|  |
| --- |
|  |

|  |
| --- |
| CHANGE REQUEST |
| Meeting ID:\* | SDS#49 |
| Source:\* | Cyrille Bareau, Orange, cyrille.bareau@orange.comBob Flynn, Exacta, bob.flynn@exactagss.comAndreas Kraft, Deutsche Telekom, Andreas.Kraft@t-systems.comMarianne Mohali, Orange, marianne.mohali@orange.com |
| Date:\* | 2021-02-03 |
| Reason for Change/s:\* | See the introduction. |
| CR against: Release\* | Release 4 |
| CR against: WI\* | [ ]  Active WI[ ]  MNT maintenance / < Work Item number(optional)>Is this a mirror CR? Yes [ ]  No [ ] mirror CR number: (Note to Rapporteur - use latest agreed revision)[x]  STE Small Technical Enhancements / < Work Item number (optional)>Only ONE of the above shall be ticked |
| CR against: TS/TR\* | TS-0001 4.8.0 |
| Clauses \* | Clauses 9.6.35, 9.6.59, 10.2.4.18, 10.2.4.30 |
| 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

The creation of a <flexContainerInstance> resource is triggered when the parent <flexContainer> resource is updated. However, this currently happens when 2 conditions apply:

1. The parent flexContainer owns at least one custom attribute.
2. The UPDATE request contains at least one of these custom attributes.

We want to add the possibility to create flexContainerInstances:

1. for flexContainers that have no custom attributes (this corresponds to a precise use case involving SDT Actions in TS-0023, that can be mapped to flexContainers with no custom attributes)
2. on demand, even if none of the current custom attributes are modified (can be used as an optimisation, for instance for historizing a temperature that is stable).

This document is a companion to CR SDS-2021-0003R02-TS-0004\_flexContainerInstance\_on\_empty\_UPDATE.

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

### 9.6.35 Resource Type *flexContainer*

The *<flexContainer>* resource type is a customizable container for data instances. It is a template for the definition of flexible specializations of data containers. Like a <*container*> resource, specializations of this *<flexContainer>* resource type are used to share information with other entities and potentially to track the data. While the <*container*> resources includes data to be made accessible to oneM2M entities inside <*contentInstance*> children, a specialization of the *<flexContainer>* resource includes associated content directly inside the <*flexContainer*> by means of zero or more [*customAttribute*] attribute(s). The attribute name and attribute data type of [*customAttribute*] attributes are defined explicitly for each specialization of <*flexContainer>*, i.e. the specific set of attribute name and type are defined in a corresponding XSD-file.

Example usage of *<flexContainer>*: As a specialization of <*flexContainer*> that includes two [customAttribute] attributes, named "temperature"(xs:float type) and "humidity"(xs:positiveInteger type) can be specified in some TS. The actual data types of [customAttribute] will be described both in the specification document or XSD file which are referred by the value of *containerDefinition* attribute.

If a <*flexContainer>* resource is created with at least one of the attributes *maxNrOfInstances, maxByteSize* or *maxInstanceAge,* then the hosting CSE shall automatically create a *copy* of the *<flexContainer>* resource in the form of a *<flexContainerInstance>* resource, child of the current *<flexContainer>* resource; and each time a <*flexContainer*> *custom* attribute is modified, a new <*flexContainerInstance>* child resource shall be added, which contains a copy of the *<flexContainer>* after update. In this case, virtual resources *<latest>* and *<oldest>* shall be available.

The *<flexContainer>* resource shall contain the child resource specified in table 9.6.35-1.

Table 9.6.35-1: Child resources of <*flexContainer*> resource

| **Child Resources of <*flexContainer*>** | **Child Resource Type** | **Multiplicity** | **Description** | ***<****flexContainer****Annc>* Child Resource Type** |
| --- | --- | --- | --- | --- |
| *[variable]* | *<semanticDescriptor>* | 0..n | See clause 9.6.30 | *<semanticDescriptor>, <semanticDescriptorAnnc>* |
| *[variable]* | *<subscription>* | 0..n | See clause 9.6.8 | *<subscription>* |
| *[variable]* | *<container>* | 0..n | See clause 9.6.6 | *<container>**<containerAnnc>* |
| *[variable]* | *<flexContainer>* | 0..n | <flexContainer> resource can include any of its specializations as child resource | *<flexContainer>**<flexContainerAnnc>* |
| *[variable]* | *<flexContainerInstance>* | 0..n | Timestamped copy of the *<flexContainer>* resource. | *<flexContainerInstance>* |
| *la* | *<latest>* | 0..1 | See clause 9.6.27 | *None* |
| *ol* | *<oldest>* | 0..1 | See clause 9.6.28 | *None* |
| *[variable]* | *<timeSeries>* | 0..n | See clause 9.6.36 | *<timeSeries>,**<timeSeriesAnnc>* |
| *[variable]* | *<transaction>* | 0..n | See clause 9.6.48 | *<transaction>* |
| *[variable]* | *<action>* | 0..n | See clause 9.6.61 | *None* |

The *<flexContainer>* resource shall contain the attributes specified in table 9.6.35-2.

Table 9.6.35-2: Attributes of <*flexContainer*> resource

| **Attributes of *<flexContainer>*** | **Multiplicity** | **RW/****RO/****WO** | **Description** | ***<flexContainerAnnc>* 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* | 0..1 (note) | RW | See clause 9.6.1.3. | MA |
| *accessControlPolicyIDs* | 0..1 (L) | RW | See clause 9.6.1.3. | MA |
| *labels* | 0..1 (L) | RW | See clause 9.6.1.3. | MA |
| *creationTime* | 0..1(note) | RO | See clause 9.6.1.3. | NA |
| *lastModifiedTime* | 0..1(note) | RO | See clause 9.6.1.3. | NA |
| *stateTag* | 1 | RO | See clause 9.6.1.3.This *stateTag* attribute value shall be incremented when a custom attribute of the flexContainer is modified. | NA |
| *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 |
| *creator* | 0..1 | RO | See clause 9.6.1.3. | NA |
| *holder* | 0..1 | RW | See clause 9.6.1.3. | NA |
| *location* | 0..1 | RW | See clause 9.6.1.3. | OA |
| *maxNrOfInstances* | 0..1 | RW | Maximum number of direct child *<flexContainerInstance>* resources in the *<flexContainer>* resource. | OA |
| *maxByteSize*  | 0..1 | RW | Maximum size in bytes of custom attributes that is allocated for the *<flexContainer>* resource for all direct child *<flexContainerInstance>* resources. | OA |
| *maxInstanceAge* | 0..1 | RW | Maximum age of a direct child *<flexContainerInstance>* resources in the *<flexContainer>* resource. The value is expressed in seconds. | OA |
| *currentNrOfInstances* | 0..1 | RO |  Current number of direct child *<flexContainerInstance>* resources in the *<flexContainer>* resource. It is limited by the maxNrOfInstances. The currentNrOfInstances attribute of the *<flexContainer>* resource shall be updated on successful creation or deletion of direct child *<flexContainerInstance>* resource of *<flexContainer>* resource. | OA |
| *currentByteSize*  | 0..1 | RO | Current size in bytes of custom attributes stored in all direct child *<flexContainerInstance>* resources of the *<flexContainer>* resource. It is limited by the maxByteSize. The currentByteSize attribute of the *<flexContainer>* resource shall be updated on successful creation or deletion of a direct child *<flexContainerInstance>* resource of *<flexContainer>* resource. | OA |
| *resourceMappingRules* | 0..1 | RW | See clause 9.6.1.3 | OA |
| *containerDefinition* | 1 | WO | This contains an identifier reference (URI) to the <*flexContainer*> schema definition which shall be used by the CSE to validate the syntax of the <*flexContainer*> resource.This URI may refer to one of the oneM2M <*flexContainer*> definitions specified in the following documents:* Generic Interworking [6]]
* AllJoyn Interworking [7];
* Home Domain Information Model [8]

A list of oneM2M <*flexContainer*> definitions is also provided in clause 9.6.1.2.2 [3].Other URI for other *<flexContainer>* definitions may be specified. | MA |
| *ontologyRef* | 0..1 | RW | A reference (URI) of the ontology used to represent the information that is stored in the present *<flexContainer>* resource. | OA |
| *contentSize* | 1 | RO | Sum of the size in bytes of all of the custom attributes. | NA |
| *nodeLink* | 0..1 | RW | The resource identifier of a <node> resource that stores the node specific information of the NoDN on which the interworked service represented by this <flexContainer> resource resides. | OA |
| *[customAttribute]* | 0..n | RW | Specialization-specific attribute(s). Name and data type defined in each specialization of <*flexContainer>* resource. | OA |
| NOTE: When an instance of <*flexContainer*> is a child of a <*flexContainer*> resource, these attributes can be optional. Their presence is determined by the respective definition referred to by the *containerDefinition* attribute. |

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

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

### 9.6.59 Resource Type *flexContainerInstance*

The *<flexContainerInstance>* resource represents a data instance in the *<flexContainer>* resource. The *<flexContainerInstance>* resource shall be created by the Hosting CSE when

* the parent *<flexContainer>* is created, or one or more of the parent’s custom attributes are modified, or the parent is modified by an UPDATE request that contains no attribute at all, and
* at least one of the *maxNrOfInstances, maxByteSize* or *maxInstanceAge* attributes is present.

An update of any of the universal, common, or resource specific attributes of the parent <*flexContainer*>, except custom attributes, does not cause the creation of a new <*flexContainerInstance*> resource.

The <*flexContainerInstane*> shall remain unchanged once created. An AE can delete a *<flexContainerInstance>* resource explicitly or it may be deleted by the Hosting CSE following the retention policy defined by the *maxNrOfInstances, maxByteSize* and *maxInstanceAge* attributes. The *<flexContainerInstance>* resource inherits the same access control policies of the parent *<flexContainer>* resource, and does not have its own *accessControlPolicyIDs* attribute.

The <*flexContainerInstance*> resource shall contain the attributes specified in table 9.6.59-1.

Table 9.6.59-1: Attributes of <*flexContainerInstance*> resource

| Attributes of *<flexContainerInstance>* | Multiplicity | RW/RO/WO | Description |
| --- | --- | --- | --- |
| *resourceType* | 1 | RO | See clause 9.6.1.3. |
| *resourceID* | 1 | RO | See clause 9.6.1.3. |
| *resourceName* | 1 | RO | See clause 9.6.1.3. |
| *parentID* | 1 | RO | See clause 9.6.1.3. |
| *labels* | 0..1 (L) | RO | See clause 9.6.1.3. |
| *creationTime* | 1 | RO | See clause 9.6.1.3. |
| *expirationTime* | 1 | RO | See clause 9.6.1.3. |
| *contentSize* | 1 | RO | Sum of the size in bytes of all of the custom attributes. |
| *originator* | 1 | RO | This attribute is configured with the identifier of the entity that originated the request that caused the creation of this <*flexContainerInstance*> resource, i.e. the *originator* of the CREATE or UPDATE request of the parent <*flexContainer*> resource, which resulted in this creation by the Hosting CSE. |
| *[customAttribute]* | 0..n | RO | Specialization-specific attribute(s). Name and data type defined in each specialization of parent <*flexContainer>* resource. These custom attributes are copies of all of the parent <*flexContainer*> resource custom attributes when the <*flexContainerInstance*> resource is created. |

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

### \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Start of change 3 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#### 10.2.4.18 Update <*flexContainer*>

This procedure shall be used for updating the attributes and the actual data of a *<flexContainer>* resource.

Table 10.2.4.18-1: <*flexContainer*> UPDATE

|  |
| --- |
| ***<flexContainer>* UPDATE** |
| Information in Request message | All parameters defined in table 8.1.2-2 apply with the specific details for:***Content*:** attributes of the <*flexContainer*> resource as defined in clause 9.6.6 which need be updated |
| Processing at Originator before sending Request | According to clause 10.1.4 |
| Processing at Receiver | According to clause 10.1.4A child *<flexContainerInstance>* resource shall be created by the Hosting CSE when at least one of the *maxNrOfInstances, maxByteSize* or *maxInstanceAge* attributes is present and the request contains either no attribute at all or at least one custom attribute of the <*flexContainer*>. All the custom attributes from the <flexContainer>, if any, are copied to the new <*flexContainerInstance*>.If at least one of *maxNrOfInstances, maxByteSize* or *maxInstanceAge* is created, modified or deleted in the *<flexContainer>* resource update request, then the set of <*flexContainerInstances*> children resources, the *currentNrOfInstances* and the *currentByteSize* attributes shall be updated accordingly. |
| 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 |

### \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* End of change 3 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

### \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Start of change 4 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#### 10.2.4.30 Create <*flexContainerInstance*>

Creation of *<flexContainerInstance>* resource is only performed by the Hosting CSE.

The Hosting CSE shall create a *<flexContainerInstance>* resource in case the parent *<flexContainer>* is created or updated with either no attribute at all or with at least one custom attribute, and when at least one of the *maxNrOfInstances, maxByteSize* or *maxInstanceAge* attributes is present with a positive and non-zero value.

When a child *<flexContainerInstance>* resource is created by the Hosting CSE, the Hosting CSE shall

* Copy values of all custom attributes of parent *<flexContainer>* resource, if any, to the created child *<flexContainerInstance>* resource;
* Generate values of all non-custom attributes of *<flexContainerInstance>* resource (e.g. *resourceType, resourceID, resourceName, parentID, labels, creationTime, expirationTime, contentSize, originator).* The *resourceName* attribute of the *<flexContainerInstance>* resource shall be assigned by the Hosting CSE. It can for instance be configured with the *resourceName* of the parent <*flexContainer*> resource appended with the *stateTag* value of the parent <*flexContainer*> resource separated by an underscore “\_” (the value of the *stateTag* is taken *after* it has been incremented e.g. “LightBulb\_3” for a <*flexContainer*> with *resourceName* “LightBulb” and *stateTag*=3). The *originator* attribute shall be configured with the identifier of the originator of the CREATE or UPDATE request of the parent <*flexContainer*> resource that caused the creation of this *<flexContainerInstance>* resource by the Hosting CSE.
* Adjust accordingly the attributes of parent *<flexContainer>* resource related to the instance creation (*currentNrOfInstances* and *currentByteSize).*

If the newly created *<flexContainerInstance>* resource violates any of the defined *maxNrOfInstances, maxByteSize* or *maxInstanceAge* attributes, then the oldest *<flexContainerInstance>* resource(s) shall be removed by the Hosting CSE to enable the creation of this new *<flexContainerInstance>* resource, and the *currentNrOfInstances* and *currentByteSize* attributes shall be re-calculated.

### \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* End of change 4 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*