|  |
| --- |
|  |

|  |
| --- |
| CHANGE REQUEST |
| Meeting ID:\* | RDM#50 |
| Source:\* | Cyrille Bareau, Orange, cyrille.bareau@orange.comMarianne Mohali, Orange, marianne.mohali@orange.com |
| Date:\* | 2021-05-27 |
| Reason for Change/s:\* | See the introduction. |
| CR against: Release\* | Release 4 |
| CR against: WI\* | [x]  Active WI-0099[ ]  MNT maintenance / < Work Item number(optional)>Is this a mirror CR? Yes [ ]  No [ ] mirror CR number: (Note to Rapporteur - use latest agreed revision)[ ]  STE Small Technical Enhancements / < Work Item number (optional)>Only ONE of the above shall be ticked |
| CR against: TS/TR\* | TR-0067 v0.1.0 |
| Clauses \* | Annex A, Annex C |
| Type of change: \* | [ ]  Editorial change[ ]  Bug Fix or Correction[x]  Change to existing feature or functionality[ ]  New feature or functionalityOnly ONE of the above shall be ticked |
| Other TS/TR(s) impacted | N/A |
| Post Freeze checking:\* | This CR contains only essential changes and corrections? YES [x]  NO [ ] This CR may break backwards compatibility with the last approved version of the TS? YES [ ]  NO [x]  |
| Template Version: January 2020 (do not modify) |

**oneM2M Notice**

The document to which this cover statement is attached is submitted to oneM2M. Participation in, or attendance at, any activity of oneM2M, constitutes acceptance of and agreement to be bound by terms of the Working Procedures and the Partnership Agreement, including the Intellectual Property Rights (IPR) Principles Governing oneM2M Work found in Annex 1 of the Partnership Agreement.

GUIDELINES for Change Requests:

Provide an informative introduction containing the problem(s) being solved, and a summary list of proposals.

Each CR should contain changes related to only one particular issue/problem.

If this is a correction, and the change applies to previous releases, a separate “mirror CR” should be posted at the same time as this CR

Mirror CR: applies only when the text, including clause numbering are exactly the same.

Companion CR: applies when the change means the same but the baselines differ in some way (e.g. clause number).

Follow the principle of completeness, where all changes related to the issue or problem within a deliverable are simultaneously proposed to be made e.g. a change impacting 5 tables should not only include a proposal to change only 3 tables. Include any changes to references, definitions, and abbreviations in the same deliverable.

Follow the drafting rules.

All pictures must be editable.

Check spelling and grammar.

Use change bars for modifications.

The change should include the current and surrounding clauses to clearly show where a change is located and to provide technical context of the proposed change. Additions of complete clauses need not show surrounding clauses as long as the proposed clause number clearly shows where the proposed new clause is located.

Multiple changes in a single CR shall be clearly separated by horizontal lines with embedded text such as, start of change 1, end of change 1, start of new clause, end of new clause.

When subsequent changes are made to the content of a CR, then the accepted version should not show changes over changes. The accepted version of the CR should only show changes relative to the baseline approved text.

## Introduction

This CR updates Annex A (for TS-0001) and Annex C (for TS-0004) of the TR-0067 with last updates proposal.

### \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Start of change 1 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Annex A : Proposal for update of TS-0001

In this Annex, are presented the proposed changes to the TS-0001 [2] specification for flexContainer introduction for device management operations.

### ----------------------- Start of change 1 -------------------------------------------

#### 9.6.1.1 Resource Type Summary

*TBD: Add <flexContainer> in <node>’s Child Resource Types and <node> in <flexContainer>’s Parent Resource Types.*

### ----------------------- End of change 1 -------------------------------------------

### ----------------------- Start of change 2 -------------------------------------------

### 9.6.18 Resource Type *node*

The *<node>* resource represents specific information that provides properties of an M2M Node that can be utilized by other oneM2M operations. It contains resources that represent the Node's context information (e.g. memory and battery), network topology, device information, device capability etc. These resources can be mapped in two distinct models:

1. Either the *<node>* resource has specialization of the *<mgmtObj>* as its child resources. The specialized *<mgmtObj>* resources are used to perform management of the Node.
2. Or the <*node*> resource has the [*flexNode*] specialization of a <*flexContainer*> as child resource, and this [*flexNode*] has specialization of the *<flexContainer>* as its child resources. These DM *<flexContainer>* resources are used to perform management of the Node.

This node specific information stored in these resources, <*mgmtObj*> specializations such as *[deviceInfo]* and *[firmware] or <flexContainer>* specializations such as [*dmDeviceInfo*] or [*dmFirmware*], can be obtained either by the existing device management technologies (OMA DM [i.3], BBF TR-069 [i.2]) or any other way (e.g. JNI [i.18]).

For the case when the *<node>* resource belongs to an ADN, please see figure 9.6.18-1 in conjunction with the description of *nodeLink* attribute in the *<AE>* resource (clause 9.6.5).

For the case when the *<node>* resource belongs to an NoDN and the applications that correspond to interworked devices are represented by <*flexContainer>s* please see figure 9.6.18-2.



Figure 9.6.18-1: Relationship between IN/MN and ADN

Figure 9.6.18-2: Relationship between IPE, interworked Services and NoDN

The *<node>* resource shall contain the child resources specified in table 9.6.18-1.

Table 9.6.18-1: Child resources of *<node>* resource

| Child Resources of *<node>* | Child Resource Type | Multiplicity | Description | *<nodeAnnc>* Child Resource Type |
| --- | --- | --- | --- | --- |
| *[variable]* | *<semanticDescriptor>* | 0..n | See clause 9.6.30 | *<semanticDescriptor>, <semanticDescriptorAnnc>* |
| *[variable]* | *<flexContainer> as defined in the specialization [flexNode]* | 0..1 | This resource provides the root for SDT-based <flexContainers> that correspond to Device Management related ModuleClasses (see clause 5.8 in TS-0023).See Note. | *<flexContainerAnnc>* |
| *[variable]* | *<mgmtObj>* as defined in the specialization [*memory]* | 0..1 | This resource provides the memory (typically RAM) information of the node. (E.g. the amount of total volatile memory), See clause D.4. | *<mgmtObjAnnc>* |
| *[variable]* | *<mgmtObj>* as defined in the specialization [*battery]* | 0..n | The resource provides the power information of the node. (E.g. remaining battery charge). See clause D.7. | *<mgmtObjAnnc>* |
| *[variable]* | *<mgmtObj>* as defined in the specialization [*areaNwkInfo]* | 0..n | This resource describes the list of Nodes attached behind the MN/ASN node and its physical or underlying relation among the nodes in the M2M Area Network. This attribute is defined in case the Node is MN/ASN. See clause D.5. | *<mgmtObjAnnc>* |
| *[variable]* | *<mgmtObj>* as defined in the specialization [*areaNwkDeviceInfo]* | 0..n | This resource describes the information about the Node in the M2M Area Network. See clause D.6. | *<mgmtObjAnnc>* |
| *[variable]* | *<mgmtObj>* as defined in the specialization [*firmware]* | *0..n* | This resource describes the information about the firmware of the Node include name, version etc. See clause D.2. | *<mgmtObjAnnc>* |
| *[variable]* | *<mgmtObj>* as defined in the specialization [*software]* | 0..n | This resource describes the information about the software of the Node. See clause D.3. | *<mgmtObjAnnc>* |
| *[variable]* | *<mgmtObj>* as defined in the specialization [*deviceInfo]* | 0..n | The resource contains information about the identity, manufacturer and model number of the device. See clause D.8. | *<mgmtObjAnnc>* |
| *[variable]* | *<mgmtObj>* as defined in the specialization [*deviceCapability]* | 0..n | The resource contains information about the capability supported by the Node. See clause D.9. | *<mgmtObjAnnc>* |
| *[variable]* | *<mgmtObj>* as defined in the specialization [*reboot]* | 0..1 | The resource is the place to reboot or reset the Node. See clause D.10. | *<mgmtObjAnnc>* |
| *[variable]* | *<mgmtObj>* as defined in the specialization [*eventLog]* | 0..1 | The resource contains the information about the log of events of the Node. See clause D.11. | *<mgmtObjAnnc>* |
| *[variable]* | *<mgmtObj>* as defined in the specialization *[cmdhPolicy]* | 0..n | The resource(s) contain(s) information about CMDH policies that are applicable to the CMDH processing on the CSE hosted on the node represented by this *<node>* resource and identified by the *hostedCSELink* attribute of this *<node>* resource. See clause D.12. | NA |
| *[variable]* | *<mgmtObj>* as defined in the specialization *[activeCmdhPolicy]* | 0..1 | This resource defines which of the present *[cmdhPolicy]* resource(s) shall be active for the CMDH processing on the CSE hosted on the node represented by this *<node>* resource and identified by the *hostedCSELink* attribute of this *<node>* resource. See clause D.12. | NA |
| *[variable]* | *<subscription>* | 0..n | See clause 9.6.8. |  *<subscription>* |
| *[variable]* | *<schedule>* | 0..n | See clause 9.6.9. |  *<scheduleAnnc>* |
| *[variable]* | *<transaction>* | 0..n | See clause 9.6.48 | *<transaction>* |
| *[variable]* | *<action>* | 0..n | See clause 9.6.61 | *None* |
| *NOTE: if the [flexNode] child is present, then all <mgmtObj> children from [memory] to [eventLog] cannot be present.* |

The *<node>* resource shall contain the attributes specified in table 9.6.18-2.

Table 9.6.18-2: Attributes of *<node>* resource

| Attributes of *<node>* | Multiplicity | RW/RO/WO | Description | *<nodeAnnc>* attributes |
| --- | --- | --- | --- | --- |
| *resourceType* | 1 | RO | See clause 9.6.1.3. | NA |
| *resourceID* | 1 | RO | See clause 9.6.1.3. | NA |
| *resourceName* | 1 | WO | See clause 9.6.1.3. | NA |
| *parentID* | 1 | RO | See clause 9.6.1.3. | NA |
| *expirationTime* | 1 | RW | See clause 9.6.1.3. | MA |
| *accessControlPolicyIDs* | 0..1 (L) | RW | See clause 9.6.1.3. | MA |
| *creationTime* | 1 | RO | See clause 9.6.1.3. | NA |
| *lastModifiedTime* | 1 | RO | See clause 9.6.1.3. | NA |
| *labels* | 0..1 (L) | RW | See clause 9.6.1.3. | MA |
| *announceTo* | 0..1 (L) | RW | See clause 9.6.1.3. | NA |
| *announcedAttribute* | 0..1 (L) | RW | See clause 9.6.1.3. | NA |
| *announceSyncType* | 0..1 | RW | See clause 9.6.1.3. | MA |
| *dynamicAuthorizationConsultationIDs* | 0..1 (L) | RW | See clause 9.6.1.3. | OA |
| *owner* | 0..1 | RW | See clause 9.6.1.3 | NA |
| *nodeID* | 1 | RW | The M2M-Node-ID of the node which is represented by this *<node>* resource. | MA |
| *nodeType* | 0..1 | RW | Indicates the type of node.It shall have one of the following values: * IN
* MN
* ASN
* ADN
* NoDN­­
 | OA |
| *hostedCSELink* | 0..1 | RW | This attribute allows to find the <CSEBase> or <remoteCSE> resource representing the CSE that is residing on the node that is represented by this <*node*> resource. The attribute contains the resource ID of a resource where all of the following applies:* The resource is a *<CSEBase>* resource or a *<remoteCSE>* resource.
* The resource represents the CSE which resides on the specific node that is represented by the current *<node>* resource.

In case the node that is represented by this <node> resource does not contain a CSE, this attribute shall not be present. | OA |
| *hostedAELinks* | 0..1(L) | RW | This attribute allows to find the AEs hosted by the node that is represented by this <*node*> resource. The attribute shall contain a list of resource identifiers of *<AE>* resources representing the ADN-AEs residing on the node that is represented by the current *<node>* resource.In case the node that is represented by this <node> resource does not contain an AE, this attribute shall not be present. | OA |
| *hostedServiceLinks* | 0..1(L) | RW | This attribute allows to find <*flexContainer> resources that have* been created by an IPE to represent services hosted on a NoDN, the NoDN being represented by this <*node*> resource. If the NoDN hosts a set of services represented by <*flexContainer>s,* then the attribute shall contain the list of resource identifiers of these <*flexContainer>* resources.In case the node that is represented by this <*node*> resource does not contain anservice that is represented by a <*flexContainer>,* this attribute shall not be present. | OA |
| *mgmtClientAddress* | 0..1 | RW | Represents the physical address of management client of the node which is represented by this <node> resource.This attribute is absent if management server is able to acquire the physical address of the management client. | OA |
| *roamingStatus* | 0..1 | RO | Indicates if the M2M Node is currently roaming from the perspective of the underlying network. The allowed values are “Yes” or “No”.  | OA |
| *networkID* | 0..1 | RO | Configured with the identity of the underlying network which the M2M Node is currently attached to.  | OA |

### ----------------------- End of change 2 -------------------------------------------

### ----------------------- Start of change 3 -------------------------------------------

### 10.2.8 Device management

#### 10.2.8.1 Introduction

This clause describes the procedures for managing device capabilities on MNs (e.g. M2M Gateways), ASNs and ADNs (e.g. M2M Devices), as well as devices that reside within an M2M Area Network.

Resources maintaining information and relationships that are specific to Device Management are termed Device Management Resources. This clause details the creation, retrieval, update and deletion of the information associated with the following Device Management Resources: <node> (clauses 10.2.8.2 to 10.2.8.6), <mgmtObj> (clauses 10.2.8.7 to 10.2.8.12), <mgmtCmd> and its child resource <execInstance> (clauses 10.2.8.13 to 10.2.8.21), [flexNode] and its child DM <flexContainer> resources (clauses 10.2.8.22 to 10.8.24).

These operations are used in both Device Management options available in oneM2M: one utilizing existing technology protocols (e.g. BBF TR‑069 [i.2], OMA-DM [i.3], and LWM2M [i.4]) and another utilizing the native oneM2M protocols. Clause 6.2.4 details the Device Management (DMG) CSF supporting this functionality.

#### 10.2.8.2 Node management

This clause describes node management procedures over Mca and Mcc reference points, using the *<node>* resource which represents information about M2M Nodes that can be utilized in Device Management and other operations.

M2M Nodes represented by the <node> resource are: MN-CSE, ASN-CSE, ADN and NoDN. Zero, one or more <*node*> resources may be used to represent each M2M Node, as follows.

* A <*node*> resource representing a MN-CSE or a ASN-CSE is hosted by the represented CSE or the registrar CSE. The *hostedCSELink* attribute of the resource allows to find the <CSEBase> or <remoteCSE> resource representing the MN-CSE or ASN-CSE represented by the <node> resource. All *<node>* resources hosted on M2M Node's CSE may be announced to associated IN-CSEs.
* A <*node*> resource representing an ADN is hosted by the registrar CSE. The *hostedAELink* attribute of the resource allows to find the <AE> resources representing the AEs residing on the node ADN.
* A <*node*> resource representing a NoDN is hosted by a CSE with DMG capabilities used to perform Device Management operations on the NoDN. If the NoDN is an interworked device, the *hostedServiceLink* attribute of the resource allows to find the <*flexContainer*> resources representing the services hosted on the NoDN.

An entity co-located with a CSE on an ASN or MN which is managed using oneM2M Device Management shall be represented by the same <*node*> resource

Device Management resources associated with a M2M Node that is represented by a <node> resource shall be created

* either as <flexContainer> specializations children of a [flexNode] child of the <node>,
* or as <mgmtObj> direct children of the <node>,
* or as <mgmtCmd> and <execInstance> resources: the <execInstance> are created as children of the <node> resource(s) referenced in the <mgmtCmd>’s *execTarget* attribute.

### ----------------------- End of change 3 -------------------------------------------

### ----------------------- Start of change 4 -------------------------------------------

#### 10.2.8.22 Device management using [flexNode] and DM <flexContainer> resources

This clause describes procedures for managing device capabilities, using DM <flexContainer> specializations. This is an alternative to the approach based on <mgmtObj> or <mgmtCmd> resources specified in clause 10.2.8.7 to 10.2.8.21.

This clause describes the management procedures over Mca and Mcc reference points. If technology specific protocols are used for management, different operations addressing a DM *<flexContainer>* resource (or its attributes or child resources) shall be translated by IN-CSE or MN-CSE into technology specific requests performed on the mapped technology specific data model object on the managed entity. In this case, the DM *<flexContainer*> resources are hosted on the IN-CSE or MN-CSE. Although management requests by the AE are agnostic to the technology specific protocol, the DM *<flexContainer>* resource exposes information about the technology specific protocol. AEs have the capability to retrieve this information within the *objectIDs* attribute of the DM *<flexContainer>* resource.

In the scenario where the DM *<flexContainer>* resource does not utilize an external management technology but instead uses the M2M Service Layer to perform the management request, the DM *<flexContainer>* resource is hosted on the CSE of the managed entity when the managed entity is an ASN, MN or IN. If the managed entity is an ADN node or the managed entity is co-located on an ASN, MN or IN, the DM *<flexContainer*> resource is hosted on the registrar CSE of the managed entity. The DM *<flexContainer>*, its parent [flexNode] and its grand-parent *<node>* resources hosted on node's CSE may be announced to associated IN-CSEs.

In the scenario where the managed entity is a NoDN, the managed entities' DM *<flexContainer>* resources are hosted by a CSE with DMG capabilities used to perform Device Management operations on the NoDN.

The Node management, as described in clauses 10.2.8.2 to 10.2.8.6, is unchanged, but in this case the only child of the <node> resource will be a [flexNode] specialization.

#### 10.2.8.23 flexNode management

##### 10.2.8.23.1 Create [flexNode]

This procedure shall be used for creating a *[flexNode]* resource.

NOTE: The creation of the *[flexNode]* resource is on discretion of the Originator.

Table 10.2.8.23.111-1: *[flexNode]* CREATE

| ***[flexNode]* CREATE** |
| --- |
| Information in Request message | All parameters defined in table 8.1.2-3 apply with the specific details for:***Content:*** The representation of the [flexNode] resource described in clause 5.8.2 in TS-0023. |
| Processing at Originator before sending Request | According to clause 10.1.2 |
| Processing at Receiver | According to clause 10.1.2 |
| Information in Response message | All parameters defined in table 8.1.3-1 apply with the specific details for:* ***Content*:** Address of the created *[flexNode]* resource, according to clause 10.1.2
 |
| Processing at Originator after receiving Response | According to clause 10.1.2 |
| Exceptions | According to clause 10.1.2 |

##### 10.2.8.23.2 Retrieve [flexNode]

This procedure shall be used for retrieving the attributes of a *[flexNode]* resource.

Table 10.2.8.23.2-1: *[flexNode]* RETRIEVE

|  |
| --- |
| *[flexNode]* RETRIEVE |
| Information in Request message | All parameters defined in table 8.1.2-3 apply with the specific details for:***Content*:** Void |
| Processing at Originator before sending Request | According to clause 10.1.3 |
| Processing at Receiver | According to clause 10.1.3 |
| Information in Response message | All parameters defined in table 8.1.3-1 apply with the specific details for:***Content*:** Attributes of the *[flexNode]* resource as defined in clause 5.8.2 in TS-0023 |
| Processing at Originator after receiving Response | According to clause 10.1.3 |
| Exceptions | According to clause 10.1.3 |

##### 10.2.8.23.3 Update [flexNode]

This procedure shall be used for updating the attributes and the actual data of a *[flexNode]* resource and its child resources.

Table 10.2.8.23.3-1: *[flexNode]* UPDATE

| *[flexNode]* UPDATE |
| --- |
| Information in Request message | All parameters defined in table 8.1.2-3 apply with the specific details for:***Content***: attributes of the *[flexNode]* resource as defined in clause 5.8.2 in TS-0023 which need be updated, with the exception of the Read Only (RO) attributes cannot be modified |
| Processing at Originator before sending Request | According to clause 10.1.4 |
| Processing at Receiver | According to clause 10.1.4 with the following:* The Receiver shall check whether the provided attributes of the [flexNode] resource represent a valid request for updating *[flexNode]* resource
 |
| Information in Response message | According to clause 10.1.4 |
| Processing at Originator after receiving Response | According to clause 10.1.4 |
| Exceptions | According to clause 10.1.4 |

##### 10.2.8.23.4 Delete [flexNode]

This procedure shall be used for deleting an existing *[flexNode]* resource.

Table 10.2.8.23.4-1: *[flexNode]* DELETE

| *[flexNode]* DELETE |
| --- |
| Information in Request message | All parameters defined in table 8.1.2-3 apply |
| Processing at Originator before sending Request | According to clause 10.1.5 |
| Processing at Receiver | According to clause 10.1.5 |
| Information in Response message | According to clause 10.1.5 |
| Processing at Originator after receiving Response | According to clause 10.1.5 |
| Exceptions | According to clause 10.1.5 |

#### 10.2.8.24 DM <flexContainer> management

DM <flexContainer> resources are <flexContainer> specializations that correspond to Smart Device Templates ModuleClasses specified in TS-0023 clause 5.8.x, that extend the abstract ModuleClass [dmBaseModule].

They are defined with two optional *custom attributes*, objectIDs and objectPaths, that have the same role as the corresponding *object attributes* in <mgmtObj> resources.

##### 10.2.8.24.1 Create DM *<flexContainer>*

This procedure shall be used to create a specific *DM <flexContainer>* resource in the Hosting CSE to expose the corresponding management function of a managed entity (i.e. M2M Device/Gateway) over the Mca reference point. Depending on the data model being used, the created *DM <flexContainer>* resource may be a partial or complete mapping from the technology specific data model object on the managed entity. If such a technology specific data model object is missing from the managed entity, it shall be added to the managed entity. Further operations performed on the created *DM <flexContainer>* resource shall be converted by the Hosting CSE into a corresponding technology specific request performed on the mapped technology specific data model object on the managed entity using technology specific protocol (e.g. OMA‑DM [i.3] or BBF TR-069 [i.2]).

Besides the generic create procedure defined in clause 10.1.2, the procedure in the following table shall be used when management is performed using technology specific protocols.

If the management is performed by service layer entities, the procedure is the same as generic create procedure defined in clause 10.1.2. In this case, local APIs (drivers) on the managed entity is required to monitor the change of the DM <flexContainer> resource and reflect the change to the managed entity.

Table 10.2.8.24.1-1: *DM <flexContainer>* CREATE

|  |
| --- |
| *DM <flexContainer>* CREATE  |
| Information in Request message | ***From:*** Identifier of the AE or the CSE that initiates the Request***To:*** The address of the *[flexNode]* where the *DM <flexContainer>* resource is intended to be Created***Content:*** The representation of the *DM <flexContainer>* resource. |
| Processing at Originator before sending Request | The Originator shall be an AE, or a CSE:* The Originator is a CSE: In this case, the CSE first collects the original technology specific data model object (the management tree structure or also the value of the tree nodes if needed) of the local device and transforms the object into the *DM <flexContainer>* resource representation, then requests the Hosting CSE to create the corresponding *DM <flexContainer>* resource.
* The Originator is an AE: In this case, the AE requests the Hosting CSE to add the corresponding technology specific data model object to the managed entity by creating an DM <flexContainer> resource in the Hosting CSE

(See notes 1 and 2) |
| Processing at Receiver | For the CREATE operation, besides the common create operation defined in clause 10.1.2, the Receiver shall:* If the Originator is an AE: Check if there is existing management session between the management server and the managed entity. If not, request the management server to establish a management session towards the managed entity. Send the technology specific request to the managed entity or to the management server to add the corresponding technology specific data model object to the managed entity based on technology specific protocol
* Maintain the mapping relationship between the created *DM <flexContainer>* resource and the technology specific data model object on the managed entity
* Respond to the Originator with the appropriate responses based on the technology specific response. It shall also provide in the response the address of the created new resource
 |
| Information in Response message | Error code if the new technology specific data model object is not created |
| Processing at Originator after receiving Response | None |
| Exceptions | * The creation of the technology specific data model object is not allowed
* The created technology specific data model object already exists
* Corresponding technology specific data model object cannot be added to the managed entity for some reason (e.g. not reachable, memory shortage)
 |
| NOTE 1: The CSE can create the *DM <flexContainer>* resource locally by itself. The details are out of scope. In this case, the Hosting CSE first collects the original technology specific data model object on the managed entity via technology specific protocol (e.g. OMA DM [i.3], BBF TR-069 [i.2] or LWM2M [i.4]), then transforms the object into the *DM <flexContainer>* resource representation and create the *DM <flexContainer>* resource locally in the CSE.NOTE 2: The *DM <flexContainer>* resource can be created in the Hosting CSE by other offline provisioning means which are out of scope. |

##### 10.2.8.24.2 Retrieve *DM <flexContainer>*

This procedure shall be used to retrieve information from an existing *DM <flexContainer>* resource. Besides the generic retrieve procedure defined in clause 10.1.3, the procedure in the following table shall be used when management is performed using technology specific protocols. If the management is performed by service layer entities, the procedure is the same as generic retrieve procedure defined in 10.1.3.

Table 10.2.8.24.2-1: *DM <flexContainer>* RETRIEVE

|  |
| --- |
| *DM <flexContainer>* RETRIEVE  |
| Information in Request message | ***From:*** Identifier of the AE or the CSE that initiates the Request***To:*** The address of the *DM <flexContainer>* resource |
| Processing at Originator before sending Request | None |
| Processing at Receiver | For the RETRIEVE operation, besides the common retrieve operation defined in clause 10.1.3, the Receiver shall:* If the Originator is an AE and if the requested information of the *DM <flexContainer>* resource is not available, identify the corresponding technology specific data object on the managed entity according to the mapping relationship that the CSE maintains. Check if there is an existing management session between the management server and the managed entity. If not, request the management server to establish a management session towards the managed entity. Send the technology specific request to get the corresponding technology specific data model object from the managed entity based on the external management technology, then return the result to the Originator based on the technology specific response
 |
| Information in Response message | Error code if the new technology specific data model object cannot be retrieved |
| Processing at Originator after receiving Response | None |
| Exceptions | * Corresponding technology specific data model object data cannot be retrieved from the managed entity (e.g. technology specific data model object not found)
 |

##### 10.2.8.24.3 Update DM <flexContainer>

This procedure shall be used to update information of an existing *DM <flexContainer>* resource. Besides the generic update procedure defined in clause 10.1.4, the procedure in the following table shall be used when management is performed using technology specific protocol. If the management is performed by service layer entities, the procedure is the same as generic update procedure defined in clause 10.1.4. In this case, local APIs (drivers) on the managed entity is required to monitor the change of the DM <flexContainer> resource and reflect the change to the managed entity.

Table 10.2.8.24.3-1: *DM <flexContainer>* UPDATE

|  |
| --- |
| *DM <flexContainer>* UPDATE |
| Information in Request message | ***From:*** Identifier of the AE or the CSE that initiates the Request***To:*** The address of the *DM <flexContainer>* resource***Content:*** The representation of the *DM <flexContainer>* resource. |
| Processing at Originator before sending Request | None |
| Processing at Receiver | For the UPDATE operation, besides the common update operation defined in clause 10.1.4, the Receiver shall:* If the Originator is an AE, identify the corresponding technology specific data model object on the managed entity according to the mapping relationship it maintains. Check if there is an existing management session between the management server and the managed entity. If not, request the management server to establish a management session towards the managed entity. Send the technology specific request to update the corresponding technology specific data model object in the managed entity accordingly based on technology specific protocol
* Respond to the Originator with the appropriate response based on the technology specific response from the external management technology
 |
| Information in Response message | Error code if the technology specific data model object cannot be updated |
| Processing at Originator after receiving Response | None |
| Exceptions | * Corresponding technology specific data model object cannot be updated to managed entity (e.g. not reachable, technology specific data model object not found)
 |

##### 10.2.8.24.4 Delete DM <flexContainer>

This procedure shall be used to delete an existing *DM <flexContainer>* resource. An Originator uses this procedure to remove the corresponding technology specific data model object (e.g. an obsolete software package) from the managed entity. Besides the generic delete procedure defined in clause 10.1.5, the procedure in the following table shall be used when management is performed using external management technologies. If the management is performed by service layer entities, the procedure is the same as generic delete procedure defined in clause 10.1.5. In this case, local APIs (drivers) on the managed entity is required to monitor the change of the DM <flexContainer> resource and reflect the change to the managed entity.

Table 10.2.8.24.4-1: *DM <flexContainer>* DELETE

|  |
| --- |
| *DM <flexContainer>* DELETE |
| Information in Request message | ***From:*** Identifier of the AE, or the CSE that initiates the Request***To:*** The address of the *DM <flexContainer>* resource |
| Processing at Originator before sending Request | The Originator shall be an AE or CSE:* The Originator is a CSE: In this case, the CSE issues the request to the Hosting CSE to hide the corresponding management function from being exposed by the *DM <flexContainer>* resource
* The Originator is an AE: In this case, the AE requests the Hosting CSE to delete the *DM <flexContainer>* resource from the Hosting CSE and to remove the corresponding technology specific data model object from the managed entity

(See notes 1 and 2) |
| Processing at Receiver | For the DELETE operation, besides the common create operation defined in clause 10.1.5, the Receiver shall:* If the Originator is an AE, identify the corresponding technology specific data model object on the managed entity according to the mapping relationship the CSE maintains. Check if there is an existing management session between the management server and the managed entity. If not, request the management server to establish a management session towards the managed entity. The CSE sends technology specific request to remove the corresponding technology specific data model object from the managed entity based on technology specific protocol
* Respond to the Originator with the appropriate generic responses based on the technology specific response
 |
| Information in Response message | Error code if the technology specific data model object cannot be deleted |
| Processing at Originator after receiving Response | None |
| Exceptions | * Corresponding technology specific data model object cannot be deleted from managed entity (e.g. not reachable, technology specific data model object not found)
 |
| NOTE 1: The Hosting IN-CSE can delete the *DM <flexContainer>* resource locally by itself. This internal procedure is out of scope.NOTE 2: The *DM <flexContainer>* resource can be deleted in the Hosting CSE by offline provisioning means which are out of scope. |

### ----------------------- End of change 4 -------------------------------------------

### ----------------------- Start of change 5 -------------------------------------------

## D.12 Resource *cmdhPolicy*

*TBD: Write DM <flexContainers> that correspond to the 8 [\*cmdh\*] <mgmtObj>.*

*Waiting for new version of CMDH processing through flexContainers (WI-0096).*

### ----------------------- End of change 5 -------------------------------------------

### \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* End of change 1 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

### \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Start of change 2 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Annex C : Proposal for update of TS-0004

In this Annex, are presented the proposed changes to the TS-0004 [3] specification for flexContainer introduction for device management operations.

### ----------------------- Start of change 1 -------------------------------------------

### 7.3.4 Management common operations

This clause describes common operations on Device Management resources that are associated with a M2M Node that is represented by a <node> resource. These resources are:

* either DM <flexContainer> specializations children of a [flexNode] child of the <node>,
* or <mgmtObj> direct children of the <node>,
* or <mgmtCmd> and <execInstance> resources: the <execInstance> are created as children of the <node> resource(s) referenced in the <mgmtCmd>’s *execTarget* attribute.

#### 7.3.4.1 Identify the managed entity and the technology specific protocol

Where a managed entity is being addressed via a <mgmtObj> or DM <flexContainer> resource, the Hosting CSE shall identify the managed entity via the <node> resource that is the parent resource of the <mgmtObj>, or the parent of the [flexNode] parent of the DM <flexContainer> resource. In case of a <mgmtCmd> resource the entity to be managed is indicated by its *execTarget* attribute. This addresses either a <node> resource or a group of resources of type <node>. Hence, in all cases the managed entity is ultimately identified through a <node> resource, from which the identifier of the device can be retrieved.

The Hosting CSE shall determine the technology specific protocol to be used for communicating with the managed entity based on the *objectIDs* attribute of the addressed <mgmtObj> or DM <flexContainer> resource.

If the managed entity cannot be identified, the Hosting CSE shall reject the request with the ***Response Status Code*** indicating "EXTERNAL\_OBJECT\_NOT\_REACHABLE" in the Response primitive.

#### 7.3.4.2 Locate the technology specific data model objects to be managed on the managed entity

The Hosting CSE shall locate the technology specific data model object to be managed on the managed entity by the *objectPaths* attribute of the <mgmtObj> or DM <flexContainer> resource addressed by the URI provided in the ***To*** primitive parameter. In the case that the ***To*** addresses an [objectAttribute] (resp. [customAttribute]), the Hosting CSE shall locate the technology specific data model object on the managed entity through the *objectPaths* attribute of the <mgmtObj> (resp. DM <flexContainer>) resource of the addressed [objectAttribute] / [customAttribute], combined with their relative position in the technology specific data model object tree. If the technology specific data model object cannot be located, the Hosting CSE shall reject the request with the ***Response Status Code*** indicating "EXTERNAL\_OBJECT\_NOT\_FOUND" in the Response primitive.

In the case that the management server is external to the Hosting CSE, the Hosting CSE shall identify the management server that is capable of performing the operation on the technology specific data model object. If the management server cannot be identified, the Hosting CSE shall reject the request with the ***Response Status Code*** indicating "EXTERNAL\_OBJECT\_NOT\_REACHABLE" in the Response primitive.

### ----------------------- End of change 1 -------------------------------------------

### ----------------------- Start of change 2 -------------------------------------------

#### 7.3.4.4 Send the management request(s) to the managed entity corresponding to the received Request primitive

The Hosting CSE shall send the management request(s) to the managed entity or management server in the established management session in order to perform the management operation as requested by the received Request primitive. The management request shall address the technology-specific data model object on the managed entity as determined in clause 0 or in the primitive-specific clauses. The management request being used is specific to the technology specific protocol according to a pre-defined mapping relationship with the Request primitive. The internal data structure of the technology specific data model object addressed by the technology specific request shall be determined based on the mapping relationship of the <mgmtObj>, DM <flexContainer> or <mgmtCmd> resources and the technology specific data model objects or based on the generic mapping rule as specified in oneM2M TS-0001 [**Erreur ! Source du renvoi introuvable.**], clauses 9.6.15, 9.6.16, 9.6.17 and 9.6.35. The Hosting CSE shall extract the management results received from the managed entity or management server in order to prepare a Response primitive to be sent to the originator later. Unless explicitly stated, if the management request cannot be performed successfully, the Hosting CSE shall reject the Request primitive with the management server in the Response primitive according to the mapping relationship with the technology specific protocol.

### ----------------------- End of change 2 -------------------------------------------

### ----------------------- Start of change 3 -------------------------------------------

### 7.4.18 Resource Type <node>

#### 7.4.18.1 Introduction

The <node> resource represents specific information that provides properties of an oneM2M Node that can be utilized by other oneM2M operations. The <node> resource has either a [flexNode] <flexContainer> specialization or <mgmtObj> as its child resources.

Table 7.4.18.1‑1: Data type definition of <node> resource

|  |  |  |
| --- | --- | --- |
| **Data Type ID** | **File Name** | **Note** |
| node | CDT-node-v4\_2\_0.xsd |  |

Table 7.4.18.1‑2: Universal/Common Attributes of <node> resource

|  |  |
| --- | --- |
| Attribute Name | Request Optionality  |
| Create | Update |
| *@resourceName* | O | NP |
| *resourceType*  | NP | NP |
| *resourceID* | NP | NP |
| *parentID* | NP | NP |
| *accessControlPolicyIDs* | O | O |
| *creationTime* | NP | NP |
| *expirationTime* | O | O |
| *lastModifiedTime* | NP | NP |
| *labels* | O | O |
| *announceTo* | O | O |
| *announcedAttribute* | O | O |
| *dynamicAuthorizationConsultationIDs* | O | O |

Table 7.4.18.1‑3: Resource Specific Attributes of <node> resource

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute Name | Request Optionality  | Data Type | Default Value and Constraints |
| Create | Update |
| *nodeID* | M | O | m2m:nodeID |  |
| *hostedCSELink* | O | O | m2m:ID |  |
| *hostedAELinks* | O | O | m2m:listOfM2MID |  |
| *hostedServiceLinks* | O | O | m2m:listOfM2MID |  |
| *mgmtClientAddress* | O | O | xs:string |  |
| *roamingStatus* | NP | NP | xs:boolean | No default. True means that the Node is currently roaming.When this attribute is not present, it indicates that no information is available. |
| *networkID* | NP | NP | xs:string | No default. When this attribute is not present, it indicates that no information is available. |
| *nodeType* | O | O | m2m:nodeType | Default is UNSPECIFIED. |

Table 7.4.18.1‑4: Child resources of <node> resource

|  |  |  |  |
| --- | --- | --- | --- |
| Child Resource Type  | Child Resource Name | Multiplicity | Ref. to Resource Type Definition |
| <mgmtObj> | [variable] | 0..n | Clause **Erreur ! Source du renvoi introuvable.**,and **Erreur ! Source du renvoi introuvable.** |
| <flexContainer> | [flexNode] | 0..1 | Clause 5.8.2 in TS-0023. |
| <subscription> | [variable] | 0..n | Clause **Erreur ! Source du renvoi introuvable.** |
| <semanticDescriptor> | [variable] | 0..n | Clause **Erreur ! Source du renvoi introuvable.** |
| <transaction> | [variable] | 0..n | Clause 7.4.61 |
| <schedule> | [variable] | 0..n | Clause 7.4.9 |

### ----------------------- End of change 3 -------------------------------------------

### \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* End of change 2 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*