Comments to ES 203 119 V00.00.01 (2013-05): Methods for Testing and Specification (MTS); The Test Description Language (TDL)

Source: Jens Grabowski, University of Göttingen (UGOE)

Date: 30 May

1 General

1.1 Missing Process Definition, Missing Possibilities for Referencing other Specs

It is still unclear for what and how TDL will be used within ETSI. The statements from Ina Schieferdecker (and the following comment from Shicheng Hu) concerning TDL positioning and necessary references to other documents indicated this problem. Potentially, TDL specs may include references to TTCN-3, TPLan and probably requirements specs. Apart from the possibility of adding comments, the current TDL draft includes no concepts for referencing other specs.

The motivation (Section 4.1) tries to explain such a process, but this description is very vague and does not explain the relations and differences between TPLan, TDL and TTCN-3.

Please note: I do not expect that the STF defines a complete ETSI process, but this comment may also be read by other MTS representatives.

1.2 Missing Examples

It is still impossible for me to judge if the current TDL language features are able to model any of the use cases without examples. The examples may be done on a very high level, i.e., by identifying the TDL language concepts needed to model a use case. Just present a use case and list the TDL concepts needed to model the use case and indicate how this is reflected in the use case.

For example (the following Table has been taken from Proposed_TDL_features.pptx in docbox):

Test Purpose					
Identifier:	**TP_IMST2_GM_INI_06**				
Summary:	When a P-CSCF receives any other response other than a SIP 1xx or a 2xx to an initial request to a UE for a dialogue and if the list of Via headers does not match the saved list of Via headers received in the request corresponding to the same dialog, it either sends no message or forwards it to the originating UE.				
Clause:	5.2.6.4.4 second numbered list				
References:	-		**Config Ref:**	CF_2Gm	
IUT Role:	IMS		**Selection Expression:**	**PICS A.2/1, A.3/25.4.1**	
	Entities			**Condition**	
	UE1	**IUT**	**UE2**		
	✓	✓	✓	UE1 and UE2 registered in IUT	
		✓		IUT configured for establishing digest without TLS security association	
		✓	✓	IUT has sent INVITE to UE2	
	UE1	**IUT**	**UE2**		
Step	**Direction**			**Message**	**IF**
1		⇐	⇐	**4xx response** for UE1 ✓ Via header not matching stored Via header	
2a	⇐‖	‖⇐		**no message**	
2b	⇐	⇐		**4xx response** ✓ Via header ➔ stored Via header	**Gm**

If TDL should be able to model the TP above, then just indicate where (possibly also how) the TDL concepts test configuration (seems to be a reference), behavior (kind of behavior, i.e., looks like a short sequence), ports (??), components (seems to be UE1, IUT, UE2), time concepts (possibly the “no message”), data, etc. are represented or references in the TP.

I still do not know how to express conditions for performing a test (i.e., stable testing states) in TDL.

Without having seen such example evaluations for the targeted use case examples, I cannot judge whether the TDL features are sufficient for modeling the targeted used cases, or whether some of the TDL features are already too fine granular.

1.3 Granularity of Language Concepts

I got the feeling that some language concepts are much more elaborated when others, i.e., some concepts remain abstract. For example, the test architecture related concepts are very detailed whereas all kinds of data types need to be constructed by the user. The verdict values seem to be adopted from TTCN-3, but the arbitration is left open

The STF should evaluate the level of detail needed for the different concepts in TDL. For example, by looking to the TP example shown above it is unclear if e.g., gates are needed, if all verdict values or verdict arbitration are needed).

2 Comments on Document Sections

2.1 Section 3.1 "Definitions"

- The terms attribute, key attribute and value attribute are mentioned but not defined. I also didn't find a section on attribute values in the document.

- Definitions for atomic, atomic behavior, atomic behavior elements, and combined behavior elements are missing. Maybe the term element should be deleted in this context and only atomic behavior needs to be defined.

- Please check usage of space and highlighting. I believe that in the definition of **Branch**, the term GateInstance may either be replaced by "Gate Instance" or "*GateInstance"*.

- Definitons of "default", "interrupt" and "viewpoint" are missing or you need to refer to some other document in which the missing definitions can be found.

- Please check whether you by accident redefine well-established definitions commonly used at ETSI. Such a redefinition may cause problems. I am not sure if e.g., the terms test configuration, test description, test objective and timeout event should be redefined here.

2.2 Section 3.3 "Abbreviations"

You may have to add:

- ICS/IFS
- TSS
- TP
- TD
- TC
- EMOF
- TPLan
- TTCN-3
- MSC

2.3 Section 5.2 "Packaging"

The terms "grouping" and "scoping" are mentioned.

Are packages the mechanisms for grouping and scoping?

2.4 Section 7.3.3 "Connections"

Please reconsider Constraint 12: It may be necessary to express broadcast communication. This may be clumsy if you are restricted to 1:1 communication. Example test situation: A broadcast message is sent out via an air interface to detect all (mobile) stations reachable by the sender.

2.5 Section 8.1 "Test Description"

Please reconsider Constraint 15: The reusability of test descriptions on similar test configuration may be decreased by this constraint.

2.6 Section 8.3 "Atomic Events"

Having a defined (and closed?) set of test verdicts but not arbitration concepts looks unbalanced. A simple solution would be to allow global verdicts only and possibly to remove the verdict values error and none.

2.7 Section 8.4 "Combined Behavior"

I recommend thinking about the semantics of communication alternatives. Should TDL follow the TTCN-3 snapshot semantics or take the SDL approach of one (theoretic) input queue where all messages are queued and processed according the order of their arrival. Currently nothing is said, but communication via several channels and queuing is assumed. In combination with exceptional behavior this can become even more complicated.

Semantics "Parallel": Please change the description to "All specified blocks may be executed in parallel"

2.8 Section 8.5 "Exceptional and Periodic Behavior"

From looking into the use cases, I have some doubts that "default exceptional behavior" and "interrupt exceptional behavior" are needed in the first version of TDL. There have also been lots of theoretical discussions in MSC related to the semantics of alternative behavior.