Heterogeneous Device Management with Eclipse OM2M based on oneM2M abstraction layer

Towards a Unified IoT Device Management Federative Platform

Eclipse IoT Day Grenoble 2020
Sébastien BOLLE / Cyrille BAREAU
Orange
IoT Research Domain
Device Management matters!

Economical

Maintenance of device costs: a call to Customer Care Call center costs 20 € - Sending a technician costs 100 €

Environmental

In the past, devices were considered as disposable - IoT becomes more environment respectful

Business

Openness to Third parties / partnership

Devices with advanced configuration capabilities

Not dedicated to a single usage, to fit technical configuration to functional requirements
What is Device Management?
A little piece of history

Gateways, STB firmware management

Assistance (Customer Care)

More stories to tell with IoT

Service configuration (e.g. VoIP)

QoS and QoE Tracking

History of Device Management
What is Device Management?
Device Management in industry

Very few research academic work on Device Management
   Even if the issues are complex in IoT context

Some standardisation initiatives:
   - Broadband Forum: success of TR-069 – new USP proposal
   - CableLabs: DOCSIS
   - OMA: OMA-DM and OMA-LWM2M
   - oneM2M
   - IETF: COMI, SUIT

A trend towards a fusion of Device and Service Management

And, obviously, many, many, many dedicated specific solutions…

“One Service – One Platform” syndrom
The reality with IoT? Silos and dedicated solutions

With already well known issues

- **Heterogeneity** (protocols, datamodels, lifecycles)
- **Scaling** (e.g. number of devices)

New challenges and issues brought by IoT Device Management

- **Static Dependencies** (firmware or configuration)
- **Dynamic Dependencies** between devices
- **Provisioning of services on multi-devices**
Our convictions
Federative and dynamicity

Facts
- Billions of devices
- Multiplicity and heterogeneity of protocols
- Multiplicity and heterogeneity of DM platforms

Target
- **Federative platform** to integrate multiple and heterogeneous Device Management Platform
- **Dynamic integration** of new management systems
- **Context-aware** management
- **Advanced smarter** (e.g. IA based) coordination features

From a **centralized paradigm**

To a **federated one**
Ongoing work
A Generic API for Device Management

Managing heterogeneous systems of devices

Protocol-agnostic Device Management API

Internal DM Server API
Single device focused

Open External Federative DM Server API
Wider scope including massive operations

Abstracting DM with oneM2M and SDT
Make it a standard API

Demo
oneM2M Smart Device Template

Brings an uniform description (abstraction) of connected objects, independently of underlying IoT technologies

Simplify developer work: one API for N technologies

Example: oneM2M ‘deviceLight’ device, its services, datapoints, and actions

Connect numerous light bulbs from distinct providers and technologies

Play with lights: switch light on/off, tune brightness & colour saturation, ...

Application to Device Management

Manage the lights: reboot, read management status, update software, ...

Light device modelling using SDT
Remote software management through oneM2M

OM2M software slot 1
EMPTY

OM2M software slot 2
EMPTY

install

OM2M software slot 1
LED strip (installed)

OM2M software slot 2
EMPTY

activate

OM2M software slot 1
LED strip (active)

OM2M software slot 2
EMPTY

disable
Conclusion - Ongoing work
The Road to a Smarter Device Management

Semantic Model of DM

Dynamic Coordination Dependencies Management

AI for DM: Autonomic, Machine Learning

Delegation in Edge Computing nodes
Thank you