STF 454: Sample TDL Instantiations

Internal Technical Report Based on Examples from 3GPP and IMS

Author: Philip Makedonski

The examples below illustrate different views on model instances derived from 3GPP and IMS examples. These views include:

- MSC-like interaction visualization (e.g. Figure 2),
- MSC-like interaction visualization with notes notes describing individual procedures and related objectives (e.g. Figure 3, mostly for 3GPP examples),
- Data concepts including data types and data instances related to the example (e.g. Figure 1), as well as mappings to concrete data instances in corresponding data resources (e.g. Figure 21),
- Configuration visualization of component instances and connections among them (e.g. Figure 5), also including gate instances (e.g. Figure 4),
- A textual serialization of the interaction behavior based on the PlantUML syntax (e.g. Listing 1),
- A tabular presentation of the interaction sequence, mostly for 3GPP examples, partially empty for all others (e.g. Table 1),
- A full xText-based textual representatoin of the model instance derived from the IMS example.

The purpose of this document is mostly illustrative and should serve as supporting material for discussions and explanations. The rest of this document is mostly automatically generated and does not include any further clarifications at this point. These will be added as necessary, and the examples will be updated and extended as necessaru as well. Any changes to the examples, the meta-model, the concrete syntax presentations, and the generation processes, can be easily incorporated due to the generative nature of this document. This will help to keep this document up to date with little to no effort.

The original materials on which the examples are based are supplied as a separate document.

TODO: References to further materials.

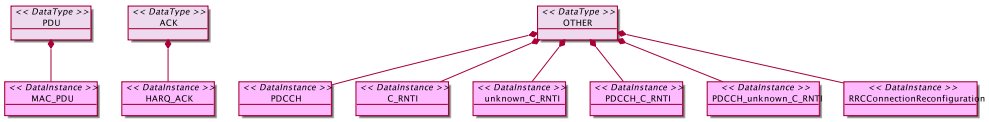

Figure 1: Data concepts visualization for sample-3GPP-7.1.3 project.

1 3GPP 7.1.3

```
@startuml
box "Tester SS <defaultCT>" #white
  participant SSgate
end box
box "SUT UE <defaultCT>" #white
  participant UEgate #CCCCFF
end box
=="Pre-test Conditions:\nRRC Connection Reconfiguration
   "==
=="Preamble:\nThe generic procedure to get UE in test
   state Loopback\nActivated (State 4) according to TS
   36.508 clause 4.5\nis executed, with all the
   parameters as specified in the\nprocedure except that
   the RLC SDU size is set to return no\ndata in uplink.\
   n(reference corresponding behavior once implemented"==
SSgate --> UEgate: ["1"] PDCCH_C_RNTI (C_RNTI)
SSgate --> UEgate: ["2"] MAC_PDU
UEgate --> SSgate: ["3"] HARQ_ACK
note left: PASS
SSgate --> UEgate: ["4"] PDCCH_unknown_C_RNTI (
   unknown_C_RNTI)
SSgate --> UEgate: ["5"] MAC_PDU
UEgate --> SSgate: ["6"] HARQ_ACK
note left: FAIL
@enduml
```

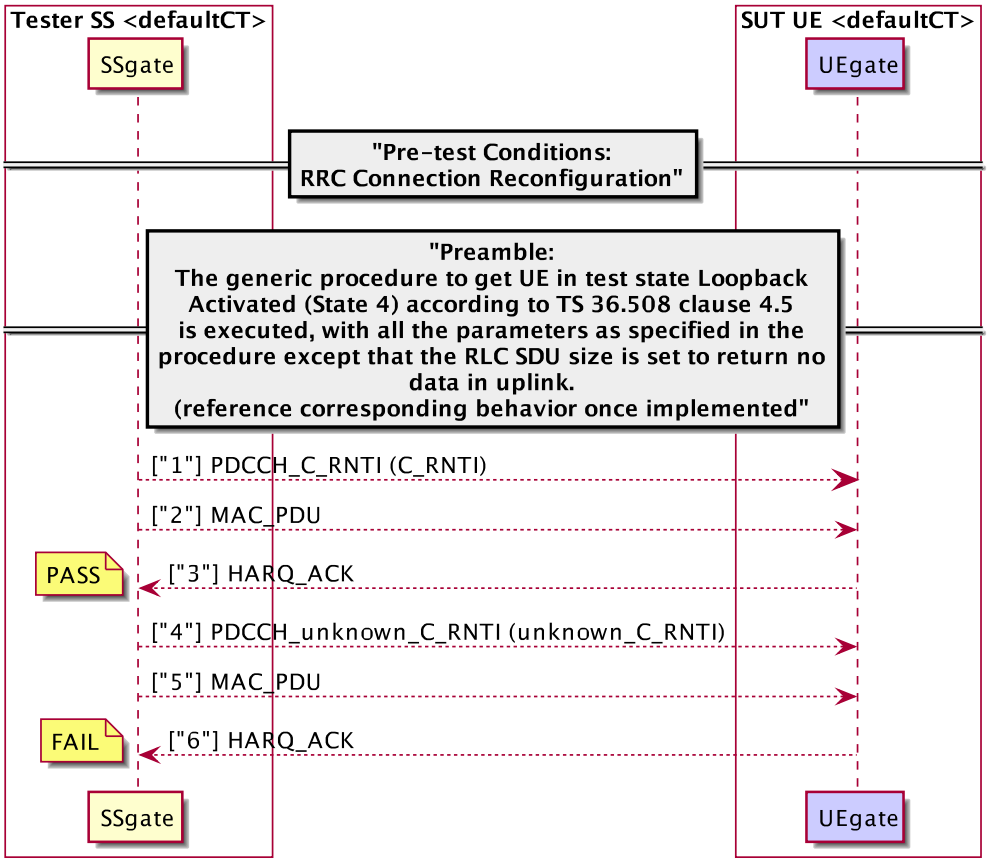

Figure 2: Interactions visualization for sample-3GPP-7.1.3 project.

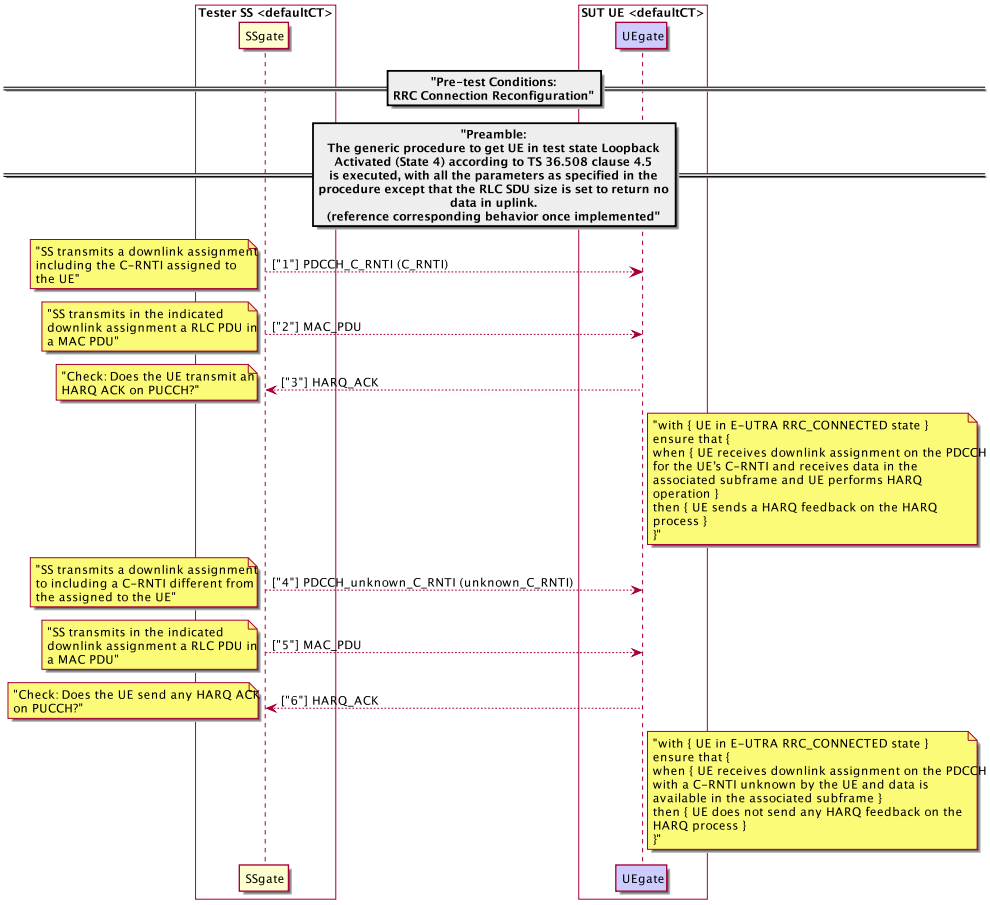

Figure 3: Interactions visualization with notes for sample-3GPP-7.1.3 project.

Step	Procedure	Direction	Message
1	SS transmits a downlink assignment including the C-RNTI assigned to the UE	SS –> UE	PDCCH_C_RNTI (C_RNTI)
2	SS transmits in the indicated downlink assignment a RLC PDU in a MAC PDU	SS –> UE	MAC_PDU
3	Check: Does the UE transmit an HARQ ACK on PUCCH?	UE –> SS	HARQ_ACK
4	SS transmits a downlink assignment to including a C-RNTI different from the assigned to the UE	SS –> UE	PDCCH_unknown_C_RNTI (unknown_C_RNTI)
5	SS transmits in the indicated downlink assignment a RLC PDU in a MAC PDU	SS –> UE	MAC_PDU
6	Check: Does the UE send any HARQ ACK on PUCCH?	UE –> SS	HARQ_ACK

Table 1: Tabular syntax.

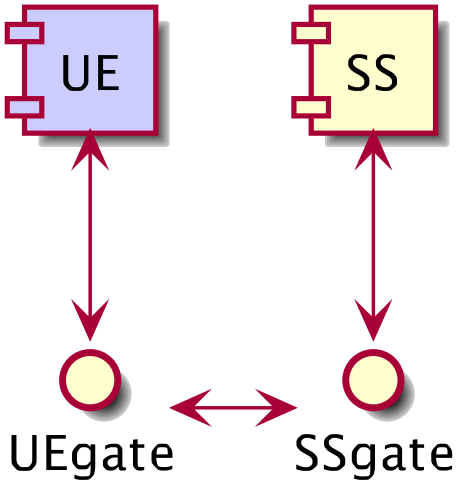

Figure 4: Configuration visualization for sample-3GPP-7.1.3 project.

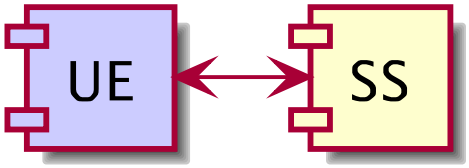

Figure 5: Configuration visualization for sample-3GPP-7.1.3 project (components only).

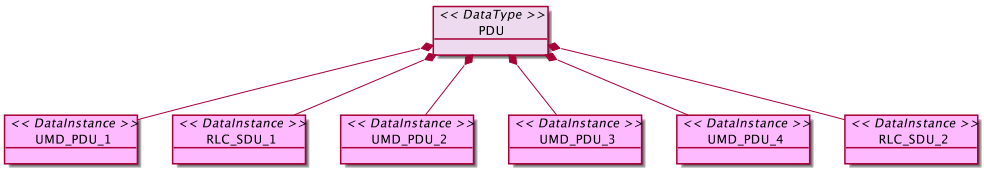

Figure 6: Data concepts visualization for sample-3GPP-7.2.2 project.

2 3GPP 7.2.2

Listing 1: PlantUML-based textual syntax.

```
@startuml
box "Tester SS <defaultCT>" #white
  participant SSgate
end box
box "SUT UE <defaultCT>" #white
  participant UEgate #CCCCFF
end box
=="Preamble:\n- The UE is in state Loopback Activated (
   state 4) according\nto [18] with the exceptions listed
    in table 7.2.2.1.3.1-1\napplicable for the configured
    UM DRB.\n- The condition SRB2-DRB(1,1) is used for
   step 8 in 4.5.3A.3\naccording to [18]."==
SSgate --> UEgate: ["1"] UMD_PDU_1
UEgate --> SSgate: ["2"] RLC_SDU_1
note left: PASS
SSgate --> UEgate: ["3"] UMD_PDU_2
SSgate --> UEgate: ["4"] UMD_PDU_3
SSgate --> UEgate: ["5"] UMD_PDU_4
UEgate --> SSgate: ["6"] RLC_SDU_2
note left: PASS
@enduml
```

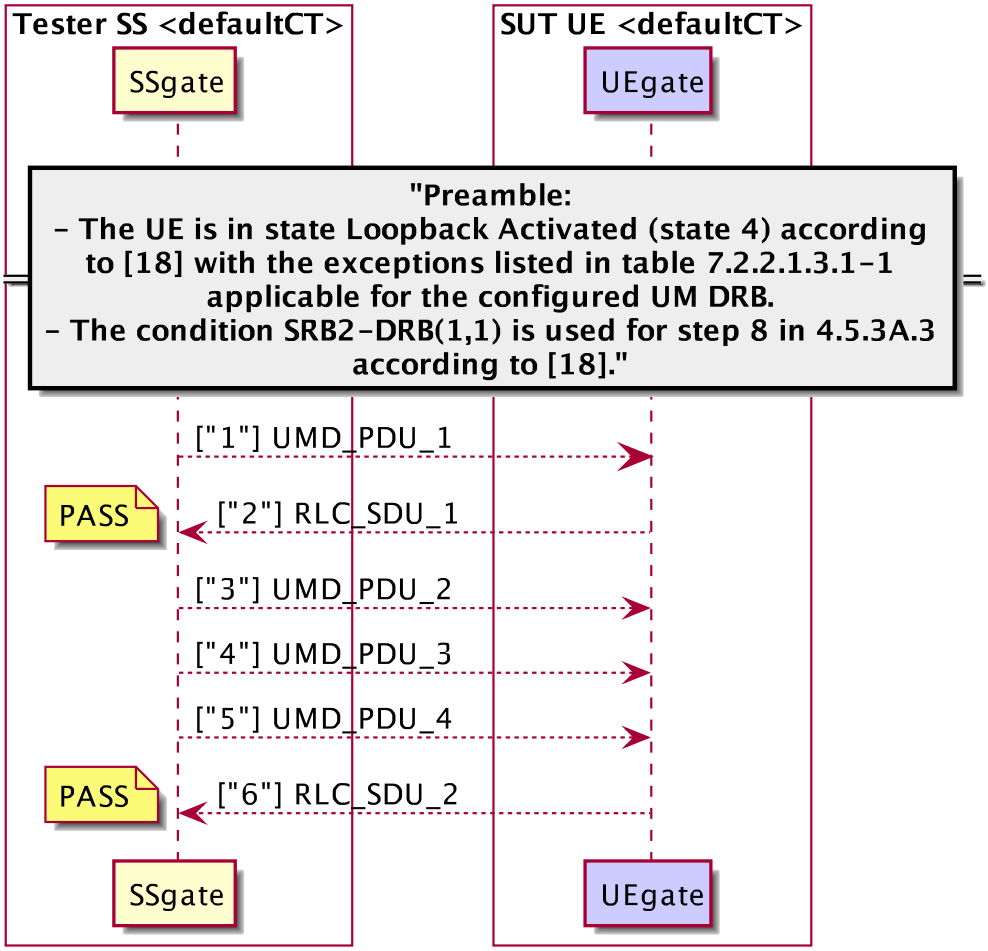

Figure 7: Interactions visualization for sample-3GPP-7.2.2 project.

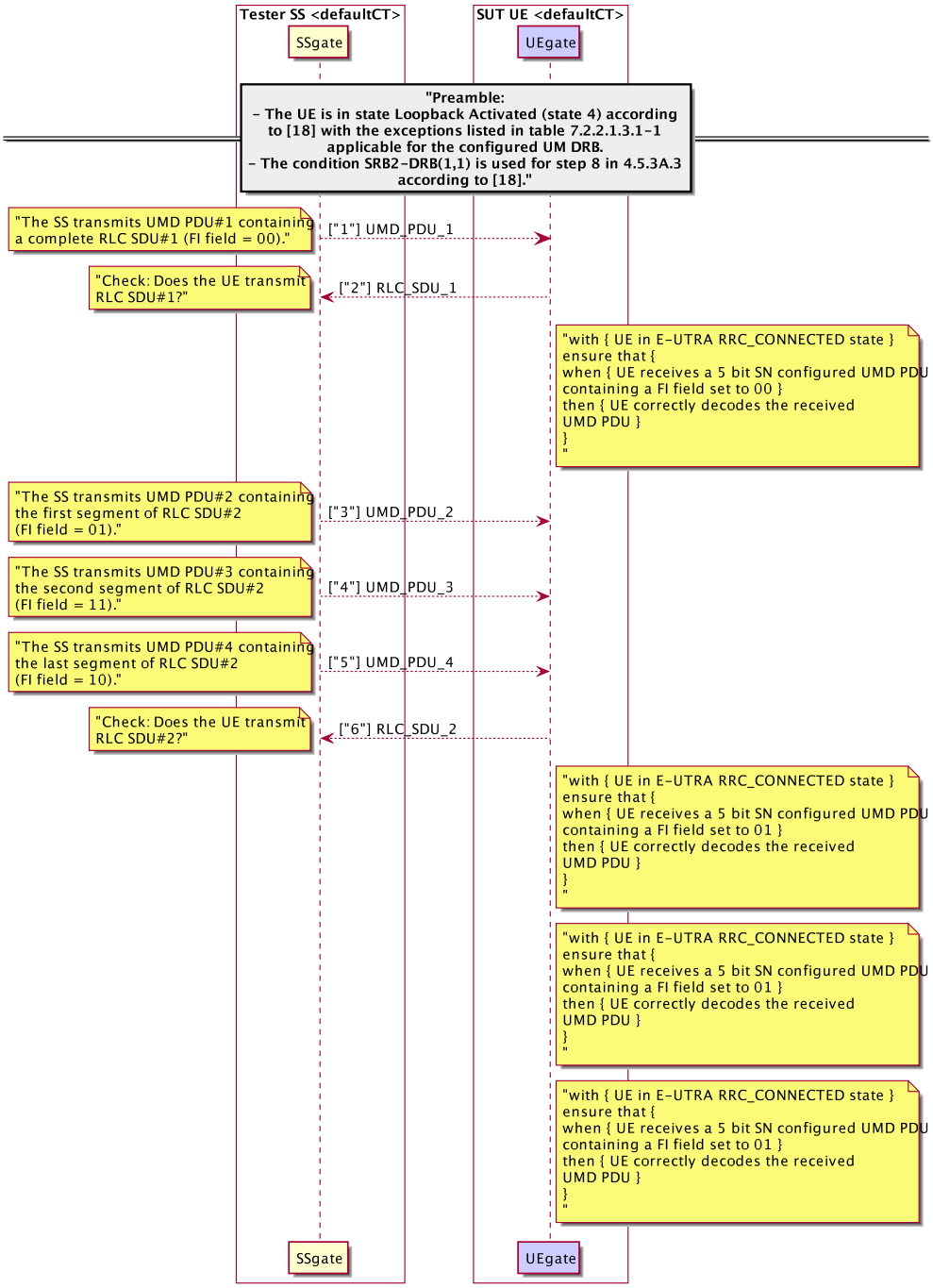

Figure 8: Interactions visualization with notes for sample-3GPP-7.2.2 project.

Step	Procedure	Direction	Message
1	The SS transmits UMD PDU1 containing a complete RLC SDU1 (FI field = 00).	SS –> UE	UMD_PDU_1
2	Check: Does the UE transmit RLC SDU1?	UE –> SS	RLC_SDU_1
3	The SS transmits UMD PDU2 containing the first segment of RLC SDU2 (FI field = 01).	SS –> UE	UMD_PDU_2
4	The SS transmits UMD PDU3 containing the second segment of RLC SDU2 (FI field = 11).	SS –> UE	UMD_PDU_3
5	The SS transmits UMD PDU4 containing the last segment of RLC SDU2 (FI field = 10).	SS –> UE	UMD_PDU_4
6	Check: Does the UE transmit RLC SDU2?	UE –> SS	RLC_SDU_2

Table 2: Tabular syntax.

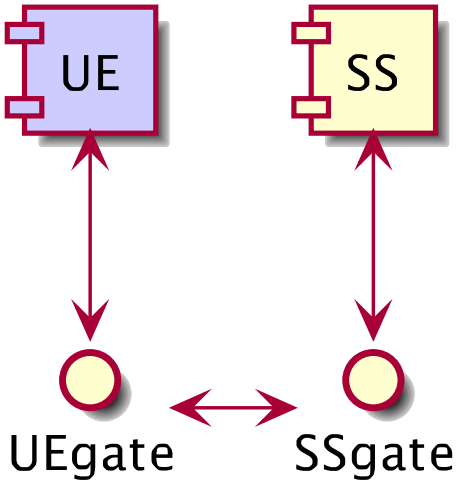

Figure 9: Configuration visualization for sample-3GPP-7.2.2 project.

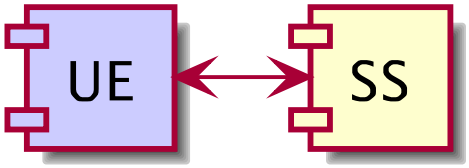

Figure 10: Configuration visualization for sample-3GPP-7.2.2 project (components only).

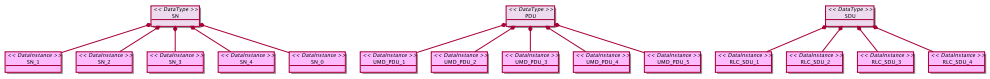

Figure 11: Data concepts visualization for sample-3GPP-7.2.2.3 project.

3 3GPP 7.2.2.3

Listing 2: PlantUML-based textual syntax.

```
@startuml
box "Tester SS <defaultCT>" #white
  participant SSgate
end box
box "SUT UE <defaultCT>" #white
  participant UEgate #CCCCFF
end box
=="Preamble:\n- The UE is in state Loopback Activated (
   state 4) according\nto [18] with the exceptions listed
    in table 7.2.2.3.3.1-1\napplicable for the configured
    UM DRB.\n- The condition SRB2-DRB(1,1) is used for
   step 8 in 4.5.3A.3\naccording to [18]."==
group Parallel
  SSgate --> UEgate: ["1"] UMD_PDU_1 (SN_0)
  SSgate --> UEgate: ["2"] UMD_PDU_2 (SN_1)
  SSgate --> UEgate: ["3"] UMD_PDU_3 (SN_2)
  SSgate --> UEgate: ["4"] UMD_PDU_4 (SN_3)
  SSgate --> UEgate: ["5"] UMD_PDU_5 (SN_4)
else
  UEgate --> SSgate: ["1"] RLC_SDU_1
  UEgate --> SSgate: ["2"] RLC_SDU_2
note left: FAIL
  UEgate --> SSgate: ["3"] RLC_SDU_3
note left: FAIL
  UEgate --> SSgate: ["4"] RLC_SDU_4
else
end
@enduml
```

Step	Procedure	Direction	Message
1	The SS transmits UMD PDU1 containing first segment of RLC SDU1.	SS –> UE	UMD_PDU_1 (SN_0)
2	The SS transmits UMD PDU2 containing last segment of RLC SDU1 and first segment of RLC SDU2.	SS –> UE	UMD_PDU_2 (SN_1)
3	The SS transmits UMD PDU3 containing last segment of RLC SDU2, first segment of RLC SDU3 and with Length Indicator that points beyond the end of the UMD PDU3..	SS –> UE	UMD_PDU_3 (SN_2)
4	The SS transmits UMD PDU4 containing last segment of RLC SDU3.	SS –> UE	UMD_PDU_4 (SN_3)
5	The SS transmits UMD PDU5 containing RLC SDU4.	SS –> UE	UMD_PDU_5 (SN_4)
else 1	The UE transmits RLC SDU1.	UE –> SS	RLC_SDU_1
2	Check: Does the UE transmit RLC SDU2?	UE –> SS	RLC_SDU_2
3	Check: Does the UE transmit RLC SDU3?	UE –> SS	RLC_SDU_3
4	The UE transmits RLC SDU4.	UE –> SS	RLC_SDU_4
else			

Table 3: Tabular syntax.

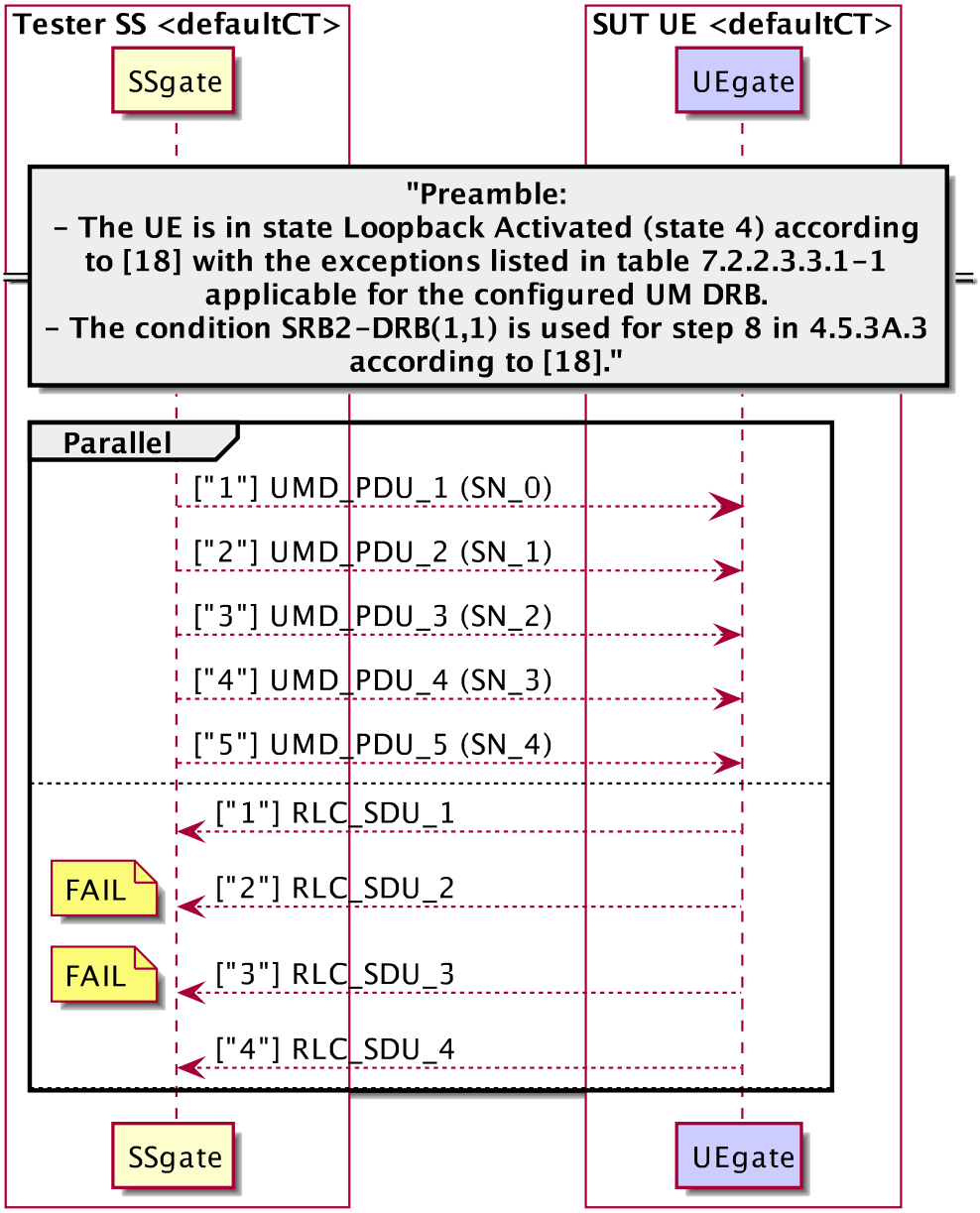

Figure 12: Interactions visualization for sample-3GPP-7.2.2.3 project.

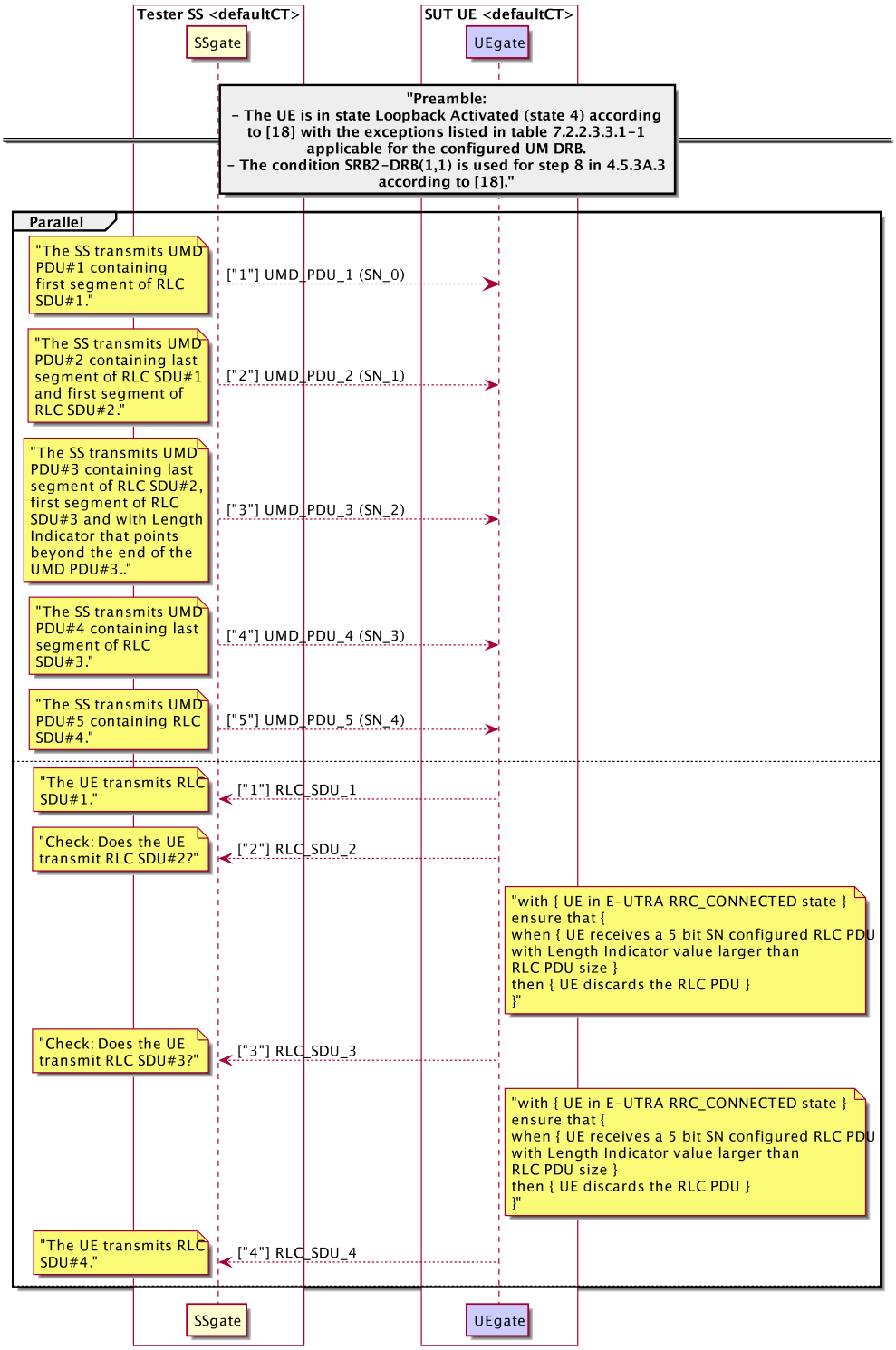

Figure 13: Interactions visualization with notes for sample-3GPP-7.2.2.3 project.

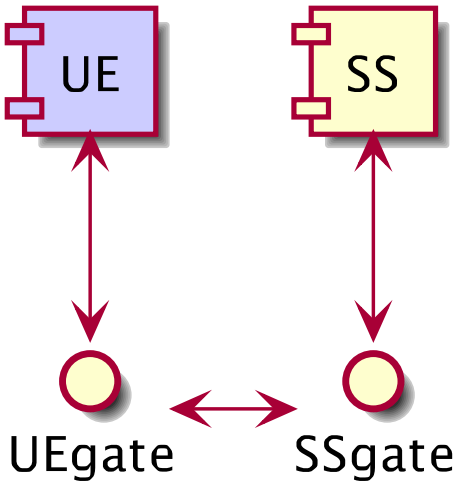

Figure 14: Configuration visualization for sample-3GPP-7.2.2.3 project.

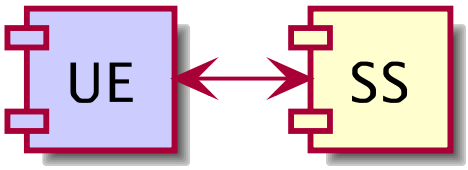

Figure 15: Configuration visualization for sample-3GPP-7.2.2.3 project (components only).

4 IMS 4.5.1

Listing 3: PlantUML-based textual syntax.

```
@startuml
box "Tester USER_A <USER>" #white
  participant gUSER_A
end box
box "Tester UE_A <UE>" #white
  participant gUE_A
end box
box "Tester IMS_A <IMS>" #white
  participant gIMS_A
end box
box "Tester IBCF_A <IBCF>" #white
  participant gIBCF_A
end box
box "Tester IBCF_B <IBCF>" #white
  participant gIBCF_B
end box
box "SUT IMS_B <IMS>" #white
  participant gIMS_B #CCCCFF
end box
box "Tester UE_B <UE>" #white
  participant gUE_B
end box
box "Tester USER_B <USER>" #white
  participant gUSER_B
end box
=="Pre-test conditions:\n- HSS of IMS_A and of IMS B is
   configured according to table 1\n- UE_A and UE_B have
   IP bearers established to their respective\nIMS
   networks as per clause 4.2.1\n- UE_A and IMS_A
   configured to use TCP for transport\n- UE_A is
   registered in IMS_A using any user identity\n- UE_B is
    registered user of IMS_B using any user identity\n-
   MESSAGE request and response has to be supported at II
   -NNI\n(TS 129 165 [16]\nsee tables 6.1 and 6.3)"==
gUSER_A --> gUE_A: ["1"] MESSAGE
gUE_A --> gIMS_A: ["2"] MESSAGE
```

Step	Procedure	Direction	Message
1		$\text{USER}_A - - > UE_A$	MESSAGE
2		$\text{UE}_A - - > IMS_A$	MESSAGE
3		$\text{IMS}_A - - > IBCF_A$	MESSAGE
4		$\text{IBCF}_A - - > IBCF_B$	MESSAGE
5		$\text{IBCF}_B - - > IMS_B$	MESSAGE_TCP (TCP)
6		$\text{IMS}_B - - > UE_B$	MESSAGE
7		$\text{UE}_B - - > USER_B$	DING
8		$\text{UE}_B - - > IMS_B$	M_200_OK
9		$\text{IMS}_B - - > IBCF_B$	M_200_OK
10		$\text{IBCF}_B - - > IBCF_A$	M_200_OK
11		$\text{IBCF}_A - - > IMS_A$	M_200_OK
12		$\text{IMS}_A - - > UE_A$	M_200_OK
13		$\text{UE}_A - - > USER_A$	DELIVERY_REPORT

Table 4: Tabular syntax.

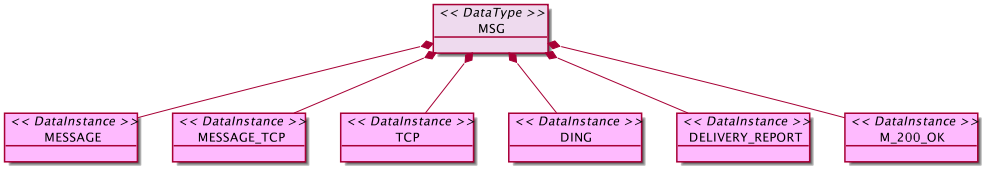

Figure 16: Data concepts visualization for sample-IMS-4.5.1 project.

```
gIMS_A --> gIBCF_A: ["3"] MESSAGE
gIBCF_A --> gIBCF_B: ["4"] MESSAGE
gIBCF_B --> gIMS_B: ["5"] MESSAGE_TCP (TCP)
gIMS_B --> gUE_B: ["6"] MESSAGE
gUE_B --> gUSER_B: ["7"] DING
gUE_B --> gIMS_B: ["8"] M_200_OK
gIMS_B --> gIBCF_B: ["9"] M_200_OK
gIBCF_B --> gIBCF_A: ["10"] M_200_OK
gIBCF_A --> gIMS_A: ["11"] M_200_OK
gIMS_A --> gUE_A: ["12"] M_200_OK
group Optional
  gUE_A --> gUSER_A: ["13"] DELIVERY_REPORT
end
@enduml
```

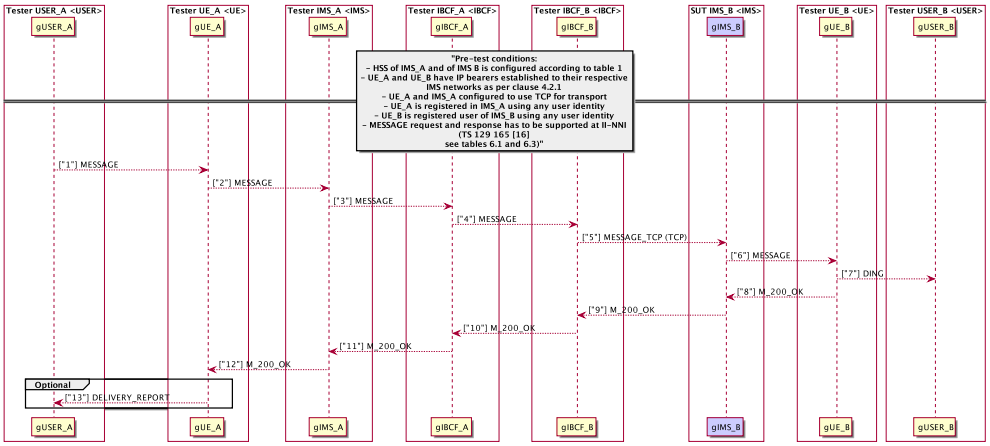

Figure 17: Interactions visualization for sample-IMS-4.5.1 project.

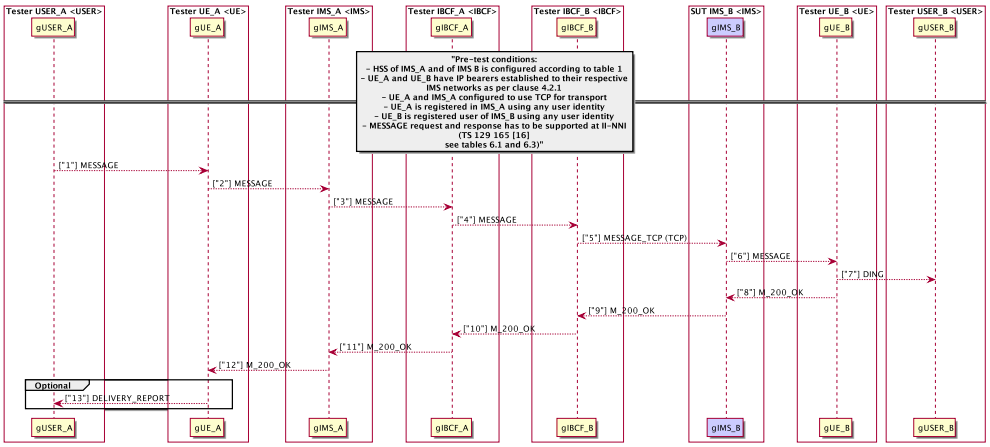

Figure 18: Interactions visualization with notes for sample-IMS-4.5.1 project.

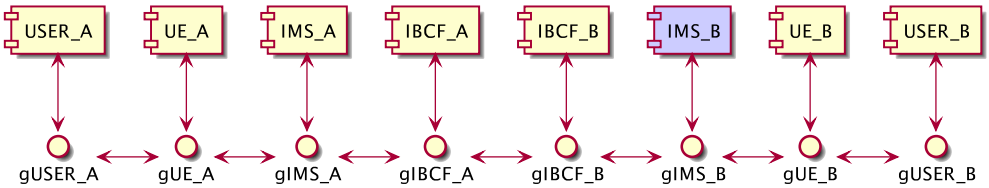

Figure 19: Configuration visualization for sample-IMS-4.5.1 project.

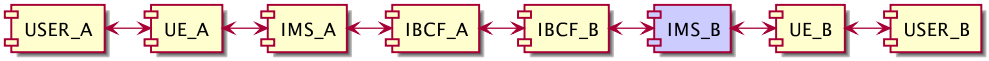

Figure 20: Configuration visualization for sample-IMS-4.5.1 project (components only).

Step	Procedure	Direction	Message
		ci1 –> ci2	MSG_1 (SESSION_ID_1)
		ci2 –> ci1	RESPONSE_1 (SESSION_ID_1)
		ci1 –> ci2	ANY_STATUS (STATUS)

Table 5: Tabular syntax.

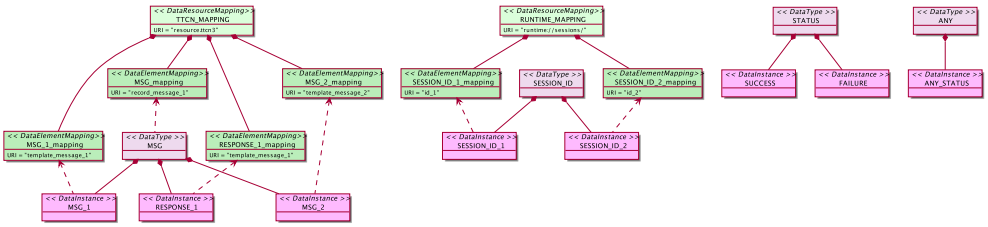

Figure 21: Data concepts visualization for sample-data project.

5 Data Concepts

Listing 4: PlantUML-based textual syntax.

```
@startuml
box "SUT ci1 <ct1>" #white
  participant gi1 #CCCCFF
  participant gi3 #CCCCFF
end box
box "Tester ci2 <ct1>" #white
  participant gi2
end box
gi1 --> gi2: MSG_1 (SESSION_ID_1)
gi2 --> gi1: RESPONSE_1 (SESSION_ID_1)
gi1 --> gi2: ANY_STATUS (STATUS)
@enduml
```

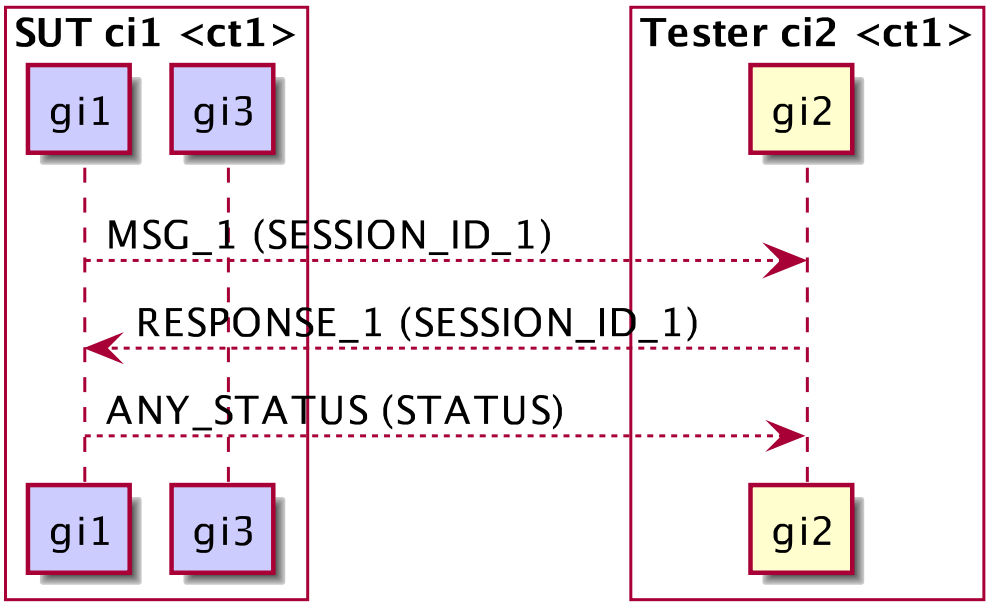

Figure 22: Interactions visualization for sample-data project.

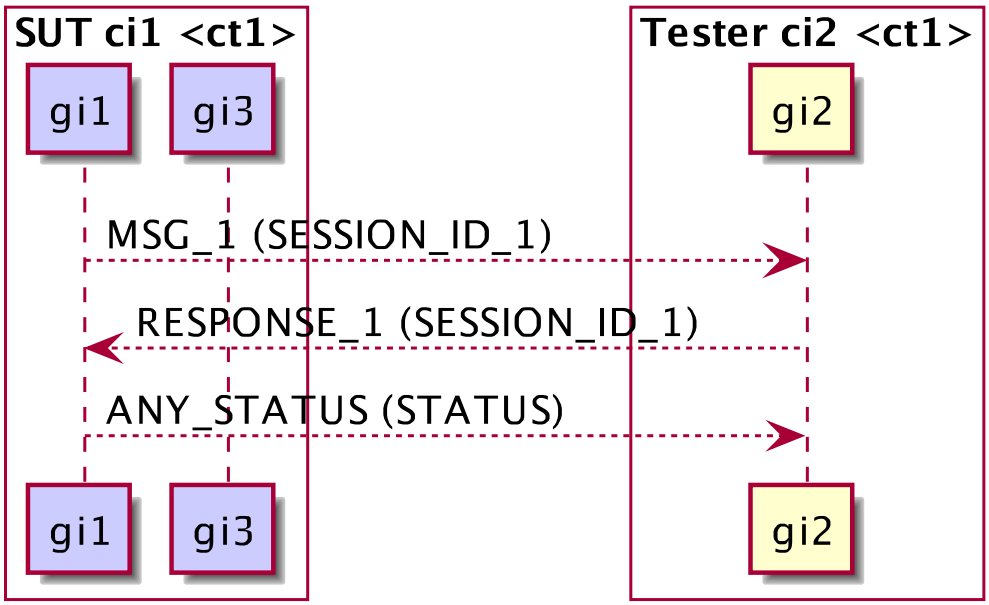

Figure 23: Interactions visualization with notes for sample-data project.

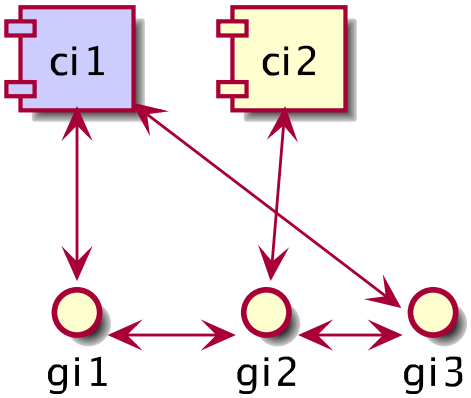

Figure 24: Configuration visualization for sample-data project.

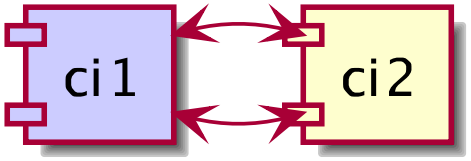

Figure 25: Configuration visualization for sample-data project (components only).

Step	Procedure	Direction	Message
		ciT –> ci	I2 (T1)

Table 6: Tabular syntax.

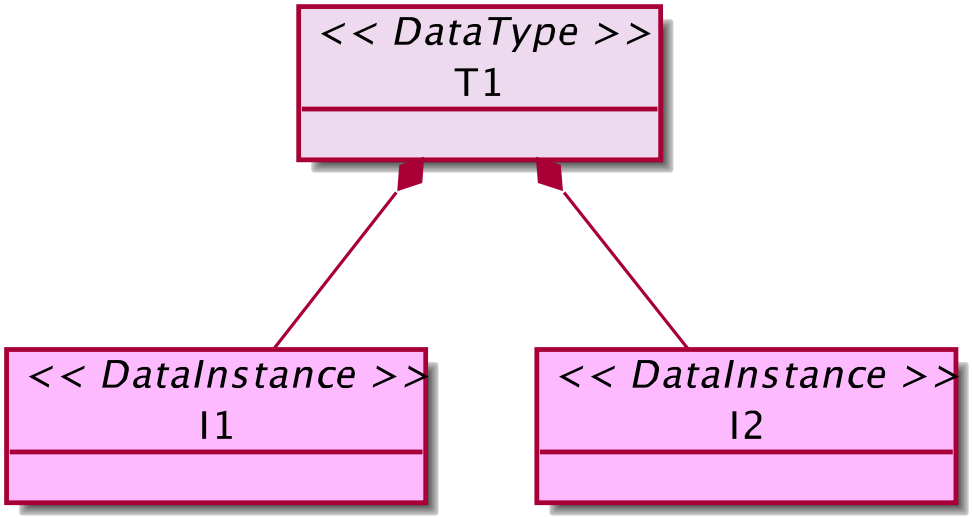

Figure 26: Data concepts visualization for sample-basic project.

6 Behaviour Concepts

Listing 5: PlantUML-based textual syntax.

```
@startuml
box "SUT ci <ct>" #white
  participant gi1 #CCCCFF
end box
box "Tester ciT <ct>" #white
  participant gi2
end box
gi2 --> gi1: I2 (T1)
note left: PASS
@enduml
```

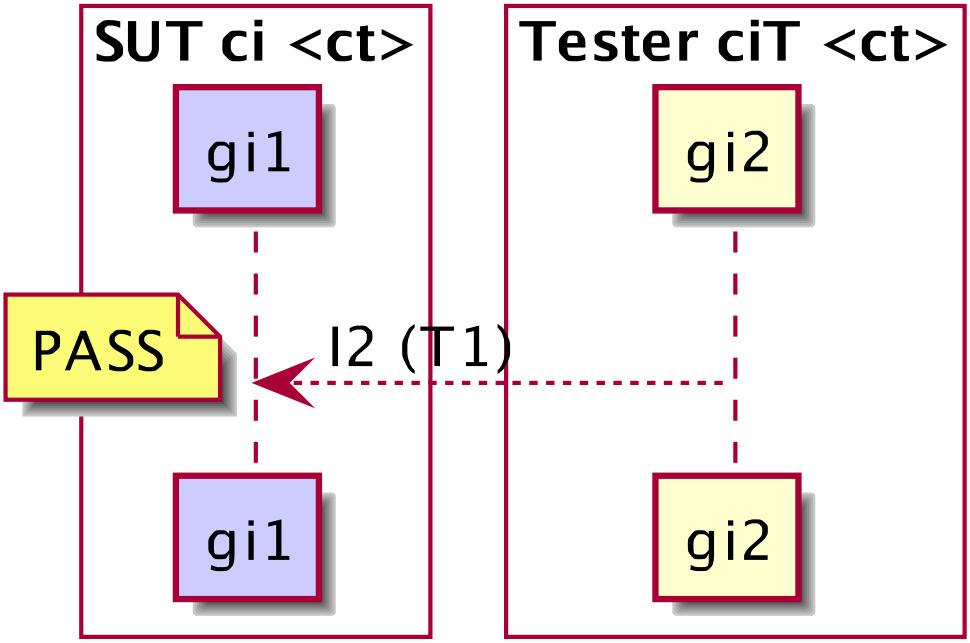

Figure 27: Interactions visualization for sample-basic project.

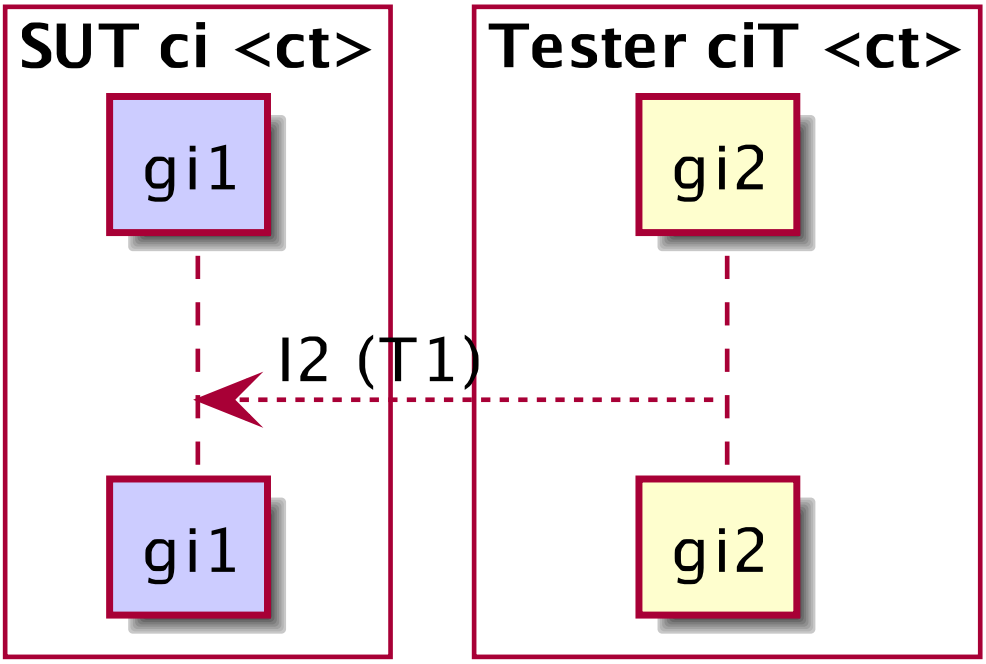

Figure 28: Interactions visualization with notes for sample-basic project.

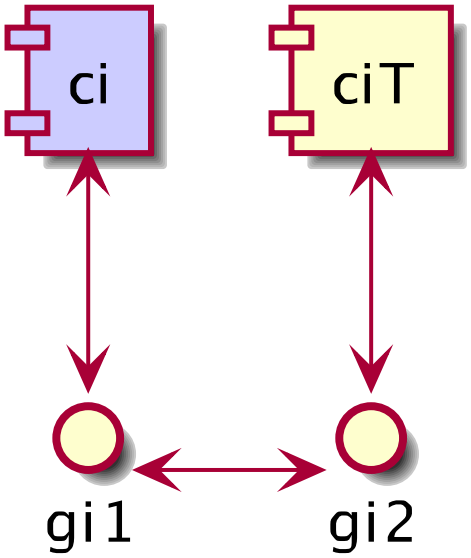

Figure 29: Configuration visualization for sample-basic project.

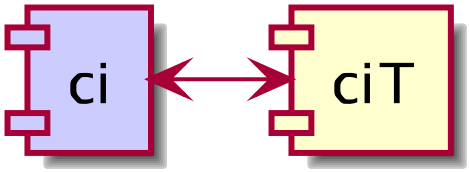

Figure 30: Configuration visualization for sample-basic project (components only).

7 Annex: IMS 4.5.1 Complete Textual Model Instance

The complete textual model instance for the IMS example is included below for illustrative purposes. Similar textual representaions of the other examples are also available, but omitted here for brevity. These textual representations are very verbose and illustrate 1-to-1 mapping to meta-model concepts.

Listing 6: Full xText-based representation of IMS example.

```
TDLSpecification {
        name IMS_NNI_General_Capabilities
        comment {
                Comment {
                        body "Taken from ETSI TS 186 011-2 V3.1.1 (2011-06)"
                }
        }

        annotation {
                Annotation {
                        key TITLE
                        value "SIP messages longer than 1 500 bytes"
                }
        }
        content {
                Action {
                        name preConditions
                        body "Pre-test conditions:
                                 - HSS of IMS_A and of IMS B is configured according to table 1
                                 - UE_A and UE_B have IP bearers established to their respective
                                   IMS networks as per clause 4.2.1
                                 - UE_A and IMS_A configured to use TCP for transport
                                 - UE_A is registered in IMS_A using any user identity
                                 - UE_B is registered user of IMS_B using any user identity
                                 - MESSAGE request and response has to be supported at II-NNI
                                   (TS 129 165 [16]
                                 see tables 6.1 and 6.3)"
                },
                VerdictType {
                        name PASS
                },
                VerdictType {
                        name FAIL
                },
                AnnotationType {
                        name STEP
                },
                AnnotationType {
                        name PROCEDURE
                },
                AnnotationType {
                        name TITLE
                },
                AnnotationType {
                        name SUMMARY
                },
                TestObjective{
                        name TP_IMS_4002_1
                        objectiveURI {"ts_18601102v030101p.pdf::4.5.1.1 (CC 1)",
                                        "TS 124 229 [1], clause 4.2A, paragraph 1"}
                        description "ensure that {
                                          when { UE_A sends a MESSAGE to UE_B
                                                         containing a Message_Body greater
                                                               than 1 300
                                                 bytes }
                                          then { IMS_B receives the MESSAGE containing the
                                                     Message_Body greater than 1 300 bytes }
                                        }"
                },
                TestObjective {
```

```
        name UC_05_I
        objectiveURI {"ts_18601102v030101p.pdf::4.4.4.2"}
},
DataType {
        name MSG
        dataInstance {
                DataInstance {
                        name MESSAGE
                },
                DataInstance {
                        name MESSAGE_TCP
                        dataInstanceParameter (TCP)
                },
                DataInstance {
                        name TCP
                },
                DataInstance {
                        name DING
                },
                DataInstance {
                        name DELIVERY_REPORT
                },
                DataInstance {
                        name M_200_OK
                }

        }
},
ComponentType{
        name USER
        gate (
                defaultGT
        )
},
ComponentType{
        name UE
        gate (
                defaultGT
        )
},
ComponentType{
        name IMS
        gate (
                defaultGT
        )
},
ComponentType{
        name IBCF
        gate (
                defaultGT
        )
},

GateType {
        name defaultGT
        dataType (
                MSG
        )
},
TestConfiguration {
        name CF_INT_CALL
        componentInstance {
                ComponentInstance {
                        name USER_A
                        role Tester
                        type USER
                        gateInstance {
                                GateInstance {
                                        name gUSER_A
                                        type defaultGT
                                }
                        }
                },
                ComponentInstance {
                        name UE_A
                        role Tester
                        type UE
```

```
                gateInstance {
                        GateInstance {
                                name gUE_A
                                type defaultGT
                        }
                }
        },
        ComponentInstance {
                name IMS_A
                role Tester
                type IMS
                gateInstance {
                        GateInstance {
                                name gIMS_A
                                type defaultGT
                        }
                }
        },
        ComponentInstance {
                name IBCF_A
                role Tester
                type IBCF
                gateInstance {
                        GateInstance {
                                name gIBCF_A
                                type defaultGT
                        }
                }
        },
        ComponentInstance {
                name IBCF_B
                role Tester
                type IBCF
                gateInstance {
                        GateInstance {
                                name gIBCF_B
                                type defaultGT
                        }
                }
        },
        ComponentInstance {
                name IMS_B
                role SUT
                type IMS
                gateInstance {
                        GateInstance {
                                name gIMS_B
                                type defaultGT
                        }
                }
        },
        ComponentInstance {
                name UE_B
                role Tester
                type UE
                gateInstance {
                        GateInstance {
                                name gUE_B
                                type defaultGT
                        }
                }
        },
        ComponentInstance {
                name USER_B
                role Tester
                type USER
                gateInstance {
                        GateInstance {
                                name gUSER_B
                                type defaultGT
                        }
                }
        }
}
connection {
        Connection {
                endPoint (
```

```
                                gUSER_A, gUE_A
                        )
                },
                Connection {
                        endPoint (
                                gUE_B, gUSER_B
                        )
                },
                Connection {
                        endPoint (
                                gUE_A, gIMS_A
                        )
                },
                Connection {
                        endPoint (
                                gIMS_A, gIBCF_A
                        )
                },
                Connection {
                        endPoint (
                                gIBCF_A, gIBCF_B
                        )
                },
                Connection {
                        endPoint (
                                gIBCF_B, gIMS_B
                        )
                },
                Connection {
                        endPoint (
                                gIMS_B, gUE_B
                        )
                }

        }
},
TestDescription {
        name TD_IMS_MESS_0001
        testConfiguration CF_INT_CALL
        annotation {
                Annotation {
                        key SUMMARY
                        value "IMS network shall support SIP messages greater than
                                    1 500 bytes"
                }
        }
        behaviour CompoundBehaviour {
                block Block {
                        behaviour {
                                ActionReference {
                                        action preConditions
                                },
                                Interaction {
                                        argument MESSAGE
                                        sender gUSER_A
                                        receiver (gUE_A)
                                        annotation {
                                                Annotation {
                                                        key STEP
                                                        value "1"
                                                }
                                        }
                                },
                                Interaction {
                                        argument MESSAGE
                                        sender gUE_A
                                        receiver (gIMS_A)
                                        annotation {
                                                Annotation {
                                                        key STEP
                                                        value "2"
                                                }
                                        }
                                },
                                Interaction {
                                        argument MESSAGE
                                        sender gIMS_A
```

```
        receiver (gIBCF_A)
        annotation {
                Annotation {
                        key STEP
                        value "3"
                }
        }
},
Interaction {
        argument MESSAGE
        sender gIBCF_A
        receiver (gIBCF_B)
        annotation {
                Annotation {
                        key STEP
                        value "4"
                }
        }
},
Interaction {
        argument MESSAGE_TCP
        sender gIBCF_B
        receiver (gIMS_B)
        annotation {
                Annotation {
                        key STEP
                        value "5"
                }
        }
},
Interaction {
        argument MESSAGE
        sender gIMS_B
        receiver (gUE_B)
        annotation {
                Annotation {
                        key STEP
                        value "6"
                }
        }
},
Interaction {
        argument DING
        sender gUE_B
        receiver (gUSER_B)
        annotation {
                Annotation {
                        key STEP
                        value "7"
                }
        }
},
Interaction {
        argument M_200_OK
        sender gUE_B
        receiver (gIMS_B)
        annotation {
                Annotation {
                        key STEP
                        value "8"
                }
        }
},
Interaction {
        argument M_200_OK
        sender gIMS_B
        receiver (gIBCF_B)
        annotation {
                Annotation {
                        key STEP
                        value "9"
                }
        }
},
Interaction {
        argument M_200_OK
        sender gIBCF_B
```

```
                                                        receiver (gIBCF_A)
                                                        annotation {
                                                                Annotation {
                                                                        key STEP
                                                                        value "10"
                                                                }
                                                        }
                                                },
                                                Interaction {
                                                        argument M_200_OK
                                                        sender gIBCF_A
                                                        receiver (gIMS_A)
                                                        annotation {
                                                                Annotation {
                                                                        key STEP
                                                                        value "11"
                                                                }
                                                        }
                                                },
                                                Interaction {
                                                        argument M_200_OK
                                                        sender gIMS_A
                                                        receiver (gUE_A)
                                                        annotation {
                                                                Annotation {
                                                                        key STEP
                                                                        value "12"
                                                                }
                                                        }
                                                },
                                                OptionalBehaviour {
                                                        block Block {
                                                                behaviour {
                                                                        Interaction {
                                                                                argument
                                                                                     DELIVERY_REPORT
                                                                                sender gUE_A
                                                                                receiver (gUSER_A)
                                                                                annotation {
                                                                                        Annotation {
                                                                                                key
                                                                                                     STEP

                                                                                                value

                                                                                                     "13"

                                                                                        }
                                                                                }
                                                                        }
                                                                }
                                                        }
                                                }
                                        }
                                }
                        }
                }

        }
}
```