

# HAAIS-DSL: DSL to develop Home Automation and Ambient Intelligence Systems

Pedro J. Clemente, Juan Hernández, José M. Conejero and Lara  
Sánchez

Quercus Software Engineering Group. <http://quercusseg.unex.es>  
University of Extremadura, Spain

This project has been financed by CICYT, project number TIN2008-02985

March 31th, 2009



# Outline

- 1 Introduction
- 2 Motivation
  - Example: Service building (saving energy)
  - Traditional development process
- 3 HAAIS-DSL
  - Model Driven Development and Domain Specific Modeling
  - Development process based on HAAIS-DSL
  - HAAIS-DSL Metamodel
  - Model to code transformation
- 4 Conclusions

# Outline

- 1 Introduction
- 2 Motivation
  - Example: Service building (saving energy)
  - Traditional development process
- 3 HAAIS-DSL
  - Model Driven Development and Domain Specific Modeling
  - Development process based on HAAIS-DSL
  - HAAIS-DSL Metamodel
  - Model to code transformation
- 4 Conclusions



# Introduction

## Future home and buildings

- People and intelligent buildings interact with each other

## Device coordination

- The number of devices is increasing
- A coordination among devices is required: comfort, security and energy saving.

## Home automation and Ambient Intelligence

- Intuitive interfaces , which are usually embedded in any kind of objects
- →Intelligence relations among users and devices

# Introduction

## Future home and buildings

- People and intelligent buildings interact with each other

## Device coordination

- The number of devices is increasing
- A coordination among devices is required: comfort, security and energy saving.

## Home automation and Ambient Intelligence

- Intuitive interfaces , which are usually embedded in any kind of objects
- →Intelligence relations among users and devices

# Introduction

## Future home and buildings

- People and intelligent buildings interact with each other

## Device coordination

- The number of devices is increasing
- A coordination among devices is required: comfort, security and energy saving.

## Home automation and Ambient Intelligence

- Intuitive interfaces , which are usually embedded in any kind of objects
- →Intelligence relations among users and devices

## Introduction (II)

In order to be able to apply the elements of Aml and HA:

- 1 non-invasive hardware,
- 2 land/mobile communication infrastructures,
- 3 dynamic and distributed net devices,
- 4 managing all these technologies in a secure and reliable way

How do we develop software to these new kind of environments?

- These factors implies a serious technology challenge, especially in the development of **new software to control and monitor** all these systems.
- However, current developments of these systems are carried out by programming them at a **very low level of abstraction**.
- The systems may be **very tied to a specific technology**.



## Introduction (II)

In order to be able to apply the elements of Aml and HA:

- 1 non-invasive hardware,
- 2 land/mobile communication infrastructures,
- 3 dynamic and distributed net devices,
- 4 managing all these technologies in a secure and reliable way

How do we develop software to these new kind of environments?

- These factors implies a serious technology challenge, especially in the development of **new software to control and monitor** all these systems.
- However, current developments of these systems are carried out by programming them at a **very low level of abstraction**.
- The systems may be **very tied to a specific technology**.

## Introduction (III)

### Model Driven Development for these systems

- Increase the abstraction level to develop these systems →MDD
- The **models** could be **the key** to develop these kind of systems
- The process to develop software systems is based on **model transformations** →.NET, Java/Jndi, OSGI, KNX/EIB, etc.

### The objective

- To develop systems for Home automation and Ambient intelligence using Model Driven Development →Increasing the abstraction level required to develop this kind of systems
- Characteristics: rapid system reconfiguration, monitorization, system back-end and system front-end generation (PC, PDA, Phone Mobile)



## Introduction (III)

### Model Driven Development for these systems

- Increase the abstraction level to develop these systems →MDD
- The **models** could be **the key** to develop these kind of systems
- The process to develop software systems is based on **model transformations** →.NET, Java/Jndi, OSGI, KNX/EIB, etc.

### The objective

- To develop systems for Home automation and Ambient intelligence using Model Driven Development →Increasing the abstraction level required to develop this kind of systems
- Characteristics: rapid system reconfiguration, monitorization, system back-end and system front-end generation (PC, PDA, Phone Mobile)

# Outline

- 1 Introduction
- 2 Motivation
  - Example: Service building (saving energy)
  - Traditional development process
- 3 HAAIS-DSL
  - Model Driven Development and Domain Specific Modeling
  - Development process based on HAAIS-DSL
  - HAAIS-DSL Metamodel
  - Model to code transformation
- 4 Conclusions



## Example characteristics

- Several kinds of devices (lights, timers, sensors, etc.)
- Configurations about each devices
- Coordination among devices to obtain an specific behavior: comfort, security, to save energy, etc.

### The system implementation

- Back-End:
  - The code for embedded devices (sensors, actuators, etc.)
  - The code for the configuration system
- Front-End: Web application to manage and monitor the systems (PDA, PC, Phone mobile)

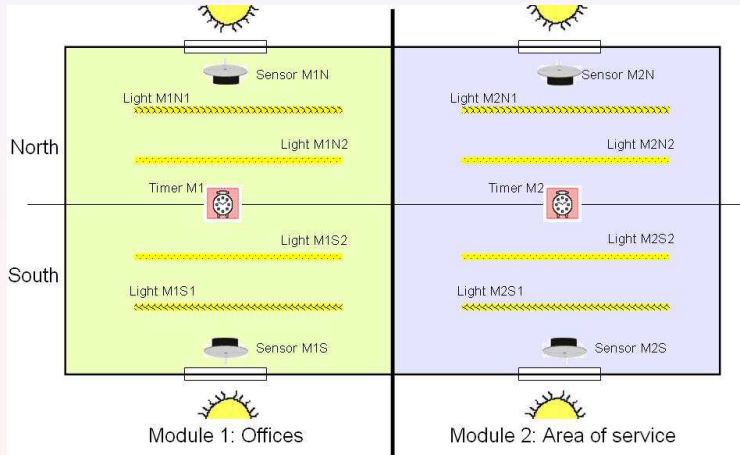
## Example characteristics

- Several kinds of devices (lights, timers, sensors, etc.)
- Configurations about each devices
- Coordination among devices to obtain an specific behavior: comfort, security, to save energy, etc.

### The system implementation

- Back-End:
  - The code for embedded devices (sensors, actuators, etc.)
  - The code for the configuration system
- Front-End: Web application to manage and monitor the systems (PDA, PC, Phone mobile)

# Example: Service building (saving energy)



## Traditional development process

- The development process is based on specific technology (e.g. KNX/EIB, X10, etc.)
- The configurations and deployments are implemented at very low abstraction levels (e.g. instructions level) → Tedious tasks and error prone
- Two kind of systems to develop
  - Physical installation control and configurations, database configurations (**back-ends**)
  - User **front-ends** tools (for example, Web tools)

The system presented is quite simple but then imagine that we also want to control the temperature in the building, there are more devices implied (heating, air conditioning, temperature sensor, blind...) and the relations become much more complex.



## Traditional development process

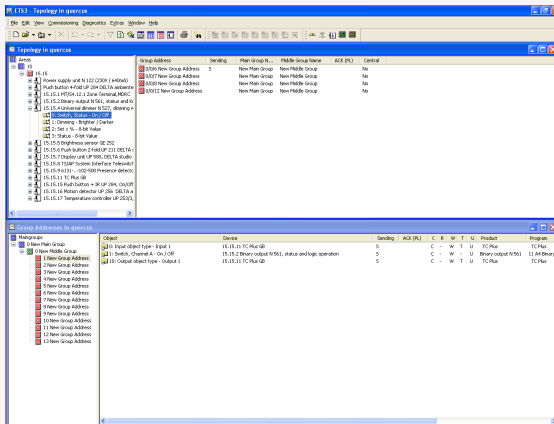
- The development process is based on specific technology (e.g. KNX/EIB, X10, etc.)
- The configurations and deployments are implemented at very low abstraction levels (e.g. instructions level) → Tedious tasks and error prone
- Two kind of systems to develop
  - Physical installation control and configurations, database configurations (**back-ends**)
  - User **front-ends** tools (for example, Web tools)

The system presented is quite simple but then imagine that we also want to control the temperature in the building, there are more devices implied (heating, air conditioning, temperature sensor, blind... ) and the relations become much more complex.





# ETS3 tool for the KNX/EIB systems configuration



To increase the abstraction level of systems development: Model Driven Development

# Outline

- 1 Introduction
- 2 Motivation
  - Example: Service building (saving energy)
  - Traditional development process
- 3 HAAIS-DSL**
  - Model Driven Development and Domain Specific Modeling
  - Development process based on HAAIS-DSL
  - HAAIS-DSL Metamodel
  - Model to code transformation
- 4 Conclusions



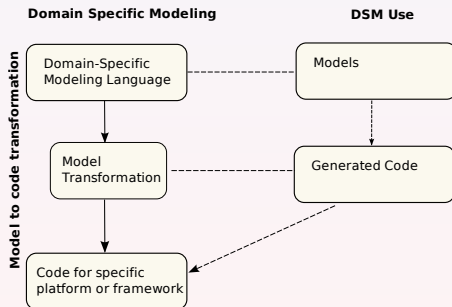
# Model Driven Development (MDD) Overview

- MDD is an emerging software engineering area
- MDD focuses on the **difference** between a system **designed** and its **implementation**
- The **models** can guide the development process
- Using abstracts models, the code for different platforms may be generated using **model transformations**

## Several approaches:

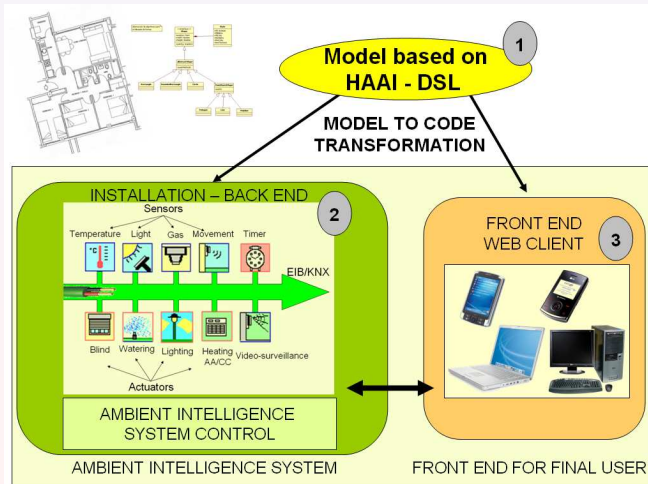
- Model Driven Architectures (MDA)
- Domain Specific Language (DSL)
- Software Factories (SF)

# Domain Specific Modeling (DSM) Overview



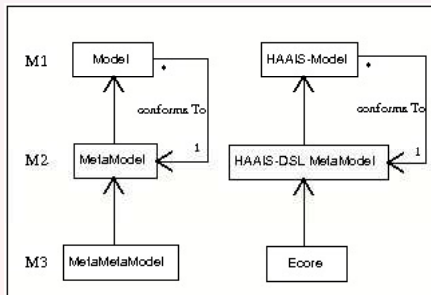
- DSL are based on specific domain concepts
- DSL improve the development of systems based on the same concepts
- Generators and frameworks can be changed to manage different technologies

# Concrete MDD approach to develop these kind of systems



## Defining our DSL: Modeling Stack

- A **technical space** is a model management framework.. For example, a typical M3, M2, M1 stack model.
- It represents specific **working context with specific implementation technologies**



# HAAIS-DSL Metamodel (I)

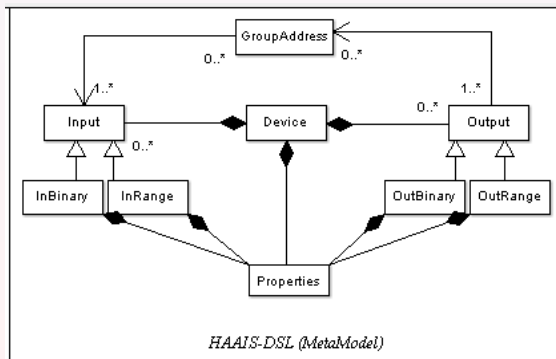
Different kinds of devices to taken into account:

- Light sensors, temperature sensors, movement sensor, timer, buttons, etc. These devices have one or more **outputs**
- Actuators like lights, blinds, air conditioning, heating, etc. are devices with one or more **inputs**.
- Devices can exist with both **inputs** and **outputs**. For example, a camera with a movement sensor.

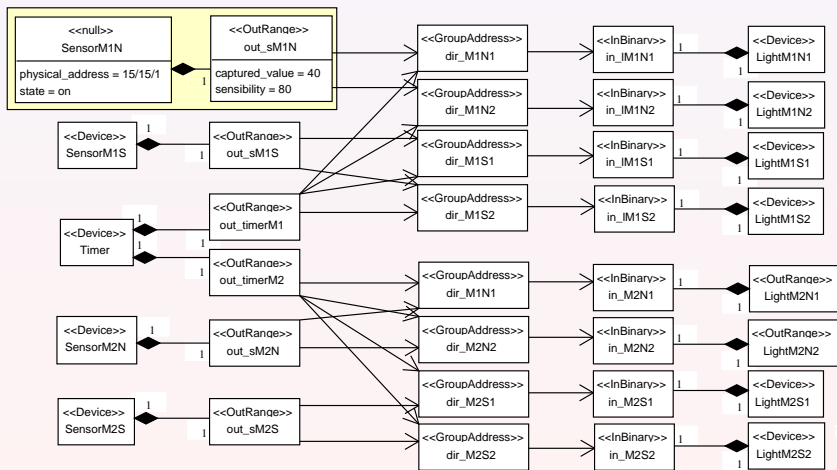


## HAAIS-DSL Metamodel (II)

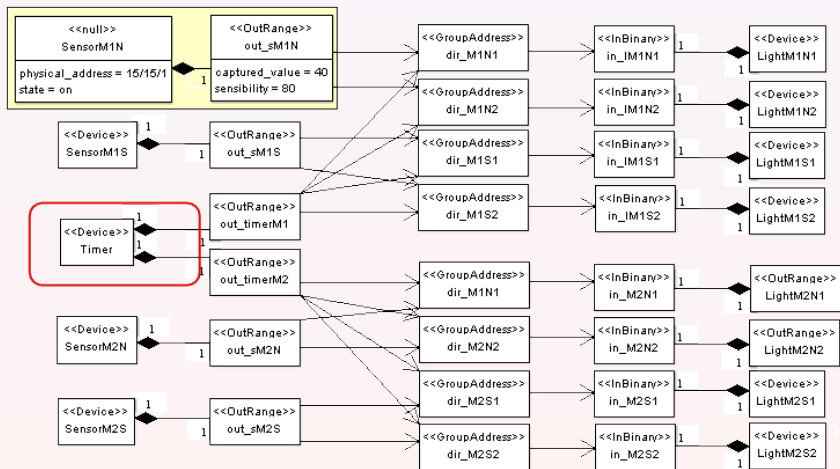
Define the main basic concepts related with HAAI systems: **device**, **input** and **outputs**, **binary or range input and outputs**, **properties** and **group address**.



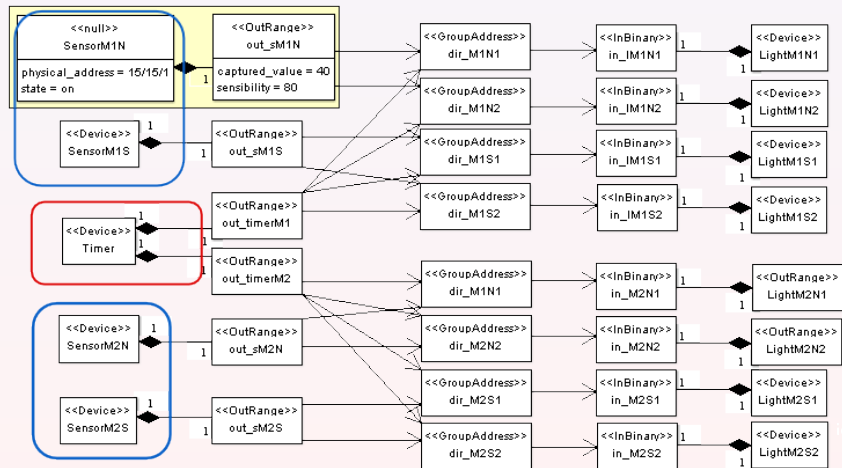
# Example of Model based on HAAIS Metamodel



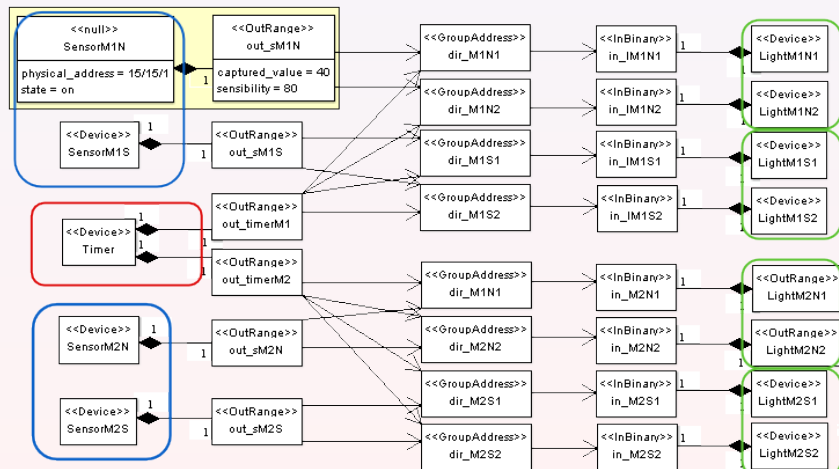
# Example of Model based on HAAIS Metamodel



# Example of Model based on HAAIS Metamodel



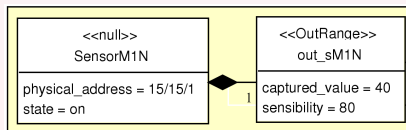
# Example of Model based on HAAIS Metamodel



## Example detail of Model using HAAIS DSL

### How model in detail an specific device?

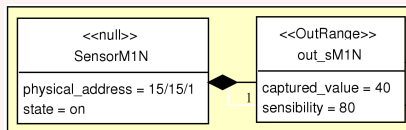
- Modeling each device, input or output using their specific characteristics or properties
- These characteristics or properties are semantic information
- The characteristics or properties could be added and configured at the model, allowing a better knowledge of the system model



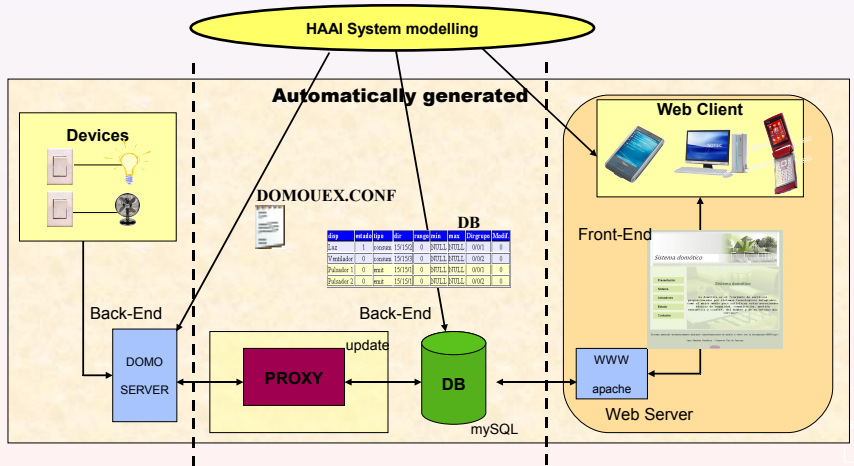
## Example detail of Model using HAAIS DSL

### How model in detail an specific device?

- Modeling each device, input or output using their specific characteristics or properties
- These characteristics or properties are semantic information
- The characteristics or properties could be added and configured at the model, allowing a better knowledge of the system model



# Model to code transformation



## Model to code transformation

### Processing the model using specific languages

- The model normally are stored in XMI (XML Model Interchange - OMG Specification)
- Currently there are several languages to implement model transformations: QVT, ATL, MTL, RubyTL, MofScript ...

### Code example of MofScript

```
texttransformation Ecore2XMLconfigurations (in model_in:'haais_mm') { // HAAIS metamodel
model_in.EPackage::main() {
  file('system_configuration.xml')
  model_in.objectsOfType(model_in.EClass)-> foreach (ecc1) {
    ecc1.devices() // procesing devices information
  }
  model_in.objectsOfType(model_in.EReference)-> foreach (ref) {
    ref.groupAdress() // processing group address
  }
  self.createConfiguration()
...
}
```

# Model to code transformation

## Processing the model using specific languages

- The model normally are stored in XMI (XML Model Interchange - OMG Specification)
- Currently there are several languages to implement model transformations: QVT, ATL, MTL, RubyTL, MofScript ...

## Code example of MofScript

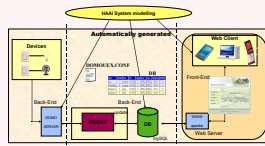
```
texttransformation Ecore2XMLconfigurations (in model_in:'haais_mm') { // HAAIS metamodel
model_in.EPackage::main() {
  file('system_configuration.xml')
  model_in.objectsOfType(model_in.EClass) -> forEach (ecc1) {
    ecc1.devices() // procesing devices information
  }
  model_in.objectsOfType(model_in.EReference)-> forEach (ref) {
    ref.groupAddress() // processing group address
  }
  self.createConfiguration()
...
}
```



## Code generated for specific platforms

### Code generated

- SQL Database schema, configuration and data.
- Complete configuration devices and group devices which are managed by DomoServer (an specific software layer to manage EIB devices).
- Web application implemented on PHP and Ajax to manage the devices installed on the system. Its is accessible by laptops, PDAs, mobile phones, etc.



# Outline

- 1 Introduction
- 2 Motivation
  - Example: Service building (saving energy)
  - Traditional development process
- 3 HAAIS-DSL
  - Model Driven Development and Domain Specific Modeling
  - Development process based on HAAIS-DSL
  - HAAIS-DSL Metamodel
  - Model to code transformation
- 4 Conclusions



# Conclusions

- Model Driven Development process **can be used** to develop Home Automation and Ambient Intelligence Systems
- The models **guide the development process**
- Model transformation allow us to obtain **back-end** and **front-end** Home Automation and Ambient Intelligence
- The **code and configuration generation** decreases the hand written code and the number of errors
- Model Driven Development can be used in **other kind of embedded devices**
- Currently, we are extending the set of devices and their properties
- Developing an specific modeling-tool to help the modeled phase

# Conclusions

- Model Driven Development process **can be used** to develop Home Automation and Ambient Intelligence Systems
- The models **guide the development process**
- Model transformation allow us to obtain **back-end** and **front-end** Home Automation and Ambient Intelligence
- The **code and configuration generation** decreases the hand written code and the number of errors
- Model Driven Development can be used in **other kind of embedded devices**
- Currently, we are extending the set of devices and their properties
- Developing an specific modeling-tool to help the modeled phase

## Questions ?

Thanks for your attention

- mail: [pjclemente@unex.es](mailto:pjclemente@unex.es)
- web page: <http://quercussegu.unex.es/pjclemente>

