Draft ETSI GR MEC 035 V2.0.11 (2020-12)

Multi-access Edge Computing (MEC); Study on Inter-MEC systems and MEC-Cloud systems coordination

Reference
DGR/MEC-0035InterMEC

Keywords
HANDOVER, INTERWORKING

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-préfecture de Grasse (06) N° 7803/88

Important notice

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status.
Information on the current status of this and other ETSI documents is available at
https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Foreword

This Group Report (GR) has been produced by ETSI Industry Specification Group <long ISGname> (<short ISGname>).

Modal verbs terminology

In the present document "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Executive summary

Introduction

1 Scope

The present document studies the applicability of MEC specifications to inter-MEC systems and MEC-Cloud systems coordination that supports e.g., application instance relocation, synchronization, and similar functionalities. Another subject of this study is the enablement and/or enhancement of functionalities for application lifecycle management by third parties (e.g. application developers). Firstly, the study analyses the current specifications. Secondly, the study documents the use cases that require inter-system coordination, including those in multi-MNO environments. Thirdly, the study clarifies the requirements and any missing parts. Finally, the study indicates possible solutions to close the gaps. The document considers the relevant work of other industry bodies relating to inter system coordination and all relevant work done in ETSI.

2 References

2.1 Normative references

Normative references are not applicable in the present document.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI GS MEC 001: “Multi-access Edge Computing (MEC); Terminology”.

[i.2] ETSI GS MEC 003: “Multi-access Edge Computing (MEC); Framework and Reference Architecture”.

[i.3] ETSI GS MEC 030: “Multi-access Edge Computing (MEC); V2X Information Service API”.

[i.4] GSMA White Paper, “Operator Platform Concept – Phase 1: Edge Cloud Computing”, Jan. 2020. Online: https://www.gsma.com/futurenetworks/resources/operator-platform-concept-whitepaper/

[i.5] XW2-200048, Huawei, Intel: “High-level Architectural Considerations on MEC in Multi-MNO Scenarios”, Attachment for LS to GSMA on High-level Architectural Considerations on MEC in Multi-MNO Scenarios (XW2-200047), presented at 5GAA ‘F2F’/Virtual WG Meeting Week #14 (11 – 15 May 2020).

[i.6] ETSI GS MEC 010-2: “Multi-access Edge Computing (MEC); Application lifecycle, rules and requirements management”.

[i.7] ETSI GS MEC 016: “Multi-access Edge Computing (MEC); UE application interface”.

[i.8] ETSI GR MEC 018: End to End Mobility Aspects

[i.9] ETSI GS MEC 021: Application Mobility Service API

[i.10] ETSI GS MEC 002: “Multi-access Edge Computing (MEC); Phase 2: Use Cases and Requirements”.

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the [following] terms [given in ... and the following] apply:

In the following, some terms and definition used in the present document are listed:

- **MEC federation**: a federated model of MEC systems enabling shared usage of MEC services and applications.

3.2 Symbols

For the purposes of the present document, the [following] symbols [given in ... and the following] apply:

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GS MEC 001 [i.1] and the following apply:

GSMA	Global System for Mobile Communications Association
MNO	Mobile Network Operator
V2X	Vehicle-to-everything
OEM	Original Equipment Manufacturer

4 Overview

4.1 Introduction

The present document studies the applicability of MEC specifications to inter-MEC systems and MEC-Cloud systems coordination.

Clause 5 documents the use cases that require inter-system communication, including those in multi-MNO environments, and consequently clarifies the requirement/recommendations. Also, evaluation is provided for each use case to clarify the any missing parts/gaps to be solved/closed.

Editor's note: Some requirement might be commonly introduced by multiple use cases. Therefore, the rapporteur intends to summarize a list of gaps in the end of Clause 6 and treat corresponding solutions in Clause 7.

Clause 6 proposes the possible solutions for closing the gaps. Clause 7 finally concludes this study.

4.2 Inter-MEC system communication

Inter-MEC system communication has been identified by ETSI ISG MEC as an important technical topic, primarily impactful to Mobile Network Operators (MNOs). ETSI MEC GS 003 [i.2] specifies three high-level requirements for inter-MEC system communication, along with a hierarchical framework for inter-MEC system discovery and communication as described by the following excerpt (Clause 9 of [i.2]):

"Inter-MEC system communication addresses the following high-level requirements:

1) A MEC platform should be able to discover other MEC platforms that may belong to different MEC systems;
2) A MEC platform should be able to exchange information in a secure manner with other MEC platforms that may belong to different MEC systems.
3) A MEC application should be able to exchange information in a secure manner with other MEC applications that may belong to different MEC systems.

To enable the inter-MEC system communication, the following hierarchical inter-MEC system discovery and communication framework is assumed:

- MEC system level inter-system discovery and communication.
- MEC host level inter-system communication between the MEC platforms.

NOTE: It is for further study if MEC platforms in different MEC systems should be able to discover each other without the involvement of the MEC system level functional elements."

In parallel, driven by the MNOs' interest to form federated MEC environments, e.g., to achieve V2X service continuity in multi-operator operation scenarios, as per ETSI GS MEC030 [i.3] (see Clauses 5.1-5.3 of the GS), ETSI ISG MEC has introduced the present Work Item (MEC035) on "Study on Inter-MEC systems and MEC-Cloud systems coordination".

At the same time, GSMA has published a White Paper on the "Operator Platform" concept with focus on "Phase 1" of Edge Cloud Computing in January 2020 [i.4]. In this White Paper, GSMA envisages that: "operators will collaborate to offer a unified "operator platform". In Phase 1, the Operator Platform will federate multiple Operators' edge computing infrastructure to give application providers access to a global edge cloud to run innovative, distributed and low latency services through a set of common APIs".

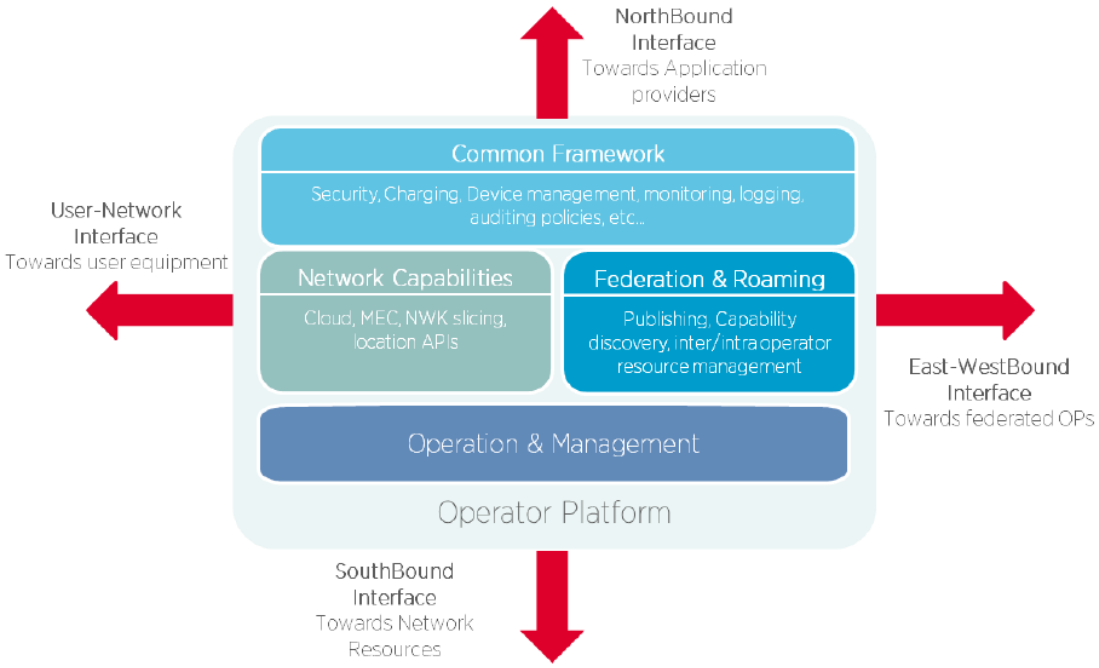


Figure 4.2-1: High level GSMA Operator Platform building blocks (source: [i.4]).

From all the above, it is concluded that inter-MEC system communication is an imperative need in today's as well as future's edge computing industry and ecosystem. However, to unlock the full potential of federated MEC environments (as the exemplary one in Figure 4.2-1), an effective and well-defined signaling framework among MEC system entities, is needed, both at system level and at host level. Such a framework has not yet been proposed so far, and the present document is the appropriate place to discuss this topic.

4.3 MEC-Cloud system communication

MEC-Cloud communication is recognized as another important technical topic. ETSI GS MEC 003 [i.2], has referred to application instance relocation between the MEC system and an external cloud environment, (Appendix A.4.2.2.4 of

[i.2]), which is applicable in the context of MEC applications sensitive to UE mobility. According to that, the application instance relocation is conducted under the supervision of MEO.

> *In some cases, and when it is supported, the UE can request the MEC system to move application instances out of the MEC system to an external cloud environment, or from an external cloud environment to the MEC system. In that case, the application instance relocation is triggered between the MEC system and the external cloud environment under the supervision of the multi-access edge orchestrator.*

Furthermore, OSS is responsible for receiving requests from device applications for relocating applications between external clouds and the MEC system (Clause 7.1.4.2 of [i.2]) and for receiving a request to run applications from the third parties. In the case of relocation between MEC and Cloud systems, it may include a request from the third parties. Virtualization infrastructure manager is expected to interact with external cloud manager to perform the application relocation (Clause 7.1.5.2 of [i.2]). As for the interfaces, [i.2] specifies the reference points connecting to external entities, i.e., Mx1 and Mx2 (Clause 6.1 of [i.2]).

As a summary, MEO supervises the application relocation between external cloud and MEC system. OSS interacts with external cloud system via Mx1 or the combination of Mx2 and Mm8.

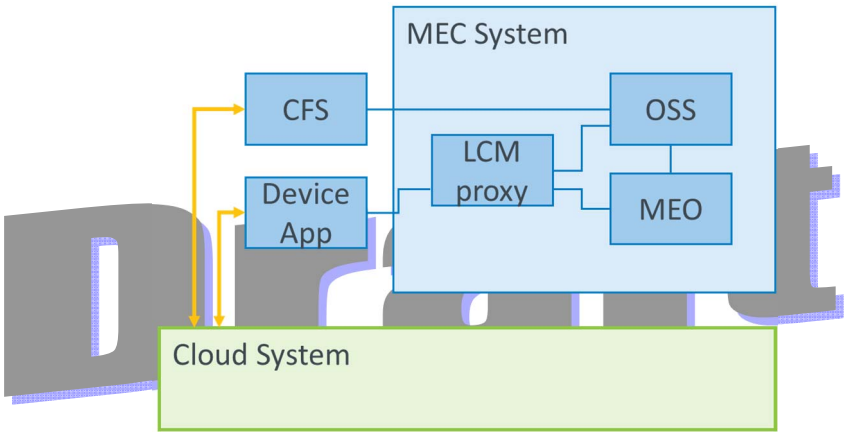


Figure 4.3-1: interactions between a MEC and a Cloud system; the blue-coloured reference points are specified by ETSI GS MEC 003 [i.2]

Application mobility is a unique feature of MEC system, which supports relocation of user context and/or application instance from one MEC host to another, or between a MEC host and a Cloud.

In this document, all the works should align with the current specifications. The further recommendations should be clarified based on the use case. Then, the gap from the current specifications will be clarified as well. Then, solutions will be introduced.

As a matter of fact, there exist many de-facto specifications for cloud systems. Therefore, proposing recommendations for the operation of the cloud system is outside the scope of this document. The intention is to rather clarify the involved reference points and functional entities in the MEC system. Fig. 4.3-2 illustrates the high-level architecture.

Note: Infrastructure level communication is out of scope in the present document.

Editor's notes:

- Business relationship between MEC and Cloud system to be introduced in the GR. That should cover MEC-Centralized Public Cloud, MEC-Public Cloud co-located with MEC, and MEC- Private Cloud (Application provider's environment).
- How to treat major de-facto standards, e.g., GCP, AWS, Azure, etc., is for further study.

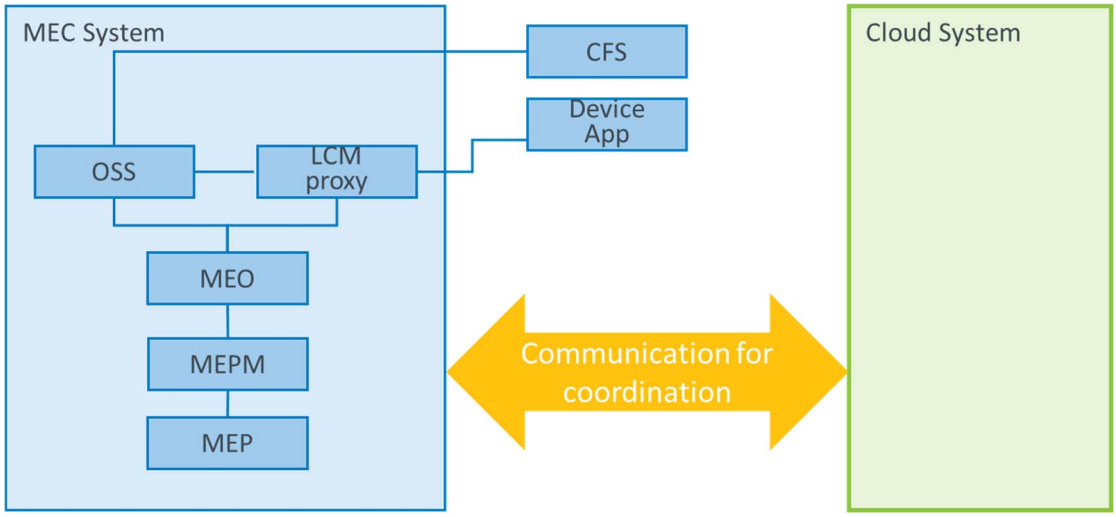


Figure 4.3-2: high-level architecture view of MEC-Cloud system communication; the blue-coloured reference points are specified by ETSI GS MEC 003 [i.2]

4.4 Patterns of Business relationship between MEC and external systems

In this study, the following patterns of business relationship between MEC and external systems are considered as illustrated in Fig. 4.3-3.

1) MEC system and MEC system/Edge Cloud:
As a main case of this category, one MEC system is in MNO 1's network and the other is in MNO 2's network. Both systems are located in the different MNOs' network but those systems are structured with the same functions that are specified in MEC 003. This category includes the following subcases relating Edge Cloud. Here, by "Edge Cloud" we refer to a cloud point-of-presence on the same "operator's premises" as the MNO but which is outside the MNO's control and therefore trust space. For practical purposes the difference may be understood as one of interconnections: the Edge Cloud is connected to the MEC System via a high-performance L2 interconnect over which the MNO can enforce L2-like strict SLAs on throughput, latency, etc.; whereas Private Cloud and Public Cloud do not presume such an interconnect (although it may presume other interconnects with their own SLAs).
   1') MEC system and Edge Cloud in different MNO's network:
   This pattern is also considered as a subcase of 1). Edge Cloud is located inside the MNO's network but the associating MNO is different from that of MEC system. It shares the virtualized infrastructure with the centralized cloud system.

   1'') MEC system and Edge Cloud in the same MNO's network:
   This pattern is considered as a subcase of 1). Edge Cloud is located inside the same MNO's network as MEC system. It shares its virtualized infrastructure with the centralized cloud system. but its resources are distributed in the associating MNO's network.

2) MEC system and Central Cloud system:
Central Cloud system is located out of MNO's network. Architecture of Central Cloud system is out of scope of the present document.
   2') MEC system and Private Cloud system in an application provider's own environment:
   This pattern is considered as a subcase of 2). Private Cloud system is located in the application provider's environment. It can be just an application server or on-premise cloud system. Architecture of Private Cloud system is out of scope of the present document.

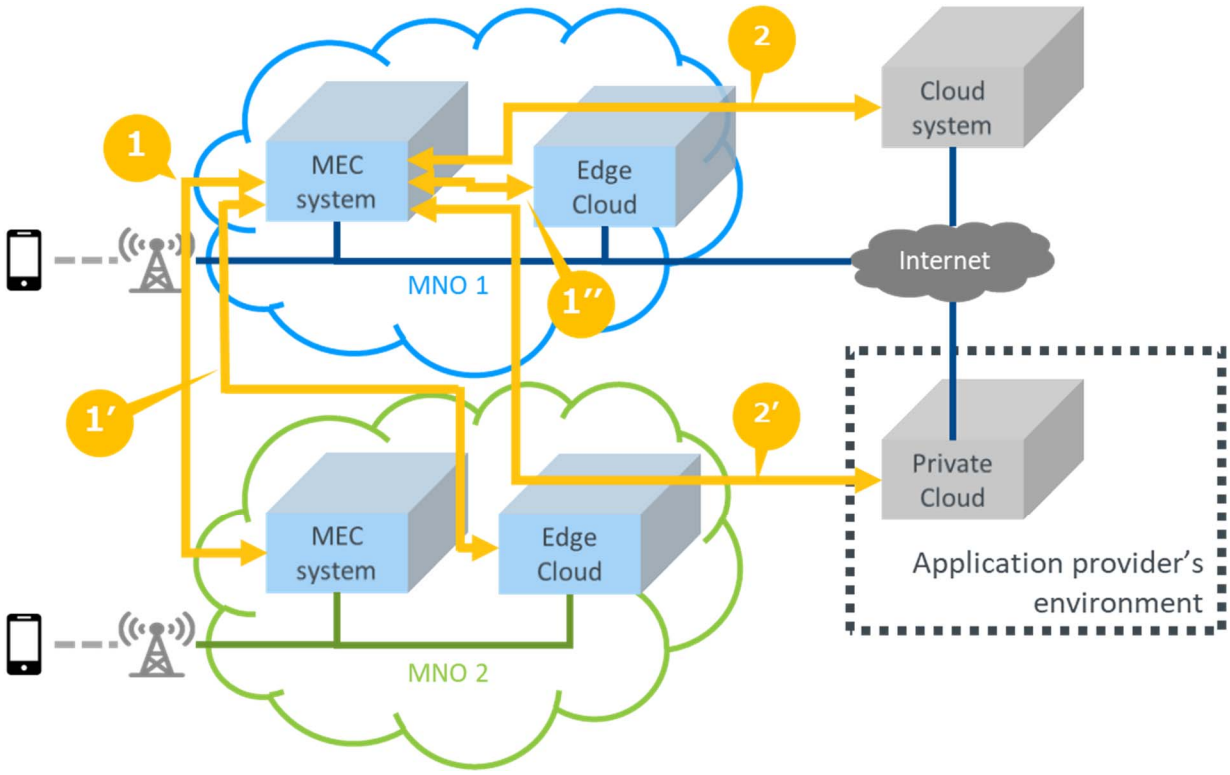


Figure 4.4-1: Patterns of business relationship

The use cases and key issues in the following clauses cover the patterns of business relationship.

5 Use cases

5.1 Use case #1: MEC federation scenario of V2X services

5.1.1 Description

We consider a typical MEC federation scenario of V2X services (i.e., multi-MNO, multi-OEM, multi-MEC), as the one illustrated in Figure 5-1.

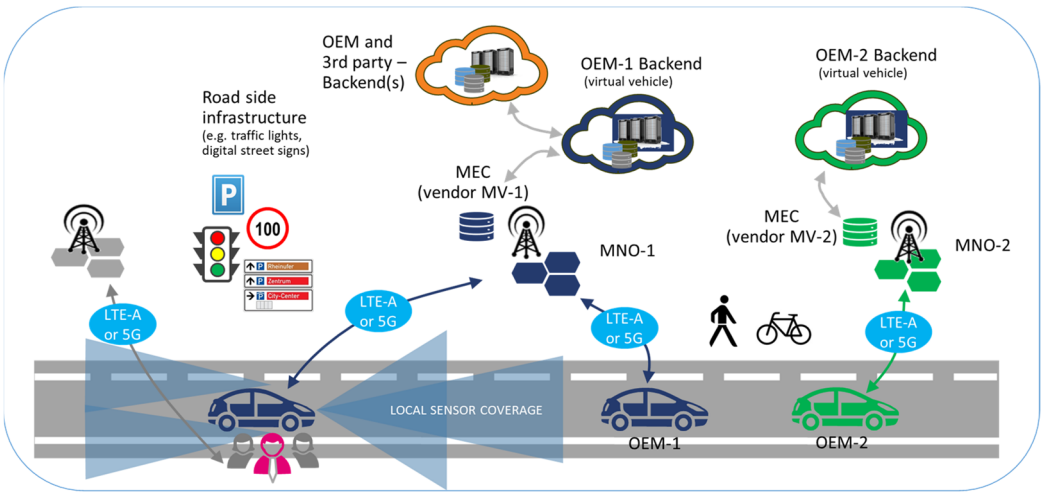


**Figure 5-1: Typical V2X multi-stakeholder scenario
(source: 5GAA member's symposium in Turin, November 2019).**

In this scenario, a V2X application instance may be running on a car connected to MNO 1 which is equipped with a MEC system from vendor 1, and communicating with another V2X application instance, running on a server, or, in general, on a second car connected to MNO 2, which, in its turn, is equipped with a MEC system from vendor 2.

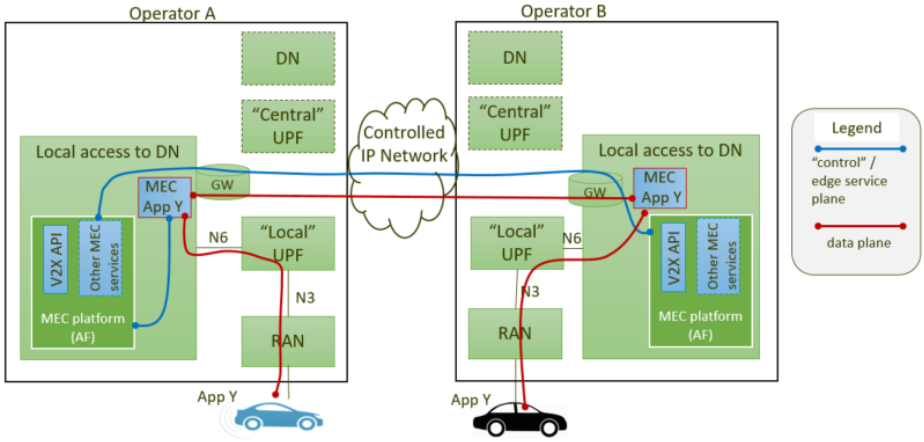


**Figure 5-2: Illustration of a MEC federation reference scenario where both MNOs have MEC platforms and a MEC application Y ("MEC App Y") is instantiated (Multiple OEM vehicle use case)
(source: 5GAA document XW2_200048, May 2020). [i.5]**

From an architectural point of view, this scenario is also depicted in Figure 5-2, where a certain V2X service is implemented with two instances of the "MEC App Y", each of which communicates with its corresponding Client App, i.e., "App Y", and is also connected with a MEC platform in each respective MEC system (domain). The "MEC App Y" instances may need to direct communicate with each other and/or consume platform services of the other MEC system.

5.1.2 Recommendations

To enable a MEC federation, the following hierarchical inter-MEC system communication levels should be introduced:

1. MEC system (i.e., below business level) discovery, including security (authentication/ authorization, system topology hiding/ encryption), charging, identity management and monitoring aspects as an essential prerequisite to form a MEC federation;
2. MEC platform discovery, by means of the MEC systems exchanging information about their MEC platforms, i.e., their identities, a list of their shared services, as well as authorization and access policies;
3. Information exchange at MEC platform level, for the needs of MEC service consumption, or for MEC app-to-app communication.

The ultimate goal is to address the needs of information exchange for MEC/edge service consumption and MEC app-to-app communication, which is related to the third item in the above list. Such information exchange refers to either a MEC application in need of consuming a MEC platform service, or a MEC application in need of communicating with other (e.g., service-producing) MEC applications.

Editor's note: Identifiers for MEC platforms and MEOs may need to be defined.

5.1.3 Evaluation

The addressment of the requirements of clause 5.1.2 is technically feasible, provided that ETSI MEC will introduce a proper hierarchical signaling framework needed to realize a MEC federation constituting of MEC systems, possibly owned and operated by different parties (e.g., MNOs).

Clause 6 includes the related key issues and proposed solutions.

5.2 Use case #2: multi-operator agreements enabling MEC Federation for V2X services

5.2.1 Description

Some federation use cases are described below:

TYPE-1 USE CASE

- A possible use case for federation can be associated to a national roaming like scenario where customers of an MNO#1 could access the edge infrastructure of MNO#2 if this operator has a complementary footprint. An end user is customer of MNO#1 but the best edge location for the MEC App to be used is in the edge infrastructure of MNO#2. When triggering the app in his device, the MEC system of MNO#1, through its federation agreement, identifies that the best edge location is in MNO#2. Then, the edge system of MNO#1 redirects the App to the MEC system of MNO#2 to ensure the best possible service.

TYPE-2 USE CASE

- An application developer has a commercial relationship with MNO#1. Through his federation agreements MNO#1 allows also the application developer to deploy its App in the MEC systems of MNO#2, MNO#3 to access their respective subscribers. Through its existing federation agreements MNO#1 provides visibility of the availability zones that can be used in MNO#1, MNO#2, MNO#3 networks. The app developer then decides of its deployment approach based upon his commercial strategy.

TYPE-3 USE CASE

- MNO#1 wants to reach the maximum possible number of federation agreements with other MNOs. To achieve this goal MNO#1 decides to make use of a federation broker who has a pre-established set of agreements with a large number of MNOs. Then MNO#1 offers to his App developers/customers the possibility to deploy in the availability zones of the MEC systems of all the MNOs part of the direct federation agreement of MNO#1 but also to the MNOs part of the federation broker portfolio.

5.2.2 Recommendations

Editor's Note: Recommendations to be added.

5.2.3 Evaluation

Editor's Note: Evaluation to be added.

5.3 Use case #3: Application instance transfer between MEC and Cloud systems

5.3.1 Description

For the better QoS or cost efficiency, the application instances are transferred from the cloud system to MEC host, e.g., in cases of shortage of backhaul network resources, activation of the MEC host, or entering the coverage of the MEC host. The current MEC specifications support the on-boarding of the application package and the instantiation of the application instance based on the request from outside. Other relevant functions are not fully specified. Furthermore,

regarding the other way, in the case where the MEC application leaves the MEC system, e.g., in the case of leaving the coverage of the corresponding MEC host, the shortage of computing resource on MEC host, or service down due to hardware errors, the application are transferred from a MEC host to the cloud system. In this context, cooperative application instance transfer between the MEC system and the Cloud system will be an essential operation for service quality and continuity. High level of the behaviour is illustrated in Fig. 5.3.1-1. Note that regarding the second application transfer that is from MEC system to Cloud system, the application instance on Cloud system is stopped and deleted during the device connects to the application instance on MEC system, e.g., in the case where the application instance on Cloud system does not associate with any other devices. Therefore, Cloud system needs to start the application instance again when the device comes back to Cloud system. Since Cloud system keeps the application package, the application package transfer is not necessary. If the application instance stays active on Cloud system after the first application transfer, the second transfer will not happen.

After the second transfer, an application instance on Cloud system may need to continue using MEC services on the MEC system, e.g., RNIS, location service, etc. In this case, the relevant information maintained by the MEC system may need to be transferred to the Cloud system for the purpose of MEC service remote consumption or equivalent service continuity.

As shown in Fig. 5.3.1-2, there are two operations for application transfer between MEC system and cloud system, (1) Distribution of the application that includes check the availability of platform service, dissemination of application package, instantiation of application instance, and synchronization of the application data, (2) Switch communication path that includes the continuity of the application and check availability of the physical resource. Recommendations are introduced based on these processes.

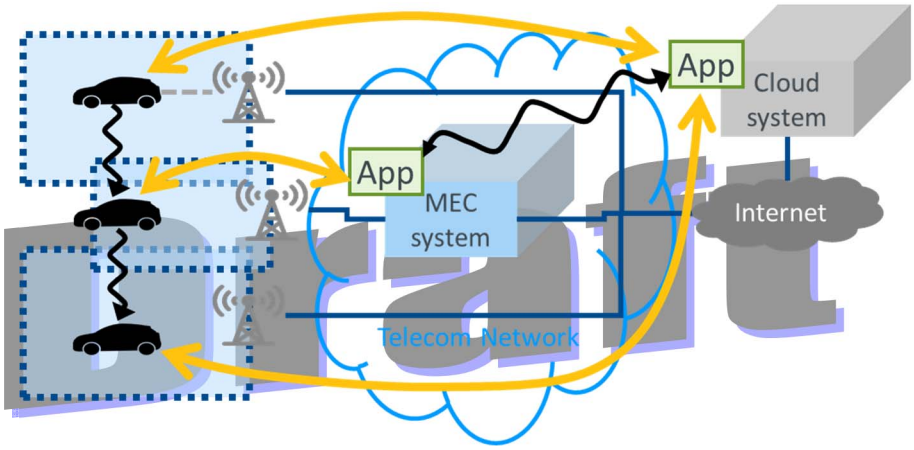


Figure 5.3.1-3: Abstract level of the behavioir

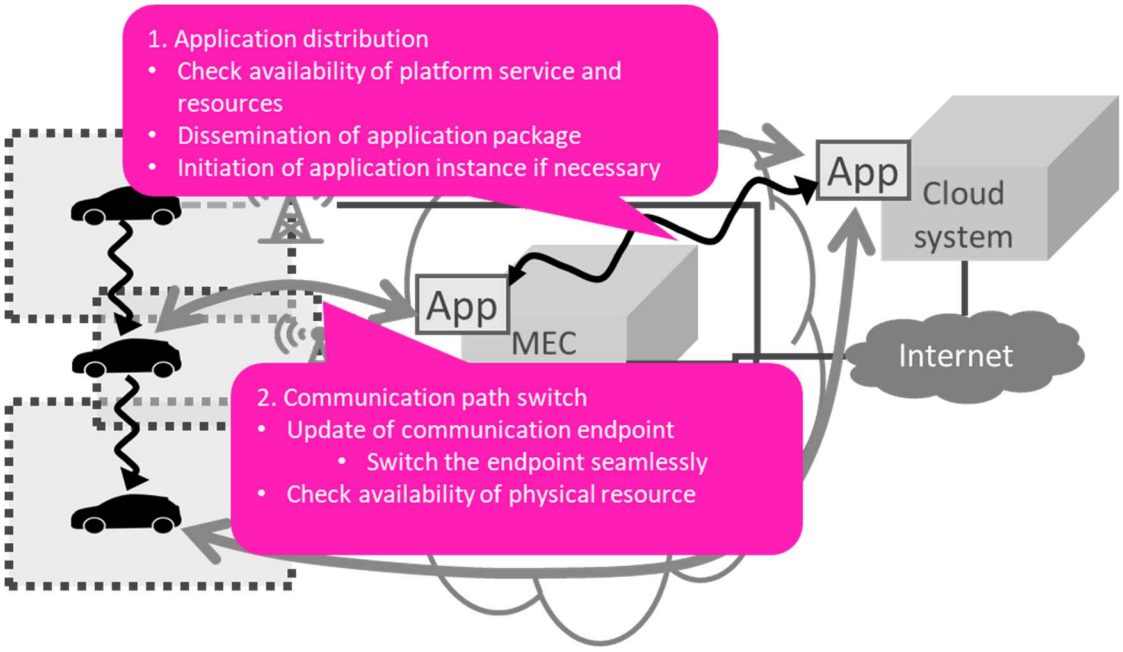


Figure 5.3.1-2: Corresponding operations

5.3.2 Recommendations

The list of recommendations for relocating application between Cloud system and MEC system are as follows.

[Recommendation 5.3.2-1]

In order to distribute the application package to the appropriate MEC host, MEC system should support Cloud system or application instance on the cloud to discover the appropriate MEC host.

[Recommendation 5.3.2-2]

In the case of transfer from cloud to MEC, MEC system should support the cloud system to check if the availability of MEC system prior to the application instance transfer/distribution. The relevant information is provided if needed. In the case of transfer from MEC to cloud, The MEC system should support to confirm the availability of the cloud system if needed.

[Recommendation 5.3.2-3]

Same application packages need to be distributed in the MEC system prior to the application onboarding. For this purpose, the MEC system should support to validate the application package.

[Recommendation 5.3.2-4]

If needed, user context should be transferred for the service continuity. The information of service subscription, e.g., list of registered identifiers for RNIS, and subscription for event notification from NEF) should be handled, e.g., transferred, synchronized, and deleted, among MEC system and cloud system, in order for the application on cloud system to remotely consume MEC services.

[Recommendation 5.3.2-5]

The MEC system should support to instantiate application instance. The instantiation is based on the request from application via the cloud system or directly from the cloud system.

[Recommendation 5.3.2-6]

The MEC system should support to request to instantiate or re-start application instance on the cloud system when transferred from MEC system to Cloud system.

[Recommendation 5.3.2-7]

The MEC system should support to switch the endpoint of the communication path from the Cloud host to the MEC host, MEC system should notify of the relevant information after the application relocation is completed.

5.3.3 Evaluation

The list of evaluations that corresponds with the recommendations is as follows.

[Evaluation for Recommendation 5.3.2-1]

MEC platform discovery from the external system is not specified in the current specifications.

Editor's note: A potential solution for MEC platform discovery should be dealt in Clause 6 Key issues.

[Evaluation for Recommendation 5.3.2-2]

Advertisement, notification, or exposure of service availability should be treated as items for further study.

Editor's note: A potential solution for exposure of service availability should be dealt in Clause 6 Key issues.

[Evaluation for Recommendation 5.3.2-3]

In order to transfer or distribute the same application packages among MEC system and the cloud system, the coordination among them are needed. The current MEC003 [i.2] specification supports the instantiation of the application instance based on the request via Mx1 or Mx2from the outside. Application package onboarding is specified in MEC010-2 [i.6]. MEC010-2 supports general check of the application package prior to application package on-boarding based on application manifest file and application descriptor. The requirement is satisfied.

[Evaluation for Recommendation 5.3.2-4]

User context transfer should be conducted via a user plane, therefore, the recommendation is satisfied. However, information of service subscription, i.e., MEC application's subscription to MEC services (e.g., list of identifiers to associate the information for a specific UE or flow) and MEC service's subscription to the external system (e.g., subscription for event notification from NEF) is not supported to be handled, e.g., transferred, synchronized, and deleted. Corresponding reference point is missing in MEC003, interface is not specified in MEC010-2, and call flow is not specified in the current specifications.

Editor's notes:

- A potential solution for handling of service subscription information should be dealt in Clause 6 Key issues.

[Evaluation for Recommendation 5.3.2-5]

According to MEC003, the reference points for the instantiation of application instance from the external system are Mx1 or Mx2. OSS or LCM proxy are responsible to forward the request to MEO. The instantiation of the application instance is specified in MEC010-2 via Mm1. The requirement is satisfied.

[Evaluation for Recommendation 5.3.2-6]

According to MEC003, MEO supervise the relocation of the application instance between MEC system and the external systems. However, MEC003 does not specify the corresponding reference points.

[Evaluation for Recommendation 5.3.2-7]

DNS rules are updated by MEP as specified in MEC003. MEC system support to notify the device of the appropriate URI/IP address of the endpoint via Mx2 as specified in MEC 016 [i.7]. However, the way to notify the application instance on the cloud system of the appropriate URI/IP address of the endpoint is not specified.

Editor's note: A potential solution for notification of the appropriate URI/IP address should be dealt in Clause 6 Key issues. What information should be exchanged between MEC system and Cloud system should also be considered in the key issue.

5.4 Use case #4: Combination of different access networks

5.4.1 Description

An example for the inter MEC system mobility is to combine both cellular and Wi-Fi networks. A mobile network operator provides a Wi-Fi network as an efficient alternative option to mitigate the cellular network congestion or to offload network traffic. A Wi-Fi network is complementarily deployed for the cellular network and its access points are distributed in cities, especially in the dense area or the specific location where requires high throughput, e.g., a user device is likely to transfer the enormous volume of data via Wi-Fi network. In this case, the resource capabilities of corresponding MEC environment are different as well as the network topology and capacity. It is logically possible to integrate those MEC environment, which means that only one orchestrator controls the entire MEC system linking both cellular and Wi-Fi network. However, due to the asymmetry of those resources or limitation of the facilities, the availability or performance of MEC services are also asymmetry. Therefore, it might be better to deploy/operate/manage those MEC systems separately. In this context, user device likely to handover from one to the other as depicted in Fig. 5.4.1-1.

From the view point of system behaviour, that process includes mainly two operations as illustrated in Fig. 5.4.1-2. (1) Distribution of the application package, instantiation of application instance, check the availability, and synchronization of the application data, and (2) Switch communication path that includes the continuity of the application and check availability of the physical resource. Recommendations correspond to these operations.

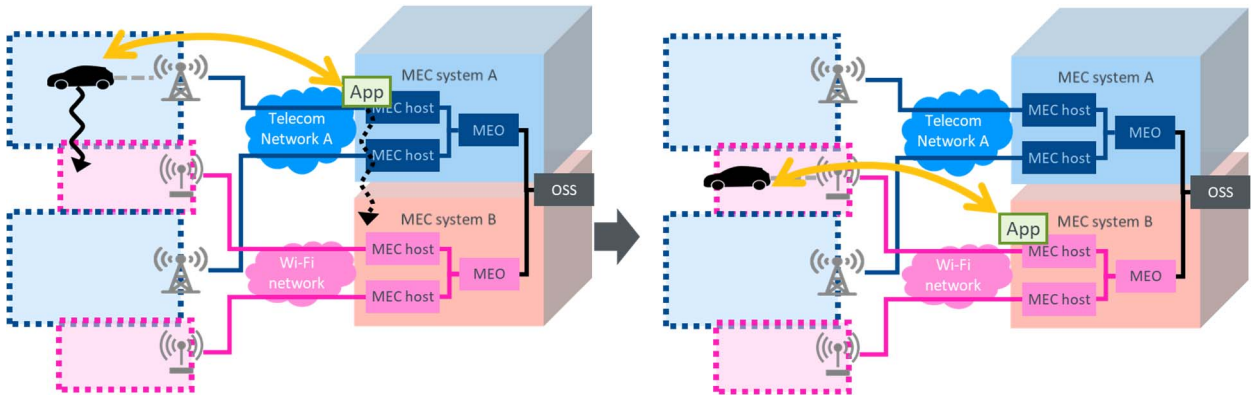


Figure 5.4.1-1

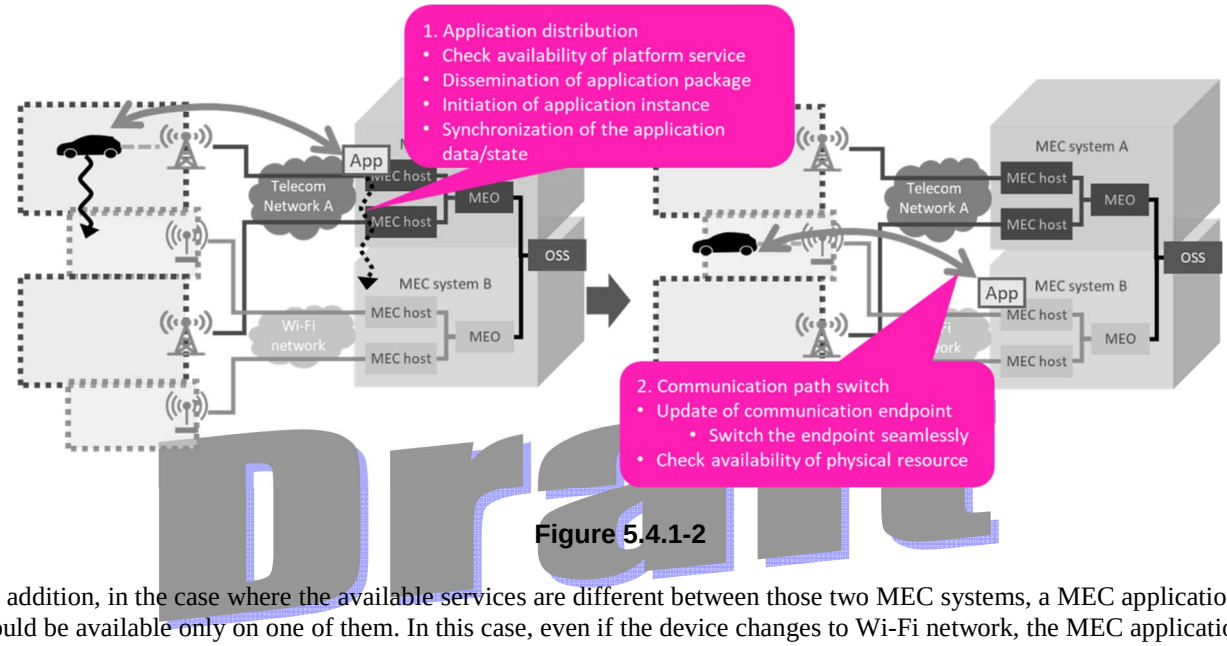


Figure 5.4.1-2

In addition, in the case where the available services are different between those two MEC systems, a MEC application could be available only on one of them. In this case, even if the device changes to Wi-Fi network, the MEC application stays on the source MEC host. The device expects to connect to the application through Wi-Fi network, through MEC system B if necessary. The high-level behaviour is described in Fig. 5.4.1-3.

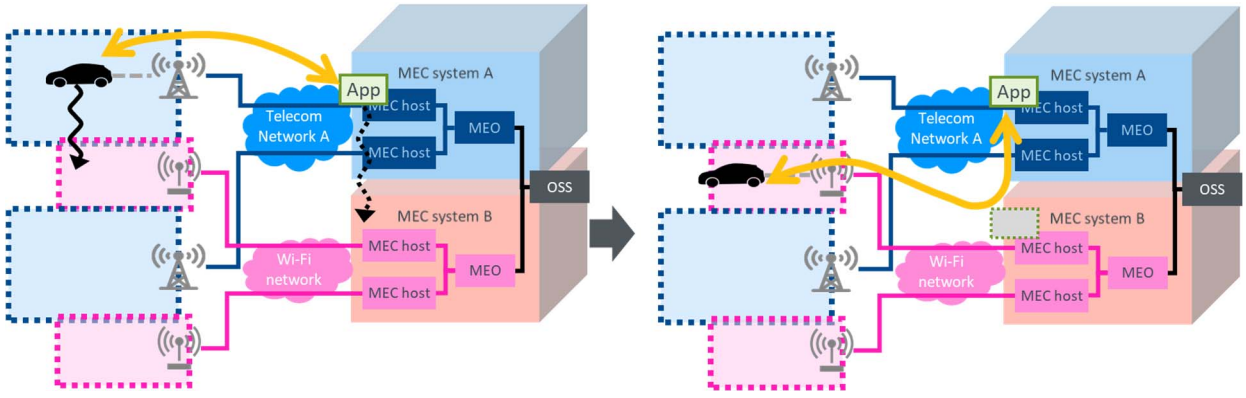


Figure 5.4.1-3

5.4.2 Recommendations

The list of recommendations are as follows.

[Recommendation 5.4.2-1]

In the case of transfer from the source MEC system to the target MEC system, the target MEC system should support the source MEC system to check the availability of the target MEC system prior to the application instance transfer/distribution. The relevant information is provided if needed.

[Recommendation 5.4.2-2]

Same application packages need to be distributed in both the source and target MEC systems prior to the application onboarding. For this purpose, the MEC systems should support to validate the application package.

[Recommendation 5.4.2-3]

If needed, user context should be transferred for the service continuity. The information of service subscription, e.g., list of registered identifiers for RNIS, and subscription for event notification from NEF) should be handled, e.g., transferred, synchronized, and deleted, among the source and target MEC systems.

[Recommendation 5.4.2-4]

The MEC system should support to instantiate application instance. The instantiation is based on the request from application via the source MEC system or directly from the source MEC system.

[Recommendation 5.4.2-5]

The MEC system should support to switch the endpoint of the communication path from the source MEC host to the target MEC host, the target MEC system should notify of the relevant information after the application relocation is completed.

[Recommendation 5.4.2-6]

The target MEC system should support to provide the connection between a device and MEC application on the source MEC host if needed. If the access network provides the connectivity between them (e.g., roaming), it is not necessary. The source MEC system should allow devices to connect to the application via different MEC systems. It should expose its own MEC platform services to other MEC systems if necessary.

5.4.3 Evaluation

The list of evaluations that corresponds with the recommendations is as follows.

[Evaluation for Recommendation 5.4.2-1]

Advertisement, notification, or exposure of service availability should be treated as items for further study.

[Evaluation for Recommendation 5.4.2-2]

In order to transfer or distribute the same application packages among multiple MEC systems, the coordination among them are needed. In this context, two direction of the transfer/distribution should be considered, i.e., receiving and sending. MEC010-2 supports the case where the application on-boarding request is received via OSS. Since this case considers an inter MEC systems deployment, the extension of Mx1 or Mx2 are not necessary. The extension may need if the application on-boarding is triggered via other interfaces or the other direction.

[Evaluation for Recommendation 5.4.2-3]

User context transfer should be conducted via user plane, therefore, the recommendation is satisfied. However, information of service subscription, i.e., MEC application's subscription to MEC services (e.g., list of identifiers to associate the information for a specific UE or flow) and MEC service's subscription to the external system (e.g., subscription for event notification from NEF) is not supported to be handled, e.g., transferred, synchronized, and deleted. Corresponding reference point is missing in MEC003, interface is not specified in MEC010-2, and call flow is not specified in the current specifications.

Editor's notes:

- A potential solution for handling of service subscription information should be dealt in Clause 6 Key issues.

[Evaluation for Recommendation 5.4.2-4]

According to MEC003, the corresponding reference points for receiving the request for instantiation are specified as Mx1 and Mx2. The instantiation of the application instance is specified in MEC010-2 via Mm1. However, sending the request for instantiation to the external MEC system is not supported. Corresponding call flow and relevant interfaces should be further specified.

Editor's note: A potential solution for sending the request for instantiation to the external MEC system should be dealt in Clause 6 Key issues.

[Evaluation for Recommendation 5.4.2-5]

DNS rules are updated by MEP as specified in MEC003. MEC system support to notify the appropriate URL/IP address of the endpoint via Mx2 as specified in MEC 016. However, currently MEP has no way to obtain the appropriate DNS rules that steer to the external systems. How to define the appropriate DNS rules should be further specified.

[Evaluation for Recommendation 5.4.2-6]

Application traffic path update is studied in MEC 016 and specified in MEC 021 [i.9]. However, corresponding operations are limited to intra-MEC system. Application traffic path update between different MEC systems is for further study.

5.5 Use case #5 MEC federation scenario for connecting different services

5.5.1 Description

Nowadays, it is very common to provide new functionalities through collaborating with other services rather than developing all of them. For example, a voice recognition function can work as a key feature within other services, such as a navigation application. In that case, the voice recognition service provider is not necessarily same as the navigation service provider, and each service can be deployed on different MEC systems in the MEC environment.

This scenario is depicted in Fig 5.5.1-1. "MEC App X" (e.g., a navigation service) provides a service to a user through a user's client application "App X" and improve its service quality in cooperation with "MEC App Y" (e.g., a voice recognition service). "MEC App Y" supports its functions by connecting with "MEC App X" not "APP X". Even if "MEC App X" and "MEC App Y" are deployed on different MEC systems or on different MNOs, the communicating path is supported in case of a MEC federation.

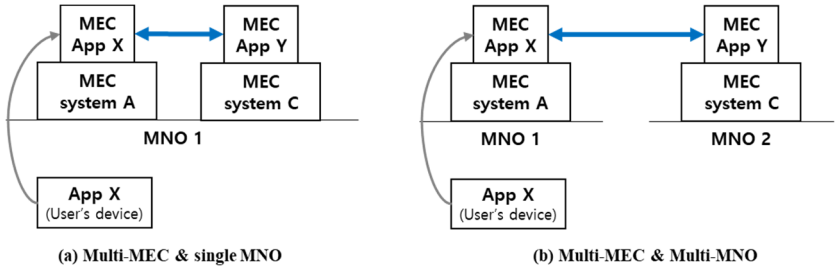


Figure 5.5.1-1 Communicating between different MEC Apps in multi-MEC environment over single or multi MNO

5.5.2 Recommendations

The list of recommendationsare as follows:

[Recommendation5.5.2-1]

When federating, each MEC system should register relevant information with its federation management entities including computing resources, Network resources, MEC application information, and etc.

[Recommendation5.5.2-2]

Federation management entities should support the exchange of information among MEC federation members.

[Recommendation5.5.2-3]

When a MEC federation is formed and MEC apps are deployed on hosts of MEC federation members, MEC app-to-app communication should be supported when multiple MEC application providers are involved.

[Recommendation5.5.2-4]

It should be supported to find the appropriate MEC platform related to the MEC application instance to communicate with on the basis of requests from the federation management entity. In that case, the relevant information (e.g. user location) should be considered to maintain service quality that the MEC service can provide.

Editor's note: The term of "federation management entities" needs to be defined

5.5.3 Evaluation

Editor's Note: Evaluation to be added.

5.6 Use case #6 MEC federation scenario for immersive AR game

5.6.1 Description

Augmented reality (AR) provides an interactive experience of a real-world environment mixed with computer-generated perceptual information and contents.

Entertainment looks to become one of the biggest applications of immersive AR content. Sport, music etc. applications will target the attendees of a specific event to provide on-site entertainment services. Also, this introduces a new class of games, in which the physical environment, where the users are located, becomes an integral part of the game.

AR games incorporate diverse scenarios based on real-world settings and users' context such as viewpoint and player actions to provide them with fully immersive experience. Network latency and data rate play critical roles in delivering uninterrupted gaming experience. In this regard, one of the biggest hurdles in expanding AR applications widely is the need for E2E QoS assurance with high-bandwidth and low-latency. Battery capability of the mobile device is another indispensable consideration because running AR applications requires intensive computing resource use which results in massive battery consumption.

However, with the emergence of 5G and MEC, those are becoming less and less of obstacles. MEC is envisioned as a promising means to deliver better quality of experience (QoE) for immersive AR applications by reducing the delay and by addressing computation-intensive and battery-consuming tasks offloaded from the mobile devices.

Here, we focus on a location-based immersive AR game whose scenario is designed to be played by all players at a specific geographical area. MEC fits well to these kinds of location-based immersive AR games in a sense that they are played by users in a certain location.

Without a MEC federation, however, there is a limitation in providing interactive AR application with users connected to different MNOs. For example, a multiplayer interactive AR game can be supported only when the users joining the game are connected to the same MNO. Users of different MNOs cannot join the multiplayer interactive AR game even when they are located nearby. This scenario is illustrated in Figure 5.6.1-1.

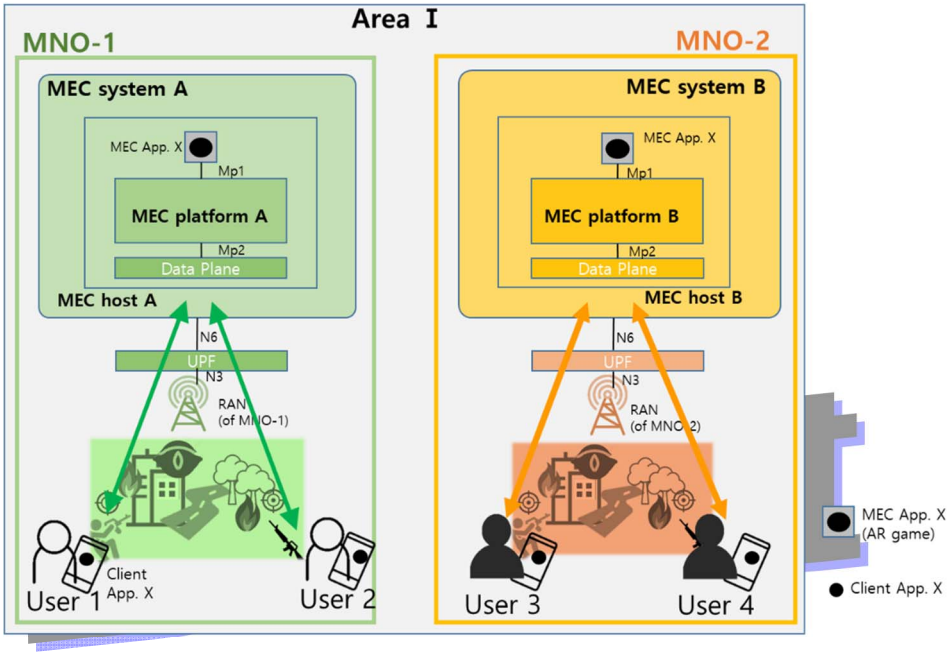


Figure 5.6.1-1: Illustration of a multiplayer interactive AR game scenario without a MEC federation. In this environment, user 1 and user 2 of MNO-1 can play together by the help of MEC platform A. User 3 and user 4 of MNO-2 can play together by the help of MEC platform B respectively. User 1 and User 4 connected to different MNOs cannot play together even when they are located nearby.

A MEC federation can be a solution to this limitation.

By a MEC federation, a multiplayer interactive AR game can be enjoyed by users connected to different MNOs and this scenario is illustrated in Figure 5.6.1-2 and Figure 5.6.1-3. Two options may be possible in incorporating multiplayer interactive games under MEC federation environment.

The first option, illustrated in Fig. 5.6.1-2, is to coordinate multiple MEC application instances of same kind where each of them is providing game service to the users connected to a MNO equipped with its respective MEC system.

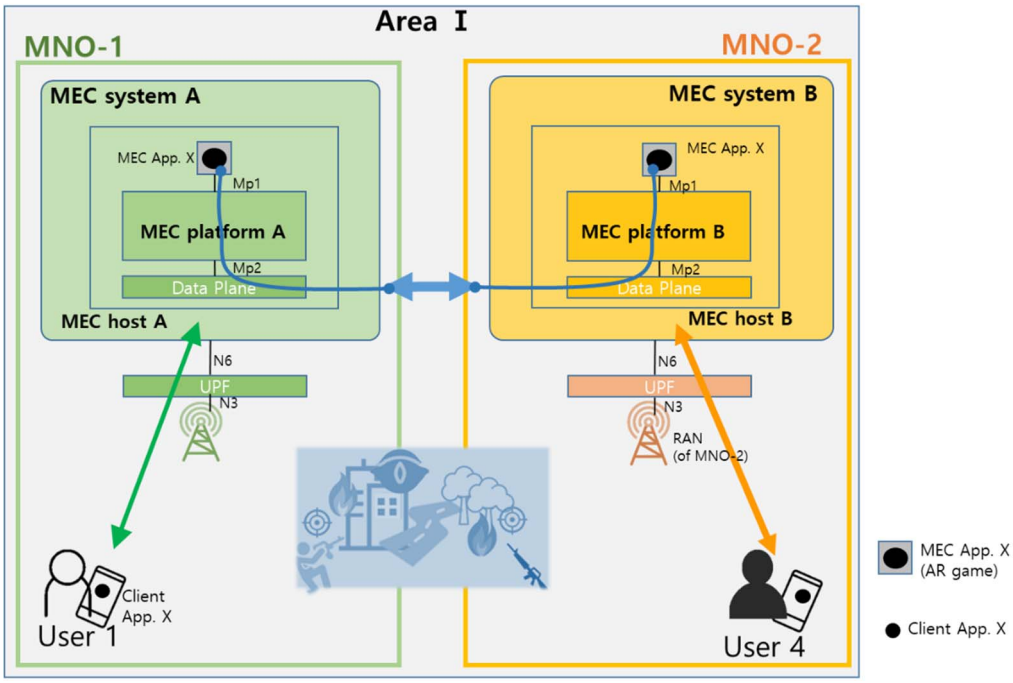


Figure 5.6.1-2: Illustration of a multiplayer interactive AR game scenario under a MEC federation. Option (1): In this environment, users of different MNOs, user 1 of MNO-1 and user 4 of MNO-2, can join a multiplayer interactive AR game and play together. The two AR game MEC application X instances coordinate for real-time synchronization.

In this example case, the two MEC application Xs, instantiated on MEC hosts of MEC system A and MEC system B respectively, communicate and coordinate together for synchronizing the game scenario. Information to be exchanged between the two MEC applications for coordination mostly include users' game play actions such as players' position, movement, direction, game control and the status of game contents virtually created.

The coordination and synchronization mechanism is specific to application implementation. However, the basic idea of how the applications are associated is represented below since it is closely related to a MEC federation.

A user – e.g., user 1 in this case and let's call this a leader - needs to create a 'multiplayer game room' to enjoy a multiplayer mode on a game server running on MEC host, MEC host A in this case. The leader can set a secret key for the multiplayer game room and share it with the desired users he wants to play together.

Thereafter, the MEC application X instantiated on MEC host A transfers the 'multiplayer room' information to other MEC application X instance on the other MEC hosts within the MEC federation, MEC host B in this example.

The desired user – user 4 in this case - can enjoy the multiplayer game by entering the 'multiplayer game room' when he connects to the game server, i.e., the MEC application X running on MEC host B in this case.

Following MNO agreement, there exists a direct IP network between the associated MEC systems owned and operated by different MNOs.

In the other possible option, as illustrated in Fig. 5.6.1-3, one main application instance plays the main role in providing the game scenarios to all the users who joined the multiplayer mode including users connected to different MNOs.

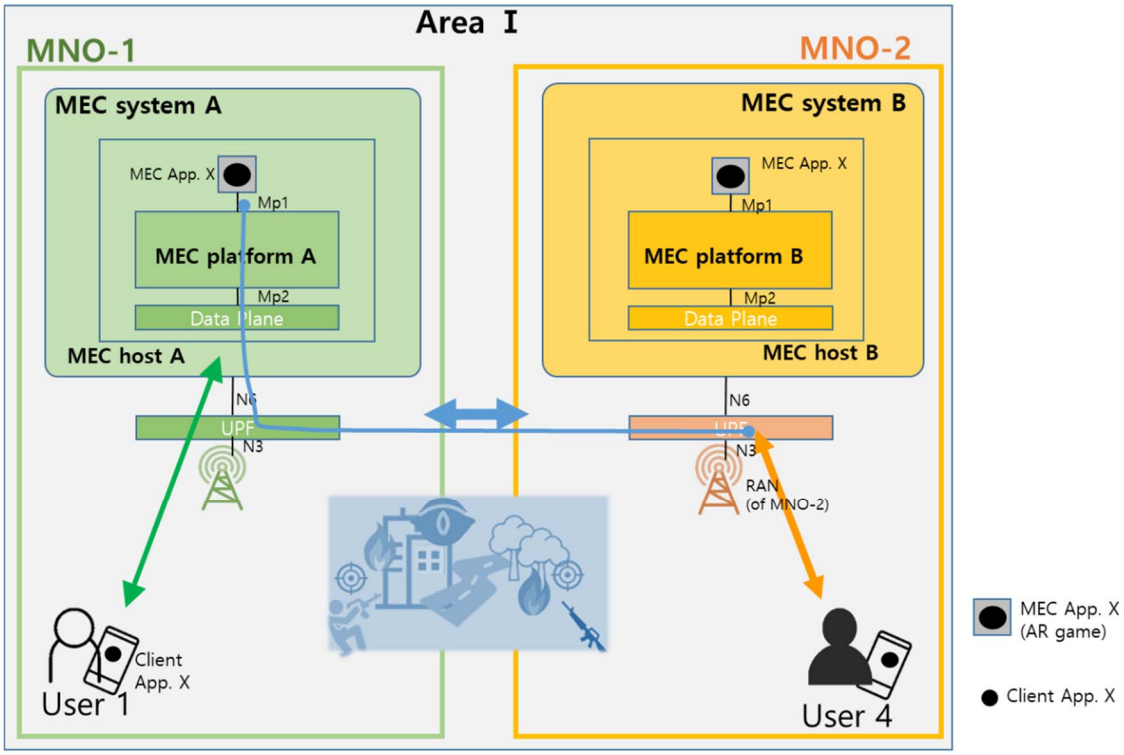


Figure 5.6.1-3: Illustration of a multiplayer interactive AR game scenario under a MEC federation. Option (2): In this environment, users of different MNOs, user 1 of MNO-1 and user 4 of MNO-2 can join a multiplayer interactive AR game and play together. The MEC platform B switches the traffic from user 4 for MEC application X to MEC platform A.

In this example case, MEC application X running on the MEC host A of MNO-1 is the main application instance.

This may be decided by the MEC application instance where a user – the leader, user 1 in this case - creates a multiplayer game room.

Thereafter, this main instance – in this case, the MEC application X on MEC host A – transfers this information to other MEC application X instantiated on the other MEC hosts of the MEC federation, MEC host B in this example.

The MEC application X running on MEC host B needs to set a traffic rule so that the traffic from user 4 to it can be switched to the main MEC application X instance – the one running on MEC host A in this case.

In this way, both user 1 and user 4 can enjoy the multiplayer mode together while being served by MEC application X running on MEC host A.

Following MNO agreement, there exists a direct IP network between the UPFs of different MNOs within the MEC federation

5.6.2 Recommendations

The list of recommendations is as follows:

[Recommendation 5.6.2-1] For option 1, it is recommended to enable a MEC application instance to discover another MEC application instance (of the same application) in the same or different MEC system. This includes the further recommendation that key performance indicators (e.g. latency) offered (i.e. achievable KPIs) by the discovered MEC application instance and the inter-domain connectivity are made available in the response and that filtering criteria (e.g. KPIs, location constraints) can be applied in the request to support discovery of appropriate MEC application instances.

[Recommendation 5.6.2-2]

For option 1, it is recommended to, subject to the agreement of the involved parties (e.g. operators and App providers), support the on-boarding and/or instantiation of a MEC application in a MEC system in response to a request with the key performance indicator (e.g. latency) by another MEC system.

[Recommendation 5.6.2-3]

1. For option 2, it is recommended to support the MEC application (server) selection in an MNO's MEC system for a group of clients that may be subscribers of different MNOs. The suitable MEC application (server) should meet the performance requirements (e.g. latency) for the group of clients.

2. For option 2, it is recommended to support a suitable rule for the efficient handling of the traffic between the MEC application (server) hosted in an MNO's MEC system and the another MNO's access network where the UE (that the App client resides in) is connected.

3. For option 2, it is recommended to support the MEC application (server) instance assessing the achievable key performance indicators that could be provided to potential App clients.

[Recommendation 5.6.2-4]
In the case where there are three or more clients in the group, both options 1 and 2 can be selected at the same time.

5.6.3 Evaluation

The addressment of the requirements of clause 5.6.2 is technically feasible, provided that there is a prior MNO agreement enabling inter-domain IP-based connectivity between MEC systems and/or UPFs operated by the involved MNOs, MEC federation management entities enabling MEC application instantiation within the MEC federation per a QoS requirement, along with the ability of a MEC application instantiated at a MEC host of a MEC system to request the setting, deactivation and deletion of traffic rules from the UPF of the 3GPP network where the said MEC system has connectivity to.

Editor's Note: Additional evaluation to be added.

5.7 Use case #7: MEC federation scenario for Edge Service availability on visited networks

5.7.1 Description

When a subscriber of one operator is roaming on another operator's network (visited network), the MEC service should still be delivered with the same performance as on the home network.

For that purpose, MEC applications should optimally be delivered from the visited network, including the proper service access from the client app to the MEC hosts and the control of the MEC host where the service will be delivered from.

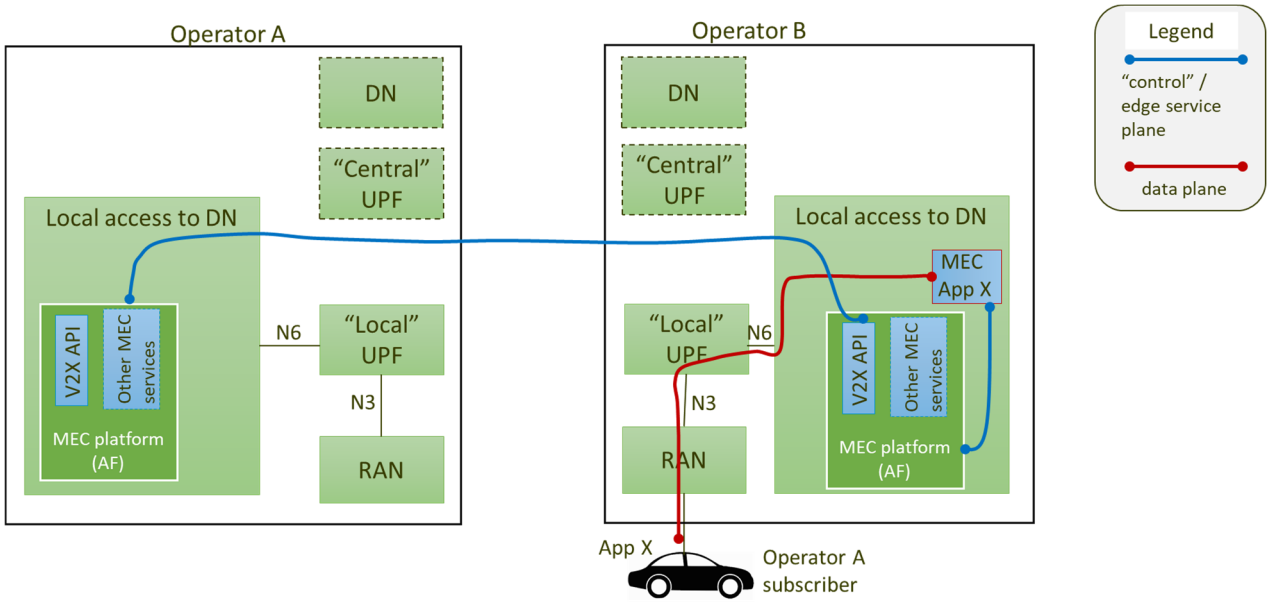


Figure 5.7.1-1: Scenario according to which a subscriber of Operator A is roaming on Operator B's network; each operator owns and manages its own MEC system (source: 5GAA)

Without a MEC federation, users will remain attached to their home MEC system and the application traffic (from the client app) will need to travel to home operator MEC platforms, with a degradation of the service performance. The MEC federation will allow the home MEC system to direct users to the system on the visited network to join the service there.

Home MEC system should be able to identify that a user is camping on a roaming environment and, if local breakout (LBO) is available, direct the user's MEC application traffic to the visited MEC system. User credentials should be shared between the two MEC systems, so that the visited MEC platform can identify the user.

Concurrently, MEC systems should ensure that the application backend is available on the visited MEC system, so the federation interface may be used to share applications from one MEC system to the other.

5.7.2 Recommendations

[Recommendation 5.7.2-1]

Authentication and authorization of the users is only available on the home MEC system, since identities are supposed to be handled by its own network operator. First attachment of the user should then always be driven to the home MEC system, which may then get in charge of driving the user to other MEC system, including the credentials, or allow the visited MEC system to retrieve those credentials from the home MEC system.

[Recommendation 5.7.2-2]

User attachment should remain on visited MEC system until a network change is triggered (i.e. radio handover), so that the binding to the system is not based on application request.

[Recommendation 5.7.2-3]

Network implications, including local breakout configuration of the operators interconnections, should be considered.

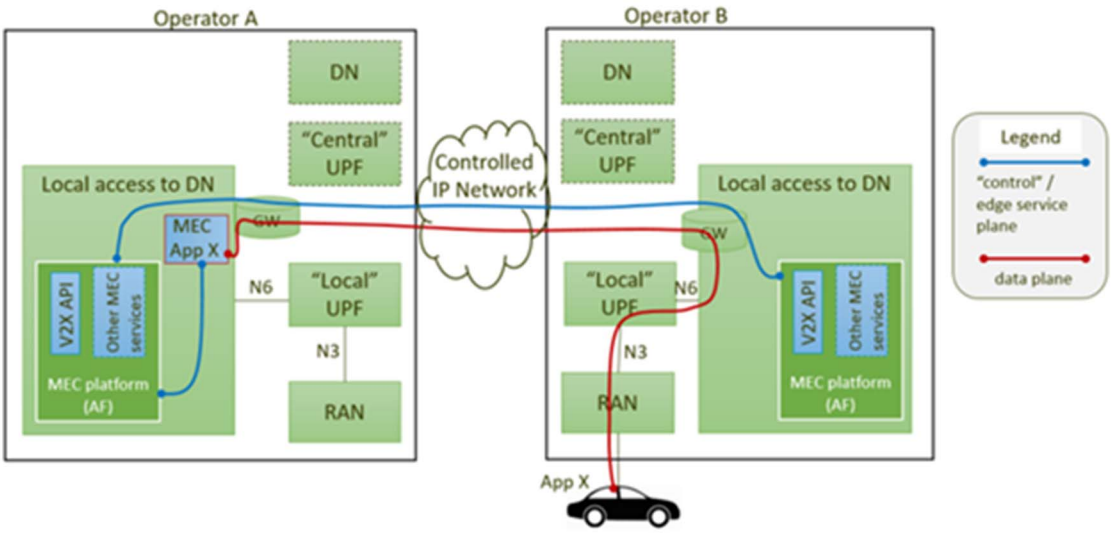
5.7.3 Evaluation

Editor's Note: Evaluation to be added.

5.8 Use case #8: MEC federation scenario for edge node sharing

5.8.1 Description

The MEC federation may be also used to share edge capabilities form one operator to other, on those situations where one of them has no edge resources on a certain region.



With edge node sharing, one subscriber of Operator B will remain attached to operator's B network, while accessing the edge services from an edge Platform of Operator A.

Federation interface should enable:

- Application discovery/publishing, so that operator's B MEC system is aware of operator's A application availability, and can determine that it is the most optimal location to deliver service to the subscriber.
- Subscriber redirection, following the procedures on use case #7 Service availability on visited network.

5.8.2 Recommendations

[Recommendation 5.8.2-1]

Connectivity between MEC platforms of operator A and network gateway of operator B (and vice versa) should be considered to optimize the service delivery form one operator to the other.

[Recommendation 5.8.2-2]

Same considerations as in use case #7. Service availability on the visited network should be considered.

5.8.3 Evaluation

Editor's Note: Evaluation to be added.

5.X Use case #X

5.X.1 Description

5.X.2 Requirement and recommendations

5.X.3 Evaluation

5.Y Summary

6 Solutions for closing the gaps

6.1 Gap/Key issue #1 - Structuring the needed signalling for secure communication among different MEC systems

6.1.1 Description

We consider typical MEC federation scenarios and key issues, as described in use case #1,#2, and #5 (clause 5).

The problem to be addressed is how to structure the needed signaling/messages for secure communication among different MEC systems, possibly owned and operated/managed by different entities (e.g., MNOs) for the needs of information exchange. Such information exchange refers to the following:

- For systems to establish a security trust by authenticating and authorizing each other;
- for an application provider /customer to deploy its load/application across multiple MEC systems using a single MNO relationship and integration (same Northbound interface);
- for a MEC application in need of consuming a MEC platform service, or,
- for a MEC application in need of communicating with other (service-producing) MEC applications.

6.1.2 Solution proposal #1-1

Signalling among specific functional entities of the involved MEC systems should be performed to address the recommendations of clauses 5.1.2 and 5.2.2. Figure 6.1-1 illustrates the considered hierarchical functional levels based on which a MEC federation can be formed by means of a proper signalling. In figure 6.1-1, a Federation Manager is newly considered in this document and described in the clause 6.2.

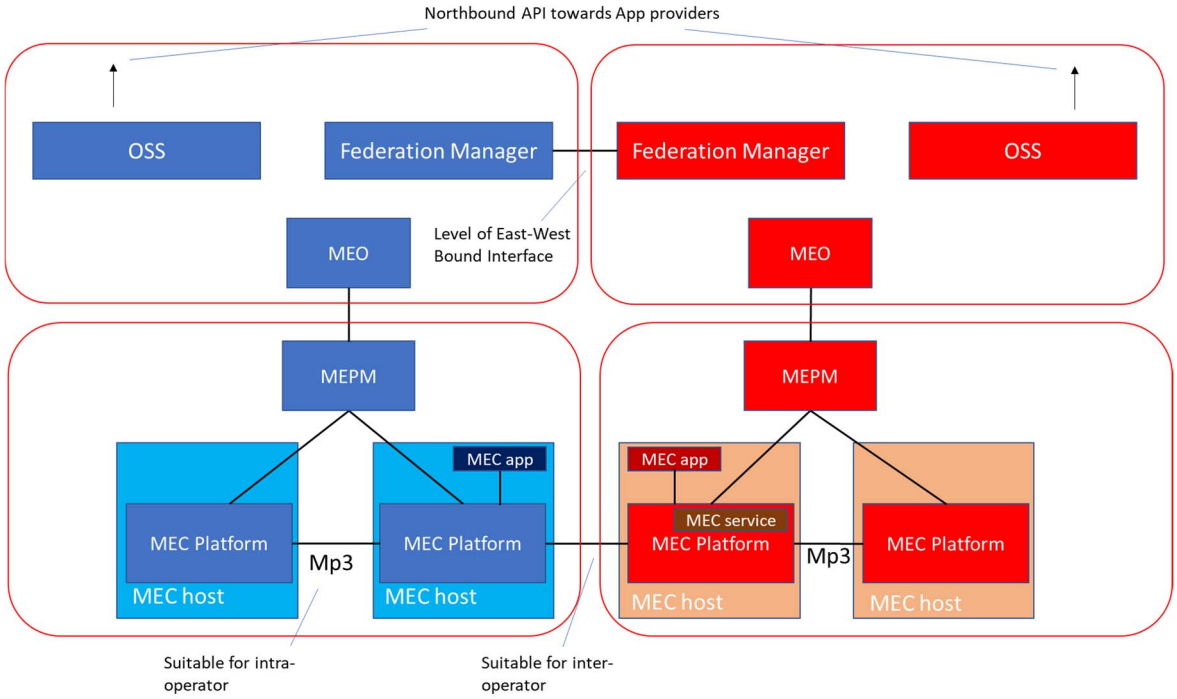


Figure 6.1-1: The considered hierarchical functional levels based on which a MEC federation can be formed by means of a proper signaling.

6.2 Gap/Key issue #2 – Considering entities for MEC federation

We consider typical MEC federation scenarios including V2X services(i.e., multi-MNO, multi-OEM, multi-MEC), as described in clause 5.1, 5.2, and 5.5. To support these scenrios, the entity which is responsible for MEC federation is required.

6.2.1 Description

Under the current MEC architecture, there is no role and entity that manages all MEC system information and discover and communicates other MEC systems. However, in case of MEC federation, inter-MEC system communicationis required, and it's needed newly to consider appropriate entities, a Federation Manager and a Federation Broker

It is supposed that the Federation Manager and Federation Broker deal with all the policies defined among the various MEC systems (and, in particular, the respective MEOs), according to which inter-MEC-system communication is allowed and can be realized.

6.2.2 Solution proposal #1 Federation Manager

The Federation Manager is located in the MEC system level and connected to MEO depicted in figure 6.2.1.The new reference points can proposed. The first one, Mff-fed is for connecting between Federation Managers of different MEC systems and the second one, Mfm-fed, is connecting with its own MEO and delivering requests from other Federation Managers.

The Federation Manager is mainly responsible for supporting inter-MEC system communication with these following functionalities:

- Authorization, authentication and control access for MEC federation members;
- Security, flow control and topology/identity hiding/encryption;
- Applicaton life cycle management (e.g., forwarding intantiation/termination request)

- Resources/platform publishing and discovery;
- Exposure of the catalog of MEC systems;
- Publishing and discovering dealing all assurance functionalities (charging, monitoring, etc);

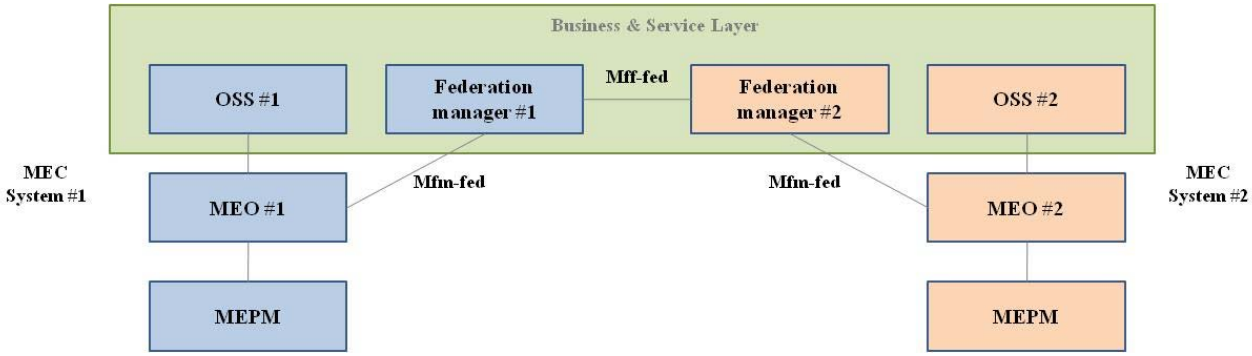


Figure 6.2-1: High-level Framwork for Federation Mangager and reference points

Editor's Note: Business and Service layers to be introduced in the GR (in an introductory clause).

Editor's Note: Federation Manager discovery is FFS.

6.2.2 Solution proposal #2 Federation Broker

We consider primarily a Federation Manager entity for each MEC system with P2P agreements between them. Nevertheless, as an alternative option, also a Federation Broker could be considered in order to reduce complexity to reach a high number of federation agreements, as illustrated in Figure 6.2.2. The present solution proposal is applicable to both variants. In case of considering a Federatin Broker, a new reference point between a Federation Manager and Federation Broker, Mfb-fed, can be considered.

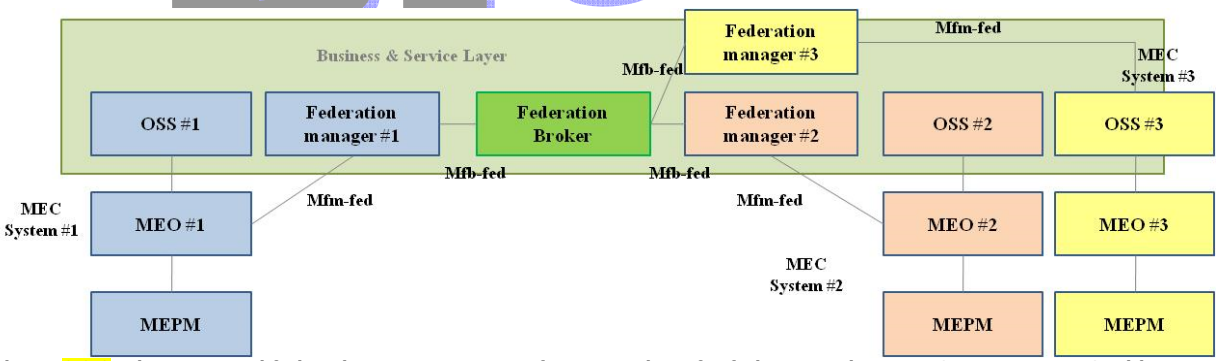


Figure 6.2-2: The proposed federation management reference point Mfm-fed connecting a MEC system's MEO with a Federation Manager. In this implementation variant we consider a single Federation Broker for the whole MEC federation

Editor's note: The scope of reference points Mfb-fed and Mff-fed needs to be defined.

6.3 Gap/Key issue #3 – MEC system discovery

We consider typical MEC federation scenarios, as described in Clause 5.

As described in Clause 5.1, to form a MEC federation, the following inter-MEC system communication level should be introduced:

- MEC system (i.e., below business level) discovery, including security (authentication/ authorization, system topology hiding/ encryption), charging, identity management and monitoring aspects as an essential prerequisite to form a MEC federation;

The ultimate goal is to address the needs of information exchange, for the needs of MEC/edge service consumption. Such information exchange refers to either a MEC application in need of consuming a MEC platform service, or a MEC application in need of communicating with other (e.g., service-producing) MEC applications.

The present gap/key issue analyzes the case of MEC system discovery.

6.3.1 Description

As a prerequisite, before inter-MEC system communication takes place to enable platform service consumption, or, MEC app-to-app communication, the MEC system #1 (and, in particular, MEO #1) needs to identify which MEC systems are members of an already established MEC federation, or, which MEC systems are available to form a new MEC federation. This identification phase of MEC systems is made possible by a Federation Manager entity described in clause 6.2

When it comes to identifying the MEC systems, which are part of a MEC federation, prior to inter-MEC system communication for the needs of edge service consumption, or MEC app-to-app communication, the following categories (types) of use cases, from an application point of view, may be encountered:

TYPE-1 USE CASE

- The Client App at car #1 knows only its own App ID (i.e., “App Y”) and, eventually, the service ID to be consumed (or the MEC API, or, again, the service produced by another MEC App running in another MEC system).
    - In this case, a certain car, with Car ID#1 is unaware of (and potentially even uninterested in) the other cars’ IDs, but simply wants to be admitted to a pool/cluster of cars using a specific App ID (or consuming a certain service with a given ID).
    - A first example is the one of an Intersection Movement Assistant (IMA), provided by a Smart City (or a software company realizing the use case for the urban administration), where different cars have the App Y installed, and the corresponding MEC Apps are instantiated at different MEC systems. It should be noted that this is the most general case.
    - Another example is the one of In-Vehicle Entertainment (IVE), which can consist in a generic video streaming service, that car #1 wants simply to consume, without knowing actually which other cars are consuming it.
    - Another example is the one of software/ firmware over-the-air (SOTA/ FOTA) updates.
- In all these type-1 use cases, the MEC systems hosting the MEC App corresponding to other cars in the pool are not necessarily known.

TYPE-2 USE CASE

- The Client App at car #1 (with its MEC App instantiated in MEC system #1) knows also the ID of a car #2 (with its MEC App instantiated in MEC system #2) - target peer for communication.
    - As a first example, car #1 wants to communicate expressly with a car #2, since, perhaps they belong to drivers who are friends travelling together (in a sort of platooning), or belonging to a “social network” of cars consuming a certain V2X service, and thus knowing by definition their respective IDs. The only information known at car #1 is the car ID#2 (i.e. UE#2). As a result, the MEC system hosting the MEC App corresponding to car #2 is not necessarily known.
    - Another example: See-through among cars belonging to different MEC systems. After an initial phase of neighbor discovery (e.g. via PC5), the car #1 can get a list of other cars (and their IDs) that could provide the see-through service (i.e., offering their front cameras as a view for car#1). Then, there is a need of establishing an on-demand communication between two cars belonging to different MEC systems. In this case, we suppose that, after a preliminary phase (thanks to a Federation Manager), the MEO #1 correctly identifies the MEC system #2, in relation to car #2 application activity.
- Thus, in type-2 use cases, MEO #1 wants to discover the target MEO which is hosting the MEC App corresponding to car #2 (based on the ID of car #2). We, thus, suppose that in this preliminary MEC system discovery phase, made possible by the Federation Manager (with the catalog of MEC systems involved in the federation), the MEO #1 correctly identifies the MEC system #2, in relation to car #2 application activity. Consequently, after this phase, MEO #1 and MEO #2 can directly communicate.

TYPE-3 USE CASE

- The Client App at car #1 (with reference to MEC system #1) knows the ID of a Car #2 (target peer for communication), together with the target MEC system #2, in advance.
  - Example can be any of the previous use cases, where the information about some of the other MEC systems is known in advance, e.g. because of the presence of an "aggregator" between few operators (not necessarily all operators in the federation).
- In this case, it is reasonable that also the target MEO #2 could be known, but, for sake of generality, the other MEOs in the federation are not known. Thus, still the role of the Federation Manager is needed to ensure interoperability and generality (i.e., guarantee a standard approach to MEC federation independent from the particular deployment / agreement among some operators).

6.3.2 Solution proposal #3-1 – Federation Manager interactions

In all occurrences of cases, after a service communication query is issued by a MEC App instantiated at MEC system #1, the MEO #1 contacts the Federation Manager, as a very first step, before starting the communication with other MEOs (known or not).

NOTE: It should be noted that formation of a MEC federation is performed once, whereas, identification (or, look-up) of MEC systems being part of a MEC federation is performed per service communication query.

For this reason, in the context of the present key issue, the first phase of the communication between MEC systems is made possible with the addition of a new federation management reference point Mfm-fed (between the MEO and the Federation Manager), as appears in Figure 6.3.2-1.

And, the role of the federation manager described in the clause 6.2 can be supported by combining Mff-fed and Mfm-fed references. Each MEO shares relevant MEC system information to the federation manager via Mfm-fed, and these shared information can be exchanged to other federation managers via Mff-fed. The information may include MEO ID, which supports direct MEO-to-MEO communication in clause 6.4.2.

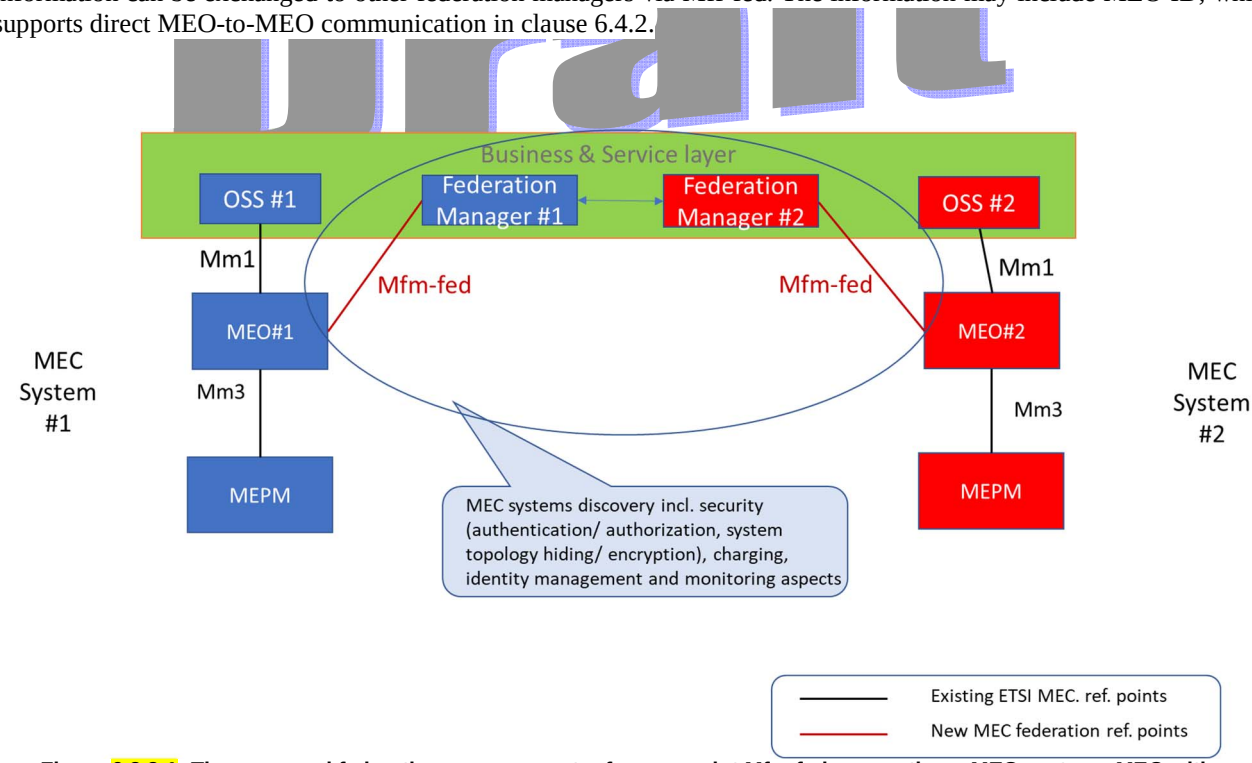


Figure 6.3.2-1: The proposed federation management reference point Mfm-fed connecting a MEC systems MEO with a Federation Manager. In this implementation variant we consider a Federation Manager per each MEC system.

6.4 Gap/Key issue #4 – MEC platform discovery

6.4.1 Description

We consider typical MEC federation scenarios, as described in Clause 5.1.

As described in Clause 5.1, to form a MEC federation, the following inter-MEC system communication level should be introduced after MEC system discovery to allow interworking between MEC systems:

- MEC platform discovery, by means of the MEC systems exchanging information about their MEC platforms, i.e., their identities, a list of their shared services, as well as authorization and access policies.

The ultimate goal is to address the needs of information exchange, for the needs of MEC/edge service consumption. Such information exchange refers to either a MEC application in need of consuming a MEC platform service, or a MEC application in need of communicating with other (e.g., service-producing) MEC applications.

> The present gap/key issue analyzes the case of MEC platform discovery. For this key issue, the assumption is that a preliminary phase is handling the MEC system (i.e., below business level) discovery, including security (authentication/ authorization, system topology hiding/ encryption), charging, identity management and monitoring aspects as an essential prerequisite for MEC federation.

In the following, a solution is proposed, to address the subsequent step, i.e., MEC platform discovery.

6.4.2 Solution proposal #4-1 – MEC platform discovery via direct MEO-to-MEO interactions

As mentioned in clause 5.1.2, MEC platform discovery is one of the key requirements to enable MEC federation, derived from the generic requirement contained in the ETSI GS MEC 003 [i.2]:

"A MEC platform should be able to discover other MEC platforms that may belong to different MEC systems;"

This "MEC platform discovery" phase is made possible by a communication between MEOs, which are aware of their MEC system topologies and all information about the MEC platforms in their respective systems. However, taking into account that, in general, MNOs would not be eager to share details of the internal structure of their managed MEC systems to other MNOs, only information essential to the subsequent information exchange for the needs of e.g., MEC service consumption would need to be exchanged. Consequently, as part of MEC platform discovery, the MEOs exchange information about their MEC platforms (i.e., their identities), and their capabilities, i.e., a list of their shared services, as well as authorization and access policies.

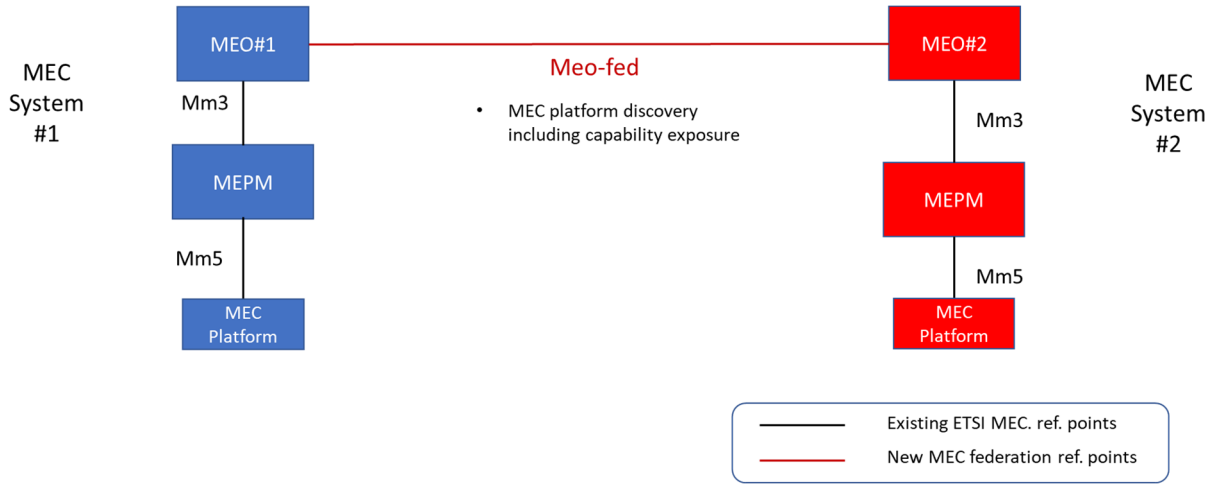


Figure 6.4.2-1: The role of the Meo-fed reference point connecting configured MEOs is to enable inter-MEC system platform discovery including capability exposure.

This solution is technically feasible through a new Meo-fed reference point connecting MEOs as configured, as introduced in Figure 6.4.2-1.

NOTE: Inter-MEC system platform discovery (including capability exposure) with the involvement of the MEOs may be especially applicable to scenarios involving MEC systems consisting of a large number of MEC hosts.

6.4.3 Solution proposal #4-2 – MEC platform discovery involving Federation Manager modules

As shown in the architecture proposals illustrated in Figures 6.3.2.-1, Federation Manager modules are in charge of the communication among MEC Systems are responsible of providing the list of functionalities for the needs of MEC platform discovery. In this solution, exchange of the relevant MEC system information should be supported via Mfm-fed and Mff-fed reference as depicted in Figure 6.4.3-1.

Editor's Note: the nature of MEC system information is FFS and should be aligned with OPG direction

Because the federation manager performs exposure of the catalog of MEC systems, each MEO shares relevant information with each federation manager via Mfm-fed. After that, when the MEC system #1 send a request (e.g., application instantiation) to MEC systems #2 via Mff-fed, the federation manager #2 chooses an appropriate MEO and forwards the request to MEO#2. MEO #2 finds the appropriate MEC platform and sends back information about the MEC platform to the MEC systems #1 via Mfm-fed and Mff-fed.

The overall purpose of this solution is same as the solution #4-1, but Mfm-fed and Mff-fed are used for MEC platform discovery instead of Meo-fed. This solution can be useful in that case multiple MEOs are connected to a single federation manager

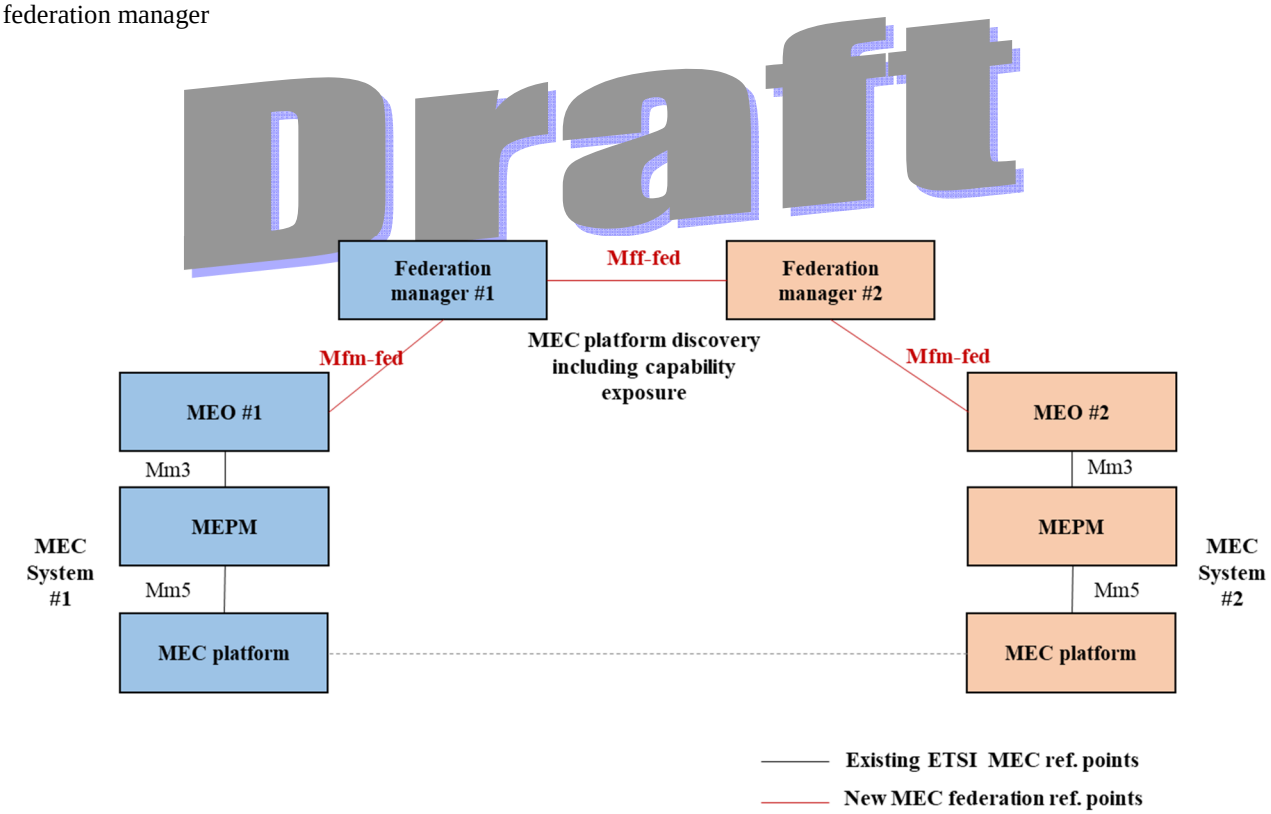


Figure 6.4.3-1: Inter-MEC platform discovery by using the federation manager modules

For instance, regarding exchanging a list of shared MEC services, the high-level information flow is illustrated in Fig. 6.4.3-2.

3) S-MEO sends a request to obtain the information of service availability via both federation managers of source and target MEC systems.
4) T-MEO replies with the information of MEC service availability via both federation managers.

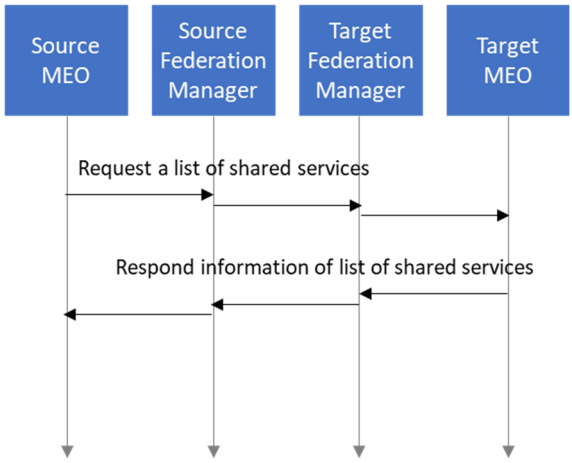


Figure 6.4.3-2

6.5 Gap/Key issue #5 – Information exchange for MEC service consumption or for MEC app-to-app communication

6.5.1 Description

We consider typical MEC federation scenarios of V2X services (i.e. multi-MNO, multi-OEM, multi-MEC), as described in Clause 5.1.

As described in Clause 5.1, as part of the operation of a MEC federation, the following inter-MEC system communication level is introduced after MEC system discovery and MEC platform discovery:

- Information exchange at MEC platform or higher level, for the needs of MEC service consumption, or for MEC app-to-app communication.

Such information exchange refers to either a MEC application in need of consuming a MEC platform service, or to a MEC application in need of communicating with other (e.g., service-producing) MEC applications. The present gap/key issue analyzes the case of information exchange.

For this key issue, the assumption is that a preliminary phase is handling the following steps:

- MEC system (i.e., below business level) discovery, including security (authentication/ authorization, system topology hiding/ encryption), charging, identity management and monitoring aspects as an essential prerequisite to form a MEC federation;
- MEC platform discovery, by means of the MEC systems exchanging information about their MEC platforms, i.e., their identities, a list of their shared services, as well as authorization and access policies.

Current definitions in MEC are only enabling edge service consumption within a single MEC system. In a single MEC system, the most general case corresponds to a MEC app running on a MEC host, which needs to consume MEC services instantiated on a MEC host (within the MEC system). The queried services are assumed available in the MEC system, however according to ETSI MEC specifications they may run at different localities. In Figure 6.5.1-1, the three general cases of edge services consumption are depicted, where it is worth noticing that, for both remote service consumption cases (i.e., the one of a MEC app consuming a remote -i.e., not instantiated at the same MEC host- MEC platform service and the one of a

MEC app consuming a remote service produced by another MEC app), the Mp3 reference point is involved that connects different MEC platforms of the same MEC system.

In the following, two solutions are proposed to address the key issue of inter-MEC system information exchange for the needs of MEC service consumption, or for MEC app-to-app communication. The aim of the solution proposals is to close the gap of having no reference points defined for such information exchange.

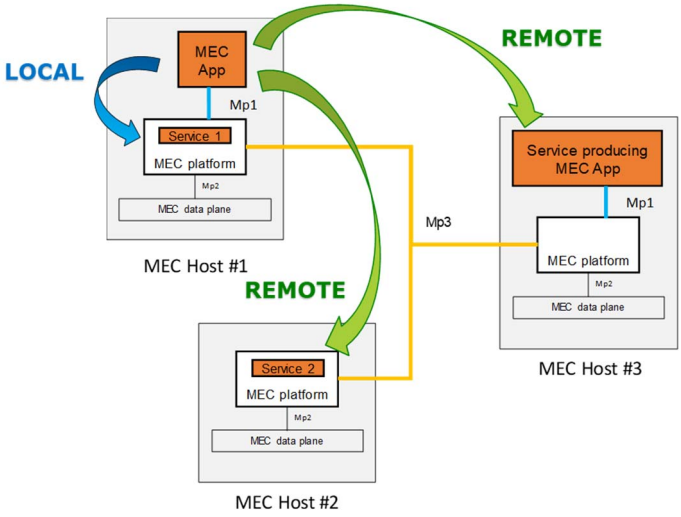


Figure 6.5.1-1: Edge service consumption options within a single MEC system.

6.5.2 Solution proposal #5-1 – overall solution addressable to key issues #1-2-3-4-5 involving information exchange at MEC platform level

Let us consider a MEC federation scenario, that involves multiple MEC systems, belonging to different (technical and/or administrative) domains. In the most general case, MEC hosts belong to different MEC systems (i.e., provided by different MEC vendors), potentially running on different MNOs networks, or in different domains.

In this context, a MEC application can consume MEC services available by other MEC hosts, belonging to other MEC systems, by defining in MEC a new "federated MEC" Mpp-fed reference point connecting inter-system MEC platforms and, hence, allowing edge service consumption in MEC federation scenarios.

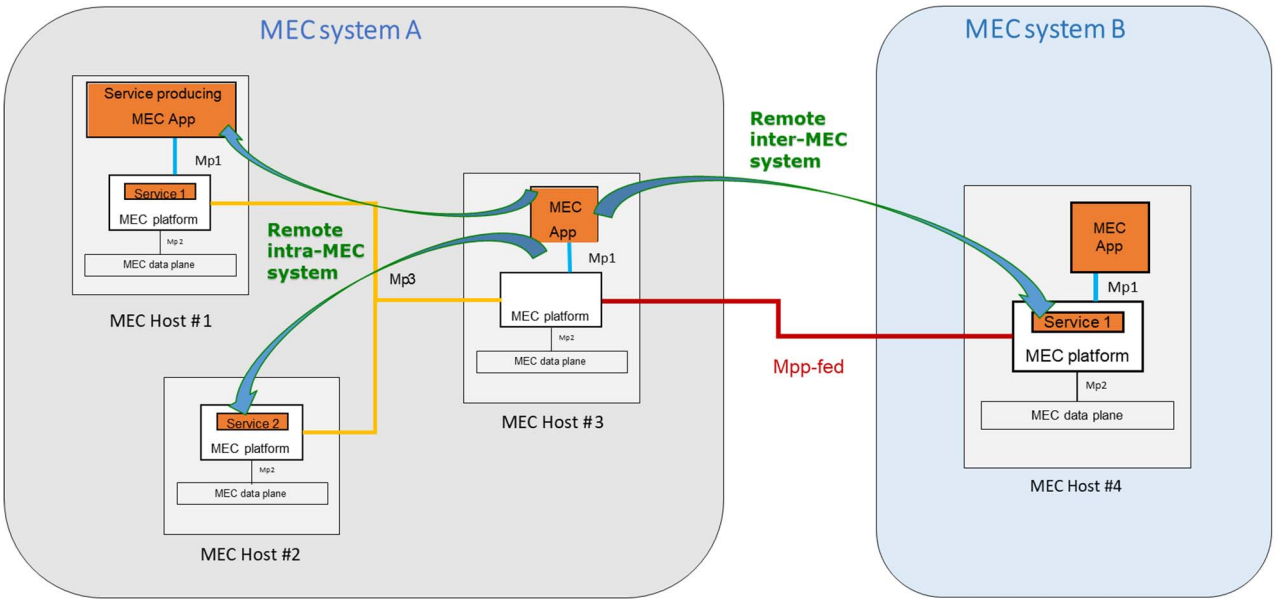


Figure 6.5.2-1: MEC federation scenario enabling edge service consumption across MEC systems.

Figures 6.5.2-1 and 6.5.2-2 clarify upon how service consumption is defined, in the context of a MEC federation, with the addition of the new Mpp-fed reference point. Another alternative is to enhance the definition of the Mp3 reference point, by enriching it with signaling capability among MEC platforms belonging to different MEC systems.

Both options are possible, and even both can be standardized, i.e., leaving as optional the choice of implementers to add a Mpp-fed reference point, or to implement an enhanced Mp3 reference point in their system. Nevertheless, it is worth noticing that a new Mpp-fed reference point should be defined only for MEC federation communication (i.e., only connecting MEC platforms belonging to different MEC systems).

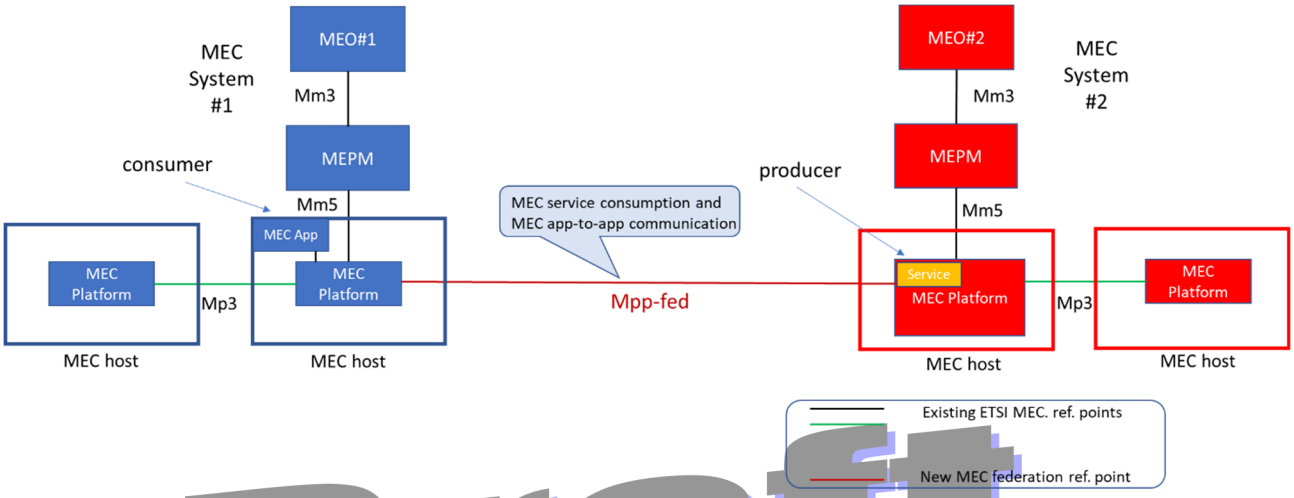


Figure 6.5.2-2: The role of the proposed Mpp-fed reference point (connecting two MEC platforms belonging to two MEC systems of a MEC federation) is to enable MEC service consumption and MEC app-to-app communication.

Given the above proposal to define a proper reference point that may support information exchange at MEC platform level, for the needs of MEC service consumption, or for MEC app-to-app communication, the whole communication framework composed of MEC system discovery including security (authentication/ authorization, system topology hiding/ encryption), charging, identity management and monitoring aspects along with MEC platform discovery (therefore, addressing Key Issue #1 – Key Issue #4) is covered by a hierarchical communication approach. The signaling sequence that follows this approach, focusing, as an example, on Type-1 use cases, as described in clause 6.3.1 (i.e., the ones, where car #1 only knows the service & application IDs to be consumed/ communicate with), is illustrated in Figure 6.5.2-3:

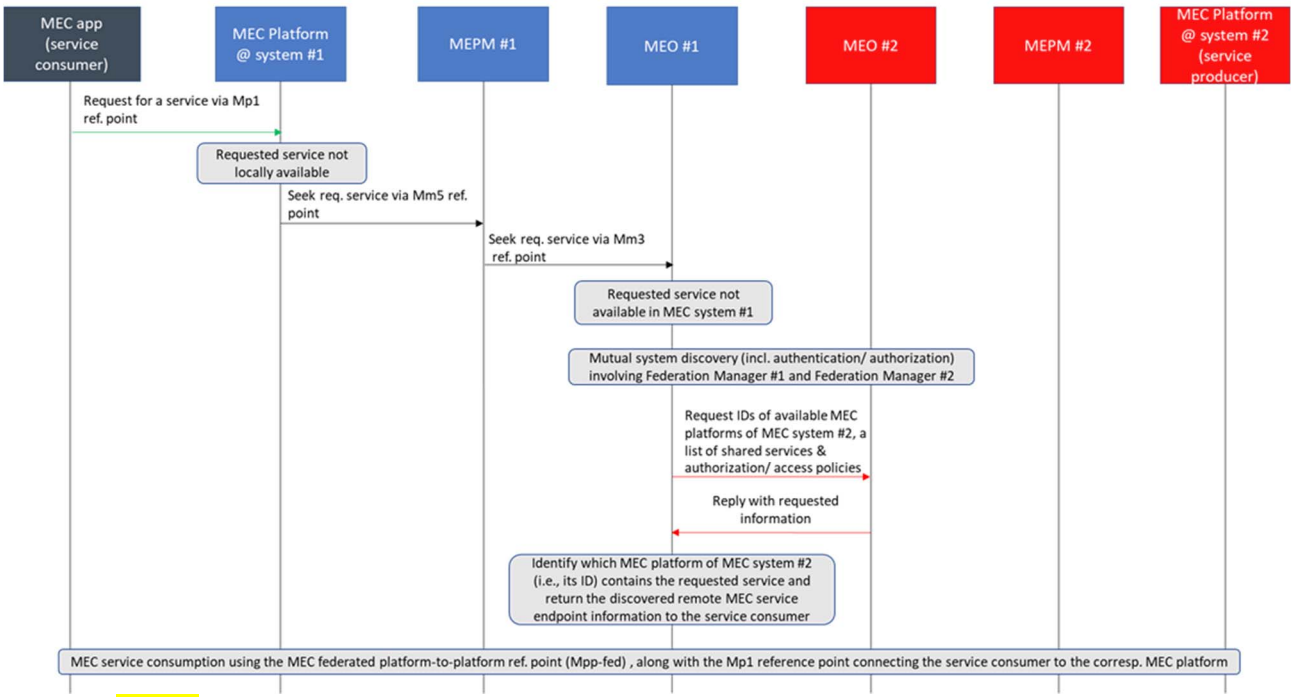


Figure 6.5.2-3: Sequence diagram explaining the involved signaling to establish hierarchical inter-MEC system communication for the needs of service consumption. A Federation Manager per system (and, possibly, per operator) is assumed; Federation Managers #1 and #2 are assumed already discovered.

In terms of signaling, the exchanged messages are the following:

1. The service consumer (i.e., a MEC application instantiated in MEC system #1) requests a needed service via the Mp1 reference point by means of its ID.
2. The respective MEC platform in MEC system #1 finds that the requested service is not locally available and forwards the service request to the MEC Platform Manager of MEC system #1 (MEPM #1).
3. MEPM #1, in its turn, forwards the service request to MEO #1.
4. MEO #1, which has an overview of the topology and available services of MEC system #1 finds that the requested service is not available across MEC system #1. This triggers the need for out-of-system service consumption; to accomplish that, MEC system discovery is performed as a first step of forming a new (or, joining an already established) MEC federation (i.e., the Federation Manager of MEC system #1, following a request by MEO #1 via the Mfm-fed ref. point informs MEO #1 of the MEO #2 ID).
5. Mutual discovery of MEC systems #1 and #2, including security (authentication/ authorization, system topology hiding/ encryption), charging, identity management and monitoring aspects is performed by the two corresponding Federation Managers (or a common Federation Manager).
6. After MEC system discovery, MEO #1 knows the ID of MEO #2 and communicates with MEO #2 via the Meo-fed reference point, requesting the IDs of the available MEC platforms of MEC system #2, a list of their shared services, as well as authorization and access policies.
7. MEO #2 replies with the requested information.
8. MEO #1 identifies which MEC platform of MEC system #2 (i.e., its ID) contains the service requested by the service consumer, i.e., the MEC App instantiated at MEC system #1 and returns the discovered remote MEC service endpoint information to the service consumer.
9. MEC service consumption is carried out using the MEC federated platform-to-platform reference point (Mpp-fed), along with the Mp1 reference point connecting the service consumer with its corresponding MEC platform of MEC system #1.

NOTE: It is noteworthy that the procedure depicted in Figure 6.5.2-3 concerns the case where MEC system #1 and MEC system #2 are, after the needed signaling, part of the same MEC federation (i.e., business agreement), but the UE Apps installed in the cars (in general belonging to different MEC systems) are not necessarily aware of this federation. Thus, any upcoming service requests that cannot be satisfied within MEC system #1 will be forwarded to the corresponding MEO which will identify whether the sought service is available anywhere in the MEC federation (e.g., other MEC system #2).

The overall set of the proposed new MEC federation reference points is depicted in Figures 6.5.2-4, 6.5.2-5 and 6.5.2-6 for all three cases of having: i) multiple directly interacting Federation Managers via a dedicated Mff-fed reference point; ii) a single, overall Federation Manager, or, iii) a Federation Broker communicating with each Federation Manager via a dedicated Mfb-fed reference point, respectively.

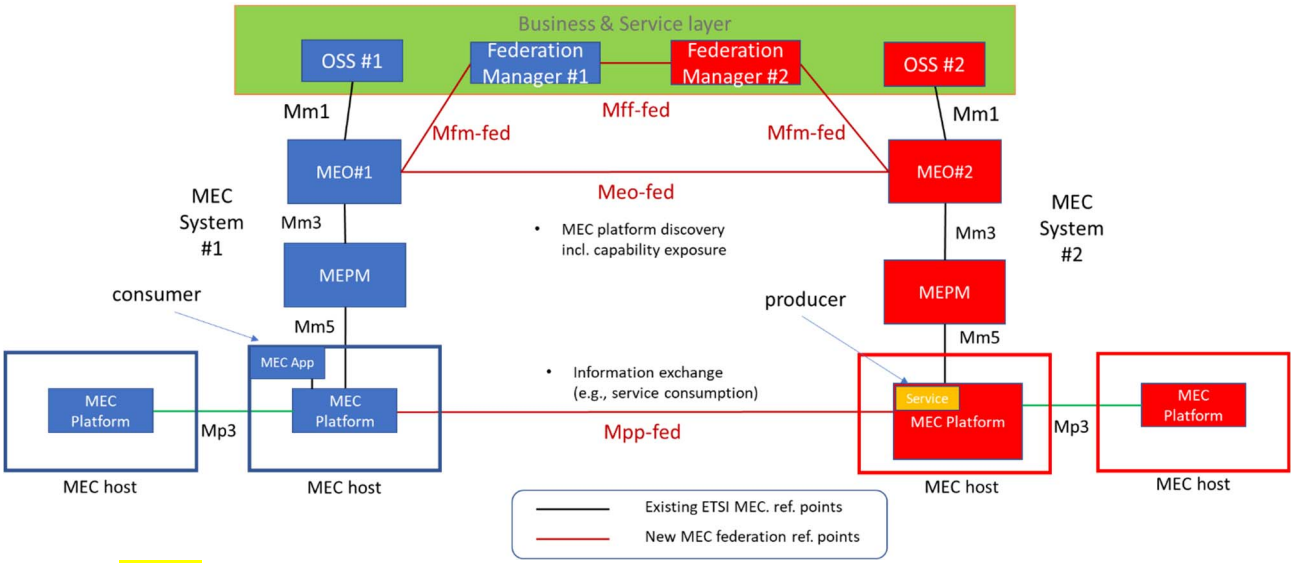


Figure 6.5.2-4: All proposed MEC federation reference points assuming a Federation Manager per MEC system.

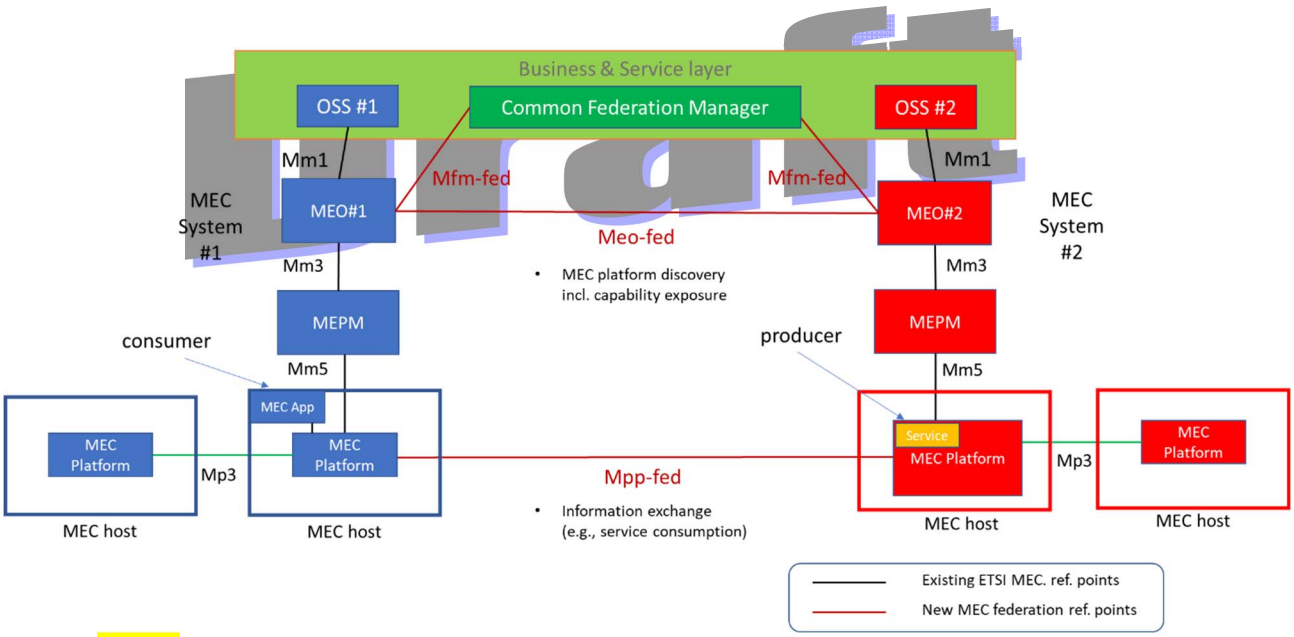


Figure 6.5.2-5: All proposed MEC federation reference points assuming a single Federation Manager, the scope of which is the whole MEC federation.

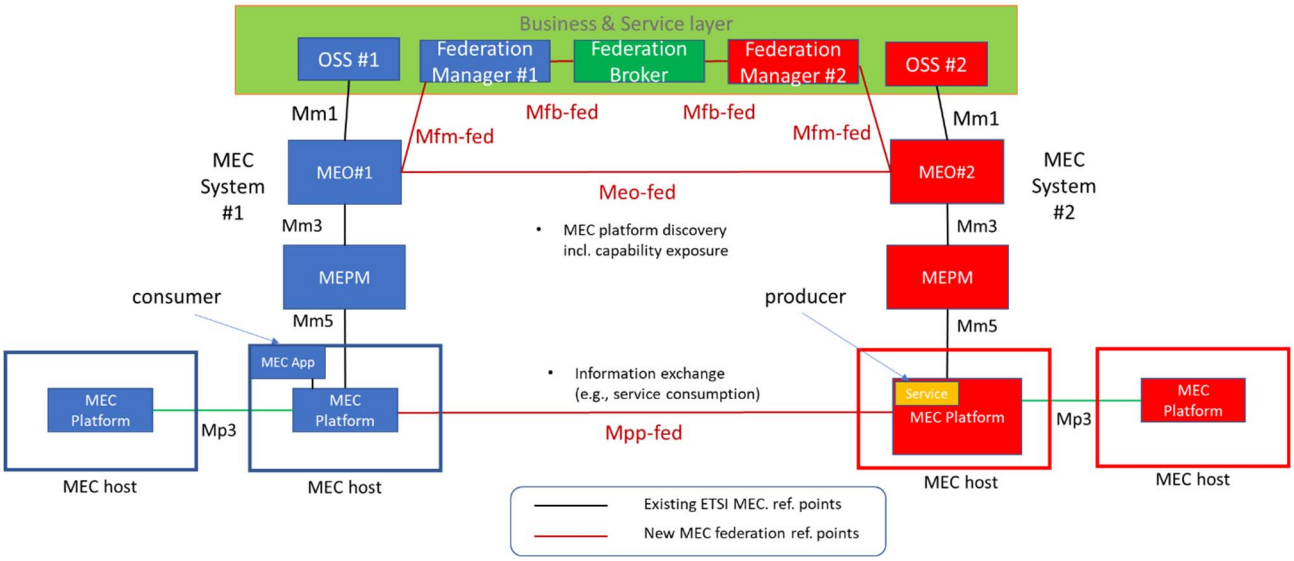


Figure 6.5.2-6: All proposed MEC federation reference points assuming the existence of a Federation Broker communicating with multiple Federation Managers.

6.5.3 Solution proposal #5-2 – overall solution addressable to key issues #1-2-3-4-5 involving information exchange at MEC federation management level

This proposed overall solution is working along with solutions #3-1 and # 4-2, as described in clauses 6.3.2 and 6.4.3, respectively. To facilitate inter-MEC system information exchange towards MEC service consumption or MEC app-to-app communication, the processes for discovering another MEC system and its MEC platforms are handled via the MEC federation entities. All control signals between MEC systems for the needs to establish a MEC federation are exchanged via the federation management entities to avoid direct MEC host-level communication. The detailed process is illustrated in Fig 6.5.3-1.

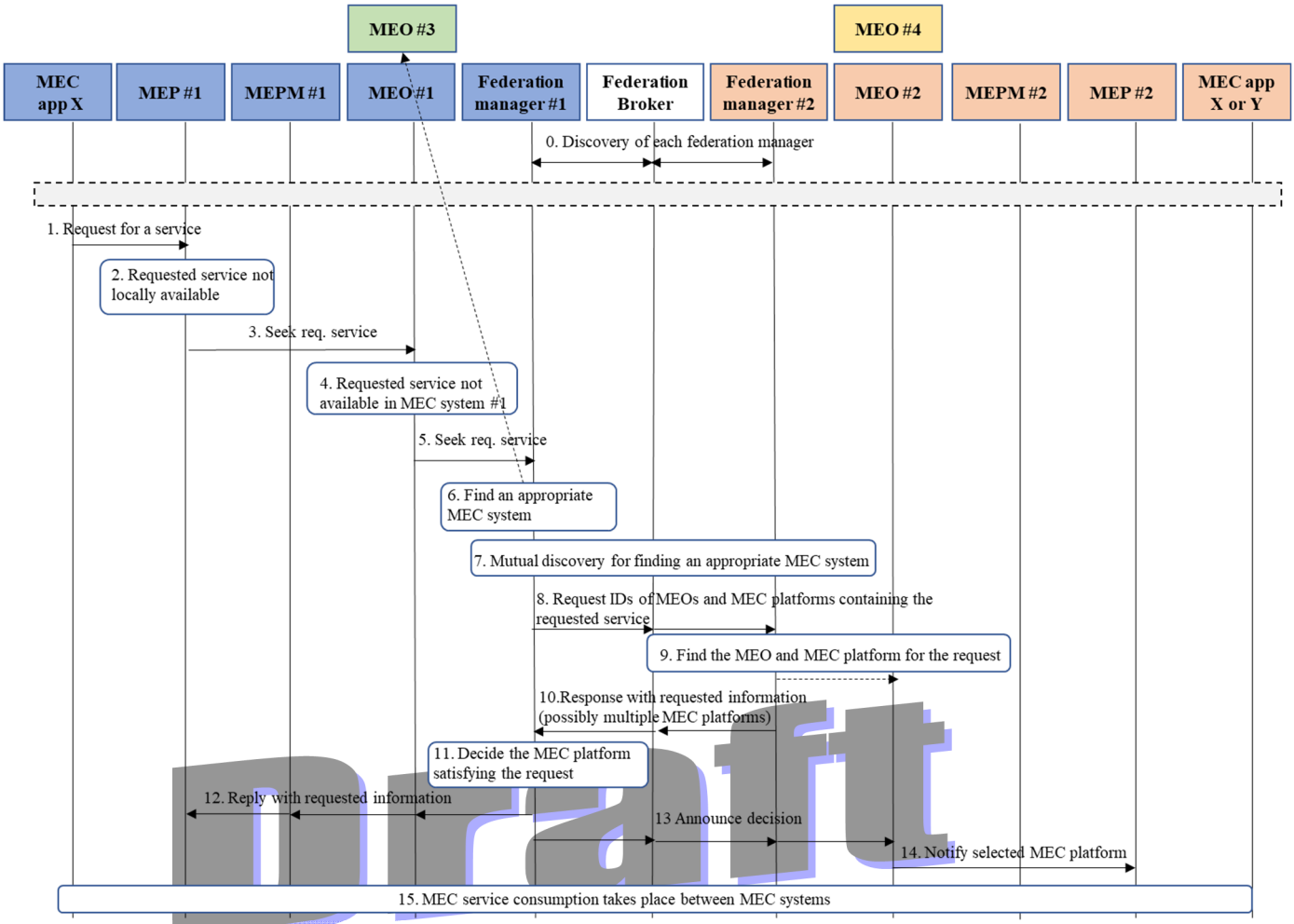


Figure 6.5.3-1 Sequence diagram explaining the involved signaling via MEC federation management reference points to establish inter-MEC system communication for the needs of service consumption

0. Discovering of each federation manager and each MEC system (e.g., charging, monitoring, etc) described in clause 6.2 and 6.3 can be performed before the request of step 1.

1-4. Steps 1-4 follow same procedures as described in the solution #5-1. When sending a request, the service consumer can include some information that can be helpful to maintain service quality. For example, to support the use case #5, the ID of MEC application Y can be included as well.

5. If MEO #1 cannot find the requested service within MEC system #1, it sends a request to the respective federation manager (i.e., federation manager #1) to find the requested service in other MEC systems.

6. Federation manager #1 can discover other MEC systems that are already federated with it (e.g., MEC system #3), before trying to connect with other federation managers.

7. Step 7 is aligned with the solution #3-1 of clause 6.3, but if step 0 is already performed, step 7 is not required.

8. After MEC system discovery, the federation manager #1 sends a request to other federation managers via the Mff-fed reference point to obtain the IDs of the available MEC platforms of the discovered MEC systems containing the requested MEC service, as well as authorization and access policies. If there are several federation managers, its request can be delivered through a federation broker involving the Mfb-fed reference point. Step 8 is aligned with the solution #4-2 of clause 6-3.

9. The federation manager #2 identifies which MEC system contains the requested service (e.g., MEC system #2) and also which MEC platform of MEC system #2 (i.e., its ID) contains the service requested by the service consumer.

If the federation manager #2 does not have enough information to identify the requested service, it sends the request to connected MEOs to identify an appropriate MEC Platform. In this case, MEO #2 can identify which MEC Platform can be applicable and sends back to the federation manager #2 with the ID of MEC Platform containing the requested MEC service.

10. The federation manager #2 sends the response with ID(s) of MEC platform(s) containing the requested MEC service to the federation manager #1.

11. The federation manager #1 can decide which of the identified MEC platforms in step 10 can satisfy the request based on its policy, if there are several MEC systems that have responded.

12. The federation manager #1 sends the response with ID of MEC platform #2 to MEO #1 and MEP #1.

13. The federation manager #1 announces its decision to the selected federation manager #2.

14. The federation manager #2 notifies the indicated MEO #2 and MEC Platform #2 containing the requested MEC service.

15. MEC service consumption takes place between the MEC service consumer (MEC app X of MEC system #1) and the identified MEC platform of MEC system #2 containing the MEC service.

The overall set of the proposed new MEC federation reference points is depicted in Figures 6.5.3-2, 6.5.3-3 and 6.5.3-4. "Consumer" refers "MEC app instantiated in MEC Host #1" in Figure 6.5.1-1 and "producer" refers "Service producing MEC App in MEC Host #2 and #3" in Figure 6.5.1-1

.

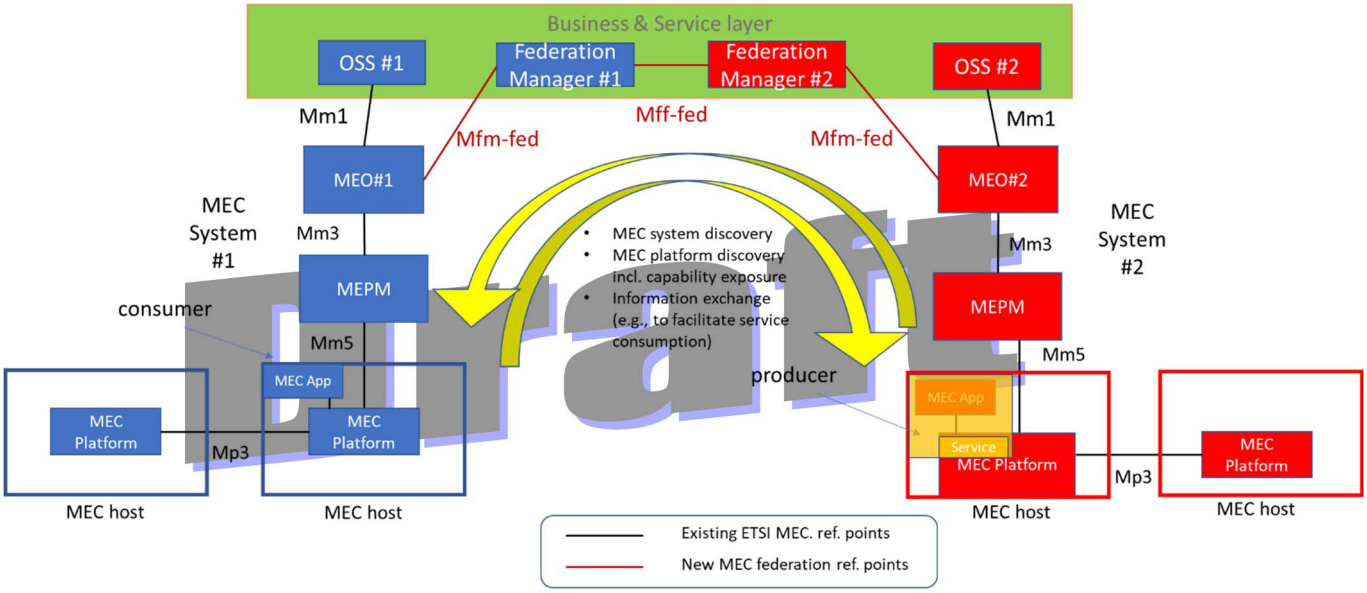


Figure 6.5.3-2: Conceptual diagram with all proposed MEC federation reference points assuming a Federation Manager per MEC system

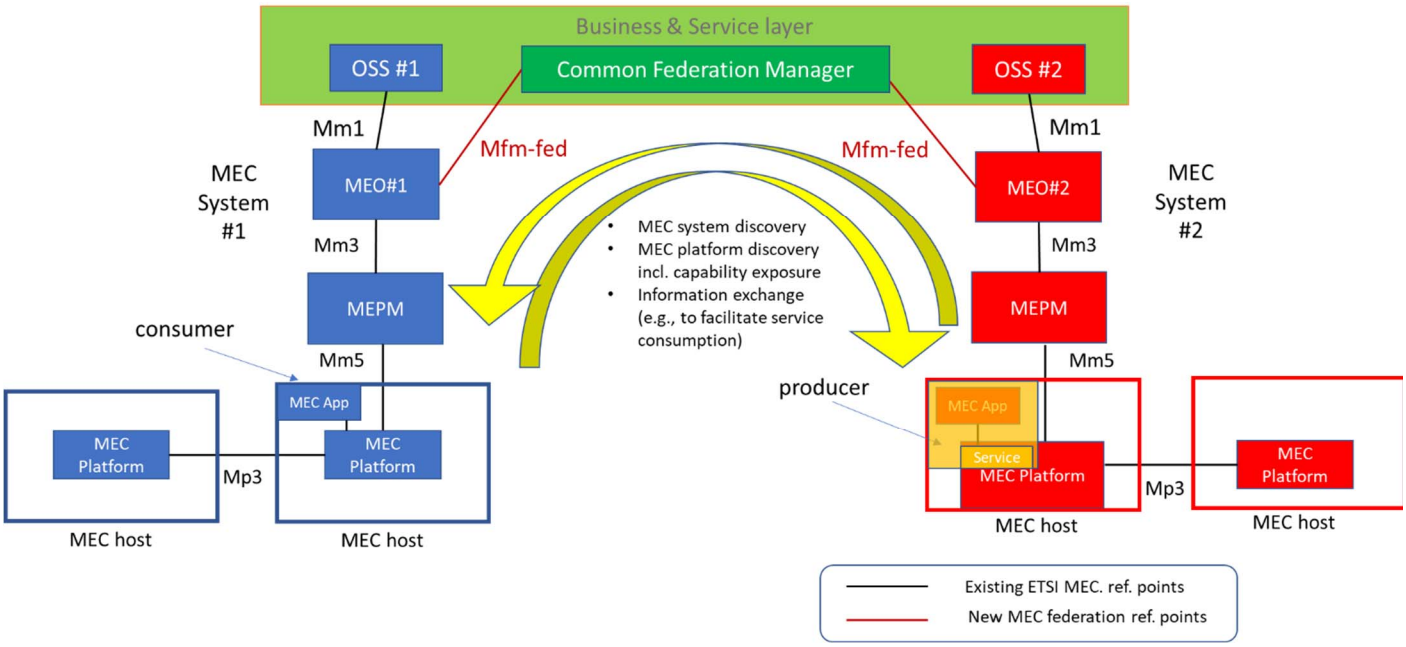


Figure 6.5.3-3: Conceptual diagram with all proposed MEC federation reference points assuming a single Federation Manager, the scope of which is the whole MEC federation.

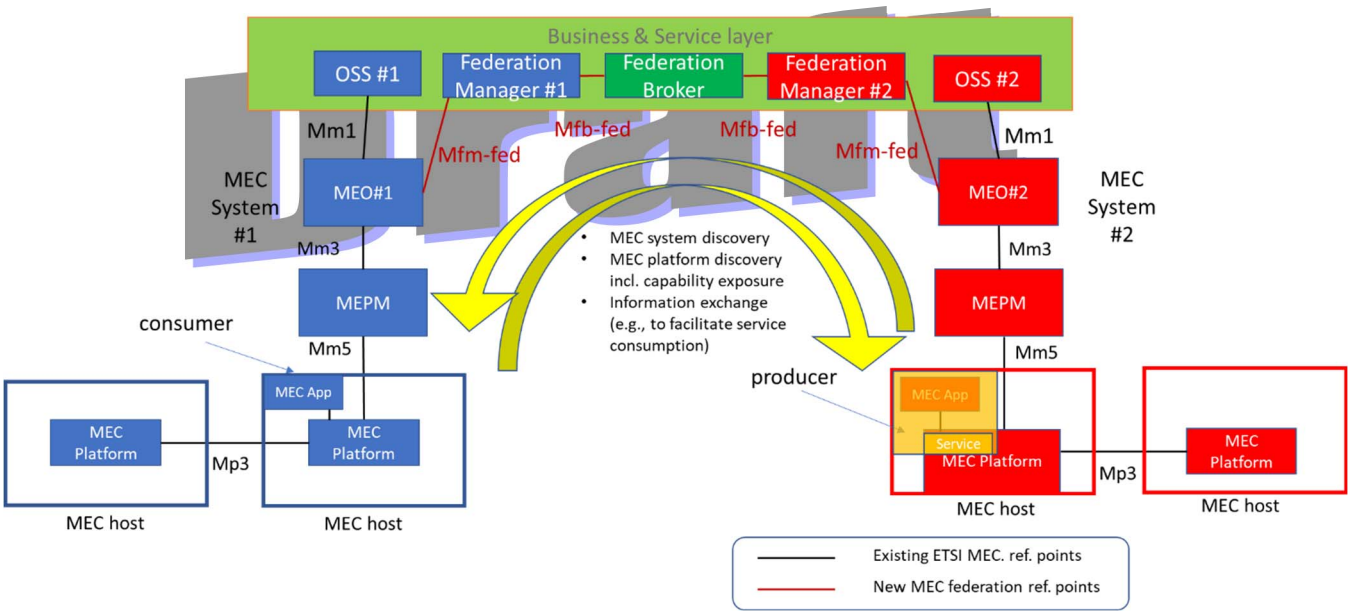


Figure 6.5.3-4: Conceptual diagram with all proposed MEC federation reference points assuming the existence of a Federation Broker communicating with multiple Federation Managers

6.6 Gap/Key issue #6 Way to request the instantiation of application on Cloud system

6.6.1 Description

As introduced in Use Case 5.3 MEC-Cloud coordination, the following recommendation should be solved.

- [Recommendation 5.3.2-6]
  The MEC system should support to request to instantiateor re-start application instance on the cloud system when transferred from MEC system to Cloud system.

In the case where the User Device goes out of the coverage of the MEC system while the device is communicating with the application on the MEC system, the server-side application is expected to be generated on a cloud system and the device expects to continue the application service by means of interaction between client application and server-side application instance on the cloud system. While OSS is responsible for receiving a request via Mx2 to instantiate the application instance on the MEC host, the way to send a request to instantiate or re-start application instance on the cloud system is not specified in MEC003, e.g., reference point and interface.

Since MEO is responsible for maintaining an overall view of the MEC system based on deployed MEC hosts, available MEC services, and topology, MEC should be a starting point to send a request for the instantiation.

6.6.2 Solution proposal #6-1 leveraging OSS

As described in the previous Clause, in the case of receiving a request to instantiate application instance on the MEC host, OSS receives the request and forward it to MEO. Then, MEO triggers instantiation process inside the MEC system. First option to realize to send a request to the cloud system is the reverse way. The high-level message flow is depicted in Fig. 6.6.2-1.

1) MEO decides to change the endpoint of the interaction from the application instance on the MEC host to the application on the cloud system.
2) MEO sends a request to OSS to instantiate or re-start the corresponding application instance.
3) OSS forwards the request to the external system, i.e., the Cloud system.

Note: the format of the request message is out of scope for the present document.

4) Cloud system instantiates or restarts an application instance.
5) Start interaction.

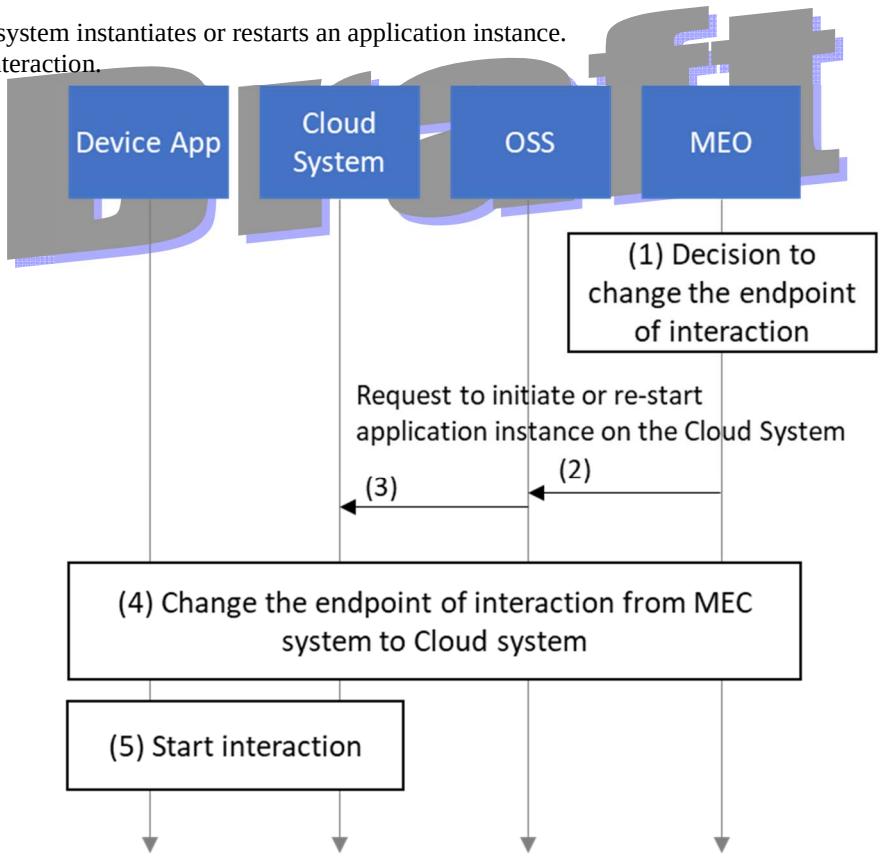


Figure 6.6.2-5 High-level information flow of sending request to instantiate or re-start the application instance via OSS.

6.6.3 Evaluation

The solution proposal #6-1 is technically feasible under the following conditions.

- OSS is capable of communicating with the Cloud system.

6.X Gap/Key issue #X

6.X.1 Description

6.X.2 Solution proposal #X-1

6.X.3 Solution proposal #X-2

7 Conclusion and recommendation

The present document has described various use cases in inter-MEC systems and MEC-Cloud systems coordination, and also has defined key issues and proposed potential solutions based on analysis the current ETSI MEC architecture.

The mapping of the key issues to their associated solutions is provided in table 7-1. This includes highlighting any identified gaps between the current scope of ETSI MEC. In a summary, new entities and reference points for MEC federation are required and it is needed to consider supporting signaling based on them. In addition, some entities (e.g., MEO and OSS) can be enhanced for MEC-Cloud coordination.

Table 7-1 Key issue and solution

Key issue	Clause #	Solution	Gap
#1 Structuring the needed signalling for secure communication among different MEC systems	6,1	Solution #1	Yes Entities for MEC federation are not present in ETSI GS MEC 003 [i.2].
#2 Considering entities for MEC federation	6.2	Solution proposal #1 Federation Manager	Yes Entities and reference points for MEC federation are not present in ETSI GS MEC 003 [i.2].
		Solution proposal #2 Federation Broker	Yes Entities and reference points for MEC federation are not present in ETSI GS MEC 003[i.2].
#3 MEC system discovery	6.3	Solution proposal #3-1 – Federation Manager interactions	Yes Entities and reference points for MEC federation are not present in ETSI GS MEC 003[i.2].
#4 MEC platform discovery	6.4	Solution proposal #4-1 – MEC platform discovery via direct MEO-to-MEO interactions	Yes Entities and reference points for MEC federation are not present in ETSI GS MEC 003[i.2].
		Solution proposal #4-2 – MEC platform discovery	Yes

		involving Federation Manager modules	Entities and reference points for MEC federation are not present in ETSI GS MEC 003[i.2].
#5 Information exchange for MEC service consumption or for MEC app-to-app communication	6.5	Solution proposal #5-1 – overall solution addressable to key issues #1-2-3-4-5 involving information exchange at MEC platform level	Yes Entities and reference points for MEC federation are not present in ETSI GS MEC 003[i.2].
		Solution proposal #5-2 – overall solution addressable to key issues #1-2-3-4-5 involving information exchange at MEC federation management level	Yes Entities and reference points for MEC federation are not present in ETSI GS MEC 003[i.2].
#6 Way to request the instantiation of application on Cloud system	6.6	Solution proposal #6-1 leveraging OSS	Yes The functionalities of OSS and MEO need to be updated.

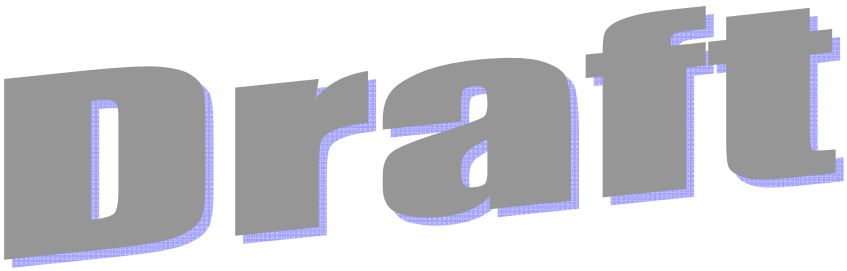
Even though this analysis has been performed carefully, there is the possibility that during the normative work additional gaps and aspects that require resolution may be discovered.

Furthermore, GSMA Operator Platform WG has been defining requirements about MEC federation concepts, APIs, mechanisms and associated procedures between operator systems, which are similar to the scope of this present document. It is worth considering to coordinate with GSMA to avoid market fragmentation, ensuring the end-to-end follow-up of the use cases related to federation among these SDOs.

Taking into account of the gap analysis provided in table 7-1 and the external environment, it is therefore recommended the following topics need to be addressed in normative follow-up work in ETSI MEC:

- to add new requirements and related use cases that currently are not covered in ETSI GS MEC 002 [i.10].
- to include new entities and reference points for MEC federation in ETSI GS MEC 003 [i.2], and to define new APIs and data models enabling MEC federation.
- to update existing entities and functionalities in ETSI GS MEC 003[i.2]. to collaborate with other organizations (i.e. GSMA, 5GAA, etc.) that have similar approaches for aligning their requirement and complementing each other.

Annex A:
Title of annex



Annex B:
Title of annex

B.1 First clause of the annex

B.1.1 First subdivided clause of the annex

Annex: Bibliography

-

Annex :
Change History

Date	Version	Information about changes
<Month year>	<#>	<Changes made are listed in this cell>

History

Document history		
<Version>	<Date>	<Milestone>
0.0.1	2020-02	Initial version of GR
2.0.2	2020-03	Implements documents MEC (20)000066 and MEC (20)000067. Fix an error on the number of version of this document.
2.0.3	2020-06	Implements documents MEC(20)000186r1, MEC(20)000191r1, and MEC(20)000188r1.
2.0.4	2020-07	Implements documents MEC(20)000187r5, MEC(20)000189r5, and MEC(20)000237r1.
2.0.5	2020-07	Implements documents MEC(20)000178r2, MEC(20)000179r2, MEC(20)000190r3, MEC(20)000247r2, and MEC(20)000249r1. Fix editorial errors on normative expression, format of itemization, and format of figure and caption.
2.0.6	2020-08	Replaces content of MEC(20)000190r3 with that of MEC(20)000190r4.
2.0.7	2020-09	Implements a document MEC(20)000275r2.
2.0.8	2020-10	Implements documents MEC(20)000268, MEC(20)000301, MEC(20)000302r3, MEC(20)000303r1,MEC(20)000300r2, MEC(20)000323r2, and MEC(20)000324r3. Fixes editorial errors on the number of figure and number of Clause.
2.0.9	2020-10	Implements documents MEC(20)000360, MEC(20)000361, MEC(20)000362, and MEC(20)000363r1.
2.0.10	2020-11	Implements documents MEC(20)000369, MEC(20)000368r1, and MEC(20)000376r2. Fixes several typos. Changes format of figures' title.
2.0.11	2020-12	Implements documents MEC(20)000383r4, MEC(20)000412r1, MEC(20)000423, MEC(20)000426, and MEC(20)000439r2.

Latest changes made on2020-12-15.