THE ETSI TEST DESCRIPTION LANGUAGE (TDL)

Philip Makedonski, Andreas Ulrich (on behalf of MTS-TDL Working Group)
Where does TDL come from?

- European Telecommunication Standards Institute (ETSI)
  - develops standards and test specifications for ICT to facilitate interoperability
  - domains: fixed, mobile, radio, aeronautical, broadcast and internet technologies
- Technical Committee on Methods for Testing and Specification (TC MTS)
  - standardising test and specification methods and languages, guidelines, frameworks
  - Testing and Test Control Notation version 3 (TTCN-3)
  - Test Description Language (TDL)
- Centre for Testing and Interoperability (CTI)
  - evaluates test specification technologies
  - provides hands-on support and assistance to TCs and projects
Where does TDL come from?

**Agile**
- support of test-driven / behaviour-driven development
- derive scenario-based tests from user stories
- address different stakeholders through multiple representations

**Models**
- describe test-related interfaces, configurations, behaviour, and data
- generate of abstract tests from test specifications
- integrate into model-driven software development processes

**Automation**
- common and frequently used test patterns
- clearly defined execution semantics
- generation of concrete (executable) tests from test specifications
What is TDL?

- Test Description Language
- Design, documentation, and representation of formalised test descriptions
- Scenario-based approach
- Standardised at ETSI by TC MTS
  - STF 454 (2013)
  - STF 476 (2014)
  - STF 492 (2015-2016)
  - STF 522 (2017)
What is TDL?

Part 1: MM Meta Model and Semantics

Part 2: GR Graphical Syntax

Part 3: XF Exchange Format

Part 4: TO Structured Test Objective Specification

Part 5: UML Profile for TDL

Part 6: Mapping to TTCN-3

Part 7: Extended Test Configurations
What is TDL?

- TDL main ingredients
- Test data
- Test configuration
- Test behaviour
- Test objectives
- Time
Structured Test Objectives with TDL-TO

- Requirements to be tested
- Behaviour-driven approach
- Prose syntax

<table>
<thead>
<tr>
<th>TP Id</th>
<th>TP/GEONW/FDV/BAH/BV/01</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Objective</td>
<td>Check defined values of default \texttt{Gn} parameters in the basic header</td>
</tr>
<tr>
<td>Reference</td>
<td></td>
</tr>
<tr>
<td>PICS Selection</td>
<td>PICS_F1</td>
</tr>
</tbody>
</table>

### Initial Conditions

```plaintext
Given

with {
  the IUT entity being in the initial state
}
```

### Expected Behaviour

```plaintext
When

ensure that{
  when {
    the IUT entity is requested to send a "GUC packet"
  }
  then {
    the IUT entity sends a "GUC packet" containing
    BasicHeader containing
    "version field" indicating value "itsGnProtocolVersion MIB parameter",
    "RNL field" indicating value "itsGnDefaultHopLimit MIB parameter"
  }
}
```

### Final Conditions
TDL Pipelines
TDL Open Source Project (TOP)
TDL Open Source Project (TOP)
TDL Open Source Project (TOP)
Mapping TDL to TTCN-3

• Test Description Language
  • Design, documentation, representation of formalised test descriptions
  • Scenario-based approach

• Testing and Test Control Notation
  • Specification and implementation of all kinds of black-box tests
  • Component-based approach
Mapping TDL to TTCN-3

- Establish a connection between TDL and TTCN-3
- Generation of executable tests from test descriptions
- Standardised, ensuring compatibility and consistency
- Re-use existing tools and frameworks for test execution
- Re-use existing TTCN-3 assets (data, behaviour)
ETSI ES 203 119-6
Methods for Testing and Specification (MTS); The Test Description Language (TDL); Part 6: Mapping to TTCN-3

MBT Workflow
System Requirements Specification
Model
Test Generator
TDL
Test Code Generator
TTCN-3
Adaptation
Executable Tests

Manual Workflow
TPLan
TDL-TO

Requirements Level
Test Design Level
Test Implementation Level
Test Execution Level
Mapping TDL to TTCN-3

Gate Type \( gt \) accepts Login, Response;

Component Type \( ct \) having {
    gate \( g \) of type \( gt \);
}

Test Configuration \( tc \) {
    create Tester tester of type \( ct \);
    create SUT sut of type \( ct \);
    connect tester.\( g \) to sut.\( g \);
}

```plaintext
<packagedElement xsi:type="tdl:ComponentType" xml:id="_gKt23nasEeWrfP0MdfQpng" name="ct">
    <gateInstance xml:id="_gKt24HasEeWrfP6MdQpng" name="g" type="_gKt23nasEeWrfP6MdQpng"/>
</packagedElement>
```

```plaintext
function tc() runs on MTC_CT {
    // Test Configuration \( tc \), mappings, connections
    TESTER_tester := ct.create;
    map (TESTER_tester.g_to_map, system: g_to_map);
}
```

```plaintext
type port gt_to_map message {
    //port type for SUT-Tester connections
    inout Login, Response
}

type port gt_to_connect message {
    //port type for Tester-Tester connections
    inout Login, Response
}

type component MTC_CT {
    //component type for MTC
    //variable for the PTC(s)
    var ct TESTER_tester;
}

type component ct {
    port gt_to_map g_to_map;
    port gt_to_connect g_to_connect;
}
```
Mapping TDL to TTCN-3
Mapping TDL to TTCN-3
Mapping TDL to...
Why not UML / UTP?

- Semantic fuzz of UML
- different notations
- different interpretations
- UML Testing Profile (UTP)
  - extension of UML to support (model-based) testing
  - wide scope of modelling notations inherited from UML
  - may still not capture all needs
  - further profiles needed, e.g. MARTE
TDL so far...

• A standardised approach for the design of test descriptions
  • graphical, textual, and user-defined syntaxes, common exchange format
  • first extensions: test purposes with TDL-TO, extended test configurations

• Design-first approach
  • higher level test design before rushing towards detailed test code
  • facilitate better quality of tests and higher productivity in testing

• Harmonise and ease development of tools for scenario-based testing
  • editors mapped to TDL meta-model, e.g. graphical, textual
  • model-based re-usable back-end tools, e.g. code and documentation generators
  • Eclipse ecosystem enables quick and low-cost tool development
What would you like to see in TDL?

tdl.etsi.org