STF 522: TDL Phase 4

Status Report

Document History

- 2017-05-11: Document submitted for SG#2
- 2017-02-27: Document submitted for SG#1
- 2017-01-25: Document submitted for MTS#70

From the Terms of Reference…

TDL Phase 4: Objectives

- New Part 6 for mapping TDL to TTCN-3
- Adaptation and extension of MM addressing CRs
  - new test configuration features (clarifications needed)
  - related to the mapping to TTCN-3
- Adaptation and extension of GR, XF, TO
- New Part 5 to relocate the UML profile (currently annex to MM)
- Requirements for security and performance testing with TDL (TR)

TDL Phase 4: Deliverables

Deliv.	Work Item code Standard number	Working title Scope
D1	RES/ES 203 119-1 V1.4.1	Test Description Language; Meta-Model and Semantics Scope: common concepts, meta-model, semantics
D2	RES/ES 203 119-2 V1.3.1	Test Description Language; Graphical Syntax Scope: TDL graphical concrete syntax for end users
D3	RES/ES 203 119-3 V1.3.1	Test Description Language; Exchange Format Scope: TDL exchange format for tool interoperability
D4	RES/ES 203 119-4 V1.3.1	Test Description Language; Structured Test Objective Specification Scope: TDL extension for structured test objectives
D5	DES/ES 203 119-5 V1.1.1	Test Description Language; UML profile for TDL Scope: TDL to UML meta-model mapping
D6	DES/ES 203 119-6 V1.1.1	Test Description Language; Mapping of TDL to TTCN-3 Scope: Mapping rules to automatically generate TTCN-3 test case skeletons from TDL test descriptions
D7	DTR/ MTS-1029504TDLSecPerfReq	TDL and its usage for security and performance testing; consolidated requirements (technical report)

Status Update

Task 0: Work Plan

- Timescale: Jan, 2017 (+4 months) - Jan, 2018 (+1 month)
  - delays due to administrative overhead at respective organisations
  - final confirmation received on Dec 15, 2016
- Kick-off conference call on Jan 16, 2017
- 1 working session planned so far, further sessions as needed
  - WK7: Feb 14-17 Session 1 @ UG (4 days)
  - Work remotely, meet in person only if necessary
  - Coordinated remote sessions scheduled as needed, based on availability

Task 0: Milestones (Current Planning)

N	Task / Milestone / Deliverable	ToR Targets	Current Targets
M0	Start of work	Sep-2016	Jan-2017
T0	Project management	Sep-2016–Sep-2017	Jan-2017–Jan-2018
T1	TDL-to-TTCN-3 mapping	Sep-2016–Sep-2017	Jan-2017–Jan-2018
T2	Advanced test configuration features	Sep-2016–Apr-2017	Jan-2017–Aug-2017
T3	Language maintenance	Sep-2016–Apr-2017	Jan-2017–Aug-2017
T4	Requirements for security and performance testing	Mar-2017–Sep-2017	Jun-2017–Dec-2017
M1	Informal report on planning	16-Dec-2016	Jan-2017
M2	1st drafts	28-Apr-2017	May-2017
M3	2nd drafts	28-Jul-2017	Sep-2017
M4	Final drafts for MTS review	01-Sep-2017	Dec-2017
M5	Final report, end of work, TB approval	27-Sep-2017	Jan-2018
M6	Membership vote	27-Sep-2017	Feb-2018
M7	Publication	01-Dec-2017	Apr-2018

Task 0: Overall Timeline (Current)

Task Milest.	Description	J	F	M	A	M	J	J	A	S	O	N	D	J	F	M	A
T0	Project management																
T1	TDL-to-TTCN-3 mapping																
T2	Advanced test configuration																
T3	Language maintenance																
T4	Requirements solicitation																
M0	Start of work																
M1	Informal report on planning																
M2	1st drafts																
M3	2nd drafts																
M4	Final drafts for MTS review																
M5	Final report, TB approval																
M6	Membership vote																
M7	Publication																

Notes from Last Meeting

- Inter-SUT communication remains as future work
  - no code-generation for Inter-SUT communications
- Introduce local ordering
  - explicit property of test description
- Focus on message-based communication at first
  - details of procedure-based specification in TDL need to be discussed
- Data mapping required at first
- Test configuration to be discussed

Deliverable Planning

- Focus on Part 1, 5 and 6 for Milestone 1
  - Parts 2, 3, 4 need to be updated according to changes in Part 1
  - changes in Part 1 need to be approved and validated first
- Examples in all parts need to be updated and aligned

Status: Part 1

- Extracted Annex C into Part 5
- Added support for local ordering (constraints pending)
  - added scope to combined behaviour and assertion
  - no mixing of local and global ordering (TestDescriptionReference)
  - no mixing of scope and global ordering (CombinedBehaviour, Assertion)
- Added support for collections
  - variable use rather than variable reference in assignment

Status: Part 1

- Added support for procedure-based communication
  - interaction refined to message or procedure call
  - further refinements can be added if necessary
- Added proposal for extended test configurations
  - currently as annex, see subsequent discussion slides

Status: Part 6

- Partial mapping specifications
  - corresponding updates for changes in MM pending
- Mappings only for local ordering
  - test descriptions that have the corresponding property set to true
- Contents from Part 1 copied for reference
  - will be removed once mappings for an element are done
- Examples

Status: Part 6, 3GPP Example

```
Type PDU;
Type ACK ;
Type C_RNTI;
Type PDCCH (optional c_rnti of type C_RNTI);
Type CONFIGURATION;

PDU mac_pdu ;
ACK harq_ack ;
C_RNTI ue;
C_RNTI unknown;
PDCCH pdcch;
CONFIGURATION RRCConnectionReconfiguration ;
```

```
type charstring SimpleType;
type float Second;
type SimpleType PDU;
type SimpleType ACK;
type SimpleType C_RNTI;

type record PDCCH {
  C_RNTI c_rnti optional
}

type SimpleType CONFIGURATION;

template PDU mac_pdu := "mac_pdu";  // some value is needed
template ACK harq_ack := "harq_ack";
template C_RNTI ue := "ue";
template C_RNTI unknown := "unknown";
template PDCCH pdcch := {};
template CONFIGURATION RRCConnectionReconfiguration :=
    "RRCConnectionReconfiguration";
```

Status: Part 6, 3GPP Example

```
Gate Type defaultGT accepts
    ACK, PDU, PDCCH, C_RNTI, CONFIGURATION ;

Component Type defaultCT having {
    gate g of type defaultGT;
}

Test Configuration defaultTC {
    create Tester SS of type defaultCT;
    create SUT UE of type defaultCT ;
    connect UE.g to SS.g ;
}
```

```
type port defaultGT_to_map message {
  //this is a port type for SUT-Tester connections
  inout charstring, PDCCH /* ACK, PDU, C_RNTI, CONFIGURATION ; */
}

type port defaultGT_to_connect message {
  //this is a port type for Tester-Tester connections
  inout charstring, PDCCH /* ACK, PDU, C_RNTI, CONFIGURATION ; */
}

type component MTC_CT {
  //component type for MTC
  //variable for the PTC(s) --TESTER component(s) in TDL
  var defaultCT TESTER_SS;
}

type component defaultCT {
  port defaultGT_to_map g_to_map;
  port defaultGT_to_connect g_to_connect;
}

function defaultTC() runs on MTC_CT {
  // Test Configuration defaultTC, behaviour to be extracted
  TESTER_SS := defaultCT.create;
  map (TESTER_SS:g_to_map,system:g_to_map);
  TESTER_SS.start(behaviourOfTESTER_SS());
}
```

Status: Part 6, 3GPP Example

```
altstep to_handle_deviations_from_TDL_description_AS () {
  [] any port.receive {
    setverdict(fail);
    //QUESTION TO SG: in case of deviation shall we stop the test case?
    mtc.stop;
  }
  //here can also be handled if nothing arrives, but in this case a timer shall be started
  //before every receive instruction and the timer must be handled somehow
  //or we can leave the timeout for the execute instruction called with the optional
  //timer parameter - but in this case the final verdict will be 'error'
}

altstep quiescence_handler_AS (timer quiescence) {
  //for all quiescence that is not connected to a gate
  [] any port.receive{
     setverdict(fail);
     //QUESTION TO SG: in case of deviation shall we stop the test case?
     mtc.stop;
   }
   [] quiescence.timeout {
     setverdict(pass);
   }
}
```

Status: Part 6, 3GPP Example

```
Test Description TD_7_1_3_1
  uses configuration defaultTC {
    perform action preCondition;
    perform action preamble;

    SS.g sends pdcch (c_rnti=ue) to UE.g;
    SS.g sends mac_pdu to UE.g;
    UE.g sends harq_ack to SS.g with {
        test objectives : TP1 ;
    };
    set verdict to PASS ;
    SS.g sends pdcch (c_rnti=unknown) to UE.g;
    SS.g sends mac_pdu to UE.g;

    //Interpolated original step 6 into
    //an alternative behaviour, covering both
    //the incorrect and the correct behaviours
    alternatively {
        UE.g sends harq_ack to SS.g ;
        set verdict to FAIL ;
    } or {
        gate SS.g is quiet for five ;
        set verdict to PASS ;
    } with {
       test objectives : TP2 ;
    }
}
```

```
function behaviourOfTESTER_SS() runs on defaultCT {
  timer quiescence;

  activate(to_handle_deviations_from_TDL_description_AS());

  g_to_map.send(modifies pdcch := {c_rnti := ue})
  // multiple level modifications cannot be handled in this way
  g_to_map.send(mac_pdu);
  g_to_map.receive(harq_ack);
  /*Test Objective Statisfied: TP1 */
  setverdict(pass);

  g_to_map.send(modifies pdcch := {c_rnti := unknown});
  // multiple level modifications cannot be handled in this way
  g_to_map.send(mac_pdu);
  quiescence.start(five);
  alt{
    [] g_to_map.receive(harq_ack){
       setverdict(fail);
    }
    [] quiescence_handler_AS(quiescence);
    /*Test Objective Statisfied:  TP2 */
  }
}
```

Status: Part 6, 3GPP Example

```
Test Description TD_7_1_3_1
  uses configuration defaultTC LOCALLY ORDERED {
    perform action preCondition;
    perform action preamble;

    SS.g sends pdcch (c_rnti=ue) to UE.g;
    SS.g sends mac_pdu to UE.g;
    UE.g sends harq_ack to SS.g with {
        test objectives : TP1 ;
    };
    set verdict to PASS ;
    SS.g sends pdcch (c_rnti=unknown) to UE.g;
    SS.g sends mac_pdu to UE.g;

    //Interpolated original step 6 into
    //an alternative behaviour, covering both
    //the incorrect and the correct behaviours
    alternatively {
        UE.g sends harq_ack to SS.g ;
        set verdict to FAIL ;
    } or {
        gate SS.g is quiet for five ;
        set verdict to PASS ;
    } with {
       test objectives : TP2 ;
    }
}
```

```
testcase TD_7_1_3_1() runs on MTC_CT
    system defaultCT // It works if there is only one SUT
{
  preCondition();
  preamble()

  activate(to_handle_deviations_from_TDL_description_AS());

  //Test Configuration defaultTC(), behaviour to be extracted
  defaultTC();

  all component.done;
}
```

Questions: Extended Test Configurations

- Integrated -> **Separate extension** / **Annex**?
  - non-essential, burden for tool providers and users
  - additional discussion on generic notion of inheritance throughout TDL
- Self-contained
  - support can be provided if required by supporting extension
  - extension can be reduced to base TDL (“flattening”)
  - otherwise more substantial changes to all of TDL

Questions: Unexpected Behaviour

- Minimal base solution + recommendations
  - focus on what is specified in TDL and not on what is not
  - handle basic assumptions and set verdict to fail
- Assumptions: receive all, timeout, stop?
- Set verdict, generate at most skeleton for further actions
  - can be customised by users or tool vendors

Questions: Reference Implementation

- Necessary for validation
- Initial proof of concept
  - UG to share initial prototype
  - Ericsson / Budapest University to clarify legal aspects
- Under TOP?
- Use STF resources?

Communications

- Proposals for UCAAT (presentation and/or tutorial)
- Interest in UP4TDL from S. Maag / TSP
- Discussions via official channels
  - official policy to use only official channels
  - private mails disregarded in the future
- Philosophical discussion regarding different feature sets

Communications: TDL Philosophy

- Distinction
  - no strict distinction
  - added local ordering feature upon request
    - mapping only for test descriptions that are locally ordered
- Semantics
  - no semantical difference beyond interpretation of ordering
  - no new concepts, minor extension to semantics for new properties
  - only affects test description, combined behaviour, assertion

Communications: TDL Philosophy

- Mapping (global to local)
  - not standardised, up to users
  - proposed sync construct left out of scope unless required
    - synchronisation can be achieved by other means if needed, within or outside of TDL
- impact on TestDescriptionReference
  - effectively identical to copying the content where it is referenced
- assumption that all components start at the same time shall be clarified
  - can also be achieved manually if required

Communications: TDL Philosophy

- Executable subset
  - question not clear
  - effectively everything but ParallelBehaviour will be mapped
    - some restrictions still apply (only locally ordered TestDescriptions)
    - no technical solutions within mapping document, up to users

Communications: TDL Philosophy

- Transformation rules
  - handling of unexpected behaviour (see also previous slides)
    - base solution + recommendation
      - set verdict, at most skeleton for further actions
      - up to users, beyond scope of STF
  - based on what is specified in TDL and not on what is not
- Test objectives
  - specification of test objectives, no impact on mapping

Any other business?