|  |
| --- |
|  |

|  |
| --- |
| 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-0004 4.3.0 |
| Clauses \* | Clauses 7.4.37.2.3 |
| 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; in this case, the creation of a <flexContainerInstance> will timestamp the triggering of this action.
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-2021-0002R02-TS-0001\_flexContainerInstance\_on\_empty\_UPDATE.

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

##### 7.4.37.2.3 Update

***Originator:***

No change from the generic procedures in clause 7.2.2.1.

***Receiver:***

Primitive specific operation on Recv-1.0 with the following additional operations.

1. The hosting CSE shall validate the received resource representation against the schema value present in the resource *containerDefinition* attribute. If the received resource is not valid then the Hosting CSE shall return a response primitive with a ***Response Status Code*** indicating "BAD\_REQUEST" error.

Primitive specific operation on Recv-6.5 "Create/Update/Retrieve/Delete/Notify operation is performed" with the following additional operations:

1. If at least one of the *maxNrOfInstances, maxByteSize* or *maxInstanceAge is* present in the request with zero value, all children <flexContainerInstances> resources shall be deleted and the *currentNrOfInstances, currentByteSize, maxNrOfInstances, maxByteSize* and *maxInstanceAge* attributes shall be deleted.

In the following steps, any occurrence of *maxNrOfInstances, maxByteSize* or *maxInstanceAge* will refer to their non-zero value in the updated <flexContainer> resource.

1. If a *custom* attribute of the targeted <flexContainer> resource is modified in the request, and *maxByteSize* attribute is present, the Hosting CSE shall check whether the size in bytes of the *custom* attributes of the targeted <flexContainer> resource is greater than the *maxByteSize* of attribute.
2. If true, the Hosting CSE shall return the response primitive with a ***Response Status Code*** indicating "NOT\_ACCEPTABLE" error and skip step 3 below.
3. If false, the Hosting CSE shall update the *contentSize* attribute to the sum of the size in bytes of all of the custom attributes.

3) If the request’s Content parameter contains either no attribute at all or at least one *custom* attribute of the targeted <flexContainer> resource, and at least one of *maxNrOfInstances, maxByteSize* or *maxInstanceAge* attributes is present in the targeted <flexContainer>, then *after* updating the attributes of the targeted <flexContainer> resource:

a) The Hosting CSE shall create a <flexContainerInstance> resource child of the targeted <flexContainer> resource. The attribute *contentSize* of the created <flexContainerInstance> resource will have the same value as in the targeted <flexContainer> resource. The custom attributes of the targeted <flexContainer> resource, if any, shall be copied into the created <flexContainerInstance> resource. The *originator* attribute of the created <flexContainerInstance> resource shall be set to the identifier of the originator of this UPDATE request.

b) The Hosting CSE shall update the *currentNrOfInstances* of the targeted <flexContainer > resource with the number of <flexContainerInstance> resources in the targeted <flexContainer > resource. The Hosting CSE shall update the *currentByteSize* of the targeted <flexContainer > resource with the sum of the *contentSize* attributes of the <flexContainerInstance> resources in the targeted <flexContainer > resource.

c) If *maxNrOfInstances* of the targeted <flexContainer > resource is present then if the *currentNrOfInstances* when modified to reflect the addition of the new *<*flexContainerInstance*>* exceeds *maxNrOfInstances*, the Hosting CSE shall remove the oldest *<*flexContainerInstance*>* resource from the targeted <flexContainer*>* resource. The attributes *currentNrOfInstances* and *currentByteSize* will be updated accordingly.

d) If *maxByteSize* of the targeted <flexContainer > resource is present then if the *currentByteSize* when modified to reflect the addition of the new *<*flexContainerInstance*>* exceeds *maxByteSize* the Hosting CSE shall remove the oldest *<*flexContainerInstance*>* resource(s) from the targeted <flexContainer*>* resource until *maxByteSize* conditions are met. The attributes *currentNrOfInstances* and *currentByteSize* will be updated accordingly.

e) When removing the oldest *<*flexContainerInstance*>* resources, the Hosting CSE shall not generate notifications even if there exists a <subscription> to the targeted <flexContainer> resource and this <subscription> is configured to generate a notification on "Delete\_of\_Direct\_Child\_Resource".

f) If the *maxInstanceAge* attribute is present in the targeted <flexContainer> resource, then the Hosting CSE shall set the *expirationTime* attribute in created <flexContainerInstance> child resource such that the time difference between *expirationTime* and the *creationTime* of the <flexContainerInstance>. The <flexContainerInstance> child resource shall not exceed the *maxInstanceAge* of the targeted <flexContainer> resource.

No other changes from the generic procedures in clause 7.2.2.2.

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