Journal of Systems and Software
Volume 95, September 2014, Pages 19-31

# Empirical research methods for technology validation: Scaling up to practice

Roel Wieringa



https://doi.org/10.1016/j.jss.2013.11.1097



## Highlights

- When scaling up RE technology to practice, researchers want to empirical validate new technology.
- We discuss research methods and techniques that can be used for empirical validation of new RE technology.
- We discuss expert opinion, single-case experiments, technical action research and statistical experiments.
- We also discuss the statistical, analogic, and abductive inference techniques used in these methods.
- We illustrate these research methods and techniques with examples from both empirical SE and empirical RE research.

## Abstract

Before technology is transferred to the market, it must be validated empirically by simulating future practical use of the technology. Technology prototypes are first investigated in simplified contexts, and these simulations are scaled up to conditions of practice step by step as more becomes known about the technology. This paper discusses empirical research methods for scaling up new requirements engineering (RE) technology.

When scaling up to practice, researchers want to generalize from validation studies to future practice. An analysis of scaling up technology in drug research reveals two ways to generalize, namely inductive generalization using statistical inference from samples, and analogic generalization using similarity between cases. Both are supported by abductive inference using mechanistic explanations of phenomena observed in the simulations. Illustrations of these inferences both in drug research and empirical RE research are given. Next, four kinds of methods for empirical RE technology validation are given, namely expert opinion, single-case mechanism experiments, technical action research and statistical difference-making experiments. A series of examples from empirical RE will illustrate the use of these methods, and the role of inductive generalization, analogic generalization, and abductive inference in them. Finally, the four kinds of empirical validation methods are compared with lists of validation methods known from empirical software engineering. The lists are combined to give an overview of some of the methods, instruments and data analysis techniques that may be used in empirical RE.

## Introduction

Empirical assessment of technology comes in two flavors, which in this paper will be called technology validation and technology evaluation, respectively. *Technology validation* is defined here as the assessment of a simulation of the technology in a simulation of its intended context of use, in order to predict what would happen if the technology were actually used by stakeholders in this intended context. We take the term "simulation" in a very wide sense as the representation of the functioning of one system or process by means of the functioning of another.[2] For example, a new requirements prioritization technique may be tested by experimenting with it in a classroom. This is a validation if the classroom experiment represents some aspects of what would happen if the technique was used in practice.

Validation always involves *scaling up to practice*, which means that successive tests take place under increasingly realistic conditions. For example, the inventor of a requirements prioritization technique may use this technique in a real-world project. This validation would resemble real-world use of the technique more than a classroom experiment, except that it is still the inventer herself who uses the technique.

A technology has been *transferred to practice* if it has been packaged, marketed, distributed, sold or otherwise made available to users, and is now being used independently from the context in which it was invented or tested. After transfer to practice other people than its inventors are using it, and they are using it to achieve their own goals, without help or other kind of intervention from its inventors, and after investment of their own time and/or money to learn to use the technology.

Technology validation is to be contrasted with *technology evaluation*, defined here as the empirical assessment of a technology as and when used in practice. For example, an RE researcher may study how a prioritization technique is used in real-world projects by means of observational case studies. Where a validation study aims to make predictions, based on simulations, about how a technology would perform if transferred to practice, an evaluation study assesses what has happened in the actual use of the technology after it has been transferred in practice. This follows terminology commonly used in the social sciences, where an evaluation study is an empirical assessment of some social intervention that has been performed, such as a recently implemented teaching method in schools, to investigate its impact in practice(Babbie, 2007).

Technology validation is a process of scaling up to practice in all engineering sciences. For example, the inventors of the jet engine validated their designs by building increasingly realistic prototypes and testing them in increasingly realistic environments(Constant, 1980). In this paper I will summarize and analyze the ways in which we can scale requirements engineering (RE) technology up to practice.

The RE technology being validated could be techniques, methods, notations, algorithms, etc. used for various requirements engineering tasks such as requirements elicitation, goal analysis, requirements specification, requirements prioritization, traceability management, requirements maintenance, etc. *Requirements* in this paper are defined as desired properties of a system. *Goals*, by contrast, are states of the world desired by stakeholders, and for which the stakeholders have committed a budget (time or money) to achieve them. All requirements are goals because they are desired by stakeholders, and stakeholders have committed a budget to achieve them. But not all stakeholder goals are system requirements. Stakeholders have many goals not stated in terms of desired system properties at all.

Davis and Hickey (2004) proposed using the methodology of New Drug Development for scaling up RE technology. I will pursue this idea further in Section2 and focus in particular on the inferences used in New Drug Development when generalizing from the object of validation research to instances of real-world use of the technology, and show that these inferences can be used in RE research too. In Section3, I present four methods for empirical technology validation, and show how the generalization methods identified in Section 2 can be used in them. This is illustrated by a series of examples from empirical requirements engineering. Finally, in Section 4, I review the empirical software validation methods identified by Zelkowitz and Wallace, 1997, Zelkowitz and Wallace, 1998 and by Glass et al. (2001) and show how they fit into the framework presented in this paper, and add a list of examples of techniques for measurement and data analysis used in empirical software engineering. Section 5 ends the paper with a brief summary and outlook.



## Section snippets

### Scaling up in drug research

Davis and Hickey (2004) were the first to apply the New Drug Development and Review Process of the U.S. Food and Drug Administration to RE technology transfer. I summarize the process in Fig. 1. The following description is based on information provided by the FDA[3] and the explanations given by Davis and Hickey (2004), Cowan (2002) and Molzon and Pharm (2005). My analysis goes beyond that of Davis and …

### Methods for validation research

We will discuss the empirical validation methods using the structure of Fig. 4. We have used this structure earlier to make a checklist for empirical research reports(Wieringa et al., 2012). The researcher uses an object of study (OoS) to represent elements of the population, where in our case the population elements have the structure [artifact×context]. Therefore, the OoS has this structure too, consisting of a model of the artifact and a model of the context. The OoS is a *model* of an …

### Related work

The reasoning schema "[Artifact×Context] → Effect by Mechanisms" has been proposed in slightly different forms in social science(Pawson and Tilley, 1997) and in management science(Van Aken, 2004). It has some similarity with the satisfaction argument as proposed by Jackson in software engineering(Jackson, 2000). Wieringa (2003) calls it the systems engineering argument, because it shows how a component must interact with other components to produce desired behavior of a composite system. It …

## Summary and conclusion

Empirical validation of technology before it is transferred to practice requires investigating the effects of the interaction of the artifact with its context, and explaining these effects by means of the underlying mechanisms that produces these effects. Scaling up to practice thus produces a design theory of the form "[Artifact×Context] produces Effects by Mechanisms".

Producing support for such a theory involves two kinds of inferences, along the vertical and horizontal dimensions of the …

## Acknowledgements

This paper benefitted from comments by Vincenzo Gervasi and Walter Tichy. I would like to thank the anonymous reviewers of this paper for their constructive critique. …

**Roel Wieringa** is Chair of Information Systems at the University of Twente, The Netherlands. His research interests include requirements engineering, risk assessment, and design research methodology. He has written two books, on Requirements Engineering and on the Design of Reactive Systems. His next book, Design Science Methodology for Information Systems and Software Engineering will appear in 2014 with Springer. …

…

…

Recommended articles

## References (63)

W. Bechtel *et al.*
Explanation: a mechanistic alternative
Studies in the History and Philosophy of Biological and Biomedical Sciences (2005)

V. Kampenes *et al.*
A systematic review of quasi-experiments in software engineering
Information and Software Technology (2009)

A. Nehlig *et al.*

Caffeine and the central nervous system: mechanisms of action, biochemical, metabolic and psychostimulant effects

Brain Research Reviews (1992)

M. Zelkowitz *et al.*

Experimental validation in software engineering

Information and Software Technology (1997)

A. Al-Emran *et al.*

Decision support for product release planning based on robustness analysis

L. Apostel

Towards a formal study of models in the non-formal sciences

E. Babbie

The Practice of Social Research

(2007)

W. Bechtel *et al.*

Discovering Complexity: Decomposition and Localization as Strategies in Scientific Research

(2010)

M. Bunge

How does it work? The search for explanatory mechanisms

Philosophy of the Social Sciences (2004)

E. Constant

The Origins of the Turbojet Revolution

(1980)

View more references

## Cited by (59)

A systematic literature review on agile requirements engineering practices and challenges

2015, Computers in Human Behavior

Show abstract

Six strategies for generalizing software engineering theories

2015, Science of Computer Programming

Show abstract

A Systematic Review on Validated Precision Livestock Farming Technologies for Pig Production and Its Potential to Assess Animal Welfare ↗

2021, Frontiers in Veterinary Science

Defining Blockchain Governance: A Framework for Analysis and Comparison ↗

2021, Information Systems Management

Software startups-A research agenda ↗

2016, E Informatica Software Engineering Journal

Design science methodology: For information systems and software engineering ↗

2014, Design Science Methodology for Information Systems and Software Engineering

View all citing articles on Scopus ↗

**Roel Wieringa** is Chair of Information Systems at the University of Twente, The Netherlands. His research interests include requirements engineering, risk assessment, and design research methodology. He has written two books, on Requirements Engineering and on the Design of Reactive Systems. His next book, Design Science Methodology for Information Systems and Software Engineering will appear in 2014 with Springer.

1 http://www.cs.utwente.nl/~roelw ↗.

View full text