

Pervasive Monitoring of Built Environments

Chris Conway, Graham Booth, and
Will Cooke

Overview

- Problem
- System Design
- Demo
- Conclusion

Overview: Problem

- Brief: Design and implement a secure, reliable, and scalable pervasive monitoring system for use in the Informatics Forum
- The system should utilise a heterogeneous Wireless Sensor Network (WSN)
- The team should identify the WSN requirements of the application (low-power utilisation)
- The team should identify the needs of the system's target user (providing salient environmental data in a clear and informative way)

Overview: Problem

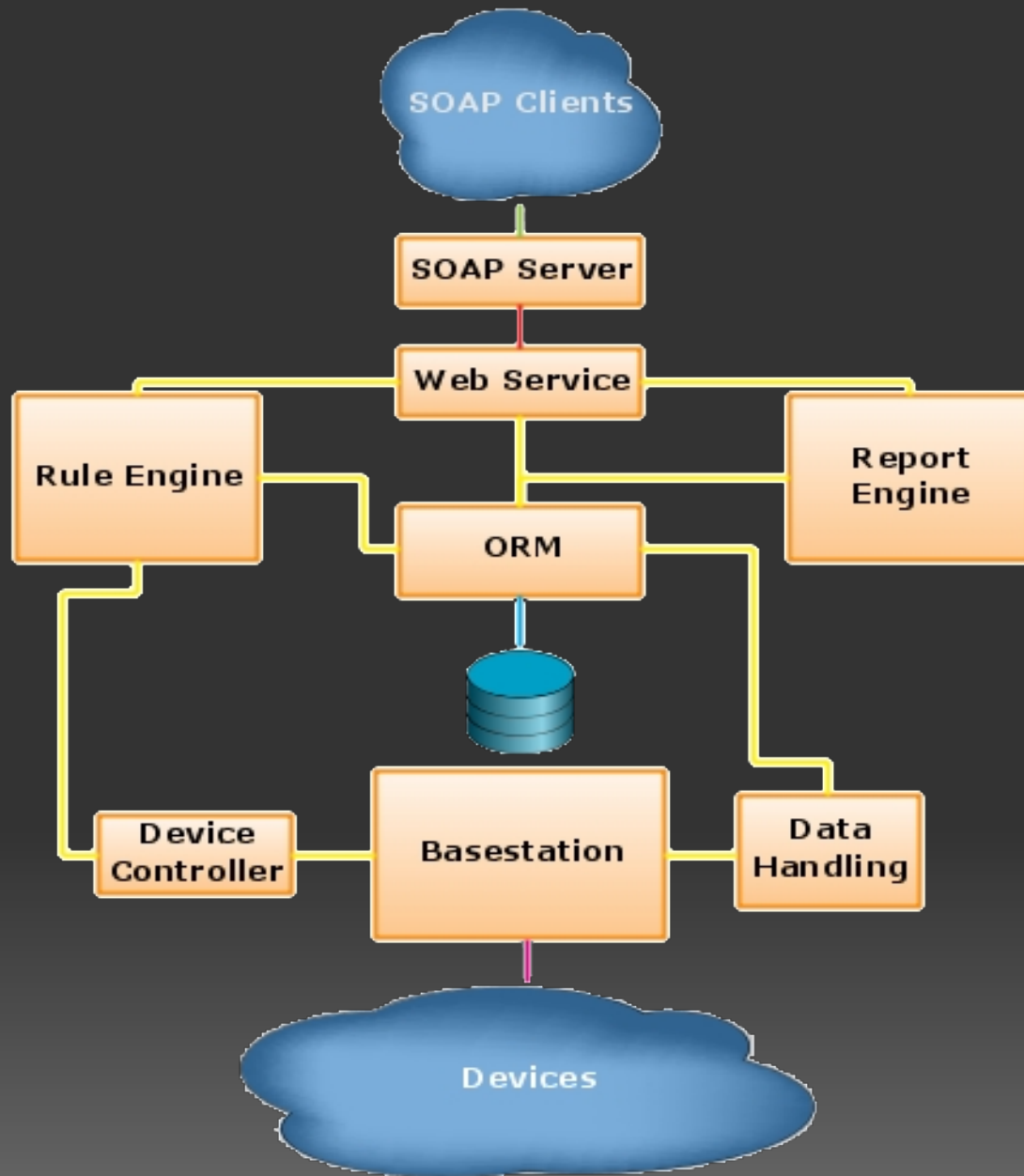
- We approached Shona Buchanan, the Energy and Sustainability Manager for The University of Edinburgh
- The feedback we received suggested that the system would be well suited to assisting Utility Managers in their day-to-day activities
- For example the system could be used to identify the rooms in the building which do not fall within health and safety temperature regulations

Overview: Problem

- The system itself should consist of spatially distributed devices that use sensors to cooperatively monitor environmental conditions, such as temperature, light, or humidity at different locations.
- The system should be able to interact with a building automation system, and so control heating and lighting within the building itself.

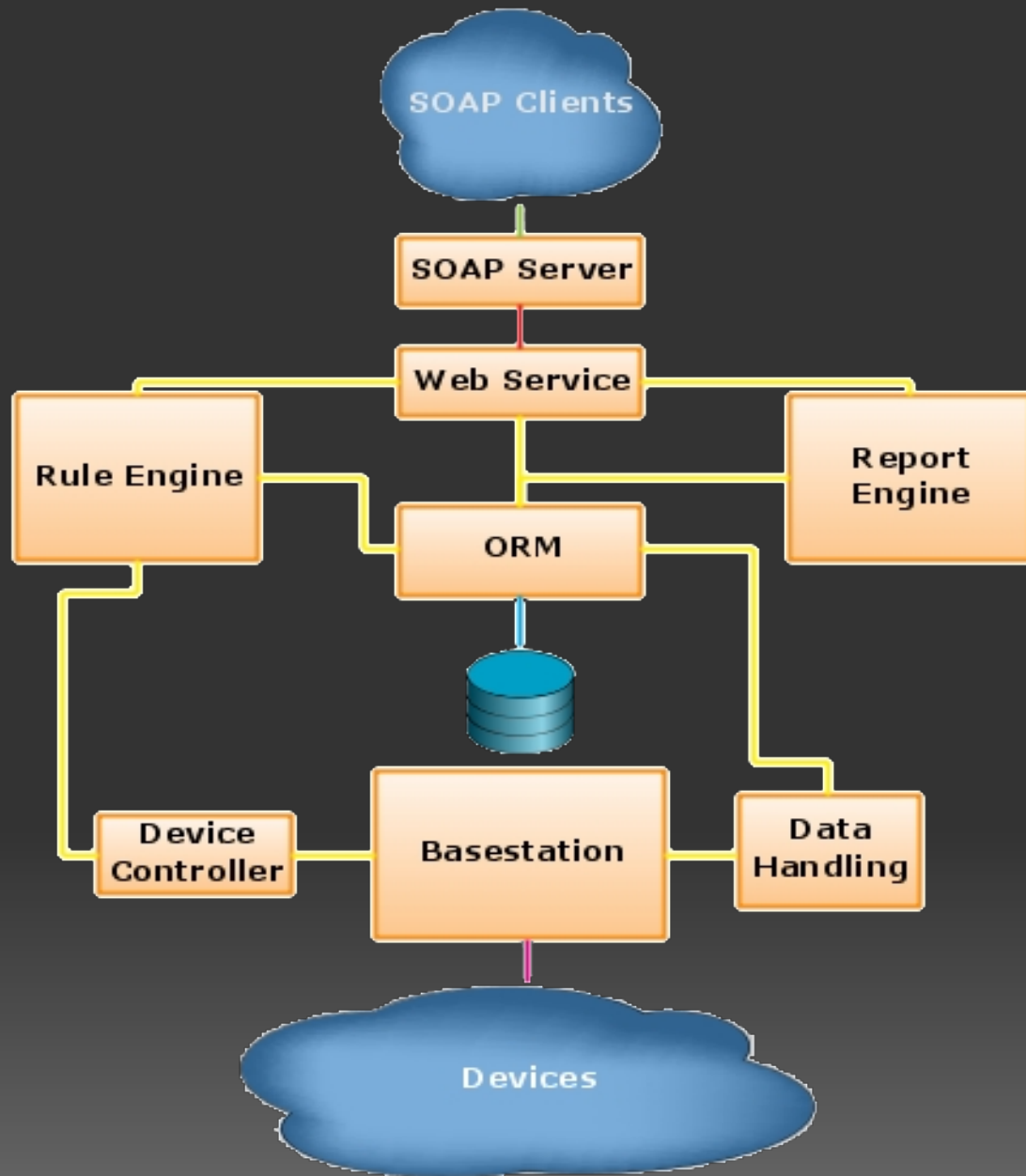


Overview: System Design



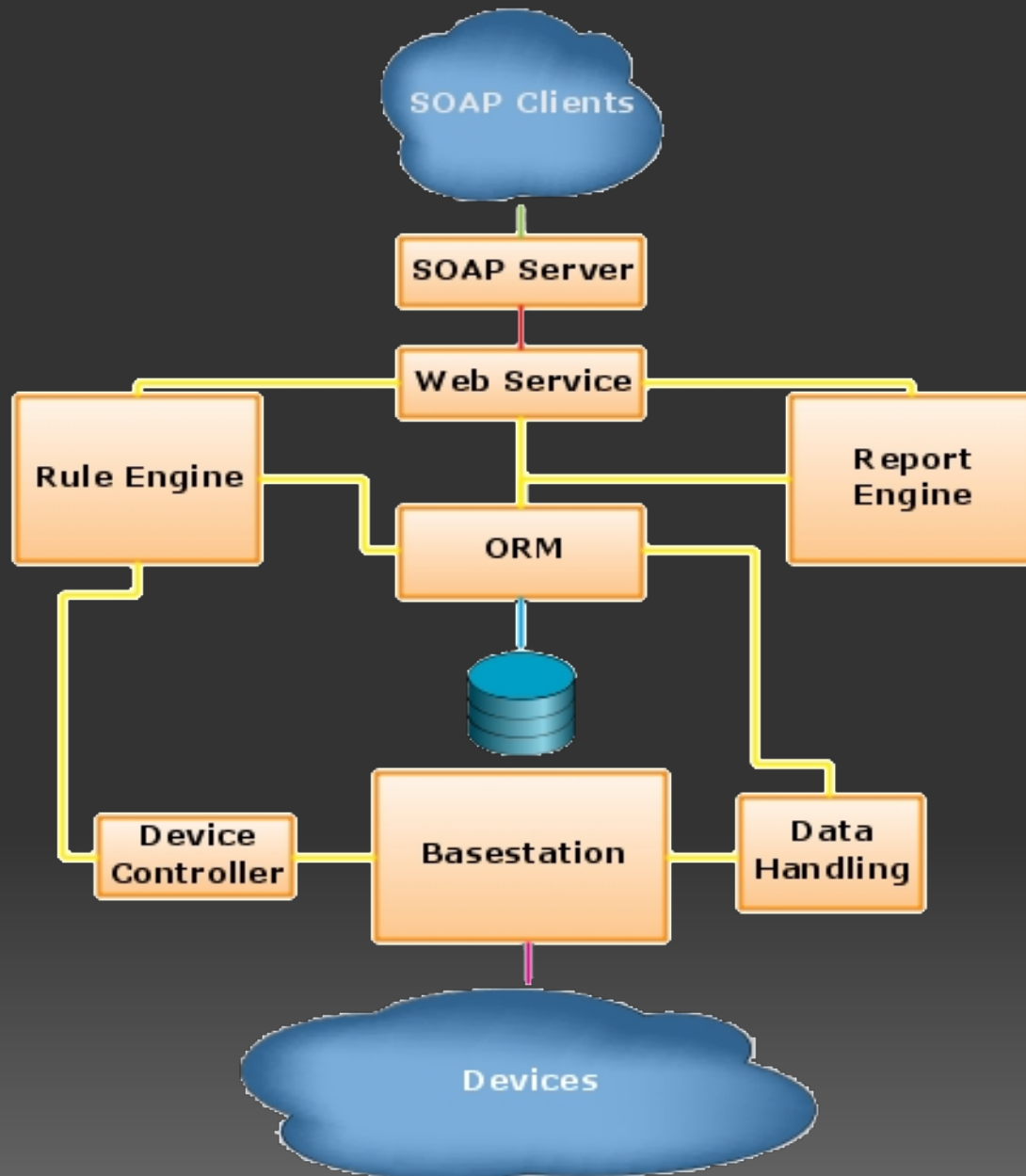
- Devices, talk over radio to:
- Basestation, which passes messages to
- Data Handling, which acts on them and stores them in the database
- The Report Engine reads from the database and produce interesting things which can then be passed to.
- the Web Service, allows access from outside clients over SOAP.
- The Rule Engine performs analyses the data and uses
- the Device Controller to take action to correct the state of the system.

Overview: System Design



- Componentised
 - all internal systems can run on separate machines using RMI.
 - can switch out separate systems easily.
 - clearly defined interfaces between each section shows clear boundary between components and data.
 - better performance - can run on multiple machines / completely distributed.

Overview: System Design



- Secure
 - internal system can be run on completely separate network from external API.
 - internal system can require authentication at each component
 - links between each system can be encrypted if desired.

Java Client Demo

Conclusion

- Pervasive Monitoring
- Reliability
 - Operational Independence
- Scalability
 - Distributed Nature
- Security
 - Signed Packets
 - Error Recovery

Questions?