Models @ Run.time to Support Dynamic Adaptation

B. Morin, O. Barais, J-M. Jezequel, F. Fleurey, A. Solberg

(presented by Michalis Famelis)

October 17, 2012
“We are all Keynesians now.”
“We are all Keynesians now.”

Sometimes, if-then-else is not enough. Sometimes, you have to change the structure itself.
Introduction

Dynamically Adaptive Systems

- Systems capable of changing with their environment.
- Self-optimize for QoS, for offered services, etc.
- Embedded systems, systems of systems...
- *Dynamic Software Product Lines*
Example: DCRM System

Available features:

A DSPL state machine (fragment):
The dynamic structure of the system as a state machine. Monitor the environment, adapt to change.

The problems:

- State space explosion!
  - Not just the number of configurations.
  - But also the *transitions* between them (quadratic).
- Unplug to evolve.
  - Need to hard-code all those reconfiguration transitions.
Solution:

Models@Runtime for Dynamic Adaptation

Models to:

- define the features as Aspects
- express the environment/context
- express the transition logic

At Runtime the system:

- analyzes the context, and if necessary...
- ...weaves a configuration using AOM
- Also: validates the configuration

Dynamic Adaptation:

- Generate reconfiguration scripts
- Only adapt if config is valid (no need for rollbacks)
Architectural Overview
**Models**

- **DSPL**: the feature model of the system
- **Context**: set if environment variables
- **Reasoning**: selection of features based on context
  Can be:
  - Event-condition-action rules
  - Goal model
- **Architecture**: components and bindings (e.g. UML)

Note: the approach does not prescribe specific metamodels. But once chosen, they “strongly type” the system.
Architectural Overview
Components

- **Event Processor**: observe runtime, update context

- **Goal-based reasoner**:
  - Configured with DSPL and Reasoning
  - When Context is updated, derive new, adapted DSPL with necessary features

- **Aspect Weaver**: Weave new Architecture from the DSPL

- **Configuration Checker**: Verify the new Architecture (e.g. w.r.t. invariants or user constraints)

- **Configuration Manager**: Apply new Architecture to runtime system.
Features as Aspects

+Composition protocol: how to weave Advice in Pointcut
Models @ Run.time to Support Dynamic Adaptation

Introduction
Problem
Solution
Case Study
Conclusion

Adaptation

(a) s:offline DiVA studio → r:goal-based reasoning → m:configuration manager
   -> load DSPL (DSPL model) → load reasoning (reasoning model) → load configuration (architecture model)

(b) cep: complex event processing
   [bandwidth = 43] → [bandwidth = 48] → [bandwidth = 46] → [bandwidth = 51]
   c:context listener → bwidth CE:context element
   notify (value) → set value (bandwidth.HIGH) → notify ()

(c) r:goal-based reasoning
    notify → reason
    load DSPL (derived DSPL model)

opt
   build product
   check (architecture model)

alt
   load configuration (architecture model)
   is invalid (derived DSPL model)
Notes and Details

- The Event Processor decouples the DAS from the environment.
  - The Context “reflects” the environment but is not “causally connected” to it.
    
    *(so …it is an abstraction?)*
  - The Context serves for the DAS to reason about the environment.

- The system is updated online.
  - Adaptation only happens once a valid Architecture has been woven.
  - If a non-valid is one produced, it is discarded without the running system ever being touched (never rollback).
“In Action”

- DCRM system developed by industrial partner.
- 92,160 configurations $\longrightarrow \sim 8.5$ billion transitions
- Adjust bandwidth, physical location.
- Adapt the DCRM w.r.t the features below.
Red: adapted in 890ms. Purple: adapted in 78ms.
Conclusion

- Sometimes, to adapt you need to change the entire internal structure.
- **Dynamic Software Product Line:**
  - configurations $\leftrightarrow$ states
  - reconfigurations $\leftrightarrow$ transitions
- Hard to build/maintain manually!

**Solution** DAS with Models@Runtime:
- Analyze environment.
- Use AOM to weave new Architecture
- Validate the new Architecture
- Generate reconfiguration scripts
- Safely/online adapt the system
Very interesting, albeit very high level presentation.

Hard to argue against with so little detail.

Hard to assess claims with so little in terms of evaluation. Authors’ ICSE’09 paper: case study on home-automation, but also highly abstract. [MBNJ09]

...“engineering” vs “science”? 
I LOVE MY CRM – Special offers extended!

CRM solutions from the market leader – as individual as your company

CRM is the key to success: a good CRM system helps you attract new customers and nurture existing relationships. CAS CRM provides CRM solutions that are precisely tailored to your business and customer needs – whether you're looking for on-premise or on-demand software, customized or industry solutions, or whether you want to buy or lease, the choice is yours. With over 25 years of experience, CAS CRM is the partner of choice for those who want to implement CRM.

Questions?
Brice Morin, Olivier Barais, Gregory Nain, and Jean-Marc Jezequel.
Taming dynamically adaptive systems using models and aspects.