Model-Driven Software Engineering (MDSE)

Bahman Zamani, Ph.D.
bahmanzamani.com
Computer Engineering Dept.
University of Isfahan

Presented at
Sheikhbahaee University
Baharestan - Isfahan

Feb. 4th, 2010 (1388/11/15)
What is a model?
- A model is a representation of a system

Engineering model?
- A representation of a system that hides some of the properties and highlights the ones that are of interest for the user [Sel 06].
Model Forms

- Mathematical, eg Linear Programming
- Physical, eg for aircraft or bridge
- Diagrammatic, eg Use case modeling

\[
\begin{align*}
\text{minimize} & \quad \sum_{j=1}^{n} c_j x_j \\
\text{subject to} & \quad \sum_{i=1}^{m} \alpha_{ij} x_j \leq b_i \quad (i = 1, 2, \ldots, m) \\
& \quad x_j \geq 0 \quad (j = 1, 2, \ldots, n)
\end{align*}
\]

http://www.cise.ufl.edu/~davis/Morgan/Image13.gif


http://yourmilitaryaircraft.com/pics/c130-20.jpg

http://texnrails.com/images/track/tms/1200701.jpg

Model-Driven Software Engineering
Why Model?

- In most of the engineering disciplines, it is de rigueur to use models when designing a complex system [Sel 03].
- Since today's software systems are becoming more and more complex, benefiting from using models is inevitable [Sel 03].
Model-Driven Approach of Software Development

- Despite the processes that are code-centric, in these approaches, models are the main artifacts which drive the development ➔ Model-Driven.
- The ultimate goal is to automatically generate programs from the corresponding models [Sel 03].
Model-Driven Approaches

- **MDA**: Model-Driven Architecture
- **MDD**: Model-Driven Development
- **MDE**: Model-Driven Engineering
- **MDSD**: Model-Driven Software Development
- **MDSE**: Model-Driven Software Engineering

Don’t worry, just go **Model-Driven**!
A metaphor for MDE

http://www.theenterprisearchitect.eu/archive/2009/08/05/a-metaphor-for-model-driven-engineering
The Unified Modeling Language (UML)

- UML is a graphical language for
  - specifying
  - visualizing
  - constructing
  - documenting
  the software artifacts.

- In the graphical modeling of OO software systems, Unified Modeling Language (UML) is the dominant approach.
- UML 1.1 adopted by OMG in November 1997
- Current release is UML 2.3 (released 9/9/09).

http://www.uml.org/
UML 2.0 Diagrams

[http://www.agilemodeling.com/essays/umlDiagrams.htm ]

1. Activity Diagram: Depicts high-level business processes
2. Class Diagram: Shows a collection of static model elements such as classes
3. Communication Diagram: Shows instances of classes, their interrelationships
4. Component Diagram: Depicts the components that compose an application
5. Composite Structure Diagram: Depicts the internal structure of a classifier
6. Deployment Diagram: Shows the execution architecture of systems
7. Interaction Overview Diagram: Overviews the control flow within a system
8. Object Diagram: Depicts objects and their relationships at a point in time
9. Package Diagram: Shows how model elements are organized into packages
10. Sequence Diagram: Models the sequential logic
11. State Machine Diagram: Describes the states an object or interaction may be in
12. Timing Diagram: Depicts the change in state or condition of a classifier instance
13. Use Case Diagram: Shows use cases, actors, and their interrelationships
Does Model-Driven work?

- Working paradigms:
  - MDA
  - xUML
Model-Driven Architecture (MDA)

- Separates
  - the specification of functionality (business and application logic)
  - the specification of the implementation (underlying platform technology)
- There are two types of models
  - Platform-Independent Model (PIM)
  - Platform-Specific Model (PSM)
- Design once, build it on any platform (.NET, J2EE, CORBA,...)

http://www.omg.org/mda/
MDA: Platform-Independent Model (PIM)

- An abstract “formal” specification of the structure and function of a system, leaving out the technical details
  - Expressed using UML
- Each PIM captures an aspect (or viewpoint) of the system
  - Viewpoints are called domains
  - Domains are combined to produce the system we want to construct
MDA: Platform-Specific Model (PSM)

- The PSM comprises all the functionality expressed in the PIM with the added design concerns

- PSM is expressed using UML extended by adding platform specific details
  - Using UML extension mechanisms (UML Profiles)
MDA: Mapping from PIM to PSM

The PSM is derived from the PIM by applying a set of systematic transformation rules called **mapping**.
Executable UML (xUML)

- An executable version of UML
- Automated methodology utilizing a highly specialized subset of the UML notation

- xUML → Semantically weak elements Action Semantic

- UML Class Diagrams, State Charts, Action Language
xUML tool: BridgePoint Development Suite
My Research

- Model-Driven Software Engineering
- Design Patterns
- Software Engineering
- Quality Assessment
- Pattern Language Verification (PLV)
For more information:

- Visit my page at [bahmazamani.com](http://bahmazamani.com)

Thank you!

Questions

http://www.greateracadianaregion.net/edu/Portals/0/images/cct/Questions.jpg
References