



# STF 476: TD-LTE Phase 2

## Status Report

# Document History

---

- 2014-03-19: Document submitted for SG-2

# Goal and Objectives of TDL Phase 2

---

- Goal
  - Supporting ETSI and industrial users in using TDL
- Objectives
  - Extended TDL meta-model for supporting test automation
  - Default concrete syntaxes
    - Graphical syntax for end-users
    - Textual exchange syntax for tool interoperability
  - Analysis on the needs for a textual syntax to support ETSI use cases

# Deliverables for TDL2

---

<b>Del.</b>	<b>Work Item Code / Standard Number</b>	<b>Working Title / Scope</b>
D1	RES/ES 203 119-1 V1.2.1	Test Description Language; Meta-Model and Semantics Scope: common concepts, meta-model, semantics
D2	DES/ ES 203 119-2 V1.1.1	Test Description Language; Graphical Syntax Scope: TDL graphical concrete syntax for end users
D3	DES/ ES 203 119-3 V1.1.1	Test Description Language; Exchange Format Scope: TDL exchange format for tool interoperability

Optionally: ES Part 4 on TDL textual concrete syntax (no WI created yet)

# TDL2 Organisation of Work

---

- Start: 02/2014
  - Task 0: Project management
  - Task 1: Extension of TDL meta-model (02-12/2014)
  - Task 2: Graphical concrete syntax (02-12/2014)
  - Task 3: Exchange syntax (06-12/2014)
  - Task 4a: Analysis on ETSI concrete syntax (02-05/2014)
- Potential STF extension: 06/2014 (decision at MTS#62)
  - Task 4b: ETSI concrete syntax (06-12/2014)
- End: 12/2014
  - WI: updated meta-model description + semantics
  - WI: concrete syntax + meta-model mapping
  - WI: exchange syntax + meta-model mapping

# TDL2 Milestones

---

- M0: 02/2014
  - Start of work of Tasks 0, 1, 2, 4
- M1: 05/2014 <- Discussion at MTS#62 (05/2014)
  - (T1) Early draft: updated meta-model, (T2) Early draft: graphical syntax,
  - (T4) Decision paper on textual syntax
  - (T3) Start of Task 3,
- M2: 09/2014 <- Discussion at MTS#63 (10/2014)
  - (T1) Stable draft: updated meta-model, (T2) Stable draft: graphical syntax, (T3) Early draft: exchange syntax
- M3: 12/2014 <- Approval at MTS#64 (02/2015)
  - (T1) Final draft: updated meta-model, (T2) Final draft: graphical syntax, (T3) Final draft: exchange syntax

# Status Report



## Task 0: Session Overview

---

- WK09 Feb 2014
- WK15 Apr 2014
- WK23 Jun 2014
- WK36 Sep 2014
- WK42 Oct 2014
- WK49 Dec 2014

## Task 1: Extended TDL Meta-Model

---

- Targeted for  
• specification  
• types and c  
• Open for M2  
• types and c  
• time  
• behaviour  
• editorial cla

## Task 2: Graphical Concrete Syntax

---

- Different sp
- OMG Diagr
- Means for u
- Decision or

## Task 4: TDL Textual Syntax Analysis

---

- Understanding ETSI's requirements for a textual syntax
  - review and discussion of notes and input collected during STF 454
  - discussion of CTI proposal based on ITS and GeoNetworking examples
  - initial focus on provided examples, (TPLan-like) format pushed within ETSI



# Task 0: Session Planning

---

- 6 sessions in total
- 2 sessions per milestone
  - 1 preparatory / debriefing
  - 1 finalising
- Homework and remote coordinated work in between

# Task 0: Session Overview

---

- WK09 Feb 24-28 - Session 1 @ETSI
- WK15 Apr 07-11 - Session 2 @ETSI
- WK23 Jun 02-06 - Session 3 @FOKUS
- WK36 Sep 01-05 - Session 4 @ETSI
- WK42 Oct 13-17 - Session 5 @Siemens
- WK49 Dec 01-05 - Session 6 @ETSI

# Task 0: Milestone 1 Timeline

---

- WK09 Feb 24-28 - Session 1 @ETSI
  - 4 experts, 16 days, define roadmaps, prepare early drafts
- WK15 Apr 07-11 - Session 2 @ETSI
  - 4 experts, 16 days, finalise early drafts, analysis report, ToR if applicable
  - deadline for requests to extend/move sessions, contracts extension
- WK18 Apr 28-May 05 - Deliverables ready
- WK20 May 14-15 - MTS #62 @Siemens

# Task 0: Milestone 2 Timeline

---

- WK23 Jun 02-06 - Session 3 @FOKUS
  - 5 experts, 20-25 days, define roadmaps, prepare stable drafts
- WK36 Sep 01-05 - Session 4 @ETSI
  - 5 experts, 20-25 days, finalise stable drafts
- WK37 Sep 08-12 - Deliverables ready
- WK39 Sep 22-26 - MTS #63 TBD

# Task 0: Milestone 3 Timeline

---

- WK42 Oct 13-17 - Session 5 @Siemens
  - 5 experts, 20-25 days, define roadmaps, prepare final drafts
- WK49 Dec 01-05 - Session 6 @ETSI
  - 5 experts, 20-25 days, finalise final drafts
- WK50 Dec 15-19 - Deliverables ready
- WKXX Jan/Feb 2015 - MTS #64 TBD

# Task 0: Milestone Resources

---

- ~15 days/expert per milestone
  - assuming roughly equal resource allocation per expert
  - 2x4 days sessions, ~7 days homework
  - other options to consider?
- Milestone 1: ~60 days (4 experts)
- Milestone 2: ~75 days (5 experts)
- Milestone 3: ~75 days (5 experts)

# Task 0: Operational Risks

---

- Task (inter-)dependencies hinder progress due to distributed work
  - Severity: Medium, Likelihood: Low
  - Mitigation strategies
    - make dependencies explicit where these are inevitable in order to raise awareness
    - ensure communication and collaboration among experts working on inter-dependent tasks
    - reassign experts where applicable



# Task 0: Operational Risks

---

- Misunderstandings and communication barriers hinder progress
  - Severity: Medium, Likelihood: Medium
  - Mitigation strategies
    - recognise and differentiate between misunderstandings and technical disagreements (moderation and awareness)
    - emphasis on examples and written word
    - identify fundamental differences between alternative proposals and their impact (pragmatist approach)

# Task 0: User Acceptance Risks

---

- Lack of essential tool support considered a limiting factor
  - Severity: High, Likelihood: Medium
  - Mitigation strategies
    - means to use and access the language need to be provided early on
    - early drafts need to be discussed and aligned with users' needs
    - simplified initial interface to TDL may be favourable (Task 4)
    - integration in existing processes and awareness among potential users (board report, collaboration with CTI)

# Task 0: Tool Vendor Adoption Risks

---

- Lack of user base and technical challenges raise barrier to entry
  - Severity: High, Likelihood: Medium
  - Mitigation strategies
    - awareness and collaboration with users seeks to create initial demand
    - early prototypical validation seeks to create a sound technical foundation and reduce inherent technical challenges
    - participating commercial and in-house tool vendors assure that their perspectives are considered in the design and execution of the standards

# Session 1 Overview

---

- Goal: Prepare and define roadmaps for Milestone 1
  - created initial pool of tasks
  - selected targets for Milestone 1
  - performed first analysis tasks
  - proposed conceptual solutions for analysed targets
- Targeted for Session 2
  - implementation and validation of analysed targets, progress on remaining targets

# Task 1: Extended TDL Meta-Model

---

- Targeted for M1
  - specification of sub-test configurations
  - types and data refinement (initial proposal)
- Open for M2 and M3
  - types and data (finalised)
  - time
  - behaviour
  - editorial clarifications and refinements

# Task 1: Sub-configurations

---

- Understanding of sub-configurations
- 2 core aspects
  - configuration composition
  - relationship between test configurations and test descriptions
    - ongoing discussion on two different approaches to binding
    - binding upon declaration (“global constant”), allows for 1 : n
    - binding upon reference (“formal parameters”), allows for m : n

# Task 1: Sub-configurations Concept

---

- “Factoring out” reusable behaviours and configurations
- Support multiple instances of the same sub-configuration
- Support reassignment of roles
  - scope and ensuring valid behaviour description (alternative, exceptional) under discussion
- Support for merging multiple components into one

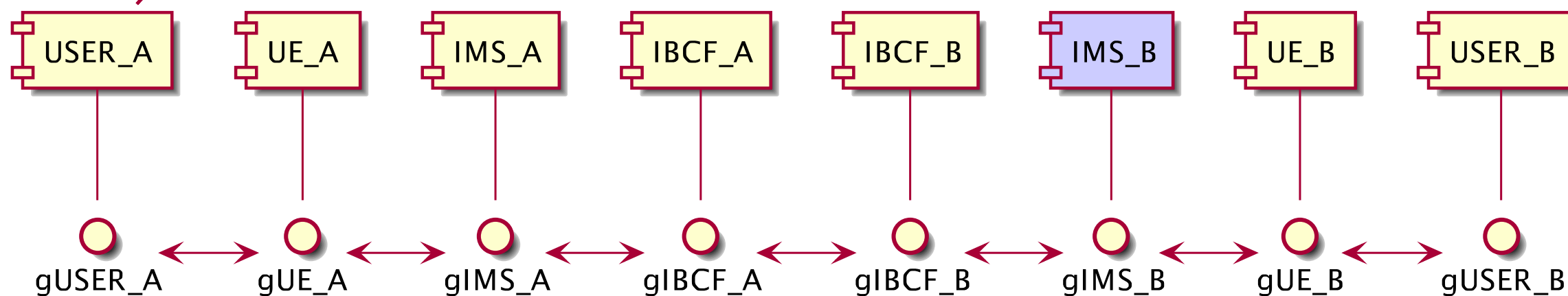


# Task 1: Sub-configurations Example

- Base configuration

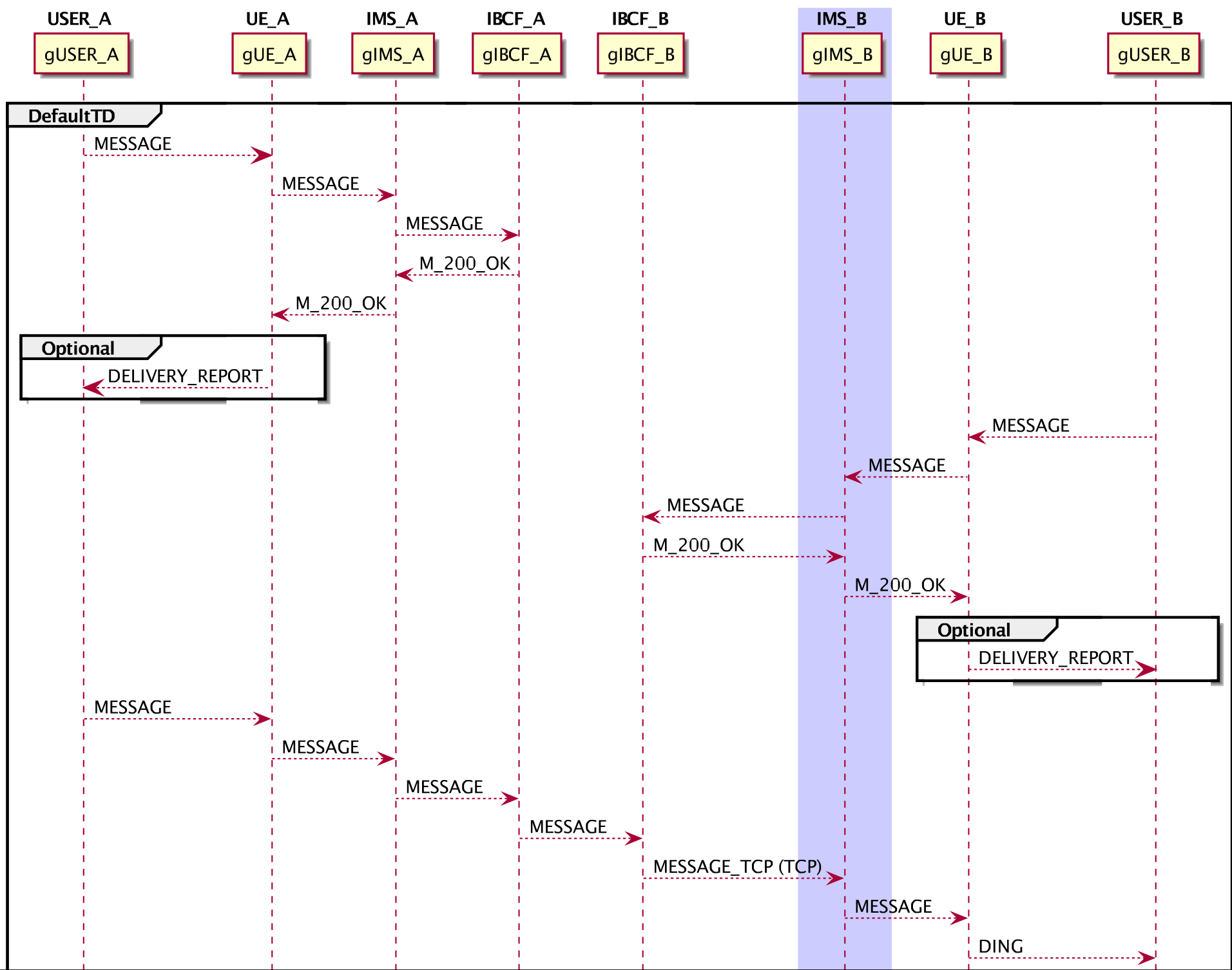
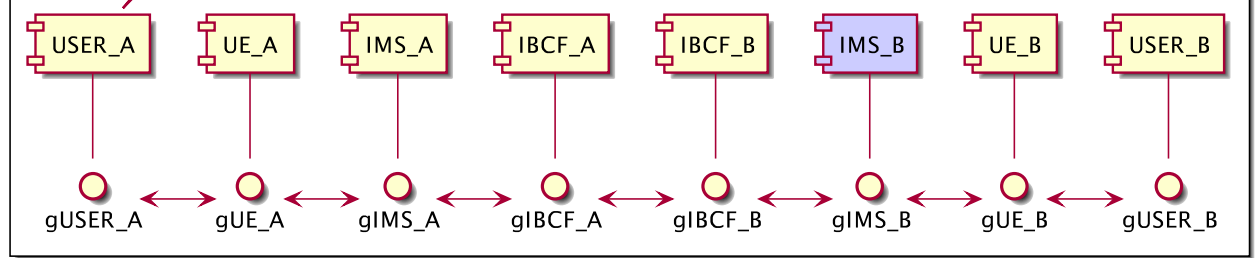
## TestConfiguration IMSbase

```
"instantiate USER_CT as USER_A with  
gate gUSER_A of type defaultGT  
}"
```



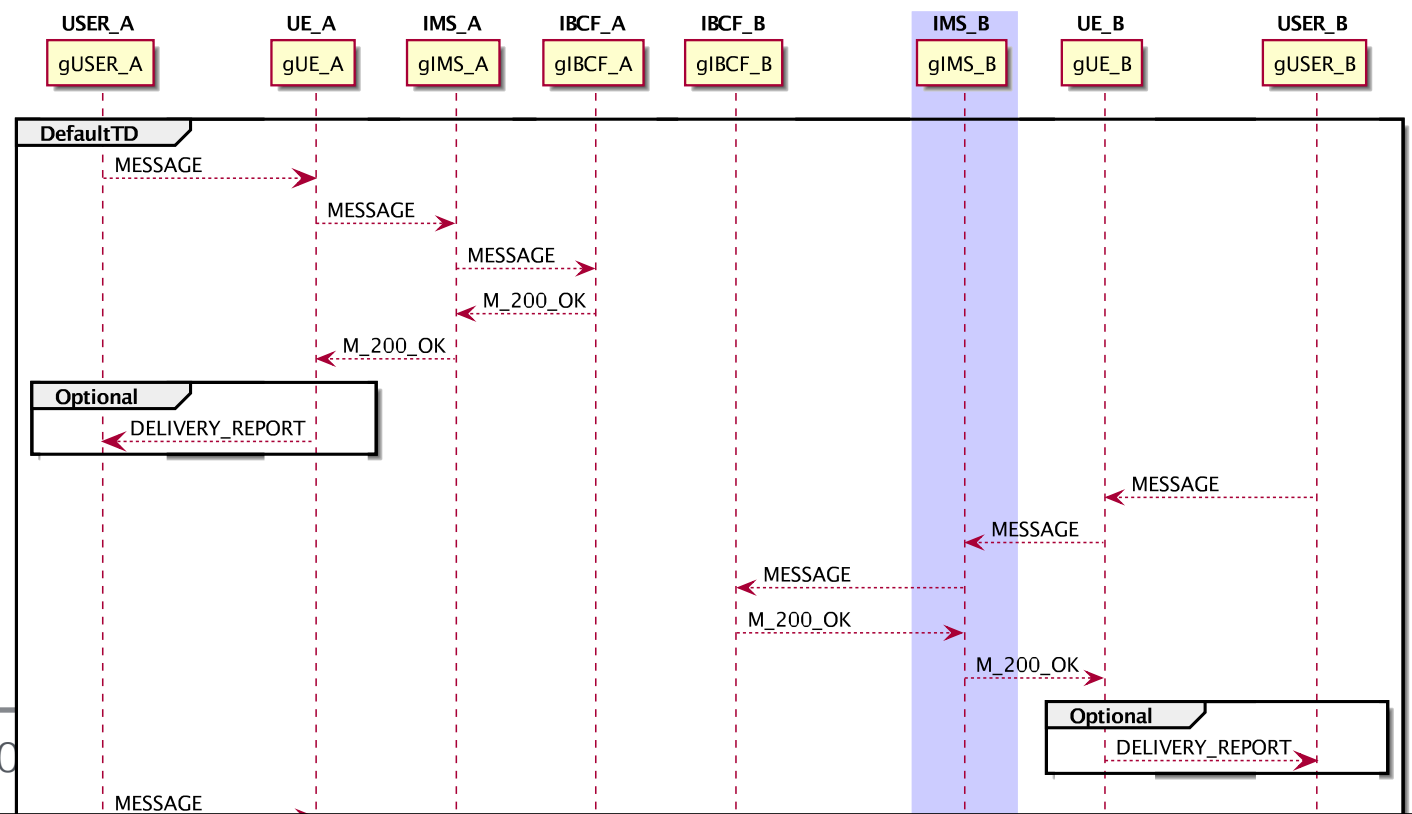
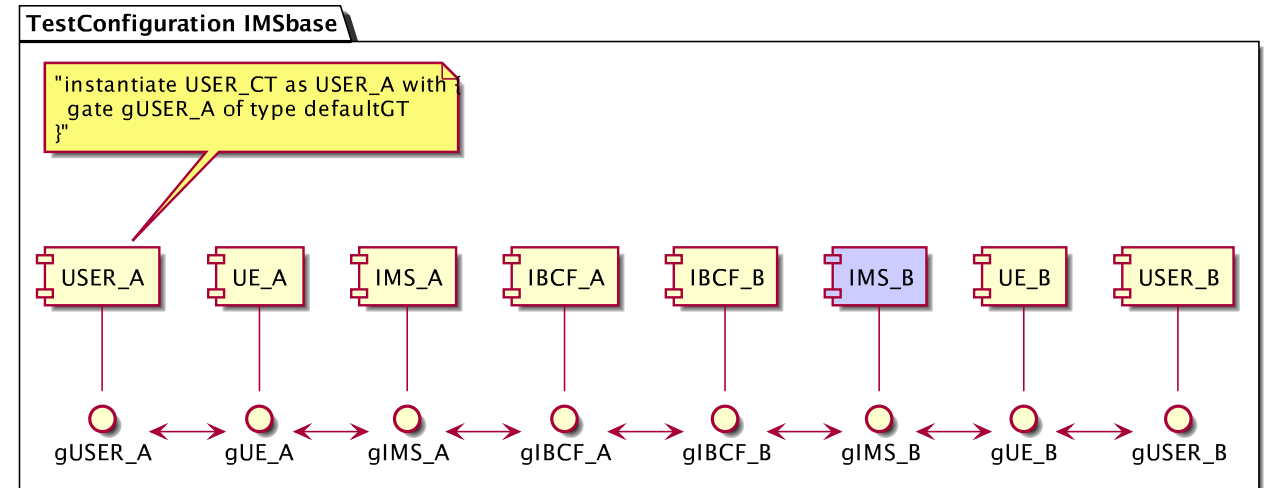
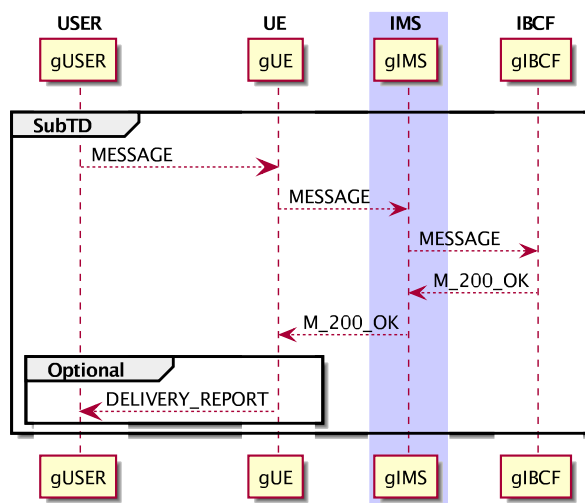
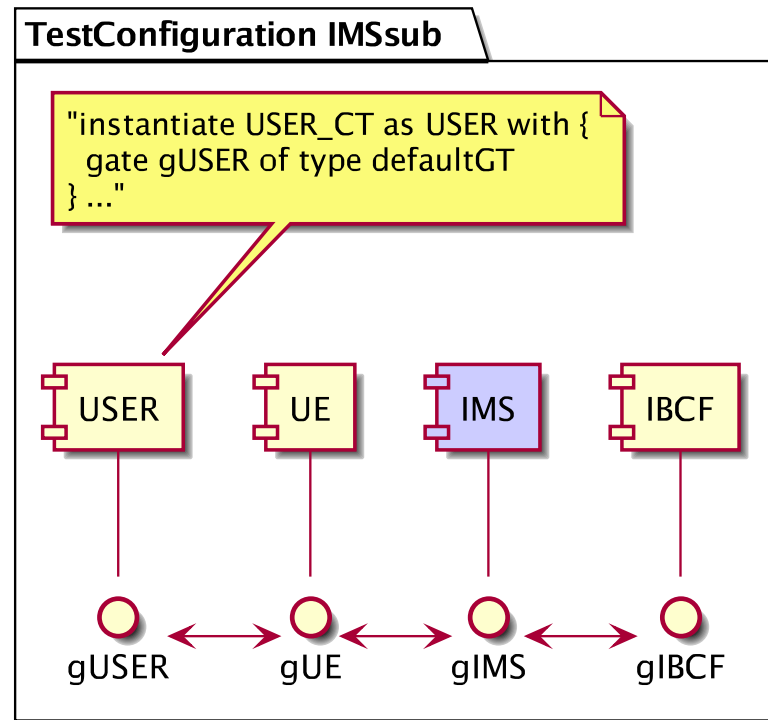
**TestConfiguration IMSbase**

"instantiate USER\_CT as USER\_A with gate gUSER\_A of type defaultGT"



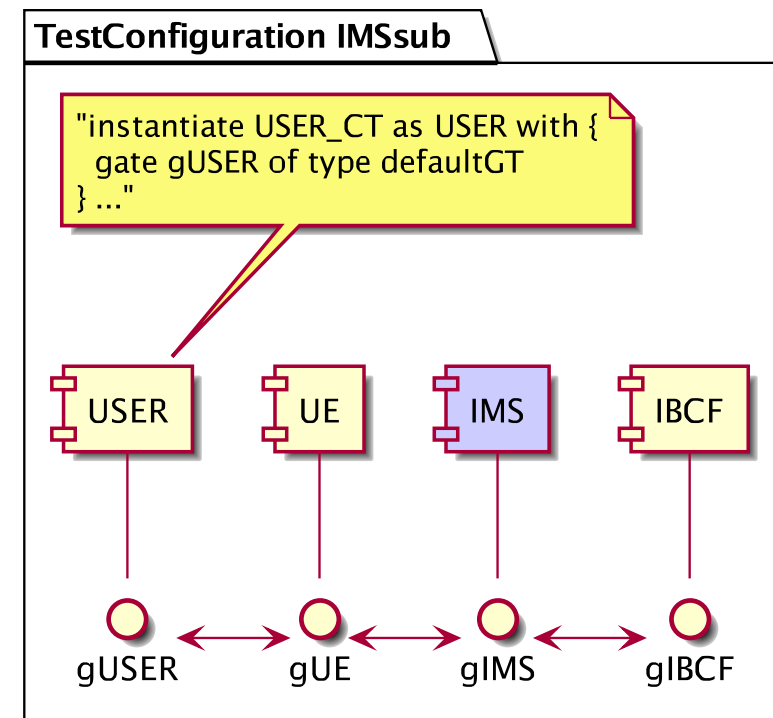
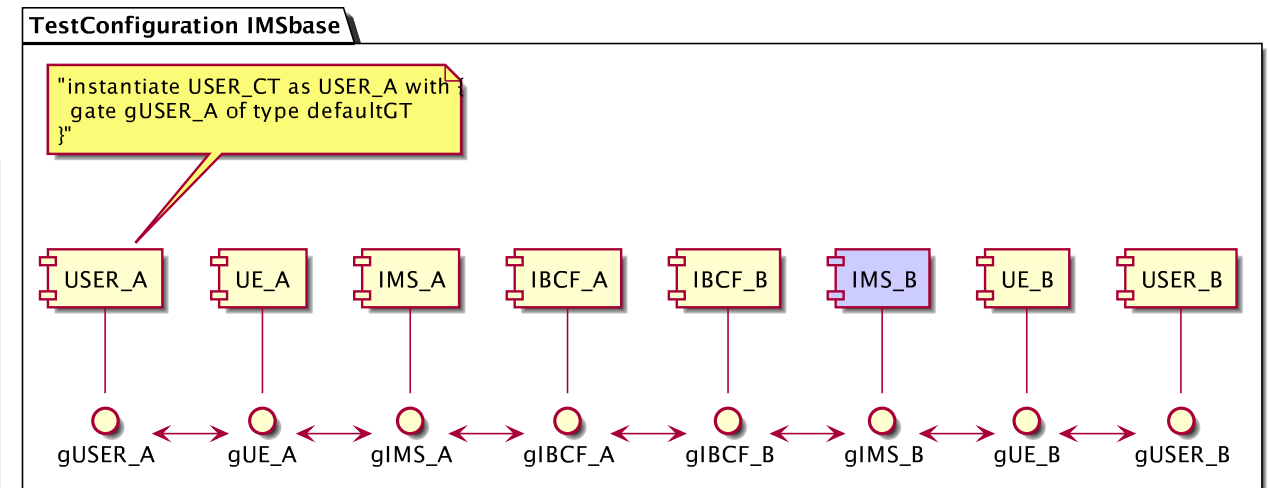
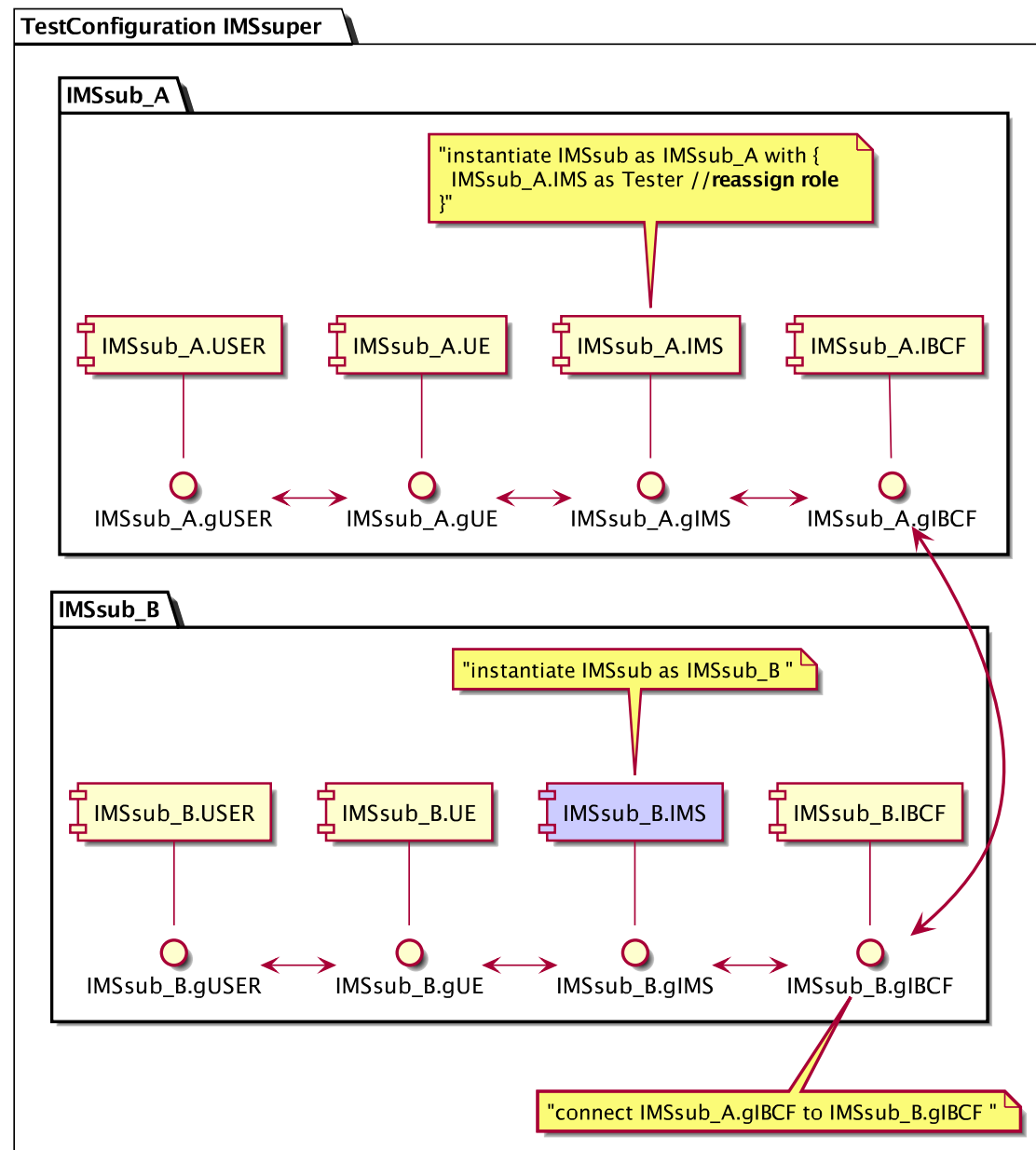
# Task 1: Sub-configurations Example

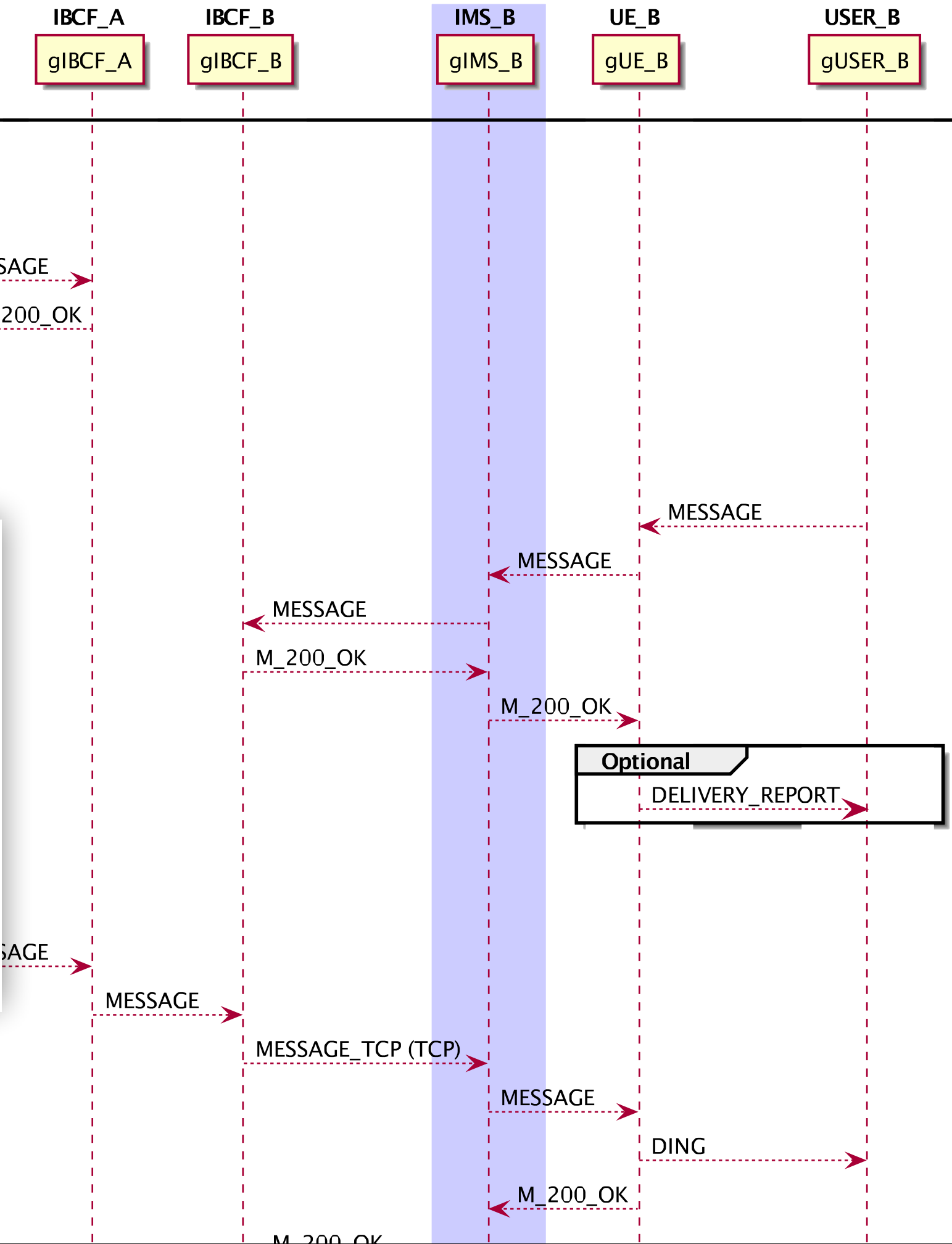
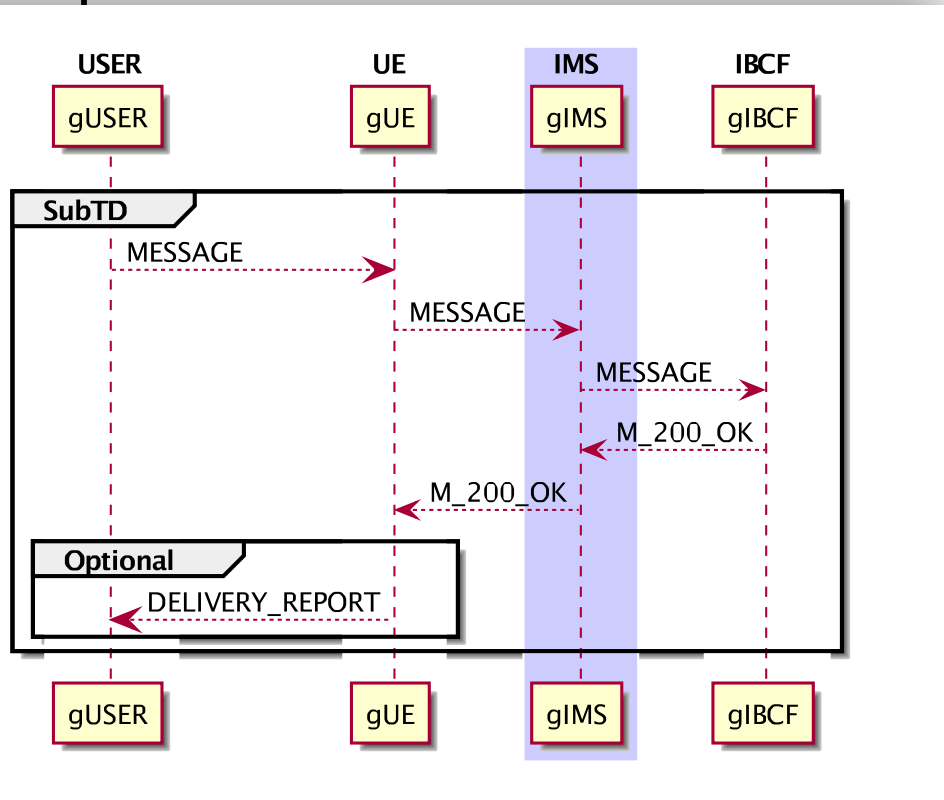
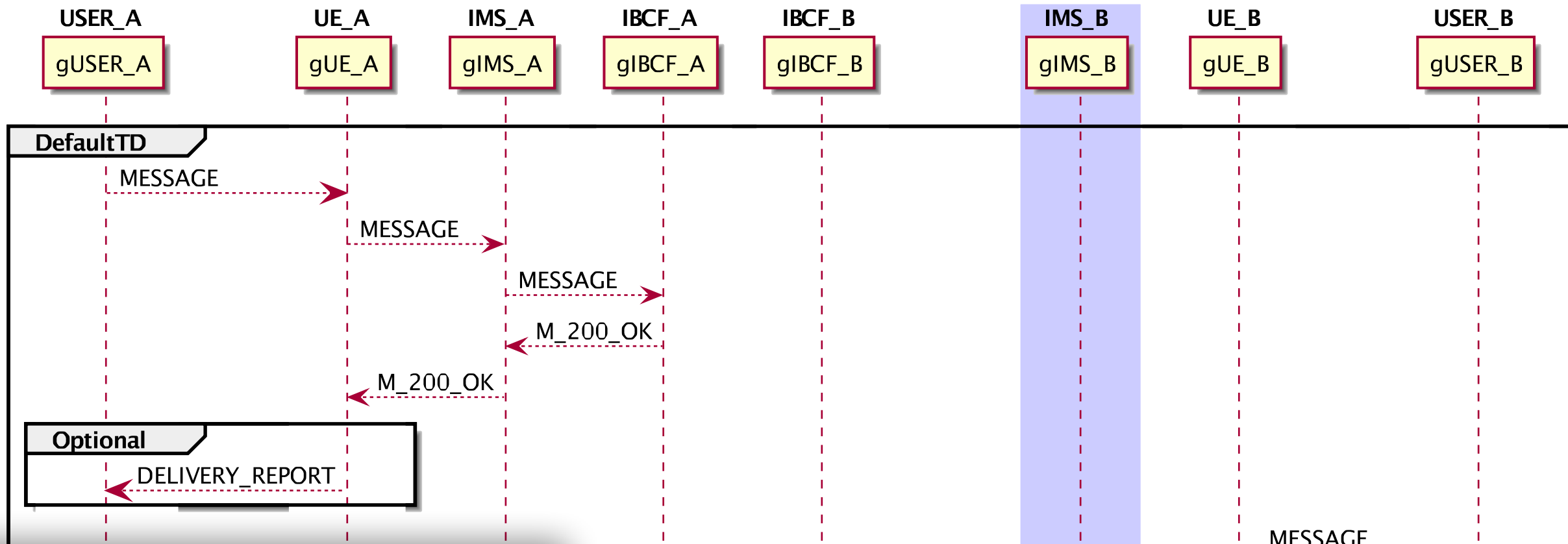
- Extracted sub-configuration

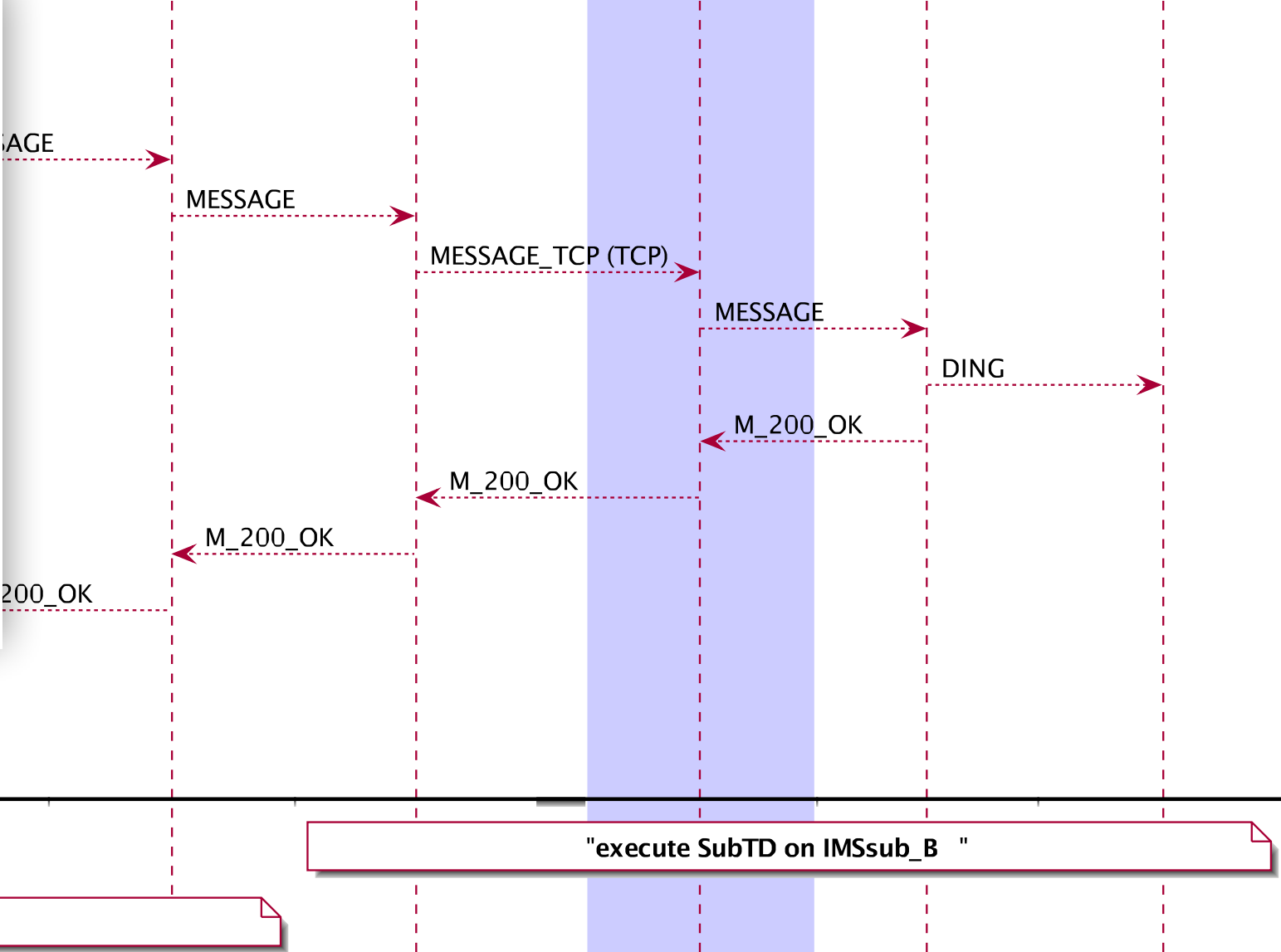
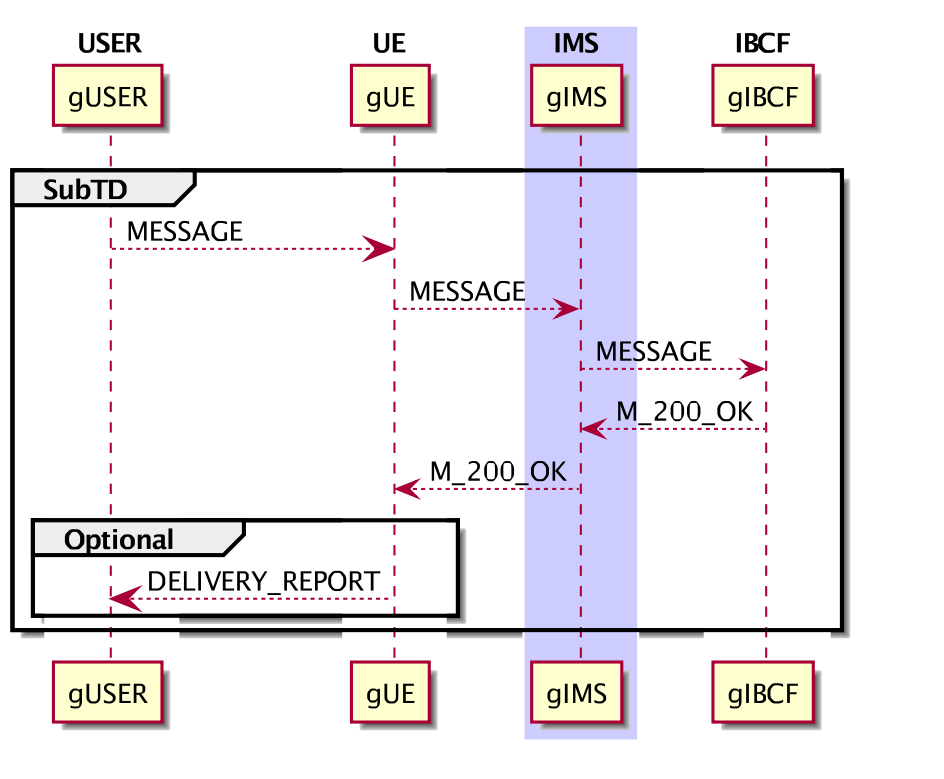
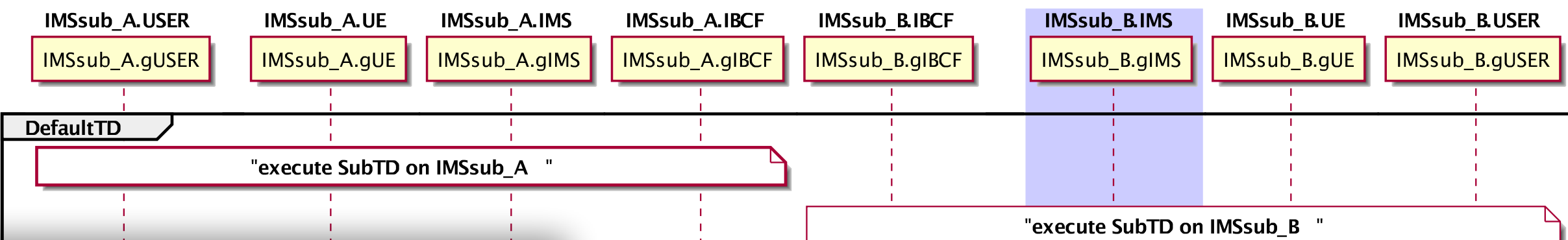
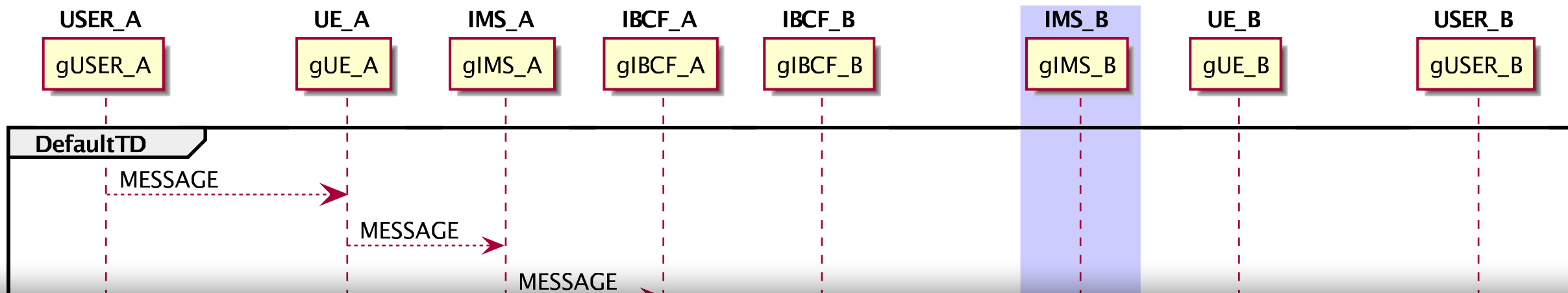


# Task 1: Sub-configurations Example

- “Super”-configuration

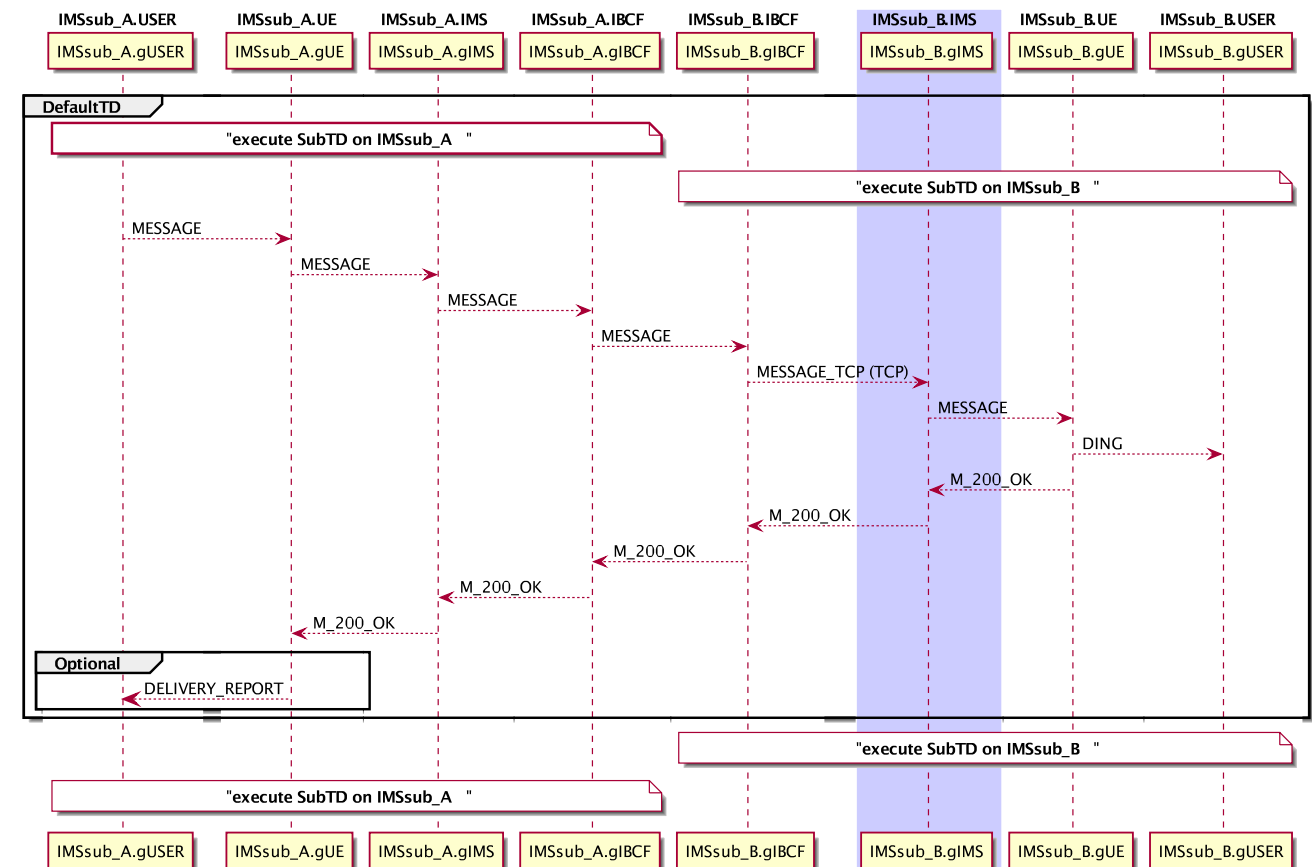
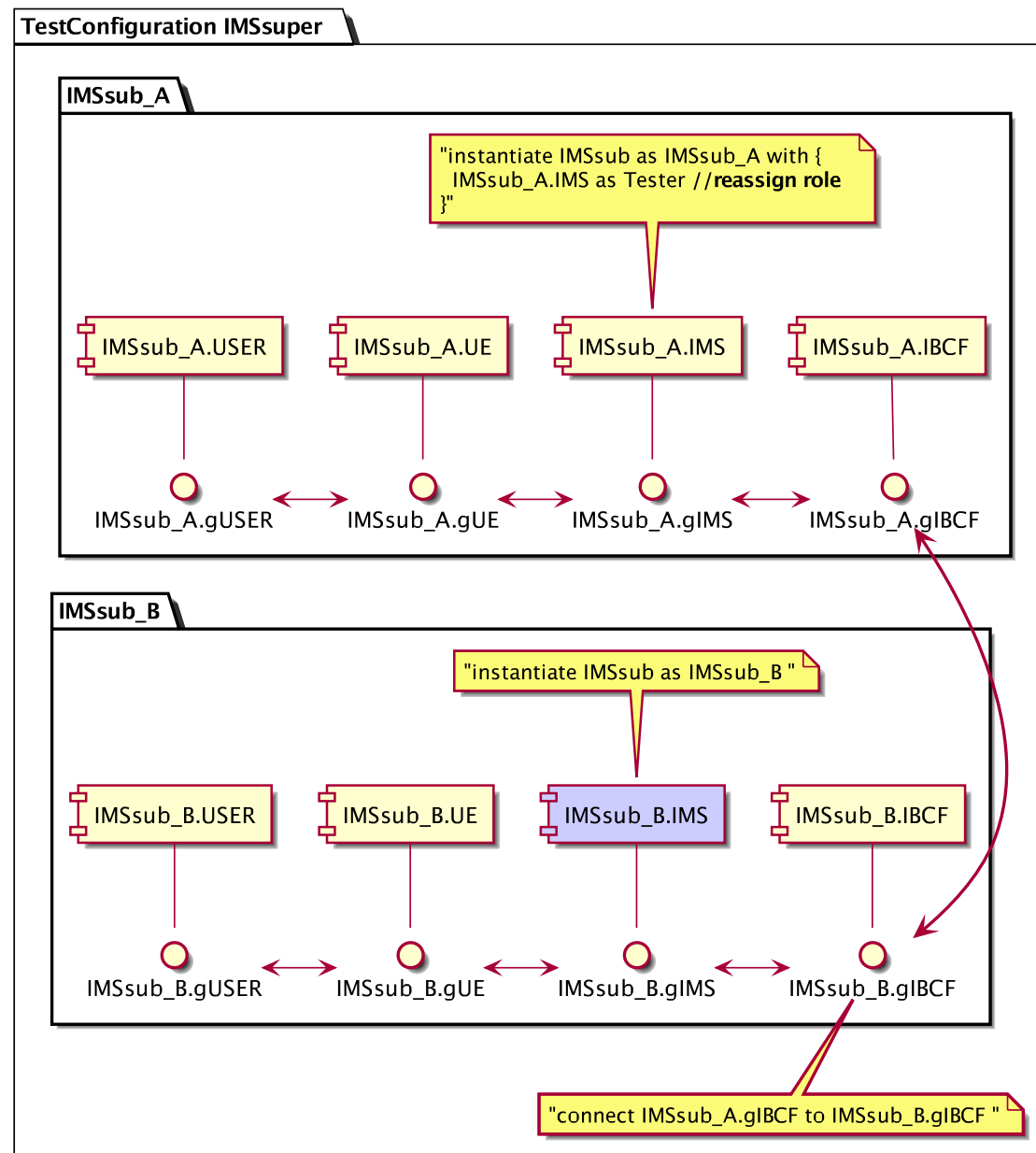






# Task 1: Sub-configurations Example

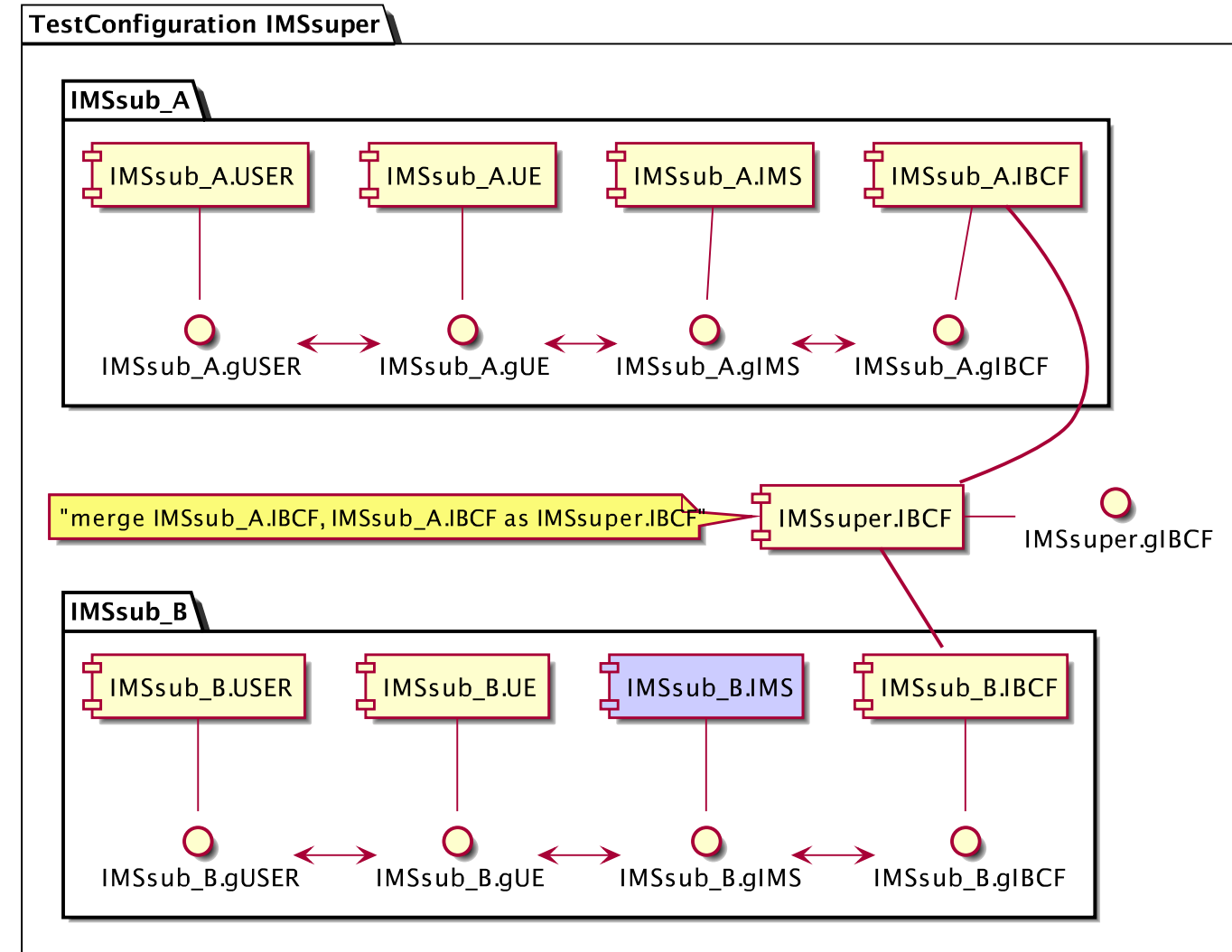
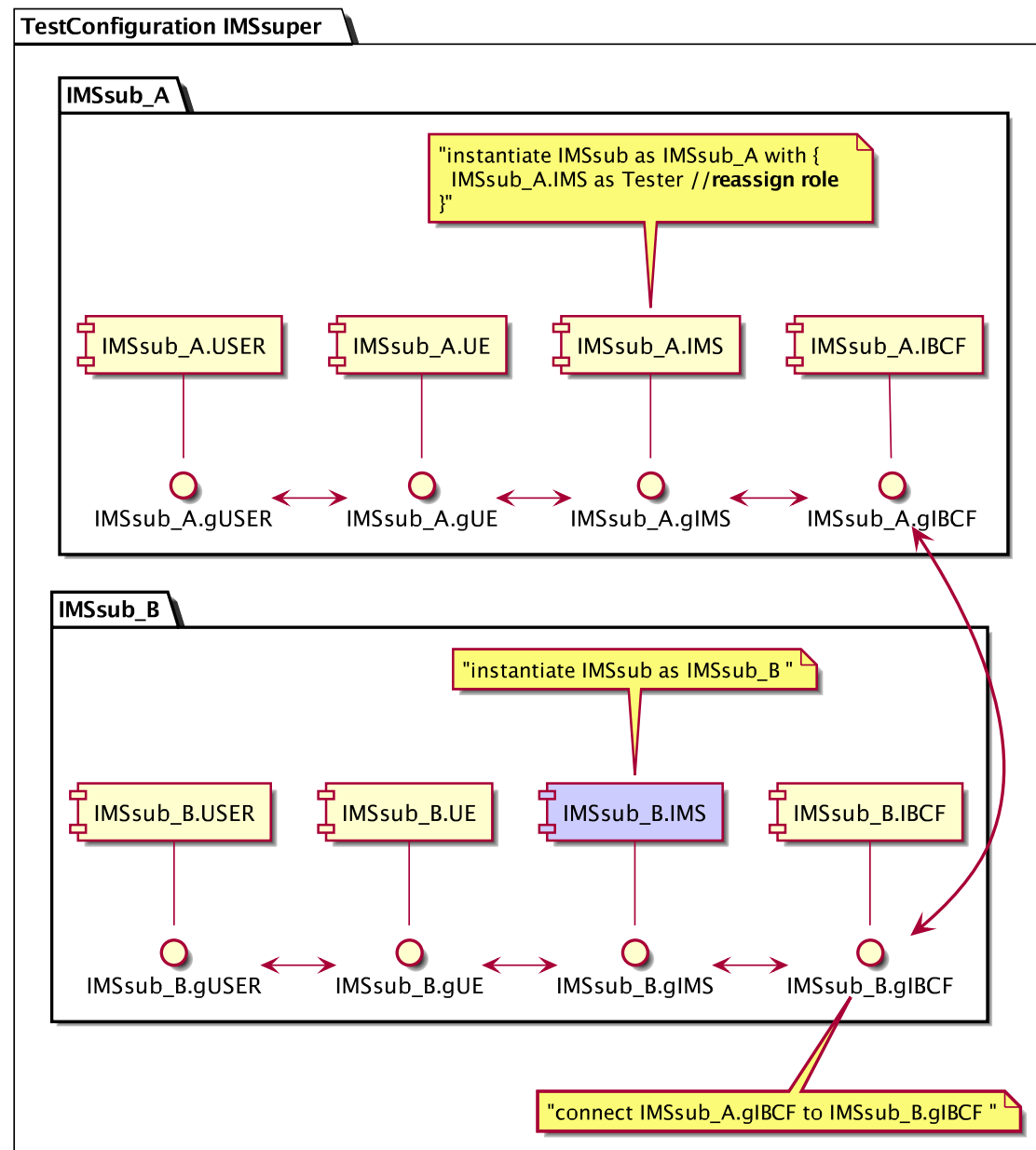
- “Super”-configuration

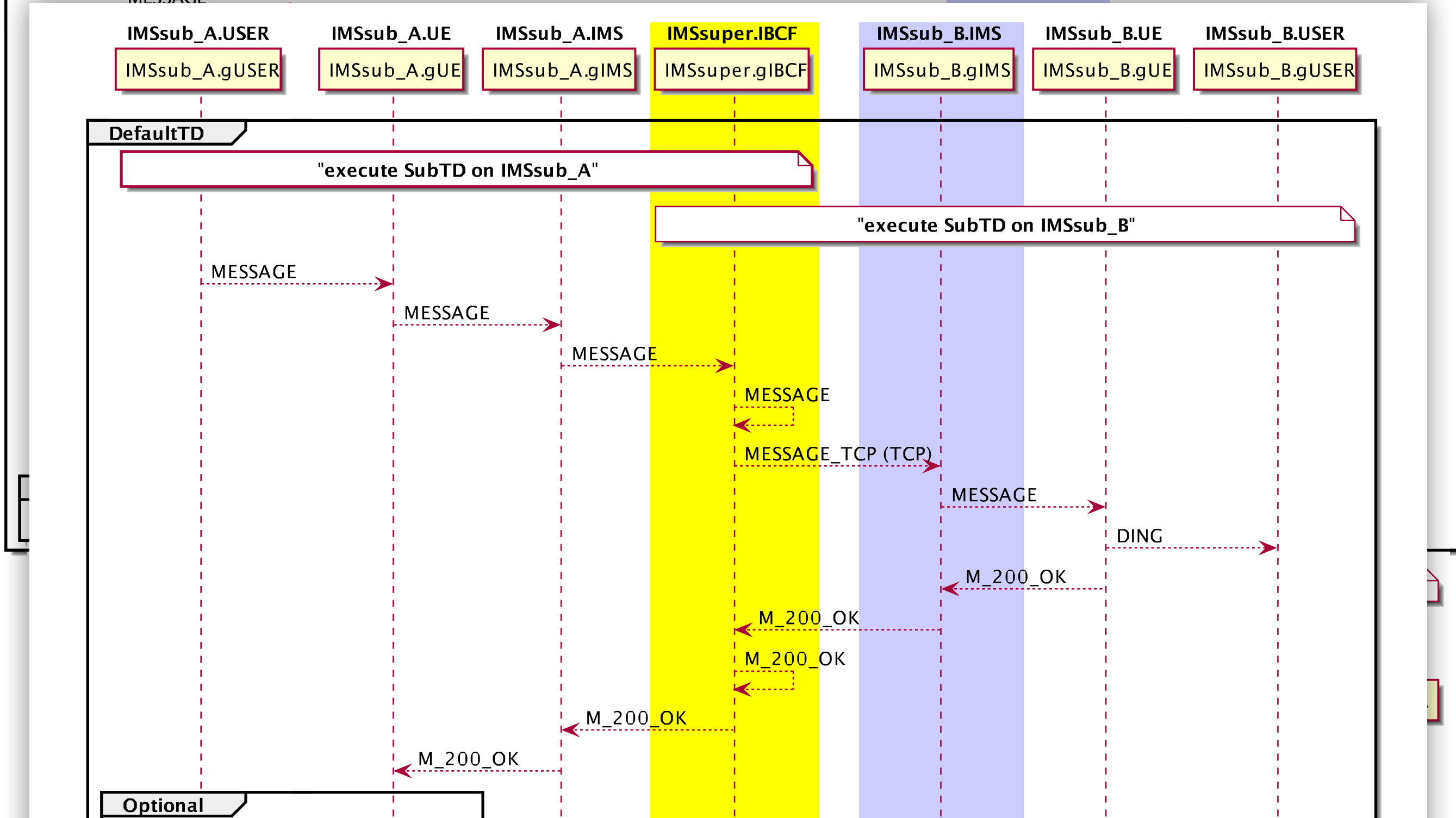
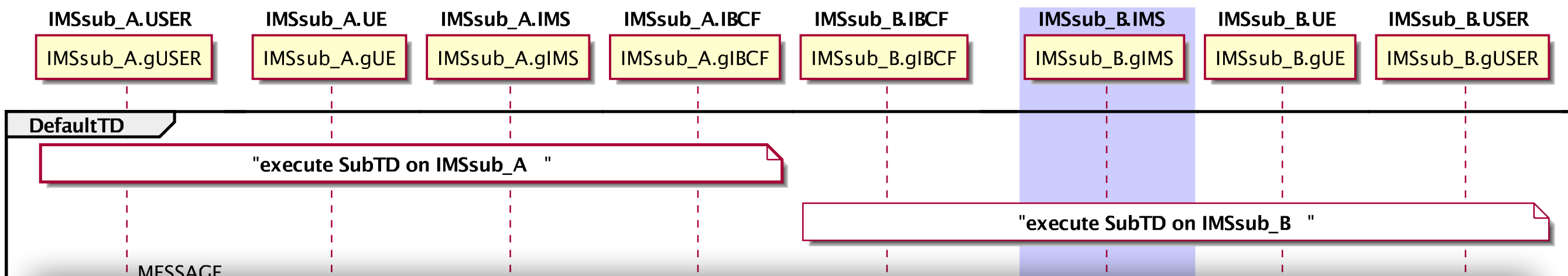




# Task 1: Merging Sub-configurations

- Merged "Super"-configuration





# Task 1: Data Concepts Summary

---

- Targeted for M1:
  - clarification and refinement of separation between notions for types and templates and their TDL counter-parts
  - conceptual proposal for local variables in order to make the specification of data flows more explicit
  - conceptual proposal for handling of data expressions and data operations

# Task 2: Graphical Concrete Syntax

---

- Different specification approaches studied
- OMG Diagram Definition (DD) selected “in principle”
- Means for user-accessible specification under study
- Decision on description and mapping structure pending

# Task 2: Diagram Definition Summary

---

- Kinds of graphical syntax information
  - User can control Diagram Interchange (DI)
    - position of nodes, interconnections
    - interchangeable between tools
  - Defined by language standard Diagram Graphics (DG)
    - shape, style of symbol
    - not interchangeable, shall be identical and is known a priori
- Common basic elements, types Diagram Common (DC)

# Task 2: Diagram Definition Architecture

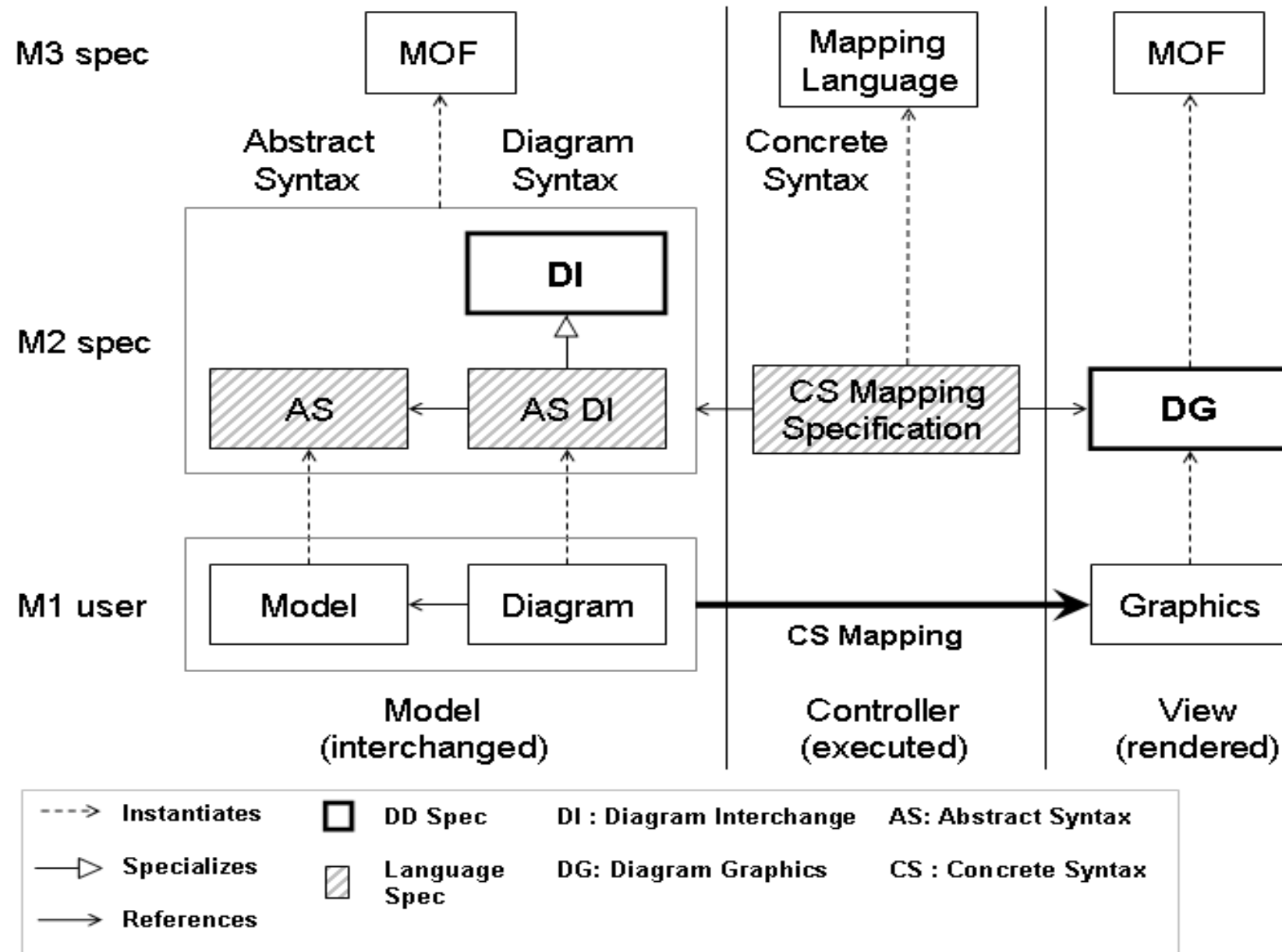


Figure 7.1 - Diagram Definition Architecture

# Task 2: Diagram Definition Architecture

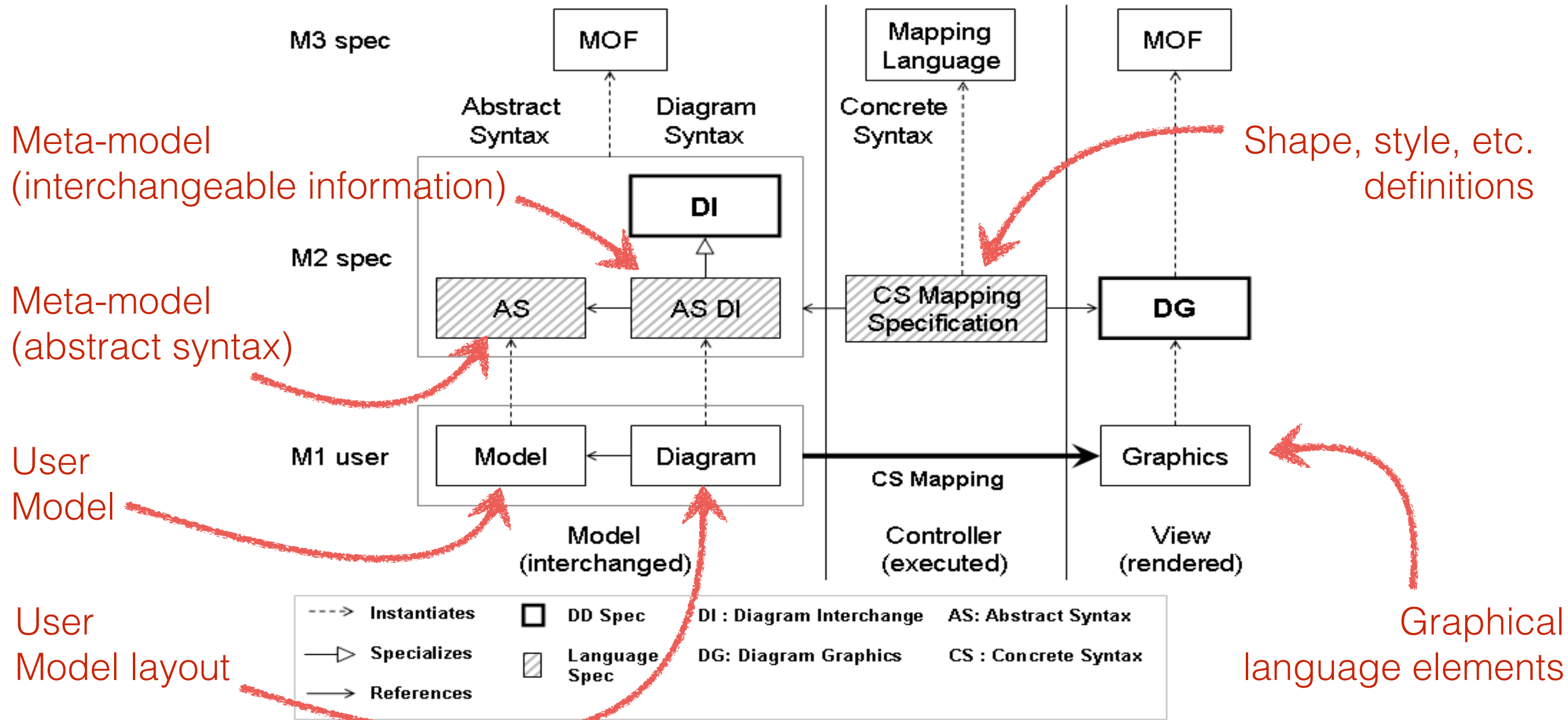


Figure 7.1 - Diagram Definition Architecture



# Task 2: Diagram Definition UML

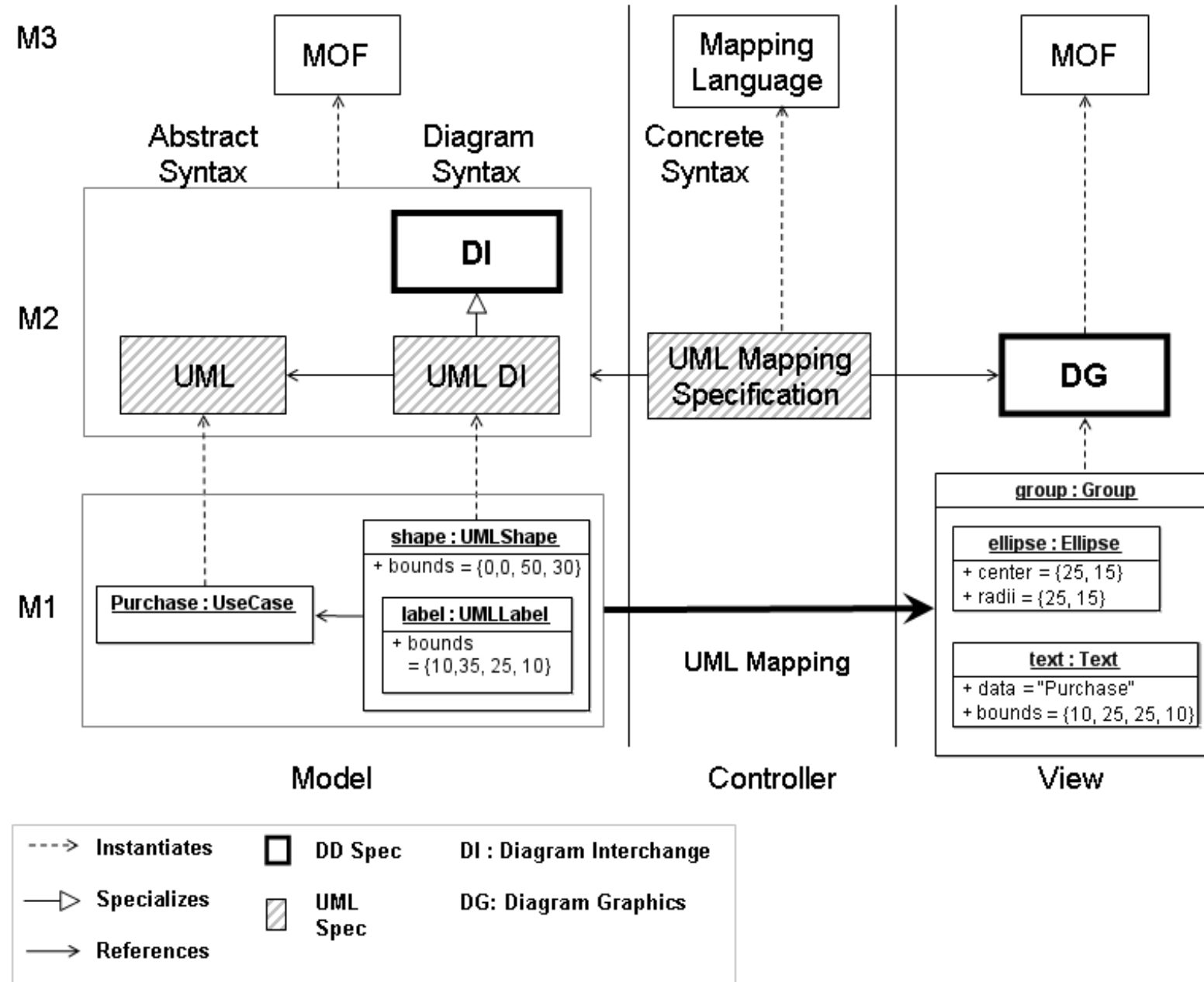


Figure 7.2 - Example of Diagram Definition Architecture For UML

# Task 2: Diagram Definition TDL

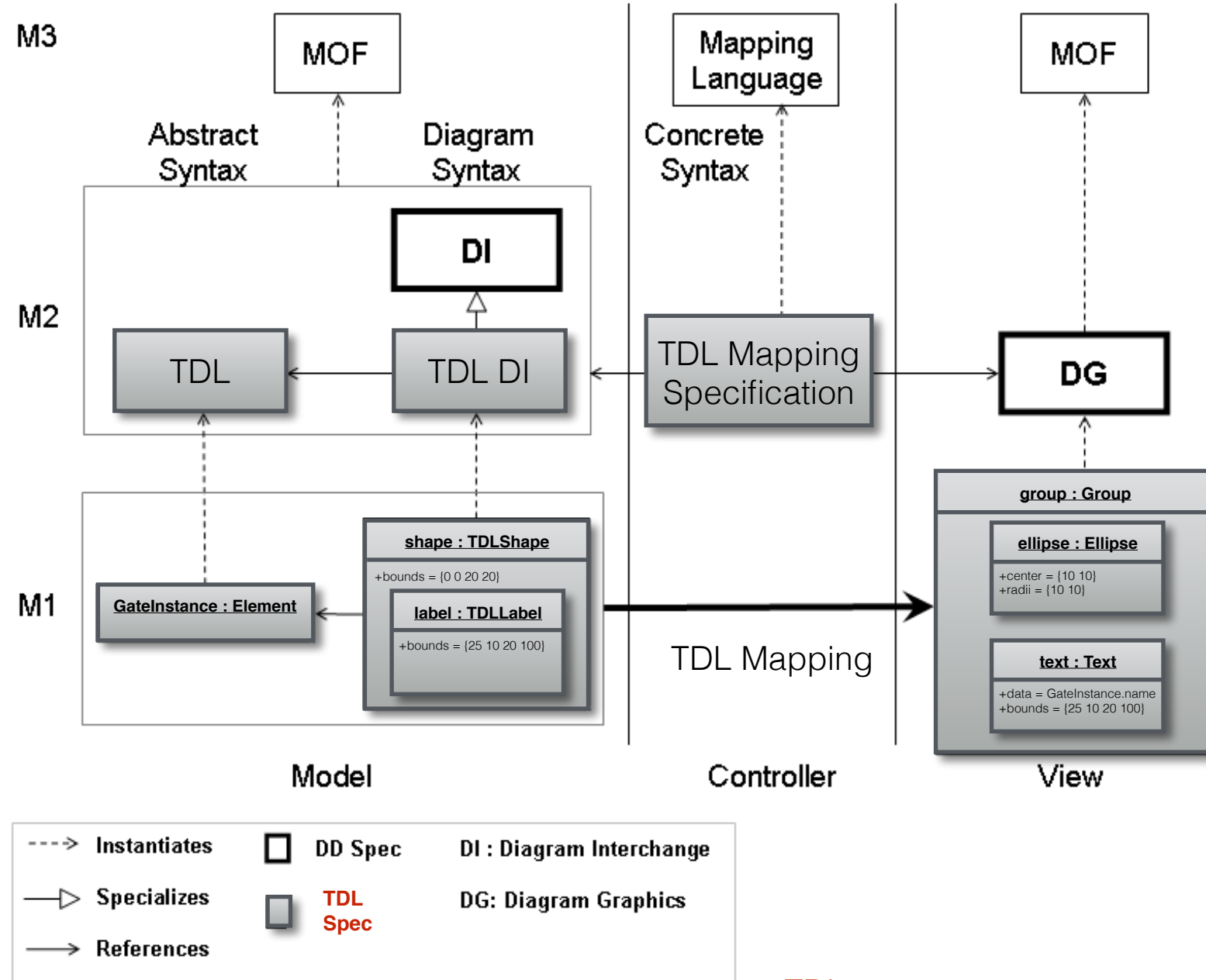


Figure 7.2 - Example of Diagram Definition Architecture For ~~UML~~ **TDL**

# Task 2: Diagram Definition TDL

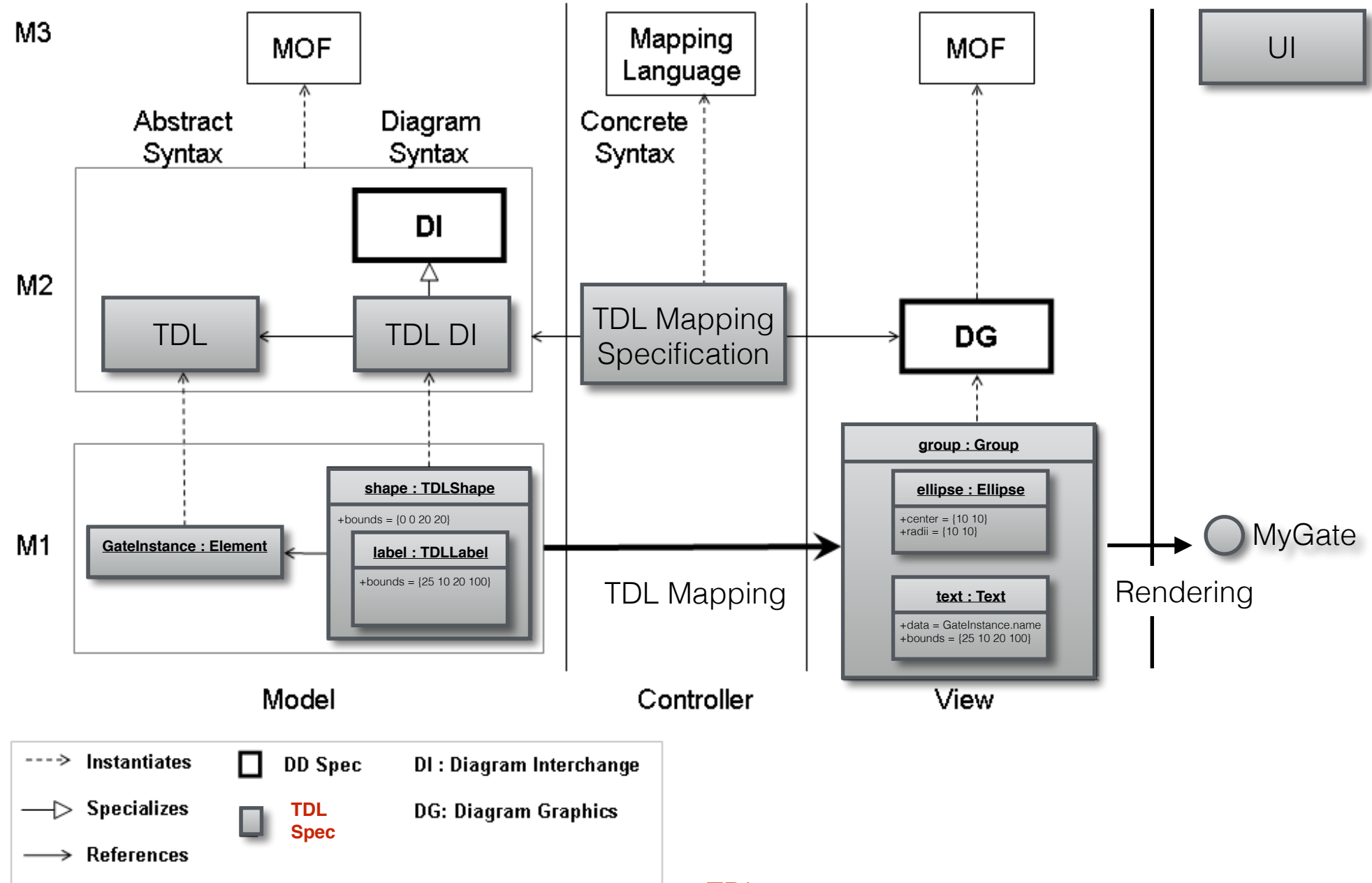


Figure 7.2 - Example of Diagram Definition Architecture For ~~UML~~ <sup>TDL</sup>

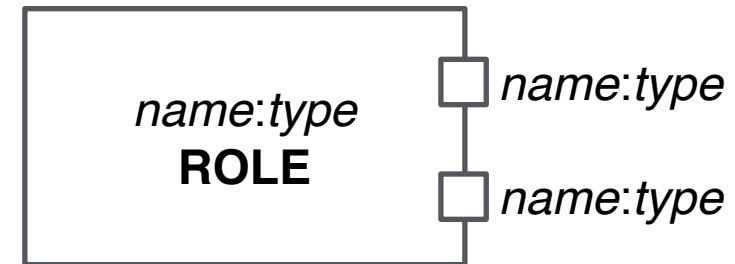
# Task 2: Graphical Syntax Examples

## Gate Instances

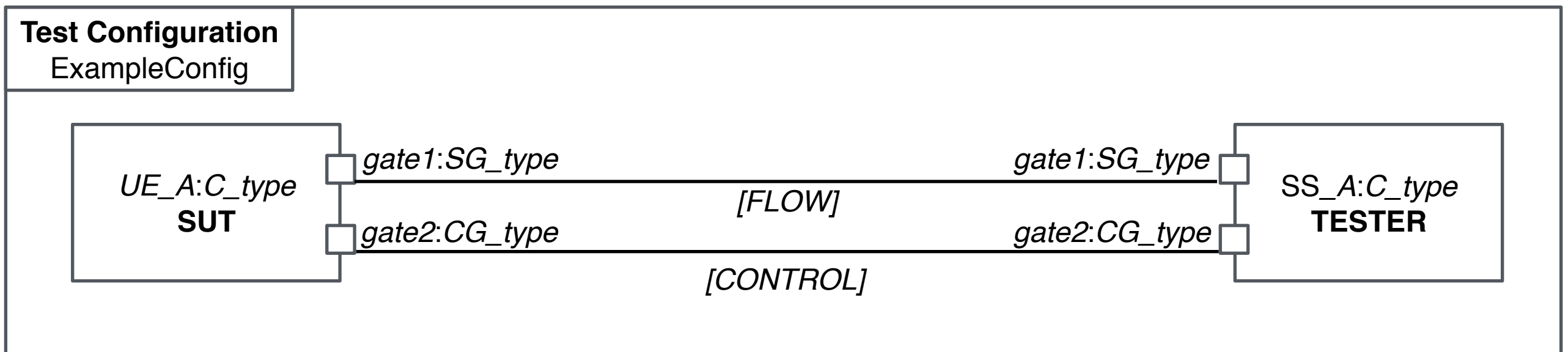
○ *name:type*

□ *name:type*

## Component Instances

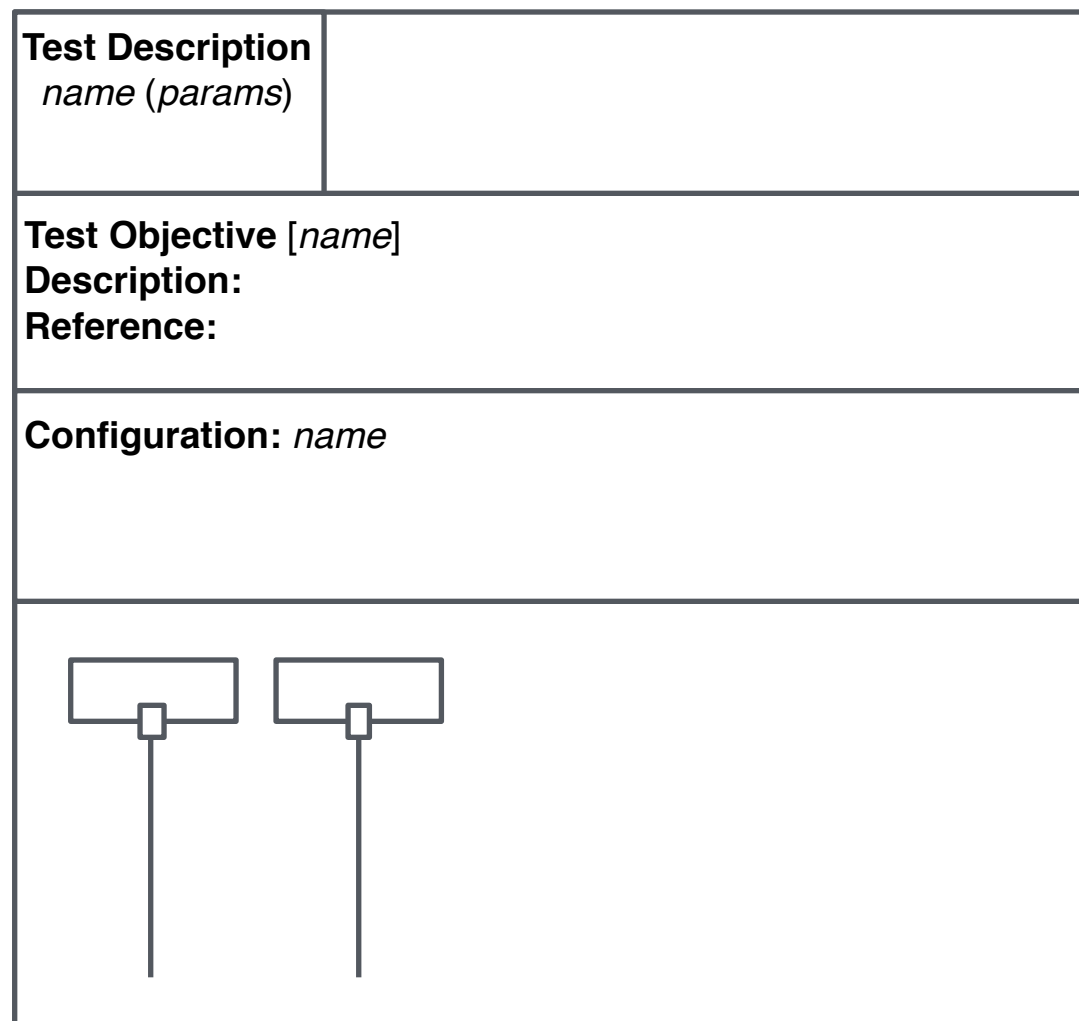


## Test Configurations



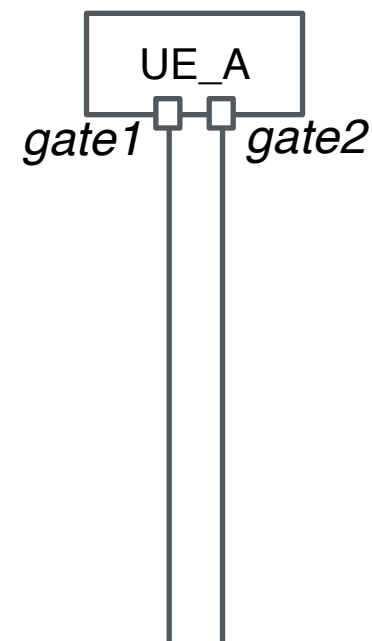
# Task 2: Graphical Syntax Examples

## Test Descriptions

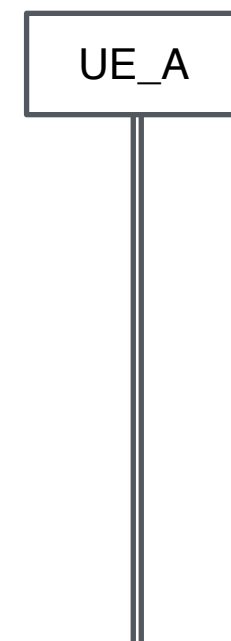


## Behaviour

### Detailed View



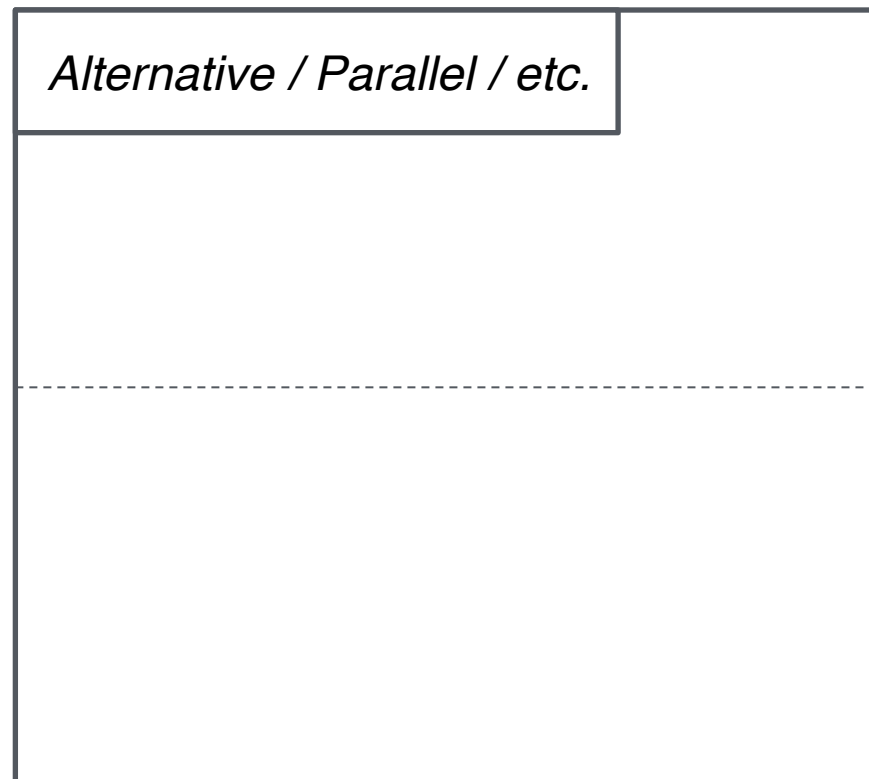
### Summary View



# Task 2: Graphical Syntax Examples

---

## Combined Behaviour



## Interactions

Point-to-point



Broadcast



Further proposals for time operations, atomic behaviour (not shown)

# Task 2: Scope of M1 and Beyond

---

- Targeted for M1:
  - initial syntax proposals for Test architecture, Test behaviour, Time
  - initial draft structure proposal
  - identification of elements that can or shall not be graphically represented
- Feedback appreciated on:
  - level of detail for formal specification
  - need for a graphical syntax meta-model

# Task 2: Prototyping and Validation

---

- Support definition and validation
- Improve STF process
- Different prototyping options evaluated, Sirius selected
  - interpretative, fast prototyping
  - supports DD “in principle”
  - template for sequence diagrams



# Task 4: TDL Textual Syntax Analysis

---

- Understanding ETSI's requirements for a textual syntax
  - review and discussion of notes and input collected during STF 454
  - discussion of CTI proposal based on ITS and GeoNetworking examples
  - focus initial analysis on provided examples, TPLan-like format pushed within ETSI

# Task 4: TDL Textual Syntax Analysis

---

- Target audiences and use cases
  - documentation used in discussions at meetings
    - level of unnecessary technical detail shall be reduced as much as possible
- input / design for test specifications
  - previously done based on test purpose description (1-2 sentence description of objectives), or on requirements directly
  - current format is considered an improvement over these earlier approaches, has high acceptance among test engineers

# Task 4: Examples

---

## 5.1.1 TP definition conventions

The TPs are defined by the rules shown in table 2.

**Table 2: TP definition rules**

<b>TP Header</b>	
TP ID	The TP ID is a unique identifier. It shall be specified according to the TP naming conventions defined in clause 5.1.2.
Test objective	Short description of test purpose objective according to the requirements from the base standard.
Reference	The reference indicates the sub-clauses of the reference standard specifications in which the conformance requirement is expressed.
PICS Selection	Reference to the PICS statement involved for selection of the TP. Contains a Boolean expression.
<b>TP Behaviour</b>	
Initial conditions	The initial conditions defines in which initial state the IUT has to be to apply the actual TP. In the corresponding Test Case, when the execution of the initial condition does not succeed, it leads to the assignment of an Inconclusive verdict.
Expected behaviour (TP body)	Definition of the events, which are parts of the TP objective, and the IUT are expected to perform in order to conform to the base specification. In the corresponding Test Case, Pass or Fail verdicts can be assigned there.
Final conditions	Definition of the events that the IUT is expected to perform or shall not perform, according to the base standard and following the correct execution of the actions in the expected behaviour above. In the corresponding Test Case, the execution of the final conditions is evaluated for the assignment of the final verdict.

# Task 4: Examples

---

16

ETSI TS 102 868-2 V1.1.1 (2011-03)

<b>TP Id</b>	TP/CAM/INA/DOP/BV/02
<b>Test objective</b>	Checks that CAM message includes DoorOpen information 30s after closed
<b>Reference</b>	TS 102 637-2 [1], clauses 7.1 and 7.2
<b>PICS Selection</b>	PICS_PUBTRANSVEH
<b>Initial conditions</b>	
with { the IUT being in the "initial state" and the IUT having sent a valid CAM message containing DoorOpen TaggedValue }	
<b>Expected behaviour</b>	
ensure that { when { the door is closed } then { the IUT sends CAM messages containing DoorOpen TaggedValue during the 30s following the door closing event } }	

# Task 4: TDL Textual Syntax Analysis

---

- Study TPLan
  - determine the extent to which it *can be* mapped to TDL as a concrete syntax
  - determine the extent to which it *needs to be* mapped to TDL in order to cover the examples from ITS, GeoNetworking, and others
  - investigate the limitations of both TDL and TPLan with regard to their integration and prepare suggestions for discussion and adaptation

# Task 4: TDL Textual Syntax Analysis

---

- Evaluate options for mapping to TDL
  - study applicability and constraints of graphical syntax
  - study examples and possible mapping means (direct vs indirect)
    - preliminary analysis indicates indirect mapping as a better, more flexible solution with fewer compromises
  - assess impact and implications of potential restrictions of mappings
    - level of formalisation (and by extension scope of mapping to TDL) pending discussion
- Propose a mapping approach (report, ToR if applicable)

# Any Other Business?

---

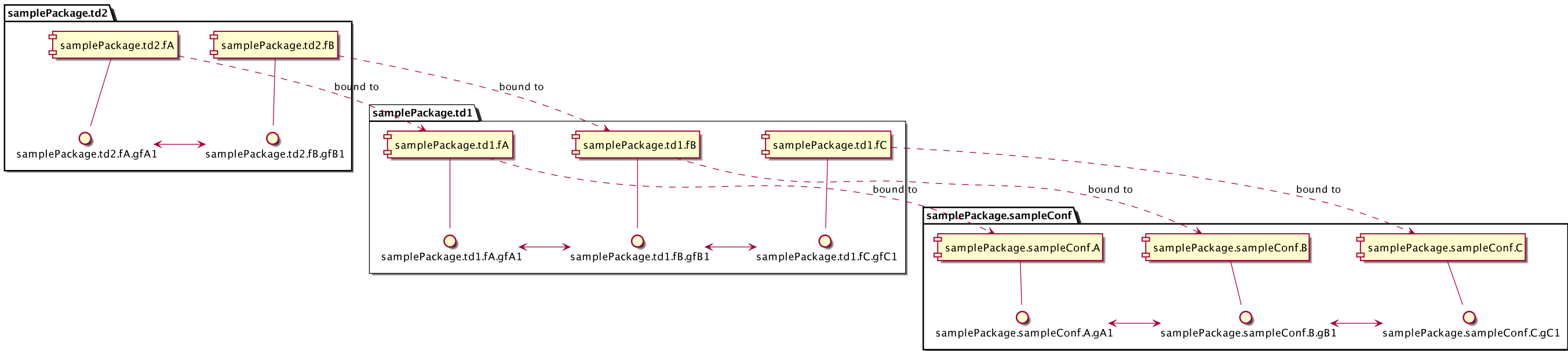
# Backup Slides



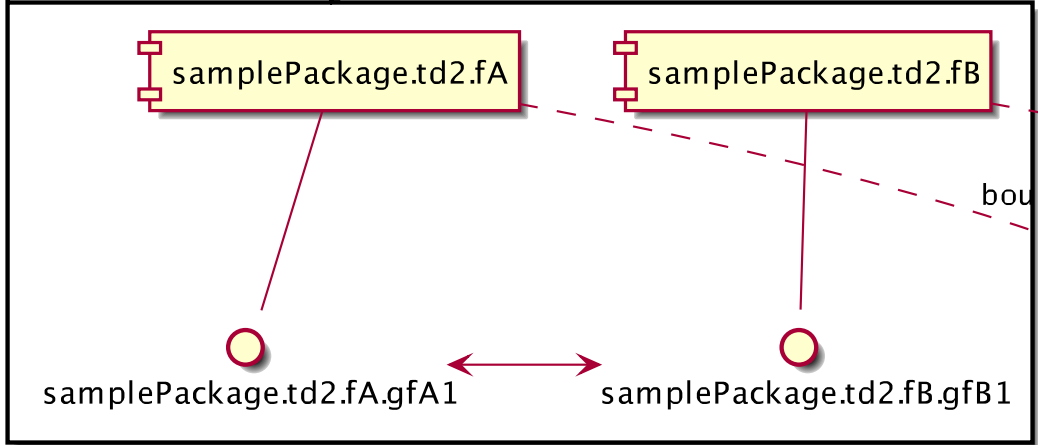
# Task 1: Sub-Configurations

---

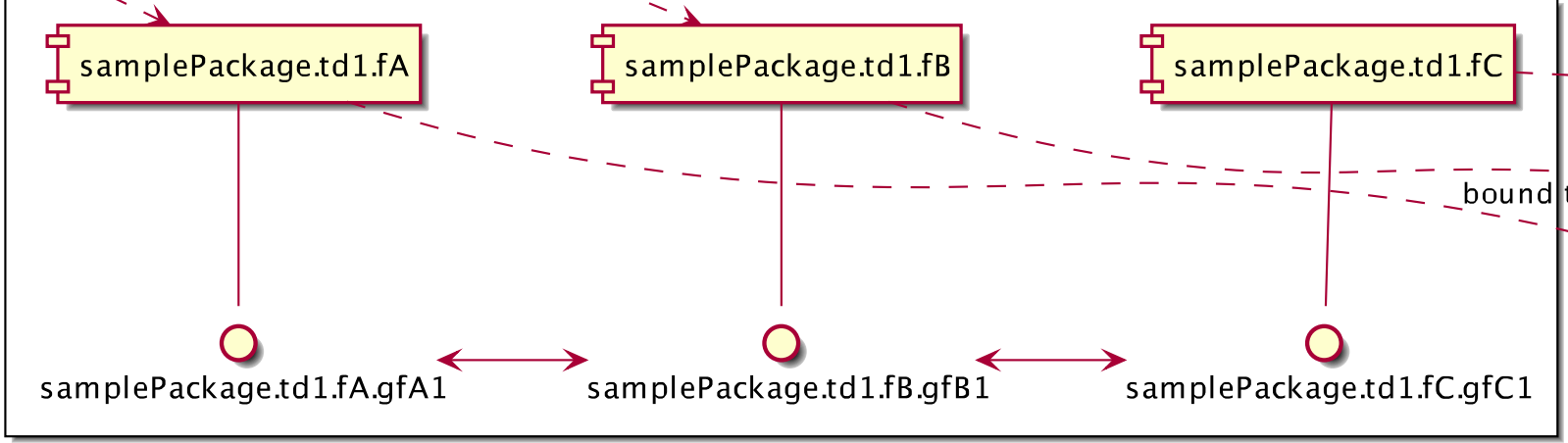
- Parameterisation-based approach (minor meta-model changes)
- Test Descriptions define their own local component instances
  - existing configurations and instances can be reused as copies
  - connections can be explicit (consistent) or implicit (convenient)
- Formal and actual component instances bound in references
  - binding concrete instances at reference time vs declaration time
  - $m : n$  vs  $1 : n$  relationships (configuration : test description)



**samplePackage.td2**



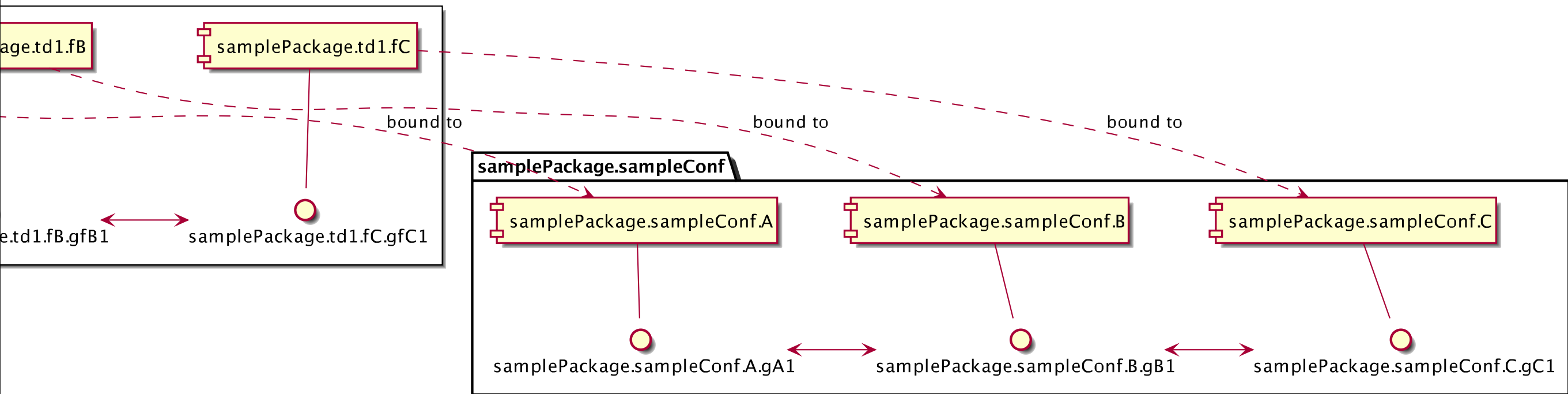
**samplePackage.td1**



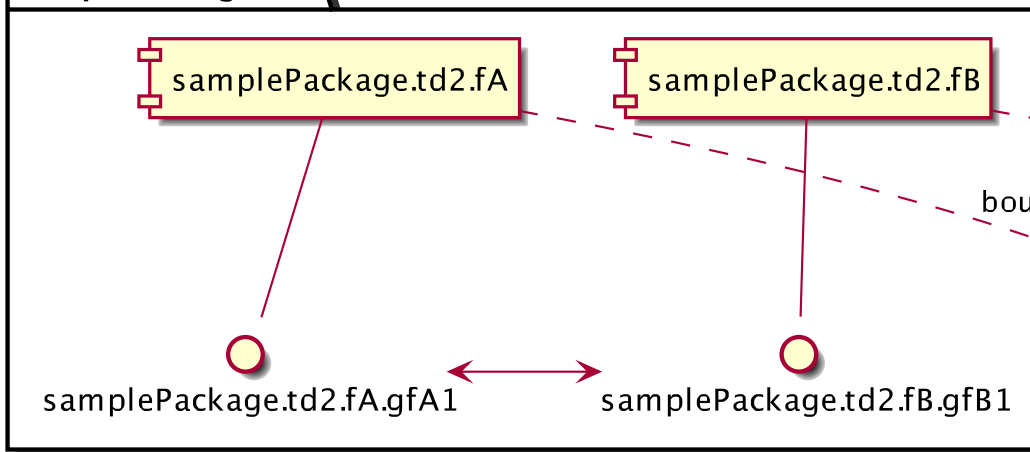
bound to

bound to

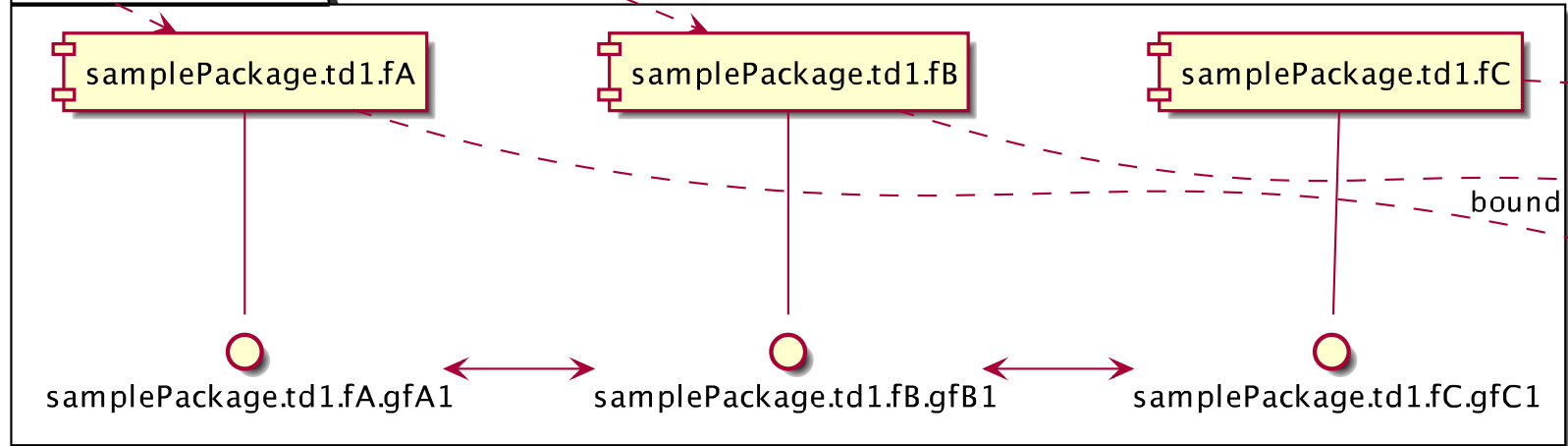
bound to



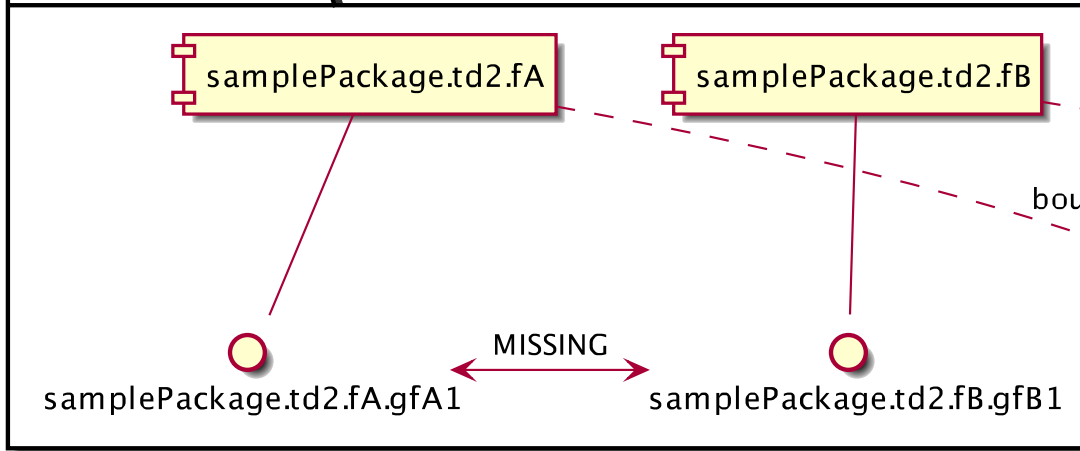
samplePackage.td2



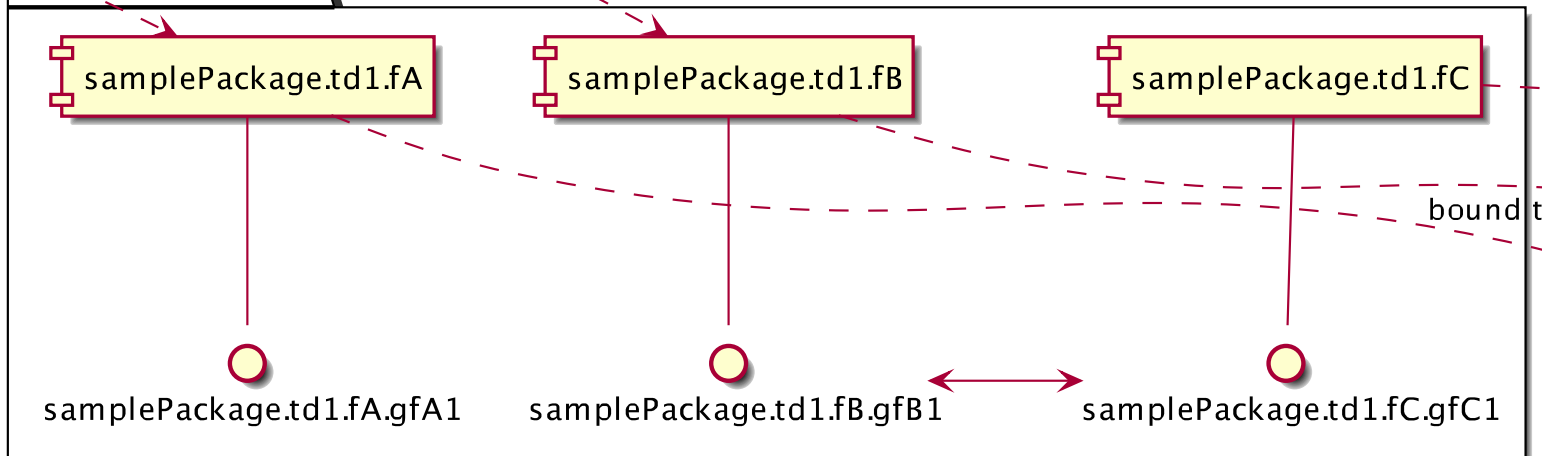
samplePackage.td1

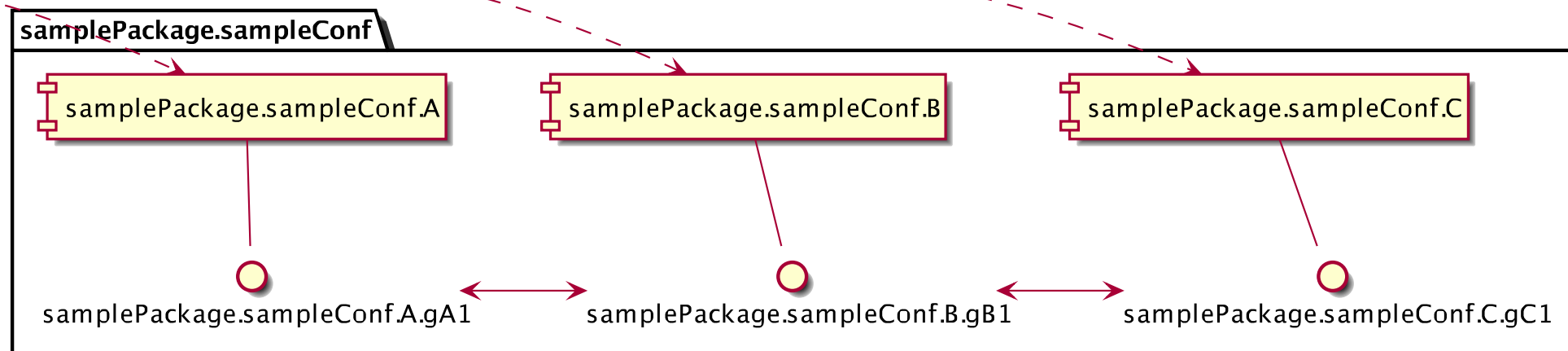
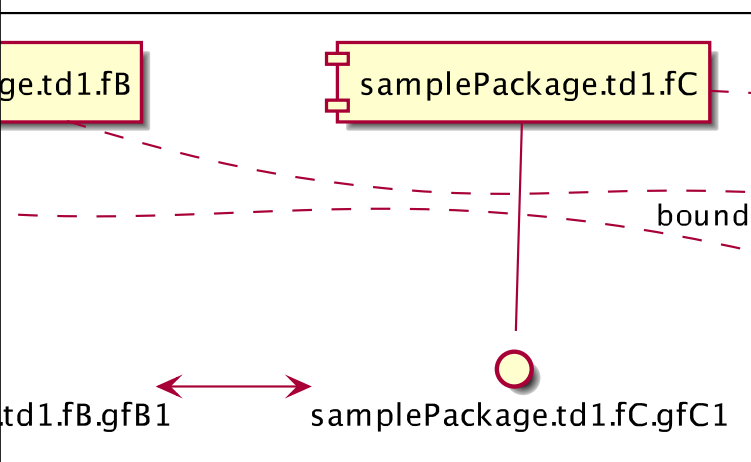
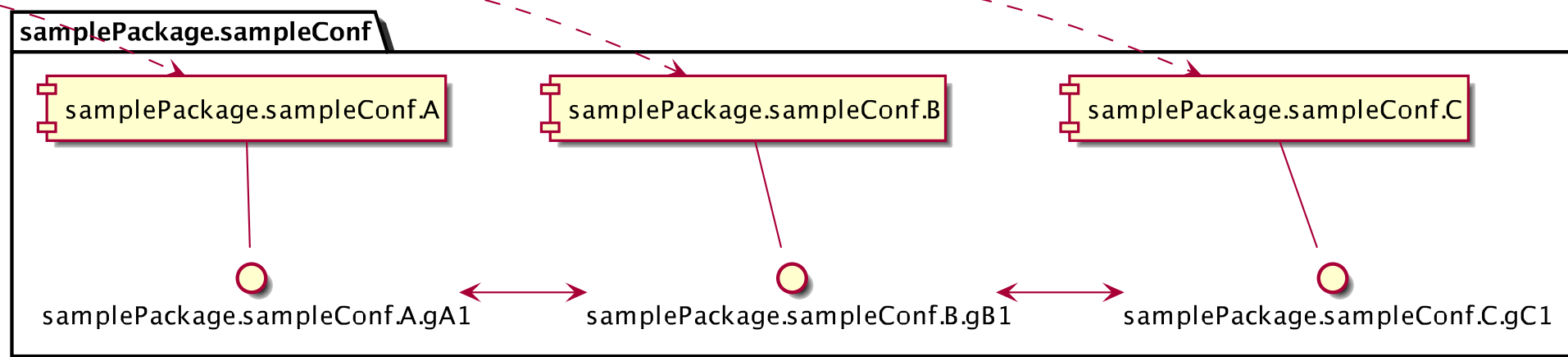
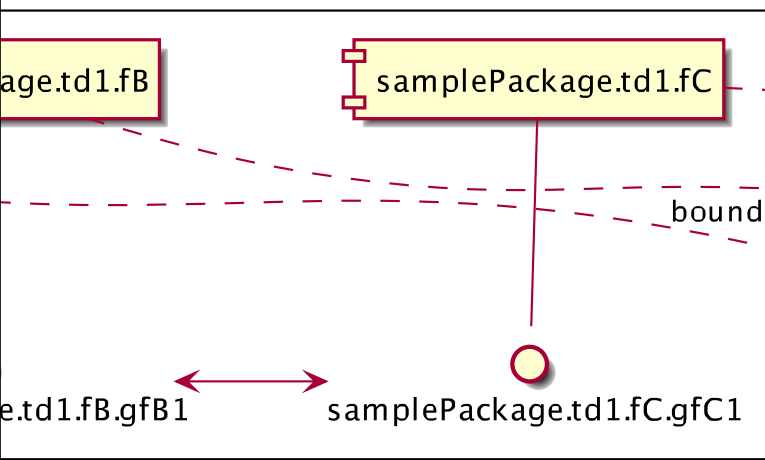


samplePackage.td2



samplePackage.td1





## STF 476: Rough Overall Resource Planning and Allocation for 2014

Description	Goals	4-day Sessions	5-day Sessions	Experts
First Session (Feb 2014)	Kick off, Post Phase 1, Prep M1	16	16	4
Homework and coordinated sessions				
Second Session (Apr 2014)	Finalise M1: Early Drafts	16	20	4
Finalisation homework if needed				
Third Session (Jun 2014)	Post M1, Prep M2, MFW available	20	25	5
Homework and coordinated sessions				
Fourth Session (Sep 2014)	Finalise M2: Stable Drafts	20	25	5
Finalisation homework if needed				
Fifth Session (Oct/Nov 2014)	Post M2, Prep M3	20	25	5
Homework and coordinated sessions				
Sixth Session (Dec 2014)	Finalise M3: Final Drafts	20	25	5
Finalisation homework if needed				
Milestone 1	April 2014	32	36	
Milestone 2	September 2014	40	50	
Milestone 3	December 2014	40	50	
Average resources per milestone		69,3	69,3	
Average contingency per milestone	For homework / session extension	32,0	24,0	
Contingency per milestone per expert	For homework / session extension	6,4	4,8	
<b>Planned / used resources</b>		<b>112</b>	<b>136</b>	
<b>Total resources available</b>		<b>208</b>	<b>208</b>	
<b>Contingency</b>		<b>96</b>	<b>72</b>	
<b>Per homework session (all experts)</b>		<b>16</b>	<b>12</b>	

# Task 0: STF Process

---

- Define and set target goals for milestone
  - distributed among experts based on task responsibilities
  - approved by STF
- Execute and refine goals
  - analysis
  - implementation
  - validation



# Task 0: Execution

---

- Analysis
  - assigned expert understands goal and proposes a conceptual solution
  - STF approves conceptual solution
  - redo analysis and propose an improved conceptual solution in case of deficiencies
- Implementation
  - assigned expert implements the conceptual solution in the respective document

# Task 0: Execution

---

- Validation
  - STF reviews the implementation
  - prototypical realisation checks technical soundness where applicable
  - go back to implementation in case of minor deficiencies
  - go back to analysis in case of major deficiencies

# Communication STF – SG

---

- Ensure overall direction of TDL design is OK
- Resolve deadlocks in STF internal discussions by providing guidance and advice
- Assist in developing a roadmap for TDL

# TDL Use Cases

---

U	Short Description	Example
A	TDL for documentation (incl. informal parts)	3GPP test specs
B	TDL for generation of tests that can be made executable (i.e. all parts are formal)	Automatic mapping of a TDL spec to partial TTCN-3 code
C	TDL for representation of generated tests (i.e. output from MBT tools)	Test cases generated from system models
D	TDL for representation of test logs	Test execution log of a TTCN-3 tool
E	TDL for test generation (i.e. input to MBT tools)	Test models as activity diagrams
F	TDL for performance testing	On-the-fly testing from a TDL spec
G	TDL for interoperability testing	Use case models, from which tests are derived

# TDL Feature Description Structure

---

- TDL feature name (title of sub-clause)
- Overview (covered in early draft)
  - Free description of the feature
- Abstract syntax
  - Representation of the feature and its elements in the meta-model
- Semantics
  - Preferably formal description of the semantics of the feature
- Constraints
  - Constraints on the feature that can be statically analysed
- Classifier description
  - Description of all elements contained in the meta-classes