Contents

Intellectual Property Rights .......................................................................................................................... 4
Foreword .......................................................................................................................................................... 4
Modal verbs terminology ............................................................................................................................... 4
1 Scope .......................................................................................................................................................... 5
2 References ................................................................................................................................................ 5
   2.1 Normative references ............................................................................................................................. 5
   2.2 Informative references ........................................................................................................................... 6
3 Definition of terms, symbols and abbreviations ....................................................................................... 7
   3.1 Terms ..................................................................................................................................................... 7
   3.2 Symbols .................................................................................................................................................. 7
   3.3 Abbreviations ......................................................................................................................................... 7
4 ATS conventions .......................................................................................................................................... 7
   4.1 Introduction ............................................................................................................................................ 7
   4.2 Testing conventions ................................................................................................................................. 7
      4.2.1 TTCN-3 Testing conventions ............................................................................................................ 7
         4.2.1.1 Testing states .............................................................................................................................. 7
         4.2.1.1.1 Initial state .............................................................................................................................. 7
         4.2.1.1.2 Final state .................................................................................................................................. 8
         4.2.1.2 Message types - JSON definitions ............................................................................................ 8
     4.2.2 Robot Testing conventions .............................................................................................................. 8
   4.3 Naming conventions ............................................................................................................................... 8
      4.3.1 TTCN-3 Naming conventions ......................................................................................................... 8
         4.3.1.1 Introduction ............................................................................................................................... 8
         4.3.1.2 General guidelines ................................................................................................................... 8
     4.3.2 Usage of Log statements ................................................................................................................ 8
   4.3.3 Robot Naming conventions .............................................................................................................. 10
Annex A (normative): TTCN-3 Abstract Test Suite (ATS) ........................................................................... 11
Annex B (normative): Robot Framework Abstract Test Suite (ATS) ............................................................ 12
Annex C (informative): Test System Execution .......................................................................................... 13
   C.1 Information on execution of the TTCN-3 test suite ............................................................................. 13
   C.2 Information on execution of the Robot Framework test suite ............................................................. 13
Annex D (informative): Change History .................................................................................................... 14
History .......................................................................................................................................................... 15
Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are 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 Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, 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.

DECT™, PLUGTESTSTM, UMTSTM and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. 3GPP™ and LTE™ are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. oneM2M™ logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. GSM® and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

This Group Specification (GS) has been produced by ETSI Industry Specification Group (ISG) Multi-access Edge Computing (MEC).

The present document is part 3 of a multi-part deliverable. Full details of the entire series can be found in part 1 [i.7].

Modal verbs terminology

In the present document "shall", "shall not", "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

Based on the testing methodology guidelines and framework specified in ETSI GR MEC-DEC 025 [i.1], the present document specifies part 3 of a multi-part deliverable on conformance test specification. Part 3 (the present document) provides the Abstract Test Suites (ATS) in TTCN-3 [i.3] and the Robot Framework [i.5] for:

- Application Package Management and Application Lifecycle Management as specified in ETSI GS MEC 010-2 [2];
- the MEC Application Enablement API specified in ETSI GS MEC 011 [3]; and
- the MEC service APIs.

The MEC service APIs in scope of the present document are specified in:

- ETSI GS MEC 012 [4];
- ETSI GS MEC 013 [5];
- ETSI GS MEC 014 [6];
- ETSI GS MEC 015 [7];
- ETSI GS MEC 016 [8];
- ETSI GS MEC 021 [9];
- ETSI GS MEC 028 [10];
- ETSI GS MEC 029 [11]; and
- ETSI GS MEC 030 [12].

2 References

2.1 Normative 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.

Referenced documents which are not found to be publicly available in the expected location might be found at https://docbox.etsi.org/Reference.

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 necessary for the application of the present document.

[2] ETSI GS MEC 010-2 (V2.2.1): "Multi-access Edge Computing (MEC); MEC Management; Part 2: Application lifecycle, rules and requirements management".
[3] ETSI GS MEC 011 (V2.2.1): "Multi-access Edge Computing (MEC); Edge Platform Application Enablement".
[4] ETSI GS MEC 012 (V2.1.1): "Multi-access Edge Computing (MEC); Radio Network Information API".
[5] ETSI GS MEC 013 (V2.1.1): "Multi-access Edge Computing (MEC); Location API".
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 GR MEC-DEC 025 (V2.1.1) (06-2019): "Multi-access Edge Computing (MEC); MEC Testing Framework".

[i.2] ETSI ETS 300 406 (1995): "Methods for testing and Specification (MTS); Protocol and profile conformance testing specifications; Standardization methodology".

[i.3] ETSI ES 201 873-1 (V4.12.1): "Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; Part 1: TTCN-3 Core Language".


[i.6] ETSI EG 202 810 (V1.1.1) (03-2010): "Methods for Testing and Specification (MTS); Automated Interoperability Testing; Methodology and Framework".

[i.7] ETSI GS MEC-DEC 032-1: "Multi-access Edge Computing (MEC); API Conformance Test Specification; Part 1: Test Requirements and Implementation Conformance Statement (ICS)".

[i.8] ETSI GR MEC 001 (V3.1.1): "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 GR MEC 001 [i.8] apply.

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GR MEC 001 [i.8] and the following apply:

- ATS: Abstract Test Suite
- ICS: Implementation Conformance Statement
- IUT: Implementation Under Test
- PICS: Protocol Implementation Conformance Statement
- TTCN: Testing and Test Control Notation

4 ATS conventions

4.1 Introduction

The abstract test suites specified in the present document provide conformance tests for MEC API servers according to the ETSI MEC specifications.

In order to promote adoption and ease deployment of MEC technology, the present document targets a broad set of communities involved in the specific context of MEC. This fundamental and specific goal of MEC is addressed by requiring application of best practices for testing, most relevant for the telecommunication and software engineering communities.

In this regard, the Test Purposes specified in ETSI GS MEC-DEC 032-2 [13] have been the basis for the development of two Abstract Test Suites, formalized in two different languages for the testing domain: TTCN-3 [i.3] and Robot Framework [i.5].

The quality and coherency between the two ATS are sustained by the common Test Purposes and by a set of conventions used by the contributors. The conventions applied are described in the following clauses of the present document.

4.2 Testing conventions

4.2.1 TTCN-3 Testing conventions

4.2.1.1 Testing states

4.2.1.1.1 Initial state

All test cases start with the function f_cf_01_http_up. This function connects the Test System to the IUT and activates all the default messages processing for error handling.

As necessary, further actions may be included in the f_cf_01_http_up function.
4.2.1.2 Final state

All test cases end with the function f_cf_01_http_down. This function disconnects the Test System and stops all default messages processing for error handling.

As necessary, further actions may be included in the f_cf_01_http_down function.

4.2.1.2 Message types - JSON definitions

JSON definitions from MEC APIs are not directly imported in TTCN-3 but they are implemented within the respect of the JSON import method specified in ETSI ES 201 873-11 [i.4].

4.2.2 Robot Testing conventions

Test system configuration is implemented via configuration file name "variables.txt", distributed among individual folders.

Test steps are defined as Robot Framework Keywords either in the common "GenericKeywords.robot" file or within individual test cases files.

PICS are modelled as Tags in the Robot Framework language and can be therefore flagged at runtime to select the set of tests for execution.

4.3 Naming conventions

4.3.1 TTCN-3 Naming conventions

4.3.1.1 Introduction

This test suite follows the naming convention guidelines provided in the ETSI ETS 300 406 [i.2].

4.3.1.2 General guidelines

The naming convention is based on the following underlying principles:

- in most cases, identifiers should be prefixed with a short alphabetic string (specified in table 4.3.1.2-1) indicating the type of TTCN-3 element it represents;
- suffixes should not be used;
- prefixes and suffixes should be separated from the body of the identifier with an underscore ("_");
- only module names, data type names and module parameters should begin with an upper-case letter. All other names (i.e. the part of the identifier following the prefix) should begin with a lower-case letter;
- the start of second and subsequent words in an identifier should be indicated by capitalizing the first character. Underscores should not be used for this purpose.

EXAMPLE 1: c_sixteen, t_wait.

EXAMPLE 2: f_initialState.

Table 4.3.1.2-1 specifies the naming guidelines for each element of the TTCN-3 language indicating the recommended prefix, suffixes (if any) and capitalization.

<table>
<thead>
<tr>
<th>Language element</th>
<th>Naming convention</th>
<th>Prefix</th>
<th>Example identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module</td>
<td>Use upper-case initial letter</td>
<td>none</td>
<td>AtsMec_AppEnablementAPI_TestCases</td>
</tr>
</tbody>
</table>

Table 4.3.1.2-1: ETSI TTCN-3 generic naming conventions
<table>
<thead>
<tr>
<th>Language element</th>
<th>Naming convention</th>
<th>Prefix</th>
<th>Example identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group within a module</td>
<td>Use lower-case initial letter</td>
<td>none</td>
<td>app_saq</td>
</tr>
<tr>
<td>Data type</td>
<td>Use upper-case initial letter</td>
<td>none</td>
<td>Headers</td>
</tr>
<tr>
<td>Message template</td>
<td>Use lower-case initial letter</td>
<td>m_</td>
<td>m_security_info</td>
</tr>
<tr>
<td>Message template with wildcard</td>
<td>Use lower-case initial letter</td>
<td>mw_</td>
<td>mw_end_point_uris</td>
</tr>
<tr>
<td>Signature template</td>
<td>Use lower-case initial letter</td>
<td>s_</td>
<td>s_callSignature</td>
</tr>
<tr>
<td>Port instance</td>
<td>Use lower-case initial letter</td>
<td>none</td>
<td>httpPort</td>
</tr>
<tr>
<td>Test component instance</td>
<td>Use lower-case initial letter</td>
<td>none</td>
<td>userTerminal</td>
</tr>
<tr>
<td>Constant</td>
<td>Use lower-case initial letter</td>
<td>c_</td>
<td>c_maxRetransmission</td>
</tr>
<tr>
<td>Constant (defined within</td>
<td>Use lower-case initial letter</td>
<td>cc_</td>
<td>cc_minDuration</td>
</tr>
<tr>
<td>component type</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>External constant</td>
<td>Use lower-case initial letter</td>
<td>cx_</td>
<td>cx_macId</td>
</tr>
<tr>
<td>Function</td>
<td>Use lower-case initial letter</td>
<td>f_</td>
<td>f_authentication()</td>
</tr>
<tr>
<td>External function</td>
<td>Use lower-case initial letter</td>
<td>fx_</td>
<td>fx_calculateLength()</td>
</tr>
<tr>
<td>Altstep (incl. Default)</td>
<td>Use lower-case initial letter</td>
<td>a_</td>
<td>a_cf_01_http_notif_down()</td>
</tr>
<tr>
<td>Test case</td>
<td>Use ETSI numbering</td>
<td>TC_</td>
<td>TC_MEC_SRV_APPSAQ_001_OK</td>
</tr>
<tr>
<td>Variable (local)</td>
<td>Use lower-case initial letter</td>
<td>v_</td>
<td>v_headers</td>
</tr>
<tr>
<td>Variable (defined within a</td>
<td>Use lower-case initial letter</td>
<td>vc_</td>
<td>vc_systemName</td>
</tr>
<tr>
<td>component type</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Timer (local)</td>
<td>Use lower-case initial letter</td>
<td>t_</td>
<td>t_wait</td>
</tr>
<tr>
<td>Timer (defined within a</td>
<td>Use lower-case initial letter</td>
<td>tc_</td>
<td>tc_authMin</td>
</tr>
<tr>
<td>component)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Module parameters for PICS</td>
<td>Use all upper case letters</td>
<td>PICS_</td>
<td>PICS_APP_ENABLEMENT_API_SUPPORTED</td>
</tr>
<tr>
<td>Module parameters for other</td>
<td>Use all upper case letters</td>
<td>PX_</td>
<td>PX_APP_INSTANCE_ID</td>
</tr>
<tr>
<td>parameters</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Formal Parameters</td>
<td>Use lower-case initial letter</td>
<td>p_</td>
<td>p_headers</td>
</tr>
<tr>
<td>Enumerated Values</td>
<td>Use lower-case initial letter</td>
<td>e_</td>
<td>e_success</td>
</tr>
</tbody>
</table>

### 4.3.2 Usage of Log statements

All TTCN-3 log statements use the following format using the same order:

- Three asterisks.
- The TTCN-3 test case or function identifier in which the log statement is defined.
- One of the categories of log: INFO, WARNING, ERROR, PASS, FAIL, INCONC, TIMEOUT.
- Free text.
- Three asterisks.

**EXAMPLE 1:**

```log
log("*** " & testcasename() & ": INFO: Registration for notification succeed ***");
```

**NOTE:** The INCONC category of log refer to the case of inconclusive test verdict as defined in ETSI EG 202 810 [i.6] and ETSI ES 201 873-1 [i.3], i.e. “test verdict given when the observed test outcome is such that neither a pass nor a fail verdict can be given”.

*ETSI*
Furthermore, the following rules are applied for the all ATS:

- Log statements are used in the body of the functions, so that invocations of functions are visible in the test logs.
- All TTCN-3 setverdict statements are combined (as defined in ETSI ES 201 873-1 [i.3]) with a log statement following the same above rules (see example 2).

**EXAMPLE 2:**  
```tcc
setverdict(pass, "*** " & testcasename() & : PASS: IUT successfully responds with a ServiceInfoList ***").
```

### 4.3.3 Robot Naming conventions

None applicable.
Annex A (normative):
TTCN-3 Abstract Test Suite (ATS)

The TTCN-3 MEC API Conformance Test Suite has been developed using the Testing and Test Control Notation (TTCN) according to ETSI ES 201 873-1 [i.3].

This test suite has been compiled error-free using a third party TTCN-3 compiler. The TTCN-3 library modules, which form parts of the present document, are accessible from the ETSI Forge at https://forge.etsi.org/rep/mec/gs032p3-ttcn-test-suite.
Annex B (normative):
Robot Framework Abstract Test Suite (ATS)

The Robot Framework MEC API Conformance Test Suite has been developed using the Robot Framework [i.5].

This test suite has been validated error-free using the official Robot Framework Python® distribution. The Robot Framework test definitions, keyword definitions and configuration files, which form parts of the present document, are accessible from the ETSI Forge at https://forge.etsi.org/rep/mec/gs032p3-robot-test-suite.
Annex C (informative):
Test System Execution

C.1 Information on execution of the TTCN-3 test suite

The detailed information needed to compile and run the TTCN-3 test suite is available at https://forge.etsi.org/rep/mec/gs032p3-ttcn-test-suite/-/blob/3.1.1/README.md.

The documentation provided suggests possible approach by means of virtualized solutions to install and setup the TTCN-3 environment.

C.2 Information on execution of the Robot Framework test suite

The execution of the Robot Framework test suite comprises three main steps:

1) installation and configuration of the required software, in particular the prerequisites for the Robot Framework;

2) configuration of the IUT specifics (e.g. ICS selection);

3) execution of the test against the IUT.

Annex D (informative):
Change History

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Information about changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>July 2019</td>
<td>V0.0.1</td>
<td>Initial proposal: MECDECODE(19)000015.</td>
</tr>
<tr>
<td>November 2019</td>
<td>V0.0.2</td>
<td>Incorporated MECDECODE(19)000076.</td>
</tr>
<tr>
<td>November 2019</td>
<td>V0.0.3</td>
<td>Move to “stable draft” after editHelp! review.</td>
</tr>
<tr>
<td>November 2020</td>
<td>V0.0.4</td>
<td>Incorporated MECDECODE(20)000057r3, which is considered sufficient to move the document to “final draft”.</td>
</tr>
<tr>
<td>May 2021</td>
<td>V3.0.1</td>
<td>Updated references to the repositories.</td>
</tr>
<tr>
<td>July 2021</td>
<td>V3.0.2</td>
<td>Updated references to the repositories.</td>
</tr>
<tr>
<td>February 2022</td>
<td>V3.0.4</td>
<td>TTF T012 updates.</td>
</tr>
</tbody>
</table>
## History

<table>
<thead>
<tr>
<th>Document history</th>
</tr>
</thead>
<tbody>
<tr>
<td>V2.1.1 December 2020 Publication</td>
</tr>
<tr>
<td>V3.1.1 April 2022 Publication</td>
</tr>
</tbody>
</table>