

# ***Desarrollo de inteligencia ambiental con SociAAL***

Pablo Campillo-Sanchez, **Jorge J. Gómez-Sanz**

{pabcampi,jjgomez,jpavon}@ucm.es  
Universidad Complutense de Madrid

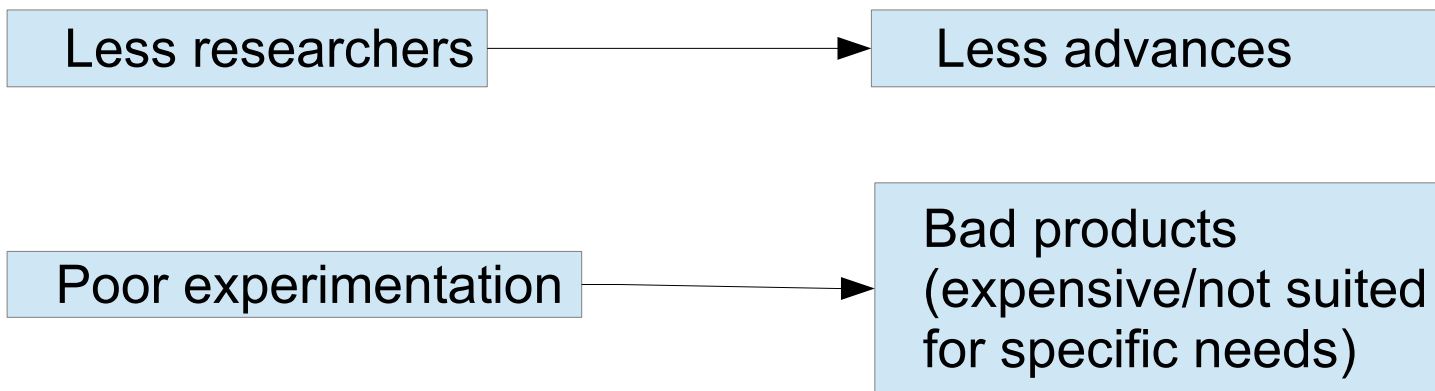
TIN2011-28335-C02-01.

<http://grasia.fdi.ucm.es/sociaal>



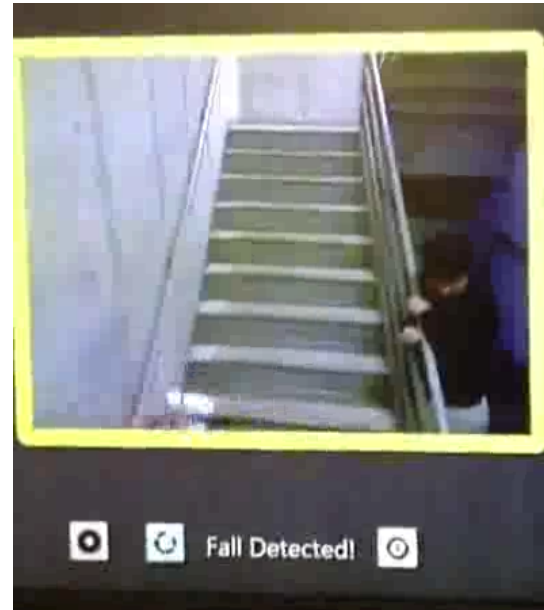
# Introduction

- Model driven solution for supporting Ambient Assisted Living development (AAL)
- AAL is expensive: equipment, human resources, time



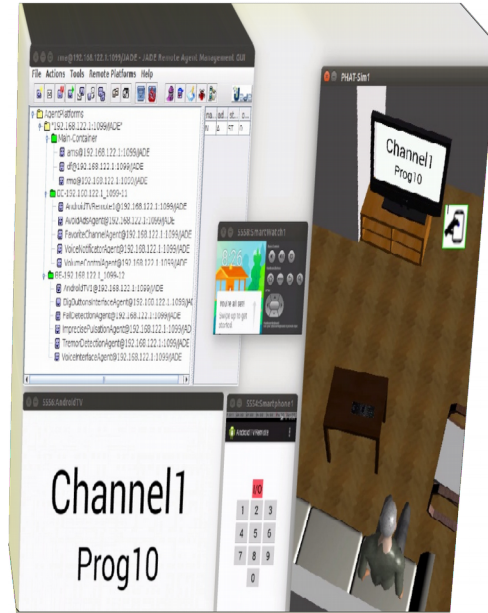
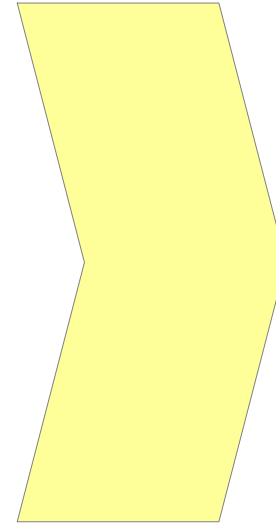
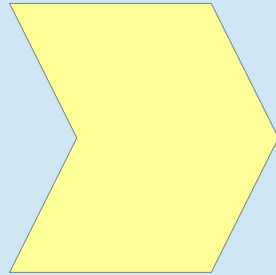
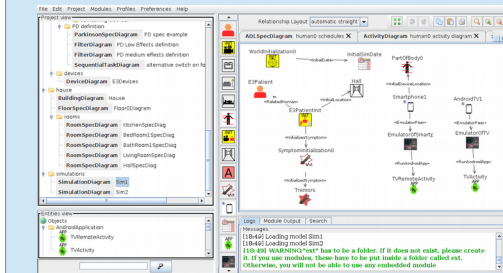
# What a AAL development may look like

- First of all, develop an application that detects when one person falls (videos from LTU CS)
- Once developed, it's testing time
  - It can be dangerous
  - You cannot repeat the experiment in the same way

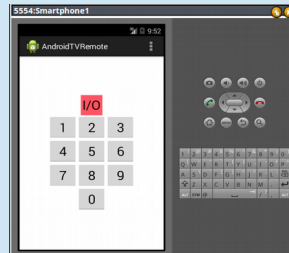


# Virtual Living Lab: Elements

SociAALML : Scenarios, actors  
PHAT: 3D Rendering



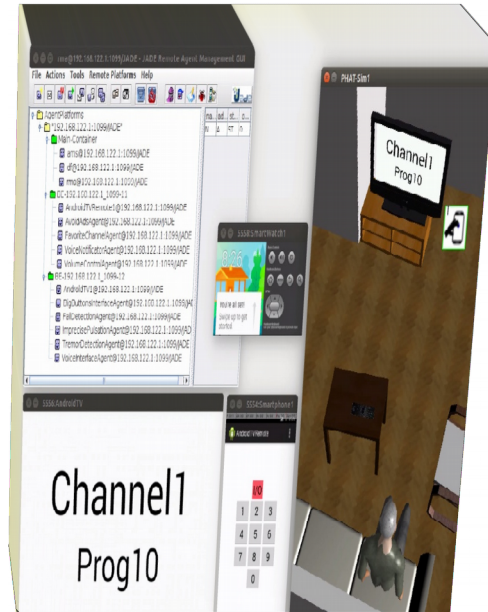
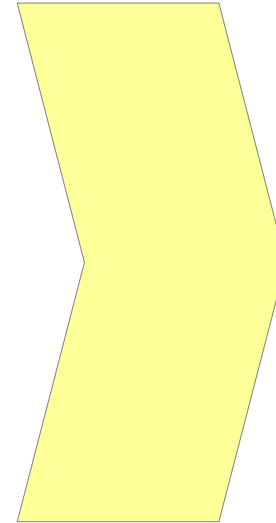
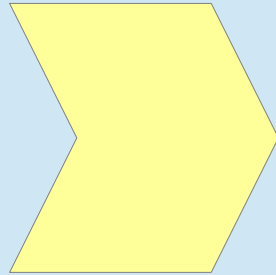
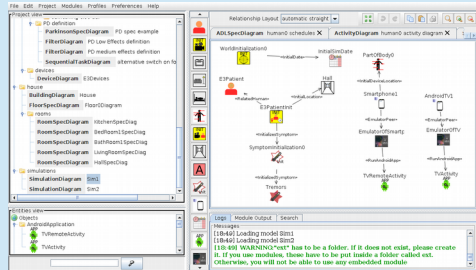
jALI: Aml Devices



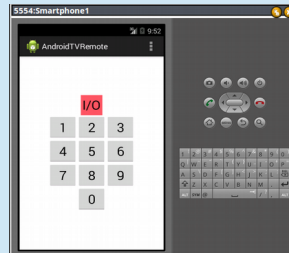
# Virtual Living Lab: Elements

**Model Driven Based: Visual Language + Code generation**

SociAALML : Scenarios, actors  
PHAT: 3D Rendering

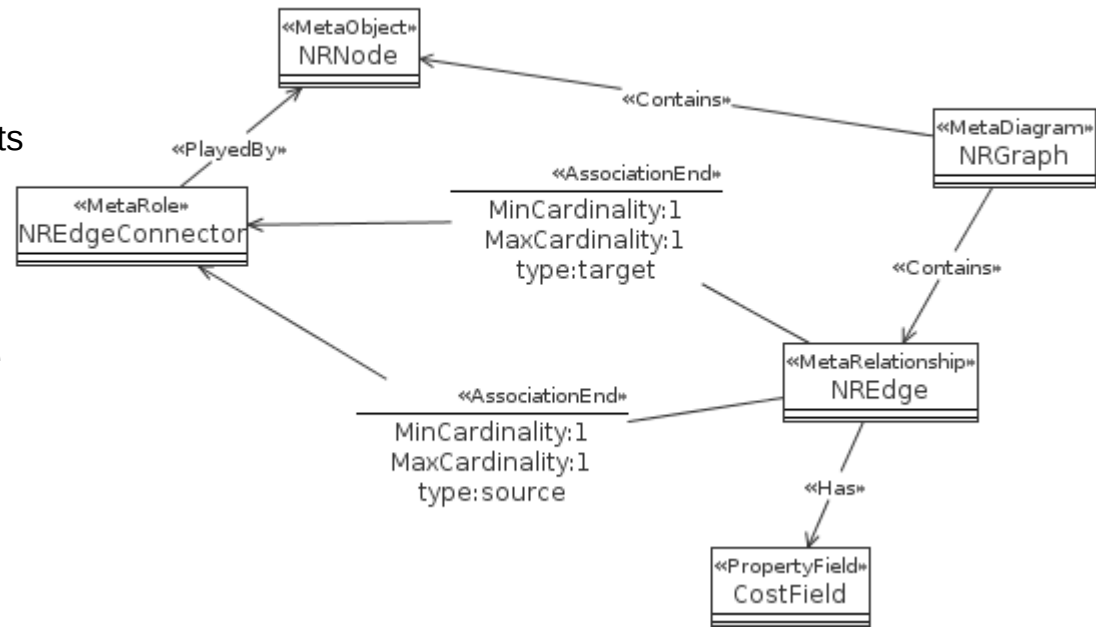


jALI: Aml Devices



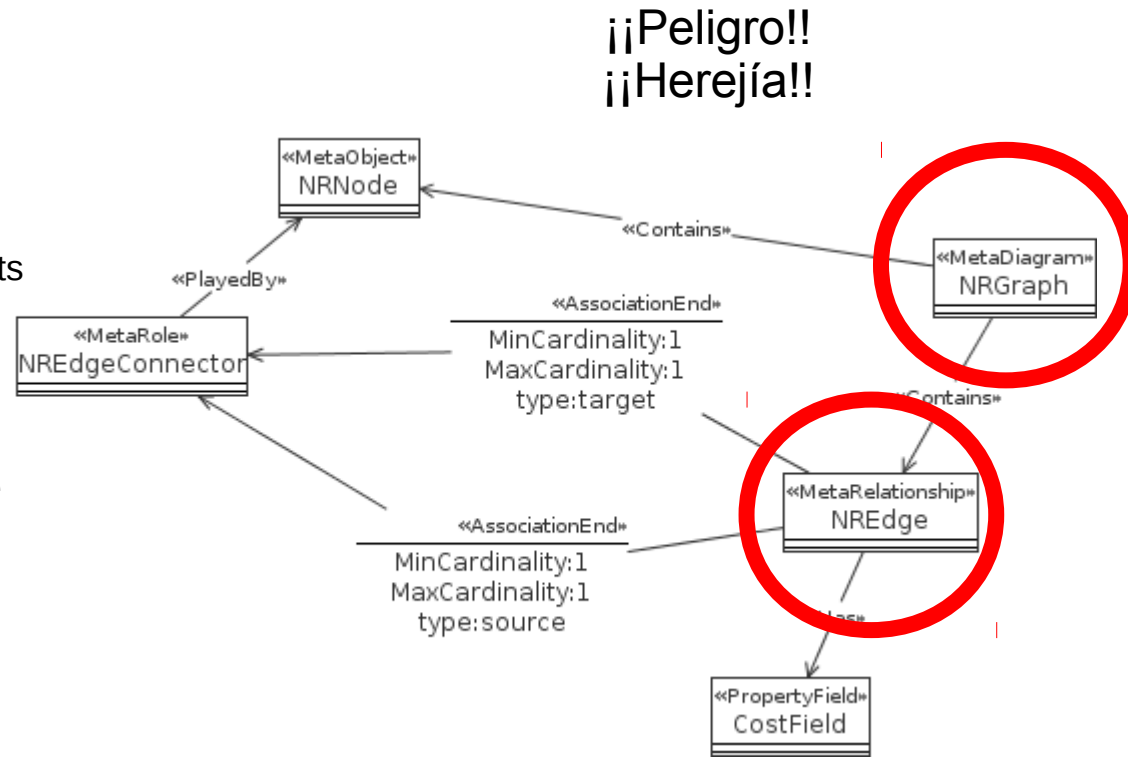
# A custom solution based on open source meta-modeling solution: INGENME

- Goal: quick abstract+concrete grammar for visual languages
  - GOPRR based
    - Couples notation with the grammar
    - First class representation for diagrams & relationship
  - Produces a visual editor + parser: both maven artifacts
    - Share your editor, let others try it!!
- Template in github
  - <https://github.com/Grasia/template-ingenme>
- Not made to be a replacement to EMF but to cope with a simpler alternative
  - Transition to EMF is possible
- Different applications: SelfMML, INGENIAS, SociAAL
- GPL v3



# A custom solution based on open source meta-modeling solution: INGENME

- Goal: quick abstract+concrete grammar for visual languages
  - GOPRR based
    - Couples notation with the grammar
    - First class representation for diagrams & relationship
  - Produces a visual editor + parser: both maven artifacts
    - Share your editor, let others try it!!
- Template in github
  - <https://github.com/Grasia/template-ingenme>
- Not made to be a replacement to EMF but to cope with a simpler alternative
  - Transition to EMF is possible
- Different applications: SelfMML, INGENIAS, SociAAL
- GPL v3







# Produced Editor

The screenshot displays the Produced Editor interface, which is divided into several main sections:

- Project view (top left):** A list of project elements including FilterDiagram, SymptomEvolutionDiagram, and SequentialTaskDiagram, organized into folders like 'tasks' and 'sleepDisorder'. The selected element is 'E3MediumEquilibriumFilter'.
- Entities view (bottom left):** A tree view showing the structure of entities, with 'BActivity' expanded to show sub-entities like 'E3GetUp', 'E3HaveBreakfast', 'E3HaveShower', and 'E3GettingDressed'.
- Diagram editor (center):** A workspace for creating and editing diagrams. The current diagram is a 'Relationship Layout' showing a central 'FTaskSelectorFilter8' node with relationships to 'BGoToTask1', 'COutside0', and 'FUseWalkerOutside'. The relationships are labeled with stereotypes like '<<AllowedTask>>', '<<FP precondition>>', and '<<NextFilter>>'. A 'Walker0' node is also shown as a tool for 'FUseWalkerOutside'.
- Message log (bottom right):** A window displaying system messages, such as 'Loading model E3HusbandHaveLuchActivities' and 'Loading model E3Devices'.

# Coding

- Using 9 INGENME templates
  - Actor behavior
  - Scenario configuration
  - Norm definition
- Information extraction and template instantiation are made through INGENME utilities

```
@@@program xmlns:xsi="http://www.w3.org/2000/10/XMLSchema-instance" @@@
@@@repeat id="diseaseProfile" @@@
  @@@saveto@@@
  @@@file overwrite="yes" @@@
  @@@v@@@output@@@/v@@@/phat/agents/filters/@@@v@@@dpName@@@/v@@@_
  @@@v@@@actorname@@@/v@@@.java@@@/file@@@
  @@@text@@@
package phat.agents.filters;
....
public class @@@v@@@dpName@@@/v@@@_@@@v@@@actorname@@@/v@@@
extends DiseaseManager {
  public @@@v@@@dpName@@@/v@@@_@@@v@@@actorname@@@/v@@@(Agent
agent, String simulation) {
  super(agent);
  initSymptoms(simulation);
  }
....
}
@@@/text@@@
  @@@/saveto@@@
  @@@/repeat@@@
@@@/program@@@
```



# Conclusions

- A Maven based model driven development is possible
  - Ensures stability and eases the software reuse
  - Homogenizes the development
    - regardless the technology, it is always **mvn compile && mvn install**
- If support tools are slowing you down, perhaps they are not as supportive as you thought
  - There are alternatives to EMF
  - Choosing them is not a limiting choice: you can choose again
- SociAALML is a complex code generation case and it is expected to keep on evolving
  - INGENIAS is even more complex

