Standard ECMA-282

December 1998

ECMA

Standardizing Information and Communication Systems

Private Integrated Services Network (PISN) -
Inter-Exchange Signalling Protocol -
Private User Mobility (PUM) -
Registration Supplementary Service

ECMA

Standardizing Information and Communication Systems

Private Integrated Services Network (PISN) -
Inter-Exchange Signalling Protocol -
Private User Mobility (PUM) -
Registration Supplementary Service

(QSIG-PUMR)

.

Brief History

This Standard is one of a series of ECMA Standards defining services and signalling protocols applicable to Private Integrated Services Networks (PISNs). The series uses ISDN concepts as developed by ITU-T and conforms to the framework of International Standards for Open Systems Interconnection as defined by ISO/IEC. It has been produced under ETSI work item DEN/ECMA-00152.

This particular Standard specifies the signalling protocol for use at the Q reference point in support of the Private User Mobility Registration supplementary service.

This Standard is based upon the practical experience of ECMA member companies and the results of their active and continuous participation in the work of ISO/IEC JTC1, ITU-T, ETSI and other international and national standardization bodies. It represents a pragmatic and widely based consensus.

This ECMA Standard is contributed to ISO/IEC JTC1 under the terms of the fast-track procedure, for adoption as an ISO/IEC International Standard.

This Standard has been adopted by the ECMA General Assembly of December 1998.

.

Table of contents

.

1 Scope

This Standard specifies the signalling protocol for the support of the Private User Mobility Registration supplementary service (SS-PUMR) at the Q reference point between Private Integrated Services Network Exchanges (PINXs) connected together within a Private Integrated Services Network (PISN).

SS-PUMR is a supplementary service which enables a PUM user to register at, or de-register from, any wired or cordless terminal within the PISN. The ability to register at different locations in the PISN at different times enables the PUM user to maintain the provided services (including the ability to make and receive calls) at different access points. De-registration is used to inform the PISN that the PUM user is temporarily unable to make use of the provided services (including the receipt of calls). SS-PUMR may provide the PUM user with the ability to obtain information on the current registration sessions (interrogation).

The Q reference point is defined in ECMA-133.

Service specifications are produced in three stages and according to the method specified in ETS 300 387. This Standard contains the stage 3 specification for the Q reference point and satisfies the requirements identified by the stage 1 and stage 2 specifications in ECMA-281.

The signalling protocol for SS-PUMR uses certain aspects of the generic procedures for the control of supplementary services specified in ECMA-165.

This Standard also specifies additional signalling protocol requirements for the support of interactions at the Q reference point between SS-PUMR and other supplementary services and ANFs.

NOTE 1

Additional interactions that have no impact on the signalling protocol at the Q reference point can be found in the relevant stage 1 specifications.

This Standard is applicable to PINXs which can interconnect to form a PISN.

This Standard applies to PUM only within a single PISN.

2 Conformance

In order to conform to this Standard, a PINX shall satisfy the requirements identified in the Protocol Implementation Conformance Statement (PICS) proforma in annex A.

Conformance to this Standard includes conforming to those clauses that specify protocol interactions between SS-PUMR and other supplementary services and ANFs for which signalling protocols at the Q reference point are supported in accordance with the stage 3 standards concerned.

3 References (normative)

The following standards contain provisions which, through reference in this text, constitute provisions of this Standard. All standards are subject to revision, and parties to agreements based on this Standard are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below.

In the case of references to ECMA Standards that are aligned with ISO/IEC International Standards, the number of the appropriate ISO/IEC International Standard is given in brackets after the ECMA reference.

ECMA-133	Private Integrated Services Network - Reference Configuration for PISN Exchanges (PINX) (International Standard ISO/IEC 11579-1)
ECMA-165	Private Integrated Services Network (PISN) - Generic Functional Protocol for the Support of Supplementary Services - Inter-Exchange Signalling Procedures and Protocol (International Standard ISO/IEC 11582)
ECMA-216	Private Integrated Services Network (PISN) - Cordless Terminal Mobility (CTM) - Inter-Exchange Signalling Protocol - Cordless Terminal Location Registration Supplementary Service
ECMA-242	Private Integrated Services Network (PISN) - Inter-Exchange Signalling Protocol - Message Waiting Indication Supplementary Service (International Standard ISO/IEC 15506)

ECMA-281	Private Integrated Services Network (PISN) - Specification, Functional Model and Information Flows - Private User Mobility (PUM) - Registration Supplementary Service
ISO/IEC 11571	Information technology - Telecommunications and information exchange between systems - Numbering and sub-addressing in private integrated services networks (1994)
ETS 300 387	Private Telecommunication Network (PTN); Method for the specification of basic and supplementary services (1994)
ETS 300 692	Private Integrated Services Network (PISN); Cordless Terminal Mobility (CTM); Location handling services; Functional capabilities and information flows (1996)
ITU-T Rec. I.112	Vocabulary of terms for ISDN (1993)
ITU-T Rec. I.210	Principles of telecommunication services supported by an ISDN and the means to describe them (1993)
ITU-T Rec. Q.950	Digital Subscriber Signalling System No. 1 (DSS 1) - Supplementary services protocols, structure and general principles (1997)
ITU-T Rec. Z.100	Specification and description language (1993)

4 Definitions

For the purpose of this Standard the following definitions apply.

4.1 External definitions

This Standard uses the following terms defined in other documents:

– AllCall registration	(ECMA-281)
– Application Protocol Data Unit (APDU)	(ECMA-165)
– Basic Service	(ITU-T Rec. I.210)
– Call, Basic Call	(ECMA-165)
– Complete Number	(ISO/IEC 11571)
– Coordination Function	(ECMA-165)
– De-registration	(ECMA-281)
– Directory PINX	(ETS 300 692)
– End PINX	(ECMA-165)
– Gateway PINX	(ECMA-165)
– Home data base (HDB)	(ECMA-281)
– Home PINX	(ECMA-281)
– Hosting address	(ECMA-281)
– InCall registration	(ECMA-281)
– Network Facility Extension (NFE)	(ECMA-165)
– PISN Number	(ISO/IEC 11571)
– Originating PINX	(ECMA-165)
– OutCall registration	(ECMA-281)
– Private Integrated Services Network (PISN)	(ECMA-133)
– Private Integrated Services Network Exchange (PINX)	(ECMA-133)
– Private User Mobility (PUM)	(ECMA-281)

– PUM number	(ECMA-281)
– PUM user	(ECMA-281)
– Registration	(ECMA-281)
– Registration session	(ECMA-281)
– Signalling	(ITU-T Rec. I.112)
– Supplementary Service	(ITU-T Rec. I.210)
– Supplementary Service Control Entity	(ECMA-165)
– Terminating PINX	(ECMA-165)
– Transit PINX	(ECMA-165)
– Visitor data base (VDB)	(ECMA-281)
– Visitor PINX	(ECMA-281)

4.2   Remote PINX

Any PINX in the private network not coincident with the Visitor PINX, from which a PUM user or another authorized user has access to the features (registration/de-registration and interrogation) of SS-PUMR.

4.3   Previous Visitor PINX

A PINX in the private network, at which a PUM user is still registered and has to be de-registered from when he attempts to register at another Visitor PINX.

4.4   Alternative identifier

An identifier, other than the PISN number, which identifies the PUM user uniquely.

4.5   Activating user

Either the PUM user himself or an authorized user acting on behalf of the PUM user.

5   List of acronyms

ANF	Additional Network Feature
APDU	Application Protocol Data Unit
ASN.1	Abstract Syntax Notation no.1
CT	Cordless Terminal
CTM	Cordless Terminal Mobility
HDB	Home Data Base
ISDN	Integrated Services Digital Network
MWI	Message Waiting Indication
NFE	Network Facility Extension
PICS	Protocol Implementation Conformance Statement
PINX	Private Integrated Services Network Exchange
PISN	Private Integrated Services Network
PUM	Private User Mobility
SDL	Specification and Description Language
SS	Supplementary Service
SS-PUMR	Private User Mobility Registration supplementary service

VDB   Visitor Data Base

6 Signalling protocol for the support of SS-PUMR

6.1 SS-PUMR description

SS-PUMR is a supplementary service which enables a PUM user to register at, or de-register from, any wired or cordless terminal within the PISN. The ability to register for incoming calls (InCall registration) is mandatory, whereas the ability to register for outgoing calls (OutCall registration) or for both incoming and outgoing calls (AllCall registration) is optional. The ability to register at different locations in the PISN at different times enables the PUM user to maintain the provided services (including the ability to make and receive calls) at different access points. De-registration is used to inform the PISN that the PUM user is temporarily unable to make use of the provided services (including the receipt of calls). SS-PUMR may provide the PUM user with the ability to obtain information on the current registration sessions (interrogation).

6.2 SS-PUMR operational requirements

6.2.1 Requirements on the Visitor PINX

Generic procedures for the call independent control (connection oriented) of supplementary services, as specified in ECMA-165 for an Originating and Terminating PINX, shall apply.

6.2.2 Requirements on the Home PINX

Generic procedures for the call independent control (connection oriented) of supplementary services, as specified in ECMA-165 for a Terminating and an Originating PINX, shall apply.

6.2.3 Requirements on the Previous Visitor PINX

Generic procedures for the call independent control (connection oriented) of supplementary services, as specified in ECMA-165 for a Terminating PINX, shall apply.

6.2.4 Requirements on a Transit PINX

Generic procedures for the call independent control (connection oriented) of supplementary services, as specified in ECMA-165 for a Transit PINX, shall apply.

6.2.5 Requirements on the Directory PINX

Generic procedures for the call independent control (connection oriented) of supplementary services, as specified in ECMA-165 for a Terminating PINX, shall apply.

6.2.6 Requirements on the Remote PINX

Generic procedures for the call independent control (connection oriented) of supplementary services, as specified in ECMA-165 for an Originating PINX, shall apply.

6.3   SS-PUMR coding requirements

6.3.1   Operations

The operations defined in Abstract Syntax Notation number 1 (ASN.1) in table 1 shall apply.

Table 1 - Operations in Support of SS-PUMR

```
PUM-Registration-Operations
                { iso (1) identified-organization (3) icd-ecma (0012) standard (0)
                qsig-pum-registration (282) pum-registration-operations (0) }


DEFINITIONS EXPLICIT TAGS ::=


BEGIN

IMPORTS    OPERATION, ERROR FROM Remote-Operation-Notation
                { joint-iso-ccitt (2) remote-operations (4) notation (0) }

           Extension FROM Manufacturer-specific-extension-definition
                { iso (1) standard (0) pss1-generic-procedures (11582) msi-definition (0) }

           notAvailable, invalidServedUserNumber, supplementaryServiceInteractionNotAllowed
                FROM General-Error-List
                { ccitt recommendation q 950 general-error-list (1) }
                -- Note: The definition of these errors is reproduced in annex D

           PartyNumber FROM Addressing-Data-Elements
                { iso (1) standard (0) pss1-generic-procedures (11582)
                  addressing-data-elements (6) }
                -- Note: The definition of PartyNumber is reproduced in annex D

           BasicService FROM Basic-Service-Elements
                { iso (1) standard (0) pss1-call-diversion (13873) call-diversion-operations (0) }
                -- Note: The definition of BasicService is reproduced in annex D

           pisnEnquiry FROM CTM-Location-Registration-Operations
                { iso (1) identified-organization (3) icd-ecma (0012) standard (0)
                  qsig-location-registration (216) ctlr-operations (0) };
                -- Note: The definition of pisnEnquiry is reproduced in annex D

PumRegistr ::=     OPERATION
           -- Registration (sent from the Visitor PINX to the Home PINX or
           --                      from a Remote PINX to the Visitor PINX)
                ARGUMENT      PumRegistrArg
                RESULT        PumRegistrRes
                ERRORS        { invalidServedUserNumber, notAuthorized, unspecified,
                              notAvailable, temporarilyUnavailable,
                              supplementaryServiceInteractionNotAllowed,
                              pumUserNotSubscribedToThisServiceOpt,
                              pumUserFailedAuthentication, hostingAddrInvalid }

PumDelReg ::=     OPERATION
           -- Delete Registration (sent from the Home PINX to the Previous Visitor PINX)
                ARGUMENT      PumDelRegArg
                RESULT        DummyRes
                ERRORS        { notAvailable, temporarilyUnavailable, unspecified,
                              supplementaryServiceInteractionNotAllowed }
```

```
PumDe-reg ::=    OPERATION
        -- De-registration (sent from the Visitor PINX or Remote PINX to the Home PINX)
                ARGUMENT    PumDe-regArg
                RESULT      DummyRes
                ERRORS      { invalidServedUserNumber, notAuthorized, unspecified,
                            supplementaryServiceInteractionNotAllowed,
                            pumUserNotSubscribedToThisServiceOpt,
                            pumUserFailedAuthentication, hostingAddrInvalid,
                            pumUserNotRegistered }
PumInterrog ::=    OPERATION
        -- Interrogation (sent from the Visitor PINX or Remote PINX to the Home PINX and
        --                     from the Home PINX to the Visitor PINX)
                ARGUMENT    PumInterrogArg
                RESULT      PumInterrogRes
                ERRORS      { invalidServedUserNumber, notAuthorized, unspecified,
                            supplementaryServiceInteractionNotAllowed,
                            pumUserFailedAuthentication, hostingAddrInvalid,
                            pumUserNotRegistered }
PumRegistrArg ::= SEQUENCE   { pumUserId  CHOICE { pumNumber   PartyNumber,
                                                -- The PISN number of the PUM user,
                                                -- always a Complete Number.
                                                alternativeId    AlternativeId },
                            basicService    BasicService,
                                -- specific basic service or all basic services,
                            hostingAddr    PartyNumber,
                                -- The PISN number of the hosting user,
                                -- always a Complete Number.
                            activatingUserAddr    [0]    PartyNumber    OPTIONAL,
                                -- The PISN number of the activating user,
                                -- always a Complete Number.
                                -- Mandatory if sent from a Remote PINX, else not included.
                            serviceOption   ServiceOption  DEFAULT inCallRegistration,
                                -- Type of registration (InCall, OutCall or AllCall)
                            sessionParams SessionParams OPTIONAL,
                                -- Duration of session, Number of outgoing calls
                            userPin  CHOICE  { pumUserPin [6] IMPLICIT  UserPin,
                                activatingUserPin [7] IMPLICIT   UserPin } OPTIONAL,
                            argExtension    PumrExtension         OPTIONAL }
PumRegistrRes ::= SEQUENCE   { pumNumber   PartyNumber,
                            serviceOption   ServiceOption  OPTIONAL,
                                -- Type of registration (InCall, OutCall or AllCall)
                            sessionParams SessionParams OPTIONAL,
                                -- Duration of session, Number of outgoing calls
                            argExtension    PumrExtension         OPTIONAL }
DummyRes ::=     CHOICE       { null          NULL,
                            extension      [ 1 ] IMPLICIT Extension,
                            sequOfExtn     [ 2 ] IMPLICIT SEQUENCE OF Extension }
```

```
PumDelRegArg ::= SEQUENCE    { pumUserId  CHOICE { pumNumber    PartyNumber,
                                            -- The PISN number of the PUM user,
                                            -- always a Complete Number.
                                            alternativeId    AlternativeId },
                             basicService    BasicService,
                                 -- specific basic service or all basic services,
                             hostingAddr    PartyNumber,
                                 -- The PISN number of the hosting user,
                                 -- always a Complete Number.
                             serviceOption    ServiceOption,
                                 -- Type of registration session (InCall, OutCall or AllCall)
                             argExtension    PumrExtension        OPTIONAL }

PumDe-regArg ::= SEQUENCE    { pumUserId  CHOICE { pumNumber    PartyNumber,
                                            -- The PISN number of the PUM user,
                                            -- always a Complete Number.
                                            alternativeId    AlternativeId },
                             basicService    BasicService,
                                 -- specific basic service or all basic services,
                             hostingAddr    [0]    PartyNumber    OPTIONAL,
                                 -- The PISN number of the hosting user,
                                 -- always a Complete Number.
                                 -- Not included if serviceOption indicates 'inCallRegistration',
                                 -- optional if serviceOption indicates 'outCallRegistration'
                                 -- or 'allCallRegistration'.
                             activatingUserAddr    [1]    PartyNumber    OPTIONAL,
                                 -- The PISN number of the activating user,
                                 -- always a Complete Number.
                                 -- Mandatory if sent from a Remote PINX, else not included.
                             serviceOption    ServiceOption DEFAULT inCallRegistration,
                                 -- Type of registration session (InCall, OutCall or AllCall)
                                 -- If serviceOption indicates 'outCallRegistration' and
                                 -- hostingAddr is omitted, the de-registration applies to
                                 -- all OutCall registrations of this PUM user.
                                 -- If serviceOption indicates 'allCallRegistration' and
                                 -- hostingAddr is omitted, the de-registration applies to the
                                 -- AllCall and all OutCall registrations of this PUM user.
                             userPin  CHOICE   { pumUserPin [6] IMPLICIT  UserPin,
                                 activatingUserPin [7] IMPLICIT    UserPin } OPTIONAL,
                             argExtension    PumrExtension        OPTIONAL }

PumInterrogArg ::= SEQUENCE    { pumUserId  CHOICE { pumNumber    PartyNumber,
                                            -- The PISN number of the PUM user,
                                            -- always a Complete Number.
                                            alternativeId    AlternativeId },
                             basicService    BasicService,
                                 -- specific basic service or all basic services,
                             hostingAddr    [0]    PartyNumber    OPTIONAL,
                                 -- The PISN number of the hosting user,
                                 -- always a Complete Number.
                                 -- Omission indicates 'all hosting addresses'.
                             activatingUserAddr    [1]    PartyNumber    OPTIONAL,
                                 -- The PISN number of the activating user,
                                 -- always a Complete Number.
                             homeInfoOnly    BOOLEAN      DEFAULT TRUE,
                                 -- True = Only Home PINX information (default)
                                 -- False = Complete information
```

```
                                      userPin  CHOICE  { pumUserPin [6] IMPLICIT  UserPin,
                                           activatingUserPin [7] IMPLICIT    UserPin } OPTIONAL,
                                      argExtension    PumrExtension        OPTIONAL }
PumInterrogRes ::= SET SIZE(1..8) OF
                    SEQUENCE { basicService  [0] IMPLICIT  BasicService     OPTIONAL,
                                  -- specific basic service or all basic services,
                                  -- (Home PINX information)
                              hostingAddr     [1]     PartyNumber   OPTIONAL,
                                  -- The PISN number of the hosting user,
                                  -- always a Complete Number.
                                  -- (Home PINX information)
                              serviceOption  [2] IMPLICIT  ServiceOption     OPTIONAL,
                                  -- Type of registration session
                                  -- (InCall, OutCall or AllCall)
                                  -- (Home PINX information)
                              interrogParams         SessionParams         OPTIONAL,
                                  -- Time left in registration session,
                                  -- Number of outgoing calls left
                                  -- (Visitor PINX information)
                              argExtension    PumrExtension        OPTIONAL }
AlternativeId ::= OCTET STRING (SIZE(1..20))
ServiceOption ::=  ENUMERATED  { inCallRegistration        (0),
                                 outCallRegistration       (1),
                                 allCallRegistration       (2) }
SessionParams ::= SEQUENCE   { durationOfSession    [ 1 ] IMPLICIT INTEGER  OPTIONAL,
                                  -- Duration of session in seconds,
                                  -- default if omitted: duration of session unlimited.
                              numberOfOutgCalls    [ 2 ] IMPLICIT INTEGER  OPTIONAL }
                                  -- Default if omitted: number of outgoing calls unlimited.
UserPin       ::=     OCTET STRING (SIZE(1..20))
PumrExtension       ::=             CHOICE       {
                                    extension    [ 4 ] IMPLICIT Extension,
                                    sequOfExtn   [ 5 ] IMPLICIT SEQUENCE OF Extension }


pumRegistr                   PumRegistr    ::= localValue 89
pumDelReg                    PumDelReg     ::= localValue 90
pumDe-reg                    PumDe-reg     ::= localValue 91
pumInterrog                  PumInterrog   ::= localValue 92


notAuthorized                ERROR        ::= localValue 1007
temporarilyUnavailable       ERROR        ::= localValue 1000
unspecified                  Unspecified  ::= localValue 1008
Unspecified ::=              ERROR        PARAMETER Extension
pumUserNotSubscribedToThisServiceOpt ERROR ::= localValue 1019
pumUserFailedAuthentication          ERROR ::= localValue 1020
hostingAddrInvalid                   ERROR ::= localValue 1021
pumUserNotRegistered                 ERROR ::= localValue 1022


END          -- of PUM-Registration-Operations
```

6.3.2 Information elements

6.3.2.1 Facility information element

The operations defined in 6.3.1 shall be coded in the Facility information element in accordance with ECMA-165.

When conveying the invoke APDU of operations defined in 6.3.1, the destinationEntity data element of the NFE shall contain value endPINX.

When conveying the invoke APDU of operations defined in 6.3.1, the Interpretation APDU shall either be omitted or be included with value rejectAnyUnrecognisedInvokePdu.

6.3.2.2 Other information elements

Any other information elements (e.g. Calling party number, Called party number) shall be coded in accordance with the rules of ECMA-165.

6.3.3 Messages

The Facility information element shall be conveyed in the messages as specified in clause 10 of ECMA-165.

6.4 SS-PUMR state definitions

6.4.1 States at the Visitor PINX

The procedures for the Visitor PINX are written in terms of the following conceptual states existing within the SS-PUMR Supplementary Service Control entity in that PINX in association with a particular registration request.

6.4.1.1 State VisitIdle

SS-PUMR is not operating.

6.4.1.2 State VisitReg

A pumRegistr invoke APDU has been sent.

6.4.1.3 State VisitEnquiry

A pisnEnquiry invoke APDU has been sent.

6.4.1.4 State VisitInterrogation

A pumInterrog invoke APDU has been sent.

6.4.1.5 State VisitDe-reg

A pumDe-reg invoke APDU has been sent.

6.4.2 States at the Home PINX

The procedures for the Home PINX are written in terms of the following conceptual states existing within the SS-PUMR Supplementary Service Control entity in that PINX in association with a particular registration request.

6.4.2.1 State HomeIdle

SS-PUMR is not operating. Ready for receipt of a pumRegistr or pumDe-reg invoke APDU.

6.4.2.2 State HomeDelete

A pumDelReg invoke APDU has been sent.

6.4.2.3 State HomeInterrogation

A pumInterrog invoke APDU has been sent.

6.4.3 States at the Directory PINX

The procedures for the Directory PINX are written in terms of the following conceptual states existing within the SS-PUMR Supplementary Service Control entity in that PINX in association with a particular registration request.

6.4.3.1 State DirectoryIdle

Ready for receipt of a pisnEnquiry invoke APDU.

6.4.4 **States at the Previous Visitor PINX**

The procedures for the Previous Visitor PINX are written in terms of the following conceptual states existing within the SS-PUMR Supplementary Service Control entity in that PINX in association with a particular registration request.

6.4.4.1 **State PreVisitorIdle**

Ready for receipt of a pumDelReg invoke APDU.

6.4.5 **States at the Remote PINX**

The procedures for a Remote PINX are written in terms of the following conceptual states existing within the SS-PUMR Supplementary Service Control entity in that PINX in association with a particular registration request.

6.4.5.1 **State RemoteIdle**

SS-PUMR is not operating.

6.4.5.2 **State RemoteReg**

A pumRegistr invoke APDU has been sent.

6.4.5.3 **State RemoteEnquiry**

A pisnEnquiry invoke APDU has been sent.

6.4.5.4 **State RemoteInterrogation**

A pumInterrog invoke APDU has been sent.

6.4.5.5 **State RemoteDe-reg**

A pumDe-reg invoke APDU has been sent.

6.5 **SS-PUMR signalling procedures for registration**

Examples of message sequences are shown in annex B.

6.5.1 **Actions at the Visitor PINX for registration**

The SDL representation of procedures at the Visitor PINX is shown in C.1 of annex C.

6.5.1.1 **Normal procedures**

On receipt of a valid registration request from a PUM user and if the Visitor PINX possesses sufficient addressing information to make a registration request to the Home PINX, the Visitor PINX shall send a pumRegistr invoke APDU to the Home PINX using the call reference of a call independent signalling connection. The call independent signalling connection shall be established (or used, if an appropriate connection is already available) in accordance with the procedures specified in 7.3 of ECMA-165. Within the argument of the invoke APDU, the PISN number or an alternative identifier of the PUM user shall be included in element pumUserId, the basic service for which the PUM user is to be registered shall be included in element basicService, and the PISN number of the hosting user shall be included in element hostingAddr. This PISN number shall be suitable for use in the Called party number information element of a call or call independent signalling connection that is to be routed to the hosting user. If provided by the registration request from the PUM user, the Visitor PINX shall additionally include in element serviceOption the type of registration (InCall, OutCall, or AllCall). If provided by the registration request from the PUM user, the Visitor PINX shall further include the PUM user's PIN in element userPin. The Visitor PINX shall enter state VisitReg and start timer T1.

The registration request from a PUM user may optionally contain session parameters (element sessionParams) specifying the maximum duration of the registration session or the maximum number of outgoing calls (in case of OutCall or AllCall registration). If the Visitor PINX supports these optional parameters, it shall set a duration timer and a counter for the number of outgoing calls to the values specified by these parameters. On expiry of the duration timer or when reaching the specified number of outgoing calls, the Visitor PINX shall terminate the registration session by means of a pumDe-reg invoke APDU to the Home PINX, as specified in 6.6.1.1 (conditional de-registration).

NOTE 2

Validation of the registration request can involve authentication of the PUM user.

NOTE 3

In the absence of sufficient addressing information, the Visitor PINX can, before sending a pumRegistr invoke APDU, use the procedures of 6.8.1 to make an enquiry to a Directory PINX in order to translate an identifier provided by the PUM user in the registration request into a PISN number.

NOTE 4

The number to be used in the Called party number information element when establishing the call independent signalling connection to the Home PINX is outside the scope of this Standard. It can, for example, be a PISN number provided by the PUM user in the registration request or optionally the PISN number provided by the Directory PINX.

On receipt of the pumRegistr return result APDU, the Visitor PINX shall stop timer T1, update the information in the VDB for the requested basic service and hosting address, revert to state VisitIdle and indicate acceptance to the PUM user.

The Visitor PINX is responsible for clearing the call independent signalling connection towards the Home PINX. This may occur on receipt of a return result APDU. Alternatively, the signalling connection may be retained for other applications, if appropriate.

6.5.1.2 Exceptional procedures

If the Visitor PINX is unable to proceed with the registration request, it shall enter state VisitIdle and indicate failure to the PUM user.

On receipt of a pumRegistr return error or reject APDU from the Home PINX, the Visitor PINX shall stop timer T1, revert to state VisitIdle and indicate rejection to the PUM user.

If timer T1 expires (i.e. the pumRegistr invoke APDU is not answered by the Home PINX), the Visitor PINX shall indicate rejection to the PUM user and enter state VisitIdle.

The Visitor PINX is responsible for clearing the call independent signalling connection towards the Home PINX. This may occur on receipt of a return error or reject APDU or on expiry of timer T1. Alternatively, the signalling connection may be retained for other applications, if appropriate.

6.5.2 Additional actions at the Visitor PINX in case of remote registration

The SDL representation of procedures at the Visitor PINX is shown in C.1 of annex C.

6.5.2.1 Normal procedures

On receipt of a pumRegistr invoke APDU from a Remote PINX using the call reference of a call independent signalling connection (as specified in 7.3 of ECMA-165), the actions specified in 6.5.1.1 shall apply with the term "PUM user" replaced by "Remote PINX". On receipt of the pumRegistr return result APDU from the Home PINX, the Visitor PINX shall send a pumRegistr return result to the Remote PINX. The contents of this pumRegistr return result APDU shall be the same as the contents of the pumRegistr return result APDU from the Home PINX.

6.5.2.2 Exceptional procedures

If the Visitor PINX is unable to proceed with the registration request, it shall send a pumRegistr return error APDU containing an appropriate error to the Remote PINX and enter state VisitIdle.

On receipt of a pumRegistr return error APDU from the Home PINX, the Visitor PINX shall send a pumRegistr return error APDU to the Remote PINX containing the same error as received in the pumRegistr return error APDU from the Home PINX, stop timer T1 and revert to state VisitIdle.

On receipt of a pumRegistr reject APDU from the Home PINX, the Visitor PINX shall send a pumRegistr return error APDU to the Remote PINX containing an appropriate error corresponding to the reject problem received in the pumRegistr reject APDU from the Home PINX, stop timer T1 and revert to state VisitIdle.

If timer T1 expires (i.e. the pumRegistr invoke APDU is not answered by the Home PINX), the Visitor PINX shall send a pumRegistr return error APDU containing the error temporarilyUnavailable to the Remote PINX and enter state VisitIdle.

6.5.3 Actions at the Home PINX for registration

The SDL representation of procedures at the Home PINX is shown in C.2 of annex C.

6.5.3.1 Normal procedures

On receipt of a pumRegistr invoke APDU using the call reference of a call independent signalling connection (as specified in 7.3 of ECMA-165), the Home PINX shall check the received basic service (element basicService) and the PUM user's PISN number or alternative identifier (element pumUserId), and verify that the PUM user is allowed to register.

If the PUM user is allowed to register, the Home PINX shall update the information in the HDB and answer the pumRegistr invoke APDU with a return result APDU.

If the received pumRegistr invoke APDU contained an InCall or AllCall registration (in element serviceOption), the Home PINX shall send a pumDelReg invoke APDU to the previous Visitor PINX using the call reference of a call independent signalling connection, in order to terminate the PUM user's previous InCall or AllCall registration session (if applicable).

If the received pumRegistr invoke APDU contained an OutCall registration (in element serviceOption), the Home PINX may optionally send a pumDelReg invoke APDU to the previous Visitor PINX using the call reference of a call independent signalling connection, in order to terminate the PUM user's previous InCall or AllCall registration session (if applicable).

The Home PINX may optionally leave the PUM user's existing OutCall registration session(s) unaffected (if applicable).

If the received pumRegistr invoke APDU does not contain the optional element serviceOption, i.e. does not specify the type of registration, the Home PINX shall assume an InCall registration and shall take the actions according to that type.

Termination of previous registration session(s) shall be performed by sending appropriate pumDelReg invoke APDU(s) to the previous Visitor PINX(s) using the call reference(s) of (a) call independent signalling connection(s). The call independent signalling connection(s) shall be established (or used, if an appropriate connection is already available) in accordance with the procedures specified in 7.3 of ECMA-165.

Within the argument of the pumDelReg invoke APDU(s), the PISN number or an alternative identifier of the PUM user shall be included in element pumUserId, the basic service for which the PUM user has registered shall be included in element basicService, a PISN number identifying the hosting user shall be included in element hostingAddr, and the type of registration (InCall, OutCall, or AllCall) shall be included in element serviceOption. The Home PINX shall enter state HomeDelete and start timer T4.

On receipt of all outstanding responses to the pumDelReg invoke APDUs, the Home PINX shall stop timer T4 and enter state HomeIdle.

The Home PINX is responsible for clearing the call independent signalling connection(s) towards the previous Visitor PINX(s). This may occur on receipt of a return result APDU. Alternatively, the signalling connection(s) may be retained for other applications, if appropriate.

6.5.3.2 Exceptional procedures

If the Home PINX is unable to proceed with the registration request, it shall answer the pumRegistr invoke APDU with a return error APDU containing an appropriate error and enter state HomeIdle.

If the PUM user is not found in the HDB, the Home PINX shall answer the pumRegistr invoke APDU with a return error APDU containing the error invalidServedUserNumber.

If the PUM user is not allowed to register, the Home PINX shall answer the pumRegistr invoke APDU with a return error APDU containing the error notAuthorized.

On receipt of a pumDelReg return error or reject APDU from a previous Visitor PINX and if there are still responses to the pumDelReg invoke APDU(s) outstanding, no action shall be taken at the Home PINX.

If timer T4 expires (e.g. a pumDelReg invoke APDU is not answered by a previous Visitor PINX), the Home PINX shall enter state HomeIdle.

NOTE 5

Expiry of timer T4 or receipt of a pumDelReg return error or reject APDU may invoke management actions to correct the VDB data in the previous Visitor PINX(s).

6.5.4 **Actions at a Transit PINX for registration**

No special actions are required in support of registration.

6.5.5 **Actions at the previous Visitor PINX for registration**

The SDL representation of procedures at the previous Visitor PINX is shown in C.3 of annex C.

6.5.5.1 **Normal procedures**

On receipt of a pumDelReg invoke APDU using the call reference of a call independent signalling connection (as specified in 7.3 of ECMA-165), the previous Visitor PINX shall check the received basic service (element basicService), the PUM user's PISN number or alternative identifier (element pumUserId), the hosting address (element hostingAddr) and the type of registration (element serviceOption), delete the specified VDB entry for the PUM user and answer the pumDelReg invoke APDU with a return result APDU.

6.5.5.2 **Exceptional procedures**

If the specified VDB entry for the PUM user is not found in the VDB, the previous Visitor PINX shall answer the pumDelReg invoke APDU with a return result APDU.

If the VDB is temporarily unavailable, the previous Visitor PINX shall answer the pumDelReg invoke APDU with a return error APDU containing the error temporarilyUnavailable.

6.5.6 **Actions at the Remote PINX for registration**

The SDL representation of procedures at the Remote PINX is shown in C.5 of annex C.

6.5.6.1 **Normal procedures**

On receipt of a valid registration request from an activating user, the Remote PINX shall send a pumRegistr invoke APDU to the Visitor PINX using the call reference of a call independent signalling connection. The call independent signalling connection shall be established (or used, if an appropriate connection is already available) in accordance with the procedures specified in 7.3 of ECMA-165. Within the argument of the invoke APDU, the PISN number or an alternative identifier of the PUM user shall be included in element pumUserId, the basic service for which the PUM user is to be registered shall be included in element basicService, a PISN number identifying the hosting user shall be included in element hostingAddr, and the PISN number of the activating user shall be included in element activatingUserAddr. The PISN number of the hosting user shall be suitable for use in the Called party number information element of a call or call independent signalling connection that is to be routed to the hosting user. If provided by the registration request from the activating user, the Remote PINX shall additionally include the type of registration (InCall, OutCall, or AllCall) in element serviceOption and the session parameters (maximum duration of the registration session, maximum number of outgoing calls) in element sessionParams. If provided by the registration request from the activating user, the Remote PINX shall further include the activating user's PIN in element userPin. The Remote PINX shall enter state RemoteReg and start timer T1.

NOTE 6

Validation of the registration request can involve authentication of the activating user.

NOTE 7

The number to be used in the Called party number information element when establishing the call independent signalling connection to the Visitor PINX is outside the scope of this Standard. It can, for example, be a PISN number provided by the activating user in the registration request.

On receipt of the pumRegistr return result APDU, the Remote PINX shall stop timer T1, revert to state RemoteIdle and indicate acceptance to the activating user.

The Remote PINX is responsible for clearing the call independent signalling connection towards the Visitor PINX. This may occur on receipt of a return result APDU. Alternatively, the signalling connection may be retained for other applications, if appropriate.

6.5.6.2 **Exceptional procedures**

On receipt of a pumRegistr return error or reject APDU from the Visitor PINX, the Remote PINX shall stop timer T1, revert to state RemoteIdle and indicate rejection to the activating user.

If timer T1 expires (i.e. the pumRegistr invoke APDU is not answered by the Visitor PINX), the Remote PINX shall indicate rejection to the activating user and enter state RemoteIdle.

The Remote PINX is responsible for clearing the call independent signalling connection towards the Visitor PINX. This may occur on receipt of a return error or reject APDU or on expiry of timer T1. Alternatively, the signalling connection may be retained for other applications, if appropriate.

6.6 SS-PUMR signalling procedures for de-registration

Examples of message sequences are shown in annex B.

6.6.1 Actions at the Visitor PINX for de-registration

The SDL representation of procedures at the Visitor PINX is shown in C.1 of annex C.

6.6.1.1 Normal procedures

On receipt of a valid de-registration request from a PUM user or an authorized user, the Visitor PINX shall:

− If the de-registration request is for a particular session or all sessions of a PUM user, send a pumDe-reg invoke APDU to the Home PINX using the call reference of a call independent signalling connection.

− If the de-registration request is for all PUM users being registered at a specific hosting address, send a pumDe-reg invoke APDU for each registered PUM user to the Home PINX using the call reference of a call independent signalling connection.

The call independent signalling connection shall be established (or used, if an appropriate connection is already available) in accordance with the procedures specified in 7.3 of ECMA-165. Within the argument of the invoke APDU(s), the PISN number or an alternative identifier of the PUM user shall be included in element pumUserId and the basic service for which the PUM user is to be de-registered shall be included in element basicService. If provided by the de-registration request from the PUM user or authorized user, the Visitor PINX shall additionally indicate in element serviceOption which type of registration session is to be de-registered. If only sessions at a specific hosting address are to be de-registered, a PISN number identifying the hosting user shall be included in element hostingAddr. The Visitor PINX shall enter state VisitDe-reg and start timer T3.

NOTE 8

The same procedures apply if the duration timer for a session expires or if the maximum number of outgoing calls is reached (conditional de-registration).

NOTE 9

Validation of the de-registration request can involve authentication of the PUM user or the authorized user.

On receipt of a pumDe-reg return result APDU, the Visitor PINX shall delete the information in the VDB for the requested PUM user and basic service and, if applicable, indicate acceptance to the PUM user or authorized user.

On receipt of all outstanding responses to the pumDe-reg invoke APDUs, the visitor PINX shall stop timer T3 and revert to state VisitIdle.

The Visitor PINX is responsible for clearing the call independent signalling connection towards the Home PINX. This may occur on receipt of all outstanding responses. Alternatively, the signalling connection may be retained for other applications, if appropriate.

On receipt of a pumDelReg invoke APDU from the Home PINX using the call reference of a call independent signalling connection (as specified in 7.3 of ECMA-165), the Visitor PINX shall check the received basic service (element basicService), the PUM user's PISN number or alternative identifier (element pumUserId), the hosting address (element hostingAddr) and the type of registration (element serviceOption), delete the specified VDB entry for the PUM user, and answer the pumDelReg invoke APDU with a return result APDU.

6.6.1.2 Exceptional procedures

On receipt of a pumDe-reg return error or reject APDU from the Home PINX and if there are still responses to the pumDe-reg invoke APDU(s) outstanding, no action shall be taken at the Visitor PINX, except, if applicable, indicate rejection to the PUM user or authorized user.

If timer T3 expires (i.e. a pumDe-reg invoke APDU is not answered by the Home PINX), the Visitor PINX shall enter state VisitIdle and, if applicable, indicate rejection to the PUM user.

The Visitor PINX is responsible for clearing the call independent signalling connection towards the Home PINX. This may occur on receipt of all outstanding responses or on expiry of timer T3. Alternatively, the signalling connection may be retained for other applications, if appropriate.

If, on receipt of a pumDelReg invoke APDU from the Home PINX, the specified VDB entry for the PUM user is not found in the VDB, the Visitor PINX shall answer with a pumDelReg return result APDU.

If the VDB is temporarily unavailable, the previous Visitor PINX shall answer the pumDelReg invoke APDU with a return error APDU containing the error temporarilyUnavailable.

6.6.2   Actions at the Home PINX for de-registration

The SDL representation of procedures at the Home PINX is shown in C.2 of annex C.

The procedures specified in the following apply for de-registration initiated by a PUM user at a the Visitor PINX and for remote de-registration.

6.6.2.1   Normal procedures

On receipt of a pumDe-reg invoke APDU using the call reference of a call independent signalling connection (as specified in 7.3 of ECMA-165), the Home PINX shall check the received basic service (element basicService) and the PUM user's PISN number or alternative identifier (element pumUserId) and verify that the PUM user may de-register.

The pumDe-reg invoke APDU may contain either a request for de-registration of a specific type of session (InCall, OutCall, AllCall) at a specific hosting address or a request for de-registration of all sessions at all hosting addresses.

If element serviceOption in the received pumDe-reg invoke APDU indicates 'outCallRegistration' and hostingAddr is omitted, the de-registration shall apply to all OutCall registrations of this PUM user.

If serviceOption indicates 'allCallRegistration' and hostingAddr is omitted, the de-registration shall apply to the AllCall and all OutCall registrations of this PUM user.

If a hosting address (in element hostingAddr) is contained in the received pumDe-reg invoke APDU, the de-registration shall apply only to the session indicated in element serviceOption of this PUM user at this hosting address.

If the received pumDe-reg invoke APDU does not contain the optional element serviceOption, i.e. does not specify which type of registration session is to be de-registered, the Home PINX shall assume an InCall de-registration and shall take the actions according to that type.

If the PUM user may de-register, the Home PINX shall update the location information in the HDB and answer the pumDe-reg invoke APDU with a return result APDU.

If required according to the contents of the pumDe-reg invoke APDU, the Home PINX shall additionally send (a) pumDelReg invoke APDU(s) to the Visitor PINX(s) using the call reference(s) of (a) call independent signalling connection(s). The call independent signalling connection(s) shall be established (or used, if an appropriate connection is already available) in accordance with the procedures specified in 7.3 of ECMA-165.

Within the argument of the pumDelReg invoke APDU(s), the PISN number or an alternative identifier of the PUM user shall be included in element pumUserId, the basic service for which the PUM user is to be de-registered shall be included in element basicService, a PISN number identifying the hosting user shall be included in element hostingAddr, and the type of registration (InCall, OutCall, or AllCall) shall be included in element serviceOption. The Home PINX shall enter state HomeDelete and start timer T4.

On receipt of all outstanding responses to the pumDelReg invoke APDUs, the Home PINX shall stop timer T4 and enter state HomeIdle.

The Home PINX is responsible for clearing the call independent signalling connection(s) towards the Visitor PINX(s). This may occur on receipt of a return result APDU. Alternatively, the signalling connection(s) may be retained for other applications, if appropriate.

6.6.2.2 **Exceptional procedures**

If the Home PINX is unable to proceed with the de-registration request, it shall answer the pumDe-reg invoke APDU with a return error APDU containing an appropriate error and enter state HomeIdle.

If the PUM user is not found in the HDB, the Home PINX shall answer the pumDe-reg invoke APDU with a return error APDU containing the error invalidServedUserNumber.

If the PUM user is not allowed to de-register, the Home PINX shall answer the pumDe-reg invoke APDU with a return error APDU containing the error notAuthorized.

On receipt of a pumDelReg return error or reject APDU from a Visitor PINX and if there are still responses to the pumDelReg invoke APDU(s) outstanding, no action shall be taken at the Home PINX.

If timer T4 expires (e.g. a pumDelReg invoke APDU is not answered by a Visitor PINX), the Home PINX shall enter state HomeIdle.

NOTE 10

Expiry of timer T4 or receipt of a pumDelReg return error or reject APDU may invoke management actions to correct the VDB data in the previous Visitor PINX(s).

6.6.3 **Actions at a Transit PINX for de-registration**

No special actions are required in support of SS-PUMR.

6.6.4 **Actions at the Remote PINX for de-registration**

The SDL representation of procedures at the Remote PINX is shown in C.5 of annex C.

6.6.4.1 **Normal procedures**

On receipt of a valid de-registration request from an activating user, the Remote PINX shall send a pumDe-reg invoke APDU to the Home PINX using the call reference of a call independent signalling connection. The call independent signalling connection shall be established (or used, if an appropriate connection is already available) in accordance with the procedures specified in 7.3 of ECMA-165. Within the argument of the invoke APDU, the PISN number or an alternative identifier of the PUM user shall be included in element pumUserId, the basic service for which the PUM user is to be de-registered shall be included in element basicService, and the PISN number of the activating user shall be included in element activatingUserAddr. The de-registration request from the activating user may contain a request for de-registration of either a specific type of session (InCall, OutCall, AllCall) at a specific hosting address or all sessions of a PUM user at all hosting addresses. If provided by the de-registration request from the activating user, the Remote PINX shall indicate in element serviceOption which type of registration session is to be de-registered. If only sessions at a specific hosting address are to be de-registered, a PISN number identifying the hosting user shall be included in element hostingAddr. The Remote PINX shall enter state RemoteDe-reg and start timer T3.

NOTE 11

In the absence of sufficient addressing information, the Remote PINX can, before sending a pumDe-reg invoke APDU, use the procedures of 6.5.3 to make an enquiry to a Directory PINX in order to translate an identifier provided by the activating user in the de-registration request into a PISN number.

NOTE 12

Validation of the de-registration request can involve authentication of the activating user.

On receipt of the pumDe-reg return result APDU, the Remote PINX shall stop timer T3, revert to state RemoteIdle and, if applicable, indicate acceptance to the activating user.

The Remote PINX is responsible for clearing the call independent signalling connection towards the Home PINX. This may occur on receipt of a return result APDU. Alternatively, the signalling connection may be retained for other applications, if appropriate.

6.6.4.2 **Exceptional procedures**

On receipt of a pumDe-reg return error or reject APDU from the Home PINX, the Remote PINX shall stop timer T3, revert to state RemoteIdle and, if applicable, indicate rejection to the activating user.

If timer T3 expires (i.e. the pumDe-reg invoke APDU is not answered by the Home PINX), the Remote PINX shall enter state RemoteIdle and, if applicable, indicate rejection to the activating user.

The Remote PINX is responsible for clearing the call independent signalling connection towards the Home PINX. This may occur on receipt of a return error or reject APDU or on expiry of timer T3. Alternatively, the signalling connection may be retained for other applications, if appropriate.

6.7 SS-PUMR signalling procedures for interrogation

Examples of message sequences are shown in annex B.

6.7.1 Actions at the Visitor PINX for interrogation

The SDL representation of procedures at the Visitor PINX is shown in C.1 of annex C.

6.7.1.1 Normal procedures

On receipt of a pumInterrog invoke APDU from the Home PINX using the call reference of a call independent signalling connection (as specified in 7.3 of ECMA-165), the Visitor PINX shall check the received basic service (element basicService) and the PUM user's PISN number or alternative identifier (element pumUserId) and answer the pumInterrog invoke APDU with a return result APDU containing the required information (interrogParams).

On receipt of a valid interrogation request from a PUM user (interrogation of existing registration sessions / hosting addresses), the Visitor PINX shall send a pumInterrog invoke APDU to the Home PINX using the call reference of a call independent signalling connection. The call independent signalling connection shall be established (or used, if an appropriate connection is already available) in accordance with the procedures specified in 7.3 of ECMA-165.

Within the argument of the invoke APDU, the PISN number or an alternative identifier of the PUM user shall be included in element pumUserId, the basic service shall be included in element basicService. In accordance with the interrogation request from the PUM user, the invoke APDU may additionally include the element homeInfoOnly indicating whether only Home PINX information or complete information is required, and a PISN number identifying the hosting user in element hostingAddr. Omission of the element hostingAddr indicates that information about all hosting addresses of this PUM user and this basic service is required. The Visitor PINX shall enter state VisitInterrogation and start timer T5.

NOTE 13

Validation of the interrogation request can involve authentication of the PUM user.

On receipt of the pumInterrog return result APDU, the Visitor PINX shall stop timer T5, revert to state VisitIdle and forward the received information to the PUM user.

6.7.1.2 Exceptional procedures

If the Visitor PINX is unable to proceed with the interrogation request, it shall answer the pumInterrog invoke APDU with a return error APDU containing an appropriate error and enter state VisitIdle.

If the PUM user is not found in the VDB, the Visitor PINX shall answer the pumInterrog invoke APDU with a return error APDU containing the error pumUserNotRegistered.

On receipt of a pumInterrog return error or reject APDU the Visitor PINX shall stop timer T5, revert to state VisitIdle and indicate rejection to the PUM user.

If timer T5 expires (i.e. the pumInterrog invoke APDU is not answered), the Visitor PINX shall enter state VisitIdle and indicate rejection to the PUM user.

The Visitor PINX is responsible for clearing the call independent signalling connection towards the Home PINX. This may occur on receipt of a return error or reject APDU or on expiry of timer T5. Alternatively, the signalling connection may be retained for other applications, if appropriate.

6.7.2 Actions at the Home PINX for interrogation

The SDL representation of procedures at the Visitor PINX is shown in C.2 of annex C.

6.7.2.1 Normal procedures

On receipt of a pumInterrog invoke APDU using the call reference of a call independent signalling connection (as specified in 7.3 of ECMA-165), the Home PINX shall check the received basic service (element basicService) and the PUM user's PISN number or alternative identifier (element pumUserId) and verify that the PUM user may interrogate.

If the PUM user may interrogate, the Home PINX shall:

− If the received pumInterrog invoke APDU contained an interrogation of Home PINX information for existing registration sessions, obtain the required information from the HDB and answer the pumInterrog invoke APDU with a return result APDU containing the required information (elements basicService, hostingAddr, serviceOption).

− If the received pumInterrog invoke APDU (additionally) contained an interrogation of complete information (i.e. including session parameters), obtain this information from the HDB and from the corresponding Visitor PINX(s) by means of an interrogation at this (these) Visitor PINX(s) and then answer the pumInterrog invoke APDU with a return result APDU containing the required information (elements basicService, hostingAddr, serviceOption, and interrogParams).

To interrogate the session parameters of a particular session at a Visitor PINX, the Home PINX shall send a pumInterrog invoke APDU to this Visitor PINX using the call reference of a call independent signalling connection. The call independent signalling connection shall be established (or used, if an appropriate connection is already available) in accordance with the procedures specified in 7.3 of ECMA-165.
Within the argument of the invoke APDU, the PISN number or an alternative identifier of the PUM user shall be included in element pumUserId, the basic service shall be included in element basicService, a PISN number identifying the hosting user shall be included in element hostingAddr. The Home PINX shall start timer T5 and enter state HomeInterrogation.

The received pumInterrog invoke may require more than one interrogation to various other Visitor PINXs. In this case the Home PINX shall collect the information received in the pumInterrog return result APDUs.

On receipt of the last response to the pumInterrog invoke APDU, the Home PINX shall stop timer T5, send the obtained information to the interrogating PINX using the pumInterrog return result APDU and enter state HomeIdle.

The Home PINX is responsible for clearing the call independent signalling connection towards the interrogated Visitor PINX(s). This may occur on receipt of a return result APDU. Alternatively, the signalling connection may be retained for other applications, if appropriate.

6.7.2.2 Exceptional procedures

If the Home PINX is unable to proceed with the interrogation request, it shall answer the pumInterrog invoke APDU with a return error APDU containing an appropriate error and enter state HomeIdle.

If the PUM user may not interrogate, the Home PINX shall answer the pumInterrog invoke APDU with a return error APDU containing the error notAuthorized.

If the PUM user is not found in the HDB, the Home PINX shall answer the pumInterrog invoke APDU with a return error APDU containing the error invalidServedUserNumber.

If timer T5 expires (i.e. the pumInterrog invoke APDU is not answered by all of the other Visitor PINXs), the Home PINX shall answer the pumInterrog invoke APDU with a return result APDU containing the available information and enter state HomeIdle.

6.7.3 Actions at the Remote PINX for interrogation

The SDL representation of procedures at the Visitor PINX is shown in C.5 of annex C.

PUM user is either the PUM user or an authorized user acting on behalf of the PUM user.

6.7.3.1 Normal procedures

On receipt of a valid interrogation request from a PUM user, the Remote PINX shall send a pumInterrog invoke APDU to the Home PINX using the call reference of a call independent signalling connection. The call independent signalling connection shall be established (or used, if an appropriate connection is already available) in accordance with the procedures specified in 7.3 of ECMA-165.

Within the argument of the invoke APDU, the PISN number or an alternative identifier of the PUM user shall be included in element pumUserId, the basic service shall be included in element basicService. In accordance with the interrogation request from the PUM user, the invoke APDU may additionally include the element homeInfoOnly indicating whether only Home PINX information or complete information is required, and a PISN number identifying the hosting user in element hostingAddr. Omission of the element hostingAddr indicates that information about all hosting addresses of this PUM user and this basic service is required. The PISN number of the activating user shall be included in element activatingUserAddr. The Remote PINX shall enter state RemoteInterrogation and start timer T5.

NOTE 14

In the absence of sufficient addressing information, the Remote PINX can, before sending a pumInterrog invoke APDU, use the procedures of 6.5.3 to make an enquiry to a Directory PINX in order to translate an identifier provided by the PUM user in the interrogation request into a PISN number.

NOTE 15

Validation of the interrogation request can involve authentication of the PUM user.

On receipt of the pumInterrog return result APDU, the Remote PINX shall stop timer T5, revert to state RemoteIdle and forward the received information to the PUM user.

6.7.3.2 Exceptional procedures

On receipt of a pumInterrog return error or reject APDU the Remote PINX shall stop timer T5, revert to state RemoteIdle and indicate rejection to the PUM user.

If timer T5 expires (i.e. the pumInterrog invoke APDU is not answered), the Remote PINX shall enter state RemoteIdle and indicate rejection to the PUM user.

The Remote PINX is responsible for clearing the call independent signalling connection towards the Home PINX. This may occur on receipt of a return error or reject APDU or on expiry of timer T5. Alternatively, the signalling connection may be retained for other applications, if appropriate.

6.8 Signalling procedures for enquiry to a Directory PINX

Examples of message sequences are shown in annex B.

6.8.1 Actions at a PINX for enquiry to a Directory PINX

The SDL representation of these procedures is shown in C.4 of annex C.

6.8.1.1 Normal procedures

In order to make an enquiry to a Directory PINX to translate an identifier provided by the PUM user into a PISN number (the PUM number), the PINX may send a pisnEnquiry invoke APDU to the Directory PINX using the call reference of a call independent signalling connection. The call independent signalling connection shall be established (or used, if an appropriate connection is already available) in accordance with the procedures specified in 7.3 of ECMA-165. Within the argument of the invoke APDU, element alternativeId shall contain the identifier provided by the PUM user. The PINX shall enter state Enquiry and start timer T2.

NOTE 16

The number to be used in the Called party number information element when establishing the call independent signalling connection to the Directory PINX is outside the scope of this Standard.

On receipt of the pisnEnquiry return result APDU, the PINX shall stop timer T2. The PISN number received in the pisnEnquiry return result APDU can be used e.g. for registration.

The PINX is responsible for clearing the call independent signalling connection towards the Directory PINX. This may occur on receipt of a return result APDU. Alternatively, the signalling connection may be retained for other applications, if appropriate.

6.8.1.2 Exceptional procedures

On receipt of a pisnEnquiry return error or reject APDU from the Directory PINX, the PINX shall stop timer T2 and enter state Idle.

If timer T2 expires, the PINX shall enter state Idle.

NOTE 17

If, in any of these situations the PINX is unable to proceed, it should indicate rejection to the PUM user or answer a received invoke APDU with an appropriate return error APDU, respectively, unless there is another means available for obtaining the PISN number of the PUM user.

The PINX is responsible for clearing the call independent signalling connection towards the Directory PINX. This may occur on receipt of a return error or reject APDU or on expiry of timer T2. Alternatively, the signalling connection may be retained for other applications, if appropriate.

6.8.2 Actions at the Directory PINX for enquiry

The SDL representation of procedures at the Directory PINX is shown in C.4 of annex C.

6.8.2.1 Normal procedures

On receipt of a pisnEnquiry invoke APDU using the call reference of a call independent signalling connection (as specified in 7.3 of ECMA-165), the Directory PINX shall attempt to translate the PUM user's identity received in the pisnEnquiry invoke APDU to a PISN number. The PUM user's PISN number shall be returned in a pisnEnquiry return result APDU.

6.8.2.2 Exceptional procedures

If the Directory PINX is unable to translate the identity received in the pisnEnquiry invoke APDU, a pisnEnquiry return error APDU containing the error invalidServedUserNumber shall be returned.

6.9 SS-PUMR impact of interworking with public ISDNs

Not applicable.

6.10 SS-PUMR impact of interworking with non-ISDNs

Not applicable.

6.11 Protocol interactions between SS-PUMR and other supplementary services and ANFs

This clause specifies protocol interactions with other supplementary services and ANFs for which stage 3 standards had been published at the time of publication of this Standard. For interactions with supplementary services and ANFs for which stage 3 standards are published subsequent to the publication of this Standard, see those other stage 3 standards.

NOTE 18

Simultaneous conveyance of APDUs for SS-PUMR and other supplementary services or ANFs in the same message, each in accordance with the requirements of its respective stage 3 standard, does not, on its own, constitute a protocol interaction.

6.11.1 Interaction with Calling Name Identification Presentation (SS-CNIP)

No interaction.

6.11.2 Interaction with Connected Name Identification Presentation (SS-CONP)

No interaction.

6.11.3 Interaction with Call Forwarding Unconditional (SS-CFU)

No interaction.

6.11.4 Interaction with Call Forwarding Busy (SS-CFB)

No interaction.

6.11.5 Interaction with Call Forwarding No Reply (SS-CFNR)

No interaction.

6.11.6 Interaction with Call Deflection (SS-CD)

No interaction.

6.11.7   Interaction with Path Replacement (ANF-PR)

No interaction.

6.11.8   Interaction with Call Transfer (SS-CT)

No interaction.

6.11.9   Interaction with Call Completion to Busy Subscriber (SS-CCBS)

No interaction.

NOTE 19

SS-CCBS may need to be cancelled if the PUM user (as either the calling user or called user) changes location or de-registers.

6.11.10   Interaction with Call Completion on No Reply (SS-CCNR)

No interaction.

NOTE 20

SS-CCNR may need to be cancelled if the PUM user (as either the calling user or called user) changes location or de-registers.

6.11.11   Interaction with Call Offer (SS-CO)

No interaction.

6.11.12   Interaction with Do Not Disturb (SS-DND)

No interaction.

6.11.13   Interaction with Do Not Disturb Override (SS-DNDO)

No interaction.

6.11.14   Interaction with Call Intrusion (SS-CI)

No interaction.

6.11.15   Interaction with Call Interception (ANF-CINT)

No interaction.

6.11.16   Interaction with Transit Counter (ANF-TC)

No interaction.

6.11.17   Interaction with Recall (SS-RE)

No interaction.

6.11.18   Interaction with Advice Of Charge (SS-AOC)

No interaction.

6.11.19   Interaction with Cordless Terminal Incoming Call (SS-CTMI)

No interaction.

6.11.20   Interactions with Cordless Terminal Outgoing call (SS-CTMO)

No interaction.

6.11.21   Interactions with Cordless Terminal Location Registration (SS-CTLR)

Interactions with CTLR occur when a PUM user is registered at a CT and the CT changes location as well as in case of an initial remote registration at a CT.

An example message sequence is shown in annex B.

6.11.21.1   Actions at the previous Visitor PINX

On receipt of a locationDelete invoke APDU from a CTM Home PINX and if the previous Visitor PINX detects by checking its VDB that a PUM user is registered at the corresponding CT, it shall act as a Remote PINX and send a pumRegistr invoke APDU to the CTM Home PINX according to the procedures specified in

6.5.6. The pumRegistr invoke APDU may be sent together with the locationDelete return result APDU in one message.

On receipt of a pumDelReg invoke APDU the previous Visitor PINX shall act as specified in 6.5.5.

6.11.21.2 Actions at the CTM Home PINX in case of location update

On receipt of a pumRegistr invoke APDU from the previous Visitor PINX, the CTM Home PINX shall act as a Remote PINX and send a corresponding pumRegistr invoke APDU to the new Visitor PINX according to the procedures specified in 6.5.6.

On receipt of the pumRegistr return result APDU from the new Visitor PINX, the CTM Home PINX shall act as a Remote PINX as specified in 6.5.6 and additionally send a corresponding pumRegistr return result APDU to the previous Visitor PINX.

6.11.21.3 Actions at the CTM Home PINX in case of initial remote registration

On receipt of a pumRegistr invoke APDU from the initiating Remote PINX, the CTM Home PINX shall act as a Remote PINX and send a corresponding pumRegistr invoke APDU to the Visitor PINX according to the procedures specified in 6.5.6.

On receipt of the pumRegistr return result APDU from the Visitor PINX, the CTM Home PINX shall act as a Remote PINX as specified in 6.5.6 and additionally send a corresponding pumRegistr return result APDU to the initiating Remote PINX.

6.11.21.4 Actions at the (new) Visitor PINX

On receipt of a pumRegistr invoke APDU from the CTM Home PINX, the (new) Visitor PINX shall act as a Visitor PINX in case of remote registration and send a corresponding pumRegistr invoke APDU to the (PUM) Home PINX according to the procedures specified in 6.5.2.

6.11.21.5 Actions at the (PUM) Home PINX

On receipt of a pumRegistr invoke APDU from the (new) Visitor PINX, the Home PINX shall send a pumRegistr return result APDU to the Visitor PINX and a pumDelReg invoke APDU to the previous Visitor PINX according to the procedures specified in 6.5.3 (if applicable).

6.11.22 Interactions with Cordless Terminal Authentication of the CTM User (SS-CTAT)

No interaction.

6.11.23 Interactions with Cordless Terminal Authentication of the PISN (SS-CTAN)

No interaction.

6.11.24 Interactions with Message Waiting Indication (SS-MWI)

6.11.24.1 Actions at the Home PINX for registration

On receipt of a pumRegistr invoke APDU containing the request for an InCall or AllCall registration and if any stored mwiActivate invoke APDU for that PUM user is available, the Home PINX shall send a mwiActivate invoke ADPU to the Visitor PINX, in addition to the pumRegistr return result APDU which is sent to the Visitor PINX or, in case of remote registration, to the Remote PINX, according to 6.5.3.

On receipt of a mwiActivate invoke APDU together with the pumDelReg return result APDU from a previous Visitor PINX, the Home PINX shall forward the mwiActivate invoke APDU to the (new) Visitor PINX.

6.11.24.2 Actions at the previous Visitor PINX for registration

On receipt of a pumDelReg invoke APDU and if any stored mwiActivate invoke APDU for that PUM user is available, the previous Visitor PINX shall send a mwiActivate invoke ADPU together with the pumDelReg return result APDU to the Home PINX according to 6.5.5.

6.11.24.3 Actions at the Visitor PINX for de-registration

On receipt of a pumDelReg invoke APDU and if any stored mwiActivate invoke APDU for that PUM user is available, the Visitor PINX shall send a mwiActivate invoke ADPU together with the pumDelReg return result APDU to the Home PINX according to 6.6.1.

6.11.25 Interactions with Common Information (ANF-CMN)

No interaction.

6.11.26   Interactions with Call Priority Interruption (SS-CPI)

No interaction.

6.11.27   Interactions with Call Priority Interruption Protection (SS-CPIP)

No interaction.

6.11.28   Interactions with Call Distribution (ANF-DIST)

No interaction.

6.12   SS-PUMR parameter values (timers)

The following timers apply:

6.12.1   Timer T1

Timer T1 operates at the Visitor PINX during state VisitReg and at the Remote PINX during state RemoteReg. Its purpose is to protect against the absence of a response to the pumRegistr invoke APDU.

Timer T1 shall have a value not less than 15 s.

6.12.2   Timer T2

Timer T2 operates at the PINX during state Enquiry. Its purpose is to protect against the absence of a response to the pisnEnquiry invoke APDU.

Timer T2 shall have a value not less than 15 s.

6.12.3   Timer T3

Timer T3 operates at the Visitor PINX during state VisitDe-reg and at the Remote PINX during state RemoteDe-reg. Its purpose is to protect against the absence of a response to the pumDe-reg invoke APDU.

Timer T3 shall have a value not less than 15 s.

6.12.4   Timer T4

Timer T4 operates at the Home PINX during state HomeDelete. Its purpose is to protect against the absence of a response to the pumDelReg invoke APDU.

Timer T4 shall have a value not less than 15 s.

6.12.5   Timer T5

Timer T5 operates at the Visitor PINX during state VisitInterrogation, at the Remote PINX during state RemoteInterrogation, and at the Home PINX during state HomeInterrogation. Its purpose is to protect against the absence of a response to the pumInterrog invoke APDU.

Timer T5 shall have a value not less than 15 s.

Annex A

(normative)

Protocol Implementation Conformance Statement (PICS) Proforma

A.1   Introduction

The supplier of a protocol implementation which is claimed to conform to this Standard shall complete the following Protocol Implementation Conformance Statement (PICS) proforma.

A completed PICS proforma is the PICS for the implementation in question. The PICS is a statement of which capabilities and options of the protocol have been implemented. The PICS can have a number of uses, including use:

–   by the protocol implementor, as a check list to reduce the risk of failure to conform to the Standard through oversight;

–   by the supplier and acquirer, or potential acquirer, of the implementation, as a detailed indication of the capabilities of the implementation, stated relative to the common basis for understanding provided by the Standard's PICS proforma;

–   by the user or potential user of the implementation, as a basis for initially checking the possibility of interworking with another implementation - while interworking can never be guaranteed, failure to interwork can often be predicted from incompatible PICS's;

–   by a protocol tester, as the basis for selecting appropriate tests against which to assess the claim for conformance of the implementation.

A.2   Instructions for completing the PICS proforma

A.2.1   General structure of the PICS proforma

The PICS proforma is a fixed format questionnaire divided into sub-clauses each containing a group of individual items. Each item is identified by an item number, the name of the item (question to be answered), and the reference(s) to the clause(s) that specifies (specify) the item in the main body of this Standard.

The "Status" column indicates whether an item is applicable and if so whether support is mandatory or optional. The following terms are used:

m   mandatory (the capability is required for conformance to the protocol);

o   optional (the capability is not required for conformance to the protocol, but if the capability is implemented it is required to conform to the protocol specifications);

o.<n>   optional, but support of at least one of the group of options labelled by the same numeral <n> is required;

x   prohibited;

c.<cond>   conditional requirement, depending on support for the item or items listed in condition <cond>;

<item>:m   simple conditional requirement, the capability being mandatory if item number <item> is supported, otherwise not applicable;

<item>:o   simple conditional requirement, the capability being optional if item number <item> is supported, otherwise not applicable.

Answers to the questionnaire items are to be provided either in the "Support" column, by simply marking an answer to indicate a restricted choice (Yes or No), or in the "Not Applicable" column (N/A).

A.2.2　Additional Information

Items of Additional Information allow a supplier to provide further information intended to assist the interpretation of the PICS. It is not intended or expected that a large quantity will be supplied, and a PICS can be considered complete without any such information. Examples might be an outline of the ways in which a (single) implementation can be set up to operate in a variety of environments and configurations.

References to items of Additional Information may be entered next to any answer in the questionnaire, and may be included in items of Exception information.

A.2.3　Exception Information

It may occasionally happen that a supplier will wish to answer an item with mandatory or prohibited status (after any conditions have been applied) in a way that conflicts with the indicated requirement. No pre-printed answer will be found in the Support column for this. Instead, the supplier is required to write into the support column an x.<i> reference to an item of Exception Information, and to provide the appropriate rationale in the Exception item itself.

An implementation for which an Exception item is required in this way does not conform to this Standard. A possible reason for the situation described above is that a defect in the Standard has been reported, a correction for which is expected to change the requirement not met by the implementation.

A.3   PICS proforma for ECMA-282

A.3.1   Implementation Identification

Supplier	
Contact point for queries about the PICS	
Implementation Name(s) and Version(s)	
Other information necessary for full identification, e.g., name(s) and version(s) for machines and/or operating systems; system name(s)	

Only the first three items are required for all implementations; other information may be completed as appropriate in meeting the requirement for full identification.

The terms Name and Version should be interpreted appropriately to correspond with a suppliers terminology (e.g., Type, Series, Model).

A.3.2   Protocol Summary

Protocol version	1.0
Addenda Implemented (if applicable)	
Amendments Implemented	
Have any exception items been required (see A.2.3)	No [ ] Yes [ ] (The answer Yes means that the implementation does not conform to this Standard)

Date of Statement	

A.3.3   General

Item	Question/feature	References	Status	N/A	Support
A1	Support of functionality of a Visitor PINX		o.1		Yes[ ] No[ ]
A2	Support of functionality of a Home PINX		o.1		Yes[ ] No[ ]
A3	Support of functionality of a Remote PINX		o.1		Yes[ ] No[ ]
A4	Support of functionality of a Directory PINX		o		Yes[ ] No[ ]
A5	Support of InCall registration	6.1	c.1	[ ]	m:Yes[ ]
A6	Support of OutCall registration	6.1	c.2	[ ]	Yes[ ] No[ ]
A7	Support of AllCall registration	6.1	c.2	[ ]	Yes[ ] No[ ]

c.1 = IF (A1 OR A2) THEN m, ELSE N/A
c.2 = IF (A1 OR A2) THEN o, ELSE N/A

A.3.4   Procedures

Item	Question/feature	References	Status	N/A	Support
B1	Support of generic procedures for the call independent control (connection oriented) of supplementary services as specified in ECMA-165 for an originating and terminating PINX in the Visitor PINX	6.2.1, 6.2.3	A1: m	[ ]	m:Yes[ ]
B2	Support of generic procedures for the call independent control (connection oriented) of supplementary services as specified in ECMA-165 for an originating and terminating PINX in the Home PINX	6.2.2	A2: m	[ ]	m:Yes[ ]
B3	Support of generic procedures for the call independent control (connection oriented) of supplementary services as specified in ECMA-165 for an originating PINX in the Remote PINX	6.2.6	A3: m	[ ]	m:Yes[ ]
B4	Support of generic procedures for the call independent control (connection oriented) of supplementary services as specified in ECMA-165 for a terminating PINX in the Directory PINX	6.2.5	A4: m	[ ]	m:Yes[ ]
B5	Support of generic procedures for the call independent control (connection oriented) of supplementary services as specified in ECMA-165 for a Transit PINX	6.2.4	m		Yes[ ]
B6	Support of signalling procedures for Registration at the Visitor PINX	6.5.1	A1: m	[ ]	m:Yes[ ]
B7	Support of additional signalling procedures for Remote registration at the Visitor PINX	6.5.2	A1: o	[ ]	Yes[ ] No[ ]
B8	Support of signalling procedures for Registration at the Home PINX	6.5.3	A2: m	[ ]	m:Yes[ ]

B9	Support of signalling procedures for Delete registration in a Home PINX	6.5.3, 6.6.2	A2: m	[ ]	m:Yes[ ]
B10	Support of signalling procedures for Delete registration in a previous Visitor PINX	6.5.5	A1: m	[ ]	m:Yes[ ]
B11	Support of signalling procedures for Registration at the Remote PINX	6.5.6	A3: m	[ ]	m:Yes[ ]
B12	Support of signalling procedures for De-registration at the Visitor PINX	6.6.1	A1: m	[ ]	m:Yes[ ]
B13	Support of signalling procedures for De-registration at the Home PINX	6.6.2	A2: m	[ ]	m:Yes[ ]
B14	Support of signalling procedures for De-registration at the Remote PINX	6.6.4	A3: m	[ ]	m:Yes[ ]
B15	Support of signalling procedures for Interrogation at the Visitor PINX	6.7.1	A1: o	[ ]	Yes[ ] No[ ]
B16	Support of signalling procedures for Interrogation at the Home PINX	6.7.2	A2: o	[ ]	Yes[ ] No[ ]
B17	Support of signalling procedures for Interrogation at the Remote PINX	6.7.3	A3: o	[ ]	Yes[ ] No[ ]
B18	Support of signalling procedures for Enquiry to a Directory PINX	6.8.1	c.3	[ ]	Yes[ ] No[ ]
B19	Support of signalling procedures for Enquiry at a Directory PINX	6.8.2	A4: m	[ ]	m:Yes[ ]
B20	Support of timer for duration of session and counter for number of outgoing calls at a Visitor PINX	6.5.1	A1: o	[ ]	Yes[ ] No[ ]
B21	Support of the option to leave existing OutCall registrations unaffected at a Home PINX	6.5.3	A2: o	[ ]	Yes[ ] No[ ]

c.3 = IF (A1 OR A3) THEN o, ELSE N/A

A.3.5   Coding

Item	Question/feature	References	Status	N/A	Support
C1	Sending of pumRegistr invoke and receipt of pumRegistr return result/error by a Visitor PINX	6.3.1	m		Yes[ ]
C2	Sending of pumRegistr invoke and receipt of pumRegistr return result/error by a Remote PINX	6.3.1	B11: m	[ ]	m:Yes[ ]
C3	Sending of pumDelReg invoke and receipt of pumDelReg return result/error by a Home PINX	6.3.1	m		Yes[ ]
C4	Sending of pumDe-reg invoke and receipt of pumDe-reg return result/error by a Visitor PINX	6.3.1	m		Yes[ ]
C5	Sending of pumDe-reg invoke and receipt of pumDe-reg return result/error by a Remote PINX	6.3.1	B14: m	[ ]	m:Yes[ ]
C6	Sending of pumInterrog invoke and receipt of pumInterrog return result/error by a Visitor PINX	6.3.1	B15: m	[ ]	m:Yes[ ]

C7	Sending of pumInterrog invoke and receipt of pumInterrog return result/error by a Remote PINX	6.3.1	B17: m	[ ]	m:Yes[ ]
C8	Sending of pumInterrog invoke and receipt of pumInterrog return result/error by a Home PINX	6.3.1	B16: m	[ ]	m:Yes[ ]
C9	Receipt of pumRegistr invoke and sending of pumRegistr return result/error by a Visitor PINX	6.3.1	B7: m	[ ]	m:Yes[ ]
C10	Receipt of pumRegistr invoke and sending of pumRegistr return result/error by a Home PINX	6.3.1	m		Yes[ ]
C11	Receipt of pumDelReg invoke and sending of pumDelReg return result/error by a Visitor PINX	6.3.1	m		Yes[ ]
C12	Receipt of pumDe-reg invoke and sending of pumDe-reg return result/error by a Home PINX	6.3.1	m		Yes[ ]
C13	Receipt of pumInterrog invoke and sending of pumInterrog return result/error by a Visitor PINX	6.3.1	B15: m	[ ]	m:Yes[ ]
C14	Receipt of pumInterrog invoke and sending of pumInterrog return result/error by a Home PINX	6.3.1	B16: m	[ ]	m:Yes[ ]
C15	Sending of pisnEnquiry invoke and receipt of pisnEnquiry return result/error by a PINX	6.3.1	B18: m	[ ]	m:Yes[ ]
C16	Receipt of pisnEnquiry invoke and sending of pisnEnquiry return result/error by a Directory PINX	6.3.1	B19: m	[ ]	m:Yes[ ]

A.3.6　Timer

Item	Question/feature	References	Status	N/A	Support
D1	Support of Timer T1 at a Visitor PINX	6.12.1	A1: m	[ ]	m:Yes[ ]
D2	Support of Timer T1 at a Remote PINX	6.12.1	A2: m	[ ]	m:Yes[ ]
D3	Support of Timer T2 at a PINX	6.12.2	B18: m	[ ]	m:Yes[ ]
D4	Support of Timer T3 at a Visitor PINX	6.12.3	A1: m	[ ]	m:Yes[ ]
D5	Support of Timer T3 at a Remote PINX	6.12.3	A2: m	[ ]	m:Yes[ ]
D6	Support of Timer T4 at a Home PINX	6.12.4	A2: m	[ ]	m:Yes[ ]
D7	Support of Timer T5 at a Visitor PINX	6.12.5	B15: m	[ ]	m:Yes[ ]
D8	Support of Timer T5 at a Remote PINX	6.12.5	B16: m	[ ]	m:Yes[ ]
D9	Support of Timer T5 at a Home PINX	6.12.5	B17: m	[ ]	m:Yes[ ]

A.3.7   Interactions between SS-PUMR and SS-CTLR

Item	Question/feature	References	Status	N/A	Support
E1	Support of SS-CTLR	ECMA-216	o		Yes[ ] No[ ]
E2	Support of procedures at a Home PINX for interaction with SS-CTLR	6.11.21.5	E1: m	[ ]	m:Yes[ ]
E3	Support of procedures at a previous Visitor PINX / Visitor PINX for interaction with SS-CTLR	6.11.21.1 6.11.21.4	E1: m	[ ]	m:Yes[ ]

A.3.8   Interactions between SS-PUMR and SS-MWI

Item	Question/feature	References	Status	N/A	Support
F1	Support of SS-MWI	ECMA-242	o		Yes[ ] No[ ]
F2	Support of procedures at a Home PINX for interaction with SS-MWI	6.11.24.1	F1: o	[ ]	Yes[ ] No[ ]
F3	Support of procedures at a previous Visitor PINX / Visitor PINX for interaction with SS-MWI	6.11.24.2 6.11.24.3	F1: m	[ ]	m:Yes[ ]

Annex B

(informative)

Examples of Message Sequences

This annex describes some typical message flows for SS-PUMR. The following conventions are used in the figures of this annex.

1. The following notation is used:

   ▬ ▬ ▬ ▬ ▬ ▬▶   Call independent signalling connection message containing SS-PUMR information
   — — — — — —▶   Call independent signalling connection message without SS-PUMR information
   - - - - - - - - - - - -▶   Symbolic primitive carrying SS-PUMR information

   xxx.inv   Invoke APDU for operation xxx
   xxx.rr   Return result APDU for operation xxx

2. The figures show messages exchanged via Protocol Control between PINXs involved in SS-PUMR. Only messages relevant to SS-PUMR are shown.

3. Only the relevant information content (e.g. remote operation APDUs, information elements) is listed below each message name. The Facility information elements containing remote operation APDUs are not explicitly shown. Information with no impact on SS-PUMR is not shown.

4. Some interactions with users are included in the form of symbolic primitives. The actual protocol at the terminal equipment interface is outside the scope of this Standard.

B.1 Successful registration

B.1.1 Successful registration using the PUM number

Figure B.1 shows an example message flow of successful registration in the case where the PUM number is provided for identification.
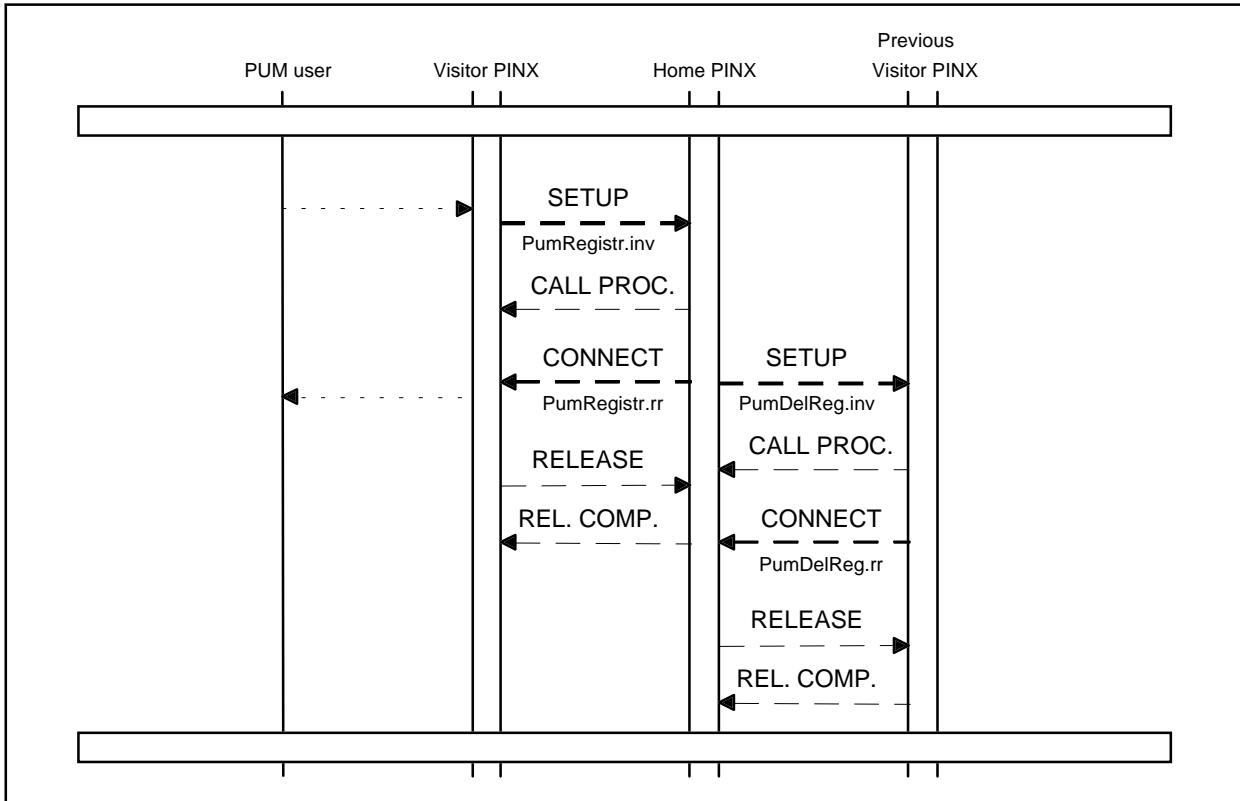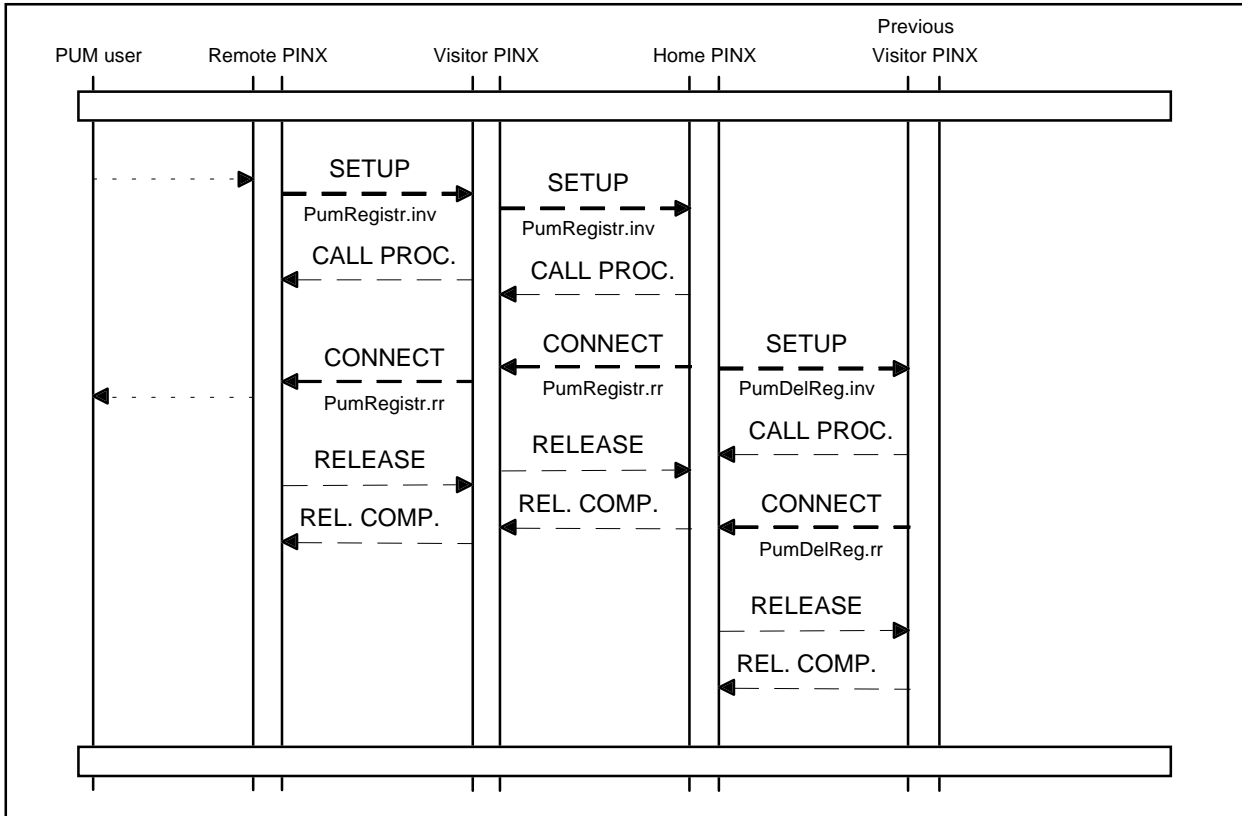

Figure B.1 - Example message flow for registration using the PUM number

NOTE

It depends on the contents of the original registration request from Visitor to Home PINX whether the previous OutCall registration(s) shall be deleted. A previous InCall registration will in any case be deleted. Therefore, more than one message with pumDelReg invoke APDUs to more than one Previous Visitor PINX might be required.

B.1.2   Successful registration with additional enquiry to a Directory PINX

Figure B.2 shows an example message flow of successful registration in the case where an alternative identifier (rather than the PUM number) is provided for identification, i.e. using an additional procedure with enquiry to the Directory PINX to obtain the PUM user's PISN number.
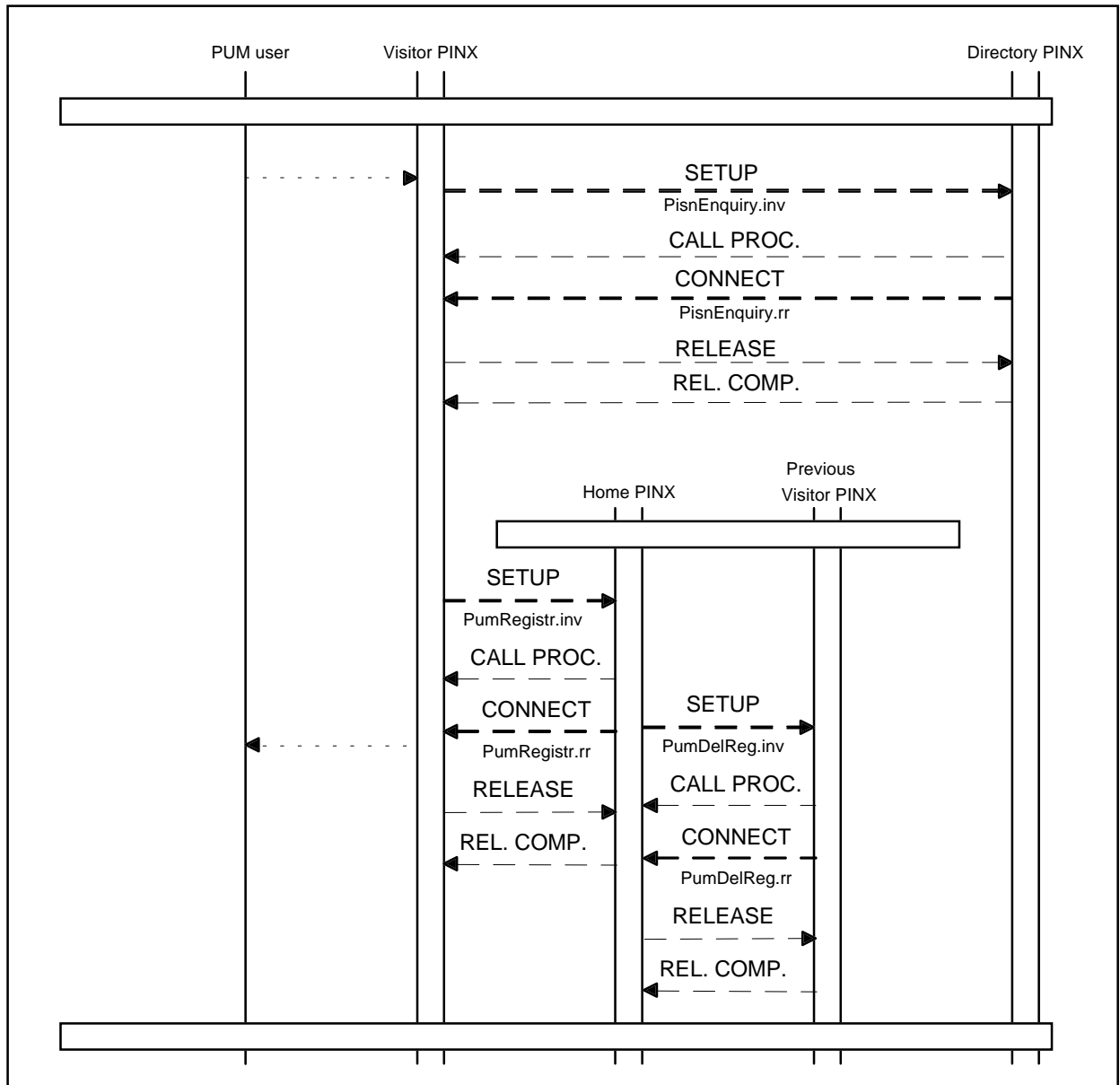
Figure B.2 - Example message flow for registration with enquiry to a Directory PINX

NOTE

It depends on the contents of the original registration request from Visitor to Home PINX whether the previous OutCall registration(s) shall be deleted. A previous InCall registration will in any case be deleted. Therefore, more than one message with pumDelReg invoke APDUs to more than one Previous Visitor PINX might be required.

B.1.3 Successful registration invoked by a Remote PINX

Figure B.3 shows an example message flow of successful registration in the case where registration is invoked by a Remote PINX.

Figure B.3 - Example message flow for registration invoked by a Remote PINX

NOTE

It depends on the contents of the original registration request whether the previous OutCall registration(s) shall be deleted. A previous InCall registration will in any case be deleted. Therefore, more than one message with pumDelReg invoke APDUs to more than one Previous Visitor PINX might be required.

B.2   Successful de-registration

B.2.1   Successful local de-registration

Figure B.4 shows an example message flow of successful de-registration invoked by (a local PUM user at) the Visitor PINX.
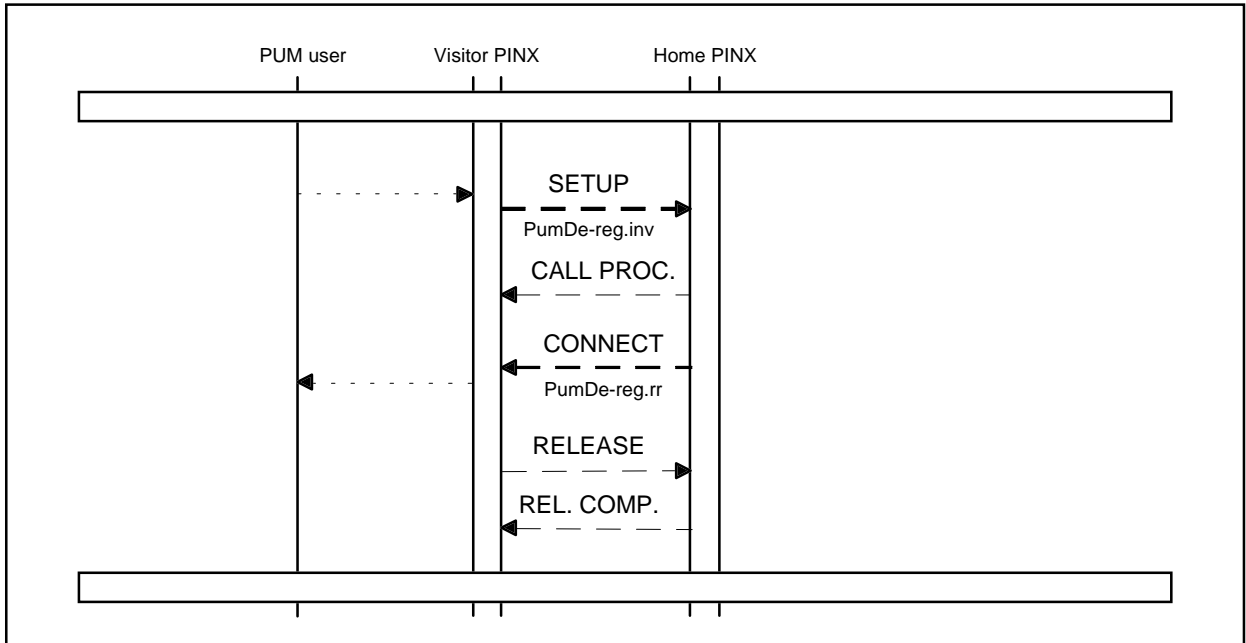
Figure B.4 - Example message flow for local de-registration

B.2.2 Successful forced de-registration

Figure B.5 shows an example message flow of successful forced de-registration of a particular session requested by an authorized user at the Visitor PINX.
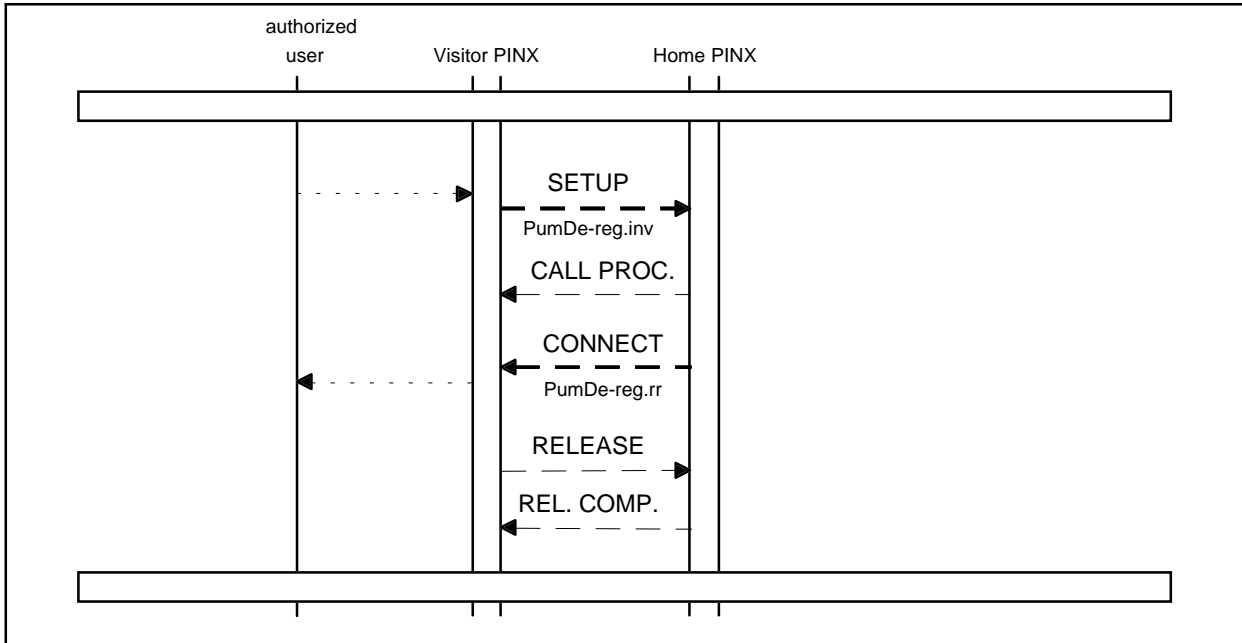
Figure B.5 - Example message flow for forced local de-registration

NOTE

In case of forced de-registration of all PUM users at a specific hosting address, more than one pumDe-reg invoke APDU to the Home PINX might be required.

B.2.3 Successful conditional de-registration

Figure B.6 shows an example message flow of successful conditional de-registration of a particular session.
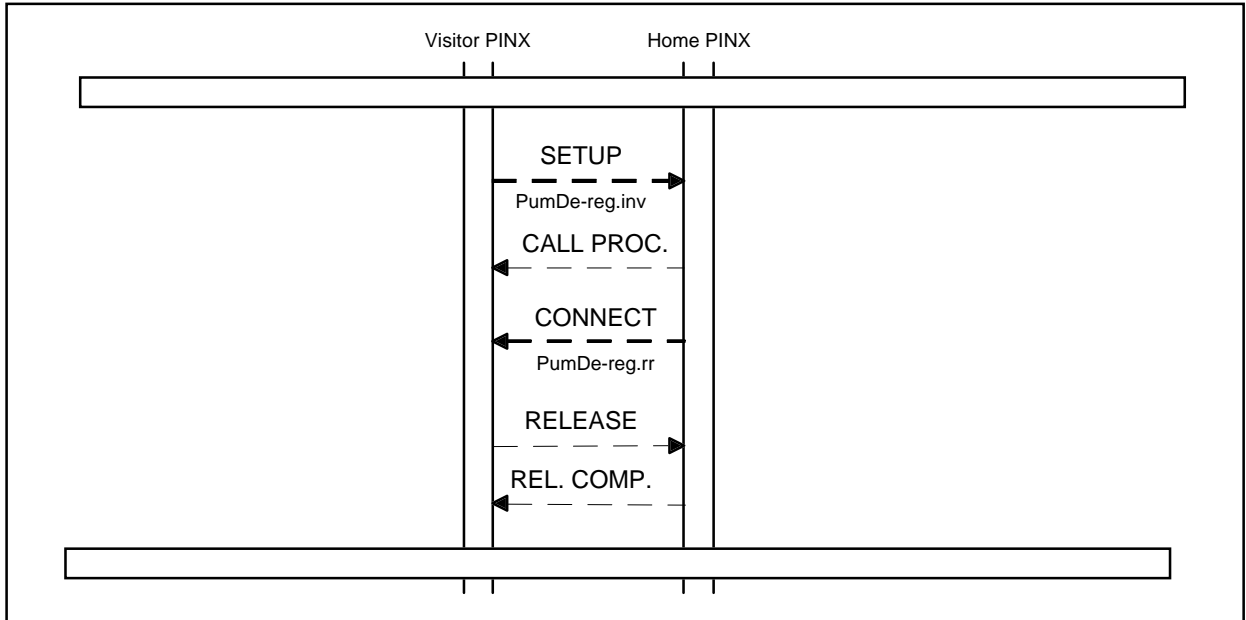
Figure B.6 - Example message flow for conditional de-registration

B.2.4 Successful remote de-registration

Figure B.7 shows an example message flow of successful de-registration invoked by (a PUM user at) a Remote PINX. De-registration can be invoked for a single / multiple / all registration sessions of a PUM-user.
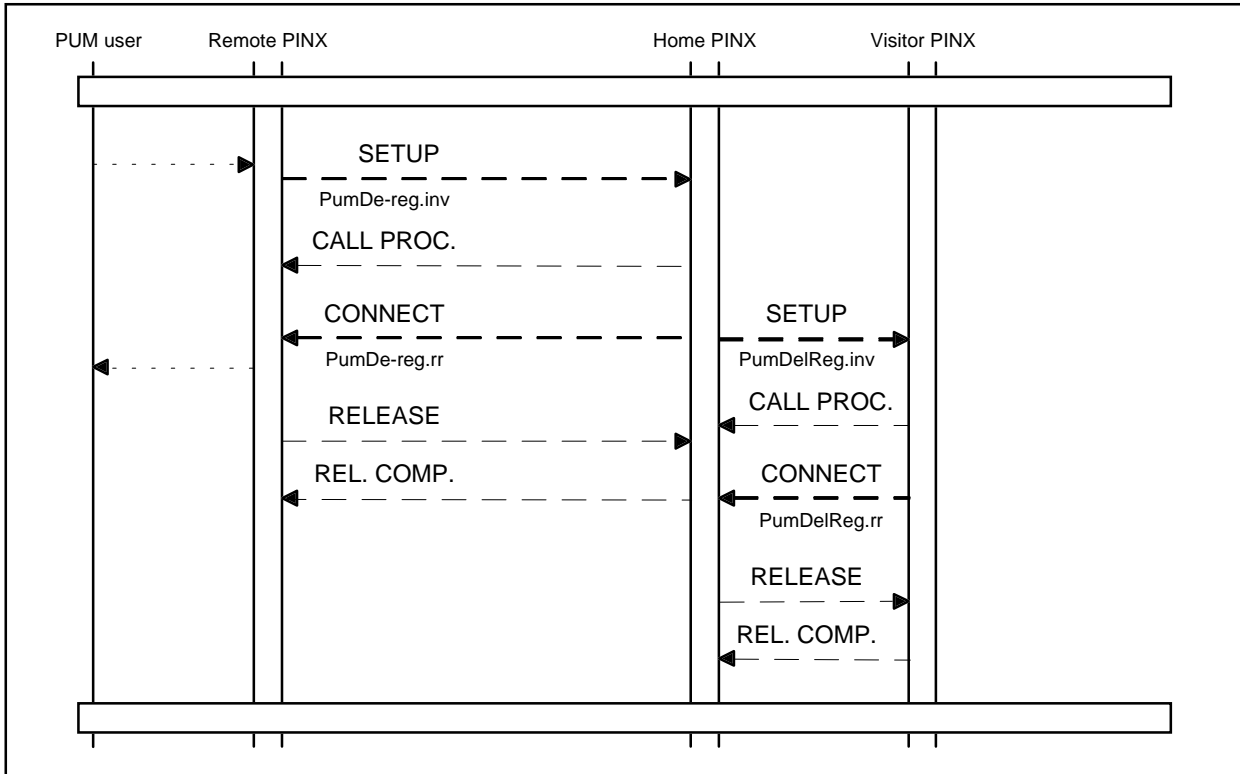
Figure B.7 - Example message flow for de-registration invoked by a Remote PINX

NOTE

More than one message with pumDelReg invoke APDUs to more than one Visitor PINX might be required.

The Remote PINX can also be a Visitor PINX.

An enquiry procedure to a Directory PINX may be invoked (if necessary) prior to sending the pumDe-reg invoke APDU to the Home PINX.

B.3   Successful interrogation

B.3.1   Successful local interrogation of registration sessions

Figure B.8 shows an example message flow of successful interrogation of registration sessions at the Visitor PINX invoked by a local PUM user.
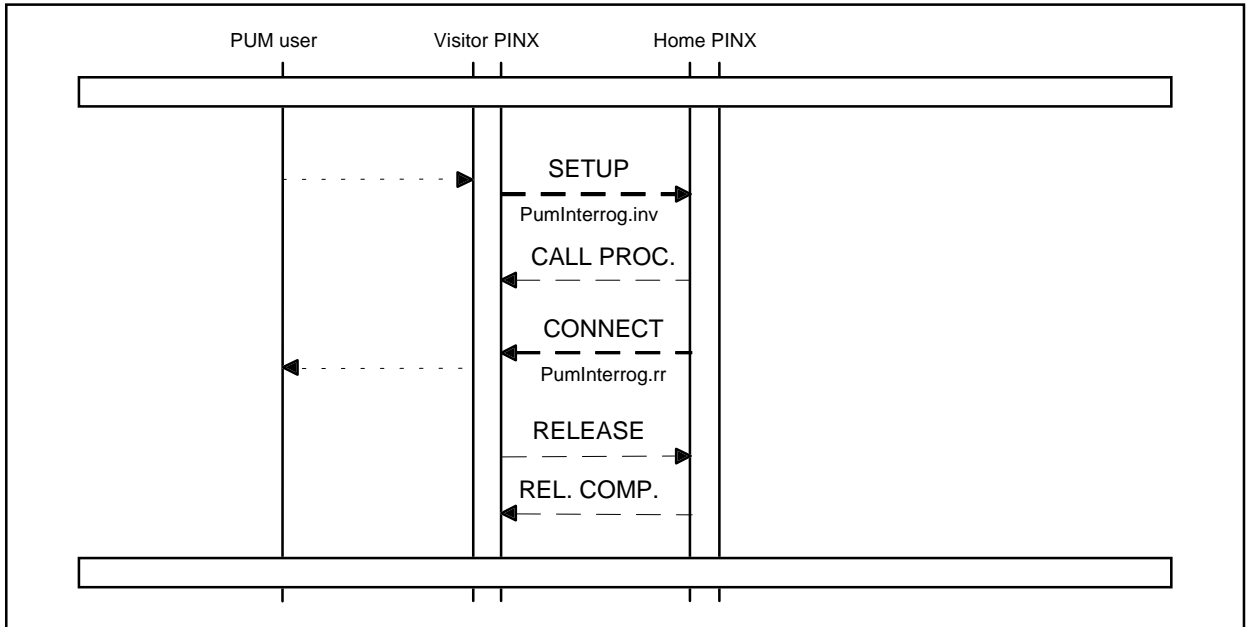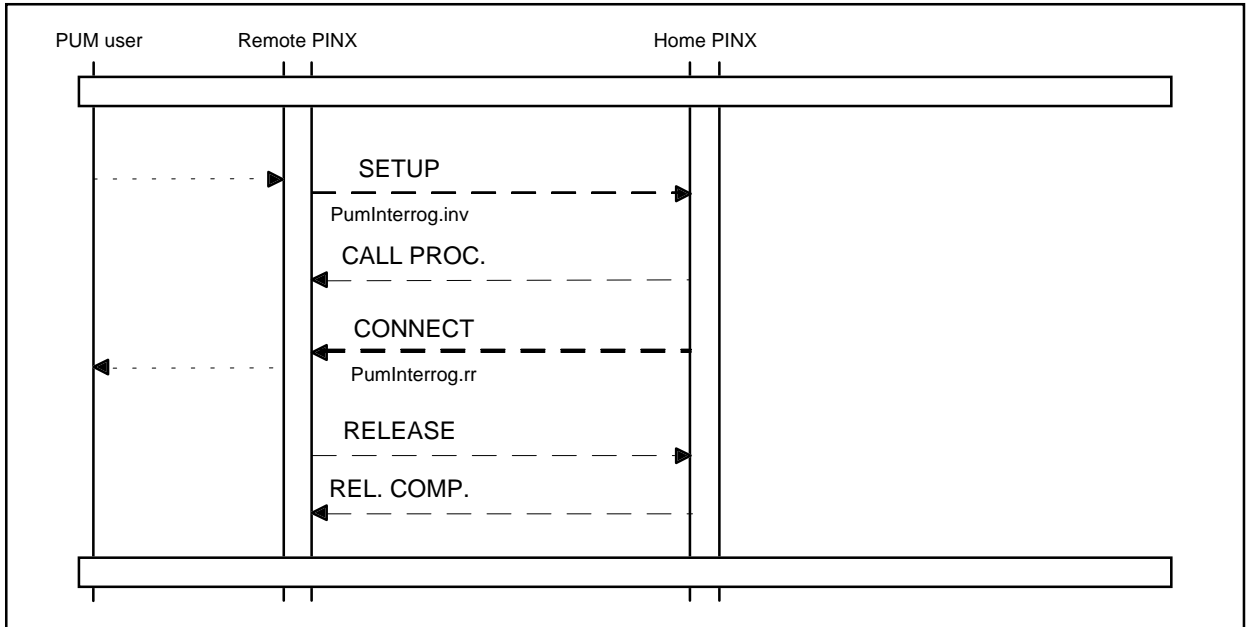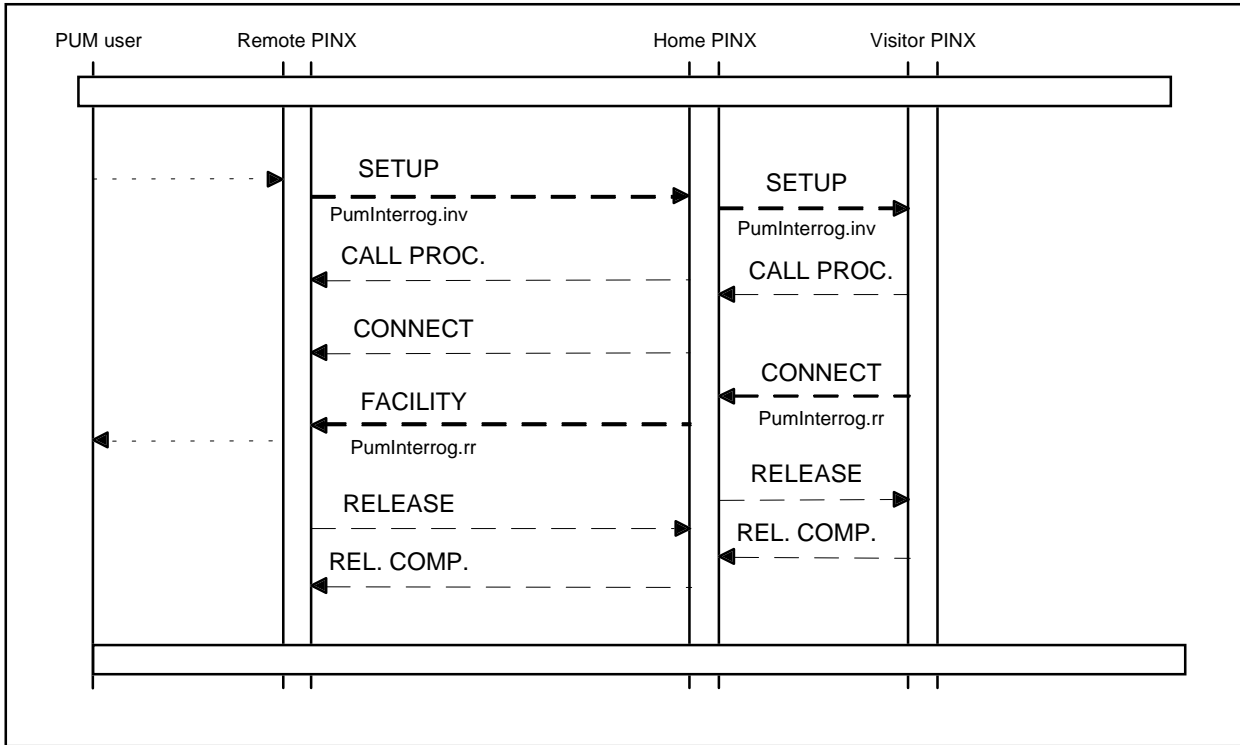

Figure B.8 - Example message flow for successful interrogation of registration sessions

B.3.2 Successful local interrogation of registration sessions including session parameters

Figure B.9 shows an example message flow of successful interrogation of registration sessions including session parameters at the Visitor PINX invoked by a local PUM user.
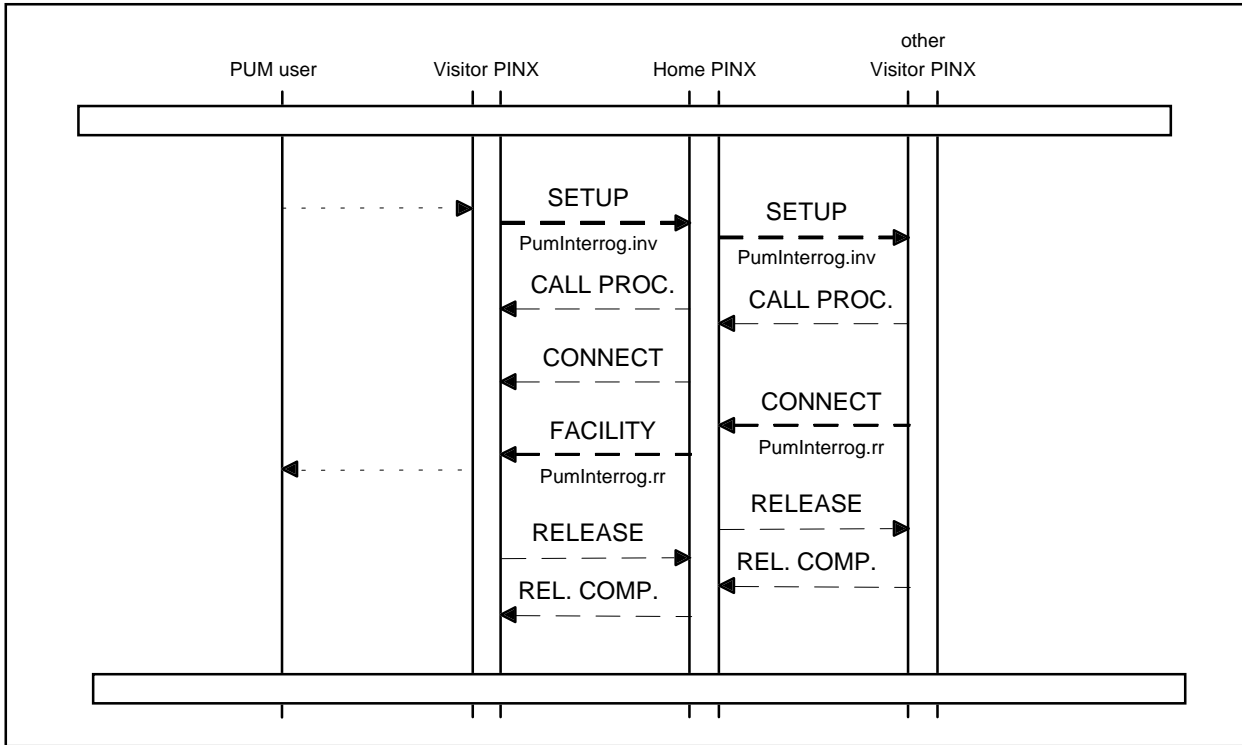
**Figure B.9 - Example message flow for successful interrogation of registration sessions
and session parameters**

NOTE

More than one message with pumInterrog invoke APDUs to more than one other Visitor PINX might be required.

B.3.3 Successful remote interrogation of registration sessions

Figure B.10 shows an example message flow of successful interrogation of registration sessions at the Home PINX invoked by a remote PUM user.

Figure B.10 - Example message flow for successful remote interrogation of registration sessions

NOTE

An enquiry procedure to a Directory PINX may be invoked (if necessary) prior to sending the pumInterrog invoke APDU to the Home PINX.

B.3.4 **Successful remote interrogation of registration sessions and/or session parameters**

Figure B.11 shows an example message flow of successful interrogation of registration sessions and/or session parameters at the Home PINX invoked by a remote PUM user.

**Figure B.11 - Example message flow for successful interrogation of registration sessions
and/or session parameters**

NOTE

More than one message with pumInterrog invoke APDUs to more than one other Visitor PINX might be required.

The Remote PINX can also be a Visitor PINX.

An enquiry procedure to a Directory PINX may be invoked (if necessary) prior to sending the pumInterrog invoke APDU to the Home PINX.

B.4 Interaction with SS-CTLR

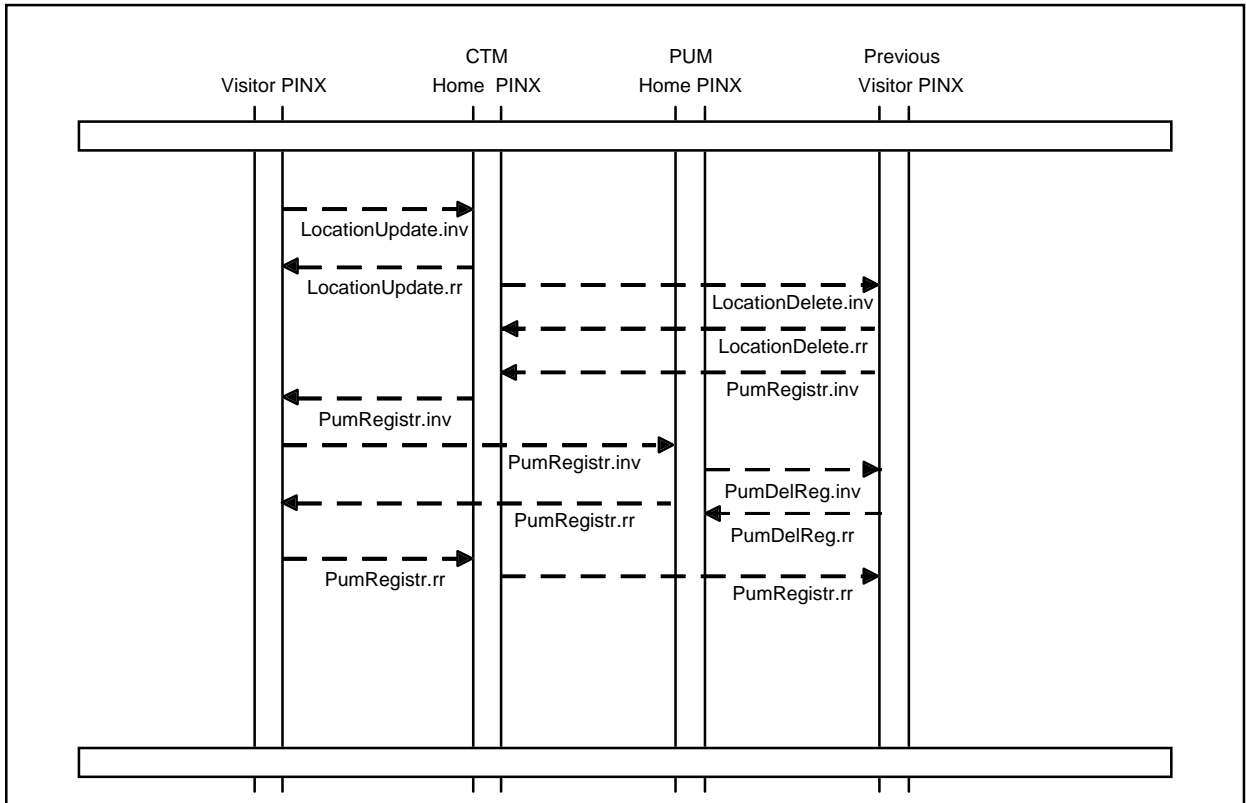Figure B.12 shows an example message flow of interactions of SS-PUMR with SS-CTLR.

Figure B.12 - Example message flow for interactions with CTLR

NOTE

The locationDelete return result APDU and the pumRegistr invoke ADPU from the previous Visitor PINX may be sent within one message.

Annex C

(informative)

Specification and Description Language (SDL) representation of procedures

The diagrams in this annex use the Specification and Description Language defined in ITU-T Recommendation Z.100 (1993).

Each diagram represents the behaviour of a SS-PUMR Supplementary Service Control entity at a particular type of PINX. In accordance with the protocol model described in ECMA-165, the Supplementary Service Control entity uses, via the Coordination Function, the services of Generic Functional Procedures Control and Basic Call Control.

Where an output symbol represents a primitive to the Coordination Function, and that primitive results in a message being sent, the output symbol bears the name of the message and any remote operations APDU(s) or notification(s) contained in that message.

Where an input symbol represents a primitive from the Coordination Function, and that primitive is the result of a message being received, the input symbol bears the name of the message and any remote operations APDU(s) or notification(s) contained in that message.

The following abbreviations are used:

inv.   invoke APDU

res.   return result APDU

err.   return error APDU

rej.   reject APDU

C.1   SDL representation of SS-PUMR at the Visitor PINX

Figures C-1 show the behaviour of a SS-PUMR Supplementary Service Control entity within the Visitor PINX.

The source of each input signal is identified in a comment attached to the input symbol and the destination of each output signal is identified in an SDL "TO" statement within the output symbol.
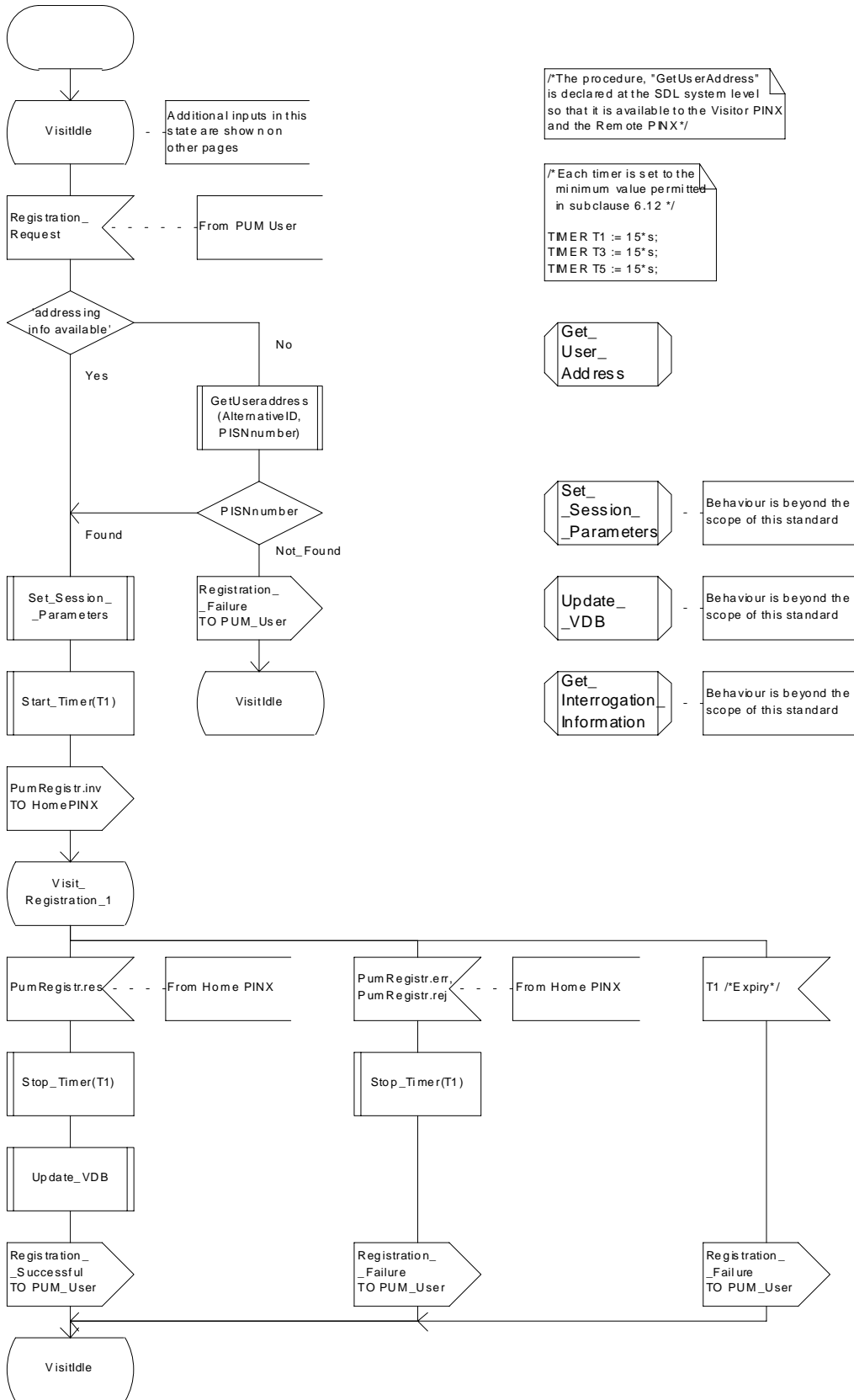
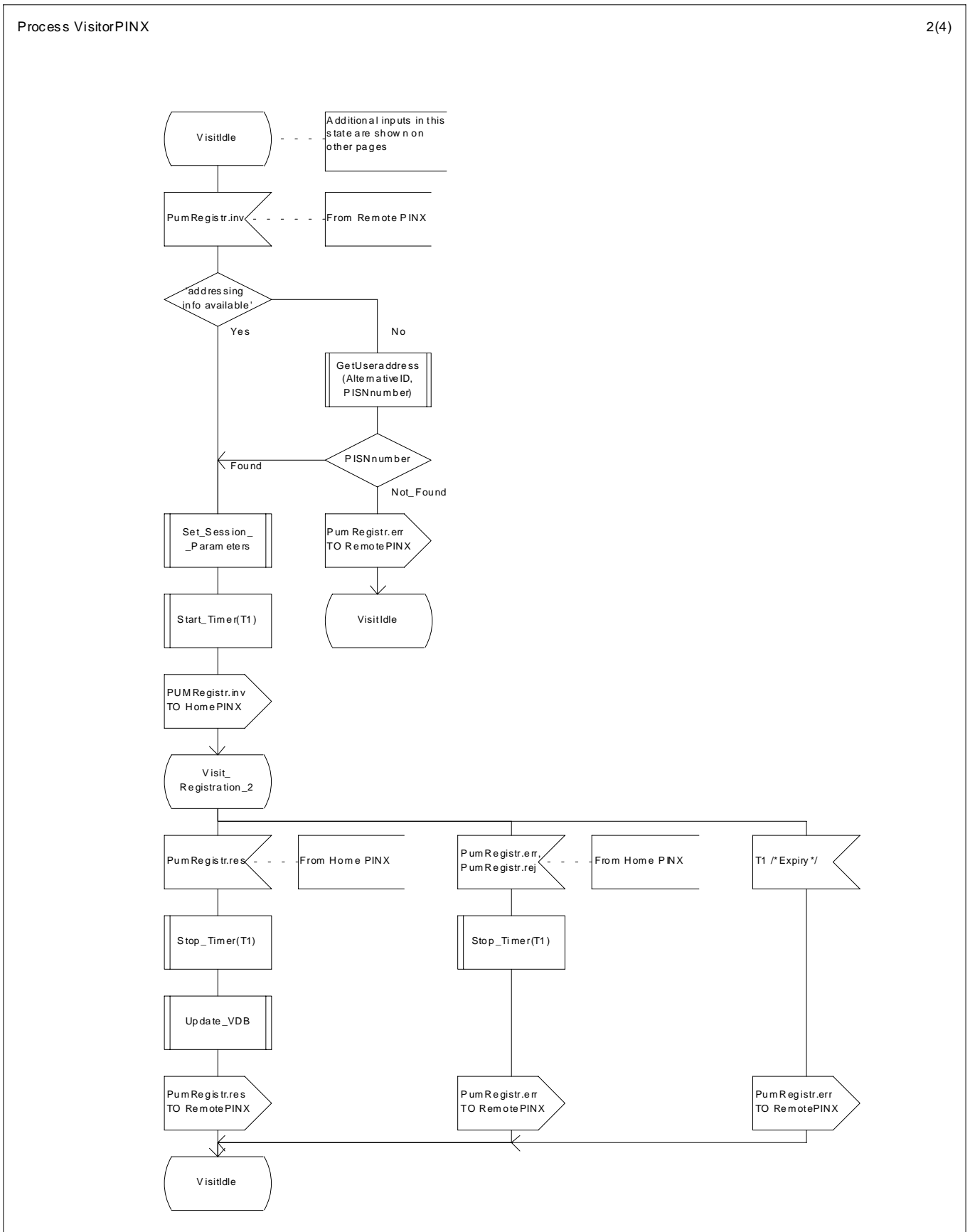**Figure C-1.1a - SDL Representation of SS-PUM Registration at the Visitor PINX**

Process VisitorPINX   2(4)

VisitIdle - - - - - Additional inputs in this
                     state are shown on
                     other pages

PumRegistr.inv - - - - - From Remote PINX

'addressing
info available'

Yes   No

                    GetUseraddress
                    (AlternativeID,
                    PISNnumber)

                    PISNnumber

Found   Not_Found

Set_Session_   PumRegistr.err
_Parameters   TO RemotePINX

Start_Timer(T1)   VisitIdle

PUMRegistr.inv
TO HomePINX

Visit_
Registration_2

PumRegistr.res - - - From Home PINX   PumRegistr.err, - - - From Home PINX   T1 /*Expiry*/
                                        PumRegistr.rej

Stop_Timer(T1)   Stop_Timer(T1)

Update_VDB

PumRegistr.res   PumRegistr.err   PumRegistr.err
TO RemotePINX   TO RemotePINX   TO RemotePINX

VisitIdle

Figure C-1.1b - SDL Representation of SS-PUM Registration at the Visitor PINX (cont.)

Process VisitorPINX   3(4)

VisitIdle

Additional inputs in this state are shown on other pages

De-registration request ⟵ - - From PUM User

Start_Timer(T3)

PumDeReg.inv TO HomePINX

Visit_De_Reg

PumDeReg.res ⟵ - - From Home PINX

Stop_Timer(T3)

Update_VDB

De_Registration_ _Successful TO PUM_User

VisitIdle

PumDelReg.inv ⟵ - - From Home PINX

'delete registration possible'
— No
Yes

Update_VDB

PumDelReg.res TO HomePINX

PumDelReg.err TO Home PINX

VisitIdle

PumDeReg.err, PumDeReg.rej ⟵ - - From Home PINX

T3 /* Expiry */ ⟵

Stop_Timer(T3)

De_Registration_ _Failure TO PUM_User

De_Registration_ Failure TO PUM_User

Figure C-1.2 - SDL Representation of SS-PUM De-registration at the Visitor PINX

Process VisitorPINX   4(4)
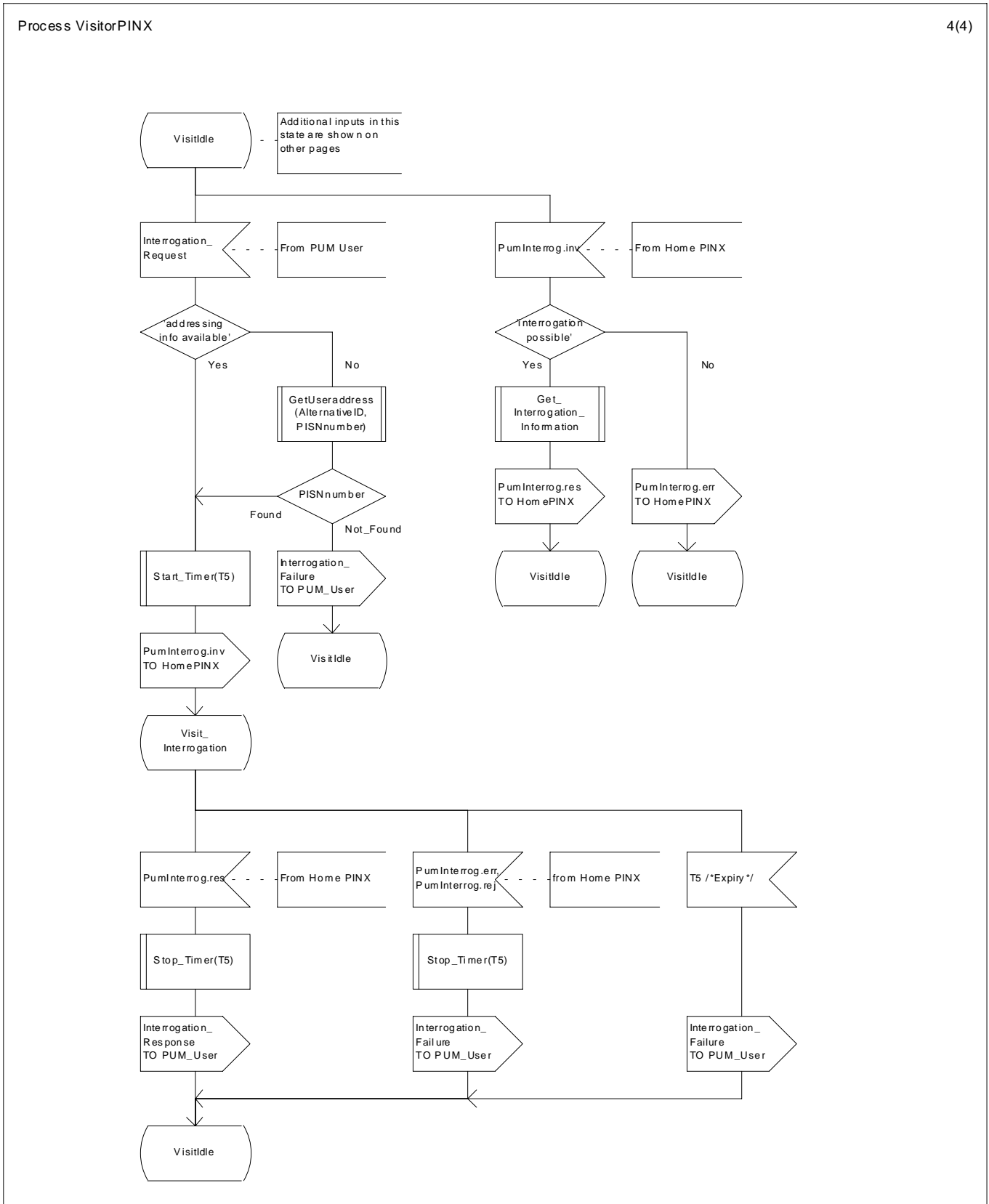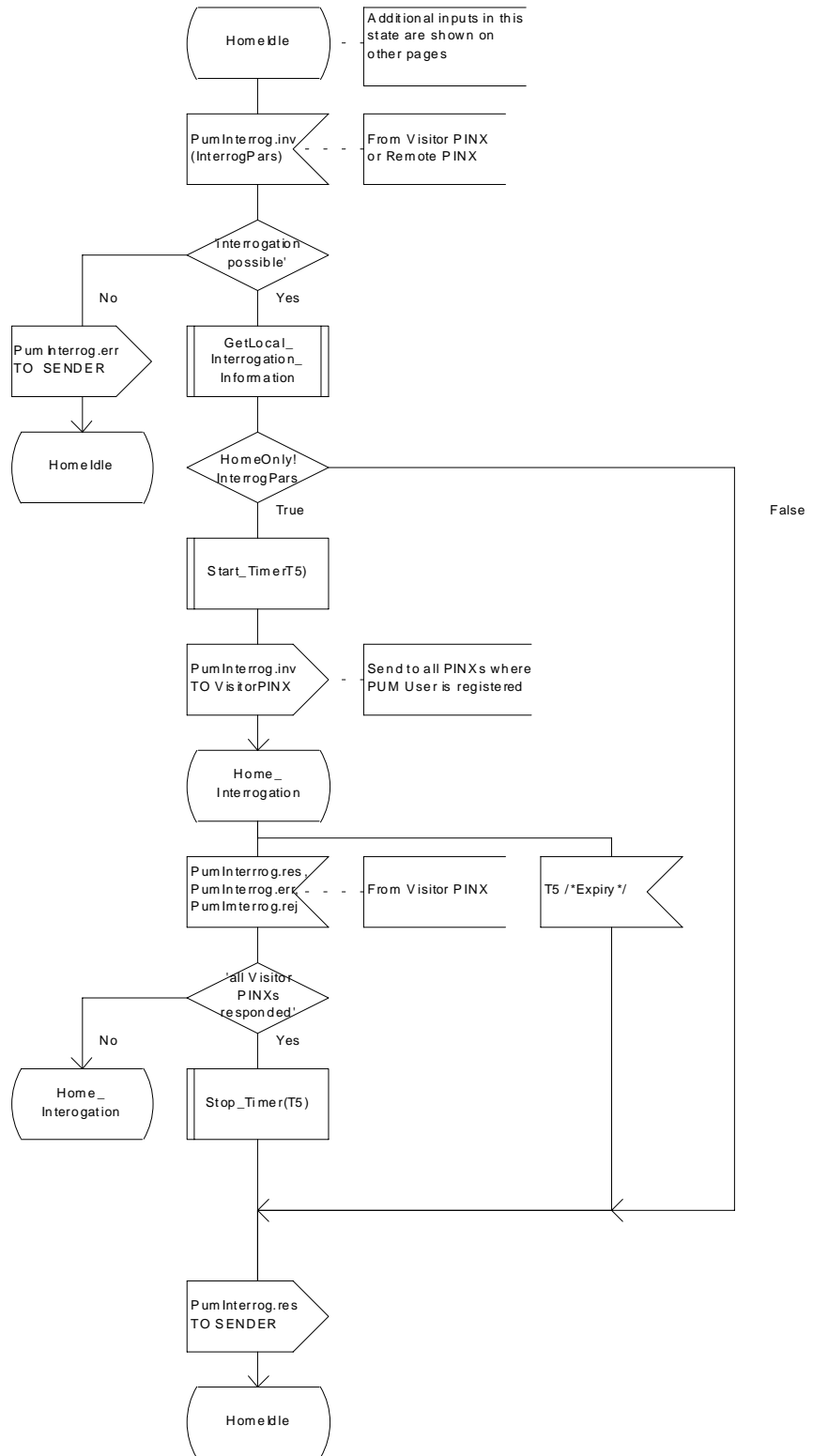

Figure C-1.3 - SDL Representation of SS-PUM Interrogation at the Visitor PINX

Figure C-1.4 shows the behaviour of a SS-PUMR Supplementary Service Control entity within a Visitor PINX for the determination of the PUM user's PISN address from an alternative identifier.

The source of each input signal is identified in a comment attached to the input symbol and the destination of each output signal is identified in an SDL "TO" statement within the output symbol.
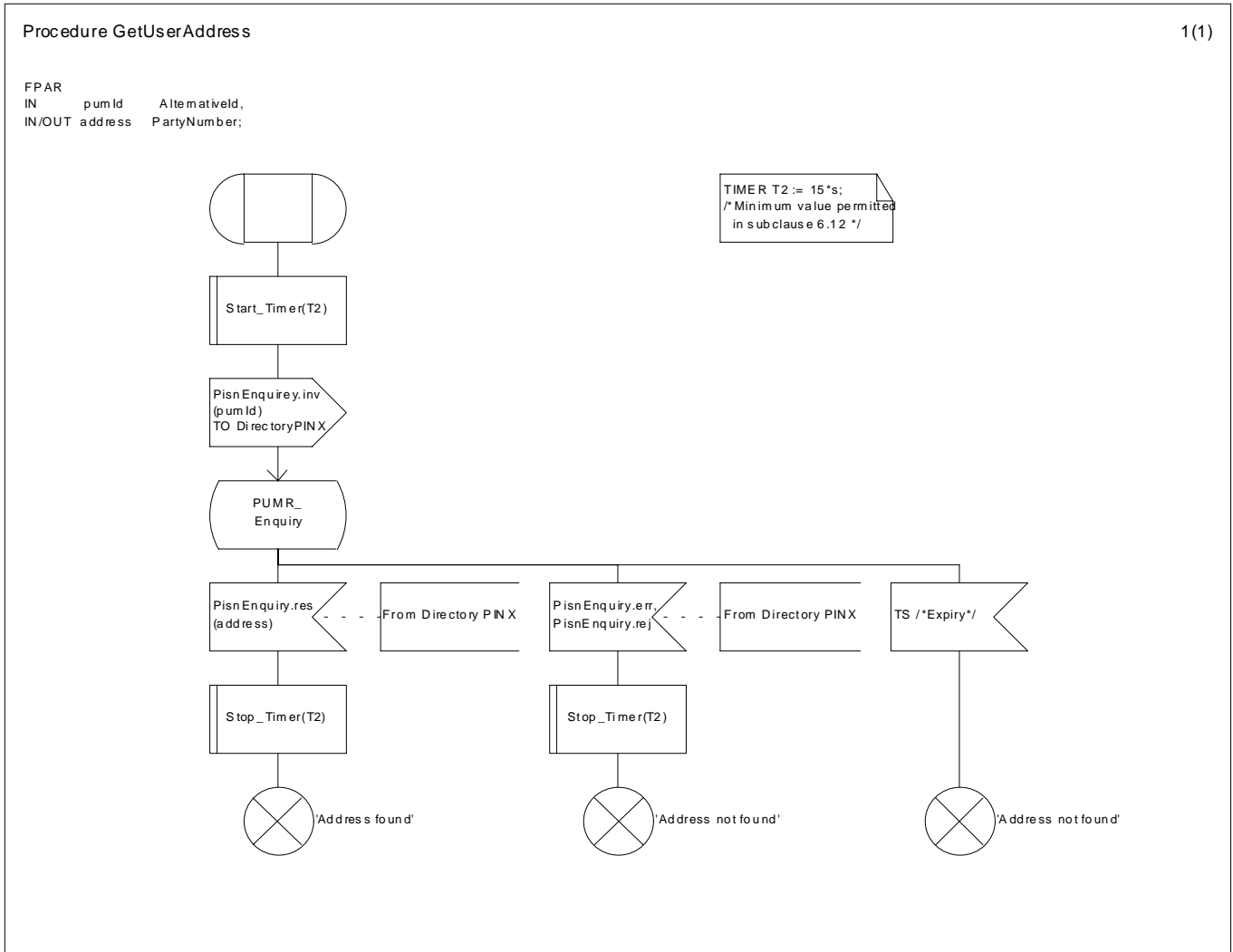
Figure C-1.4 - SDL Representation of conversion of alternative identifier to PISN number (also used by the Remote PINX)

C.2   SDL representation of SS-PUMR at the Home PINX

Figures C-2 show the behaviour of a SS-PUMR Supplementary Service Control entity within the Home PINX.

The source of each input signal is identified in a comment attached to the input symbol and the destination of each output signal is identified in an SDL "TO" statement within the output symbol.
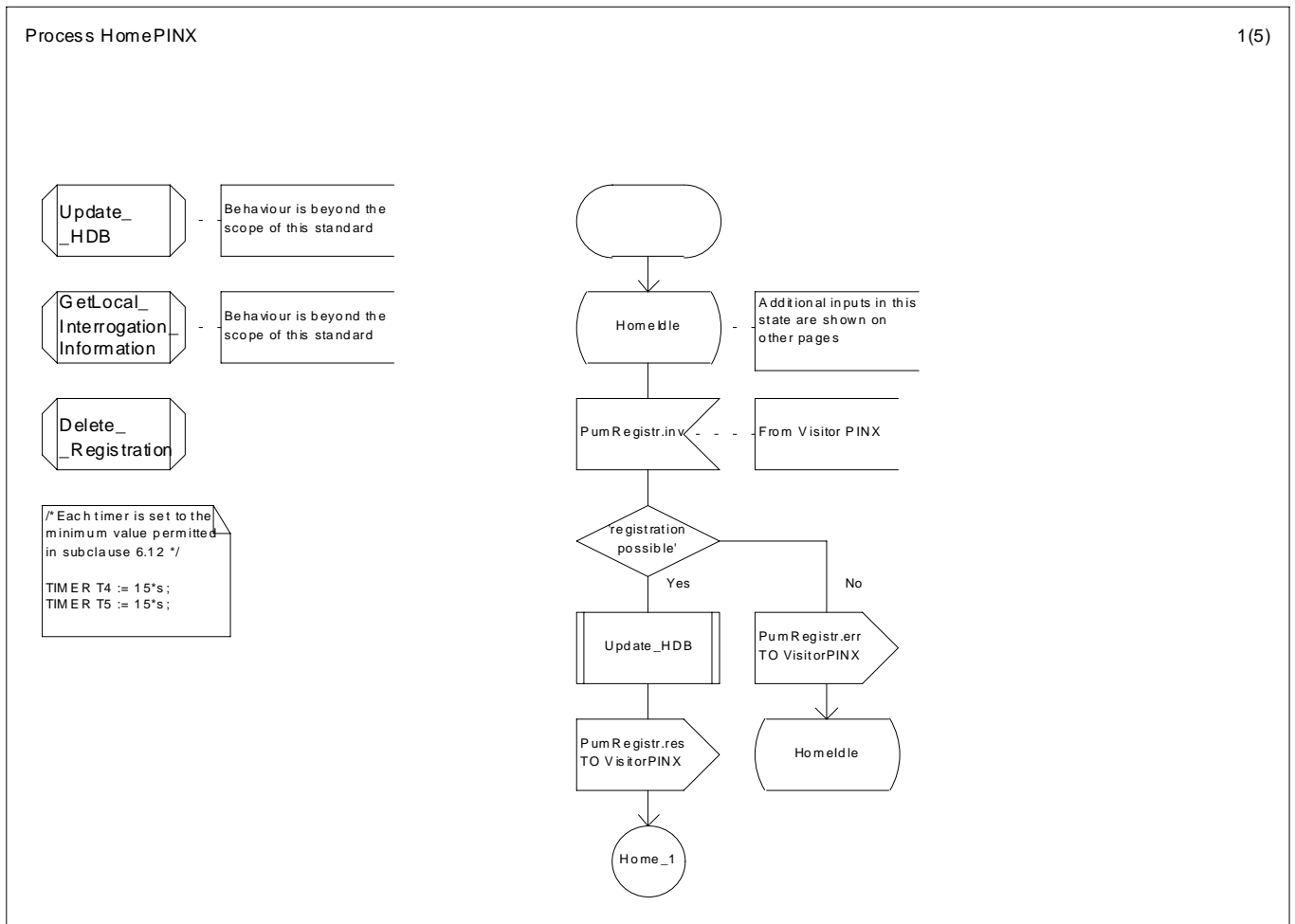
Figure C-2.1a - SDL Representation of SS-PUM Registration at the Home PINX
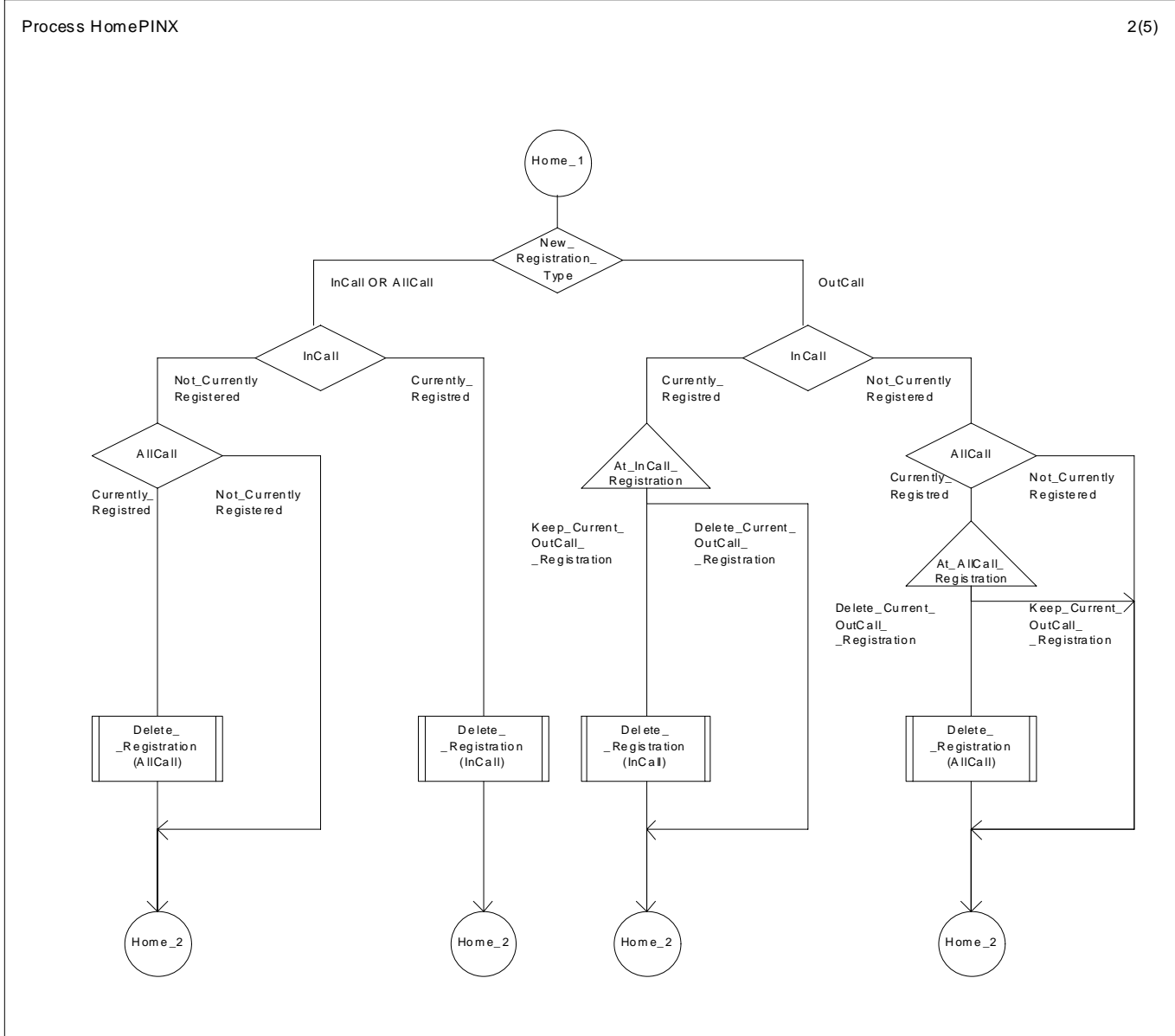
Figure C-2.1b - SDL Representation of SS-PUM Registration at the Home PINX (cont.)

Process HomePINX   3(5)
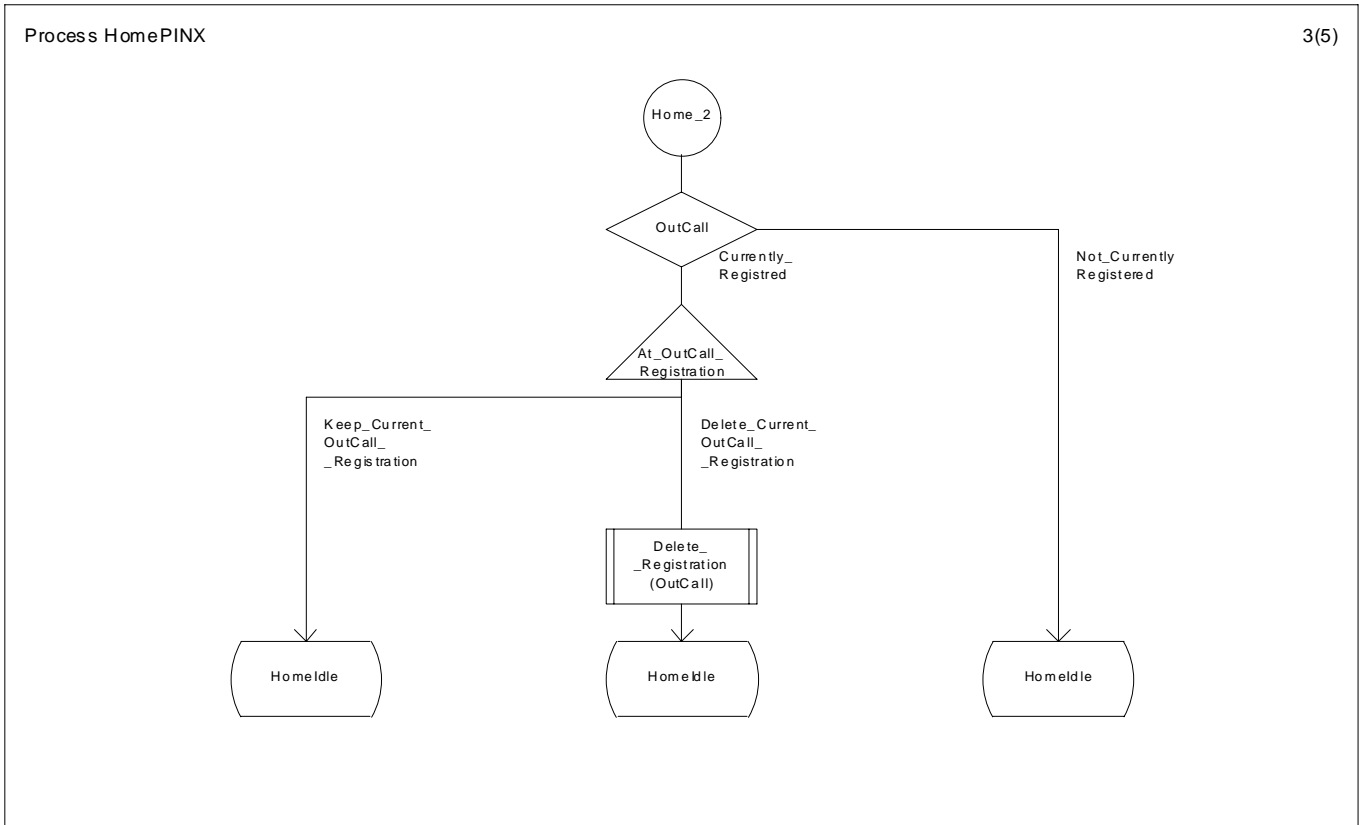
Figure C-2.1c - SDL Representation of SS-PUM Registration at the Home PINX (cont.)
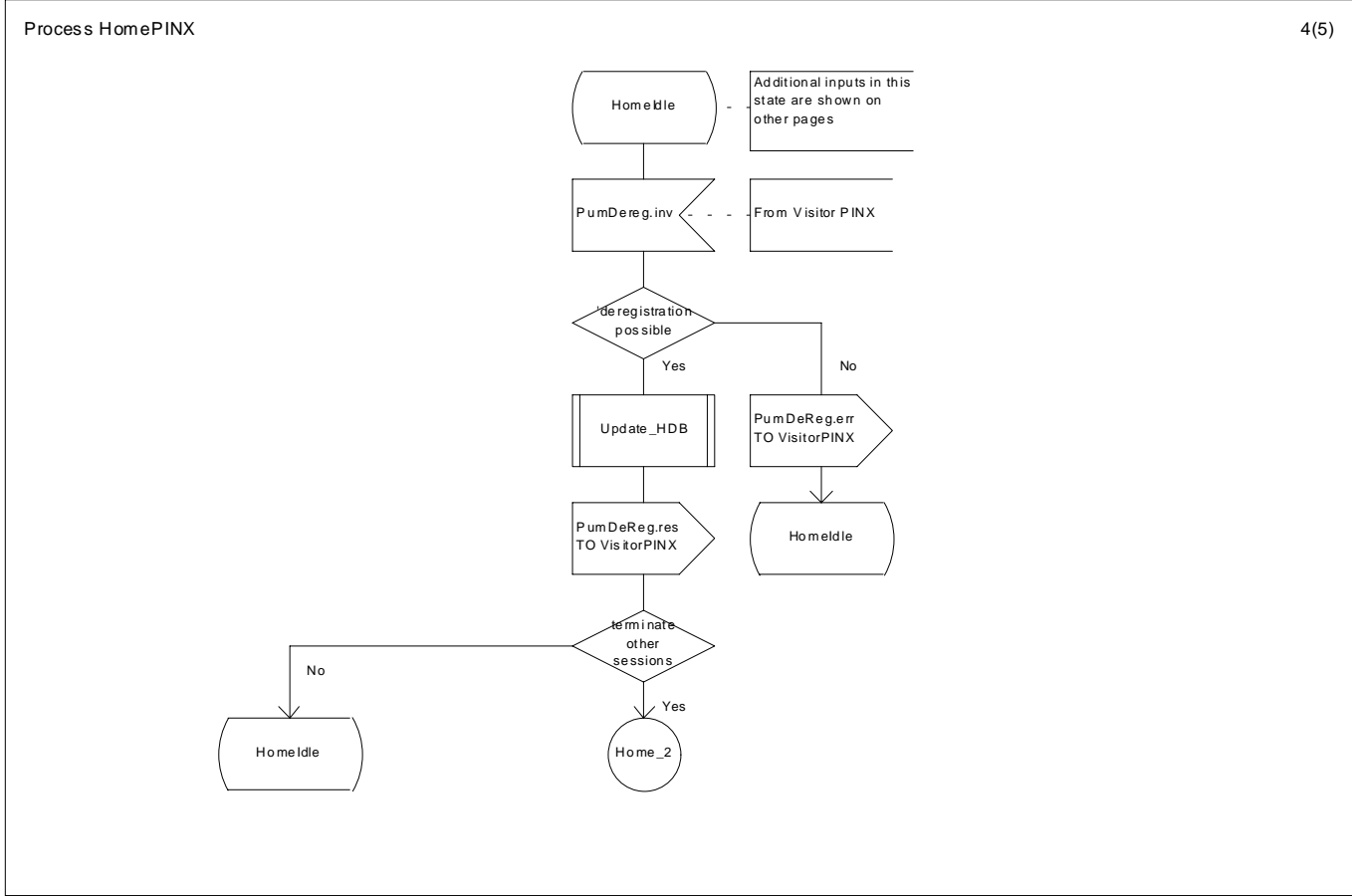
Figure C-2.2 - SDL Representation of SS-PUM De-registration at the Home PINX

Figure C-2.3 - SDL Representation of SS-PUM Interrogation at the Home PINX

Figure C-2.4 shows the behaviour of a SS-PUMR Supplementary Service Control entity within the Home PINX for deleting a PUM user's registrations at previous Visitor PINXs.
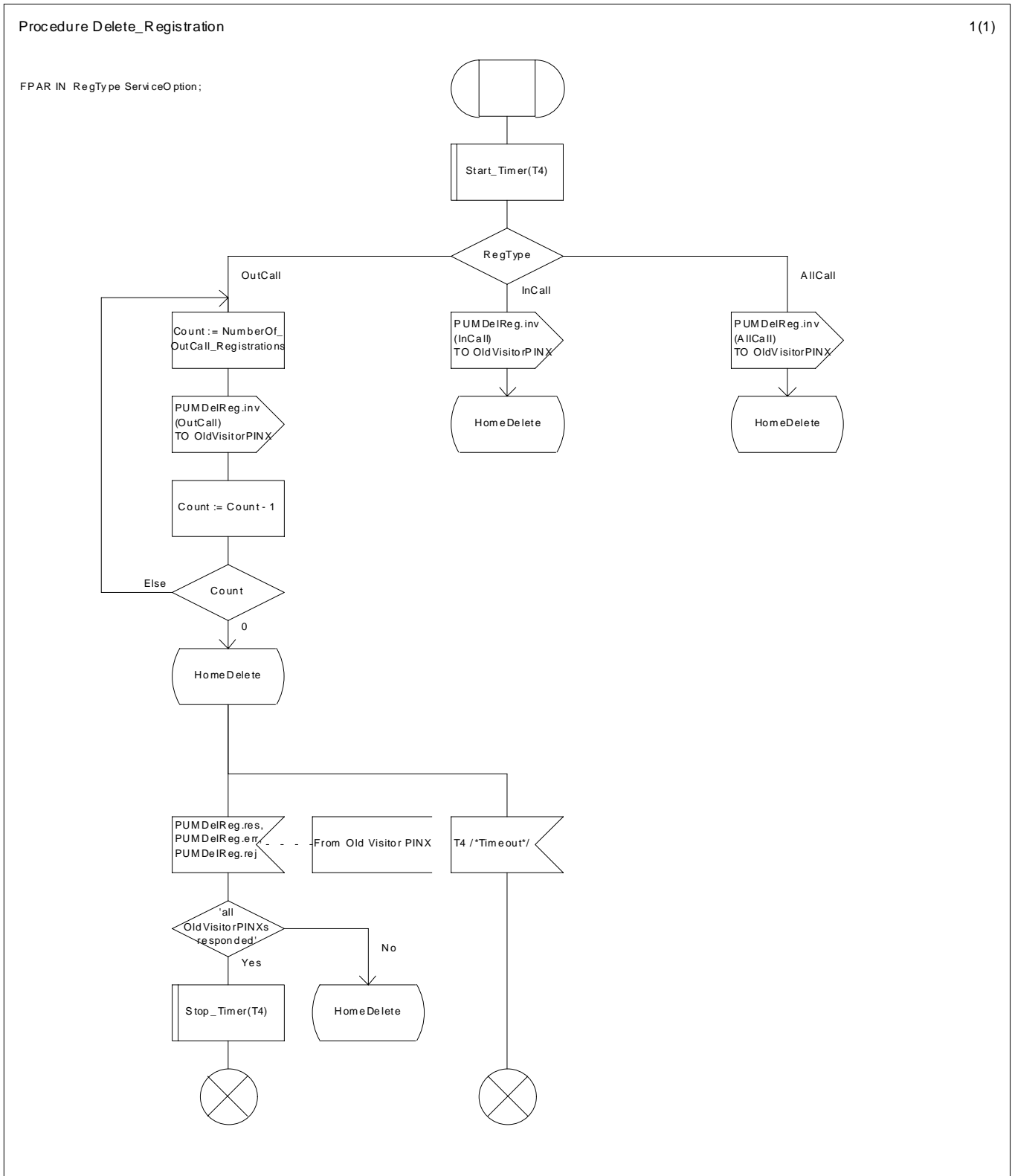
Figure C-2.4 - SDL Representation of SS-PUM registration deletion by the Home PINX

C.3 SDL representation of SS-PUMR at the previous Visitor PINX

Figure C-3 shows the behaviour of a SS-PUMR Supplementary Service Control entity within the previous Visitor PINX.

The source of each input signal is identified in a comment attached to the input symbol and the destination of each output signal is identified in an SDL "TO" statement within the output symbol.
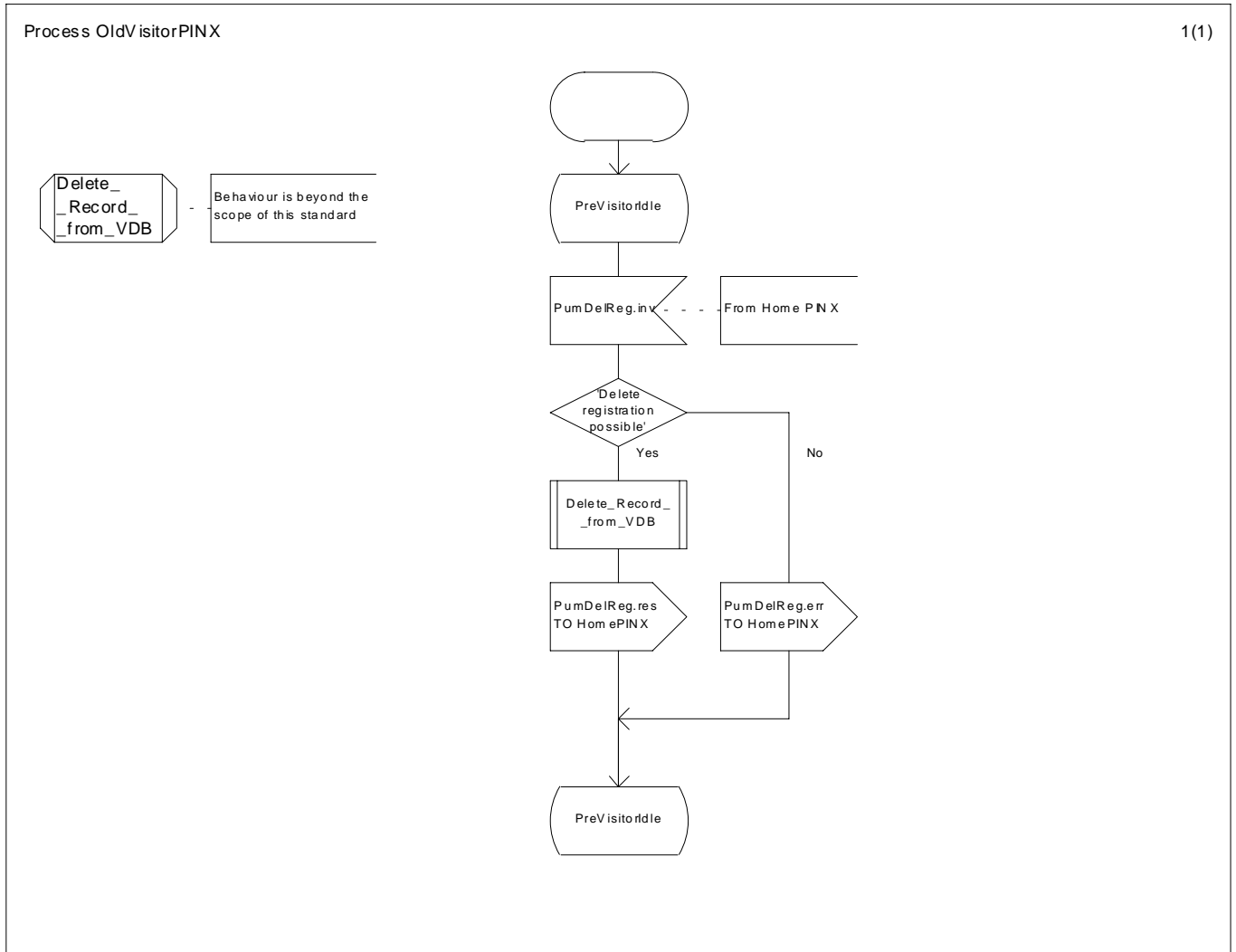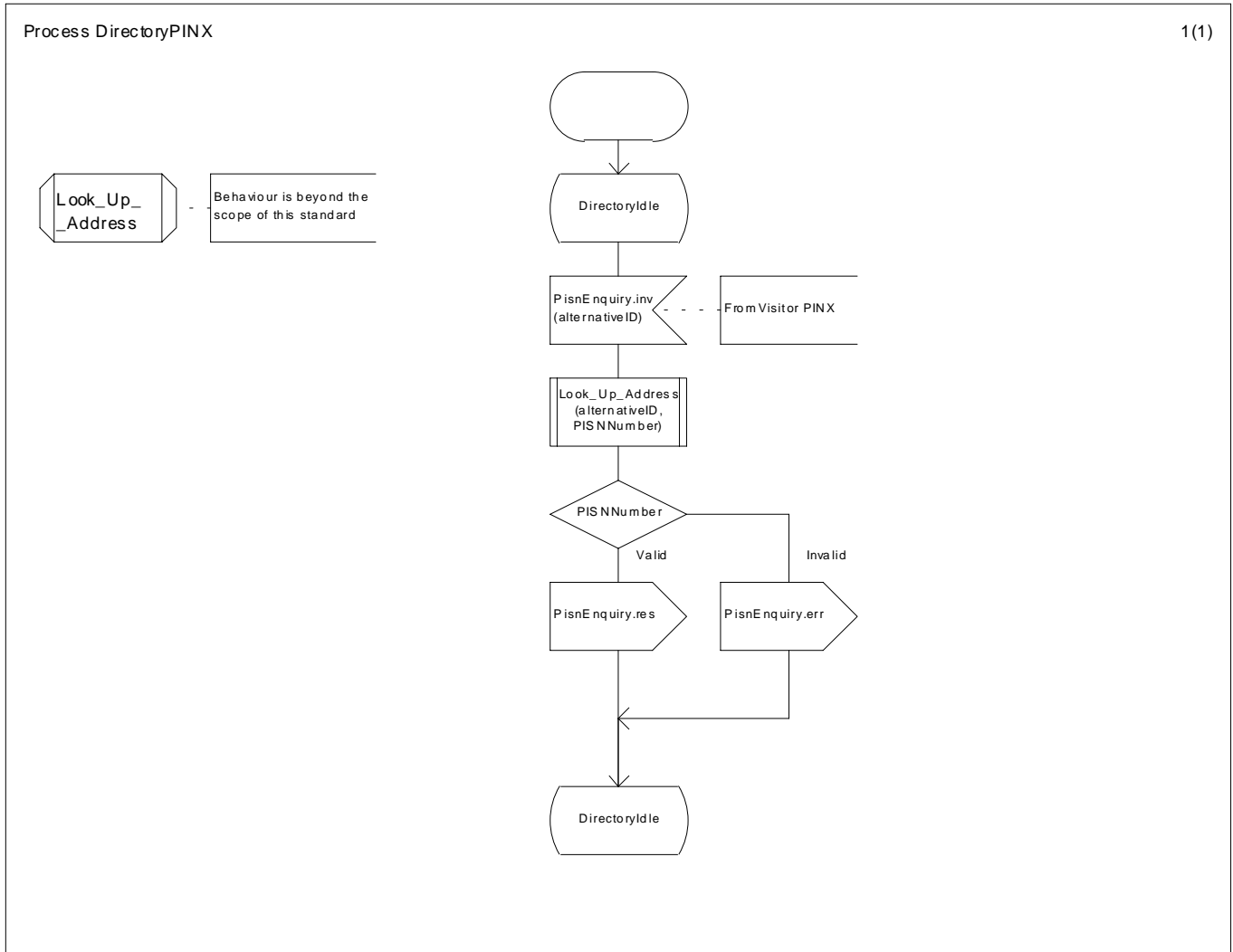

Figure C-3 - SDL Representation of SS-PUMR at the previous Visitor PINX

C.4 SDL representation of the PISN Enquiry procedure

Figure C-4 shows the behaviour of a SS-PUMR Supplementary Service Control entity within the Directory PINX.

The source of each input signal is identified in a comment attached to the input symbol and the destination of each output signal is identified in an SDL "TO" statement within the output symbol.

Figure C-4 - SDL Representation of PISN Enquiry at the Directory PINX

C.5   SDL representation of SS-PUMR at the Remote PINX

Figures C-5 show the behaviour of a SS-PUMR Supplementary Service Control entity within the Remote PINX.

The source of each input signal is identified in a comment attached to the input symbol and the destination of each output signal is identified in an SDL "TO" statement within the output symbol.
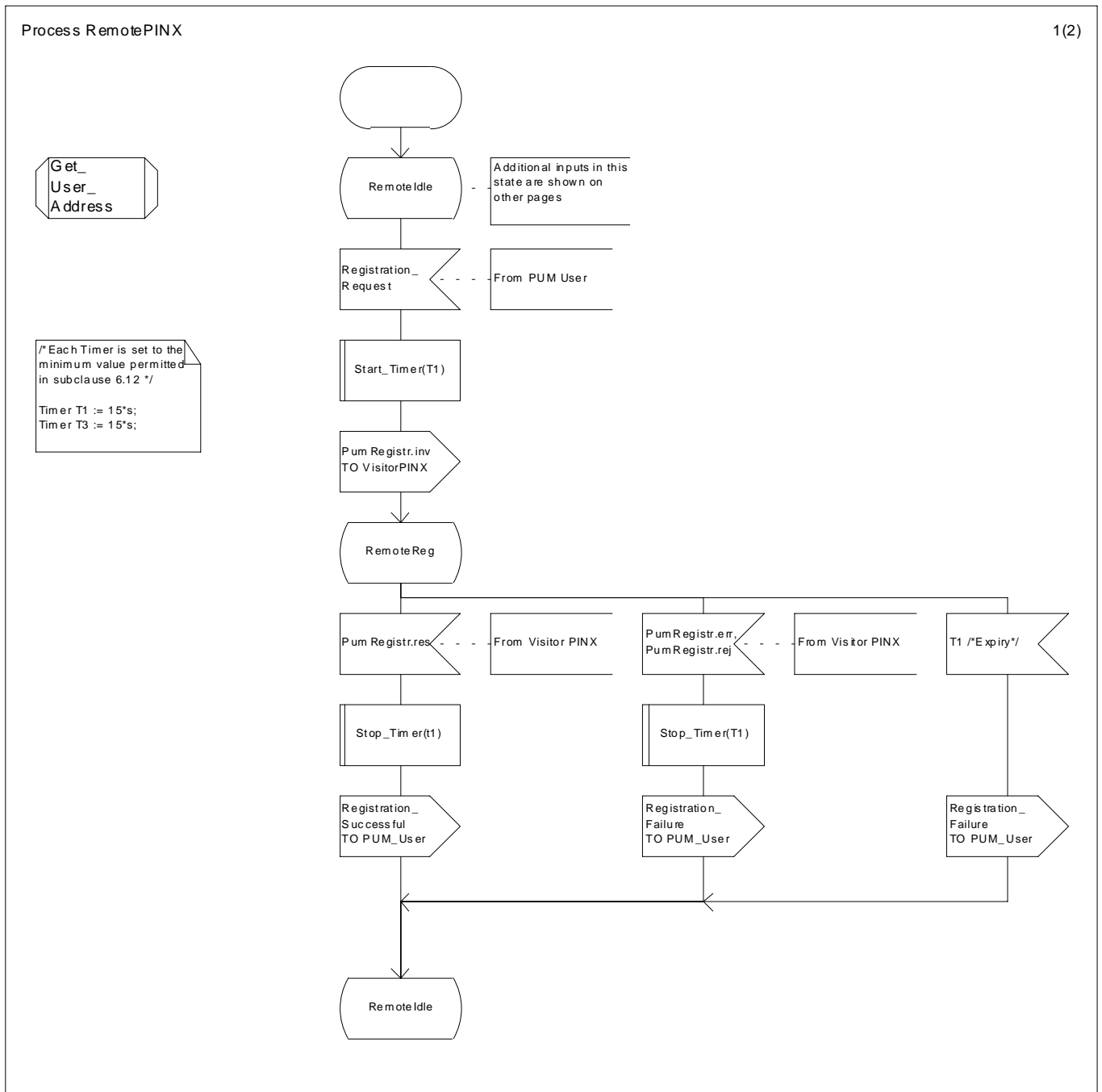

Figure C-5.1 - SDL Representation of SS-PUM Registration at the Remote PINX
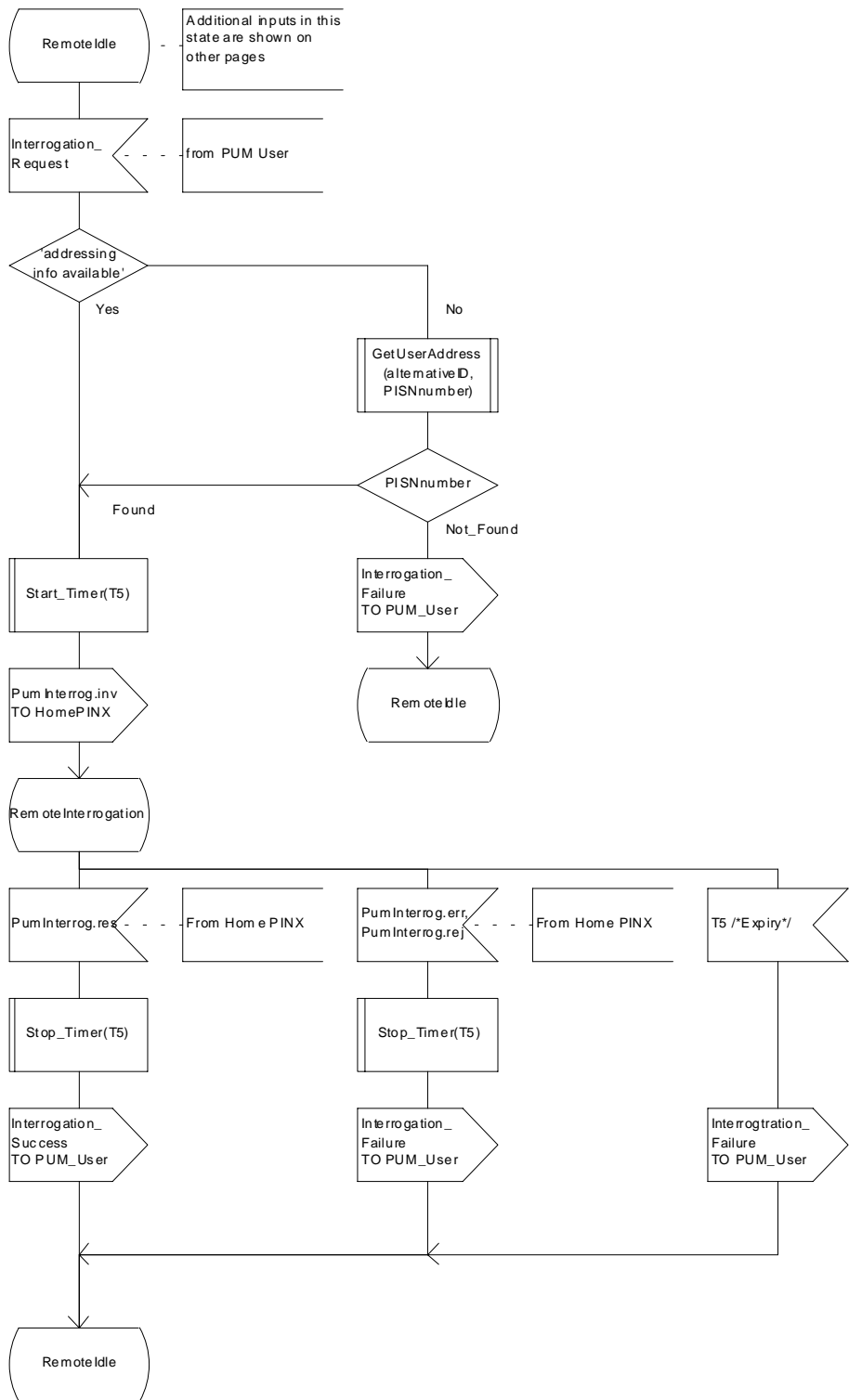
Figure C-5.2 - SDL Representation of SS-PUM Interrogation at the Remote PINX

Annex D

(informative)

Imported ASN.1 Definitions

Table D.1 is an extract from module General-Error-List in Q.950.

Table D.1 - Imported ASN.1 Definitions General-Errors

```
notAvailable                         ERROR ::= 3
          -- The requested supplementary service or function is not available for the
          -- basic service, and optionally for the served users ISDN number (e.g. temporary fault).

invalidServedUserNumber         ERROR ::= 6
          -- The served users number provided is not a valid number.

supplementaryServiceInteractionNotAllowed      ERROR ::= 10
          -- The performance of the requested supplementary service or function is prohibited
          -- by another supplementary service or function.
```

Table D.2 is an extract from module Call-Diversion-Operations in ISO/IEC 13873.

Table D.2 - Imported ASN.1 Definitions BasicService

```
BasicService    ::= ENUMERATED
                {allServices                    (0),
                 speech                         (1),
                 unrestrictedDigitalInformation (2),
                 audio3100Hz                    (3),
                 telephony                      (32),
                 teletex                        (33),
                 telefaxGroup4Class1            (34),
                 videotexSyntaxBased            (35),
                 videotelephony                 (36)}
```

Table D.3 is an extract from module Addressing-Data-Elements in ISO/IEC 11582.

Table D.3 - Imported ASN.1 Definitions Addressing-Data-Elements

PartyNumber	::= CHOICE {
	unknownPartyNumber [0] IMPLICIT NumberDigits,
	-- the numbering plan is the default numbering plan of the network.
	-- It is recommended that this value is used.
	publicPartyNumber [1] IMPLICIT PublicPartyNumber,
	-- the numbering plan is according to Recs. E.163 and E.164.
	dataPartyNumber [3] IMPLICIT NumberDigits,
	-- not used, value reserved.
	telexPartyNumber [4] IMPLICIT NumberDigits,
	-- not used, value reserved.
	privatePartyNumber [5] IMPLICIT PrivatePartyNumber,
	nationalStandardPartyNumber [8] IMPLICIT NumberDigits}
	-- not used, value reserved.
PublicPartyNumber	::= SEQUENCE {
	publicTypeOfNumber PublicTypeOfNumber,
	publicNumberDigits NumberDigits}
PrivatePartyNumber	::= SEQUENCE {
	privateTypeOfNumber PrivateTypeOfNumber,
	privateNumberDigits NumberDigits}
NumberDigits	::= NumericString (SIZE(1..20))
PublicTypeOfNumber	::= ENUMERATED {
	unknown (0),
	-- if used number digits carry prefix indicating type of number
	-- according to national recommendations
	internationalNumber (1),
	nationalNumber (2),
	networkSpecificNumber (3),
	-- not used, value reserved.
	subscriberNumber (4),
	abbreviatedNumber (6)}
	-- valid only for called party number at the outgoing access,
	-- network substitutes appropriate number
PrivateTypeOfNumber	::= ENUMERATED {
	unknown (0),
	level2RegionalNumber (1),
	level1RegionalNumber (2),
	pTNSpecificNumber (3),
	localNumber (4),
	abbreviatedNumber (6)}

Table D.4 is an extract from CTM-Location-Registration-Operations in ECMA-216.

Table D.4 - Imported ASN.1 Definition PisnEnquiry Operation

PisnEnquiry	::=	OPERATION	
		-- Sent from the Visitor PINX [or Remote PINX] to a Directory PINX.	
		ARGUMENT	PisnEnqArg
		RESULT	PisnEnqRes
		ERRORS	{ invalidServedUserNumber, unspecified, supplementaryServiceInteractionNotAllowed }
PisnEnqArg	::=	SEQUENCE	{ alternativeId AlternativeId,
			-- Can be a temporary identifier, e.g. Network Assigned
			-- Identity structure, or a fixed handset identifier.
			argExtension LrExtension OPTIONAL }
PisnEnqRes	::=	SEQUENCE	{ pisnNumber PartyNumber,
			-- The PISN number of the CTM user,
			-- always a Complete Number.
			dummyRes DummyRes OPTIONAL }
AlternativeId	::=	OCTET STRING (SIZE(1..20))	
LrExtension	::=	CHOICE	{ none NULL,
			extension [1] IMPLICIT Extension,
			sequOfExtn [2] IMPLICIT SEQUENCE OF Extension }
pisnEnquiry		PisnEnquiry	::= localValue 53

.