Model-driven engineering: a practical experience

Oscar Pastor, Sergio España, Marcela Ruiz
Centro de Investigación ProS
Departamento de Sistemas Informáticos y Computación
Universitat Politècnica de València

Talk @ CRI February 28, 2014
Model-driven development

Fundamentals of model-driven development for information systems

This material is the result of the joint effort of Sergio España, Arturo González, Óscar Pastor and Marcela Ruiz
Model-driven development
Model-driven development

Model-driven architecture

- REQUIREMENTS ENGINEERING
  - COMPUTATION INDEPENDENT MODEL
- SW SYSTEM ANALYSIS
  - PLATFORM INDEPENDENT MODEL
- SW SYSTEM DESIGN
  - PLATFORM SPECIFIC MODEL
- IMPLEMENTATION
  - CODE MODEL
Model-driven development

Model-to-model transformations

INPUT MODEL

LANGUAGE USED

SOURCE METAMODEL

SOURCE LANGUAGE

TRANSFORMATION

TARGET LANGUAGE

OUTPUT MODEL

LANGUAGE USED

TARGET METAMODEL

MODEL-DRIVEN DEVELOPMENT

MODEL-TO-MODEL TRANSFORMATIONS

Añadir otra traspa con transformaciones

Mejorar los colores de esta para hacerla más atrac3va
<table>
<thead>
<tr>
<th><strong>Model-driven development</strong></th>
<th><strong>Tool support</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>AADL from Carnegie-Mellon Software Engineering Institute</td>
<td><strong>Graphical Modeling Framework</strong> (GMF)</td>
</tr>
<tr>
<td><strong>Acceleo</strong> an open source code generator from Obeo</td>
<td>HyperSenses and <strong>ANGIE</strong> from DELTA Software Technology</td>
</tr>
<tr>
<td><strong>Actifsource</strong></td>
<td>Innovator from MID GmbH</td>
</tr>
<tr>
<td><strong>Apollo for Eclipse</strong> from Gentleware</td>
<td>Integranova from CARE Technologies</td>
</tr>
<tr>
<td><strong>AndroMDA</strong> an open source MDA tool</td>
<td>LEONARDI</td>
</tr>
<tr>
<td><strong>ArcStyler</strong> from Interactive Objects Software GmbH</td>
<td><strong>MagicDraw</strong> from No Magic Inc</td>
</tr>
<tr>
<td><strong>Artisan Studio</strong> from Artisan Software Tools</td>
<td><strong>ManyDesigns</strong> Portofino</td>
</tr>
<tr>
<td><strong>ASCET</strong> from ETAS</td>
<td><strong>MetaEdit+</strong> from MetaCase</td>
</tr>
<tr>
<td><strong>AtomWeaver</strong> from Isomeris</td>
<td><strong>objectIF</strong> from microTOOL</td>
</tr>
<tr>
<td><strong>CoCoViLa</strong> from Tallinn University of Technology</td>
<td><strong>openArchitectureWare</strong></td>
</tr>
<tr>
<td><strong>EnterpriseCoreObjects</strong> by CapableObjects.com</td>
<td><strong>OptimalJ</strong> from Compuware</td>
</tr>
<tr>
<td><strong>Eclipse Modeling Framework</strong> (EMF)</td>
<td><strong>Rhapsody</strong> from IBM</td>
</tr>
<tr>
<td><strong>Enterprise Architect</strong> from Sparx Systems</td>
<td><strong>RISE Editor</strong> from RISE to Bloome Software</td>
</tr>
<tr>
<td><strong>GenerateXY</strong> from DotXY</td>
<td><strong>SCADE Suite</strong> from Esterel Technologies</td>
</tr>
<tr>
<td><strong>Generic Eclipse Modeling System</strong> (GEMS)</td>
<td><strong>Sculpture Platform</strong> from Modelingsoft</td>
</tr>
<tr>
<td><strong>GeneXus</strong></td>
<td><strong>Select Architect</strong> from Select Business Solutions</td>
</tr>
<tr>
<td><strong>Together Architect</strong> from Borland</td>
<td><strong>TOPCASED</strong></td>
</tr>
<tr>
<td><strong>Uniface</strong> from Compuware</td>
<td><strong>Uniservice</strong> from Uniservice</td>
</tr>
</tbody>
</table>
Model-driven development

Think of artefacts that can be engineered...
Model-driven development

Think of artefacts that can be engineered...

- An automatic teller machine (ATM)
- A spaceship
- A building
- A dam
- A bicycle
- An information system
- A multimedia player software
- A conference website
- A mobile phone
- A nanorobot

Do all these artefacts need the same requirements engineering approach?
ELEVATION

Technical Schematic for Experimental Hover Machine Salvage One, also known as THE VOLUTE.

HALF - SECTION
Model-driven development

Model-driven architecture

- REQUIREMENTS ENGINEERING
  - COMPUTATION INDEPENDENT MODEL
- SW SYSTEM ANALYSIS
  - PLATFORM INDEPENDENT MODEL
- SW SYSTEM DESIGN
  - PLATFORM SPECIFIC MODEL
- IMPLEMENTATION
  - CODE MODEL

ABSTRACTION LEVEL
What is an information system?
ORGANISATIONAL SYSTEM

DECISION SYSTEM

INFORMATION SYSTEM

OPERATIONAL SYSTEM
method

technology
Model-driven development

Business process modelling
and requirements specification
Model-driven development

Model-driven architecture

- REQUIREMENTS ENGINEERING
- SW SYSTEM ANALYSIS
- SW SYSTEM DESIGN
- IMPLEMENTATION
- COMPUTATION INDEPENDENT MODEL
- PLATFORM INDEPENDENT MODEL
- PLATFORM SPECIFIC MODEL
- CODE MODEL
REQUIREMENTS MODELLING

CONCEPTUAL MODELLING
Communication Analysis

- **Organisational Modelling**
  - Organisational structure

- **Business Process Modelling**
  - Communicative Event Diagrams

- **Requirements Modelling**
  - Communicative Event Templates

- **Conceptual Modelling**
  - Conceptual Model
    - Object Model
    - Dynamic Model
    - Functional Model
    - Presentation Model

- **Structural Modelling**
- **Reaction Modelling**
- **Interface Modelling**

MODEL COMPILATION

Final application

Interface layer

Business layer

Persistence layer

Communication Analysis
Communication Analysis

- Organisational Modelling
  - Organisational structure

- Business Process Modelling
  - Communicative Event Diagrams

- Requirements Modelling
  - Communicative Event Templates

- Conceptual Modelling
  - Conceptual Model
  - Object Model
  - Dynamic Model
  - Functional Model
  - Presentation Model

- Structural Modelling

- Reaction Modelling

- Interface Modelling
Object Model

Dynamic Model

Functional Model

Presentation Model

Conceptual Model

 boon

oo-Method
**OO-Method**

**Conceptual Model**
- Object Model
- Dynamic Model

**Communicative Event Diagrams**

**Organisational structure**
- Functional Model

**Interface layer**
- Business layer
- Persistence layer

**Final application**
- Interface layer

**Model Compilation**
- Requirements Model
- Conceptual Model
- Object Model
- Dynamic Model

**Presentation Models**
- Requirements Model
- Communicative Event Templates

**Source:** Gartner (2003)

**Productivity (FPs/day)**

<table>
<thead>
<tr>
<th>Category</th>
<th>CARE TECHNOLOGIES</th>
<th>PEER GROUP (AVG.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Management</td>
<td>1845</td>
<td>37</td>
</tr>
<tr>
<td>Quality</td>
<td>856</td>
<td>174</td>
</tr>
<tr>
<td>Specifications</td>
<td>502</td>
<td>18</td>
</tr>
<tr>
<td>Programming</td>
<td>127</td>
<td>13</td>
</tr>
<tr>
<td>Acceptance Tests</td>
<td>1193</td>
<td>40</td>
</tr>
<tr>
<td>Initial &amp; Post-Start-Up</td>
<td>15860</td>
<td>50</td>
</tr>
<tr>
<td>Other Activities</td>
<td>13703</td>
<td>69</td>
</tr>
</tbody>
</table>

Source: Gartner (2003)
Model-driven development

Platform-independent object-oriented conceptual modelling

This material is the result of the joint effort of Sergio España, Arturo González, Óscar Pastor and Marcela Ruiz
Object Model

**Specification of the static view of the software system**

- Defined by an extended **Class Diagram**.
  - **Classes**
    - Attributes
    - Preconditions and Services
    - Integrity Constraints
  - **Relationships between classes**
    - Association
    - Inheritance
  - **Agents**
### Object Model attributes

**OO-Method**

#### Object Model

![Object Model attributes](image)

**Attributes Table**

<table>
<thead>
<tr>
<th>Name</th>
<th>Attribute type</th>
<th>Data type</th>
<th>Id</th>
<th>Size</th>
<th>Default value</th>
<th>Request u...</th>
<th>Nulls</th>
</tr>
</thead>
<tbody>
<tr>
<td>NumVendedor</td>
<td>Constant</td>
<td>Autonumeric</td>
<td>0</td>
<td>30</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Nombre</td>
<td>Variable</td>
<td>String</td>
<td>30</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Apellidos</td>
<td>Variable</td>
<td>String</td>
<td>30</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>NombreComercial</td>
<td>Variable</td>
<td>String</td>
<td>100</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Dirección</td>
<td>Variable</td>
<td>String</td>
<td>50</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Teléfono</td>
<td>Variable</td>
<td>String</td>
<td>50</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Fax</td>
<td>Variable</td>
<td>String</td>
<td>50</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Tipo</td>
<td>Constant</td>
<td>String</td>
<td>1</td>
<td></td>
<td>Yes</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Saldo</td>
<td>Derived</td>
<td>Real</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPostal</td>
<td>Variable</td>
<td>String</td>
<td>10</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Poblacion</td>
<td>Variable</td>
<td>String</td>
<td>50</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Class:** Vendedor
Object Model services
Object Model
transactions

OO-Method
**OO-Method**

- **Requirements Modelling**
  - Conceptual Model
  - Communicative Event Templates

- **Modelling**
  - Organisational Modelling
  - Business Process Modelling
  - Reaction Modelling
  - Interface Modelling

- **Conceptual Model**
  - Object Model
  - Dynamic Model
  - Functional Model
  - Presentation Model

- **Model Compilation**
  - Final application
    - Interface layer
    - Business layer
    - Persistence layer
Dynamic Model

Specifying the Dynamic view of the software system

- It represents the software system behaviour
- It is defined by two diagrams:
  - State Transition Diagram (STD)
    - It specifies the Object’s valid life
    - It defines the available services in each state.
  - Object Interaction Diagram (OID)
    - It specifies valid Interaction between objects such as Global Transactions and Operations and Triggers.
Dynamic Model

OO-Method
A service is activated when its condition is fulfilled
Dynamic Model

- **Inter-Object** Transactions/Operations
- A Global Transaction Involves **different objects** of the same class or different classes

```
(pig_AunqueYu = TRUE)
FOR ALL Moneda WHERE (Moneda.ParticipaEnFrec = TRUE) DO Moneda.ACTPRECISM2PARA(Moneda, ptg_Parcela,
ptg_precio, ptg_Fechaln, ptg_FechaFin).
(pig_AunqueYu = FALSE)
FOR ALL Moneda WHERE (Moneda.ParticipaEnFrec = TRUE) DO Moneda.CREAPRECISM2PARA(Moneda, ptg_Parcela,
ptg_precio, ptg_Fechaln, ptg_FechaFin).
```
Functional Model

Specifies reaction of the software system

- It defines the semantics related to state transitions
- It describes how the execution of events changes the value of class attributes (valuations)
OO-Method

Functional Model

Class: Vendedor

Attribute: TodoVendedor

Event: cargarTodoVendedor

Attribute: Event

TodoVendedor: cargarTodoVendedor =.. NombreComer...
TodoVendedor: cargarTodoVendedor =.. Apellidos
TodoVendedor: cargarTodoVendedor =.. Apellidos + "."...
TodoVendedor: cargarTodoVendedor =.. Nombre
TodoVendedor: cargarTodoVendedor =.. NULL

Valuation:
Attribute: Event

Categories:
- State
- Cardinal
- Situation

Evaluation condition:
Apellidos ◄ NULL AND Nombre ◄ NULL

Event effect:
Apellidos + "." + Nombre

Comment:
Almacena un string correspondiente a "Apellidos, Nombre" en el caso en que ni los apellidos ni el nombre sean nulos.
OO-Method

- Conceptual Model
  - Object Model
  - Dynamic Model
  - Functional Model
  - Presentation Model

Model Compilation

Final application
- Interface layer
- Business layer
- Persistence layer

Requirements Modelling

Organisational modelling

Business Process Modelling

Conceptual Modelling

Structural Modelling

Reaction Modelling

Interface Modelling

OO-Method
Specification of the user interface

- Abstract specification of the user interface
- Independent of the target device
- View-oriented
- Language of patterns organized in three layers
Presentation Model

Hierarchical Action Tree

- IU Service
  - Introduction
  - Defined Selection
  - Argument Grouping
  - Status Recovery
  - Dependency
  - Supplementary Information

- UI Population
  - Filter
    - Order Criterium
  - Display Set
  - Offered Actions
  - Navigation

- UI Instance

- UI Master/Detail
  - Master UI
  - Details UI

Legend

A uses B

17 Existing presentation patterns
Presentation Model

OO-Method
R&D, The Heart and the Soul of the company

Integranova is promoting the development of engineering solutions within the field of University graduates. The scholarship program whose aim is to finance the investigation being made by the professors plus their talented students.

For the development of this investigation work Integranova supports the consultation team from SAB (Scientific Advisory Board), a multidisciplinary council of International Investigators with renowned prestige.

Read more

* Average calculated by Gartner Group
Model-Driven Architecture in Practice
A Software Production Environment Based on Conceptual Modeling

Sergio España Cubillo

Methodological integration of Communication Analysis into a model-driven software development framework

ADVISORS
Arturo González del Río Rams
Óscar Pastor López

December 2011

Tesis de Máster en Ingeniería de Software
Métodos Formales y Sistemas de Información
DEPARTAMENTO DE SISTEMAS INFORMÁTICOS Y COMPUTACIÓN

A model-driven framework to integrate Communication Analysis and OO-Method

Luz Marcela Ruiz Carmona

Advisors: Óscar Pastor López
Sergio España Cubillo
“An entity or collection of entities may be said to be evolving if their value or fitness is increasing over time. Individually or collectively they are becoming more meaningful, more complete or more adapted to a changing environment.”
Motivation

1) To be aligned with the requirements of its environment.
2) Innovate, reduce cost and regulate.
3) Implement strategies for continued progress.

J. Visser. (2012) Change is the constant. ERCIM news - Special theme: Evolving Software. 3.
Motivation

- Industrial motivation
  - An SME approached us with the need to improve their business process management suite.
  - As part of a national project we explored goal-driven evolution of conceptual models.
  - It is interesting to explore how to support the evolution of business process models.
  - Industry partners exposes their needs to support model evolution in reengineering environments.
Challenges

- **Understand** the evolution process.
- **Guidelines** to facilitate the evolution process.
- **Techniques** to record and specify evolution.
  - Traceability support to link the old with the new.
- **Technological** support.
  - Partial automatisation.
  - Support for report generation.
Reverse engineering for forward engineering evolution process: To-be models as-is models system. ABSTRACTION LEVEL.

Main goal
To design a method that involves model-driven analysis of conceptual model evolution
Main goal

- Pattern-based support for organisational evolution
- Delta-based analysis among conceptual models
- Traceability-based support
- Migration and system interoperability
- Goal-driven model evolution
Delta models

1. DERIVATION OF DELTA MODELS
2. MEASUREMENT OF DELTA MODELS
3. ANALYSIS OF CHANGES

LEGEND
MODEL
ASSOCIATION RELATIONSHIP
CHANGE PROCESS
Delta models

\[ \Delta_{\text{models}} \]
Metrics to measure delta models
Rationale for evolution

1. Specification of goals

2. Specification of delta models

3. Relation of goal and delta models

MODEL M1

MODEL M2

DELTA MODEL

GOAL MODEL

CONCEPTUAL MODEL

METAMODEL

METAMODEL

METAMODEL

EVOLUTION PROCESS

Delta (d) between models a and b

INSTANCE OF ASSOCIATION RELATIONSHIP
Pattern-based evolution

Diagram showing the evolution process involving as-is and to-be models, with reverse and forward engineering, pattern repositories, and metamodels.
Patern-based evolution

P1: PATTERN
NAME: Exception in internal treatments
CREATION_DATE: 01/12/2011
PROBLEM_DESCRIPTION: A business activity that involves an authorization can demand an exceptional behaviour related to approvals or decision taking
GOAL_DESCRIPTION: Offer possibility of rejections or alternative decisions.
DISCOVERY_GUIDELINES: The analyst should ask the stakeholders the following questions. Is this business activity just a formal acknowledgement step, or is it a seal of approval? Can it occur that the decision is negative, that the approval is denied? In case the stakeholders keep dismissing such possibility, the analyst should insist. Then, it never has occurred that the decision maker communicated a rejection, right? And you consider impossible that this can ever happen in the future, right?...

APPLICATION1: PATTERN_APPLICATION
DESCRIPTION: During the meeting held on 09/10/2012, we analysed that supplies do not always have enough stock.

AS-IS MODEL

SALE 2
SALES MANAGER ASSIGNS SUPPLIER

SALES MANAGER

SALE 3
SUPPLIER AUTHORISES THE ORDER
SALES MANAGER

SALE 4
TRANSPORT MNGR ARRANGES LOGISTICS
TRANSP. ASSISTANT

SALE 5
INSUR. DEPT. CLERK SPECIFIES CLAUSES
INSUR. DEPT. CLERK

TO-BE MODEL

SALE 2
SALES MANAGER ASSIGNS SUPPLIER

SALES MANAGER

SALE 3 SUPPLIER EVALUATES THE ORDER

SALE 3.1 ORDER IS REJECTED

SALE 3.2 ORDER IS ACCEPTED

SALES MANAGER

SALE 4
TRANSPORT MNGR ARRANGES LOGISTICS
TRANSP. ASSISTANT

SALE 5
INSUR. DEPT. CLERK SPECIFIES CLAUSES
INSUR. DEPT. CLERK
Practical applications
BPMS migrations

User interface v3

Business process model v3

Business process engine v3

Database v3

Extractor from v3

Intermediate XML file to v5

Loader to v5

Intermediate XML schema to v5

TimeProcess migration framework generator

Business process model v5

Business process engine v5

Database v5

Database schema v5

Business process grammar v3

Business process grammar v5

Database schema v3

Database schema v5

LEGEND
- program
- (meta) model
- database
- interpreter / engine
- input / output flows
- other relationship (qualified)
## Practical applications

### European project quality requirements

### Long-term stakeholder. Maintainer of CDD

<table>
<thead>
<tr>
<th>No.</th>
<th>Requirement description</th>
<th>Source</th>
<th>Artefact</th>
<th>Mod.</th>
<th>Characteristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>QR-H.1</td>
<td><strong>Changeability of meta-model</strong></td>
<td></td>
<td>Method</td>
<td>Prod</td>
<td>Maintainability</td>
</tr>
<tr>
<td>QR-H.1.1</td>
<td>It should be possible to change the metamodel. Metrics:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Changes to the metamodel</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Unit: Number</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Measurement method: Analysis of changes between versions of the metamodel whenever a new CDD release is issued, by means of a delta analysis.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Target value: We do not pre-establish a target value. The metric will be informative and serve as evidence of metamodel changeability.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QR-H.2</td>
<td><strong>Changeability of process model</strong></td>
<td></td>
<td>Method</td>
<td>Prod</td>
<td>Maintainability</td>
</tr>
<tr>
<td>QR-H.3</td>
<td><strong>Scalability of the tool</strong></td>
<td>Tool</td>
<td>Prod</td>
<td></td>
<td>Maintainability</td>
</tr>
<tr>
<td>QR-H.4</td>
<td><strong>Changeability of the tool</strong></td>
<td>Tool</td>
<td>Prod</td>
<td></td>
<td>Maintainability</td>
</tr>
</tbody>
</table>

### Long-term stakeholder. Standards organisation

<table>
<thead>
<tr>
<th>No.</th>
<th>Requirement description</th>
<th>Source</th>
<th>Artefact</th>
<th>Mod.</th>
<th>Characteristic</th>
</tr>
</thead>
<tbody>
<tr>
<td>QR-I.1</td>
<td><strong>Innovation with respect to specific standards</strong></td>
<td></td>
<td>Participative modelling</td>
<td>Method</td>
<td></td>
</tr>
</tbody>
</table>
I hope DELTA is full of (+) traces :)}
Model-driven development

www.pros.upv.es