Comments to the TDL early draft, from e-mails and discussion at MTS#59 and the STF’s response

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Issue** | **STF Response** | **Proposer** |
| Major | Data Types in TDL  Proposal: At least the value sets and the expected syntaxes of the simple data types Integer, Float and Boolean shall be specified. | To be investigated.  In the early draft the possibility of referencing external data types is not elaborated. TDL will provide support for a set of basic data types.  The combined use of TDL and external data type language is for further investigation. | E/// |
| Major | Timers, times  Proposal: MTS should discuss handling of timing and timers in TDL should make the needed decisions | To be discussed | E///,  MTS |
| Major | When referencing a Test Description (TD), the referenced TD must have the same configuration as the referencing TD. Can it be a “sub”configuration instead? | To be investigated.  At the moment, interdependent TDs share the same test configuration (TC). TCs are not compatible among each other. | Discussed in MTS though from minutes left out |
| Major | Proposal to remove the Exit Action completely. Unclear what the added value of having it is. MTS believe the removal would enable referencing of subset of “lifelines” in TDL specifications (or remove the current limitation that you can NOT specify that just because of the Exit event – which MTS believes to be a major limitation) | The purpose of ‘ExitEvent’ is to allow the explicit termination of the entire flow of a TD at any time and not just at the end of the TD (implicit exit). A similar feature occurs in other specification languages as well and is considered useful.  To avoid confusion the element should be renamed to ‘StopEvent’. | MTS, Confor-miq |
| Major | Evaluation of Behaviour branches  The semantics of evaluating combined behaviour branches needs to be elaborated in more detail. E.g. what should happen if more than one conditional branch evaluates to true? Also the relation of combined behaviour and exceptional behaviour needs to be defined (e.g. exceptional behaviour branches are evaluated after the combined behaviour branches) | To be investigated.  It is agreed that the semantics on executing branching behavior must be clearly defined.  The intention is to resolve exceptional behavior to a form of branching behavior to simplify the discussion on its semantics. | E/// |
| Major | The current specification of test configuration should provide more detail to reflect properties for links such as observability (PO, PCO, CP etc.). Preferred solution is to simplify test configuration specification to a purely informal specification. | To be investigated.  PO, PCO, CP are no concepts in the TDL MM; they occurred only as illustrative elements in the slides.  The role of a connection (PO, PCO, CP) can be derived from the events of the gates associated with a component and the role (SUT, Tester) of this component.  Test configuration cannot be informal in the meta-model as it would turn them into a comment.  In concrete syntax it may be created implicitly (“optional” element in concrete syntax) | MTS |
| Minor | Request: to align type definitions for data (no type) with type definitions for gates and components (with types) | To be investigated | MTS,  Confor-miq |
| Minor | Proposal: exceptional behaviour should be possible to be added at different levels: at least to test description (e.g. referencing its name), interaction flow, combined behaviour | To be clarified  ‘ExceptionalBehaviour’ is an optional element of a ‘CombinedBehaviour’ which in turn can occur at all levels where ‘Behaviour’ can be specified, i.e. at the top level of a TD or in any nested blocks. | E/// |
| Minor | Timeout event  Clause 8.3 defines a timeout event as “denotes the end point of a time interval, in which no gate interaction events at the associated gate instance occur that could be received by a tester component”. It is unclear, why a relation of a timeout event and other, previous events is needed and why this limitation?  Proposal: delete the above limitation | The ‘TimeOutEvent’ denotes a quiescence observation by the tester since the last SUT output was observed (or since start of TD if nothing at all was observed). Therefore the reference to a previous output is needed to calculate the time for a tester to wait until it can state quiescence and continue execution.  The text in the draft will be updated. | E/// |
| Minor | Verdicts  Proposal: Delete None and Error | Accepted | E/// |
| Minor | Where is TDL supposed to be in between TPLan and TTCN-3 (right?). How can one interrelate TPLan, TDL, and TTCN-3 specification for a given technology/system? How can one trace between the three? For example, how can one refer to a test suite structure in TDL? | TDL is intended to support the Test Description Specification phase at ETSI. Tracing to a TPLan spec is supported by the ‘TestObjective’ element in the MM. Tracing to TTCN-3 code is out of scope for TDL.  Introduction chapter will be extended. | Ina |
| Editorial | Test Case Flag: At least rename or even better drop the Test \_Case\_ Flag, create a consistent solution (e.g., by using instead annotation for this purpose) | To be investigated  Possible solution is to rename it ‘isEntry’ flag. | MTS, Ina |
| Editorial | May I ask to position TDL in the figure (would be good to have a number for it) on page 10. Does it target Test Descriptions (TD) or also other tasks of the test specification development process? | TDL is intended to support the Test Description Specification phase at ETSI.  Introduction chapter will be extended | Ina |
| Editorial | May I also ask to have UTP added as a reference? | Accepted | Ina |
| Editorial | Role: Tester/Sut: Proposal to replace Tester by: Test System – term “Tester” is overloaded with meaning | To be discussed further.  ‘Tester’ is a standard term. TDL makes no assumptions whether it refers to a person or a machine or anything else. | MTS, Confor-miq |
| Editorial | The term “Test Objective” is confusing although the intent was to generalize on test purpose – MTS would prefer to use test purpose in alignment with ETSI’s current test specification processes. TPLan must be referenced here. | The ‘TestObjective’ element can be renamed to ‘TestPurpose’ to better reflect the ETSI process. | MTS, Ina |
| Editorial | Proposal : Use “Termination” instead of “Exit” (note also item 8 in this list) | To be investigated  The element could be renamed to ‘StopEvent’. The need of an additional ‘ExitEvent’ that only terminates the flow of the currently executed TD (in case of nested TD calls) needs to be investigate. | MTS |
| Editorial | Proposal : Use more generic “Output/input event” in text instead of ‘send/receive message’ | Accepted | MTS |
| Editorial | TDL model vs. TDL specification  These two terms are used as equivalent terms (see abstract syntax definition in clause 4.2). In the document mostly the term TDL model is used, except for clause 10.2 and Annex A  Proposal: Use the term TDL model (or similar) for an instance of a TDL meta-model and TDL specification for a concrete specification using a concrete syntax | Accepted | E/// |
| Editorial | Reflexive import  It is not a widely used term, and is not defined in the draft. Circular import (with definition) is proposed instead | Accepted | E/// |
| Editorial | Chapter 3.1 to be moved to an appendix | Accepted  It will be moved to the appendix discussing the MM and replaced by a more suitable definition list. | Confor-miq |
| N.A. | In order to meet the most TDL user requirements, my preferred approach is to see 2 - 3 TDL sub-deliveries to be oriented to the different levels of the usages. | This is a concrete syntax related question, the meta-model must support all use cases | Shi Cheng |

# Shi Cheng, e-mail 30.4.

In order to meet the most TDL user requirements, my preferred approach is to see 2 - 3 TDL sub-deliveries to be oriented to the different levels of the usages.

1. **TDL minimum**, which is a TPLan extension, more prose-oriented. To be completed by STF454. It can be easily and quickly introduced to 3GPP, Plugtest, M2M and the other ETSI TCs for testing. This level is important to initially make TDL a name in the testing world.
2. **TDL extended**, including a set of well-selected features to be completed in the frame of STF454. Here the MTS guidelines are required (STF454 should NOT be overloaded).
3. **TDL enhanced** (an optional technical level, you could also merge into the level above, it's better to split here from the view of STF management)

More features to be added-on by a follow-up STF in 2014. There will be another Stephan's task for the funding request.

Another important point is that STF454 should find a generic way for the evolved TDL transitions between the three levels.