Abstract Telephone Communication
Service Constituent module
IN-Lab2 report 93/5

Trygve Reenskaug, Taskon
Knut Lillegraven, NTR
28 July 1993

The module defined in this report defines the main objects involved in establishing and removing a one-to-one telephone connection together with a prescription of the interaction between these objects. There are two objects, they are responsible for the calling (A) and called (B) side of the connection respectively.

The module is abstract in the sense that it only defines the A- and B- roles without any object type specifications. The corresponding concrete objects will play additional roles, e.g. an invocation model, but this specification is kept independent of the choice of the additional models.
Report overview.
Chapter 1
Introduction

This document describes the Abstract Telephone Communication OOram Module on the service sub-layer of the Service Constituent Creator Layer.

![Diagram of IN-Lab2 support value chain.]

This module covers the interaction between the telephone connection services of two users (A and B). Both the A- and B-services are capable of handling real-time telephony calls. Service A handles telephony calls in a way that suits A, while service B handles calls in a way that suits user B. In this role model service A is capable of generating new outgoing calls, while service B handles incoming calls.

The stimulus of this role model is a desire by service A to create a connection (possibly initiated by some user interface). It is assumed that service B initially knows little about the state of service A and the type of call A wants to make, just that service A is capable of generating calls.
This role model does not cover a complete protocol between service A and service B, just messages that are relevant for the TINA demo. It is also expected that this role model can be split into different role models covering different aspects of service interaction, e.g. charging, connection setup, security, etc. For the sake of simplicity and due to our relatively simple services we merge this into one role model at this stage.

The models of this module are as follows:

1. **Import models**
   - None

2. **Role Models**
   - *Telephone Service Interaction (Tel2IntRM).* This model shows the basic interaction between the A- and B- service objects.

3. **Object Specifications**
   - None

4. **Export models**
   - *Telephone Service Interaction (Tel2IntEXP).* This mechanism shows the basic interaction between the A- and B- service objects. In our simple example, it is identical to the above role model.

---

**Figure 2. The model import/export structure of the Abstract Telephone Connection module**
Chapter 2
Abstract Telephone Service export summary

2.1 Abstract Telephone Interaction EXP (Abstr2IntEXP) {Export Model}

Report including the following selections automatically generated 28 July 1993: Export Models
Area of Concern - stimuli Role list - diagram Scenarios - explanation

2.1.1 Area of Concern

This role model covers the interaction between the services of two users (A and B). Both the A- and B-services are capable of handling real time telephony calls. Service A handles telephony calls in a way that suits A, service B handles calls in a way that suits user B. In this role model service A is capable of generating new outgoing calls, while service B handles incoming calls.

The stimulus messages of this role model is a request by service B for context information from service A. This message is typically the first message sent from service B after it has been created and started. It is assumed that service B initially knows little about the state of service A and the type of call A wants to make, just that service A is capable of generating calls.

This role model does not cover a complete protocol between service A and service B, just messages that are relevant for the TINA demo. It is also expected that this role model can be split into different role models covering different aspects of service interaction, e.g. charging, connection setup, security, etc. For the sake of simplicity and due to our relatively simple services we merge this into one role model at this stage.
2.1.2 The Roles

2.1.3 Interaction Scenarios

Figure 3. Release connection from A [Scenario]
This MSC shows how a connection is released with the initiative coming from the calling side.
Figure 4. Establish connection {Scenario}
This Message Sequence Chart shows the typical message sequence for the successful establishment of a telephone connection.

Figure 5. Copy of Release connection from A (Release connection from B) {Scenario}
This MSC shows how a connection is released with the initiative coming from the called side.
Chapter 3
Role Models

3.1 Telephone Interaction RM (Tel2IntRM) {Role Model}

Report including the following selections automatically generated 28 July 1993: Role Models; Area of Concern stimuli; Role list diagram, role explanation, role import map; Message Sets port explanation, contract explanation, message explanation; FSM

3.1.1 Area of Concern

This role model covers the interaction between the services of two users (A and B). Both the A- and B-services are capable of handling real time telephony calls. Service A handles telephony calls in a way that suits A, service B handles calls in a way that suits user B. In this role model service A is capable of generating new outgoing calls, while service B handles incoming calls.

The stimulus messages of this role model is a request by service B for context information from service A. This message is typically the first message sent from service B after it has been created and started. It is assumed that service B initially knows little about the state of service A and the type of call A wants to make, just that service A is capable of generating calls.

This role model does not cover a complete protocol between service A and service B, just messages that are relevant for the TINA demo. It is also expected that this role model can be split into different role models covering different aspects of service interaction, e.g. charging, connection setup, security, etc. For the sake of simplicity and due to our relatively simple services we merge this into one role model at this stage.

Stimulus messages
1. Tel2AServ>>>open Start the B-service process.
3.1.2 The Roles

1. **Telephone Service-A (Tel2AServ) {Role}**. This role represents the calling part of a telephone connection service. It takes the initiative to establish a service. It is responsible for establishing a Connection Point service in the Switching Domain, and for establishing a leg from the Calling User to this connection point. We assume this functionality is encapsulated in the object, and do not specify how it accomplishes it.

2. **Telephone Service-BB-Serv (Tel2BServ) {Role}**. This role represents the called part of a telephone connection service. It is responsible for establishing a Leg in the Switching Domain from the Called User to a Connection Point provided by the A-Service. We assume this functionality is encapsulated in the object, and do not specify how it accomplishes it.

3.1.3 Message Sets

1. **Telephone Service-A (Tel2AServ) {Role}**
   - bService (sb) {Port}.
   - Tel2BServ<Tel2AServ {Contract}.
     - call: callTyp qoc: qoc {Message}. This message provides input to service B about the type of call A wants to make, and some information about the state of service A. Just relevant state information for service B to be able to proceed should be given.
       Parameters:
       \[ \text{callTyp} = \{ \#Simplex, \#Duplex \} \]
       \[ \text{qoc} \text{ is Integer representing required bandwidth.} \]
     - connected {Message}. This message tells service B that connections have been established and that the call has been successfully completed.
     - releaseFromA {Message}. Release the connection and all associated resources. Terminate the service.
     - open {Message}. Start the B-service process.
     - createLeg: tp direction: dir qoc: qoc toConnPt: conPt {Message}. Create a Leg to the specifies ConnectionPoint.
2. **Telephone Service-BB-Serv (Tel2BServ) {Role}**

   - aService (sa) (Port).
   - Tel2AServ<Tel2BServ (Contract).
     - `provideContext (Message)`. This message requests service A to provide service B with a context for the call.
     - `userBWaiting: accessPoint (Message)`. This message tells service A that Service B has accepted the call request and that Service B is waiting for connections at a certain access point.
     - `releaseFromB (Message)`. Release the connection and all associated resources. Terminate the service.
Chapter 4
Object Type Specifications

None
Chapter 5
Abstract Telephone Service export models

The single export model is identical to the role model described above.
Appendix 1
Implementation for monitored execution

This module has no implementation since it does not have an object type specification.
TABLE OF CONTENTS

Chapter 1  Introduction .................................................................................................................. 2

Chapter 2  Abstract Telephone Service export summary ......................................................... 4
  2.1  Abstract Telephone Interaction EXP (Abstr2IntEXP) {Export Model} ......................... 4
  2.1.1  Area of Concern ............................................................................................................. 4
  2.1.2  The Roles ....................................................................................................................... 5
  2.1.3  Interaction Scenarios ................................................................................................... 5

Chapter 3  Role Models ............................................................................................................. 7
  3.1  Telephone Interaction RM (Tel2IntRM) {Role Model} .................................................... 7
  3.1.1  Area of Concern ........................................................................................................... 7
  3.1.2  The Roles ..................................................................................................................... 8
  3.1.3  Message Sets ............................................................................................................... 8

Chapter 4  Object Type Specifications ...................................................................................... 10

Chapter 5  Abstract Telephone Service export models ............................................................ 11

Appendix 1  Implementation for monitored execution .............................................................. 12