### SAFESPOT Certification Reference Framework – Part B

<table>
<thead>
<tr>
<th>Deliverable No. (use the number indicated on technical annex)</th>
<th>D7.4.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>SubProject No.</td>
<td>SP7</td>
</tr>
<tr>
<td>SubProject Title</td>
<td>Score</td>
</tr>
<tr>
<td>Workpackage No.</td>
<td>WP4</td>
</tr>
<tr>
<td>Workpackage Title</td>
<td>Exploitation Convergence &amp; Certification</td>
</tr>
<tr>
<td>Task No.</td>
<td>T7.4.2</td>
</tr>
<tr>
<td>Task Title</td>
<td>ITS Cooperating System Certification Approach</td>
</tr>
<tr>
<td>Authors (per company, if more than one company provide it together)</td>
<td>Main Authors: A. Plaza, F.J. Nuñez, J. Baños (AT4 wireless)</td>
</tr>
<tr>
<td>Status (F: final; D: draft; RD: revised draft)</td>
<td>F</td>
</tr>
<tr>
<td>Version No.</td>
<td>1.6</td>
</tr>
<tr>
<td>File Name</td>
<td>SF_D7.4.3_SCRF-Part B_v1.6.doc</td>
</tr>
<tr>
<td>Planned Date of submission according to TA</td>
<td>31/03/2008</td>
</tr>
<tr>
<td>Issue Date</td>
<td>24/09/2010</td>
</tr>
<tr>
<td>Project start date and duration</td>
<td>01 February 2006, 48 Months</td>
</tr>
</tbody>
</table>
## Revision Log

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Reason</th>
<th>Name and Company</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>2007-04-18</td>
<td>First Input</td>
<td>A. Plaza, F J. Nuñez, J. Baños (AT4 wireless)</td>
</tr>
<tr>
<td>0.2</td>
<td>2008-01-03</td>
<td>New Index Proposal</td>
<td>A. Plaza, F J. Nuñez, J. Baños (AT4 wireless)</td>
</tr>
<tr>
<td>0.3</td>
<td>2008-01-21</td>
<td>Part B.2 Content</td>
<td>A. Plaza, F J. Nuñez, J. Baños (AT4 wireless)</td>
</tr>
<tr>
<td>0.5</td>
<td>2008-03-07</td>
<td>Performance Testing</td>
<td>A. Plaza, F J. Nuñez, J. Baños (AT4 wireless)</td>
</tr>
<tr>
<td>0.6</td>
<td>2008-03-28</td>
<td>Certification Content</td>
<td>A. Plaza, F J. Nuñez, J. Baños (AT4 wireless)</td>
</tr>
<tr>
<td>0.9</td>
<td>2008-03-31</td>
<td>Version ready to be reviewed</td>
<td>A. Plaza, F J. Nuñez, J. Baños (AT4 wireless)</td>
</tr>
<tr>
<td>1.0</td>
<td>2008-04-24</td>
<td>Final Version after Peer Review</td>
<td>A. Plaza, F J. Nuñez, J. Baños (AT4 wireless)</td>
</tr>
<tr>
<td>1.1</td>
<td>2008-10-29</td>
<td>EU reviewers comments included</td>
<td>A. Plaza, F J. Nuñez, J. Baños (AT4 wireless)</td>
</tr>
<tr>
<td>1.2</td>
<td>2009-10-30</td>
<td>Second EU reviewers comments included</td>
<td>A. Plaza, F J. Nuñez, J. Baños (AT4 wireless)</td>
</tr>
<tr>
<td>1.4</td>
<td>2010-01-26</td>
<td>Conclusion section added</td>
<td>A. Plaza, F J. Nuñez, J. Baños (AT4 wireless)</td>
</tr>
<tr>
<td>1.5</td>
<td>2010-09-15</td>
<td>EU reviewers from 4th review included</td>
<td>A. Plaza, F J. Nuñez, J. Baños (AT4 wireless)</td>
</tr>
<tr>
<td>1.6</td>
<td>2010-09-24</td>
<td>Further minor checks and validation</td>
<td>G. Vivo (CRF)</td>
</tr>
</tbody>
</table>
## Abbreviation List

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>3GPP</td>
<td>3rd Generation Mobile System</td>
</tr>
<tr>
<td>ADAS</td>
<td>Advanced Driver Assistance System</td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>ASN.1</td>
<td>Abstract Syntax Notation</td>
</tr>
<tr>
<td>ATC</td>
<td>Abstract Test Case</td>
</tr>
<tr>
<td>ATCRF</td>
<td>Automobile Telematics Certification Reference Framework</td>
</tr>
<tr>
<td>ATS</td>
<td>Abstract Test Suite</td>
</tr>
<tr>
<td>ATSO</td>
<td>Automobile Telematic Stakeholders Organization (ATSO)</td>
</tr>
<tr>
<td>C2C – CC</td>
<td>Car to Car Communication Consortium</td>
</tr>
<tr>
<td>CA</td>
<td>Certification Authority</td>
</tr>
<tr>
<td>CALM</td>
<td>Continuous Air Interface, Long and Medium range</td>
</tr>
<tr>
<td>CAN</td>
<td>Controller Area Network</td>
</tr>
<tr>
<td>Car OEM</td>
<td>Car Original Equipment Manufacturer</td>
</tr>
<tr>
<td>DECT</td>
<td>Digital Enhanced Cordless Telecommunications</td>
</tr>
<tr>
<td>CEN</td>
<td>Comité Européen de Normalisation</td>
</tr>
<tr>
<td>CVIS</td>
<td>Co-operative Vehicle-Infrastructure Systems (IP project, IST 027 293)</td>
</tr>
<tr>
<td>DoC</td>
<td>Declaration of Compliance</td>
</tr>
<tr>
<td>DSRC</td>
<td>Dedicated Short Range Communications</td>
</tr>
<tr>
<td>EC</td>
<td>European Commission</td>
</tr>
<tr>
<td>EITSFA</td>
<td>European Intelligent Transport Systems Framework Architecture</td>
</tr>
<tr>
<td>ETSI</td>
<td>European Telecommunications Standards Institute</td>
</tr>
<tr>
<td>EUCAR</td>
<td>European Council for Automotive R &amp; D</td>
</tr>
<tr>
<td>EUT</td>
<td>Equipment Under Test</td>
</tr>
<tr>
<td>FOT</td>
<td>Field Operation Testing</td>
</tr>
<tr>
<td>FP6</td>
<td>Framework Programme 6</td>
</tr>
<tr>
<td>FTP</td>
<td>File Transfer Protocol</td>
</tr>
<tr>
<td>GPS</td>
<td>Global Positioning System</td>
</tr>
<tr>
<td>GSM</td>
<td>Global System for Mobile</td>
</tr>
<tr>
<td>HMI</td>
<td>Human Machine Interface</td>
</tr>
<tr>
<td>ICS</td>
<td>Implementation Conformance Statement</td>
</tr>
<tr>
<td>IDL</td>
<td>Interface Definition Language</td>
</tr>
<tr>
<td>IEC</td>
<td>International Electrotechnical Commission</td>
</tr>
<tr>
<td>IEE</td>
<td>Institution of Electrical Engineers</td>
</tr>
<tr>
<td>IEEEE</td>
<td>Institute of Electrical and Electronics Engineers</td>
</tr>
<tr>
<td>Acronym</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>IP</td>
<td>Integrated Project</td>
</tr>
<tr>
<td>ISM</td>
<td>Industrial, Scientific and Medical</td>
</tr>
<tr>
<td>ISO</td>
<td>International Organization for Standardization</td>
</tr>
<tr>
<td>ISO/TC204</td>
<td>ISO Standardization of Transport Information and Control Systems (TICS)</td>
</tr>
<tr>
<td>ITS</td>
<td>Intelligent Transportation Systems</td>
</tr>
<tr>
<td>ITU</td>
<td>International Telecommunication Union</td>
</tr>
<tr>
<td>IUT</td>
<td>Implementation Under Test</td>
</tr>
<tr>
<td>IXIT</td>
<td>Implementation eXtra Information for Testing</td>
</tr>
<tr>
<td>KAREN</td>
<td>Keystone Architecture Required for European Networks</td>
</tr>
<tr>
<td>LDM</td>
<td>Local Dynamic Map</td>
</tr>
<tr>
<td>MAC</td>
<td>Medium Access Control</td>
</tr>
<tr>
<td>NGN</td>
<td>Next Generation Networks</td>
</tr>
<tr>
<td>OBU</td>
<td>On Board Unit</td>
</tr>
<tr>
<td>OSCP</td>
<td>On-line Certificate Status Protocol</td>
</tr>
<tr>
<td>OSGi</td>
<td>Open Services Gateway initiative</td>
</tr>
<tr>
<td>OSI</td>
<td>Open System Interconnection</td>
</tr>
<tr>
<td>PA</td>
<td>Platform Adapter</td>
</tr>
<tr>
<td>PHY</td>
<td>Physical Layer</td>
</tr>
<tr>
<td>PICS</td>
<td>Protocol Implementation Conformance Statement</td>
</tr>
<tr>
<td>PIXIT</td>
<td>Protocol Implementation eXtra Information for Testing</td>
</tr>
<tr>
<td>PKI</td>
<td>Public Key Infrastructure</td>
</tr>
<tr>
<td>QE</td>
<td>Qualified Equipment</td>
</tr>
<tr>
<td>QOS</td>
<td>Quality of Service</td>
</tr>
<tr>
<td>QP</td>
<td>Quality Plan</td>
</tr>
<tr>
<td>RFID</td>
<td>Radio Frequency IDentification</td>
</tr>
<tr>
<td>RQ</td>
<td>Requirement Catalogue</td>
</tr>
<tr>
<td>RSU</td>
<td>Road Side Unit</td>
</tr>
<tr>
<td>SA</td>
<td>System Adapter</td>
</tr>
<tr>
<td>SAFESPOT</td>
<td>Global System for Telematics (IP project)</td>
</tr>
<tr>
<td>SCORE</td>
<td>SAFESPOT CORE architecture</td>
</tr>
<tr>
<td>SCRF</td>
<td>SAFESPOT Certification Framework</td>
</tr>
<tr>
<td>SEVECOM</td>
<td>Secure Vehicle Communication</td>
</tr>
<tr>
<td>SFCDA</td>
<td>SAFESPOT Certification Administrator</td>
</tr>
<tr>
<td>SFCB</td>
<td>SAFESPOT Certification Body</td>
</tr>
<tr>
<td>SFTL</td>
<td>SAFESPOT Authorized Trusted Test Laboratory</td>
</tr>
<tr>
<td>SFTCRL</td>
<td>SAFESPOT Test Case Reference List</td>
</tr>
<tr>
<td>Acronym</td>
<td>Definition</td>
</tr>
<tr>
<td>---------</td>
<td>------------</td>
</tr>
<tr>
<td>SFTS</td>
<td>SAFESPOT Test System</td>
</tr>
<tr>
<td>SLA</td>
<td>Service Level Agreement</td>
</tr>
<tr>
<td>SLS</td>
<td>System Level Specification</td>
</tr>
<tr>
<td>SMA</td>
<td>Safety Margin Assistance</td>
</tr>
<tr>
<td>SoA</td>
<td>State of Art</td>
</tr>
<tr>
<td>SP</td>
<td>Sub Project</td>
</tr>
<tr>
<td>STRF</td>
<td>SAFESPOT Testing Reference Framework</td>
</tr>
<tr>
<td>SUT</td>
<td>System Under Test</td>
</tr>
<tr>
<td>SUUT</td>
<td>SAFESPOT Unit Under Test</td>
</tr>
<tr>
<td>T3RTS</td>
<td>TTCN-3 Runtime System</td>
</tr>
<tr>
<td>TCI</td>
<td>TTCN-3 Control Interface</td>
</tr>
<tr>
<td>TCU</td>
<td>Telematic Control Unit</td>
</tr>
<tr>
<td>TE</td>
<td>TTCN-3 Executable</td>
</tr>
<tr>
<td>TELCO</td>
<td>Telecommunication Company</td>
</tr>
<tr>
<td>TM</td>
<td>Test Management</td>
</tr>
<tr>
<td>TO</td>
<td>Test Operator</td>
</tr>
<tr>
<td>TP</td>
<td>Test Purpose</td>
</tr>
<tr>
<td>TRI</td>
<td>TTCN-3 RunTime Interface</td>
</tr>
<tr>
<td>TS</td>
<td>Test Site</td>
</tr>
<tr>
<td>TSS</td>
<td>Test Suite Structure</td>
</tr>
<tr>
<td>TTCN-3</td>
<td>Testing and Test Control Notation</td>
</tr>
<tr>
<td>UML</td>
<td>Unified Modelling Language</td>
</tr>
<tr>
<td>UWB</td>
<td>Ultra Wide Band</td>
</tr>
<tr>
<td>V2I</td>
<td>Vehicle to Infrastructure</td>
</tr>
<tr>
<td>V2V</td>
<td>Vehicle to Vehicle</td>
</tr>
<tr>
<td>VANET</td>
<td>Vehicle Ad Hoc Network</td>
</tr>
<tr>
<td>WAVE</td>
<td>Wireless Access in Vehicular Environments</td>
</tr>
<tr>
<td>WLAN</td>
<td>Wireless Local Area Network</td>
</tr>
<tr>
<td>WP</td>
<td>Work Package</td>
</tr>
<tr>
<td>XML</td>
<td>eXtensible Markup Language</td>
</tr>
</tbody>
</table>
Table of contents

Revision Log ................................................................. 2
Abbreviation List .......................................................... 3
Table of contents ......................................................... 6
List of Figures ............................................................... 7
List of Tables ............................................................... 7
EXECUTIVE SUMMARY .................................................... 8
  1. Introduction ................................................................ 9
  2. Basic Concepts ....................................................... 10
PART B.1 – CERTIFICATION FRAMEWORK .......................... 13
  3. SAFESPOT Certification Reference Framework (SCRF) ....... 14
      3.1. Introduction ....................................................... 14
      3.2. SAFESPOT Main Certification Players ....................... 14
      3.3. SAFESPOT Certification Process ............................ 16
          3.3.1. Cooperative System Forum Organisation ............... 16
          3.3.2. Cooperative System Forum Activities .................. 17
          3.3.3. Overview of the SAFESPOT Certification Process .... 17
          3.3.4. Test Specifications and associated Test Case Reference List .... 19
          3.3.5. SAFESPOT Authorized Trusted Test Laboratory (SFTL) Activities 22
          3.3.6. SAFESPOT Certification Body (SFCB) Activities ....... 23
          3.3.7. SAFESPOT Certification Authority (SFCA) Activities . 24
      3.4. Certification Scheme Implementation ....................... 24
          3.4.1. Tasks to build a certification scheme .................... 24
          3.4.2. Criteria for selecting SAFESPOT Test Facilities (SFTL) ... 24
PART B.2 – TESTING FRAMEWORK ...................................... 26
  4. Testing Introduction .................................................. 27
      4.1. Plugfest Events .................................................. 29
      4.2. Conformance Testing ............................................ 32
      4.3. Interoperability Testing ...................................... 36
          4.3.1. Basic Interoperability Testing ......................... 36
          4.3.2. Advanced Interoperability Testing .................... 39
      4.4. Performance Testing ............................................. 40
      4.5. Virtual Testing .................................................. 44
  5. SAFESPOT Testing Reference Framework (STRF) ............... 47
      5.1. Phase 1: Elements to be tested / certified ............... 48
      5.2. Phase 2: SAFESPOT Test Specification ..................... 53
          5.2.1. Requirements Catalogue ................................ 55
          5.2.2. Implementation Conformance Statement (ICS) ....... 55
          5.2.3. Test Strategy ................................................ 56
          5.2.4. Test Purpose & Test Suite Structure (TSS&TP) ....... 56
          5.2.5. Abstract Test Suite (ATS) ............................... 56
      5.3. Phase 3: SAFESPOT Test System .............................. 57
  6. SAFESPOT Test System Description (SFTS) ...................... 58
      6.1. SAFESPOT TTCN-3 Conformance Test System .......... 60
          6.1.1. TTCN-3: Formal Language for writing test specifications .... 60
          6.1.2. TTCN-3 Test System ..................................... 61
      6.2. SAFESPOT Interoperability Test System ................. 64
PART B.3 – Guidance for External Applicants ....................... 67
  7. Background ............................................................ 68
  8. Added-Value of SAFESPOT Certified Products ................. 69
  9. The SAFESPOT Approach to Certification ....................... 70
 10. Conclusions .......................................................... 72
 11. References ........................................................... 73
ANNEXES ................................................................. 75
Annex 1: SAFESPOT Test Specification Template ................ 76
Annex 2: UML Testing Profile - Formal modelling for testing .... 80
List of Figures

Figure 1. Cooperative System Forum .................................................... 15
Figure 2. SAFESPOT Certification Entities ............................................. 16
Figure 3. SAFESPOT High-level certification process ............................... 17
Figure 4. SAFESPOT Certification Process .......................................... 19
Figure 5. SAFESPOT Test Plan Generation .......................................... 20
Figure 6. Detailed Certification Process Flow ....................................... 22
Figure 7. Cooperative System label Proposal ...................................... 24
Figure 8. Plugfest Events ..................................................................... 30
Figure 9. Plugfest process ................................................................... 30
Figure 10. SAFESPOT Test Sites .......................................................... 32
Figure 11. Conformance Testing Examples ............................................. 33
Figure 12. Conformance Testing architecture ......................................... 33
Figure 13. IUT from OSI point of view .................................................. 34
Figure 14. Test System components ..................................................... 35
Figure 15. Generic Architecture for Interoperability Testing .................... 36
Figure 16. EUT, SUT & IUT relationship .............................................. 37
Figure 17. Example of EUT, SUT and IUT in a vehicle .............................. 38
Figure 18. Combining Conformance and Interoperability Testing ............... 40
Figure 19. Response Time Measurement .............................................. 42
Figure 20. Generic performance testing architecture ............................... 43
Figure 21. Virtual Testing Approach ................................................... 45
Figure 22. SAFESPOT High Level Vision ............................................ 47
Figure 23. SAFESPOT Phases in STRF .............................................. 48
Figure 24. SAFESPOT Node Architecture .......................................... 49
Figure 25. Activities to select Conformance Point .................................... 50
Figure 26. SAFESPOT Conformance Points .......................................... 53
Figure 27. Steps to generate SAFESPOT Test Specifications .................... 54
Figure 28. Beaconing Test Specification Overview .................................. 55
Figure 29. SAFESPOT Scope of Testing .............................................. 58
Figure 30. SAFESPOT Test System General Architecture ......................... 59
Figure 31. Basic Components of a TTCN-3 Test System .......................... 62
Figure 32. Interoperability Test Bed Approach ....................................... 64
Figure 33. VANET Functionalities ...................................................... 65
Figure 34. SAFESPOT Interoperability Test Bed ..................................... 65
Figure 35. ATS Example using U2TP ................................................... 81

List of Tables

Table 1. SAFESPOT Test Categories ...................................................... 21
Table 2. Organisations working in Testing ............................................. 28
Table 3. List of Objects for SAFESPOT High Level Architecture ................ 50
Table 4. Interworking Reference Points for SAFESPOT High Level Architecture .................................................. 51
Table 5. List of objects for SAFESPOT Node Architecture ......................... 51
Table 6. Perceptive Reference Points for a SAFESPOT Vehicle ................... 51
Table 7. Programmatic Reference Points for a SAFESPOT Vehicle .......... 52
Table 8. Interworking Reference Points for a SAFESPOT Vehicle .......... 52
Table 9. SF Architecture and TTCN-3 Test System Matching ..................... 63
Table 10. Added value for the SAFESPOT applicants ............................... 69
EXECUTIVE SUMMARY

The objective of this document is to define a certification and testing reference framework for SAFESPOT [1] to produce test specifications and test system specifications for heterogeneous and co-operative networks such as the VANET [22] implemented in the SAFESPOT IP.

In order to increase multi-vendor interoperability in SAFESPOT technology, a well-defined, accurate and unambiguous standard or specification and a systematic testing of the new SAFESPOT products is needed.

The work is divided in two parts. The result of the first part is the document D7.4.3 Part A, which was presented in SAFESPOT meeting in August 2007. That document introduced some of the most relevant certification programs for wireless technologies worldwide.

This document, D7.4.3 Part B focuses on the definition of a systematic testing methodology. The characteristics of co-operative and heterogeneous networks are challenging. On this document existing standards have been reviewed and based on the combination of standard ISO 10746 Reference Model for Open Distributed Processing (RM-ODP) [2] and standard ISO-9646 Conformance Testing Methodology and Framework [3] and ETSI recommendations, a certification framework and a testing framework have been defined.

The certification framework outlines the key actors, the basic certification process and the main steps to start the scheme. The proposed framework relies on the establishment of a Forum (here referred as the Cooperative System Forum) that incorporates the stakeholders of the technology. Furthermore as the SAFESPOT technology is based on some existing wireless technologies, liaison with those bodies and Forums that promote those technologies and rely on their certification schemes is also foreseen.

The key element in the defined testing framework is the “conformance point”. RM-ODP defines “conformance point” as an interaction point to be tested. A critical step in the development of the test specifications and later the test system development is the definition of the conformance points for the system.

Finally, this deliverable will provide details on the various testing procedures in a way that external applicants will have a better understanding on the contents of the various testing and their added value for the pre-implementation activities.

Considering the system complexity, this deliverable should be considered as a document subject to further step of harmonization deriving from the input of the other subprojects or feedback in the specification phase.
1. Introduction

This document is targeted to SAFESPOT partners with the objective to build a general consensus on the certification and testing technology (test specification and test system requirements) that will be necessary for the SAFESPOT certification of telematic components, systems and services. Then, this document will be published largely to support the setting up of the European organisation necessary for the testing and certification of telematic components, system and services focused to the vehicle safety.

In the context of the most common certification regimes, testing is usually understood as the testing of the lower layers of the air-interface (radio characteristics and MAC protocol). However, the evolution of the networks toward heterogeneous and co-operative networks such as the one adopted by SAFESPOT are being proposed as means to support complex distributed systems interconnecting information processing systems. From that point of view onwards, testing needs to evolve and adapt to these new environment requirements and it is need to define a new testing reference framework, more suitable for this kind of networks.

SAFESPOT IP is developing specifications for co-operative systems and applications, which intend to be the origin of new products and services. The SAFESPOT base specifications need to be complemented with SAFESPOT testing specification. SAFESPOT specifications will guarantee the interoperability of co-operative equipment, and test tools that allow testing of conformance to the specifications, and interoperability between equipment developed by different manufacturers.
2. Basic Concepts

For the purposes of the present document, the following definitions and concepts apply:

**Abstract Test Case:** A complete and independent specification of the actions required to achieve a specific test purpose (or a specified combination of test purposes).

**Abstract Test Suite:** A collection of Abstract Test Cases.

**Accreditation:** The act of granting credit or recognition.

**Accreditation body:** An organization that assesses the qualifications and capabilities of certifiers (certification bodies) to operate independently and reliable. It verifies their competence and controls their operation. The accreditation body shall accredit independent entities (certification bodies, test houses, etc.) in accordance with standards and procedures (e.g., ISO 17025).

**Certify:** To state something officially, usually in writing, especially that something is true or correct.

**Certification:** Confirmation that some fact or statement is true. It can be also defined as the procedure through which an official designation is obtained often involving standardized testing.

**Certification body:** A third party that assesses and certifies with respect to standards. It is an impartial organization possessing the necessary competence to operate a certification program.

**Conformance:** Compliance with requirements specified in applicable standards.

**Conformance testing:** Testing the extent to which an IUT is a conforming implementation. Conformance Testing checks whether an implementation is implemented correctly with respect to the requirements stated in the relevant standard or specification. The 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.

**Implementation Under Test:** It is an implementation of one or more OSI protocols in an adjacent user/provider relationship being part of a real open system, which is going to be studied by testing.

**Interoperability:** It is the ability of two systems to interoperate using the same communication protocol.

**Interoperability testing:** Activity of proving that end-to-end functionality between (at least) two communicating systems is as required by the base standard(s) on which those systems are based. Interoperability happens when two or more entities interact by exchanging or sharing information to perform a certain task as in the case of distributed applications and communication protocols. It can be said that two or more entities interoperate if they behave together as expected.
Performance: It is the degree to which a system or component accomplishes its designated functions within given constraints, such as speed, accuracy, or memory usage.

Performance Testing: This testing is conducted to evaluate the compliance of a system or component with specific performance requirements.

Plugfest: It refers to events where engineers get together to test the interoperability of their implementations between each other.

Protocol Implementation Conformance Statement: A statement made by the supplier of an OSI implementation or system, stating which capabilities have been implemented, for a given OSI protocol.

Qualified Equipment: It is a group of one or more devices that have been shown by rigorous and well-defined testing, to interoperate with other equipment.

Self-certification: An official statement that you make about yourself.

Standardization: The process needed to produce Standards.

Standardization body: It is any entity whose primary activities are developing, coordinating, promulgating, revising, amending, reissuing, interpreting, or otherwise maintaining standards that address the interests of a wide base of users outside the standards development organization.

Standards: They are produced by many organizations (or groups of cooperating entities), some for internal usage only, others for use by groups of people, groups of companies, or an entire industry. They form interest groups, that obtain mutual gains in a coordinated action if they ensure a "group-wide" uniformity in a measure (for comparative evaluations), or in a technical reference level of quality or attainment.

System Under Test: It is the real open system in which the IUT resides. It contains the IUT, which usually refers to a single protocol and SUT refers to a group of them.

Test Case: Is the detailed set of instructions (or steps) that need to be taken in order to perform the test.

Test Laboratory: An organization that carries out conformance and/or interoperability testing. This can be a third party, a user organization, a telecommunications administration or recognised private operating agency, or an identifiable part of a supplier organization.

Test Purpose: A prose description of a narrowly defined objective of testing, focusing on a single conformance requirement as specified in the appropriate OSI standard.

Test Suite Structure: Structure of the Test Purposes. Test Purposes are grouped into a logical Test Suite Structure according to suitable criteria (e.g., basic interconnection, error handling, functionality etc.).

Testing: The execution of tests with the intent of providing that the system and application under test does or does not perform according to the requirements specification.
Verdict: A statement of “pass”, “fail” or “inconclusive”, specified in an abstract test case, concerning conformance of an IUT with respect to that test case when it is executed.
PART B.1 – CERTIFICATION FRAMEWORK
3. SAFESPOT Certification Reference Framework (SCRF)

3.1. Introduction

The mission and main objective in developing SAFESPOT technology is to obtain a breakthrough in road safety and increase traffic efficiency. The concept consists of several key components in vehicles and road infrastructure and in addition these should work seamlessly together with the backend systems built.

Only following the SAFESPOT specifications in full will guarantee the interoperability of SAFESPOT equipment. There will be test tools available for product manufacturers to test the conformance to the specifications and interoperability between equipment developed by different manufacturers. However a comprehensive certification framework is highly required to ensure interoperability and conformance of the SAFESPOT technology.

The purpose of the SAFESPOT Certification Framework (SCRF) is to guarantee compliance to SAFESPOT Specifications and that the SAFESPOT devices will be interoperable with other vendor equipments. Because VANET is the cornerstone of SAFESPOT System to achieve interoperability in the whole system, the SCRF certifies VANET standalone systems.

Nevertheless, the need to update and extend the SAFESPOT certification program has to be carefully balanced with the need to preserve continuity in the marketplace and to ensure that equipment already deployed will interoperate with new equipment. The Cooperative System Forum is committed to ensure backward compatibility for all Cooperative System Forum Certified devices and update and enhance the performance of the conformance and interoperability testing including new functionalities on them.

3.2. SAFESPOT Main Certification Players

The SAFESPOT certification process is a process by which the compliance to the SAFESPOT technical specifications is ensured. There is a need to establish the following certification players in SCRF:

- **Cooperative System Forum** is an organized group of companies, laboratories and bodies that are aiming to promote, develop, certify, test and sell products containing SAFESPOT technology. A specific part of the Forum will be in charge of establishing the policies and practices of the SAFESPOT certification. It will also approve the test

---

1 In order to test the concept proposed, the test specifications and test system requirements to be developed with the scope of SAFESPOT will focus on beaconing functionality
specifications to be used for certification. A typical Forum structure is outline in the figure below.

![Cooperative System Forum Diagram](image)

**Figure 1. Cooperative System Forum**

- **SAFESPOT Certification Administrator (SFCA)** is a body inside Cooperative System Forum or a company outsourced to the Cooperative System Forum. The SFCA is in charge of administering the SAFESPOT certification scheme.

- **SAFESPOT Certification Body (SFCB)** is competent and qualified, accredited body that provides part or the whole Certification service to the SAFESPOT Applicant seeking for SAFESPOT Product Certification. It is responsible for checking and approving test plans, as well as checking Technical Documentation against requirements.

- **SAFESPOT authorized trusted Test Laboratory (SFTL)** is a third party test facility authorized by SFCB to perform certification testing according to certification requirements. It is competent and qualified to conduct tests of SAFESPOT Products within its accreditation scope. For this purpose, it needs qualified personnel and required test equipment.

- **SAFESPOT Applicant** can be a car manufacturer, service provider or device supplier (hardware and software) seeking for Certification of its SAFESPOT Product. SAFESPOT Applicant must first become a Cooperative System Forum member.
3.3. SAFESPOT Certification Process

SAFESPOT Applicant must demonstrate that each particular product design complies with the SAFESPOT specifications. To demonstrate that a product complies with the SAFESPOT specifications, a product must pass certification tests at a SFTL and reports and documents must be reviewed by a SFCB. After receiving an official test report from SFTL and a certificate from SFCB, the SFCA grants the right to use the Cooperative System label.

3.3.1. Cooperative System Forum Organisation

In the Cooperative System Forum Organisation there is Cooperative System Forum Board of Directors and the following four working groups for specified areas. The workgroups are formed from the representatives of Cooperative System Forum members.

- SAFESPOT Marketing Working Group
- SAFESPOT Technical Working Group
- SAFESPOT Certification Working Group
- SAFESPOT Legal Working Group

Further part of the Cooperative System Forum Organisation is SFCA and possibly also SFCB. The SFCA can be contracted outside the Cooperative System Forum too, like mentioned before.
3.3.2. Cooperative System Forum Activities

The main player is the Cooperative System Forum. The main activities carried out by the Cooperative System Forum are the following:

− It promotes and authorizes the use of the SAFESPOT technology.
− It defines the compliance requirements for the technology.
− It maintains a web site, where the information related is published.
− It is in charge of establishing and managing all the policies and practices of the SAFESPOT certification scheme.
− It defines the criteria for selection and recognizes SFTL and SFCB and supervises the SFCA as well. Test facilities are selected by Cooperative System Forum.

3.3.3. Overview of the SAFESPOT Certification Process

The goal of the SAFESPOT Certification Requirements and Certification Process is to ensure both the conformance and interoperability of all SAFESPOT branded products. The Cooperative System Forum will define the minimum compliance requirements that apply to all products containing SAFESPOT technology. The products are certified through the SAFESPOT Certification Process according to the certification rules. Figure 3 shows a high-level certification process overview diagram.

![Figure 3. SAFESPOT High-level certification process](image-url)
The detailed description of the SAFESPOT certification process is listed below:

1. The SAFESPOT Certification Administrator (SFCA) will receive a Certification Request, including a summary of the Technical File from the SAFESPOT Applicant member.

2. The SFCA analyses the SAFESPOT Applicant’s request to characterize and classify the SAFESPOT Product(s) to be certified.

3. The SFCA produces a high level Test Plan and Certification Plan.

4. The SAFESPOT Applicant gets into contact with the SFCBs and SFTLs that are recognised for the scope of the work.

5. A Detailed Test Plan is produced by SFTL and approved between SFTL and SFCB. This allows all parties to estimate test efforts, costs and duration and to conclude a contract between the SAFESPOT Applicant and the SFTL.

6. The SAFESPOT Applicant schedules a test date with the SFTL using a method specified by the SFTL.

7. Following the agreed SAFESPOT Detailed Test Plan, the SAFESPOT Applicant sends the SAFESPOT device sample to the SFTL to be tested.

8. The SFTL performs testing according to the Detailed Test Plan. SAFESPOT samples have to be tested as part of a system under test that allows execution of the SAFESPOT Test Cases as specified in the applicable Detailed Test Plan. The certification testing is performed using validated Test Equipment and Test tools in the testing process.

9. The SFTL issues test results and a comprehensive test report to the SFCB and SAFESPOT Applicant.

10. The SFCB evaluates test reports for conformity with the applied standards and test procedures and produces a certificate of conformity to all the parts it has reviewed.

11. When there is a complete certificate of conformity for all the tests, the SFCA grants the right to use Cooperative System label and lists product in the web site.
3.3.4. Test Specifications and associated Test Case Reference List

Based on the SAFESPOT specifications, the testing requirements must be defined. Each product needs to satisfy all applicable SAFESPOT Testing Requirements and the list of applicable SAFESPOT Testing Requirements is called a Detailed Test Plan.
Test documents intend to guarantee that equipment from multiple vendors has been tested the same way, to the same interpretation of the standard, increasing the interoperability of the equipment.

The full set of SAFESPOT Test Documents is composed of four documents: Requirement Catalogue (RQ), Protocol Implementation Conformance Statement (PICS), Test Suite Structure and Test Purpose (TSS&TP) and Abstract Test Suite (ATS). Each document is explained in more detail in section 5.2 of this deliverable.

The SAFESPOT Test Plan is based on TSS&TP document and is generated when each SAFESPOT vendor fills in its own PICS documents including all the features its own equipment supports.

Test Categories will be defined because the certification requirements are evolving over the time and there is a need to control this. Therefore new test cases could be introduced. To satisfy the SAFESPOT Testing Requirements for the different test categories, the testing shall be performed on validated tools or test beds. **Errore. L’origine riferimento non è stata trovata.** summarized the SAFESPOT Test Categories.

The SAFESPOT Test Case Reference List (SFTCRL) lists and uniquely identifies all SAFESPOT Test Cases. Basically, a SFTCRL is a list summarizing all active and upcoming test cases. This SFTCRL may be updated periodically according to new testing and certification requirements. A short description of SFTCRL content is listed below:

- Test Case categorization.
- Test Cases availability and active dates.
- Mandatory test equipment (if applicable).
- Introduction of the test cases.

![Figure 5. SAFESPOT Test Plan Generation](image-url)
Table 1. SAFESPOT Test Categories

<table>
<thead>
<tr>
<th>Test Category</th>
<th>Short name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Mandatory at an accredited SFTL.</td>
<td>This test case is fully validated and commercially available in at least one implementation. The test case is mandatory and has to be performed at an accredited SFTL.</td>
</tr>
<tr>
<td>B</td>
<td>Declaration with evidence</td>
<td>The test case is mandatory and shall be performed by the Member. The Member declares that the IUT design meets the test case’s conformance and interoperability requirements and justifies this declaration by reporting the testing results and test set-up to the SFCB. If the Member does not follow the instructions in the test specification, it must specify how the test was performed.</td>
</tr>
<tr>
<td>C</td>
<td>Declaration without submittal of evidence</td>
<td>The test case is mandatory and shall be performed by the Member. The Member declares that the IUT design meets the test case’s conformance and interoperability requirements, and that the IUT has successfully passed the test case. No evidence is required to be submitted to the SFCB.</td>
</tr>
</tbody>
</table>
3.3.5. SAFESPOT Authorized Trusted Test Laboratory (SFTL) Activities

Certification Testing implies conformance and interoperability testing. The SAFESPOT device has to pass both types of testing in order to get certified and the whole testing process has to be done by a SFTL. To obtain certification, it is required to obtain a “pass” verdict for all test cases in the test plan for the each product under test. SAFESPOT Test Requirements may be satisfied by previously established test results where applicable. If product fails during the conformance or interoperability testing, it will go back to the vendor. Next figure depicts the SAFESPOT Certification Test flow within the overall Certification Process flow.

![SAFESPOT Certification Test Flow](image)

**Figure 6. Detailed Certification Process Flow**

Conformance Test System and Interoperability Test Bed are described in sections 6.1 and 6.2 respectively.

It is important to use a standardized Test Report to report testing results and inform the Applicant and the SFCB. A test report shall contain at a minimum:

- Administrative details of test facility that conducted the tests
- Administrative details of the Member submitting the product under test
- The identifying information of the product under test including HW and SW version numbers
- Reference to SAFESPOT Specification and SAFESPOT Test Specification versions
- Details of test setup
- List of test equipment including version information where relevant
- ICS information relevant for tests performed
- Summary list of all performed test cases with test case identifier and description, date of test and verdict
- Calibration record of SAFESPOT Test Equipment.

Test reports must be signed by the Laboratory duly authorized person. For automatically generated test reports a digital signature could be used if approved by the Cooperative System Forum.
3.3.6. SAFESPOT Certification Body (SFCB) Activities

The SFCB will review test reports and the Technical File to check if all requirements have been meet. The Technical File should include:

- Descriptive name
- Exact model number
- Hardware version number
- Software version number
- SAFESPOT profiles supported

This information will appear on the SAFESPOT Qualified Products List when the product is fully qualified. Product updates and product ranges are covered by one listing. In addition to the identifying information, the SAFESPOT Applicant must also submit to the SFCB sufficient technical documentation to allow the SFCB to determine whether the product should be qualified. This documentation includes at a minimum:

- Preliminary user manual or user guide
- Functional block diagram and technical description

Once the SFCB has reviewed all the documents, it will issue a certificate.

As well, Declaration of Compliance (DoC) to SAFESPOT specifications and associated documentation must be provided by the Applicant. The Cooperative System Forum Member shall fill in and sign a Declaration of Compliance (DoC) proforma as provided by Cooperative System Forum. In the DoC the Cooperative System Forum Member shall declare that:

- The product complies with the SAFESPOT Specification referenced
- The product implements the referenced design
- The product meets the Testing and Documentation part of the SAFESPOT Qualification Requirements.

The DoC shall list the following:

- Design ID or version number
- Product identification
- PICS pertaining to product/design
- SAFESPOT Specification identifier
3.3.7. SAFESPOT Certification Authority (SFCA) Activities

An important player in the certification process is the SAFESPOT Certification Administrator (SFCA). The information about all SAFESPOT certified products will be publicly available though the official SAFESPOT web site that is maintained by SFCA. Once the SFCBs have certified all parts of the SAFESPOT product the SFCA grants a license to use Cooperative System label. Next figure illustrates the Cooperative System label proposal.

![Cooperative System label Proposal](image)

Figure 7. Cooperative System label Proposal

3.4. Certification Scheme Implementation

3.4.1. Tasks to build a certification scheme

The following identifies the tasks to be completed in order to build a certification scheme:

1. Establish the Cooperative System Forum.
2. Develop a web site for repository of documents and certificates.
3. Develop certification Policy documents.
4. Develop Test Specifications
5. Select a SAFESPOT Certification Administrator.
6. Select SAFESPOT Test Laboratories and SAFESPOT Certification Bodies.
7. Validate SAFESPOT Test Tools.
8. Run the certificate scheme.

3.4.2. Criteria for selecting SAFESPOT Test Facilities (SFTL)

A pre-requisite for recognition as SFTL is that the facility is accredited for SAFESPOT Conformance and Interoperability Testing. Accreditation for this requires that the facility has been successfully assessed, regarding its technical capabilities and its quality according to the SAFESPOT Test Facility
Evaluation Criteria. A test facility may receive such accreditation through having evaluated by Contracted Accreditation Organizations, which are party to have a signed agreement with SFCB to perform assessment and accreditation services against the SAFESPOT Test Facility Evaluation Criteria.

The formal process of SFTL recognition is an activity separate from the accreditation. The SFTL Recognition is based upon that the facility:

- meets the accreditation pre-requisite, and
- the facility is a Member or affiliated with a Member, and
- no other issues become known that would preclude SFTL Recognition
PART B.2 – TESTING FRAMEWORK
4. Testing Introduction

The primary task of any testing activity is to collect information in order to help vendors make informed decisions related to the global quality of the application being tested.

So, costs of network technology and implementations also continue increasing, and testing can actually help to reduce cost and implementations time, due to the fact that a good testing methodology can localize errors quickly.

Testing is the way by which either a product or a prototype can be checked to be compliant to the technical requirements (conformance, interoperability or performance requirements) specified for such a product or prototype.

The execution of testing has two perspectives: on one hand informal testing that can be executed internally in a company, and on the other hand, formal testing that is usually associated to a certification process and is fulfilled by a test laboratory accredited compliant to ISO/IEC 17025 [4].

When a product is not tested, it will hardly interoperate properly with other product implementations and there shall be high risk of fatal failures. Nevertheless, if a product is successfully tested then there will be several advantages from the company’s point of view:

- Savings costs and development time.
- Ensuring products are compliant to a standard or specification.
- Increasing the likely of successful interoperability with other product implementations.
- Offering a good impact in the final user because if the product fails the confidence in the company and in the product will decrease.

Testing increases the probability that different products implementations interoperate properly.

Moreover, different organisations and standardization bodies are working in the specification of testing methodologies and procedures using a homogeneous testing language.
Next table summarizes some of the most important organisations working in this area.

Table 2. Organisations working in Testing

<table>
<thead>
<tr>
<th>Organisations</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANSI</td>
<td>American National Standards Institute</td>
</tr>
<tr>
<td>CENELEC</td>
<td>Comité Européen de Normalisation Electrotechnique</td>
</tr>
<tr>
<td>ETSI</td>
<td>European Telecommunication Standard Institute</td>
</tr>
<tr>
<td>IEEE</td>
<td>Institute of Electrical and Electronical Engineers</td>
</tr>
<tr>
<td>IETF</td>
<td>Internet Engineering Task Force</td>
</tr>
<tr>
<td>ISO</td>
<td>International Standard Organizartion</td>
</tr>
<tr>
<td>ISTQB</td>
<td>International Software Técnicas de medida Qualifications Board</td>
</tr>
</tbody>
</table>

Techniques of testing are a wide field and not only are focused on telecommunication testing concerning to protocols or radiofrequency. Because of the heterogeneity of SAFESPOT, the diversity of techniques of testing that can be applied to in this vehicular environment is really important. This environment could cover software testing, performance testing and event robustness testing as well. Also present, a new technique of testing is becoming more relevant during the deploy either of a network or a complex heterogeneous system, this type of testing is called Field Operation Testing (FOT), however this technique of testing is out of the scope of SAFESPOT.

This section intends to provide a state of the art of several methods of testing such as Plugfest Events, Conformance Testing, Interoperability Testing, Performance Testing and Virtual Testing. Next sections describe these methods of testing and explain the objectives, specifications, benefits and the basic testing architecture of each method.
4.1. Plugfest Events

A Plugfest event is an open technical test event where teams of engineers from different companies gather to test their implementations against each other in an open environment. The duration of time of these events is usually short (1-5 days).

The purpose of a Plugfest Event is to bring all implementation from a particular technology together at a single location to validate components, software and platforms based on the specification developed.

This event aims at improving interoperability by:

- Creating an opportunity for companies to test their prototypes against a standard with their partners and competitors.
- Enhancing the quality of specifications.
- Speeding up the standardization process.
- Reducing time to market.
- Supporting the deployment of the technology.

Plugfest events are part of the standardization process at an early stage of the technology. They are optional events, but it is recommended to arrange them, similarly as it was adopted in many other different technologies such as Bluetooth, GSM, WiMAX, IPv6, IMS, etc: 6th SIM/HANDSET Plugtest, 3rd Mobile WiMax Plugfest, SIGTRAN, 3G & Mobile Broadband User Experience, IPv6 Interoperability 4th IPv6 Plugtest, 5th IPv6, Plugtest Bluetooth UnplugFest#1, Bluetooth UnplugFest#3, Bluetooth UnplugFest#6, Bluetooth UnplugFest#9. Next figure depicts some Plugfest events organized all over the World. Several Plugfests have been arranged by ETSI, WiMAX Forum and other organizations for these technologies.
The specification and implementation of the Plugfest is also considered a transversal activity in the process of developing and specifying a well-defined specification of the technology.

Plugfest events offer some benefits from companies and technology point of view:
Each participant who brings an implementation of the technology to the Plugfest Event is allowed to test his or her own implementation against other during the event and note the results of the testing.

Enhancing the quality of a standard and implementations because:

- Plugfest events find errors or ambiguities in standards and thereby help improve the stability and quality of standards, test specifications and final products.
- There is a close interaction and cross fertilization between the people drafting and those implementing the standard.
- Engineers find errors in their implementation that they would not have found otherwise.
- Implementations are tested against a large number of implementations from other companies.
- Improving the understanding of system and test specifications concentrated all the efforts in test activities.
- Fast feedback into the standards process. The earlier bugs are discovered, the cheaper it is to correct them.
- Publicity of Plugtest raises awareness in the market about the standard.
- Meeting other developers familiar with the standard by making personal contact with developers of other companies and getting to know them, questions and answers that one does not want to post on a public mailing list can be exchanged after the event.
- Overall, Plugfest save companies and the industry lost of time and money. Engineers rank the usefulness of these events consistently very high. In addition to discovering problems, these events allow testing with a wide range of vendors in a short amount of time. According to participants, much time and money is therefore saved as opposed to one-on-one scheduling.

The main goal of Plugfests is Interoperability, so Plugfests are mostly considered as interoperability events. Although each Plugfest event will have a specific technical goal, the general goals for any Plugfest event are usually the following:

- To remove ambiguities and misinterpretations in the currently developing standards, as well as in their implementations.
- To provide feedback into the standardization process.
- To improve and increase the interoperability between different implementations by debugging the standard and companies' implementation at an early stage.
- To detect implementation errors or bugs because the implementations tested are usually prototypes.
As conclusion, this final purpose is very similar to the purpose of Test Sites in SAFESPOT where different applications, technologies, platforms, etc. from SAFESPOT Subprojects are validated and the impacts and end-user acceptances are also evaluated. This test sites reproduce several vehicular scenarios characterizing real life driving contexts where Highway, rural roads, test tracks, tunnels will be equipped as well as several vehicles. Finally, different test sites are also a means to demonstrate the interoperability of implemented applications.

![SAFESPOT Test Sites](image)

**Figure 10. SAFESPOT Test Sites**

### 4.2. Conformance Testing

Conformance testing checks whether an implementation is implemented correctly with respect to the requirements stated in the relevant standard or specification.

The concepts of this type of testing are extracted from the standard ISO/IEC 9646 Conformance Testing Methodology and Framework [3] which specifies a general methodology for conformance testing of products to standards.

This type of testing is applied to different branch of telecommunication mainly focused on protocol testing (layer-by-layer testing), however this testing is perfectly applicable in radiofrequency (RF) testing (Figure 11). The first example of figure below represents a usual protocol testing, where one mobile phone must send an ‘Error Message’ at least 3 ms after receiving a ‘Erroneous Request Message’ generated by the Test System. The second example illustrates a RF testing where a hardware board, which implements the physical layer of a technology, generates the carrier frequency in the band specified in the standard. These examples introduce the Test System and the Implementation Under Test (IUT) concepts that will be explained in following sections.
The architecture of conformance testing is usually composed of one test system and the Implementation Under Test (IUT) such as Figure 12 shown.

Figure above introduces two important concepts in conformance testing: Test System and Implementation Under Test.

- The IUT is the implementation to be tested. The IUT is usually one or more protocols in an adjacent user/provider relationship being part of a real open system, which is going to be studied by testing. The most illustrative example is the implementation of an OSI (Open Systems Interconnection) layer such as Figure 13 depicts. In this example, the IUT is the layer 3 (usually network layer) in an OSI layer.
A Test System is a controlled environment that contains hardware components, measurement instrumentation, simulators, software tools, and other support elements needed to conduct a test in a right way in order to execute the tests specified. The test system implementation must be able to have full control over the IUT’s behaviour. Another important aspect is the configuration of the Test System because the configuration can be dependent of the type of measure and the goals of the test. As example, one test system could be composed of one oscilloscope, one spectrum analyzer and the software in charge of controlling these devices, as Figure 14 illustrates, and some tests only need the oscilloscope to execute the measurements (Configure 1), and other tests only need the spectrum analyzer for the same purpose (Configure 2).

If conformance requirements are comprehensively specified, conformance testing is able to accomplish the following for a given component:

- Giving a high-level of confidence
- Determining whether the behaviour of an implementation is compliant to the requirements laid out in its base specification including the full range of errors and exception conditions which can only be induced or replicated by dedicated test equipment.
- Test Execution can be automated and repeated, but in this case, tests are executed in an ideal environment, so conformance testing owns a high IUT control and observability.

On the other hand, conformance testing has the following limitations:

- Does not prove end-to-end interoperability of functions between the two similar communicating systems. This means that one IUT, which is tested and is compliant to a specific standard, will not always be interoperable with other devices base on the same standard. This is often a specification problem rather than a testing problem
- Does not test the user’s ‘perception’ of the system because conformance testing only tests individual components.
- Standardised conformance tests do not include proprietary ‘aspects’, this means that does not prove the operation of proprietary features, functions, interfaces, and systems that are not in the public domain.
However, these proprietary facilities may be exercised indirectly as part of the configuration or execution of the conformance tests.

Figure 14. Test System components

At the beginning, one criticism of this testing is that it was expensive, especially in test system developed for radio environments such as GSM or DECT. At present, this perception about conformance testing is changing, because different organizations, such as ETSI, started to work in this aspect in order to improve the process of specifying test specifications through an adequate methodology and writing test cases based on a specific and standardized test language such a TTCN-3 (Testing and Test Control Notation) [11].

TTCN-3 provides both a standardized test language for describing test cases and a test architecture for developing test systems. Different technologies are using TTCN-3 as testing language; some examples of these technologies are WiMAX and IPv6. Protocol testing is the main area of application in which TTCN-3 is being applied, but other areas (software, system, etc.) and other testing techniques (interoperability, robustness, etc.) are starting to use TTCN-3 as test language. Section 6.6.1 describes in more detail this language.
4.3. Interoperability Testing

4.3.1. Basic Interoperability Testing

Interoperability testing is usually considered as the next step in the logical process of testing. There are several definitions of interoperability and the most extent definition is that interoperability is the ability of two systems to interoperate using the same communication protocol or technology.

However, interoperability has other definition depending on the context, for example interoperability is defined as the ability of equipment of different manufacturers (or different systems) to communicate together on the same infrastructure (same system), or on another while roaming in a Next Generation Networks (NGN) context and interoperability is defined as the ability of two or more systems or components to exchange data and use information in a 3GPP context [6].

All these definitions have a common aspect from a user point of view and to some extent; interoperability shows that a system works properly from an end-to-end functionality point of view.

Based on these definitions, interoperability testing is the activity of proving that end-to-end functionality between (at least) two communicating systems is as required by the base standards on which those systems are based; and it is very important that interoperability testing is carried out in accordance with a comprehensive and structured suite of tests as ETSI EG 202-237 [7] defined.

Moreover, ETSI EG 202-237 defines a general approach and architecture of this technique of testing. The implementation of this architecture is usually known as TestBed and test operators can execute the interoperability test cases in this TestBed. Next figure illustrates the basic interoperability architecture.

![Figure 15. Generic Architecture for Interoperability Testing](image-url)
Several concepts are introduced in figure above: Qualified Equipment (QE) and System Under Test (SUT).

- Qualified Equipment is a grouping of one or more devices that have been verified and validated, by rigorous and well-defined testing, to interoperate with other equipments and against the System Under Test in order to demonstrate the SUT’s interoperability.

- System Under Test is the system to be tested and must interoperate with the QE and must be able to work according to standards in more real environments emulated in a test bed with the adequate conditions. Making an example based on OSI Layer as Figure 5 shows in previous conformance testing section, SUT could be composed of one or more OSI layers.

- Equipment Under Test is composed of one or more SUTs and this EUT must be able to interoperate to QEs. An EUT could be considered as the high level device under test that is under the conditions to be tested and it would be tested the whole functionality of the different SUTs included in the EUT and sometimes it is possible to say that the integration of the whole system is being tested.

There are several differences between interoperability and conformance testing. In interoperability testing the EUT comprises the SUT, and one or more pieces of IUTs, as Figure 16 illustrated, and in this type of testing the SUT and the EUT are the main actors of the testing, although different IUTs can take part of the SUT. However, it is very important to be sure that the IUTs are compliant with the standards in which are based on. This can be summarized with the next sentences extracted from ETSI: test the component first, then test the system. In this case, the IUT plays the role of component, and the SUT or EUT plays the role of system.

Next figure illustrates the concepts of IUT, SUT and EUT in a vehicle composed of one Telematic Control Unit (TCU), and this TCU implements different applications and protocols.
Interoperability testing has the following characteristics:

- Interoperability testing tests a complete device or a collection of devices.
- It shows that several devices are able to interoperate but within a limited scenarios.
- It tests the end-to-end functionality of the system testing the system as a whole from a user’s point of view, but not how.
- It gives a high-level of confidence that devices will interoperate with other devices based on the same technology.
However, interoperability testing has some disadvantages:

- It does not prove that a device is conformant. Nevertheless, devices may interoperate even though they are not compliant with the base specification.
- It cannot test error behaviour or unusual scenarios, so that invalid conditions may need to be forced and this technique of testing has a lack of controllability.
- It is difficult to automate, but not impossible. These tests usually use more time to be prepared than executed.

Initially, interoperability testing could not assure full interoperability of a SUT against others implementations that have not been tested; this means, if a SUT-A interopes with a SUT-B and SUT-B interopes with a SUT-C; and all of them have been tested, it does not necessary mean that SUT-A interoperates with SUT-C.

In order to reduce this fact, several organisations are currently working in producing interoperability guidelines to be applied in the entire standards development of a technology. From this point of view, interoperability is one of the main objectives of ICT standardization. ETSI is one of these organisations and is proposing several initiatives and procedures to assure interoperability in the next generation of standards [6].

**4.3.2. Advanced Interoperability Testing**

Conformance and interoperability are both important and useful approaches to the testing of standardized protocol implementations although it is unlikely that one will ever fully replace the other. Conformance testing is able to show that a particular implementation complies with all of the protocol requirements specified in the associated base standard. However, it is difficult for such testing to be able to prove that the implementation will interoperate with similar implementations in other products. On the other hand, interoperability testing, especially when performed with two devices from different vendors, can clearly demonstrate that two implementations will cooperate to provide the specified end-to-end functions but cannot easily prove that either of them conforms to the detailed requirements of the protocol specification.

For example, when you move up a system protocol stack the emphasis should change from conformance to interoperability testing. Lower layer protocols, such as PHY or MAC, are usually candidates to be tested through a conformance testing perspective; however testing services, applications or systems are usually focused on interoperability testing environments.

Nowadays, combining interoperability testing with conformance testing is becoming increasingly popular and is known as Advanced Interoperability Testing. This technique is possible because both methods of testing are complementary, so conformance testing in conjunction with interoperability testing provides both the proof of conformance and the guarantee of
interoperation. Conformance testing can be considered as a prerequisite for interoperability testing in many certification schemes.

This combination is the best probability of getting interoperability between products based on the same technology.

A first approach of architecture for advanced interoperability testing is shown in Figure 18. This combination permits a complete interoperability testing to be undertaken while limited protocol conformance monitoring takes place through a sniffer. The sniffer is part of the architecture in order to allow the test operator to check whether protocol between the SUT and QE is conformed to the appropriate standard(s).

![Figure 18. Combining Conformance and Interoperability Testing](image)

4.4. Performance Testing

Performance is the degree to which a system or component accomplishes its designated functions within given constraints, such as speed, accuracy, or memory usage [17].

There are many definitions of performance testing, each one for the context where tests carries out.

In general, performance testing is a rigorous type of testing oriented to determine the responsiveness, throughput, reliability, capacity and scalability of a system under test working at a given and specified load in realistic conditions. Performance testing can also be used as a diagnostic aid in locating communication bottlenecks, which heavy decrease efficiency of any system, component or software program.

A more formal definition from IEEE says [17]:

“Performance testing is a kind of testing conducted to evaluate the compliance of a system or component with specified performance requirements.”
It is necessary to have a strong knowledge of the system under test extracting its basic characteristics in order to define the main testing goals and test them correctly.

Basically, performance testing explores several system qualities, most important of them could be simplified to:

- **Speed**: by testing the speed of a system or component it is possible to know if the system responds quickly enough when working at a given load. Response time or delays could be examples of that kind of measurements.

- **Capacity**: in the context of performance testing, capacity means the necessity to know if a given system or software program is adequately dimensioned. The number of users is usually a very common measure.

- **Scalability**: it tries to ascertain if a system or component could grow if the traffic, user number or any other qualitative parameter increases.

- **Stability**: other desired quality of any system is to keep on the stability of the system when this one is working at heavy conditions. So, stability ascertains if the system works correctly under heavy load.

The mainly reasons to do performance testing could be resumed in:

- It enables to estimate the performance characteristics of a system under test in an early production phase and evaluate if it would be suitable to adapt the performance testing strategy.

- It evaluates the adequacy of current capacity levels.

- It determines the criteria of acceptability of the performance metrics.

- It compares different system configurations to determine which works better.

- It verifies that the system exhibits the desired performance characteristics.

- It analyzes the behaviour of the system at different load levels.

- It identifies bottlenecks which heavy reduces the efficiency of the system under test.

- It provides information related to speed, scalability and stability of a given system or component in an early design phase.

The selected type of performance testing depends on what type of results you want to achieve. Performance testing is divided into four different types:

- **Load tests** are used to verify application behaviour under normal and peak load conditions, so that kind of tests are conducted to verify if our system is able to meet the performance requirements. That kind of test enables to measure metrics like throughput, response times and latency. It is possible to know the break point of the system by going to the peak load condition.
Stress tests are useful when looking for any kind of problem on our system or application which only appears when working at high load conditions. It provides an estimate about system performance before causing general failures or unexpected behaviours. Relevant information by doing that kind of test, like the robustness of the system, how well the system recovers from a high load or how many concurrent users can be handled for the system, for example.

Capacity tests provide information about how much load can be tolerated by a system. The mayor goal is to show how far a given system can scale to adapt a specific set of circumstances.

The more usual metrics that any performance testing should take attention on systems are listed below:

- Capacity is the maximum workload it can handle without violating the performance acceptance criteria parameters. That limit values can’t be exceeded.
- Latency represents any delay that increases the response time of a given system or network well above the desired value. So, it is a measure about the time to finish a request to the system.
- Response time represents how fast the system responds to a given request. A possible method to get this parameter is shown in Figure 19. This method consists on placing two networks analyzers on the two extremes of the network under test, and then sends a ping packet. After that, it is necessary to wait until the arrival of the packet and then calculate the elapsed time.

![Figure 19. Response Time Measurement](image)

- Scalability involves the ability of a system to grow and then handle a higher workload by adding additional elements.
- Stability is used to verify the behaviour of a system while working under high load conditions.
Throughput represents the number of units of works that can be handled per unit of time. It is usually useful to estimate the theoretical maximum data transfer rate. Typically, testers use the FTP protocol and the Kbps (Kilobits per second) unit to predict the throughput by analyzing the data transfer rate between two entities over a period of time.

Bottlenecks are the location in software or hardware where the performance is lower than that in other parts of the system and thereby limits the overall throughput.

Performance testing is typically understood as one kind of “black-box” testing or functionality testing. That refers to the fact that performance testing generally ignores the internal implementation of the system under test and focuses on the generated outputs due to the given excitation and conditions.

Next figure shows an architecture used for performance testing.

![Figure 20. Generic performance testing architecture](image)

The architecture is one of the most used in performance testing. It is usually possible to get a software system to make the measurements, and that software is usually implemented on several and distributed servers.

- Controller manages and controls the behaviour of the monitors, load generators and environment emulators as well as executes different performance test cases. The controller is handled by the test operator.
- Load Generator receives the workload information from the controller as input, and generates the requests towards the SUT.
− Environment Emulator is an entity which mission is to emulate realistic conditions during the execution of a test case.

− Monitors are special software that supervise and control the behaviour of other systems, such as its performance. When the test ends, its time to analyze the results obtained and retest if the desirable level in performance metrics have not been obtained.

The more relevant characteristics of performance testing could be resumed to:

− Determine the speed, scalability and stability characteristics of a system, providing an input used to make consistent decisions.

− Focus on determining if the user of a given system is satisfied with its performance.

− Identify mismatches between performance related expectations and real world.

− Support tuning and performance refinement providing an structures testing methodology.

− Provide information about how much workload could be handled for the system.

However, performance testing has some disadvantages:

− Tests are quite difficult to create.

− Performance testing tools have usually a great impact on the own system, so empirical results could not be totally exact. Then, it is very convenient to include that effect on the system calibration.

− It is complex to correctly simulate the users’ behaviour and environment conditions.

### 4.5. Virtual Testing

During prototyping and developing phases, much of system’s design and implementation shall be done in software or in prototype platforms. Only when the development and debugging is totally completed, the design should become reality in a real platform. If testing could be conducted in parallel during design of protocol prototypes in software or not real platforms, then product development and testing would be cheaper and quicker.

The development of advanced and complex protocols, interfaces and means of communications require expensive and complex test equipments. For instance, in wireless communication and protocol, the test system must implement the radio communication channel to transport the test stimulus – response signals to the IUT and back. This is usually done with instruments, which may be expensive or not available at the early stages.

Main challenges are detected in test systems at the early stages if they are only based on the real platform:
- The radio based test tool may not be available at all for commercial reasons, may become available late for commercial and technical reasons and may be excessively expensive because of real costs combined with limited number of potential vendors.

- The process of assuring the correct execution of test cases in a real platform, it may be possibly available later than desired for in-house prototype debugging and for providing feedback and making corrections to a standard.

- Before testing against conformance test tools or other implementations, each company individually has to spend a lot of effort on in-house testing/debugging schemes.

For these reasons, and mainly focused on conformance testing, ETSI [8] developed the concept of virtual testing that is a basically a test system that replaces the communication layer by a virtual communication tunnel. Figure 21 shows the virtual tester approach.

Thus, one reason for a virtual tester is to test protocols destined for an expensive interface in their prototyping and development stage. For instance, the tester could use a substitute wire interface for the lower transport layers. Another reason for a virtual tester is to allow conformance testing during early stages of design and development.

Consequently, virtual testing methodology has the following:

- Make test cases validation faster and cheaper without using expensive test equipments.
- Enable sharing efforts (costs) on protocol conformance testing and company in-house testing/debugging.
- Make product testing available at the early technologies with alternative solutions.
- Facilitate development of final test tools.

Protocol development/debugging and the writing/running tests can occur in parallel thereby reducing the time-to-market and reducing costs because prototype and test debugging occur during the development process.

Finally, Virtual Tester could be defined as a mock-up test system and for these reason, SAFESPOT can benefit from virtual testing concepts in the development of the SAFESPOT Test System (SFTS) mock-up from conformance point of view because several platforms are still under development and the final platform is not decided yet.
5. SAFESPOT Testing Reference Framework (STRF)

SAFESPOT architecture defines a co-operative distributed telematics system using a communication network between vehicles and infrastructures. This implies that SAFESPOT is a complex distributed system interconnecting information processing systems and exchanging information over networks through communication media, different data features and using heterogeneous wireless networks.

This means that SAFESPOT testing requires new challenges and approaches to cover the more relevant functionalities of the whole SAFESPOT system with the objective of increasing the likelihood of interoperability in SAFESPOT products.

An innovative methodology and principles of SAFESPOT testing is the cornerstone of the SAFESPOT Testing Reference Framework (STRF) and it is relied on the combination of the following standards:

- ETSI EG 202-237 - Generic approach to interoperability testing [7]

The complexity of the SAFESPOT architecture including the convergence of fixed and mobile requires an inspection methodology and tools to verify interoperability of network elements.

A clearly defined, univocal and unambiguous SAFESPOT Testing Reference Framework is required. This framework is clearly structured into three different phases:
- **Phase 1: SAFESPOT Elements to be tested / certified.** This phase is based on ISO 10746-1 with the main objective to select the functionalities to be tested / certified in SAFESPOT.

- **Phase 2: SAFESPOT Test Specification.** Once the functionalities to be tested / certified have been decided, it is necessary to extract from the SAFESPOT deliverables the SAFESPOT test specification and to decide the test strategy to be applied for.

- **Phase 3: SAFESPOT Test System.** It is final phase, and covers the implementation and validation of the test system based on the test strategy decided.

![SAFESPOT Phases in STRF](image)

Next sections explain each phase in more detail.

### 5.1. Phase 1: Elements to be tested / certified

The objective of this phase is to identify the potential SAFESPOT interfaces and functionalities, from a high level point of view, to be tested or certified. Because of the decision of this potential objects to be tested is from a high level point of view, the input for this phase is the SAFESPOT Architecture defined in SP7 – SCORE (Figure 22) and the architecture of a SAFESPOT Node (Figure 24).
In order to handle and help the uncertainty of potential SAFESPOT products, conformance assessment and conformance testing will be derived from the ISO 10746-1: Reference Model Object Distributed Processing (RM-ODP). SAFESPOT architecture defines a “reference point” as a potential interaction point to become tested. Finally, the reference points selected for testing are named “conformance points”.

There are four classes of reference points:

- **Programmatic reference point**: interface which is realized through a programming language binding, for instance databases, SW applications, etc.

- **Perceptive reference point**: a reference at which there is some interaction between the system and the physical world, for instance HMI, GUI, etc.

- **Interworking reference point**: a reference point at which an interface can be established to allow communication between two or more systems (or objects), for instance OSI protocol and communications between SAFESPOT nodes.

- **Interchange reference point**: a reference point at which an external physical storage medium can be introduced into the system, for instance hard disk, card, etc.

---

2 Conformance is a relation between a specification and a real implementation, for instance of a product. Conformance is assessed using conformance testing; its behavior is evaluated (stimulus – responses) at specific (interaction) points.
In general, reference points of the class interworking are usually those most susceptible to become conformance points.

SAFESPOT SP7 subproject clearly identifies the reference points in the architecture. In order to obtain the functionality to be tested in SAFESPOT project, the following activities are proposed to be executed in the SAFESPOT Architecture:

1. Enumerate the objects that participate in each SAFESPOT interface.
2. Enumerate and identify the reference points defined in the subproject.
3. Decide which reference points become conformance points.

Next Figure 25 represents these activities.

**Figure 25. Activities to select Conformance Point**

![SAFESPOT Architecture](image)

These activities are initially applied over SAFESPOT High Level Architecture (Figure 22). The list of objects identified is shown in next table.

**Table 3. List of Objects for SAFESPOT High Level Architecture**

<table>
<thead>
<tr>
<th>Main Objects</th>
<th>Composed of</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAFESPOT Vehicle</td>
<td>SAFESPOT Vehicle Node</td>
</tr>
<tr>
<td>SAFESPOT Road Site Unit</td>
<td>SAFESPOT Road Side Node</td>
</tr>
<tr>
<td>Non SAFESPOT Vehicle</td>
<td></td>
</tr>
<tr>
<td>Non SAFESPOT Road Site Unit</td>
<td></td>
</tr>
<tr>
<td>Road Operator</td>
<td></td>
</tr>
<tr>
<td>Safety Centre</td>
<td></td>
</tr>
</tbody>
</table>

Next step is to identify the reference points between these object. Next table classifies the references point detected in SAFESPOT High Level Architecture based on objects detected previously. All reference points are interworking.
Table 4. Interworking Reference Points for SAFESPOT High Level Architecture

<table>
<thead>
<tr>
<th>ID Ref</th>
<th>RP Name</th>
<th>Object</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>RP-SF-HLA-I-001</td>
<td>RP I-C</td>
<td>Safety Centre</td>
<td>Road Operator</td>
</tr>
<tr>
<td>RP-SF-HLA-I-002</td>
<td>RP V-C</td>
<td>SAFESPOT Vehicle</td>
<td>Safety Centre</td>
</tr>
<tr>
<td>RP-SF-HLA-I-003</td>
<td>RP V-V</td>
<td>SAFESPOT Vehicle</td>
<td>SAFESPOT Vehicle</td>
</tr>
<tr>
<td>RP-SF-HLA-I-004</td>
<td>RP V-I</td>
<td>SAFESPOT Vehicle</td>
<td>SAFESPOT Road Side Unit</td>
</tr>
<tr>
<td>RP-SF-HLA-I-005</td>
<td>RP I-I</td>
<td>SAFESPOT Road Side Unit</td>
<td>Road Operator</td>
</tr>
</tbody>
</table>

The reference points selected are RP-SF-HLA-I-003 and RP-SF-HLA-I-004 in order to improve and test the interoperability between SAFESPOT Nodes.

The next step is applied again these activities over a SAFESPOT Vehicle node and a SAFESPOT Road Side Node (Figure 24) to find the functionality to be tested and certified.

The list of objects identified is shown in table below.

Table 5. List of objects for SAFESPOT Node Architecture

<table>
<thead>
<tr>
<th>SAFESPOT Vehicle</th>
<th>SubProject</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main Objects</td>
<td></td>
</tr>
<tr>
<td>Applications</td>
<td>SP4 - SCOVA</td>
</tr>
<tr>
<td>LDM</td>
<td>SP3 - SINTECH</td>
</tr>
<tr>
<td>Message Management</td>
<td>SP3 - SINTECH</td>
</tr>
<tr>
<td>In-vehicle Data Processing &amp; Fusion</td>
<td>SP1 - SAFEPROBE</td>
</tr>
<tr>
<td>In-vehicle Sensing</td>
<td>SP1 – SAFEPROBE</td>
</tr>
<tr>
<td>SAFESPOT Road Side</td>
<td>SubProject</td>
</tr>
<tr>
<td>Main Objects</td>
<td></td>
</tr>
<tr>
<td>Applications</td>
<td>SP5 – COSSIB</td>
</tr>
<tr>
<td>LDM</td>
<td>SP3 – SINTECH</td>
</tr>
<tr>
<td>Message Management</td>
<td>SP3 – SINTECH</td>
</tr>
<tr>
<td>In-Road Data Processing &amp; Fusion</td>
<td>SP2 – INFRASENS</td>
</tr>
<tr>
<td>In-Road Sensing</td>
<td>SP2 – INFRASENS</td>
</tr>
<tr>
<td>SAFESPOT External</td>
<td></td>
</tr>
<tr>
<td>Human User</td>
<td></td>
</tr>
</tbody>
</table>

The reference points identified in a SAFESPOT Vehicle are perceptive, interworking and programmatic. Next table lists each reference point.

Perceptive Reference Points

Table 6. Perceptive Reference Point for a SAFESPOT Vehicle

<table>
<thead>
<tr>
<th>ID Ref</th>
<th>RP Name</th>
<th>Object</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>RP-SF-V-PER-001</td>
<td>RP V-H</td>
<td>SCOVA Application</td>
<td>Human User</td>
</tr>
</tbody>
</table>
Programmatic Reference Points

Table 7. Programmatic Reference Points for a SAFESPOT Vehicle

<table>
<thead>
<tr>
<th>ID Ref</th>
<th>RP Name</th>
<th>Object</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>RP-SF-V-PRO-001</td>
<td>RP V-4</td>
<td>SCOVA Application</td>
<td>LDM</td>
</tr>
<tr>
<td>RP-SF-V-PRO-002</td>
<td>RP V-6</td>
<td>SCOVA Application</td>
<td>Message Management</td>
</tr>
<tr>
<td>RP-SF-V-PRO-003</td>
<td>RP V-7</td>
<td>SCOVA Application</td>
<td>In-Vehicle Data Processing &amp; Fusion</td>
</tr>
<tr>
<td>RP-SF-V-PRO-004</td>
<td>RP V-2</td>
<td>LDM</td>
<td>In-Vehicle Data Processing &amp; Fusion</td>
</tr>
<tr>
<td>RP-SF-V-PRO-005</td>
<td>RP V-3</td>
<td>Message Management</td>
<td>In-Vehicle Data Processing &amp; Fusion</td>
</tr>
<tr>
<td>RP-SF-V-PRO-006</td>
<td>RP V-1</td>
<td>In-Vehicle Sensing</td>
<td>In-Vehicle Data Processing &amp; Fusion</td>
</tr>
<tr>
<td>RP-SF-V-PRO-007</td>
<td>RP V-5</td>
<td>LDM</td>
<td>Message Management</td>
</tr>
</tbody>
</table>

Interworking Reference Points

Table 8. Interworking Reference Points for a SAFESPOT Vehicle

<table>
<thead>
<tr>
<th>ID Ref</th>
<th>RP Name</th>
<th>Object</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>RP-SF-V-INT-001</td>
<td>RP V-V</td>
<td>Message Management</td>
<td>Message Management (In a SAFESPOT Vehicle)</td>
</tr>
<tr>
<td>RP-SF-V-INT-002</td>
<td>RP V-I</td>
<td>Message Management</td>
<td>Message Management (In a SAFESPOT Road Side Unit)</td>
</tr>
</tbody>
</table>

The reference points identified in a SAFESPOT Road Side Unit are the same that in a SAFESPOT Vehicle because of SAFESPOT Nodes present a symmetric architecture.

The final decision was taken at SP7 Malaga Meeting on November 13th 2007 by all SP7 members. Finally, the conformance points selected were RP-SF-V-INT-001 and RP-SF-V-INT-002. This conformance points covers all functionality related to VANET and interoperability between SAFESPOT Nodes. Figure 26 shows an overview of the system indicating the conformance point for SAFESPOT.

Finally, it is important to highlight that whole VANET functionality will not be tested by the SAFESPOT Test System mock-up. At a first stage, the main functionality to be tested will be Beaconing.
5.2. Phase 2: SAFESPOT Test Specification

Once the elements to be tested / certified have been selected, it is necessary to produce a complete SAFESPOT Test Specification derived from the specifications written in the different SAFESPOT subproject.

Test specifications are necessary to:

− Ensure that equipment and systems claiming compliance to the standard or a profile have been sufficiently tested to demonstrate that compliance.

− Guarantee that equipment from multiple vendors has been tested the same way, to the same interpretation of the standard, increasing the interoperability of the equipment.

In order to achieve a complete SAFESPOT Test Specification, this phase is divided into five steps based on ISO 9646 and ETSI Recommendations:

− **Step 1 - Production of the Requirement Catalogue:** The requirements catalogue collects all features that are mandatory and optional for implementations conforming to the SAFESPOT Specification.

− **Step 2 - Production of ICS (Implementation Conformance Statement) proforma:** ICS provides an overview of the features, capabilities, functionalities and options that are supported by a SAFESPOT Implementation conforming to the SAFESPOT Specification.
- **Step 3 - Decision on Testing Strategy**: What aspects should be run as conformance testing and which ones as interoperability, if any.

- **Step 4 - Development of the Test Suite Structure and Test Purpose (TSS & TP)**: this document derived from the test strategy and requirement catalogue documents. Test Purposes (TPs) provides a short description of each test objective using words, focusing on the meaning of the test rather than detailing how it may be achieved. TSS provides a logical grouping for the TPs.

- **Step 5 - Production of the ATS (Abstract Test Suite)**: ATS is a collection of Test Cases. Each Test Case specifies the preconditions for setting up the test and the steps that must be taken in order to perform the test.

![Diagram of SAFESPOT Specifications]

**Figure 27. Steps to generate SAFESPOT Test Specifications**

Test Specification is composed of the following documents: Requirement Catalogue Document, ICS Document, TSS & TP Document and ATS Document. Next sections describe each document in more detail and introduce a template to be used in order to specify each document.

As it was described before, beaconing functionality was selected to be tested. The test specifications of this functionality have not been included in this deliverable, but they have been completely specified in external documents which are located on SAFESPOT web site (http://bscw.safespot.eu.org/bscw/bscw.cgi/163215). Next figure illustrates the specifications used to extract all the beaconing test specifications.
Specifications and standards may be understood as a set of requirements related to the technology defined in these documents. For this reason, requirements catalogue collects all features that are mandatory and optional for implementations conforming to the specification or standard to be used. This step is laborious and expensive, but it is an essential step in the whole testing process, because this catalogue is a summary of the specifications and standards and can help to understand them much better in order to achieve a well-defined test specifications.

The template chosen to document requirements catalogue is presented in Annex 1.

### 5.2.2. Implementation Conformance Statement (ICS)

The ICS is a checklist of the capabilities supported by the IUT embedded in the SUT. It provides an overview of the features, capabilities, functionality and options that are implemented by a product under test. The ICS can be used to select and parameterize the test cases as an indicator for basic interoperability between different products. Consequently, the ICS must be filled in by the manufacturer.

The template chosen to document ICS is presented in Annexe 1.
5.2.3. **Test Strategy**

The next step would be to define the most adequate test strategy according on the focus of testing. In one hand, conformance testing is useful when the implementation interfaces are clear and unambiguous conformance requirements are set, in the other interoperability testing is appropriate when the details of the underlying technology are not strictly defined; e.g., when a specific wireless technology is not binding. Also, network integration testing shall be used for those providing the network operation or the service provisioning.

In most cases and due to the complementary of the approaches a combination of two or more types of testing provides the best solution.

As a result of the test strategy and the test tools to use some extra information may be needed in the process. The Implementation eXtra Information for Testing (IXIT) contains additional information necessary for testing.

5.2.4. **Test Purpose & Test Suite Structure (TSS&TP)**

The TSS&TP development is based on SAFESPOT specifications; they provide an informal, natural language description of every test, focus on the meaning of the test rather than detailing how it may be achieved.

Test purposes are grouped into a logical Test Suite Structure according to suitable criteria (e.g., basic interconnection, functionality, tests architecture, etc.).

The template chosen to document TSS & TP is presented in Annex 1. The language used to describe the test purpose itself is TPLan (Test Purpose Language), which is a notation developed by ETSI for expressing Test Purpose in a formal way.

5.2.5. **Abstract Test Suite (ATS)**

The ATS is the total collection of all Abstract Test Cases (ATC) contained in the TSS & TP document. Each Test Case specifies the preconditions for setting up the test and the steps that must be taken by the test operators in order to perform the test.

An ATC is a complete and independent specification of the actions required to achieve a specific test purpose. The main difference with respect to TSS & TP is that ATC defines HOW not WHAT at a certain level of abstraction.

ATS may be seen at the first step in the design of the test system. It is very useful to model the ATS though a language such as UML [9] using a special profile called U2TP (UML 2.0 Testing Profile) [10] (See Annex 2).
The template chosen to document conformance and interoperability ATS is presented in Annex 1.

5.3. **Phase 3: SAFESPOT Test System**

This phase is detailed in chapter 6.
6. SAFESPOT Test System Description (SFTS)

SAFESPOT Test System (SFTS) description is the final phase of the STRF, the phase 3: Test System Implementation and Validation. Based on the conformance point extracted from phase 1 of STRF and SP7 Malaga meeting decision, the first approach of System Under Test, from now on calling SAFESPOT Unit Under Test (SUUT), is the VANET implementation based on D3.3.4 from SINTECH subproject [22].

The general functional architecture of the SFTS is illustrated in Figure 30. This architecture is mainly composed of four components: Test Operator, Test Management, Test System Management and SUUT.

- **Test Operator**: it is usually a human user. Test Operator is in charge of selecting the test cases to be executed and filling in the adequate parameters for each test case. The interaction between the Test Operator and Test Management will be through the TO-TM interface.

- **Test Handler**: it is responsible of:
  - Selecting of the test cases to be executed.
  - Managing the execution of the test system
  - Generating test report of the test cases executed.
  - Managing the several parameters used in a test case.
Figure 30. SAFESPOT Test System General Architecture

- **Test System Management**: it is the core of the whole test system because it includes the test cases developed. Furthermore, this component is composed of other relevant functionalities:
  - Logging manager is in charge of providing the logs generated during the execution of a test case.
  - Automatization manager will be able to automate the whole process of execution of the test case in order to avoid the human interaction.
  - Data manager will gather all the information needed for the correct configuration and behaviour of the test case, for instance PICS and parameters filled in Test Management component. These data will be transferred from the Test Management to the Test System Management through the TM-TS interface.
  - Timer Manager will manage the time source of the test system. The time could be extracted from an operating system or a external hardware.
− External Manager will manage the possible interaction between the test cases with other hardware, such as protocol analyzer, oscilloscopes, etc. in order to measure special information from the SUUT.

− Communication Manager will manage all the communication between the SUUT and the Test System Management through the TS-SUUT interface.

− **SUUT**: it is the SAFESPOT implementation to be tested and will be developed in other SAFESPOT projects. **It is very important to highlight that the SUUT must be provided to the test system developers to validate and improve the whole test system.**

The SAFESPOT test system mock-up described in the following sections (conformance and interoperability test systems) are based on this general architecture. Performance testing is out of the scope of this deliverable, because performance is considered in the SAFESPOT technical and applicative subprojects as internal testing.

### 6.1. SAFESPOT TTCN-3 Conformance Test System

This test system will be in charge of carrying out the conformance testing in SAFESPOT. The test system architecture selected is TTCN-3 Test System Architecture ([11], [12], [13]). This architecture makes testing available and independent of the mean of connection selected in the SUUT.

TTCN-3 is a standardized testing language and test system architecture that is being used and accepted widely in the telecom industry. TTCN-3 is being successfully used in the certification of new challenging technologies such as IPv6 and WiMAX. Basically, TTCN-3 provides a common language used by many different people involved in testing.

#### 6.1.1. TTCN-3: Formal Language for writing test specifications

Once, ATS are modelling through U2TP, the next step is to write the ATC using a formal specification language called TTCN-3 (Testing and Test Control Notation).

TTCN-3 is a formal language applicable to the specification of all types of reactive systems tests over several communication interfaces. This language is addressed for a broad user community and provides a platform independent specification. The syntax is similar to a conventional programming language such as C, C++ and Java and therefore it is easy to understand for programmers. The TTCN-3 Core is described in the standard ETSI ES 201 873-1 [11]: Methods for Testing and Specification and Test Control Notation version 3; Part 1: TTCN-3 Core Language.

TTCN-3 has a well defined syntax, syntactic and operational semantic as well as being constantly maintained and developed by ETSI.
TTCN-3 can be represented to the TTCN-3 developer through different presentation formats. ETSI ES 201 873-2 [14] describes the tabular presentation format and ETSI ES 201-873-3 [15] describes the graphical presentation format. Furthermore, the operational semantic of the TTCN-3 is explained in the standard ETSI ES 201 873-4 [16].

TTCN-3 is based on new concepts being independent of any syntax. The main capabilities of TTCN-3 are summarized as follows:

- Dynamic concurrent testing configurations.
- Various communication mechanisms (synchronous and asynchronous).
- Data and signature templates with powerful matching mechanism.
- Specification of encoding information.
- Display and user-defined attributes.
- Test suite parameterization.
- Control of Test Case execution and selection mechanisms.
- Control of complex test configurations.
- Assignment and handling test verdicts.
- Harmonized with ASN.1 (XML and IDL coming).
- Different presentation formats.
- Well-defined syntax, static-and operational semantics.

The main advantage of using is that the test cases are not dependent on the test platform used. Thus the behaviour part of the test case is the same for all test system implementations.

TTCN-3 reduces deeply the development effort in designing complex abstract test suites and enables the possibility of re-usability, allowing better use of human resources in companies.

### 6.1.2. TTCN-3 Test System

TTCN-3 test cases are executed on a specific test system. SAFESPOT Conformance Test System Architecture will follow the architecture presented in Figure 31 based on TTCN-3 Test System Architecture and SAFESPOT general architecture presented above. Below the main components of a TTCN-3 test system are defined.

- **TTCN-3 Executable (TE):** TE is the entity responsible for the interpretation or execution of the TTCN-3 test cases. Conceptually, the TE can be divided into two entities: Executable Test Suite handles the execution of test cases, the TTCN-3 Runtime System (T3RTS) performs all the actions necessary to execute correctly a test case; this entity interacts with the Test Management, SA and PA via TCI and TRI interfaces and manages Executable Test Suite.
Figure 31. Basic Components of a TTCN-3 Test System

- **CODECS**: the CODECS entity is responsible for encoding and decoding each data type exchanged between TE entity and SA and PA entities via TRI. An encoder function will be called when a value is sent to the SUT. A decoder function will be called whenever received data have to be converted into a TTCN-3 value.

- **System Adaptor (SA)**: the SA adapts message and procedure based communication of the TTCN-3 test system with the SUT to the particular execution platform of the test system. It is responsible to propagate send requests and SUT action operations from the TTCN-3 Executable (TE) to the SUT, and to notify the TE of any received test events by appending them to the port queues of the TE. These capabilities are implemented through the TRI (TTCN-3 RunTime Interface) Interface.

- **Platform adaptor (PA)**: the PA implements TTCN-3 external functions and provides a TTCN-3 test system with a single notion of time. In this entity, external functions are to be implemented as well as all timers. Notice that timer instances are created in the TE. The interface with the TE enables the invocation of external functions and the starting, reading, and stopping of timers as well as the inquiring of the status of timers using their timer ID. The PA notifies the TE of expired timers. Finally, external functions are the main component of Platform Adaptor (PA). External functions are a powerful resources supported by TTCN-3 language. External function is a function declared at TTCN-3 level.
but implemented at native level. These capabilities are implemented through the TRI (TTCN-3 RunTime Interface) Interface.

- **Test Management (TM):** the TM entity is responsible for the overall management of a test system. The aim of this entity is to manage the execution control.

One of the key features of TTCN-3 Test System architecture is that adaptors components abstract the test cases from the technology used for each application. For example, in order to upgrade the VANET service testing from an 802.11a to 802.11p communication technology it is only necessary to update the SA functionality and therefore the TTCN-3 code is not modified.

Next table illustrated the mapping between SAFESPOT general architecture and TTCN-3 Test System architecture.

<table>
<thead>
<tr>
<th>Table 9. SF Architecture and TTCN-3 Test System Matching</th>
</tr>
</thead>
<tbody>
<tr>
<td>TE - Test Cases</td>
</tr>
<tr>
<td>Reports</td>
</tr>
<tr>
<td>Test Cases Handler</td>
</tr>
<tr>
<td>PICS</td>
</tr>
<tr>
<td>Parameters</td>
</tr>
<tr>
<td>Logging Manager</td>
</tr>
<tr>
<td>Automatization Manager</td>
</tr>
<tr>
<td>Data Manager</td>
</tr>
<tr>
<td>External Manager</td>
</tr>
<tr>
<td>Timer Manager</td>
</tr>
<tr>
<td>Test Cases</td>
</tr>
<tr>
<td>Communication Manager</td>
</tr>
<tr>
<td>External HW / Timers</td>
</tr>
</tbody>
</table>
Several functionalities, such as Reports, PICS and Parameters, are not implemented in TTCN-3 Test System Architecture. These functionalities could be implemented through a GUI interface that allows managing them.

### 6.2. SAFESPOT Interoperability Test System

This testing phase is executed only after the implementation under test has passed conformance testing as discussed in the previous sections. SAFESPOT testing architecture to interoperability testing shall follow the architecture presented in Figure 32 which is based on architecture described in interoperability section (section 4.3). This type of interoperability testing is called Advanced Interoperability Testing. This architecture is derived from Figure 15 and it is a combination between interoperability and conformance testing. This combination permits complete interoperability testing to be undertaken while limited protocol conformance monitoring takes place.

![Figure 32. Interoperability Test Bed Approach](image)

For interoperability, the system under test is a complete VANET module composed of all its functionality as next figure shows and it is described in [22].
Figure 34 shows the SAFESPOT test-bed configuration for interoperability testing. SAFESPOT interoperability testing consists of the execution of the interoperability Test Purposes that are applicable to the VANET Under Test (VUT). Unlike conformance testing, for the execution of the interoperability Test Purposes the VUT is tested against a set of pre-selected SAFESPOT VANETs Golden Units (VGUs) from different equipment vendors.

This test bed is mainly composed of several building blocks:

- **VUT**: it is the VANET module under test.
- **VGUs**: a set of products from different vendors of VANETS module that are considered as “golden” reference.
- **Test Controller**: it is in charge of executing interoperability test cases, managing the whole test bed functionality and gathering all the information needed to establish the verdict of the test cases.

- **VANET Sniffer**: this block gets all the VANET messages exchanged between the VUT and the VGUs. These messages are also sent to the Test Controller.

- **Switching Unit**: this block provides the necessary paths to perform testing.

- **SAFESPOT Components**: this block is composed of the LDM, Data Processing, set of SAFESPOT applications and the application coordinator based on all subproject specifications developed during this project. These components will emulate a real environment in order to provide all the information needed to be exchanged during the execution of the tests.

- **Test Control Network**: this network will transport all the information related to control each block in the test bed in order to automate the functionality of the test bed. Highlighting that the information exchanged between SAFESPOT components will be based on format message specified in [User Data & Format – SP7].

- **Test Data Network**: this network will transport all the VANET messages between VUGs and the VUT through the switching unit.

It is assumed that this test bed will be validated and considered as the “Gold IOT SAFESPOT Test Bed”, with all interoperability tests initially run in this testbed during validation.
PART B.3 – Guidance for External Applicants
7. Background

The ultimate goal of the SAFESPOT certification program (SCRF), and for the future ITS certification program, would be to promote and accelerate the introduction of cost-effective conformant wireless vehicles-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications into the marketplace. This will be achieved using standards-based, which are currently being specified by ETSI in strong cooperation with COMeSAFETY, C2C-Consortium and other R&D projects. Therefore, SAFESPOT Certified products attest to their conformance to V2V/V2I standards enabling complete interoperability worldwide.

A clear example of success is Wi-Fi technology thanks to Wi-Fi Alliance certification program. This certification program has been a key driver for worldwide Wi-Fi adoption. Wi-Fi Certified devices have been tested for interoperability to improve the user experience. Consumers show a clear preference for Wi-Fi Certified devices and many enterprise users require Wi-Fi Alliance certification. The Wi-Fi Alliance certification program has been a catalyst for the industry because it has led to a positive user experience. The Wi-Fi Alliance approach to certification is based on principles that help ensure interoperability, backward compatibility and innovation. Wi-Fi Alliance certification is entirely voluntary, but equipment vendors worldwide view it as a prerequisite to market acceptance and success. Wi-Fi networks now support diverse functionality and must be capable of operating in more complex environments, where a higher number and variety of devices have to coexist and interoperate.

Having in mind the tremendous success of this technology, and also of WiMAX, Bluetooth, etc., ITS should consider seriously the fact of creating, firstly, a Forum (called Cooperative System Forum in this deliverable, but C2C-Consortium could be considered the proper one in the future) with the aim of promoting ITS Technologies, and secondly, a certification program, here the SCRF (SAFESPOT Certification Reference Framework), to achieve maximum interoperability between V2V/V2I products from different vendors.
8. Added-Value of SAFEPOT Certified Products

SAFEPPOST Certified products that have been through the certification program will reduce investment uncertainties for all applicants in the ITS network value chain. This is why, the creation of a SCRF will bring substantial benefits to the main SAFEPOT applicants in the ITS value chain: OEMs (Original Equipment Manufacturers), Application developers, Car Manufacturers and Drivers (Consumer Users). The SCRF is important for all these players, and will become increasingly crucial for them to succeed in the future ITS market.

Table 10. Added value for the SAFEPOT applicants

<table>
<thead>
<tr>
<th>SAFEPOT Applicant</th>
<th>Added-Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original Equipment Manufacturers (OEMs)</td>
<td>Certification will help to ensure interoperability with other equipment vendors’ products and allows OEMs a further measure to test quality in their equipment before introducing it into the market, thus reducing overall support costs. In addition, Car manufacturers can easily identify certified products by the SAFEPOT logo, showing a quality and reliable product.</td>
</tr>
<tr>
<td>ITS Application Developers</td>
<td>It is expected that ITS will have a worldwide impact, so ITS applications will be also able to be marketed worldwide, lowering development costs and enlarging the addressable market. The SCRF will provide additional functionality that enables application developers to bring more advanced applications to market.</td>
</tr>
<tr>
<td>Car Manufacturers</td>
<td>The SAFEPOT Certified Products will give to Car Manufacturers, the confidence and flexibility of deploying ITS infrastructures from multiple vendors, thereby reducing the overall investment risk and creating a price-competitive marketplace.</td>
</tr>
<tr>
<td>Drivers (Consumer users)</td>
<td>SAFEPOT Certified Products are a trusted brand that offers interoperability, standards-based security, and reliability. It is an assurance that an independent third party has tested the product in numerous configurations and with a diverse sampling of other devices to ensure compatibility with other SAFEPOT Certified Products.</td>
</tr>
</tbody>
</table>
9. The SAFESPOT Approach to Certification

The SCRF would play an essential role in enabling and accelerating the adoption of ITS products and services, concretely V2V/V2I products, and the design of this certification program satisfies the following goals:

- **Interoperability**: it is the primary target of certification. Rigorous test cases are used to ensure that products from different equipment vendors can interoperate in a wide variety of configurations. Interoperability testing makes certification testing more demanding and requires a higher level of collaboration among vendors, but it also facilitates more comprehensive troubleshooting ahead of commercial availability.

- **Backward compatibility**: One of the major challenges of certification is to balance the need to retain backward compatibility with the drive to upgrade to the latest technology and market developments. To retain continuity in the market and technological relevance, the SCRF carefully manages the evolution path and ensures that the certification testing evolves gradually.

- **Support flexible deployments**: ITS technology, and concretely V2V/V2I communications, is based on open standards and therefore can support deployments in which vehicles and elements from infrastructure developed from multiple vendors seamlessly coexist. As they develop products, vendors may interpret differently some standards specifications, and this may limit interoperability. The SCRF ensures that these differences are resolved by including tests for conformance to the standard, and by operating test with equipment from multiple vendors to assess interoperability explicitly.

- **Establish the SAFESPOT program as the trusted resource for products selection**: In certification, trust is an essential ingredient. Drivers would choose a vehicle where SAFESPOT Products Certified are installed as long as they are confident that the certified product is going to work as expected. Similarly, Car manufacturers will require SAFESPOT Products Certified from their OEMs.

- **Trust Third Party**: In addition, to ensure independence and consistency, a SAFESPOT authorized trusted Test Laboratory (SFTL) will conduct the certification testing under the guidance of the SAFESPOT Forum. The SFTL will be selected under a rigorous selection process.

For true interoperability, SAFESPOT Products will have to perform and support the capabilities they have been certified for. The SCRF defines and conducts conformance and interoperability testing on differing vendor products. Through comprehensive testing, the SCRF would ensure that these products from multiple manufacturers work together properly. Passing the suite of tests achieve the “SAFESPOT Certified” designation. The SCRF includes three types of tests to ensure whole interoperability.
Conformance: the equipment conforms to specific critical elements of the V2V/V2I standards (still under developments). Conformance testing usually involves standalone analysis of individual products and establishes whether the equipment responds to inputs as expected and specified.

Interoperability: this testing has always been, and still is, the predominant component of interoperability testing itself, and it is the testing that most people associate with “interoperability”. It involves tests with multiple devices from different equipment vendors. This testing is the component that helps to ensure devices purchased today will work with SAFESPOT Certified Products in the future.

Performance: the equipment meets the performance levels required to meet end-user expectations in support of key applications. Nevertheless, performance tests are not designed to measure and compare performance among products, but simply to verify that the product meets the minimum performance requirements for a good user experience (Performance testing is out of the scope of this deliverable).

The future success of ITS Certification will be one of the key drivers to ITS adoption worldwide. Users have the confidence that certified products will work seamlessly with other certified products. The SCRF focuses on real-world conditions by testing equipment interoperability with market-ready products, by running common applications, and by keeping certification close to the requirements of the equipment users. Backward compatibility helps to ensure ease of integration of new equipment into future ITS networks and can prolong the longevity of equipment investments. Finally, all main car manufacturers would have enthusiastically to embrace certification as a requirement to enter and succeed in the market, so the Cooperative System label is expected to be one of the main elements that guides purchase decisions among consumer Drivers and Car Manufacturers.
10. Conclusions

This deliverable has identified how important is to identify and define a certification program (Part A) and a testing methodology (Part B) for ITS environment with the main aim of increasing interoperability in future ITS technologies.

The certification program described in this deliverable has established a starting point in ITS certification and has identified a set of procedures which could be useful for a future and formal ITS certification program. The certification program has identified the key actors, the basic process to be performed and the main steps to start the scheme.

One of the key points in the certification program is the testing process; this is why this deliverable has defined a testing methodology as formal as possible following standards and recommendations from relevant organisations such as ETSI and ISO. Part of this methodology, concretely phase 1, has been applied in this deliverable; and has considered that key reference points in ITS are the V2V and V2I. Phase 2 and Phase 3 are mainly part of D7.4.4; although a potential architecture of a conformance test system and an interoperability testbed has been written in this deliverable.

Finally, Part C of this deliverable provides details on the various testing procedures in a way that external applicants will have a better understanding on the contents of the various testing and their added value for the pre-implementation activities.
11. References


[22] D3.3.4 – Vehicular Adhoc Network Specification v1.1, SAFESPOT SP3
ANNEXES
Annex 1: SAFESPOT Test Specification Template

Requirement Catalogue Template

Identifier  Unique identifier for each requirement
Spec Clause  Reference to the source of the requirement
Type  The type of requirement (Mandatory, Optional or Conditional)
Applies to  Type of device to which the requirement applies
Requirement  The requirement in English
Spec Text  The actual text from which the requirement was extracted

ICS Template

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Status</th>
<th>Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[replace with Feature 1]</td>
<td>M</td>
<td>O</td>
</tr>
<tr>
<td>2</td>
<td>[replace with Feature 2]</td>
<td>M</td>
<td>O</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>M</td>
<td>O</td>
</tr>
<tr>
<td>N</td>
<td>[replace with Feature N]</td>
<td>M</td>
<td>O</td>
</tr>
</tbody>
</table>

[Replace, if applicable, with

Comment:
O.X:
CXXX: ]

Item column: The item column contains a number which identifies the item in the table.

Description column: The description column describes in free text each respective item (e.g., parameters, timers, etc.). It implicitly means "is <item description> supported by the implementation?"

Status column: The following notations are used for the status column:

- M (mandatory): the capability is mandatory to be supported.
- O (optional): the capability may be supported or not.
- O.X (qualified optional) – for mutually exclusive or selectable options from a set. "X" 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.

− CXXX (conditional) - the requirement on the capability ("M" or "O") depends on the support of other optional or conditional items. "XXX" is an integer identifying a unique conditional status expression which is defined immediately following the table. A conditional expression notation is used following the following style:

CXXX: IF <Boolean expression> THEN … ELSE …

The Boolean expression is composed of either one item reference or more item references connected via boolean connectors such as AND, OR, etc. An item reference is defined as the table identifier, followed by a character "/", followed by the item number in the table, e.g., A.5/4 is the reference to the answer of item 4 in table A.5. Some examples are shown below:

EXAMPLE 1: IF (A.2/1 AND A.2/2) THEN M ELSE O
EXAMPLE 2: IF A.3/3 THEN O ELSE M

**Support column:** The support column shall be filled in by the supplier of the implementation. The following notation is used for the support column:

− Y - Supported by the implementation.
− N – Not supported by the implementation.

**Comment:** The comment may be filled in to give additional information.

### TP Template

<table>
<thead>
<tr>
<th>TP Id</th>
<th>[An unique Test Purpose Identifier]</th>
</tr>
</thead>
<tbody>
<tr>
<td>RQ Reference</td>
<td>RQ_XXX_YYY_1</td>
</tr>
<tr>
<td></td>
<td>[Description Requirement Catalogue 1 from [XXX RQ]]</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>RQ_XXX_YYY_N</td>
</tr>
<tr>
<td></td>
<td>[Description Requirement Catalogue N from [XXX RQ]]</td>
</tr>
<tr>
<td>Role</td>
<td>[List of one or more identifier indicating the role or roles of the implementation being testing by the TP]</td>
</tr>
<tr>
<td>Test Purpose</td>
<td>ensure that</td>
</tr>
<tr>
<td></td>
<td>{</td>
</tr>
<tr>
<td></td>
<td>when { Stimuli described from viewpoint of the IUT }</td>
</tr>
<tr>
<td></td>
<td>then { IUT generates its responses }</td>
</tr>
<tr>
<td></td>
<td>}</td>
</tr>
<tr>
<td>Comments</td>
<td>[Any comment needed]</td>
</tr>
</tbody>
</table>
Test cases for protocol testing are coded using MSCs (message sequence charts) through UML language.

The message format is as shown: Message-Name (field1=value1, field2=value2, ...., fieldN=valueN);

Three values for the fields are possible:

- Fixed value: this field must be present and only a fixed value is valid.
- ?: this field must be present but any value is valid.
- *: this field may be present and, if present, any value is valid.
## Interoperability ATS Template

<table>
<thead>
<tr>
<th>TP ID</th>
<th>[Unique identifier of the test purpose]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Purpose</td>
<td>[Description of the test purpose]</td>
</tr>
<tr>
<td>Configuration</td>
<td></td>
</tr>
<tr>
<td>Preamble</td>
<td>[Previous steps to be performed before test execution]</td>
</tr>
</tbody>
</table>

### Step

<table>
<thead>
<tr>
<th>Step</th>
<th>Test description</th>
<th>Verdict</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[First step to be performed]</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>n</td>
<td>[Is the appropriated test execution obtained?]</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### Postamble

[Conditions to obtain after test execution]

### Observations

[Additional comments]
Annex 2: UML Testing Profile - Formal modelling for testing

UML [6] is a standardized specification language for object modeling. It is a general-purpose modeling language that includes a graphical notation used to create an abstract model of a system. UML is not restricted to modeling software and in this methodology is used for modeling the ATS through a UML profile called U2TP (UML 2.0 Testing Profile) [7].

U2TP defines a language for design, visualization, specification, analysis, building and documentation of test systems. It is a testing modeling language that can be used in stand-alone mode for holding testing artifacts or can be used integrated with UML for holding system and test artifacts.

The basic characteristics of this profile are:

1. Separation between testing behavior and testing evaluation by introducing a new test component: the arbiter. This one allows easy reusing of the testing behavior for other types of testing.

2. Test control, test group and test case concepts integration as the unique test case concept, which can be split up/decomposed into lower level test cases. This also allows easy reusing of the test case definitions within some hierarchy levels. A test context may be considered as a high level test case.

3. Defaults integration to support unpredictable behavior, verdicts, and wildcards.

4. Support for data partitions both for observations and stimulus. This allows test case description without needing to completely define stimulus data. This may be specified just as a range or group of values.

Some other added features to extend the basic use of UML Testing Profile are:

A test configuration (as an internal test context structure) is used to define the test components that are executing a test case, by describing the initial test component configuration and the connection between SUT and themselves.

In addition, U2TP provides some concepts about:

- Testing behavior addressed with/by observations and activities during a test.
- Testing architecture.
- Testing data.
- Time.
An example of one conformance ATS test configuration using U2TP is illustrated in Figure 35. The ATS shows the SUT and two elements (LT and MTC) that compose the test system. Moreover, the test configuration depicts the different connections between each entity.

![Figure 35. ATS Example using U2TP](image-url)