

# Semantic Modeling of Smart Homes in the ABS Real-Time Modeling Language

Ming-Chang Lee and Olaf Owe

Department of Informatics, University of Oslo

IoTSec meeting at NR

May 6 2016

# Motivation of abstract modeling

## Interfaces:

- ▶ Abstraction mechanisms
- ▶ High level
- ▶ Generality
- ▶ Future
- ▶ Interfaces more abstract than concrete those of manufacturers

## Model Level Analysis:

- ▶ Comparison of different solutions
- ▶ Security issues
- ▶ Privacy issues
- ▶ Efficiency
- ▶ Properties

# Context

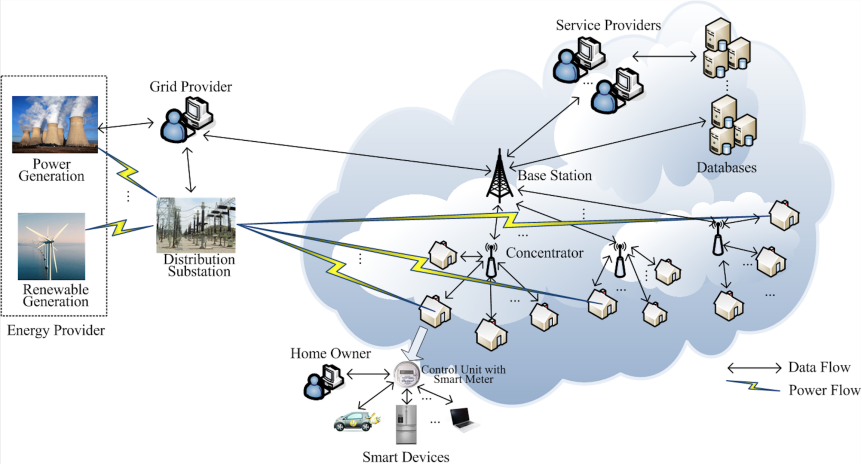


Figure: The architecture of Smart Home Systems.

# Motivation of executable abstract modeling

## Formal and Executable:

- ▶ Semantic specification
- ▶ Reasoning
- ▶ Simulation
- ▶ Model exploration
- ▶ Analysis

## Adaptability

- ▶ Dealing with new devices with new interfaces/functionality
- ▶ Adapting without shutting down the smart home
- ▶ Seamless updates (Creol)

# Focus of this work

- ▶ Interfaces control unit and grid
- ▶ Interfaces control unit and service
- ▶ Interfaces control unit and owner
- ▶ Interfaces control unit and devices (appliances)
- ▶ Interfaces control unit and database

## ControlUnit

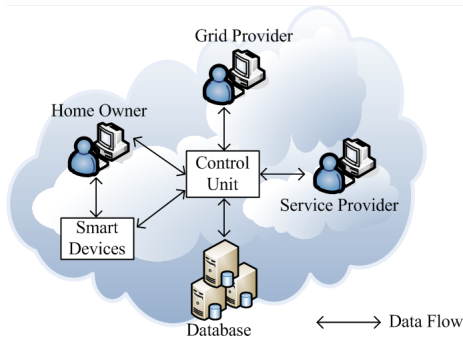


Figure: The interactions of each component in the Smart Home System

# ABS Interfaces

```
interface Owner{ -- methods used by owner
  Unit mode(ControlUnit controlunit, DBase dbase);
  Unit tariff(String tariff,ControlUnit cu, Grid grid, DBase dbase);
  Unit subscription(String subscr,ControlUnit cu, Service s, DBase dbase);
}
```

```
interface Grid{ -- set tariff by owner
  Bool tariffMode(String tariff); }
```

```
interface Service{ -- subscribe subscription by owner
  Bool subscription(String subscription);}
```

```
interface DBase{ -- store data settings in database
  Bool insertRefrigerator(Int volumeFood, String Name, String message);
  Bool insertHeating(Rat indoorTemp, String message);
  Bool insertWindow(Rat outdoorWeather, String message);
  Bool insertLamp(String switch, String message);
  Bool insertEnergyMode(Int id, String energymode);
  Bool insertTariffMode(String tariffMode);
  Bool insertSubscriptionCase(String subscription); }
```

# ABS Device Interfaces

```
interface Refrigerator{ -- purchase food for owner
    Unit orderFood(ControlUnit cu, DBase dbase);}
interface Heater{ -- control smart heater
    Unit controlHeater(ControlUnit cu, DBase dbase);}
interface Window{ -- control smart window
    Unit controlWindow(ControlUnit cu, DBase dbase);}
interface Lamp{ -- control smart lamp
    Unit controlLamp(ControlUnit cu, DBase dbase);}
```

# The interfaces of each component

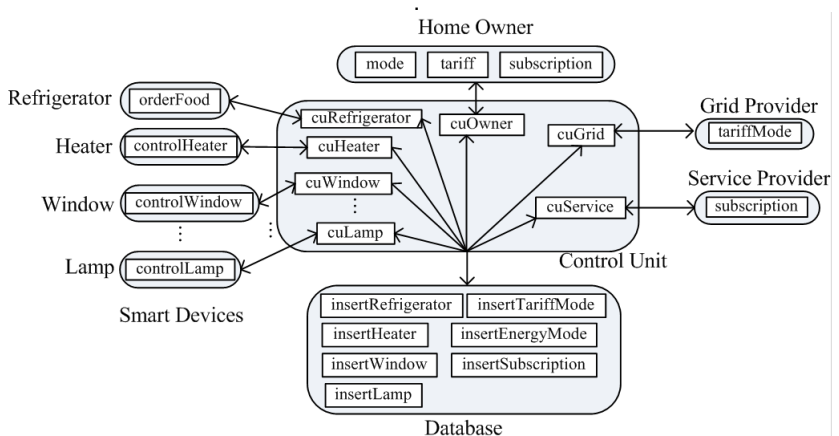


Figure: The interactions of each component in the Smart Home System



# Overview Current Implementation

## Progress report:

- ▶ Interfaces
- ▶ Classes/Deployment components
- ▶ Case example
- ▶ Simulation

# Overview Current Implementation

## Interfaces:

- ▶ Refrigerator, Heater, Window, and Lamp (Smart Devices)
- ▶ Owner: decide energy mode, tariff mode, and subscribe subscription.
- ▶ ControlUnit
- ▶ Grid
- ▶ Service
- ▶ DBase

## Classes/Deployment Components:

- ▶ Refrigerator, Heater, Window, and Lamp
- ▶ Owner
- ▶ ControlUnit
- ▶ Grid
- ▶ Service
- ▶ DBase

# Simulation of Case study: What and How

## Case example:

- ▶ Smart refrigerator
  - ▶ How much food does home owner have (now simulated by random values)
  - ▶ Notify home owner to buy it
- ▶ Smart window: check wind and rain (now also simulated by random values) then open/close window
- ▶ Smart heater: temperature control, simulation uses random temp. values
- ▶ Smart lamp
- ▶ DBase interaction
- ▶ ControlUnit interaction
- ▶ Owner interaction: Energy mode setting
- ▶ Grid interaction: tariff
- ▶ Service interaction
- ▶ Time is used for durations and relative speeds