|  |
| --- |
| Input contributionUse case |
| Use Case Title:\* | Semantic discovery in presence of a “network” of M2M Service Providers (M2MSP) |
| Group Name:\* | RDM |
| Source:\* | INRIA (on behalf of ETSI STF 589) |
| Contact: | INRIALuigi Liquori, Luigi.Liquori@inria.fr Marie-Agnès Peraldi-Frati, marie-agnes.peraldi\_frati@inria.fr  |
| Date:\* | 2020-04-23 |
| Abstract:\* | This use case could be considered as either the “use-case zero”, or a “parametric use-case” for Advanced Semantic Discovery because it is suitable to be **instantiated in many concrete cases**. It shows the importance of fixing, formalizing and extending:a) *Formal Graph Topologies* to capture most common scenarios involving networks of M2MSP;b) *Formal Semantic Discovery Routing Mechanism* (SDRM) to route a Semantic Query between M2MSP with exhaustivity/non exhaustivity constraints and iterative vs. recursive routing modality;c) *Formal Semantic Discovery Query Language* (SDQL) to express a large type of queries;d) *Formal Semantic Resolution Query Mechanism* (SRQM) necessary to reduce locally a complex query into a number of simpler queries. |
| Agenda Item:\* | [WI-0101](http://member.onem2m.org/Application/documentapp/downloadLatestRevision/?docId=31941) - Advanced Semantic Discovery |
| Work item(s): | See proposed WI in TP-2020-0040 |
| Document(s) Impacted\* | TR 001 |
| Intended purpose ofdocument:\* | [x]  Decision[ ]  Discussion[ ]  Information[ ]  Other <specify> |
| Decision requested or recommendation:\* | Include in TR 001 |
| '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.

## Advanced Semantic Discovery in presence of as \*network\* of M2M Service Providers

### Description

This use case is the “use-case zero”, a “parametric use-case” is suitable to be instantiated in many concrete cases of Advanced Semantic Discovery. It shows the importance of fixing, formalizing and extending:

a) *Formal Graph Topologies* to capture most common scenarios involving networks of M2MSP;

b) *Formal Semantic Discovery Routing Mechanism* (SDRM) to route a Semantic Query between M2MSP with exhaustivity/non exhaustivity constraints and iterative vs. recursive routing modality;

c) *Formal Semantic Discovery Query Language* (SDQL) to express a large type of queries;

d) *Formal Semantic Resolution Query Mechanism* (SRQM) necessary to reduce locally a complex query into a number of simpler queries.

### Source

ETSI TR 103 714: “SmartM2M; Study for oneM2M Discovery and Query use cases and requirements”.

###  Actors

1. 5 *Application Entities* (AE) X of type T1, Y of type T2, Z of type T3, V of type T4, and W of type T5.
2. 2 *Middle Node Common Service Entities* (MN-CSE) P, and Q.

A MN-CSE has a local database containing information on their registered AE. The local database includes location information (where each device is currently located), the device type, etc. Let P and Q have some “Semantic Discovery Agreement” (SDA) with A. SDA can be relaxed inside a single Service Provider, see Note 2 in Subsection 12.1.4.

1. 4 *Infrastructure Node Common Service Entities* (IN-CSE) A, B, C, and D.

A IN-CSE has a local database containing information on their registered MN-CSE and AE. The local database includes location information (where each device is currently located), the device type, etc. Let A,B,C, and D have some “Semantic Discovery Agreement” (SDA) each other’s.

### Kinds of SDA

In an hypothetical analogy with the Border Gateway Protocol 4 (BGP4, <https://tools.ietf.org/html/rfc4271>), 3 kinds of SDA between MN-CSE and IN-CSE are possible, namely:

SDA IN {CUSTOMER, PROVIDER, PEER}.

The three SDA can intuitively be explained as follows:

1. CSE1 CUSTOMER of CSE2: CSE1 takes advantage of the infrastructure, MN-CSE, and AE registered in CSE2, and also shares security policies of CSE2.
2. CSE2 PROVIDER of CSE1: conversely, CSE2 offer usage of its infrastructure, and registered MN-CSE, and AE to CSE1. CSE1 again shares security policies of CSE2.
3. CSE1 PEER CSE2: CSE1 and CSE2 mutually shares infrastructure, MN-CSE, and AE and common security policies.

**Note 1.** CUSTOMER and PROVIDER SDAs are *asymmetric relations*, while PEER SDA is a *symmetric* *relation* one.

**Note 2.** Inside a single Service Provider SDA is not required (i.e. SDA=PEER).

### Pre-conditions

Consider the following topology:



### Triggers

Let AND be a nonterminal of a *Semantic Discovery Query Language* (SDQL). Let X:T1 send a *Semantic Discovery Query Request* (SDREQ) to MN-CSE P as follows:

 SDREQ1 = ?T2|FC2 AND ?T3|FC3 AND ?T4|FC4 AND ?T5|FC5

The query can be intuitively read as looking for

some AE of type T2 registered in any CSE satisfying the filter criteria FC2, AND

some AE of type T3 registered in any CSE satisfying the filter criteria FC3, AND

some AE of type T4 registered in any CSE satisfying the filter criteria FC4, AND

some AE of type T5 registered in any CSE satisfying the filter criteria FC5

Let OR be a nonterminal. Consider queries such as

SDREQ1’ = ?T2|FC2 OR ?T3|FC3 OR ?T4|FC4 OR ?T5|FC5

The query can be intuitively read as looking for

some AE of type T2 registered in any CSE satisfying the filter criteria FC2, OR

some AE of type T3 registered in any CSE satisfying the filter criteria FC3, OR

some AE of type T4 registered in any CSE satisfying the filter criteria FC4, OR

some AE of type T5 registered in any CSE satisfying the filter criteria FC5

By combining AND and OR, it is possible to consider semantic queries such as:

SDREQ1’’ = (?T2|FC2 OR ?T3|FC3) AND (?T4|FC4 OR ?T5|FC5)

SDREQ1’’’ = (?T2|FC2 AND ?T3|FC3) OR (?T4|FC4 AND ?T5|FC5)

By adding the NOT nonterminal, it is possible to consider queries such as:

SDREQ1’’’’ = (?T2|FC2 AND ?T3|FC3) OR (?T4|FC4 AND (NOT ?T5|FC5))

It is also possible to consider other restricted semantic routing directives such as (list not exhaustive):

ANY = search in all CSE databases

CURRENT = search in the CSE local database

CUSTOMER[N] = search in the databases of N CUSTOMER CSE

PROVIDER[N] = search in the databases of N PROVIDER CSE

PEER[N] = search start on the databases of N PEER CSE

The shape of a logical query could be a restricted kind of either a *Conjunctive Normal Form* (CNF) or a *Disjunctive Normal Form* (DNF). A CNF/DNF will be translated into a set of simpler unitary Queries by the CSE receiving the query using the *Semantic Resolution Query Mechanism* (SRQM) to be defined in Task 2 of STF 589. Complexity of SRQM can be an issue and should be discussed as part of Task 2 of STF 589.

### Normal Flow

A possible “trace” of the semantic discovery routing mechanism, inspired to [1,2] proceeds as follows (the real routing mechanism will be described and formally specified in Task 2 of STF 589 and simulated in Task 3 of STF 589).

* X sends a semantic discovery request (SDREQ1) to P via an mca-pointer
* P verifies the integrity of SDREQ1 and forward a semantic discovery request SDREQ1 to A via an mcc-pointer that starts the semantic routing into the network of CSE
* SDREQ1 is resolved using SRQM locally in A into four subqueries, namely SDREQ2, SDREQ3, SDREQ4, and SDREQ5, where:

SDREQ2 = ?T2|FC2 AND

SDREQ3 = ?T3|FC3 AND

SDREQ4 = ?T4|FC4 AND

SDREQ5 = ?T5|FC5

1. A starts lookups in its local database, trying to solve {SDREQ2,3,4,5} but fail
2. A down-forwards SQREQ1 to Q via an mcc pointer
3. Q solve the subquery SQREQ2 ?T2|FC2 in its local database returning Y to A
4. A send back Y to P and X
5. A up-forwards SQREQ3 & SQREQ4 & SQREQ5 to B via an mcc’ pointer
6. B solve the SQREQ3 ?T3|FC3 in its local database returning Z to A (and back to P and X)
7. B side-forwards SQREQ4 & SQREQ5 to C via an mcc’ pointer
8. C solve the SQREQ4 ?T4|FC2 in its local database returning V to B (and back to A, P and X)
9. C down-forwards SQREQ5 to D via an mcc’ pointer
10. D solve the SQREQ5 ?T5|FC5 in its local database returning W to C (and back to B, A, P and X)

**Note 1**. The presented semantic routing trace is “recursive” i.e. A waits for B, which waits for C, which waits for D. It builds an “iterative” routing flow where X communicate directly with the all other actors.

**Note 2.** This use case also discriminate two kinds of routing, namely:

1. “Exhaustive”. As example, in the case a semantic resource that exists for sure, the system will explore all the distributed data bases until it will found it.
2. “Non-exhaustive”. As example, even in the case a semantic resource that exists for sure, the system will explore the data bases spaces until it will reach a fixed number N of hops.

**Note 3.** When A up-forwards to B, it follows that A respect the customer-provider SDA with B (e.g. A respect the SDA directives of B). When B side-forwards to C, it follows that B and C respect the peer-peer SDA common agreement. When C down-forwards to D, it follows that D respect the provider-customer SDA with C (e.g. D respect the SDA directives of C).

**The moral is**: B and C should be “acknowledged’’ for their “routing job”.

### Alternative flow

In the following alternative topology the customer-provider SDA are reversed:



A possible “trace” of the semantic discovery routing mechanism, again inspired to [1,2,3,4] proceeds as in Subsection 12.1.6, excepting for the following caveat.

**Caveat**. When A down-forwards to B, it follows that B should respect the provider-customer SDA with A (e.g. B *should acknowledge* A. This is not intuitive since *B do a favour to A and moreover have also acknowledge A*). When B side-forwards to C, it follows that B and C have a common SDA agreement and, as such, they not acknowledge it each other. When C up-forwards to D, it follows that C and D have a common SDA agreement (e.g. C *should acknowledge* D). This is not intuitive since *C do a favour to D and moreover have also acknowledge D).*

**The moral is**: B and C does a job for their providers and, moreover, they have to *acknowledge* for their “routing job”.

Alternative traces happen in practice. Because of the distributed nature of the Semantic Discovery Routing, it is beneficial to try to incentivize routing respecting the SDA, and, as such, avoid routing not respecting the SDA. Those situations are not new in Internet and are referred as VALLEY ROUTING by Gao [1]. “Good routing” should guarantee that routing is always “valley preserving” (or “no valley”). Valley routing property is also preserved in the network aware Resource Discovery Protocol of Liquori *et al*. [2].

### Post-conditions

X can start to interact with Y, Z, V, and W.

### High Level Illustrations



### Potential requirements

The following requirements extend the oneM2M requirements TS-0002-V4.6.0.

1. The M2M system SHALL provide a
	1. Additional feature in the current SEMANTIC DISCOVERY QUERY LANGUAGE (SDQL) to manage the discovery among different SP and within a single SP, and to support complex queries pointing to set of resources distribute in a single system as well in multiple oneM2M systems
	2. A mechanism able to rewrite a complex query into a set of unitary ones, as detailed in Subsection 12.1.6, and collect the results. (SEMANTIC DISCOVER RESOLUTION QUERY MECHANISM - SDQM)
2. The M2M system SHALL provide mechanism Semantic Discovery Agreements between CSE.
3. The M2M system SHALL provide a SEMANTIC DISCOVERY ROUTING MECHANISM (SDRM) that can be RECURSIVE or ITERATIVE and EXAUSTIVE or NON-EXAUSTIVE) to support flexible scope of the discovery and queries originated by the AEs e.g. when the targets of the query are not known at the application (e.g. searching for the values of an unknown distributed set of containers).
4. The M2M system SHALL provide security and access control mechanisms to support the distributed nature of Semantic Network Discovery.

References

[1] Lixin Gao. On inferring autonomous system relationships in the internet. IEEE/ACM Trans. Netw. 9(6): 733-745 (2001) <https://dl.acm.org/doi/10.1109/90.974527>).

[2] Luigi Liquori, Rossano Gaeta, and Matteo Sereno. A Network Aware Resource Discovery Service. EPEW 2019 - 16th European Performance Engineering Workshop, Nov 2019, Milano, Italy, Volume 12039 of Lecture Notes in Computer Science, Springer Verlag, pages 84-99, 2019.

[3] oneM2M. TR 0001 Use\_Cases\_Collection-V4\_3\_0. Ex 12.9. Semantics query for device discovery across M2M Service Providers.

[4] oneM2M. REQ-2014-0005R01-Semantics\_query\_for\_device\_discovery\_on\_Inter-M2M\_SP.