Design(S1, Advanced, Structural Testing, P1)*

2.

*Factor = (Expertise, Technique, Program)*  
*for Expertise, Treatment = {Advanced, Intermediate, Junior}*  
*for Technique, Treatment = {Code Reading, Functional Testing, Structural Testing}*  
*for Program, Treatment = {P1, P2, P3}*  
*SetOfTreatment = (Advanced, Code Reading, P3)*

3. and 4.

*Execution() →*

*Training(S1, {Code Reading ∨ Functional Testing ∨ Structural Testing}, --, --) ∧ ...*  
*Applying(S1, {Code Reading ∨ Functional Testing ∨ Structural Testing}, P3, --) ∧ ...*

After the instantiating of the ontology, we can observe that all pieces of information given in the conducted study can be encapsulated and saved in EXPEROntology without losing any information.

Besides, we can observe the missing values on the predicate: Training - artifact and period of training, Applying - period for the application. After looking in the study we can see that

it does not bring them indeed. It allows us to make one more important conclusion «the ontology can also be used as a mechanism to improve the obtained data set from the Lab Package»[8, p. 6]

Also, the authors intend in **Future works** combine the presented EXPEROntology with OntoTest. By merging these two ontology it will be possible to create an whole architecture that «supports the development of environments/tools to automate, software testing activities and related experimental studies»[8, p.6]

We have just looked at the ontology for the controlled experiments. However, there exists another work from alternative perspective, which was suggested by H. Siy and Y. Wu[18]. Although they had the similar **Objectives**:

- Present an ontology for analyzing empirical studies of SE, in particular the design of software engineering experiment
- Encapsulate the experience experts by means of an ontology for experimental designs using Protege OWL

they are aware of the **Problem** wider in enterprise way, namely a bad designed experiment can increase the cost and risk of invalid results. And the solving this problem includes not only the fundamental knowledge of the mechanisms, methods and tools, thus improving our understanding of which one works best under what situation, but also the way software engineers work, think and interact with each other.

So as to solve this problem, the author **suggest** the following ontology concept. First of all, the main concepts and their roles(Figure 4 are presented:

- *Treatment* Software Engineering Technique/Method/Process being studied
- *Subject* Person, Developer/Student participated on experiment
- *Object* Entity, Program/Model
- *Assignment* Relation between all of them; in an assignmentInstance: a subjectInstance is assigned to apply a treatmentInstance in an objectInstance.

Secondly, in order to bring all these concepts working, the connection axioms are set. In sum, we get four necessary constraints and the purposes they fulfill:

1. No *Subject* can be assigned a *Treatment* that is less sophisticated than the other ones he was already assigned to. What for: subject assigned to one treatment may use the knowledge and experience gained from that treatment.
2. No *Treatment* was applied by only one *Subject*. What for: experiment subjects have varying backgrounds and abilities, what implies different results. If only one subject is related, then it is not scientifically meaningful.

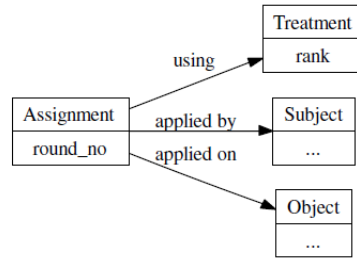


Figure 4: Ontology fragment depicting the concepts involved in the design of experiments[18, p. 13]

3. *Subject* is assigned to several *Treatments*. What for: assess the variability introduced by that subject
4. An *Object* is treated by several *Subjects*. What for: provide a way to untangle subject performance from object complexity

Finally, the created ontology concept was evaluated on experiments on software inspections[3]. Aligning to presented concepts the concrete values i. e. *Treatment* - reading technique, *Subject* - reviewer, *Object* - software requirements document. The **Results** were «encouraging»[18, p. 15]. The author states no it was not found any validation after the applying all constraints. Moreover, if the ontology was fed by invalid assignments, it was possible to observe inconsistencies as expected.

It remains to be said that the authors consider the **Future** of their presented ontology and the preliminary results that it has shown as a step towards organizing and accommodating such a ontology, which can also support other Software Engineering knowledge domains(not only the systematic review process).

To the very end, I would like to discuss the similarities and differences of suggested in this sections ontologies. Firstly, it is remarkable that all researchers point on the advantages and opportunities which the adoption of ontologies give them in realization of their intentions. In the researcher's opinion, an ontology is the best tool to accumulate any experience or knowledge(especially, from experiments), formalize them for later representing, sharing and transferring. Nevertheless, ontology is not a silver bullet, because of apparent drawbacks. It cannot depict preciously all objects of the real world and relations between them. An ontology is still restricted by the first-order logic. But sometimes it can be enough to get desired results. Moreover, we could observe that the modeling of each presented ontology was done by using of different ontology languages or at least different kinds of the same language what can also lead to the barriers by applying of the presented ontologies and making them standards in the research field. Secondly, TODO

## 4 Ontology-based systems in scientific search

During the searching and reading of the suggested and found by reference-based search papers the two facts appeared in sight: Ontologies are used in contemporary search engines and the the scientific world is using ontologies successfully for a big amount of tasks. Therefore, based on database search (because this was not the original accent of the suggested literature list) the papers about Ontology-based systems in scientific search were found and will be analyzed further in this section.

### 4.1 Semantic Web

Definition, change from syntax Web (WWW) to the semantic Web, motivation and importance of that [5]

### 4.2 Semantic Search Engine

Semantic search is required by a search engine to properly interpret the meaning of a user's query and the inherent relations among the terms that a document contains with respect to a specific domain. Traditional approaches do not care about the semantic relations among the terms that a document contains with respect to a specific domain. How to make it possible for the search engine to work even in case that there are no available domain ontologies for user requests [7]

### 4.3 Examples

In medicine: Semantic search engine for cancer [16] with Objective:  
to build an Ontology, using Medical Knowledge Base, in order to analyze the knowledge about cancer, its categories, cause, symptoms etc., thereby translating the human natural language in a machine and human readable language

In general (with evaluation in medicine): information extraction from scientific abstracts [15] with Objective:  
to describe a novel ontology-based interactive information extraction (OBIIE) framework and how this system enables life scientists to make ad hoc queries similar to using a standard search engine specific OBIIE system (evaluating for extracting co-factors from EMBASE and MEDLINE (medical databases))

## 5 Conclusion

## References

- [1] Jorge Calmon de Almeida Biolchini et al. “Scientific research ontology to support systematic review in software engineering”. In: *Adv. Eng. Informatics* 21.2 (2007), pp. 133–151.
- [2] V. R. Basili and R. W. Selby. “Comparing the Effectiveness of Software Testing Strategies”. In: SE-13 (1987), pp. 1278–1296. ISSN: 0098-5589. DOI: 10.1109/tse.1987.232881.
- [3] V. R. Basili, F. Shull, and F. Lanubile. “Building knowledge through families of experiments”. In: 25 (1999), pp. 456–473. ISSN: 0098-5589. DOI: 10.1109/32.799939.
- [4] Sean Bechhofer et al. *OWL Web Ontology Language Reference*. 2004. URL: <http://www.w3.org/TR/2004/REC-owl-ref-20040210/>.
- [5] Tim Berners-Lee, James Hendler, and Ora Lassila. “The Semantic Web”. In: 284 (2001), pp. 34–43. ISSN: 0036-8733. DOI: 10.1038/scientificamerican0501-34.
- [6] Dan Brickley and R.V. Guha. *RDF Vocabulary Description Language 1.0: RDF Schema*. W3C Recommendation. World Wide Web Consortium, Feb. 2004. URL: <http://www.w3.org/TR/2004/REC-rdf-schema-20040210/>.
- [7] Wei-Dong Fang et al. *Toward a semantic search engine based on ontologies*. 2005. DOI: 10.1109/icmlc.2005.1527258.
- [8] Rogério Eduardo Garcia et al. “An Ontology for Controlled Experiments on Software Engineering”. In: *Proceedings of the Twentieth International Conference on Software Engineering & Knowledge Engineering (SEKE’2008), San Francisco, CA, USA, July 1-3, 2008*. Knowledge Systems Institute Graduate School, 2008, pp. 685–690.
- [9] John P. A. Ioannidis, Kevin W. Boyack, and Richard Klavans. “Estimates of the Continuously Publishing Core in the Scientific Workforce”. In: 9 (), e101698. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0101698.
- [10] John P. A. Ioannidis et al. “Meta-research: Evaluation and Improvement of Research Methods and Practices”. In: 13 (), e1002264. ISSN: 1545-7885. DOI: 10.1371/journal.pbio.1002264.
- [11] Barbara Kitchenham and Stuart M. Charters. *Guidelines for performing Systematic Literature Reviews in Software Engineering*. Tech. rep. 2007. URL: [https://www.researchgate.net/publication/302924724\\_Guidelines\\_for\\_performing\\_Systematic\\_Literature\\_Reviews\\_in\\_Software\\_Engineering](https://www.researchgate.net/publication/302924724_Guidelines_for_performing_Systematic_Literature_Reviews_in_Software_Engineering).
- [12] Barbara A. Kitchenham, Tore Dyba, and Magne Jorgensen. “Evidence-Based Software Engineering”. In: *26th International Conference on Software Engineering (ICSE ’04)*. Edinburgh, Scotland, May 2004, pp. 273–281.

- 
- [13] Boris Konev. *Lecture notes of course Ontology and knowledge representation*. 2010. URL: <https://www.lektorium.tv/speaker/2680>.
  - [14] Malcolm R. Macleod et al. "Biomedical research: increasing value, reducing waste". In: 383 (2014), pp. 101–104. ISSN: 0140-6736. DOI: 10.1016/S0140-6736(13)62329-6.
  - [15] David Milward et al. "Ontology-Based Interactive Information Extraction From Scientific Abstracts". In: 6 (2005), pp. 67–71. ISSN: 1531-6912. DOI: 10.1002/cfg.456.
  - [16] Syam RajBS and Sarumathi S. "Ontology based Semantic Search Engine for Cancer". In: (2014), pp. 39–43. ISSN: 0975-8887. DOI: 10.5120/16594-6308.
  - [17] Lilian Passos Scatalon, Rogério Eduardo Garcia, and Ronaldo Celso Messias Correia. "Packaging Controlled Experiments Using an Evolutionary Approach Based on Ontology(S)". In: *Proceedings of the 23rd International Conference on Software Engineering & Knowledge Engineering (SEKE'2011), Eden Roc Renaissance, Miami Beach, USA, July 7-9, 2011*. Knowledge Systems Institute Graduate School, 2011, pp. 408–413.
  - [18] Harvey Siy and Yan Wu. *An Ontology to Support Empirical Studies in Software Engineering*. en. Mar. 2012. URL: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.216.8899;%20http://csalpha.ist.unomaha.edu/~hsiy/research/icc09.pdf>.
  - [19] Claes Wohlin et al. *Experimentation in Software Engineering - An Introduction*. Vol. 6. The Kluwer International Series in Software Engineering. Kluwer, 2000. ISBN: 978-1-4613-7091-8. DOI: 10.1007/978-1-4615-4625-2.