

Editorial: Systematic Reviews in Information and Software Engineering

Barbara Kitchenham and Claes Wohlin

In 2005, the Editors of Information and Software Technology (IST) first introduced readers to the concept of systematic reviews [1]. IST published the first two systematic reviews in 2005 ([2], [3]). These studies appeared so promptly because they were authored by researchers involved in the initial proposal for evidence-based software engineering ([4], [5]).

We are already able to identify that these papers have proved of interest to our readers. For example, with 21 citations found by SCOPUS, Dybå et al.'s paper on the use of statistical power in software engineering papers is currently the most cited IST paper published in 2005.

After the initial two reviews, only two more were published in 2006 and 2007, but since 2008 we have seen an increasing number of reviews. This time lag is an indication of the time and effort required to undertake such studies. The purpose of this special collection is to bring to your attention some of the systematic reviews recently published by IST.

Systematic reviews (sometimes called systematic literature reviews in software engineering to ensure they are not confused with inspection research) are a form of *secondary study* that assesses the impact of individual empirical studies (referred to as *primary studies*). They aim to search for, and aggregate, all relevant information on a specific research topic, where within in the context of evidence-based software engineering, “information” implies empirical evidence. The basic methodology for systematic reviews was developed first for medical research. Kitchenham and Charters [6] have tried adapted the procedures to the context of software engineering and included guidelines from recent social science texts ([7], [8]). The guidelines adapted for software engineering can be found [here](#). We recommend looking at all three sources and reading some of the systematic reviews included in this collection before attempting to undertake a systematic review, since the difficulty of performing such studies should not be underestimated.

Systematic Review Collection

The systematic reviews included in this collection are shown in Table 1. They were chosen to provide examples of recent systematic reviews, and include a mapping study (MS) and a meta-analysis (MA) as well as more conventional systematic reviews (SR). To maximise reader interest, we have concentrated on studies concerned with software engineering topics as opposed to research methodology topics. The collection is intended to demonstrate the range of topics suitable for systematic reviews, and illustrate a variety of different aggregation methods.

Studies S1 and S6 will be of interest to the many people in academia and industry researching, using, or planning to use agile methods. In S1, Dybå and Dingsøyrr review holistic empirical studies of agile methods i.e. the results of applying an integrated set of agile methods. From a methodological point of view they have used a qualitative method to integrate the results of different primary studies and present their results in a narrative form. In contrast, in S6, Hannay and his co-workers look at a specific agile technique – pair-programming. Here they have gone beyond simple tabulation of results and have used statistical meta-analysis to aggregate their results. Both studies suggest we know a lot less about agile techniques than we might think. Dybå and Dingsøyrr highlight the general lack of trustworthy empirical studies, particularly related to management methods – they only found one study investigating SCRUM. Hannay et al. found limited evidence that agile methods exhibit greater quality and are faster than conventional methods but are less productive. However, all these effects are small and detailed meta-analysis casts some doubts on the reliability of the effects. However, Hannay et al. offer some guidance regarding the conditions under which agile methods may be effective.

Two studies demonstrate the way in which researchers can use systematic review results to support their research goals. In S3, Beecham and her co-workers investigate empirical studies of motivation in Software Engineering. This topic should be of interest to software managers and soft factor researchers. In a subsequent study, this systematic review was used as the basis for constructing a motivation model [9]. Walia and Carver present a similar use of the results of a systematic review. In S8, they present their systematic review of software requirements errors and organise the results into a fault taxonomy they propose as a starting point for quality improvement. This study also includes a review of studies published in the field of human psychology and psychology emphasizing that software engineering researchers may find it useful to consider research in other related disciplines.

Afzal et al. (S7) illustrate the progression from mapping study to more detailed systematic review. Their discussion of search-based testing for non-functional properties was preceded by a mapping study looking at general research trends in the topic area [10]. In S2, Björnson and Dingsøyrr present an example of how a systematic review can provide an excellent introduction to a topic area. Their study has some aspects of a mapping study since they classify the literature, but they also provide a narrative discussion of the results concluding with advice for researchers and practitioners.

The final two reviews in this collection address issues that should be of particular interest to software managers. In S5, Jadhav and Sonar present a mapping study classifying the literature relating to software package selection. This study categorizes the literature so it is easy to see if there is any research related to a specific package selection issue. In S4, Staples and Niazi investigate the motives that organizations have for adopting CMM-based software process important. This study was based on aggregating qualitative data i.e. textual comment embedded in primary studies, and discusses some of the issues involved in using this approach.

If the papers in this collection have proved of interest, you can obtain information about other systematic reviews published between 2004 and June 2007 by consulting [11] which is a mapping study of systemic literature reviews.

Table 1 Recent systematic reviews published by IST

ID	Authors	Title	Year	Type	Who should read it
S1	Dybå and Dingsøy	Empirical studies of agile software development: A systematic review	2008	SR	Researchers, consultants managers and software engineers interest in Agile methods
S2	Björnson and Dingsøy	Knowledge management in software engineering: A systematic review of studies concepts, finding and research methods used	2008	SR/MS	Knowledge management researchers and consultants
S3	Beecham et al.	Motivation in Software Engineering: A systematic Literature review	2008	SR	Soft factors researchers SW Managers
S4	Staples & Niazi	Systematic review of organisational motivations for adopting CMM	2008	SR	CMM researchers, and consultants and SW managers
S5	Jadhav and Sonar	Evaluating and selecting software packages: A review	2009	MS	Researchers or SE managers involved in package selection
S6	Hannay et al.	The effectiveness of pair programming: A meta-analysis	2009	MA	Researchers, consultants managers and software engineers interest in Agile methods
S7	Afzal et al.	A systematic review of search-based testing for non-functional system properties	2009	SR	Researcher in the field of testing and non-functional requirements
S8	Walia and Carver	A systematic literature review to identify and classify software requirements errors	2009	SR	Researchers and practitioners (particularly quality managers) interested in error classification.

References

- [1] Dyer, M., Shepperd, M. and Wohlin, C. Editorial: Systematic Reviews in Evidence-based Software Technology and Software Engineering, Information and Software Technology, 47, 2005, p 1.
- [2] Grimstad, S., Jørgensen, M. and Molokken-Ostfold, K. Software effort estimation terminology: The tower of Babel, Information and Software Technology, 48 (4), 2006, pp 302-310.
- [3] Dybå, T., Kampenes, V.B. and Sjøberg, D.I.K. A systematic review of statistical power in software engineering experiments, Information and Software Technology, 48(8), 2006, pp 745-755.
- [4] Kitchenham, B.A., Dybå, T. and Jørgensen, M. Evidence-based Software Engineering. Proceedings of the 26th International Conference on Software

- Engineering, (ICSE '04), IEEE Computer Society, Washington DC, USA, 2004, pp 273 – 281.
- [5] Dybå, T., Kitchenham, B.A. and Jørgensen, M. Evidence-based Software Engineering for Practitioners, *IEEE Software*, 22 (1), 2005, pp 58-65.
 - [6] Kitchenham, B.A. and Charters, S. Guidelines for performing Systematic Literature Reviews in Software Engineering Technical Report EBSE-2007-01, 2007.
 - [7] Fink, A. *Conducting Research Literature Reviews. From the Internet to Paper*, Sage Publication, Inc., 2005.
 - [8] Petticrew, M. and Roberts, H. *Systematic Reviews in the Social Sciences: A Practical Guide*, Blackwell Publishing, 2005.
 - [9] Sharp, H., Baddoo, N., Beecham, S., Hall, T. and Robinson, H. Models of Motivation in Software Engineering, *Information and Software Technology*, 51(1), 2009, pp 219-233.
 - [10] Afzal, W., Torkar, R. and Feldt, R. A Systematic Mapping Study on Non-Functional Search-Based Software testing, in 20th International Conference on Software Engineering and Knowledge Engineering (SEKE), 2008.
 - [11] Kitchenham, B., Brereton, O.P., Budgen, D., Turner, M., Bailey, J. and Linkman, S. Systematic Literature Reviews in Software Engineering – a Systematic Literature Review, *Information and Software Technology*, 51, 2009, pp 7-15.