|  |
| --- |
|  |

|  |  |
| --- | --- |
| CHANGE REQUEST | |
| Meeting ID:\* | Joint RDM/SDS |
| Source:\* | Cyrille Bareau, Orange, [cyrille.bareau@orange.com](mailto:cyrille.bareau@orange.com)  Bob Flynn, Exacta, [bob.flynn@exactagss.com](mailto:bob.flynn@exactagss.com)  Andreas Kraft, Deutsche Telekom, [a.kraft@telekom.de](mailto:a.kraft@telekom.de)  Marianne Mohali, Orange, [marianne.mohali@orange.com](mailto:marianne.mohali@orange.com) |
| Date:\* | 2022-03-30 |
| Reason for Change/s:\* | See the introduction. |
| CR against: Release\* | Release 4 |
| CR against: WI\* | Active WI-0109  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-0035 v0.2.1 |
| Clauses \* | Modified clauses: 2.2, 3, 5.1, 5.2, 5.3.1, 5.3.2, 5.3.3, 5.4.1, 5.4.2, 5.4.3  New clause 7 |
| Type of change: \* | Editorial change  Bug Fix or Correction  Change to existing feature or functionality  New feature or functionality  Only 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  NO  This CR may break backwards compatibility with the last approved version of the TS? YES  NO |
| 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 draft is part of a series of CRs related to the Work Item WI-0109: IPE-based Device Management with FlexContainers. For a full introduction, see clause 2 “Justification” in WI-0109-IPE-based\_Device\_Management\_with\_FlexContainers-V0\_0\_1.DOCX.

In this specific draft, the proposed changes are as follows:

1. Add TS-0023 and TS-0033 to the references list.
2. Generalize the notion of DM Resource: <mgmtObj> *or* SDT DM <flexContainer>.

3,4,5. Replicate the use cases with SDT approach.

6. Add a new clause for IPE-based DM.

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

## 2.2 Informative references

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

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: "Functional Architecture".

[i.3] BBF TR-069: "Abstract Test Plan".

[i.4] oneM2M TS-0023: “SDT based Information Model and Mapping for Vertical Industries”

[i.5] oneM2M TS-0033: “Interworking Framework”

# 3 Abbreviations

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

ADN Application Dedicated Node

ADN-AE AE which resides in the Application Dedicated Node

AE Application Entity

ASN Application Service Node

BBF Broadband Forum

CoAP Constrained Application Protocol

CSE Common Services Entity

CSE-ID Common Service Entity Identifier

DM Device Management

HTTP HyperText Transfer Protocol

IN Infrastructure Node

IN-CSE CSE which resides in the Infrastructure Node

JSON JavaScript Object Notation

Mca Reference Point for M2M Communication with AE

MN Middle Node

MN-CSE CSE which resides in the Middle Node

NoDN Non-oneM2M Node

OMA Open Mobile Alliance

SDT Smart Device Template

URI Uniform Resource Identifier

URL Uniform Resource Link

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

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

## 5.1 Introduction

*In the rest of this document, “Device Management Resource” is a generic term used to represent either <mgmtObj> or SDT DM <flexContainer> resources. SDT DM <flexContainer> specializations are defined in oneM2M TS-0023 [i.4], based on the Smart Device Template (SDT) data model, a model used to represent IoT devices in various IoT domains. When device management is performed with this type of resource, a [flexNode] <flexContainer> specialization shall be created as child of the <node> resource that represents the managed entity, and the <flexContainer> specializations that represent the SDT modules to perform DM operations are created as children of this [flexNode].*

*In order to distinguish <mgmtObj> and <flexContainer> specializations, both between square brackets, the <flexContainer> specializations will be underlined: e.g. [memory] for a <mgmtObj>, [dmAgent] for a <flexContainer>.*

oneM2M uses Device Management Resources for device management. For device management over the service layer, the Device Management Resource that is used for the management of the managed entity is located at different places following the characteristic of the managed entity. In each case, there exists one driver or client on the managed entity that observes, directly or not, the updates of the Device Management Resource to perform management operations.

For ASN, MN and IN which have a CSE, the Device Management Resource is hosted on the corresponding ASN-CSE, MN-CSE and IN‑CSE as child resource of the <node> resource that represents the ASN, MN and IN, either as *direct* child of the <node> resource in the case of <mgmtObj>, or as child of the [*flexNode*] <flexContainer> child of the <node> for SDT DM <flexContainer>. The managed entity observes the Device Management Resource directly to perform the management operations.

For ADN, the Device Management Resource is hosted on the registrar CSE of the ADN-AE as child resource of the <node> resource that represents the ADN. The ADN-AE in this case should subscribe to the Device Management Resources under the corresponding <node> resource to receive notifications on any update of the Device Management Resource. The ADN which is the managed entity in this case further performs the management operations based on the notifications received.

For NoDN, the Device Management Resource is hosted on the CSE which the NoDN is connected to. As NoDN is not an oneM2M defined entity, the connection between NoDN and the CSE is out of scope of oneM2M and is implementation specific. In this case, the Device Management Resource on the CSE under the corresponding <node> resource should be monitored to determine if any management operation needs to be performed. This part is implementation specific and is out of scope of oneM2M as well.

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

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

## 5.2 Use case

An application wants to retrieve the current available memory of the device. In this case, it is either possible to use the specialization of <mgmtObj> [*memory*] defined in clause D.4 of [i.2], or the [*dmAgent*] <flexContainer> specialization defined in clause 5.8.3 of [i.4]. In the [*memory*] specialization, *memAvailable* is the attribute that contains the required information, or *ramAvailable* and *storageAvailable* attributes in [*dmAgent*]. The application can get to know the available memory by retrieving the resource.

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

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

## 5.3 Architecture

### 5.3.1 Management of ASN, MN and IN



Figure 5.3.1-1: Architecture for management of IN,MN,ASN

In Figure 5.3.1-1 is represented the architecture for management of IN, MN and ASN, using <mgmtObj> resources. As those Managed Entities have their own CSEBase resource, they have the capability to host oneM2M resources. When using device management over the service layer, the *[memory]* (resp. [*dmAgent*])resource is directly hosted under the <node> resource of the CSEBase (resp. under the [*flexNode*] resource child of the <node> resource). In this case, the <node> resource represents the IN, MN, ASN themselves.

The Driver is the software that interacts with the memory of the device that acquires the total memory and available memory from the system.

In this case, the *[memory]* or [*dmAgent*] specialization is modified by the Driver using an internal interface. The modification may be done periodically or following some other policies, which is out of scope of oneM2M standard.

Whenever the AE issues a retrieve request to the *[memory]* or [*dmAgent*] specialization, the value from the resource is returned.

### 5.3.2 Management of ADN



Figure 5.3.2-1: Architecture for management of ADN

In Figure 5.3.2-1 is represented the architecture of management of ADN, using <mgmtObj> resources. The *[memory]* or [*dmAgent*] specialization is hosted on the Registrar CSE of the ADN. According to the supported configuration of oneM2M architecture, the Registrar CSE could be IN-CSE or MN-CSE.

In this case, the <node> resource hosted under the CSEBase of the Registrar CSE represents the ADN which is the managed entity. The driver gets the memory information using internal interfaces and makes the ADN-AE update the memory value to the *[memory]* or [*dmAgent*] specialization over the Mca reference point. The update over Mca is triggered periodically or based on the policy of the ADN which is implementation specific.

### 5.3.3 Management of NoDN



Figure 5.3.3-1: Architecture for management of NoDN

In Figure 5.3.3-1 is represented the architecture of management of NoDN, using <mgmtObj> resources. The *[memory]* or [*dmAgent*] specialization is hosted on the connection point of NoDN. The connection point is the entity that the NoDN is connected to. The entity has the adaptor that shares the same network protocol with NoDN.

In this case, the <node> resource hosted under the CSEBase represents the NoDN. And the connection between Driver and the CSE is out of scope of oneM2M. It may be Bluetooth, ZigBee or even proprietary.

The update of the memory information happens over the interface between driver and CSE which is out of scope of oneM2M.

Two architectures are possible:

* Either the CSE handles the interactions with the NoDNs via a Management Server that uses an external management technology, such as BBF TR-069 and OMA DM or LwM2M. In this case, the mapping between the underlying device management protocol data model and oneM2M <mgmtObj> resources is specified in oneM2M TS-0005 (for OMA protocols) or TS-0006 (for BBF protocols). This architecture is described in clause 6.
* Or the interactions with the NoDNs are handled by an Interworking Proxy application Entity (IPE) that is registered as an <AE> on the CSE. In this case, the IPE is responsible for mapping the technology specific data model to oneM2M SDT DM <flexContainer> resources. This architecture is described in clause 7.

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

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

## 5.4 Procedures

### 5.4.1 Management of ASN, MN and IN



Figure 5.4.1-1: Procedures for management of IN, MN, ASN

001: The AE sends an oneM2M Retrieve primitive to the ResourceID of *[memory]* or [*dmAgent*] resource.

002: The CSE receives the Retrieve primitive, triggers the Driver to acquire the current available memory value, and updates the *[memory]* or [*dmAgent*] resource.

003: The CSE responds to the AE with the representation of the *[memory]* or [*dmAgent*] resource.

NOTE: The AE could also subscribe to the [memory] or [*dmAgent*] resource to get the notification on the update of the [memory] or [*dmAgent*] resource.

The [memory] or [*dmAgent*] resource in this case is locally created before hand by the Hosting CSE using internal interfaces.

### 5.4.2 Management of ADN



Figure 5.4.2-1: Procedures for management of ADN

001: The AE sends Retrieve request to Registrar CSE of ADN-AE, requesting the *[memory]* or [*dmAgent*] resource.

002: The ADN internally acquires the available memory information.

003: The Registrar CSE responds with the representation of the *[memory]* or [*dmAgent*] resource.

004: The ADN-AE updates the *[memory]* or [*dmAgent*] resource using Mca reference point.

005: Update successful

NOTE: 001, 003 and 002, 004 don't have a specific order. The ADN is updating the *[memory]* or [*dmAgent*] resource periodically. In 003, the Registrar CSE always responds with the current resource.

Subscription and notification could also be used by the AE to monitor the current status of the *[memory]* or [*dmAgent*] resource.

The [*memory*] or [*dmAgent*] resource in this case is created by the ADN-AE.

### 5.4.3 Management of NoDN



Figure 5.4.3-1: Procedures for management of NoDN

001: The AE sends Retrieve request to connection point of NoDN, requesting the *[memory]* or [*dmAgent*] resource.

002: The NoDN internally acquires the available memory information.

003: The CSE of the connection point responds with the representation of the *[memory]* or [*dmAgent*] resource.

004: The NoDN updates the *[memory]* resource using protocol that is non-oneM2M (see clause 6.4) or the IPE that manages the NoDN acquires the memory information from the NoDN, using protocol that is non-oneM2M, and updates the *[memory]* or [*dmAgent*] resource using Mca reference point (see clause 7.4).

NOTE: 001, 003 and 002, 004 don't have a specific order. The NoDN, or the IPE that manages it, updates the *[memory]* or [*dmAgent*] resource periodically. In 003, the Registrar CSE always responds with the current resource.

The [memory] or [*dmAgent*] resource in this case is created by NoDN via non oneM2M interface (see clause 6), or by the IPE that manages it via oneM2M interface (see clause 7).

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

### ----------------------- Start of change 6 -------------------------------------------

# 7 Device management using IPE

## 7.1 Introduction

In the case of IPE-based device management, management is done through an IPE that uses a device management protocol to communicate using the underlying M2M Area Network. Therefore, in this case, each <node> resource that represents a managed entity is hosted under the <CSEBase> of the Registrar CSE of the <AE> that represents the IPE. The managed entity may be IN, MN, ASN, ADN or NoDN. Then the SDT DM <flexContainer>s that are used for management are hosted as child resources of the [*flexNode*] child of the <node> resource.

The generic guidelines for managing these resources by an IPE are defined in oneM2M TS-0033 [i.5].

## 7.2 Use case

An application wants to retrieve the current available memory of the device. In this case, the [*dmAgent*] specialization of <flexContainer> is used (clause 5.8.3 of [i.4]). In the [*dmAgent*] specialization, *ramAvailable* and *storageAvailable* are the attributes that contain the required information. The application can get to know the available memory by retrieving the resource.



Figure 7.2-1: Use case for management

## 7.3 Architecture

AE

IPE

CSE

CSEBase

[dmAgent]

[flexNode]

node

Mca

Mca

Managed Entity

NoDN

RAM

Driver

Out of scope

Figure 7.3-1: Architecture for management using IPE

In the case of management using IPE, all management related resources are hosted on a CSE (IN or MN or ASN) that is the Registrar CSE of the IPE. The IPE interacts with the managed entity through interfaces that are out of scope.

## 7.4 Procedures

AE

Managed

Entity

IPE

CSE

000: Create [dmAgent]

001: Get memory

002: Update [dmAgent]

003: Retrieve [dmAgent]

004: Response [dmAgent]

Register ME

Register IPE

Register AE

Figure 7.4-1: Procedures for management of NoDN

NOTE: the registrations phases (AE on CSE, IPE on CSE, Managed Entity on IPE) are independent, and in no specific order.

000: After registration of the managed entities to the IPE, through non-oneM2M interface, the IPE creates Device Management Resources on its Registrar CSE, over Mca interface.

001: Information on memory is transferred from the managed entity to the IPE, through non-oneM2Minterface. This can be done in various ways, depending on the underlying technology: ‘push’ of events from the managed entity to the IPE, regular ‘pull’ from the IPE, or synchronous reading from the IPE to the managed entity when the 003 request is received.

002: The IPE updates the [*dmAgent*] resource, over Mca interface, when new values of memory information are detected.

003: AE retrieves the [*dmAgent*] resource from CSE over the Mca interface.

004: The CSE returns the [*dmAgent*] resource over the Mca interface.

## 7.5 Implementation

The [*dmAgent*] resource name is "dmAgent002", its resource ID is "fcnt-117074326", and the parent [*flexNode*] resource ID is "fcnt-160864454". The IPE is registered as an <AE> resource with ID “CAE801969269”.

Table 7.5-1

|  |  |
| --- | --- |
| Step | Message Example |
| 000 | HTTP Request    POST /~/CSE0034234/server/flexNode002 HTTP/1.1  Host : in.management.server.com  X-M2M-RI : 1357  X-M2M-Origin: CAE801969269  Content-Type: application/json;ty=28  {  "mad:dmAgt": {  "rn": "dmAgent002",  "cnd": "org.onem2m.management.moduleclass.dmAgent",  "powSe": "NORMAL",  "ramAe": "222",  "ramTl": "300",  "stoAe": "4321",  "stoTl": "5000",  "state": "ready"  }  } |
| 001 | Out of scope |
| 002 | HTTP Request    PUT /~/CSE0034234/server/dmAgent002 HTTP/1.1  Host : in.management.server.com  X-M2M-RI : 1399  X-M2M-Origin: C324352841DAS  Content-Type: application/json  {  "mad:dmAgt": {  "ramAe": "217"  }  } |
| 004 | HTTP Request    GET /~/CSE0034234/server/dmAgent002 HTTP/1.1  Host : in.management.server.com  X-M2M-RI : 2345  X-M2M-Origin: C324352841DAS  Accept : application/json |
| 004 | HTTP Response  X-M2M-RI : 2345  X-M2M-RSC : 2000  {  "mad:dmAgt": {  "rn": "dmAgent002",  "ty": 28,  "ri": "fcnt-117074326",  "pi": "fcnt-160864454",  "ct": "20220218T103523",  "lt": "20220218T103523",  "et": "20230218T103523",  "cnd": "org.onem2m.management.moduleclass.dmAgent",  "powSe": "NORMAL",  "ramAe": "217",  "ramTl": "300",  "stoAe": "4321",  "stoTl": "5000",  "state": "ready"  }  } |

### ----------------------- End of change 6 -------------------------------------------