



Model Driven Engineering



Modeling RPG game using **WebGME**

Raha Naseri

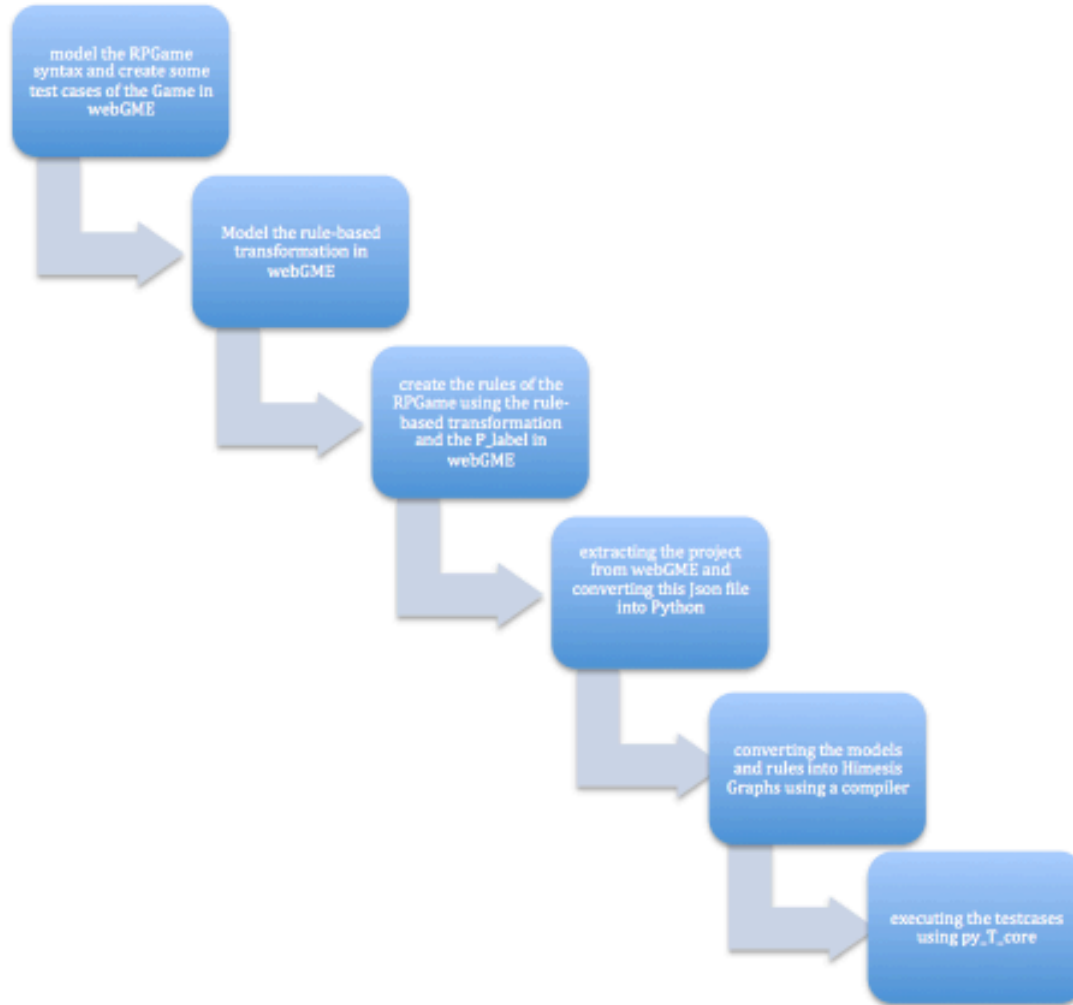
- What is WebGME?
- How does it look like?
- Why should I use WebGME?

The screenshot displays the WebGME editor interface for a project named "RPGR @ master". The browser address bar shows the URL: `editor.webgme.org/?user=demo&create=true&project=demo_87_64_72_61`. The interface is divided into several panels:

- PANEL 1:** Contains navigation options: Composition, Meta, Set membership, Crosscut, Graph view.
- PANEL 2:** Contains additional navigation options: Composition, Meta, Set membership, Crosscut, Graph view.
- Workspace:** A central area showing a graph-based model with nodes and edges. Nodes include HERO, VILLAIN, ITEM, GOAL, FCO, TILECONTENT, HASLABEL, LAI, FILTER, and CONTAINS. Edges represent relationships like "dist" and "wpc".
- OBJECT BROWSER:** A tree view on the right showing the hierarchy of the model, starting from ROOT and branching into Bottom, Contains, Element, and FCO.
- PROPERTY EDITOR:** A panel at the bottom right showing the properties of the selected node (ROOT), including GUID, ID, Attributes (name: ROOT), META, isAbstract, isPort, validPlugins, Pointers, and base.

At the bottom of the interface, there is a status bar with the text "© 2014 Vanderbilt University version: 0.6.5" and several status indicators: "master", "IN SYNC", "CONNECTED", "LOG: WARNING", and "ON".

The plan for this project



Advantages

- Prototypical inheritance
- Automatic save/compilation
- Scalable access to the database on cloud
- Version control
- Online colaboration

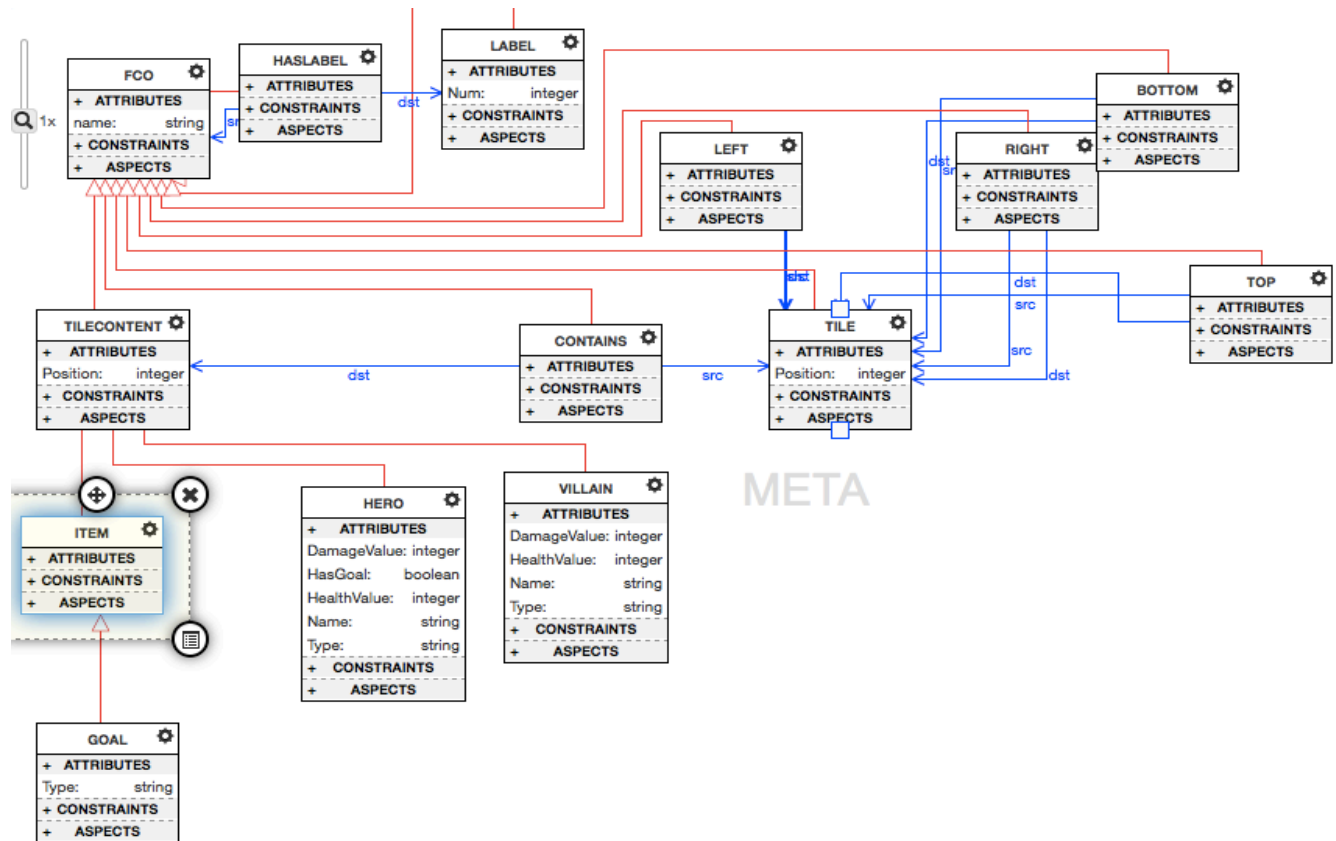
Disadvantages

- No seperate pages for models
- Practically No Multi paradigm
- Fixed inheritance relation
- Limited types available for attributes

Implementation

Modeling RPGGame in webGME

- Abstract syntax



Implementation

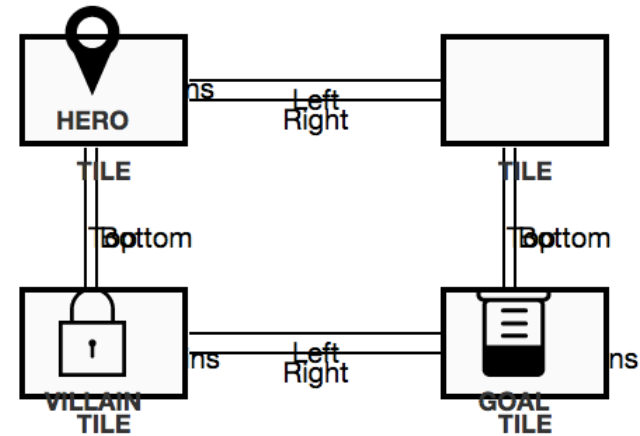
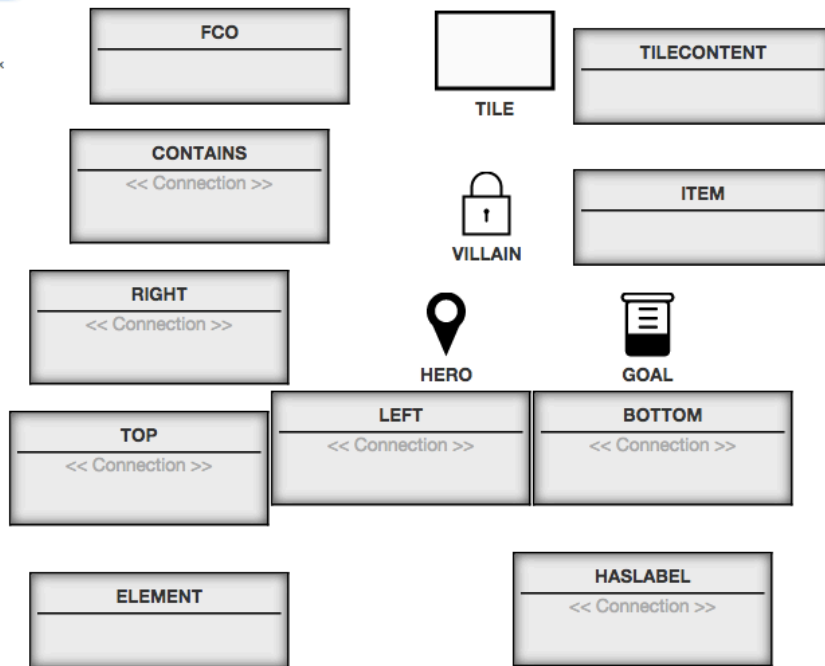
Modeling RPGGame in webGME

- Concrete syntax
- TestCase



ROOT

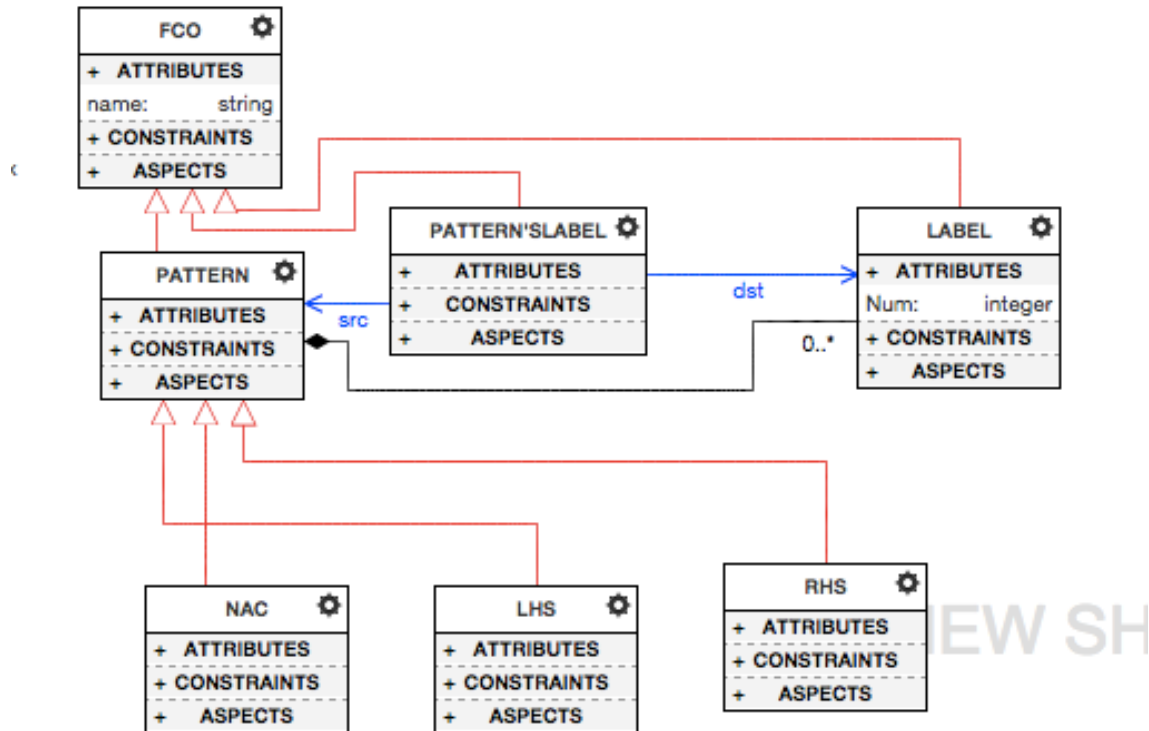
1.5x



Implementation

Modeling Rule-Based Transformation in webGME

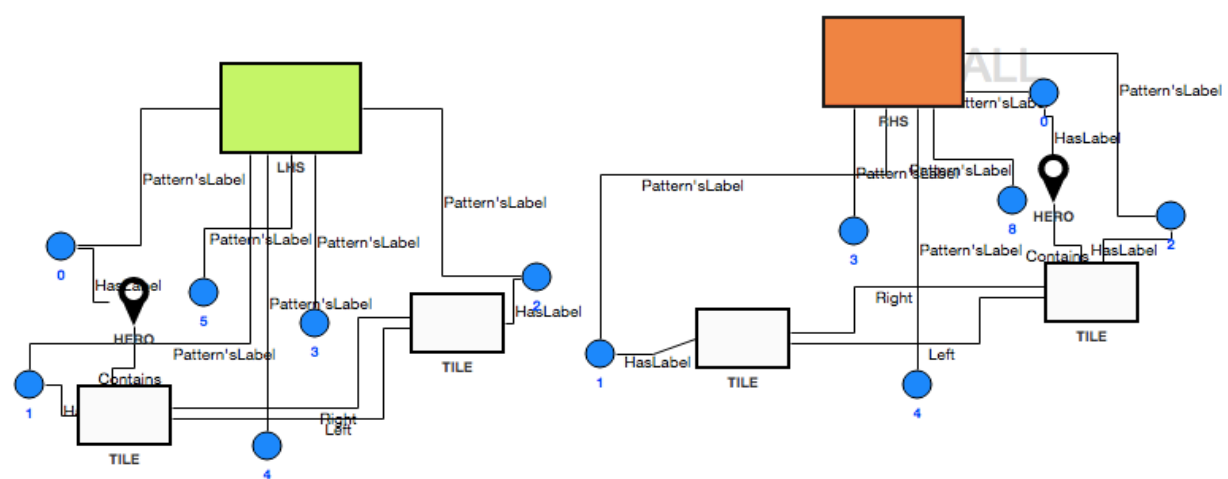
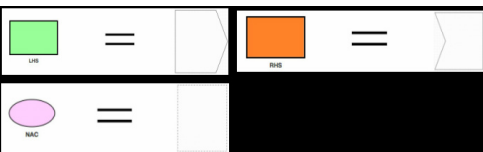
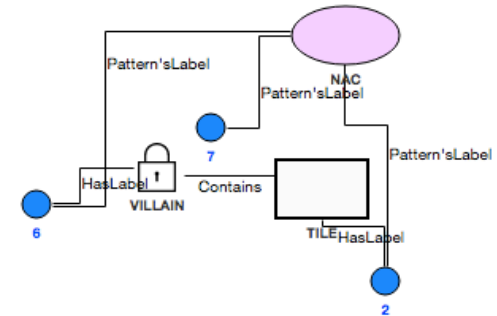
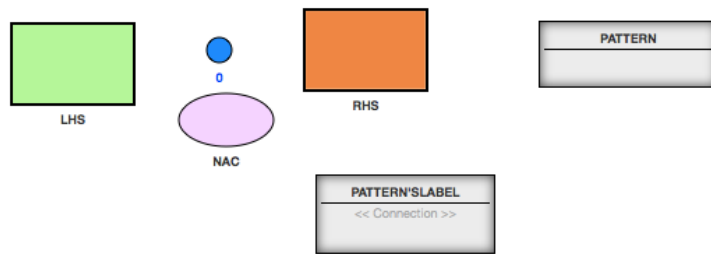
- Abstract Syntax



Implementation

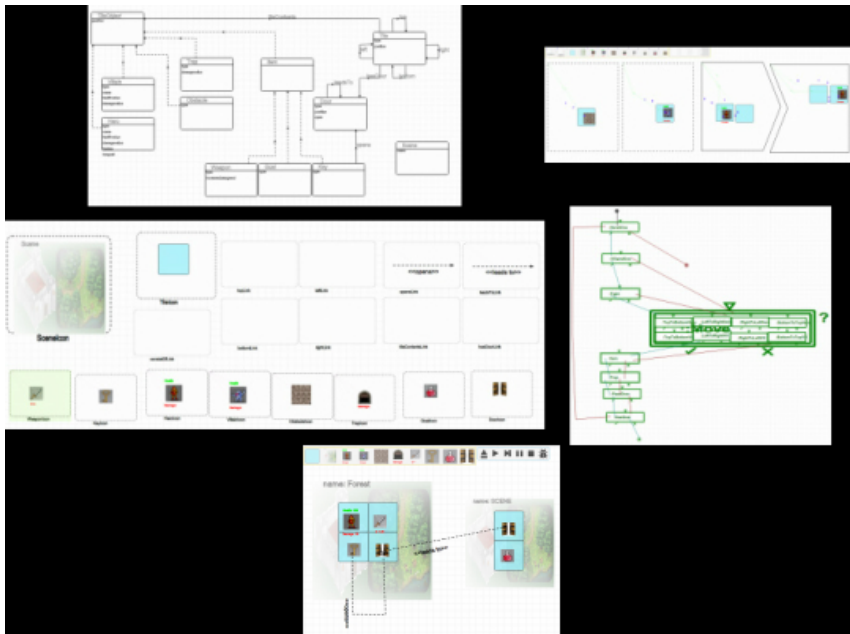
Modeling Rule-based Transformation in webGME

- Concrete syntax
- A RPGame Rule

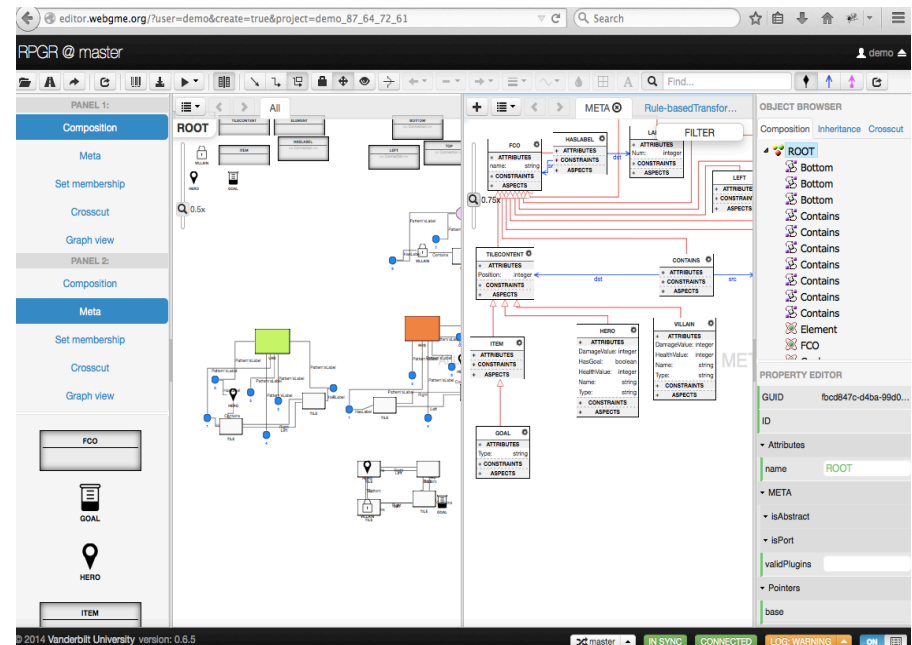


WebGME vs. AToMPM

- AToMPM a complete package for modeling while in webGME the syntax modeling stages is possible
- Simple and structure of AToMPM vs complex structure of WebGME
- Importing images is not possible in WebGME while it is possible in AToMPM
- Compilation and saving Automatic in WebGME while it is Manual in AToMPM
- Online colaboration easier and faster in webGME compared to AToMPM



VS



Questions?