From test design to validation
(with the example of the IPv6 test bed)

Sebastian Müller
Centre for Testing and Interoperability

4th e-Infrastructure Concertation
Sophia Antipolis, 5/6 Dec 2007
ETSI Standardized Testing

- Two complementary forms of testing
  - Conformance testing
  - Interoperability testing

- ETSI Test Specification Development Process
  - Requirement Catalogue
  - Test Purposes
  - Conformance tests/Interoperability tests
  - Test Validation

- ETSI techniques
  - TPLan
  - TTCN-3
Conformance and Interoperability Testing

Product testing against the standard

A

Test System

Conformance testing
(of a network element)

B

Test System

Conformance testing
(of terminal equipment)

Product against product testing

Interoperability testing
(of terminal equipment)
Characteristics of Conformance Testing

- Tests IUT against requirements specified either in a base specification or profile (standard)
  - Gives a high-level of confidence that key components of a device or system are working as they were specified and designed to do
  - Usually limited to one requirement per test
- High IUT control and observability
  - Can explicitly test error behaviour
  - Can provoke and test non-normal (but legitimate) scenarios
  - Can be extended to include robustness tests
- Test execution can be automated and repeated
  - Requires a test system development (and executable test cases)
  - Can be expensive (e.g., radio-based test system)
  - Tests under ideal conditions
- Tests are thorough and accurate but limited in scope
  - At level of detailed protocol messages, service primitives, or procedure calls
Limitations of Conformance Testing

- Does not prove end-to-end functionality (interoperability) between communicating systems
  - Conformance tested implementations may still not interoperate
    - This is often a specification problem rather than a testing problem!
    - Need for minimum requirements coverage or profiles

- Tests individual system components
  - But a system is often greater than the sum of its parts!
  - Does not test the user’s ‘perception’ of the system

- Standardised conformance tests do not include proprietary ‘aspects’
  - Difficult to test via proprietary interfaces, e.g., user APIs
  - Can however be done by a manufacturer with own conformance tests for proprietary requirements
Characteristics of Interoperability Testing

- It is **system testing**
  - Tests a complete device or a collection of devices
- Shows that (two or more) devices interoperate
  - But within a limited scenario
- Tests at a ‘high’ level (perception of users)
  - Tests the ‘whole’, not the parts or layers
    - E.g., protocol stacks integrated with applications
  - Tests functionality
    - Shows function is accomplished (but not how)
- Does not usually involve test systems
  - Uses existing interfaces (standard/proprietary)
- Interoperability testing looks at end-to-end functionality
  - Less thorough but wide in scope
  - Gives a high-level of confidence that devices (or components in a system) will interoperate with other devices (components)
Limitations of Interoperability Testing

- Does not prove that a device is conformant
  - Interoperable devices may still interoperate even though they are non-conformant to a base specification or standard
- Cannot explicitly test error behaviour or unusual scenarios
  - Invalid conditions may need to be forced (lack of controllability)
  - Has limited coverage (does not fully exercise the device)
- Usually performed manually
  - Difficult to automate, tests use more time to be prepared than executed.
- Does not prove 100% interoperability with other implementations with which no testing has been done
  - ‘A’ may interoperate with ‘B’ and ‘B’ may interoperate with ‘C’. But it doesn’t necessarily follow that ‘A’ will interoperate with ‘C’.
World Class Standards

Conformance and Interoperability Testing are Complementary

- **ETSI experience**
  - As you move up a system stack the emphasis should change from conformance to interoperability testing

- **Lower layer protocols, infrastructure**
  - Emphasis on conformance

- **Middleware, enablers**
  - Combination of conformance and interoperability testing

- **Services, applications, systems**
  - Emphasis on interoperability testing

- **Conformance testing can be viewed as a pre-requisite to interoperability testing**

4th e-Infrastructure Concertation, Sophia Antipolis, 5/6 December 2007
ETSI Test Specification Development Process

Testing provides vital feedback into the base standard

Web interface, browsing by function, user-defined search and filter, traceability, document generation, …!
Requirements Catalogue

- Each requirement is identified, extracted, and catalogued as follows (example from IPv6):
  - Requirement type:
    - Mandatory (MUST, MUST NOT, SHALL, SHALL NOT)
    - Recommended (SHOULD, SHOULD NOT)
    - Optional (MAY, MAY NOT, COULD)
  - Requirement role (tested entity):
    - E.g., Host, Router, Node (Host or Router)
  - Requirement context
  - Requirement text
  - Exact base document citation and reference
  - Functional grouping
    - E.g., Process Fragmented packet, Generate ICMPv6 Error Type etc.
World Class Standards

The Requirements Catalog Online

Numbered Requirements and Catalogue

Context:
The implementation receives an Echo Request.

Requirement:
The implementation sends an Echo Reply in response to the request.

Requirement Type: MUST

RFC Reference: RFC 2463, 54.1.10

Conformance Test Purpose:
Interop Test Purpose:

Requirement: RQ_COR_1482

Applicability: Node

Context:
The implementation receives an ICMPv6 Echo Request Message.

Requirement:
The implementation passes the Echo Request message to [upper-layer] processes receiving ICMP messages.

Requirement Type: MAY

RFC Reference: RFC 2463, 54.1.12

Conformance Test Purpose:
Interop Test Purpose:

Requirement: RQ_COR_1483

Applicability: Node

Context:
The implementation has received an Echo Request message and generates an ICMPv6 Echo Reply Message.

Requirement:
The implementation includes the following information in the Echo Reply IPv6 Fields: (a) Destination Address - Copied from the Source Address field of the invoking Echo Request packet in ICMPv6 Fields. (b) Type Field set to 128. (c) Code Field set to 0. (d) Checksum Field set to the calculated checksum. (e) Identifier Field - The identifier from the invoking Echo Request message. (f) Sequence Number Field - The sequence number from the invoking Echo Request message.

Requirement Type: MUST

RFC Reference: RFC 2463, 54.2.18

Conformance Test Purpose: TP_COR_1483_01
Interop Test Purpose:
Test Purposes

- Precise descriptions of the purpose of a test in relation to a particular (base standard) requirement
  - Meaningful to a larger audience than technical experts
- Define the functionality being tested, i.e., WHAT?
  - Should be formulated using a similar level of information content and wording as the corresponding requirement description
  - Should only mention directly relevant aspects of interactions but not describe them in detail
  - Reference a test configuration (i.e., architecture)
  - Identify one or more test purposes per requirement
- Specified at ETSI in
  - Natural language, or
  - ETSI’s Test Purpose notation (TPLan): www.tplan.info
  - But definitely not coded!
TPLan Example from Digital Public Mobile Radio

**TP id**: TP_PMR_0406_01
**summary**: 'Header frame acknowledges connect request'
**TP type**: conformance
**RQ ref**: RQ_001_0406 -- Catalogue Identifier
**IUT Role**: CSF -- Configured Service Function (CSF)
**config ref**: CF_dPMR_CSF_01 -- CSF Implementation Under Test (IUT) and TESTER

with { IUT in standby }

ensure that {
    when { IUT receives a Connection_Request }
    then { IUT sends an Acknowledgement_Frame }
}
Conformance Test Suite

- A collection of detailed test cases or scripts that implement test purposes
  - Can be compiled and executed, e.g., when using TTCN-3
- Specifies HOW to test
  - Implements also handling of, e.g., unexpected or background behaviour, or returning the SUT to the initial state after a test
- Composition of test cases
  - Each individual test has a preamble, test body (i.e., implementation of the Test Purpose), and postamble
  - Test components may be used clearly separate interactions with different logical IUT interfaces during a test
- Assigns test verdicts
- Should be parameterizable at run-time, e.g., with SUT address
- A test suite is not a test system
  - Test focus on IUT behaviour
  - Test system handles also message encoding and transport
What is TTCN-3?

- Testing and Test Control Notation Version 3
- Internationally standardized language developed specifically for executable test specification
- Look and feel of a regular programming language
- Good tool support (today 6 commercial tools available)

Advantages of Using TTCN-3

- Access to standardized test suites
  - UMTS by 3GPP
  - WiMAX, SIP, IPv6 and more by ETSI
- One language enables reusage of test elements
- Cost reduction in testing process and higher quality of products
- Is independent of a test execution environment
  - Standardized runtime interfaces (TRI/TCI)
- Build-in test automation and system specification features
- Further development and maintenance of TTCN-3
Interoperability Test Description

- Specifies detailed steps to be followed to achieve stated test purpose, i.e., HOW?
- State steps clearly and unambiguously without unreasonable restrictions on actual method:
  - Example:
    - Answer incoming call
    - NOT
    - Pick up telephone handset
- Written in a structured and tabulated natural language so tests can be performed manually
- At least in theory it is possible to increase automation with test scripts, e.g., written in TTCN-3
  - Proprietary nature of user APIs makes this in practice difficult
  - Requires additional adapter development from equipment vendors
Validation of Tests

- Typical test suite validation requires that:
  - Tests compile with several TTCN-3 tools
  - Tests are executed on one or more platforms against various implementations of the standard (usually by ETSI members)
- Requires very close co-operation with test tool suppliers, test platform providers, equipment manufacturers etc.

Test Tool Supplier: Testing Tech
Test Platform Provider: ETSI testbed
Equipment Manufacturers:
- Linux: USAGI, MIPL, OpenIKEv2
- FreeBSD: KAME, SHISA, OpenIKEv2
World Class Standards

TTCN-3
Graphical Logging
World Class Standards

TTCN-3 Result Visualisation

[Image of TTCN-3 Test Result Visualisation]

- TTCN Type: Test Control
- User Type: Prototype
- Property: Value

- ID: TC COR
- Prototype: 00

- Test Result Visualisation:
  - Test Case: TCP Connection
  - Result: Pass

- TTCN-3 Graphical Language:
  - Test Script: TCP Connection
  - Protocol: TCP/IP
  - Message: ECHO Request
Test Report Generation

World Class Standards

Report Number: Test-Campaign
Report Title: NEA-DS-32
Company Name: ETSI
Test Lab: Linux Network
System Under Test (SUT): Linux Kernel at ETSI

Module: Adspsp_Core_TestControl
Module Path: Adspsp_Core_TestControl\..
Test Adapter: com.testadapter.integra6.HiveAdapter
Test Average Path: /Hive6-Adapter

Number of Test Cases: 6
Passed: 3
Failed: 1
Aborted: 1
Duration: 0

<table>
<thead>
<tr>
<th>Test Case Name</th>
<th>Test Purpose</th>
<th>Start Time</th>
<th>End Time</th>
<th>Passed</th>
</tr>
</thead>
<tbody>
<tr>
<td>TG_COR_1601_01</td>
<td>U1 sends 'TCP/IP' error message containing: 'unrecognized Next Header type encountered and containing pointer indicating buffer of unrecognized value and unrecognized header'</td>
<td>2005-09-22 16:51:02.967</td>
<td>2005-09-22 16:51:03.386</td>
<td>PASS</td>
</tr>
<tr>
<td>TG_COR_1692_01</td>
<td>U1 sends 'TCP/IP' error message containing: 'unrecognized Next Header type encountered and containing pointer indicating buffer of unrecognized value and unrecognized header'</td>
<td>2005-09-22 16:51:02.967</td>
<td>2005-09-22 16:51:03.386</td>
<td>PASS</td>
</tr>
</tbody>
</table>
Conclusions

- Standards and Test Specification development should be systematic
- Standards should be designed for interoperability and testability
- Synchronize testing with standards and product development
- Perform Interoperability events!
- Validate Standards and Products through testing
- ETSI can help you achieve the above!
Thank you for your attention

Contact:
Sebastian Müller, Senior Technical Expert
sebastian.mueller@etsi.org