Calling Telephone Service (Tele-A)  
Service Constituent module  
IN-Lab report 93/6

Trygve Reenskaug, Taskon  
Knut Lillegraven, NTR

28 July 1993

The main purpose of this module is to provide a Configuration Service Constituent which may be instantiated and loaded into a User's Invocation Analyzer as a service.

The module defines one possible example design and implementation of the calling side of a telephone communication.

It brings together the functionality defined in the Network module, the Invocation module and the Abstract Telephone Communication module and adds very little new functionality in itself.
Report overview
Chapter 1
Introduction

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

The main purpose of this module is to provide a Configuration Service Constituent which may be instantiated and loaded into a User's Invocation Analyzer as a service.

The functionality of this module is to define the calling (A) side of a telephone connection service.
The models of this module are as follows:

1. **Imported Role Models**
   - *Network Manager (Netw2MngrEXP)* The network manager for access to the Switching Domain.
   - *Network Connection Service (Netw2ConnEXP)* Access to the Switching Domain connection service.
   - *Invocation Environment (Inv2InvEXP)* Mechanism defining the Service Domain invocation environment.
   - *Invocation Environment (Inv2InvFW)* Framework defining the Service Domain invocation environment. (Identical model to the above, but implemented and causes automatic subclassing in its importing models).
   - *Abstract Telephone Interaction EXP (Abstr2IntEXP)* The abstract telephone connection service.

2. **Own Role Models**
   - *Calling Telephone Service User Interaction (TelA2UserRM)*. Defining the minimal interaction between the User object and the calling telephone service object.

3. **Object Type Specifications**
   - *Calling Telephone Service SPC (TelA2SCSPC)*. Object type specification for a calling telephone service.

4. **Exported Role Models**
   - *Connecting telephone service (TelA2SrvCSC)*. Configuration Service Constituent used to define an abstract type for the configuration level.
   - *TelA2 User-Service Interaction (TelA2UserEXP)*. Mechanism specifying the interaction between the calling User object and the Calling Telephone service object.

*Figure 2. Export/import role model structure of the Calling Telephone Service Module.*
Chapter 2
Calling Telephone module export models summary

2.1 Connecting telephone Service Configuration Service Constituent
(TelA2SrvCSC) {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 model is a type specification for the calling end of a telephone connection service.

2.1.2 The Roles

2.2 TelA2 User-Service Interaction EXP (TelA2UserEXP) {Export Model}

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

Defines the interaction between the User object and the Calling Telephone service.

2.2.2 The Roles
Chapter 3
Own Role Models

3.1 Calling Telephone Service User Interaction (TelA2UserRM) {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; Scenarios explanation; Message Sets port explanation, contract explanation, message explanation; FSM

3.1.1 Area of Concern

The minimal interaction between a User and its Calling Telephone Service.

Stimulus messages

1. TelA2User>>connect:from:to:qoc: Requests the creation of a one-to-one telephone connection to an entity addressed in the service Domain.

3.1.2 The Roles

1. TelA2User {Role}. This object represents a Terminal which can initiate, manage and support a full duplex, point-to-point connection in the Switch.

2. TelA2Service (TelA2Srvc) {Role}. This object represents a Service which can initiate, manage and support a full duplex, point-to-point connection in the Switch.
3.1.3 Message Sets

1. **TelA2User** *(Role)*
   - service (s) *(Port)*.
     - TelA2Srvc<TelA2User *(Contract)*.
       - connect: typ from: aTerm to: srvcID qoc: qoc *(Message)*. Requests the creation of a one-to-one telephone connection to an entity addressed in the service Domain.
       - releaseFromA *(Message)*. Release the connection and all associated resources. Terminate the service.

2. **TelA2Service** *(TelA2Srvc) *(Role)*
   - user (u) *(Port)*.
     - TelA2User<TelA2Srvc *(Contract)*.
       - userBWaiting: accessPoint *(Message)*. This message tells the terminal that the service has accepted the call request and that Service B is waiting for connections at a certain access point.
Chapter 4  
Object Type Specification

4.1  Calling Telephone Service SPC (TelA2SCSPC) {Type Specification}

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

4.1.1  Area of Concern

This model is a type specification for the calling end of a telephone connection service.

Stimulus messages

1.  TelA2User>>connect:from:to:qoc: Requests the creation of a one-to-one telephone connection to an entity addressed in the service Domain.
4.1.2 The Roles

[Diagram showing the roles and connections between Inv Mngr, Inv2 A Analzr, Inv2 B Analzr, TelA2 User, TelB2 Srvc, TelA2 Srvc, Netw2 Mngr, Netw2 Conn Point, and Netw2 Conn Leg.]
1. **InvManager (InvMngr) [Type]**. The Invocation Manager's role is to accept requests from Clients and start the Invocation Analyzer (IA) for further processing and to start specific services on request by the IA. The Invocation Analyzer is unique for a given User, and an Invocation Analyzer can be responsible for analyzing requests for many services. The Invocation Manager can be considered a part of the operating system of the network.

The Invocation Manager's role is to accept requests from Clients and start the Invocation Analyzer (IA) for further processing and to start specific services on request by the IA. The Invocation Analyzer is unique for a given User, and an Invocation Analyzer can be responsible for analyzing requests for many services. The Invocation Manager can be considered a part of the operating system of the network.

Setting the status of the Invocation Manager from roles outside this model may be necessary, for example when the network is going down. Then the Manager must be told not to accept any Clients or only a certain set of privileged Clients. Selective rejection of invocation requests must be handled by the invocator. There are many different schemas for rejection of clients and they may depend on the mode of the invocator. Normally all clients are accepted. During major maintenance all clients may be rejected for a short time period (this is not acceptable in a telecommunications network, but theoretically possible); Certain clients are refused; Most clients are refused. Screening functions may be introduced to decide who are acceptable or who rejectable and so on... This can be useful during crisis situations like war etc. Currently this sort of mechanisms are not contained in the invocator role because we assume all clients are accepted.

This globally available object is the first point of entry to the Service Domain.

2. **Inv2AAnalzr [Type]**. The Invocation Analyzer must perform a client-dependent analysis that will identify the particular service requested. This implies that the Invocation Analyzer must analyze the information received from the client, look up relevant information in its internal databases, load the service object and finally initiate it with the required information.

3. **User object AATerm (TelA2User) [Type]**. This object represents a User (or more properly, the user's terminal) which can initiate, manage and support a full duplex, point-to-point telephone connection in the Switch.

4. **TelA2ServiceAServ (TelA2Srvc) [Type]**. This object represents a Service which can initiate, manage and support a full duplex, point-to-point connection in the Switch.

5. **TelB2ServiceB-Serv (TelB2Srvc) [Type]**. 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.

6. **Network Access Manager (Netw2Mngr) [Type]**. This object is a main access point to the Switching domain, giving access to the specialized Switching Domain services.

The service client views the network as a virtual switch object and represents a user of the Connection Control.
7. **Netw2Connection Leg (Netw2ConnLeg) {Type}.** This role represents a communication path from a connection point towards an addressable entity in the network (e.g. an end user terminal). A Leg is always associated with a ConnectionPoint. Leg is responsible for storing the information characterizing the leg.

8. **Netw2Connection Point (Netw2ConnPoint) {Type}.** This role represents an interconnection or logical bridge allowing information to flow between legs.

9. **Inv2BAnalzr {Type}.** The Invocation Analyzer of the called User.

<table>
<thead>
<tr>
<th>TelA2SCSPC</th>
<th>Abstr2IntEXP</th>
<th>Inv2InvFW</th>
<th>TelA2UserRM</th>
<th>Netw2ConnEXP</th>
<th>Netw2MngrEXP</th>
<th>Inv2InvMCH</th>
</tr>
</thead>
<tbody>
<tr>
<td>InvMngr</td>
<td>InvMngr</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>InvMngr</td>
</tr>
<tr>
<td>Inv2AAnalzr</td>
<td>Inv2Analzr</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TelA2User</td>
<td>Inv2Client</td>
<td>TelA2User</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TelA2Srvc</td>
<td>Abstr2AServ</td>
<td>Inv2Serv</td>
<td>TelA2Srvc</td>
<td>Netw2ConnClient</td>
<td>Netw2MngrClient</td>
<td>Inv2Client</td>
</tr>
<tr>
<td>TelB2Srvc</td>
<td>Abstr2BServ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Inv2Serv</td>
</tr>
<tr>
<td>Netw2Mngr</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Netw2Mngr</td>
<td></td>
</tr>
<tr>
<td>Netw2ConnLeg</td>
<td></td>
<td></td>
<td></td>
<td>Netw2ConnLeg</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Netw2ConnPoint</td>
<td></td>
<td></td>
<td></td>
<td>Netw2ConnPoint</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Inv2BAnalzr</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Inv2Analzr</td>
</tr>
</tbody>
</table>

*Table 3. Import relationships for Calling Telephone Service SPC {RMTTypeSpecification}.*
Chapter 5
Models exported from the Calling Telephone Service SC

5.1 Connecting telephone Service Configuration Service Constituent
(TelA2SrvCSC) {Export Model}

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

5.1.1 Area of Concern

This model is a type specification for the calling end of a telephone connection service.

5.1.2 The Roles

1. Inv2Analzr (Type). The Invocation Analyzer must perform a client-dependent analysis that will identify the particular service requested. This implies that the Invocation Analyzer must analyze the information received from the client, look up relevant information in its internal databases, load the service object and finally initiate it with the required information.
2. TelA2ServiceAServ (TelA2Srvc) \{Type\}. This object represents the calling part of a Service which can initiate, manage and support a full duplex, point-to-point connection in the Switch.

5.1.3 Message Sets

1. Inv2Analzr \{Type\}
   - service \(s\) \{Port\).
     - InvServ<InvAnalzr \{Contract\}.
       - setUp: parameters client: aClient analyzer: anAnalyzer invocator: anInvoc \{Message\}. Create and return an instance of a Service with the given parameters (one of them being owner UserID, another being the current Client).
       - kind \{Message\}. Return aSymbol, which represents the kind of service.
       - protocol \{Message\}. Answer the Client protocol permitted by this service.

2. TelA2ServiceAServ (TelA2Srvc) \{Type\}
   - invocationAnalyzer \(iaa\) \{Port\).
     - InvAnalzr<InvServ \{Contract\}.
       - userID \{Message\}. Return the user ID of the user.
       - user \{Message\}. Return the User object of the InvocationAnalyzer.

5.2 TelA2 User-Service Interaction EXP (TelA2UserEXP) \{Export Model\}

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

5.2.1 Area of Concern

Defines the interaction between the User object and the Calling Telephone service.
5.2.2 The Roles

1. **User object AATerm (TelA2User) {Type}**. This object represents a User (or more properly, the user's terminal) which can initiate, manage and support a full duplex, point-to-point telephone connection in the Switch.

2. **TelA2ServiceAServ (TelA2Srvc) {Type}**. This object represents a Service which can initiate, manage and support a full duplex, point-to-point connection in the Switch.

5.2.3 Message Sets

1. **User object AATerm (TelA2User) {Type}**
   - service (s) {Port}.
     - TelA2Srvc<TelA2User {Contract}.
       - connect: typ from: aTerm to: srvcID qoc: qoc {Message}. Requests the creation of a one-to-one telephone connection to an entity addressed in the service Domain.
       - releaseFromA {Message}. Release the connection and all associated resources. Terminate the service.

2. **TelA2ServiceAServ (TelA2Srvc) {Type}**
   - client (c) {Port}.
     - TelA2User<TelA2Srvc {Contract}.
       - userBWaiting: accessPoint {Message}. This message tells the terminal that the service has accepted the call request and that Service B is waiting for connections at a certain access point.
Appendix 1
Programs for monitored execution

App 1.1   Calling Telephone Service SPC (TelA2SCSPC) {Type Specification}

Report including the following selections automatically generated 28 July 1993: Type Specifications; Implementation

App 1.1.1  Calling Telephone Service SPC {Implementation}

Inv2Serv1 subclass: #TelA2Srvc0
   instanceVariableNames: 'bIncovAnalyzer bService connectionPoint legs netwManager'
   classVariableNames:"
   poolDictionaries:"
   category: 'ZZZInLab-TeleA2+

This object represents a Service which can initiate, manage and support a full duplex, point-to-point connection in the Switch.

Instance variables:
   bIncovAnalyzer <RMSObject1>
   bService <RMSObject1>
   connectionPoint <RMSObject1>
   legs <RMSObject1> No Explanation
   netwManager <RMSObject1>

Subclasses should not implement methods for:

Subclasses must implement the following messages:
   InvClient<InvAnalzr
   clientIdRequest
   obeysProtocol:
   TelA2Srvc<TelA2User
   connect:from:to:qoc:
   releaseFromA
   Tel2AServ<Tel2BServ
   provideContext
   releaseFromB
   userBWaiting:

TelA2Srvc0 subclass: #TelA2Srvc1
   instanceVariableNames: 'callType qualityOfConnection aAccessPoint bAccessPoint apID'
classVariableNames: "
poolDictionaries: "
category: 'ZZZInLab-TeleA2+'

TelA2Srvc1 class>>protocol (constants)
protocol
self halt: 'illegal'.
^#TelA

TelA2Srvc1 class>>protocolConst (constants)
protocolConst
^#TelA

TelA2Srvc1 class>>tmInitialize (class initialization)
tmInitialize
Inv2Install1 addService: self protocolConst asClass: self

TelA2Srvc1>>createConnection (private)
createConnection
| legA legB |
connectionPoint := self networkManager
createCp: callType
with: #OnlyNack
for: self guard.
legA := self
createLeg: callType
direction: #None
qoc: qualityOfConnection
toConnPt: connectionPoint.
legA notNil
ifTrue:
[legB := bService
createLeg: callType
direction: #None
qoc: qualityOfConnection
toConnPt: connectionPoint.
legB notNil
ifTrue: [legs := Array with: legA with: legB]
ifFalse: [legs := Array with: legA with: legB]
ifFalse:
[legs := Array new].
legs size = 2
ifTrue:
[self dpsTrace: 'commenceCommunication']
ifFalse:
[invocationAnalyzer user releaseFromB.
bService releaseFromA].

TelA2Srvc1>>networkManager (private)
networkManager
netwManager isNil ifTrue: [netwManager := Netw2MngrServer1 default].
^netwManager

TelA2Srvc1>>kind (InvServ<InvAnalzr)
kind
" Return aSymbol, which represents the kind of service. "
^#TelA
TelA2Svc1>>clientIdRequest (InvClient<InvAnalzr)

clientIdRequest
  ^client isNil
  ifTrue: [invocationAnalyzer userID]
  ifFalse: [client clientIdRequest]

TelA2Svc1>>obeysProtocol: (InvClient<InvAnalzr)

obeysProtocol: aType
  " Return true iff the Client obeys the given protocol. "
  ^#TelB = aType

TelA2Svc1>>apID (access)
apID
  ^apID

TelA2Svc1>>createLeg:direction:qoc:toConnPt: (access)
createLeg: tp direction: dir qoc: qoc toConnPt: conPt
  ^conPt createLeg: tp direction: dir to: self apID qoc: qoc

TelA2Svc1>>initializeFrom: (InvService<InvInstaller)
initializeFrom: aDict
  " 930706 trygve(4.1): IN-Lab Service Contract. "
  apID := aDict at: 'apID'.

TelA2Svc1>>postCopy (copying)
postCopy
  super postCopy.
  bInvocAnalyzer := nil.
  bService := nil.
  connectionPoint := nil.
  legs := OrderedCollection new.
  netwManager := netwManager.
  callType := callType copy.
  qualityOfConnection := qualityOfConnection copy.
  aAccessPoint := aAccessPoint copy.
  bAccessPoint := bAccessPoint copy.
  apID := apID copy.

TelA2Svc1>>provideContext (Tel2AServ<Tel2BServ)
provideContext
  " This message requests service A to provide service B with a context "
  " for the call. "
  bService call: callType qoc: qualityOfConnection.

TelA2Svc1>>releaseFromB (Tel2AServ<Tel2BServ)
releaseFromB
  self halt: 'not implemented'

TelA2Svc1>>userBWaiting: (Tel2AServ<Tel2BServ)
userBWaiting: accessPoint
  " 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. "
  bAccessPoint := accessPoint.

TelA2Svc1>>connect:from:to:qoc: (TelA2Svc<TelA2User)
connect: typ from: aTerm to: srvcID qoc: qoc
  callType := typ.
qualityOfConnection := qoc.
accessPoint := invocationAnalyzer user.
blInvocAnalyzer := invocationManager invocReq: srvcID.
bservice := blInvocAnalyzer getService: #TelB for: self.
bservice open.
self createConnection.

TelA2Srv1>>releaseFromA (TelA2Srv<TelA2User)
releaseFromA
  self halt: 'not implemented'
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Chapter 1</strong></td>
<td>Introduction</td>
<td>2</td>
</tr>
<tr>
<td><strong>Chapter 2</strong></td>
<td>Calling Telephone module export models summary</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>2.1 Connecting telephone Service Configuration Service Constituent</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>(TelA2SrvCSC) {Export Model}</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>2.1.1 Area of Concern</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>2.1.2 The Roles</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>2.2 TelA2 User-Service Interaction EXP (TelA2UserEXP) {Export Model}</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>2.2.1 Area of Concern</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>2.2.2 The Roles</td>
<td>5</td>
</tr>
<tr>
<td><strong>Chapter 3</strong></td>
<td>Own Role Models</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>3.1 Calling Telephone Service User Interaction (TelA2UserRM) {Role Model}</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>3.1.1 Area of Concern</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>3.1.2 The Roles</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>3.1.3 Message Sets</td>
<td>7</td>
</tr>
<tr>
<td><strong>Chapter 4</strong></td>
<td>Object Type Specification</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>4.1 Calling Telephone Service SPC (TelA2SCSPC) {Type Specification}</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>4.1.1 Area of Concern</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>4.1.2 The Roles</td>
<td>9</td>
</tr>
<tr>
<td><strong>Chapter 5</strong></td>
<td>Models exported from the Calling Telephone Service</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>5.1 Connecting telephone Service Configuration Service Constituent</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>(TelA2SrvCSC) {Export Model}</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>5.1.1 Area of Concern</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>5.1.2 The Roles</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>5.1.3 Message Sets</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>5.2 TelA2 User-Service Interaction EXP (TelA2UserEXP) {Export Model}</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>5.2.1 Area of Concern</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>5.2.2 The Roles</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>5.2.3 Message Sets</td>
<td>14</td>
</tr>
<tr>
<td><strong>Appendix 1</strong></td>
<td>Programs for monitored execution</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>App 1.1 Calling Telephone Service SPC (TelA2SCSPC) {Type Specification}</td>
<td>15</td>
</tr>
</tbody>
</table>
INDEX

TelA2Srvc1 class>>protocol (constants) ......................................................... 16
TelA2Srvc1 class>>protocolConst (constants) .................................................. 16
TelA2Srvc1 class>>tmInitialize (class initialization) ......................................... 16
TelA2Srvc1>>apID (access) ............................................................................ 17
TelA2Srvc1>>clientIdRequest (InvClient<InvAnalzr) ........................................... 17
TelA2Srvc1>>connect:from:to:qoc: (TelA2Srvc<TelA2User) ............................. 17
TelA2Srvc1>>createConnection (private) ........................................................ 16
TelA2Srvc1>>createLeg:direction:qoc:toConnPt: (access) .............................. 17
TelA2Srvc1>>initializeFrom: (InvService<InvInstaller) ..................................... 17
TelA2Srvc1>>kind (InvServ<InvAnalzr) ............................................................. 16
TelA2Srvc1>>networkManager (private) .......................................................... 16
TelA2Srvc1>>obeysProtocol: (InvClient<InvAnalzr) .......................................... 17
TelA2Srvc1>>postCopy (copying) .................................................................. 17
TelA2Srvc1>>provideContext (Tel2AServ<Tel2BServ) ....................................... 17
TelA2Srvc1>>releaseFromA (TelA2Srvc<TelA2User) ......................................... 18
TelA2Srvc1>>releaseFromB (Tel2AServ<Tel2BServ) .......................................... 17
TelA2Srvc1>>userBWaiting: (Tel2AServ<Tel2BServ) ........................................ 17