Introduction to OneM2M

Suraj and Lavanya
3rd April, 2019
IIIT-H

Today's topics

1. Case study : Existing IoT Solutions
2. Introduction to OneM2M
3. Why OneM2M
4. Ontology
5. Using OneM2M
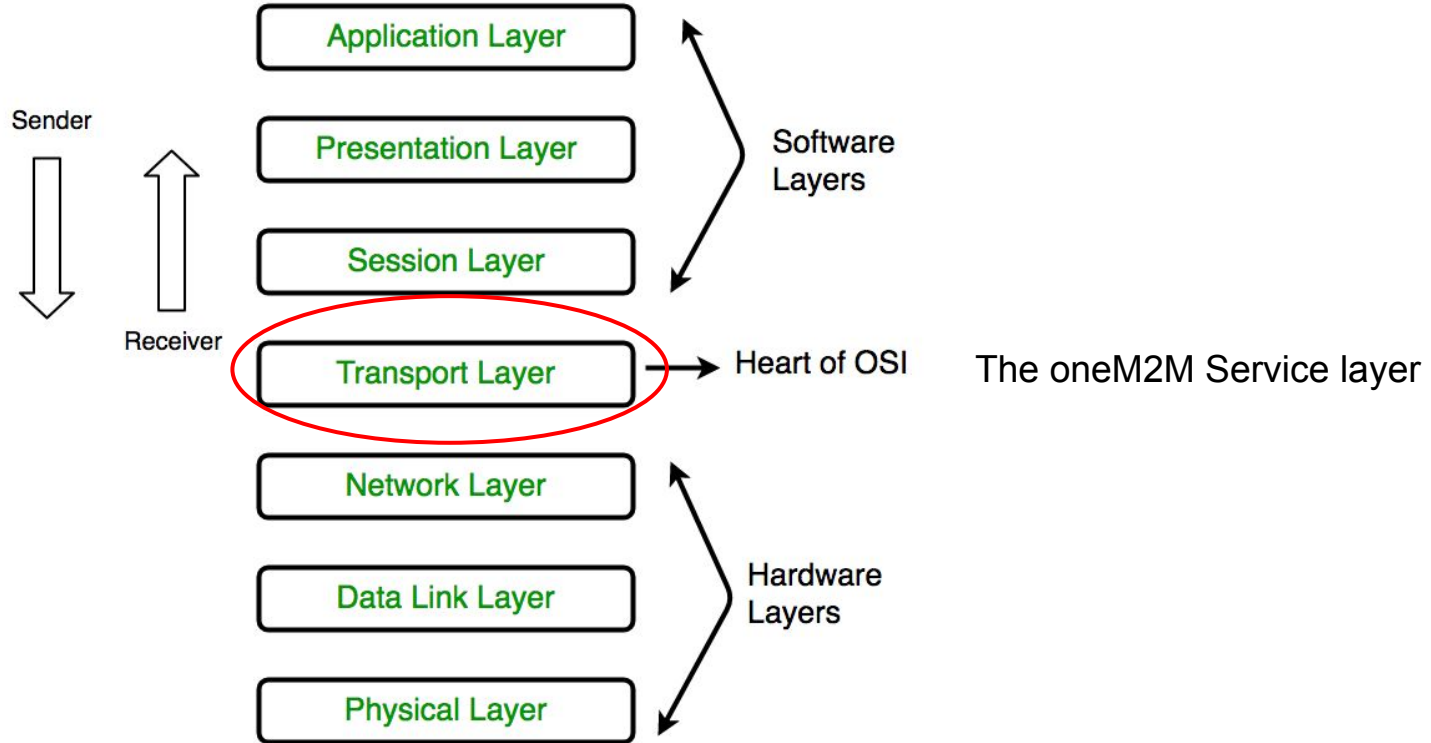6. Intro to the upcoming lab experiment

A simple case

- Highly fragmented market with limited vendor-specific applications
- Re-inventing the wheel : Same services developed again and again
- Each silo contains its own technologies without interoperability

Source : http://www.onem2m.org/images/app_dev_guide/tr-0057/sillos.png

The challenges

1. Interoperability
2. Scalability
3. Device management

Network Layer

Sender

Receiver

	Software Layers
Application Layer	
Presentation Layer	
Session Layer	
Transport Layer	Heart of OSI
Network Layer	
Data Link Layer	Hardware Layers
Physical Layer	

The oneM2M Service layer

https://cdncontribute.geeksforgeeks.org/wp-content/uploads/computer-network-osi-model-layers.png

What is oneM2M

- Global standardization for M2M and IoT
  - Not across all the layers of the networking stack
- Provides a software framework by creating a **horizontal** layer across domains.
- This service layer is located between applications and hardware infrastructure
- Enables reusability
- Members consist of various standard bodies, ICTs and companies.
- Work was initiated in 2008

What oneM2M IS NOT

- Standardize interfaces, not entire environment across networks
- Device interoperable but is not network agnostic
    - Only IP (Internet Protocol) is supported

"ANY APP"

"ANY NETWORK"

"ANY DEVICE"
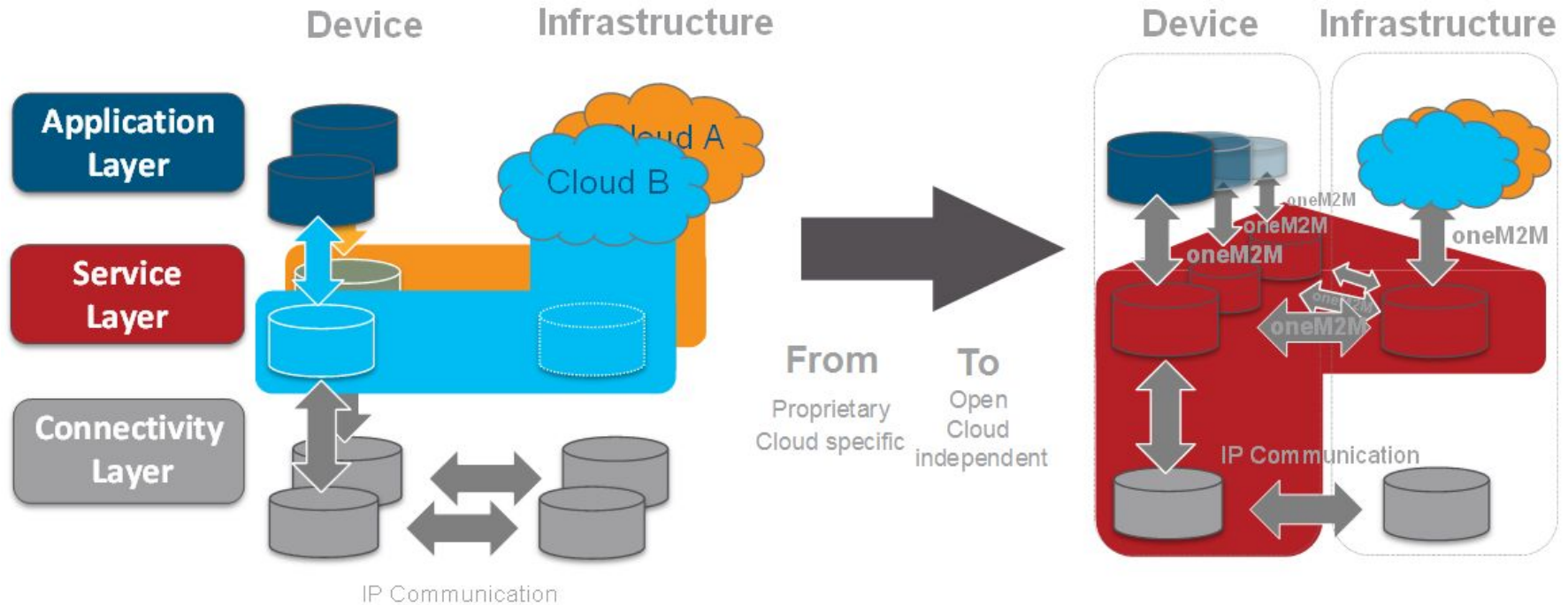
Need for standardization

- To provide scalability and flexibility
- Improves functionality-cost-quality trade off
- Set of APIs communicating with the service layer reduces:
    - Time-to-market
    - Development and on-boarding costs
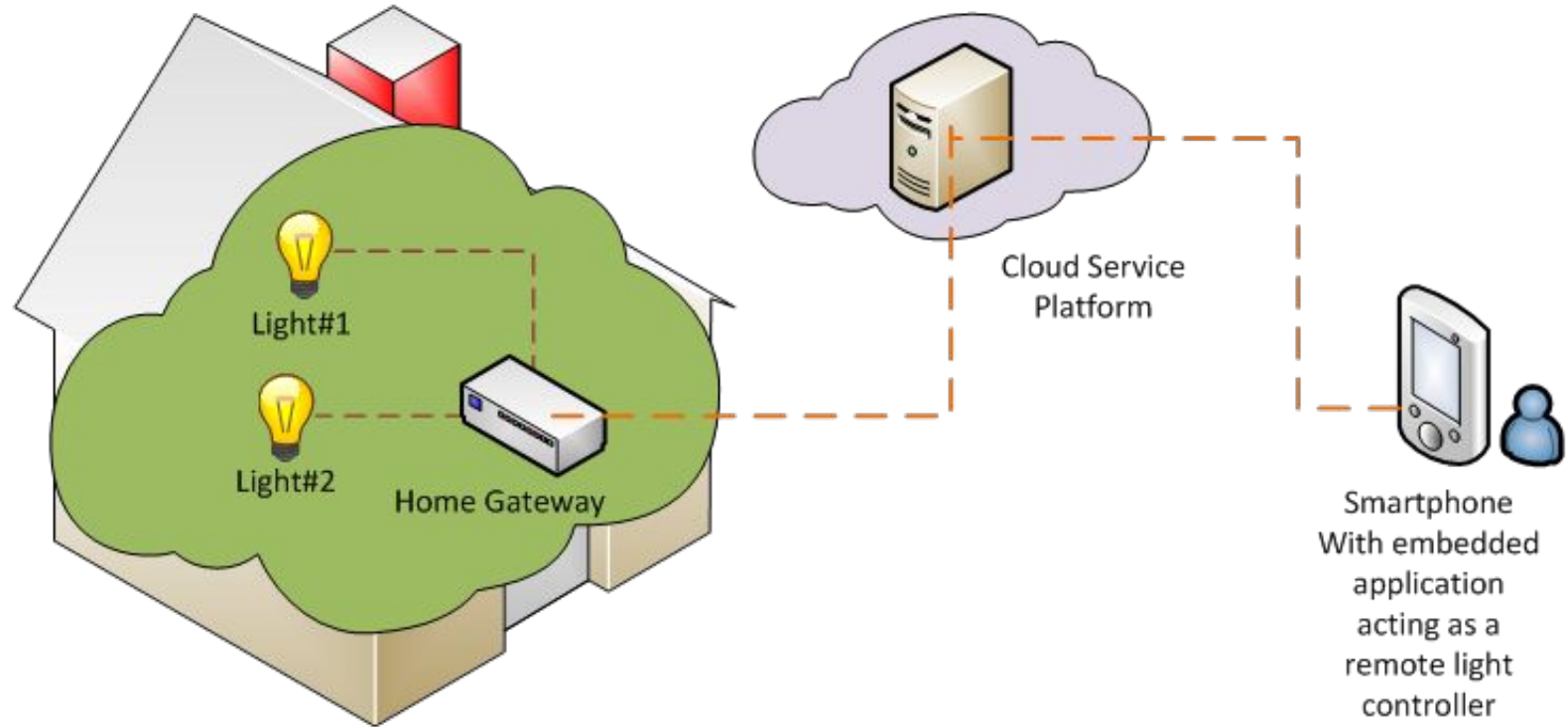   - Management of devices and applications

With oneM2M
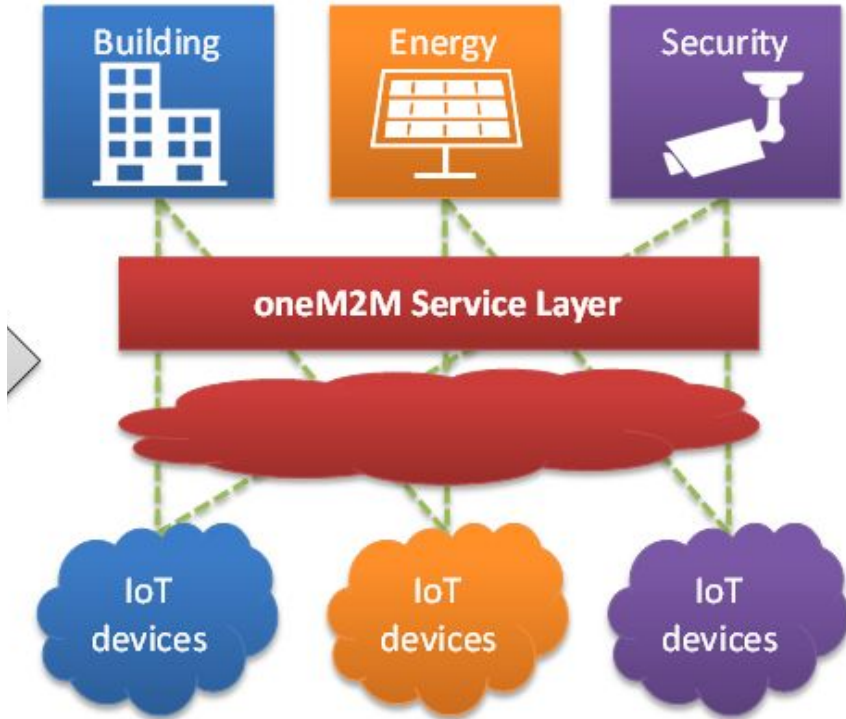
- End-to-end platform : common service capabilities layer
- Interoperability at the level of data and control exchanges via uniform APIs
- Seamless interaction between heterogeneous applications and devices
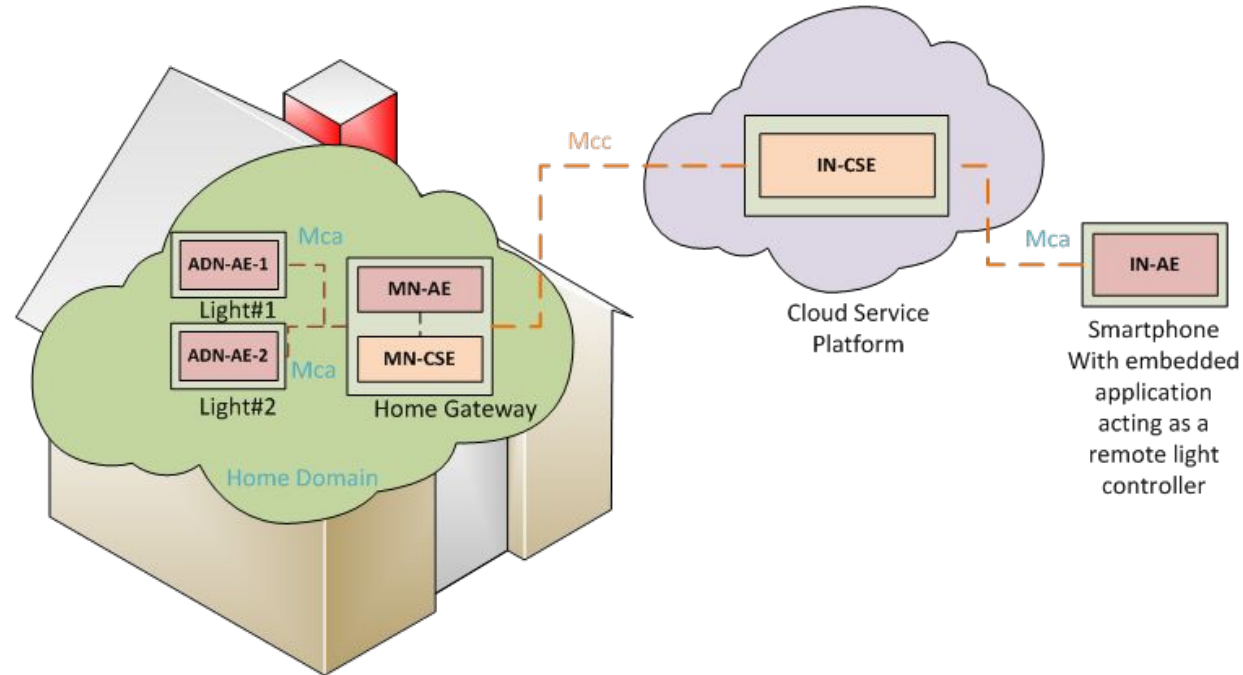
Cloud Interoperability

Source : http://www.onem2m.org/images/app_dev_guide/tr-0057/cloud_provider.png

Ontology

http://www.onem2m.org/images/app_dev_guide/UseCase.png

IN : Infrastructure Node
MN : Middle Node

CSE : Common service entity
AE : Application Entity

ADN : Application Device Node

http://www.onem2m.org/images/app_dev_guide/UseCase.png

Steps to use OneM2M

1. Registration
2. Initial Resource Creation
3. Discovery of Container Resource
4. Discovery and Retrieval of Light states
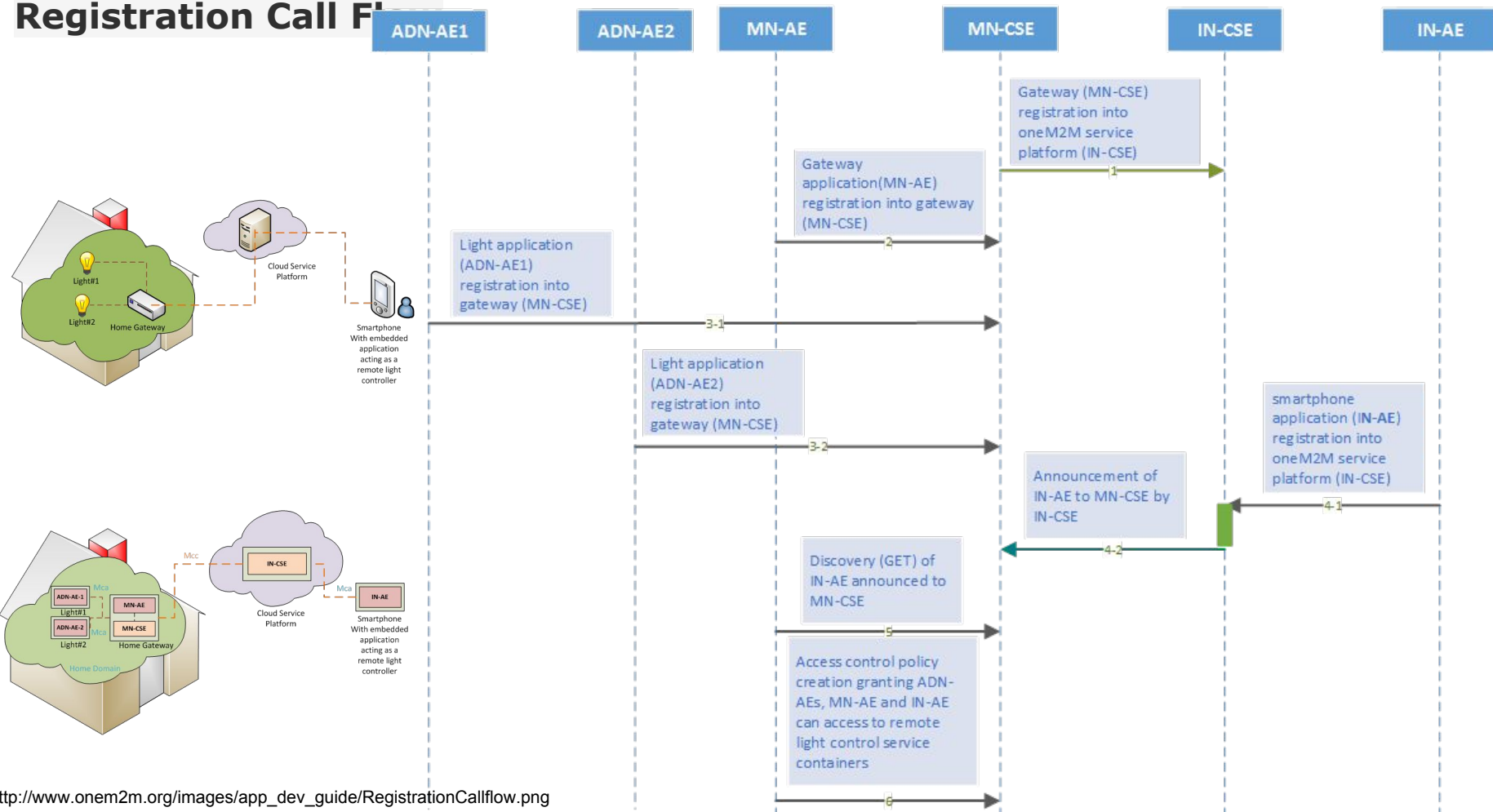5. Controlling the lights
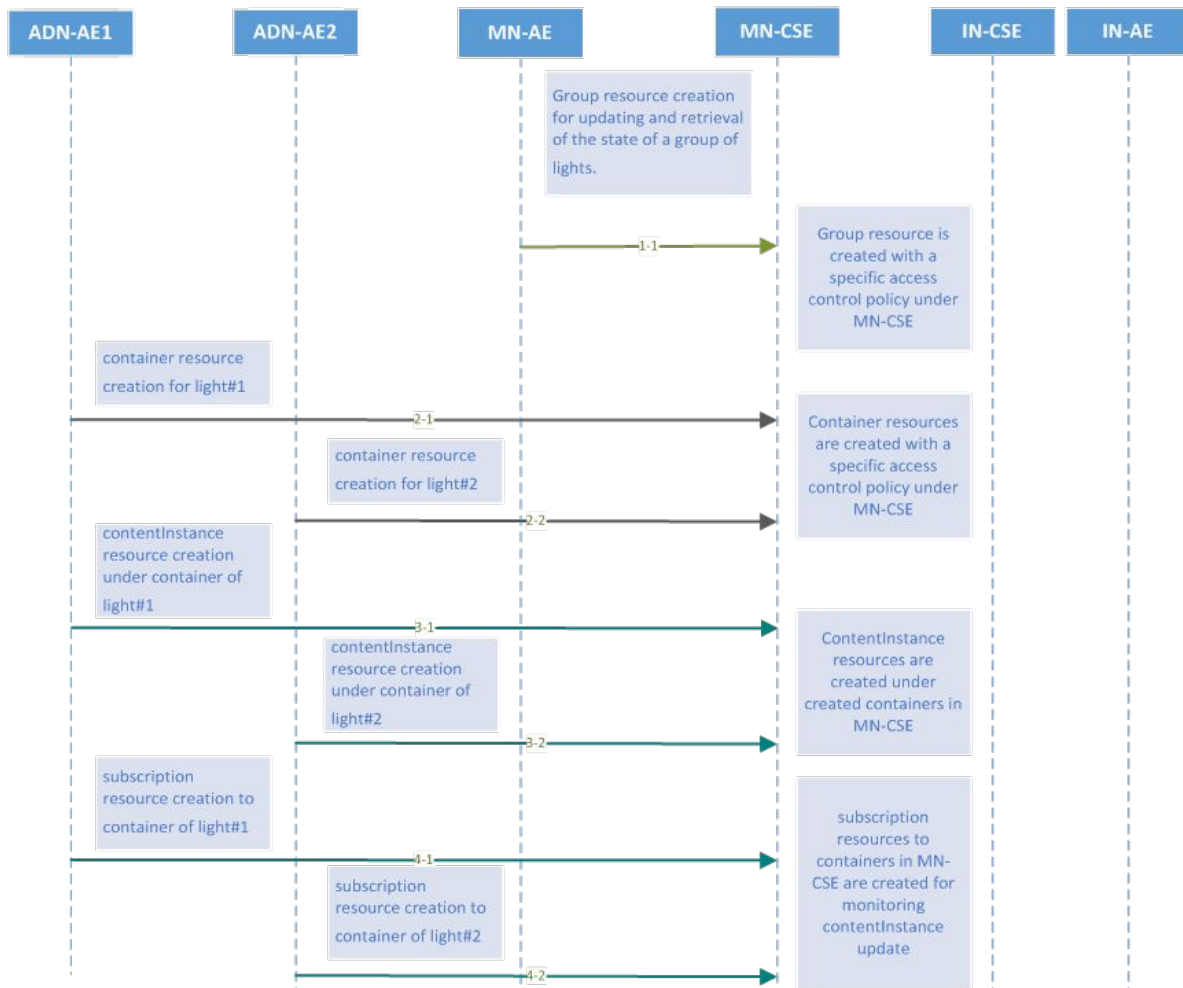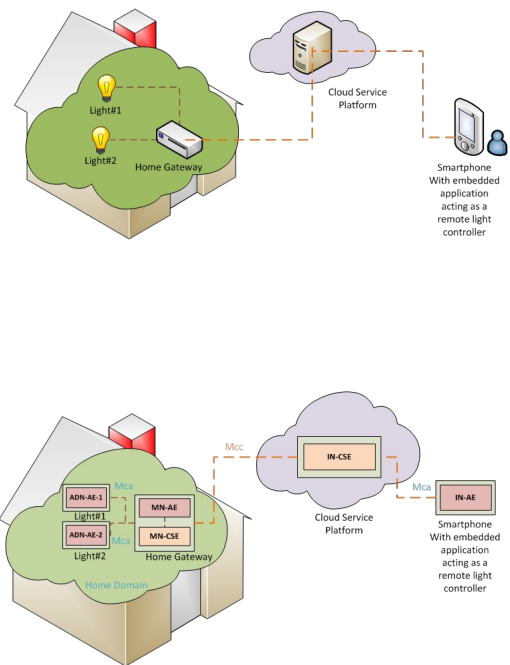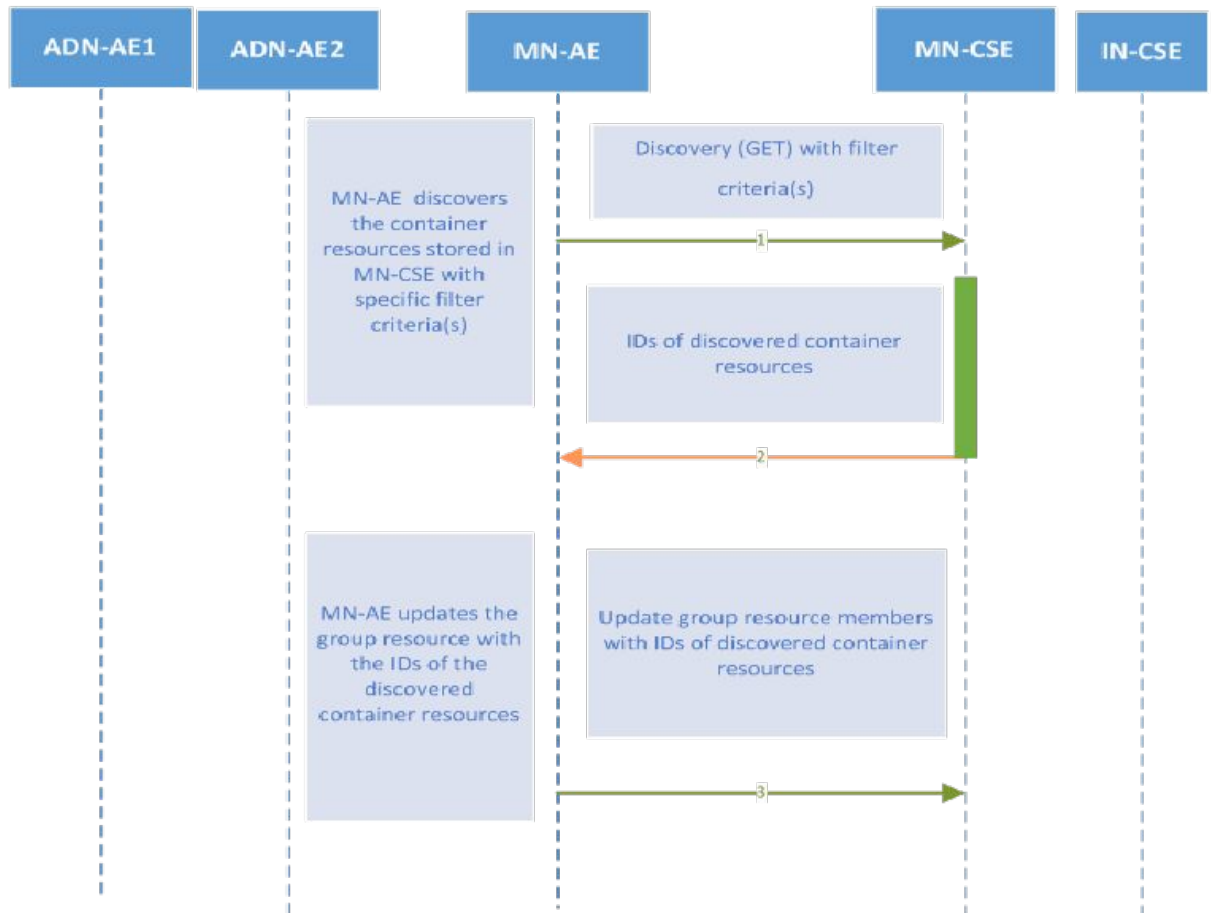
Registration Call Flow

Lights — **ADN-AE1** | **ADN-AE2**
Home Gateway — **MN-AE** | **MN-CSE**
Cloud — **IN-CSE**
Mobile — **IN-AE**

Gateway (MN-CSE) registration into oneM2M service platform (IN-CSE) — 1

Gateway application(MN-AE) registration into gateway (MN-CSE) — 2

Light application (ADN-AE1) registration into gateway (MN-CSE) — 3-1

Light application (ADN-AE2) registration into gateway (MN-CSE) — 3-2

smartphone application (IN-AE) registration into oneM2M service platform (IN-CSE) — 4-1

Announcement of IN-AE to MN-CSE by IN-CSE — 4-2

Discovery (GET) of IN-AE announced to MN-CSE — 5

Access control policy creation granting ADN-AEs, MN-AE and IN-AE can access to remote light control service containers — 6

http://www.onem2m.org/images/app_dev_guide/RegistrationCallflow.png

Resource Creation

Container Discovery

ADN-AE1 | ADN-AE2 | MN-AE | MN-CSE | IN-CSE

MN-AE discovers the container resources stored in MN-CSE with specific filter criteria(s)

Discovery (GET) with filter criteria(s)

1

IDs of discovered container resources

2

MN-AE updates the group resource with the IDs of the discovered container resources

Update group resource members with IDs of discovered container resources

3
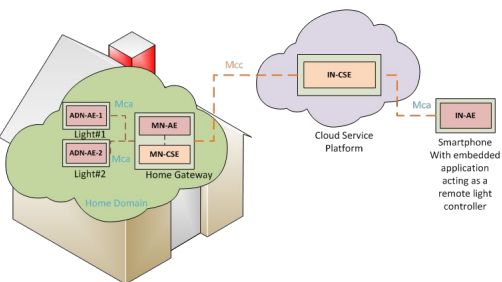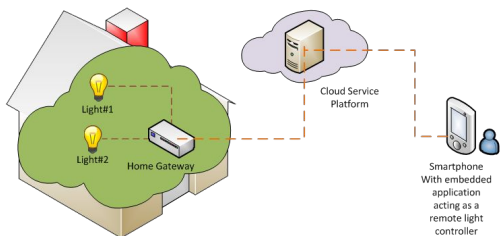
Discovery & Content Instance Retrieval

http://www.onem2m.org/images/app_dev_guide/DiscoveryRetrieval.png

Light Control

http://www.onem2m.org/images/app_dev_guide/LightControl.png
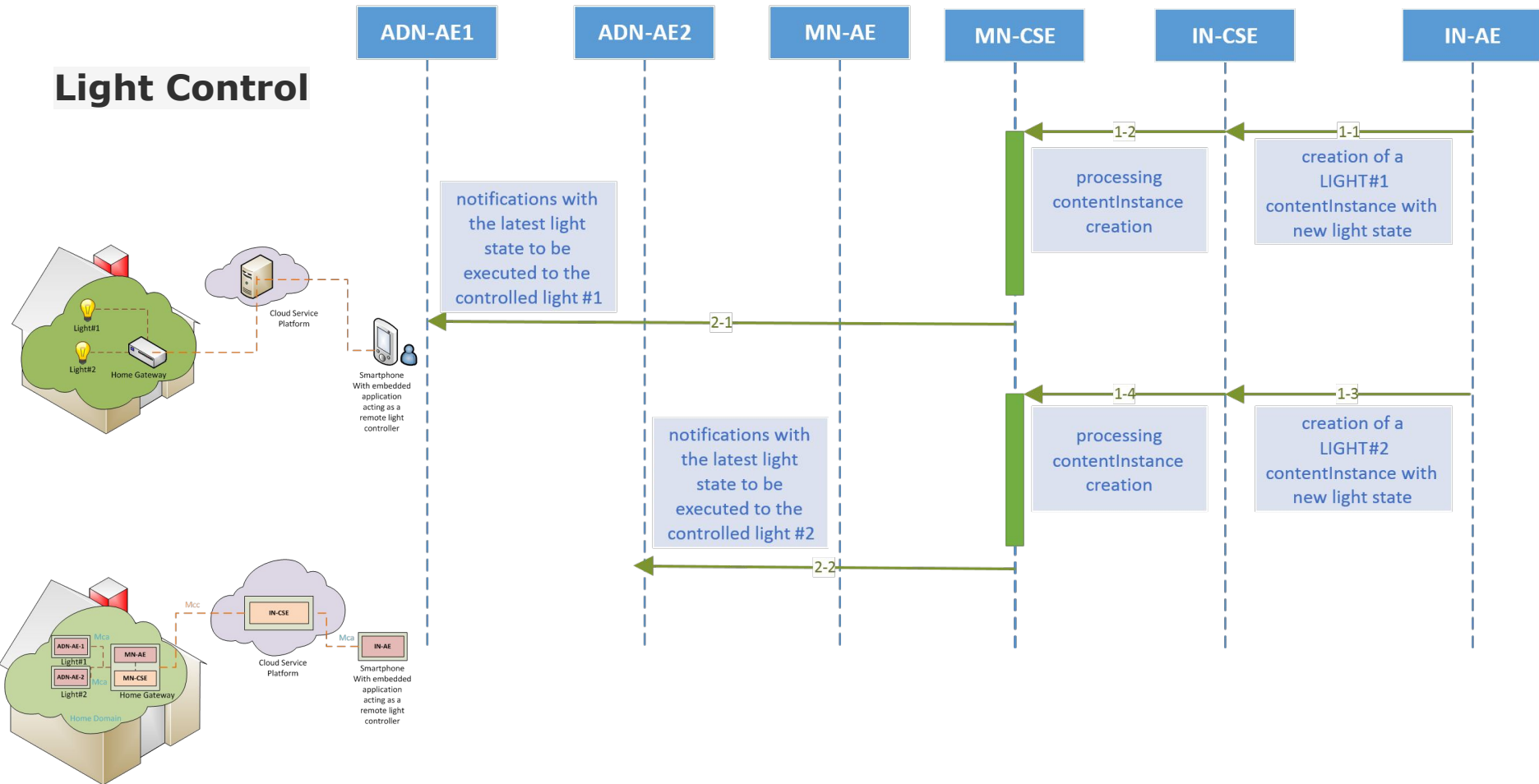
Features of OneM2M

- authentication, authorization, encryption
- **remote provisioning & activation**
- **connectivity setup**
- **buffering**
- scheduling
- synchronization
- **aggregation**
- **group communication**
- **device management**

Summary

- OneM2M facilitates interoperability
- Simple procedures allow developers to
    - build scalable systems
    - Focus on the idea/business rather than rewriting code
    - IoT products Easily maintainable

Upcoming Lab Session

- What will you be doing?
    - Focus on the basics
    - Semi-Building OM2M methods on your own
    - Establishing Communication with sensors and actuators
- Here is the "Requirements" document: (link)

References

- [Smart Cities with OneM2M](#)
- [REST APIs](#)
- [OneM2M Developer Guide](#)
- [Node-RED : A Dashboard for visualizing data](#)