|  |  |
| --- | --- |
| Input contribution  Use 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: | INRIA  Luigi Liquori, [Luigi.Liquori@inria.fr](mailto:Luigi.Liquori@inria.fr)  Marie-Agnès Peraldi-Frati, [marie-agnes.peraldi\_frati@inria.fr](mailto: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 (ASD) that can be **instantiated in many specific cases**. It aims at showing the importance of formalizing:   * *Advanced Semantic Discovery Query Language* (ASDQL) able to express *Advanced Semantic Discovery Queries* (ASDQ); * *Semantic Discovery Routing Protocol* (SDRP) to route an Advanced Semantic Discovery Query (ASDQ) between different CSEs; * *Semantic Discovery Agreement* (SDA), to state some communication agreements between CSE; * *Semantic Query Resolution System* (SQRS) allowing to locally resolving an Advanced Semantic Discovery Query (ASDQ) into some elementary standard oneM2M Semantic Discovery Queries (SDQ). |
| 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 of  document:\* | 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 - a network of nodes across IoT Domains

### Description

This use case could be considered as either the “use-case zero”, or a “parametric use-case” for Advanced Semantic Discovery (ASD) and it can be instantiated in many domain specific cases.

This use case illustrates the needs for an Advanced Semantic Discovery (ASD) within distributed network of CSEs belonging a single Service Provider and across different IoT Service Providers. This distributed scenario is partially faced in the current document in Clause 12.9 (Semantics query for device discovery across M2M Service Providers).

It shows the importance of formalizing:

* *an Advanced Semantic Discovery Query Language* (ASDQL) able to write Advanced Semantic Discovery Query (ASDQ);
* *a Semantic Discovery Routing Protocol* (SDRP) to route an *Advanced Semantic Discovery Query* (ASDQ) between different CSEs;
* *a Semantic Discovery Agreement* (SDA), to state some communication agreements between CSEs;
* *a Semantic Query Resolution functionality*(SQR) allowing to locally resolve an Advanced Semantic Discovery Query (ASDQ) into some elementary standard oneM2M Semantic Discovery Queries (SDQ).

The concepts included in the current use clause

is intensively used in the following clauses of the current document, namely:

*12.21 Advanced\_Semantic\_Discovery - Semantic\_Recommendation in a network of nodes across IoT Domains*

*12.22 Advanced\_Semantic\_Discovery -Facility\_Management\_of\_a\_Supermarket\_Chain,* and

*12.23 Advanced\_Semantic\_Discovery -Healthcare\_Network\_and\_Clinical\_Knowledge\_Administration.*

T and **:**

**Advanced Semantic Discovery (ASD)**

We define Advanced Semantic Discovery (ASD) allowing discovery across a network of CSEs statically connected among each other’s in a *tree-like* topology inside a single Service Provider (SP) and/or Administrative Domain (AD) and in a *mesh-like* topology between different SPs. The root of each tree is represented by an IN-CSE, instead the mesh contains only IN-CSE nodes.

**Semantic Discovery Agreement (SDA)**

By Semantic Discovery Agreement (SDA) we aim at adding a “direction” to the edges of the tree (DAG) and the edges of the mesh-graphs formed by the MN-CSEs and the IN-CSEs. With an analogy with the Border Gateway Protocol 4 (BGP4, <https://tools.ietf.org/html/rfc4271>), we set 2 kinds of directions:

* CSE1 to CSE2 meaning that CSE1 takes advantage of the infrastructure, MN-CSEs, and AEs registered in CSE2, and also shares security policies of CSE2. We say that CSE1 is a *CUSTOMER* and CSE2 is a *PROVIDER.*
* CSE1 to CSE2 means that CSE1 and CSE2 mutually share infrastructure, MN-CSEs, and AEs and common security policies. We say that CSE1 and CSE2 are *PEERS.*

CUSTOMER and PROVIDER are roles that conform an *asymmetric relationship*, while PEER is a role conforming a *symmetric* *relationship*.

**Note 2.** Inside a single Service Provider, the SDA is not mandatory since it can be considered as PEER.

**Semantic Routing Table (SRT)**

A Semantic Routing Table (SRT) contained in each CSE annotates all semantic routes and SDA relations: ex. SRT\_CSE1: [PEER:CSE2,3,4 ; PROVIDER:CSE5,6 ; CUSTOMER: CSE7,8,9].

**Advanced Semantic Discovery Query Language (ASDQL)**

The Advanced Semantic Discovery Query Language (ASDQL) is an extension of the actual oneM2M Semantic Discovery Query Language (SDQL), which has to be suitable enough to describe queries that will be resolved in a *cooperative* way by a distributed network of CSEs. Each CSE involved in the resolution participates in resolving subqueries and aggregating results by coordinating and cooperating among each other’s.

**Advanced Semantic Discovery Query (ASDQ)**

According to the Theory of Formal Languages, we define Advanced Semantic Discovery Query (ASDQ) as a word in the Advanced Semantic Discovery Query Language (ASDQL).

**Semantic Query Resolution (SQR)**

Each CSE contains a Semantic Query Resolution capability (SQR) that takes as input an Advanced Semantic Discovery Query (ASDQ) and

1. produces as output a *normalized* Advanced Semantic Discovery Query (ASDQ) equivalent to *Conjunctive Normal Form* (CNF) or a *Disjunctive Normal Form* (DNF);

**Note**. It is well-known that complexity of this transformation could be an issue;

1. produces a set of ordinary oneM2M Semantic Discovery Query (SDQ) from the normalized Advanced Semantic Discovery Query (ASDQ) one.

**Semantic Discovery Routing (SDR)**

The CSEs support a distributed Semantic Discovery Routing (SDR) that listens for Advanced Semantic Discovery Query (ASDQ) and:

1. It extracts the Advanced Semantic Discovery Query (ASDQ);
2. It reduces the Advanced Semantic Discovery Query (ASDQ) into a set of #m Semantic Discovery Queries (SDQ) by means of the Semantic Query Resolution System (SQRS);
3. It solve as much Semantic Discovery Queries (SDQ) as it cans, using resources registered in the CURRENT CSE, say #n;
4. It forwards the remaining #m-n Semantic Discovery Queries (SDQ) to α-CSE Customer (we say in "downstream"), taken from the CURRENT Semantic Routing Table (SRT), resolving say #p;
5. It forwards the remaining #(m-n-p) Semantic Discovery Queries (SDQ) to β-CSE PEER (we say in "sidestream"), taken from the CURRENT Semantic Routing Table (SRT), resolving say #q;
6. It forwards the remaining #(m-n-p-q) Semantic Discovery Queries (SDQ) to γ-CSE PROVIDERS (we say in "upstream"), taken from the CURRENT Semantic Routing Table (SRT), resolving say #r ≤ #(m-n-p-q);
7. It reconstructs in a reverse order the partial results, sending back to the originator of the Advanced Semantic Discovery Query (ASDQ).

**Note 1**. α, β, and γ are protocol specific parameters that can be locally modified in each CSE;

**Note 2.** if #r=#(m-n-p-q), then we say that the query is *exhaustive,* and this is in contrast with *non-exhaustive* routing*.*Generally, we

1. As example, in the case that a semantic resource exists somewhere in the CSEs network, then the system will explore the entire distributed network until it will found it.
2. As example, even in the case a semantic resource that exists somewhere in the CSEs network, then the system will explore part of the distributed network until it will be stopped.

**Note 3**. A similar approach is described in [i.24].

### 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.

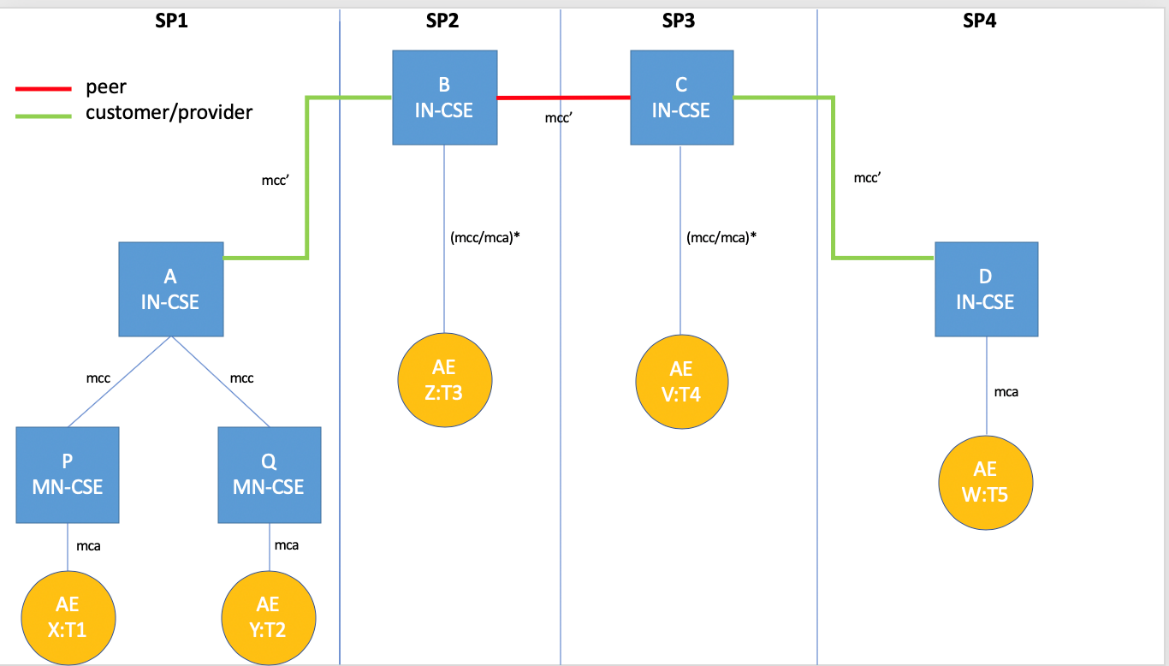
A MN-CSE has a local database containing information on their registered AE. The local database includes location information (where each device is currently located), the device type, etc. Let P and Q have some Semantic Discovery Agreement (SDA) with A. Semantic Discovery Agreement (SDA) can be relaxed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

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

An IN-CSE has a local database containing information on their registered MN-CSEs and AEs. 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) among each other’s.

### Pre-conditions

Consider the following topology:



### Triggers

This section presents, informally, three examples of the Advanced Semantic Discovery Query Language (ASDQL). Let AND, OR, NOT be *non-terminals* and ?T means a *meta-variable* of type T to be resolved.

**Example 1.** X:T1 send to MN-CSE P

ASDQ1 = ?T2|FC2 AND ?T3|FC3 AND ?T4|FC4 AND ?T5|FC5

The query can be intuitively read as follows: X is 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

**Example 2.** X:T1 send to MN-CSE P

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

The query can be intuitively read as follows: X is 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

**Example 3**. X:T1 send to MN-CSE P

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

**Example 4**. X:T1 send to MN-CSE P

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

**Example 5**. X:T1 send to MN-CSE P

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

It is also possible to consider other non-terminals, 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;

BETWEEN\_TIME[SEC] = search should return in SEC;

BETWEEN\_SPACE[METER] = search should give results in METER;

OF\_BRAND[NAME] = search should give results of brand NAME.

### Normal Flow

We present a “trace” of the Semantic Discovery Routing (SDR) generated by Example 1, the other examples can be easily traced following the same logic. This trace is inspired to a semantic discovery routing as described in [i.24] and [i.25] and proceeds as follows:

* X sends an Advanced Semantic Discovery Query (ASDQ1) to P;
* P verifies the integrity of ASDQ1 and forward the Advanced Semantic Discovery Query ASDQ1 to A that starts the Semantic Discovery Routing Protocol (SDPR) into the network of CSE;
* ASDQ1 is resolved using the Semantic Query Resolution System (SQRS) locally in A into four subqueries, namely ASDQ2, ASDQ3, ASDQ4, and ASDQ5, where:

ASDQ2 = ?T2|FC2

ASDQ3 = ?T3|FC3

ASDQ4 = ?T4|FC4

ASDQ5 = ?T5|FC5

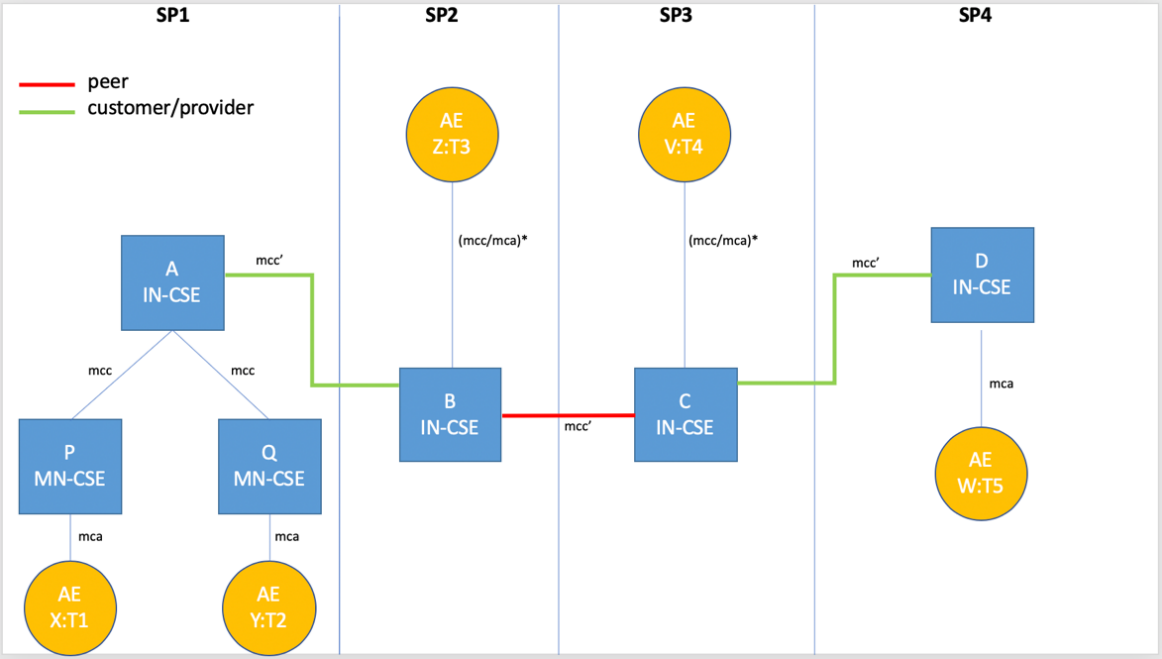
1. A starts lookup in its local database, trying to solve {ASDQ2,3,4,5} but fail
2. A down-forwards ASDQ1 to Q via an mcc pointer
3. Q solve the subquery ASDQ2 ?T2|FC2 in its local database returning Y to A
4. A send back Y to P and X
5. A up-forwards ASDQ3 and ASDQ4 and ASDQ5 to B
6. B solve the ASDQ3 ?T3|FC3 in its local database returning Z to A (and back to P and X)
7. B side-forwards ASDQ4 & ASDQ5 to C
8. C solve the ASDQ4 ?T4|FC2 in its local database returning V to B (and back to A, P and X)
9. C down-forwards ASDQ5 to D
10. D solve the ASDQ5 ?T5|FC5 in its local database returning W to C (and back to B, A, P and X)

**Note 3.** When A up-forwards to B, it follows that A respect the CUSTOMER-PROVIDER SDA with B (e.g. A respect the Semantic Discovery Agreement (SDA) directives of B). When B side-forwards to C, it follows that B and C respect the PEER-PEER Semantic Discovery Agreement (SDA) directives. When C down-forwards to D, it follows that D respect the PROVIDER-CUSTOMER Semantic Discovery Agreement (SDA) with C (e.g. D respect the Semantic Discovery Agreement (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 Semantic Discovery Agreement (SDA) are reversed:



A possible “trace” of the Semantic Discovery Routing Protocol (SDRP), again inspired to [2] and [3] proceeds as in clause 12.20.6, excepting for the following caveat.

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

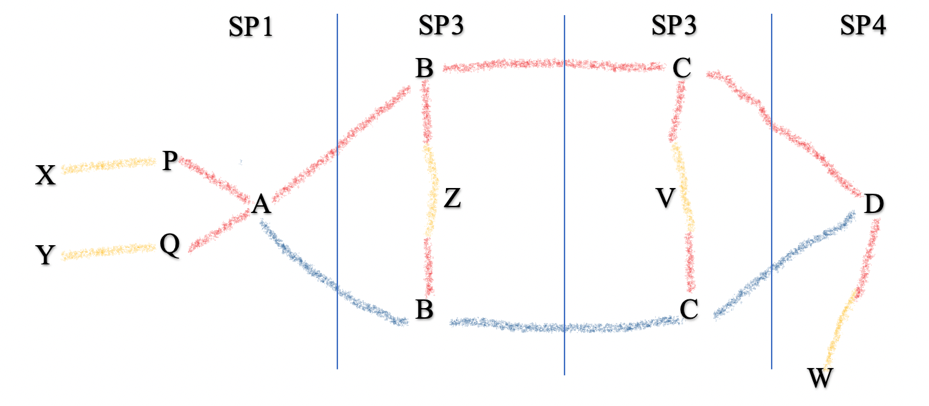
**The moral is**: B and C do a job for their providers and, moreover, they have to *acknowledge* for their “routing job”.

Alternative traces happen in practice. Because of the distributed nature of the Semantic Discovery Routing Protocol (SDRP), it is beneficial to try incentivizing routing respecting the Semantic Discovery Agreement (SDA), and, as such, avoid routing not respecting the Semantic Discovery Agreement (SDA). Those situations are not new in Internet and are referred as VALLEY ROUTING by [i.23]. “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 [i.24].

### Post-conditions

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

### High Level Illustrations



### Potential requirements

The oneM2M system shall provide mechanisms for Advanced Semantic Discovery (ASD) across a distributed network of IoT nodes within a single oneM2M Service Provider and across different IoT Service Providers.

A CSE receiving an Advanced Semantic Discovery Query (ASDQ) shall extract the Semantic Discovery Query (SDQ), embedded in the packet payload, and shall resolve the query with respect to the locally available information and shall forward to other suitable CSEs the Advanced Semantic Discovery Query (ASDQ) to complete the discovery.

More specifically, the oneM2M system shall provide:

1. An Advanced Semantic Discovery Query Language (ASDQL) that the ability to write Advanced Semantic Discovery Query (ASDQ);
2. A Semantic Discovery Agreement (SDA) to state some communication agreements between CSE;
3. A Semantic Query Resolution (SQR) that allows to locally translate an Advanced Semantic Discovery Query (ASDQ) into some elementary oneM2M Semantic Discovery Queries (SDQ);
4. A Semantic Discovery Routing (SDR) to route an Advanced Semantic Discovery Query (ASDQ) between different CSEs.

References (to be included in section 2.2 of TR0001)

[i.23] 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>).

[i.24] 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.

[i.25] Raphael Chand and Michel Cosnard and Luigi Liquori. Powerful resource discovery for Arigatoni overlay network. Future Generation Computing System, volume 23, number 1, pages 31-38, 2008.