

|  |  |
| --- | --- |
| **oneM2M**  **Technical Specification** | |
| Document Number | TS-0024-V3.0.0 |
| Document Name: | OCF Interworking |
| Date: | 2017-December-21 |
| Abstract: | This document specifies details on interworking between oneM2M-specified entitiesand OCF-specified clients and/or servers. |
| Template Version: 08 September 2015 (Dot not modify) | |

This Specification is provided for future development work within oneM2M only. The Partners accept no liability for any use of this Specification.

The present document has not been subject to any approval process by the oneM2M Partners Type 1. Published oneM2M specifications and reports for implementation should be obtained via the oneM2M Partners' Publications Offices.

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

© 2017, oneM2M Partners Type 1 (ARIB, ATIS, CCSA, ETSI, TIA, TSTDI, TTA, TTC).

All rights reserved.

The copyright extends to reproduction in all media.

Notice of Disclaimer & Limitation of Liability

The information provided in this document is directed solely to professionals who have the appropriate degree of experience to understand and interpret its contents in accordance with generally accepted engineering or other professional standards and applicable regulations. No recommendation as to products or vendors is made or should be implied.

NO REPRESENTATION OR WARRANTY IS MADE THAT THE INFORMATION IS TECHNICALLY ACCURATE OR SUFFICIENT OR CONFORMS TO ANY STATUTE, GOVERNMENTAL RULE OR REGULATION, AND FURTHER, NO REPRESENTATION OR WARRANTY IS MADE OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR AGAINST INFRINGEMENT OF INTELLECTUAL PROPERTY RIGHTS. NO oneM2M PARTNER TYPE 1 SHALL BE LIABLE, BEYOND THE AMOUNT OF ANY SUM RECEIVED IN PAYMENT BY THAT PARTNER FOR THIS DOCUMENT, WITH RESPECT TO ANY CLAIM, AND IN NO EVENT SHALL oneM2M BE LIABLE FOR LOST PROFITS OR OTHER INCIDENTAL OR CONSEQUENTIAL DAMAGES. oneM2M EXPRESSLY ADVISES ANY AND ALL USE OF OR RELIANCE UPON THIS INFORMATION PROVIDED IN THIS DOCUMENT IS AT THE RISK OF THE USER.

Contents

1 Scope 4

2 References 4

2.1 Normative references 4

2.2 Informative references 4

3 Definitions and abbreviations 5

3.2 Abbreviations 5

4 Conventions 5

5 Introduction to OCF interworking 5

6 OCF interworking architecture 6

6.1 Exposure of OCF Functions to the oneM2M System 6

6.1.1 Summary of Interworking Architecture for exposure of OCF Functions 6

6.1.2 OCF-IPE Responsibilities to support exposure of OCF Functions to the oneM2M System 8

6.2 Exposure of native oneM2M services to an OCF Proximal IoT Network 9

6.2.1 Summary of Interworking Architecture for exposure of native oneM2M services 9

6.2.2 OCF-IPE Responsibilities to support exposure of oneM2M services to an OCF Proximal IoT Network 11

7 Representation of OCF and/or oneM2M functions 12

7.1 Representation of OCF functions by oneM2M resources 12

7.1.1 Representation of OCF Devices by oneM2M <*node*> resources 12

7.1.2 Representation of OCF Devices by oneM2M <*AE*> resources 14

7.1.2.1 OCF Clients 14

7.1.2.2 OCF Servers 15

7.1.3 Representation of OCF services by oneM2M <*flexContainer*> resources 16

7.2 Representation of oneM2M services by OCF resources 17

8 OCF Interworking Procedures 18

8.1 Procedures supporting exposure of OCF Functions to the oneM2M Sytsem 18

8.1.1 Determination of OCF Functions to be exposed to the oneM2M System 18

8.1.1.1 Summary 18

8.1.1.2 Provisioning 18

8.1.1.3 Discovery 18

8.1.1.4 On demand determination 18

8.1.2 Creation / Deletion of oneM2M Resource representing exposed OCF Functions 18

8.1.3 Mirroring state of exposed OCF Functions in oneM2M Resources 19

8.1.4 Detection of requests to execute OCF Functions and invocation thereof 19

8.2 Procedures supporting exposure of native oneM2M services to an OCF Proximal IoT Network 19

8.2.1 Determination of oneM2M services to be exposed to the OCF Proximal IoT Network 19

8.2.1.1 Summary 19

8.2.1.2 Provisioning 19

8.2.1.3 Discovery 19

8.2.1.4 On demand determination 19

8.2.2 Instantiation / removal of OCF Servers representing exposed oneM2M services 19

8.2.3 Mirroring state of oneM2M Resources representing exposed oneM2M services in OCF resources hosted on OCF Servers 19

8.2.4 Detection of requests to consume exposed oneM2M services and execution thereof 19

Annex A (Informative) 20

Examples 20

History 20

# 1 Scope

The present document specifies the detailed methods for oneM2M and OCF interworking using the architecture identified in TS-0033 [5] and annex F of oneM2M TS-0001 [2] for the following scenario:

* Interworking with full mapping of the semantics of the non-oneM2M data model to Mca, see scenario number 1 listed in Clause F.2 of oneM2M TS-0001 [2]. This is also in line with the interworking concepts specified in TS-0033 [5].

This interworking scenario allows for interworking between OCF devices and oneM2M entities purely based on the common understanding of aligned information models – such as the models defined in TS-0023 [6]. There is no limitation regarding the direction of exposure of services: Services provided by OCF devices (OCF servers) can be exposed to oneM2M entities or vice versa. The oneM2M entities do not need to be aware of any details of the OCF protocols or interfaces.

# 2 References

## 2.1 Normative 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 necessary for the application of the present document.

[1] oneM2M TS-0011: "Common Terminology".

[2] oneM2M TS-0001: "Functional Architecture".

[3] OIC-Core-Specification-V1.0.0: "OIC Core Specification".

[4] oneM2M TS-0003: "Security solutions".

[5] oneM2M TS-0033: "Interworking Framework".

[6] oneM2M TS-0023: “Home Appliances Information Model and Mapping”.

[7] OCF-Core-Specification-V1.3.0.

NOTE: Available at https://openconnectivity.org/specs/OCF\_Core\_Specification\_v1.3.0.pdf

[8] OCF Device-Specification-V1.3.0.

NOTE: Available at https://openconnectivity.org/specs/OCF\_Device\_Specification\_v1.3.0.pdf

## 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 reference 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>.

# 3 Definitions and abbreviations

For the purposes of the present document, the terms and definitions given in oneM2M TS-0011 [1], oneM2M TS‑0001 [2], and oneM2M TS-0033 [5] apply.

**OCF Client**: A logical entity that accesses a Resource on an OCF Server.

**OCF Device**: A logical entity that assumes one or more roles (e.g., OCF Client, OCF Server).

**OCF Framework**: A set of related functionalities and interactions defined in the OCF Core Specification [7], which enable interoperability across a wide range of networked devices, including the Internet of Things.

**OCF-IPE**: IPE providing interworking functions for OCF-oneM2M interworking.

**OCF Physical Entity:** An aspect of the physical world that is exposed through an OCF Device. An example of an OCF Physical Entity is an LED.

**OCF Platform**: a physical device containing one or more Devices

**OCF Resource:** Represents an OCF Entity modelled and exposed by the OCF Framework

**OCF Server**: An OCF Device with the role of providing access to OCF Resource state information and facilitating remote interaction with those resources.

**OCF Functions**: Services or Device information provided by one or more OCF Servers by exposing access to OCF Resources via OCF-specified interfaces.

NOTE: A term defined in the present document takes precedence over the definition of the same term, if any, in oneM2M TS-0011 [1], oneM2M TS-0001 [2], and oneM2M TS-0033 [5].

## 3.2 Abbreviations

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

ACP Access Control Policy

AE Application Entity

AE-ID Application Entity Identifier

CBOR Concise Binary Object Representation

CMDH Communication Management and Delivery Handling

CSE Common Services Entity

IPE Interworking Proxy Entity

JSON JavaScript Object Notation

OIC Open Interconnect Consortium

URI Uniform Resource Identifier

XML eXtensible Markup Language

# 4 Conventions

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

# 5 Introduction to OCF interworking

OCF specifies an architecture enabling resource-based interactions among OCF Devices, see OCF-Core-Specification-V1.0.0 [7]. OCF Devices can expose aspects of the physical world like a lightbulb and/or logical entities like an application. The OCF Devices in the OCF architecture can provide services – in that case they play the role of an OCF Server – and/or consume services – in that case they play the role of an OCF Client.

This present document specifies details of the interworking with OCF Devices on the information model layer, see oneM2M TS-0033 [5], and the implications on how to represent and execute external OCF functions with means of resource instances in the oneM2M system and vice versa.

Service provided and/or consumed on either side (OCF versus oneM2M) are represented by resources hosted by service providers (OCF Servers or oneM2M CSEs) and accessible to service consumers (OCF Clients or oneM2M AEs/CSEs). In order to support seamless interworking across boundaries of OCF & oneM2M deployments, OCF and oneM2M have aligned several information models for such services so that they became technology agnostic. Definitions of such information models are contained in the following specifications:

* oneM2M TS-0023 [6].
* OCF Device-Specification-V1.3.0 [8].

In the remainder of this present document it is assumed that mutual exposure of services between oneM2M and OCF deployments is restricted to services represented by resource types defined in these specifications.

In line with oneM2M TS-0033 [5], this present document specifies two major directions of mutual OCF / oneM2M interworking:

* Expose OCF Functions provided by OCF Servers to oneM2M entities by using one or more OCF-IPE(s) which are acting as OCF Client(s). For this direction - OCF services exposed to oneM2M - the OCF-IPE(s) are creating and managing the oneM2M resources representing exposed services provided by OCF Servers and provide the required procedures to allow consumption of OCF services on the oneM2M side. Details on this exposure direction are specified in Claus 6.1 of the present document.
* Expose services provided by oneM2M entities by using one or more OCF-IPE(s) which are acting as OCF Server(s). For this direction – oneM2M services provided to OCF – the OCF-IPE(s) are interacting with already created oneM2M resource representations that are offering the native oneM2M services and provide the required procedures to allow consumption of the oneM2M services on the OCF side. Details on this exposure direction are specified in Claus 6.1 of the present document.

While it is possible that a single OCF-IPE can simultaneously support exposure of OCF Functions to oneM2M and exposure of oneM2M services to OCF, it is not mandated to implement OCF-IPEs in that way. It is also possible to instantiate separate OCF-IPEs for the different exposure directions. This is an implementation choice. In case that exposure in one direction needs to be correlated with a different Service Subscription Profile than for the other direction, separate OCF-IPEs are required. Details on the representation of OCF Functions by oneM2M Resources are defined in Clause 7 of the present document.

# 6 OCF interworking architecture

## 6.1 Exposure of OCF Functions to the oneM2M System

### 6.1.1 Summary of Interworking Architecture for exposure of OCF Functions

An OCF-IPE exposing OCF Functions to the oneM2M System is responsible for the creation of oneM2M Resources representing the exposed OCF Functions on its own Registrar CSE. A single OCF-IPE may expose OCF Functions provided by one or more OCF Servers to the oneM2M System. A high-level summary of the relationship of OCF Servers providing OCF Functions to be exposed to oneM2M, one or more OCF-IPE(s) and sets of oneM2M Resources representing the exposed OCF Functions is depicted in Figure 6.1.1-1 below.

 **Figure 6.1.1-1: Summary of Interworking Architecture for Exposure OCF → oneM2M**

More than one OCF-IPE may be instantiated for exposure of multiple non-overlapping sets of OCF Functions to the oneM2M System. In particular, if OCF Functions provided by different sets of OCF Servers shall be exposed to the oneM2M System and there is a deployment requirement to differentiate sets of OCF Functions provided by different OCF Servers – e.g. for the purpose of correlating them with different Service Subscription Profiles – then instantiating more than one OCF-IPE is required. Further details on the implications using more than one OCF-IPE for exposure of OCF Functions to the oneM2M System are defined in Clause 7.1.

A specific instance of an OCF-IPE exposing OCF Functions to the oneM2M System shall play the role of a single OCF Client on the OCF side. If there is a deployment requirement to differentiate multiple OCF Clients who act as interworking proxies for exposing a common set of OCF Functions to the oneM2M System – for instance due to the need to assign different access control properties to them on the OCF side – then the OCF-IPE exposing those OCF Functions would need to play the role of multiple OCF Clients. However, mapping rules between the identifiers of Originators on the oneM2M side who attempt to consume the exposed OCF Functions and OCF Client identifiers to use for triggering the execution of the exposed OCF Functions on the OCF side are not specified in this present document. Therefore, it is assumed that each OCF-IPE is only playing the role of a single OCF Client on the OCF side. This restriction implies that all requests originating from oneM2M entities which are successfully modifying the corresponding oneM2M Resources and then get translated by a specific OCF-IPE to the corresponding interactions with OCF Servers are treated on the OCF side as coming from one single OCF Client and, therefore, will be handled by OCF Servers with the same access control properties, irrespective of the identifier of the Originator on the oneM2M side. However, by means of setting appropriate Access Control Privileges on the oneM2M side, it is certainly possible to define which entities on the oneM2M side will get which mode of access to the exposed OCF services. Support of multiple OCF Clients for a single OCF-IPE is for further study and might be subject of future releases of the present document.

After creation of oneM2M Resources representing exposed OCF Functions by a specific OCF-IPE, this particular OCF-IPE is also responsible for monitoring relevant parts of these created resources in order to detect any operations that need to be translated into an execution of corresponding OCF Functions. In case such an operation on the previously created oneM2M Resources gets detected, the corresponding Function on the OCF side shall be executed by the OCF-IPE.

State changes within the OCF Functions exposed by a specific OCF-IPE to the oneM2M System that are impacting the state of the corresponding oneM2M Resources which were previously created by that specific OCF-IPE need to be monitored on the OCF side and shall be translated by that specific OCF-IPE into corresponding state changes of the associated oneM2M Resources.

### 6.1.2 OCF-IPE Responsibilities to support exposure of OCF Functions to the oneM2M System

When exposing OCF Functions to the oneM2M System, an OCF-IPE shall be responsible to support the following procedures:

1. Determination of OCF Functions to be exposed to the oneM2M System  
   The OCF-IPE needs to determine which OCF Functions need to be exposed to the oneM2M System. This determination can be done in different ways (e.g. through provisioning, discovery, on-demand signalling, or combinations thereof). Further details of this procedure are defined in Clause 8.1.1
2. Creation / Deletion of oneM2M Resource representing exposed OCF Functions  
   The OCF-IPE needs to perform creation / deletion of resource instances representing OCF Functions according to the – possibly dynamically changing – need to expose them to the oneM2M System using resource types that are have been aligned between oneM2M and OCF in order to become technology independent. Further details on this procedure are defined in Clause 8.1.2 Resource types that meet the requirement to be technology agnostic between OCF and oneM2M are defined in:
   * OCF Device-Specification-V1.3.0 [8] and oneM2M TS-0023 [6].
3. Mirroring state of exposed OCF Functions in oneM2M Resources  
   An OCF-IPE exposing OCF Functions provided by OCF Servers is responsible to modify the resource instances it has created in order to represent the exposed OCF Functions according to any state changes occurring in the corresponding OCF Servers. This implies that such an OCF-IPE shall monitor the state of the associated OCF Servers and upon detection of OCF Server state changes relevant for the exposed OCF Functions the OCF-IPE shall modify the previously created oneM2M Resources accordingly. Further details on this procedure are defined in Clause 8.1.3
4. Detection of requests to execute OCF Functions and invocation thereof  
   The OFC-IPE is responsible for monitoring relevant changes in the resource instances it has previously created for the purpose of representing the exposed OCF Functions. Upon detection of any valid operation meant to trigger the execution of the exposed OCF Functions, the OCF-IPE is responsible for the invocation of the corresponding OCF Functions via its own OCF Client. Further details of this procedure are defined in Clause 8.1.4.

The set of responsibilities of the OCF-IPE when exposing OCF Functions to the oneM2M system is summarized in Figure 6.1.2-1 below. The dashed boxes describe optional/alternative means to determine the set of exposed OCF Functions. Note that, in this Figure one OCF-IPE is responsible for all interworking procedures to support exposure of OCF Functions to oneM2M. More than one OCF-IPE may be used to expose different sets of OCF Functions to oneM2M. Details on how to map exposed OCF Function into oneM2M Resources are defined in clause 7.

 **Figure 6.1.2-1: Exposure of OCF Functions to the oneM2M System**

## 6.2 Exposure of native oneM2M services to an OCF Proximal IoT Network

### 6.2.1 Summary of Interworking Architecture for exposure of native oneM2M services

An OCF-IPE exposing oneM2M services to an OCF Proximal IoT Network is responsible for the creation of OCF Server instances in the OCF Proximal IoT Network representing the exposed oneM2M services. A single OCF-IPE may expose one or more oneM2M services to the OCF Proximal IoT Network. A high-level summary of the relationship of oneM2M Resources providing the services to be exposed to the OCF Proximal IoT Network, one or more OCF-IPE(s) and sets of OCF Servers representing the exposed oneM2M services is depicted in Figure 6.2.1-1 below.

 **Figure 6.2.1-1: Summary of Interworking Architecture for Exposure oneM2M → OCF**

More than one OCF-IPE may be instantiated for exposure of multiple non-overlapping sets of oneM2M services to the OCF Proximal IoT Network. In particular, if there is a deployment requirement to differentiate the exposure of different sets of oneM2M services – e.g. for the purpose of correlating them with different Service Subscription Profiles – then instantiating more than one OCF-IPE is required. Further details on the implications using more than one OCF-IPE for exposure of oneM2M services to the OCF Proximal IoT Network are defined in Clause 7.1.

A specific oneM2M service shall only be exposed to a given OCF Proximal IoT Network by at most one instance of an OCF-IPE connected to that OCF Proximal IoT Network. If there is a deployment requirement to differentiate multiple AE instances who act as interworking proxies on behalf of OCF Clients for exposing a common set of oneM2M services to the OCF Proximal IoT Network – for instance due to the need to assign different access control privileges to them on the oneM2M side – then the exposure of that common set of oneM2M services would require multiple instances of OCF-IPEs with a shared set of OCF Servers exposing the same set of oneM2M services to a given OCF Proximal IoT Network. However, mapping rules between the identifiers of OCF Clients who attempt to consume the exposed oneM2M services and the respective OCF-IPE AE-IDs to use for triggering the execution of the exposed oneM2M services are not specified in this present document. Therefore, it is assumed that each exposed oneM2M service is only interacting with one OCF-IPE for a given OCF Proximal IoT Network. This restriction implies that all requests originating from OCF Clients received by any of the OCF Servers instantiated by a given OCF-IPE which are meant to be translated by the OCF-IPE to the corresponding operations on oneM2M Resources representing the exposed oneM2M services are treated on the oneM2M side as coming from one single oneM2M AE and, therefore, will be handled by oneM2M CSEs with the same access control privileges, irrespective of the identifier of the requesting OCF Client. However, by means of setting appropriate access control properties on the OCF side – i.e. access control governing the acceptance or requests at the OCF Servers instantiated by the OCF-IPE – it is certainly possible to define which OCF Client will get which mode of access to the exposed oneM2M services. Therefore, access control can be imposed on OCF Client basis if the appropriate access control properties are provisioned properly into the OCF Servers instantiated by the OCF-IPE. Support of multiple OCF-IPEs exposing the same set of oneM2M services is for further study and might be subject of future releases of the present document.

After creation of OCF Server instances representing the exposed oneM2M services by a specific OCF-IPE, this particular OCF-IPE is also responsible for monitoring incoming OCF Client requests reaching these created OCF Servers in order to detect any requests that need to be translated into a corresponding operation on oneM2M Resources representing the exposed oneM2M services. In case such an request directed to the previously created OCF Servers gets detected, the corresponding operation(s) on the oneM2M side shall be executed by the OCF-IPE.

State changes within the oneM2M Resources representing the oneM2M services exposed to the OCF Proximal IoT Network by a specific OCF-IPE that are impacting the state of the corresponding resources on any of the OCF Servers that were previously created by that specific OCF-IPE need to be detected on the oneM2M side and shall be translated by that specific OCF-IPE into corresponding state changes of the associated resources in the OCF Servers.

### 6.2.2 OCF-IPE Responsibilities to support exposure of oneM2M services to an OCF Proximal IoT Network

When exposing oneM2M services to an OCF Proximal IoT Network, an OCF-IPE shall be responsible to support the following procedures:

1. Determination of oneM2M services to be exposed to the OCF Proximal IoT Network  
   The OCF-IPE needs to determine which oneM2M services need to be exposed to the OCF Proximal IoT Network. This determination can be done in different ways (e.g. through provisioning, discovery, on-demand signalling, or combinations thereof). Further details of this procedure are defined in Clause 8.2.1
2. Instantiation / removal of OCF Servers representing exposed oneM2M services  
   The OCF-IPE needs to perform instantiation / removal of OCF Server instances representing oneM2M services according to the – possibly dynamically changing – need to expose them to the OCF Proximal IoT Network using OCF Servers hosting OCF resource types that are have been aligned between oneM2M and OCF in order to become technology independent. Further details on this procedure are defined in Clause 8.2.2. OCF resource types and equivalent oneM2M resource types that meet the requirement to be technology agnostic between OCF and oneM2M are defined in:
   * oneM2M TS-0023 [6] and OCF Device-Specification-V1.3.0 [8].
3. Mirroring state of oneM2M Resources representing exposed oneM2M services in OCF resources hosted on OCF Servers  
   An OCF-IPE exposing oneM2M services to an OCF Proximal IoT Network is responsible to modify the resource instances in the OCF Servers it has instantiated in order to represent the exposed oneM2M services according to any state changes occurring in the corresponding oneM2M Resources. This implies that such an OCF-IPE shall monitor the state of the associated oneM2M Resources and upon detection of oneM2M Resource state changes relevant for the exposed oneM2M services, the OCF-IPE shall modify the corresponding resource state of OCF resources hosted in the previously instantiated OCF Servers accordingly. Further details on this procedure are defined in Clause 8.2.3
4. Detection of requests to consume exposed oneM2M services and execution thereof  
   The OFC-IPE is responsible for monitoring relevant changes in the resources hosted by the OCF Servers the OCF-IPE has previously instantiated for the purpose of representing the exposed oneM2M servies. Upon detection of any valid operation on the OCF side meant to trigger the consumption of the associated exposed oneM2M services, the OCF-IPE is responsible for the requesting the corresponding oneM2M operation on the oneM2M Resources representing the exposed oneM2M service to be consumed. Further details of this procedure are defined in Clause 8.2.4.

The set of responsibilities of the OCF-IPE when exposing oneM2M services to an OCF Proximal IoT Network is summarized in Figure 6.2.2-1 below. The dashed boxes describe optional/alternative means to determine the set of exposed oneM2M services. Note that, in this Figure one OCF-IPE is responsible for all interworking procedures to support exposure of oneM2M services to an OCF Proximal IoT Network. More than one OCF-IPE may be used to expose different sets of oneM2M services to the same OCF Proximal IoT Network. Details on how to map exposed oneM2M services into OCF resources are defined in clause 7.



**Figure 6.2.2-1: Exposure of native oneM2M functions to the OCF Proximal IoT Network**

# 7 Representation of OCF and/or oneM2M functions

## 7.1 Representation of OCF functions by oneM2M resources

### 7.1.1 Representation of OCF Devices by oneM2M <*node*> resources

An OCF Device which provides functions that are exposed to the oneM2M system can be represented by a oneM2M <*node*> resource that contains device-specific information which can be used e.g. for the purpose of device management. Whether a specific OCF device shall be represented by a oneM2M <*node*> resource depends on the type of device:

* OCF Devices providing services that are exposed to the oneM2M system using specializations of [*flexContainer*] resources as defined in oneM2M TS-0023 [6], see also clause 7.1.3 below, shall be represented by oneM2M <*node*> resources as well. Each instance of a [*flexContainer*] resource representing a service provided by a specific OCF device is linked via the *nodeLink* attribute to a specific <*node*> resource instance that represents this specific OCF device.

For OCF Devices that are not matching with any of the listed categories, the present document does not specify any normative procedure to represent such OCF Devices as oneM2M <*node*> resources.

When a <*node*> resource is used to represent an OCF Device, it actually needs to reflect the details of the OCF Platform that hosts the OCF Device, Attributes of the <*node*> resource listed in Table 7.1.1-1 below shall have the specified value settings in Table 7.1.1-1. All other attributes of the <*node*> resource shall be used as specified in oneM2M TS-0001 [2]. A <*node*> resource representing an OCF Device shall include exactly one [*deviceInfo*] child resource instance that represents specifics of the OCF Platform hosting the OCF Device. All other child resources of the <*node*> resource shall be used as specified in oneM2M TS-0001 [2]. Attributes of a [*deviceInfo*] resource listed in Table 7.1.1-2 below shall have the specified value settings in Table 7.1.1-2 for the [*deviceInfo*] child resource instance representing the OCF Platform of the OCF Device. All other attributes or child resources of the [*deviceInfo*] resource shall be used as specified in oneM2M TS-0001 [2].

Table 7.1.1-1: Attribute settings for <*node*> resources representing an OCF Platform.

| Attribute Name | Setting |
| --- | --- |
| *resourceName* | This attribute shall be set to the value of the “pi” property (Platform ID) of the OCF resource with the pre-defined URI “/oic/p” of the respective OCF Device as defined in the OCF Core Specification [7]. |
| *labels* | This attribute shall include a Key:Value pair equal to “Iwked-Technology:OCF”. Other Key:Value pairs may also be present. |
| *nodeID* | This attribute shall be set to the value of the “pi” property (Platform ID) of the OCF resource with the pre-defined URI “/oic/p” of the respective OCF Device as defined in the OCF Core Specification [7]. |

Table 7.1.1-2: Attribute settings for a [*deviceInfo*] child-resource of a <*node*> resources representing an OCF Platform.

| Attribute Name | Setting |
| --- | --- |
| *labels* | This attribute shall include a Key:Value pair equal to “Iwked-Technology:OCF”. Other Key:Value pairs may also be present. |
| *deviceLabel* | This attribute shall be set to the value of the “pi” property (Platform ID) of the OCF resource with the pre-defined URI “/oic/p” of the respective OCF Device as defined in the OCF Core Specification [7]. |
| *manufacturer* | This attribute shall be set to the value of the “mnmn” property (Manufacturer Name) of the OCF resource with the pre-defined URI “/oic/p” of the respective OCF Device as defined in the OCF Core Specification [7], if present. |
| *manufacturerDetailsLink* | This attribute shall be set to the value of the “mnml” property (Manufacturer Details Link) of the OCF resource with the pre-defined URI “/oic/p” of the respective OCF Device as defined in the OCF Core Specification [7], if present. |
| *manufacturingDate* | This attribute shall be set to the value of the “mndt” property (Date of Manufacture) of the OCF resource with the pre-defined URI “/oic/p” of the respective OCF Device as defined in the OCF Core Specification [7], if present. |
| *model* | This attribute shall be set to the value of the “mnmo” property (Model Number) of the OCF resource with the pre-defined URI “/oic/p” of the respective OCF Device as defined in the OCF Core Specification [7], if present. |
| *subModel* | This attribute shall be set to the value of the “mnpv” property (Platform Version) of the OCF resource with the pre-defined URI “/oic/p” of the respective OCF Device as defined in the OCF Core Specification [7], if present. |
| *deviceType* | This attribute shall be set to the value “OCF Platform” |
| *fwVersion* | This attribute shall be set to the value of the “mnfv” property (Firmware Version) of the OCF resource with the pre-defined URI “/oic/p” of the respective OCF Device as defined in the OCF Core Specification [7], if present. |
| *hwVersion* | This attribute shall be set to the value of the “mnhw” property (Hardware Version) of the OCF resource with the pre-defined URI “/oic/p” of the respective OCF Device as defined in the OCF Core Specification [7], if present. |
| *osVersion* | This attribute shall be set to the value of the “mnos” property (OS Version) of the OCF resource with the pre-defined URI “/oic/p” of the respective OCF Device as defined in the OCF Core Specification [7], if present. |
| *systemTime* | This attribute shall expose the system time of the device in line with the value of the “st” property (SystemTime) of the OCF resource with the pre-defined URI “/oic/p” of the respective OCF Device as defined in the OCF Core Specification [7], if present. |
| *supportURL* | This attribute shall be set to the value of the “mnsl” property (Support link) of the OCF resource with the pre-defined URI “/oic/p” of the respective OCF Device as defined in the OCF Core Specification [7], if present. |

### 7.1.2 Representation of OCF Devices by oneM2M <*AE*> resources

#### 7.1.2.1 OCF Clients

The present document does not specify any normative procedure to represent OCF Clients as oneM2M <*AE*> resources. Since the OCF specifications do not define any procedure for OCF Clients needing to register themselves with any other entity or needing to respond to any discovery procedures before trying to consume services provided by OCF Servers, it is not possible to determine which OCF Clients exist before they make any requests to consume services provided by OCF Servers. Therefore, it would be very complex to associate OCF Clients in an OCF Proximal IoT Network with oneM2M <*AE*> resources. In order to do that, the OCF-IPE would have to detect OCF Clients dynamically at the time when they try to request any services from that OCF-IPE and then create on the fly some corresponding <*AE*> resources. Furthermore, such dynamically created <*AE*> resources would also require adjustment of access control privileges when the corresponding AEs would need to access oneM2M resources. For these reasons, there are no provisions in this present document to represent OCF Clients by oneM2M <*AE*> resources.

#### 7.1.2.2 OCF Servers

OCF Devices playing the role of an OCF Server can be represented by an <*AE*> resource that corresponds to an OCF-IPE, see Figure 6.1.1-1. Depending on implementation and deployment needs, one or more OCF Servers can be represented on the oneM2M side by a single OCF-IPE with a single <*AE*> resource.

In case an OCF Server is intended to be exposed to the oneM2M system, exactly one OCF-IPE shall be responsible for the exposure of services provided by that specific OCF Server in a given OCF Proximal Network. Therefore, a specific OCF Server shall be represented by at most one OCF-IPE, i.e. by at most one <*AE*> resource.

In case of OCF Servers, the details of the OCF Server can be determined in the OCF Proximal IoT Network by means of executing OCF discovery and introspection procedures. For an OCF-IPE representing a specific OCF-Server, any exposed services provided by that OCF Server that are meant to be exposed to the oneM2M System shall be represented by child resources of the <*AE*> resource representing that specific OCF-IPE, see Clause 7.1.3 below.

Representing one or more OCF Servers by the <*AE*> resource associated with an OCF-IPE has the following implications:

* The set of represented OCF Servers represented by a specific OCF-IPE can be associated with M2M Service Subscriptions just like any other oneM2M Application
* The OCF-IPE allowing interworking with specific type of OCF Servers can be registered in the App-ID registry

Attributes of an <*AE*> resource listed in Table 7.1.2-1 below shall have the specified value settings in Table 7.1.2-1 when that <*AE*> resource represents and OCF-IPE. All other attributes of the <*AE*> resource shall be used as specified in oneM2M TS-0001 [2].

Table 7.1.2-1: Attribute settings for <*AE*> resources representing an OCF-IPE.

| Attribute Name | Setting |
| --- | --- |
| *labels* | In order to indicate the supported external technology and which specific OCF Server(s) are represented by a specific OCF-IPE, the *labels* attribute of the <*AE*> resource of an OCF-IPE exposing OCF Server Functions to oneM2M shall contain the following information:   * A Key:Value pair set to “Iwked-Technology:OCF” indicating that this AE supports interworking with OCF entities. * A Key:Value pair indicating the specific OCF Servers being exposed to the oneM2M system by this specific OCF-IPE. The Key should be set to “Iwked-Entity-IDs”. The Value shall contain a coma-separated list of IDs in square brackets, e.g.. “[ID1, ID2, ID3]”, where each listed ID identifies one specific exposed OCF Server and is equal to the value of the “piid” property (Protocol Independent ID) of the device resource with the pre-defined URI “/oic/d” of the respective OCF Server as defined in the OCF Core Specification [7].   Other Key:Value pairs may also be present. |

### 7.1.3 Representation of OCF services by oneM2M <*flexContainer*> resources

Services provided by OCF Servers that are intended to be exposed to the oneM2M system shall be represented by specializations of <*flexContainer*> resources in order to be exposed to the oneM2M system. Services provided by a specific OCF Server and intended to be exposed to the oneM2M system shall be represented by <*flexContainer*> child resources of the <*AE*> resource instance that represents the OCP-IPE responsible for the exposure of services provided by that specific OCF Server to oneM2M.

Services provided by OCF Servers are characterized by the device types defined in the OCF Device Specifications [8]. For each device type there is a minimum set of OCF resources defined in the OCF Device Specifications [8]. Exposure of a specific service provided by an OCF Server is accomplished by representing the corresponding set of OCF resources hosted on the OCF Server with matching oneM2M <*flexContainer*> resources hosted on the Registrar-CSE of the OCF-IPE responsible for the exposure of that OCF Server as child resources of the <*AE*> resources representing the OCF-IPE.

In oneM2M TS-0023 [6] a set of information models is defined which describes the exposure of services provided by oneM2M Devices via specializations of <*felxContainer*> resources. The specification oneM2M TS-0023 [6] also contains a mapping between OCF Device Types and the corresponding oneM2M Devices.

When exposing services of an OCF Server that complies with any of the OCF Device Types for which oneM2M TS-0023 [6] includes a normative mapping to oneM2M Devices, an instance of the respective <*flexContainer*> specialization as defined in oneM2M TS-0023 [6] and any required child resources shall be used to represent the exposed service provided by an OCF Server.

When exposing services of an OCF Server that does not comply with any of the OCF Device Types for which oneM2M TS-0023 [6] includes a normative mapping to oneM2M Devices, an instance of a customized <*flexContainer*> resource shall be used. In line with <*flexContainer*> resources defined in oneM2M TS-0023 [6], such customized <*flexContainer*> resources shall include a *nodeLink* attribute which links to a <*node*> resource that represents the OCF Platform hosting the exposed OCF Server. Further details of such customized < *flexContainer*> resources are not in scope of this specification.

Attributes of an <*flexContainer*> resource representing services provided by an OCF Server as listed in Table 7.1.3-1 below shall have the specified value settings in Table 7.1.3-1. All other attributes of the <*flexContainer*> resource shall be used as specified in oneM2M TS-0001 [2] or in oneM2M TS-0023 [6] if applicable.

Table 7.1.2-1: Attribute settings for <*flexContainer*> resources representing services provided by an OCF Server.

| Attribute Name | Setting |
| --- | --- |
| *labels* | In order to indicate the supported external technology and which specific OCF Server is represented by this <*flexContainer*> resource, the *labels* attribute of the <*flexContainer*> resource shall contain the following information:   * A Key:Value pair set to “Iwked-Technology:OCF” indicating that this <*flexContainer>* supports interworking with OCF entities. * A Key:Value pair indicating the specific OCF Server being exposed to the oneM2M system by this specific <*flexContainer>*. The Key should be set to “Iwked-Entity-ID”. The Value shall be equal to the value of the “piid” property (Protocol Independent ID) of the device resource with the pre-defined URI “/oic/d” of the respective OCF Server as defined in the OCF Core Specification [7].   Other Key:Value pairs may also be present. |
| *nodeLink* | The *resource identifier* of a *<node>* resource that stores the node specific information of the node on which the OCF Server resides which provides services that are exposed by this <*flexContainer*> resource. See clause 7.1.1 for details on representing OCF Devices by a <*node*> resource |

The procedural aspects on how to create, delete and interact with <*flexContainer*> resources for the purpose of exposing OCF Functions to the oneM2M system are specified in Claus 8.1 of the present document.

## 7.2 Representation of oneM2M services by OCF resources

Services provided by oneM2M Devices can be exposed to an OCF Proximal IoT Network by an OCF-IPE acting as one or more OCF Servers – see clause 6.2.1.

For a service intended to be exposed to the OCF Proximal IoT Network and provided by a oneM2M Device represented by a <*flexContainer*> resource, the corresponding OCF Device Type shall be determined according to the normative mapping defined in the oneM2M specification(s)

* oneM2M TS-0023 [6].

Then for each OCF Device Type that has been determined in the mapping, the minimum set of resources for that OCF Device Type as specified in the OCF Device Specification [8] shall be exposed to the OCF Proximal IoT Network by the OCF-IPE responsible for the exposure. The OCF-IPE responsible for the exposure to the OCF Proximal IoT Network shall act as an OCF Server for each identified OCF Device Type.

Each OCF Server which is instantiated by an OCF-IPE needs to provide a minimum set of resources depending on the Device Type it represents. Common to all OCF Devices is the need to support the following mandatory core resources:

* “/oic/res” for discovery of resources hosted by the OCF Server
* “/oic/p” for discovery of platform specific parameters of the node hosting the OCF Server
* “/oic/d” for discovery of device information

Beyond these core resources, an OCF Server shall also support the resources required by the specific OCF Device Type. For the mandatory core resources supported by all OCF Servers, the property settings defined in Table 7.2-1 and Table 7.2-2 shall apply. Properties not listed in Table 7.2-1 and Table 7.2-2 may be present while the present specification does not define any settings for those properties.

Table 7.2-1: Property settings for a “/oic/p” resource provided by an OCF-IPE acting as an OCF Server.

| Property Name | Setting |
| --- | --- |
| pi | This property shall be generated in line with the OCF Core Specification [7]:Unique identifier for the physical  platform (UIUID); this shall be a UUID in accordance with  IETF RFC 4122. It is recommended that the UUID be created using the random generation  scheme (version 4 UUID) specific in the RFC.  If the OCF-IPE that instantiated this OCF Server is instantiating multiple OCF Servers, it shall use the same value for all pi properties of the /oic/p resources of all its instantiated OCF Servers. |
| mnmn | In case the <*AEr*> resource representing the OCF-IPE that instantiated this OCF Server is linked to a <*node*> resource that has a [*deviceInfo*] child resource that includes a *manufacturer* attribute, the value of that *manufacturer*  shall be used for this mnmn property. Otherwise, the string to be used for this property is implementation dependent but needs to be present.. |

Table 7.2-2: Property settings for a “/oic/d” resource provided by an OCF-IPE acting as an OCF Server.

| Property Name | Setting |
| --- | --- |
| n | An implementation dependent OCF Device name prefixed by the string “oneM2M-“. |
| icv | Spec version of the OCF Core Specification this OCF Server instance of the OCF-IPE is implemented to, The syntax is "ocf.<major>.<minor>.<subversion>”  where <major>, <minor>,and <sub-version> are the major,  minor and sub-version numbers of the OCF Core Specification, respectively. |
| di | Unique identifier of the device in line with the requirements in the OCF Core Specifications [7]. |
| dmv | Version of the Resource Specification to which this OCF Server instance of the OCF-IPE is implemented. This needs to be in line with the requirements on the dmv property defined in the OCF Core Specifications [7]. |
| piid | A unique and immutable identifier for this OCF Server instance of the OCF-IPE generated in line with the OCF Core Specifications [7] |

# 8 OCF Interworking Procedures

## 8.1 Procedures supporting exposure of OCF Functions to the oneM2M Sytsem

### 8.1.1 Determination of OCF Functions to be exposed to the oneM2M System

#### 8.1.1.1 Summary

TBD

#### 8.1.1.2 Provisioning

TBD

#### 8.1.1.3 Discovery

TBD

#### 8.1.1.4 On demand determination

TBD

### 8.1.2 Creation / Deletion of oneM2M Resource representing exposed OCF Functions

TBD

### 8.1.3 Mirroring state of exposed OCF Functions in oneM2M Resources

TBD

### 8.1.4 Detection of requests to execute OCF Functions and invocation thereof

TBD

## 8.2 Procedures supporting exposure of native oneM2M services to an OCF Proximal IoT Network

### 8.2.1 Determination of oneM2M services to be exposed to the OCF Proximal IoT Network

#### 8.2.1.1 Summary

TBD.

#### 8.2.1.2 Provisioning

TBD

#### 8.2.1.3 Discovery

TBD

#### 8.2.1.4 On demand determination

TBD

### 8.2.2 Instantiation / removal of OCF Servers representing exposed oneM2M services

TBD

### 8.2.3 Mirroring state of oneM2M Resources representing exposed oneM2M services in OCF resources hosted on OCF Servers

TBD

### 8.2.4 Detection of requests to consume exposed oneM2M services and execution thereof

TBD

Annex A (Informative)

# Examples

# History

|  |  |  |
| --- | --- | --- |
| **Publication history** | | |
| V2.0.0 | 30-Aug-2016 | Release 2 - Publication |
| V2.0.1 | July 2017 | Clean-up done by ***editHelp!*** e-mail: <mailto:edithelp@etsi.org> |
| V3.0.0 | 21-Dec-2017 | Initial Releae 3-Version  Implementation of agreed CR in: ARC-2017-0410R01 |
|  |  |  |
|  |  |  |