|  |
| --- |
|  |

|  |  |
| --- | --- |
| CHANGE REQUEST | |
| Meeting ID:\* | SDS 40.3 |
| Source:\* | Bob Flynn, Convida Wireless , Bob.Flynn@convidawireless.com |
| Date:\* | 2019-05-15 |
| Reason for Change/s:\* | Enforcement of subscription with “missingData” under <timeSeries> during create of subscription only |
| CR against: Release\* | Rel-2 |
| CR against: WI\* | Active <Work Item number>  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\* | TS-0001 v2.23.0 |
| Clauses \* | 9.6.8 |
| 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 contribution addresses

* Subscription under <timeSeries>

As per section 10.2.4.29, specification mentions that if *missingData* field is present in the *eventNotificationCriteria* attribute of *<subscription>* resource during subscription create operation under *<timeSeries>* resource, then CSE shall start the timer on first missing data detection and report the same to the subscriber with a NOTIFY request if the conditions to meet reporting criteria are met.

As per the specification TS-0004 v3.10.1 Table 7.4.8.1‑3, *eventNotificationCriteria* attribute of *<subscription>* resource is O (optional) during update operation of *<subscription>.* This means that *missingData* field can be included in *eventNotificationCriteria* during update operation of *<subscription>* under *<timeSeries>.*

Now, CSE’s implementation can get little complex if the values of *missingData* are modified during update operation of *<subscription>* under *<timeSeries>.* For example, during create of *<subscription>,* if the sub-fileds *number* and *duration* of missingData are configured to certain values, then corresponding to this, CSE starts timer for *duration* and monitors the number of missing data based on *number*. Later, if any or all of the sub-fields namely *number* and *duration* of *missingData* are modified during subscription update operation, the implementation on the CSE can get complex to handle this scenario since CSE would have started timer for the old *duration* and would have been calculating the number of missed data based on *number*.

To keep it simple, contribution is to make the field *missingData* of *eventNotificationCriteria* attributeof <*subscription*> resource NP during update of subscription.

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

Table 9.6.8-3: *eventNotificationCriteria* conditions

| Condition tag | Multiplicity | Matching condition |
| --- | --- | --- |
| *createdBefore* | 0..1 | The *creationTime* attribute of the resource is chronologically before the specified value. |
| *createdAfter* | 0..1 | The *creationTime* attribute of the resource is chronologically after the specified value. |
| *modifiedSince* | 0..1 | The *lastModifiedTime* attribute of the resource is chronologically after the specified value. |
| *unmodifiedSince* | 0..1 | The *lastModifiedTime* attribute of the resource is chronologically before the specified value. |
| *stateTagSmaller* | 0..1 | The *stateTag* attribute of the resource is smaller than the specified value. |
| *stateTagBigger* | 0..1 | The *stateTag* attribute of the resource is bigger than the specified value. |
| *expireBefore* | 0..1 | The *expirationTime* attribute of the resource is chronologically before the specified value. |
| *expireAfter* | 0..1 | The *expirationTime* attribute of the resource is chronologically after the specified value. |
| *sizeAbove* | 0..1 | The *contentSize* attribute of the *<contentInstance>* resource is equal to or greater than the specified value. |
| *sizeBelow* | 0..1 | The *contentSize* attribute of the *<contentInstance>* resource is smaller than the specified value. |
| *notificationEventType* | 0..5 | The type of event. Possible notification event type values are:  a) Update to attributes of the subscribed-to resource.  b) Deletion of the subscribed-to resource.  c) Creation of a direct child of the subscribed-to resource.  d) Deletion of a direct child of the subscribed-to resource.  e) An attempt to retrieve a <contentInstance> direct-child-resource of a subscribed-to <container> resource is performed while this <contentInstance> child resource is an obsolete resource or the reference used for retrieving this resource is not assigned. This retrieval is performed by a RETRIEVE request targeting the subscribed-to resource with the *Result Content* parameter set to either "child-resources" or "attributes+child-resources".  The other conditions in *eventNotificationCriteria conditions apply to the selected notificationEventType.*  For example, if notificationEventType is "Creation of a direct child of the subscribed-to resource" then other *eventNotificationCriteria* conditions is applied to the direct child resources of the subscribed-to resource.  If this condition is not specified, the default value is "Update to attributes of the subscribed-to resource".  This default value shall be apply only if operationMonitor is not present in the resource.  The notion of "obsolete resource" is defined in clause 9.6.1.3.2 (Common attributes). |
| *operationMonitor* | 0..5 | The operations accessing the subscribed-to resource matches with the specified value. It allows monitoring which operation is attempted to the subscribed-to resource regardless of whether the operation is performed. This feature is useful when to find malicious AEs. Possible string arguments are: create, retrieve, update, delete. |
| *attribute* | 0..n | A list of attribute names of a subscribed-to-resource. This list is only applicable when *notificationEventType* has a value of "Update to attributes of the subscribed-to resource".  If this list is present, then it is used to specify a subset of a subscribed-to-resource's attributes for which updates shall result in a notification. If ANY attribute specified on this list is updated, then a notification shall be generated. If an attribute that is not specified in this list is updated, then a notification shall not be generated.  If this list is not presented, then the default attribute list is the full set of a subscribed-to-resource's attributes. If ANY attribute of a subscribed-to-resource is updated, then a notification shall be generated. |
| *missingData* | 0..1 | The *missingData* includes two values: a minimum specified missing number of the Time Series Data within the specified window duration, and the window duration. The condition only applies to subscribed-to resources of type *<timeSeries>*. This attribute may only be specified in CREATE operation.  The first detected missing data point starts the timer associated with the window duration.  The window duration is restarted upon its expiry until such time as the entire subscription is terminated or not refreshed. More details about NOTIFICATIONS related to data reporting is found in clause 10.2.39. |
| *filterOperation* | 0..1 | Indicates the logical operation (AND/OR) to be used for different condition tags. The default value is logical AND. |

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