



Application developer guide

SeungMyeong JEONG (sm.jeong@keti.re.kr)

Korea Electronics Technology Institute

2020.11.12

The project "International Digital Cooperation - ICT Standardisation" is funded by the European Union



Motivation



- See how oneM2M APIs seen from the previous session really work
- Then you can better understand the APIs and you can use them for your applications
- This provides the sample service application that consists of commonly used oneM2M APIs
- Reference: oneM2M TR-0025

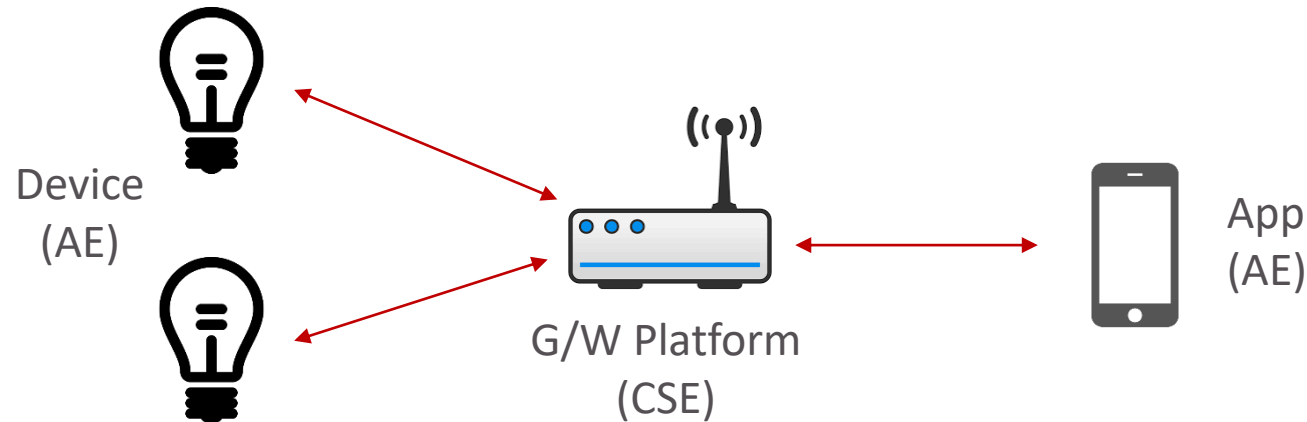
- Use Case
- Procedures and Call Flows
- Implementation



Use Case

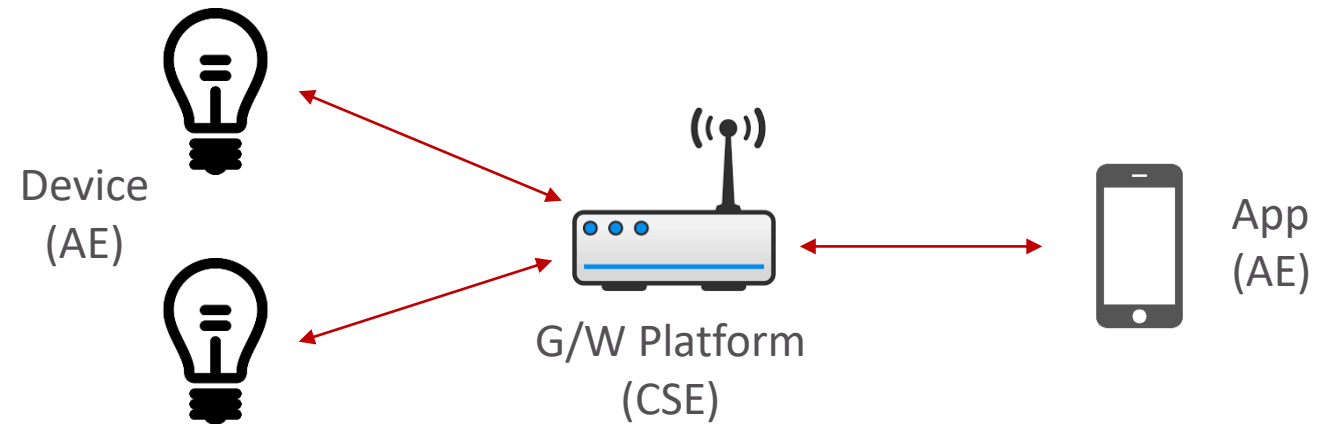
Simple Smart Home Service

- There is a house having a home G/W, light bulbs and an user application
- The devices and the user app are connected to the platform
- Sensing and actuation can be done by user application via the platform



APIs to try

- AE registration
- Resource discovery
- Data resource creation
- Group fan-out
- Subscription/notification

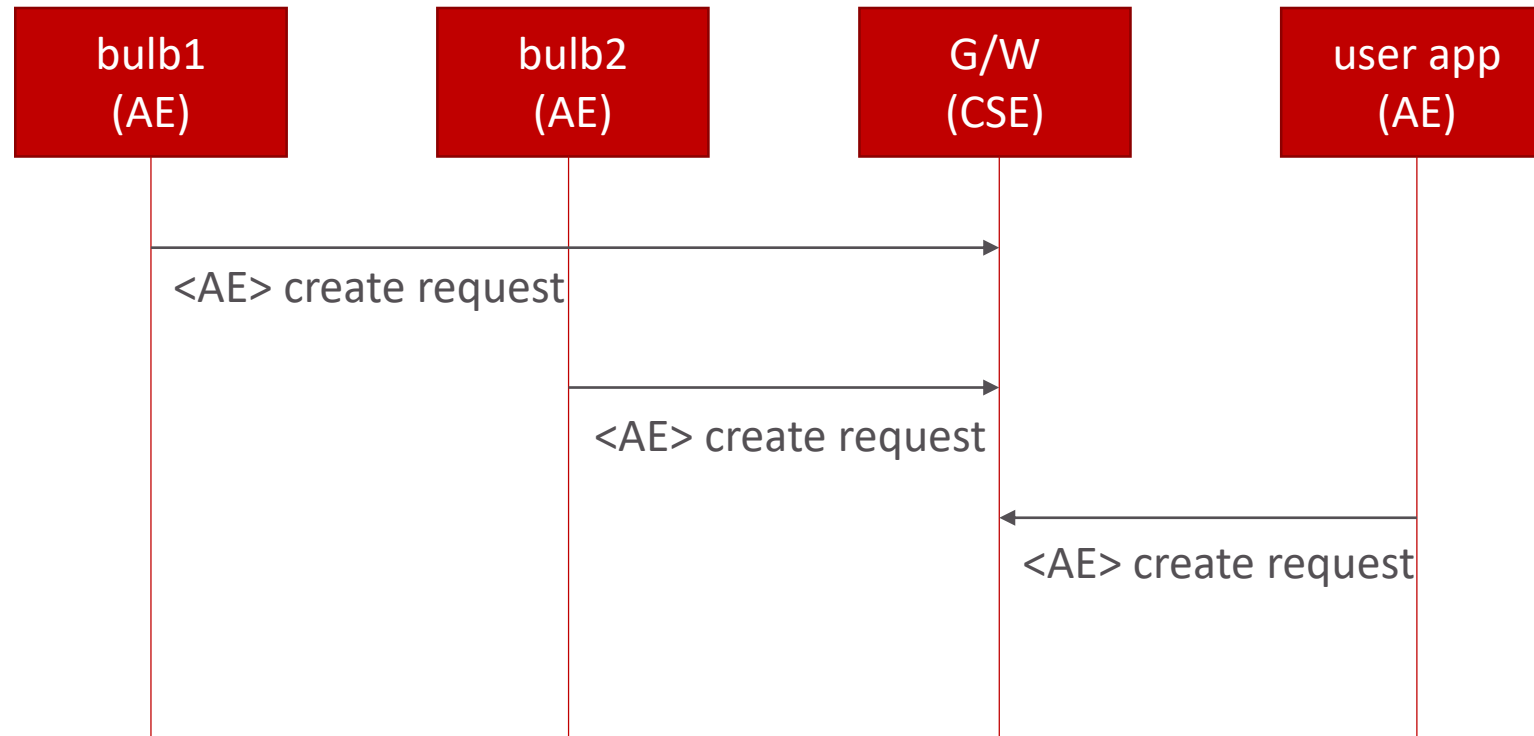




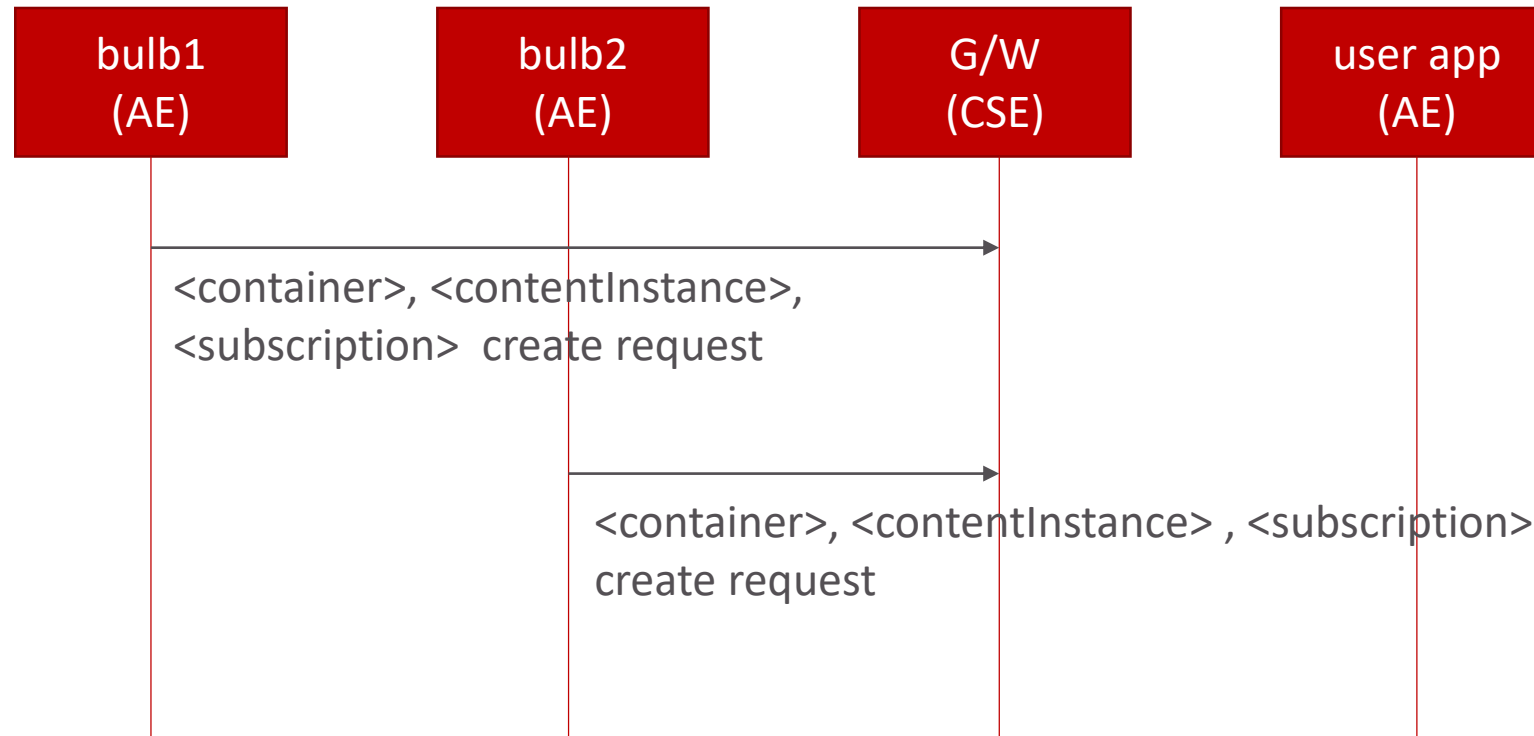
Procedures and Call Flows

1. Registration
 - Devices and user app (AEs) registers to the platform (CSE)
2. Initial resource creation
 - <container>, <contentInstance> and <subscription> resources
3. Target container discovery
 - group creation
4. Retrieve current light bulb status
5. Single switch on/off
 - watching notification event
6. Multiple switches on/off
 - group fan-out

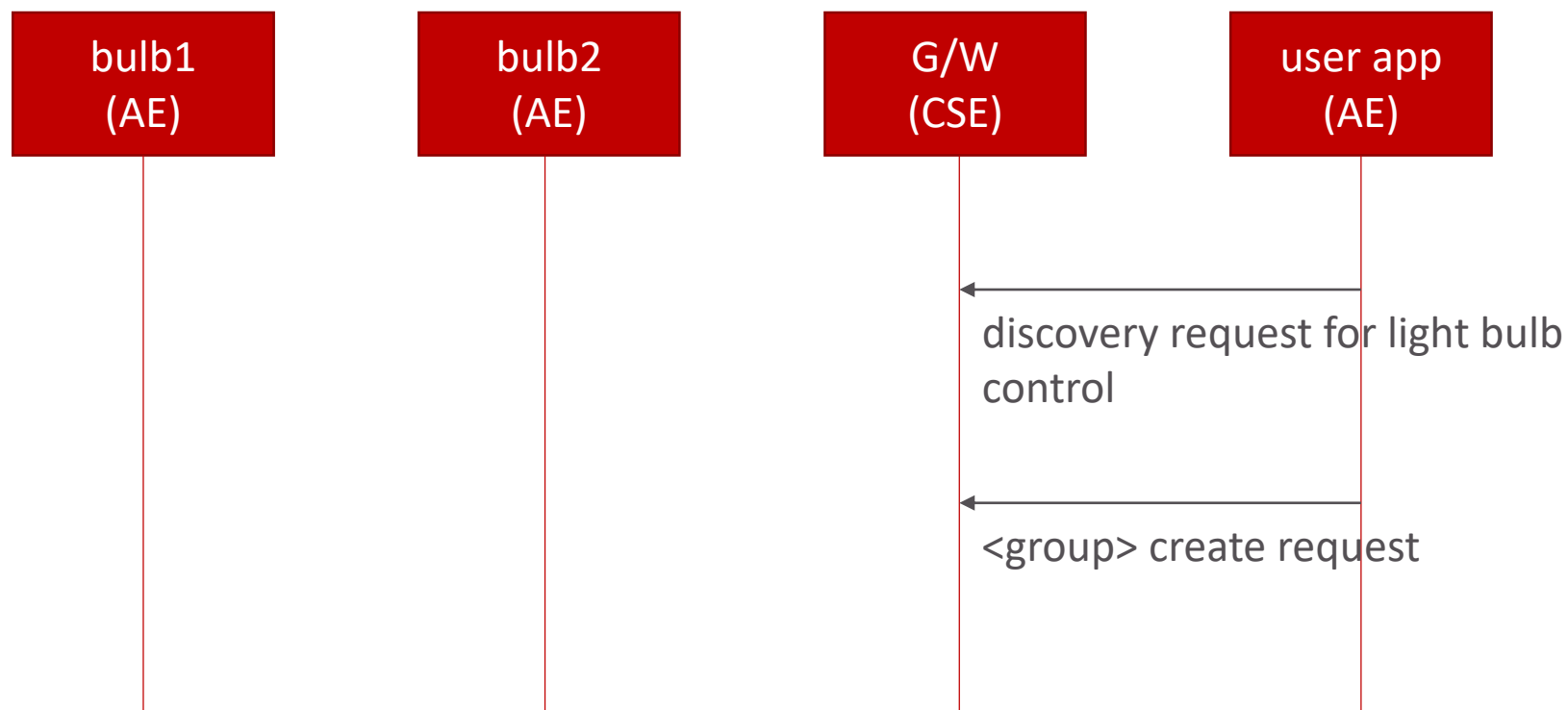
- Registration
 - Devices and user app (AEs) registers to the platform (CSE)



- Initial resource creation
 - <container>, <contentInstance> and <subscription> resources

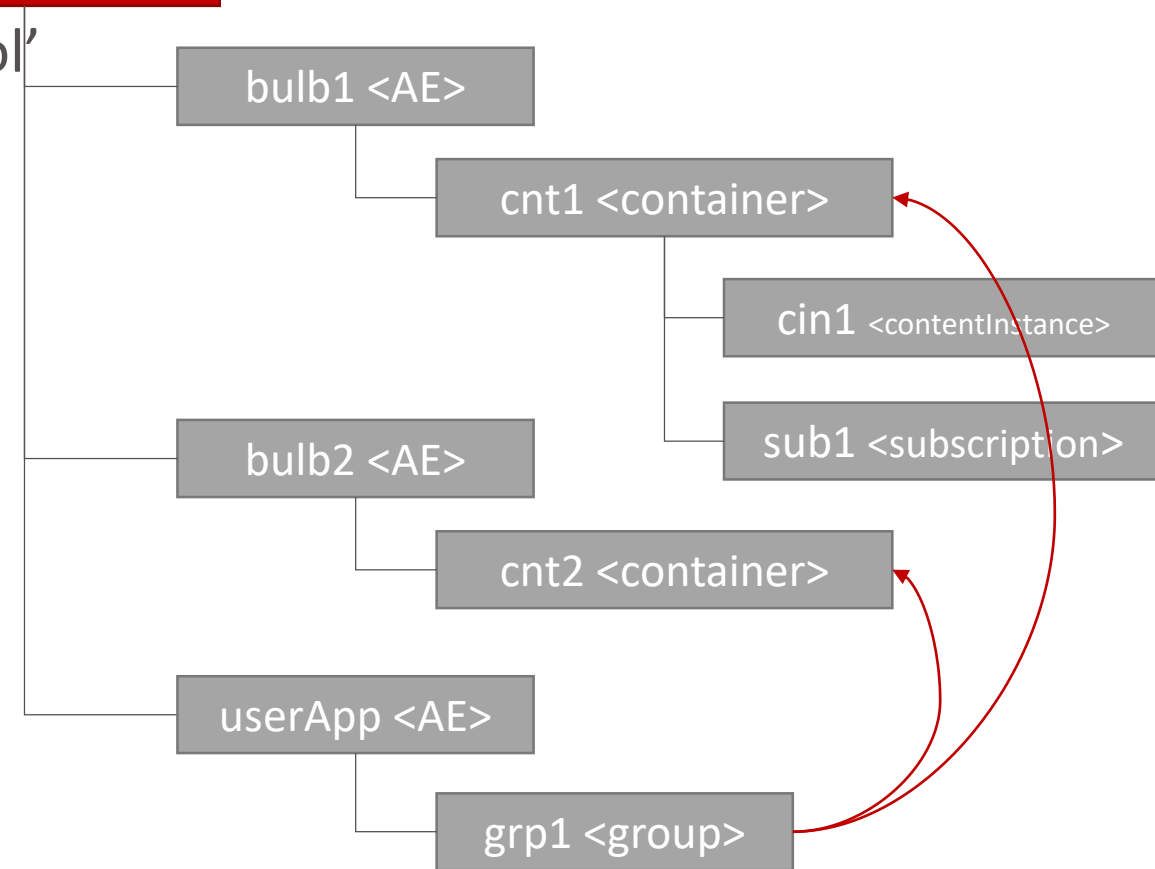


- Target container discovery
 - Group creation

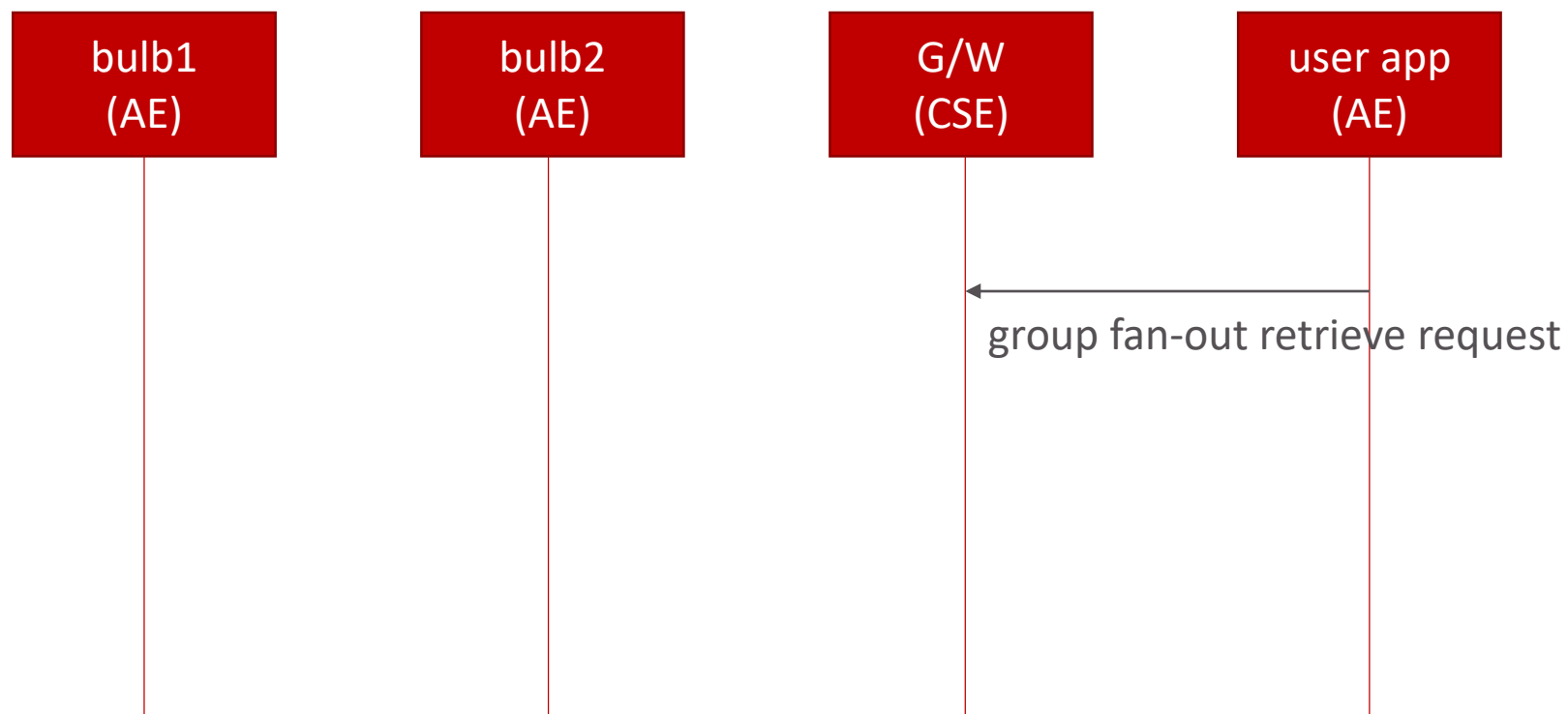


- Resource tree so far (abstract ver.)
 - bulbs have containers of 'status' and 'control'
 - containers have data instances
 - containers have subscriptions
 - userApp has groups for multiple status and control

CSE <CSEBase>

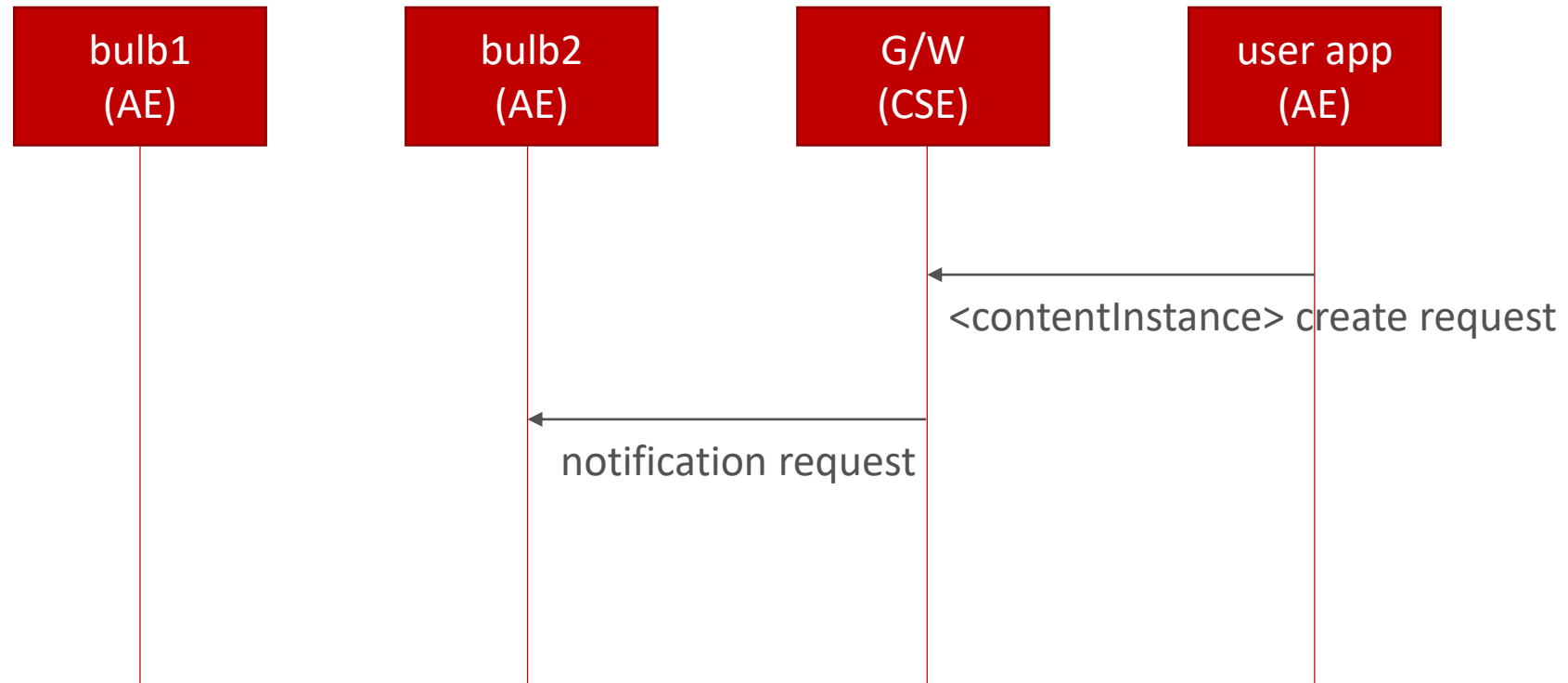


- Retrieve current light bulb status

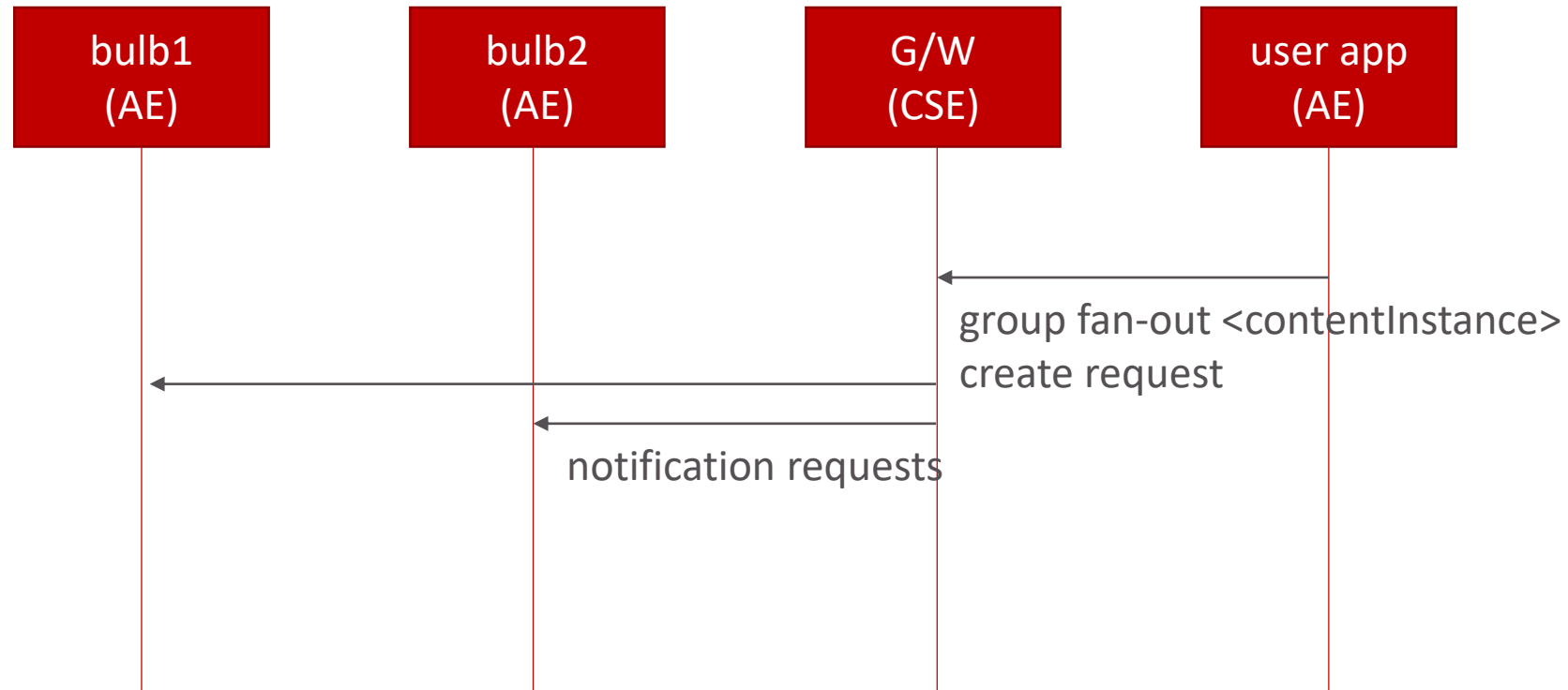


Call Flows

- Single switch on/off
 - watching notification event



- Multiple switches on/off
 - group fan-out





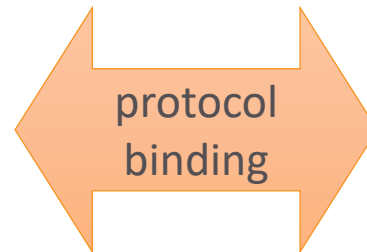
Implementation

AE Registration

- <AE> resource creation with mandatory attributes
 - Note that we use HTTP binding in this session

oneM2M primitive

```
{
  "to": "Mobius",
  "fr": "CAE1",
  "op": 1,
  "rqi": "1234",
  "pc": {
    "m2m:ae": {
      "rn": "bulb1",
      "api": "lightBulb",
      "rr": true
    }
  }
}
```



HTTP request

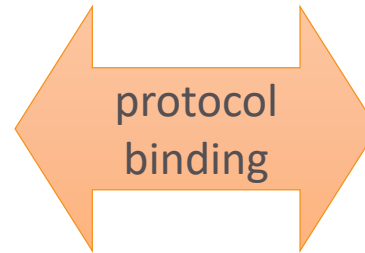
POST /Mobius HTTP/1.1	
Host:	localhost:7579
X-M2M-Origin:	CAE1
X-M2M-RI:	1234
Content-Type:	application/json; ty=2
Accept:	application/json
<pre>{ "m2m:ae": { "rn": "bulb1", "api": "lightBulb", "rr": true } }</pre>	

Initial Resource Creation

- <container> resources creation for sensing and control
 - Iterate this for two light bulbs 'status' and 'control' containers

oneM2M primitive

```
{
  "to": "Mobius/bulb1",
  "fr": "CAE1",
  "op": 1,
  "rqi": "1234",
  "pc": {
    "m2m:cnt": {
      "rn": "status",
      "mni": 100
    }
  }
}
```



HTTP request

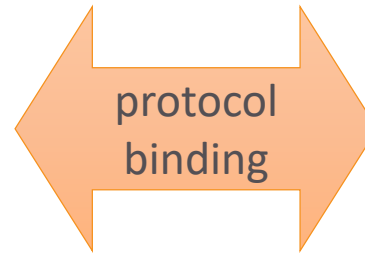
POST /Mobius/bulb1 HTTP/1.1	
Host:	localhost:7579
X-M2M-Origin:	CAE1
X-M2M-RI:	1234
Content-Type:	application/json; ty=3
Accept:	application/json
<pre>{ "m2m:cnt": { "rn": "status", "mni": 100 } }</pre>	

Initial Resource Creation

- <contentInstance> resources creation for initial status
 - Iterate this for status containers of the two light bulbs

oneM2M primitive

```
{
  "to": "Mobius/bulb1/status",
  "fr": "CAE1",
  "op": 1,
  "rqi": "1234",
  "pc": {
    "m2m:cin": {
      "con": "ON"
    }
  }
}
```



HTTP request

POST /Mobius/bulb1/status HTTP/1.1	
Host:	localhost:7579
X-M2M-Origin:	CAE1
X-M2M-RI:	1234
Content-Type:	application/json; ty=4
Accept:	application/json
<pre>{ "m2m:cin": { "con": "ON" } }</pre>	

Initial Resource Creation

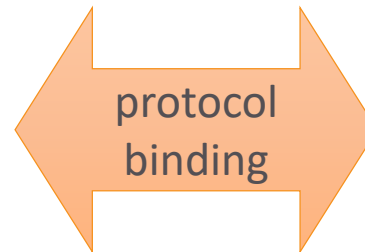
- <subscription> resources creation for initial
 - Iterate this for control containers of the two light bulbs

notificationEventType (TS-0004)

Value	Interpretation
1	Update_of_Resource
2	Delete_of_Resource
3	Create_of_Direct_Child_Resource
4	Delete_of_Direct_Child_Resource

oneM2M primitive

```
{
  "to": "Mobius/bulb1/control",
  "fr": "CAE1",
  "op": 1,
  "rqi": "1234",
  "pc": {
    "m2m:sub": {
      "rn": "mySub",
      "enc": {"net": [3]},
      "nu": ["http://127.0.0.1:7000?ct=json"],
      "nct": 1
    }
  }
}
```



HTTP request

POST /Mobius/bulb1/control HTTP/1.1	
Host:	localhost:7579
X-M2M-Origin:	CAE1
X-M2M-RI:	1234
Content-Type:	application/json; ty=23
Accept:	application/json

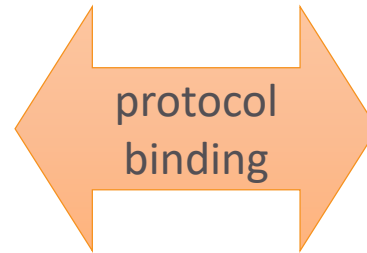
```
{
  "m2m:sub": {
    "rn": "mySub",
    "enc": {"net": [3]},
    "nu": ["http://127.0.0.1:7000?ct=json"],
    "nct": 1
  }
}
```

Resource Discovery

- Discover a specific resource type
 - Discover with other filter criteria

oneM2M primitive

```
{
  "to": "Mobius",
  "fr": "CAE3",
  "op": 2,
  "rqi": "1234",
  "fc": {
    "fu": 1,
    "ty": 3,
    "rn": "status"
  }
}
```



HTTP request

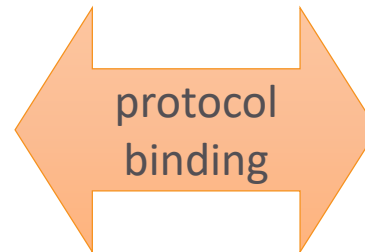
GET /Mobius?fu=1&ty=3&rn=status HTTP/1.1	
Host:	localhost:7579
X-M2M-Origin:	CAE3
X-M2M-RI:	1234
Accept:	application/json

Group creation

- <group> resource creation with the discovery result
 - Iterate this for status and control

oneM2M primitive

```
{
  "to": "Mobius/userApp",
  "fr": "CAE3",
  "op": 1,
  "rqi": "1234",
  "pc": {
    "m2m:grp": {
      "rn": "statusGroup",
      "mnm": 10,
      "mid": [
        "Mobius/bulb1/status",
        "Mobius/bulb2/status"
      ]
    }
  }
}
```



HTTP request

POST /Mobius/userApp HTTP/1.1	
Host:	localhost:7579
X-M2M-Origin:	CAE3
X-M2M-RI:	1234
Content-Type:	application/json; ty=9
Accept:	application/json
<pre>{ "m2m:grp": { "rn": "statusGroup", "mnm": 10, "mid": ["Mobius/bulb1/status", "Mobius/bulb2/status"] } }</pre>	

Fan-out Retrieval

- Group fan-out retrieve request for status containers of the bulbs
 - Then how to get latest data of the two status containers?

oneM2M primitive

```
{  
  "to": "Mobius/userApp/statusGroup/fopt",  
  "fr": "CAE3",  
  "op": 2,  
  "rqi": "1234"  
}
```

protocol
binding

HTTP request

POST /Mobius/userApp/statusGroup/fopt HTTP/1.1	
Host:	localhost:7579
X-M2M-Origin:	CAE3
X-M2M-RI:	1234
Accept:	application/json

- Single switch on/off with <contentInstance> resource creation
 - Watch event notification
 - Notification will be handled by the device application for actuation

HTTP request

POST /Mobius/bulb1/control HTTP/1.1	
Host:	localhost:7579
X-M2M-Origin:	CAE1
X-M2M-RI:	1234
Content-Type:	application/json; ty=4
Accept:	application/json

```
{
  "m2m:cin": {
    "con": "OFF"
  }
}
```

event triggers
notification

notification HTTP body

```
{
  "m2m:sgn": {
    "sur": "Mobius/bulb1/control",
    "nev": {
      "net": 3,
      "rep": {
        "m2m:cin": {
          "con": "OFF"
        }
      }
    }
  }
}
```


Device Control over Group

- Multiple switches on/off with <contentInstance> resource creation using group fan-out
 - Watch incoming event notifications
 - Notification will be handled by the device applications for actuation

oneM2M primitive

```
{
  "to": "Mobius/userApp/controlGroup",
  "fr": "CAE3",
  "op": 1,
  "rqi": "1234",
  "pc": {
    "m2m:cin": {
      "con": "ON"
    }
  }
}
```

protocol
binding

HTTP request

POST /Mobius/userApp/controlGroup/fopt HTTP/1.1

Host:	localhost:7579
X-M2M-Origin:	CAE3
X-M2M-RI:	1234
Content-Type:	application/json; ty=4
Accept:	application/json

```
{
  "m2m:cin": {
    "con": "ON"
  }
}
```