Supporting Information Sharing for Reuse and Analysis of Scientific Research Publication Data

Fajar J. Ekaputra[1], Marta Sabou[2], Estefanía Serral[3], Stefan Biffl[1]

[1] Vienna University of Technology, Christian Doppler Laboratory CDL-Flex
Favoritenstrasse 9/188, 1040 Vienna, Austria
{fajar.ekaputra, stefan.biffl}@tuwien.ac.at

[2] MODUL University Vienna, Department of New Media Technology
Am Kahlenberg 1, 1190 Vienna, Austria
sabou@modul.ac.at

[3] Department of Decision Sciences and Information Management, KU Leuven
Naamsestraat 69, 3000 Leuven, Belgium
estefania.serralasensio@kuleuven.be

Abstract. Effective and efficient information sharing for reuse and analysis of scientific data from published research papers is an important challenge for researchers working within the empirical software engineering (EMSE) domain. Currently, there is only limited support for storing empirical research data and results in a way that is easy to access and reuse for other researchers. In this paper, we propose the Systematic Knowledge Engineering Tool (SKET), an ontology-based tool to provide researchers in the EMSE domain with capabilities for storing, sharing, and verifying results within their research community. The initial evaluation results show that SKET can address relevant needs in the EMSE community and can be considered as a foundation for advanced tool capabilities.

Keywords: Empirical Software Engineering, Empirical Research, Publication Database, and Ontology.

1 Introduction

Publication databases (PDs) are large collections of scientific publications such as research papers and books. PDs help editors represent and publish research-related books and papers and have become the main source of information for researchers to discover and survey recent research results in their respective areas. The current approach to structure and represent a PD (e.g., IEEEXplore[1], Google Scholar and Sco-

---

[1] http://ieeexplore.org/; http://scholar.google.at/; http://www.elsevier.com/online-tools/scopus

pus) mostly focuses on syntactic search in generic publication meta-data elements e.g., title, keywords (with the notable exception of MeSH[2] that also provides term-based search). Therefore, these approaches provide only limited support for effective reuse and analysis of scientific data from research papers. This hinders the users in reenacting the analysis reported in a paper or conducting other advanced analysis.

Empirical software engineering (EMSE) is one research area affected by the limited support of the current PD approach. A typical example of EMSE research data that is currently difficult to access in traditional PDs includes: the research hypotheses, key result variables, the outcomes of the experiment, and the raw experiment data / materials from EMSE experiments. With the available tools, users usually need significant expertise, resources, and time to extract detailed information from a published paper. This typically cost several person/hours every time a user wants to extract different elements of research data from a paper [1][2]. This phenomenon is also true for meta-researchers, who analyze several experiments in a particular field, e.g., as part of a Systematic Literature Review (SLR) process [3].

Since the number of EMSE studies is considerably growing, it is needed to adopt systematic approaches in order to provide objective evidences on a particular EMSE topic in an efficient way [9]. To deal with this problem, our work extends the publication database approaches to allow the extraction, storage and efficient querying of experimental level data from publication datasets.

Key requirements of EMSE researchers for a good solution are:

- An effective and efficient process for data import, including quality assurance, to enable the incremental building and reuse of knowledge. In a context that needs high-quality data (e.g., EMSE), humans need to be involved to assure the quality of extracted data and relationships between data elements.
- Concept-level search capabilities on EMSE research documentation, e.g., result variables of an experiment.
- An effective and efficient process for querying and data exports from the PD to enable advanced data analysis processes based on the collected knowledge.

Within this paper, we report on the design and implementation of the Systematic Knowledge Engineering Tool (SKET). SKET is a PD that combines the power of an ontology-based approach [6] and domain experts' knowledge to further enhance the value of the PD. The usage of ontologies provides a platform for flexible data modeling, strong querying support with SPARQL, and also inference capability for enhancing the data access. Furthermore, the usage of ontologies as data storage will make it easier for SKET to interact with other related data sources in the future.

Two distinct SKET prototypes covering different EMSE topics are available online[3,4]. We evaluated these prototypes based on the most relevant requirements and queries from EMSE researchers with data from more than 40 EMSE experiments. The initial evaluation shows that SKET can help to effectively and efficiently find detailed

---

[2] http://www.ncbi.nlm.nih.gov/mesh

[3] http://cdlflex.org/prototypes/ske/

[4] http://cdlflex.org/prototypes/ske/theory/

experimental data for meta-analysis or reuse purposes, thus, improving on the functionalities offered by mainstream PDs.

The remainder of this paper is organized as follows. Section 2 explains the EMSE domain and clarifies the typical users and queries that SKET supports. Section 3 describes the design of the SKET tool. Section 4 describes the concrete SKET prototypes and their initial evaluation. Section 5 summarizes related work. Section 6 concludes and discusses the future work.

2 The Empirical Software Engineering Domain and its Requirements

This section introduces the Empirical Software Engineering (EMSE) domain, user roles, and typical questions as context for designing and evaluating a tailored Publication Database (PD).

2.1 Empirical Software Engineering Research

Researchers in EMSE collaborate on research topics, such as software quality assurance, to build up a body of knowledge (BoK). An EMSE BoK includes theory models [7], hypotheses derived from the theory models, and results from empirical studies that test those hypotheses [8], to explain and/or predict EMSE phenomena.

A major goal of researchers in EMSE is to investigate the effects of software engineering concepts, methods, and tools, e.g., whether a new method on software inspection performs differently from established methods, and the influence of factors, such as the level of expertise of the users of a software engineering method. The researchers design and conduct experiments in controlled environments, case studies in academic and industrial environments, and surveys to collect data for analysis from literature and from domain experts. The analysis results of the collected data help to provide practitioners with information on the strengths and limitations of methods and tools as well as on success and risk factors in order to support their informed decision on which methods and tools best to use in and adapt to their software engineering environment. In addition, researchers test hypotheses to develop theories on the effects of software engineering concepts, methods, and tools and drive the planning of the most relevant future experiments, case, studies, and surveys.

2.2 User Groups in the EMSE Domain

In our previous research [10], we distinguished between three types of target users for ontology-based system, which fits well into EMSE domain: (1) casual users or lay users, (2) domain experts, and (3) ontology / knowledge experts. Lay users consist of entry-level EMSE researchers and students. These users typically have little knowledge about the domain, so they need more explanation about the domain and its concepts. Domain experts are EMSE experts and meta-researchers in the area. They typically have a deeper knowledge about the area and require a system with complex query capabilities. Knowledge expert represent maintainer of the domain knowledge. The challenge here is to provide casual users and domain expert, which we assume as the typical users, with a simple yet powerful approach to query the EMSE research data. Therefore, the user interface will be designed with these users as primary target.

2.3 Typical Questions of EMSE Researchers

EMSE researchers typically use a PD to conduct three major tasks: Meta analysis research, replication research, and research network analysis.

Meta analysis research is conducted by posing a set of questions that focus on the aggregation of research results analysis, for example, "*Which are the hypotheses investigated in experiments on <inspection method Perspective Based Reading> and <theory construct 'effectiveness'> and its synonyms*?"

Replication research questions aim at retrieving information that could help users to replicate an experiment. While this kind of information could be retrieved from reading the whole paper, we are aiming to simplify this process by providing users with the predefined queries with parameters to reduce the number of papers to be analyzed. One example question here is "*Which are the reported findings of experiments on <a certain inspection method>?*"

Research network analysis focuses on a community-level understanding of the EMSE research, through questions such as "*Which research groups are working on topics with response variables similar to <a certain domain concept>?*"

3 The Systematic Knowledge Engineering Tool (SKET)

The Systematic Knowledge Engineering Tool was created as part of a toolset that supports the Systematic Knowledge Engineering (SKE) process approach [1], which aims at storing the data results of a Systematic Literature Review (SLR) [3] for incrementally building up contributions to a specific scientific body of knowledge (BoK) in the context of Empirical Software Engineering (EMSE). The SKE process consists of four stages:

(1) Planning an EMSE BoK Knowledge Base (KB) creation,
(2) Conducting data extraction from published research reports,
(3) Creating/updating the EMSE BoK KB, and
(4) Data analysis and processing.

SKET automates a significant part of the SKE process, mainly in the third and fourth stages, which will be explained in the Section 3.1 to 3.3. The SKET workflows and module implementation will be explained in Section 3.4.

3.1 Data Input

The data input to SKET consists of data extracted from EMSE research publications. These research publications typically contain implicit and explicit information related to the experiments conducted as the basis for the publication. Several typical data elements extracted from the experiments are: *hypotheses, threats to validity, measurements, and experiment results*.

EMSE domain experts extract experiment related information from these publications as part of the SKE process and store this information in a predefined but customizable data format, such as excel spreadsheets. EMSE experts prefer spreadsheets as the main data storage format due to their easy handling during data collection and verification. Therefore SKET accepts spreadsheets as its primary input data format, while being designed to support a wide variety of data import formats from a range of heterogeneous data sources.

3.2 Data Storage

As foundation for the storing process of the raw data, we designed an ontology model based on the EMSE domain context. The high level abstraction of the data model is depicted in Figure 1.

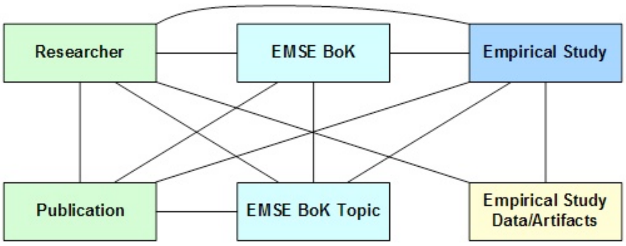


Figure 1. Major areas of the EMSE data model

Figure 1 illustrates that researchers and publications contribute the source information, which contains the typical metadata of a publication database (PD), e.g., publication title, authors, and publication events. The light blue parts of the figure depict the data elements relevant for the EMSE BoK and BoK topics. Typical information in this area is the categorization of research inside their respective BoK, e.g., an EMSE BoK may be “Software Quality research” and a specific BoK topic “software inspection”. Empirical study information (dark blue) and empirical study data/artifacts (yellow) consist of detailed, experiment-level information, which is typical for SKET and therefore sets the tool apart from mainstream generic PDs.

The data model used for storing the EMSE information is depicted in Figure 2. The main concepts in this data model are: (1) *Publication*, which holds the common bibliographic metadata, and (2) *Experiment*, which glues together all concepts and properties related to the setup of the empirical research experiment, and (3) *Experiment Run*, which stores the experiment execution details. Currently the ontology data model is not yet re-using any existing ontology, e.g., publication ontology[5] and PRO[6] (Publishing Roles Ontology). Such an alignment is part of future work.

---

[5] http://ebiquity.umbc.edu/ontology/publication.owl

[6] http://purl.org/spar/pro

The storage of the publication metadata extracted by experts into the ontology is currently performed automatically based on a manually crafted mapping between the two data structures [7].

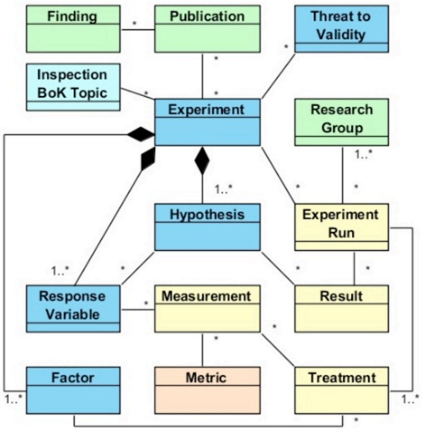
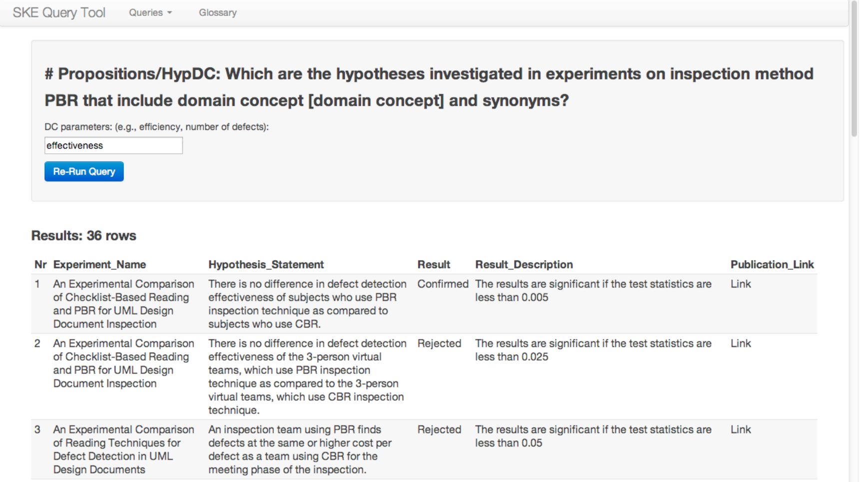


Figure 2. SKET KB data model[8] overview

3.3 Querying and Result Visualization

To enable both lay-users and domain experts to use SKET, we developed a user-friendly interface for the tool as shown in Figure 3. To access the queries, users click the *Queries* tab in the top-left part of the SKET prototype page and choose the desired query. The list of queries can be further extended based on user requests.



 # Propositions/HypDC: Which are the hypotheses investigated in experiments on inspection method PBR that include domain concept [domain concept] and synonyms?

DC parameters: (e.g., efficiency, number of defects): effectiveness

Results: 36 rows

Nr	Experiment_Name	Hypothesis_Statement	Result	Result_Description	Publication_Link
1	An Experimental Comparison of Checklist-Based Reading and PBR for UML Design Document Inspection	There is no difference in defect detection effectiveness of subjects who use PBR inspection technique as compared to subjects who use CBR.	Confirmed	The results are significant if the test statistics are less than 0.005	Link
2	An Experimental Comparison of Checklist-Based Reading and PBR for UML Design Document Inspection	There is no difference in defect detection effectiveness of the 3-person virtual teams, which use PBR inspection technique as compared to the 3-person virtual teams, which use CBR inspection technique.	Rejected	The results are significant if the test statistics are less than 0.025	Link
3	An Experimental Comparison of Reading Techniques for Defect Detection in UML Design Documents	An inspection team using PBR finds defects at the same or higher cost per defect as a team using CBR for the meeting phase of the inspection.	Rejected	The results are significant if the test statistics are less than 0.05	Link

Figure 3. SKET prototype screenshot

---

[7] SKET mapping sheet: http://juang.me/documents/public/mapping.xlsx

[8] SKET ontology data model: http://juang.me/documents/public/sket.owl

This user interface mainly consists of a landing page, which explains the contexts of the domain problem, a set of predefined SPARQL queries that reflect the main EMSE expert questions, and a link to a glossary[9] of terms related to the domain (in this case the EMSE BoK of software inspection). The result of queries is visualized in a tabular view, which is simple yet sufficient to cater for the needs of the users and can be extended with a graphical visualization of complex relationships.

3.4 SKET Workflow and Modules

The SKET workflow and modules are shown in Figure 4. The arrows show process steps within SKET and its interaction with users. SKET consists of three modules, which corresponds with the number in the figure.

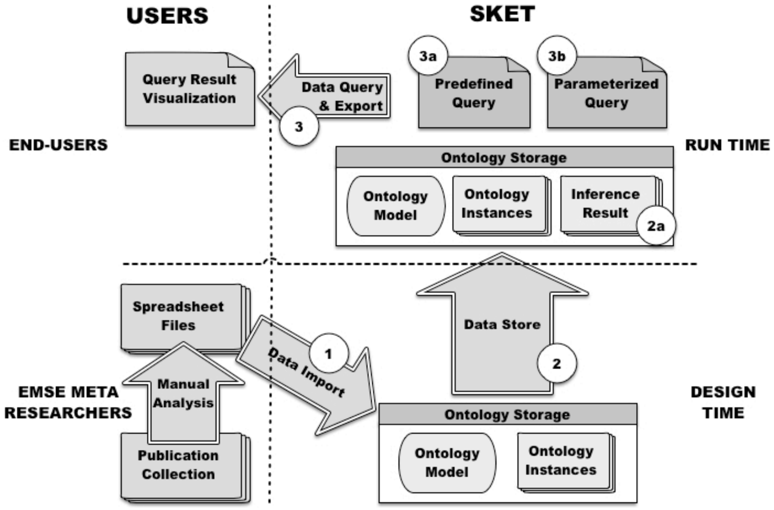


Figure 4. SKET modules and workflow

1. SKET data import module (DIM). To enable the data import from the meta-researchers, we provided a spreadsheet import tool to convert their gathered data into ontology-based metadata. There are already several open source implementations of spreadsheet data importers, see, e.g., the tool survey in Kovalenko et al. [11]. Unfortunately, these tools do not allow customizing the behavior for importing the spreadsheet data efficiently. Therefore, we developed a flexible importer, the *XlsxToOwl*[10], which enables to easily define mappings between spreadsheets and ontology. The tool was created using the *Apache Jena*[11] and *Apache POI*[12] libraries. Further, it also supports the customization of an ontology model based on parameters given in the

---

[9] The glossary is an external system integrated with the SKET design.

[10] https://github.com/fajarjuang/XlsxToOwl

[11] http://jena.apache.org/

[12] http://poi.apache.org/

spreadsheet file. This enables building a flexible data model based on a given spreadsheet from domain expert.

2. SKET data storage module (DSM). The DSM enables the usage of ontology data within our environment. For this module, the availability of an inference engine is very important, since it will enable inference result enhancement (2a) based on predefined rules, implemented as generic Jena rules in the current system.

3. SKET ontology-querying tool (OQT). The OQT provides the user interface for lay users and domain experts. We already proposed a framework for analyzing OQT for different kinds of users [10]. However, for the current implementation, we only implemented a simple interface that provides users with a set of predefined queries (3a) with optional parameters (3b) based on domain-experts request, with result visualization in a tabular format. Future work will explore more sophisticated user interfaces based on forms or natural language question answering.

4 SKET Prototypes and their Evaluation

For evaluation, we tested SKET with empirical software engineering experts. Figure 5 shows how SKET addresses the challenges posed in the EMSE domain by introducing a Knowledge Base (KB) and the role of a knowledge engineer (KE). In this context, researchers extract data from EMSE studies published in digital libraries (depicted with (1) in Figure 5) and the KE integrates the extracted data into the SKET Publication Database KB. Therefore, the knowledge collected is available for semantic querying also to the general readership, including other researchers and practitioners.

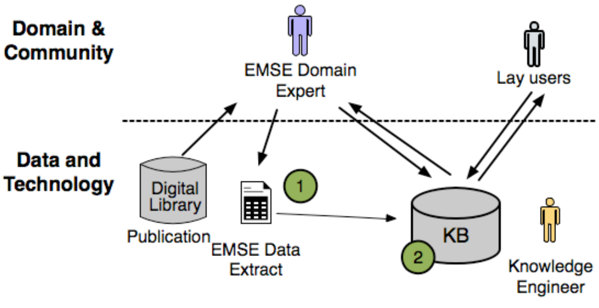


Figure 5. EMSE stakeholder and technology with SKET

To identify the most relevant query candidates to be answered in the context of an EMSE Body of Knowledge (BoK) on software inspection, we focused on EMSE researchers as main stakeholders, who conduct meta-analyses on study reports or conduct empirical studies and need to be aware of relevant research in their area. Based on an informal survey with software inspection researchers in six research groups (located in Austria, Brazil, Chile, Ecuador, and Spain), we identified a set of query candidates.

For the evaluation of SKET we observed two concrete instances of the SKET prototype: SKET prototype P1[3] for general EMSE BoK domain [1] and SKET prototype P2[4] for theory construct identification in EMSE [2]. The structure of the underlying data model for both prototypes is very similar; however, their query sets are different since they have been developed to address different goals: P1 for providing an overview on experiment reports in EMSE BoK based on EMSE domain concepts (e.g., hypothesis, experiment) and P2 for identifying theory constructs in a focused EMSE BoK topic (e.g., cost of quality, defect detection).

P1 contains data from the 30 most recent papers in the specific area of software inspection in EMSE out of 102 total papers that were found in the area in a Systematic Literature Review (SLR) [3] study covering a specific publication time window, plus 2 papers that the domain expert used for testing the extraction process. P2 added information from 10 more recent papers into SKET about a specific part of the software inspection area, called Perspective-Based Reading.

Table 1. SKET prototypes summary

Prototype	**P1 (Original)**	**P2 (Addition)**	**Total**
Number of publications	32	10	42
Size of semantic data (triples)	13,292	6,228	19,520
Number of domain experts involved	10	1	10
Effort for publication analysis and data extraction (in person hours)	80	25	105

Since the data from P1 and P2 could be merged together without affecting the result of the queries, we decided to update P1 and P2 ontology with the aggregated version of extracted data from both. Hence, the current versions of the prototypes P1 and P2 use the same ontology. Table 1 provides more detailed information on both prototypes and the effort for publication analysis. The size of the semantic data shows the growing data instance within the ontology, while effort for publication analysis and data extraction reflect the effort needed by experts within the approach.

Initial evaluations of processes using P1 and P2 show that SKET can successfully answer the queries of the researchers in the EMSE domain. However, we are aware that these initial evaluations can provide only limited insight and a future larger scale evaluation is likely to bring up new needs regarding the usability and usefulness of SKET.

The SKET evaluation showed the feasibility of building a software inspection EMSE PD from controlled experiments. A major result of the SKET evaluation was that the effort for data extraction and storing is similar to the effort for data extraction in a traditional SLR process. Therefore, the benefits of SKET can be achieved with little extra cost for data extraction. The resulting KB may be used as input to a future SLR protocol to identify relevant research on a specific topic or for exploring semantic search facilities usually not available in current PDs. SLR extraction sheets, on the other hand, can be used as input for extending the knowledge contained in the SKET.

The SKET prototypes support users in locating empirical evidence and reusing it according to their specific needs. However, it is important to state that support for meta-analyses and for applying specific research syntheses methods are beyond the scope of SKET, which provides exports of relevant data as input to user-specific tool chains for advanced data analysis. Moreover, the relevant queries of our evaluation were chosen focusing on a survey with a specific type of stakeholder, the EMSE researchers in software inspection. Other stakeholders may have different needs.

The straightforward approach for data importing and querying enables researchers to contribute towards building up additional knowledge, and to query for knowledge with low effort. Using ontologies facilitates extensions of the underlying KB common data model and semantic search. The querying capabilities were found efficient in the evaluation on answering the EMSE researchers' most relevant queries.

The main lessons learned relate to success factors for SKET. A major success factor is properly involving a knowledge engineer. The overhead of this new role is likely to be offset soon by benefits obtained by the established KB. Another success factor is getting the EMSE research community involved for long-term collection and use of data. Therefore, incentives and benefits of contributing should be clarified. More detail regarding these evaluation could be found in [1][2].

5 Related Work

The use of Semantic Web technologies for representing, publishing and making sense of research related data is a broad and active field of research. *CS AKTive space*, for example, was one of the first projects, which facilitated access to research data by using semantic techniques [12]. The project gathered large amounts of data through scraping from the web-pages of computer science departments in the UK, represented this data in terms of a set of ontologies and then developed a visual interface for interacting with the RDF data and understanding UK's Computer Science research landscape in terms of the main researchers, their topics, publications, grants as well as geo location. The *Flink* system [13] gathered social network data from web pages and FOAF profiles and allowed the analysis of the social structure of the Semantic Web research community. The recent *Rexplore* [4] system, focuses on facilitating "sensemaking" of research data extracted from a variety of bibliographic sources, which go beyond the keyword-based search offered by most Web-based bibliographic systems (e.g., Google Scholar, Scopus[1]). Sensemaking tasks include (i) extracting and monitoring the various research areas in a field and their evolution; (ii) detecting "semantically" related researchers, which work on the same topics or have similar research trajectories; and (iii) fine-grained expert search. Making sense of large publication collections has also been a topic of works that do not rely on semantic technologies. For example, the *Action Science Explore* (ASE) tool [5] uses statistics, text analysis, and visualization techniques to provide a rapid insight into publication collections in terms of the main authors, key papers, controversies, and hypotheses.

Another line of research focuses on using semantic web technologies to represent and make use of experimental data, including experimental workflows [14] or actual research data in experimentally heavy natural science domains such as biology, biomedicine, and chemistry [15].

Our work focuses on research data from academic publications, as many of the works above. However, we are interested in representing fine-grained knowledge in terms of used research data and results in the EMSE domain, which brings specific challenges as discussed, and, to our knowledge, has not yet been considered as an application area for semantic publishing.

6 Conclusions and Future Work

In this paper we introduced the *Systematic Knowledge Engineering Tool* (SKET), which supports building publication databases (PDs) from empirical studies. SKET supports the Systematic Knowledge Engineering process and provides a Knowledge Base (KB) as storage for the extracted data. SKET allows the research community to find gathered knowledge and reuse it with semantic search facilities, as building blocks for a variety of analyses. An important aspect of SKET is to provide high-quality data, which is precise and correct within the scope of the target research area. Therefore, humans need to be involved in quality assurance of the imported data and to provide semantically correct relationships between data elements.

SKET was evaluated by building a PD for the software inspection topic from the knowledge acquired through controlled experiments. Information was extracted from more than 40 research papers and integrated into the KB. The resulting PD on software inspection is available online being implemented in two prototype systems[3,4]. The main evaluation results were:

- Data extraction of inspection experiments into spreadsheets based on the common data model was successful. However, support for flexible mapping should be provided to reuse the data model for different research communities.
- The SKET KB was effective and efficient in answering the most relevant stakeholder queries. The standardized query support and the flexible data model provided a stable background for this result. However, a better data storage should be provided for better scalability.
- SKET enables knowledge reuse (by applying queries) for analysis and meta-analysis purposes. Moreover, new knowledge, i.e., new data from literature, can be included in the KB as a foundation for a growing EMSE PD.

SKET showed promising results in the software inspection context and should also be evaluated in other contexts. The overall effort of data extraction and import to SKET is comparable to the effort of conducting traditional systematic literature reviews (SLRs) and, for the specific purpose of building an EMSE PD, several advantages of using SKET can be identified, such as analysis of specific topic from EMSE research area and EMSE theory identification support.

As future work we propose: (a) investigating additional EMSE research stakeholder needs; (b) extending the set of empirical studies on software inspections in the SKET KB; (c) instantiating SKET in other research domains beyond EMSE; (d) extending SKET with a platform to allow building on the collective intelligence of the research community for quality assurance and recommendation; (e) Ontology Querying Tool extension to provide users a graphical visualization of complex relationships; (f) reuse of currently existing publication ontologies and (g) providing support for data instance versioning and model evolution for the SKET back-end.

Acknowledgement

This work was supported by the Christian Doppler Forschungsgesellschaft, the Federal Ministry of Economy, Family and Youth, Österreichischer Austauschdienst (ÖAD) and the National Foundation for Research, Technology and Development - Austria. We also wanted to thank the reviewers for their valuable inputs to the paper.

References

1. Biffl, S., Kalinowski, M., Ekaputra, F.J., Serral, E., Winkler, D.: Building Empirical Software Engineering Bodies of Knowledge with Systematic Knowledge Engineering. (2013). Technical Report IFS-CDL 13-03, TU Vienna, Austria, October 2013. Online available at http://qse.ifs.tuwien.ac.at/publication/IFS-CDL-13-03.pdf
2. Biffl, S., Kalinowski, M., Ekaputra, F.J., Martins, G.G., Winkler, D.: Towards Efficient Support for Theory Identification from Published Experimental Research. (2014). Technical Report IFS-CDL 14-01, Vienna University of Technology, Austria, January 2014. Online available at http://qse.ifs.tuwien.ac.at/publication/IFS-CDL-14-01.pdf
3. Keele, S.: Guidelines for performing Systematic Literature Reviews in Software Engineering. (2007).
4. Osborne, F., Motta, E., Mulholland, P.: Exploring Scholarly Data with Rexplore. Semant. Web–ISWC 2013. 460–477 (2013).
5. Dunne, C., Shneiderman, B., Gove, R., Klavans, J., Dorr, B.: Rapid understanding of scientific paper collections: Integrating statistics, text analytics, and visualization. JASIST. 63, 2351–2369 (2012).
6. Chandrasekaran, B., Josephson, J.R., Benjamins, V.R., Others: What are ontologies, and why do we need them? IEEE Intell. Syst. (1999).
7. Sjøberg, D.I.K., Dybå, T., Anda, B.C.D., Hannay, J.E.: Building Theories in Software Engineering. In: Shull, F., Singer, J., and Sjøberg, D.K. (eds.) Guide to Advanced Empirical Software Engineering SE - 12. pp. 312–336. Springer London (2008).
8. Wohlin, C., Runeson, P., Höst, M., Ohlsson, M.C., Regnell, B., Wesslén, A.: Experimentation in Software Engineering. Springer. (2012).
9. Brereton, P., Kitchenham, B.A., Budgen, D., Turner, M., Khalil, M.: Lessons from applying the systematic literature review process within the software engineering domain. J. Syst. Softw. 80, 571–583 (2007).
10. Ekaputra, F.J., Serral, E., Winkler, D., Biffl, S.: An Analysis Framework for Ontology Querying Tools. I-SEMANTICS '13. p. 1. ACM Press, New York, USA (2013).
11. Kovalenko, O., Serral, E., Biffl, S.: Towards evaluation and comparison of tools for ontology population from spreadsheet data. I-SEMANTICS '13. p. 57. ACM Press, New York, USA (2013).
12. Shadbolt, N.R., Gibbins, N., Glaser, H., Harris, S., others: CS AKTive Space or how we stopped worrying and learned to love the Semantic Web. IEEE IS 19, 41. (2004).
13. Mika, P.: Flink: Semantic web technology for the extraction and analysis of social networks. JWS. Services and Agents on the World Wide Web. 3, 211–223 (2005).
14. Corcho, Ó., Garijo Verdejo, D., Belhajjame, K., Zhao, J., Missier, P., Newman, D., Palma, R., Bechhofer, S., a Cuesta, E., Gómez-Pérez, J.M., others: Workflow-centric research objects: First class citizens in scholarly discourse. SePublica 2012 (2012).
15. Wiljes, C., Cimiano, P.: Linked data for the natural sciences: Two use cases in chemistry and biology. SePublica 2012 (2012).