TDL – TEST DESCRIPTION LANGUAGE

GUSZTÁV ADAMIS

ERICSSON HUNGARY
TEST SOLUTIONS AND COMPETENCE CENTER

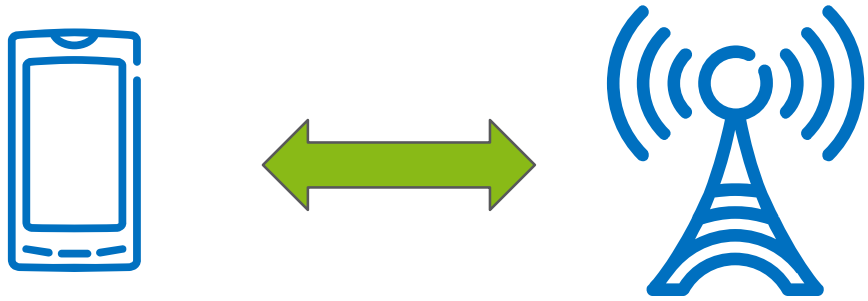
CONTENTS

TESTING

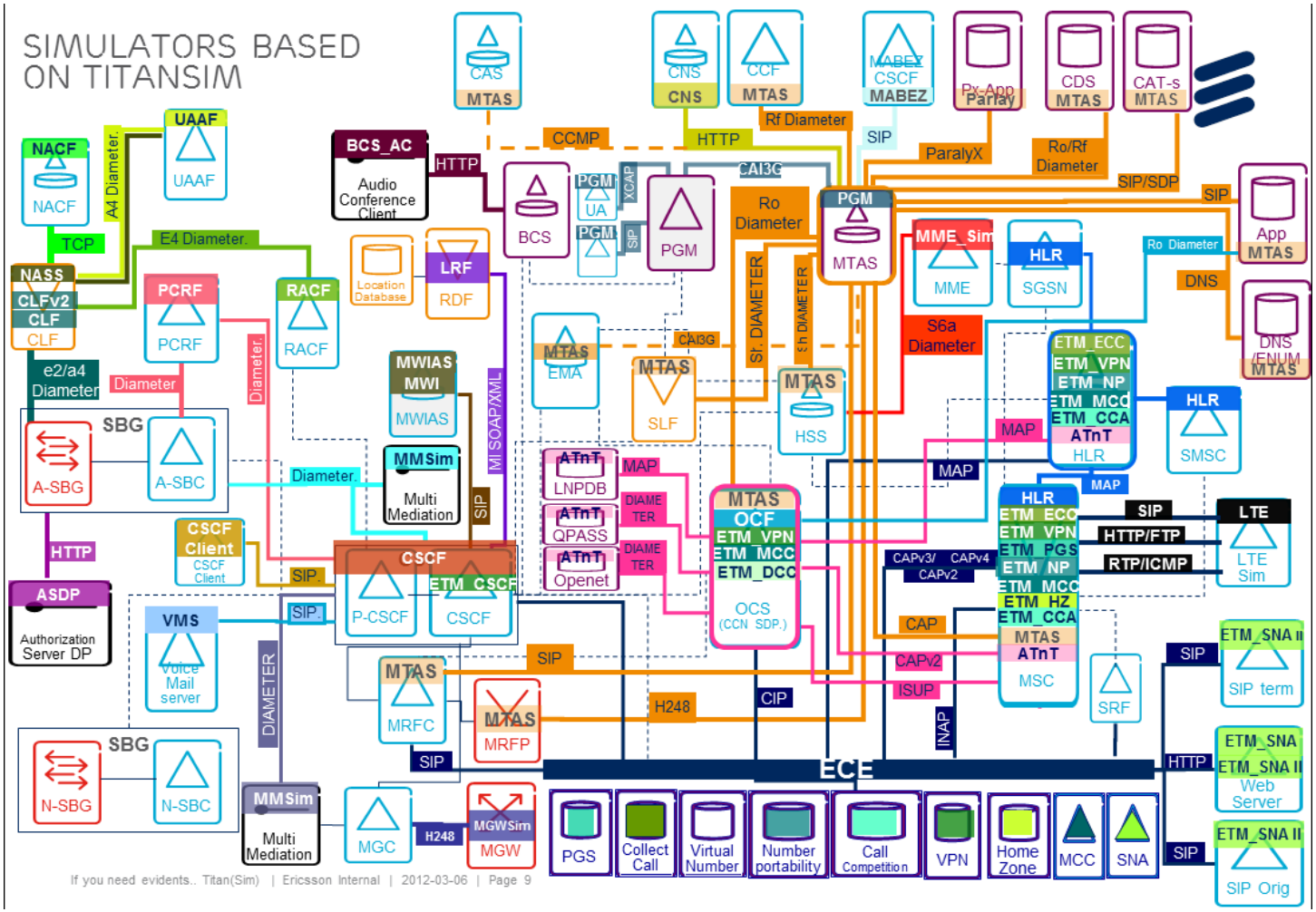
Telecommunications systems become more and more complex

- Complex architecture (system of systems)
- Complex behaviour (concurrency, complex protocols)
- Complex data (complex data structure, „big data”)

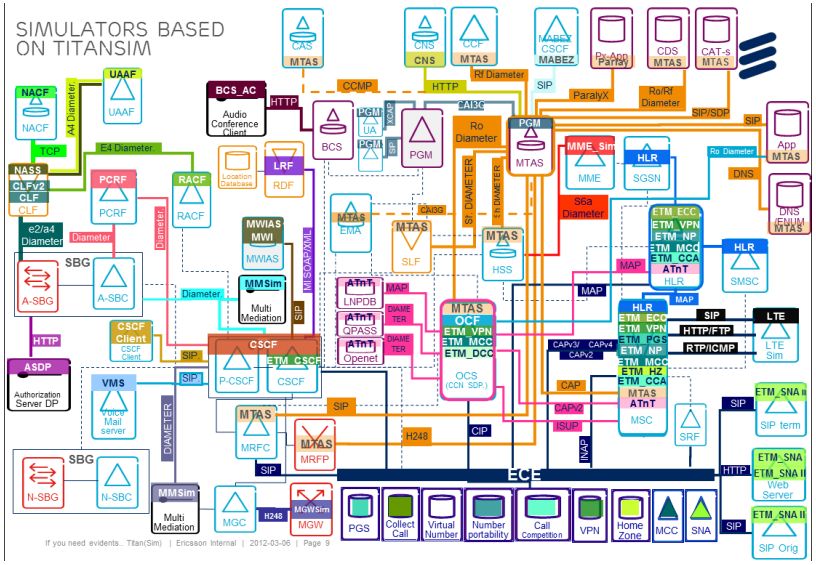
... and these systems shall be tested



TESTING


SIMULATORS BASED ON TITANSIM
CAS
MTAS
CNS
CNS
CCF
MTAS
MABEZ
CSCF
MABEZ
Px-App
Parlay
CDS
MTAS
CAT-s
MTAS
NACF
NACF
A4 Diameter.
UAAF
UAAF
BCS_AC
Audio Conference Client
HTTP
CCMP
HTTP
Rf Diameter
SIP
ParalyX
Ro/Rf Diameter
SIP/SDP
SIP
App
MTAS
TCP
E4 Diameter.
NASS
CLFv2
CLF
CLF
PCRF
PCRF
RACF
RACF
Location Database
LRF
RDF
BCS
PGM
UA
PGM
SIP
XCAP
PGM
CAI3G
Ro Diameter
Sr. DIAMETER
Sh DIAMETER
PGM
MTAS
MME_Sim
MME
HLR
SGSN
Ro Diameter
DNS
DNS /ENUM
MTAS
S6a Diameter
MTAS
EMA
CAI3G
MTAS
SLF
MTAS
HSS
ETM_ECC
ETM_VPN
ETM_NP
ETM_MCC
ETM_CCA
ATnT
HLR
HLR
SMSC
MAP
MAP
MAP
e2/a4 Diameter
Diameter
Diameter.
MWIAS
MWI
MWIAS
MI SOAP/XML
SBG
A-SBG
A-SBC
Diameter.
MMSim
Multi Mediation
SIP
ATnT
LNPDB
ATnT
QPASS
ATnT
Openet
DIAME TER
DIAME TER
MTAS
OCF
ETM_VPN
ETM_MCC
ETM_DCC
OCS
(CCN SDP.)
HLR
ETM_ECC
ETM_VPN
ETM_PGS
ETM_NP
ETM_MCC
ETM_HZ
ETM_CCA
MTAS
ATnT
MSC
SIP
HTTP/FTP
RTP/ICMP
LTE
LTE Sim
HTTP
CSCF Client
CSCF Client
SIP
CSCF
P-CSCF
CSCF
ETM_CSCF
CAPv3/ CAPv2
CAPv4
CAP
CAPv2
ISUP
INAP
ASDP
Authorization Server DP
VMS
Voice Mail server
SIP
DIAMETER
SIP
MTAS
MRFC
MTAS
MRFP
H248
CIP
SRF
SIP
ETM_SNA II
SIP term
SBG
N-SBG
N-SBC
SIP
ECE
HTTP
ETM_SNA
ETM_SNA II
Web Server
MMSim
Multi Mediation
MGC
H248
MGWSim
MGW
PGS
Collect Call
Virtual Number
Number portability
Call Competition
VPN
Home Zone
MCC
SNA
SIP
ETM_SNA II
SIP Orig
If you need evidents.. Titan(Sim) | Ericsson Internal | 2012-03-06 | Page 9

TESTING



What to test?

- Test purposes

How to test?

- Test configuration

Which data to test with?

- Test data specification

What the test shall contain?

- Test behaviour description

OVERVIEW OF ETSI TEST LANGUAGES

TPLan

- Describes Test Purposes
- „List of wishes”

```
TP id    : TP_COR_1097_02
Summary  : 'EUT processes a traversed packet with its size equals to its
           incoming link MTU'
RQ ref   : RQ_001_1097
Config   : CF_COR_21
TD ref   : TD_COR_1097_02

with { QE1 configured 'with a unique global unicast address '
   and QE2 configured 'with a unique global unicast address'
   and EUT configured 'with two unique global unicast addresses on the link
                       connecting QE1 and EUT, and, the link connecting QE2
                       and EUT, respectively'
   and QE1 'having larger link MTU than EUT'
   and EUT 'having larger or equivalent link MTU than QE2'
     }
ensure that {
  when { EUT receives a packet 'with its size equal to its
                                incoming link MTU'
           containing QE1 as the source_address
       and containing QE2 as the destination_address }
  then { EUT sends the packet to QE2 }
       }
```

OVERVIEW OF ETSI TEST LANGUAGES

TTCN-3

- Test program

```
function f_RECEIVER() runs on SIPPhone_CT
{
  var SipMessage vl_backup;
  var integer n := 0;
  alt {
    [] SIP.receive(SipMessage:?)
       -> value vl_backup
       { n := n + 1; SIP.send(vl_backup); repeat; }
    [] MSG.receive("STOP")
       { MSG.send(int2str(n)); setverdict(pass);}
    [] any port.receive
       { setverdict(fail); repeat; }
    }
  }
}
```

OVERVIEW OF ETSI TEST LANGUAGES

Missing…

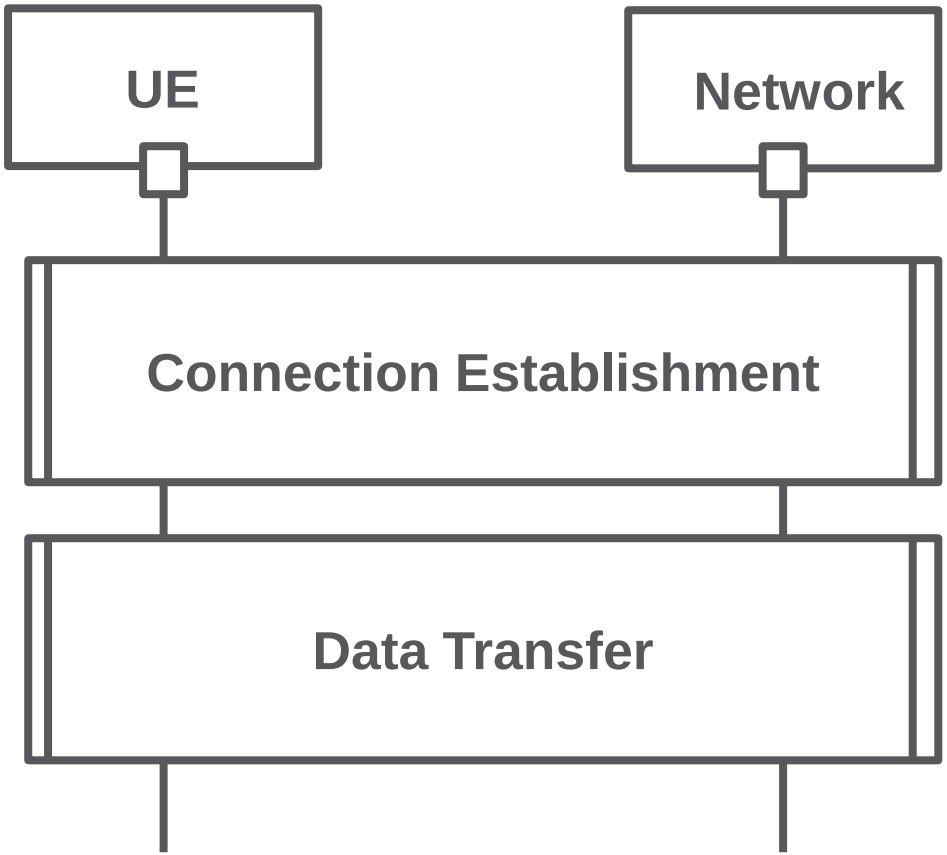
- Something between the two abstraction levels
- Which can be used also by non-programmers
- Graphical

TDL – Test Description Language

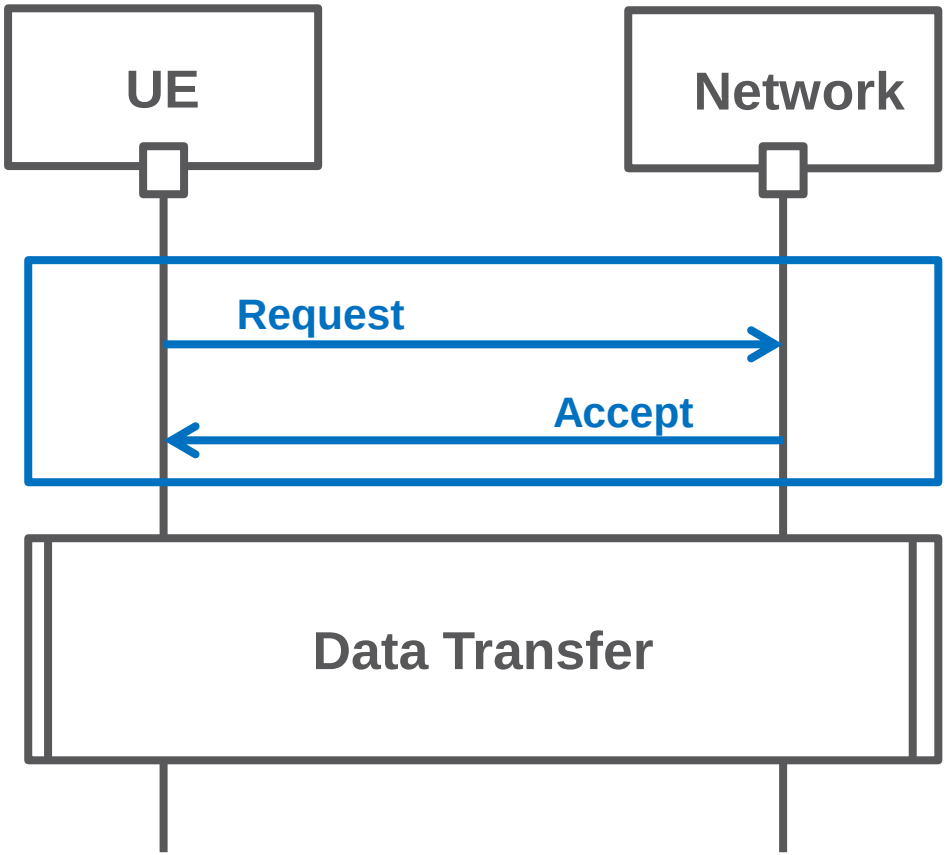
DESIGN CONSIDERATIONS OF TDL

- Test design
- Test documentation
- Test representation
- Shall be easy-to-learn, easy-to-use, intuitive
- One language throughout the whole development cycle

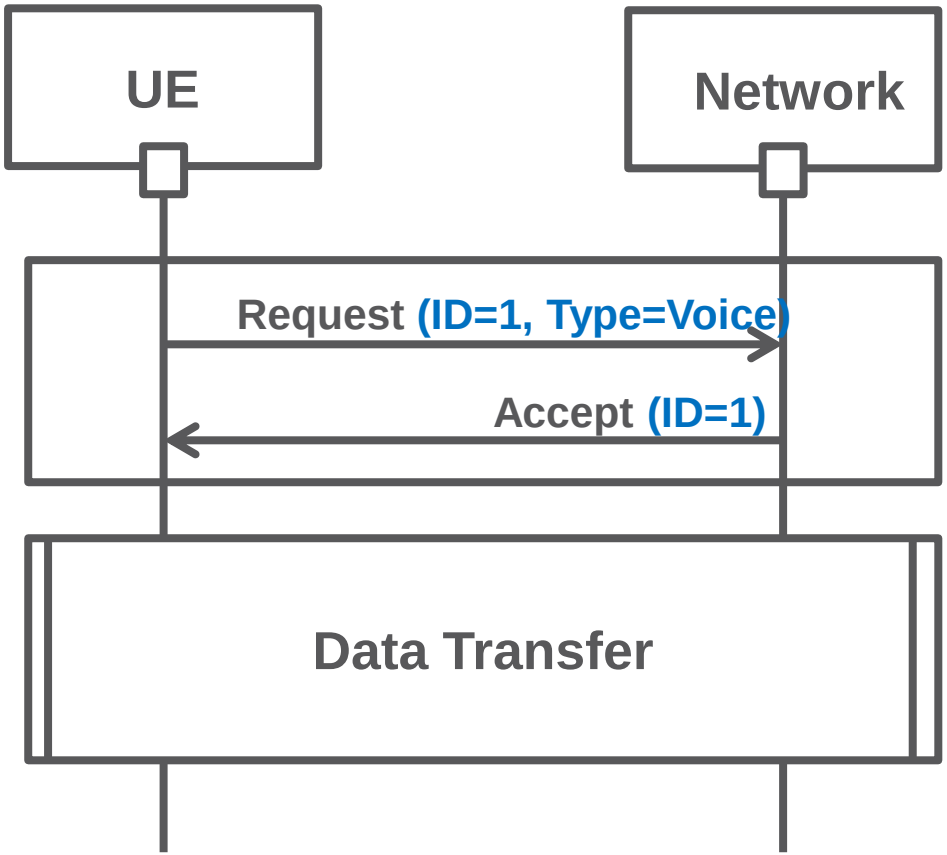
ITERATIVE DESIGN



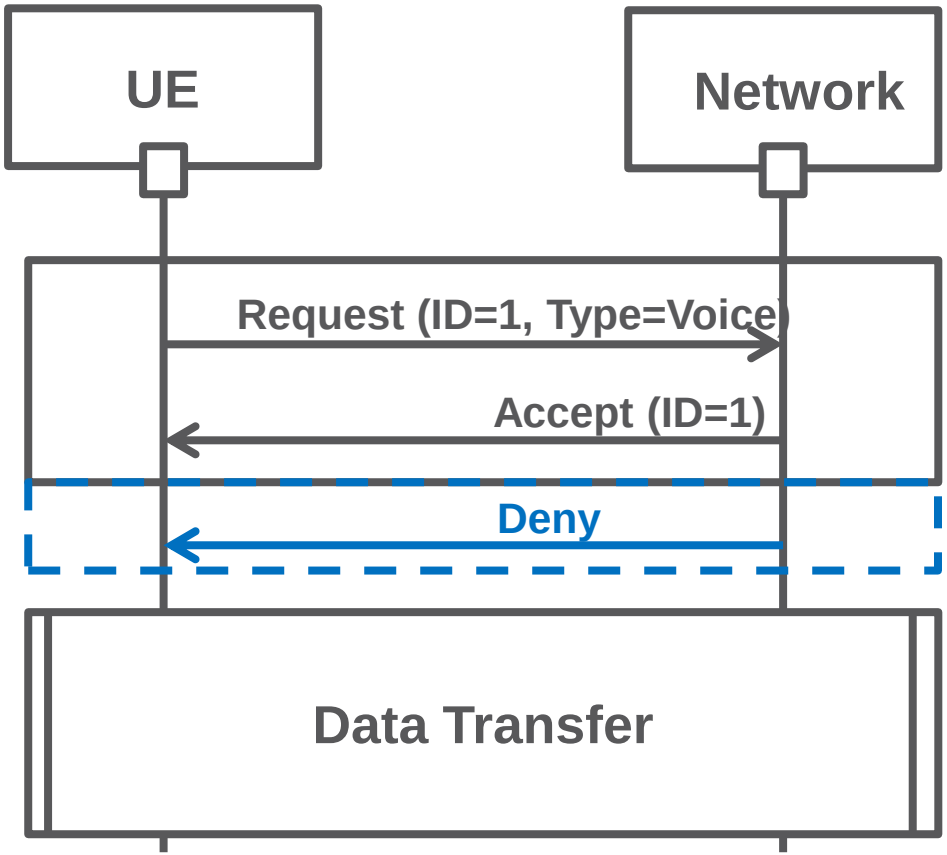
ITERATIVE DESIGN



ITERATIVE DESIGN

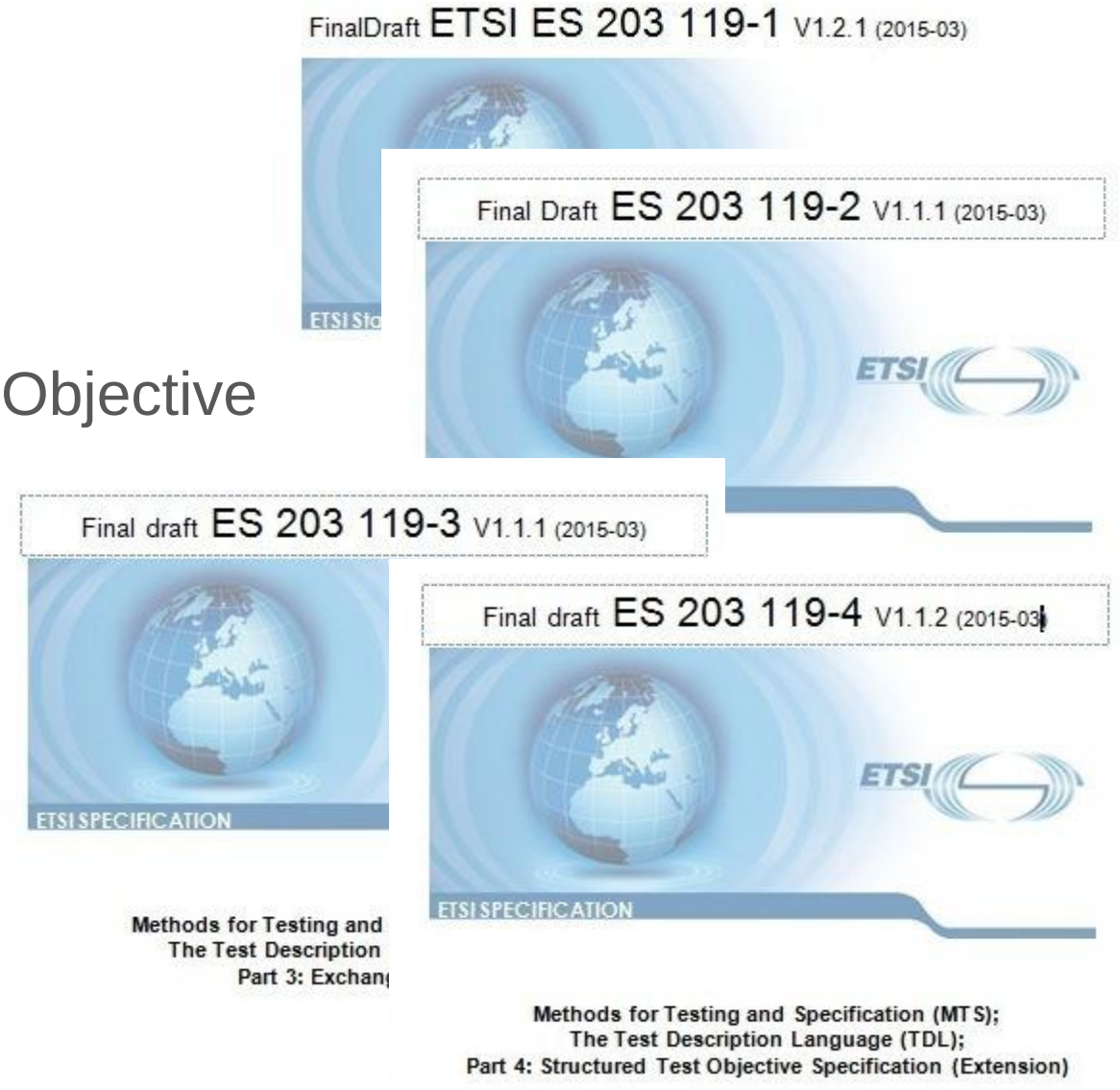


ITERATIVE DESIGN



TDL STANDARD

- Meta-model
- Graphical Syntax
- Transfer Syntax
- Structured Test Objective Specification

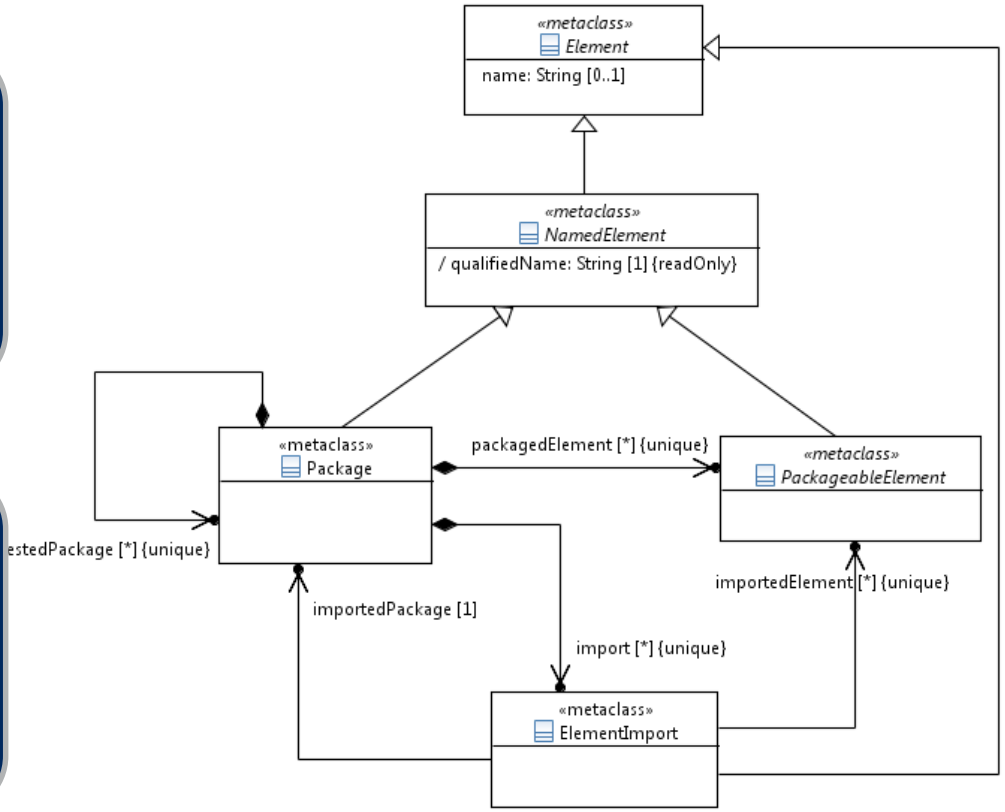


META-MODEL

Well-defined language constructs

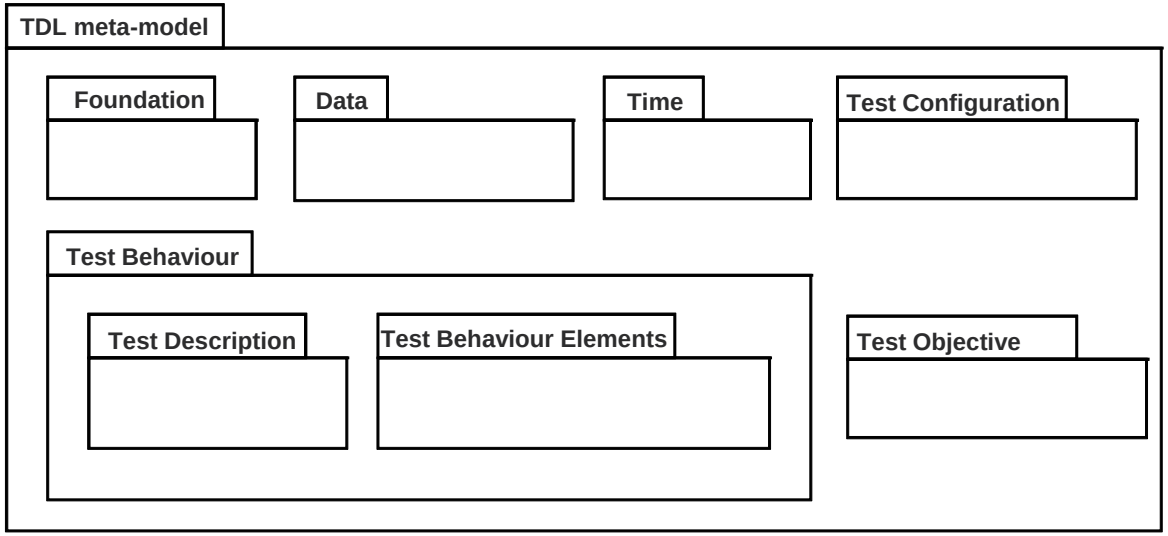
- UML MOF-based description

Makes it possible to develop different domain-specific concrete syntaxes



STRUCTURE OF TDL

- Test Data
- Time, Timers
- Test Configuration
- Test Behaviour
- Test Objectives



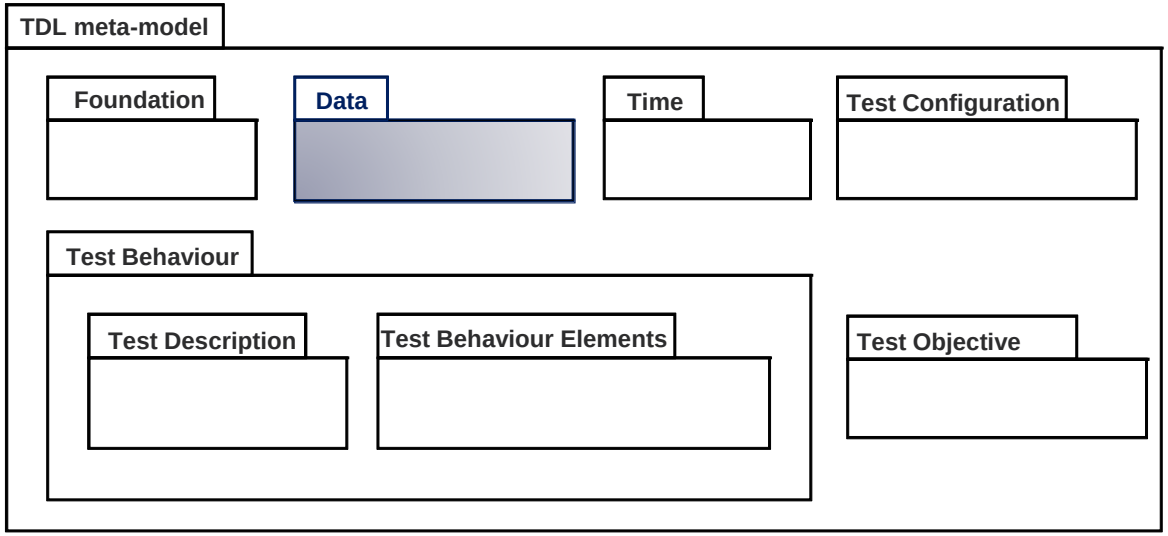
TEST DATA

Abstract data

Simple and Structured Data (declarative)

- Mandatory and optional fields

Can be mapped to ‘real’ data implementation



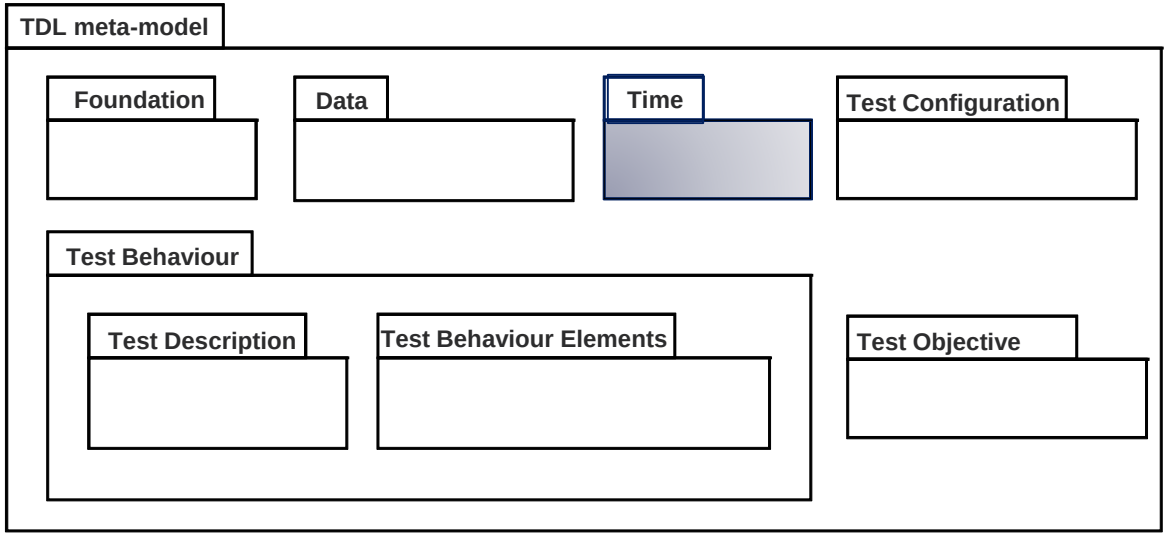
TIME, TIMERS

Time

- Time Labels
- Time Constraints
- Wait

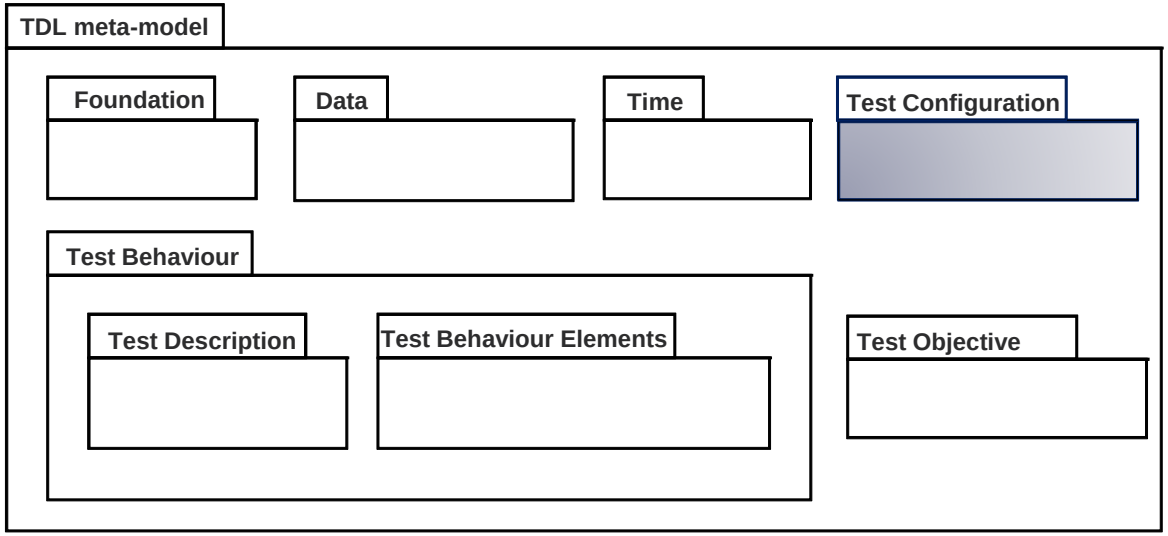
Timers

- Definition
- Operations
  - **start**, **stop**, **timeout**



TEST CONFIGURATION

- Components and Gates (Interfaces)
- TESTER and SUT roles
- Connections between Gates



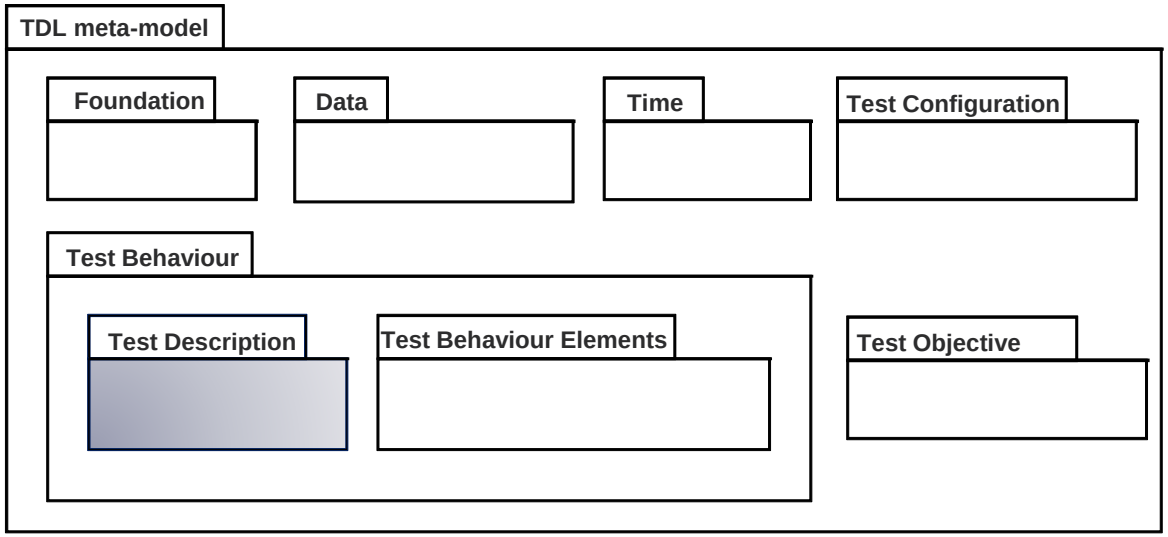
TEST BEHAVIOUR

Test Description

- ~ Testcase

Contains:

- Test Objective
- Test Configuration
- Test Behaviour



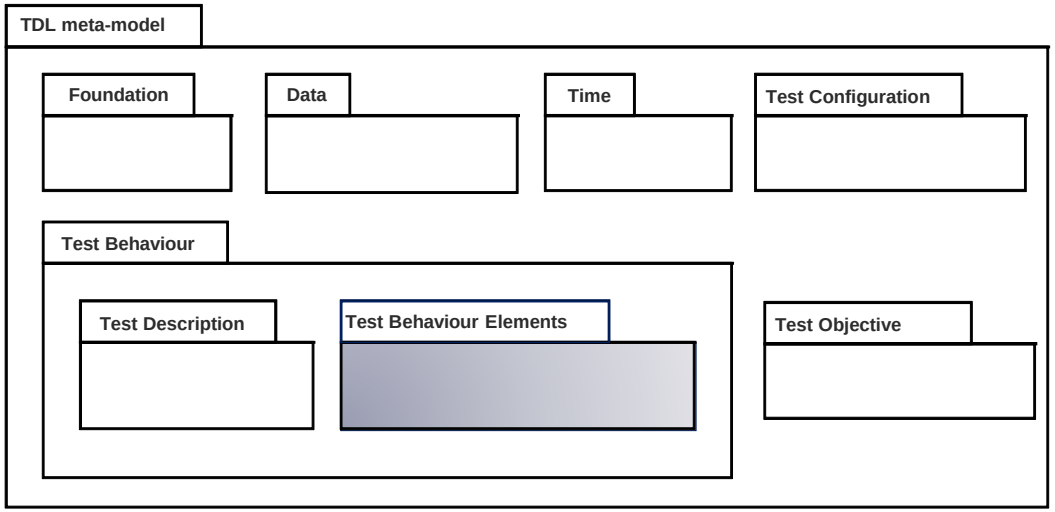
TDL describes the expected behaviour

- Any deviation – automatic **fail** verdict
- Can be overwritten by explicit verdict setting
- Pre-defined verdicts: **pass, fail, inconclusive**
  - Extensible

TEST BEHAVIOUR

Atomic Behaviours

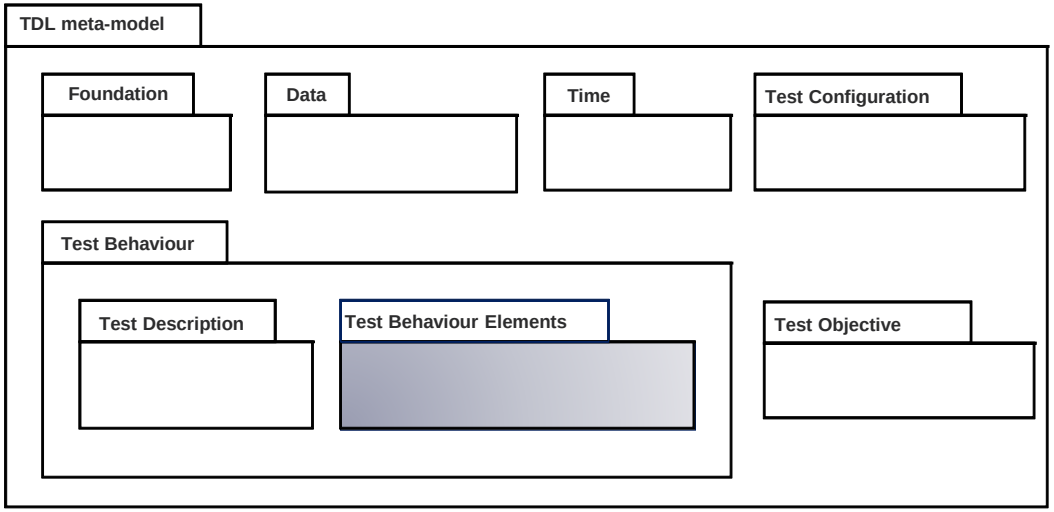
- Interaction
  - generalized communication
- Action, Function Call, Assignment
- Call other Test Descriptions
- Explicit verdict setting
- Assertion
- Stop



TEST BEHAVIOUR

Combined Behaviours

- Sequential (Compound)
- Parallel
- Alternative
- Conditional (~if..then..else)
- Cycles
  - For
  - While
- Periodical
- Handling of non-expected behaviour
  - Default

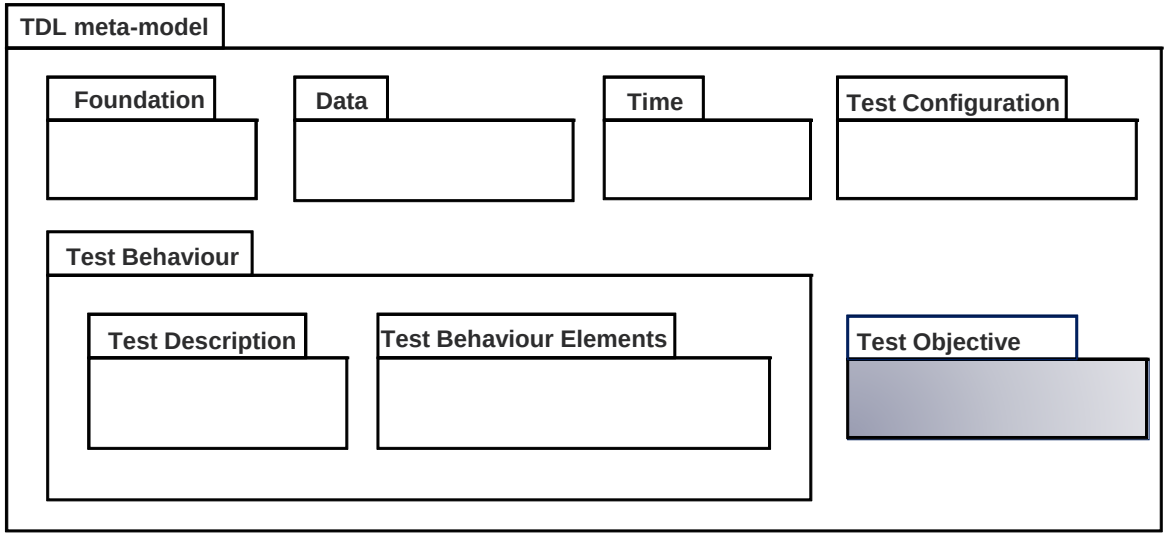


TEST OBJECTIVES

Definition of Test Objectives

- Textual description
- Reference to test documents

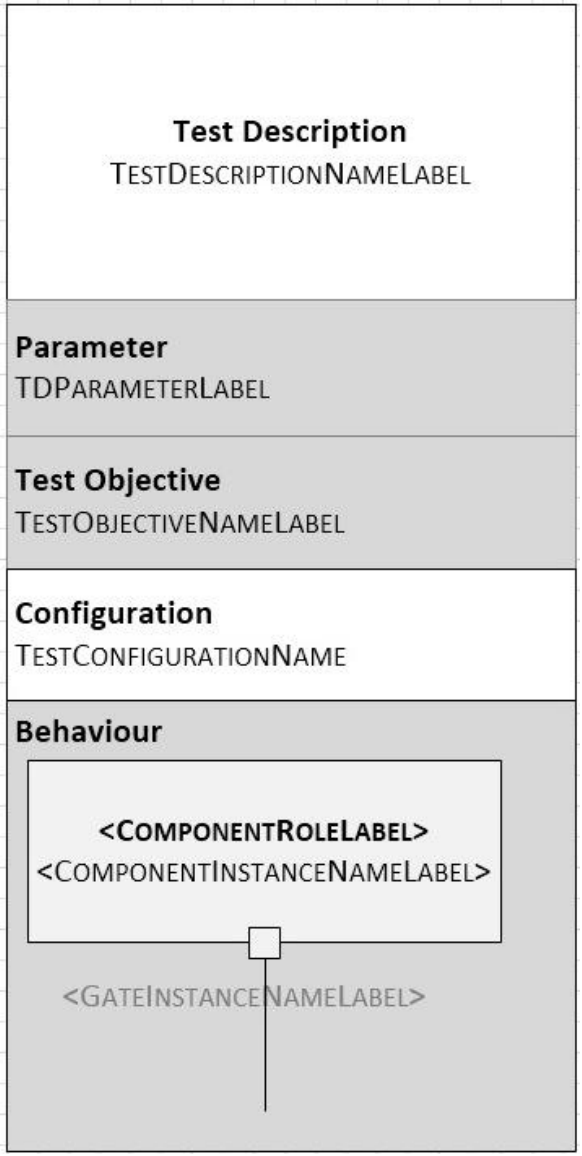
At behaviour description the satisfied test objective can be indicated



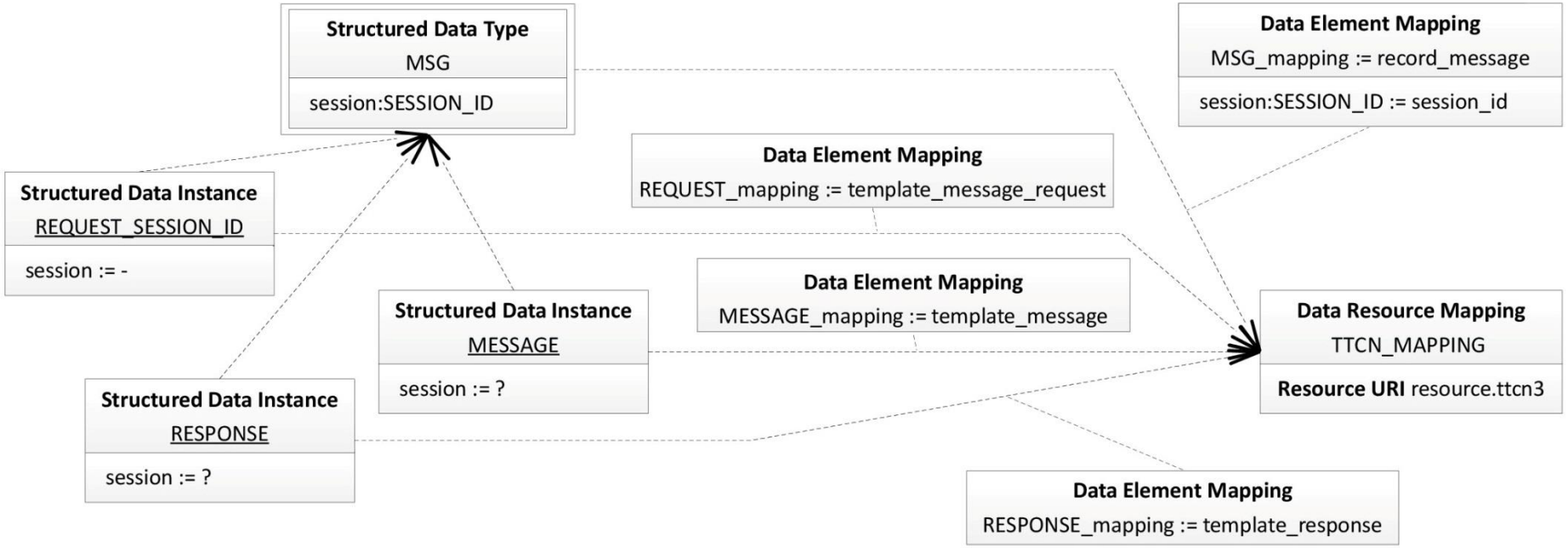
GRAPHICAL SYNTAX

- Similar approach to UML SD
- New symbols to new constructs
- Graphical symbols
  - Formal description of the contained text

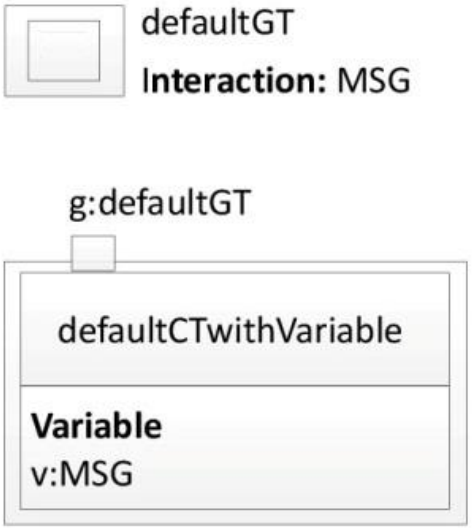
```
context TestDescription
TESTDESCRIPTIONNAMELABEL ::= self as context in <NAMEDELEMENTLABEL>

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 <NAMEDELEMENTLABEL>
                           end
TESTCONFIGURATIONNAME ::= self.testConfiguration as context in <NAMEDELEMENTLABEL>
```

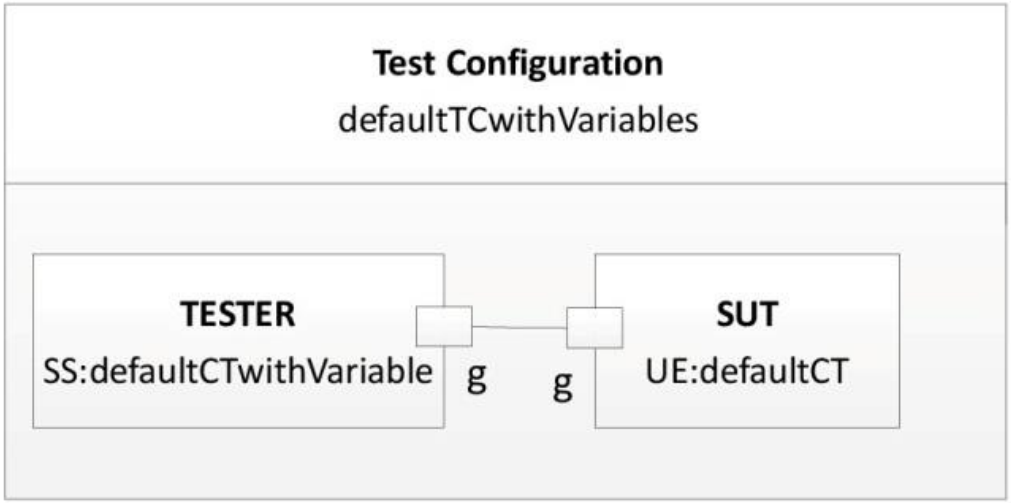


DATA SPECIFICATION



TEST CONFIGURATION



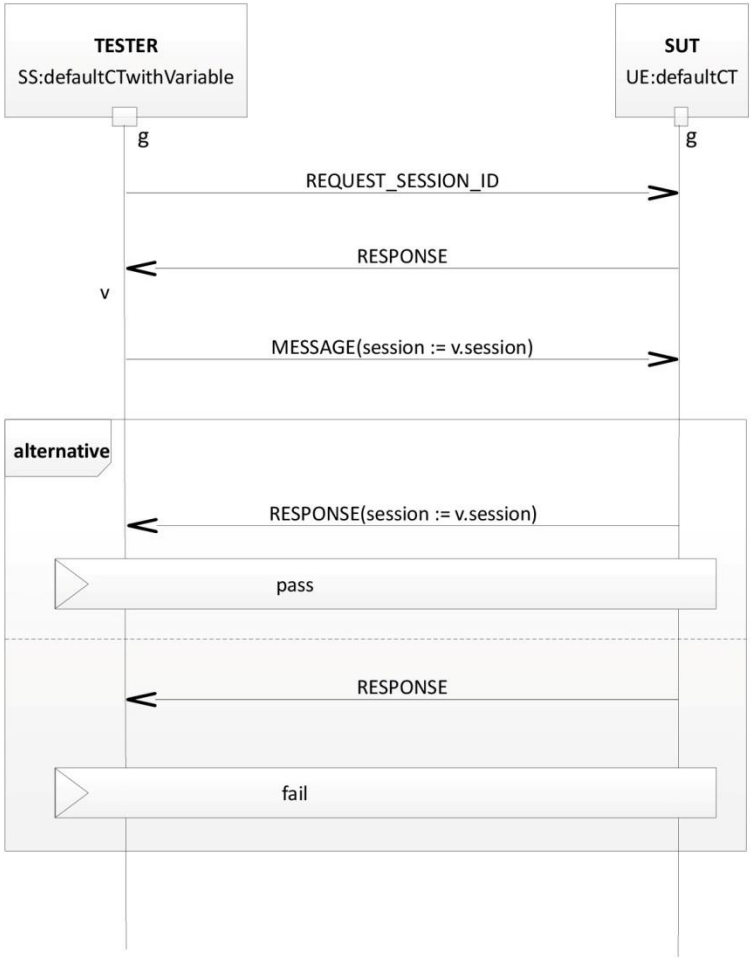


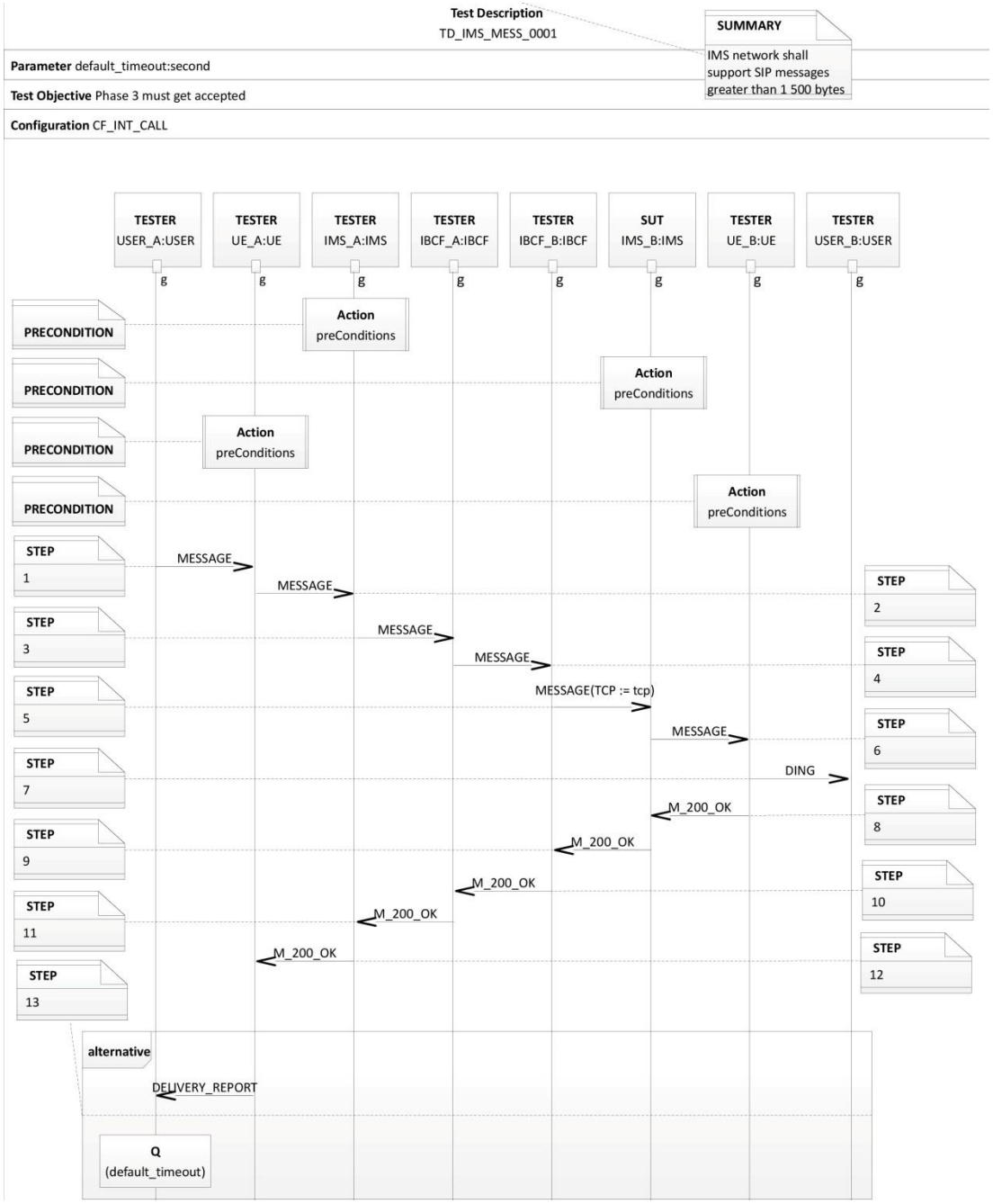
TEST BEHAVIOUR

TestDescription
exampleTD

Configuration defaultTCwithVariables

Test Objective CHECK_SESSION_ID_IS_MAINTAINED




Test Description
TD_IMS_MESS_0001
SUMMARY
IMS network shall support SIP messages greater than 1 500 bytes
Parameter default_timeout:second
Test Objective Phase 3 must get accepted
Configuration CF_INT_CALL
TESTER USER_A:USER
TESTER UE_A:UE
TESTER IMS_A:IMS
TESTER IBCF_A:IBCF
TESTER IBCF_B:IBCF
SUT IMS_B:IMS
TESTER UE_B:UE
TESTER USER_B:USER
PRECONDITION
Action preConditions
MESSAGE
STEP 1
STEP 2
STEP 3
STEP 4
STEP 5
MESSAGE(TCP := tcp)
STEP 6
STEP 7
DING
STEP 8
M_200_OK
STEP 9
STEP 10
STEP 11
STEP 12
STEP 13
alternative
DELIVERY_REPORT
Q
(default_timeout)

TRANSFER SYNTAX

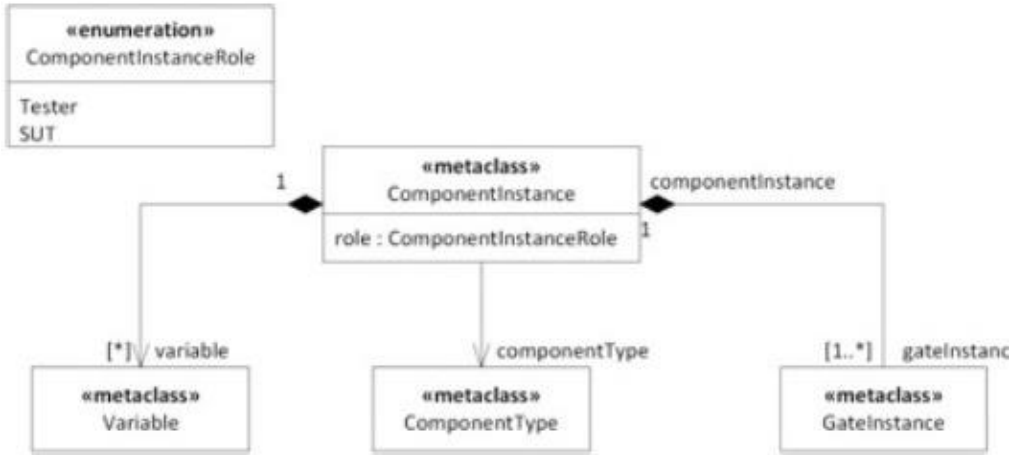
Goals:

- Interoperability between tools
- Interoperability between different concrete syntaxes

XMI (XML Metadata Interchange)

- To serialize the meta-model
- Syntactical check possible
- For semantic check constraints described in meta-model shall also be considered

TRANSFER SYNTAX




```
<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>
```

STRUCTURED TEST OBJECTIVE SPECIFICATION

Goals:

- Formalize the description of test purposes
- Unify the content
- Possibility of automatic validation
- Preserving the compatibility with TPLan

STRUCTURED TEST OBJECTIVE SPECIFICATION

TP Id	TP/GEONW/FDV/BAH/BV/01
Test Objective	Check defined values of default Gn parameters in the basic header
Reference	
PICS Selection	PICS_F1

Initial Conditions

```
with {
  the IUT entity being in the initial state
}
```

Expected Behaviour

```
ensure that{
  when {
    the IUT entity is requested to send a "GUC packet"
  }
  then {
    the IUT entity sends a "GUC packet" containing
      BasicHeader containing
        "version field" indicating value "itsGnProtocolVersion MIB parameter" ,
        "RHL field" indicating value "itsGnDefaultHopLimit MIB parameter"
      ;
    ;
  }
}
```

Final Conditions

TDL: PRESENT AND FUTURE

ETSI Standard

TDL v1 – 2013

TDL v2 – Multi-part Standard

TDL v3

- Reference editor
- UML profile

TDL v4

- TDL -> TTCN-3 Mapping

SUMMARY

One language throughout the whole development cycle

Easy-to-use, Easy-to-learn

Graphical

Supports different abstraction levels

- From very high level to close to implementation

At different standardisation/industry areas

- Telecommunication, Internet, Vehicle, Medical, etc.
- Simple and structured systems