Dalhousie University
Faculty of Computer Science

CSCI 3132 – Object Orientation and Generic Programming

Week 12 – OOAD Design Problem

The Coffee Maker

- Sources:
  - Coffee Maker problem from Robert Martin's "First Principles of OOD and UML" course.
  - All slides and images taken from "OOAD Design Problem: The Coffee Maker" paper by Jim Weirich [http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf]

The Coffee Maker

The Mark IV special makes up to 12 cups of coffee at a time.

The user places a filter in the filter holder, fills the filter with coffee grounds, and slides the filter holder into its receptacle.

The user then pours up to 12 cups of water into the water strainer and presses the "Brew" button. The water is heated until boiling.

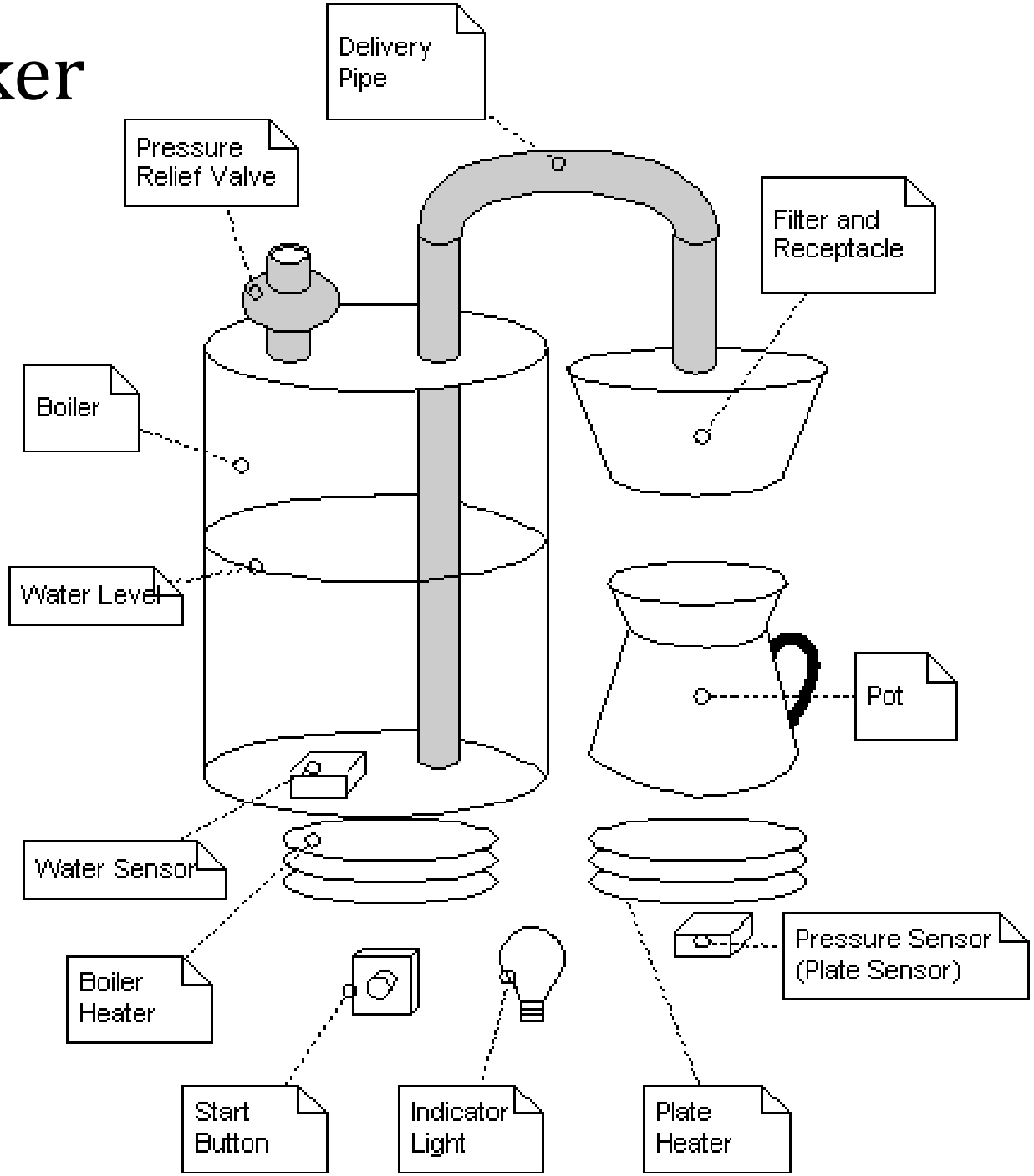
The pressure of the evolving steam forces the water to be sprayed over coffee grounds, and coffee drips through the filter into the pot

The pot is kept warm for extended periods by a warmer plate, which only turns on if there is coffee in the pot.

If the pot is removed from the warmer plate while coffee is sprayed over the grounds, the flow of water is stopped, so that brewed coffee does not spill on the warmer plate.

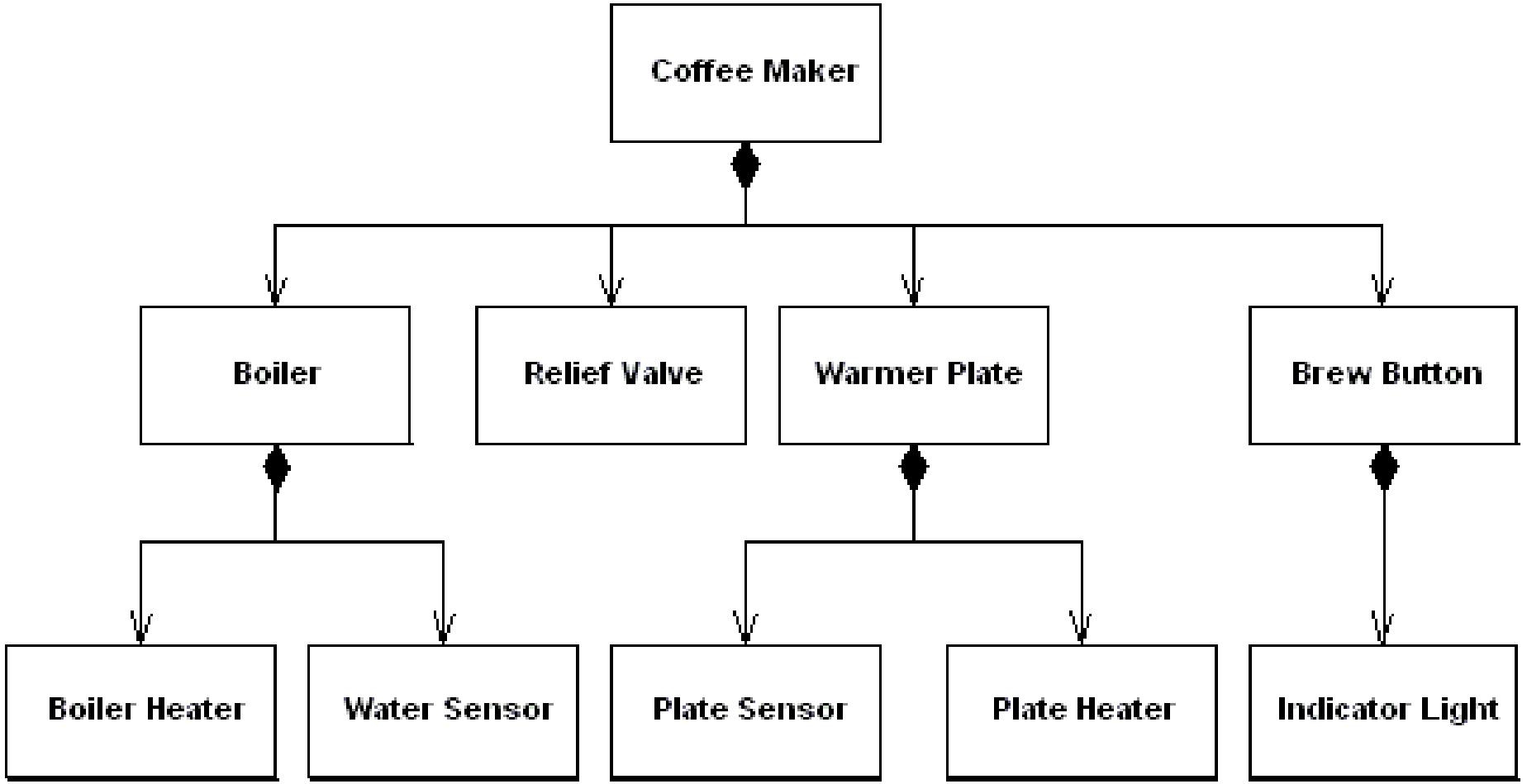
http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

The Coffee Maker



http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

The Coffee Maker – Hardware



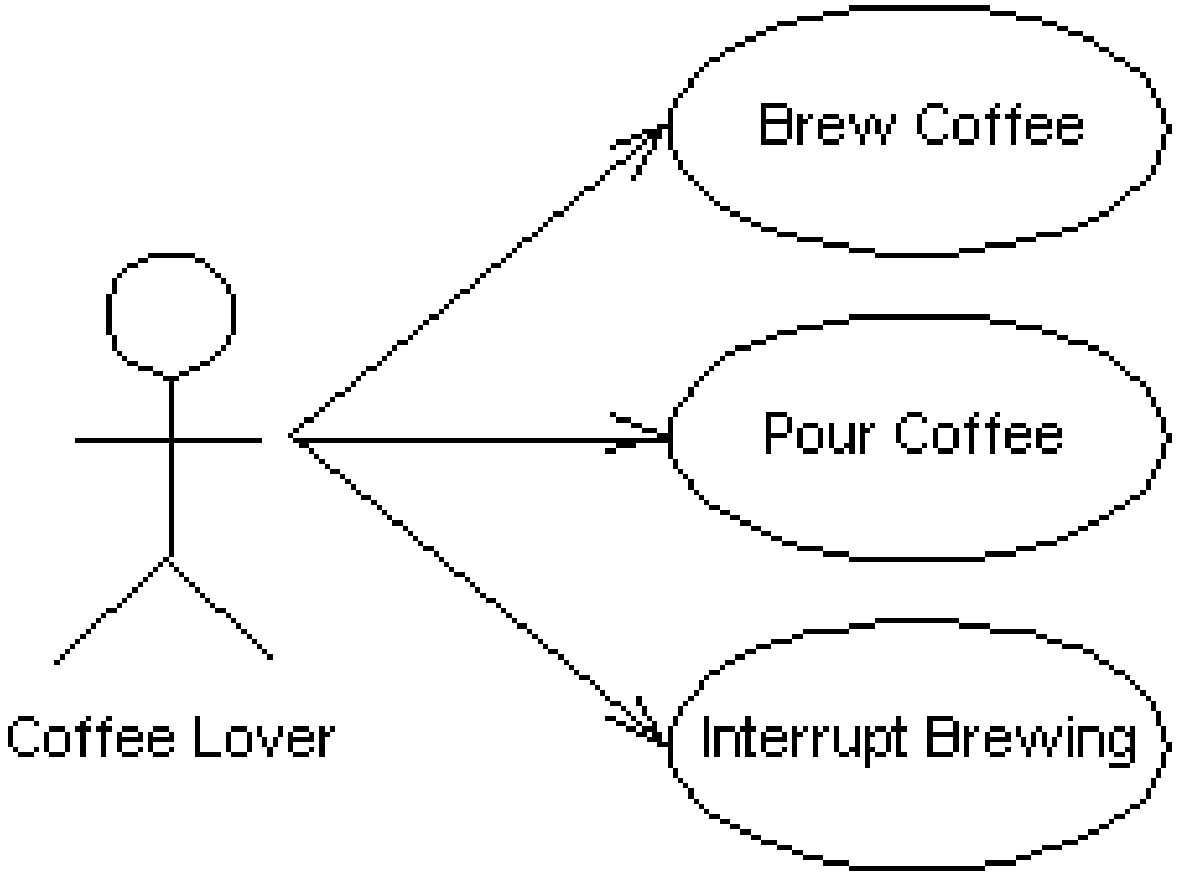
http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

The Coffee Maker – Features

- The heating element for the boiler. It can be turned on or off.
- The heating element for the warmer plate. It can be turned on or off.
- The sensor for the warmer plate. It has three states:
  - **warmerEmpty**, **potEmpty**, and **potNotEmpty**.
- A sensor for the boiler, which determines if there is water present or not. It has two states:
  - **boilerEmpty** or **boilerNotEmtpy**.
- The brew button. This momentary button starts the brewing cycle. It has an indicator that lights up when the brewing cycle is over and the coffee is ready.
- A pressure-relief valve that opens to reduce the pressure in the boiler. The drop in pressure stops the flow of water to the filter. It can be opened or closed.

http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

Basic Use Cases



http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

Use Case – Brew Coffee

Use Case: Brew Coffee
System: Coffee Maker
Actors: Coffee Lover
Precondition: None

Step	Actor (Coffee Lover)	System (Coffee Maker)
1	Puts empty pot on warmer. Fills boiler with water. Puts filter and coffee grounds into filter holder and loads it into the receptacle.	
2	Pushes "Brew Button"	
3		Closes relief valve to allow water to flow to receptacle and turns on boiler heater.
4		When boiler is empty, turns off boiler heater. The indicator light is also turned on.

http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

Use Case – Pour Coffee

Use Case: Pour Coffee
System: Coffee Maker
Actors: Coffee Lover
Precondition: The Coffee Maker is not Brewing

Step	Actor (Coffee Lover)	System (Coffee Maker)
1	Lifts pot from warmer.	
2		Detects the pot has been lifted and turns off the plate warmer. If the indicator light is on, then turn it off (see note below)
3	Replaces pot on warmer	
4		If there is coffee remaining in the pot, restart the plate warmer.

http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

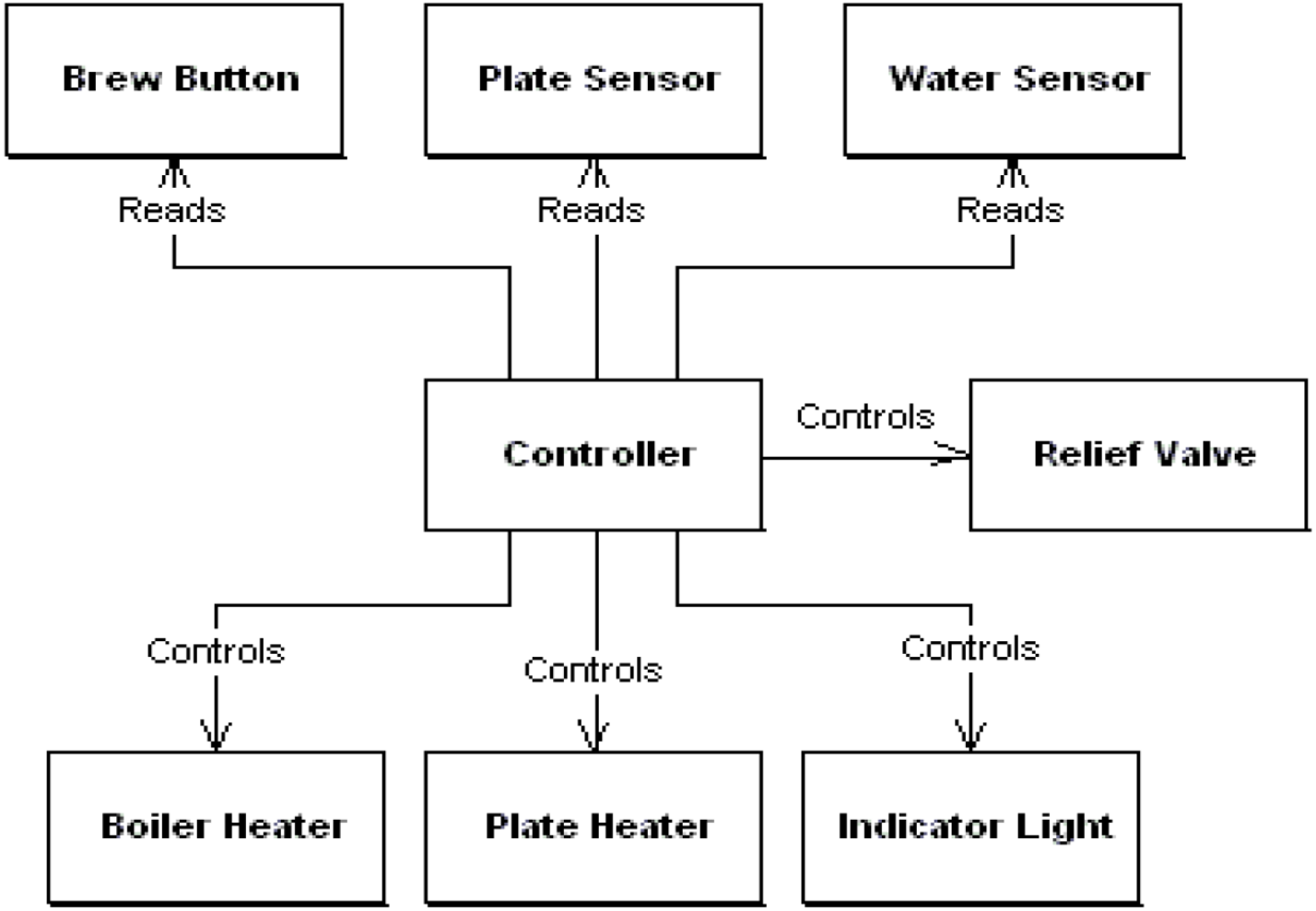
Use Case – Interrupt Brewing

Use Case: Pour Coffee
System: Coffee Maker
Actors: Coffee Lover
Precondition: Brewing in progress

Step	Actor (Coffee Lover)	System (Coffee Maker)
1	Removes the pot from the plate warmer.	
2		Turns off the boile and opens the pressure relief valve.
1	Returns the pot to the plate warmer.	
2		Turns on the boiler and closes the relief valve, continuing the normal brewing cycle.

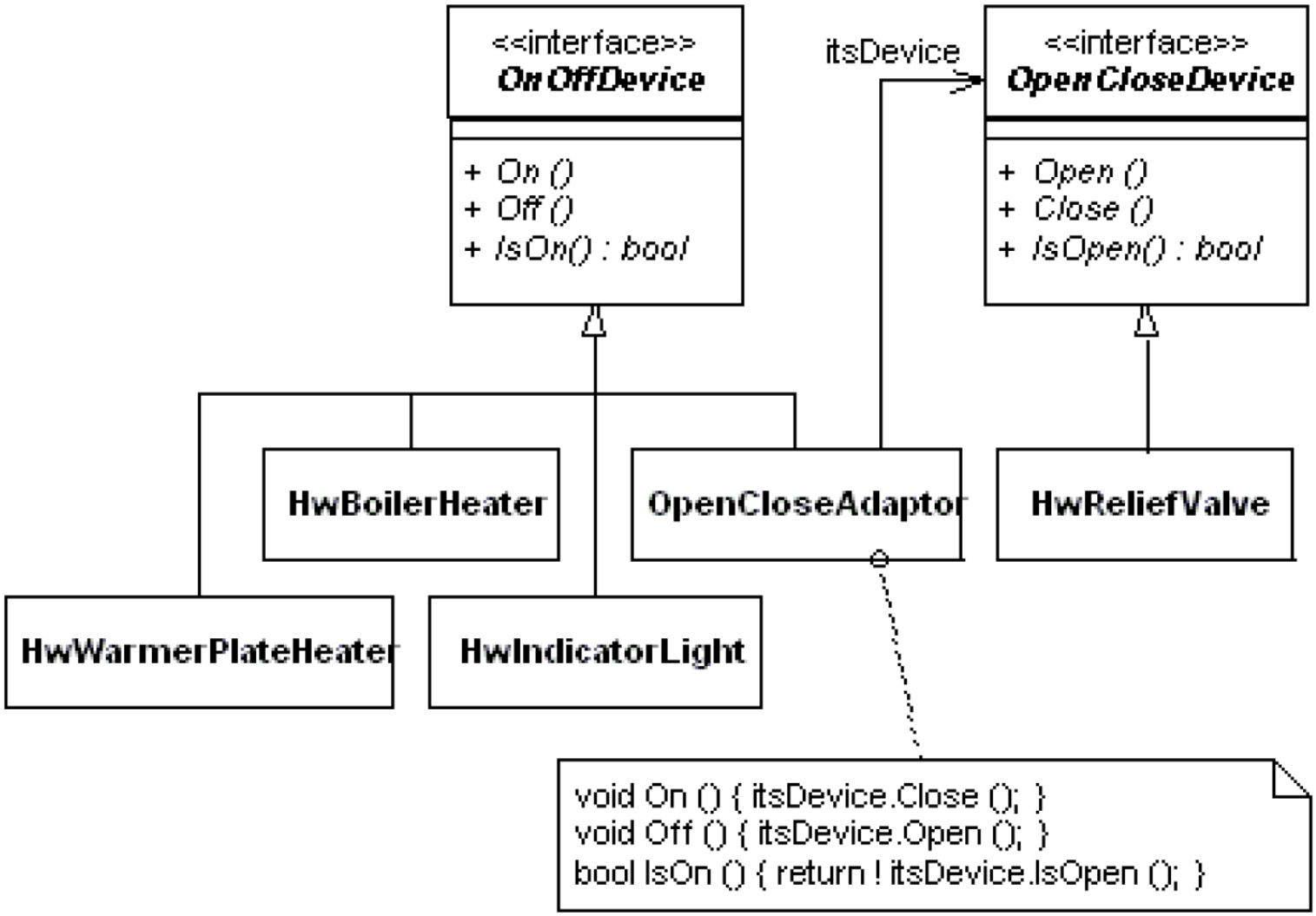
http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

Initial Design



http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

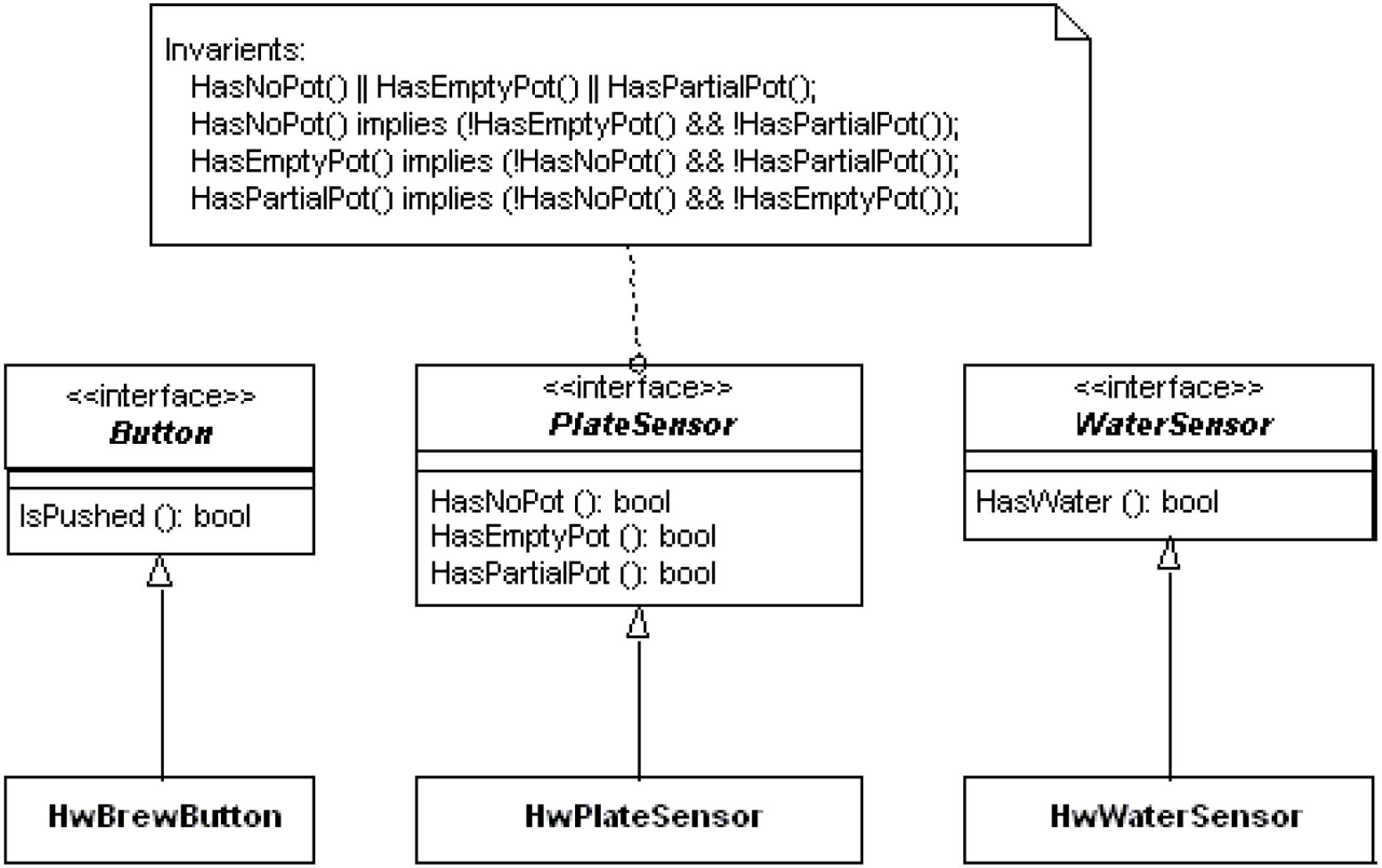
Class Diagram – Output Devices



http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

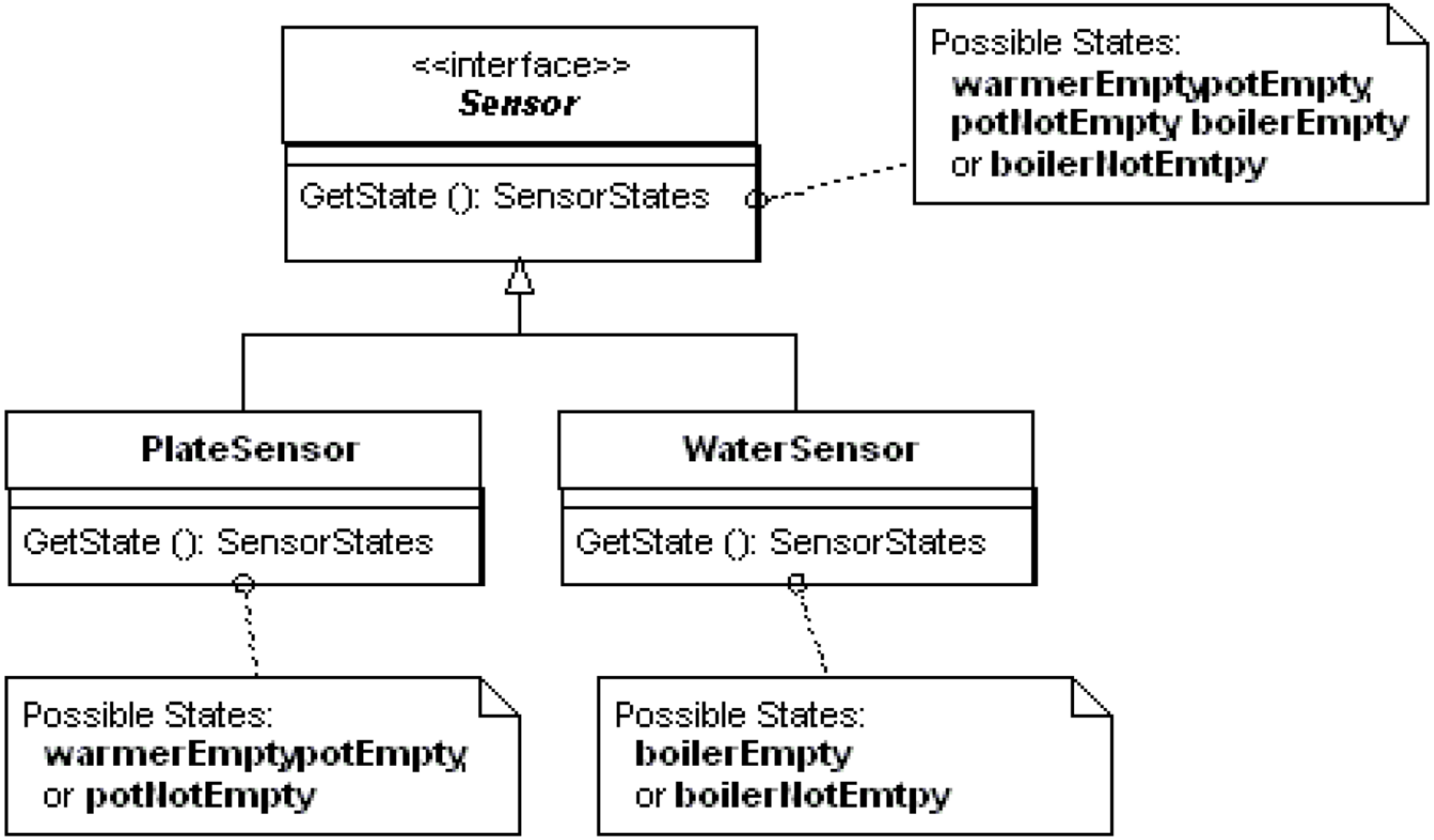
http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

Class Diagram – Input Devices



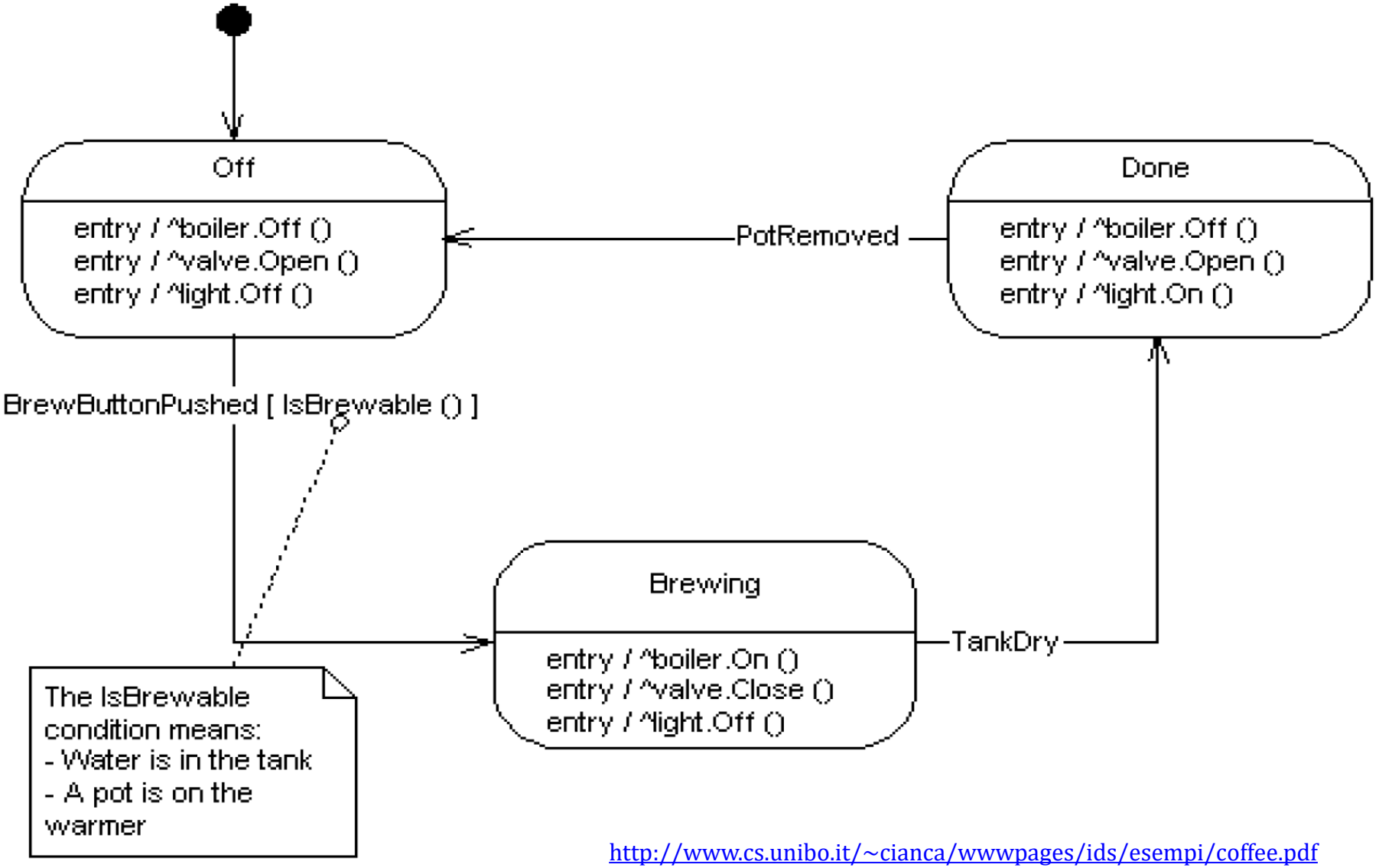
Combining Sensor Classes

- Combining different types of sensors is not a good idea



http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

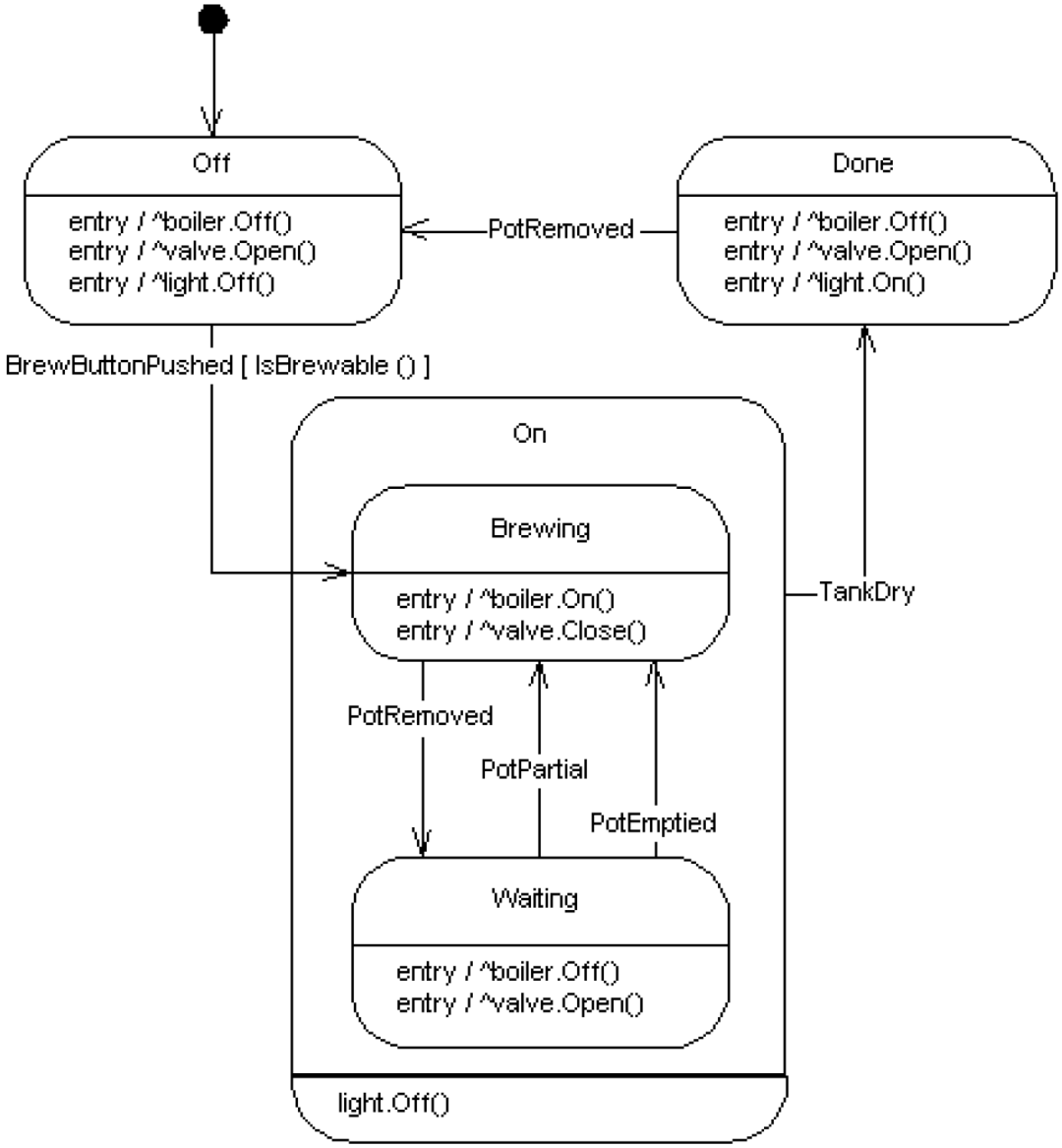
State Diagram – The Brew Controller



http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

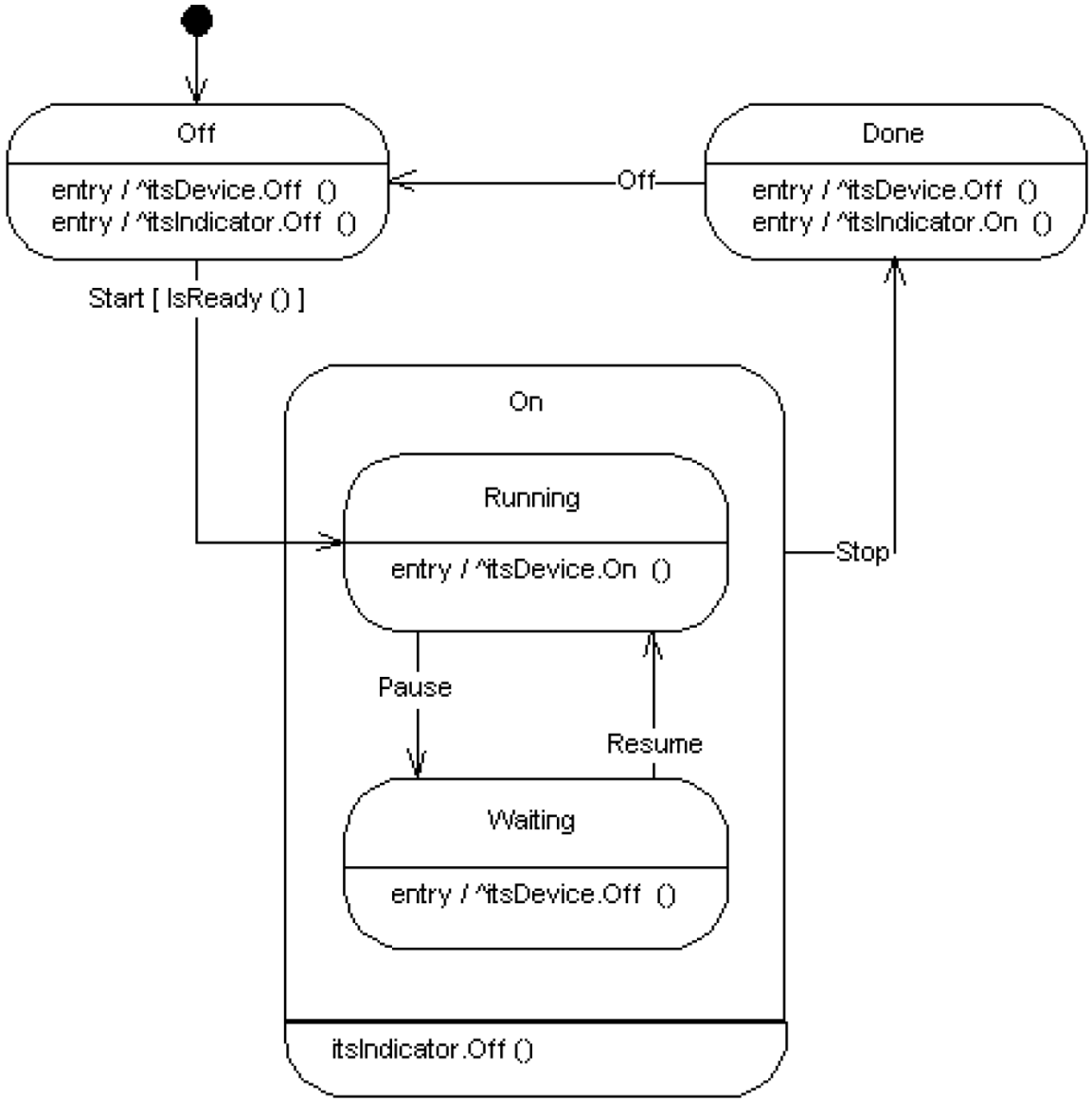
http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

The Improved Brew Controller



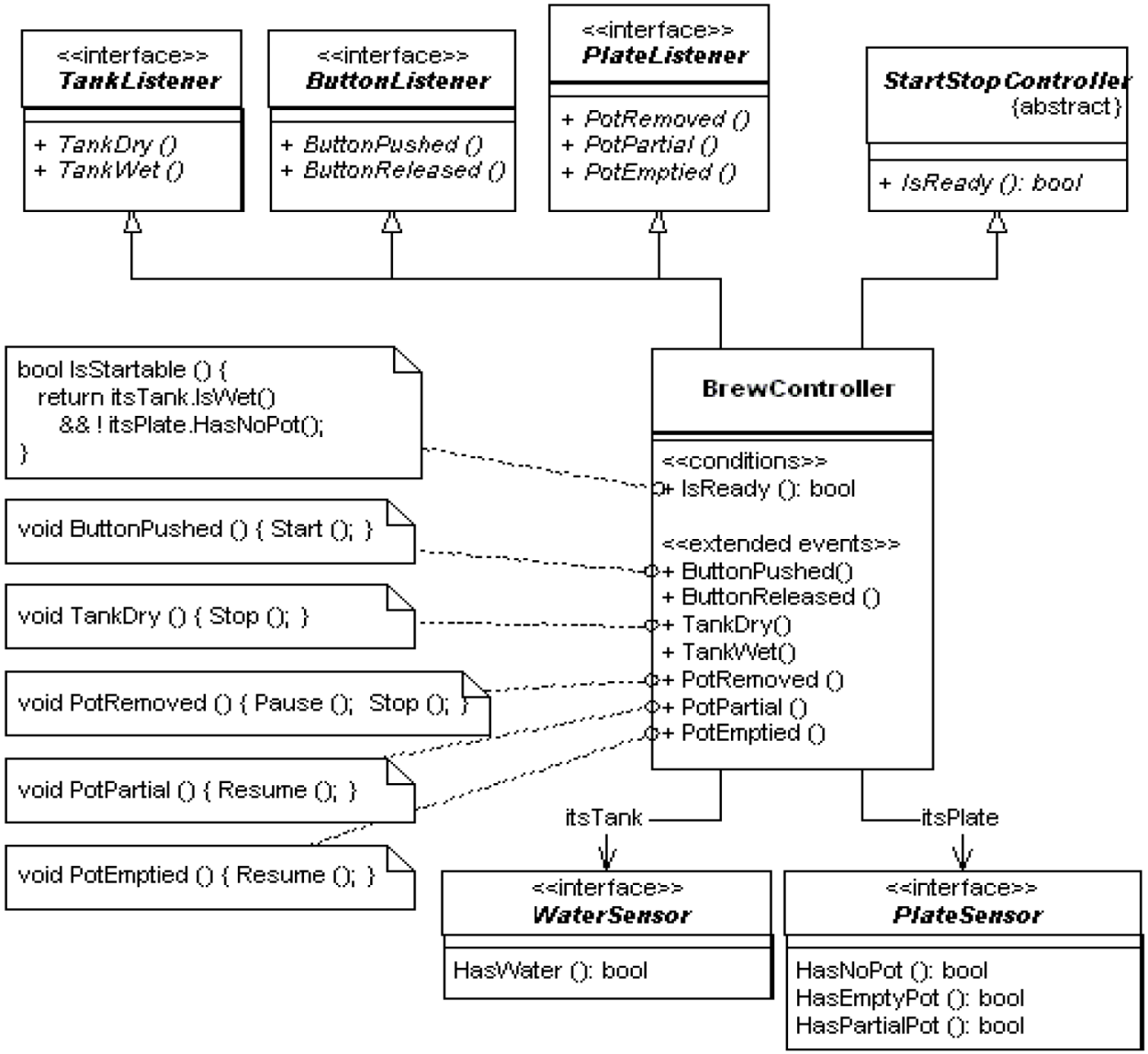
http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

A Generic Brew Controller

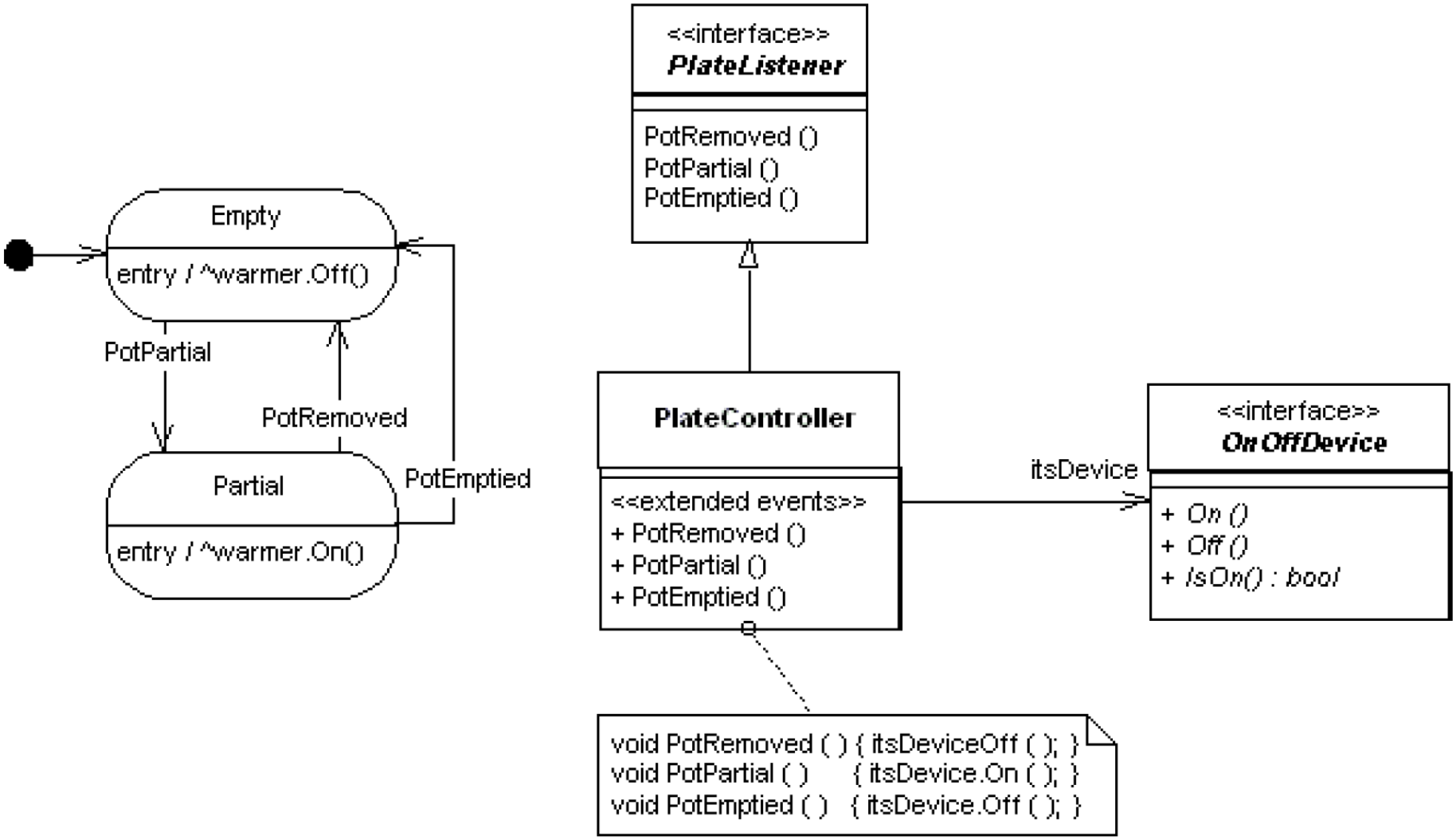


http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

Start/Stop Controller

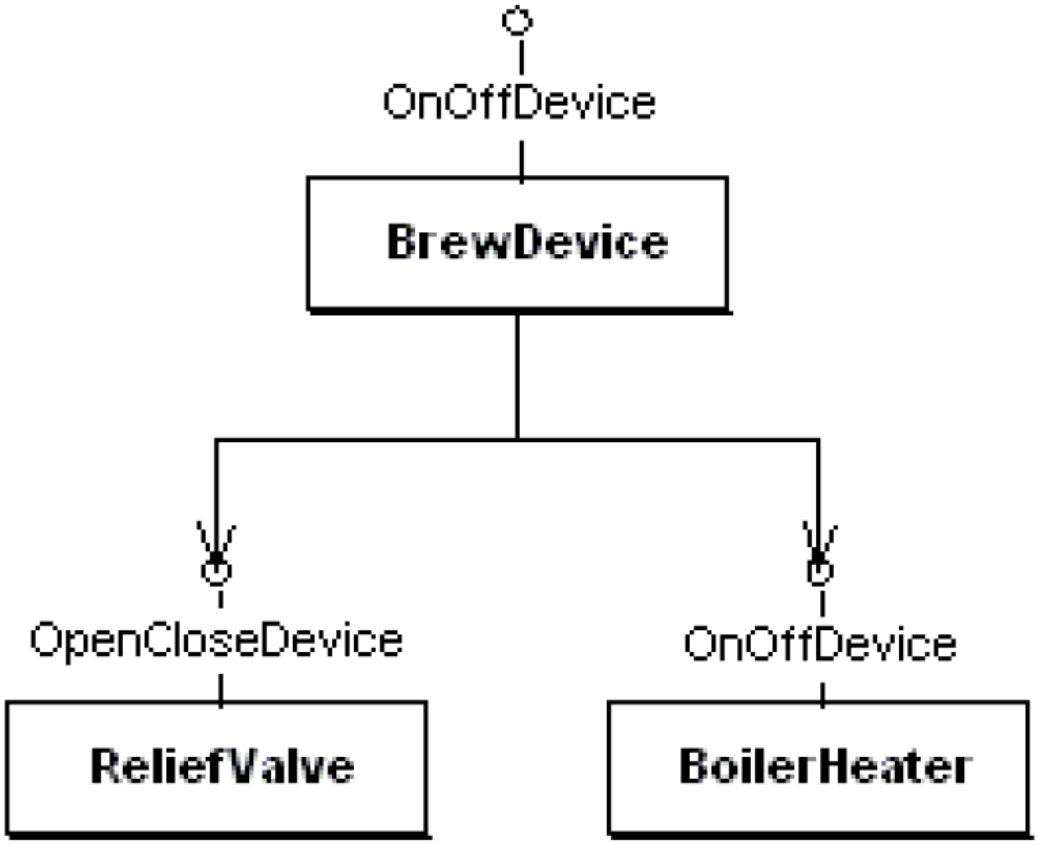


The Plate Controller



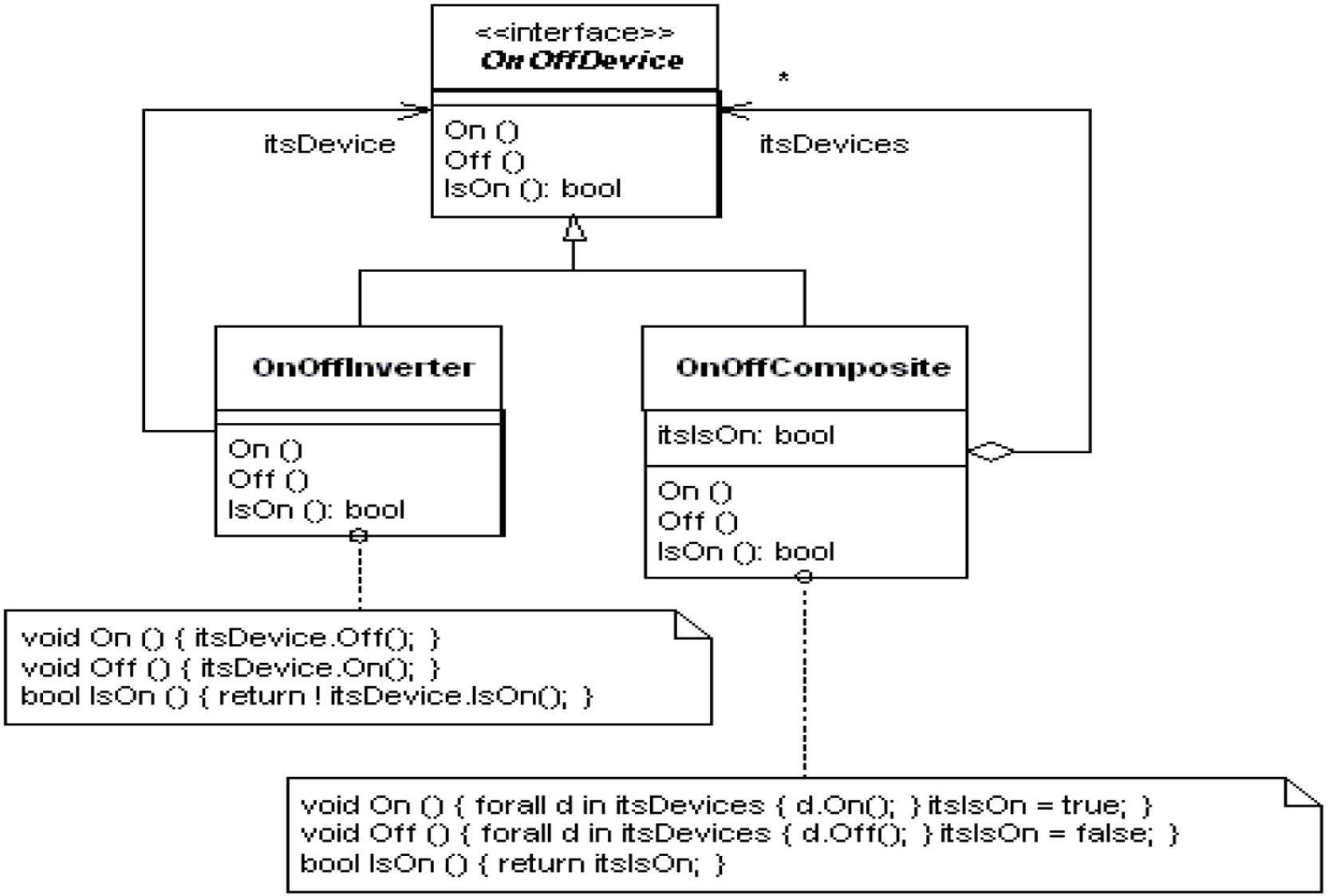
http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

The Brew Device



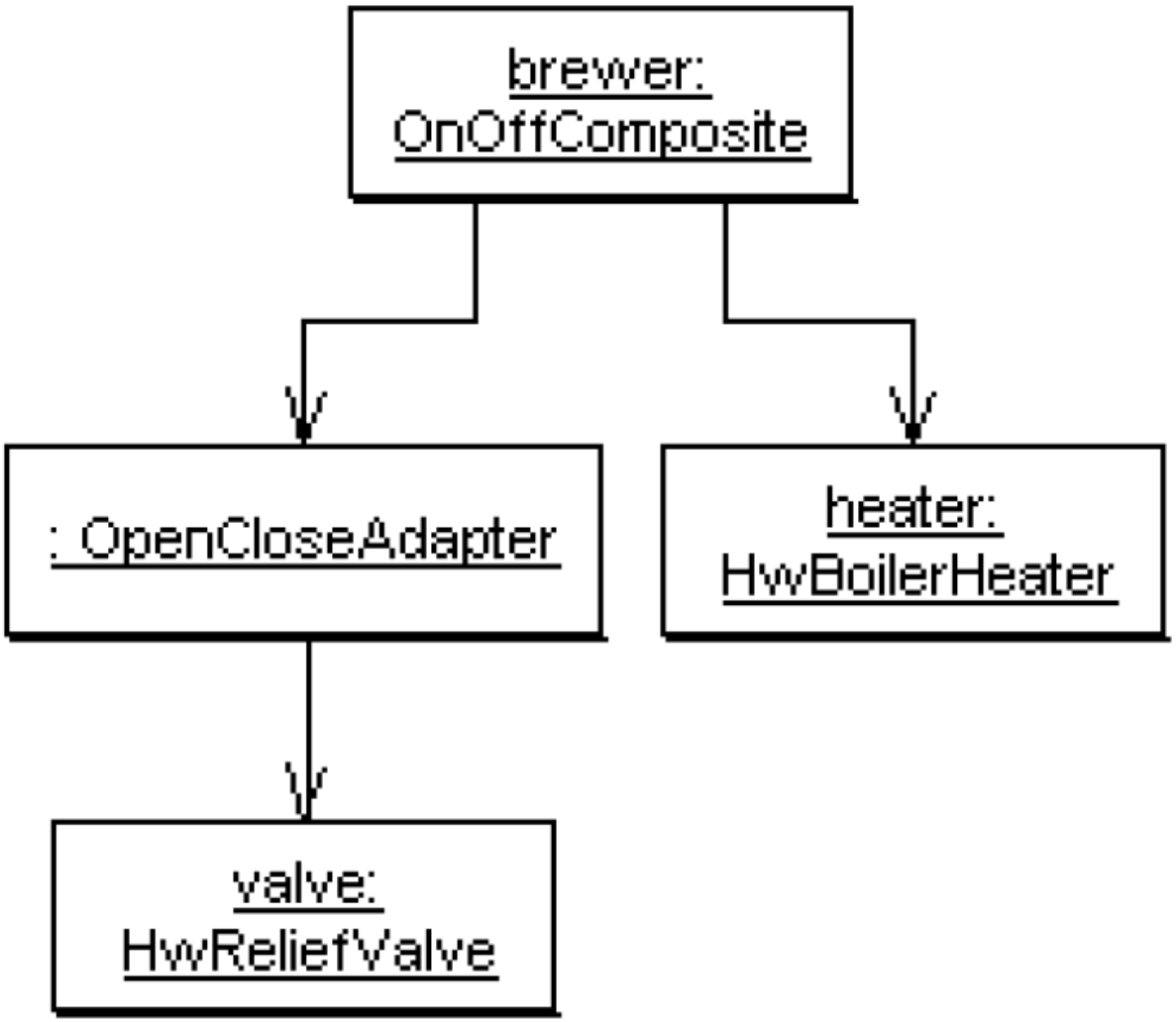
http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

On/Off Devices using Composite Pattern



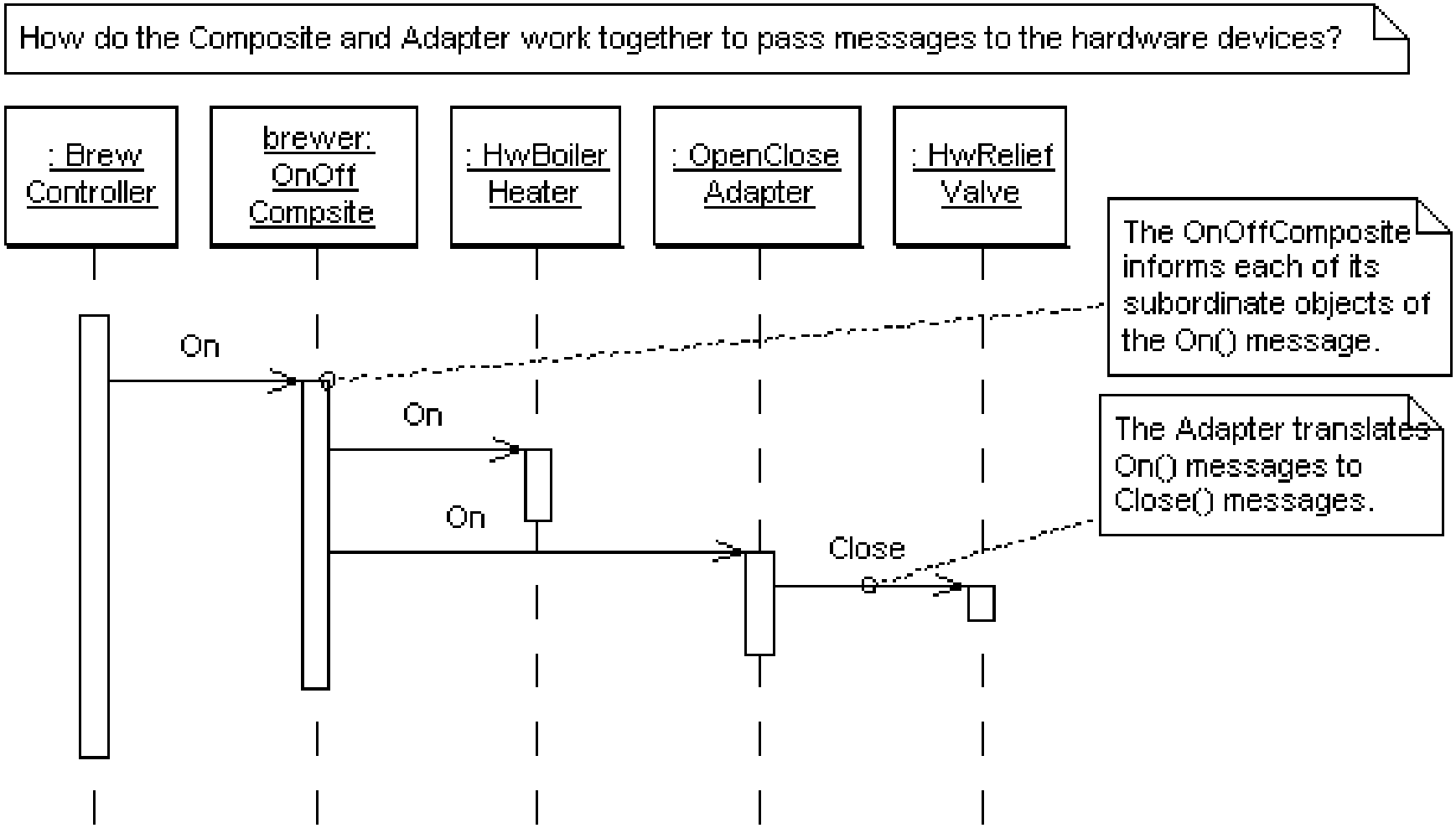
http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

Brew Device using Composite Pattern



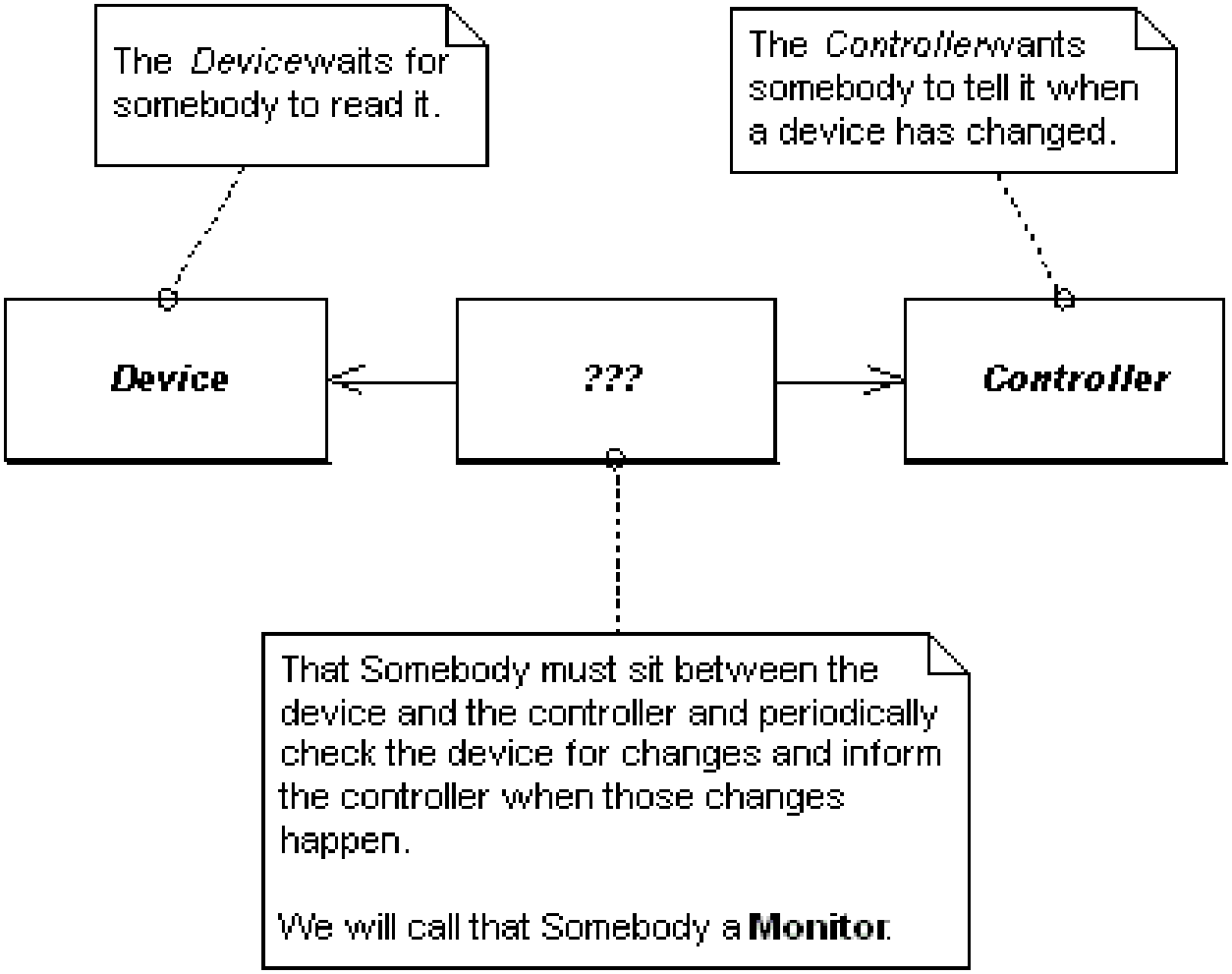
http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

Sequence Diagram



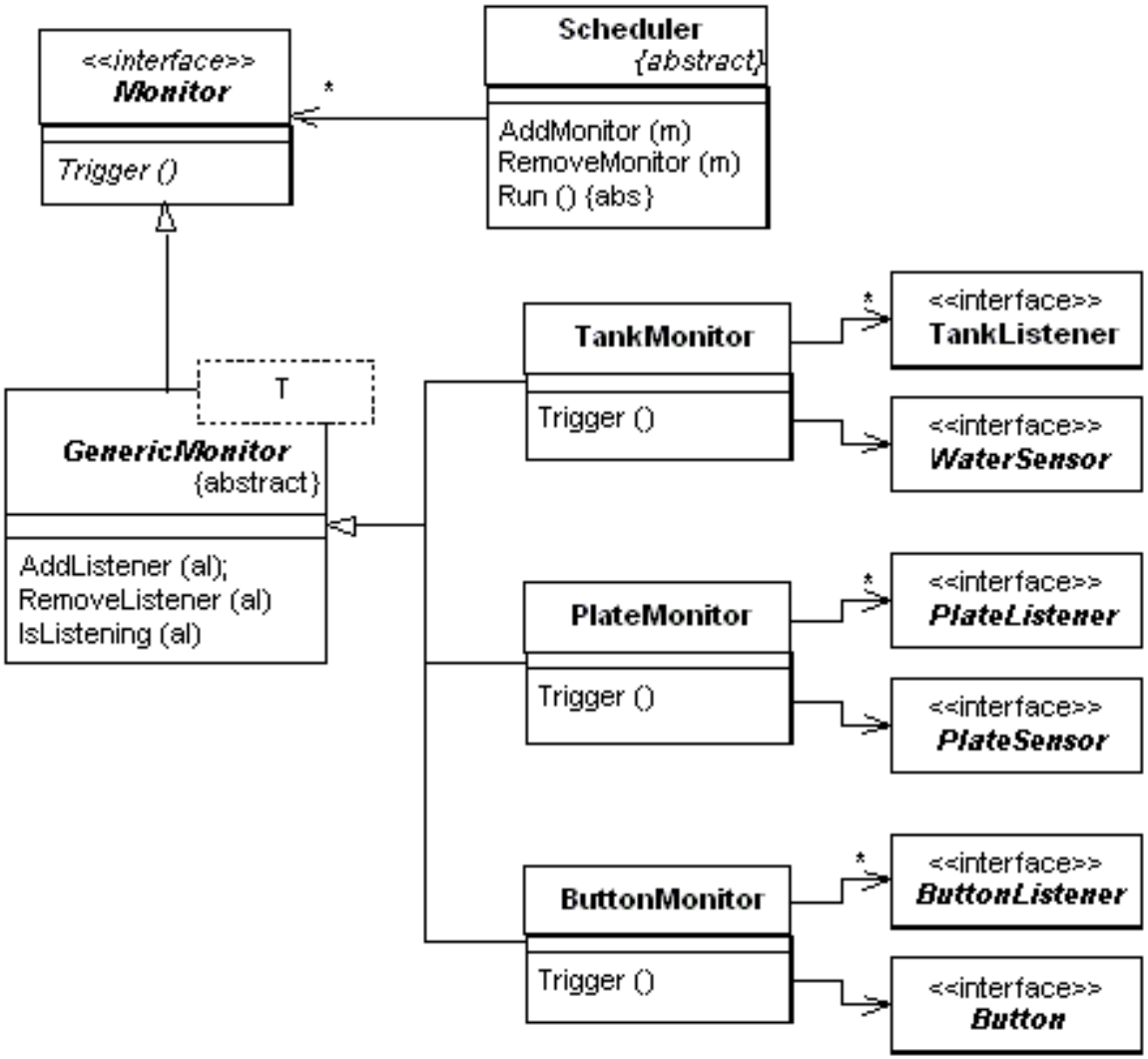
http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

Managing Events



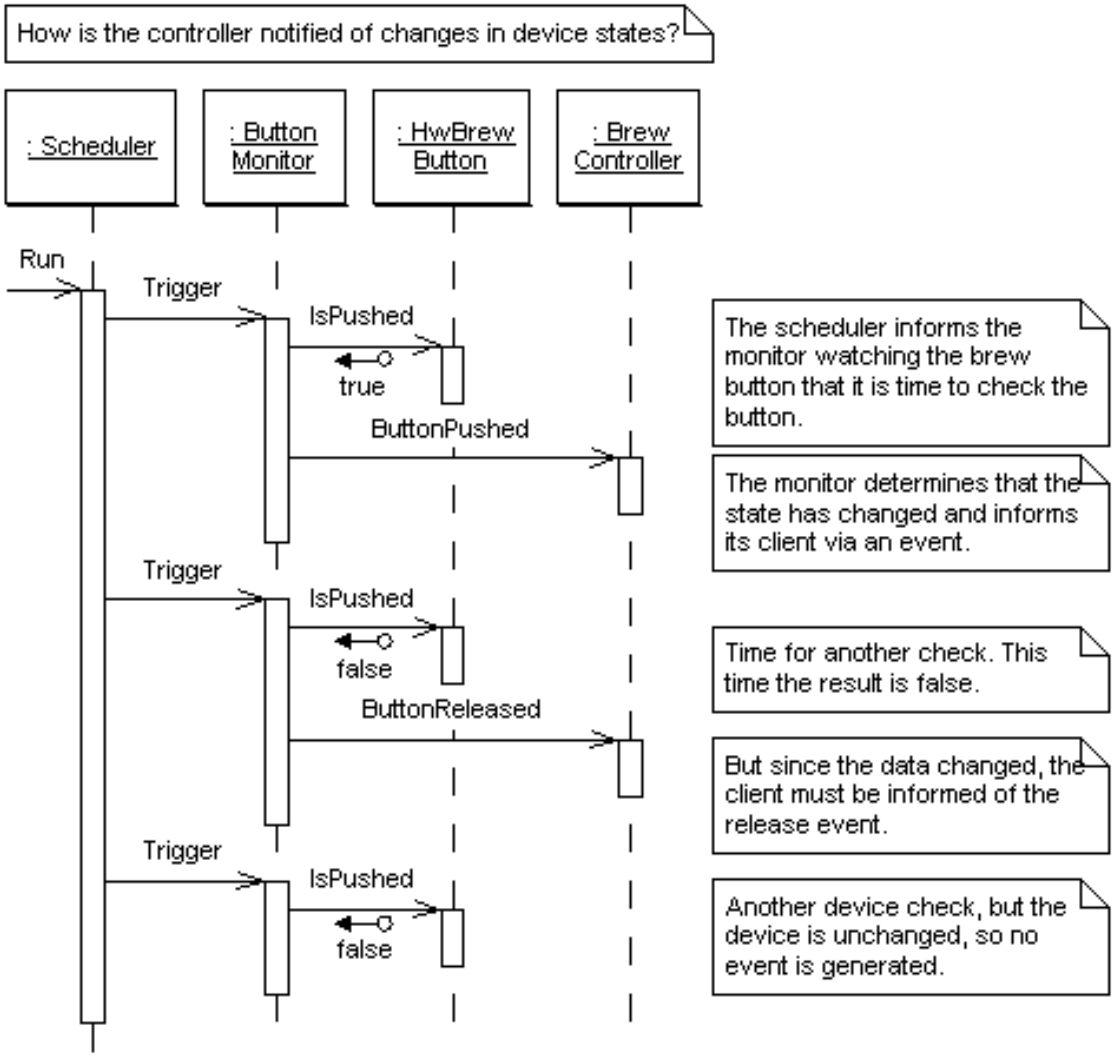
http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

Scheduler and Monitor Interaction



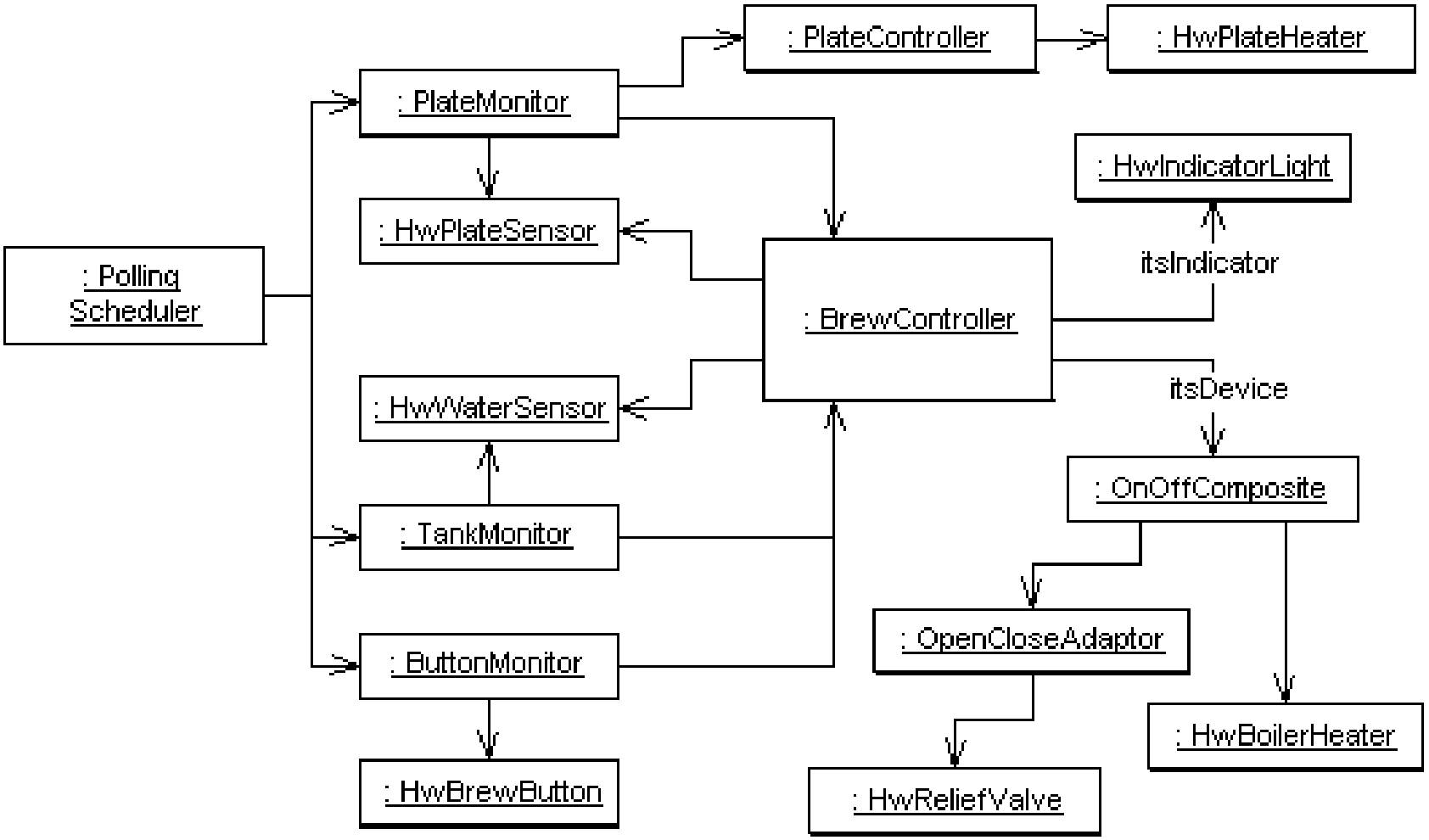
http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

Monitor Sequence Diagram



http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

System Object Diagram



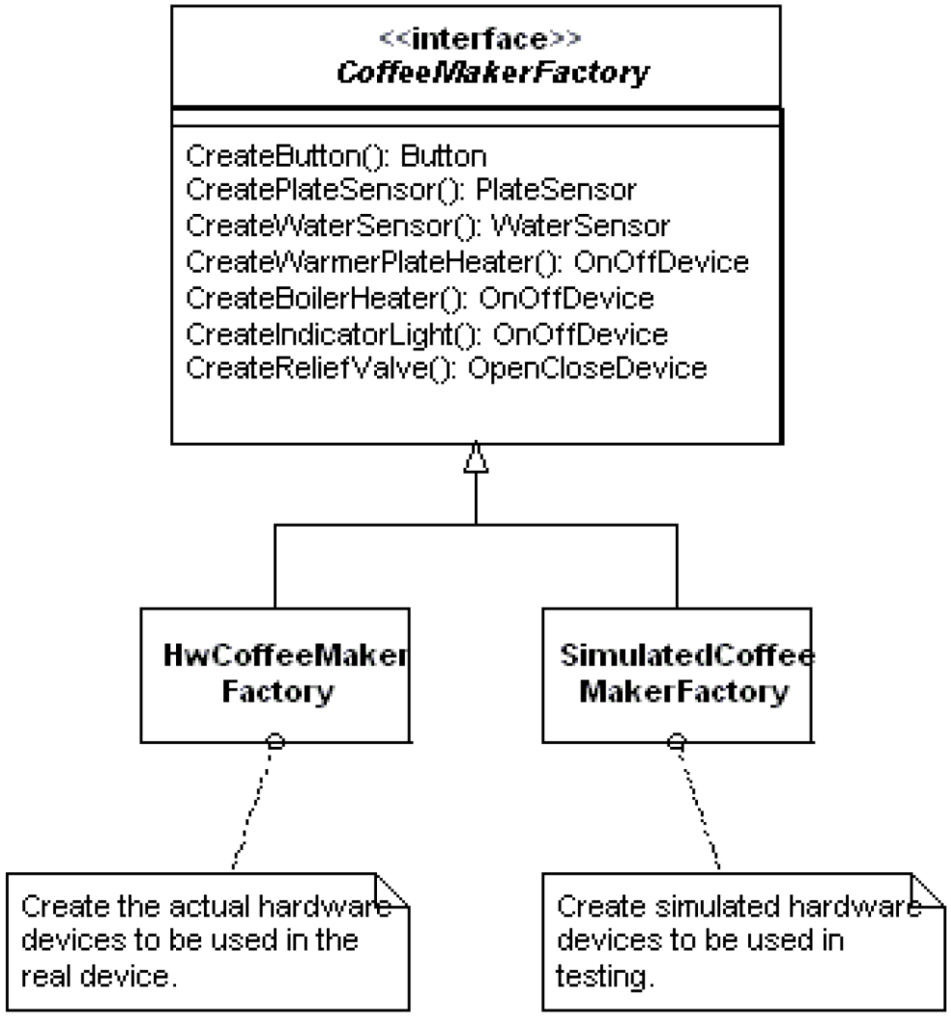
http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

Building the System

```
int main ()
{
    // Create the devices
    auto_ptr<Button> brewbutton (new HwBrewButton);
    auto_ptr<PlateSensor> plate (new HwPlateSensor);
    auto_ptr<WaterSensor> tank (new HwWaterSensor);
    auto_ptr<OnOffDevice> warmer (new HwPlateHeater);
    auto_ptr<OnOffDevice> light (new HwIndicatorLight);
    auto_ptr<OnOffDevice> heater (new HwBoilerHeater);
    auto_ptr<OpenCloseDevice> valve (new HwReliefValve);
    // Create controllers, adapters, etc., and establish
    // the required connections. Finally, kick start the
    // system by starting the scheduler.
    sch->Run();
    return 0;
}
```

http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

Coffee Maker Abstract Factory



http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

Coffee Maker Abstract Factory

```
int main (){
   // Create the factory
   auto_ptr<CoffeeMakerFactory> f (new
      SimulatedCoffeeMakerFactory);
   // Create the devices
   auto_ptr<Button> brewbutton (f->CreateButton());
   auto_ptr<PlateSensor> plate (f->CreatePlateSensor());
   auto_ptr<WaterSensor> tank (f->CreateWaterSensor());
   auto_ptr<OnOffDevice> warmer (f-
      >CreateWarmerHeater());
   auto_ptr<OnOffDevice> light (f-
      >CreateIndicatorLight());
   auto_ptr<OnOffDevice> heater (f-
>CreateBoilerHeater());
   auto_ptr<OpenCloseDevice> valve (f-
>CreateReliefValve());
   // ? and so on ?
}
```

http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf

Coffee Maker Abstract Factory

- We could even select between the two sets of hardware devices at run time.

```
// Choose which factory to use
auto_ptr<CoffeeMakerFactory> f;
if (doSimulatedRun)
{
    f.reset (new SimulatedCoffeeMakerFactory);
}
else
{
    f.reset (new HwCoffeeMakerFactory);
}
```

http://www.cs.unibo.it/~cianca/wwwpages/ids/esempi/coffee.pdf