

TDL – BRIDGING THE GAP BETWEEN SPECIFICATIONS AND TESTING

DR. GUSZTÁV ADAMIS
DR. GYÖRGY RÉTHY

ERICSSON HUNGARY
TEST SOLUTIONS AND COMPETENCE CENTER

CONTENTS



Problem definition

TDL - the new ETSI test language

Role of TDL in testing

Activities on TDL

TEST DEVELOPMENT MANUAL WORKFLOW



- › Requirement and System specification
 - Feature description, IW Description etc.

Input

ERICSSON

Prepared (also subject responsible if other)
ETXJASO Jack Song

Approved
BICPAAHAB [Jack Song]

2.2 Use

The following

ERICSSON

Ericsson Internal
FUNCTIONAL SPEC. 20 (135)

Prepared (also subject responsible if other)
ETXJASO Jack Song

Approved
BICPAAHAB [Jack Song]

No.
29/155 17-HSD 101 96/1 Uen

Date
2016-08-30

Rev
PJ1

Reference
I

3.2.2 Main Scenarios

3.2.2.1 Successful originating session

The Figure 7 below is a simplified signaling sequence.

```
sequenceDiagram
    participant UE as Served UE in PS network
    participant SBG
    participant ATGW
    participant PCRF
    participant SCCAS

    UE->>SBG: 1. INVITE (SDP UE-A)
    SBG->>ATGW: 2. ADD
    ATGW-->>SBG: 3. ADD reply
    SBG->>SCCAS: 4. INVITE (SDP ATGW-B)
    SCCAS-->>SBG: 5. 200 OK (SDP UE-B)
    SBG->>ATGW: 6. MODIFY
    ATGW-->>SBG: 7. MODIFY reply
    SBG->>PCRF: 8. AAR
    PCRF-->>SBG: 9. AAA
```

TEST DEVELOPMENT MANUAL WORKFLOW



> Test design

– Done at the test execution language level

Output

Test cases

```
testcase TC_BTP_PP_BV_02() runs on ItsBtp system ItsBtpSystem {  
  
    // Preamble  
    f_prInitialState();  
    f_selfOrClientSyncAndVerdictPreamble(c_prDone, e_success);  
  
    // Test Body  
    v_btpReq := valueof(m_btpReq( m_btpBWithPorts (f_getBtpDstPort(), f_getBtpDstPortInfo(), f_getBtpPayload()
```

API
(product
specific)

Test
Framework

```
/** @desc Requests to bring the IUT in an initial state  
 * @param p_init The initialisation to trigger.  
 */  
function f_utInitializeIut(template (value) UtInitialize p_init) runs on ItsBtp {  
  
    //deactivate btpPort default alts  
    vc_btpDefaultActive := false;  
  
    utPort.send(p_init);  
    tc_wait.start;  
    alt {  
        [] utPort.receive(UtInitializeResult:true) {  
            tc_wait.stop;  
            log("*** f_utInitializeIut: INFO: IUT initialized ***");  
        }  
    }  
}
```



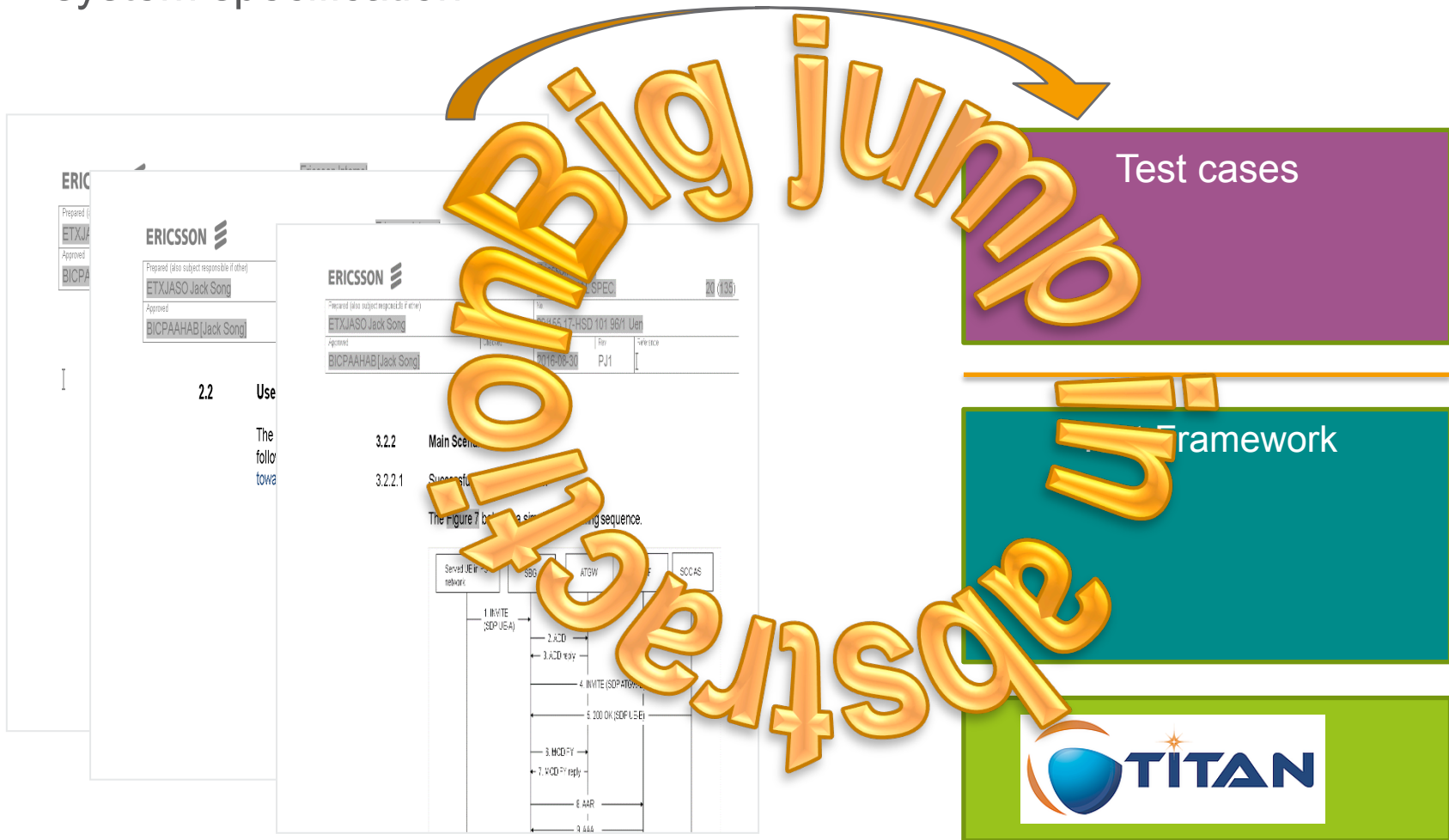
Test tool
(IDE, execution environment)

ABSTRACTION GAP

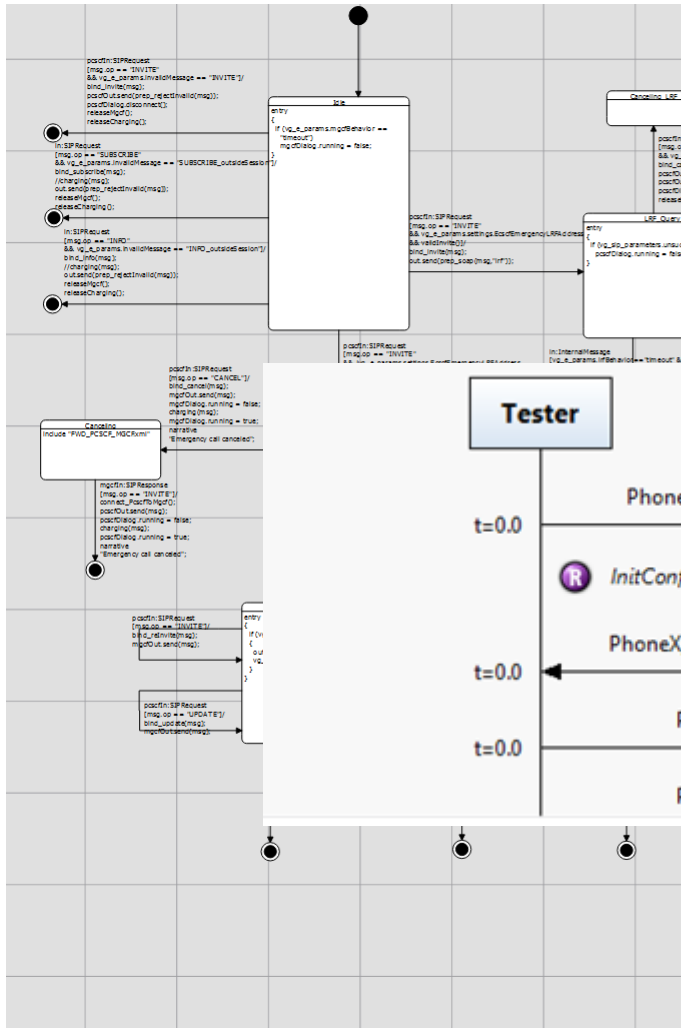


Requirement management/
system specification

Test design/execution



TEST DEVELOPMENT MBT WORKFLOW



```
public void reselection(String p_legId)
```

```
{
```

```
mgcfIn);
```

```
vg_sip_parameters.withoutRinging = true;
```

```
mgcfDialog = new SIPCallTerm(mgcfOut,
```

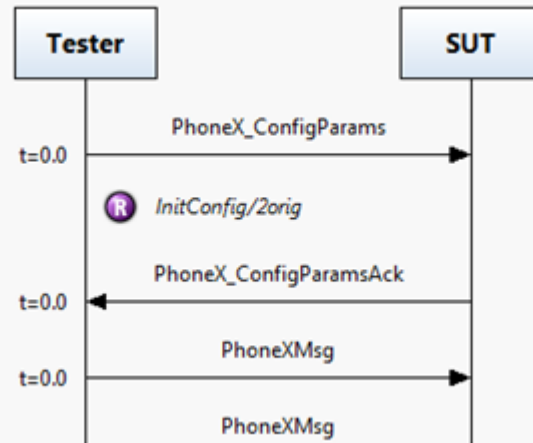
```
mgcfDialog.legId = "mgcf1";
```

```
mgcfDialog.teardown = false;
```

```
GCF1");
```

```
:f2Behavior ==
```

```
log.finalResp = 503;
```



Test harness
in destination language, e.g.



THE ABSTRACTION GAP



MBT requires knowledge on

- Modeling level
- Programming level

WE NEED A LANGUAGE THAT...



Can describe the test cases...

- Both generated and manually designed

at the abstraction level of the model

Can be used also by non-programmers

Enables incremental development

- From system specification to test case design level

Standardised

- Can be used by different tools

Has a graphical representation



TDL – Test Description Language

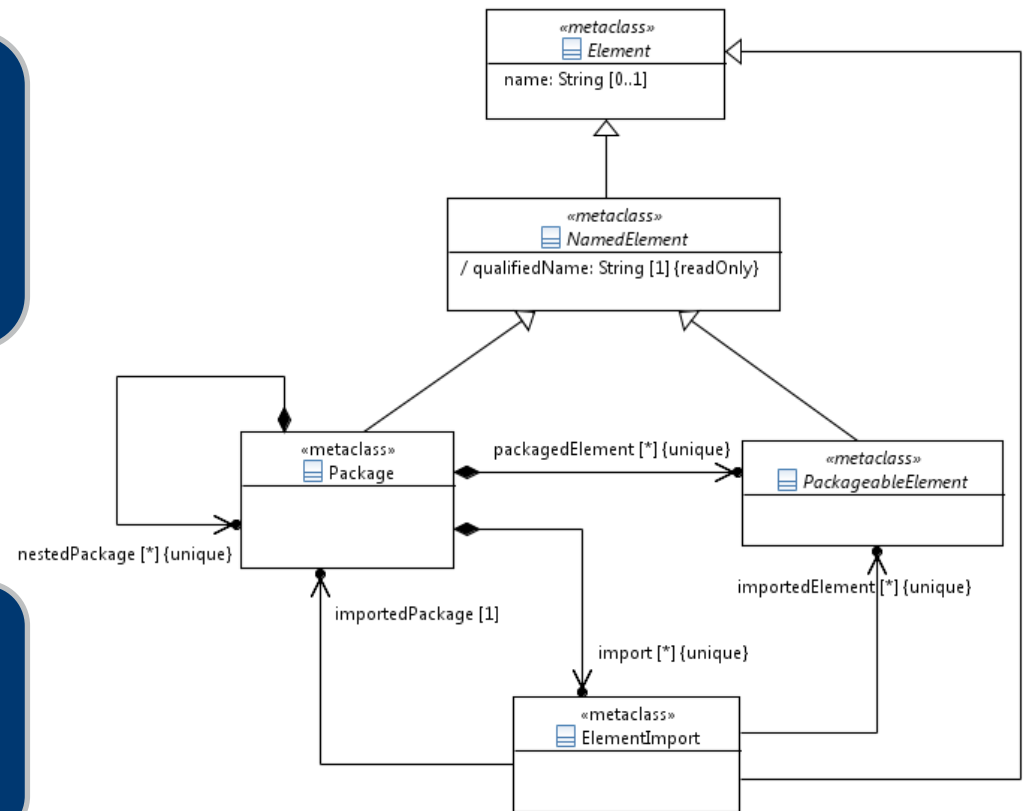
TDL META-MODEL



Well-defined language constructs

- UML MOF-based description
- OCL constraints

Makes it possible to develop different domain-specific concrete syntaxes



STRUCTURE OF TDL



Test Data

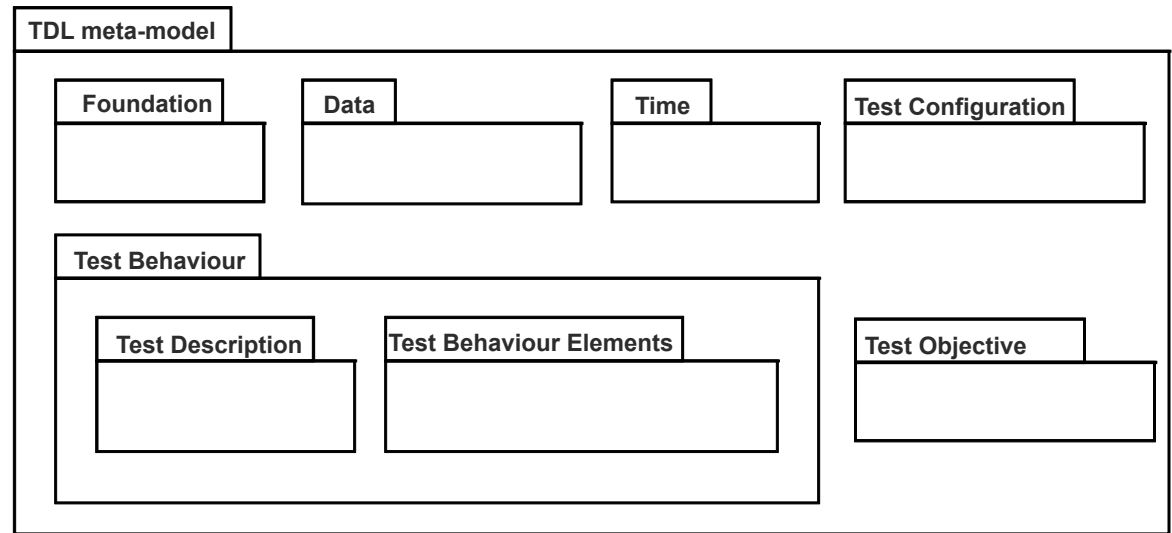
Time, Timers

Test Configuration

Test Behaviour

- Including test-specific constructs, like
 - alternative
 - default
 - verdict

Test Objectives



TDL GRAPHICAL SYNTAX



Standardised by ETSI

Similar approach to UML SD

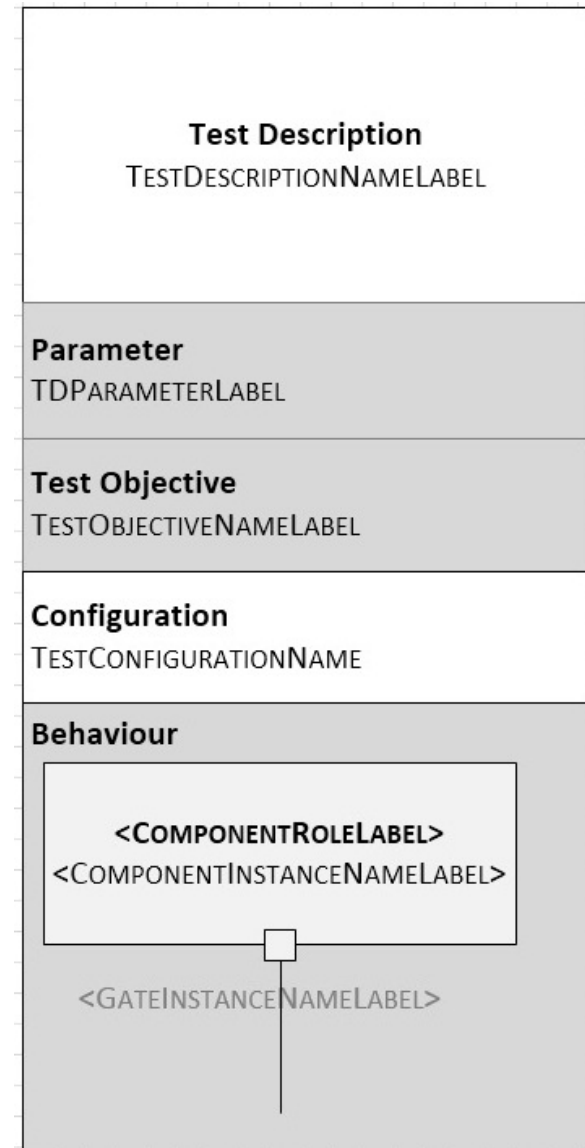
- But new symbols to new constructs

Graphical symbols

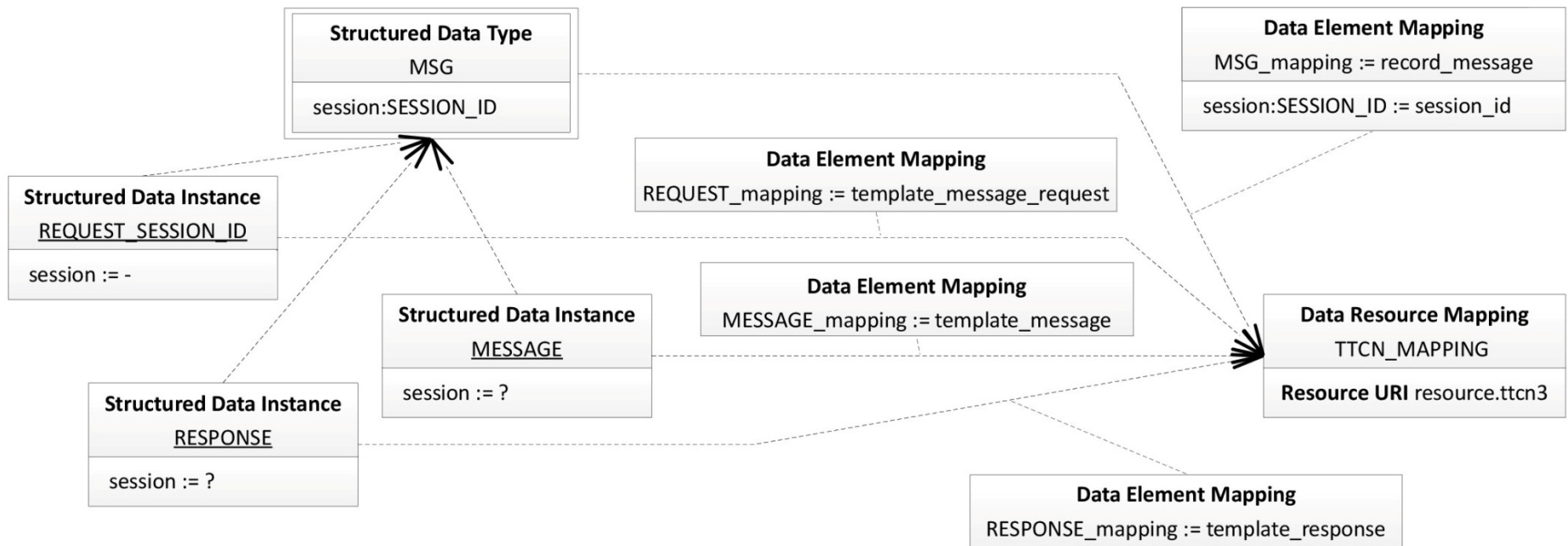
- Formal description of the contained text

```
context TestDescription
TESTDESCRIPTIONNAMELABEL ::= self as context in <NAMEELEMENTLABEL>

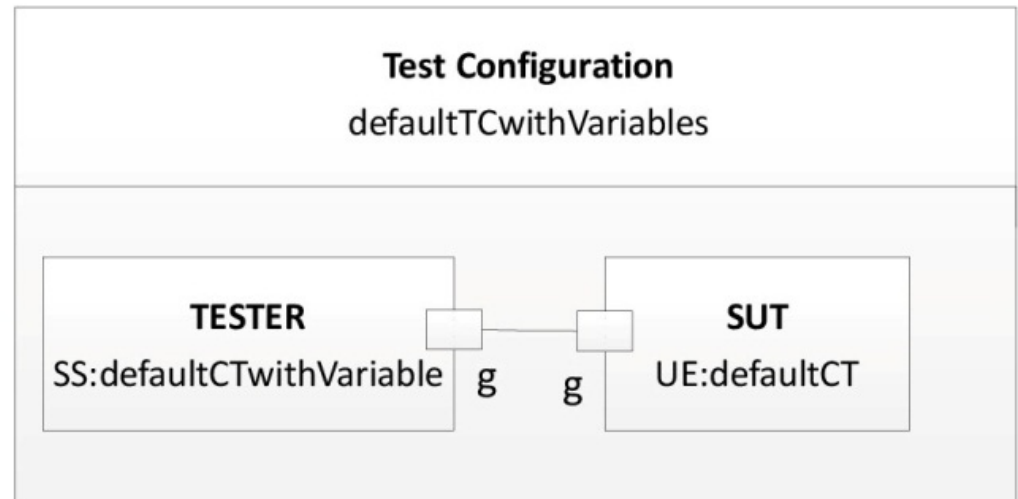
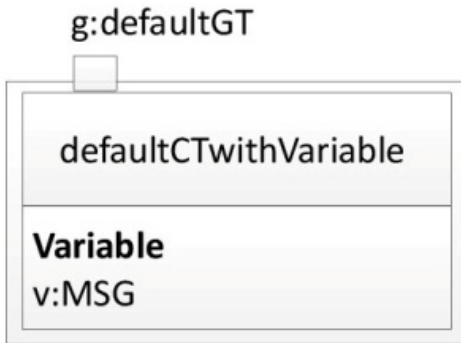
TDPARAMETERLABEL ::= foreach p:Parameter in self.formalParameter separator(',')
                    p as context in <ParameterLabel>
                    end
TESTOBJECTIVENAMELABEL ::= foreach t:TestObjective in self.testObjective newline()
                          t as context in <NAMEELEMENTLABEL>
                          end
TESTCONFIGURATIONNAME ::= self.testConfiguration as context in <NAMEELEMENTLABEL>
```



TDL GRAPHICAL SYNTAX DATA SPECIFICATION



TDL GRAPHICAL SYNTAX TEST CONFIGURATION



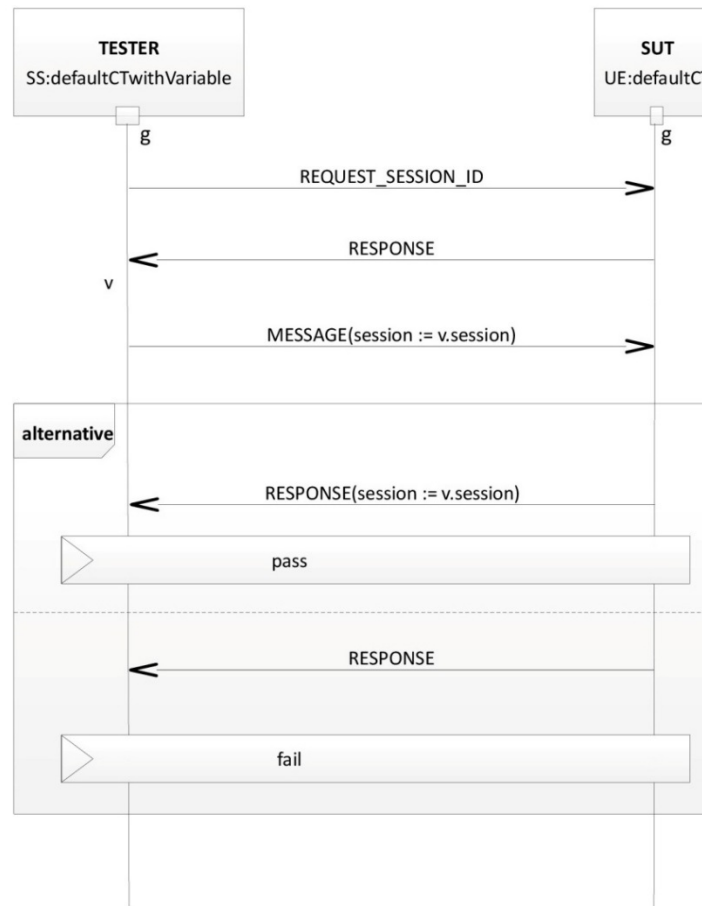
TDL GRAPHICAL SYNTAX TEST BEHAVIOUR



TestDescription
exampleTD

Configuration defaultTCwithVariables

Test Objective CHECK_SESSION_ID_IS_MAINTAINED



TRANSFER SYNTAX

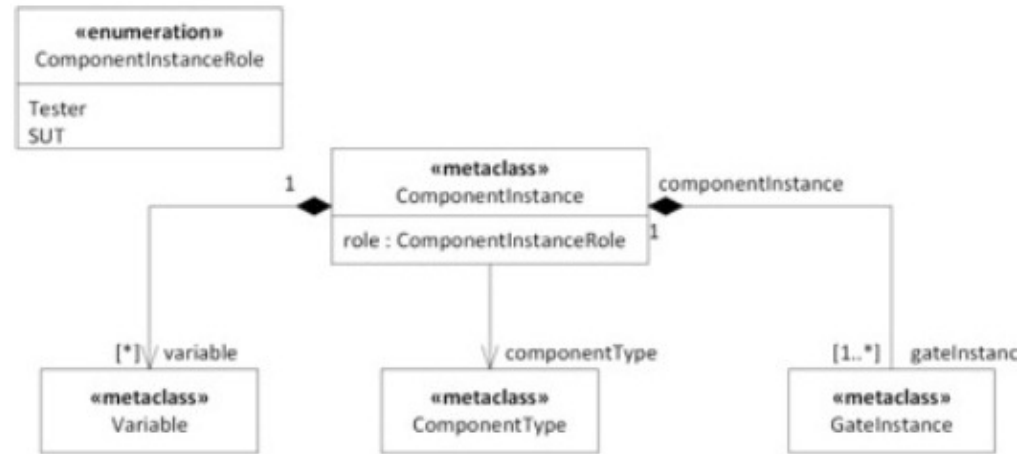


Goals:

- Interoperability between tools
- Interoperability between different concrete syntaxes

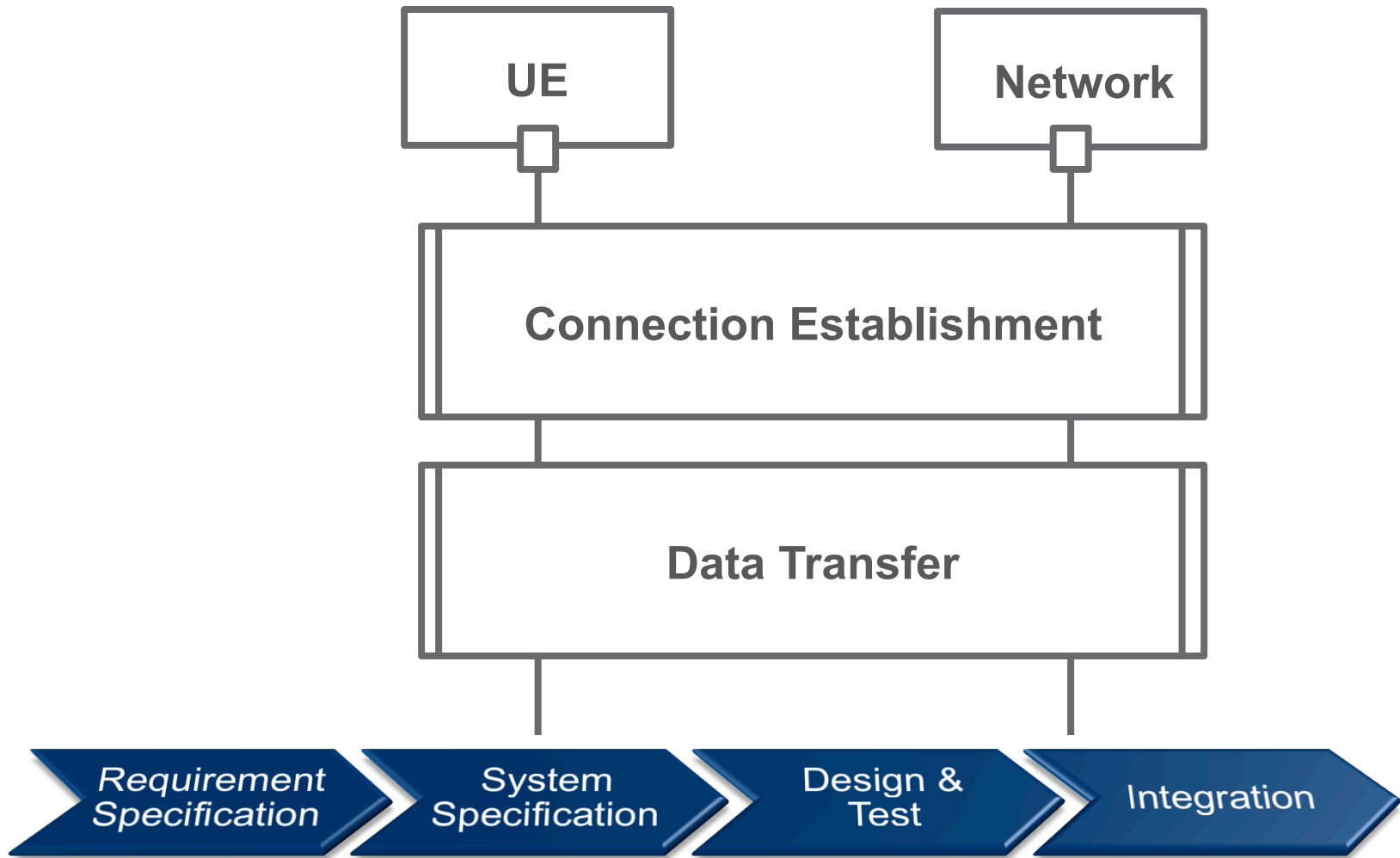
XMI (XML Metadata Interchange)

- To serialize the meta-model
- Syntactical check possible

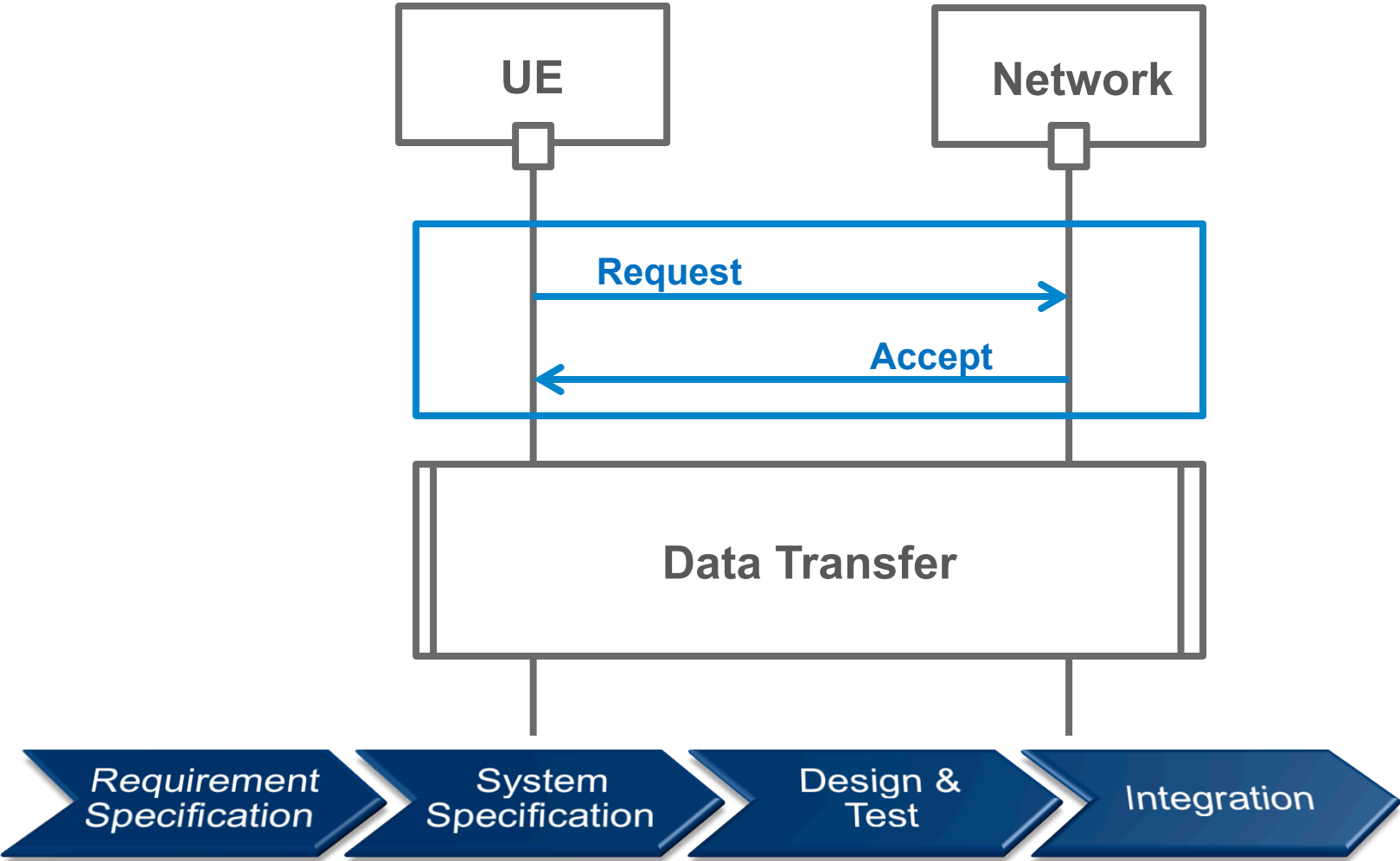


```
<xsd:complexType name="ComponentInstance">
  <xsd:complexContent>
    <xsd:extension base="tdl:Element">
      <xsd:choice maxOccurs="unbounded" minOccurs="0">
        <xsd:element name="gateInstance" type="tdl:GateInstance"/>
        <xsd:element name="variable" type="tdl:Variable"/>
      </xsd:choice>
      <xsd:attribute name="componentType" type="xsd:anyURI">
      <xsd:attribute name="role" type="tdl:ComponentInstanceRole">
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

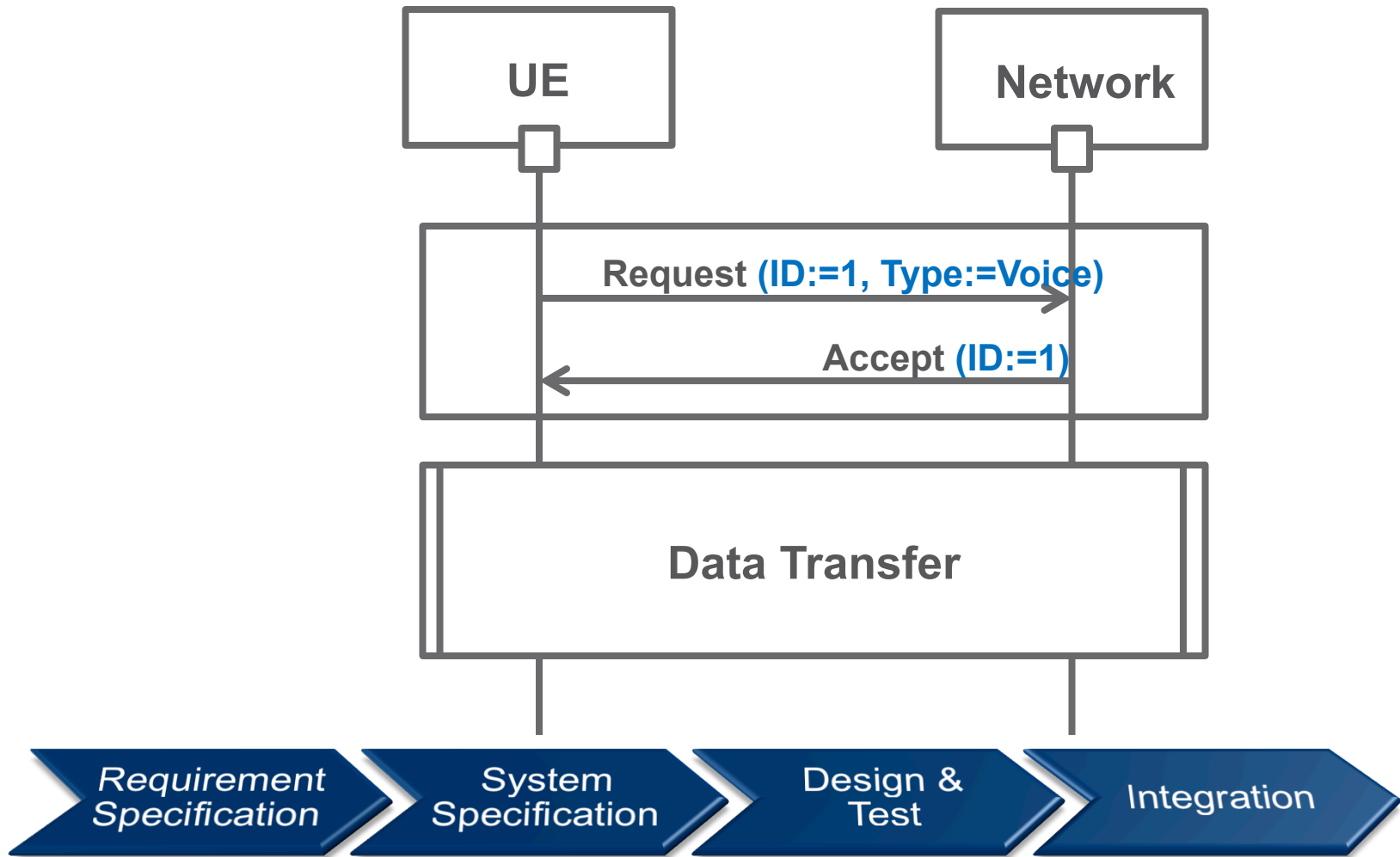
INCREMENTAL DESIGN



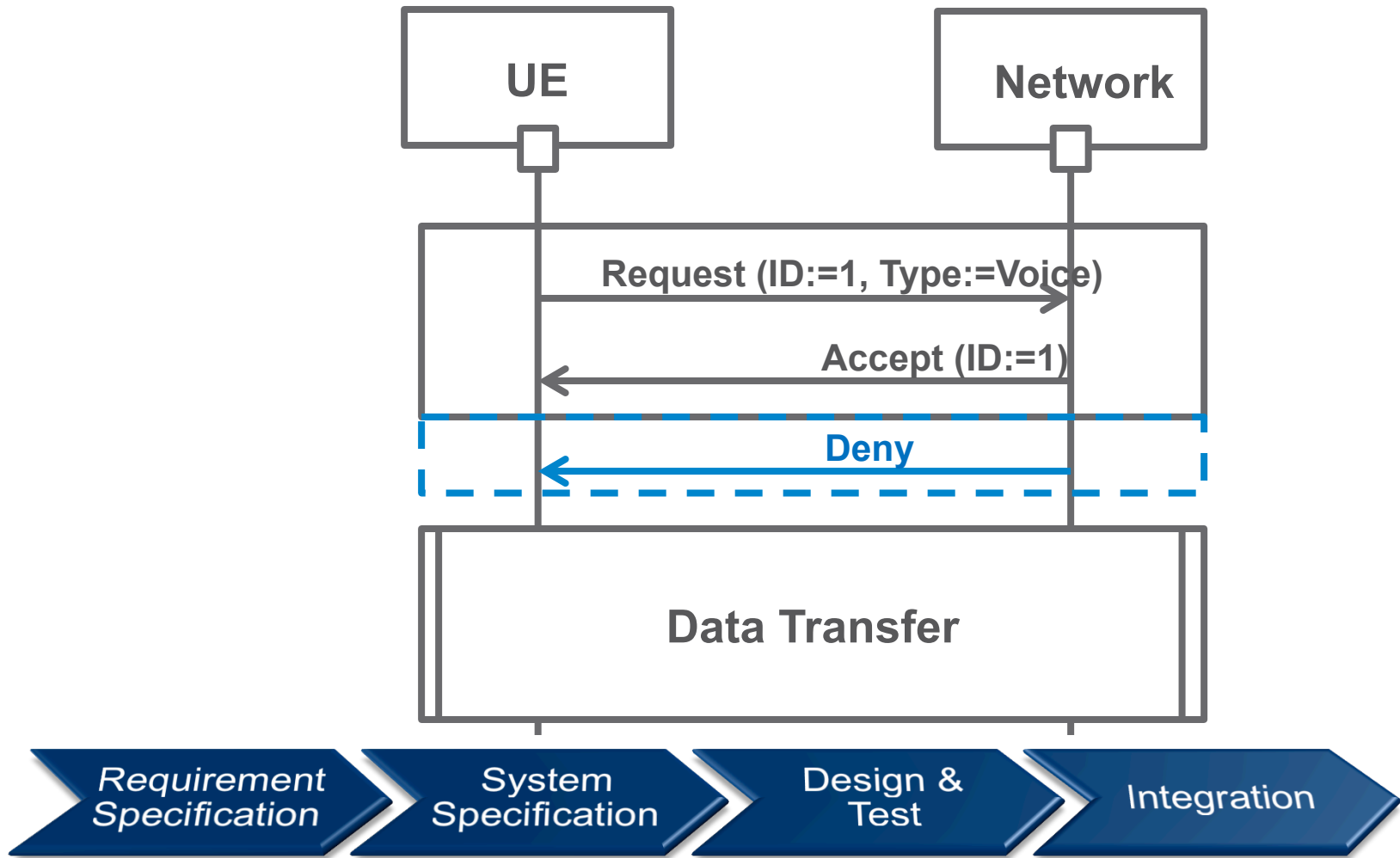
INCREMENTAL DESIGN



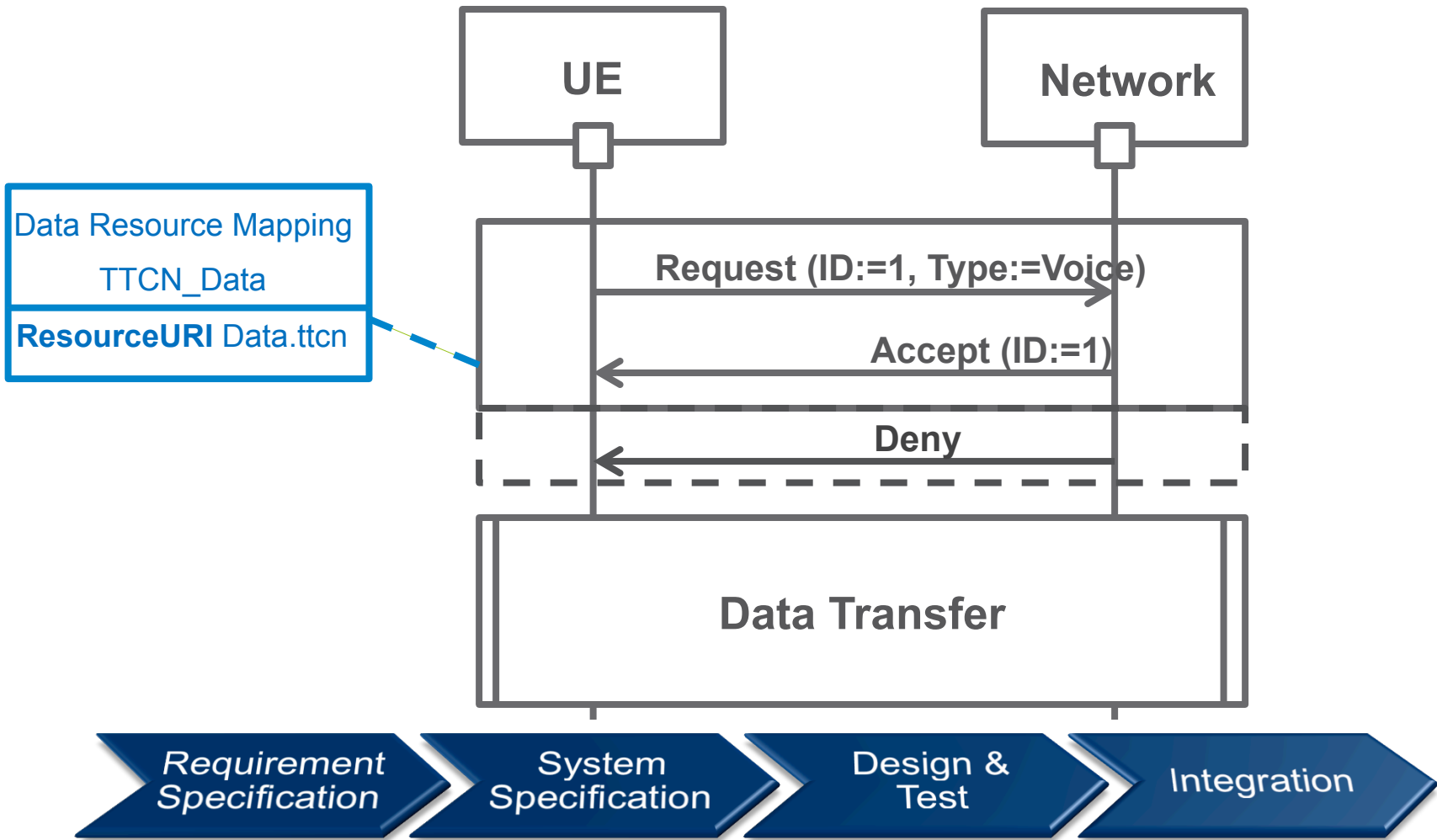
INCREMENTAL DESIGN



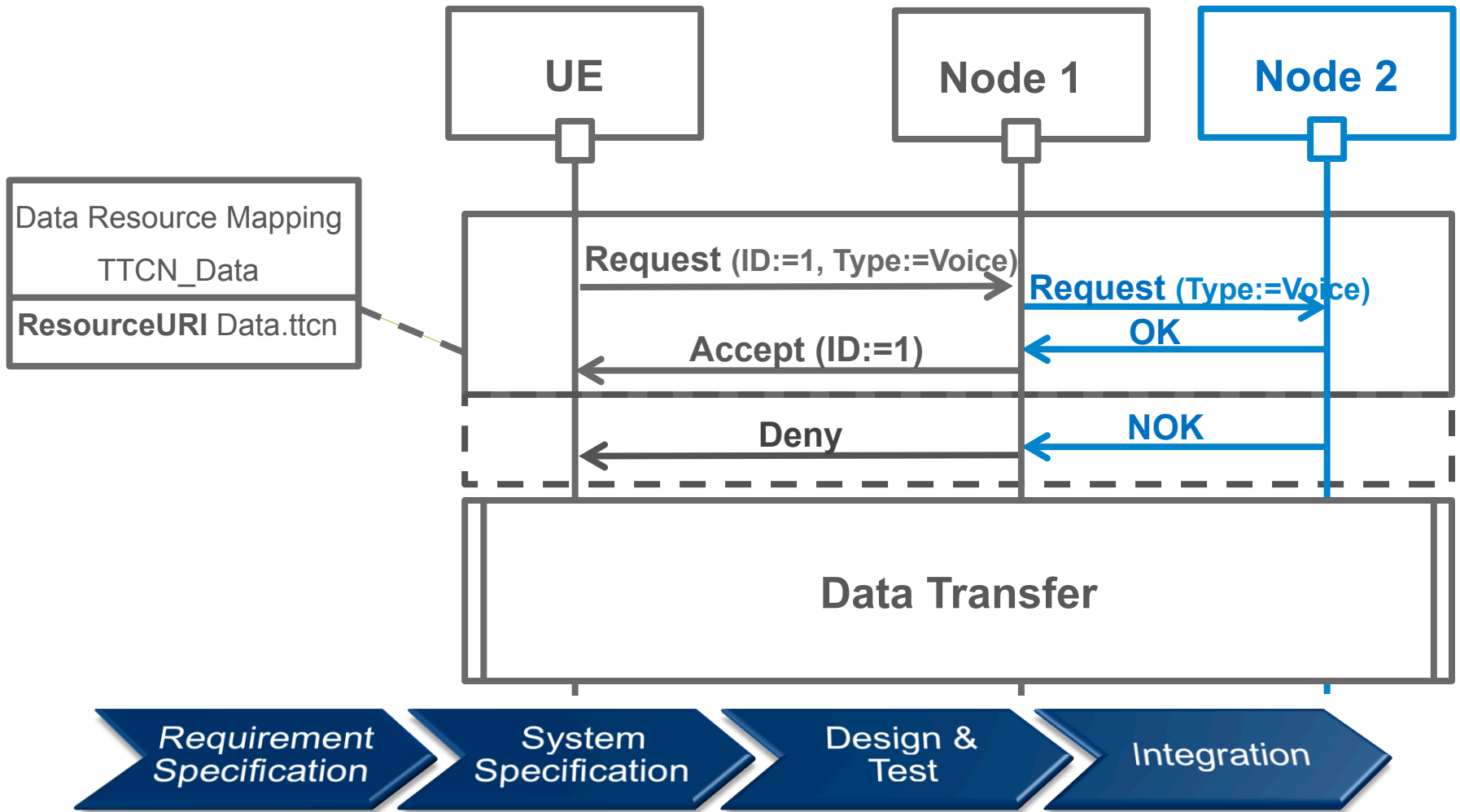
INCREMENTAL DESIGN



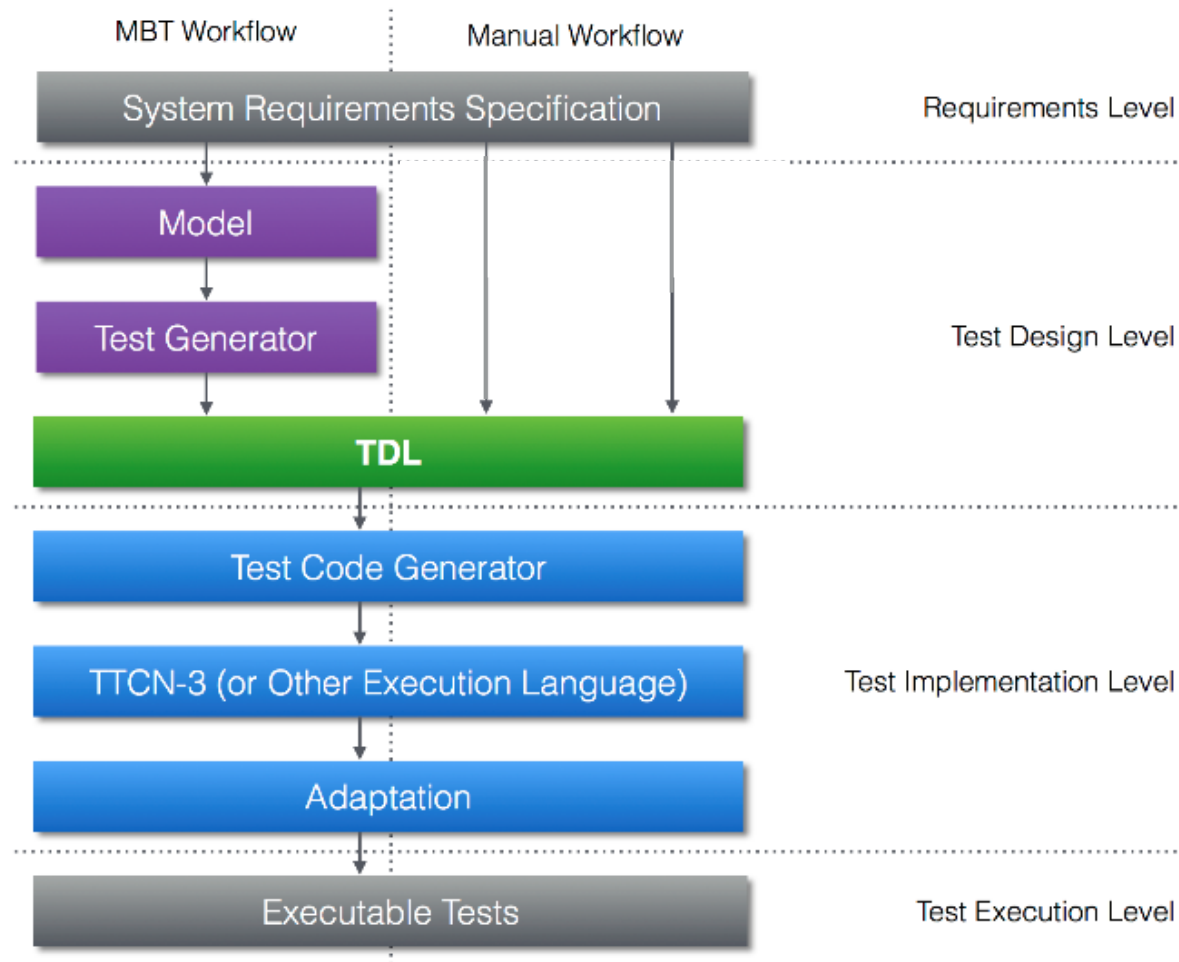
INCREMENTAL DESIGN



INCREMENTAL DESIGN



TDL IN TEST DEVELOPMENT



<http://tdl.etsi.org/index.php/about/introduction>

ACTIVITIES ON TDL



At ETSI:

- › Latest versions of TDL standards are published in 05/2016
See more at <http://tdl.etsi.org>

- › Tool Prototypes – (will be open sourced)
 - Reference TDL-GR viewer
 - › TDL XMI -> TDL GR
 - Papyrus-based TDL editor
 - › TDL -> UML mapping

ACTIVITIES ON TDL



At Ericsson TCC:

› Participation in **all** ETSI TDL activities

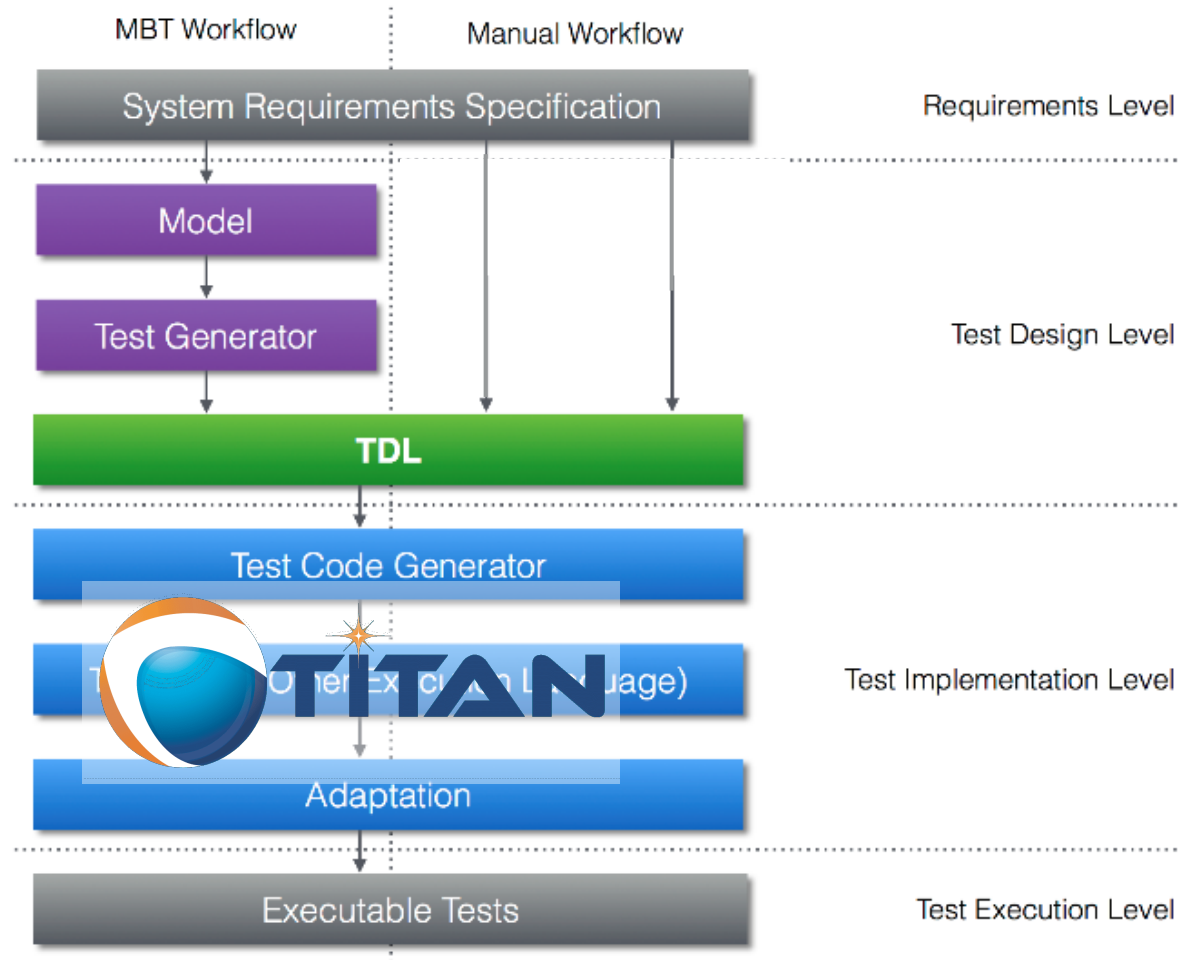
- Language development projects (STFs – Specialized Task Force)
- TB MTS (approving the TDL standards and TDL STF requests)
- TDL Steering Group

› Prototype TDL GR editor integrated into Intuitive tool



› Working on TDL -> TTCN-3 M2M transformation

- › With cooperation of Budapest University of Technology and Economics
 - Mapping of TDL to TTCN-3
 - Prototype tool
 - The coming new ETSI STF will standardize the mapping





ERICSSON