

Bridging the Research-Industry Gap: The Case for Domain Modeling

Pamela Zave

AT&T Advanced Technology—Research
Bedminster, New Jersey 07921, USA
Email: pamela@research.att.com

Abstract—The abstract goes here.

I. INTRODUCTION

The organizers of the Workshop on Software Engineering Research and Industrial Practice, in the workshop aims, have provided an eloquent description of the gap between research and practice. The gap is wide and affects all researchers, regardless of whether they work in universities, government labs, or even industrial research labs.

Transfer of research to industrial practice is always challenging, particularly because researchers are asking practitioners to change the way they do their work, and practitioners usually have the power to refuse change without apparent consequences. From the workshop aims, here are some of the reasons practitioners refuse:

- Practitioners have a view that case studies in research do not represent the complexities of real projects, and doubt that results are realistic.
- Practitioners believe that researchers are working on either dated or futuristic challenges.
- Practitioners want quick and powerful solutions that will generate profits immediately, while research takes a long time and yields limited, cautious results.

In this talk I will diagnose one aspect of the problem, and propose a solution to it. My goal is to make researchers more productive, as well as to improve their prospects for technology transfer.

II. A PROBLEM DIAGNOSIS

In today's world, software systems—usually large and complex—are integrated into every technological domain. In many of these domains, there are few artifacts filling the conceptual gap between the needs of the domain and the software itself. The word “artifacts” refers to requirements, specifications, domain descriptions, design documents, models, test suites, manuals, *etc.* When such artifacts do exist, they are often of such poor quality that they are unread, obsolete, and forgotten.

This situation is a detriment to industry. It is also an obstacle to doing research that transfers to industrial practice, because it prevents the kind of communication researchers need to understand real problems. It also prevents research that could make big contributions, as opposed to cautious incremental ones. In software engineering, making a big contribution

requires making a big generalization, which is rarely possible without some viewpoint more abstract than code.

III. DOMAIN MODELS

The missing information all comes under the general category of *domain models*. Domain models cover any part of the territory of environment and operating assumptions, requirements, software specifications, design, software platforms and frameworks, and use. As important as their subject matter, domain models should be *domain-specific*, *functional*, and *re-usable*.

??? why each, what does it mean ???

include formal in functional; domain models need to be formal, not just slides; domain models need to be functional, not just block diagrams (here functional is not opposed to performance, but to component assemblies);

??? there has been no lack of calling for these things, so why do we not have them? because they are really difficult to get; because they take a long time to get right; because practitioners cannot do it ???

IV. A SOLUTION

???

researchers should do domain modeling, no one else can or will

if there is killer research, this is where it will come from
???

V. OVERCOMING THE RESEARCH OBSTACLES

belief that these cannot be evaluated: ridiculous if you have ever tried

VI. EXAMPLE: THE SEMICONDUCTOR INDUSTRY

this is a great success story, with a great history; semiconductors and design automation grew up together; virtuous cycle started with an early standardized DSL; unfortunately it is too late for most domains to do this

VII. EXAMPLE: THE COMPUTING INDUSTRY

definition: artifacts are the principal interests of CS (compilers, OS, hardware architectures, databases)

VIII. EXAMPLE: THE TELECOMMUNICATIONS INDUSTRY

this is an industry with a great history that has been lost

IX. EXAMPLE: THE NETWORKING INDUSTRY
SDN and what is happening

the geomorphic view versus ambients and the pi calculus