Building Empirical Software Engineering Bodies of Knowledge with Systematic Knowledge Engineering

Stefan Biffl¹ Marcos Kalinowski² Fajar Ekaputra¹
Estefania Serral³ Dietmar Winkler¹

¹Vienna University of Technology, Austria
²Fluminense Federal University, Brazil
³KU Leuven, Belgium

² kalinowski@acm.org
Agenda

- I. Introduction
- II. Research Issues
- III. Systematic Knowledge Engineering
- IV. Evaluation Results
- V. Conclusions
Empirical Software Engineering (EMSE) researchers conduct Systematic Literature Reviews (SLRs) to build Bodies of Knowledge (BoKs).

An EMSE BoK includes theory models, hypotheses derived from the theory models, and results from empirical studies that investigate those hypotheses, to explain and/or predict phenomena.

SLRs have become a widely used and reliable research method, but:

- SLR reports often provide only discussions on selected research questions.
- Knowledge collected during the SLR process is usually not publicly available for future analyses and extensions.
Introduction

- Context and challenges:
  - Data extracted during SLRs usually stays in a local archive.
  - There is seldom a way for other researchers to access the extracted data to apply different analyses and research synthesis methods, or to extend the data in the BoK.
Introduction

- SE requires a “community supported living experience base” (Basili, 2013).

- We introduce the Systematic Knowledge Engineering (SKE) process to support the community in efficiently building BoKs from empirical studies.
  - SKE is based on the SLR process and on Knowledge Engineering (KE) practices.
  - Provides a Knowledge Base (KB) with semantic technologies that enable reusing intermediate data extraction results and querying of empirical evidence.

Research Issues

Solution concept:

- Researchers extract data from empirical studies published in digital libraries and have it integrated into the KB by the knowledge engineer, enabling incremental content extensions.
- The collected knowledge is available for semantic querying from the KB, enabling to apply different analyses.
Research Issues

- **RI-1: SKE Requirements Analysis.** Which queries to knowledge on empirical studies are most relevant to EMSE BoK stakeholders?

- **RI-2: SKE Process & Data Modeling.** How can the traditional SLR process be adapted to support incrementally building EMSE BoKs? Which data elements are necessary to address the most relevant queries of EMSE BoK researchers?

- **RI-3: SKE Tool Support.** Which functions are necessary to automate key steps in the SKE process, i.e., efficient data integration and querying?
RI-1: SKE Requirement Analysis

- Relevant stakeholder queries:
  - Q1: Which inspection methods were effective (or efficient) in finding defects in requirements artifacts?
  - Q2: What are the results of experiments that report on a given BoK Topic Parameter <BTP>, e.g., inspection method PBR?
  - Q3: Which experiments were conducted with the response variable <RV>, e.g., number of defects?
  - Q4: Which hypotheses include the domain concept <DC> (and its synonyms), e.g., effectiveness?
  - Q5: Which synonyms have been used for domain concept <DC>, e.g., effectiveness?
  - Q6: Who are researchers working on topics with response variables in their experiments similar to the domain concept <DC>, e.g., efficiency?

- Gathered based on a survey with software inspection EMSE BoK researchers in six research groups (located in Austria, Brazil, Chile, Ecuador, and Spain).
RI-2: SKE Process & Data Model

- Process:

1. Planning the Review
   - Ident. the need for a review
   - Commissioning a review
   - Specifying the research question(s)
   - Developing the review protocol

2. Conducting the Review
   - Identification of research
   - Selection of primary studies
   - Study quality assessment
   - Data extraction and monitoring

3. Reporting the Review
   - Specifying dissemination
   - Formatting main report
   - Evaluating the report

Systematic Knowledge Engineering

1. Planning BoK Creation
   - Ident. the need for the BoK
   - Commissioning BoK creation
   - Specifying the BoK topic and evidence type
   - Developing the protocol

2. Designing the BoK KB
   - Designing the KB common data model

3. Conducting Data Extraction
   - Identification of research
   - Selection of primary studies
   - Study quality assessment
   - Data extraction and monitoring

4. Populating the BoK KB
   - Integrating data extraction into KB
   - Providing KB query facilities

Data Synthesis + Reporting the Review
Systematic Knowledge Engineering

- RI-2: SKE Process & Data Model
  - Data Model:
Systematic Knowledge Engineering

- RI-2: SKE Process & Data Model
  - Data Model:
RI-2: SKE Process & Data Model

Data Model:

The designed model allows querying knowledge acquired from experimental studies:

- Hypotheses of experiments related to specific BoK Topics (or their synonyms).
- Results for each hypothesis in the available experiment runs (confirmed/rejected) and information on their statistical confidence.
- Measurements that led to each of those results.
- ...
RI-3: SKE Tool Support

- The KB was implemented as an ontology using the Protégé framework.
- Tool support:
  - Spreadsheet data contribution interface.
    - Automated in Java by using a spreadsheet reader library (Apache POI) and an ontology library (Apache Jena).
    - Enables integration of partial and heterogeneous data by applying the Interchange Standard Approach.
  - Web prototype for querying.
    - Queries are implemented using SPARQL query language.
    - The knowledge engineer enhanced the KB by implementing semantic search functions (e.g., searching on domain concepts, their synonyms and related concepts).
  - A glossary tool.
    - Facilitates identifying and defining domain concepts.

Available online http://cdlflex.org/prototypes/ske
Evaluation Results

- **Evaluation concept:**
  - We applied and measured the SKE process steps with tool support to build a software inspection EMSE BoK KB based on knowledge acquired from controlled experiments.
  - Information was extracted from 30 systematically identified research papers and integrated into the KB.
  - While the authors of this paper provided the SKE process and data model design, the data extraction was conducted by an independent expert team.
Evaluation Results

- **Process feasibility:**
  - The software inspection EMSE BoK KB was successfully built following the SKE process.
  - Accurate and semantically enhanced query results could be obtained.
    - Queries Q2 to Q6 could be directly formulated, listing experiment results, experiments, hypotheses, synonyms, and research groups.
    - **Query Q1:** “Which inspection methods were effective (or efficient) in finding defects in requirements artifacts?”
      - Strategy: Identify experiments that report on effectiveness or efficiency (or synonyms) in their hypotheses or response variables and that were related to BoK Topics associated to inspection methods and to the artifact type requirements.
        - Q1.1 List their hypotheses and results in all experiment runs.
        - Q1.2 List the findings of the papers related to them.
Evaluation Results

- Process feasibility:

<table>
<thead>
<tr>
<th>Hypothesis Text</th>
<th>Hypothesis Result</th>
<th>Experiment Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Defects detected by PBR are more evenly distributed over the whole SRS document than those detected by checklist.</td>
<td>Confirmed</td>
<td>PBR versus Checklist: A Replication in the N-Fold Inspection Context</td>
</tr>
<tr>
<td>Increased Effectiveness Regarding Major Defects: Teams using GRIP find a higher number of major reference defects than teams following a manual inspection process.</td>
<td>Rejected</td>
<td>Effect of groupware for software inspection.</td>
</tr>
<tr>
<td>Individual reviewers using PBR and Checklist find different defects.</td>
<td>Partly</td>
<td>Perspective-Based Reading: A Replicated Experiment Focused on Individual Reviewer Effectiveness</td>
</tr>
</tbody>
</table>
Evaluation Results

- Process feasibility:

  http://cdlflex.org/prototypes/ske
Evaluation Results

- Process effort:

<table>
<thead>
<tr>
<th>SKE Effort (person hrs)</th>
<th>Effort Description</th>
<th>SKE vs SLR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planning EMSE BoK Creation</td>
<td>Building the protocol.</td>
<td>↓ (-10% to -5%)</td>
</tr>
<tr>
<td>Conducting Data Extraction</td>
<td>Filtering primary studies and data extraction.</td>
<td>← (-5% to +5%)</td>
</tr>
<tr>
<td>Creating EMSE BoK KB (Population)</td>
<td>KB data integration (automated).</td>
<td>N/A</td>
</tr>
</tbody>
</table>

- Since SLRs are widely used, we ascertain that, given the similar effort, it makes sense to apply SKE for building EMSE BoKs.
Evaluation Results

Process added value:

- SKE integrates extracted data into a KB and facilitates the reuse of the aggregated knowledge by other researchers according to their specific goals.
- SKE facilitates building up knowledge incrementally by integrating new extracted data into the KB.
- The resulting KB allows exploring the gathered empirical evidence using semantic search capabilities that cannot be performed on SLR reports.
  - Effective in answering the relevant queries, enabling knowledge reuse for analyses beyond the results from the SLR process.
Conclusions

- In this paper we introduced the Systematic Knowledge Engineering (SKE) process.
  - Supports the research community in building up EMSE BoKs from empirical studies.
  - Builds on the Systematic Literature Review (SLR) process and Knowledge Engineering (KE) practices.

- By decoupling data extraction from data synthesis and providing a KB, SKE allows the community to extend gathered knowledge and reusing it with semantic search facilities, as building blocks for a variety of analyses.

- SKE showed promising results in the software inspection context and should also be evaluated in other contexts.
Conclusions

- In this paper we introduced the Systematic Knowledge Engineering (SKE) process.
  - Supports the research community in building up EMSE BoKs from empirical studies.
  - Builds on the Systematic Literature Review (SLR) process and Knowledge Engineering (KE) practices.

- By decoupling data extraction from data synthesis and providing a KB, SKE allows the community to extend gathered knowledge and reusing it with semantic search facilities, as building blocks for a variety of analyses.

- SKE showed promising results in the software inspection context and should also be evaluated in other contexts.
Conclusions

- **SKE is being evaluated in other contexts.**
  - Software Product Lines
  - Threats to Validity in Controlled Experiments
  - Software Maintenance

Building Empirical Software Engineering Bodies of Knowledge with Systematic Knowledge Engineering

Stefan Biffl\textsuperscript{1} Marcos Kalinowski\textsuperscript{2} Fajar Ekaputra\textsuperscript{1}
Estefania Serral\textsuperscript{3} Dietmar Winkler\textsuperscript{1}

\textsuperscript{1}Vienna University of Technology, Austria
\textsuperscript{2}Fluminense Federal University, Brazil
\textsuperscript{3}KU Leuven, Belgium