Software Architecture: Past, present & future

Philippe Kruchten
Helsinki, August 22\textsuperscript{nd} 2012
Philippe Kruchten, Ph.D., P.Eng., CSDP

Professor of Software Engineering
NSERC Chair in Design Engineering
Department of Electrical and Computer Engineering
University of British Columbia
Vancouver, BC Canada
pbk@ece.ubc.ca

Founder and president
Kruchten Engineering Services Ltd
Vancouver, BC Canada
philippe@kruchten.com
Reflections: the Good

• Stabilization
  – Concepts, representations, roles, processes
  – Standards
  – Books have 2\textsuperscript{nd} and 3\textsuperscript{rd} edition
  – Certifications

• Education
  – Academia: Taylor, Bass et al.
  – Industry: Gorton, or Wood\&Rozanski
Reflections: the Bad

• Proliferation of conferences...?
• Widening industry-academia gap
  – Academic game
    • Tenure, publish/perish,
    • Granularity
    • Funding
  – Access to information
  – Agile methods
  – IP
• Fragmentation of research
  – Integration
  – Scaling
  – CS vs. SW Eng.
Reflections: the Bad

- Proliferation of conferences...?
- Widening industry-academia gap
  - Academic game
    - Tenure, publish/perish,
    - Granularity
    - Funding
  - Access to information
  - Agile methods
  - IP
- Fragmentation of research
  - Integration
  - Scaling
  - CS vs. SW Eng.
Future
• The real issues? the “wetware”
  – Culture (values, beliefs, attitudes)
    • Collaboration and coordination
    • Trust
  – Process
  – (not YAMW Yet another magic wand)
  – Reasoning process
• Qualitative research
  – Hard, different, not in a lab, longer
• Larger, coordinated research efforts
  – Integrating many magic wand
Two kinds of SW architect

**Architectus Reloadus**
- External Focus
- Requirement facing
- New technologies
- Business
- Partners
- Ecosystem
- *Arch. specification*

**Architectus Aryzus**
- Internal focus
- Code-facing
- Structure
- Development process
- Communication
- Implementation
- *Arch. delivery*

Fowler 2003 Who needs an architect
Whom are we focusing on?

**Architectus Reloadus**
- External Focus
- Requirement facing
- New technologies
- Business
- Partners
- Ecosystem
- *Arch. specification*

**Architectus Aryzus**
- Internal focus
- Code-facing
- Structure
- Development process
- Communication
- Implementation
- *Arch. delivery*

Economic (Value vs. cost)
Culture (values, beliefs, attitudes...)
Ways of working (Process)
Whom are we focusing on?

Architectus Reloadus
- External Focus
- Requirement facing
- New technologies
- Business
- Partners
- Ecosystem

Architectus Azyrus
- Internal focus
- Code-facing
- Structure
- Development process
- Communication
- Implementation

Economic (Value vs. cost)
Culture (values, beliefs, attitudes...)
Ways of working (Process)
Foundations for the Study of Software Architecture

Dewayne E. Perry
AT&T Bell Laboratories
600 Mountain Avenue
Murray Hill, New Jersey 07974
dep@research.att.com

Alexander L. Wolf
Department of Computer Science
University of Colorado
Boulder, Colorado 80309
alw@cs.colorado.edu


ABSTRACT

The purpose of this paper is to build the foundation for software architecture. We first develop an intuition for software architecture by appealing to several well-established architectural disciplines. On the basis of this intuition, we present a model of software architecture that consists of three components: elements, form, more toward integrating designs and the design process into the broader context of the software process and its management. One result of this integration was that many of the notations and techniques developed for software design have been absorbed by implementation languages. Consider, for example, the concept of supporting “programming-in-the-large”. This integration has tended to blur, if not confuse, the distinction between design and implementation.
Perry & Wolf 1992

• “accept with minor revisions”
• All the ingredients are present, except maybe:
  – the process (architecting)
  – the role (architect)
  – Views? What views?
• It tells a story, not a complete story
  – Examples not prescriptions
• We can retell their story in our own ways, in our own context. Which we did. Very slowly.