From IoT Devices to Smart Objects

Marco Perez Hernandez

15 November, 2017
Outline

Context
Popular Approach for IoT Apps
Proposed approach
Case Study
Open challenges
Context

Gateway

SOA
Web services (REST, RPC-b)

Data

Mobile Apps
User control (manual)

Embedded programming
Sensors / Actuators
Low resources
Popular approach
IoT Devices as data gatherers

- Data storage, processing & control
- Service discovery, control
- Service Implementation
- Raw data gathering

Concentration:
Data, functionalities

Sub utilisation of device resources

Real autonomy?
Popular approach
IoT based on Smart Objects

- Data storage, processing & control
- Service discovery, control
- Service Implementation

- Data gathering

- Data storage, processing
- Capability Selection & Discovery
- Cooperation & Autonomy
- Service Implementation
Proposed solution
Role-based architecture

- Goal-directed agent
- Capabilities as service components
- Applications: realisation of abstract workflows based on services
- Roles: group of functionalities/responsibilities
Proposed Solution
Role-based architecture
Case Study: Home automation (I)

- Smart Air Freshener
- Resource Manager
- Room Watcher
- Home Controller

Sources:
1. src: http://photo-dictionary.com
2. src: www.noushouse.com.au
3. src: http://officestor.co.uk
Case Study: Home automation (II)

Prototype Implementation

- Java
- eve almende
  - Jetty 7.x
  - JSON-RPC over HTTP
- CouchDB:
  - Document oriented
  - JSON Documents

src: http://eve.almende.com/index.html

De Jong J., Stellingwerff L. et Al (2013)
Case Study: Home automation (III)

Role Definition

- **Role: AirFreshener**
  - `id`: "activity/keepAirFresh",
  - `type`: "activity",
  - `categories`: []
  - `input`: {}
  - `knowledge`: [{}]
    - `scope`: "saf_agent",
    - `name`: "freshener",
    - `kind`: "resource",
    - `attrNames`: ["model"]
  - `operator": "AND",
  - `operands`: [{}]
    - `operator": ">=",
    - `operand1`: {
      - `scope": "saf_agent",
      - `name": "freshener",
      - `kind": "resource",
      - `attributeName": "level",
      - `value": "10"
    }
    - `operator": ">="
    - `operand1`: {
      - `scope": "room",
      - `name": "activity",
      - `kind": "feature",
      - `attributeName": "level"
    }

- **Role: suppliesManager**
  - `description": "actions related to keep the air fresh",
  - `actions": [{}]
    - `scope": "saf_agent",
    - `name": "freshener",
    - `kind": "resource",
    - `effect": "decrease"
  - `output": [{}]
    - `scope": "room",
    - `name": "freshness",
    - `kind": "feature",
    - `effect": "increase",
    - `attrNames": ["level"]
Agent-based Simulation

- Simulation of an IoT System with up to 140 SO
- Every agent an SO where Middleware is installed
- Sos communicate and coordinate through SO protocol
Open challenges

- Machine Learning for decision-making for support deciding between on-object/off-object services
- Role-based SO functionality as part of unikernels
- Hierarchical P2P protocol for role-based coordination
- Agent-based simulation of concrete scenarios
- Performance evaluation of SO-based solutions
Any question?