THE RPG DSL: A CASE STUDY OF LANGUAGE ENGINEERING USING MDD FOR GENERATING RPG GAMES FOR MOBILE PHONES

Eduardo Marques, Valte Balegas, Bruno Barroca, Ankica Barisic, Vasco Amaral
The Project

- Development of a language to create RPG games
  - Domain analysis
  - Language Design
  - Textual/Graphical language editor
  - Generate framework code
  - Game properties verification

- Target users: domain experts
Domain Analysis

Choose target platform

Generate framework's Metamodel

RPG language Metamodell

RPG language instance

Framework's Metamodel instance

Transformed to

Transformed to

APN

Source code generation

Verification

Graphical language
**DOMAIN ANALYSIS (1)**

- Identify the RPG Domain characteristics
- Restrict the domain
  - 80% approach
- Textual description
- Feature model
**Domain Analysis (2)**

- **World Map**
  - Scenes
    - Tile map

- **Agents**
  - Hero
    - Inventory
  - Hostile
    - Interaction: Fight
  - Friendly
    - Interaction: Dialogue

- **Resources**
  - Gold, Wood and Metal

- Magics
DOMAIN ANALYSIS (3)

- Agent properties
  - Life points, magic points, strength, agility, intelligence, experience

- Objects
  - Artifacts, equipment, keys, doors, obstacles, traps and switches

- Objectives
  - Reach a specific scene
  - Pick an artifact
  - Interact with an agent

- Time
Domain Analysis (4)
DOMAIN ANALYSIS(5)
Choose target platform

Domain Analysis → RPG language Metamodel → Graphical language

Generate framework's Metamodel

RPG language instance

Source code generation

APN

Verification
TARGET PLATFORM(1)

- Platform analysis:
  - Advantages and drawbacks
  - Ease of learning/development

<table>
<thead>
<tr>
<th>Framework</th>
<th>Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slick</td>
<td>Java based language; Uses LWJGL</td>
</tr>
<tr>
<td>Sphere</td>
<td>Scripting language; Abstraction level that allows typical RPG features implementation</td>
</tr>
<tr>
<td>Corona</td>
<td>Scripting language; Cross-platform compilation for Android and iOS</td>
</tr>
</tbody>
</table>
TARGET PLATFORM(2)

- Corona Framework
- Development of an abstraction layer
  - Implements RPG features
  - Data structure library
    - Decision trees, graphs, menu lists
  - Simplify inter-model transformation
METAMODELS CREATION

- RPG language metamodel:
  - Based on domain analysis

- Framework metamodel:
  - Based on API layer developed over the framework

- 1-1 relation between models (whenever possible)
Choose target platform

Domain Analysis

Generate framework's Metamodel

RPG language Metamodel

RPG language instance

Source code generation

Transformed to APN

Transformed to Verification

Transformed to Domain Analysis

Graphical language
GRAPHICAL LANGUAGE

- Drag and drop approach
  - Ease of use
- Developed using Eugenia and Emfatic
Domain Analysis

Choose target platform

Generate framework's Metamodel

RPG language Metamodel

Graphical language

RPG language instance

Transformed to

Transformed to

Framework's Metamodel instance

APN

Verification

Source code generation
METAMODEL TRANSFORMATIONS

- Generate target platform model instances
- Generate Petri-Nets for verification
- Using ATL
  - Language and *toolkit* for model transformation
SOURCE CODE GENERATION

- From RPG language to target platform language
- To generate verification rules
- Using Xpand – Template based language
Choose target platform

Domain Analysis → RPG language Metamodel

Generate framework's Metamodel

RPG language instance

Source code generation

Transformed to APN

Verification

Graphical language

Transformed to

Framework's Metammodel instance
VERIFICATION(1)

- Create a minimal model of RPG language specific for verification purposes.
- An abstraction to simplify Petri-net generation
**Verification(2)**

- Verified properties:
  - User can finish game
    - Player completes the final objective
  - User can finish game with maximum score
    - Player completes all the objectives
  - Player can complete an objective if he can reach the scene where it is
QUESTIONS?