Exercise 1: Software Architecture Description of the HS07 System

Anders H Poder, Jesper Dalberg, Lars Kringelbach

Department of Computer Science, University of Aarhus
Aabogade 34, 8200 Århus N, Denmark

Group 11 - Kilo
19951439, 20074976, 20074842
{ahp, jdalberg, u074842}@daimi.au.dk

2008-02-04 13:32

Abstract

The HS07 system implements a closed-loop control of the heating in a private home. It monitors thermometers in the home, and based on measurements HS07 adjusts radiators in the home. This report gives a software architecture description of an architectural prototype of the HS07 system. The techniques used for architectural description are taken from [Christensen et al., 2004].

1 Introduction

Figure 1 shows a schematic overview of HS07 in a home. The home may be accessed by the home owner from the outside through the HS07 gateway. The HS07 gateway also monitors and controls the home.

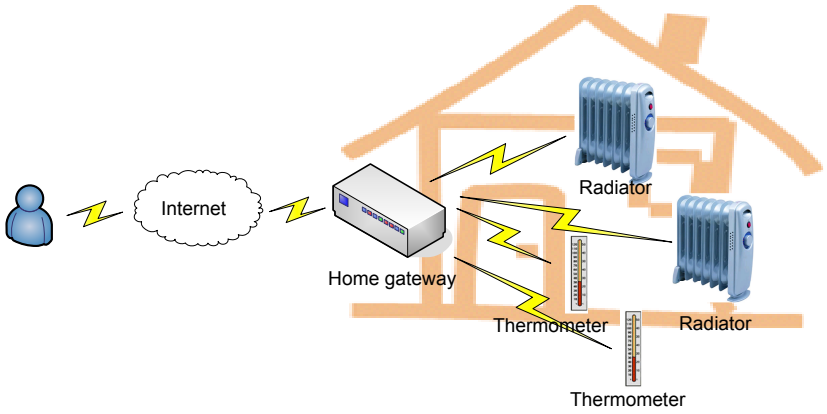

Figure 1: HS07 in a home

HS07 includes sensor and actuator hardware which runs on an embedded Java virtual machine with standard software.

2 Architectural Requirements

For our purposes there is one main use case for the HS07 system:

> *Control Temperature*: The gateway collects measurements from thermometers and reports this to radiators that then control the temperature.

The major driving qualities attributes of the HS07 system are[1]:

- *Performance.* HS07 should be performant so that a large number of thermometers and radiators may be part of the system.
- *Modifiability.* It must be possible to modify HS07 to include new types of sensors and actuators.
- <<Extra quality requirement that you consider important >>

3 Architectural Description

3.1 Module Viewpoint

The module viewpoint of the HS07 system.

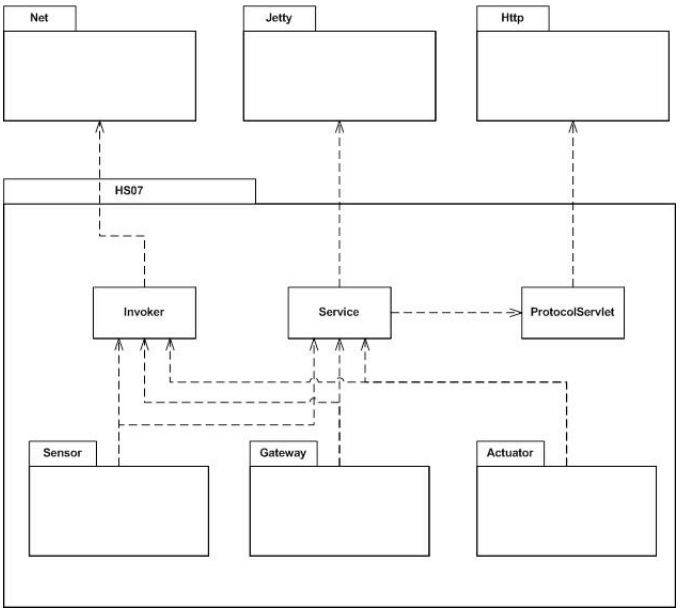

Figure 2: HS07 Package Diagram

The figure shows the packages of the HS07 system, and how they interact with java packages used.

[1] These qualities will be operationalized in Exercise 2

3.2 Component & Connector Viewpoint

The Component & Connecter view consists of an Active Objects diagram and a sequence diagram. The Active Objects diagram shows the active objects of the HS07 system, and how they interact.

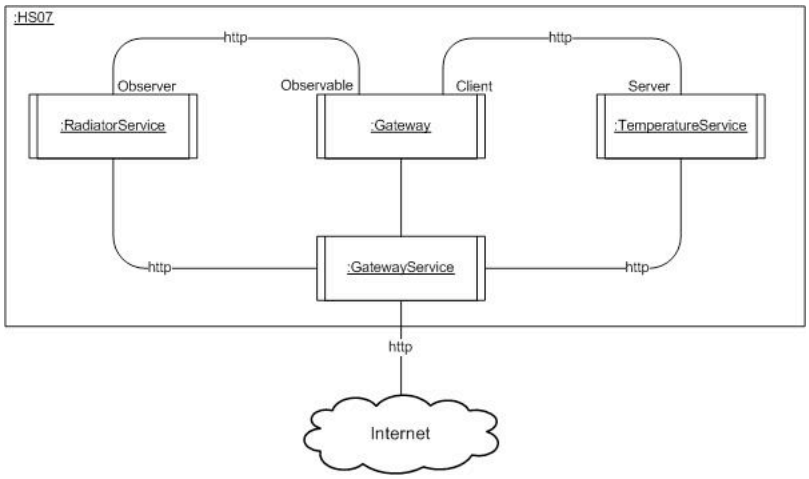

Figure 3: HS07 Active Objects

The sequence diagram illustrates the timing of the interactions.

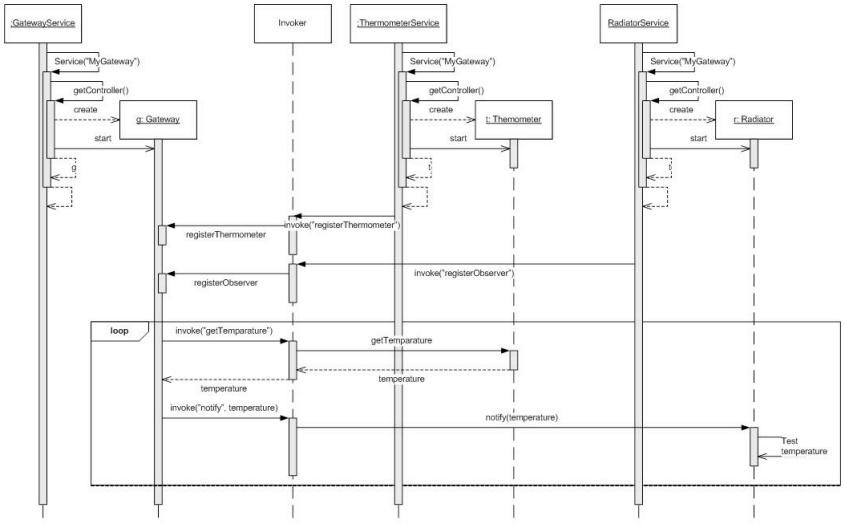

Figure 4: HS07 Sequence Diagram

3.3 Allocation Viewpoint

The allocation viewpoint illustrates how components are deployed im actaual processes within the HS07 system.

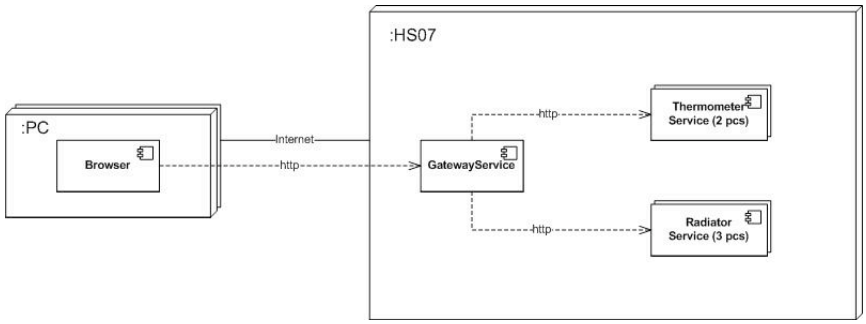

Figure 5: HS07 Allocation Diagram

4 Discussion

<<What are the strengths and limitations of this approach to architectural description judging from this case?>>

<<Are there aspects of the software architecture that have not been properly described?>>

<<...>>

5 ...

References

[Christensen et al., 2004] Christensen, H., Corry, A., and Hansen, K. (2004). An approach to software architecture description using UML. Technical report, Computer Science Department, University of Aarhus.