A Model-Driven Approach to Systems-of-Systems Engineering

Shayne Flint

The Australian National University

September 2008
Outline

1. “One Sentence”
   An application domain

2. Traditional Model-Driven Engineering (MDE)
   Advantages of MDE
   Limitations of MDE

3. Aspect-Oriented Thinking (AOT)
   Overview
   Advantages and Limitations
   Current Projects
Aspect-Oriented Thinking is a methodology for weaving together autonomous multi-disciplinary knowledge to build systems.
Ecological Science - A Software-Intensive System-of-Systems
Traditional Model-Driven Engineering

Advantages of MDE
- Aspect-Oriented Thinking (AOT)

Limitations of MDE

MODELS

PROCESSES
Automated, Human

TARGET SYSTEMS
Intellectual effort is captured in the models and processes rather than implementation artefacts.
Different implementations can be built with different processes without changing the models.
But traditional MDE makes assumptions.

Assumes that we are going to build or modify a system to satisfy a set of requirements and constraints that can be elicited from stakeholders.
But traditional MDE makes assumptions

Existing Model-Driven approaches address the development of specific systems and thus focus on the 'ACT' phase.

**ACT**
Implement improvement

**OBSERVE**
What do I/we see happening in the World?

**ORIENT**
What does it mean to me/us?

**DECIDE**
What can I/we do to improve my/our situation?
Aspect-Oriented Thinking (AOT)

- **AOT is a continuous process of learning**
  - models for exploration, understanding, decision making and communication
  - all phases of the OODA loop
- **... and change**
  - system creation
  - system operation
  - system modification
  - system retirement/destruction
- **... which deals with diverse aspects**
  - knowledge
  - architecture
  - processes
  - methods
  - technology
  - culture
- **... from all applicable perspectives**
  - technical
  - social
  - environmental
  - legal
  - financial
  - political
- **... using a multi-disciplinary approach**
The world is viewed as a dynamically complex System-of-Systems.
Aspects of the world are autonomously modeled by Subject Matter experts.

We already do this kind of thing.

eg. Science
Domain Models are statically and dynamically analysed in order to learn about a problem situation and identify necessary changes.

Specification Archetypes describe the ways in which knowledge captured in Domain Models can be ‘woven’ together to describe systems of a particular kind.
Specifications are formed to describe changes required to learn about and improve the world. Such changes will include new systems and ways of using them, as well as the modification, retirement and destruction of existing systems.
Aspect-Oriented Thinking - ACTION

Domain Models

Specifications

Specifications are implemented by automated and/or manual means

Artefacts such as models, simulations, processes, architectures, source code, hardware designs, configuration data and test harnesses.

Generated Artefacts

The World

Subject Matter Experts

Specifications

Artefacts such as models, simulations, processes, architectures, source code, hardware designs, configuration data and test harnesses.

Translation

Shayne Flint

A Model-Driven Approach to Systems-of-Systems Engineering
Aspect-Oriented Thinking - ACTION

Domain Models

Subject Matter Experts

The World

Artefacts are deployed to impact the world and help us understand it.

Specifications

Specification Archetypes

Generated Artefacts

Translation

Shayne Flint

A Model-Driven Approach to Systems-of-Systems Engineering
Aspect-Oriented Thinking - CONTINUOUS LEARNING AND CHANGE

- Domain Models
- Specification Archetypes
- Specifications
- Continuous Learning and Change
- Generated Artefacts
- Translation

Subject Matter Experts

The World

The World

Shayne Flint

A Model-Driven Approach to Systems-of-Systems Engineering
System Specifications
(including architecture, design, implementation, production, operation and retirement)

Specification

Domain Models
(Any Language)

Specifications describing the ways in which subject-matter knowledge can be assembled to form system specifications.

Artefacts

Generated Artefacts

e.g.

models
simulations
processes
procedures
source code
HDLs
Configurations

Applicable subject matter is modeled and verified autonomously

Specification Archetype

<<reference>>

<<complyWith>>

Implementation Processes

<<reference>>

<<complyWith>>
• **Advantages**
  - Multi-dimensional separation of concerns - *including cross-cutting concerns*
  - Multi-disciplinarity - *without weakening the disciplines*
  - Reuse - *corporate knowledge, intellectual assets*
  - Management of stakeholder agreement/disagreement
  - Preparedness - *assembly of capability*
  - Encourages Interoperability - *through reuse of specification archetypes*
  - Technology Adoption
  - Agility
  - Provenance - *an important issue in computational science*

• **Limitations**
  - Evaluation
  - Tool support
  - Communication
  - Implementation processes
Current Aspect-Oriented Thinking Projects

- **Helping scientists do more science**
  - Collaborators
    - ANU Fenner School of Environment and Society
    - Department of Biology, École Normale Supérieure (ENS), Paris
  - 3Worlds - new generation platform for ecological simulation
  - Re-engineering for Linux clusters
  - High Performance Computing workflow management
  - Model coupling

- **(Web) Services Orchestration**
  - Collaborators
    - CSIRO ICT Centre
  - Archetypes for orchestration of autonomous services
  - similar issues to systems-of-systems engineering

- **AOT Tool Support**
  - Multi-Language support (user defined)
  - Translation engine support
  - Demonstrated in software development domain
  - 2nd generation web-based tool under development
Aspect-Oriented Thinking Project: 3Worlds ecological modeling system

Autonomous Domain Models dealing with various aspects of ecology and simulation (UML, System Dynamics, publications)

Archetypes describe the ways in which domain knowledge can be assembled for the purpose of ecological simulation (constraint prototypes)

Specifications are formed for particular simulations (constraints)

Implementation Processes translate specifications to form implementation artefacts

Various conventional implementation technologies

Domain Experts

Multi-disciplinary domain experts

End users operating at different levels of abstraction

Domain Specific Language (DSL) for ecology sims

Ecological Models (written in DSL)

Ecological Sim DSL translator

Repast Agent-Based Modeling Toolkit

Other tools (Analysis etc.)

Domain Specific Language subsystem

Agent-Based Modeling subsystem

More Specific Needs

More Generic Needs

Minimal Programming Ability

Good Programming Ability

Shayne Flint

A Model-Driven Approach to Systems-of-Systems Engineering
A Model-Driven Approach to Systems-of-Systems Engineering

Shayne Flint

Software-Intensive Systems Engineering Group
College of Engineering and Computer Science
The Australian National University

Tel: +61-(0)2-6125-8183
shayne.flint@anu.edu.au