Executable and Translatable UML
- How Difficult Can it Be?

Håkan Burden
University of Gothenburg

Rogardt Heldal
Chalmers University of Technology

Toni Siljamäki
Ericsson AB

SWEDEN
From Idea to System

Platform-Independent (PIM)  Platform-Specific (PSM)

Idea  Reqs.  UML  Java  Binary  System

Informal  Formal
PIM but Formal?

Platform-Independent (PIM) ↔ Platform-Specific (PSM)

Idea → UML → xtUML → Java → Binary → System

Informal ↔ Formal
xtUML: Executable and Translatable UML

• Profile of UML

• Executable:
  — formal and testable models

• Translatable:
  — code generation by model compilers
Executable Model Hierarchy
How Difficult Can it Be?

• Translatable:
  — efficient code
  — generated for different platforms

• Executable:
  — can novice software modellers solve complex problems using xtUML?
Case Study

• Development teams
  — 3-4 bachelor students
  — total of 300 hours

• Hotel reservation system
  — implemented and tested using xtUML

• Analysis using UML

• Object-oriented evaluation criteria
Results

• Model Evaluation:
  — 2009: 18 of 22 teams
  — 2010: 25 of 28 teams
  — Total: 43 of 50 teams

• Experienced learning threshold in 2010:
  — 30 of 90 students confident within 20 hours
  — 75 of 90

• First time for many students to work with asynchronous calls
Discussion

• No libraries – no internet forums

• Relevance to industry

• Algorithmic vs. control problems

• UML is harder to learn and use for our students
Relationship between Model Elements

• Executable, tightly coupled:
  — Component
  — Class
  — State
  — Action

• Informal, loosely coupled
  — Use Case
  — Sequence*
  — Communication*
  — Activity