USING TDL IN THE DEVELOPMENT OF STANDARDIZED TEST SPECIFICATIONS FOR IOT AND OTHER TECHNOLOGIES

Michele Carignani, Anthony Wiles
ETSI Centre for Testing and Interoperability
STANDARDIZED TEST SPECIFICATION AT ETSI

Implementations mature from prototypes to INTEROPERABLE commercial products

Market Awareness and Pre-Standardization:
- Early Plugtests
- Proofs of Concepts

Testing and Validation:
- Methodologies
- Frameworks
- documentation, planning, and tools

Product Testing (not ETSI):
- Standardized Test Specifications
  - Conformance, Interoperability etc.

PLUGTESTS Events
- Interoperability, Technology Evaluations, Coexistence etc.

INTEROPERABLE standards development in parallel with product development

User Conference on
Advanced Automated Testing

© All rights reserved
# METHODOLOGY

<table>
<thead>
<tr>
<th>Requirement identification</th>
<th>Test Purpose</th>
<th>Test Case (TC) specification</th>
<th>Test validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Which requirements are to be tested</td>
<td>What is to be tested</td>
<td>How is it to be tested? (Executable)</td>
<td>-</td>
</tr>
<tr>
<td>Prose, Tables</td>
<td>Prose, Tables, TPLan</td>
<td>TTCN-3, Tables, MSCs, etc.</td>
<td>Is test implemented correctly?</td>
</tr>
</tbody>
</table>

- **Test validation**: Is test implemented correctly?
- **Test Case (TC) specification**: How is it to be tested? (Executable)
- **Test Purpose**: What is to be tested
- **Requirement identification**: Which requirements are to be tested
TEST DESCRIPTION LANGUAGE

• An ETSI MTS standard
• Designed to fill the gap between Test Purposes and Test Cases
• Made of 6 parts
  • Part 1: core semantics
  • Other parts: Extensions
• More to come later today..

More at tdl.etsi.org
TDL FOR STANDARDIZED TESTS?

Let’s put TDL under test in

1. OneM2M (IOT & M2M)
2. ETSI ISG NFV (Telecom & Cloud)
3. 3GPP (Mobile networks)
1. OneM2M: Conformance test

Test objective: Check that the stateTag attribute of a container resource is increased when a child resource is deleted.

Reference: TS-0001 10.1.4

Config Id: CF01

PICS Selection: PICS_CSE

Initial conditions:
- the IUT being in the "initial state"
- the IUT having registered the AE
- the IUT having created a container resource containing a child resource
- the AE having privileges to perform DELETE operation on the TARGET_CHILD_RESOURCE_ADDRESS

Expected behaviour:

<table>
<thead>
<tr>
<th>Test events</th>
<th>Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>when {</td>
<td>IUT ← AE</td>
</tr>
<tr>
<td>the IUT receives a valid DELETE Request from AE containing To set to TARGET_CHILD_RESOURCE_ADDRESS and From set to AE_ID }</td>
<td></td>
</tr>
<tr>
<td>then {</td>
<td>IUT → AE</td>
</tr>
<tr>
<td>the IUT increments the stateTag attribute of the container resource and the IUT sends a valid Response containing Response Status Code set to 2002 (DELETED) }</td>
<td></td>
</tr>
</tbody>
</table>

TDL Part 4
TDL-TO
Embed structured test purposes in the test description
2. NFV CASE STUDY: INTEROPERABILITY TEST

Test Description: standalone NS instantiation

<table>
<thead>
<tr>
<th>Step</th>
<th>Type</th>
<th>Description</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Stimulus</td>
<td>Trigger NS instantiation on the NFVO</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>IOP Check</td>
<td>Verify that the software images have been successfully added to the image repository managed by the VIM</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>IOP Check</td>
<td>Verify that the requested resources have been allocated by the VIM according to the descriptors</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>IOP Check</td>
<td>Verify that the VNF instance(s) have been deployed according to the NSD (i.e. query the VIM and VNFM for VMs, VLs and CPs)</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>IOP Check</td>
<td>Verify that the VNF instance(s) are reachable via the management network</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>IOP Check</td>
<td>Verify that the VNF instance(s) have been configured according to the VNFD(s) by querying the VNFM</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>IOP Check</td>
<td>Verify that the VNF instance(s), VL(s) and VNFFG(s) have been connected according to the descriptors</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>IOP Check</td>
<td>Verify that the NFVO indicates NS instantiation operation result as successful</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>IOP Check</td>
<td>Verify that the NS is successfully instantiated by running the end-to-end functional test</td>
<td></td>
</tr>
</tbody>
</table>

Pre-test conditions:
- NSD, its associated descriptors (VLD(s), VNFFGD(s)) and VNF Package(s) have been on-boarded to the NFVO
- The software image repository is reachable by the VIM
- The required resources are available on the NFVI
1. NFV: TD HEADERS

<table>
<thead>
<tr>
<th>Identifier</th>
<th>TD_NFV_NS_LCM_INSTANTIATE_001</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Purpose</td>
<td>To verify that a standalone NS can be successfully instantiated</td>
</tr>
<tr>
<td>Configuration</td>
<td>SUT Configuration 1</td>
</tr>
<tr>
<td>References</td>
<td>IFA005, IFA006[1], IFA007, IFA008, IFA010, IFA013</td>
</tr>
</tbody>
</table>

**Test Objective**

TP_NFV_NS_LCM_INSTANTIATE_001 {
    from: "7.6.2.1";
    description: "To verify that a **standalone NS can be successfully instantiated**";
} with {
    APPLICABILITY :
        "- NFVO/VNFM can request VIM to allocate virtualised resources [...]"
}

**Action preConditions:** 

"- NSD, its associated descriptors (VLD(s), VNFFGD(s)) and VNF Package(s) have [...]"
1. NFV: TEST CONFIGURATIONS

Type \texttt{MSG};
Gate Type \texttt{TestPort} accepts \texttt{MSG};

Component Type \texttt{NFVO} having {
\begin{verbatim}
gate tst of type TestPort;
gate toVnfm of type TestPort;
gate toVim of type TestPort;
\end{verbatim}
}
/* Similar for VNFM, VIM, NFVI, EMVNF ... */

Component Type \texttt{TestSystem} having {
\begin{verbatim}
gate tst of type TestPort;
\end{verbatim}
}

Test Configuration \texttt{NFV\_TST\_007\_conf\_1} {
\begin{verbatim}
create Tester \texttt{tstSym} of type TestSystem;
create SUT \texttt{nfvo} of type \texttt{NFVO};
/* Similar for VNFM, VIM, NFVI, EMVNF ... */
connect \texttt{tstSym.tst} to \texttt{nfvo.tst};
connect \texttt{tstSym.tst} to \texttt{vnfm.tst};
connect \texttt{tstSym.tst} to \texttt{vim.tst};
connect \texttt{tstSym.tst} to \texttt{emvnf.tst};
connect \texttt{tstSym.tst} to \texttt{nfvi.tst};
\end{verbatim}
}
1. NFV: TEST BEHAVIOURS

<table>
<thead>
<tr>
<th>Test Sequence</th>
<th>Step</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>Stimulus</td>
<td>Trigger NS instantiation on the NFVO</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>IOP Check</td>
<td>Verify that the software images have been successfully added to the image repository managed by the VIM</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>IOP Check</td>
<td>Verify that the requested resources have been allocated by the VIM according to the descriptors</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>IOP Check</td>
<td>Verify that the VNF instance(s) have been deployed according to the NSD (i.e. query the VIM and VNFM for VMs, VLs and CPs)</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>IOP Check</td>
<td>Verify that the VNF instance(s) are reachable via the management network</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>IOP Check</td>
<td>Verify that the VNF instance(s) have been configured according</td>
</tr>
</tbody>
</table>

Action triggerNSInst : "Trigger NS instantiation on the NFVO";

Test Description TD_NFV_NS_LCM_INSTANTIATE_001 uses NFV_SUT_Configuration_1 {
    perform action preConditions with { PRECONDITION ; };

    perform action triggerNSInst with { STEP : "1"; TYPE: "stimulus" ; } ;

    perform action "Verify that the software images have been successfully added to the image] repository managed by the VIM" with { STEP : "2"; TYPE: "IOP Check" ; } ;

    /* Similar for steps 3-9 ... */
}
2. 3GPP: PARALLEL TEST BEHAVIOURS

### Table 13.3.2.3.2.5: Parallel behaviour

<table>
<thead>
<tr>
<th>St</th>
<th>Procedure</th>
<th>Message Sequence</th>
<th>TP</th>
<th>Verdict</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Check: Does the UE transmit a ROUTING AREA UPDATE REQUEST message?</td>
<td>--</td>
<td></td>
<td>P</td>
</tr>
<tr>
<td>1A</td>
<td>The SS transmits a SECURITY MODE COMMAND message for the PS domain.</td>
<td>&lt;-- SECURITY MODE COMMAND</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1B</td>
<td>The UE receives a SECURITY MODE COMMAND message for the PS domain.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**EXCEPTION:** In parallel to the events described in Step 1 to 3 the steps specified in Table 13.3.2.3.2.6 take place.

### Table 13.3.2.3.2.6: Parallel behaviour

<table>
<thead>
<tr>
<th>St</th>
<th>Procedure</th>
<th>Message Sequence</th>
<th>TP</th>
<th>Verdict</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>The SS transmits a SECURITY MODE COMMAND message for the CS domain.</td>
<td>&lt;-- SECURITY MODE COMMAND</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4a1</td>
<td>The UE transmits an EPCapability/End message to request EUTRA radio access information for E-UTRA and UTRA.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4a2</td>
<td>The UE transmits an EPCapability/Info message on Cell 5.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**EXCEPTION:** Steps 7a1 – 7b1 describe behaviour that depends on the UE implementation; the "lower case letter" identifies a step sequence that take place if the UE performs a certain action.

**EXCEPTION:** Step 7a1a1 describe behaviour that depends on the UE implementation; the "lower case letter" identifies a step sequence that take place if the UE performs a certain action.

<table>
<thead>
<tr>
<th>St</th>
<th>Procedure</th>
<th>Message Sequence</th>
<th>TP</th>
<th>Verdict</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-4</td>
<td>Void</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1A</td>
<td>Void</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1a-1</td>
<td>Void</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1a-4</td>
<td>Void</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5-5</td>
<td>Void</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7a1</td>
<td>IF the UE wants to remove SRVCC media in the next 10 sec. THEN the generic procedure defined in Annex C.24 of TS 34.229-1 [35] take place.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**EXCEPTION:** In parallel to the events described in step 34 to 39 the steps specified in Table 13.3.2.3.2.5 take place.
2. 3GPP: COMPLEX TEST BEHAVIOURS CONT.

```c
TDLan Specification spec_3gpp_36523 {
    /* Inter-system mobility
        E-UTRA PS voice+PS data to UTRA CS voice+PS data
        SRVCC */
    /* ... */
}
```

**Test Description** TD_13_4_3_2 uses 3gpp_conf_1 {

```c
    // steps 1 - 34
    run {
        // steps 34-39
        } in parallel to {
            // table 13.4.3.2.3.2-5
            run {
                // steps 1-3
            } in parallel to {
                // table 13.4.3.2.3.2-6
                optionally {
                    optionally { // step 7a1a1 }
                    // steps 7a2-3
                } optionally { // step 7b1 }
                } optionally { // step 4a1 }
                } // step 40
}
```
VISION FOR TDL

• Embrace Test Purposes with TDL-TO (TPLan)
  • Treated as first class elements of the language
• Keep focus on the high level of design
  • Simplicity is the goal
    • Test configurations
    • Test behaviours
  • Implementation details out of scope
• TDL is not a replacement for TTCN-3
• Tooling support required to enable adoption of TDL
TDL from STANDARDS to PRACTICE (and back)
TDL OPEN SOURCE PROJECT

to the TOP

✓ Textual and graphical editors
✓ Generation of documentation, TTCN3 code and metadata
✓ Format exchange
✓ And more...

top.etsi.org
TOP ECOSYSTEM

More at tdl.etsi.org/wiki
LOOKING FORWARD

• TDL has its place in standardization

• Next steps
  • Digesting feedback from users
  • Tools (editor!) are needed

• What is ETSI doing?
  • Bringing TDL to standardizers and feedback to MTS
  • Bringing TOP to the testing community

• JOIN THE PROJECT! (top.etsi.org)
QUESTIONS?

Contact Details:
Michele Carignani (michele.carignani@etsi.org)
Anthony Wiles (anthony.wiles@etsi.org)

Thank you!