Identifying and Classifying Approaches for Sharing Architecture Decisions

A report on how to plan and conduct a systematic literature review in the field of software engineering.

Rayan Brouwer & Sophie Hugenholtz

8-7-2016

Version 6

Revision History

Version	Date	Author	Description
1	19/5/2016	Sophie	Outline of the thesis
2	6/6/2016	Rayan	Start writing on Introduction, Method search strategy, results preliminary search
3	16/6/2016	Sophie Rayan	Rewriting introduction, research questions, refining search strategy and results preliminary search Refining search strategy, adding QGS, evaluation of search string, start on results selection criteria
4	30/6/2016	Sophie Rayan	Write Quality assessment method + revise search strategy and selection criteria + revise results preliminary Refining parts in the Method section, writing about data extraction strategy
5	6/07/2016	Rayan Sophie	Refine structure of thesis, rewriting introduction, writing Research method and Results section. Review everything, write validation, intro, background and evaluation
6	8/7/2016	Sophie Rayan	Finalizing thesis Finalizing thesis

Inhoud

1 Introduction

A literature review is an important type of research study, used mainly for summarizing existing knowledge or identifying unexplored subjects [1]. Such a review is systematic when performed in an unbiased and replicable way, and when making use of a review protocol. Several different approaches to writing (parts of) this protocol have been published, some of which containing parts contradicting with others. This study therefore looks into how a protocol to such a systematic literature review (SLR) is constructed.

As an SLR should be conducted replicable and unbiased, a review protocol should explain exactly and in detail how every step of the SLR should be taken. The protocol should contain many specific sections, but this paper will mainly focus on some of the most important parts. These are the search strategy, the selection criteria, the quality assessment criteria, and the data extraction strategy. The result of this study is a protocol (cf. Appendix E) that can be used for conducting an SLR on identifying approaches to architectural knowledge sharing (AKS). Architectural knowledge can be defined as the sum of design and design decisions, and AKS as capturing, documenting and communicating architectural knowledge.

This paper looks into the construction of a review protocol, it discusses the resulting protocol, and validates this result through a pilot study. This paper and the resulting protocol presents, among other things, an optimal search strategy that aims at identifying all studies relevant to the SLR. This strategy includes an objective approach for evaluating searches.

The remainder of this paper is structured as follows: some background and necessary information on the subject is given in Section 2, the methodology and approach of the research is specified in Section 3, the results are presented in Section 4, the pilot study to validate the results is described and discussed in Section 5, all results are discussed and evaluated in Section 6 and in Section 7, the conclusions are presented, along with some suggestions for future work.[1]

2 Background

This section goes into more detail about specific parts important to this study, the protocol or the SLR. It gives explanations of what an SLR and review protocol are exactly and gives some information about related work. This last section does not discuss any specific papers, but does give an explanation of what is out there, and some brief background on constructions of review protocols. Section 2.1 explains what an SLR is, including its objective, Section 2.2 describes the function and objective of a review protocol, and Section2.3 gives information about related work.

2.1 Systematic Literature Review

Literature reviews are secondary studies into existing knowledge and previously published articles with the purpose of gaining more insight or obtaining new knowledge about the studied subject. Data, results and findings from several studies are combined to be able to draw new conclusions. If researchers start searching for studies without any guidelines, selecting relevant articles along the way, the selection of articles is prone to researcher bias. It is tempting to select only studies beneficiary to the conclusions the researcher wants to draw, leaving possible contradicting articles out. Even when researchers have no intention of doing this, researcher bias is always likely without guidelines. To this end, Kitchenham [1] wrote a document with guidelines for performing an SLR. These describe in detail

[1] All sections of this protocol were written by the two writers of this document together, except for Section 4, the results, and Section 5, the validation. Section 4 was mainly written by Rayan Brouwer, except for one subsection, and Section 5 was mainly written by Sophie Hugenholtz.

how and SLR should be performed, the importance of every step described, and necessary preparatory work that needs to be (largely) completed before conducting the actual SLR.

2.2 Review Protocol

The guidelines by Kitchenham [1], mentioned in the previous section, suggest and explain the importance of the use of a review protocol. Such a protocol describes in detail all the necessary steps for conducting an SLR along with a description of how those steps should be taken. When conducting an SLR it is important to precisely follow the protocol or adjust the protocol whenever this turns out to be impossible, ineffective or incorrect. Writing a review protocol is an iterative process, where all methods and processes are carefully tested several times along the way. Also, a review protocol is likely to be adjusted and refined as the SLR itself is being conducted.
A review protocol specifies how to conduct an SLR and the guidelines [1] suggest using the following sections:

- Background
  A section giving insight in the background of the subject of the SLR. For instance, related work, a brief history, definitions, challenges in the field, etc.
- Research Questions
  The title of this section speaks for itself. It states the research questions for the SLR, along with an explanation of the question, why it is asked, and what the question aims for.
- Search Strategy
  This section specifies in detail how relevant studies for the SLR are going to be searched and found. This detailed explanation is necessary for a replicable and transparent SLR.
- Study Selection Criteria
  As the search strategy is likely to result in both relevant and irrelevant studies, selection criteria are defined in this section.
- Study Selection Procedures
  In this section a process stating how the selection criteria from the previous section should be applied is specified.
- Study Quality Assessment
  This section gives means by which the empirical quality of relevant studies can be assessed, along with an explanation of this quality assessment.
- Data Extraction Strategy
  A careful description of how relevant data from the studies should be extracted is stated in this section.
- Synthesis of the Extracted Data
  In this section, one describes how the extracted data should be analysed.
- Dissemination Strategy
  This section specifies how the results of the analysed data are being reported, as well as how the entire SLR should be reported.
- Project Timetable
  As the title suggests, this section contains a planning for conducting the SLR, including when to start with what part of the research.

The last three sections of the list above are not in the scope of this paper and can therefore not be found in the resulting review protocol. The created protocol includes next to the sections mentioned above, an introduction to the subject and the rationale behind the SLR, and a revision history. This paper will focus on the construction of a Search Strategy, the Study Selection Criteria and Procedures, the Study Quality Assessment and the Data Extraction Strategy.

2.3 Related work

Many protocols for SLRs have been written already, but not many studies to the construction of specific protocols have been done. Protocols are often written, without a paper specifying and justifying how and why it was written the way it is. There do exist studies to SLR's in software engineering in general. For instance, Zhang et al. [2] conducted an empirical study to how well SLRs have been adopted within the field, Brereton et al. [3] investigated challenges when performing SLRs and presented possible solutions, MacDonnell et al. [4] investigated how reliable the SLRs within the field of software engineering generally are, and Kitchenham et al. [5] conducted a tertiary study towards the current practice, or state of the art, of SLRs in software engineering.
There exist several papers on how to write a protocol in general, or about specific parts of the protocol as well. The most important and well-known one is written by Kitchenham. This is the document giving guidelines for performing an SLR in software engineering including the suggestion that a review protocol is used. These guidelines were used for the construction of the resulting review protocol of this paper. Several other papers about specific strategies or parts of the protocol, for instance the search strategy, are mentioned elsewhere in this paper, because they were used to answer the research questions and construct the protocol.

3 Research Method

The goal of this study was to create transparent and repeatable strategies for conducting an SLR. These strategies are defined in a review protocol, which is the final result of this study (cf. Appendix E). This paper will discuss the protocol and how the protocol was piloted. In this section we describe the context of the study, the research questions, the methods used, and an overview of the project.

3.1 Context

This study was done as a Bachelor project assignment. The assignment was to create a review protocol for a systematic literature review. Conducting the systematic literature review was not in the scope of the assignment. The SLR will be conducted by the supervisor of this project, with the aim to create an abstract overview of AKS approaches in the form of a pattern language. We created the main part of the protocol with input from our supervisor.
This paper looks into how to perform an SLR in the field of software engineering on the subject AKS by means of researching the construction, writing and piloting of a protocol. The result is a carefully constructed and piloted protocol, useful for an SLR about sharing architectural knowledge. The suggested methods in the protocol are discussed in this paper, together with a validation of the methods by doing a pilot study.

3.2 Research questions

There are four parts of the protocol that were considered most important in a review protocol. The following research questions are defined, based on these four parts:

- RQ1: What is a strategy that will identify all relevant primary studies for this SLR on AKS approaches in a transparent and replicable way?
- RQ2: What are the criteria that includes appropriate studies and excludes inappropriate studies according to the topic?
- RQ3: What are the criteria to assess the quality of the primary studies found for this SLR?
- RQ4: What is the data extraction strategy for the SLR?

For conducting an SLR it is very important to identify as many studies as possible, which are related to the research topic. Therefore it is crucial to define a good and detailed search strategy in the protocol. RQ1 aims at defining such a strategy, to identify all the relevant primary studies for an SLR on AKS approaches. This strategy should be transparent and replicable.
For RQ2, selection criteria should be defined that will include all the relevant studies for the SLR, and exclude irrelevant ones. A selection process should also be defined here, such that these criteria can be applied in a systematic way.
When all the relevant studies for the SLR are identified, the quality of those studies should be assessed. RQ3 aims at defining criteria for this assessment, so that this can be done in an unbiased way.
The last research question aims to define a strategy for extracting the data from the primary studies. RQ4 is answered with such a strategy. The result of applying this strategy to the relevant studies will be used to answer the research questions of the SLR.

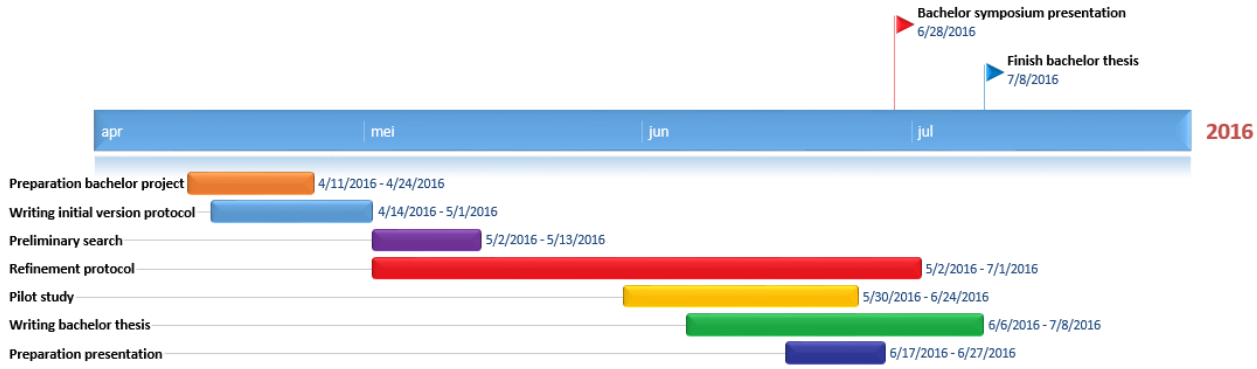
3.3 Methods

To start the study, and on every step along the way, the guidelines for performing Systematic Literature Reviews in Software Engineering [1] were consulted. These guidelines helped defining the different methods in the protocol in a transparent and replicable way. In addition to the guidelines, a lot of research into already existing SLRs and studies suggesting methods for performing an SLR was done. While writing the protocol, a lot of articles on architectural knowledge sharing were read in order to get a better understanding on the research topic.
The guidelines [1] describe the use of a preliminary study prior to conducting an SLR, and, in accordance, this search is suggested in the protocol. In parallel with constructing the protocol, a preliminary search was conducted. The aim of this unsystematic search was to get insight in the volume of potentially relevant studies, to be able to assess whether the scope of the SLR was not too broad or too narrow. The selected databases defined in the scope of the search (cf. Protocol Section 3.2.2.) were tested by doing automated searches on those databases, trying out several different search strings. The preliminary search was used to define the first selection criteria, test the scope of the search (including the selected databases and venues, and the specified time period) and establish the first search strings that are used for the automated search. A main search string is stated in the protocol (cf. Protocol Section 3.2.3), which is derived from the results of the preliminary search. Multiple search terms were identified during this search which formed an initial search string to be used for the automated search.
In order to gain some insight in the volume of the potentially relevant studies related to the research questions, articles from the selected databases in the protocol were retrieved by using the suggested initial search string. We assessed the relevancy of the retrieved articles by liberally applying the defined selection criteria on the titles, keywords and abstract. This resulted in a rough overview of how many primary studies can be found for the SLR. Because this search was preliminary, and not to identify all relevant primary studies, the search was time boxed. One man-week was considered for the preliminary search. Results of the preliminary search can be found in Appendix A.
To validate the protocol created, a pilot study was done to test several strategies suggested in the protocol. The whole pilot study was done by two people. One database that was selected in the scope of the search was chosen for piloting the protocol. The Search Strategy was piloted in one man-week and the Study Selection Criteria and Procedures also in one man-week. The Study Quality Assessment and Data Extraction Strategy were piloted together in four man-weeks. The total duration of the pilot study was four weeks, because alongside the pilot, the protocol was refined. More about the pilot study can be found in Section 5.

3.4 Project overview

Figure 1 shows an overview of the whole project. The total duration of the project was thirteen weeks. The focus in the first eleven weeks was mainly on creating and piloting the protocol, where the last two weeks the focus was on this paper. In order to process the results of the preliminary search and the pilot study, writing this paper started earlier, as shown in Figure 1.
The first few weeks of the project started with some preparation work and writing an initial version of the protocol. For preparation we familiarized ourselves with the subject by reading about SLRs and AKS. We also created an outline for the project to set some deadlines. In the end of week 3, we established an initial version of the protocol. Because of the iterative process, the protocol was refined several times until the end of the project. The pilot study started in the eighth week and ended before the final presentation of this project, in which the results were presented.

Figure 1. Timeline of project



4 Results

This section will give the results of the research questions defined in Section 3. A more detailed description of what is suggested in the protocol is stated in this section. The first subsection will state which strategy was used to identify the primary studies for this SLR. The second subsection is about how the selection criteria were defined to include relevant studies and exclude irrelevant studies and how these criteria were applied to the studies. The third subsection gives the criteria that were used to assess the quality of the relevant studies and the last subsection is about the strategy that was used to extract data from the primary studies found for the SLR.

4.1 RQ1: Search Strategy

To establish the transparency and replicability of an SLR, a detailed description of the search strategy is included in the protocol. This is crucial, since an SLR aims to identify as many articles related to the research questions as possible, without any bias.
We based the search strategy suggested in the protocol on the quasi-gold standard concept [6], which helped to construct an objective, repeatable and transparent search approach. This concept was also used to evaluate our search strategy. As stated in the protocol, the search strategy consists of three stages. Prior to conducting the SLR an unsystematic preliminary search is performed. (cf. Section 3.3.1) While conducting the SLR, two stages are left. First a subset of the primary studies will be manually established, called the quasi-gold standard, and is used for evaluation; secondly the protocol suggests to use an integrated search including an automated search complemented with a manual search.

4.1.1 Establishing the quasi-gold standard[2]

Search strategies for SLRs in software engineering are often evaluated in a subjective way [6], which means that the strategy defined in a protocol is evaluated by external experts before an automated search is performed. After the automated search these experts evaluate the results based on their own knowledge of the domain, making it difficult to test whether the search strategy is reliable.

To overcome these limitations, the quasi-gold standard (QGS) was introduced by Zhang et al. [6] as an objective way to assess the search strategy. This concept is called 'quasi' because, whereas the gold standard represents all the primary studies that can be found, the quasi-gold standard only represents a subset of all relevant primary studies [6]. The quasi-gold standard is a gold standard with the restriction of time and venues. A relevant time period has to be chosen and venues have to be selected according to the research topic. The QGS is established by manually searching all the selected venues that were defined in the scope of the SLR. The identified studies constitute a QGS. Because different databases represent different venues, the identified studies will be grouped by their respective database (the selected databases defined in the scope). This will form a set of the QGS for every database. A QGS for a database can then be used to evaluate search strings used for the automated search on this database. When evaluating the search string, the automated search will be performed only on the specific venues in the database that were chosen to represent the QGS. The evaluation is done by calculating the quasi-sensitivity and the precision of the results of a search string. The quasi-sensitivity and precision are calculated the following way:

$$\text{quasi-sensitivity} = \frac{\text{the number of articles from the QGS retrieved with the search string}}{\text{the number of articles in the QGS}} 100\%$$

$$\text{precision} = \frac{\text{the number of relevant articles retrieved with the search string}}{\text{the total number of articles retrieved with the search string}} 100\%$$

The higher the quasi-sensitivity and precision, the better the search string is. An optimum search string will have a quasi-sensitivity range of 80-99% and precision range of 20-25% [7]. Once a satisfactory search string is found, it can be used on the entire database, to find more relevant articles.

The quasi-sensitivity is derived from sensitivity, a term used mostly in the medicine domain. Sensitivity defines the amount of relevant articles found among all the relevant articles. As we do not know the set of all relevant articles, the sensitivity cannot be calculated. For this reason the quasi-sensitivity was introduced, which can only be calculated when doing the evaluation of the search strings. One could find all relevant articles in a specific database, by manually searching all articles, but this would be too time consuming, and also make the automated search irrelevant.

Compared to the quasi-sensitivity, the precision of the search can be calculated when doing an automated search on all the venues available in the database, and is expected to be close to the precision of the search limited to the selected venues.

Figure 2 shows a QGS, which is the intersection between circle 1, all articles in the database; circle 2, all articles of the selected venues in the database; and circle 3, all relevant articles in the database. Since circles 2 and 3 are entirely in circle 1, a QGS is also given by the intersection between circles 2 and 3. The area marked with 'X' contains all articles found with a search string that are in the QGS. So as can be seen in Figure 2, how more articles from the QGS are found with a search string, the greater the X will be, which results in a higher quasi-sensitivity.

In mathematical notation, a QGS, 'X', the quasi-sensitivity, and the precision are given by:

[2] This subsection was written by both authors of this document.

$QGS = 1 \cap 2 \cap 3 = 2 \cap 3$

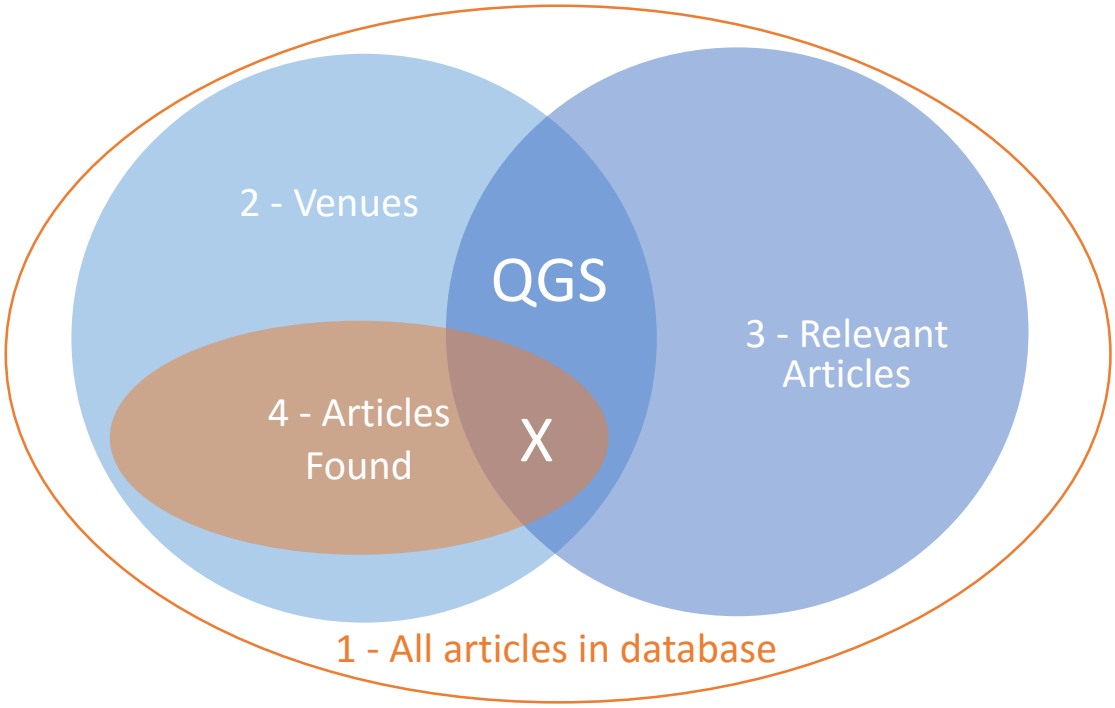
$X = QGS \cap 4$

$$\text{quasi-sensitivity} = \frac{\#X}{\#QGS} 100\%$$

$$\text{precision} = \frac{\#X}{\#4} 100\%$$

Where the numbers refer to the circles, and # denotes the amount of articles in a circle.

Figure 2. Quasi-Gold Standard, for evaluating search strings.



4.1.2 Automated and manual search

According to guidelines by Kitchenham [1], searching only the digital databases is not sufficient for an SLR. The guidelines suggest to also conduct manual searches to increase the search perimeter. So in order to increase the accuracy of the identification of the primary studies, an integrated search strategy is defined in the protocol, which is adopted from the QGS concept by Zhang et al. [6]. The strategy contains an automated search, complemented with a manual search.

The goal of the manual search is to find primary studies that are not or cannot be found with the automated search. For example, journals related to the research topic, at least the ones defined in the scope, should be searched. Also related conference proceedings, or other kinds of grey literature could be search and reference lists of relevant studies that have already been found should be reviewed (snowballing).

The goal of automated search is to find the bigger part of all the relevant studies according to the research topic, and also retrieve the relevant studies from the venues that are not included in the QGS. The search string found after evaluating with the QGS should be used on the entire database, without the restriction of the venues selected for the QGS. This should be done on all the selected databases defined in the protocol.

All the retrieved studies, from both manual and automated search, will be put in a spreadsheet, to keep track of which studies are included and which studies will be excluded after the study selection. In this spreadsheet at least the titles, authors, year of publication, abstracts and keywords, and URL to pdf from all the retrieved studies should be included.

4.2 RQ2: Study Selection Criteria and Procedures

In the protocol multiple selection criteria are defined and a three-stage process is presented for applying these criteria. The guidelines suggest the use of study selection criteria as means to evaluate the relevancy of the articles found. These criteria describe which articles to include, and which to exclude from the list with the retrieved studies. Some selection criteria were already defined in the first weeks of constructing the protocol. Because of the iterative process, these criteria were not final yet. During the process, some changes had to be made to the criteria.

There is one inclusion criterion defined in the protocol. This will include studies that present an approach for sharing architectural knowledge, a study that evaluates or compares one or multiple approaches for sharing architectural knowledge, a study that is about the state of the art of AKS or a study about industrial practices with regard to AK sharing. This inclusion criterion was first defined as "A study is directly related to at least one of the research questions in some way". Because of the research questions asking specifically about a suggested approach, relevant studies about AKS in general or comparative studies talking about multiple approaches, were excluded.

There are multiple exclusion criteria defined in the protocol that will exclude articles, for example, based on language, year of publication or on the kind of article, i.e. editorial or abstract. An exclusion criterion that was added in a later stage was to exclude studies about knowledge sharing, but not specifically about architectural knowledge. When piloting the protocol, we encountered multiple studies about knowledge sharing, and we decided to exclude those studies because the research topic is mainly about sharing design and design decisions. The final selection criteria can be found in Section 3.3 of the protocol.

The selection criteria should be applied by at least two reviewers in a three-stage process. This process suggested in the protocol entails the following steps:

- First make a broad call, interpret the criteria very liberally and assess the relevancy based on the title, keywords and abstract of an article. This results in a list of potentially relevant articles.
- Second, obtain the full text of the articles from the list established in the first step. Judge the relevancy now by precisely applying the selection criteria on the full text.
- Lastly, compare the inconsistencies between the assessments of the different reviewers. Try to come to an agreement and if this is not possible, another reviewer will come in and make the final call.

4.3 RQ3: Study Quality Assessment

Quality assessment criteria are defined in the protocol and piloted, in order to objectively assess the quality of each relevant article. The quality assessment can be done for two reasons. One can exclude articles with a poor quality score, and it can be used as an importance weight [3]. Since the selection criteria have been modified such that they include less empirical studies as well, and because an article of poor quality might still present useful information, it was decided not to use the quality assessment as an exclusion criterion.

The quality criteria in the protocol are means to assess the empirical value of an article and are based on the criteria defined by Dybå and Dingsøyr [8]. Next to assessing the empirical value of an articles, the criteria by Dybå and Dingsøyr cover three more quality issues: "Relevance, Rigour and Credibility"[8]. We did not include the relevance in our quality assessment, but our criteria are related to the validity, trustworthiness and meaningfulness [8]. The final quality criteria are defined in the Protocol Section 3.4.

The quality criteria are defined as yes/no questions and are answered with 0/0.5/1, where 0 denotes a 'no', 1 a 'yes', and 0.5 means 'to some extent'. The sum of all the answers, will give the final quality

score. The quality assessment defined in the protocol consists of nine questions, which will be put in a form, to increase the usability when performing the quality assessment.
The quality assessment should be done by one reviewer, and at least one other reviewer should check the assessment. For every criterion a motivation for the score should be given, which makes it more practicable for the second reviewer to check the quality assessment. The quality assessment will be conducted in parallel with the data extraction.

4.4 RQ4: Data Extraction Strategy

For the extraction of the data from the primary studies, a form was created, as is suggested by Kitchenham [1]. The form defined in the protocol contains question blocks, each containing one or more questions. These questions were all derived from the research questions defined in the protocol for conducting the SLR. The extracted data will then also be used to answer these research questions. The research questions that were defined are the following:

- RQ1: What approaches for sharing AK have been proposed in the literature?
- RQ2: What are the characteristics of the approaches identified in RQ1?
    - RQ2.1: What is the intended project context of the approach?
    - RQ2.2: Which knowledge management strategy does the approach support?
    - RQ2.3: What are the mechanisms of the approach to support sharing of AK?
    - RQ2.4: What are the architectural knowledge entities captured by the approach?
    - RQ2.5: How prescriptive or descriptive is the approach?
    - RQ2.6: What are reported benefits and limitations of the approaches?
    - RQ2.7: What empirical evidence do we have for each approach?
- RQ3: How are the approaches identified in RQ1 related to each other?
- RQ4: What is a pattern language for architectural knowledge sharing?

Most of the research questions were split up in a question complemented with an explanation, which is represented in a question block. The final data extraction form contains twelve question blocks (cf. Protocol Section 4). The first question block is only for documentation purposes, it contains information like the title, authors, year of publication and the abstract. The second question block is about who performed the data extraction, how was this study found and what was the final quality score of the study. The quality of a study was assessed with a separate form (cf. Section 4.3).
Question block three will answer the first research question in the protocol, which is: “What approaches for sharing AK have been proposed in the literature?” To answer this research question, the name of the approach proposed in the study and a short description will be documented in the form.
For question block four, we proposed to use a model to clearly describe the project context of the approach. This question block is related to research question 2.1. For describing the project context, the Octopus model from Phillipe Kruchten is used [9]. Using this model the different approaches can be classified with their corresponding project context, which is useful for the final goal of the SLR: creating a pattern language on AKS approaches. The factors that are used in this model to describe the project context are: size, criticality, business model, stable architecture, team distribution, governance, rate of change, and age of system.
Question blocks five till ten cover research questions 2.2 till 2.7 and question block eleven relates to RQ3, about comparing the found approaches and identifying relations between them.
The last question block is added so that there is some space for additional notes. We did not want to miss out on relevant information that we did not foresee, therefore we added question block twelve. This final block also helped us to refine our form.

5 Validation

In order to validate the results of the study, a pilot study was conducted for most methods included in the protocol. There is no pilot of the manual search, as this is outside the scope of this study. The methods that were piloted are the quasi-gold standard, the automated search, the selection criteria, the quality assessment and the data extraction. This section describes this pilot study. It describes per subject the research method, section 5.1, followed by the results of the pilot, section 5.2.

5.1 Research Method of Pilot Study

The methods of the protocol that were piloted, were only piloted on the database ScienceDirect. The pilot studies of each section of the protocol was done sequentially; first the QGS is piloted, then the automated search, and finally the quality assessment and the data extraction. ScienceDirect was chosen because the preliminary search showed many potentially relevant studies can be found here, most of the relevant studies here are articles, and the results showed clear venues with every study found, as opposed to, for instance, Google Scholar, which resulted in too many irrelevant articles including a lot of grey literature. This section describes the different parts of the pilot, how they were approached and executed. Section 5.1.1 describes the piloting of the quasi gold standard, Section 5.1.2 the pilot of the automated search, Section 5.1.3 the pilot of the selection criteria, Section 5.1.4 the pilot of the quality assessment and Section 5.1.5 the pilot of the data extraction strategy.

5.1.1 Pilot of the quasi-gold standard

Piloting the quasi-gold standard consists of two parts. First establish the QGS, and secondly, use the QGS for evaluating the search strings. The QGS for ScienceDirect was established by manually searching for relevant articles from the Journal of Systems and Software (JSS). The scope of the search, defined after the preliminary search in the protocol defines some relevant venues. Since more than half of the potentially relevant studies from ScienceDirect are from JSS, and none of the potentially relevant studies are from any of these venues, only JSS was used to establish a QGS for ScienceDirect (cf. Appendix A). Alongside the pilot of the concept of the QGS, the selection process and criteria were piloted, more about this in Section 5.1.3.

All articles published since 2004 were evaluated for their relevance, according to the selection process defined in Section 4.2 of this document and 3.3 of the review protocol. The relevant articles found, constitute a quasi-gold standard for ScienceDirect. This QGS can be found in Section 5.2.1 and Appendix B of this document.

Once the QGS was established, several search strings were tried and evaluated with the QGS, until a sufficient result was established. The search strings were performed on only JSS in ScienceDirect with a time period of 2004 until now. Starting with the search string found in the preliminary search and with the definition of AKS given in the review protocol, several different strings were evaluated on a trial-and-error basis. The strings were evaluated based on their quasi-sensitivity and precision. The final optimal search string for ScienceDirect and the rest of the result of this pilot can be found in Section 5.2.1 and Appendix C.

5.1.2 Pilot of the Automated Search

After an optimal search string for ScienceDirect has been defined, the automated search was piloted on the entire database of ScienceDirect. All the articles retrieved from ScienceDirect with this search string were evaluated according to the selection process and criteria as defined in Section 4.2 of this document and 3.3 of the review protocol. This resulted in a list of all relevant articles from ScienceDirect and a final precision of the search string for this database. The results can be found in section 5.2.2.

5.1.3 Pilot of the Selection Criteria

The selection process and criteria were piloted alongside the pilot of the QGS and the automated search. In order to say something about the relevance of a study, selection criteria and a selection process are needed to be able to select the relevant studies. Each article found went through the selection process as described in Section 4.2, but relevance was not only assessed by means of the selection criteria, but also if the article was really part of the scope of the SLR. This way, any flaws in the criteria got noticed and adjustments were made accordingly. After a change in the criteria, the list of potentially relevant studies, which are the once obtained after executing step one of the selection process, was assessed again with the new criteria. The results of this pilot can be found in Section 5.2.3.

5.1.4 Pilot of the Quality Assessment

The quality assessment criteria and the data extraction strategy were piloted at the same time for three weeks. This was possible since these two parts are independent of one another. For some of the relevant studies found with ScienceDirect, a quality assessment was done. A Google Form was used to answer the different questions described in Section 3.4 of the review protocol. By piloting the quality assessment on several different articles, more insight was gained into how well the criteria work for the different kinds of articles included. The results of this pilot can be found in Section 5.2.4.

5.1.5 Pilot Data Extraction

For all studies for which a quality assessment was done, the data was also extracted to pilot the data extraction form. The aim for this was to evaluate whether the form is complete, the questions are clear, and the form is usable [1]. A digital form was created with Google Forms, with all the questions of the data extraction form in Section 4 of the protocol. The studies were carefully read in order to be able to answer all the questions. Whenever a flaw in the form was discovered, the form was adjusted accordingly, and all articles that already had their data extracted were evaluated again with the changes. The results of this can be found in Section 5.2.5.

5.2 Results Pilot Study

This section describes the results of the different pilot studies described in the previous section.

5.2.1 Results of pilot of the Quasi-Gold Standard

After the relevance of all articles published by JSS, with the given time period, was evaluated with the selection process and after some changes in the selection criteria, a final QGS for ScienceDirect contained 18 articles. The authors, title, and year of publication of those 18 articles can be found in Appendix B.

With this QGS, several search strings were evaluated by means of their quasi-sensitivity and their precision. After seven trials, trial six turned out to be an optimal search string. This search string retrieved 69 articles from JSS including all the 18 articles from the QGS. This resulted in a quasi-sensitivity of 100% and a precision of 26.1%. The search string looks like:

> *"software architecture"*
> ***AND***
> *("architectural knowledge"* ***OR*** *"architecture knowledge"* ***OR*** *"architectural design decision"* ***OR*** *"architecture design decision")*
> ***AND***
> *("capturing"* ***OR*** *"sharing"* ***OR*** *"communicating"* ***OR*** *"documenting")*

The other trials, with their results, can be found in Table 9 in Appendix C.

5.2.2 Results of pilot of the Automated Search

The search string defined in the previous section was used for the automated search in ScienceDirect. ScienceDirect was searched for all the venues available. This resulted in 123 articles and 41 book chapters. Since there was no way of obtaining the full chapters, they were excluded from the pilot. A full copy is needed in order to be able to correctly assess the relevance of a study. These 123 articles were reviewed in the selection process to assess the relevance. This finally resulted in a precision of 21.1%, which is close to the precision that was calculated after the evaluation.

5.2.3 Results of pilot of the Selection Criteria

The selection process and criteria defined in the protocol were applied to all articles published by JSS available on ScienceDirect within the given time period and on the results of the automated search on ScienceDirect. Since the relevance of the articles was not only assessed with the selection criteria, but also with a general assessment of what are thought to be relevant in general, we gained a lot of insight in the performance of the selection criteria. This extra assessment showed no irrelevant studies were included and no relevant studies included with the selection criteria.

By applying the first step of the selection process to all articles published by JSS, from 2004 till now, a list of 96 potentially relevant articles was obtained (combined results of three reviewers). After step two and step three of the process were applied to these articles, the final result was a QGS of 23 articles. When changes were made in the selection criteria, the QGS was reviewed again, and resulted in 18 relevant articles. From the 123 retrieved articles from ScienceDirect, 26 articles were left after the selection process.

5.2.4 Results of pilot of the Quality Assessment

For fifteen of the 26 relevant articles from ScienceDirect, the online form for the quality assessment was filled in. Since the quality assessment checks the empirical value of a study, it was difficult to assess non-empirical studies. Therefore, those articles often obtained a very poor quality score. This was mainly an issue for articles presenting current practice of AKS. For all other articles it was pretty straight forward how to answer the different questions.

At first, it was impossible to motivate the answer to each question. Since there are no exact guidelines as to when to award how many points, this resulted in quality scores that appear to be quite random. To solve this problem, an extra explanation area was added to each question where one must motivate the given answer.

The lack of exact guidelines causes another difficulty. Their absence makes the score not only appear random, but even be sometimes quite random. It is for instance difficult to assess exactly when something is explained well enough. It is not possible to create such exact guidelines because each article is different, each author describes things differently, and they would have to be applicable to each relevant article found. This makes it even more important that that the answers to the questions are motivated.

5.2.5 Results of pilot of the Data Extraction

For each of the fifteen articles for which the quality was assessed, the data was also extracted. Since the questions in the data extraction form focus on specific approaches being proposed in studies, it was known beforehand that the form would not work for all studies. This turned out to be true. The form was hardly useful for studies describing the state of the art of AKS. Most questions had to remain unanswered. Since those articles do contain relevant information, the additional notes question is very useful for these types of studies.

The fact that the questions in the form are so focussed on a specific approach, they were also slightly difficult when a study was of the comparing type. This was solved by numbering the different

approaches compared in the study, and each question was answered as accurate as possible. Since these articles do not propose anything, the answers to the questions may be slightly different than when the articles do propose an approach, as the form is designed for the latter.
A last difficulty that was experienced was with the questions referring to the problem context (RQ 2.1). Often the answers to the questions were not explicitly stated in the study. Giving a specific answer was therefore often hard and since the answer often came from interpretations of parts of the description of the proposed approach, the answers might not always be completely accurate and/or objective. Therefore, it was important to motivate the answers given, especially when they are based on interpretations. Sometimes the answers were given based on the problem context of a case study, where the approach was often applied and tested in an industrial environment. The disadvantage of this is that the approach was probably not designed for that specific industrial environment, and hence answers based on interpretations of the case study probably limit the approach too much. To avoid such inaccuracies, it is clearly stated in the motivation of the answer that the answer is based on a case study.

6 Evaluation

The results presented in section 4 were carefully validated with a pilot study as described in section 5. Since the pilot study was performed on only one database, and sometimes even on only one venue, there is no way of knowing whether this works for all databases and venues. Nevertheless, it is assumed that the results of the pilot are a good representation of the expected results for the not piloted databases and venues.

6.1 Evaluation of results

The protocol suggests to aim for an optimum search string, which has a quasi-sensitivity between 80% and 99 % and a precision between 20% and 25%. In the pilot study, the objective evaluation using the concept of QGS, resulted in a search string for ScienceDirect with a quasi-sensitivity of 100% and a final precision of 26.1%. This search string (cf. Section 5.2.1) is well within limits of this optimal search string, and even exceeds expectation as it has an even higher quasi-sensitivity than the optimum search string described. The pilot study on ScienceDirect of the automated search resulted in 26 relevant articles among the 123 retrieved articles, which gives a precision of 21.1%. As expected, this precision value is close to the precision of the evaluation and within limits of the optimum search string.
The number of relevant articles found with the automated search on ScienceDirect is less than and not close to the number of potentially relevant studies found in the preliminary search. As one of the goals of the preliminary search is to gain insight in the volume of relevant studies available, one would expect these numbers to lie closer together. However, as the selection criteria were only liberally applied to the articles (only to the title, keywords, and abstract), which resulted in a list of potentially relevant studies, and since the selection criteria have changed later on, it is not an unexpected result.
The selection process, in which the selection criteria are applied, was applied to over a thousand articles. During the process we, including the supervisor, got a better understanding of the scope of the SLR. Which articles we wanted to include for the SLR, and which articles we wanted to exclude. Therefore, and due to the iterative process, some changes had to be made to the selection criteria, which was expected. With the final selection criteria, as defined in the review protocol, no relevant studies were excluded or irrelevant included from the SLR during the pilot. This result suggest that the criteria defined work well for this SLR.
As expected, the quality assessment does not work well for all relevant studies identified, as it assesses the empirical value of a study and since non-empirical studies are also included in the SLR according to the protocol. Since the main goal of the quality assessment is to assign an importance weight to the

selected studies, a little precaution is needed. Before quickly discarding a study based on a low quality score, the researcher should look why the one who extracted the data gave the study its poor score.
As with the quality assessment, the data extraction form does not work great for all types of articles selected for the SLR. The biggest struggle lies with articles describing the current practices of AKS, since hardly any of the questions in the form apply to this type of studies. The additional notes are therefore necessary, and when the data are interpreted, the researcher should take the difficulties of the form and the additional notes into account.
Overall, the methods and strategies suggested in the protocol can probably be used for other SLR, except, of course for the subject specific ones. Especially, the search strategy is probably not only efficient for this SLR, but can likely be applied to other SLR's about different subjects, with only minor changes. The selection process, apart from the selection criteria, the quality assessment, and the data extraction strategy, except for the form, are not subject specific and likely at least a good start for other researchers interested in conducting an SLR in software engineering.

6.2 Evaluation of project

The project took part over 13 weeks, a lot of time has gone into writing the protocol and refining it. While first versions of most of the sections of the protocol were on paper within two weeks, it still took quite some time before the pilot started. It might have been more efficient to start this pilot earlier, while still refining the protocol, as that would have resulted in an earlier identification of problems. While the late start of the pilot did not give any direct drawbacks, it was not beneficiary to the efficiency of the researchers, nor to the perception of enough time left to finish the project. In the end, a better planning and looking further ahead than just the next two weeks or so would have allowed for the project to be more structured.
First versions of the protocol and the thesis were finished quite soon, allowing much time for review and refinement, which was beneficial to the quality of both papers. Some parts of the pilot study took a lot of time. The data extraction and quality assessment were piloted together but were very time consuming to pilot. However, this was anticipated, as every article should be read carefully, and some parts more than once, before one was able to confidently answer the quality assessment or data extraction question. As the data extraction form was changed several times during the pilot, extracting data again from the articles that were already done did slow the process down even more. Since this pilot study was time boxed, it was no danger to our planning, but rather to the number of articles used for this pilot. In the end, the quality assessment and data extraction strategy were piloted on over half of the relevant articles from ScienceDirect.
The pilot of the QGS was expected to take a very long time, as the selection process had to be applied to over a thousand articles. It still took quite some time, but was finished quicker than expected. This was also the case with the automated search, as this was done on ScienceDirect only, the search string was used only once, after which the resulting articles, minus those from JSS, had to go through the selection process. This was all done in one day, which allowed us to move on quickly to the next elements.
At the beginning of the project, it was often a bit unclear what was expected exactly of the protocol, but as progress was made the objective became more and more clear. This made it easier to refine the protocol. The pilot study was very straightforward, and as we had learned a lot about protocols for SLRs, the evaluation and assessment of the results was not a complex task. Another reason for this is that clear objectives were often defined beforehand.
A personal reflection on the project of each of the authors of this document can be found in Appendix D.

7 Conclusion

This study resulted in a review protocol for an SLR on Architectural Knowledge Sharing approaches. In this protocol we suggest an objective method for evaluating the search strategy and constructing an objective search query. We adopted the quasi-gold standard concept introduced by Zhang et al., which helped creating the systematic and unbiased search strategy. This search strategy also increases the search perimeter by using an integrated method of an automated search complemented with a manual search. For the automated search, an optimum search string can be found by evaluating the results of the search using the quasi-gold standard. For the manual search snowballing is mainly used, which means the reference lists of the relevant primary studies will be reviewed. Also conference proceedings, or other kinds of grey literature, related to the research topic can be searched. Non-empirical studies will not be excluded because of the desire for as much relevant information as possible.

The four main parts of the review protocol, which are the Search Strategy, the Study Selection Criteria and Procedures, the Study Quality Assessment, and the Data Extraction Strategy were tested in a pilot study of four weeks. Piloting on ScienceDirect, we showed that the objective evaluation of the search and the construction of an optimum search string resulted in impressive results for the quasi-sensitivity and precision. After a few iterations the final selection criteria were established that include the relevant articles and exclude the irrelevant articles according to the research topic. These selection criteria were tested in the pilot study when establishing the QGS for ScienceDirect and when retrieving the articles from ScienceDirect with the automated search. The automated search resulted in 26 relevant articles from ScienceDirect, after the selection criteria were applied in the three-stage process. Using these 26 articles, the quality assessment and data extraction were piloted in parallel, which resulted in clear and usable forms for these methods.

Since the resulting protocol is not yet a finished protocol, as several sections were out of the scope of this study, there is no way of knowing whether the quality assessment and the data extraction have their desired effect for the next stages. We tested whether the form worked and the quality assessment questions were answerable, but we don't know whether the extracted data is useful to draw conclusions. What is known, is that the strategies proposed, the quality assessment questions, and the data extraction form work at least reasonably well for the studies they were piloted on.

The pilot study is assumed to be a good representation of how well the protocol works for the entire SLR, and hence we conclude that the resulting protocol works very well for the first stages of conducting an SLR.

The protocol developed for this study will be further refined and finalized by the supervisor of this project, after which the SLR is conducted. It is important to notice that until the SLR is finished, the protocol is not finished, and that there will likely be small revisions in the future. The methods suggested in this paper and in the review protocol have been carefully piloted and will therefore be useful for others interested in performing an SLR.

Bibliography

[1] B. Kitchenham and S. Charters, "Guidelines for performing Systematic Literature Reviews in Software Engineering," *Engineering*, vol. 2, p. 1051, 2007.

[2] H. Zhang and M. Ali Babar, "Systematic reviews in software engineering: An empirical investigation," *Inf. Softw. Technol.*, vol. 55, no. 7, pp. 1341–1354, 2013.

[3] P. Brereton, B. A. Kitchenham, D. Budgen, M. Turner, and M. Khalil, "Lessons from applying the systematic literature review process within the software engineering domain," *J. Syst. Softw.*, vol. 80, no. 4, pp. 571–583, 2007.

[4] S. MacDonell, M. Shepperd, B. Kitchenham, and E. Mendes, "How Reliable Are Systematic Reviews in Empirical Software Engineering?," *IEEE Trans. Softw. Eng.*, vol. 36, no. 5, pp. 676 – 687, 2010.

[5] B. Kitchenham, R. Pretorius, D. Budgen, O. P. Brereton, M. Turner, M. Niazi, and S. Linkman, "Systematic literature reviews in software engineering-A tertiary study," *Inf. Softw. Technol.*, vol. 52, no. 8, pp. 792–805, 2010.

[6] H. Zhang, M. A. Babar, and P. Tell, "Identifying relevant studies in software engineering," *Inf. Softw. Technol.*, vol. 53, no. 6, pp. 625–637, 2011.

[7] O. Dieste and A. G. Padua, "Developing Search Strategies for Detecting Relevant Experiments for Systematic Reviews," *Proc. - 1st Int. Symp. Empir. Softw. Eng. Meas. ESEM 2007*, pp. 215–224, 2007.

[8] T. Dybå and T. Dingsøyr, "Empirical studies of agile software development: A systematic review," *Inf. Softw. Technol.*, vol. 50, no. 9–10, pp. 833–859, 2008.

[9] P. Kruchten, "The Frog and the Octopus — Experience Teaching Software Project Management," *Proc. Can. Eng. Educ. Assoc.*, 2011.

Appendices

Appendix A. Preliminary search

The preliminary search resulted in a table of synonyms and alternative terms useful in our domain. These terms, shown in Table 1, and the research questions were used to construct the initial search string for the automated search. This search string can be found in Section 3.2.3 of the protocol.

Table 1. Synonyms of search terms

Synonyms	Source
Software development management	Clerc et al. (2007)
Knowledge engineering	Babar et al. (2007)
Knowledge-based approach	Li et al. (2012)
Architectural knowledge sharing approach	Research questions
Architecture knowledge management	Ali Babar et al. (2009)
Architectural decisions making process	Dragomir et al. (2014)
Architectural design decisions	Chen et al. (2010)

The databases considered in the preliminary search are the following:

- IEEE XPlore
- ScienceDirect
- SpringerLink
- ACM digital library
- Google Scholar

When these databases were searched with Search string 1, already good results were obtained. Only the ACM digital library resulted in only two articles. These results are shown in Table 2. The relevance was checked by two reviewers and the precision was calculated. The precision is the percentage of the relevant articles among all the retrieved articles. When the search was conducted using Search String 2, ACM digital library resulted in more relevant articles, but also more irrelevant articles. We decided that we prefer to have a search string that results in more articles, even when there are more irrelevant articles among the retrieved articles. This way it is more certain that we are not missing many relevant articles. The results of the search with Search String 2 are shown in Table 3.
The articles retrieved by Google Scholar with Search string 2, which were 1290 articles, were not all checked. There was decided that the results retrieved with Search string 1 were sufficient enough and this search string was chosen to use for the automated search. In Table 4 is shown which specific search string is used for the different databases, with in Table 5 the results of the conducted searches with these search strings.

Search string 1.
("architecture knowledge sharing" OR "architectural knowledge sharing" OR "decision sharing") AND "software architecture"

Search string 2.
("software architecture" AND "architectural knowledge" AND ("architectural design decisions" OR "sharing"))

Table 2. Preliminary search result with search string 1.

Database	Number of articles/chapters	Relevant articles	Precision
IEEE Xplore	48	44	92
ScienceDirect	16	11	69
SpringerLink	36	29	81
ACM digital library	2	2	100
Google Scholar	206	141	68

Table 3. Preliminary search result with search string 2.

Database	Number of articles/chapters	Relevant articles	Precision
IEEE Xplore	48	44	92
ScienceDirect	93	51	55
SpringerLink	192	111	58
ACM digital library	92	56	61
Google Scholar	1290	-	

Table 4. Databases with the corresponding search strings.

Database	Search string	Scope
IEEE Xplore	("architecture knowledge sharing" OR "architectural knowledge sharing" OR "decision sharing") AND "software architecture"	Full text
ScienceDirect	("software architecture" AND "architectural knowledge" AND ("architectural design decisions" OR "sharing"))	Full text
SpringerLink	("software architecture" AND "architectural knowledge" AND ("architectural design decisions" OR "sharing"))	Full text
ACM digital library	Any fields matches all "software architecture" "architectural knowledge" Any fields matches any "architectural design decisions" "sharing"	Full text
Google Scholar	("architecture knowledge sharing" OR "architectural knowledge sharing" OR "decision sharing") AND "software architecture"	Full text

Table 5. Search results of databases and search strings in table 5.

Database	Number of articles	Relevant articles	Precision
IEEE Xplore	48	44	92
ScienceDirect	93	51	55
SpringerLink	192	111	58
ACM digital library	92	56	61
Google Scholar	206	141	68

Of the 111 potentially relevant studies found with SpringerLink (Table 5), only 6 are articles. The rest of the studies are all book chapters. ACM (Table 5) also returned only 6 relevant studies published in a Journal, the rest of them are mostly proceedings, which can be called grey literature. As for Google Scholar, it was often difficult to determine the exact origin and type of a study, and the list of sources

in the excel is not complete. Both ScienceDirect and IEEE Explore gave many relevant articles. Because ScienceDirect gave more relevant articles, the protocol was piloted on this database. Below, in Table 6, one finds a dissection of the potentially relevant results from ScienceDirect based on the venue:

Table 6. Results ScienceDirect, relevance per venue

Venue	% of relevant studies from all relevant studies
JSS	56.86
Information and Software Technology	15.69
Book Chapter	9.80
Science of Computer Programming	7.84
Future Generation Computer Systems	3.92
Information Sciences	1.96
Computers in Industry	1.96
Accounting, Organizations and Society	1.96

Appendix B. Quasi-gold standard of ScienceDirect

Table 7 shows the final QGS that was established for ScienceDirect. Table 8 shows articles that were excluded from the QGS in a later stage.

Table 7. The QGS for ScienceDirect.

Author	Title	Year
Capilla, Rafael; Jansen, Anton; Tang, Antony; Avgeriou, Paris & Babar, Muhammad Ali	10 years of software architecture knowledge management: Practice and future	2016
Manteuffel, Christian; Tofan, Dan; Avgeriou, Paris; Koziolek, Heiko & Goldschmidt, Thomas	Decision architect - A decision documentation tool for industry	2016
Gaubatz, Patrick; Lytra, Ioanna & Zdun, Uwe	Automatic enforcement of constraints in real-time collaborative architectural decision making	2015
van Heesch, U.; Avgeriou, P. & Tang, A.	Does decision documentation help junior designers rationalize their decisions? A comparative multiple-case study	2013
van Heesch, U.; Avgeriou, P. & Hilliard, R.	A documentation framework for architecture decisions	2012
Weinreich, Rainer & Buchgeher, Georg	Towards supporting the software architecture life cycle	2012
Zimmermann, Olaf; Miksovic, Christoph & Küster, Jochen M.	Reference architecture, meta-model, and modeling principles for architectural knowledge management in information technology services	2012
Hoorn, Johan F.; Farenhorst, Rik; Lago, Patricia & van Vliet, Hans	The lonesome architect	2011
Tang, Antony; Avgeriou, Paris; Jansen, Anton; Capilla, Rafael & Babar, Muhammad Ali	A comparative study of architecture knowledge management tools	2010
Unphon, Hataichanok & Dittrich, Yvonne	Software architecture awareness in long-term software product evolution	2010
Christensen, Henrik Bærbak & Hansen, Klaus Marius	An empirical investigation of architectural prototyping	2010
Jansen, Anton; Avgeriou, Paris & van der Ven, Jan Salvador	Enriching software architecture documentation	2009
López, Claudia; Inostroza, Pablo; Cysneiros, Luiz Marcio & Astudillo, Hernán	Visualization and comparison of architecture rationale with semantic web technologies	2009
Zimmermann, Olaf; Koehler, Jana; Leymann, Frank; Polley, Ronny & Schuster, Nelly	Managing architectural decision models with dependency relations, integrity constraints, and production rules	2009
Jansen, Anton; Bosch, Jan & Avgeriou, Paris	Documenting after the fact: Recovering architectural design decisions	2008
Tang, Antony; Jin, Yan & Han, Jun	A rationale-based architecture model for design traceability and reasoning	2007

Tang, Antony; Babar, Muhammad Ali; Gorton, Ian & Han, Jun	A survey of architecture design rationale	2006
Roeller, Ronny; Lago, Patricia & van Vliet, Hans	Recovering architectural assumptions	2006

Table 8. Excluded papers from QGS.

Authors	Title	Year
Van Vliet, Hans & Tang, Antony	Decision making in software architecture	2016
Liang, Peng; Jansen, Anton; Avgeriou, Paris; Tang, Antony & Xu, Lai	Advanced quality prediction model for software architectural knowledge sharing	2011
Kruchten, Philippe	What do software architects really do?	2008
Sowe, Sulayman K.; Stamelos, Ioannis & Angelis, Lefteris	Understanding knowledge sharing activities in free/open source software projects: An empirical study	2008
Corbin, Richard D.; Dunbar, Christopher B. & Zhu, Qiuming	A three-tier knowledge management scheme for software engineering support and innovation	2007

Appendix C. Evaluation of search strings for ScienceDirect

Table 9. Trials for evaluating the search string.

Trial nr.	Search string	# retrieved	# QGS	Sensitivity	Precision
1	("software architecture" AND "architectural knowledge" AND ("architectural design decisions" OR "sharing"))	43	15	83.3%	34.9%
2	("software architecture" AND "architectural knowledge" AND ("architectural design decisions" OR "sharing" OR "capture"))	45	16	88.9%	35.6%
3	("software architecture" AND "architectural knowledge" AND ("architectural design decisions" OR "sharing" OR "documenting"))	45	16	88.9%	35.6%
4	("architecture knowledge sharing" OR "architectural knowledge sharing" OR "architectural design decisions" OR "decision sharing") AND "software architecture"	43	13	72.2%	30.2%
5	("architectural knowledge" OR "architecture knowledge" OR "design decision" OR "design decision making") AND "software architecture" AND ("capturing" OR "sharing" OR "communicating" OR "documenting")	161	18	100%	11.2%
6	("architectural knowledge" OR "architecture knowledge" OR "architectural design decision" OR "architecture design decision") AND "software architecture" AND ("capturing" OR "sharing" OR "communicating" OR "documenting")	69	18	100%	26.1%
7	("architectural knowledge" OR "architecture knowledge") AND "software architecture" AND ("capturing" OR "sharing" OR "communicating" OR "documenting" OR "architecture design decision" OR "architectural design decision")	49	16	88.9%	25.0%

Appendix D. Reflection

We have both written a reflection on our bachelor project, which can be found below. The STARR method is used for writing these reflections. The meaning of STARR is Situation, Task, Action, Result, and Reflection.

Reflection by Rayan Brouwer

Situation

The last phase of our bachelor Computing Science at the RUG is a bachelor project of 15 ECTS. This project started mid-April until the 8th of July. For our bachelor project we constructed a review protocol for a systematic literature review (SLR) and wrote a separate thesis on how we constructed this. The goal of this SLR is to create an overview of Architectural Knowledge Sharing approaches in the form of a pattern language. This SLR will be conducted by our supervisor Christian Manteuffel, so we created the review protocol in collaboration with him.

Task

The goal of our bachelor project was to create a review protocol, to write a thesis about how we constructed and tested the protocol, and to give a final presentation at the Bachelor symposium. For the review protocol we defined a systematic search strategy, selection criteria to assess which studies are relevant and which studies are irrelevant according to the research topic, a quality assessment, and a strategy for extracting the data from the primary studies for this SLR. These four methods are all important parts of a review protocol. After we defined some methods in the protocol, the methods also had to be tested. Consequently, did a pilot study, using the database ScienceDirect, to show that our methods were working and to refine the methods.

We did not assign specific tasks or divided the tasks in the beginning of the project. In the end, we constructed every part of the protocol together. We thought the best way was to just start writing and learn more about writing a protocol during the process. We planned four weeks for creating an initial version of the protocol and the other weeks for refinement and testing. The last two weeks we mainly focused on the thesis.

Action

In the beginning of the project it was difficult to start writing the protocol. We were unexperienced in writing such a review protocol and also did not have any experience in the subject Architectural Knowledge Sharing. Our supervisor recommended us to not read too much in the beginning but combine the writing and the reading. We both started writing on different parts and then checked and refined each other's writings. For almost every section in the protocol there was already something written in the first two/three weeks. The following weeks we kept refining the protocol until the last week, as we learned more about the subject. So, a big part of the project was refinement. In the fourth week I started with the preliminary search, with the goal of refining the selection criteria, testing the scope and try to find usable search strings for the databases. We decided to time box this search, because it was only preliminary and not a systematic search. I finally spent one and half week on the search, including assessing the relevancy of the retrieved articles based on their title, keywords and abstract. It would have probably been better to not consider the abstracts, because it would have saved a lot of time and the goal was only to assess the volume of the potentially relevant articles, not to find the final studies for the SLR.

In the middle of the project, we were a little bit slowed down. We kept refining the protocol but we did not make much progress. Finally in week eight we started piloting the manual search for establishing the QGS (using JSS). Manually searching the journal, assessing the final relevance of the articles found and discussing the inconsistencies between the three of us took us finally one week. Next, we did the evaluation of the search strings for ScienceDirect in one day, and when the optimum search string was found, we performed the automated search. The automated search resulted in 123 articles from ScienceDirect. It took us half-a-week to apply the selection process on the 123 articles,

which gave us a result of 26 relevant articles from ScienceDirect. The last four weeks we worked on piloting the quality assessment, the data extraction form, refining the protocol and finish writing our thesis.
A critical part of the protocol was defining the search strategy, which is based on the quasi-gold standard. The search strategy was already defined in the first weeks, but only later in the project after rereading and some knowledge from external experts, we had a full understanding on how the search strategy should be performed.

Result
The final results of our bachelor project are the review protocol, the thesis about how the protocol is constructed and tested, and the presentation for the Bachelor symposium. The protocol is almost done, our supervisors will keep working on it before they start with conducting the SLR, but the main part of the protocol is there. We started with the protocol in mid-April and kept working on it for thirteen weeks, because the duration of our bachelor project was only thirteen weeks. In the last weeks we wrote our thesis on how we created the protocol and how we piloted it. I think the final thesis has become a good and clear paper about how to construct a protocol for an SLR in the software engineering domain. The pilot study also showed very good results for the search strategy.
On the 28th of June we gave a presentation at the Bachelor symposium about the protocol we created and the pilot study we had done to validate the protocol. I think the presentation went well, although we could have been a little bit clearer on the actual goal of our project. We received positive feedback on the presentation.

Reflection
Overall, I think the project went smoothly. It would have probably been better to start with the pilot study earlier in the project. We started the pilot in week eight, and we were done around the time of the presentation (beginning of week twelve). The manual search (establishing QGS) and the selection process was taking a little bit longer than anticipated, because of the inconsistencies between the three reviewers. Also the piloting of the data extraction form took a lot of time, as we also had expected beforehand.
Because we did not have any experience with writing a protocol, in the beginning of the project we encountered some difficulties with the planning of our project. When we learned more about protocols and SLRs in the first few weeks, it was easier to create a clear outline for the rest of the project.
During our project we learned a lot about Systematic Literature Reviews, how to search for relevant articles according to a research topic, how to assess articles on their relevance and their quality, and how to extract data from studies, all using systematic methods in an unbiased way. We also learned a lot about the importance of Architectural Knowledge Sharing and the different kind of approaches to share AK. Personally, I also improved my English writing during the project.
I think the search strategy that we suggested in the protocol is helpful for SLRs in the software engineering domain, because the strategy is used to construct an objective search query and to objectively evaluate the search strategy. It helps refining the search string used for automated search, until the sensitivity and precision are high enough, so that one can assume that the relevant articles according to the research topic can be found.

Reflection by Sophie Hugenholtz

Situation
The last part of finishing a Bachelor Computing Science consists of doing a bachelor's project. This project contains three parts; the project itself, a thesis about the project, and a presentation of the project. The project can be done by one, two or three students, is mandatory, and counts for 15 ECTS. The bachelor's project is to be done in the fourth period of the last year of a bachelor. This means that for this year, it started on April 11th. The project lasts for eleven to thirteen week.

Task

The goal of the project is for a student to execute an assignment usually including a literature study, the development of a software component, and writing a thesis where the results are presented and evaluated. It is also possible that the project does not include developing any software or any programming in general. Some personal skills that should be obtained and are tested are that a student is able to

- analyze results and report them in academic writing,
- explain the project and the results by means of an oral presentation,
- work and function well in a team, if the project is performed by more than one student,

My project was done in a team of two, where we had to write a protocol for a systematic literature review, which is going to be conducted by our supervisor. There was no programming involved, but instead of a resulting software product, we had to produce this protocol and carefully pilot this. A had to be written about the process of the construction and the pilot study. Each student should write at least one of the chapters of the thesis individually.

The review protocol written, should be for a replicable, transparent, and unbiased SLR. It should describe a search strategy that aims at identifying as much relevant studies as possible, should contain selection criteria than, in the best way possible, include relevant studies for the SLR and exclude irrelevant studies, define quality assessment criteria that assess the quality of studies as objective as possible, and define a data extraction strategy that describes how to extract as much relevant data from the studies as possible.

Next to this protocol, we should give an oral presentation of the project, and write a thesis explaining how the protocol was constructed, how all problems were approached, how the pilot study was approached, the results of the pilot study, and discuss and evaluate the results of the thesis.

Action

We started with writing the protocol. As recommended by our supervisor, we did not read too much before starting with writing the protocol, only the guidelines by Kitchenham were consulted. I started with the introduction and the background, and Rayan with the search strategy. This division of tasks was quite set for the first days, until there was a solid base for the introduction and the background. I then moved on to certain sections of the search strategy Rayan had not written yet, and helped her with the ones she was still working on. Then Rayan started on the quality assessment section and I on the data extraction. This division of tasks of the last two chapters of the protocol, was also pretty much fixed for a while. Once a good foundation for the protocol was written, we started reviewing each other's work and moved on to the next steps.

Rayan started with the preliminary search, while I started rewriting the entire introduction and background, as a lot of feedback was given there. Afterwards, I checked the preliminary search, where the focus was mainly on which articles were relevant and which not.

After finalizing the preliminary search, process its results, and doing another round of revision of the protocol, it was time to start piloting the protocol. We began with the manual search. This meant we had to go through all articles published by the Journal of Systems and Software between 2004 and now and assess them for their relevance. We both did this individually by scrolling through ScienceDirect, through issue after issue. After all potentially relevant articles were selected, we merged the two lists together, removing the duplicates. It was then time to take a closer look at the true relevance of the articles, a process we again performed individually, without looking at the judgement of the other. Any inconsistencies were discussed and resolved in the end.

When all relevant articles from ScienceDirect were found, an automated search could be performed. Rayan started out with the string from the preliminary search, whereas I started with the definition of

AKS as defined in the protocol. As this turned out the way to go, we arrived at a great result after just a few iterations of small changes to the string I started with. The articles this string retrieved all had to be checked for their relevance, which we did in a similar way the relevance of the articles from the manual search was checked. Rayan filled in the relevance from JSS, since we already assessed those articles, so I started with all the other articles.
During this time, we also started writing the thesis. First making a general outline, and moving some parts of the protocol to the thesis, after which other sections of the thesis began getting shape. Also during this time, the presentation came closer. We made a rough overview of the slides and the presentation we wanted to give, and worked on this together almost every day up until the day of the presentation itself. We alternated during the presentation every couple of slides between who was presenting. Rayan opened and I closed.
The next phase of the pilot study was piloting the quality assessment and the data extraction form. Over half of the relevant articles from ScienceDirect were processed, where the data extraction was done parallel to the quality assessment. Each of us extracted the data of different articles.
After the presentation and the pilot of the data extraction and quality assessment, some final touches were made on the protocol, while the main focus was on the thesis. It was first co-authored almost entirely, but later noticed that each of us should write at least one chapter individually. Therefore, we decided that Rayan wrote the results, and I did the Validation. The validation section contains the entire pilot study, including its results, and sometimes a short evaluation of the results. I split it up into two parts, one stating the pilot method for each of the pilots, and one showing the results of the different pilots.

Result
This bachelor's project has three very obvious results. These are the review protocol, the thesis, and the presentation. We can say with some certainty that the resulting review protocol is a working, useful protocol for the SLR it is meant for, due to the pilot study.
Next to the obvious results, there are also a few tangible results, such as an increase in the level of certain skills, a greater knowledge about certain subjects, and a better understanding of research in the field of computer science or software engineering. Also, since a previous presentation had been a long time ago, presenting skills were refreshed or even improved.

Reflection
The resulting protocol, which was the main subject of our project, was in the beginning and at times a little difficult to write. This was because, certainly in the beginning, we began writing without any knowledge. As more literature about the subject was consulted, the whole project quickly began to gain shape in my head. Because we started writing without prior knowledge, we were so busy trying to get a grip on what we were writing, that we were less concerned with what the other one was writing. This resulted in the fact that we did not start reviewing each other's work, until a first version of the protocol was already finished. As many more versions were still to come, this did not give any problems, but for the teamwork's sake, it would probably have been better had we started this earlier. As this might be called a small planning problem, it leads me to the next point. At a few points during this project, we got stuck a little, wondering what it was exactly what we should do next. This resulted in a quite late start of the pilot study. Where we did not feel much time pressure at the beginning of the project, this lead to long days of a lot of work. It did not affect the quality of the work though. Had we watched the planning better, and evaluated the planning more often, the last few weeks of the project could have probably been more manageable.
I looked up against the manual search, as over 2000 articles had to go through the selection process. In the end, I was able to focus on this very well, and the first selection was made fairly quickly. I did

miss a few relevant articles though, but the whole reason the two of us were doing the search individually, was that we anticipated a small error on both sides. Another part of the project that I was not eager to start on, was the last pilot. This was the pilot of the data extraction form and the quality assessment. It was difficult at times to fully focus on each of the articles, but in the end, it was very doable. We anticipated on doing all of the relevant articles found with the automated search, but instead only came to a little over half. This was due to deficiencies of the initial data extraction form, research in how to answer certain research questions, and having to redo work after changes in the form were made. I don't think we could have gone through the articles any faster, and the number of articles done is sufficient to my opinion for a good pilot of the data extraction form and the quality assessment.

The teamwork between Rayan and I went fairly well, I think. We have done several projects together in the past, which means we knew exactly what to expect from the other. The contact went without problems, we understood from one another what we were trying to do or say, and were able to give each other valuable feedback.

Overall, I am happy with the process and result of this bachelor's project, even though I might do some things different in the future, and there is some room for improvements.

Appendix E. Protocol

Towards a pattern language for Architectural Knowledge Sharing

A Systematic Literature Review Protocol

Rayan Brouwer & Sophie Hugenholtz

8-7-2016

Version 10

Contents

Revision History

Version	Date	Author	Description
0	14/4/2016	Sophie Rayan	Start writing intro and background Start writing research methodology
1	15/4/2016	Sophie Sophie, Rayan Sophie Rayan	Adding section numbering, TOC and revision history Processing comments and changes Continue writing intro and background Continue writing research methodology
2	20/4/2016	Sophie Rayan	Adding reference list using Mendeley, continue writing on background, reading and adjusting research methodology Continue writing research methodology, adding part about 'quasi-gold standard'
3	25/4/2016	Rayan Sophie	Continue writing search strategy, scope, search process, search strings Start on selection criteria/process + small revisions of background
4	28/4/2016	Rayan Sophie	Writing quality assessment, organizing search methodology, adding diagram and text to search process Revise selection criteria, start on major revision introduction and background, writing data extraction
5	4/5/2016	Rayan Sophie	Refining search strategy, search process Refining introduction and background
6	19/5/2016	Rayan Sophie	Refining research methodology, writing section about search strings Rewriting introduction and background
7	3/6/2016	Rayan Sophie	Define search strings, results preliminary search, rewrite search process Start writing intro research design + little additions to manual/QGS and preliminary search + review research design + revision intro, background, selection criteria, data extraction
8	8/6/2016	Rayan Sophie Christian	Refine selection process, moved to new section + data extraction Selection criteria + data extraction Updated research questions, selection process, selection criteria
9	16/6/2016	Rayan Christian	Refining data extraction form and quality assessment Refined research questions, search process, selection procedure, scope of search, data extraction form.
10	1/7/2016	Rayan	Refined data extraction form, small refinements in Research Design

1 Introduction

In the case of large-scale embedded systems projects, in which many architects and other stakeholders, like manufacturing, marketing, or engineering, are involved in the development process, sharing architectural knowledge is a major challenge.

Many approaches to architectural knowledge sharing have been proposed in the recent years. Most of these were developed to work in specific situations, fields or organizations. So far, there exists little research to these different approaches, where they are used or how they tackle the challenge.

Therefore, we want to conduct a systematic literature review (SLR) about approaches for sharing architectural knowledge. A systematic literature review is a secondary study, intended to evaluate all relevant primary studies in a systematic way. The SLR poses questions about the different approaches; to what extent they can be reused elsewhere, the limitations and benefits of the approaches, whether they are based on any empirical evidence and how these approaches are related. The goal of the SLR is to develop a pattern language for architectural knowledge sharing, which specifies good practices for sharing architectural knowledge.

This protocol will specify the design and procedures of the systematic literature review. It poses research questions and describes how relevant articles can be found for the review. The protocol is written according to Kitchenham's guidelines[1] for performing systematic literature reviews in software engineering.

The remainder of this protocol consists of a description of the background of architectural knowledge sharing (Section 2), an overview of the research strategy to be used when performing the actual systematic review, including research questions, search queries, quality assessments and selection criteria (Section 3) and a description of the data extraction, which defines how to search for relevant information in the primary studies selected (Section 4).

2 Background

This section deals with the background of architectural knowledge sharing including a definition of architectural knowledge and a description of architectural knowledge sharing.

2.1 Architectural knowledge and architectural knowledge sharing, a definition

Architectural knowledge does not have a commonly accepted definition, making it hard to define what is included in the concept[10]. De Boer and Farenhorst found in a systematic literature review that most explicitly given definitions indicate that architectural knowledge is limited to architectural design decision making and sometimes extended with the solution or the problem domain[10]. When they looked beyond the provided definitions in the primary studies and looked how the definition was interpreted and used, they found that in practice, the authors of the studies stretched the definition of architectural knowledge from the problem domain to design decision making, to the solution and sometimes even towards the implementation[10]. In order to be able to say anything about architectural knowledge and the approaches to sharing, we need to establish a definition of architectural knowledge for this research.

For a definition of architectural knowledge, we turn to an article written by Kruchten, Lago and van Vliet. They define architectural knowledge as the sum of all design decisions and the design itself[11]. In this SLR we define architectural knowledge sharing as capturing, documenting and communicating architectural knowledge.

2.2 Importance of architectural knowledge sharing

The importance of capturing architectural knowledge has been described in many articles ([12]–[14] to name a few). It is of special importance in projects involving large-scale embedded systems, because there are usually many parties and disciplines involved, the teams are large and the knowledge is

distributed. Despite its well-known importance, architects are known to neglect the documentation of architectural decisions[12], which makes the documentation incomplete and prone to mistakes. Furthermore, if architectural knowledge is not documented and communicated to the stakeholders involved in the process, the knowledge and rationale slowly disappears as the software is further developed or maturing and growing over the years, making it impossible to maintain and improve the architecture[15] and making error detection more difficult or even impossible[15].

2.3 Challenges of architectural knowledge sharing

Much of the available knowledge of architects comes from experience and expertise[12], making it tacit knowledge that is difficult to capture and communicate. It is therefore important that the architectural knowledge can somehow be captured. Nonaka and Takeuchi defined externalization as a solution, making it possible to transform this tacit knowledge to explicit knowledge through dialog and non-conceptual observations[16]. Once the knowledge is explicit, it will be easier to write down and document. Hansen, Nohria and Tierney defined another approach split into two ways of capturing knowledge; codification, which is much like externalization, and personalization, which describes the practice of who knows what[17]. But, even if knowledge is documentable, architects are still reluctant when it comes down to documenting their decisions and everything leading up to this decision [12]. This makes capturing architectural knowledge, or as Babar and Gorton indicated: “the lack of access to knowledge underpinning architecture design decisions and process”, the key challenge in software architecture[15].

Another challenge in architectural knowledge sharing is that the concept is relatively new. Many different definitions, approaches and ideas exist, but nothing has been standardized and many approaches have not been proven to work in different situations. Each approach or idea usually answers one problem, but may create new ones. Also all the different approaches or ideas are often developed to work in a specific situation, field or organization, making it difficult to reuse them. It also leads to a lack of knowledge as to which parts of a solution can be used in other situations. The SLR and the subsequent pattern language are an attempt to solve this challenge.

3 Research Design

In order to be able to objectively say something about different approaches used for architectural knowledge sharing, we will conduct a systematic literature review. This is a secondary study that aims to systematically evaluate all relevant articles available on the subject without bias. In order to do so, a predefined search strategy, selection criteria and a data extraction form are necessary. These are defined in this protocol, and describe how the search for articles should be conducted, how relevant studies should be selected from the retrieved studies and how relevant data can be extracted from the selected articles. With the retrieved data, the conclusion for the systematic literature review can be drawn.

A systematic literature review should not be confused with a systematic mapping study. A systematic mapping study is usually used to give a broad overview of a research area, for instance to determine unexplored fields, whereas a systematic review is narrower and often used to research such unexplored fields. Systematic mapping studies can be of great help determining the areas where systematic literature reviews are useful[1].

The goal of this study has been formulated according to the GQM approach [Basili93] by Basili, which expresses the object of the study, defines the particular angle of the study, and describes the context of the study.

Analyze **the body of literature** for the purpose of **identifying and synthesizing approaches into a pattern language** with respect to **architectural knowledge sharing** from the point of view of **empirical researchers** in the context of **software architecture research.**

3.1 Research Questions

The goal of the systematic literature has been decomposed into the following research questions.

- RQ1: What approaches for sharing AK have been proposed in the literature?
- RQ2: What are the characteristics of the approaches identified in RQ1?
    - RQ2.1: What is the intended project context of the approach?
    - RQ2.2: Which knowledge management strategy does the approach support?
    - RQ2.3: What are the mechanisms of the approach to support sharing of AK?
    - RQ2.4: What are the architectural knowledge entities captured by the approach?
    - RQ2.5: How prescriptive or descriptive is the approach?
    - RQ2.6: What are reported benefits and limitations of the approaches?
    - RQ2.7: What empirical evidence do we have for each approach?
- RQ3: How are the approaches identified in RQ1 related to each other?
- RQ4: What is a pattern language for architectural knowledge sharing?

RQ1 aims at identifying approaches for sharing architectural knowledge. We pose this research question to get an overview of all approaches that have been proposed or that were reported as industrial practice in the literature.

AK sharing approaches are often designed for a particular scenario such as a particular industry or user group. Therefore, RQ2 aims to classify AK sharing approaches through a set of sub-questions. RQ2.1 aims at identifying which project context for which an approach has been designed or is being used. The project context will be classified according to Phillipe Kruchten's Octopus Model for Software Development [Kruchten2011], which includes the following criteria: size, criticality, business model, stable architecture, team distribution, governance, rate of change, and age of system. The model proposed by Kruchten has been specifically proposed to support empirical research on software development methods. Answering RQ2.1 helps us to map approaches and project contexts in order to provide guidance on which AKS approach is suitable in a particular project.

According to Farenhorst and De Boer [18][Farenhorst2009] knowledge sharing can be achieved through codification and/or personalization. They came to the conclusion that a hybrid strategy works best. Therefore, RQ2.2 aims at classifying AK sharing approaches based on the underlying AK management strategy (i.e., personalization, codification or hybrid).

In RQ2.3 we investigate how the approaches functionally support AK sharing. For instance, mechanisms that support sharing of AK could be a central knowledge repository, a publish/subscribe notification system, yellow pages of who knows what, etc.

RQ2.4 aims to identify which architectural knowledge entities are supported by the AK sharing approach. Answering this research question helps us to identify entities that are frequently supported but also to identify entities that are not yet supported by AK sharing approaches. As classification, we use ISO/IEC/IEEE 42010 standard for architecture descriptions. We also include the following two extensions of ISO42010. First, the decision documentation framework proposed by Van Heesch et al. [Heesch2012e,2012d] and the technical debt documentation framework proposed by Li et al. [Li2016]. Architecture decisions are an important aspects of architectural knowledge[Lago]. We include technical debt because it received increasing attention as first-class architectural knowledge entity in recent years. For the same reason, we also add architectural assumptions (Chen et al. [Chen2015]) as a possible AK entity.

RQ2.5 aims at classifying approaches based on how descriptive (flexible, how things actually work) or prescriptive (normative, describing how things should work) they are. Hoorn et al.[Hoorn11] found out that architects prefer descriptive approaches, due to the creative nature of architecture design.
RQ2.6 helps identifying the benefits and limitations of the approaches according to the different problem contexts, which is also useful for getting a good overview of the different methods used for architectural knowledge sharing.
RQ2.7 helps to gain insight about the empirical evidence that exists for AKS approaches. Since we aim to extract patterns of AKS from scientific literature, we cannot identify known uses of the approach in practice. As an alternative, we report the empirical evidence for an approach as means to judge the validity of a pattern.
We ask RQ3 to investigate how different approaches are related to each another. Some approaches might be inspired by others, depend on others, are compatible or incompatible with each other. RQ3 allows us to identify relations between approaches and allows us to identify approaches that can be used together.
RQ4 aims at creating a pattern language for architectural knowledge sharing. A pattern language is a method of describing good practices within a field of expertise, here architectural knowledge sharing. The approaches identified in RQ1, characterized in RQ2 and interrelated in RQ3 will be synthesized into pattern format. Patterns are documented in heterogeneous formats. However, most commonly a pattern consist at least of a description of the *context, problem, solution*, and *known uses* of the pattern [AlexanderXX, BuschmannXX]. RQ2.1 – RQ2.6 help us to establish the *context*, *problem* and *solution* of a pattern. RQ2.7 provides input for *known uses.* RQ3 helps us to identify connections between patterns.

3.2 Search Strategy

The success of a systematic literature review is determined by the ability of the search strategy to identify the relevant research available that is useful for answering one or more research questions[1]. As suggested by Kitchenham[1], we will use multiple search methods for identifying relevant primary studies. Using multiple diverse methods increases our chances to identify studies that would not have been found using a search strategy that relies on a single method. Therefore, our search strategy is comprised of the following methods:

- Preliminary search to assess the volume of potentially relevant studies, and to evaluate the scope, the search terms and the selection criteria,
- Manual search using the selected venues suggested in Section 3.2.2, to establish a subset of the primary studies (quasi-gold standard) which will be used to evaluate the automated search, and
- An automated search using the search strings defined in Section 3.2.3 and the databases suggested in Section 3.2.2, to identify as accurately as possible all the primary studies for the SLR, complemented with a manual search to identify studies that could not be found with automated search.

In the remainder of this section we describe the concrete steps in the search, determine the scope of the search, and establish the search terms for the automated search.

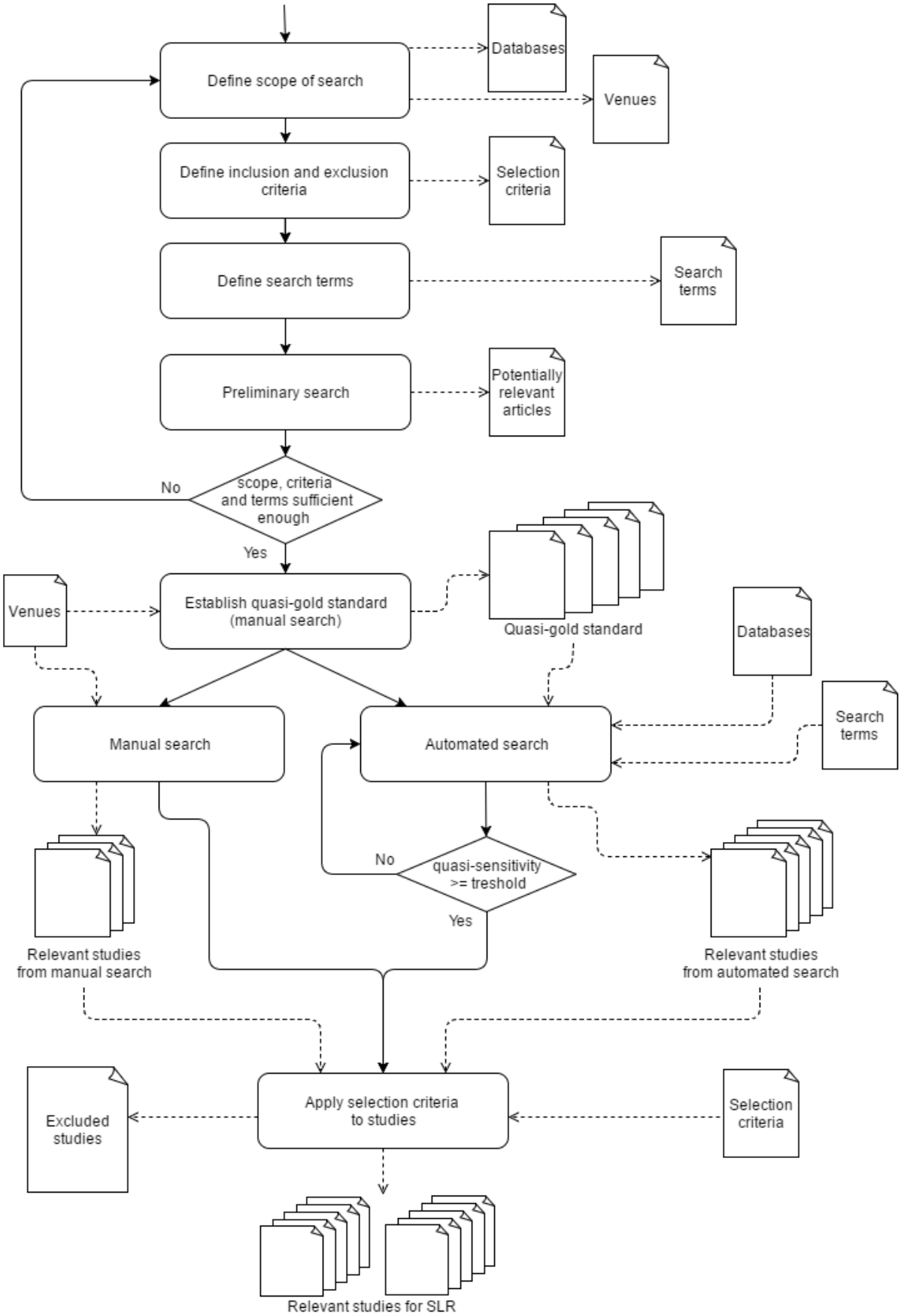
3.2.1 Search Process

As illustrated in Figure 1, our search process will be decomposed into the following steps: defining the scope, the criteria and the search terms; performing a preliminary search; conducting both a manual and automated search; applying the selection criteria; and assessing the quality of the primary studies.

Zhang et al.[6] suggest to use an integrated search strategy using both manual and automated search. The manual search has two goals. The first goal is to establish the quasi-gold standard (subset of the primary studies). The quasi-gold standard (QGS) is an instrument to assess the sensitivity and precision of a search query, which is used to evaluate search terms for the automated search. The second goal is to complement the automated search by searching manually for relevant articles that cannot be found by automatically searching the digital databases.
Next the steps of the search process will be explained.

Defining the scope, selection criteria and search terms: The first step in an SLR is to define the scope of the search (cf. Section 3.2.2), the selection criteria (cf. Section 3.2.4), and the search terms (cf. 3.2.3). We will conduct a preliminary search to evaluate the scope of search, the selection criteria, and search terms.
The preliminary search is not systematic in the sense that it does not follow a strict process or conventions. It has the goal to get a general overview of the relevant studies and important keywords in the field of study, to identify venues and databases that are relevant for the search, to familiarize with databases and their query interfaces, to define the selection criteria that distinguishes relevant from irrelevant studies, as well as to try out different search terms and synonyms of search terms.
The first steps are iterative, which means that the scope, terms and criteria will be redefined until they are considered adequate in order to progress to the manual and automated search.
Since conducting a systematic literature review is regarded an iterative process, it is possible that the scope, selection criteria, and even the search terms are refined in a later step (e.g., if the automated search yields too few primary studies).

Figure 1. Flow of different steps in the search process.



Establish Quasi-gold Standard: The quasi-gold standard represents a subset of all the primary studies for an SLR. The QGS will be established by manually searching every selected venue defined in our scope (cf. Section3.2.2). Because every database represents different venues, every database also has its own QGS. The identified studies found by manually searching the selected venues will be grouped by their respective database. Consequently, this will form a QGS for a database.

Zhang et al. [6] suggested the quasi-gold standard concept as means to evaluate the search strategy. The idea of the quasi-gold standard concept is that the same articles will be found with the automated search as that are included in the QGS. This way the search terms can be tested. When evaluating the search terms, the automated search will first be limited to search only for the one or more venues that were used to establish the QGS. When it is shown that the search terms are sufficient to find the articles included in the QGS, the actual automated search can be conducted by removing this limitation.

This concept suggested by Zhang et al.[6] is an objective way for evaluating the search strategy. This objective method assesses the search performance by quantitative criteria whereas by a subjective method the assessment of the performance depends on the reviewer's knowledge[6]. The quasi-gold standard is a concept derived from the gold standard used for improving literature search processes in systematic literature reviews in other disciplines, such as medicine. This gold standard concept is using two important criteria: sensitivity and precision. Sensitivity is defined as the ability to identify all the relevant studies for our SLR. Precision means the amount of relevant studies that are retrieved among all the retrieved studies. There are two additional constraints attached to quasi-gold standard compared with the gold standard, which are defined by asking the questions 'where' and 'when'[6], meaning the venues and time span used in the review should be defined.

In the QGS concept the quasi-sensitivity has to be calculated, which refers to the total number of relevant articles retrieved by the automated search divided by the number of articles in the quasi-gold standard (relevant articles retrieved by the manual search), as shown in Formula 1. According to Zhang et al. [6] an acceptable sensitivity is between 70% and 80%. If the calculated quasi-sensitivity is lower than this threshold, the search terms need to be refined until it reaches the desired threshold. Dieste and Padua [7] propose four different strategies to balance the amount of irrelevant articles retrieved and the ability to identify all of the relevant articles. We aim for the optimum strategy, which "strikes a balance between high sensitivity and high precision[7]". The optimum strategy has a sensitivity range of 80-99% and a precision range of 20-25%.

Formula 1. Quasi-sensitivity.

$$Quasi\text{-}sensitivity = \frac{Number\ of\ relevant\ studies\ retrieved\ by\ automated\ search}{Number\ of\ articles\ in\ Quasi-gold\ standard} * 100\%$$

Manual and Automated Search: According to the guidelines by Kitchenham[1] only searching digital databases automatically is not sufficient enough for a systematic literature review. Therefore, we use an integrated search strategy as suggested by Zhang et al.[6]. An integrated search strategy means that an automated search will be complemented with a manual search to identify studies that could not be found with the automated search and thereby increase the search perimeter. We will manually check the venues listed in Section 3.2.2 and perform a "snowball" search on the studies identified by searching the databases (automated search) and venues (manual search). Snowball search is a technique, in which the citied studies (backwards snowballing) and citing studies (forward snowballing) of an already identified primary study are checked if they are relevant for the SLR [Wohlin2014]. The automated search will be performed using the constructed search terms defined in Section 3.2.3.

Study Selection and Quality Assessment: In the final step of the search process, the previously identified studies are conclusively included or excluded from the SLR and their empirical quality is being assessed.

The study selection is performed based on a set of inclusion and exclusion criteria, which specify whether or not a study helps in answering the specified research questions. The concrete selection process is detailed in Section 3.3. The result of the study selection is a list of distinct studies relevant to answering the research questions.

After study selection, the primary studies for the SLR are established. For the primary studies the empirical quality should be assessed. According to Brereton et al. [Brereton2007a], the goal of the quality assessment is twofold. First, it acts as a criterion to exclude studies which are of questionable empirical quality. Second, it allows the weighting of specific studies during the data synthesis stage. In this study we do not use the quality assessment score as a criterion to exclude a study because we explicitly include non-empirical studies (cf. Section 3.2.2). Therefore the quality assessment is also not included in the diagram in Figure 1. The quality assessment procedure is explained in Section 3.4.

3.2.2 Scope of Search

The scope of the search determines the relevant research databases, journals, and venues and specifies the time period under scrutiny by the search.

Since the subject of SLR lies in the domain of software engineering, the automatic search is performed on databases which are commonly used for SLRs in software engineering, according to Brereton et al. [3] and Zhang et al. [6]. Table 1 lists all databases that are automatically searched in this SLR.

In addition to databases, which allow us to automatically retrieve a list of potentially relevant studies, we also include venues, primarily targeted by software architecture researchers, which are manually searched. Zhang et al. [6] investigated the most used venues that were used twice or more for doing manual search in SLRs in SE. We use the list proposed by Zhang et al. as a starting point to identify venues that are relevant in the context of this SLR. This list is extended with our own perception of relevant venues. Table 2 lists all venues considered for the manual search.

The time period relevant for the manual search and automatic search are studies published between 2004 and July 2016. We chose the lower bound because Capilla et al. [19] found that architectural knowledge management as a research area emerged around 2004. Therefore, we do not expect a considerable number of studies related to architectural knowledge sharing published before 2004 that would justify the additional effort associated with extending the timeframe of the search. We chose the upper bound to address reliability issues regarding the repeatability of our search process.

Since this SLR aims to distill a pattern language from the body of literature, we broaden the scope of the study to also include non-empirical studies in order to get an extensive overview of approaches for architectural knowledge and to acquire as much information about a single approach as possible.

Table 1. List of databases for automatic search

Name	Type	Rationale for Inclusion
IEEE Xplore	Database	Brereton et al. [3]
ACM Digital Library	Database	Brereton et al. [3]
ScienceDirect	Database	Brereton et al. [3]
GoogleScholar	Database	Brereton et al. [3]
SpringerLink	Database	Brereton et al. [3]
Web of Science	Database	Zhang et al. [6]
Wiley InterScience	Database	Zhang et al. [6]

We did not include the following commonly used databases in software engineering SLRs because the RUG has no subscription for these databases

- El Compendex
- Inspec
- SCOPUS
- Kluwer
- Citiseer library
- Keele University's electronic library

Table 2. List of journals and venues for manual search

Name	Abbreviation.	Type	Database/Catalog
IEEE Software	IEEE Software	Jour.	IEEE Explore
Empirical Software Engineering	ESEM	Jour.	SpringerLink
International Conference on Software Engineering	ICSE	Conf.	IEEE Xplore
Journal of Systems and Software	JSS	Jour.	ScienceDirect
International Journal on Software Engineering and Knowledge Engineering	JSEKE	Jour.	Worldscientific
Working IEEE/IFIP Conference on Software Architecture	WICSA	Conf.	IEEE Xplore
International Conference on Software Engineering & Knowledge Engineering	SEKE	Conf.	DBLP
European Conference on Software Architecture	ECSA	Conf.	SpringerLink LNCS

We did not consider the following journals and venues commonly used in SE SLRs[6] for manual search because they are not primarily targeted by software architecture researchers.

- IEEE Computer
- Metrics (module of Scopus)
- TOSEM (Transactions on Software Engineering and Methodology)
- ESE (Empirical Software Engineering)
- WWW (World Wide Web - Internet and Web Information Systems)
- ICSM (International Conference on Superconductivity and Magnetism)
- MISQ (Management Information Systems Quarterly)
- TSE (Transactions on Software Engineering)
- IESE (Instituto de Estudios Superiores de la Empresa)

3.2.3 Search Terms

Based on the results of the preliminary search and by considering the research questions the following terms are derived to construct the search string:

Software architecture; architectural knowledge; architecture knowledge; architectural design decisions; architectural design; sharing

The terms listed above are used to construct the main search string. Because architectural knowledge (AK) includes architectural design (AD) and architectural design decisions (ADD), we also include these two terms in the search string. Consequently, our constructed search string contains at least the parts

"software architecture" and "sharing", which is shown by the Boolean AND. The other keywords are separated by the Boolean OR which implies that at least one of the terms should match with the articles.
The main search string is as follow:

"software architecture" **AND**
"sharing" **AND**
("architecture knowledge" **OR** "architectural knowledge" **OR** "architectural design decisions" **OR** "architectural design")

Depending on the results of search string evaluation using the QGS, the search string above might be adapted to get more sufficient results from the different databases.
If supported by the search engine, the search query is applied to the full text.

3.3 Selection Process

This section will describe the selection criteria and procedure of the selection process for the systematic literature review. It describes how to determine which studies should be included and which should be excluded for the review.
For some articles we might want to include the conclusion because abstracts in Software Engineering are mostly not structured and do not contain the results and the conclusion of the research[3]. Consequently, you cannot always gain enough information about the content and the relevance of an article from the abstract alone. That is why you should not only rely on the abstract when selecting primary studies, but also assess the conclusion when necessary[3].

The selection of the primary studies used in this SLR is a three-stage process.

- **First Stage:** The list of studies retrieved in the automated or manual search will be stored in a list including (at least) title, authors, year, abstract and keywords, URL to pdf. The title, keywords and abstract will be individually reviewed by two researchers indicating if a paper is relevant or irrelevant for the SLR. In the first stage, the selection should be performed generously, if there are doubts the article should preferably be included rather than excluded.
- **Second Stage:** All studies that were selected by at least one reviewer in the first round, will be reevaluated based on the full text of the study.
- **Third Stage:** In the third round, the results of the second round will be compared and differences in the judgment of relevancy will be discussed. If no agreement can be reached, a third researcher will be consulted to resolve the conflict.

3.3.1 Inclusion Criteria

For a study to be included in the review, every inclusion criterion has to apply.

- A study presents an approach for sharing architectural knowledge, or a study evaluates/compares one or multiple approaches for sharing architectural knowledge, or a study is about the state of the art of AK sharing or industrial practices w.r.t to AK sharing.

3.3.2 Exclusion Criteria

If one or more of the exclusion criteria apply to a study, it will be excluded from the review.

- A study is not written in English.

- A study is about knowledge sharing in general and not specifically about architectural knowledge.
- A study has been published before 2004 or after July 2016 (cf. Section 3.2.2).
- The study is an editorial, position paper, abstract, keynote, opinion, tutorial summary, panel discussion, or technical report. A paper that is not a peer-reviewed scientific paper may not be of acceptable quality or may not provide reasonable amount of information.
- A study is a duplicate of an already selected primary study.

3.4 Quality Assessment

Additional to the selection criteria, the quality of the primary studies will be assessed. This quality assessment will be performed the same way as done by Ali et al. [20], who adopted the quality assessment defined by Dybå and Dingsøyr [8]. The quality of the primary studies will be assessed using a checklist containing nine questions. In this phase of the process no studies will be excluded, because of the inclusion of non-empirical studies (cf. Section 3.2.2). The final score of these questions represents a weight of the importance of these studies and will be included in the data extraction form. The quality score is also used for answering RQ2.7.

The same scale for the quality check is used as suggested by Ali et al. [20]. There are three possible answers to the questions in Table 2, which are 'Yes', 'No' or 'To some extent'. Numerical values are used to calculate the final quality score of a study. These values are 'yes'=1, 'no'=0, and 'to some extent'=0.5. The final score will be the sum of the scores of all the questions.

The quality assessment will be conducted by one reviewer in parallel with the data extraction. A second reviewer will check the quality assessment. If it appears that the reviewers disagree on the quality of a paper, they have to discuss this disagreement and come to an understanding.

Table 2. Quality checklist

Q1	Is there a rationale for why the study was undertaken?
Q2	Is there an adequate description of the context (e.g. industry, laboratory setting, products used, etc.) in which the research was carried out?
Q3	Is there a justification and description for the research design?
Q4	Has the researcher explained how the study sample (participants or cases) were identified and selected, and what was the justification for such selection?
Q5	Is it clear how the data was collected (e.g. through interviews, forms, observation, tools, etc.)?
Q6	Does the study provide description and justification of the data analysis approaches?
Q7	Is there a clear statement of the findings and has sufficient data been presented to support these findings?
Q8	Did the researcher critically examine their own role, potential bias and influence during the formulation of research questions, sample recruitment, data collection, and analysis and selection of data for presentation?
Q9	Do the authors discuss the validity and limitations of their findings?

4 Data Extraction

For the data extraction process, a data extraction form is used, as is advised in the Guidelines[1]. For each study found to be relevant by the selection criteria, this form has to be filled in.
Whenever possible, two or more researchers will independently perform the data extraction. For each article, one will be appointed the data extractor, and the other the data checker. The data extractor completes the data extraction form and the data checker confirms whether the data on the form is correct. Whenever it is not possible to have two or more researchers assessing each study, a random subset of the studies is to be assessed by at least one other researcher.
The findings from each researcher of each article must be compared and disagreements must be resolved either by consensus or by an external independent researcher or by the writer(s) of the study. This process of data extraction is as advised by the Guidelines [1] and turn out to be good practice according to P. Brereton et al.[3]. The data extraction form can be found in Table .
When a field in the data extraction form cannot be filled in please state the reason why, e.g. not reported in the study, not applicable, no constraints etc.

Table 3. Data Extraction Form

ID	Name	Value(s)	Explanation	Related RQ
1A	Title			
1B	Author(s)			
1C	Year of publication			
1D	Source			
1E	Abstract			
1F	Keywords			
2A	Who conducted the review?			
2B	Who is the data checker?			
2C	How was this study found?	MS[Venue] AS[Database] SB	MS = Manual Search AS = Auto. Search SB = Snowballing Multiple options possible.	
2D	Quality Score	0-9	According to Section 3.4	RQ2.7
3A	Which AKS approach(es) are/is discussed in the study?	Free Text	If no name is provided, fill in NO NAME. If more than one approach, number the approaches. (Also for the following questions)	RQ1
3B	Description of AKS approach(es) discussed in the study?	Free Text (max. 250 Words per approach)	Describe the approach(es) in your own words .	RQ1
4A	Size	Size of system, Size of team, Number of person-months, or development budget etc.	The overall size of the system under development is by far the greatest factor, as it will drive	RQ2.1

			in turn the size of the team, the number of teams, the needs for communication and coordination between teams, the impact of changes, etc. Larger teams are typically formed to address more complex problems.	
4B	Criticality	None, Life critical, Other, No constraints, Not reported	How many people die or are hurt if the system fails? A team may need to conform to financial regulations, privacy regulations, or even life-critical regulations.	RQ2.1
4C	Business Model	Internal, Commercial, Contracted, Open Source, Not reported, Other	Are you developing an internal system to support your internal processes, or a commercial product, or a bespoke system on contract for a customer, or maybe a component of a large system involving many different parties? Or a free-libre open-source (FLOSS) project?	RQ2.1
4D	Stable Architecture	Implicit Architecture, Reference Architecture, Normal Architectural, Volatile Architecture, Not reported, Other	Is there an implicit, architecture already in place at the start of the project? Most projects are not novel enough to require architectural effort. Some projects suffer from constant design volatility.	RQ2.1
4E	Team Distribution	Co-located, Offshore Outsourced, No constraints, Not reported, Other	How many teams are involved and are not collocated.	RQ2.1

4F	Governance	Structural, Dynamic, No constraints, Not Reported, Other	How are projects started, terminated? Who decide what happens? Who manages the software project manager? Are there external rules and regulations imposed on the product?	RQ2.1
4G	Rate of Change	Low, Medium, High, Not reported	How stable is the business environment and how much risk and uncertainty is the project facing?	RQ2.1
4H	Age of System	Exploratory, greenfield, brownfield, legacy maintenance, Not reported, Other	Multiple Choice	RQ2.1
5A	What knowledge management strategy is supported by the approach?	Personalization, codification, or hybrid		RQ2.2
6A	What are the mechanisms described in the study to support AKS?	Free Text	e.g. knowledge repository, a publish/subscribe notification system, yellow pages, viewpoint-based	RQ2.3
7A	What AK entities will be captured by the approach?	Technical Debt, Stakeholder, Assumption, Decision, Alternative, Decision Topic, Force, Concern, etc.	The complete list of entities must be extracted from the paper.	RQ2.4
8A	IS the approach more prescriptive or descriptive?	Prescriptive or descriptive		RQ2.5
8B	Motivate your answer to 8a.	Free Text		RQ2.5
9A	What are the benefits of the approach reported in the study?	List of benefits		RQ2.6

9B	What are the limitations of the approach reported in the study?	List of limitations		RQ2.6
10A	Did the article describe any case studies or experiments?	Yes/No		RQ2.7
10B	If yes, give short description of experiment/case study.	Free text		RQ2.7
10C	What kind of data is collected by the researchers?	Observations, Interviews, Questionnaire, Recordings, Not reported, Other		RQ2.7
10D	What were the results of the case study/experiment reported in the article?	Free text	What did they conclude from the results of the case study/experiment? Positive and/or negative results?	RQ2.7
11A	Is this approach compared with another approach in the study?	Yes/No		RQ3
11B	If yes, list the approaches and the outcome of the comparison.	Compared approaches and outcome of comparison.		RQ3
12	Additional useful information.		If no approach is suggested in the study, describe here why this study is still relevant and what useful information is given.	

Question block 1 is only for documentation purposes and do not relate to the data to be extracted. Question block 2 is about how the review was conducted. Question blocks 3 is for answering RQ1. Question block 4 helps describing the problem context of the approach (cf. RQ 2.1). Question blocks 5 to 10 are related to the research questions 2.2 till 2.7 and question block 11 is related to RQ3. Question block 12 gives the possibility to include relevant information that could not be answered by the former question blocks.
The Guidelines [1] suggest creating an electronic form for the data extraction.

References

[1] B. Kitchenham and S. Charters, "Guidelines for performing Systematic Literature Reviews in Software Engineering," *Engineering*, vol. 2, p. 1051, 2007.

[2] H. Zhang and M. Ali Babar, "Systematic reviews in software engineering: An empirical investigation," *Inf. Softw. Technol.*, vol. 55, no. 7, pp. 1341–1354, 2013.

[3] P. Brereton, B. A. Kitchenham, D. Budgen, M. Turner, and M. Khalil, "Lessons from applying the systematic literature review process within the software engineering domain," *J. Syst. Softw.*, vol. 80, no. 4, pp. 571–583, 2007.

[4] S. MacDonell, M. Shepperd, B. Kitchenham, and E. Mendes, "How Reliable Are Systematic Reviews in Empirical Software Engineering?," *IEEE Trans. Softw. Eng.*, vol. 36, no. 5, pp. 676 – 687, 2010.

[5] B. Kitchenham, R. Pretorius, D. Budgen, O. P. Brereton, M. Turner, M. Niazi, and S. Linkman, "Systematic literature reviews in software engineering-A tertiary study," *Inf. Softw. Technol.*, vol. 52, no. 8, pp. 792–805, 2010.

[6] H. Zhang, M. A. Babar, and P. Tell, "Identifying relevant studies in software engineering," *Inf. Softw. Technol.*, vol. 53, no. 6, pp. 625–637, 2011.

[7] O. Dieste and A. G. Padua, "Developing Search Strategies for Detecting Relevant Experiments for Systematic Reviews," *Proc. - 1st Int. Symp. Empir. Softw. Eng. Meas. ESEM 2007*, pp. 215–224, 2007.

[8] T. Dybå and T. Dingsøyr, "Empirical studies of agile software development: A systematic review," *Inf. Softw. Technol.*, vol. 50, no. 9–10, pp. 833–859, 2008.

[9] P. Kruchten, "The Frog and the Octopus — Experience Teaching Software Project Management," *Proc. Can. Eng. Educ. Assoc.*, 2011.

[10] R. C. de Boer and R. Farenhorst, "In search of 'architectural knowledge,'" *SHARK 2008 - 3rd Int. Work. Shar. reusing Archit. Knowl.*, p. 71, 2008.

[11] P. Kruchten, P. Lago, and H. Van Vliet, "Building up and reasoning about architectural knowledge," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 4214 LNCS, pp. 43–58, 2006.

[12] R. Capilla, A. Jansen, A. Tang, P. Avgeriou, and M. A. Babar, "10 years of software architecture knowledge management: Practice and future," *J. Syst. Softw.*, vol. 000, pp. 1–15, 2015.

[13] A. Tang, P. Avgeriou, A. Jansen, R. Capilla, and M. Ali Babar, "A comparative study of architecture knowledge management tools," *J. Syst. Softw.*, vol. 83, no. 3, pp. 352–370, 2010.

[14] M. A. Babar, T. Dingsøyr, P. Lago, and H. Van Vliet, "Software Architecture Knowledge Management," *Softw. Archit. Knowl. Manag. Theory Pract.*, pp. 1–279, 2009.

[15] M. A. Babar and I. Gorton, "Architecture Knowledge Management : Challenges , Approaches , and Tools," 2007.

[16] I. Nonaka and H. Takeuchi, *The Knowledge-Creating: How Japanese companies create the dynamics of innovation*, vol. 3. 1995.

[17] M. T. Hansen, N. Nohria, and T. Tierney, "What ' s Your Strategy for Managing Knowledge?," *Harv. Bus. Rev.*, vol. 72, no. 2, pp. 106–116, 1999.

[18] R. Farenhorst and R. C. de Boer, "Chapter 2: Knowledge Management in Software Architecture: State of the Art," *Softw. Archit. Knowl. Manag.*, pp. 21–38, 2009.

[19] R. Capilla, A. Jansen, A. Tang, P. Avgeriou, and M. A. Babar, "10 years of software architecture knowledge management: Practice and future," *J. Syst. Softw.*, vol. 000, pp. 1–15, 2015.

[20] M. S. Ali, M. Ali Babar, L. Chen, and K.-J. Stol, "A systematic review of comparative evidence of aspect-oriented programming," *Inf. Softw. Technol.*, vol. 52, no. 9, pp. 871–887, 2010.