|  |
| --- |
|  |

|  |  |
| --- | --- |
| CHANGE REQUEST | |
| Meeting ID:\* | SDS 57 |
| Source:\* | Miguel Angel Reina Ortega, ETSI, [MiguelAngel.ReinaOrtega@etsi.org](mailto:MiguelAngel.ReinaOrtega@etsi.org)  Andreas Kraft, DT, [A.Kraft@telekom.de](mailto:A.Kraft@telekom.de) |
| Date:\* | 2022-11-24 |
| Reason for Change/s:\* | notificationContentType attribute |
| CR against: Release\* | Rel-4 |
| CR against: WI\* | Active < WI-0077>  MNT maintenance / < Work Item number(optional)>  Is this a mirror CR? Yes  No  mirror CR number:  STE Small Technical Enhancements / < Work Item number (optional)>  Only ONE of the above shall be ticked |
| CR against: TS/TR\* | TS-0004 v4.12.0 |
| Clauses \* |  |
| 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 | None |
| 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 2019 (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.

In case of a correction, and the change apply to previous releases, a separate “mirror CR” should be posted at the same time of 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. Includes any changes to references, definitions, and acronyms in the same deliverable.

Follow the drafting rules.

All pictures must be editable.

Check spelling and grammar to the extent practicable.

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 new clause is proposed to be 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 content of a CR, then the accepted version should not show changes over changes. The accepted version of the CR should only show changes relative to the baseline approved text.

## Introduction

This CR is to allow notificationContentType attribute to be present during subscription resource creation when notificationEventType is set to 7 (Blocking\_Update).

## ----------------------- Start of Change 1--------------------------------------------

##### 7.4.8.2.1 Create

***Originator:***

No change from the generic procedures in clause 7.2.2.1.

If the Originator specifies a *missingData* condition with a duration value greater than the *periodicInterval* attribute of the <timeSeries> resource no notification on missing data points will be generated.

***Receiver:***

The following are additional Hosting CSE procedures to the generic resource handling procedures (Figure 7.2.2.2‑1 in clause 7.2.2.2). The additional procedures shall be inserted from Recv-6.2 to Recv-6.5 as below.

Recv-6.3 The following step is in addition to the procedures defined in clause 7.3.3.15:

Check if the Originator has privileges for retrieving the subscribed-to resource. If the Originator does not have the privilege, the Hosting CSE shall return the response primitive with a ***Response Status Code*** indicating an "ORIGINATOR\_HAS\_NO\_PRIVILEGE" error.

Recv-6.4 The following steps are in addition to the procedures defined in clause 7.3.3.3:

1. Check if the subscribed-to resource, addressed in the ***To*** parameter in the Request, is subscribable. Subscribable resource types are defined in TS-0001 [6]; they have <subscription> resource types as their child resources. If it is not subscribable, the Hosting CSE shall return the Notify response primitive with a ***Response Status Code*** indicating a "TARGET\_NOT\_SUBSCRIBABLE" error instead of the ***Response Status Code*** "INVALID\_CHILD\_RESOURCE\_TYPE".
2. Check if the *notificationEventType* is set to "Blocking\_Update".

* If the subscribed-to resource already has a subscription with this *notificationEventType* the Hosting CSE shall return the response primitive with an ***Response Status Code*** indicating a "BLOCKING\_SUBSCRIPTION\_ALREADY\_EXISTS" error if more than one notification of this type could be sent.
* If there is more than one *notificationURI* specified, the Hosting CSE shall return the response primitive with a ***Response Status Code*** indicating a "BAD\_REQUEST" error.
* If any resource-specific attributes of the <subscription> resource other than *eventNotificationCriteria*, *notificationContentType* or *notificationURI* are specified the Hosting CSE shall return the primitive with a ***Response Status Code*** indicating a "BAD\_REQUEST" error.
* If any condition tag of the *eventNotificationCriteria* attribute other than *attribute* condition tag is specified, the Hosting CSE shall return the response primitive with a ***Response Status Code*** indicating a "BAD\_REQUEST" error.

1. Check if the *notificationEventType* is set to “Report on missing data points”. If the *missingData* attribute is not provided as well, the request shall be rejected with a ***Response Status Code*** indicating a "BAD\_REQUEST" error***.***
2. Check if the *missingData* element of eventNotificationCriteria is provided.

* If the subscribed-to resource (i.e. the resource given by the ***To*** parameter in the Request) is not a <timeSeries>, the request shall be rejected with a ***Response Status Code*** indicating a "BAD\_REQUEST" error.

1. If any of the *notificationURI* entries are not the Originator, the Hosting CSE may send a Subscription Verification request primitive to each of them as described in clause 7.5.1.2.3.

a) If the Hosting CSE cannot send one or more Subscription Verification request primitives, the Hosting CSE shall return the Create <subscription> response primitive with a ***Response Status Code*** indicating a "SUBSCRIPTION\_VERIFICATION\_INITIATION\_FAILED" error.

b) If the Hosting CSE sent all the Subscription Verification request primitives, the Hosting CSE shall check if each Notify response primitive contains a ***Response Status Code*** indicating "OK". If not, the Hosting CSE shall return the Create <subscription> response primitive containing the ***Response Status Code*** indicating a "SUBSCRIPTION\_VERIFICATION\_INITIATION\_FAILED" error.

1. If the Originator provides a value of *childResourceType* which is not a valid child of the subscribed-to resource, the request shall be rejected with a ***Response Status Code*** indicating a “BAD\_REQUEST” error***.***
2. If the Originator provides *missingData*, check that subscribed-to resource is of type <timeSeries>. If not, the request shall be rejected with a ***Response Status Code*** indicating a "BAD\_REQUEST" error*.*
3. If both the *notificationEventType* and *operationMonitor* are present in the Request, the request shall be rejected with a ***Response Status Code*** indicating a "BAD\_REQUEST" error***.***
4. If the *notificationContentType* is invalid for a given operation (refer to oneM2M TS-0001 [6] Table 9.6.8-4: Default and allowed values of *notificationContentType*) the request shall be rejected with a ***Response Status Code*** indicating a "BAD\_REQUEST" error***.***

Recv-6.5: The following steps are in addition to the procedures defined in clause 7.3.3.5:

1. If the Originator does not provide *notificationContentType*, the Hosting CSE shall set it according to the default shown in oneM2M TS-0001 [6] Table 9.6.8-4: Default and allowed values of *notificationContentType.*
2. If the *notificationURI* is not the Originator, the Hosting CSE shall set the Originator’s ID as the <subscription> resource's *creator* attribute.
3. If the *batchNotify* attribute is present in the Request but *batchNotify*/*duration* is not provided by the Originator, the Hosting CSE shall set the value of *batchNotify*/*duration* to the default duration as given by the M2M Service Provider.
4. If the *notificationStatsEnable* attribute is set to true, the Hosting CSE shall start recording notification statistics in the *notificationStatsInfo* attribute once the <subscription> resource is created.

-------------------------------------------------- End of Change 1---------------------------------------