You Can't Make Bricks Without Straw: Designing Systematic Literature Search Systems

Short Paper

Benjamin Sturm
University of Kassel
Mönchebergstr. 19, 34109 Kassel, Germany
bsturm@uni-kassel.de

Ali Sunyaev
University of Kassel
Mönchebergstr. 19, 34109 Kassel, Germany
sunyaev@uni-kassel.de

Abstract

Although a rigorous review of literature is essential for any research endeavor, technical solutions that support systematic literature review approaches are still scarce. Systematic literature searches in particular are often described as complex, error-prone and time-consuming, due to the prevailing lack of adequate technical support. In this study, we therefore aim to learn how to design information systems that effectively facilitate systematic literature searches. Using the design science research paradigm, we develop design principles that intend to increase comprehensiveness, precision, and reproducibility of systematic literature searches. The design principles are derived through multiple design cycles that include the instantiation of the principles in form of a prototype web application and qualitative evaluations. Our design knowledge could serve as a foundation for future research on systematic search systems and support the development of innovative information systems that, eventually, improve the quality and efficiency of systematic literature reviews.

Keywords: Design science, literature review, information retrieval, systematic literature searches

Introduction

Building upon the existing knowledge base is essential for any academic project (Webster and Watson 2002). Hence, most research publications contain a review of literature in some form, either to ground the main study or as a stand-alone research contribution (Okoli and Schabram 2010). Literature reviews can be conducted in different ways, ranging from highly systematic (Kitchenham et al. 2009; Okoli and Schabram 2010) to more traditional or narrative approaches (e.g., Bandara et al. 2011; Ridley 2012; Webster and Watson 2002). Which method is best suited for a specific review, depends on individual aspects, such as the specific research question, available resources, and the topic under review (Boell and Cecez-Kecmanovic 2015; Okoli and Schabram 2010). There is, however, broad consensus that any high-quality literature review requires a systematic approach to some degree (i.e., a rigorous, methodical approach) (e.g., Fink 2014; Levy and Ellis 2006; Okoli and Schabram 2010; vom Brocke et al. 2015). Unsystematic reviews tend to be subjective, give little justification for the selection and analysis of a specific literature sample, and are often based on a partial or superficial examination of the available literature, which might result inaccurate or even false findings (Fink 2014; Levy and Ellis 2006). One of the first steps to prevent such consequences is the selection of an appropriate literature sample using a rigorous, systematic search approach (Levy and Ellis 2006). Carelessness during the search process often leads to an outdated, scattered, and irrelevant literature sample, a shortcoming that cannot be compensated by
subsequent review steps (Levy and Ellis 2006). However, applying a systematic search approach is often a complex and time-consuming task, especially for students and novice researchers (Fink 2014; vom Brocke et al. 2015).

Several factors contributed to the high complexity of systematic literature searches. A common starting point of reviewers (i.e., researchers who conduct literature reviews) are curated literature databases, like the IEEEXplore Digital Library or the AIS Electronic Library. However, due to their limited individual coverage, a rigorous, systematic database search usually requires querying multiple literature databases, each with its own peculiarities (e.g., available features, search fields, and query syntax). As a result, the necessary knowledge and effort to prepare and execute search requests and manage results as well as the risk of making mistakes multiply with each additional database (Fink 2014; vom Brocke et al. 2015). This is an even bigger issue in interdisciplinary research areas, like the information systems (IS) field, where scientific contributions are published in a wide variety of outlets (e.g., journals and conference proceedings), which are dispersed over numerous databases (Barnes 2005; vom Brocke et al. 2015). Covering the 50 top-ranked IS journals (AIS 2017) requires at least six different databases to be queried. Alternative search systems with increased coverage, like academic web search engines or discovery services, have been found to be of little use when following a systematic approach. The main issues are their lack of transparency, oversimplified interfaces, and unreliable search results (e.g., Asher et al. 2013; Hjorland 2015; Lewandowski 2010). However, apart from research evaluating existing solutions (e.g., Boeker et al. 2013; Giustini and Kamel Boulou 2013), research on the design of new systematic literature search systems (SLSS) for the specific purpose of facilitating systematic literature searches (i.e., systems that enable comprehensive and objective literature overviews) is scarce. Extant design research focuses for instance on user-system interaction (e.g., Atanassova and Bertin 2014; Gamberini et al. 2015), scientific paper recommender functions (e.g., Beel et al. 2013; Huang et al. 2014), and prototype systems to support synthesis and analysis of research articles (e.g., Jonnalagadda et al. 2013; Larsen and Bong 2016). While these efforts seek to assist researchers during the literature review process, systems or individual features for the specific purpose of conducting systematic searches to acquire a comprehensive and objective literature overview are not investigated. Yet, a better understanding of the design and effects of SLSS would provide new design knowledge on this class of systems, insights into why existing systems fail to sufficiently aid reviewers, and guidance on the construction of new information systems that improve the quality of systematic literature reviews.

To address this gap in the design knowledge base, our overarching research project tries to answer the following question: How to design a SLSS that effectively facilitate systematic literature searches? To answer this question, we base our research on the design science research (DSR) paradigm (Gregor and Hevner 2013; Hevner et al. 2004) and develop materiality oriented design principles (Chandra et al. 2015), which describe the shape and features of SLSS. Our research method consists of multiple design cycles (DC) of artifact development, evaluation, and refinement. This paper focuses on the results of the second DC, containing a first set of design principles for SLSS along with their instantiation in form of a prototype web application. The design principles are intended to be of use for both researchers and system developers. Our research thereby contributes to both science and practice. We contribute to science by providing novel design knowledge on SLSS that may serve as a starting point for future research on SLSS. We contribute to practice by introducing design principles that can guide the development of innovative information systems aiding reviewers in conducting systematic literature searches and, eventually, help to improve the quality and efficiency of systematic literature reviews.

Research Background

Supportive Systems for Systematic Literature Searches

Design knowledge drawn from existing artifacts addressing related (research) problems is an important information source in DSR projects (Gregor and Hevner 2013). Related to our research problem, different tools exist that facilitate literature searches, besides visiting a local library and communicating with peers. Commonly used systems are online literature databases (e.g., IEEEXplore, MEDLINE, or AISel), discovery services and meta-search engines (e.g., EBSCO Discovery Service or ProQuest’s Summon), digital library catalogues (e.g., ERIC or DPLA), and academic web search engines (e.g., Google Scholar or Microsoft Academic Search). Despite the variety of available systems, conducting a rigorous, systematic literature
Designing Systematic Literature Search Systems

search remains a challenging task. As aforementioned, literature databases and digital libraries provide only limited coverage due to a narrow topical focus (Asher et al. 2013; Boell and Cecez-Kecmanovic 2014; Lowry et al. 2013). Querying multiple literature databases is therefore often unavoidable. Since each database interface has its own set of features and rules (e.g., available features, search fields, and query syntax), creating semantically similar queries for multiple databases and merging their results is considered a highly complex task involving a steep learning curve (Fink 2014; Rowley and Slack 2004; vom Brocke et al. 2015). Hence, it is not surprising that discovery services (i.e., faceted search systems) and meta-search engines, which increase the reach of individual search requests, are gaining popularity among researchers (Pontis et al. 2015; Spezi 2016). These services cover a large body of literature by combining multiple data sources into a single meta-index and provide access through a unified search interface. As a result, searches are more efficient and extensive in comparison to multiple database searches (Asher et al. 2013; Olson 2007; Wells 2016). However, when considered as means for more systematic search approaches, discovery services are criticized for their low transparency (e.g., inaccessible title lists) as well as their oversimplified and imprecise search interfaces (e.g., limited advanced search functionality and export restrictions) (Asher et al. 2013; Fagan et al. 2012; Sesagiri Raamkumar et al. 2016; Wells 2016). Even more criticized are academic web search engines, i.e., special-purpose search engines crawling the entire Internet for scientific contributions. Despite their high coverage of scientific outlets, which surpasses most indexes of individual literature databases (e.g., Bramer et al. 2013; Samadzadeh et al. 2013), academic web search engines are widely criticized for their minimalistic search interfaces, fluctuating and non-transparent search indexes, low document quality, and export limitations (e.g., Asher et al. 2013; Boeker et al. 2013; Lewandowski 2010; Wu and Chen 2014). As a result, academic web search engines are ill-suited for rigorous literature searches (Boeker et al. 2013; Gehanno et al. 2013; Lewandowski 2010).

Research on systems that specifically help researchers to conduct systematic searches is also scarce. Most research on the topic evaluates existing systems regarding their fit for different search tasks (Boeker et al. 2013; Bramer et al. 2013; Falagas et al. 2007; Giustini and Kamel Boulos 2013; Samadzadeh et al. 2013), provide practical guidelines on where to search (Levy and Ellis 2006; Schryen 2015) and how to use existing systems more effectively (Bandara et al. 2015; Bandara et al. 2011; Wolfsinkel et al. 2013). Research on the design of literature search systems (i.e., prototype systems and features) comprises retrieval systems with high user interaction (Yuan and Belkin 2010a; Yuan and Belkin 2010b), search systems with faceted or symbiotic interfaces (Atanassova and Bertin 2014; Gamberini et al. 2015), scientific paper recommender systems (Beel et al. 2013; Huang et al. 2014; Kılıçkurt and al. 2013; Naak et al. 2008), systems to support synthesis and analysis of research articles (Blake and Pratt 2006; Jonnalagadda et al. 2013; Larsen and Bong 2016), meta-search engines for individual full-text articles (On and Lee 2004; Santos et al. 2010), specialized web crawler for indexing research papers (He and Hui 2001; Hoff and Mundhenk 2001; McCallum et al. 2000), systems for visualizing citation networks (Chou and Yang 2011), and citation analysis tools for mining academics’ social networks (Chen et al. 2011; Tang et al. 2007; Tang et al. 2008). Although these research contributions help us to better understand system designs and retrieval techniques in general, insights on how to design effective systems for the specific purpose of conducting systematic rigorous searches are not provided.

**Ingwersen’s Cognitive Model of Information Retrieval Interaction**

Based on what we learned so far, it becomes clear that an effective SLSS design has to consider both the technical aspects of the information retrieval (IR) process and reviewers’ search strategies, goals, and behavior. Ingwersen’s (1996) cognitive model of IR interaction might help to better understand this socio-technical design perspective. Based on ideas from cognitive psychology, the model identifies interactions between different actors during information search processes while also integrating system design issues (Wilson 1999). As depicted in Figure 1, IR interactions involve communication on a cognitive level between human actors (individual users and social/organizational environment), technology artefacts (IR systems and interfaces), and information objects. In the center of Ingwersen’s model are the users that seek information. The users’ cognitive models comprise, for instance, their current information needs, information behavior, problems, and goals. The cognitive models of technical artifacts (e.g., retrieval techniques and database structures) and information objects (e.g., knowledge representation) are explications of the cognitive models of their creators (i.e., system designers or authors). Similar to the Task-Technology Fit Theory (Goodhue and Thompson 1995) and the Cognitive Fit Theory (Vessey and Galletta 1991), Ingwersen (1996) proposes that *fit* is an essential condition for effective IR interactions, in this case...
the fit between the actors’ cognitive models. Inconsistencies between the cognitive models increase the interaction effort, which results in uncertainty and misunderstandings between the actors. For instance, an IR system’s definition of a search task (instantiated as specific search algorithms or interface design elements) might not fit a user’s individual information needs and, thus, either forces the user to adjust his/her information needs or find a sufficient workaround. The cognitive models of users are, therefore, a valuable input for successful IR system designs (Ingwersen 1996). However, it is difficult to provide a perfect fit for each individual user, if the IR system targets a large user group. Common factors that influence the cognitive models of all users could therefore provide a better starting point for actual design requirements. As depicted in Figure 1, one of the major influences (directly or indirectly) is the users’ social and organizational environment. All retrieval interactions take place in the context of a social or organizational environment, which change the cognitive models of system users, creators of information objects, intermediaries, and systems designers, and, consequently, influence all interactions within information search and retrieval processes (Ingwersen 1996). Thus, knowledge about the environment’s (collective) cognitive model (e.g., strategies, goals, tasks, and preferences) is highly relevant for effective search system designs. In the literature search context, the environment usually relates to the academic fields from which the users’ information needs arise (Ingwersen 1982). When it comes to SLSS, it is reasonable to assume that the cognitive models of the targeted research fields are of particular relevance, as the results of systematic literature searches are normally intended for later publication. Hence, reviewers should have increased incentive to establish conformity with their social environment to improve communication and increase acceptance of their research. Search strategies, goals, preferences, etc. (i.e., the cognitive model) of targeted research fields could therefore provide valuable knowledge on SLSS meta-requirements. The following section describes how we incorporated this implication into our DSR design.

![Diagram of Information Retrieval Interaction](image)

**Figure 1. Cognitive Model of Information Retrieval Interaction (figure adapted from Ingwersen (1996))**

**Research Method**

Based on the DSR paradigm (Gregor and Hevner 2013; Hevner et al. 2004), our research method comprises multiple DC. The first DC was informed by initial requirements from the application domain. We identified problems and opportunities through a requirements workshop with seven researchers from the IS field. Furthermore, we reviewed extant research on information search and retrieval systems and investigated existing artifacts in the application domain. Based on our insights a first prototype application was developed and afterwards evaluated through an expert review with five IS researchers and developers. The results of the expert review demonstrated the technical feasibility of the prototype and showed a necessity for further refinements (e.g., improvements of the search process and usability). Building on the knowledge elicited from the first DC, the goal of the second DC was to develop a first set design principles for SLSS. Design principles serve as an abstract blueprint for the construction of design products or methods (Gregor and Jones 2007). The principles developed in this paper can be classified as materiality oriented design principles (Chandra et al. 2015). These principles describe the shape and features of an artifact rather than the intended use of the artifact (i.e., action oriented design principles) (Chandra et al. 2015), similar to principles of form and function (Gregor and Hevner 2013; Gregor and Jones 2007). However, before design
principles can be developed, a clear understanding of the purpose of a design artifact in form of meta-requirements is needed (Gregor and Hevner 2013; Gregor and Jones 2007). Considering the strong influence of a social and organizational environment on the entire information search and retrieval processes, as described by Ingwersen (1996), meta-requirements for SLSS should reflect acknowledged quality criteria for the search process and its results (i.e., strategies and goals as part of the collective cognitive model). To expand our initial understanding of these criteria, we conducted a systematic literature review of literature review guidelines. Following Webster and Watson (2002), we searched the eight top IS journals (AIS Senior Scholar’s Basket) and a special issue of the Communications of the AIS (Vol. 37, 2015) on literature reviews. The eight basket journals were selected due to their high methodical rigor and diversity (Hirschheim and Klein 2012), which makes them most likely to publish or reference the acknowledged review guidelines we were looking for. To identify such guidelines and review articles referencing them, we searched in titles, abstracts, and keywords using the broad query “literature AND review”. From the resulting in 266 articles, 57 articles were either literature review guidelines or review articles referencing at least one guideline in their method section, which were also included (backward search). This way we could identify a total of 25 literature review guidelines. After coding all requirements related to either the literature search procedure or its results, we aggregated them incrementally into meta-requirements. In the second step, we derived six design principles for SLSS by reflecting on the design knowledge acquired through the first DC and on the insights from our literature review of review guidelines. In the third step, we instantiated the developed design principles by refining the existing prototype web application. This allowed us to investigate potential implementations of the derived design principles and provide a first proof-of-concept (Nunamaker and Briggs 2011). Finally, we conducted a naturalistic ex-post evaluation of the prototype implementation through nine semi-structured expert interviews. The experts were researchers from the IS field with high expertise on the literature review process. The interview transcripts were analyzed using an iterative coding process to assess the utility, necessity, and sufficiency of the instantiated design principles. The interview results are briefly discussed at the end of this paper.

**SLSS Meta-Requirements**

Our review of literature review guidelines shows that, despite their different approaches, there is a common understanding in the IS community on criteria that constitute a good literature search. The following three meta-requirements synthesize this understanding.

**Comprehensiveness** (MR1) of a literature review describes the degree to which all relevant literature on the investigated topic is covered. The main goal of literature reviews is to find the existing body of knowledge. A fragmented literature sample can lead to a partial view on a topic (Fink 2014; Levy and Ellis 2006) and increases the chance that individual biased articles influence the integrity of an entire review (Cooper 1982; Fink 2014). A comprehensive overview of extant research is, thus, essential for finding and justifying research gaps (Fink 2014). The only way to achieve a comprehensive literature review is a comprehensive literature sample (Levy and Ellis 2006). However, comprehensiveness usually does not equal completeness. Compiling a complete literature sample is usually either inefficient or even impossible (Rowe 2014; Wolfswinkel et al. 2013). Review guidelines therefore suggest “a good or reasonable coverage” (Rowe 2014, p. 246).

**Precision** (MR2) describes the fraction of documents in a result set that is relevant to the reviewer. Manually identifying relevant documents from a large result set is one of the most time-consuming tasks during a review (Rowe 2014), especially, when applying an iterative search and review approach (Boell and Cecez-Kecmanovic 2014; Wolfswinkel et al. 2013). Because reviewers’ resources are usually limited (Okoli and Schabram 2010), guidelines recommend the definition of explicit inclusion and exclusion criteria that pre-filter search results. These criteria include selecting appropriate databases (database-centered strategies) or outlets (outlet-centered strategies) as well as parameters like keywords or authors (Boell and Cecez-Kecmanovic 2015; Fink 2014; Okoli and Schabram 2010). However, a more precise search is also more restrictive and more likely to exclude relevant research contributions (Rowe 2014). A good literature search is therefore both precise enough to exclude as many irrelevant articles as possible and comprehensive enough to include all vital contributions (Levy and Ellis 2006).

**Reproducibility** (MR3) defines the degree to which results of a literature review can be reproduced. A good literature search follows an approach that is reliable (i.e., results do not vary over time) and allows to communicate and justify each process step (Levy and Ellis 2006; vom Brocke et al. 2015). Hence, one major
precondition for reproducibility of literature searches is transparency of the search process (Wolfswinkel et al. 2013). A transparent search process enables reviewers to be explicit about how a literature sample was compiled, including queried data sources (e.g., databases or outlets) and exclusion and inclusion criteria (Fink 2014; Okoli and Schabram 2010; Wolfswinkel et al. 2013). A reproducible literature search is more reliable (Cooper 1982; Okoli and Schabram 2010) and contributes to the credibility of a review (Fink 2014; vom Brocke et al. 2015). Fellow researchers are enabled to assess the exhaustiveness of a literature sample and are encouraged to use and extend a review (Barnes 2005; vom Brocke et al. 2015). Furthermore, a reproducible and well documented search process allows to refine previous search steps and increases the chance of publication (Webster and Watson 2002; Wolfswinkel et al. 2013).

SLSS Design Principles

The following six design principles provide guidance on how to design real-world systems that facilitate comprehensive, precise, and reproducible literature searches.

(DP1 – Multi-sourcing) Provide the SLSS with the ability to retrieve and combine bibliographic data from multiple sources in order for reviewers to find all relevant research contributions, given that in the specific search context these contributions cannot be accessed through a single data source. To address MR1, a comprehensive search has to cover all sources that might contain literature relevant to the topic under review (Levy and Ellis 2006; Wolfswinkel et al. 2013) and is not limited to one set of journals or geographic region (Webster and Watson 2002). In the IS field, like most interdisciplinary fields, there is no central literature database. IS related research is published in over 800 outlets (Lamp 2017), which are spread over numerous databases (e.g., ProQuest and AISeL) (Boell and Cecez-Kecmanovic 2014; Levy and Ellis 2006). Even academic web search engines, like Microsoft Academic or Google Scholar, offer only limited coverage (Boeker et al. 2013; Bramer et al. 2013). Thus, to provide a reasonable coverage for a comprehensive literature search, SLSS must access and merge data from multiple sources, when either building their own catalogue or querying on behalf of reviewers.

(DP2 – Filtering) Provide the SLSS with precise filter mechanisms in order for reviewers to exclude irrelevant literature, given that the queried data structures allow for a precise subset selection. A search request is essentially a set of filter criteria defining inclusion and exclusion criteria, like specific outlets, authors, keywords, document availability, etc. In order to allow for precise search requests (MR2), SLSS must provide a wide range of filter criteria (restricted by the granularity and structure of the queried data sources), which should be easily adjustable during the search process to enable efficient iterative search request refinement. Depending on the implemented retrieval technology, result filtering can be performed ex-ante (e.g., filter settings in request forms) or ex-post (e.g., search within initial results or faceted search filters). For instance, in order to generate facets for ex-post result filtering, a SLSS must be able to retrieve and process the entire result set in one step (Nui 2014). SLSS with a successive retrieval approach (e.g., meta-search engines) can only implement ex-ante filter mechanisms.

(DP3 – Flexibility) Provide SLSS with a sufficient flexible search interface in order for reviewers to implement their individual search strategies, given that there is more than one possible strategy in the specific search context. Reviewers require the ability to formulate search requests that balance the trade-off between comprehensiveness (MR1) and precision (MR2). Since this trade-off is unique for each search instance (Boell and Cecez-Kecmanovic 2014; Rowe 2014), providing a sufficient level of flexibility to implement individual strategies and constraints (i.e., exclusion and inclusion criteria) appropriate for a review’s goals and limitations is vital for SLSS. Furthermore, a fit between the characteristics of an IT system, in this case SLSS’s functionality, and the researchers’ needs will not only lead to a higher task performance but also increases usage acceptance of the system (Goodhue and Thompson 1995).

(DP4 – Transformation) Provide the SLSS with the ability to translate reviewers’ search requests into data-source-specific queries without semantic changes in order for reviewers to receive precise and predictable results, given that the system accesses multiple data sources or provides custom interfaces with different rulesets. Reviewers usually must prepare individual search requests for each queried data source, due to lack of database standards (Boell and Cecez-Kecmanovic 2014). Most literature databases have their own request format (e.g., syntax, parameters and wildcards), catalog style (e.g., outlet names), and restrictions (e.g., number of terms or Boolean
expressions). Overlooking such peculiarities leads to unexpected results during a cross-database search (Boell and Cecez-Kecmanovic 2014), and eventually decreases its comprehensiveness, precision, and reproducibility. Hence, SLSS must transform reviewers’ requests to take peculiarities of queried data source into account without semantic changes, either for indexing or querying purposes.

(DP5 – Transparency) Provide the SLSS with a transparent search process to enable reviewers to comprehend and communicate each process step, given that the provided information contributes to reviewers’ understanding of the process. Detailed information on how the search results were produced (e.g., queried data sources and outlets, applied parameters) enables reviewers to understand the comprehensiveness of their search and, if necessary, to either extend the search to increase comprehensiveness or document gaps to increase reproducibility (Fink 2014;Boell and Cecez-Kecmanovic 2014; vom Brocke et al. 2015). For instance, the undocumented catalogue of academic web search engines makes it impossible to determine which sources were searched (Beckmann et al. 2012; Boeker et al. 2013; Bramer et al. 2013), whereas a transparent search system provides ample information on where and how the presented results were attained.

(DP6 – Reliability) Provide the SLSS with the ability to produce similar search results for identical search requests in order for reviewers to reproduce pervious search results, given a sufficient stable publication practice in the specific search context. Unpredictable search algorithms or search catalogue with high content fluctuation, like Google Scholar (Beckmann et al. 2012; Bramer et al. 2013), will lead to unique search results depending on when the search is performed or by whom (Boeker et al. 2013), no matter how thoroughly the search process is described. To provide reproducible search results (MR3), SLSS must provide not only a transparent search process but also a stable search environment (i.e., stable literature catalogues and objective search algorithms) to replicate the results when repeating the process.

Instantiation of SLSS Design Principles

This section gives a brief overview on the instantiation of the six SLSS design principles in form of LitSonar (http://litsonar.com), a prototype web application designed to support systematic literature searches. LitSonar’s software architecture consists of three main components: a Java EE web service, a web-based user interface (PHP client), and an outlet database. Figure 2 depicts the prototype’s architecture along with indicators for the implemented design principles. For an more extensive description of the prototype and its development process we refer to Sturm et al. (2015).

LitSonar provides unified access to multiple literature databases by utilizing the meta-search approach (DP1) and, thus, acting as an intermediary between reviewers and retrieval systems, as depicted in Ingwersen’s model (Figure 1). Reviewers’ search requests are dispatched to up to six curated databases containing IS-related literature (e.g., ProQuest and EBSCOhost). By utilizing curated data sources, LitSonar passes their catalogues’ stability on to the reviewer and, thereby, contributes to the reliability of the search.

Figure 2. Abstract Architecture of LitSonar and Implemented Design Principles
results (DP6). LitSonar’s user interface provides two novel features for entering search requests to increase precision (DP2), besides typical filters, like time-span or article types. First, a flexible keyword editor lets reviewers define complex nested query structures of any depth using graphical elements, instead of the usual “expert mode” (i.e., a single text field), as most databases provide for complex requests (Figure 3). Second, a data-source-selection-mask allows reviewers to either select multiple databases directly (database-centered) or compile a list of journals and conferences (outlet-centered). In the latter case, reviewers can choose from individual outlets and predefined lists of outlets based on journal and conference rankings. Using information from an internal outlet database, LitSonar automatically identifies appropriate databases, so that all selected outlets in the specified timeframe are covered. Thus, both the data-source-selection-mask and the keyword editor showcase our efforts to facilitate the definition of more precise search requests (DP2) and enable a wide variety of search approaches through a highly flexible interface design (DP3).

After receiving a reviewer’s request, LitSonar transforms it into database-specific search queries, including the translation of syntax and parameter values (e.g., outlet names) to match the respective format (DP4). During this process, the semantic of queries is altered only with the reviewer’s knowledge and consent to keep the search process transparent (DP5) and reliable (DP6). After dispatching the requests to the queried data source, returned results are presented in a homogenous, deduplicated list. Reviewers can browse through the list, download articles, compose individual result lists, and export article references. Additionally, LitSonar provides extensive reports on the coverage of literature databases and outlets to increase transparency of the search process (DP5). The database report shows which databases were searched and how many results per database were found. If a selected database could not be searched, an explicit warning is presented. In that case database-specific search query are provided, along with instructions on how to proceed manually (i.e., using databases’ native search interfaces). LitSonar also provides an outlet coverage report, if the reviewer restricted the search to certain outlets (Figure 3). This report gives detailed information about each selected outlet by listing the searched time periods and highlighting gaps in coverage. This information enables reviewers to assess and communicate the exhaustiveness of the conducted search and, if necessary, manually complement the results.

**Qualitative Evaluation and Next Steps**

The evaluation of LitSonar through expert interviews underline reviewers’ need for SLSS. The experts describe their standard search approach (i.e., consecutive keyword searches in multiple literature databases
and manual result merging) as complex, time-consuming, and error-prone. Furthermore, we find a fit between LitSonar and the task of systematically searching literature. The interviewed experts express strong intent to use the system. Using the system is expected to have a positive outcome on performance in form of a higher comprehensiveness and efficiency of the search process, which contributes to the quality of literature reviews. These findings indicate the technical feasibility and utility of our prototype implementation and, thus, also give evidence for the relevance of our six design principles. Another interesting finding from the evaluation of LitSonar is that the interview experts mentioned a lack of support from the prototype during the early stages of a systematic literature search (e.g., for identifying relevant search keywords). It was also mentioned that these activities are probably not fully automatable and therefore difficult to implement. This is in line with Levy and Ellis (2006), who describe identifying the right keywords as a creative process and a classic cold-start problem. Often several search and analysis cycles are necessary to refine naïve search terms into a complete set of relevant keywords (Levy and Ellis 2006). This finding raises the question of whether SLSS can support the entire systematic literature search process, or even the broader question of what are the limitations of SLSS. To answer these questions, further research is necessary.

In conclusion, we learn that the SLSS design principles and their instantiation are technically feasible and carry the potential to facilitate systematic literature searches. However, to rigorously examine the utility and relevance of the designed artifact and thereby make a valuable contribution to the design knowledge base, LitSonar must be studied directly in the application domain (Gregor and Hevner 2013; Hevner et al. 2004). The results from our qualitative evaluation demonstrate that LitSonar has reached a sufficient level of maturity, allowing us to subject the prototype to a large-scale field test. Building on the output of the previous two DC, the next steps of our research incorporate a third DC of artifact refinement and evaluation. To evaluate LitSonar’s impact on systematic literature searches of real users, the prototype will be rigorously examined in a quantitative evaluation. The evaluation will be conducted in the course of a large-scale field test at two German universities, allowing us to study LitSonar directly in the application domain. Students and researchers will have open access to the system. The data collection method will include both data logs and a voluntarily questionnaire. Data analysis will focus on the on the utility of the instantiation and, in particular, on how well the SLSS meta-requirements are met. For this we will adapt established IR measurements, like precision and recall (Kent et al. 1955), as well as develop new measurements based on pretest that capture the specifics of the systematic search task. Besides completing the third DC, the quantitative evaluation will contribute to the DSR knowledge base. Studying LitSonar in its environment also allows us to assess whether the SLSS design principles adequately address the prevailing challenges and improve the application domain as intended by DSR. Additionally, we will learn more about the information needs, problems/goals, and information behavior (i.e., cognitive model) of individual users and user groups. Following Ingwersen (1996), these insights will give us a better understanding of the user-system-interaction and guide future refinements of both SLSS design principles and their implementation.

The contributions of this short paper are twofold. We contribute to research by identifying an initial set of SLSS meta-requirements through a systematic literature review, deriving design principles for SLSS, and providing first evidence for their utility. As discussed in the research background section, there is already some scientific interest in the systematic literature search process and the critical evaluations of existing artifacts. However, the existing knowledge can be summarized as dispersed, vague requirements that provide hardly any directions on how an information system could address them. To our knowledge, we are the first to provide rigorously and systematically derived meta-requirements and corresponding design principles for information systems that facilitate rigorous, systematic literature searches. Our research thereby not only contributes to the existing literature review and information retrieval literature streams but also could serve as knowledge repository and starting point for future research on SLSS. Fellow researchers might use the presented design knowledge to explore its relevance in different contexts (i.e., research areas) or develop novel evaluation instruments to measure the suitability of systematic literature search solutions. We also contribute to practice by providing meta-requirements and a first blueprint (i.e., design principles) for SLSS that facilitate the systematic search process. Developers can use this knowledge to create innovative search systems or add systematic search features to existing solutions. Our research results could help to increase comprehensiveness, precision, and reproducibility of future systematic literature searches and, eventually, have a positive effect on the overall quality of literature reviews.
References


