ColosAAL

Collaborative Ambient Assisted Living Design

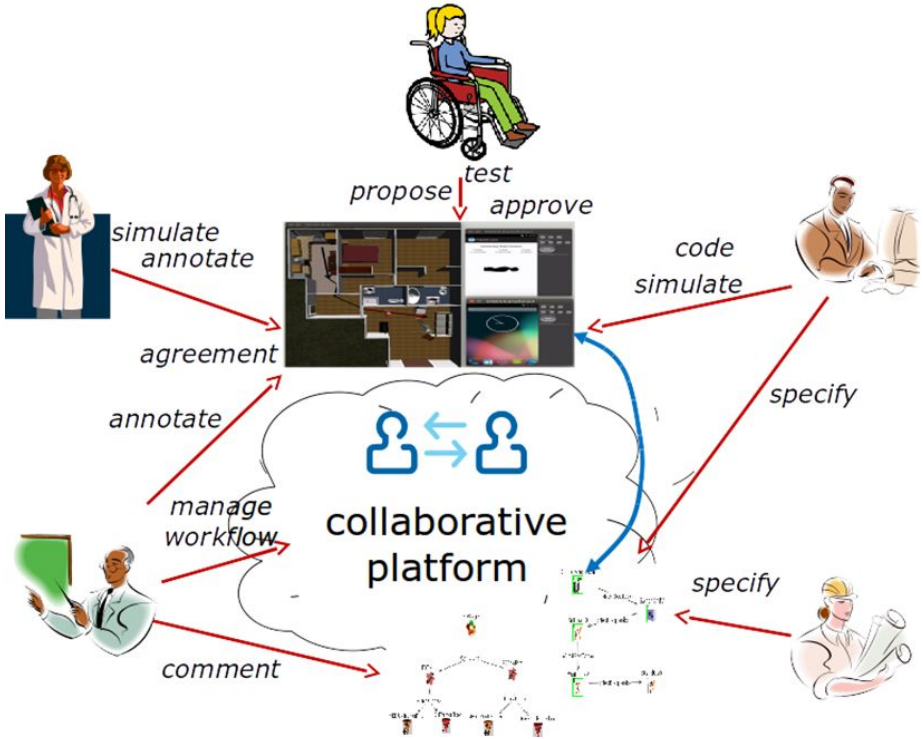
How to build scenarios for simulations

Introduction

ColosAAL Collaborative development of AAL solutions

Human Factor

Assistive Technology



Accessible Spaces

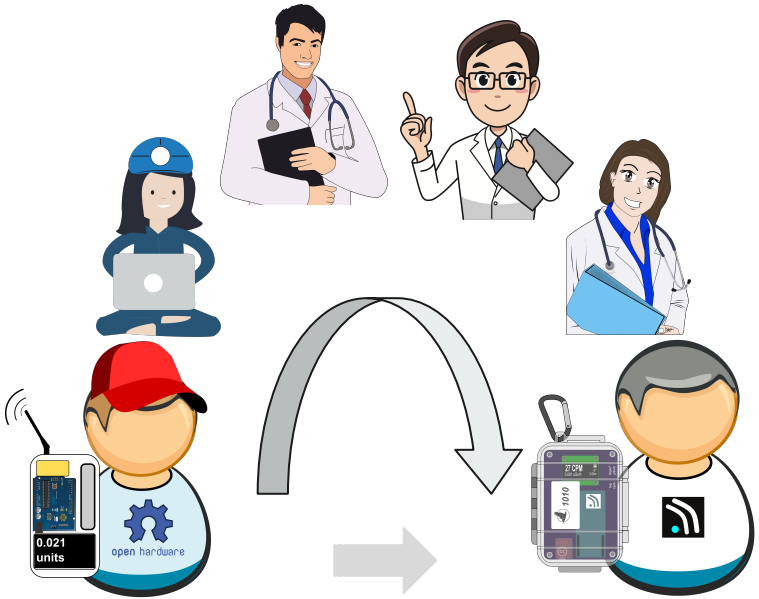
Simulations

UCM Team @ Bologna

Débora Álvarez | Juan Pavón | Marlon Cárdenas | Carlos Cervigón

What role each member of the UCM team

Human Factor



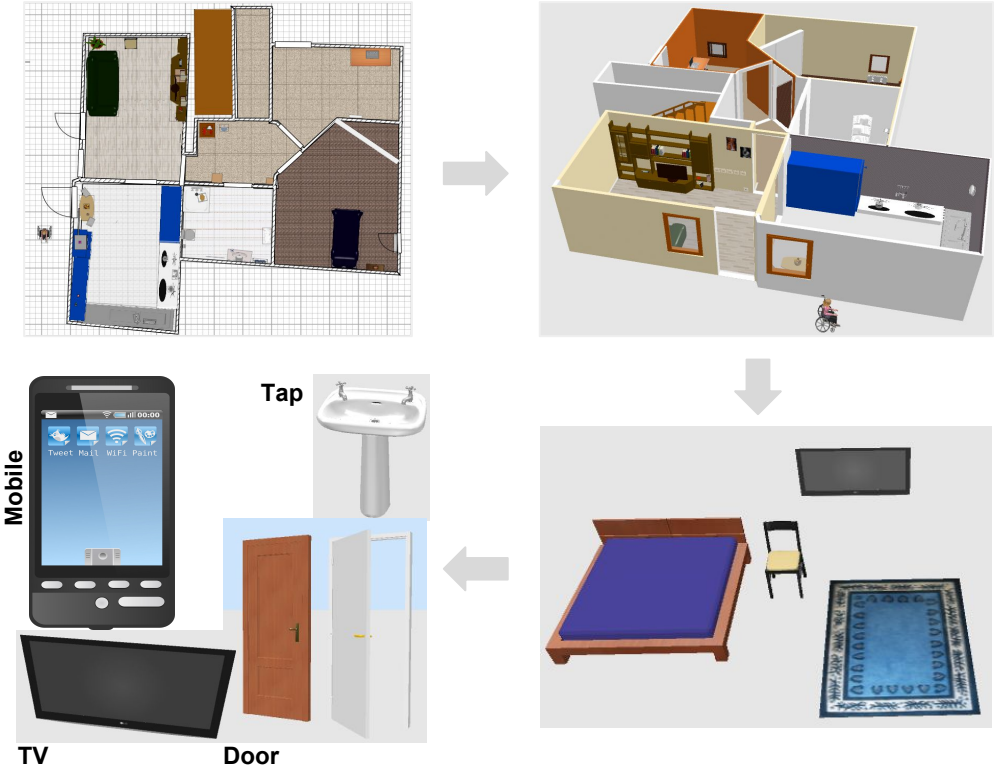
- See the ***experts*** up close
- Know how to interact with ***users***
- Scope of the ***technology***

A goal: Compare ***Reality*** vs. ***Simulations***

Accessible Spaces

- Create the drawings (***with scale***)
- Create and locate accessories or objects (***environmental objects***)
  - Areas of the house
  - Furniture
  - Decoration

Assistive Technology



Modeling Accessible Spaces

- Identify important **spaces.**
- Define the passive and active ***objects of the environment.***
- Identify **technology** in these spaces.
- Define relevant ***user actions.***
  - Thinking about user **comfort.**
  - Improve the **quality of life** and **independence** of users.

How do I build my own scenarios?

The basic steps for modeling scenarios are:

Modeling of spaces

step 1: resize the drawings

step 2: decomposition of objects

step 3: define physical structure

step 4: define logical structure

step 5: import in the simulation the objects

What tools do I need?

Tools

Sweet Home 3D: to design the drawings.

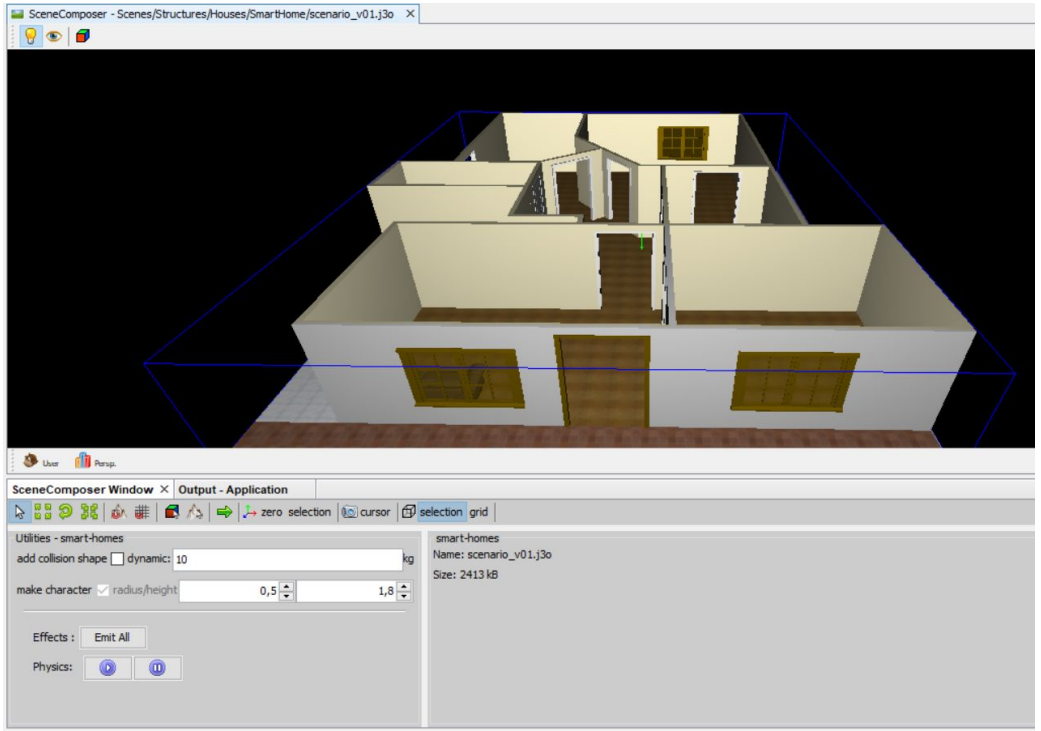
jMonkeyEngine 3.0 (Recommended): to configure the logical structure that the drawings need within the simulation.

PHAT-SIM: to use the plans and create the simulations.

Simulations

***Step 1**:* resize de drawings

- Unify the scales:
  - *design of JMonkey and simulations PHAT*
- Positioning the objects on the scene:
  - Use the spatial nodes to represent all relevant objects in the scenario



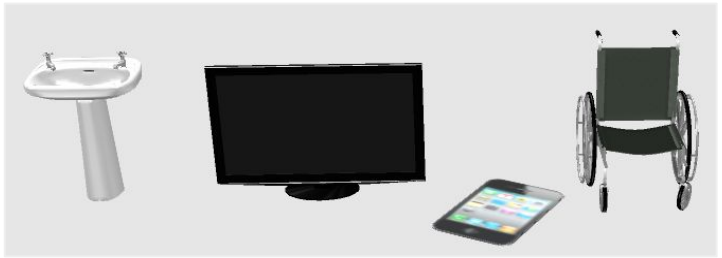
Simulations

***Step 2**:* decomposition of objects

- Identify:
  - Passive objects: no animation
  - Interactive objects: they need animation
- Tag all interactive objects in the simulation

Passive objects

interactive objects

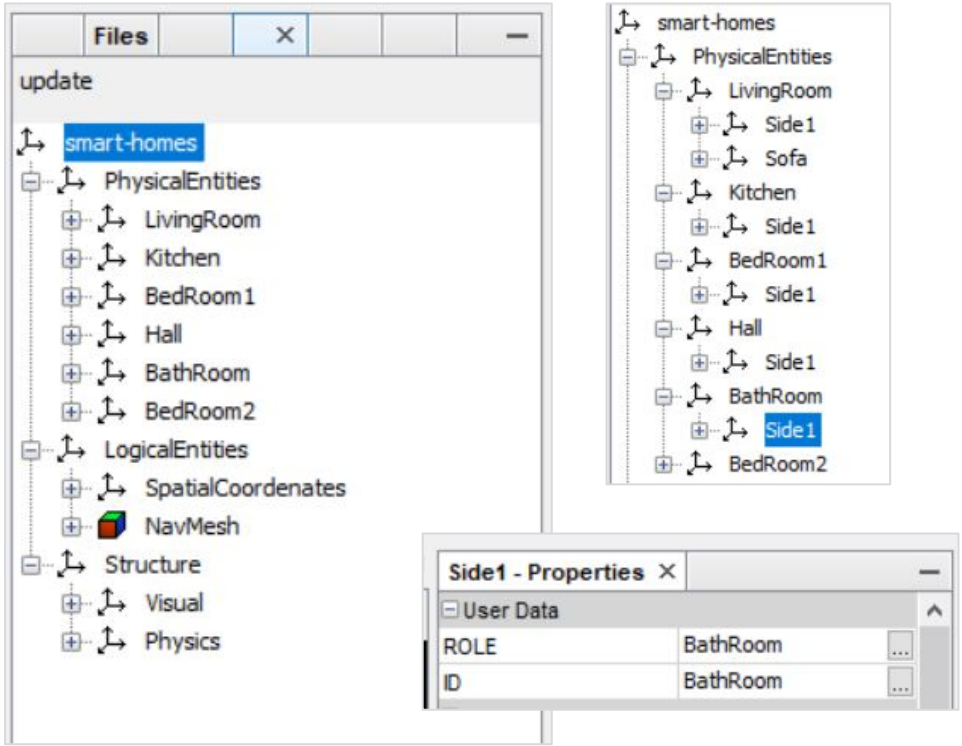


Simulations

***Step 3**: define physical structure*

“only on interactive objects”

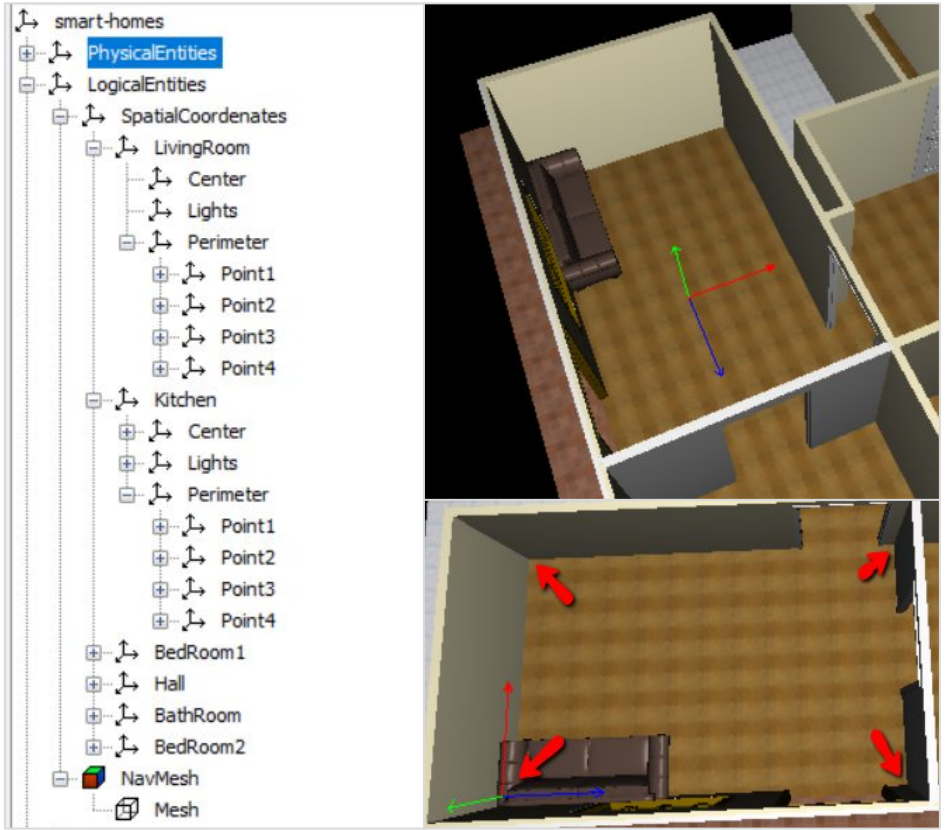
- Create nodes with the position of the object in the scenario
- Add meta information (*ID, ROLE*)



Simulations

***Step 4**:* define logical structure “only on interactive objects”

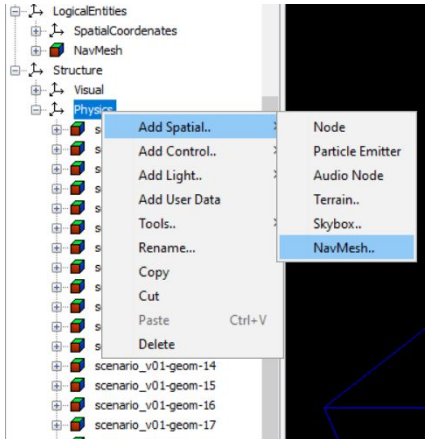
- Create nodes for each space in the scenario
- Create nodes of the objects
- Define the ***center*** and *access* points of the object
- Setting the ***light*** point and set the ***perimeter***

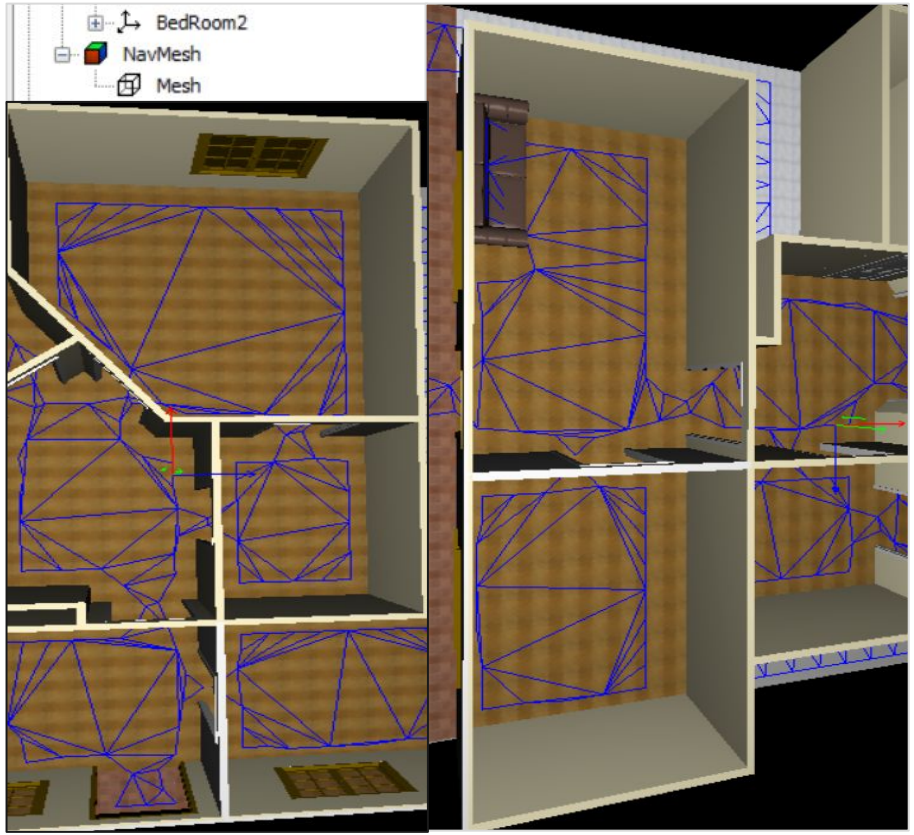


Simulations

***Step 4**:* define logical structure “only on interactive objects”

- *Generate the navigation mesh (**NavMesh**)*





Simulations

***Step 5**: import the objects in the simulation*

- *Build use cases and import modelled objects*

```
@Override
public void initBodies(BodyConfigurator bodyConfig) {
    bodyConfig.createBody(BodiesAppState.BodyType.Elder, id: "Patient1");
    bodyConfig.setInSpace( bodyId: "Patient1", house: "House1", spaceId: "Kitchen");

    bodyConfig.createBody(BodiesAppState.BodyType.ElderLP, id: "Patient2");
    bodyConfig.setInSpace( bodyId: "Patient2", house: "House1", spaceId: "BathRoom");

    bodyConfig.createBody(BodiesAppState.BodyType.Young, id: "Patient3");
    bodyConfig.setInSpace( bodyId: "Patient3", house: "House1", spaceId: "BedRoom1");

    bodyConfig.createBody(BodiesAppState.BodyType.Debok, id: "Relative");
    bodyConfig.setInSpace( bodyId: "Relative", house: "House1", spaceId: "LivingRoom");
    bodyConfig.runCommand(new TremblingHeadCommand( bodyId: "Relative", on: true));
    bodyConfig.runCommand(new SetStoopedBodyCommand( bodyId: "Relative", on: true));
    bodyConfig.runCommand(new TremblingHandCommand( bodyId: "Relative", on: true, left: true));
    bodyConfig.runCommand(new BodyLabelCommand( bodyId: "Relative", show: true));

}
```

What has been done?

step 1: resize the drawings ..... 

step 2: decomposition of objects .....  **Only passive objects**

step 3: define physical structure ..... 

step 4: define logical structure ..... 

step 5: import in the simulation the objects 

Tools and Examples

1. *Project websites*:
   https://grasia.fdi.ucm.es/newmain/language/en/
   http://grasia.fdi.ucm.es/colosaal/
   http://grasia.fdi.ucm.es/sociaal/
   http://grasia.fdi.ucm.es/aide/

2. *Modeling*:
   AIAS http://grasiagroup.fdi.ucm.es/aias/

3. *Examples*:
   https://github.com/mfcardenas/phat_examples
   https://github.com/Grasia/sample-ami-development
   https://github.com/Grasia/base-ami-prototyping

4. *Tools*:
   Hack4People (CamAssistance)
   http://grasiagroup.fdi.ucm.es/hackwithpeople/

Any questions?

Thank you