|  |  |
| --- | --- |
| Input Contribution | |
| Meeting ID\* | SDS 42 |
| Title:\* | Semantic reasoning normative texts for TS-0034 |
| Source:\* | Xu Li, Convida, [li.xu@convidawireless.com](mailto:li.xu@convidawireless.com)  Chonggang Wang, Convida, [wang.chonggang@convidawireless.com](mailto:wang.chonggang@convidawireless.com) |
| Date:\* | 2019-09-14 |
| Input related to\* | TS-0034 v 4.0.0 |
| Intended purpose of  document:\* | Decision  Discussion  Information  Other <specify> |
| Impacted other TS/TR(s) | n/a |
| Decision requested or recommendation:\* | The content is to be included in clauses 6 and 7.11 of TS-0034 v 4.0.0 |
| Template Version: January 2017 (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.

# Introduction

On TP 41, SDS-2019-0369 was discussed regarding the next step for WI-0053 (semantic enablement) and the consensus reached is that the given the maturity of technical contents in TR-0033, it is time for moving the normative contents from TR-0033 to the related TS (i.e., TS-0034 and TS-0001).

Note that, the current content organization for semantic related topics is as follows:

* The detailed descriptions for each of semantic feature as well as corresponding resource CRUD operations are mainly described in TS-0034
* The resource definitions and the brief introductions for each of semantic features are described in TS-0001 (clause 9.6 for resource definition and clauses 10.2.4 for briefly introducing semantic features)

According to the above organization, this contribution includes the following changes towards TS-0034 (another companion contribution SDS-2019-0460 is towards TS-0001):

* The detailed descriptions of CRUD operations for the three semantic reasoning related resources, i.e., <*reasoningRules*>, <*ruleRepository*>, and <*reasoningJobInstance*> (See Change #1).
* A new clause 7.11 for the semantic reasoning feature (See Change #2).

Note that, all the changes introduced in this contribution are directly from TR-0033 without any new technical-related additions/deletions/modifications, except for necessary editorial/format changes.

### -----------------------Start of change 1-------------------------------------------

### 6.X <*ruleRepository*> Operations

#### 6.X.1 Introduction

A *<ruleRepository>* resource is a child resource of the *<CSEBase>* resource. The *<ruleRepository>* resource may have one or multiple <*reasoningRules*> child resources to represent different sets of reasoning rules in the oneM2M system. A reasoning initiator can create <*reasoningJobInstance*> child resources of a *<ruleRepository>* resource to initiate desired reasoning operations.

#### 6.X.2 Create *<ruleRepository>*

This procedure is used for creating a <*ruleRepository*> resource as described in Table 6.X.2-1.

Table 6.X.2-1: <*ruleRepository*> CREATE

| ***<ruleRepository>* CREATE** | |
| --- | --- |
| Associated Reference Point | Mca, Mcc and Mcc' |
| Information in Request message | All parameters defined in TS-0001 [1] table 8.1.2-3 apply with the specific details for:  ***Content:*** The resource content provides the information as defined in the resource definition of <*ruleRepository*> resource. |
| Processing at Originator before sending Request | According to clause 10.1.2 in oneM2M TS-0001 in [1]. |
| Processing at Receiver | According to clause 10.1.2 in oneM2M TS-0001 in [1]. |
| Information in Response message | All parameters defined in table 8.1.3-1 in [1] apply with the specific details for:  **Content**: Address of the created <*ruleRepository*> resource, according to clause 10.1.2 in [1]. |
| Processing at Originator after receiving Response | According to clause 10.1.2 in oneM2M TS-0001 in [1]. |
| Exceptions | According to clause 10.1.2 in oneM2M TS-0001 in [1]. |

#### 6.X.3 Retrieve *<ruleRepository>*

This procedure is used for retrieving the attributes of a <*ruleRepository*> resource as described in Table 6.X.3-1.

Table 6.X.3-1: <*ruleRepository*> RETRIEVE

|  |  |
| --- | --- |
| ***<ruleRepository> RETRIEVE*** | |
| Associated Reference Point | Mca, Mcc and Mcc'. |
| Information in Request message | All parameters defined in Table 8.1.2-3 in [1] apply. |
| Processing at Originator before sending Request | According to clause 10.1.3 in oneM2M TS-0001 in [1]. |
| Processing at Receiver | According to clause 10.1.3 in oneM2M TS-0001 in [1]. |
| Information in Response message | All parameters defined in Table 8.1.3-1 in [1] apply with the specific details for:  **Content:** Attributes of the <*ruleRepository*> resource. |
| Processing at Originator after receiving Response | According to clause 10.1.3 in oneM2M TS-0001 in [1]. |
| Exceptions | According to clause 10.1.3 in oneM2M TS-0001 in [1]. |

#### 6.X.4 Update *<ruleRepository>*

This procedure is used for updating the attributes of a <*ruleRepository*> resource as described in Table 6.X.4-1.

Table 6.X.4-1: <*ruleRepository*> UPDATE

|  |  |
| --- | --- |
| ***<ruleRepository> UPDATE*** | |
| Associated Reference Point | Mca, Mcc and Mcc' |
| Information in Request message | All parameters defined in Table 8.1.2-3 in [1] apply with the specific details for:  **Content**: Attributes of the <*ruleRepository*> resource to be updated. |
| Processing at Originator before sending Request | According to clause 10.1.4 in oneM2M TS-0001 in [1]. |
| Processing at Receiver | According to clause 10.1.4 in oneM2M TS-0001 in [1]. |
| Information in Response message | According to clause 10.1.4 in oneM2M TS-0001 in [1]. |
| Processing at Originator after receiving Response | According to clause 10.1.4 in oneM2M TS-0001 in [1]. |
| Exceptions | According to clause 10.1.4 in oneM2M TS-0001 in [1]. |

#### 6.X.5 Delete *<ruleRepository>*

This procedure is used for deleting a <*ruleRepository*> resource as described in Table 6.X.5-1.

Table 6.X.5-1: <*ruleRepository*> DELETE

| ***<ruleRepository> DELETE*** | |
| --- | --- |
| Associated Reference Point | Mca, Mcc and Mcc' |
| Information in Request message | All parameters defined in table 8.1.2-3 in [1] apply. |
| Processing at Originator before sending Request | According to clause 10.1.5 in oneM2M TS-0001 in [1]. |
| Processing at Receiver | According to clause 10.1.5 in oneM2M TS-0001 in [1]. |
| Information in Response message | According to clause 10.1.5 in oneM2M TS-0001 in [1]. |
| Processing at Originator after receiving Response | According to clause 10.1.5 in oneM2M TS-0001 in [1]. |
| Exceptions | According to clause 10.1.5 in oneM2M TS-0001 in [1]. |

### 6.Y <*reasoningRules*> Operations

#### 6.Y.1 Introduction

A <*reasoningRules*> resource can be used to store a set of related reasoning rules (e.g. for supporting a particular application). *<reasoningRules>* resource is a child resource of the *<ruleRepository>* resource. By performing the CRUD operations on the <*reasoningRules*> resources, various reasoning rules (e.g., user-defined reasoning rules based on business logic) can be created, discovered, retrieved, updated and deleted inside the oneM2M system.

#### 6.Y.2 Create *<reasoningRules>*

This procedure is used for creating a <*reasoningRules*> resource as described in Table 6.Y.2-1.

Table 6.Y.2-1: <*reasoningRules*> CREATE

| ***<reasoningRules>* CREATE** | |
| --- | --- |
| Associated Reference Point | Mca, Mcc and Mcc' |
| Information in Request message | All parameters defined in Table 8.1.2-3 in TS-0001 [1] apply with the specific details for:  ***Content:*** The resource content provides the information as defined in the resource definition of <*reasoningRules*> resource. |
| Processing at Originator before sending Request | According to clause 10.1.2 in oneM2M TS-0001 in [1]. |
| Processing at Receiver | According to clause 10.1.2 in oneM2M TS-0001 in [1]. |
| Information in Response message | All parameters defined in table 8.1.3-1 in [i.3] apply with the specific details for:  **Content**: Address of the created <*reasoningRules*> resource, according to clause 10.1.2 in [i.3]. |
| Processing at Originator after receiving Response | According to clause 10.1.2 in oneM2M TS-0001 in [1]. |
| Exceptions | According to clause 10.1.2 in oneM2M TS-0001 in [1]. |

#### 6.Y.3 Retrieve *<reasoningRules>*

This procedure is used for retrieving the attributes of a <*reasoningRules*> resource as described in Table 6.Y.3-1.

Table 6.Y.3-1: <*reasoningRules*> RETRIEVE

|  |  |
| --- | --- |
| ***<reasoningRules> RETRIEVE*** | |
| Associated Reference Point | Mca, Mcc and Mcc'. |
| Information in Request message | All parameters defined in table 8.1.2-3 in [1] apply. |
| Processing at Originator before sending Request | According to clause 10.1.3 in oneM2M TS-0001 in [1]. |
| Processing at Receiver | According to clause 10.1.3 in oneM2M TS-0001 in [1]. |
| Information in Response message | All parameters defined in Table 8.1.3-1 in [1] apply with the specific details for:  **Content:** Attributes of the <*reasoningRules*> resource. |
| Processing at Originator after receiving Response | According to clause 10.1.3 in oneM2M TS-0001 in [1]. |
| Exceptions | According to clause 10.1.3 in oneM2M TS-0001 in [1]. |

#### 6.Y.4 Update *<reasoningRules>*

This procedure is used for updating the attributes of a <*reasoningRules*> resource as described in Table 6.Y.4-1.

Table 6.Y.4-1: <*reasoningRules*> UPDATE

|  |  |
| --- | --- |
| ***<reasoningRules> UPDATE*** | |
| Associated Reference Point | Mca, Mcc and Mcc' |
| Information in Request message | All parameters defined in Table 8.1.2-3 in [1] apply with the specific details for:  **Content**: Attributes of the <*reasoningRules*> resource to be updated. |
| Processing at Originator before sending Request | According to clause 10.1.4 in oneM2M TS-0001 in [1]. |
| Processing at Receiver | According to clause 10.1.4 in oneM2M TS-0001 in [1]. |
| Information in Response message | According to clause 10.1.4 in oneM2M TS-0001 in [1]. |
| Processing at Originator after receiving Response | According to clause 10.1.4 in oneM2M TS-0001 in [1]. |
| Exceptions | According to clause 10.1.4 in oneM2M TS-0001 in [1]. |

#### 6.Y.5 Delete *<reasoningRules>*

This procedure is used for deleting a <*reasoningRules*> resource as described in table 6.Y.5-1.

Table 6.Y.5-1: <*reasoningRules*> DELETE

| ***<reasoningRules> DELETE*** | |
| --- | --- |
| Associated Reference Point | Mca, Mcc and Mcc' |
| Information in Request message | All parameters defined in Table 8.1.2-3 in [1] apply. |
| Processing at Originator before sending Request | According to clause 10.1.5 in oneM2M TS-0001 in [1]. |
| Processing at Receiver | According to clause 10.1.5 in oneM2M TS-0001 in [1]. |
| Information in Response message | According to clause 10.1.5 in oneM2M TS-0001 in [1]. |
| Processing at Originator after receiving Response | According to clause 10.1.5 in oneM2M TS-0001 in [1]. |
| Exceptions | According to clause 10.1.5 in oneM2M TS-0001 in [1]. |

### 6.Z <*reasoningJobInstance*> Operations

#### 6.Z.1 Introduction

A Reasoning Initiator (RI), such as an AE or CSE, may trigger two types of reasoning operations. One type is “one-time reasoning operation. This is applicable to the case where a reasoning operation can be executed over a Fact Set (FS) and a Rule Set (RS) that may not change over time. In comparison, the other type is a “continuous” reasoning operation. The second type is applicable to the cases where the input FS and RS for reasoning may change over time, and accordingly the previously inferred knowledge may not be valid anymore. Therefore, new reasoning is executed over the latest version of FS and RS in order to generate up-to-date inferred knowledge.

A <*reasoningJobInstance*> resource represents a specific reasoning job instance for enabling the two types of reasoning operations. A RI initiates a desired reasoning operation by creating a <*reasoningJobInstance*> resource as a child resource of a *<ruleRepository>* resource.

#### 6.Z.2 Create *<reasoningJobInstance>*

This procedure is used for creating a <*reasoningJobInstance*> resource as described in Table 6.Z.2-1.

Table 6.Z.2-1: <*reasoningJobInstance*> CREATE

| ***<reasoningJobInstance>* CREATE** | |
| --- | --- |
| Associated Reference Point | Mca, Mcc and Mcc' |
| Information in Request message | All parameters defined in [1] table 8.1.2-3 apply with the specific details for:  ***Content:*** The resource content provides the information as defined in the resource definition of <*reasoningJobInstance*> resource. |
| Processing at Originator before sending Request | According to clause 10.1.2 in oneM2M TS-0001 in [1]. |
| Processing at Receiver | The Receiver follows the basic procedure according to clause 10.1.4 in oneM2M TS-0001 [1], with the following specific details:   1. The receiver first retrieves the facts from the resources referred to by the *factSet* attribute. For example,  * If a referred resource is a type of <*semanticDescriptor*> resource, the RDF triples included in the *descriptor* attribute will be collected. * If a referred resource is a type of <*ontology*> resource, the data included in the *ontologyContent* attribute will be collected.  1. The receiver retrieves all the related reasoning rules for the resources referred to by the *ruleSet* attribute. For example,  * If a referred resource is a <*reasoningRules*> resource, the rules included in the *ruleRepresentation* attribute will be collected.  1. The receiver includes the retrieved facts and rules from the previous steps, as well as optional facts/rules based on local policies, as inputs for the semantic reasoning operation. The receiver performs semantic reasoning processing using these inputs and produces the reasoning result and stores it in the *result* attribute of the created <*reasoningJobInstance*> resource. 2. If the created <*reasoningJobInstance*> resource represents a continuous reasoning operation (i.e., the *reasoningType* attribute is set to “continuous”), subsequent reasoning processing will be automatically triggered and performed according to the values of *mode* and *period* attributes and the *result* attribute will be overwritten with the latest reasoning result. |
| Information in Response message | All parameters defined in table 8.1.3-1 in [1] apply with the specific details for:  ***Content***: Address of the created <*reasoningJobInstance*> resource, according to clause 10.1.2 in [1]. |
| Processing at Originator after receiving Response | According to clause 10.1.2 in oneM2M TS-0001 in [1]. |
| Exceptions | According to clause 10.1.2 in oneM2M TS-0001 in [1]. |

#### 6.Z.3 Retrieve *<reasoningJobInstance>*

This procedure is used for retrieving the attributes of a <*reasoningJobInstance*> resource as described in Table 6.Z.3-1.

Table 6.Z.3 -1: <*reasoningJobInstance*> RETRIEVE

|  |  |
| --- | --- |
| ***<reasoningJobInstance> RETRIEVE*** | |
| Associated Reference Point | Mca, Mcc and Mcc'. |
| Information in Request message | All parameters defined in table 8.1.2-3 in [1] apply. |
| Processing at Originator before sending Request | According to clause 10.1.3 in oneM2M TS-0001 in [1]. |
| Processing at Receiver | According to clause 10.1.3 in oneM2M TS-0001 in [1]. |
| Information in Response message | All parameters defined in table 8.1.3-1 in [i.3] apply with the specific details for:  ***Content*:** Attributes of the <*reasoningJobInstance*> resource. |
| Processing at Originator after receiving Response | According to clause 10.1.3 in oneM2M TS-0001 in [1]. |
| Exceptions | According to clause 10.1.3 in oneM2M TS-0001 in [1]. |

#### 6.Z.4 Update *<reasoningJobInstance>*

This procedure is used for updating the attributes of a <*reasoningJobInstance*> resource as described in Table 6.Z.4-1.

Table 6.Z.4-1: <*reasoningJobInstance*> UPDATE

|  |  |
| --- | --- |
| ***<reasoningJobInstance> UPDATE*** | |
| Associated Reference Point | Mca, Mcc and Mcc' |
| Information in Request message | All parameters defined in table 8.1.2-3 in [1] apply with the specific details for:  ***Content***: Attributes of the <*reasoningJobInstance*> to be updated. |
| Processing at Originator before sending Request | According to clause 10.1.4 in oneM2M TS-0001 in [1]. |
| Processing at Receiver | According to clause 10.1.4 in oneM2M TS-0001 in [1]. |
| Information in Response message | According to clause 10.1.4 in oneM2M TS-0001 in [1]. |
| Processing at Originator after receiving Response | According to clause 10.1.4 in oneM2M TS-0001 in [1]. |
| Exceptions | According to clause 10.1.4 in oneM2M TS-0001 in [1]. |

#### 6.Z.5 Delete *<reasoningJobInstance>*

This procedure is used for deleting a <*reasoningJobInstance*> resource as described in Table 6.Z.5-1.

Table 6.Z.5-1: <*reasoningJobInstance*> DELETE

| ***<reasoningJobInstance> DELETE*** | |
| --- | --- |
| Associated Reference Point | Mca, Mcc and Mcc' |
| Information in Request message | All parameters defined in table 8.1.2-3 in [1] apply. |
| Processing at Originator before sending Request | According to clause 10.1.5 in oneM2M TS-0001 in [1]. |
| Processing at Receiver | According to clause 10.1.5 in oneM2M TS-0001 in [1]. |
| Information in Response message | According to clause 10.1.5 in oneM2M TS-0001 in [1]. |
| Processing at Originator after receiving Response | According to clause 10.1.5 in oneM2M TS-0001 in [1]. |
| Exceptions | According to clause 10.1.5 in oneM2M TS-0001 in [1]. |

### -----------------------End of change 1-------------------------------------------

### -----------------------Start of change 2-------------------------------------------

## Semantics Reasoning

~~The procedures are not fully defined in this release.~~

Semantic reasoning is a mechanism to derive implicit facts that are not explicitly expressed in the existing knowledge/facts (such as RDF triples) by leveraging a set of reasoning rules. A Semantic Reasoning Function (SRF) is defined in this clause in order to support semantic reasoning function in oneM2M system. The key features of SRF is shown in Figure 7.11-1:



Figure 7.11-1: Key Features of Semantic Reasoning Function (SRF)

**Feature-1: Enabling semantic reasoning related data**

The major functionality of Feature-1 is to enable the semantic reasoning related data (referring to facts and reasoning rules) by making those data be discoverable, publishable/sharable across different entities in oneM2M system (which is illustrated as the dark yellow arrow in the Figure 7.11-1). The semantic reasoning related data can be a Fact Set (FS) and/or a Rule Set (RS):

* A FS refers to a set of facts and each RDF triple in fact describes a piece of fact, so a set of RDF triples stored in a <semanticDescriptor> resource can be regarded as an FS. In general, a FS can be used as an input for a semantic reasoning process (i.e. an input FS) or it can be a set of inferred facts as the result of a semantic reasoning process (i.e. an inferred FS).
* A RS refers to a set of semantic reasoning rules. For example, oneM2M applications may define their own reasoning rules (user-defined reasoning rules) for different application needs.

Overall, Feature-1 involves with enabling the publishing/discovering/sharing semantic reasoning related data (including both FSs and RSs) through appropriate oneM2M resources. The general flow of Feature-1 is that oneM2M users (as originator) can send requests to certain receiver CSEs in order to publish/discover/update/delete the FS/RS-related resources through the corresponding CRUD operations. Once the processing is done, the receiver CSE will send the response back to the originator.

**Feature-2: Optimizing other semantic operations with background semantic reasoning support**

The existing semantic operations supported in oneM2M system (e.g., semantic resource discovery and semantic query) may not yield desired results without semantic reasoning support. The major functionality of Feature-2 of SRF is to leverage semantic reasoning as a “background support” to optimize other semantic operations (which are illustrated by the pink arrows in the Figure 7.11-1). In this case, users trigger/initiate specific semantic operations (e.g., a semantic query). During the processing of this operation, semantic reasoning may be further triggered in the background, which is however fully transparent to the user.

Overall, the general flow of Feature-2 is that oneM2M users (as originator) can send requests to certain receiver CSEs for the desired semantic operations (such as semantic resource discovery, semantic query, etc.). During the request processing, the receiver CSE, assuming it supports SRF, can further leverage reasoning capability. In general, the reasoning capability of SRF is realized by an underlying semantic reasoner software. By leveraging the outputs of the semantic reasoning (i.e., reasoning result), the receiver CSE will further produce the optimal result for the semantic operation as requested by the originator (e.g., the semantic query result, or semantic discovery result) and then send the response back to the originator.

**Feature-3: Enabling individual semantic reasoning process**

oneM2M users may also directly interact with SRF by triggering individual semantic reasoning process, which is the Feature-3 of SRF. When using this feature, oneM2M user shall first identify the interested facts (as input FS) as well as the desired reasoning rules based on their application needs. When the input FS and RS are identified, the oneM2M user shall send a request to SRF for triggering a specific semantic reasoning process by specifying the inputs (i.e. the input FS and RS). The SRF will then initiate a desired semantic reasoning process. Once the SRF works out the semantic reasoning result, it will be returned to the oneM2M users for further usage.

### -----------------------End of change 2-------------------------------------------