

Accepted Manuscript

Six strategies for generalizing software engineering theories

Roel Wieringa, Maya Daneva

PII: S0167-6423(14)00545-0
DOI: [10.1016/j.scico.2014.11.013](https://doi.org/10.1016/j.scico.2014.11.013)
Reference: SCICO 1851

To appear in: *Science of Computer Programming*

Received date: 11 October 2013
Revised date: 5 March 2014
Accepted date: 25 June 2014

Please cite this article in press as: R. Wieringa, M. Daneva, Six strategies for generalizing software engineering theories, *Sci. Comput. Program.* (2015), <http://dx.doi.org/10.1016/j.scico.2014.11.013>

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.



Six Strategies for Generalizing Software Engineering Theories

Roel Wieringa, Maya Daneva
University of Twente, The Netherlands

Abstract

General theories of software engineering must balance between providing full understanding of a single case and providing partial understanding of many cases. In this paper we argue that for theories to be useful in practice, they should give sufficient understanding of a sufficiently large class of cases, without having to be universal or complete. We provide six strategies for developing such theories of the middle range.

In lab-to-lab strategies, theories of laboratory phenomena are developed and generalized to other laboratory phenomena. This is a characteristic strategy for basic science. In lab-to-field strategies, theories are developed of artifacts that first operate under idealized laboratory conditions, which are then scaled up until they can operate under uncontrolled field conditions. This is the characteristic strategy for the engineering sciences.

In case-based strategies, we generalize about components of real-world cases, that are supposed to exhibit less variation than the cases as a whole. In sample-based strategies, we generalize about the aggregate behavior of samples of cases, which can exhibit patterns not visible at the case level. We discuss three examples of sample-based strategies.

Throughout the paper, we use examples of theories and generalization strategies from software engineering to illustrate our analysis. The paper concludes with a discussion of related work and implications for empirical software engineering research.

Keywords:

Generalization, external validity, scaling up, causal theory, architectural

Email addresses: r.j.wieringa@utwente.nl (Roel Wieringa),
m.daneva@utwente.nl (Maya Daneva)

Download English Version:

<https://daneshyari.com/en/article/6875349>

Download Persian Version:

<https://daneshyari.com/article/6875349>

[Daneshyari.com](https://daneshyari.com)