Figure 1: oneM2M logo

oneM2M Technical Report

Document Number	TR-0025 V2.0.3
Document Name:	Application Developer Guide
Date:	2019-June-06
Abstract:	Provides a use case for guiding application developers to develop applications using functionalities provided by a oneM2M service platform.
Template Version:23 February 2015 (Dot not modify)	Template Version:23 February 2015 (Dot not modify)

About oneM2M

The purpose and goal of oneM2M is to develop technical specifications which address the need for a common M2M Service Layer that can be readily embedded within various hardware and software, and relied upon to connect the myriad of devices in the field with M2M application servers worldwide.

More information about oneM2M may be found at: http//www.oneM2M.org

Copyright Notification

Contents

1 Scope

The present document provides a guide for application developers to develop applications using functionalities provided by any oneM2M compliant service

platform with the scope of as follows:

- Objective of the use case,

- The architecture of the use case mapped into an oneM2M service platform,

- The execution procedures for implementation of the use case, and

- Implementation details of the use case.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or nonspecific. For specific references, only the cited version applies. For non-specific references, the latest version of the reference document (including any amendments) applies.

The following referenced documents are necessary for the application of the present document.

Not applicable.

2.2 Informative references

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

- [i.1] oneM2M Drafting Rules

    NOTE: Available at http://www.onem2m.org/images/files/oneM2M-Drafting-Rules.pdf

- [i.2] oneM2M TS-0001 (V12.12.1): "Functional Architecture".

- [i.3] oneM2M TS-0004 (V2.9.0): "Service Layer Core protocol Specification".

- [i.4] oneM2M TS-0009 (V2.8.0): "HTTP Protocol Binding".

- [i.5] oneM2M TS-0011: "Common Terminology".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in oneM2M TS-0011 [i.5] and the following apply.

NOTE: A term defined in the present document takes precedence over the definition of the same term, if any, in oneM2M TS-0011 [i.5].

M2M service provider domain: part of the M2M System that is associated with a specific M2M Service Provider

registrar CSE: CSE where an Application or another CSE has registered

resource: uniquely addressable entity in oneM2M architecture

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

```
ACP       Access Control Policy
ADN       Application Dedicated Node
ADN-AE    AE which resides in the Application Dedicated Node
AE        Application Entity
CoAP      Constrained Application Protocol
CSE       Common Services Entity
CSE-ID    Common Service Entity Identifier
DNS       Domain Name System
FQDN      Fully Qualified Domain Name
HTTP      HyperText Transfer Protocol
IN        Infrastructure Node
IN-AE     Application Entity that is registered with the CSE in
the Infrastructure Node
IN-CSE    CSE which resides in the Infrastructure Node
IP        Internet Protocol
JSON      JavaScript Object Notation
M2M       Machine to Machine
Mca       Reference Point for M2M Communication with AE
Mcc       Reference Point for M2M Communication with CSE
MN        Middle Node
MN-AE     Application Entity that is registered with the CSE in
Middle Node
MN-CSE    CSE which resides in the Middle Node
PoA       Point of Access
SP        Service Provider
URI       Uniform Resource Identifier
XML       eXtensible Markup Language
```

4 Conventions

The key words "Shall", "Shall not", "May", "Need not", "Should", "Should not" in this document are to be interpreted as described in the oneM2M Drafting Rules [i.1].

5 Use case

This guide is based on a home lighting use case involving lights in a home that can be remotely controlled by a user's smartphone leveraging the capabilities of oneM2M. An overview of the use case is shown in figure 5-1. The main components are introduced as follows:

- The lights are deployed in a home and are attached to a home gateway.

- The home gateway communicates with a cloud service platform allowing the lights to be controlled remotely by the smartphone.
- The cloud service platform supports a set of services to enable the smartphone to more easily control the lights in the home. Some examples of services include registration, discovery, data management, group management, subscription/notification etc
- The smartphone hosts an application used to remotely control the lights in the home and supports the following capabilities:
  - Discovery of lights deployed in the home.
  - Sending commands to change light states i.e. ON and OFF.
  - Retrieval of light states.
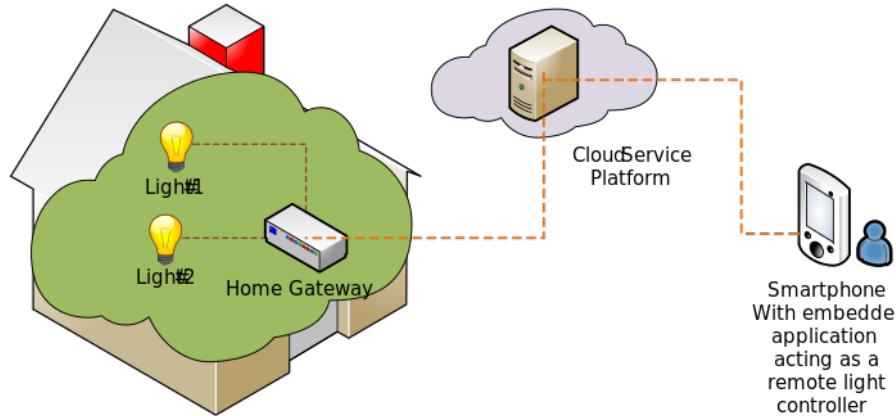
Figure 2: Figure 5-1: Overview of remote lights control use case

6 Architecture

This clause describes how the different components of this use case can be represented by corresponding oneM2M architectural entities as shown in figure 6-1.
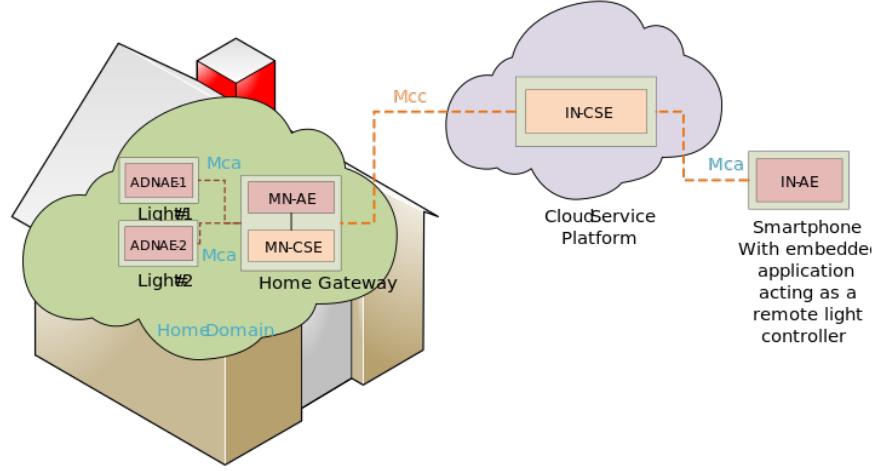
Figure 3: Figure 6-1: oneM2M functional architecture of remote lights control use case

In the oneM2M functional architecture two basic types of entities are defined. One is an AE (short for Application Entity) and the other is a CSE (short for Common Services Entity). In this use case, the lights and smartphone each host an AE. Also an IN-CSE (short for Infrastructure Node CSE) is hosted in the cloud by the oneM2M Service Provider and a MN-CSE (short for Middle Node CSE) is hosted on the Home Gateway.

The oneM2M defined Mca reference point is used to interface an AE and CSE. The oneM2M defined Mcc reference point is used to interface CSEs. In this use case, the reference point used between a Light AE and home gateway MN-CSE or Smartphone AE and IN-CSE is Mca while reference point used between the home gateway MN-CSE and oneM2M service platform IN-CSE is Mcc

In summary, applications used in the current use case are classified as follows:

- ADN-AE1: an application embedded in *Light#1* with capabilities to control *Light#1* and interact with the home gateway MN-CSE through *M ca* reference point;
- ADN-AE2: an application embedded in *Light# 2* with capabilities to control *Light# 2* and interact with the home gateway MN-CSE through *M ca* reference point;
- IN-AE: a smartphone application embedded in the smartphone device with capabilities to interact directly with the oneM2M service platform IN-CSE through *M cc* reference point and thereby remotely control *Light#1* and *Light# 2* ;
- MN-AE: a gateway application embedded into the home gateway that

interacts with the MN-CSE through *M ca* reference point.

7 Procedures

7.1 Introduction

Editor note: Most request are missing the RVI request attribute

The deployment of the oneM2M standard in the present use case requires procedures that are classified as follows:

- **Registration** : The current procedure contains light application registration, gateway application registration, and accessControlPolicy resource creation for selective access to data storage resources.
- **Initial resource creation** : The current procedure contains group resource creation, container resources creation with specific access control policies, content instance resources creation with initial light states, subscription resources creation for notifications.
- **Discovery of container resource** : all containers with a specific filter criteria are discovered by the gateway application and then configured as members of a group resource.
- **Discovery and retrieval lights states** : all containers with a specific filter criteria are discovered and retrieved using resource identities through a smartphone application which gains access to oneM2M service platform so content information can be retrieved.
- **Single light switch on/off:** Any light that is discovered by and connected to the smartphone application is able to be switched on and off via a smartphone application.
- **Multiple lights switch on/off:** Multiple lights that are discovered are able to be switched on and off together via a smartphone application.

7.2 Call Flows

7.2.1 Application registration and Access control policy creation

Call flows regarding the registration phase depicted in figure 7.2.1-1 are ordered as follows:

1. Gateway (MN-CSE) registers with the oneM2M service platform (IN-CSE).
2. Gateway application (MN-AE) registers with the gateway (MN-CSE).
3. Light applications (ADN-AE1 and ADN-AE2) register with the gateway (MN-CSE).
4. Smartphone application (IN-AE) registers with the oneM2M service platform (IN-CSE).
5. Gateway application (MN-AE) discovers the smartphone application (IN-AE) from gateway (MN-CSE) with specific filter criteria. The discovered IN-AE is granted access to the remote light control service containers.

6. Gateway application (MN-AE) creates an accessControlPolicy resource granting all the entities playing roles in the current use case including ADN-AE1, ADN-AE2, MN-AE and IN-AE access to the created container and content instance resources.
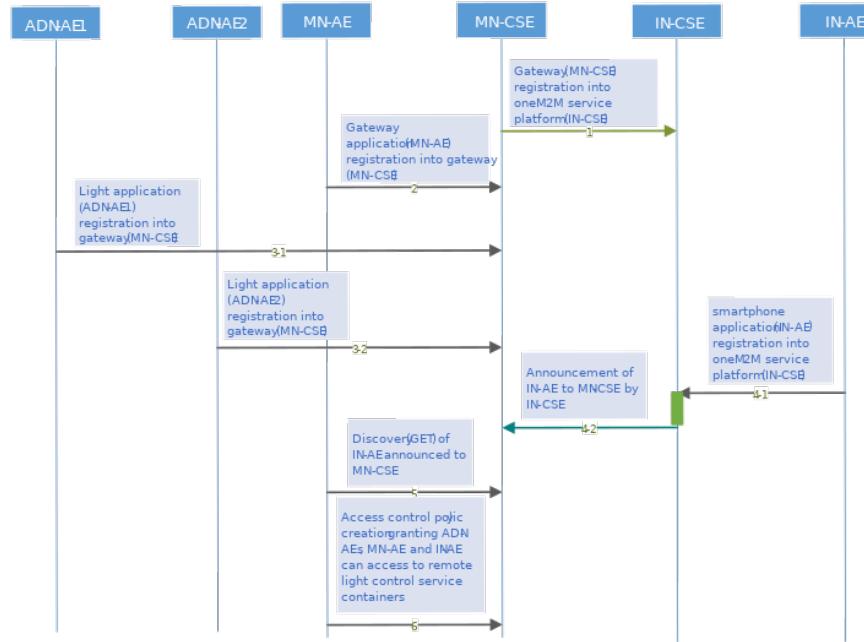
Figure 4: Figure 7.2.1-1: Registration phase call flows

7.2.2 Initial resource creation

Call flows regarding the initial resource creation phase depicted in figure 7.2.2-1 are ordered as follows:

1. Gateway application (MN-AE) creates a group resource on gateway (MN-CSE), for updating and retrieving group light state named as *containers _group* . The group members are added from the list of discovered container resources that the MN-AE discovers. The group resource is created with a link to the same access control policy.
2. Two container resources are created in the gateway (MN-CSE) to store the light states under the registered light application ADN-AE1 and ADN-AE2, respectively. The containers are created with a link to the same access control policy.
3. Content Instance resources are created by light applications (ADN-AE1 and ADN-AE2) under each created container and represent the controlled light states.

4. Subscription resources are created under the containers in the gateway (MN-CSE) so that subscribers, i.e. light applications, can be notified whenever there is a new contentInstance resource created by the IN-AE.
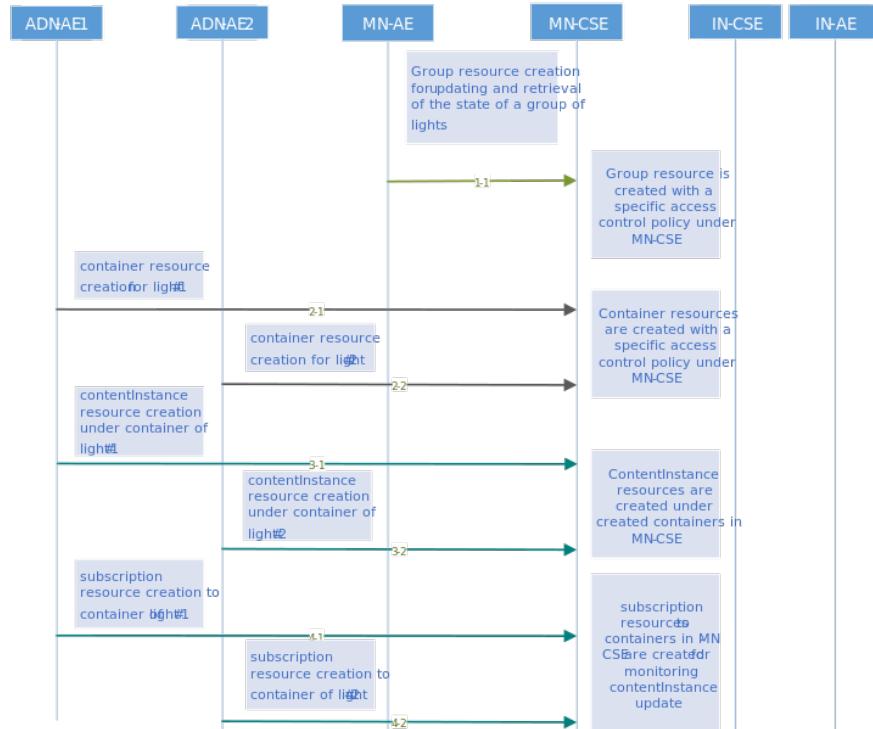
Figure 5: Figure 7.2.2-1: Initial resource creation phase call flows

7.2.3 Discovery of group resources

Call flows regarding the discovery and update of group resources are depicted in figure 7.2.3-1 and ordered as follows:

1. Gateway application (MN-AE) periodically sends a RETRIEVE request including the parameter *filterUsage* and specific filter criteria condition(s) as a query string for discovery of container resources stored in the MN-CSE of gateway. The filter criteria conditions for the discovery operation include *createdBefore, createdAfter, modifiedSince, unmodifiedSInce, label, creator, expireAfter, resourceType* etc.
2. Gateway (MN-CSE) responds with URIs of the discovered container resources, if any, to the gateway application (MN-AE) according to the filter criteria(s).

3. Gateway application (MN-AE) sends a update request to update the list of group members within the previously created group resource with identifiers of the discovered containers.
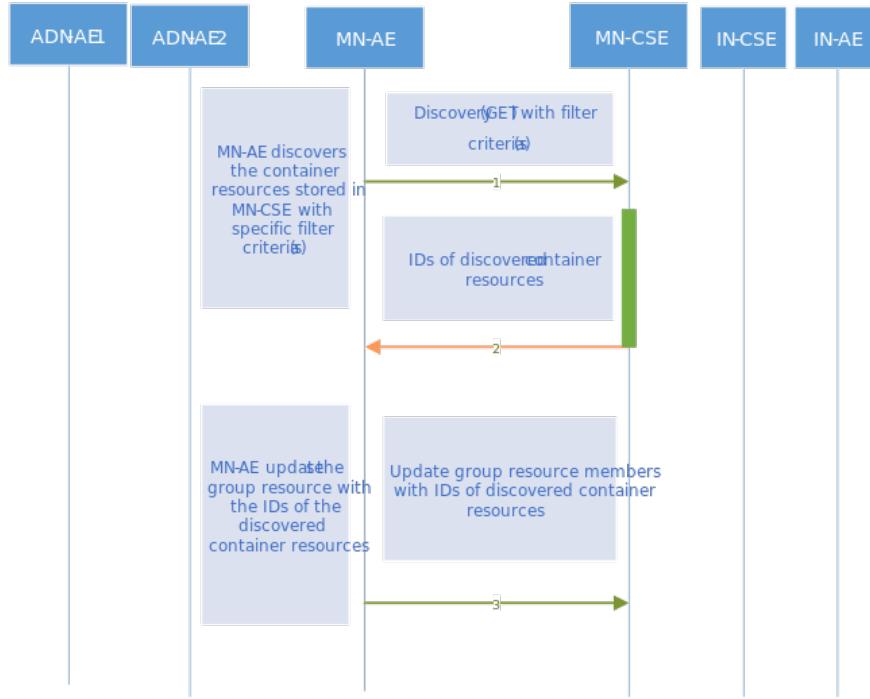
Figure 6: Figure 7.2.3-1: Discovery and group light state update phase call flows

7.2.4 Discovery and retrieval of contentInstance resources

Call flows regarding the discovery and retrieval of contentInstance resources depicted in figure 7.2.4-1 and 7.2.4-2 are ordered as follows:

1. The smartphone application (IN-AE) periodically sends a RETRIEVE request including the parameter *filterUsage* and specific filter criteria condition(s) as a query string for discovery of container resources stored in the MN-CSE of gateway.
2. The IN-AE also sends a Discovery request to the MN-CSE for the discovery of the group resources located in the MN-CSE.
3. The gateway (MN-CSE) responds to the IN-AE with URIs of the discovered container resources under ADN-AE1 and ADN-AE2, if any.
   For the case where the IN-AE sends a Discovery request for the discovery of group resources, the MN-CSE responds to the IN-AE with the URIs of the discovered group resources located in the MN-CSE, if any.

4. The IN-AE sends GET requests for retrieval of the latest contentInstance resources from each discovered light container resource.
   In the case of retrieval of the latest contentInstance resources of the group of containers, the IN-AE sends a retrieve request to the *fanOutPoint* of the discovered group resource.
5. The MN-CSE responds to the IN-AE with the latest light state(s).
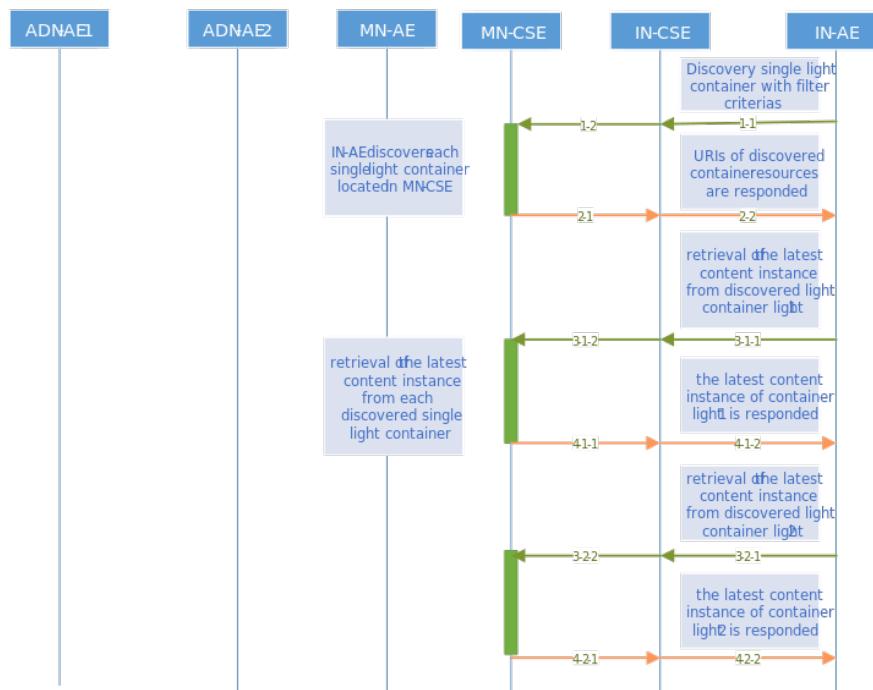
Figure 7: Figure 7.2.4-1: Discovery and single light retrieval phase call flows

7.3 Remote control scenarios

7.3.1 Introduction

Lights are able to be controlled remotely through the smartphone application accessing the oneM2M service platform. Two scenarios are introduced in clauses 7.3.2 and 7.3.3.

7.3.2 Single light control

Light#1 and *Light#2* are controlled remotely by a human user through a smartphone application (IN-AE). A call flow for single light control is depicted in figure 7.3.2-1 and the steps are ordered as follows:

13
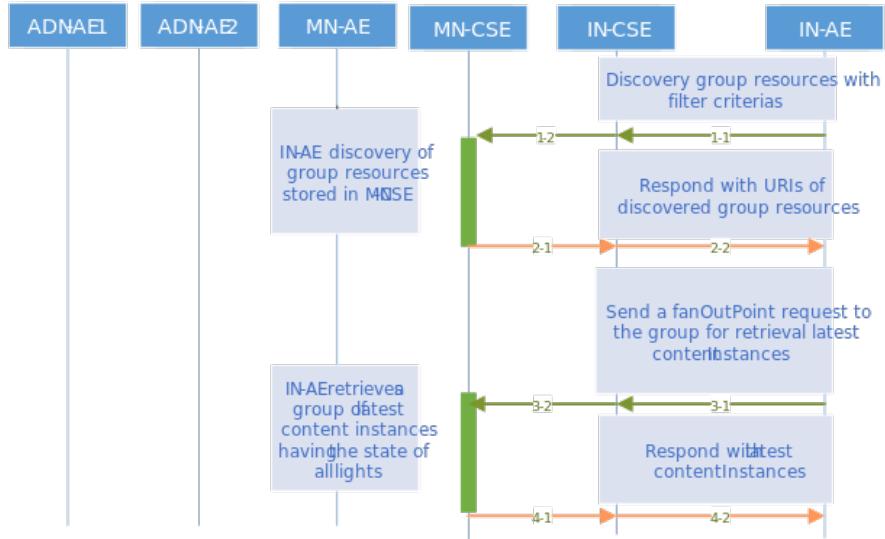
Figure 8: Figure 7.2.4-2 Discovery and a group of lights retrieval phase call flows

1. When the user updates the light state on her/his smartphone, the IN-AE creates a new contentInstance representing a new light state under the targeted container of a Light ADN-AE stored in the MN-CSE.
2. If the contentInstance is created sucessfully_,_ the MN-CSE sends a notification to the corresponsding Light ADN-AE to notify it that a new contentInstance resource was created.

7.3.3 Multiple light control

Users can also remotely control multiple lights through the smartphone application (IN-AE) by sending a single light control command to the group resource. A call flow for multiple lights control is depicted in figure 7.3.3-1 and the steps are ordered as follows:

1. When the user updates the state of a group of lights on her/his smartphone, the IN-AE sends a contentInstance create request targeting the group resource on the MN-CSE. The MN-CSE then fans out the request to the individual Light container member resources on the MN-CSE..
2. For each contentInstances created sucessfully_,_ the MN-CSE sends a notification to the corresponsding Light ADN-AE.

14
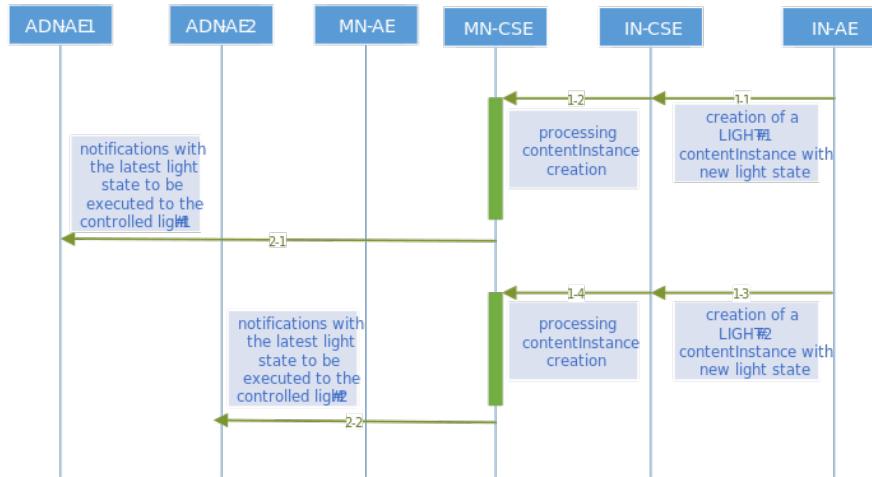
Figure 9: Figure 7.3.2-1: Single light remote control phase call flows
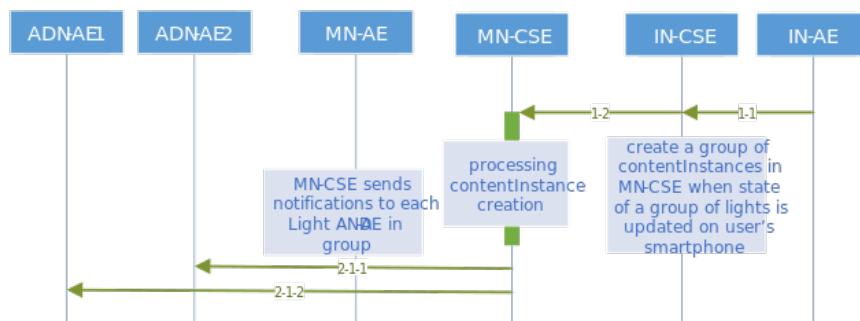
Figure 10: Figure 7.3.3-1: Multiple lights remote control phase call flows

8 Implementation

8.1 Introduction

Clause 8 presents necessary procedures required for the implementation of the remote lights control use case, including conditions that are met for the correct implementation of the current use case, and resource tree etc.

8.2 Assumptions

Assumptions are presented as below in order to ensure the remote lights control use case can be correctly implemented.

- All the applications are server capable;
- Devices and application entites are independently addressable with host names resolved by DNS network services;
- Host port number 8080 is reserved for oneM2M services;
- Security is not considered in the current use case;
- HTTP binding of oneM2M primitives is used in the current use case;
- Both XML and JSON serializations of oneM2M primitives are used in the current use case;
- All mandatory HTTP headers are presented in the HTTP requests while optional headers are selectively used in the current use case;
- All mandatory resource attributes for resources presented in the current use case are presented in the HTTP requests while optional resource attributes are selectively used in the current use case;
- The IN-CSE and MN-CSE in the current use case are deployed within the same oneM2M Service Provider domain;
- All AEs in the current use case are initially registered with CSEs and the identifier of the AEs are assigned by the Registrar CSE of the AE accordinlgy, starting with a character of 'C';
- All resources created in the current use case are addressable with the oneM2M Resource Identifier form of *Hierarchical address* ;
- Short names for the representation of the resources and attributes are used in the current use case;
- Default access control policy has already been created under IN-CSE and it is used for MN-CSE registration with IN-CSE;
- All request originators send *Blocking Requests* for accessing resources located in CSEs.

8.3 Addressing for Entities

Each oneM2M entity including AE and CSE are addressable with correct host address that can be IP addresses or FQDN addresses resolved to IP addresses by DNS network services according to addressing rules specified in oneM2M standards.

The IN-CSE and MN-CSE entities presented in this use case are addressable with the following identifiers.

- IN-CSE:
    - CSE-ID: `in-cse`
    - resourceName of IN-CSE's CSEBase resource: `server`
- MN-CSE:
    - CSE-ID: `mn-cse`
    - resourceName of MN-CSE's CSEBase resource: `home_gateway`

8.4 Modelling for Light State Data

The light state *ON* or *OFF* stored as the content of content instance resource is modelled as string in XML representation and can be represented as `< con >ON</ con >` or `<con>OFF</ con>` , while represented as *{ . . . "con" : "ON" . . . }* or *{. . . "con" : "OFF" . . . }* , respectively.

8.5 Resource Structure

8.5.0 Introduction

The development of an oneM2M application includes the design of the resource trees of service capability layers i.e. IN-CSE and MN-CSE in the current use case. The resource tree is constructed with child resources created according to the high level procedures presented in oneM2M application developer guide clause 7. All the child resources shown in the resource trees are mandatorily required in order to correctly implement the remote lights control use case.

8.5.1 Resource Structure of IN-CSE

The resource tree of IN-CSE starts with a CSEBase named *server* depicted in figure 8.5.1-1.

The root CSEBase has two direct child resources, a remoteCSE named *home_gateway* and an AE named *smartphone_ae* .

8.5.2 Resource Structure of MN-CSE

The resource tree of MN-CSE starts with a CSEBase named *home_gateway* depicted in figure 8.5.2-1.

The resource tree of MN-CSE is constructed with child resources as follows:

- an accessControlPolicy named *gateway_acp* ,
- an ADN-AE named *light_ae1* which contains sub-resources of a container named *light* and multiple contentInstances,
- an ADN-AE named *light_ae2* which contains sub-resources of a container named *light* and multiple contentInstances,
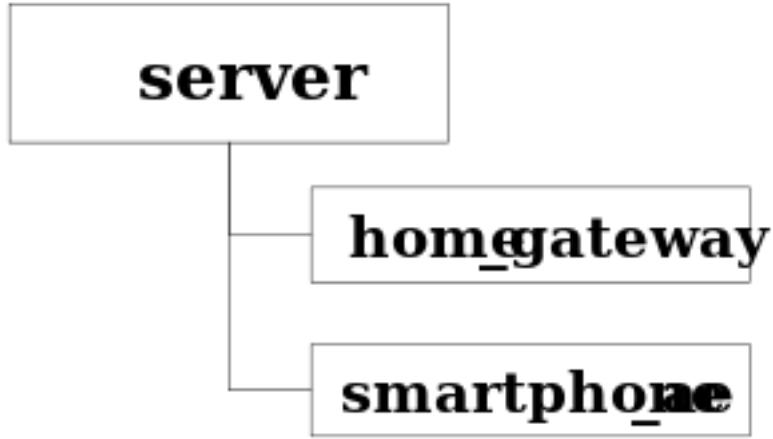
Figure 11: Figure 8.5.1-1: IN-CSE resource structure

- light_state_sub1 and light_state_sub2 subscription resources that are child resources of the two *light* containers, and
- a MN-AE named *gateway_ae* which contains a group child resource,
- A group resource named *containers_group* whose members are the *light* containers of each AND-AEs. Smartphone application users with proper access control privileges can send a *fanOutPoint* request to this group to create and retrieve content instances in the two_light_ containers.

8.6 Role of Entities

8.6.1 oneM2M service platform (IN-CSE)

The oneM2M service platform is modelled as an IN-CSE and is responsible for

- handling the registration requests from *the* smartphone AE and home gateway MN-CSE

8.6.2 Home gateway application (MN-AE)

The home gateway application is modelled as a MN-AE and is responsible for

- initializing the home gateway device,
- creating an access control policy resource gateway_acpin the MN-CSE,
- registering the home gateway application with the MN-CSE,
- creating the group resource with access control policy gateway_acp in the MN-CSE,
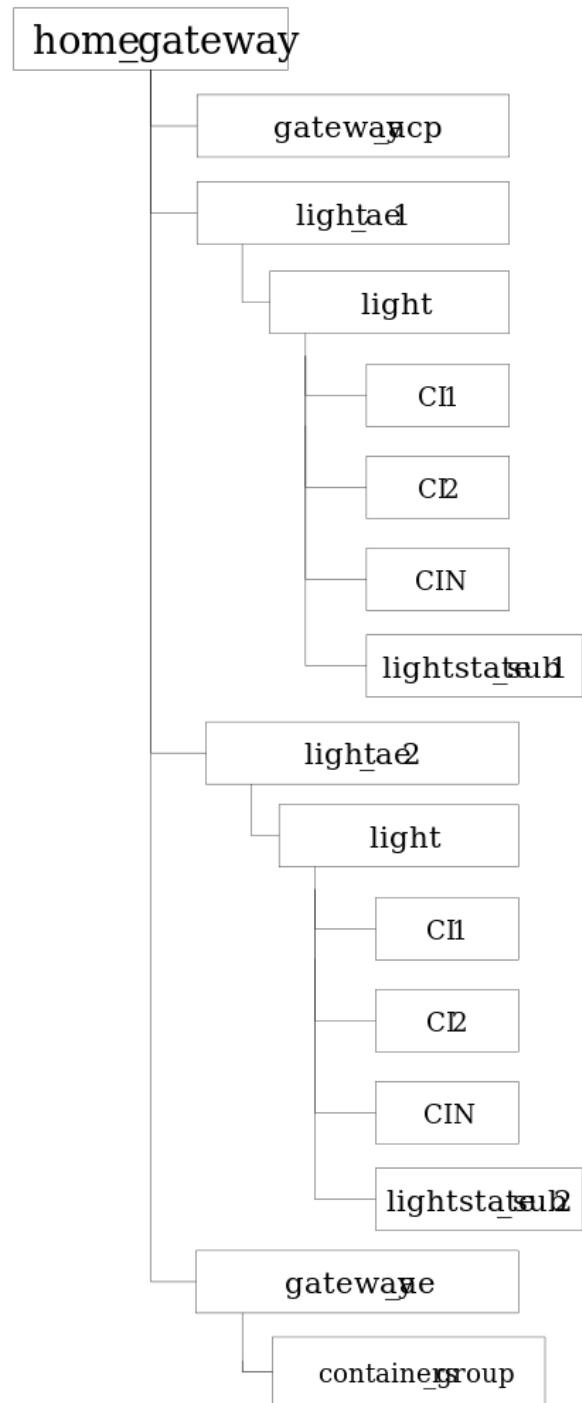
Figure 12: Figure 8.5.2-1: MN-CSE resource tree

- discovering device applications registered with the MN-CSE, and

8.6.3 Light applications (ADN-AE1 and ADN-AE2)

Each of the light applications are modelled as an ADN-AE and are responsible for

- initializing the light control device,
- registering the light devices with the MN-CSE,
- creating container resources named "*light*" with access control policy gateway_acp in the MN-CSE, respectively,
- creating subscription resources lightstate_sub1 and lightstate_sub2 under the two *light* containers, and
- creating content instance resources under containers light1 and light2 with initial light state, respectively.

8.6.4 Smartphone application (IN-AE)

The smartphone application is modelled as a IN-AE, which directly communicates with the oneM2M service platform IN-CSE and is responsible for

- initializing the smartphone light control application,
- registering the smartphone application with the IN-CSE,
- discovering the two *light* containers,
- displaying the discovered light states,
- accepting the light state modification commands from the smartphone application user,
- executing the light state modification commands for single and multiple lights.

8.7 Implementation Procedures

8.7.1 Introduction

The implementation procedures in the current use case are mapped into HTTP bindings with both XML and JSON serializations of oneM2M primitives according to the standard APIs describing the reference points Mca and Mcc, as defined in oneM2M TS-0001 [i.2], oneM2M TS-0004 [i.3], the HTTP binding TS-0009 [i.4].

In addition, *short names* for the representation of the resources and attributes are used in the implementation procedures.

8.7.2 MN-CSE registration

The implementation starts with the registration of MN-CSE with IN-CSE as shown in the following procedure.

The following example shows the MN-CSE registration request and response using XML serialization.

HTTP Request:

```
POST /~/in-cse/server?rcn=0 HTTP/1.1
Host: in.provider.com:8080
X-M2M-Origin: /mn-cse
Content-Type: application/xml;ty=16
X-M2M-RI: incse-88643

<?xml version="1.0" encoding="UTF-8"?>
<m2m:csr xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="home_gateway">
  <csi>mn-cse&lt;/csi>
  <cb>mn.provider.com/mn-cse</cb>
  <rr>true</rr>
  <poa>http://mn.provider.com:8080</poa>
  <cst>2</cst>
  <acpi>/in-cse/acp-666957710</acpi>
</m2m:csr>
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: incse-88643
Content-Location: /in-cse/csr-299409504
```

The following example shows the MN-CSE registration request and response using HTTP with JSON serialization:

HTTP Request:

```
POST /~/in-cse/server?rcn=0 HTTP/1.1
Host: in.provider.com:8080
X-M2M-Origin: /mn-cse
Content-Type: application/json;ty=16
X-M2M-RI: incse-88643

{
  "m2m:csr": {
    "rn": "home_gateway",
    "csi": "mn-cse",
    "cb": "mn.provider.com/mn-cse",
    "rr": true,
    "poa": [
      "http://mn.provider.com:8080"
    ],
    "cst": 2,
    "acpi": [
      "/in-cse/acp-666957710"
    ]
```

```
  }
}
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: incse-88643
Content-Location: /in-cse/csr-299409504
```

8.7.3 Access control policy creation

When an access control policy resource is created, a list of one or more allowed request originators is specified in the *acor* field and the allowed operations in the *acop* field.

The value of *acop* is set to 63 which indicates that the specified originator is granted privileges to conduct *CREATE, RETRIEVE, UPDATE, DELETE, DISCOVERY,* and *NOTIFY* operations.

The creation of access control policy resource *gateway_acp* in MN-CSE is implemented in the following procedure.

The following example shows an access control policy create request and response using HTTP with XML serialization.

HTTP Request:

```
POST /~/mn-cse/home_gateway?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: Cgateway_ae
Content-Type: application/xml;ty=1
X-M2M-RI: mncse-62948

<m2m:acp xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="gateway_acp">
  <pv>
    <acr>
      <acor>Cgateway_ae Clight_ae1 Clight_ae2 /in-cse/Csmartphone_ae</acor>
      <acop>63</acop>
    </acr>
  </pv>
  <pvs>
    <acr>
      <acor>Cgateway_ae</acor>
      <acop>51</acop>
    </acr>
  </pvs>
</m2m:acp>
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-62948
Content-Location: /mn-cse/acp-805496226
```

The following example shows an access control policy create request and response using HTTP with XML serialization.

HTTP Request:

```
POST /~/mn-cse/home_gateway?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: Cgateway_ae
Content-Type: application/json;ty=1
X-M2M-RI: mncse-62948

{
  "m2m:acp": {
    "rn": "gateway\_acp",
    "pv": {
      "acr": [
        {
          "acor": [
            "Cgateway_ae",
            "Clight_ae1",
            "Clight_ae2",
            "/in-cse/Csmartphone_ae"
          ],
          "acop": 63
        }
      ]
    },
    "pvs": {
      "acr": [
        {
          "acor": [
            "Cgateway_ae"
          ],
          "acop": 51
        }
      ]
    }
  }
}
```

HTTP Response:

```
201 Created
```

```
X-M2M-RSC: 2001
X-M2M-RI: mncse-62948
Content-Location: /mn-cse/acp-805496226
```

The access control policy resource is used to grant applications the access rights
to conduct specific operations and access to specific resources. The list of
applications could be obtained with a discovery procedure using filter criteria
conditions. For more details about the discovery procedure, please go to clause
8.7.9. Here it is required that the list of applications has been discovered before
creating the access control policy resource.

8.7.4 Application entities registration

8.7.4.1 Light application ADN-AE1 The registration of ADN-AE1 with
MN-CSE is shown in the following procedure. Note that the access control policy
identifier (unstructured SP-relative resourceID) which is assigned to ADN-AE1
is `/mn-cse/acp-805496226`.

The following example shows an ADN-AE registration request and response
using HTTP with XML serialization.

HTTP Request:

```
POST /home_gateway?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: C
Content-Type: application/xml;ty=2
X-M2M-RI: mncse-92345

<xml version="1.0" encoding="UTF-8"?>
<m2m:ae xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="light_ae1">
  <api>A01.com.company.lightApp1</api>
  <rr>true</rr>
  <poa>http://192.168.0.10:9090</poa>
  <acpi>/mn-cse/acp-805496226</acpi>
</m2m:ae>
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-92345
Content-Location: /mn-cse/ae-CAE340304071
```

The following example shows an ADN-AE registration request and response
using HTTP with JSON serialization.

HTTP Request:

```
POST /home_gateway?rcn=0 HTTP/1.1
```

```
Host: mn.provider.com:8080#
X-M2M-Origin: C
Content-Type: application/json;ty=2
X-M2M-RI: mncse-92345

{
  "m2m:ae": {
    "rn": "light_ae1",
    "api": "A01.com.company.lightApp1",
    "rr": true,
    "poa": [
      "http://192.168.0.10:9090"
    ],
    "acpi": [
      "/mn-cse/acp-805496226"
    ]
  }
}
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-92345
Content-Location: /mn-cse/ae-CAE340304071
```

8.7.4.2 Light application ADN-AE2 The registration of ADN-AE2 with MN-CSE is shown in the following procedure. Note that the access control policy identifier (unstructured SP-relative resourceID) which is assigned to ADN-AE2 is /mn-cse/acp-805496226.

The following example shows an ADN-AE registration request and response using HTTP with XML serialization.

HTTP Request:

```
POST /home_gateway?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: C
Content-Type: application/xml;ty=2
X-M2M-RI: mncse-18346

<?xml version="1.0" encoding="UTF-8"?>
  <m2m:ae xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="light_ae2">
  <api>A01.com.company.lightApp2</api>
  <rr>true</rr>
  <poa>http://192.168.0.20:9090</poa>
```

```
   <acpi>/mn-cse/acp-805496226</acpi>
</m2m:ae>
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-18346
Content-Location: /mn-cse/ae-CAE340304042
```

The following example shows an ADN-AE registration request and response using HTTP with JSON serialization.

HTTP Request:

```
POST /home_gateway?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: C
Content-Type: application/json;ty=2
X-M2M-RI: mncse-18346

{
  "m2m:ae": {
    "rn": "light_ae2",
    "api": "A01.com.company.lightApp2",
    "rr": true,
    "poa": [
      "http://192.168.0.20:9090"
    ],
    "acpi": [
      "/mn-cse/acp-805496226"
    ]
  }
}
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-18346
Content-Location: /mn-cse/ae-CAE340304042
```

8.7.4.3 Home gateway application MN-AE   The registration of MN-AE with MN-CSE is shown in the following procedure. Note that the access control policy identifier (unstructured SP-relative resourceID) which is assigned to MN-AE is /mn-cse/acp-805496226.

The following example shows an MN-AE registration request and response using HTTP with XML serialization.

HTTP Request:

```
POST /home_gateway?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: C
Content-Type: application/xml;ty=2
X-M2M-RI: mncse-19347

<?xml version="1.0" encoding="UTF-8"?>
  <m2m:ae xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="gateway_ae">
  <api>A01.com.company.gatewayApp</api>
  <rr>false</rr>
  <acpi>/mn-cse/acp-805496226</acpi>
</m2m:ae>
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-19347
Content-Location: /mn-cse/ae-CAE340303271
```

The following example shows an MN-AE registration request and response using HTTP with JSON serialization.

HTTP Request:

```
POST /home_gateway?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: C
Content-Type: application/json;ty=2
X-M2M-RI: mncse-19347

{
  "m2m:ae": {
    "rn": "gateway_ae",
    "api": "A01.com.company.gatewayApp",
    "rr": false,
    "acpi": [
      "/mn-cse/acp-805496226"
    ]
  }
}
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-19347
Content-Location: /mn-cse/ae-CAE340303271
```

8.7.4.4 Smartphone application IN-AE The registration of IN-AE with IN-CSE is shown in the following procedure. Note that the access control policy identifier (unstructured SP-relative resourceID) which is assigned to IN-AE is `/in-cse/acp-666957710`.

The following example shows an IN-AE registration request and response using HTTP with XML serialization.

HTTP Request:

```
POST /server?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: C
Content-Type: application/xml;ty=2
X-M2M-RI: incse-16346

<?xml version="1.0" encoding="UTF-8"?>
<m2m:ae xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="smartphone_ae">
  <api>A01.com.company.lightControlApp</api>
  <rr>false</rr>
  <acpi>/in-cse/acp-666957710</acpi>
</m2m:ae>
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: incse-16346
Content-Location: /in-cse/ae-CAE340304178
```

The following example shows an IN-AE registration request and response using HTTP with JSON serialization.

HTTP Request:

```
POST /server?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: C
Content-Type: application/json;ty=2
X-M2M-RI: incse-16346

{
  "m2m:ae": {
    "rn": "smartphone_ae",
    "api": "A01.com.company.lightControlApp",
    "rr": false,
    "acpi": [
      "/in-cse/acp-666957710"
    ]
```

```
    }
}
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: incse-16346
Content-Location: /in-cse/ae-CAE340304178
```

8.7.5 Containers creation

8.7.5.1 Create a container of ADN-AE1 The creation of a container resource for ADN-AE1 is shown in the following procedure.

The following example shows a container create request and response using HTTP with XML serialization.

Editor note: <?xml version="1.0" encoding="UTF-8"?> header is missing

HTTP Request:

```
POST /home_gateway/light\_ae1?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: Clight_ae1
Content-Type: application/xml;ty=3
X-M2M-RI: mncse/13345

<m2m:cnt xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="light">
</m2m:cnt>
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-13345
Content-Location: /mn-cse/cnt-582759912
```

The following example shows a container create request and response using HTTP with JSON serialization.

HTTP Request:

```
POST /home\_gateway/light_ae1?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: Clight_ae1
Content-Type: application/json;ty=3
X-M2M-RI: mncse-13345

{
  "m2m:cnt": {
```

```
      "rn": "light"
  }
}
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-13345
Content-Location: /mn-cse/cnt-582759912
```

8.7.5.2 Create a container of ADN-AE2 The creation of a container resource for ADN-AE2 is shown in the following procedure.

The following example shows a container create request and response using HTTP with XML serialization.

Editor note: <?xml version="1.0" encoding="UTF-8"?> header is missing

HTTP Request:

```
POST /home_gateway/light_ae2?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: Clight_ae2
Content-Type: application/xml;ty=3
X-M2M-RI: mncse-62345

<m2m:cnt xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="light">
</m2m:cnt>
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-62345
Content-Location: /mn-cse/cnt-582769893
```

The following example shows a container create request and response using HTTP with JSON serialization.

HTTP Request:

```
POST /home_gateway/light_ae2?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: Clight_ae2
Content-Type: application/json;ty=3
X-M2M-RI: mncse-62345

{
  "m2m:cnt": {
    "rn": "light"
```

```
  }
}
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-62345
Content-Location: /mn-cse/cnt-582769893
```

8.7.6 ContentInstances creation

8.7.6.1 Create a content instance of ADN-AE1 The creation of a content instance resource under the light container of ADN-AE1 with initial content OFF is shown in the following procedure.

The following example shows a contentInstance create request and response using HTTP with XML serialization.

Editor note: <?xml version="1.0" encoding="UTF-8"?> header is missing

HTTP Request:

```
POST /home_gateway/light_ae1/light?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: Clight_ae1
Content-Type: application/xml;ty=4
X-M2M-RI: mncse-24345

<m2m:cin xmlns:m2m="http://www.onem2m.org/xml/protocols">
  <cnf>text/plain:0&lt;/cnf>
  <con>OFF</con>
</m2m:cin>
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-24345
Content-Location: /mn-cse/cin-394798749
Content-Type: application/xml
```

The following example shows a contentInstance create request and response using HTTP with JSON serialization.

HTTP Request:

```
POST /home_gateway/light_ae1/light?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: Clight_ae1
Content-Type: application/json;ty=4
```

```
X-M2M-RI: mncse-24345

{
  "m2m:cin": {
    "cnf": "text/plains:0",
    "con": "OFF"
  }
}
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-24345
Content-Location: /mn-cse/cin-394798749
Content-Type: application/json
```

8.7.6.2 Create a content instance of ADN-AE2   The creation of a content instance resource under the light container of ADN-AE2 with initial content OFF is shown in the following procedure.

The following example shows a contentInstance create request and response using HTTP with XML serialization.

Editor note: <?xml version="1.0" encoding="UTF-8"?> header is missing

HTTP Request:

```
POST /home_gateway/light_ae2/light?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: Clight_ae1
Content-Type: application/xml;ty=4
X-M2M-RI: mncse-22345

<m2m:cin xmlns:m2m="http://www.onem2m.org/xml/protocols">
<cnf>text/plain:0/cnf>
  <con>OFF</con>
</m2m:cin>
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-22345
Content-Location: /mn-cse/cin-256599578
```

The following example shows a contentInstance create request and response using HTTP with JSON serialization.

HTTP Request:

32

```
POST /home_gateway/light_ae2/light?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: Clight_ae1
Content-Type: application/json;ty=4
X-M2M-RI: mncse-22345

{
  "m2m:cin": {
    "cnf": "text/plains:0",
    "con": "OFF"
  }
}
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-22345
Content-Location: /mn-cse/cin-256599578
```

8.7.7 Group creation

The creation of a group resource by the MN-AE is shown in the following procedure. The group resource is created with two initial member ids of the light container resources..

The following example shows a group create request and response using HTTP with XML serialization.

HTTP Request:

```
POST /home_gateway/gateway_ae?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: Cgateway_ae
Content-Type: application/xml;ty=9
X-M2M-RI: mncse-76905

<?xml version="1.0" encoding="UTF-8"?>
<m2m:grp xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="containers_grp">
  <mt>3</mt>
  <mid>/mn-cse/cnt-582759912 /mn-cse/cnt-582769893</mid>
  <mnm>10&lt;/mnm>
</m2m:grp>
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
```

```
X-M2M-RI: mncse-76905
Content-Location: /mn-cse/grp-977978327
```

The following example shows a group create request and response using HTTP with JSON serialization.

HTTP Request:

```
POST /home_gateway/gateway_ae?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: Cgateway_ae
Content-Type: application/json;ty=9
X-M2M-RI: mncse-76905

{
  "m2m:grp": {
    "rn":"containers_grp",
    "mt": 3,
    "mid": [
      "/mn-cse/cnt-582759912",
      "/mn-cse/cnt-582769893"
    ],
    "mnm": 10
  }
}
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-76905
Content-Location: /mn-cse/grp-977978327
```

8.7.8 Subscriptions creation

8.7.8.1 Subscription to the content instance of ADN-AE1 When a subscription resource is created, the *notification content type* (short for *nct*) parameter is set to a value 1 to indicate that all attributes of the subscribed resource will be notified to the subscriber.

ADN-AE1 creates a subscription resource including the notification URI set to the resource identifier of ADN-AE1 so that the ADN-AE1 will get notified whenever a content instance child resource is created in the container . The corresponding subscription create request is shown in the following procedure.

The following example shows a subscription create request and response using HTTP with XML serialization.

HTTP Request:

```
POST /home_gateway/light_ae1/light?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: Clight_ae1
Content-Type: application/xml;ty=23
X-M2M-RI: mncse-67891

<?xml version="1.0" encoding="UTF-8"?>
<m2m:sub xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="lightstate_sub1">
  <enc>
    <net>3</net>
  </enc>
  <nu>Clight_ae1</nu>
  <nct>1</nct>
</m2m:sub>
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-67891
Content-Location: /mn-cse/sub-856593979
```

The following example shows a subscription create request and response using HTTP with JSON serialization.

HTTP Request:

```
POST /home_gateway/light_ae1/light?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: Clight_ae1
Content-Type: application/json;ty=23
X-M2M-RI: mncse-67891

{
  "m2m:sub": {
    "rn": "lightstate_sub1",
    "enc": {
      "net":<a href="#_ref_3">[
        3
      ]</a>
    },
    "nu": ["Clight_ae1"
    ],
    "nct": 1
  }
}
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-67891
Content-Location: /mn-cse/sub-856593979
```

8.7.8.2 Subscription to the content instance of ADN-AE2 When a
subscription resource is created, the *notification content type* (short for *nct*)
parameter is set to a value 1 to indicate that all attributes of the subscribed
resource will be notified to the subscriber.

ADN-AE1 creates a subscription resource including the notification URI set
to the resource identifier of ADN-AE1 so that the ADN-AE1 will get notified
whenever a content instance child resource is created in the container. The
corresponding subscription create request is shown in the following procedure.

When a subscription resource is created, the *notification content type* (short for
nct) parameter is set to value 1 to indicate that all attributes of the subscribed
resource will be notified to the subscriber.

ADN-AE2 creates a subscription resource including the notification URI set
to the resource identifier of ADN-AE2 so that the ADN-AE2 will get notified
whenever a content instance child resource is created in the container. The
corresponding subscription create request is shown in the following procedures.

The following example shows a subscription create request and response using
HTTP with XML serialization.

HTTP Request:

```
POST /home_gateway/light_ae2/light?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: Clight_ae2
Content-Type: application/xml;ty=23
X-M2M-RI: mncse-29387

<?xml version="1.0" encoding="UTF-8"?>
<m2m:sub xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="lightstate_sub2">
  <enc>
    <net>3</net>
  </enc>
  <nu>Clight_ae2</nu>
  <nct>1</nct>
</m2m:sub>
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
```

```
X-M2M-RI: mncse-29387
Content-Location: /mn-cse/sub-856463728
```

The following example shows a subscription create request and response using HTTP with JSON serialization.

HTTP Request:

```
POST /home_gateway/light_ae2/light?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: Clight_ae2
Content-Type: application/json;ty=23
X-M2M-RI: mncse-29387

{
  "m2m:sub": {
    "rn": "lightstate_sub2",
    "enc": {
      "net":<a href="#_ref_3">[
        3
      ]</a>
    },
    "nu": ["Clight_ae2"
    ],
    "nct": 1
  }
}
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-29387
Content-Location: /mn-cse/sub-856463728
```

8.7.9 Discovery

8.7.9.1 Introduction The discovery functionality in oneM2M is implemented using a RETRIEVE operation along with one or multiple filter criteria parameters.

In order to enable the retrieve operation for resource discovery, parameter *filterUsage* (short for *fu)* is included in the RETRIEVE request as a query string.

In addition, parameter *resource type* (short for *rty)* is used as a *filterCriteria* condition for the discovery of single light and group light members. The parameter *discovery result type* (short for *drt*) is set to 2 to indicate that the format of elements of URIList is unstructured. The detailed discovery procedures are presented in clauses 8.7.9.2 and 8.7.9.3.

8.7.9.2 Discovery of single light registered with MN-CSE The discovery of containers for each light registered with the MN-CSE by the smartphone AE is shown in the following procedure.

If the discovery response is preferred to be returned with a XML representation, the HTTP request message is sent as following example.

HTTP Request:

```
GET /~/mn-cse/home_gateway?fu=1&rty=3&drt=2 HTTP/1.1
Host: in.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
X-M2M-RI: mncse-99882
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-99882
X-M2M-CNST: 2
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<m2m:uril xmlns:m2m="http://www.onem2m.org/xml/protocols">
  /mn-cse/cnt-582759912
  /mn-cse/cnt-582769893
</m2m:uril>
```

If the discovery response is preferred to be returned with a JSON representation, the HTTP request message is sent as following example.

HTTP Request:

```
GET /~/mn-cse/home_gateway?fu=1&rty=3&drt=2 HTTP/1.1
Host: in.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
X-M2M-RI: mncse-99882
Accept: application/json
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-99882
X-M2M-CNST: 2
Content-Type: application/json

{
  "m2m:uril": [
    "/mn-cse/cnt-582759912",
```

```
      "/mn-cse/cnt-582769893"
   ]
}
```

The smartphone application retrieves a list of URIs representing containers registered with MN-CSE from the response message, e.g. /mn-cse/cnt-582759912 which is the URI of container created in ADN-AE1. The retrieved URIs of the discovered containers are then used for the group member update operation.

8.7.9.3 Discovery of groups located in MN-CSE   The discovery of groups located in MN-CSE by the smartphone AE is shown in the following procedures.

If the discovery response is preferred to be returned with a XML representation, the HTTP request message is sent as following example:

HTTP Request:

```
GET /~/mn-cse/home_gateway?fu=1&rty=9&drt=2 HTTP/1.1
Host: in.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
X-M2M-RI: mncse-15001
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-15001
X-M2M-CNST: 2
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<m2m:uril xmlns:m2m="http://www.onem2m.org/xml/protocols">
   /mn-cse/grp-977978327
</m2m:uril>
```

If the discovery response is preferred to be returned with a JSON representation, the HTTP request message is sent as following example.

HTTP Request:

```
GET /~/mn-cse/home_gateway?fu=1&rty=9&drt=2 HTTP/1.1
Host: in.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
X-M2M-RI: mncse-15001
Accept: application/json
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
```

```
X-M2M-RI: mncse-15001
X-M2M-CNST: 2Content-Type: application/json

{
  "m2m:uril": [
    "/mn-cse/grp-977978327"
  ]
}
```

The smartphone application retrieves a list of URIs representing group resources located in MN-CSE from the response message, e.g. `/mn-cse/grp-977978327` which is the URI of the group resource. The retrieved URIs of the discovered group resource are then used for the group member update operation.

8.7.10 Latest content instances retrieval

8.7.10.1 Introduction The smartphone application can retrieve the latest light states via sending a RETRIEVE request targeting a container's *la* (short for *latest*) virtual resource.

The smartphone application can also retrieve a group of latest light states via sending a RETRIEVE request targeting the group *fanOutPoint* virtual resource.

8.7.10.2 Retrieve the latest content instance of ADN-AE1 The latest content instance of the *light* container resource for ADN-AE1 can be retrieved by the following procedure.

If the response is preferred to be returned with a XML representation, the following is a HTTP request message example.

HTTP Request:

```
GET /~/mn-cse/home_gateway/light_ae1/light/la HTTP/1.1
Host: in.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
X-M2M-RI: mncse-11223
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-11223
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<m2m:cin xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="cin-394798749">
  <ty>4</ty>
  <ri>cin-394798749</ri>
```

```
  <pi>cnt-181049109</pi>
  <ct>20150925T045938</ct>
  <lt>20150925T045938</lt>
  <et>20151107T154802</et>
  <st>0</st>
  <cnf>text/plain:0</cnf>
  <cs>3</cs>
  <con>OFF</con>
</m2m:cin>
```

If the response is preferred to be returned with a JSON representation, the following is a HTTP request message example.

Editor note: correct the "lat" typo below.

```
GET /~/mn-cse/home_gateway/light_ae1/light/lat HTTP/1.1
Host: in.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
X-M2M-RI: mncse-11223
Accept: application/json
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-11223
Content-Type: application/json

{
  "m2m:cin": {
    "ty": 4,
    "ri": "cin-394798749",
    "pi": "cnt-181049109",
    "ct": "20150925T045938",
    "lt": "20150925T045938",
    "et": "20151107T154802",
    "st": 0,
    "cnf": "text/plain:0",
    "cs": 3,
    "con": "OFF"
  }
}
```

8.7.10.3 Retrieve the latest content instance of ADN-AE2   The latest content instance of the *light* container resource for ADN-AE2 can be retrieved by the following procedures.

If the response is preferred to be represented in XML, the following is a HTTP

request message example.

HTTP Request:

```
GET /~/mn-cse/home_gateway/light_ae2/light/la HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
X-M2M-RI: mncse-22336
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-22336
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<m2m:cin xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="cin-256599578">
  <ty>4</ty>
  <ri>cin-256599578</ri>
  <pi>cnt-790965889</pi>
  <ct>20150925T050515</ct>
  <lt>20150925T050515</lt>
  <et>20151107T154802</et>
  <st>0</st>
  <cnf>text/plain:0</cnf>
  <cs>3</cs>
  <con>OFF</con>
</m2m:cin>
```

If the response is preferred be returned in representation of JSON, the following is a HTTP request message example.

HTTP Request:

```
GET /~/mn-HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
X-M2M-RI: mncse-22336
Accept: application/json
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-22336
Content-Type: application/json
```

```
{
  "m2m:cin": {
    "ty": 4,
    "ri": "cin-256599578",
    "pi": "cnt-790965889",
    "ct": "20150925T050515",
    "lt": "20150925T050515",
    "et": "20151107T154802",
    "st": 0,
    "cnf": "text/plain:0",
    "cs": 3,
    "con": "OFF"
  }
}
```

8.7.10.4 Retrieve a group of latest content instances for all light states

A group of latest content instances can be retrieved via sending a RETRIEVE request targeting the group *fanOutPoint* virtual resource and appending *latest* as shown in the following procedures.

If the response is preferred to be returned with a XML representation, the following is a HTTP request message example.

HTTP Request:

```
GET /~/mn-cse/home_gateway/containers_grp/fopt/la HTTP/1.1
Host: in.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
X-M2M-RI: mncse-55667
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-55667
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<m2m:agr xmlns:m2m="http://www.onem2m.org/xml/protocols">
  <m2m:rsp>
    <rsc>2000</rsc>
    <rqi>mncse-55667</rqi>
    <pc>
      <m2m:cin rn="cin-394798749">
        <ty>4</ty>
        <ri>cin-394798749</ri>
        <pi>cnt-181049109</pi>
```

```
                    <ct>20150925T045938</ct>
                    <lt>20150925T045938</lt>
                    <et>20151107T154802</et>
                    <st>0</st>
                    <cnf>text/plain:0</cnf>
                    <cs>3</cs>
                    <con>OFF</con>
                </m2m:cin>
            </pc>
            <to>/in-cse/Csmartphone_ae</to>
            <fr>/mn-cse/cnt-582759912/la</fr>
        </m2m:rsp>
        <m2m:rsp>
            <rsc>2000</rsc>
            <rqi>mncse-55667</rqi>
            <pc>
                <m2m:cin rn="cin-256599578">
                    <ty>4</ty>
                    <ri>cin-256599578</ri>
                    <pi>cnt-790965889</pi>
                    <ct>20150925T050515</ct>
                    <lt>20150925T050515</lt>
                    <et>20151107T154802</et>
                    <st>0</st>
                    <cnf>text/plain:0</cnf>
                    <cs>3</cs>
                    <con>OFF</con>
                </m2m:cin>
            </pc>
            <to>/in-cse/Csmartphone_ae</to>
            <fr>/mn-cse/cnt-582769893/la</fr>
        </m2m:rsp>
</m2m:agr>
```

If the response is preferred to be returned with a JSON representation, the following is a HTTP request message example.

HTTP Request:

```
GET /~/mn-cse/home_gateway/containers_grp/fopt/la HTTP/1.1
Host: in.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
X-M2M-RI: mncse-55667
Accept: application/json
```

HTTP Response:

```
200 OK
```

```
X-M2M-RSC: 2000
X-M2M-RI: mncse-55667
Content-Type: application/json

{
  "m2m:agr": {
    "m2m:rsp": [
      {
        "rsc": 2000,
        "rqi": "mncse-55667",
        "pc": {
          "m2m:cin": {
            "ty": 4,
            "ri": "cin-394798749",
            "pi": "cnt-181049109",
            "ct": "20150925T045938",
            "lt": "20150925T045938",
            "et": "20151107T154802",
            "st": 0,
            "cnf": "text/plain:0",
            "cs": 3,
            "con": "OFF"
          }
        },
        "to": "/in-cse/Csmartphone_ae",
        "fr": "/mn-cse/cnt-582759912/la"
      },
      {
        "rsc": 2000,
        "rqi": "mncse-55667",
        "pc": {
          "m2m:cin": {
            "ty": 4,
            "ri": "cin-256599578",
            "pi": "cnt-790965889",
            "ct": "20150925T050515",
            "lt": "20150925T050515",
            "et": "20151107T154802",
            "st": 0,
            "cnf": "text/plain:0",
            "cs": 3,
            "con": "OFF"
          }
        },
        "to": "/in-cse/Csmartphone_ae",
        "fr": "/mn-cse/cnt-582769893/la"
```

```
        }
      ]
    }
}
```

8.7.11 Light state modification

8.7.11.1 Introduction   Once the smartphone application is registered with the IN-CSE, it can be granted access to resources including containers located in the MN-CSE so that smartphone application users can send light control commands for modifying the light states.

When the user makes a change to the light state via the smartphone user interface, the smartphone application performs a new content instance creation procedure carrying the new state.

The modification of a single light state is implemented by creating a new content instance resource for the specific container with access control policy acp1 (*gateway_acp*) while the modification of all light states is implemented by creating a new content instance resource for each member of group (*containers_grp*) with access control policy acp1 (*gateway_acp*). The implementation of the latter case is to target the <fopt> virtual resource of *containers_grp* resource with a content instance create request so that the content of all members of the group is updated together.

8.7.11.2 Create a content instance under container of ADN-AE1   If the contentInstance create request body is represented in XML, the following is a HTTP request message example.

HTTP Request:

Editor note: <?xml version="1.0" encoding="UTF-8"?> header is missing

```
POST /~/mn-cse/home_gateway/light_ae1/light?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
Content-Type: application/xml;ty=4
X-M2M-RI: mncse-11123

<m2m:cin xmlns:m2m="http://www.onem2m.org/xml/protocols">
  <cnf>text/plain:0</cnf>
  <con>ON</con>
</m2m:cin>
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
```

```
X-M2M-RI: mncse-11123
Content-Location: /mn-cse/cin-789356234
```

If the contentInstance create request body is represented in JSON, the following is a HTTP request message example.

HTTP Request:

```
POST /~/mn-cse/home_gateway/light_ae1/light?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
Content-Type: application/json;ty=4
X-M2M-RI: mncse-11123

{
  "m2m:cin": {
    "cnf": "text/plains:0",
    "con": "ON"
  }
}
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-11123
Content-Location: /mn-cse/cin-789356234
```

8.7.11.3 Create a content instance under container of ADN-AE2 If the contentInstance create request body is represented in XML, the following is an HTTP request message the example.

Editor note: <?xml version="1.0" encoding="UTF-8"?> header is missing

HTTP Request:

```
POST /~/mn-cse/home_gateway/light_ae2/light?rcn=0 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
Content-Type: application/xml;ty=4
X-M2M-RI: mncse-12222

<m2m:cin xmlns:m2m="http://www.onem2m.org/xml/protocols">
  <cnf>text/plain:0</cnf>
  <con>ON</con>
</m2m:cin>
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-12222
Content-Location: /mn-cse/cin-237896783
```

If the contentInstance create request body is represented in JSON, the following is a HTTP request message example.

HTTP Request:

```
POST /~/mn-cse/home_HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
Content-Type: application/json;ty=4
X-M2M-RI: mncse-12222

{
  "m2m:cin": {
    "cnf": "text/plains:0",
    "con": "ON"
  }
}
```

HTTP Response:

```
201 Created
X-M2M-RSC: 2001
X-M2M-RI: mncse-12222
Content-Location: /mn-cse/cin-237896783
```

8.7.11.4 Update the state of all lights using group fanout   If the fanOut-Point request body is represented in XML, the following is a HTTP request message example:.

HTTP Request:

```
POST /~/mn-cse/home_gateway/gateway_ae/containers_grp/fopt HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
Content-Type: application/xml;ty=4
X-M2M-RI: mncse-33344

<xml version="1.0" encoding="UTF-8"?>
<m2m:cin xmlns:m2m="http://www.onem2m.org/xml/protocols">
  <cnf>text/plain:0</cnf>
  <con>ON</con>
</m2m:cin>
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-33344
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<m2m:agr xmlns:m2m="http://www.onem2m.org/xml/protocols">
  <m2m:rsp>
    <rsc>2001</rsc>
    <rqi>mncse-33344</rqi>
    <pc>
      <m2m:cin rn="cin-479874939">
        <ty>4</ty>
        <ri>cin-479874939</ri>
        <pi>cnt-181049109</pi>
        <ct>20151025T045938</ct>
        <lt>20151025T045938</lt>
        <et>20151207T154802</et>
        <st>0</st>
        <cs>2</cs>
      </m2m:cin>
    </pc>
    <to>/in-cse/Csmartphone_ae</to>
    <fr>/mn-cse/cnt-582759912</fr>
  </m2m:rsp>
  <m2m:rsp>
    <rsc>2001</rsc>
    <rqi>mncse-33344</rqi>
    <pc>
      <m2m:cin rn="cin-659957825">
        <ty>4</ty>
        <ri>cin-659957825</ri>
        <pi>cnt-790965889</pi>
        <ct>20151025T045938</ct>
        <lt>20151025T045938</lt>
        <et>20151207T154802</et>
        <st>0</st>
        <cs>2</cs>
      </m2m:cin>
    </pc>
    <to>/in-cse/Csmartphone_ae</to>
    <fr>/mn-cse/cnt-582769893</fr>
  </m2m:rsp>
</m2m:agr>
```

If the fanOutPoint request body is represented in JSON, the following is a HTTP

request message example.

HTTP Request:

```
POST /~/mn-cse/home_gateway/gateway_ae/containers_grp/fopt HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
Content-Type: application/json;ty=4
X-M2M-RI: mncse-33344

{
  "m2m:cin": {
    "cnf": "text/plains:0",
    "con": "ON"
  }
}
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-33344
Content-Type: application/json

{
  "m2m:agr": {
    "m2m:rsp": [
      {
        "rsc": 2001,
        "rqi": "mncse-33344",
        "pc": {
          "m2m:cin": {
            "ty": 4,
            "ri": "cin-479874939",
            "pi": "cnt-181049109",
            "ct": "20151025T045938",
            "lt": "20151025T045938",
            "et": "20151207T154802",
            "st": 0,
            "cs": 2
          }
        },
        "to": "/in-cse/Csmartphone_ae",
        "fr": "/mn-cse/cnt-582759912"
      },
      {
        "rsc": 2001,
        "rqi": "mncse-33344",
```

```
      "pc": {
        "m2m:cin": {
          "ty": 4,
          "ri": "cin-659957825",
          "pi": "cnt-790965889",
          "ct": "20151025T045938",
          "lt": "20151025T045938",
          "et": "20151207T154802",
          "st": 0,
          "cs": 2
        }
      },
      "to": "/in-cse/Csmartphone_ae",
      "fr": "/mn-cse/cnt-582769893"
    }
  ]
  }
}
```

8.7.12 Notifications

8.7.12.1 Introduction   Each time a content instance is created under a container of an ADN-AE, then a notification containing the whole created content instance is posted to the targeted subscriber i.e. ADN-AE1 or ADN-AE2, that can actuate the light with the new state received in the notification.

8.7.12.2 Post a notification to ADN-AE1   If the notification request body is represented in XML, the following is a HTTP request message example.

HTTP Request:

```
POST / HTTP/1.1
Host: 192.168.0.10:9090
X-M2M-Origin: /mn-cse
X-M2M-RI: notif-12345
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<m2m:sgn xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="cin-394798749">
  <nev>
    <rep>
      <m2m:cin>
        <ty>4</ty>
        <ri>cin-394798749</ri>
        <pi>cnt-790965889</pi>
        <ct>20150925T050534</ct>
        <lt>20150925T050534</lt>
```

```
        <et>20151107T154802</et>
        <st>0</st>
        <cnf>text/plain:0</cnf>
        <cs>3</cs>
        <con>ON</con>
      </m2m:cin>
    </rep>
    <net>3</net>
  </nev>
  <sur>/mn-cse/sub-856593979</sur>
</m2m:sgn>
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: notif-12345
```

If the notification request body is represented in JSON, the following is a HTTP request message example.

HTTP Request:

```
POST / HTTP/1.1
Host: 192.168.0.10:9090
X-M2M-Origin: /mn-cse
X-M2M-RI: notif-12345

Content-Type: application/json

{
  "m2m:sgn": {
    "nev": {
      "rep": {
        "m2m:cin": {
          "ty": 4,
          "ri": "cin-394798749",
          "pi": "cnt-790965889",
          "ct": "20150925T050534",
          "lt": "20150925T050534",
          "et": "20151107T154802",
          "st": 0,
          "cnf": "text/plain:0",
          "cs": 3,
          "con": "ON"
        }
      },
      "net": 3
```

```
    },
    "sur": "/mn-cse/sub-856593979"
  }
}
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: notif-12345
```

8.7.12.3 Post a notification to ADN-AE2 If the notification request body
is represented in XML, the following is a HTTP request message example.

HTTP Request:

```
POST / HTTP/1.1
Host: 192.168.0.20:9090
X-M2M-Origin: /mn-cse
X-M2M-RI: notif-12346
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<m2m:sgn xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="cin-256599578">
  <nev>
    <rep>
      <m2m:cin>
        <ty>4</ty>
        <ri>cin-256599578</ri>
        <pi>cnt-790965889</pi>
        <ct>20150925T050623</ct>
        <lt>20150925T050623</lt>
        <et>20151107T154802</et>
        <st>0</st>
        <cnf>text/plain:0</cnf>
        <cs>3</cs>
        <con>ON</con>
      </m2m:cin>
    </rep>
    <net>3</net>
  </nev>
  <sur>/mn-cse/sub-856463728</sur>
</m2m:sgn>
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
```

```
X-M2M-RI: notif-12346
```

If the notification request body is represented in JSON, the following is a HTTP request message example.

HTTP Request:

```
POST / HTTP/1.1
Host: 192.168.0.20:9090
X-M2M-Origin: /mn-cse
X-M2M-RI: notif-12346
Content-Type: application/json

{
  "m2m:sgn": {
    "nev": {
      "rep": {
        "m2m:cin": {
          "ty": 4,
          "ri": "cin-256599578",
          "pi": "cnt-790965889",
          "ct": "20150925T050623",
          "lt": "20150925T050623",
          "et": "20151107T154802",
          "st": 0,
          "cnf": "text/plain:0",
          "cs": 3,
          "con": "ON"
        }
      },
      "net": 3
    },
    "sur": "/mn-cse/sub-856463728"
  }
}
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: notif-12346
```

Annex A: Reading Resources

Annex A.1 Introduction

The information of resources created during each call flow of current use case can be retrieved via smartphone application IN-AE initiates a RETRIEVE request

to the target resources as following.

Annex A.2 CSE resources

Annex A.2.1 IN-CSE

HTTP Request:

```
GET /~/in-cse/server HTTP/1.1
Host: in.provider.com:8080
X-M2M-Origin: /mn-cse/Cgateway_ae
X-M2M-RI: incse-12345
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: incse-12345
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<m2m:cb xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="server">
  <ty>5</ty>
  <ri>/in-cse</ri>
  <ct>20150925T045938</ct>
  <lt>20150925T045938</lt>
  <acpi>/in-cse/acp-666957710</acpi>
  <cst>1</cst>
  <lt;csi>in-cse</csi>
  <srt>1 2 3 4 5 9 14 15 16 17 23</srt>
  <poa>http://in.provider.com:8080/</poa>
</m2m:cb>
```

Annex A.2.2 MN-CSE

HTTP Request:

```
GET /~/in-cse/server/home_gateway HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
X-M2M-RI: incse-12346
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: incse-12346
```

```
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<m2m:csr xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="home_gateway">
  <ty>16</ty>
  <ri>csr-299409504</ri>
  <pi>/in-cse</pi>
  <ct>20150925T045938</ct>
  <lt>20150925T045938</lt>
  <et>20171005T105550</et>
  <acpi>/in-cse/acp-666957710</acpi>
  <poa>http://mn.provider.com:8080/</poa>
  <cb>mn.provider.com/mn-cse</cb>
  <csi>/mn-cse</csi>
  <rr>true</rr>
</m2m:csr>
```

Annex A.3 Gateway device application MN-AE

HTTP Request:

```
GET /~/mn-cse/home_gateway/gateway_ae HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
X-M2M-RI: mncse-12347
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-12347
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<m2m:ae xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="gateway_ae">
  <ty>2</ty>
  <ri>ae-CAE340303271</ri>
  <pi>/mn-cse</pi>
  <ct>20150925T052438</ct>
  <lt>20150925T052438</lt>
  <et>20171005T105550</et>
  <acpi>/mn-cse/acp-805496226</acpi>
  <api>A01.com.company.gatewayApp</api>
  <aei>CAE340303271</aei>
  <rr>false</rr>
</m2m:ae>
```

Annex A.4. Light device applications

Annex A.4.1 ADN-AE1

HTTP Request:

```
GET /~/mn-cse/home_gateway/light_ae1 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
X-M2M-RI: mncse-12348
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-12348
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<m2m:ae xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="light_ae1">
  <ty>2</ty>
  <ri>ae-CAE340304071</ri>
  <pi>/mn-cse</pi>
  <ct>20150925T052455</ct>
  <lt>20150925T052455</lt>
  <et>20171005T105550</et>
  <acpi>/mn-cse/acp-805496226</acpi>
  <api>A01.com.company.lightApp1</api>
  <aei>CAE340304071</aei>
  <rr>true</rr>
</m2m:ae>
```

Annex A.4.2 ADN-AE2

HTTP Request:

```
GET /~/mn-cse/home_gateway/light_ae2 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /mn-cse/Cgateway_ae
X-M2M-RI: mncse-12348
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-12348
Content-Type: application/xml
```

```xml
<?xml version="1.0" encoding="UTF-8"?>
<m2m:ae xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="light_ae2">
  <ty>2</ty>
  <ri>ae-CAE340304042</ri>
  <pi>/mn-cse</pi>
  <ct>20150925T052542</ct>
  <lt>20150925T052542</lt>
  <et>20171005T105550</et>
  <acpi>/mn-cse/acp-805496226</acpi>
  <api>A01.com.company.lightApp2</api>
  <aei>CAE340304042</aei>
  <rr>true</rr>
</m2m:ae>
```

Annex A.5 Smartphone application IN-AE

HTTP Request:

```
GET /~/in-cse/server/smartphone_ae HTTP/1.1
Host: in.provider.com:8080
X-M2M-Origin: /mn-cse/Cgateway_ae
X-M2M-RI: incse-12349
Accept: application/xml
```

Editor note: The response format here is different from the others examples

Resonse status: 200 OK X-M2M-RSC: 2000 X-M2M-RI: incse-12349 Content-Type: application/xml

Resonse message:

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<m2m:ae xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="smartphone_ae">
  <ty>2</ty>
  <ri>ae-CAE340304178</ri>
  <pi>/in-cse</pi>
  <ct>20150925T052622</ct>
  <lt>20150925T052622</lt>
  <et>20171005T105550</et>
  <acpi>/in-cse/acp-666957710</acpi>
  <api>A01.com.company.lightControlApp</api>
  <aei>CAE340304178</aei>
  <rr>false</rr>
</m2m:ae>
```

Annex A.6 Access control policy

HTTP Request:

```
GET /~/mn-cse/home_gateway/gateway_acp HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /mn-cse/Cgateway_ae
X-M2M-RI: mncse-12350
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-12350
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<m2m:acp xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="gateway_acp">
  <ty>1</ty>
  <ri>acp-805496226</ri>
  <pi>/mn-cse</pi>
  <ct>20150925T050238</ct>
  <lt>20150925T050238</lt>
  <et>20171005T105550</et>
  <pv>
    <acr>
      <acor>/in-cse/home_gateway /mn-cse/Cgateway_ae /mn-cse/Clight_ae1 /mn-cse/Clight_ae2 /
      <acop>63</acop>
    </acr>
  </pv>
  <pvs>
    <acr>
      <acor>/in-cse/home_gateway /mn-cse/Cgateway_ae /mn-cse/Clight_ae1 /mn-cse/Clight_ae2 /
      <acop>51</acop>
    </acr>
  </pvs>
</m2m:acp>
```

Annex A.7 Containers

Annex A.7.1 Container under ADN-AE1

HTTP Request:

```
GET /~/mn-cse/home_gateway/light_ae1/light HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /mn-cse/Cgateway_ae
X-M2M-RI: mncse-12351
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-12351
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<m2m:cnt xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="light">
    <ty>3</ty>
    <ri>cnt-582759912</ri>
    <pi>/mn-cse/ae-CAE340304071</pi>
    <ct>20150925T052955</ct>
    <lt>20150925T052955</lt>
    <et>20171005T105550</et>
    <acpi>/mn-cse/acp-805496226</acpi>
    <st>0</st>
    <cni>1</cni>
    <cbs>3</cbs>
</m2m:cnt>
```

Annex A.7.2 Container under ADN-AE2

HTTP Request:

```
GET /~/mn-cse/home_gateway/light_ae2/light HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /mn-cse/Cgateway_ae
X-M2M-RI: mncse-12352
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-12352
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<m2m:cnt xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="light">
    <ty>3</ty>
    <ri>cnt-582769893</ri>
    <pi>/mn-cse/ae-CAE340304042</pi>
    <ct>20150925T053135</ct>
    <lt>20150925T053135</lt>
    <et>20171005T105550</et>
    <acpi>/mn-cse/acp-805496226</acpi>
    <st>0</st>
    <cni>1</cni>
```

```
    <cbs>3</cbs>
</m2m:cnt>
```

Annex A.8 ContentInstances

Annex A.8.1 Latest contentInstance in ADN-AE1

HTTP Request:

```
GET /~/mn-cse/home_gateway/light_ae1/light/la HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /mn-cse/Cgateway_ae
X-M2M-RI: mncse-12353
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-12353
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<m2m:cin xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="cin-394798749">
    <ty>4</ty>
    <ri>cin-394798749</ri>
    <pi>/mn-cse/cnt-582759912</pi>
    <ct>20150925T053225</ct>
    <lt>20150925T053225</lt>
    <et>20171005T105550</et>
    <st>0</st>
    <cnf>text/plain:0</cnf>
    <cs>2</cs>
    <con>ON</con>
</m2m:cin>
```

Annex A.8.2 Latest contentInstance in ADN-AE2

HTTP Request:

```
GET /~/mn-cse/home_gateway/light_ae2/light/la HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /mn-cse/Cgateway_ae
X-M2M-RI: mncse-12354
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
```

```
X-M2M-RI: mncse-12354
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<m2m:cin xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="cin-256599578">
    <ty>4</ty>
    <ri>cin-256599578</ri>
    <pi>/mn-cse/cnt-582769893</pi>
    <ct>20150925T053425</ct>
    <lt>20150925T053425</lt>
    <et>20171005T105550</et>
    <st>0</st>
    <cnf>text/plain:0</cnf>
    <cs>2</cs>
    <con>ON</con>
</m2m:cin>
```

Annex A.9 Subscriptions

Annex A.9.1 Subscription to container in ADN-AE1

HTTP Request:

```
GET /~/mn-cse/home_gateway/light_ae1/light/lightstate_sub1 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /mn-cse/Cgateway_ae
X-M2M-RI: mncse-12355
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-12355
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<m2m:sub xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="lightstate_sub1">
    <ty>23</ty>
    <ri>sub-856593979</ri>
    <pi>/mn-cse/cnt-582759912</pi>
    <ct>20150926T052955</ct>
    <lt>20150926T052955</lt>
    <et>20171005T105550</et>
    <acpi>/mn-cse/acp-805496226</acpi>
    <enc>
        <net>3</net>
    </enc>
```

```
    <nu>/mn-cse/ae-CAE340304071</nu>
    <nct>1</nct>
</m2m:sub>
```

Annex A.9.2 Subscription to container in ADN-AE2

HTTP Request:

```
GET /~/mn-cse/home_gateway/light_ae2/light/lightstate_sub2 HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /mn-cse/Cgateway_ae
X-M2M-RI: mncse-12356
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-12356
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<m2m:sub xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="lightstate_sub2">
    <ty>23</ty>
    <ri>sub-856463728</ri>
    <pi>/mn-cse/cnt-582759912</pi>
    <ct>20150926T053055</ct>
    <lt>20150926T053055</lt>
    <et>20171005T105550</et>
    <acpi>/mn-cse/acp-805496226</acpi>
    <enc>
        <net>3</net>
    </enc>
    <nu>/mn-cse/ae-CAE340304042</nu>
    <nct>1</nct>
</m2m:sub>
```

Annex A.10 Groups

Annex A.10.1 Group1

HTTP Request:

```
GET /~/mn-cse/home_gateway/gateway_ae/containers_grp HTTP/1.1
Host: mn.provider.com:8080
X-M2M-Origin: /in-cse/Csmartphone_ae
X-M2M-RI: mncse-12357
Accept: application/xml
```

HTTP Response:

```
200 OK
X-M2M-RSC: 2000
X-M2M-RI: mncse-12357
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<m2m:grp xmlns:m2m="http://www.onem2m.org/xml/protocols" rn="containers_grp">
    <ty>9</ty>
    <ri>grp-977978327</ri>
    <pi>/mn-cse/ae-CAE340303271</pi>
    <ct>20151004T045954</ct>
    <lt>20151004T045954</lt>
    <et>20171005T105550</et>
    <acpi>/mn-cse/acp-805496226 /mn-cse/acp-805496226</acpi>
    <mt>3</mt>
    <cnm>2</cnm>
    <mnm>10</mnm>
    <mid>/mn-cse/cnt-582759912 /mn-cse/cnt-582769893</mid>
    <mtv>true</mtv>
    <csy>1</csy>
    <fopt>/mn-cse/grp-977978327/fopt</fopt>
</m2m:grp>
```

History

Publication history	Publication history	Publication history

Draft history (to be removed on publication)	Draft history (to be removed on publication)	Draft history (to be removed on publication)
V2.0.0	Fev 2017	Initial release 2 version based on TR-0025 V1.2.0 Implemented the agreed contribution from TST#27 - TST-2017-0052-TR-0025_update

Draft history (to be removed on publication)	**Draft history** (to be removed on publication)	**Draft history** (to be removed on publication)
V2.0.1	24 June 2017	Implemented the agreed contributions from TST#29 - TST-2017-0117R01-CR-TR-0025_HTTP_Host_Header_correction_Rel_2 - TST-2017-0128-CR-TR-0025_Notification_correction_Rel_2
V2.0.2	23 March 2018	Implemented the agreed contributions from TST#34 - TST-2018-0043R01-TR-0025_subscription_error_correction
V2.0.3	06 June 2019	Implemented the agreed contributions from TDE#40 - TDE-2019-0076-Missing_resource_types_in_Content-Type_header_fields