ETSI GR MEC-DEC 025 V2.1.1 (2019-06)

Multi-access Edge Computing (MEC); MEC Testing Framework

Disclaimer

The present document has been produced and approved by the Multi-access Edge Computing (MEC) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG.
It does not necessarily represent the views of the entire ETSI membership.

Reference
DGR/MEC-DEC25TestingFramework

Keywords
MEC, testing

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

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 (ISG) Multi-access Edge Computing (MEC).

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.

1 Scope

The present document lists the functionalities and capabilities required by a MEC compliant implementation. In addition, the present document specifies a testing framework defining a methodology for development of interoperability and/or conformance test strategies, test systems and the resulting test specifications for MEC standards. In additional, the testable requirements are listed and prioritized.

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 012: "Multi-access Edge Computing (MEC); Radio Network Information API".

[i.2] ETSI GS NFV-TST 002: "Network Functions Virtualisation (NFV); Testing Methodology; Report on NFV Interoperability Testing Methodology".

[i.3] ETSI GS MEC 003: "Multi-access Edge Computing (MEC); Framework and Reference Architecture".

[i.4] ISO/IEC 9646-7:1995: "Information technology -- Open Systems Interconnection -- Conformance testing methodology and framework -- Part 7: Implementation Conformance Statements".

NOTE: Available at https://www.iso.org/standard/3084.html.

[i.5] ISO/IEC 9646-1:1994: "Information technology -- Open Systems Interconnection -- Conformance testing methodology and framework -- Part 1: General concepts".

NOTE: Available at https://www.iso.org/standard/17473.html.

[i.6] TTCN-3 abstract test language.

NOTE: Available at http://www.ttcn-3.org/index.php/downloads/standards.

[i.7] ETSI GS MEC 002: "Multi-access Edge Computing (MEC); Phase 2: Use Cases and Requirements".

[i.8] ETSI GS MEC 010-1: "Mobile Edge Computing (MEC); Mobile Edge Management; Part 1: System, host and platform management".

[i.9] ETSI GS MEC 010-2: "Multi-access Edge Computing (MEC); MEC Management; Part 2: Application lifecycle, rules and requirements management".

[i.10] ETSI GS MEC 011: "Multi-access Edge Computing (MEC); Edge Platform Application Enablement".

[i.11] ETSI GS MEC 013: "Multi-access Edge Computing (MEC); Location API".

[i.12] ETSI GS MEC 014: "Mobile Edge Computing (MEC); UE Identity API".

[i.13] ETSI GS MEC 015: "Mobile Edge Computing (MEC); Bandwidth Management API".

[i.14] ETSI GS MEC 016: "Multi-access Edge Computing (MEC); UE application interface".

[i.15] ETSI Test Description Language.

NOTE: Available at https://tdl.etsi.org/index.php/downloads.

[i.16] ETSI GS MEC 001: "Multi-access Edge Computing (MEC); Terminology".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI GS MEC 001 [i.16] and the following apply:

certification/compliance assessment: major goal of a compliance assessment is to ensure the interoperability of implementations, and the conformance of implementations to the standard

conformance testing: purpose of conformance testing is to determine to what extent a single implementation of a particular standard conforms to the individual requirements of that standard

interoperability testing: purpose of interoperability testing is to prove that end-to-end functionality between (at least) two communicating systems is as required by the standard(s) on which those systems are based

Test Case (TC): complete and independent specification of the actions required to achieve a specific Test Purpose

NOTE: TCs are written in testing languages, e.g. TTCN-3.

Test Descriptions (TD): specify the sequence of actions required to realize the verdict identified in the TP and are primarily intended for use in interoperability test specifications

NOTE: However, in some instances, particularly where there is a considerable difference in complexity between the TPs and the TCs, it is worthwhile adding TDs as an extra design stage in a conformance test specification.

Test Purpose (TP): should be written for each potential test of each identified requirement

NOTE: A TP defines in broad terms what the goal of a particular test should be. A TP is defined in prose.

test suite: collection of Test Cases

testing framework: provides guidance for development of conformance and interoperability test strategies, test systems and the resulting test specifications

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GS MEC 001 [i.16] and the following apply:

API	Application Programming Interface
ATM	Abstract Test Method
ATS	Abstract Test Suite
BWMS	BandWidth Management Service
CON	CONformance

CRS	Conformance Requirement Statements
DUT	Device Under Test
FUT	Function Under Test
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
ICS	Implementation Conformance Statement
IFS	Interoperability Feature Statement
IOP	InterOPerability
IUT	Implementation Under Test
MEH	MEC Host
OAM	Operations And Maintaince
PDU	Packet Data Unit
PICS	Protocol Implementation Conformance Statement
PLMN	Public Land Mobile Network
RAB	Radio Access Bearer
RNI	Radio Network Information
RNIS	RNI Service
RP	Reference Point
SAQ	Service Availability Query
SUT	System Under Test
TC	Test Case
TCP	Transmission Control Protocol
TDL	Test Description Lanaguage
TP	Test Purpose
TSS	Test Suite Structure
TTCN	Testing and Test Control Notation
URI	Uniform Resource Identifier

4 Testing Methodology Guidelines for MEC

4.1 Introduction

Clause 4 provides:

- Identification of the implementations under test (IUT) for conformance testing and the device under test (DUTs) for interoperability, i.e. answering the question "what is to be tested".
- Definition of the applicable test procedures, i.e. answering the question "how is it to be tested".
- Definition of the procedure for development of test specifications and deliverables (for instance: Test Purposes (TP) in case of conformance testing and Test Descriptions (TD) in case of interoperability testing, documentation, etc.).

The MEC testing framework contains:

- a documentation structure:
  - catalogue of capabilities/features/functions (PICS or IFS);
  - Test Suite Structure (TSS);
  - individual tests in the form of TPs (Conformance) or TDs (Interoperability);
- a methodology linking the individual elements of a test specification together:
  - style guidelines and examples;
  - naming conventions;
  - a structured notation for TPs or TDs.

4.2 Basic concepts for conformance and interoperability testing

Conformance Testing and Interoperability Testing are the two main and complementary testing methodologies to test devices implementing standardized services [i.2]. These two testing methodologies also apply to MEC.

The basic concepts for Conformance Testing and Interoperability Testing are defined as follows:

- **Conformance Testing** can show that a product correctly implements and meets the requirements in the ETSI ISG MEC standards, which will include testing protocol message contents and formats as well as the permitted sequence of messages for the interfaces defined by ETSI ISG MEC standards.
- **Interoperability Testing** can demonstrate that a product will work with other alike products. It proves that **end-to-end functionality** between (at least) two functional elements is as required by the ETSI MEC standards on which those functions are based.

For more details about the basic concepts for conformance and interoperability testing, please refer to clause 4.1 of ETSI GS NFV-TST 002 [i.2].

4.3 Conformance Test Specifications

4.3.1 Introduction

Clause 4.3 explains how to apply the MEC conformance testing methodology in order to properly produce MEC conformance test specifications.

The conformance testing can show that a product correctly implements a particular standardized protocol, that is, it establishes whether or not the implementation under test meets the requirements specified for the protocol itself.

EXAMPLE: The scope of the testing is on protocol message content, format as well as the permitted sequences of messages. In that context, tests are performed at open standardized interfaces that are not (usually) accessible to an end user, and executed by a dedicated test system that has full control of the system under test and the ability to observe all incoming and out coming communications; the high degree of control of the test system over the sequence and contents of the protocol messages allows to test both valid and invalid behaviour.

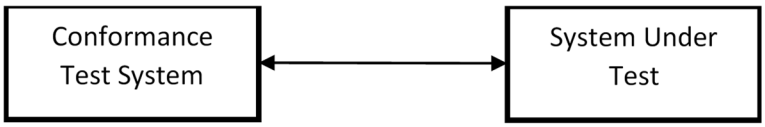

Figure 4.3.1-1: Conformance testing

Conformance test specifications should be produced following the methodology described in ISO/IEC 9646-1 [i.5]. In summary, this methodology begins with the collation and categorization of the features and options to be tested into a tabular form which is normally referred to as the "Implementation Conformance Statement" (ICS). All implemented capabilities supported by the Implementation Under Test (IUT) are listed by the implementer in the ICS, so that the tester knows which options have to be tested. This ensures that complete coverage is obtained.

The next step is to collect the requirements from the specification that is tested. For each requirement, one or more tests should be identified and classified into a number of groups which will provide a structure to the overall test suite (TSS).

A brief Test Purpose (TP) should then be written for each identified test and this should make it clear what is to be tested but not how this should be done. Finally, a detailed Test Case (TC) is written for each TP. In the interests of test automation, TCs are usually combined into an Abstract Test Suite (ATS) using a specific testing language such as TTCN-3 or others. The TCs in the ATS are then "Verified" against a number of IUTs for correct operation according to some agreed procedures, before being released for use by the industry.

In summary, the MEC Conformance Testing methodology consists of:

- Selection of Implementations Under Test (IUT).
- Identification of reference points.

- Development of test specifications, which includes:
  - Development of "Implementation Conformance Statements" (ICS).
  - Development of "Test Suite Structure and Test Purposes" (TSS&TP).
  - Development of "Abstract Test Suite" (ATS).

4.3.2 Test architecture

4.3.2.1 Selection of Implementation Under Test

4.3.2.1.1 Definition

The "Implementation Under Test" (IUT) is a protocol implementation considered as an object for testing. This means that the test process will focus on verifying the compliance of this protocol implementation (IUT) with requirements set up in the related base standard. An IUT normally is implemented in a "System Under Test" (SUT). For testing, a SUT is connected to a test system over at least a single interface. Such an interface is identified as "Reference Point" (RP) in the present document. Further details on RPs are presented in clause 6.

NOTE: Other interfaces between the test system and the IUT may be used to control the behaviour of the IUT during the test process.

Figure 4.3.2.1.1-1 shows the multi-access edge system reference architecture, see also clause 6 of ETSI GS MEC 003 [i.3].

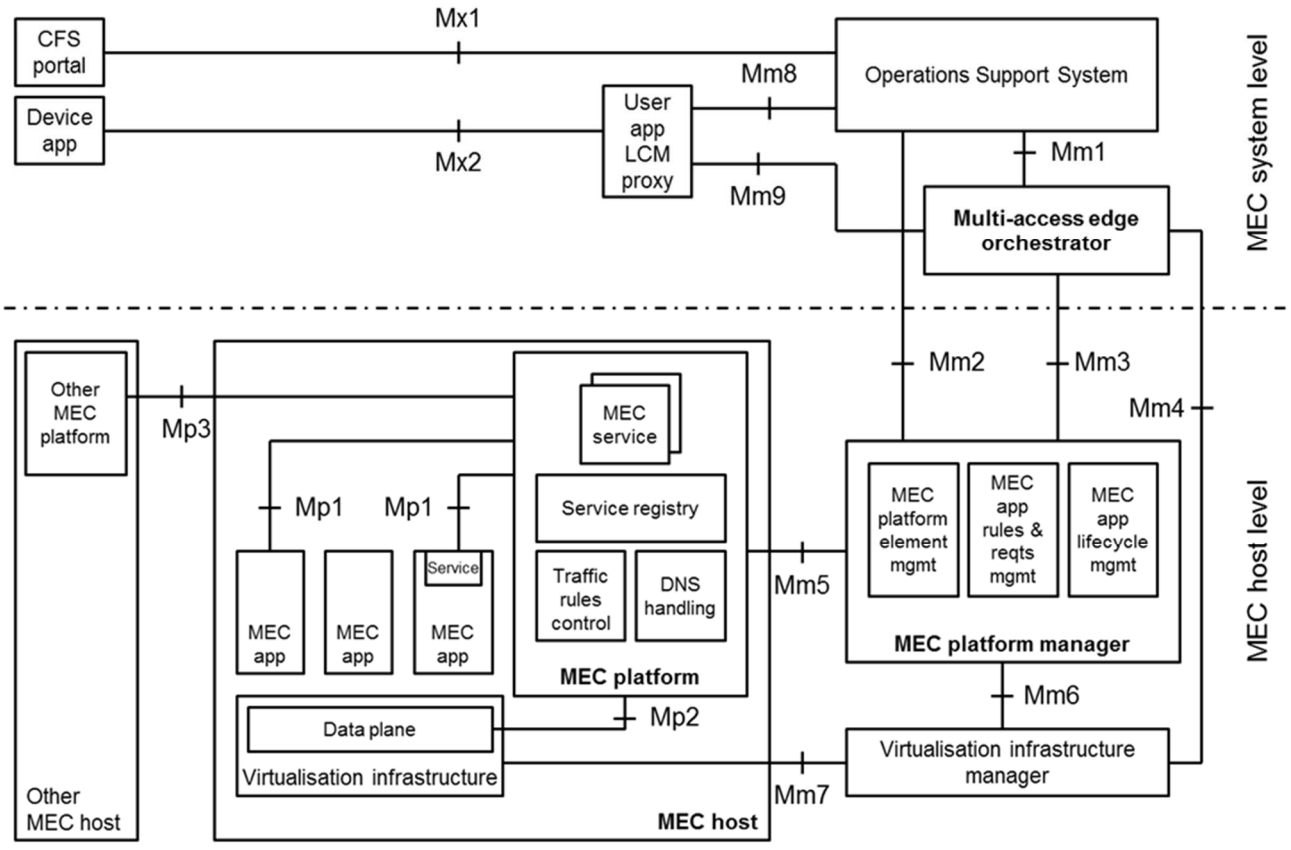

Figure 4.3.2.1.1-1: Multi-access edge system reference architecture

4.3.2.1.2 MEC IUTs and Reference Points

MEC IUTs and Reference Points are collected in tables as shown in the example below.

Table 4.3.2.1.2-1: Example of MEC IUT assessment

IUT	Reference	Reference Points	Notes
Multi-access edge application (MEC app)	Clause 6 of ETSI GS MEC 003 [i.3]	Mp1	
Multi-access platform (MEC plat)	Clause 6 of ETSI GS MEC 003 [i.3]	Mp1, Mp2, Mm5	Mp2 and Mm5 out of scope of testing

These tables need to be amended in the following cases:

- A new node or entity is defined in the base specifications.
- A new interface is defined in the base specifications between any of the existing nodes or entities.

4.3.3 Development of Conformance Test Specifications

4.3.3.1 Implementation Conformance Statement (ICS)

The purpose of an ICS is to identify those standardized functions which an IUT is required to support, those which are optional and those which are conditional on the presence of other functions. It helps to provide a means for selection of the suite of tests which will subsequently be developed.

In addition, the ICS can be used as a proforma for identifying which functions an IUT will support when performing conformance testing. The purpose of this ICS proforma is to provide a mechanism whereby a MEC implementation supplier may provide information about the implementation in a standardized manner. The information in an ICS is usually presented in tabular form as recommended in ISO/IEC 9646-7 [i.4].

The ICS can be considered as a set of "switches" which specify the capability of supporting the requirement in base standards to be tested. It is possible that with different choices in an ICS proforma, several different set of TPs will be necessary.

In clauses 5 "Requirement assessment" and 6 "Architecture assessment" assessments are made on whether requirements, features, components and other capabilities are required according to a referenced GS, or in order to achieve compliance. This assessment provides the following options:

m	mandatory - the capability is required to be supported.
o	optional - the capability may, or may not, be supported.
c.i	conditional - the requirement on the capability ("m", "o", "x" or "n/a") depends on the support of other optional or conditional items. "i" is an integer identifying a unique conditional status expression which is defined immediately following the table.
n/a	not applicable - in the given context, it is not possible to use the capability.
x	prohibited (excluded) - there is a requirement not to use this capability in the given context.
o.i	qualified optional - for mutually exclusive or selectable options from a set: "i" is an integer which identifies a unique group of related optional items and the logic of their selection which is defined immediately following the table.

An example is shown in the Table 4.3.3.1-1.

Table 4.3.3.1-1: Roles

Item	Role	Reference	Status
1	User	1.3	o.1
2	Network	1.2	o.1
o.1:	At least one item should be supported.		

4.3.3.2 Test Suite Structure & Test Purposes (TSS&TP)

4.3.3.2.1 Introduction

A test purpose is a prose description of a well-defined objective of testing. Applying to conformance testing, it focuses on a single conformance requirement or a set of related conformance requirements from the base standards.

Several types of presentation of the test purposes exist. These presentations are combining text with graphical presentations, mainly tables, and include sometimes message sequence charts. The present document presents a proposed table template to write test purposes with recommendations concerning the wording and the organization of the test purposes.

There are usually numerous test purposes, which need to be organized in structured groups. The organization of the test purposes in groups is named "Test Suite Structure".

The development of the test purposes follows the analysis of the conformance requirements, clearly expressed in the base standards. Furthermore, the analysis of a base standard leads to the identification of different groups of functionalities, which are used to define the first levels of the test suite structure.

4.3.3.2.2 Test Suite Structure

Defining the test suite structure consists of grouping the test purposes according to different criteria like for instance:

- The functional groups and sub-groups of procedures in the base standard, from which the requirement of the test purpose is derived.
- The category of test applying to the test purposes, for instance:
  - valid behaviour test;
  - invalid behaviour test;
  - timer test;
  - etc.

Usually the identification of the different functional groups of procedures leads to the definition of the top levels of the TSS. Then further levels at the bottom of the TSS is used to group test purposes belonging to the same type of test.

Table 4.3.3.2.2-1 shows an example of a two level TSS used in the TSS&TP for the MEC system.

Table 4.3.3.2.2-1: Example of test suite structure

TP_<root>_<gr>_<sgr>_<xx>_<nnn>		
<root> = root	MEC	MEC
<gr> = group	App	MEC application
	Plat	MEC platform
<sgr> = sub- group	Mp1	Reference point Multi-access platform 1
	Mp2	Reference point Multi-access platform 2
<fea> = feature	SAQ	Service Availability Query
<xx> = type of testing	BI	Invalid Behaviour tests
	BO	Inopportune Behaviour tests
	BV	Valid Behaviour tests
<nnn> = sequential number		001 to 999

Each feature is characterized by a number of functional requirements, some of them explicitly identified in the existing MEC specifications. Additionally, MEC 025 has identified further requirements, derived from the specification of the MEC services REST APIs. These requirements are identified in the rest of the document adopting a specific numbering schema, as follows: **MEC025.<if>.<fea>.<nnn>.** This schema is described in Table 4.3.3.2.2-2.

Table 4.3.3.2.2-2: Example of functional requirement identifier

MEC025.<if>.<fea>.<nn>		
<if> = interface-name	Mm1	Reference point
<fea> = feature	AppPkgm	Application Package Management
<nnn> = sequential number		001 to 999

4.3.3.2.3 Test Purpose

A test purpose is an informal description of the expected test behaviour. As such it is written in prose.

When needed to clarify the TP, it is helpful to add some graphical presentations, mainly tables, and include message sequence charts.

In order to increase the readability of the TP, the following recommendations should be followed:

- Each TP should contain:
  - The TP header, which contains the TP identifier, the TP objective and the external references (ICS, and base standard).
  - The behaviour part, which contains the test behaviour description. This part can be optionally divided in the three following parts, in order to increase the readability:
    - the initial conditions;
    - the expected behaviour;
    - the final conditions.
- Each TP should be written according to TDL-TO [i.15].

Table 4.3.3.2.3-1: Description of the fields of the TP

TP Header	
TP ID	The TP identifier identifies uniquely the test purposes. In order to ensure the uniqueness of the TP identifier, it follows a naming convention. The more useful and straightforward naming convention consists of using the test suite structure, to form the first part of the TP identifier. Then the final part consists of a number to identify the TP order within a TP group. The TP identifier is formed by the abbreviation "TP", followed by abbreviation representing the group of the following TSS levels, ending with a number representing the TP order. Each field of the TP identifier is separated by a "_". A TP identifier, following the TP naming convention of the table could be for instance TP_MEC_MEApp_MP1_BV_001. The TP numbering uses two digits for presentation, and starts with 01 rather than with 00. Exceeding 99 TPs per group is not recommended. In such a case, it is rather recommended to create sub-groups, in order to keep clarity in the Test Suite Structure.
Test objective	The test objective clearly indicates which requirement is intended to be tested in the test purpose. This part eases the understanding of the TP behaviour. This also eases the identification of the requirements, which were used as a basis for the test purpose. It is recommended to limit the length of the test objective to one sentence.
Reference	In the reference row, the TP writer indicates, in which clauses of the protocol standards, the requirement is expressed. This information is critical, because it justifies the existence and the behaviour of the TP. The reference row may refer to several clauses. When the clause containing the requirement is big (for instance, more than 1/2 page), it is recommended to indicate the paragraph of the clause where the requirement was identified. The reference to the base standard actually is precise enough to enable the TP reader to identify quickly and precisely the requirement.
Config Id	The pointer to the applicable test configuration. A test configuration defines how the test system connects to the SUT.
ICS selection	The ICS selection row contains a Boolean expression, made of ICS parameters. It is recommended to use ICS acronym, which clearly identify the role of the ICS.
TP Behaviour	
Initial conditions	The initial conditions define in which initial state the IUT has to be to apply the actual TP. In the corresponding Test Case, when the execution of the initial condition does not succeed, it leads to the assignment of an Inconclusive verdict.
Expected behaviour (TP body)	Definition of the events, which are parts of the TP objective, and the IUT are expected to perform in order to conform to the base specification. In the corresponding Test Case, Pass or Fail verdicts can be assigned there.
Final conditions (optional)	Definition of the events that the IUT is expected to perform or mandated not to perform, according to the base standard and following the correct execution of the actions in the expected behaviour above. In the corresponding Test Case, the execution of the final conditions is evaluated for the assignment of the final verdict.

4.3.3.2.4 Conventions for the expected behaviour in Test Purposes

In order to increase the comprehension of the test purposes, a number of conventions regarding data has been defined.

The data exchanged in the test execution is expressed (within the Expected Behaviour field of the TP) in three different ways, depending on their types among:

- Fixed values.
- Configurable values.
- Irrelevant values.

Fixed values, as per definition, are values that never change within or among any executions of test. In this case a literal value should be used and wrapped in quotes (i.e. "401 Unauthorized").

Configurable values may change among different executions of the tests. In this case, the value should be defined in capital letters, separating all the words by the underscore symbol (i.e. SUBSCRIPTION_HREF_URI).

Irrelevant values may be required in the exchange of the messages but have no impact on the test outcome and verdict. In this case, if the attribute name is relevant to the reader, the keyword "attribute" should be used (i.e.: attribute ID) or the keyword "set" (ID set to "any value").

When a message is meant not to have a specific field the name of the field is prepended with "not". In this case the Test System should send a message without that specific field.

Datatype naming done according to the OpenAPIs (when possible) and references are added. When a collection of object is specified without an explicit name in the OpenAPIs, the TP Expected Behaviour should create a name in the form of "<DataType>List" according to the datatype of objects in the collection.

In every data, object or message instance, the test should present only fields on which requirements or provisions are set.

In the TP, the general rules of HTTP message conformance are not explicitly reported each time but abstracted with the usage of identifiers such as vGET, vPOST, etc. Each one of these identifiers refers to HTTP requests where the following headers are conformant to the relevant MEC specification:

- Accept.
- Authorization.
- Content type.

4.3.3.3 Abstract Test Method (ATM)

4.3.3.3.1 Methodology

The Abstract Test Method (ATM) is defined in ISO/IEC 9646-1 [i.5] and supports a wide range of approaches for testing including e.g. the TTCN-3 abstract test language [i.6].

An abstract protocol tester presented in Figure 4.3.3.3.1-1 emulates a peer IUT of the same layer/the same entity. This type of test architecture provides a situation of communication which is equivalent to real operation between MEC systems. The MEC test system will simulate valid and invalid protocol behaviour, and will analyse the reaction of the IUT. Then the test verdict, e.g. pass or fail, will depend on the result of this analysis. Thus, this type of test architecture enables to focus the test objective on the IUT behaviour only.

In order to access an IUT, the corresponding abstract protocol tester needs to use lower layers to establish a proper connection to the system under test (SUT) over a physical link (Lower layers link).

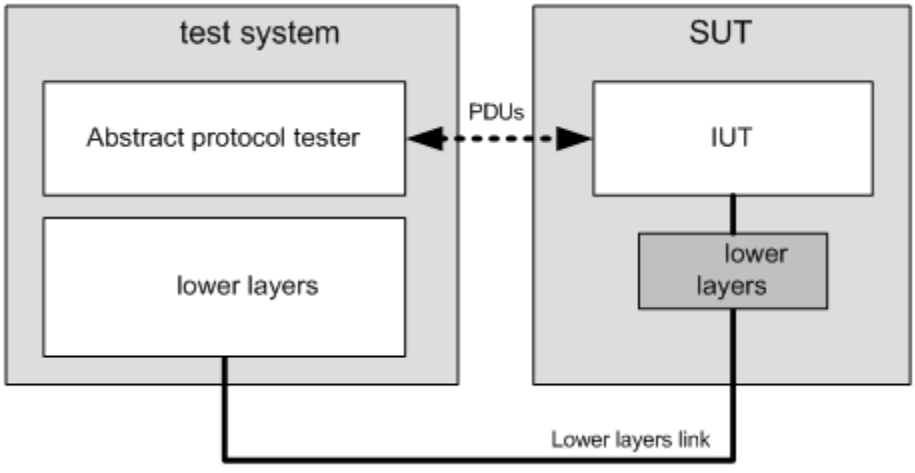

Figure 4.3.3.3.1-1: Generic abstract protocol tester

The Protocol Data Units (PDUs) are the messages exchanged between the IUT and the abstract protocol tester as specified in the base standard of the IUT. These PDUs are used to trigger the IUT and to analyse the reaction from the IUT on a trigger. Comparison of the result of the analysis with the requirements specified in the base standard allows to assign the test verdict.

For instance, to test the MEC app, the abstract protocol tester will emulate the MEC platform and use HTTP PDUs, TCP and IPV4/IPV6 protocol in the transport and networking layer and ethernet in the access layer.

4.3.3.3.2 Abstract PDU Transport Protocol

The application services described in the MEC specifications and addressed by this testing framework specify RESTful services over HTTPS as the PDUs transport protocol. Some specifications briefly mention that other protocols also may be used when concerns such as low latency or high-volume data transfers are a factor to consider.

To foster tests re-use and accommodate future evolutions of the MEC specifications on what concerns PDUs transport, conformance test suites should be designed to be able to use different transport protocols; the default implementation should support the protocol required mandatorily in the specifications. The bridge between the transport abstraction and the actual protocol to use should be done by test adapters, which translate the abstract PDU message into the intended protocol to use for the validation of the IUT.

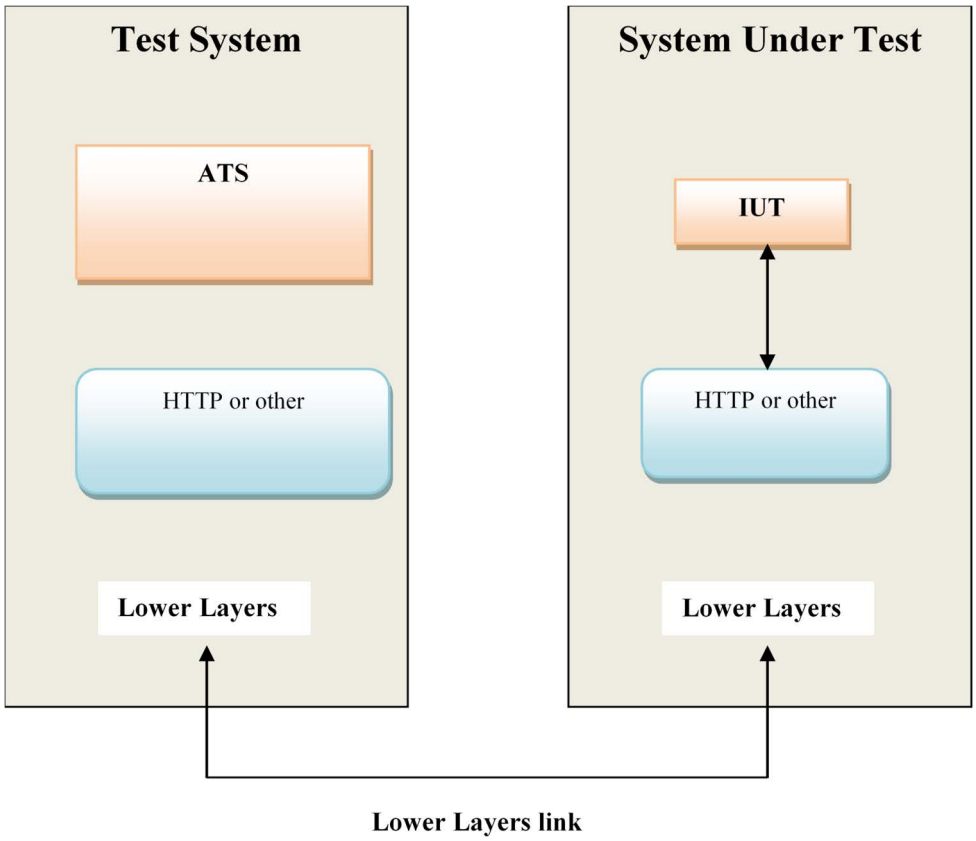

Figure 4.3.3.3.2-1: The architecture of Abstract Protocol testing with example of transport protocols for MEC

For instance, to test the MEC app, the abstract protocol tester will emulate the MEC platform, whereby the test suite will use events such as sending and receiving data, and the test adapters will use HTTP PDUs, TCP and IPV4/IPV6 protocol in the transport and networking layer and ethernet in the access layer.

4.4 Interoperability Test Specifications

4.4.1 Introduction

The MEC interoperability testing methodology follows the NFV interoperability testing methodology (defined in ETSI GS NFV-TST 002 [i.2]) where applicable. Hence references to [i.2] are used where possible and additional remarks are made where necessary.

Clause 4.1 of ETSI GS NFV-TST 002 [i.2] applies for this clause of the present document.

4.4.2 Basic concepts for interoperability testing

Clause 4.2 of ETSI GS NFV-TST 002 [i.2] applies for this clause of the present document.

Clause 4.2.3 of ETSI GS NFV-TST 002 [i.2] applies with the following modification:

- In the context of MEC, a Function Under Test is a combination of software and/or hardware items which implement the functionality of one or several MEC components and interact with other FUTs via one or more reference points, as described in ETSI GS MEC 003 [i.3].

4.4.3 Interoperability Test Specifications

Clause 4.3 of ETSI GS NFV-TST 002 [i.2] applies for this clause of the present document.

Clause 4.3.2 of ETSI GS NFV-TST 002 [i.2] applies with the following modifications:

- Modified the example of a generic SUT architecture: The test interface allows to observe message exchanges between the entities. The test interface can be implemented as a manual test operator action.

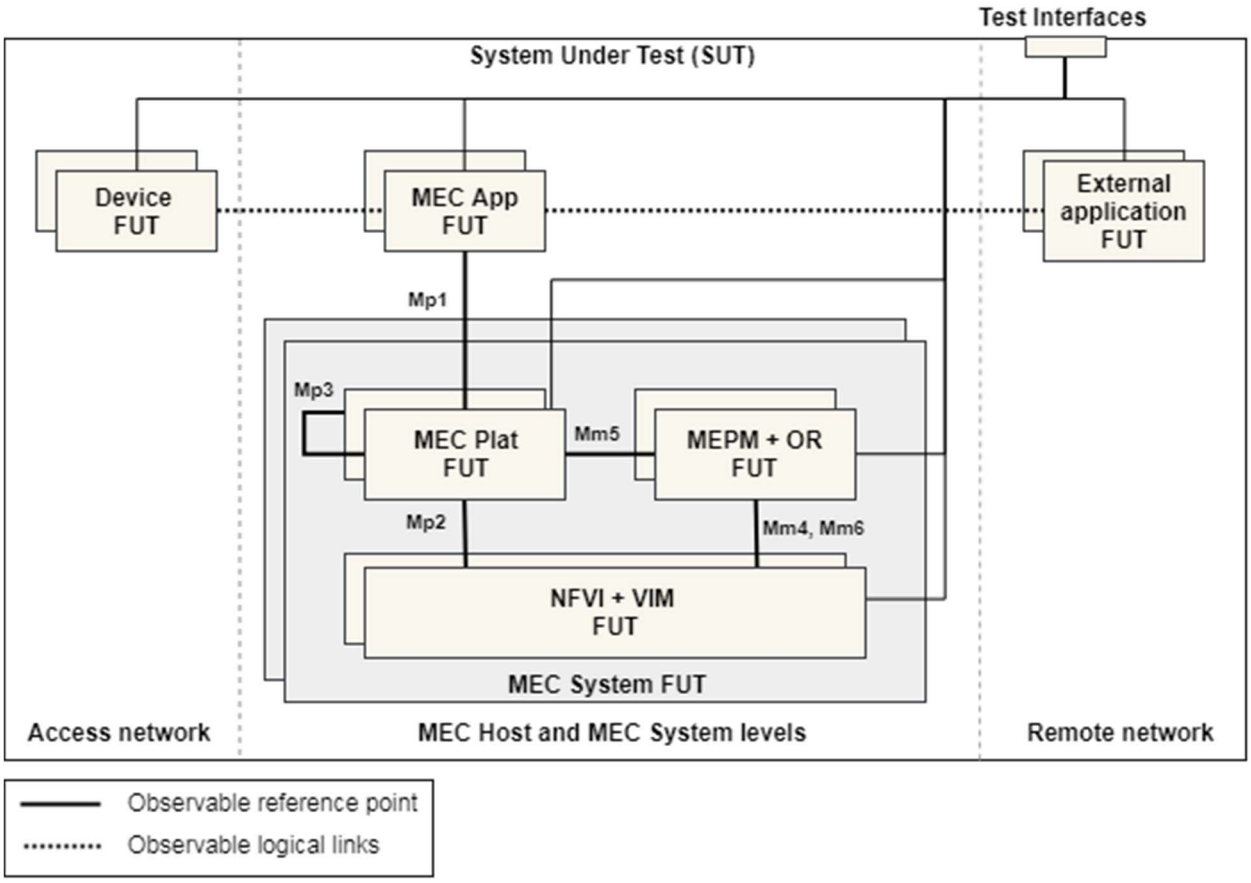

Figure 4.4.3-1: EXAMPLE: A specific SUT architecture for a specific interoperability test may contain Device FUT, MEC App FUT and MEC System FUT

NOTE: The generic SUT architecture serves as a base for identification of SUT architectures for different interoperability testing activities. The specific SUT architecture may contain only a subset of the elements in the generic SUT architecture. The specific SUT architecture should contain only the minimal set of FUTs needed to support the interoperability testing activity.

Clause 4.3.3 of ETSI GS NFV-TST 002 [i.2] applies with the following modification:

- The IFS provide the means to compile and organize all the following information:
  - FUT Identification.
  - Supported Functional Blocks: e.g. MEC App, MEC Platform, MEC Platform Manager.
  - Supported Role (when/if applicable): e.g. producer/consumer, source/sink.

- Supported Features: e.g. Radio Network Information.
- Supported options: e.g. RAB information, PLMN information.
- Applicable reference point.
- For each identified IOP feature the following information is provided:
  - A unique **identifier** - the usage of a naming convention allowing to put the feature into context (Functional Block, (Role), Functional Group, etc.) is recommended.
  - A short description of the **feature.**
  - A **reference** to the base specification.
  - The feature **status**: mandatory (M), optional (O), conditional (C).
  - In the IFS pro-forma, an additional field allows to state whether the implementation in question supports or not the feature (Y/N).
- There is no substantial difference between IFS and ICS. The IFS can be considered as a group of ICS, i.e. the IFS focuses on high level features whereas the ICS focuses on detailed implementation choices.

Clause 4.3.7 of ETSI GS NFV-TST 002 [i.2] applies with the following modification:

- Modified Test Description template.

Table 4.4.3-1: Test Description Template Example

Interoperability Test Description			
Identifier	*Unique test description ID: TD_AB_XXX_00. Follows a well-defined naming convention.*		
Test Purpose	*A concise summary of the test reflecting its purpose and allowing readers to easily distinguish this test from any other test in the document.*		
Configuration	*List of all the FUTs required devices for running this test, possibly also including a (reference to) an illustration of the SUT configuration.*		
References	*List of references to the base specification clause(s), use case(s), requirement(s), etc. which are either used in the test or define the functionality being tested.*		
Applicability	*List of features and capabilities in the IFS which are required to be supported by the SUT in order to execute this test (e.g. if this list contains an optional feature to be supported, then the test is optional).*		
Pre-test conditions	*List of test specific pre-conditions that need to be met by the SUT including information about configuration, i.e. precise description of the initial state of the SUT prior to start executing the test sequence:* • •		
Test Sequence	**Step**	**Type**	**Description**
	1	***<Type>***	*Step description*
	2		
	3		
	4		
	5		
	6		

- Added the 'CON check' type description:
  - A **stimulus** corresponds to an event that triggers a specific action on a FUT, like sending a message for instance.
  - A **configure** corresponds to an action to modify the FUT or SUT configuration.
  - An **IOP check** consists of observing that one FUT behaves as described in the standard: i.e. resource creation, update, deletion, etc. For each IOP check in the Test Sequence, a result can be recorded.

- A **CON check** consists of validating the appropriate sequence and contents of the protocol messages. For each CON check in the Test Sequence, a result can be recorded.
- The overall IOP Verdict will be considered OK if all the IOP checks in the sequence are OK.
- The overall CON Verdict will be considered OK if all the CON checks in the sequence are OK.

4.4.4 Interoperability Testing Process

Clause 4.4 of ETSI GS NFV-TST 002 [i.2] applies for this clause of the present document.

4.5 Interoperability Test Process

As described in clauses 4.2 and 4.3, the conformance testing can be performed in conjunction with the interoperability testing. Although there are quite distinct differences between conformance testing and interoperability testing, it is valid to consider using the techniques together to give combined results. Such an approach is described as the interoperability test with conformance checks.

During a Test Session, for each Test Description of interoperability test with conformance check in the Test Specification, the following actions are taken as part of the test execution:

1) Determine if the Test Description (TD) is in scope:
   - If all the Interoperable Functions Statements (IFS) listed in the Applicability field are met by all the concerned IUTs.
   - Optionally, when running Interoperability Testing with Conformance Checks, if all the Conformance Requirement Statements (CRS) listed in the Applicability field are met by the concerned IUT.
2) Setup the System Under Test (i.e. combination of IUTs) according to the Configuration described in the TD.
3) Configure each IUT to match the configuration of peering IUTs and to ensure the resulting System Under Test (SUT) will follow the expected test behaviour and meet the Test Purpose.
4) Take the System Under Test to the state described in the pre-test conditions.
5) Follow the steps Test Sequence, which will be a combination of the following:
   - Trigger an action on one of the IUTs to initiate the expected test process described in the Stimulus steps.
   - Verify that the IUTs behave according to the expectations, that is, as described in the IOP Check steps. Record the result.
   - Optionally, when running Interoperability Testing with Conformance Checks, check for the compliance of the interaction among the concerned IUTs, as described in the CON Check steps. Record the result.
6) Assess the Test Verdict as follows:
   - IOP Verdict, will be OK if all the IOP Checks are OK.
   - When running Interoperability Testing with Conformance Checks, CON Verdict will be pass if all the CON Checks pass.

For more details about the interoperability test process, please refer to clause 4.4 of ETSI GS NFV-TST 002 [i.2].

5 Requirement assessment

5.1 Introduction

The starting point for defining a MEC compliant implementation is ETSI GS MEC 002 [i.7], since this specifies the technical requirements for MEC, including whether they are mandatory, conditional or optional. Therefore, the purpose of the following clauses is to perform a per requirement assessment of the necessity to fulfil each requirement in a MEC compliant implementation. The assessment also identifies which reference point(s) and API(s) are utilized in fulfilling those requirements, if that has been specified.

Some of the requirements defined in ETSI GS MEC 002 [i.7] have not been addressed in the current MEC specifications or they have not been translated into practical and technical requirements associated to architecture components or interfaces; for this reason, they cannot be mapped into specific reference points or APIs. In these cases, the reference points, the APIs and the related GSs in the tables below are reported as "Not Specified". Moreover, when the definition of reference points, APIs and GSs is either not in scope of the present document or not directly related due to the high level nature of the requirement ,the term "Not Applicable" is used.

5.2 Generic requirements

5.2.1 Framework Requirements

Table 5.2.1-1: Framework Requirements

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
Framework-01	O	Not Applicable	Not Applicable	Not Applicable
Framework-02	M	Not Applicable	Not Applicable	Not Applicable
Framework-03	M	Not Applicable	Not Applicable	Not Applicable

5.2.2 Application lifecycle

Table 5.2.2-1: Application lifecycle

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
Lifecycle-01	M	Mm1, Mm3	apmi, alcmi	ETSI GS MEC 010-2 [i.9]
Lifecycle-02	M	Mm1, Mm3	apmi, alcmi	ETSI GS MEC 010-2 [i.9]
Lifecycle-03	M	Mm1, Mm3	apmi, alcmi	ETSI GS MEC 010-2 [i.9]
Lifecycle-04	M	Mm1, Mm3	apmi, alcmi	ETSI GS MEC 010-2 [i.9]
Lifecycle-05	M	Mm1, Mm3	apmi, alcmi	ETSI GS MEC 010-2 [i.9]
Lifecycle-06	M	Mm1, Mm3	apmi, alcmi	ETSI GS MEC 010-2 [i.9]

5.2.3 Applications environment

Table 5.2.3-1: Applications environment

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
AppEnvironment-01	M	Mm1, Mm3, Mp1	apmi, alcmi, mp1	ETSI GS MEC 010-2 [i.9], ETSI GS MEC 011 [i.10]
AppEnvironment-02	M	Not Specified	Not Specified	Not Specified
AppEnvironment-03	M	Not Specified	Not Specified	Not Specified

5.2.4 Support of mobility

Table 5.2.4-1: Support of mobility

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
Mobility-01	M	Not Applicable	Not Applicable	Not Applicable
Mobility-02	M	Not Applicable	Not Applicable	Not Applicable
Mobility-03	O	Not Applicable	Not Applicable	Not Applicable

5.3 Services requirements

5.3.1 Platform essential functionality

5.3.1.1 Mobile edge services

Table 5.3.1.1-1: Mobile edge services

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
Services-01	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Services-02	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Services-03	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Services-04	M	Not Specified	Not Specified	Not Specified
Services-05	M	Not Specified	Not Specified	Not Specified
Services-06	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Services-07	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Services-08	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Services-09	M	Not Specified	Not Specified	Not Specified

5.3.1.2 Connectivity

Table 5.3.1.2-1: Connectivity

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
Connectivity-01	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Connectivity-02	M	Not Specified	Not Specified	Not Specified
Connectivity-03	M	Not Specified	Not Specified	Not Specified
Connectivity-04	M	Not Specified	Not Specified	Not Specified

5.3.1.3 Storage

Table 5.3.1.3-1: Storage

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
Storage-01	M	Not Specified	Not Specified	Not Specified

5.3.1.4 Traffic routing

Table 5.3.1.4-1: Traffic routing

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
Routing-01	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Routing-02	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Routing-03	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Routing-04	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Routing-05	M	Not Specified	Not Specified	Not Specified
Routing-06	M	Not Specified	Not Specified	Not Specified
Routing-07	M	Not Specified	Not Specified	Not Specified
Routing-08	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Routing-09	M	Not Specified	Not Specified	Not Specified
Routing-10	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Routing-11	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Routing-12	O	Mp1	mp1	ETSI GS MEC 011 [i.10]
Routing-13	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Routing-14	M	Mp1	mp1	ETSI GS MEC 011 [i.10]

5.3.1.5 DNS support

Table 5.3.1.5-1: DNS support

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
DNS-01	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
DNS-02	M	Mp1	mp1	ETSI GS MEC 011 [i.10]

5.3.1.6 Timing

Table 5.3.1.6-1: Timing

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
Timing-01	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Timing-02	O	Not Specified	Not Specified	Not Specified

5.3.2 Features

5.3.2.1 Feature UserApps

Table 5.3.2.1-1: Feature UserApps

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
UserApps-01	O	Mx2	mx2	ETSI GS MEC 016 [i.14]
UserApps-02	C	Mx2	mx2	ETSI GS MEC 016 [i.14]
UserApps-03	C	Mx2	mx2	ETSI GS MEC 016 [i.14]
UserApps-04	C	Mx2	mx2	ETSI GS MEC 016 [i.14]
UserApps-05	C	Mx2	mx2	ETSI GS MEC 016 [i.14]
UserApps-06	C	Mx2	mx2	ETSI GS MEC 016 [i.14]
UserApps-07	C	Mx2	mx2	ETSI GS MEC 016 [i.14]
UserApps-08	C	Mx2	mx2	ETSI GS MEC 016 [i.14]

5.3.2.2 Feature SmartRelocation

Table 5.3.2.2-1: Feature SmartRelocation

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
SmartReloc-01	O	Not Specified	Not Specified	Not Specified
SmartReloc-02	C	Not Specified	Not Specified	Not Specified
SmartReloc-03	C	Not Specified	Not Specified	Not Specified
SmartReloc-04	C	Not Specified	Not Specified	Not Specified
SmartReloc-05	C	Not Specified	Not Specified	Not Specified
SmartReloc-06	C	Not Specified	Not Specified	Not Specified

5.3.2.3 Feature RadioNetworkInformation

Table 5.3.2.3-1: Feature RadioNetworkInformation

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
RNI-01	O	Mp1	rni	ETSI GS MEC 011 [i.10] and ETSI GS MEC 012 [i.1]
RNI-02	C	Mp1	rni	ETSI GS MEC 011 [i.10] and ETSI GS MEC 012 [i.1]
RNI-03	C	Mp1	rni	ETSI GS MEC 011 [i.10] and ETSI GS MEC 012 [i.1]
RNI-04	C	Mp1	rni	ETSI GS MEC 011 [i.10] and ETSI GS MEC 012 [i.1]
RNI-05	C	Mp1	rni	ETSI GS MEC 011 [i.10] and ETSI GS MEC 012 [i.1]
RNI-06	C	Mp1	rni	ETSI GS MEC 011 [i.10] and ETSI GS MEC 012 [i.1]
RNI-07	C	Mp1	rni	ETSI GS MEC 011 [i.10] and ETSI GS MEC 012 [i.1]

5.3.2.4 Feature LocationService

Table 5.3.2.4-1: Feature LocationService

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
Location-01	O	Mp1	ls	ETSI GS MEC 011 [i.10] and ETSI GS MEC 013 [i.11]
Location-02	C	Mp1	ls	ETSI GS MEC 011 [i.10] and ETSI GS MEC 013 [i.11]
Location-03	C	Mp1	ls	ETSI GS MEC 011 [i.10] and ETSI GS MEC 013 [i.11]
Location-04	C	Mp1	ls	ETSI GS MEC 011 [i.10] and ETSI GS MEC 013 [i.11]
Location-05	C	Mp1	ls	ETSI GS MEC 011 [i.10] and ETSI GS MEC 013 [i.11]
Location-06	C	Mp1	ls	ETSI GS MEC 011 [i.10] and ETSI GS MEC 013 [i.11]

5.3.2.5 Feature BandwidthManager

Table 5.3.2.5-1: Feature BandwidthManager

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
Bandwidth-01	O	Mp1	bwm	ETSI GS MEC 011 [i.10] and ETSI GS MEC 014 [i.12]
Bandwidth-02	C	Mp1	bwm	ETSI GS MEC 011 [i.10] and ETSI GS MEC 014 [i.12]
Bandwidth-03	C	Mp1	bwm	ETSI GS MEC 011 [i.10] and ETSI GS MEC 014 [i.12]

5.3.2.6 Feature UEIdentity

Table 5.3.2.6-1: Feature UEIdentity

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
UEIdentity-01	O	Mp1	ui	ETSI GS MEC 011 [i.10] and ETSI GS MEC 015 [i.13]
UEIdentity-02	C	Mp1	ui	ETSI GS MEC 011 [i.10] and ETSI GS MEC 015 [i.13]
UEIdentity-03	C	Mp1	ui	ETSI GS MEC 011 [i.10] and ETSI GS MEC 015 [i.13]
UEIdentity-04	C	Mp1	ui	ETSI GS MEC 011 [i.10] and ETSI GS MEC 015 [i.13]
UEIdentity-05	C	Mp1	ui	ETSI GS MEC 011 [i.10] and ETSI GS MEC 015 [i.13]

5.4 Operation and management requirements

Table 5.4-1: Operation and management requirements

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
OAM-01	M	Not Specified	Not Specified	Not Specified
OAM-02	M	Not Specified	Not Specified	ETSI GS MEC 010-1 [i.8]

5.5 Security, regulation, charging requirements

Table 5.5-1: Security, regulation, charging requirements

Requirement ID	Required according to GS	Reference Point(s)	API(s)	GS(s)
Security-01	M	Not Specified	Not Specified	Not Specified
Security-02	M	Mp1	mp1	ETSI GS MEC 011 [i.10]
Lawful-01	M	Not Specified	Not Specified	Not Specified
Charging-01	M	Not Specified	Not Specified	Not Specified

6 Architecture assessment

6.1 MEC components compliance

6.1.1 Mobile Edge Platform Manager

A MEC Platform Manager (MEPM) supports the following features with reference to the Mm2 reference point ETSI GS MEC 010-1 [i.8]:

- MEC Host configuration management (active).
- MEC Host configuration management (passive).
- MEC Host fault management.
- MEC Applications configuration management (active).
- MEC Applications configuration management (passive).
- MEC Applications state management.

A MEC Platform Manager (MEPM) supports the following features with reference to the Mm3 reference point ETSI GS MEC 010-2 [i.9]:

- Application Lifecycle Management.
- Application Lifecycle Change Notification.
- Application Package Management Subscription.

6.1.2 Mobile Edge Orchestrator

A MEC Orchestrator (MEO) supports the following features with reference to the Mm1 reference point [i.9]:

- Application Package Management.
- Application Lifecycle Management.

A MEC Orchestrator (MEO) supports the following features with reference to the Mm3 reference point [i.9]:

- Application Package Management.
- Application Lifecycle Change Notification.

6.1.3 MEC Platform

A MEC Platform supports the following features with reference to the Mp1 reference point ETSI GS MEC 011 [i.10]:

- MEC service assistance.
- MEC application assistance.
- Traffic routing and DNS rules.
- Timing.
- Transport information.

The MEC platform supports also the following features, which however are not fully specified and are left for proprietary implementation:

Mm5 reference point (see ETSI GS MEC 003 [i.3], clause 7.2.2):

- Platform configuration.
- Configuration of the application rules and requirements.
- Application lifecycle support procedures.
- Management of application relocation.

Mp2 reference point (see ETSI GS MEC 003 [i.3], clause 7.2.2):

- Instruction of the data plane about how to route traffic among applications, networks, services, etc.

6.1.4 MEC Apps

MEC Applications make use of the services discovered over the Mp1 reference point ETSI GS MEC 011 [i.10]. The features supported by each service are described next.

The Radio Network Information Service (RNIS) supports the following features ETSI GS MEC 012 [i.1]:

- Up-to-date radio network information regarding radio network conditions.
- Measurement information related to the user plane based on 3GPP specifications.
- Information about UEs connected to the radio node(s) associated with the MEC host, their UE context and the related radio access bearers.

The Location Service supports the following features ETSI GS MEC 013 [i.11]:

- The location information of UEs currently served by the radio node(s) associated with the MEC host.
- The distance between a specified location and a UE currently served by the radio node(s) associated with the MEC host.
- A list of UEs in a particular location area.

The UE Identity service supports the following features ETSI GS MEC 014 [i.12]:

- To register a tag (representing a UE) or a list of tags.

The Bandwidth Management Service (BWMS) supports the following features ETSI GS MEC 015 [i.13]:

- Static and dynamic bandwidth allocation.
- Bandwidth size and priority definition.
- Retrieve configured bandwidth allocation.

The UE application interface service supports the following features [i.14]:

- User application look-up.
- Application context management.
- Receiving notification events.

Annex A:
Conformance Test Purposes examples

A.1 Examples of test purposes

A.1.1 Introduction

Below are listed examples of Test Purposes for conformance testing edited in TDL-TO [i.15] and exported in a document format. The TPs are formatted according to the provisions in clause 4 of the present document. The TPs are not meant to be valid in the content but to be a validation of the methodology developed in the Framework.

A.1.2 Example of test purpose for querying DNS rules

TP Id	"TP_MEC_PLAT_MP1_DNS_BV_001"
Test Objective	Check that the IUT responds with a list of active DNS rules when queried by a MEC Application
Reference	ETSI GS MEC 011 [i.10], clause 7.12.3.1 https://forge.etsi.org/gitlab/mec/gs011-app-enablement-api/blob/master/Mp1.yaml#/definitions/DnsRule
PICS Selection	PIC_MEC_PLAT and PIC_SERVICES
Initial Conditions	

with {
the IUT being_in idle_state and
the IUT having a apps_instance containing
instance_id indicating value APP_INSTANCE_ID
;
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a vGET containing
Uri indicating value "mp1/v1/applications/{APP_INSTANCE_ID}/dns_rules"
;
from the MEC_APP
}
then {
// MEC 011, clause 7.12.3.1
the IUT sends a HttpMsg containing
status_code set to "200 OK"
body containing
DnsRuleList containing
DnsRule set to any_value
;
;
;
to the MEC_APP
}
}	

Final Conditions	

A.1.3 Example of test purpose for notification of an App Package enablement

TP Id	"TP_MEC_MEO_MM3_PKGM_002"
Test Objective	Check that the MEO sends an App Package management notification to the subscriber when the App Package is enabled
Reference	ETSI GS MEC 010-2 [i.9], clause 7.3.5 ETSI GS MEC 010-2 [i.9], Table 6.2.3.8.2-1
PICS Selection	PIC_APP_PACKAGE and PIC_APP_PACKAGE_NOTIFICATIONS

Initial Conditions

```
with {
        the IUT having a App_Package containing
        appPkgId set to APP_PKG_ID,
        appProvider set to APP_PROVIDER,
        operationalState set to "DISABLED"
      ;
      and the IUT having a App_Package_subscription containing
        subscriptionId indicating value SUBSCRIPTION_ID,
        subscriptionType indicating value "OpChange",
        callbackUri indicating value CALLBACK_URI,
        appPkgFilter containing
          appPkgFilterCriteria containing
            appProvider set to APP_PROVIDER
          ;
        ;
      ;
}
```

Expected Behaviour

```
ensure that {
  when {
        the IUT receives a vPOST containing
          appPkgId set to APP_PKG_ID,
          operationalState set to "ENABLED";
        from the MEC_OSS
  }
  then {
        the IUT sends a vPOST containing
          uri indicating value CALLBACK_URI,
          body containing
            AppPkgNotification containing
              notificationId indicating value any_value,
              notificationType indicating value "OP_CHANGE",
              subscriptionId indicating value SUBSCRIPTION_ID,
              appPkgId set to APP_PKG_ID
            ;
          ;
        ; to the MEC_SUB
  }
}
```

Final Conditions

A.1.4 Example of test purpose for malformed App Package creation

TP Id	"TP_MEC_MEO_MM3_PKGM_002"
Test Objective	Check that the MEO sends an App Package management notification to the subscriber when the App Package is enabled
Reference	ETSI GS MEC 010-2 [i.9] V1.2.2, clause 7.3.5 ETSI GS MEC 010-2 [i.9] V1.2.2, Table 6.2.3.8.2-1
PICS Selection	PIC_APP_PACKAGE and PIC_APP_PACKAGE_NOTIFICATIONS

Initial Conditions

```
with {
        the IUT having a App_Package containing
        appPkgId set to APP_PKG_ID,
        appProvider set to APP_PROVIDER,
        operationalState set to "DISABLED"
      ;
      and the IUT having a App_Package_subscription containing
        subscriptionId indicating value SUBSCRIPTION_ID,
        subscriptionType indicating value "OpChange",
        callbackUri indicating value CALLBACK_URI,
        appPkgFilter containing
          appPkgFilterCriteria containing
            appProvider set to APP_PROVIDER
          ;
        ;
      ;
}
```

Expected Behaviour

```
ensure that {
  when {
        the IUT receives a vPOST containing
          appPkgId set to APP_PKG_ID,
          operationalState set to "ENABLED";
        from the MEC_OSS
  }
  then {
        the IUT sends a vPOST containing
          uri indicating value CALLBACK_URI,
          body containing
            AppPkgNotification containing
              notificationId indicating value any_value,
              notificationType indicating value "OP_CHANGE",
              subscriptionId indicating value SUBSCRIPTION_ID,
              appPkgId set to APP_PKG_ID
            ;
          ;
        ; to the MEC_SUB
  }
}
```

Final Conditions

A.1.5 Example of test purpose for unauthorized creation of App Package

TP Id	"TP_MEC_MEO_MM1_PKGM_001_NEG_B"
Test Objective	Check that MEO responds with an error when it receives a request for creating a new App Package without credentials
Reference	ETSI GS MEC 010-2 [i.9], clause 7.3.1.3.1 ETSI GS MEC 010-2 [i.9], Table 6.2.3.2.2-1
PICS Selection	PIC_APP_PACKAGE and PIC_APP_PACKAGE_MANAGEMENT
Initial Conditions	

with {
the IUT being_in idle_state
}	

Expected Behaviour	

ensure that {
when {
the IUT receives a POST containing
uri indicating value "/apmi/v1/app_packages",
"not" authorization, //Token is mandatory
body containing
AppPkg containing
appPkgName set to any_value
;
;
; from the MEC_OSS
}
then {
the IUT sends a HTTP_RESPONSE containing
status set to "401 Unauthorized"
; to the MEC_OSS
}
}	

Final Conditions	

Annex B:
Interoperability Test Descriptions examples

B.1 Examples of test descriptions for interoperability

B.1.1 Introduction

Below are listed examples of Test Descriptions (TD) for interoperability testing edited according to the guidelines reported in clause 4.4 of the present document. The TDs are not meant to be valid in the content but to be a validation of the methodology developed in the present Framework.

The examples are constructed on the basis of a subset of MEC Use Cases, as specified in ETSI GS MEC 002 [i.7]. The selection has been done according to testability and suitability for the example.

B.1.2 Example of test description for Mobile radio network information (TD_MEC002_A2)

Interoperability Test Description			
Identifier	TD_MEC002_A2 (Mobile radio network information)		
Test Purpose	Verify that a radio analytics MEC application is able to receive relevant radio network information.		
Configuration	**MEC-config-MEC002_A2**		
References	ETSI GS MEC 011 [i.10], clauses 5.2.2 and 5.2.5. ETSI GS MEC 012 [i.1], clauses 5.2.5 and 5.2.6. ETSI GS MEC 002 [i.7], clause A.2.		
Applicability	• Application Lifecycle Management. • Radio Network Information Service. • Traffic Routing Service.		
Pre-test conditions	• MEC App is up and running. • MEC App has authN/Z rights to perform all the requests mentioned as stimulus in the test sequence. • UEs connected to the radio nodes associated with the target MEC Host.		
Test Sequence	**Step**	**Type**	**Description**
	1	Stimulus	MEC App sends request for service availability query on RNI services through Mp1 (ETSI GS MEC 011 [i.10], clause 7.3.3.1).
	2	CON Check	MEC App receives information on available RNI services.
	3	Stimulus	MEC App subscribes updates on RNI data through Mp1 (ETSI GS MEC 012 [i.1], clause 7.6.3.4).
	4	IOP Check	Subscription created.
	5	IOP Check	MEC App receives updated RNI data which is consistent with the current status of the testbed setup.
	6	CON Check	MEC App acknowledges RNI data.

B.1.3 Example of test description for active device location tracking (TD_MEC002_A7)

Interoperability Test Description			
Identifier	TD_MEC002_A7_LCM (Active Device Location Tracking)		
Test Purpose	Verify the capability of the MEC system to provide information about a specific UE, or all UEs, or a certain category of UEs currently served by the access network associated with the MEC host, to a third-party server located in external networks.		
Configuration	**MEC-config-MEC002_A7**		
References	ETSI GS MEC 013 [i.11], clause 5.3.2-3. ETSI GS MEC 002 [i.7], clause A.7.		
Applicability	• Location Service.		
Pre-test conditions	• UEs are connected to the radio nodes which the target MEC Host is associated to. • AppD for Location Service MEC App available at MEO and related instance running. • Connectivity between external apps and MEC applications deployed on the target MEC Host is enabled (see req. Connectivity-04). • API client has authN/Z rights to perform all the requests mentioned as stimulus in the test sequence.		
Test Sequence	**Step**	**Type**	**Description**
	1	Stimulus	Send request to get geographical information about UE's access point (ETSI GS MEC 013 [i.11] clause 7.3.2.2 request 2).
	2	CON Check	Response with information about UE's access point geographical location (ETSI GS MEC 013 [i.11] clause 7.3.2.2 response 2).
	3	IOP Check	The response contains the correct location information regarding the UE FUTs.

B.1.4 Example of test description for Edge data orchestration (TD_MEC002_A10)

Interoperability Test Description	
Identifier	TD_MEC002_A10 (MEC Edge Data Orchestration)
Test Purpose	Verify the capability of the MEC system to provide mechanisms for the configuration of traffic rules based on network addresses or IP addresses and to perform routing of the uplink/downlink data traffic from/to UEs to/from the data orchestration MEC application.
Configuration	**MEC-config-MEC002_A10**
References	ETSI GS MEC 010-1 [i.8], clause 6.1.5-6. ETSI GS MEC 002 [i.7], clause A.10.
Applicability	• MEH configuration management (active) - Traffic rules.
Pre-test conditions	• Data Orchestration MEC Application already deployed in the target MEC Host. • AppD for Data Orchestration MEC Application on-boarded in MEO and including traffic rules. • Data Consumer and Data Producer UEs are connected to the access nodes which the target MEC Host is associated to. • API client has authN/Z rights to perform all the requests mentioned as stimulus in the test sequence.

Test Sequence	Step	Type	Description
	1	Stimulus	Trigger the configuration of traffic rules for data traffic from data producers UEs to Data Orchestration MEC application on Mm2.
	2	CON check	Traffic rules configuration successful response.
	3	IOP check	MEC Platform with uplink traffic rules configured.
	4	Stimulus	Trigger the activation of traffic rules for data traffic from data producers UEs to data orchestration MEC application on Mm2.
	5	CON check	Traffic rules activation successful response.
	6	IOP check	MEC Platform with uplink traffic rules activated.
	7	Stimulus	Trigger the configuration of traffic rules for data traffic from data orchestration MEC application to data consumer UEs on Mm2.
	8	CON check	Traffic rules configuration successful response.
	9	IOP check	MEC Platform with downlink traffic rules configured.
	10	Stimulus	Trigger the activation of traffic rules for data traffic from data orchestration MEC application to consumer UEs on Mm2.
	11	CON check	Traffic rules activation successful response.
	12	IOP check	MEC Platform with downlink traffic rules activated.
	13	Stimulus	Start traffic from data producer UEs.
	14	CON check	Verify traffic from data producer UEs to data orchestration MEC application.
	15	IOP check	Verify traffic routing on MEC Host towards the data orchestration MEC application.
	16	Stimulus	Request data from consumer UEs.
	17	CON check	Verify traffic data orchestration MEC application to consumer UEs.
	18	IOP check	Verify traffic routing on MEC Host from the Data Orchestration MEC application.

Annex C:
Change History

Date	Version	Information about changes
12/2017	0.0.2	MEC(17)000622 addition of MEC012 example. Added placeholders to clause 7 for MEC010-1, 10-2, 11, 12, 13, 14, 15 & 16. MEC(17)000665r1 MEC025 Proposal for clause 5 Requirement Assessment. The tables were completed using the spreadsheet included in the 665r1 contribution
05/2018	0.0.3	MEC(18)000178r1, MEC(18)000179r1, MEC(18)000180r1 & MEC(18)000181r1
07/2018	0.0.4	MEC(18)000214r1, noting this replaces text in clause 4 & MEC(18)000269r1
09/2018	0.0.5	MEC(18)000381r1
10/2018	0.0.6	MEC(18)000433r1, MEC(18)000443 & MEC(18)000444
12/2018	2.0.1	Moved to stable draft based on the incorporation of MEC(18)000477r2, MEC(18)000478, MEC(18)000482, MEC(18)000483 & MEC(18)000484
04/2019	2.0.2	MEC(19)000004, plus rapporteur edits

Annex D: Bibliography

- ETSI EG 202 237: "Methods for Testing and Specification (MTS); Internet Protocol Testing (IPT); Generic approach to interoperability testing".
- ETSI EG 202 568: "Methods for Testing and Specification (MTS); Internet Protocol Testing (IPT); Testing: Methodology and Framework".

History

Document history		
V2.1.1	June 2019	Publication