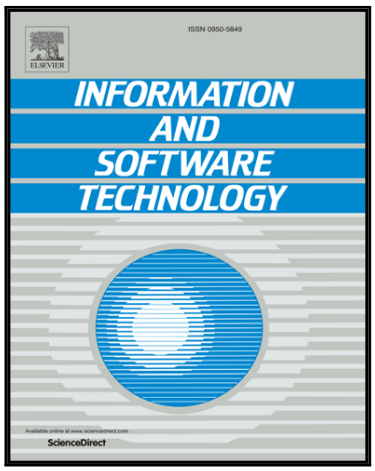

A Systematic Literature Review of Software Requirements Reuse Approaches

Mohsin Irshad, Kai Petersen, Simon Poulding

PII: S0950-5849(16)30361-5
DOI: 10.1016/j.infsof.2017.09.009
Reference: INFSOF 5882

To appear in: *Information and Software Technology*

Received date: 21 November 2016
Revised date: 11 September 2017
Accepted date: 22 September 2017

Please cite this article as: Mohsin Irshad, Kai Petersen, Simon Poulding, A Systematic Literature Review of Software Requirements Reuse Approaches, *Information and Software Technology* (2017), doi: 10.1016/j.infsof.2017.09.009

A Systematic Literature Review of Software Requirements Reuse Approaches

Mohsin Irshad, *Blekinge Institute of Technology, Karlskrona, Sweden,*
Kai Petersen, *Blekinge Institute of Technology, Karlskrona, Sweden,*
Simon Poulding, *Blekinge Institute of Technology, Karlskrona, Sweden*

Abstract

Context: Early software reuse is considered as the most beneficial form of software reuse. Hence, previous research has focused on supporting the reuse of software requirements.
Objective: This study aims to identify and investigate the current state of the art with respect to (a) what requirement reuse approaches have been proposed, (b) the methods used to evaluate the approaches, (c) the characteristics of the approaches, and (d) the quality of empirical studies on requirements reuse with respect to rigor and relevance.
Method: We conducted a systematic review and a combination of snowball sampling and database search have been used to identify the studies. The rigor and relevance scoring rubric has been used to assess the quality of the empirical studies. Multiple researchers have been involved in each step to increase the reliability of the study.
Results: Sixty-nine studies were identified that describe requirements reuse approaches. The majority of the approaches used structuring and matching of requirements as a method to support requirements reuse and text-based artefacts were commonly used as an input to these approaches. Further evaluation of the studies revealed that the majority of the approaches are not validated in the industry. The subset of empirical studies (22 in total) was analyzed for rigor and relevance and two studies achieved the maximum score for rigor and relevance based on the rubric. It was found that mostly text-based requirements reuse approaches were validated in the industry.
Conclusion: From the review, it was found that a number of approaches already exist in literature, but many approaches are not validated in industry. The evaluation of rigor and relevance of empirical studies show that these do not contain details of context, validity threats, and the industrial settings, thus highlighting the need for the industrial evaluation of the approaches.

Index Terms

Software Requirements, Requirements Reuse, Rigor, Relevance, Artefact Reuse, Reusability.

1 Introduction

Software reuse describes the practice of reusing existing software artefacts for developing a new product or maintaining an old product [70]. The field has been the subject of research for several years and different aspects of software reuse such as costs, artefacts, the level of reuse, stakeholders, and reuse processes have been investigated [71] [72] [73] [74] [101]. Software reuse can take place at any time during a project's lifecycle starting from the requirements analysis phase to the maintenance phase and researchers have pointed out that the reuse in early phases of the project leads to more benefits [73] [75]. The reuse of software requirements is one of the approaches to enable early reuse. Software requirements contain the information on the needs of a user or the details of a contract that is formally imposed. The reuse of software requirements has been the focus of investigations in the past and the reuse of requirements can help in two ways: (i) reducing the time for the analysis of requirements; (ii) identifying reusable code and test artefacts having similar requirements, thus resulting in an early reuse i.e., early in development cycle [76] [77].

Various studies on reusing software requirements are present in the literature [4] [42] [55]. Approaches to requirements reuse take different forms of requirements into consideration, such as text-based requirements, use-cases, and formal specifications. Different strategies such as variable/ configurable part of requirements or analogy based matching have been proposed to achieve requirements reuse [19] [78]. Similarly, formal methods based requirements reuse approaches have been proposed [106]. These approaches are applied in various different contexts and use different reusable artifacts to increase the reusability. This study is an attempt to identify and analyze the requirements reuse approaches present in the literature.

Empirical investigations in software engineering allow practitioners and researchers to evaluate a solution before it is applied in real world settings. This can help in assessing the benefits and limitations of a solution. Each solution (a model, guideline or method) should be described with details that others can use to reproduce the solution. Kitchenham et al. point to the need of improving the reporting of empirical research in software engineering and concluded that guidelines

- *M. Irshad, K. Petersen and S. Poulding are employed at Software Engineering Research Lab, Blekinge Institute of Technology, Karlskrona, Sweden,*

  E-mail: mohsin.irshad@bth.se, kai.petersen@bth.se, simon.poulding@bth.se