The purpose of this module is to specify a terminal which may represent a User in the context of service invocation.
Report overview
Chapter 1
Introduction

This document describes the User Terminal OOram Module on the service sub-layer of the Service Constituent Creator Layer.

The module is a skeleton module which defines the minimum terminal functionality needed to perform a telephone connection invocation.
Chapter 2
Overview of the User Terminal Module

This skeleton module only contains a role model representing the user interface and an object type specification specifying the example terminal as seen from the Service Domain (the internals of the terminal are not shown in this example).

Figure 2. Model structure of the User Terminal Module.
Chapter 3
Models Exported from the User Terminal Module

There are no export models in this module.
Chapter 4
Role Models of the User Terminal Module

There is one role model in this module which defines how a telephone connect request is initiated.

4.1 The terminal user interface (Tel2UserTerm) {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

4.1.1 Area of Concern

This role model is a minimal representation of the terminal user interface.

Stimulus messages
1. Term2Client>>telephoneConnectTo:

4.1.2 The Roles

1. Term2Client [Role]. The client object will probably be some object in the User's terminal which is associated with the user interface.

2. Term2User [Role]. This object represents an object in the user's terminal which can initiate, manage and support a full duplex, point-to-point telephone connection in the Switch.
4.1.3 Message Sets

1. Term2Client \{Role\}
   - user \((u)\) \{Port\}.
     - Term2User\(<\)Term2Client \{Contract\}.
       - telephoneConnectTo: calledUserID \{Message\}. 
Chapter 5
Object type specifications in the User Terminal Module

5.1 TelA2 User-Service Interaction SPC (TelA2UserSPC) {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

5.1.1 Area of Concern

Defines the interaction between the User object and the Calling and Called Telephone services.


5.1.2 The Roles

1. **User Terminal objectATerm (Term2User) (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.

   This object represents a User (e.g. part of a Terminal) which can accept or refuse a request for a full duplex, point-to-point connection in the Switch.

   This object represents any client of the Invocation Manager, an Analyzer and possibly one of the Analyzer's Services.

   This object represents an object in the user's terminal which can initiate, manage and support a full duplex, point-to-point telephone connection in the Switch.

2. **Term2ServiceAServ (Term2Srvc) (Type)**. This object represents a Service which can initiate, manage and support a full duplex, point-to-point connection in the Switch.

   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.

   The Service role represents the target application for the client. Any service to be invocated by this model must have an object which plays this role.
3. **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.

4. **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.

5. **Term2Client \{Type\}**. The client object will probably be some object in the User’s terminal which is associated with the user interface.

<table>
<thead>
<tr>
<th>TelA2UserSPC</th>
<th>TelA2UserEXP</th>
<th>TelB2UserEXP</th>
<th>Inv2InvFW</th>
<th>Tel2UserTerm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Term2User</td>
<td>TelA2User</td>
<td>TelB2User</td>
<td>Inv2Client</td>
<td>Term2User</td>
</tr>
<tr>
<td>Term2Srvc</td>
<td>TelA2Srvc</td>
<td>TelB2Srvc</td>
<td>Inv2Serv</td>
<td></td>
</tr>
<tr>
<td>InvMngr</td>
<td></td>
<td></td>
<td>InvMngr</td>
<td></td>
</tr>
<tr>
<td>Inv2Analzr</td>
<td></td>
<td></td>
<td>Inv2Analzr</td>
<td></td>
</tr>
<tr>
<td>Term2Client</td>
<td></td>
<td></td>
<td></td>
<td>Term2Client</td>
</tr>
</tbody>
</table>

*Table 3. Import relationships for TelA2 User-Service Interaction SPC \{RMTTypeSpecification\}.*
Chapter 6
Monitored executions

There are no Monitored executions in this module.
Appendix 1
Implementation for monitored execution

App 1.1  TelA2 User-Service Interaction SPC (TelA2UserSPC) {Type Specification}

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

App 1.1.1  TelA2 User-Service Interaction SPC {Implementation}

Inv2Client1 subclass: #Term2User0
  instanceVariableNames: ""
  classVariableNames: ""
  poolDictionaries: ""
  category: 'ZZInLab-User2+'

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. This object represents a Terminal which can accept or refuse a request for a full duplex, point-to-point connection in the Switch. This object represents any client of the Invocation Manager, an Analyzer and possibly one of the Analyzer's Services. An object representing a User in the service Domain.

Instance variables:

Subclasses should not implement methods for:

Subclasses must implement the following messages:
  TelA2User<TelA2Srvc
    userBWaiting:
    TelB2User<TelB2Service
      openRequest

Term2User0 subclass: #Term2User1
  instanceVariableNames: ""
  classVariableNames: ""
  poolDictionaries: ""
  category: 'ZZInLab-User2+'

Term2User1 class>>initialize (class initialization)
initialize
  ownProtocol := IdentityDictionary new
  at: #telephoneConnectTo: put: #(Inv2Install1Class); yourself.
Term2User1>>obeysProtocol: (InvClient<InvAnalzr)
obeysProtocol: aType
  " Return true iff the Client obeys the given protocol. "
^#TelA = aType

Term2User1>>connectTo: (examples)
connectTo: toSerId
  service := invocAnalyzer getService: #TelA for: self guard.
  service connect: #Duplex from: self guard to: toSerId qoc: 2400.

Term2User1>>example4ConnectTo: (examples)
example4ConnectTo: toSerId
  service := invocAnalyzer getService: #TelA for: self guard.
  service connect: #Duplex from: self guard to: toSerId qoc: 2400.

Term2User1>>telephoneConnectTo: (examples)
televisionConnectTo: calledUserID
  service := invocAnalyzer getService: #TelA for: self guard.
  service connect: #Duplex from: self guard to: calledUserID qoc: 2400.

Term2User1>>userBWaiting: (TelA2User<TelA2Srvc)
userBWaiting: accessPoint
  self halt: 'not implemented'

Term2User1>>openRequest (TelB2User<TelB2Service)
openRequest
  self dpsError: 'not implemented'.
  ^true
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chapter 1</td>
<td>Introduction</td>
<td>2</td>
</tr>
<tr>
<td>Chapter 2</td>
<td>Overview of the User Terminal Module</td>
<td>3</td>
</tr>
<tr>
<td>Chapter 3</td>
<td>Models Exported from the User Terminal Module</td>
<td>4</td>
</tr>
<tr>
<td>Chapter 4</td>
<td>Role Models of the User Terminal Module</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>4.1 The terminal user interface (Tel2UserTerm) {Role Model}</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>4.1.1 Area of Concern</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>4.1.2 The Roles</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>4.1.3 Message Sets</td>
<td>6</td>
</tr>
<tr>
<td>Chapter 5</td>
<td>Object type specifications in the User Terminal Module</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>5.1 TelA2 User-Service Interaction SPC (TelA2UserSPC) {Type Specification}</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>5.1.1 Area of Concern</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>5.1.2 The Roles</td>
<td>8</td>
</tr>
<tr>
<td>Chapter 6</td>
<td>Monitored executions</td>
<td>10</td>
</tr>
<tr>
<td>Appendix 1</td>
<td>Implementation for monitored execution</td>
<td>11</td>
</tr>
<tr>
<td>App 1.1</td>
<td>TelA2 User-Service Interaction SPC (TelA2UserSPC) {Type Specification}</td>
<td>11</td>
</tr>
<tr>
<td>App 1.1.1</td>
<td>TelA2 User-Service Interaction SPC {Implementation}</td>
<td>11</td>
</tr>
</tbody>
</table>
INDEX

Term2User1 class>>initialize (class initialization) .................................................. 11
Term2User1>>connectTo: (examples) ......................................................................... 12
Term2User1>>example4ConnectTo: (examples) ........................................................... 12
Term2User1>>obeysProtocol: (InvClient<InvAnalzr) ................................................... 12
Term2User1>>openRequest (TelB2User<TelB2Service) .............................................. 12
Term2User1>>telephoneConnectTo: (examples) ........................................................ 12
Term2User1>>userBWaiting: (TelA2User<TelA2Srvc) ................................................ 12