

# Escaped from the Lab: Crossing the Gap from Invention to Practice

Dennis Mancl  
Alcatel-Lucent

Steven Fraser  
Cisco Systems

William Opdyke  
Motorola

## ABSTRACT

This workshop will address many practical questions that arise in software product organizations large and small, focusing on the transition from an idea to a product. How do you go from a good research idea to a real-world product? Why is it so hard to address performance, reliability, and security issues when going from an initial prototype implementation into product-quality code? What have we learned from our successes and failures in converting research ideas into quality software products? Workshop participants will explore the challenges, compiling a set of the most important techniques to utilize when turning inventions into practice.

## Categories & Subject Descriptors:

D.2.9 Management

K.4.3 Organizational Impacts

K.6.3 Software Management

**General Terms:** Management

**Keywords:** Complexity, Reuse

## 1. Escaped from the Lab

An important dynamic in large organizations: one part of the organization over-commits (promises the earth, moon, and stars), and another part of the organization is forced to deliver. The extravagant promise of the Powerpoint presentation is converted into the trail of tears of the Gantt chart.

The grandiose project was originally supposed to be feasible. There were some small technology trials that proved out the basic ideas for low-volume transaction rates and simplified user interfaces. The product was supposed to be delivered in record time because of high rates of software reuse. So what went wrong?

One thing that compounds this effect: only a small fraction of the people working on software are actually in firms that consider themselves to be in the software business. This may lead a team to make inadequate choices of practices and approaches for the process of turning new ideas into products.

This workshop will explore the intersection of object oriented technology, high reliability and performance requirements, large organizations, and conflicts in the software development process. Some questions we will

consider include:

- Prototyping: When can a prototype be “scaled up” to production? When is it just an experiment to be thrown away?
- Reusability: How to make an honest assessment of the potential reusability of a module or subsystem?
- Reliability and Availability: How do the reliability and availability requirements get captured and how do they become action items in the product plan?
- Frameworks: When is a framework part of the solution, and when does it become part of the problem?
- Contracting and Outsourcing: Who assumes the risk when there are problems with development schedule, gaps in the feature set, stability issues, and shortcomings in the product’s usability?
- Decision making: Is it possible to define the key issues early in the project lifecycle? What kinds of development processes and management processes are most useful for highlighting the gaps?
- Innovation and innovation models: Where do the new ideas come from, and how to we make sure that an invention turns into a money-making innovation?

## 2. Some good practices

A few good practices are discussed in existing literature:

- In the development schedule, include extra time for “discovery costs”. Managers need to plan for exploration and learning time when moving from a small prototype to a large product development team.
- Prototype applications should not be a fully working application: whenever possible, leave out key pieces of functionality so that there is less temptation to ship the prototype as a product.
- Use an agile development process centered on user scenarios when transforming a simple application into a production-grade system.

For more information on the subject of this workshop (including some of the workshop materials and the workshop’s final report), see the workshop website: <http://mysite.verizon.net/~dennis.mancl/oopsla08>.

Copyright is held by the author/owner(s).

OOPSLA '08 October 19–23, 2008, Nashville, Tennessee, USA.

ACM 978-1-60558-220-7/08/10.