ITU Telecommunication Standardization Sector AVC-1117
Former Q.2&3/15 Rapporteur Meeting
Boston, ... 18-21 February 1997
Original: English

STUDY GROUP 16
CONTRIBUTION

Source:
M. Korpi, E. Horvath, B. Petri, F. Trinkwalder, – Siemens
email: korpim@sbs.de
voice: +49 89 722-34570
fax: +49 89 722-23977

TITLE: Call Transfer Supplementary Service in H.323

Date: February 13, 1997

Purpose: To add Call Transfer Supplementary Service to Annex C of H.323

Overview: At the last meeting of Q.2/15 in Boulder (12/96), it was agreed to specify ISO/IEC/ QSIG protocols. It was tentatively decided to start a new Annex C of H.323 for these specifications. Based on those agreements, this contribution proposes an initial base text for a specification of the call transfer supplementary service for H.323 terminals. According to the Boulder results, the text is tentatively structured in the style of an Annex C.2 of H.323; however it may be decided that some more protocol related parts of it (particularly the detailed coding) might be moved to a new Annex of H.225.0 rather than H.323.

Changes to H.323

2. References

ISO/IEC 13865, 1995: *Information technology - Telecommunication and information exchange between systems - Private Integrated Services Network - Specification, functional model and information flows - Call transfer supplementary service*

ISO/IEC 13869, 1995: *Information technology - Telecommunication and information exchange between systems - Private Integrated Services Network - Inter-exchange signalling protocol - Call transfer supplementary service*

Add to Annex C:

C.2 Call Transfer

C.2.1 Scope and Field of Application

This section describes the Call Transfer supplementary service (SS-CT), which is applicable to various basic services supported by H.323 Multimedia Terminals. SS-CT is based on the equivalent supplementary service for Private Integrated Services Networks (PISN) specified in ISO/IEC 13865 and 13869.

C.2.2 Definition

Call Transfer (SS-CT) is a supplementary service which enables the served user (user A) to transform an existing call with user B into a new call between user B and a user C selected by user A. User A may or may not have a call established with user C prior to transfer. Each call can either be an incoming call to user A or an outgoing call from user A. On successful completion of SS-CT user B and user C can communicate with each other and user A will no longer be able to communicate with user B or user C.

C.2.3 Description

The initial call between user A and user B (primary call) must be answered before transfer can be initiated. On invocation of call transfer, if a call between user A and user C exists (secondary call), the transferred-to endpoint (user C) is informed of the pending call transfer, and transfer only proceeds if this endpoint agrees. In this case the transferred-to endpoint returns a temporary identifier to be used in the subsequent transfer procedure to identify the secondary call.

The transferring terminal (user A) requests the transferred endpoint (user B) to call the transferred-to endpoint (user C). The transferred endpoint then establishes a call to the transferred-to endpoint (transfer by rerouteing) and includes the temporary identifier for the secondary call if this call exists.

The primary call is retained until the first acknowledgement has been received from the transferred-to endpoint (user C), and is then released. This means that the primary call remains in place if call transfer fails before that stage. If the secondary call exists it is retained until the new call request arrives at the transferred-to endpoint and is then released.

Upon answer from user C (if no secondary call exists) or successful association of user C with the new call (if user C was already involved in the secondary call) user B and user C can communicate with each other. The media exchange capabilities of the new call should be equal to those of the primary call as far as possible.

If call transfer fails after the primary call was released the action taken is outside the scope of this recommendation.

C.2.4 Messages

The APDUs of call transfer operations are transported within Facility information elements in call control and FACILITY messages as defined in H.225.0. The call transfer operations are described in H.225.0 Annex X.2.

C.2.5 Actions at the transferring terminal

C.2.5.1 Normal procedures if only the primary call exists

In order to initiate call transfer, the transferring terminal shall send a callTransferInitiate invoke APDU in a FACILITY message to the transferred endpoint, using the call reference of the primary call, start timer T3 and enter state CT-Await-Initiate-Response. The invoke APDU shall contain element rerouteingNumber with the address of user C as selected by user A. Element callIdentity shall be empty.

On receipt of a callTransferInitiate return result APDU on the primary call the transferring terminal shall stop timer T3, clear the primary call and enter state CT-Idle.

If the primary call is released while in state CT-Await-Initiate-Response, the transferring terminal shall stop timer T3 and enter state CT-Idle.

C.2.5.2 Normal procedures if also the secondary call exists

In order to initiate call transfer, the transferring terminal shall send a callTransferIdentify invoke APDU in a FACILITY message to the transferred-to endpoint, using the call reference of the secondary call, start timer T1 and enter state CT-Await-Identify-Response.

On receipt of a callTransferIdentify return result APDU on the secondary call the transferring terminal shall stop timer T1, send a callTransferInitiate invoke APDU in a FACILITY message to the transferred endpoint, using the call reference of the primary call, start timer T3 and enter state CT-Await-Initiate-Response. The invoke APDU shall contain elements rerouteingNumber and callIdentity with the values received in callTransferIdentify return result.

On receipt of a callTransferInitiate return result APDU on the primary call the transferring terminal shall stop timer T3, clear the primary and secondary call if not already cleared, and enter state CT-Idle.

If the primary and/or secondary call is released while in state CT-Await-Identify-Response or CT-Await-Initiate-Response, the transferring terminal shall stop the relevant timer, clear the other call if not already cleared, and enter state CT-Idle.

C.2.5.3 Exceptional procedures

On receipt of a callTransferIdentify reject or return result APDU while in state CT-Await-Identify-Response the transferring terminal shall stop timer T1 and enter state CT-Idle.

On receipt of a callTransferInitiate reject or return result APDU while in state CT-Await-Initiate-Response the transferring terminal shall send a callTransferAbandon invoke APDU in a FACILITY message on the secondary call if this call exists, stop timer T3 and enter state CT-Idle.

On expiry of timer T1 or T3 the transferring terminal shall send a callTransferAbandon invoke APDU in a FACILITY message on the secondary call if this call exists, and enter state CT-Idle.

C.2.6 Actions at the transferred endpoint (user B)

C.2.6.1 Normal procedures

On receipt of a callTransferInitiate invoke APDU on the primary call the transferred endpoint shall determine whether it can participate in the call transfer. If it can it shall use the address contained in element rerouteingNumber to initiate call establishment towards the transferred-to endpoint, include in the SETUP message a callTransferSetup invoke APDU with the same value of callIdentity as received in the callTransferInitiate invoke APDU, and enter state CT-Await-Setup-Response. Optionally timer T4 may be started.

On receipt of an ALERTING or CONNECT message with a callTransferSetup return result APDU from the transferred-to endpoint the transferred endpoint shall stop timer T4 if running, send a RELEASE COMPLETE message with a callTransferInitiate return result APDU on the primary call to the transferring terminal and enter state CT-Idle. Further connection establishment between user B and user C shall follow standard H.323 procedures, with media properties inherited from the primary call as far as possible.

C.2.6.2 Exceptional procedures

If on receipt of a callTransferInitiate invoke APDU the transferred endpoint cannot participate in call transfer a callTransferInitiate return error APDU shall be sent in a FACILITY message on the primary call.

On expiry of timer T4, or on receipt of a clearing indication on the new call while in state CT-Await-Setup-Response, the transferred endpoint shall send a callTransferInitiate return error APDU in a FACILITY message on the primary call, stop timer T4 if running, and enter state CT-Idle. If a callTransferSetup return error APDU was received from the transferred-to endpoint the same error value shall be used in the callTransferInitiate return error APDU.

If the primary call is cleared while in state CT-Await-Setup-Response the transferred terminal shall stop timer T4 if running, clear the new call and enter state CT-Idle.

C.2.7 Actions at the transferred-to endpoint (user C)

C.2.7.1 Normal procedures

On receipt of a callTransferIdentify invoke APDU on the secondary call the transferred-to endpoint shall determine whether it can participate in the call transfer. If it can it shall send a callTransferIdentify return result APDU in a FACILITY message on the secondary call, start timer T2 and enter state CT-Await-Setup. The return result APDU shall contain user C's address in element rerouteingNumber and a local identifier for the secondary call in element callIdentity.

On receipt of a SETUP message with a callTransferSetup invoke APDU from the transferred endpoint the transferred-to endpoint shall, if in state CT-Await-Setup, stop timer T2, check elements rerouteingNumber and callIdentity and continue with call establishment if they correctly identify the secondary call. The secondary call shall be cleared using normal H.323 procedures.

If the callTransferSetup invoke APDU is received in state CT-Idle, call setup shall continue if element callIdentity is empty and if call transfer to user C is acceptable.

Either the ALERTING or the CONNECT message on the new call shall include a callTransferSetup return result APDU.

C.2.7.2 Exceptional procedures

If on receipt of a callTransferIdentify invoke APDU the transferred-to endpoint cannot participate in call transfer a callTransferIdentify return error APDU shall be sent in a FACILITY message on the secondary call.

If on receipt of a callTransferSetup invoke APDU with an empty element callIdentity the transferred-to endpoint cannot accept call transfer to user C the endpoint shall return a RELEASE COMPLETE message with a callTransferSetup return error APDU.

If on receipt of a callTransferSetup invoke APDU with a non-empty element callIdentity the transferred-to endpoint cannot match the identifier with a secondary call the endpoint shall return a RELEASE COMPLETE message with a callTransferSetup return error APDU.

If in state CT-Await-Setup a callTransferAbandon invoke APDU is received on the secondary call, or if the secondary call is cleared, the transferred-to endpoint shall stop timer T2 and enter state CT-Idle.

On expiry of timer T2 the transferred-to endpoint shall enter state CT-Idle.

C.2.8 Further notes on SS-CT

C.2.8.1 Interworking with non-H.323 terminals

In the specification of procedures above the term “endpoint” means “terminal” or “gateway”.

If user B or user C are non-H.323 terminals the gateway on the route to user B or user C will act as transferred endpoint or transferred-to endpoint, respectively.

C.2.8.2 Hold and consultation

Call transfer procedures do not demand a particular hold state for the primary or secondary call, i.e. user B or user C may or may not be put on hold prior to call transfer, depending on the capabilities of terminal A and the specific implementation.

With existing (e.g. PISN) implementations, call transfer is frequently performed by user A putting user B on hold, setting up the secondary call to user C (“consultation call”) and then invoking transfer. This method is also shown in the example below.

C.2.8.3 Additional procedures

C.2.8.3.1 Transfer without rerouteing

ISO/IEC 13869 specifies for PISNs a procedure “transfer by join”. This procedure is not applicable for H.323, where terminals rather than network exchanges provide call transfer functionality. It is not expected from a terminal to join two calls in which it no longer participates.

However, a similar method may be desireable for the case that the transferred or the transferred-to endpoint does not support SS-CT. This is for further study.

C.2.8.3.2 Actions subsequent to call transfer

For further information exchange between transferred and transferred-to endpoint, ISO/IEC 13869 specifies operations callTransferUpdate and subAddressTransfer. These operations may also be used in H.323. See ISO/IEC 13869 for details.

C.2.8.4 Terminology

The following terms are used in the examples below:

- H.323 SplS-terminal: An H.323 SplS-terminal is an H.323 terminal which can support QSIG supplementary services based on H.225.

- **Consultation:** A secondary call set up from user A to user C.

C.2.9 Dynamic Description of Transfer with Rerouting

In addition to Te A, Te B also has to be a H.323-SplS-terminal, or the gateway of B must be able to transfer calls.

C.2.9.1 Operational model

Description from user point of view:

User A (transferring party): communicating with B; selects C; Request B to connect to C (Media inherited?); transfer accepted; idle

User B (transferred party): communicating with A; receives request of transfer; confirmes media; communicating with C;

User C (transferred to party): idle; receives notification of incoming call; accepts call; confirmes media; communicating with B

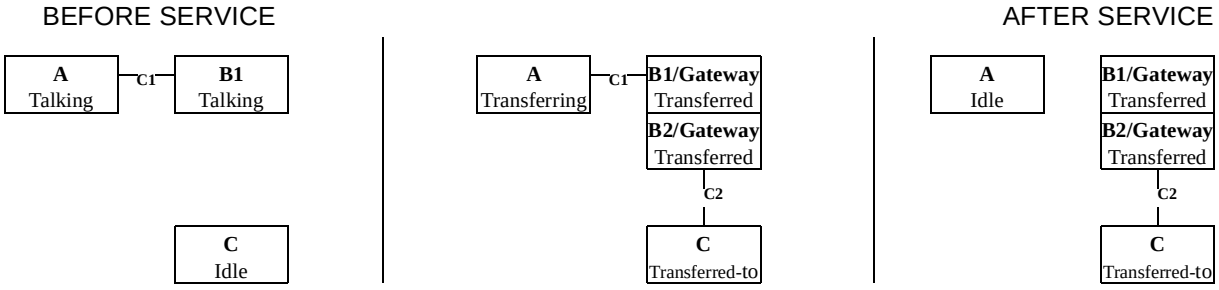


Figure C.2 -1 Operational model for transfer with rerouting

C.2.9.2 Communication between Transferring signalling entity (TRGSE) and TRGSE user

C.2.9.2.1 Table of primitives

Communication between the TRGSE and TRGSE user, is performed using the primitives shown in Table 1.

TABLE 1

Primitives and parameters

generic name	type			
	request	indication	response	confirm
CT_INITIATE	IDENTITY RRNUMBER	not defined[1]	not defined	-[2]

IDENTIFY	-	not defined	not defined	IDENTITY RRNUMBER

1 “not defined” means that this primitive is not defined
2 “-” means no parameters

C.2.9.2.2 Primitive definition

The definition of these primitives is as follows

a) The CT_INITIATE.request primitive is used to make the transfer request

b) The CT_INITIATE.confirm primitive is used to indicate the result of the transfer request

c) The IDENTIFY.request primitive is used to request identity TRTSE

d) The IDENTIFY.confirm primitive is used to confirm the identity of TRTSE

C.2.9.2.3 Parameter definition

The definition of the primitive parameters shown in Table 1 are as follows:

a) The IDENTITY parameter is the call identifier parameter. This parameter is mandatory.

b) The RRNUMBER parameter is the rerouting number parameter. This parameter is mandatory.

C.2.9.2.4 States

The following states are used to specify the allowed sequence of primitives between the TRGSE and TRGSE user

CT-Idle

- SS-CT is not operating

CT-Await-Answer-From-User-C

- A callTransferComplete invoke APDU with callStatus having value alerting has been sent to the TRDSE. This state is used during transfer by rerouting.

CT-Await-Identify-Response

- A callTransferIdentify invoke APDU has been sent to the TRTSE. This state is used during transfer by re-routing.

CT-Await-Initiate-Response

- A callTransferInitiate invoke APDU has been sent to the TRDSE. This state is used during transfer by re-routing

C.2.9.3 Communication between Transferred signalling entity TRDSE and TRDSE user

C.2.9.3.1 Table of primitives

Communication between the TRDSE and TRDSE user, is performed using the primitives shown in Table 2.

TABLE 2

Primitives and parameters

generic name	type			
	request	indication	response	confirm
CT_INITIATE	not defined[1]	IDENTITY RRNUMBER	-[2]	not defined
SETUP	IDENTITY	not defined	not defined	IDENTITY
UPDATE	RDNUMBER RDNAME	not defined	not defined	-

1 “not defined” means that this primitive is not defined
2 “-” means no parameters

C.2.9.3.2 Primitive definition

The definition of these primitives is as follows

a) The CT_INITIATE.indication primitive is used to notify about transfer iniate request

b) The CT_INITIATE.response primitive is used to indicate of the success of the transfer

c) The SETUP.request primitive is used to request for call establishment to TRTSE

d) The SETUP.confirm primitive is used to indicate of the success of call establishment to TRTSE

e) The UPDATE.request primitive is used to complete a call establishment to TRTSE

f) The UPDATE.confirm primitive is used to indicate that call establishment to TRTSE has been completed

C.2.9.3.3 Parameter definition

The definition of the primitive parameters shown in Table 2 are as follows:

a) The IDENTITY parameter is the call identifier parameter. This parameter is mandatory.

b) The RRNUMBER parameter is the rerouting number parameter. This parameter is mandatory.

c) The RDNUMBER parameter is a redirection number parameter. This parameter is mandatory

d) The RDNAME parameter is a redirection name parameter. This parameter is optional.

C.2.9.3.4 States

The following states are used to specify the allowed sequence of primitives between the TRDSE and TRDSE user

CT-Idle

- SS-CT is not operating

CT-Await-Setup-Response

- A callTransferSetup invoke APDU has been sent to the TRTSE. This state is used during transfer by rerouting.

CT-Await-Connect

- The Primary Call has been transferred to an alerting TRTSE, and the TRDSE has been notified. A CONNECT message indicating answering by the TRTSE is awaited.

C.2.9.4 Communication between Transferred-to signalling entity TRTSE and TRTSE user

C.2.9.4.1 Table of primitives

Communication between the TRTSE and TRTSE user, is performed using the primitives shown in Table 3.

TABLE 3

Primitives and parameters

generic name	type			
	request	indication	response	confirm
SETUP	not defined[1]	IDENTITY	IDENTITY	not defined
UPDATE	not defined	RDNUMBER RDNAME	RDNUMBER RDNAME	not defined
IDENTIFY	not defined	not defined	IDENTITY RRNUMBER	not defined

1 “not defined” means that this primitive is not defined

C.2.9.4.2 Primitive definition

The definition of these primitives is as follows

a) The SETUP.indication primitive is used to notify about setup request

b) The SETUP.response primitive is used to indicate of the acceptance of the call

c) The UPDATE.indication primitive is used to notify about update request

d) The UPDATE.response primitive is used to give response to update request

e) The IDENTIFY.response primitive is used to give response to identify request

C.2.9.4.3 Parameter definition

The definition of the primitive parameters shown in Table 3 are as follows:

a) The IDENTITY parameter is the call identifier parameter. This parameter is mandatory.

b) The RDNUMBER parameter is a redirection number parameter. This parameter is mandatory.

c) The RDNAME parameter is a redirection name parameter. This parameter is optional.

d) The RRNUMBER parameter is a rerouting number parameter. This parameter is mandatory.

C.2.9.4.4 States

The following states are used to specify the allowed sequence of primitives between the TRTSE and TRTSE user

CT-Idle

- SS-CT is not operating

CT-Await-Setup

- A callTransferIdentify return result APDU has been sent to the TRGSE. This state is used during transfer by rerouting.

C.2.9.5 Peer-to-peer communication for Transfer with Rerouting

C.2.9.5.1 Messages

The **FACILITY** message may be used to request or acknowledge a supplementary service. For more informarion see ITU-T Recommendation H.225.0 (May 28, 1996) section 7.4.1.

The **SETUP** message is sent by a calling entity to indicate its desire to set up a connection to a called entity. For more informarion see ITU-T Recommendation H.225.0 (May 28, 1996) section 7.3.11.

The **CONNECT** message is sent by the called entity to the calling entity to indicate acceptance of the call by the called entity. For more informarion see ITU-T Recommendation H.225.0 (May 28, 1996) section 7.3.3.

The **RELEASE COMPLETE** message is sent by a terminal to indicate release of the call if the reliable call signalling channel is open. For more informarion see ITU-T Recommendation H.225.0 (May 28, 1996) section 7.3.10.

C.2.9.5.2 Timers

T1 - Timer T1 shall operate at the TRGSE during state CT-Await-Identify-Response. Its purpose is to protect against the absence of a response to the TRGSE_identify_req.

T2 - Timer T2 shall operate at the TRTSE during state CT-Await-Setup. Its purpose is to protect against failure of completion of the call transfer operation.

T3 - Timer T3 shall operate at the TRGSE during state CT-Await-Setup-Response. Its purpose is to protect against failure to establish the new connetion.

T4 - Timer T4 may optionally operate at the TRDSE during state CT-Await-Setup-Response. Its purpose is to protect against failure to establish the new connection.

C.2.9.5.3 Counters

To be added

C.2.9.5.4 Message Flows

Table C.2 - 1 Transfer with rerouting[1]

Row no.	**User / Application action**	a) H.323 Native API primitive b) State c) Timer	Note	**IP, H.225, QSIG, CSTA, H.245**	Note	a) H.323 Native API primitive b) State c) Timer	**User / Application action**
1	**MM Terminal A**			**Network**		**MM Terminal B / Gateway**	
2	Active Basic Call between TE A and TE B Capabilities exchanged H.225 connection still exists UDP path for audio open						

Table C.2 - 1 Transfer with rerouting[1]

Row no.	User / Application action	a) H.323 Native API primitive b) State c) Timer	Note	IP, H.225, QSIG, CSTA, H.245	Note	a) H.323 Native API primitive b) State c) Timer	User / Application action
3	Request terminal B to transfer call to C enter state: CT-Await-Identify-Respond	⇓ **a) Transferring/ TRGSE** <TR_Initiate_req> b)CT_Await_Initiate_Response c) T3		H.225 <FACILITY> ⟶ Facility IE: invoke QSIG: callTransferInitiate rerouteingNumber=address C calltransferID (=0 for transfer with rerouting)		⇑ **a) Transferred** / TRDSE a)<TR_Initiate_ind> b) CT_Idle c) None	Receive transfer_request
4	**MM Terminal B / Gateway**			**Network**		**MM Terminal C**	
5	*establish TCP path for H.225 call signalling*						
6	Request for call establishment to User C Select Media User C address	⇓ **a) Transferred/ TRDSE** <Setup_req> <Update_req> b)CT_Await_Setup_Response c) T4 (optional)		H.225 <SETUP> ⟶ Facility IE: invoke QSIG: callTransferSetup invoke QSIG: callTransferUpdate redirectionNumber		⇑ **a) Transferred-to/TRTSE** <Setup_ind> <Update_ind> b) CT_Idle c) None	Receive H.225 setup_indication
7	Indication to Appl/ User B of MM Terminal C general availability	⇑ **a) Transferred/ TRDSE** <Setup_conf> <Update_conf> b) CT_Idle c) None		H.225 <CONNECT> ⟵ Facility IE: returnResult QSIG: callTransferSetup invoke QSIG: callTransferUpdate redirectionNumber		⇓ **a) Transferred-to/TRTSE** <Setup _resp> <Update_resp> b) CT_Idle c) None	Call accept by User C or by Application C
8	**MM Terminal A**			**Network**		**MM Terminal B / Gateway**	
9	*close logical channels*						
10	Receive H.225 release_indication	⇑ **a) Transferring/ TRGSE** <TR_Initiate_conf> b) CT_Idle c) None		H.225 <RELEASE COMPLETE> ⟵ Facility IE: returnResult QSIG: callTransferInitiate		⇓ **a) Transferred/ TRDSE** <TR_Initiate_resp> b) CT_Idle c) None	release call to TE A
11	**MM Terminal B / Gateway**			**Network**		**MM Terminal C**	
12	*exchange terminal capabilities* *open logical channels (H.245)*						

1Description from user point of view:

User A (transferring party): communicating with B; selects C; Request B to connect to C (Media inherited?); transfer accepted; idle

User B (transferred party): communicating with A; receives request of transfer; confirmes media; communicating with C;

User C (transferred to party): idle; receives notification of incoming call; accepts call; confirmes media;communicating with B

Figure C.2 -2 Failing case of tranfer with rerouting[1]

Row no.	User / Application action	a) H.323 Native API primitive b) State c) Timer	Note	IP, H.225, QSIG, CSTA, H.245	Note	a) H.323 Native API primitive b) State c) Timer	User / Application action
13	**MM Terminal A**			**Network**		**MM Terminal B / Gateway**	
14	Active Basic Call between TE A and TE B Capabilities exchanged H.225 connection still exists UDP path for audio open						
15	Request terminal B to transfer call to C enter state: CT-Await-Identify-Respond	⇓ **a) Transferring/ TRGSE** <CT_Initiate_req> b)CT_Await_Initiate_Response c) T3		H.225 <FACILITY> → Facility IE: invoke QSIG: callTransferInitiate rerouteingNumber=address C calltransferID (=0 for transfer with rerouting)		⇑ **a) Transferred/** TRDSE <TR_Initiate_ind> b) CT_Idle c) None	Receive transfer_request
16	**MM Terminal B / Gateway**			**Network**		**MM Terminal C**	
17	Request for call-establishment to TE C	*Establishment of basic call fails*					
18	**MM Terminal A**			**Network**		**MM Terminal B / Gateway**	
19		⇑ **a) Transferring/** TRGSE <TR_Initiate_rej> b) CT_idle c) None		H.225 <FACILITY> ← Facility IE: returnError QSIG: callTransferInitiate		⇓ **a) Transferred/** TRDSE <TR_Initiate_rej> b) CT_idle c) None	
20	A communicating with B						

1.Scenario Description from user point of view:

User A (transferring party):communicating with B; selects C; Request B to connect to C (Media inherited?); unsuccessful transfer; communicating with B

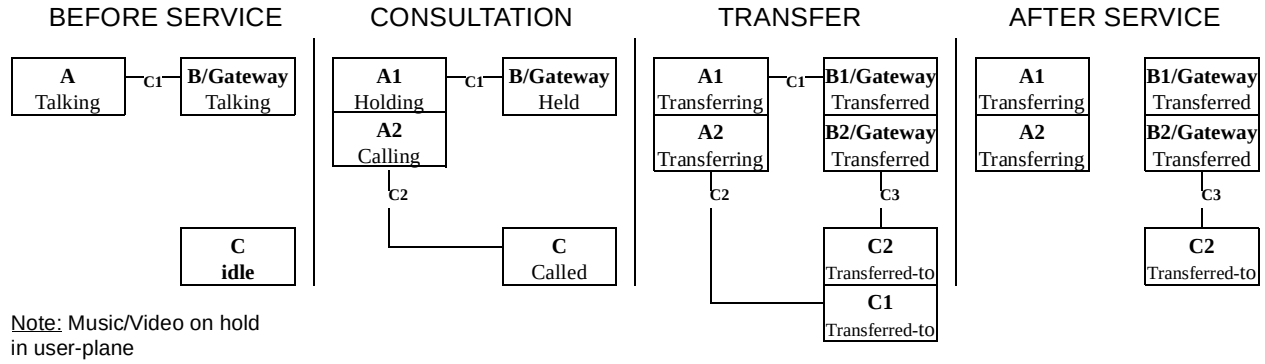
User B (transferred party):communicating with A; receives request for transfer; unsuccessful transfer; communicating with A;

User C (transferred to party): idle; unsuccessful call establishment from B; idle

C.2.10 Transfer with consultation

In addition to Te A, Te B also has to be a H.323-SplS-terminal, or the Gateway of B must be able to transfer calls.

C.2.10.1 Operational model



Note: Music/Video on hold in user-plane

Figure C.2 -3 Operational model for transfer with consultation

C.2.10.2 Description from user point of view:

User A: communicating with B; B on Hold (optional); User selects C and Media; requests connection to C; Consultation Active; Request B to connect to C; Transfer accepted; idle

User B: communicating with A; receives request for transfer; confirmes media; communicating with C

User C:idle; receives notification of incoming call; accepts call; confirmes media; communicating with A

receives notification of incoming call; accepts call; confirmes media; communicating with A

Table C.2 - 2 Transfer with consultation

Row no.	**User / Application action**	a) H.323 Native API primitive b) State c) Timer	Note	**IP, H.225, QSIG, CSTA, H.245**	Note	a) H.323 Native API primitive b) State c) Timer	**User / Application action**
21	**MM Terminal A**			**Network**		**MM Terminal B**	
22	Active Basic Call between TE A and TE B Capabilities exchanged H.225 connection still exists UDP path for audio open						
23	**MM Terminal A**			**Network**		**MM Terminal C**	
24	*Establishment of Basic Call*						
25	*Consultation active*						
26	Identification of Te C	⇓ **a) Transferring/ TRGSE** <Identify_req> b)CT_Await_Identify_Response c) T1		H.225: <FACILITY> → Facility IE: invoke QSIG: callTransferIdentify	1	⇑ **a) Transferred-to** <Identify_ind> b) CT_Idle c) None	

Row no.	User / Application action	a) H.323 Native API primitive b) State c) Timer	Note	IP, H.225, QSIG, CSTA, H.245	Note	a) H.323 Native API primitive b) State c) Timer	User / Application action
27	Receive Identification	⇑ **a) Transferring/ TRGSE** <Identify_conf> b) CT_Idle c) None		**H.225 <FACILITY>** ◄ Facility IE: returnResult QSIG: callTransferIdentify calltransferID rerouteingNumber=address C	2	⇓ **a) Transferred-to /**TRTSE <Identify_resp> b) CT_Await_Setup c) T2	determine ability of CT enter state: CT-Await-Set-up
28	**MM Terminal A**			**Network**		**MM Terminal B / Gateway**	
29	*transfer with rerouting (C.2.9)* [3]						
30	**MM Terminal A**			**Network**		**MM Terminal C**	
31	*close logical channels*						
32	Receive H.225 release_ indication	⇑ **a) Transferring/ TRGSE** <Release_ind> b) CT_Idle c) None		H.225 <RELEASE COMPLETE> ◄		⇓ **a) Transferred-to /TRTSE** <Release _req_> b) CT_Idle c) None	release call to TE A

1optional
2optional
3No transfer with rerouting?

C.2.10.3 SDLs

C.2.10.3.1 SDL Model for Call Transfer

To be added

C.2.10.3.2 Primitive parameter default values

To be added

C.2.10.3.3 Message field default values

To be added

C.2.10.3.4 Transferring Signalling Entity SDLs

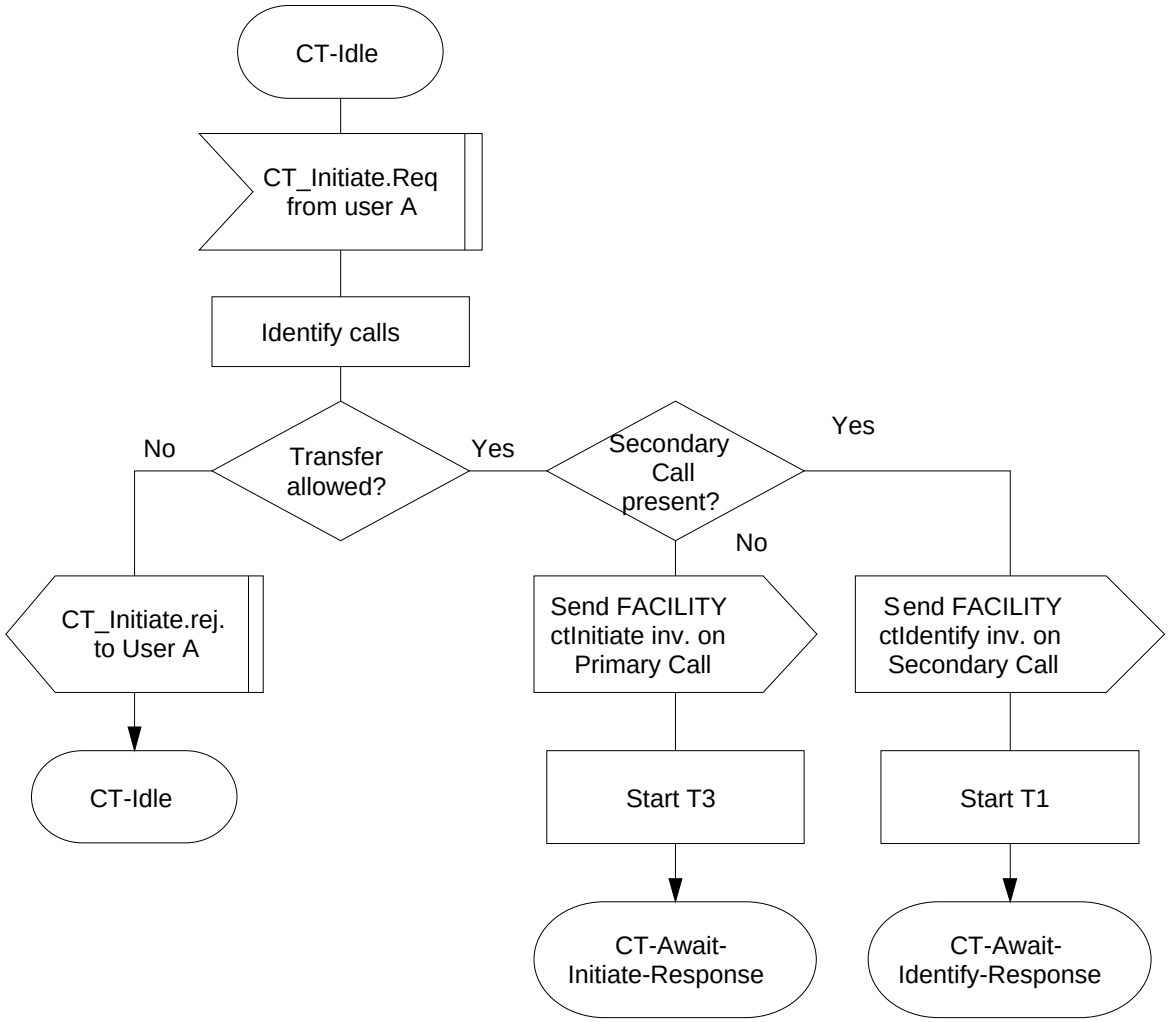


Figure 1 -4 Transferring Signalling Entity SDL (sheet 1 of 3)

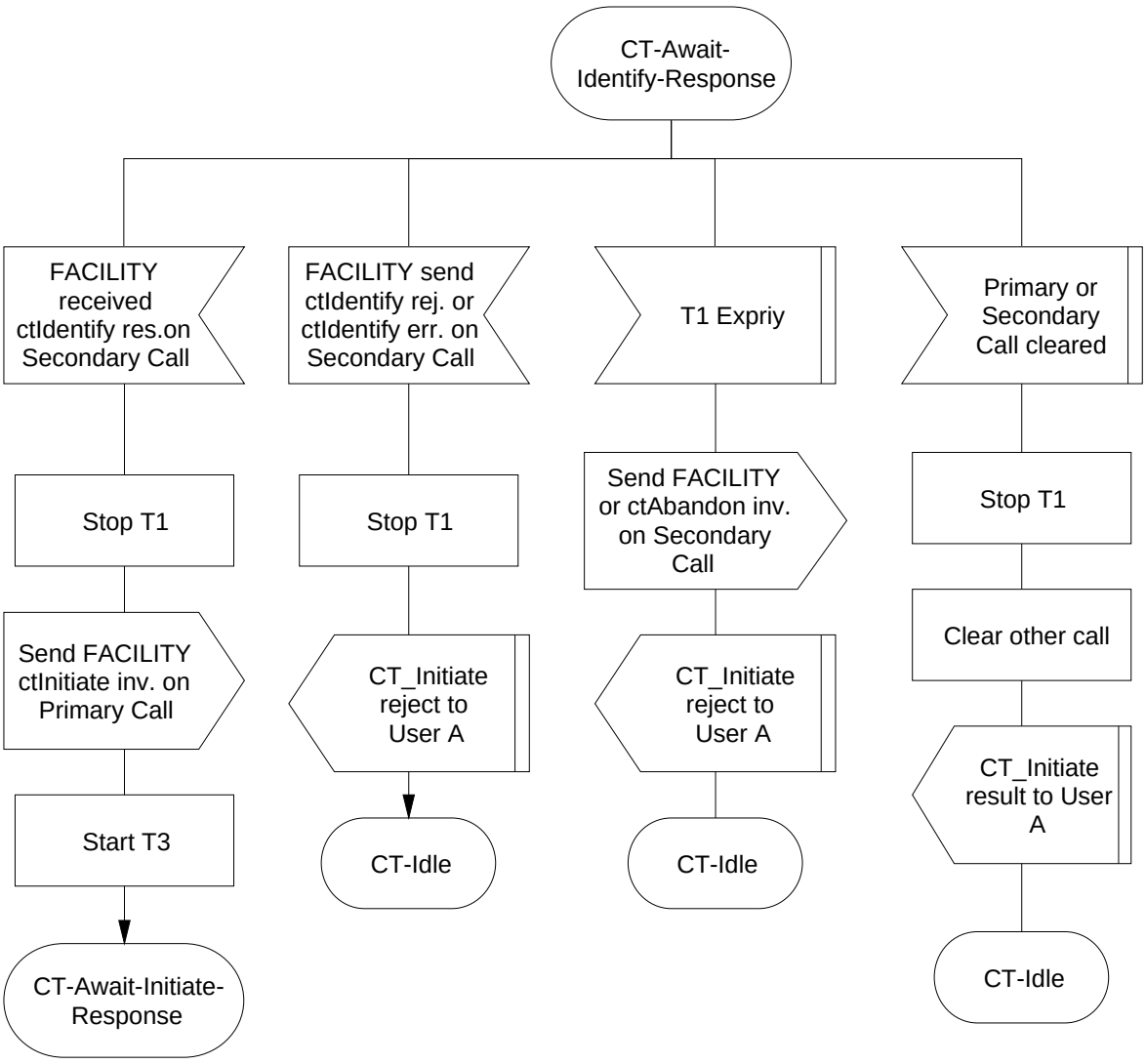


Figure 1 -5 Transferring Signalling Entity SDL (sheet 2 of 3)

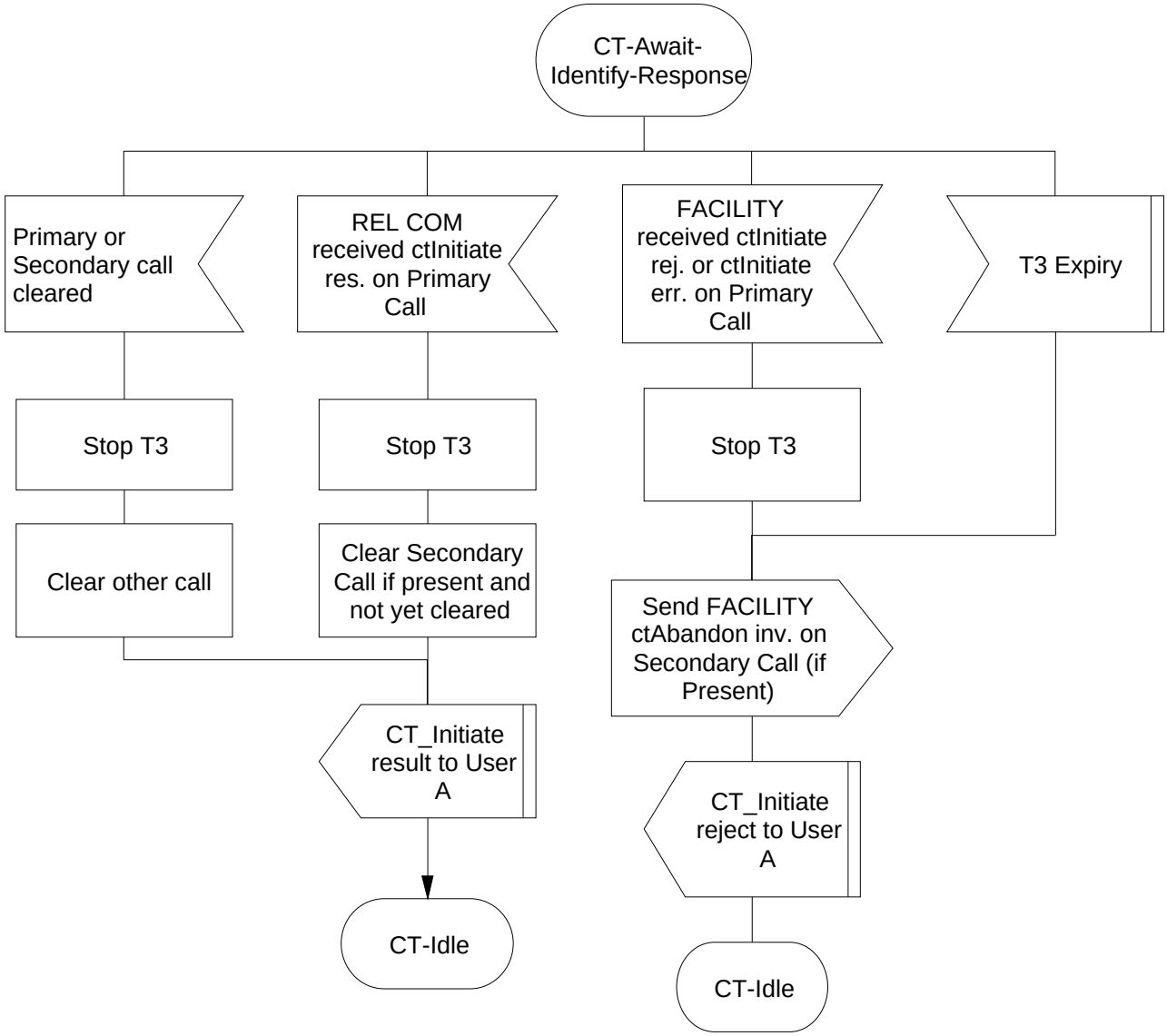


Figure 1 -6 Transferring Signalling Entity SDL (sheet 4 of 4)

C.2.10.3.5 Transferred Signalling Entity SDLs

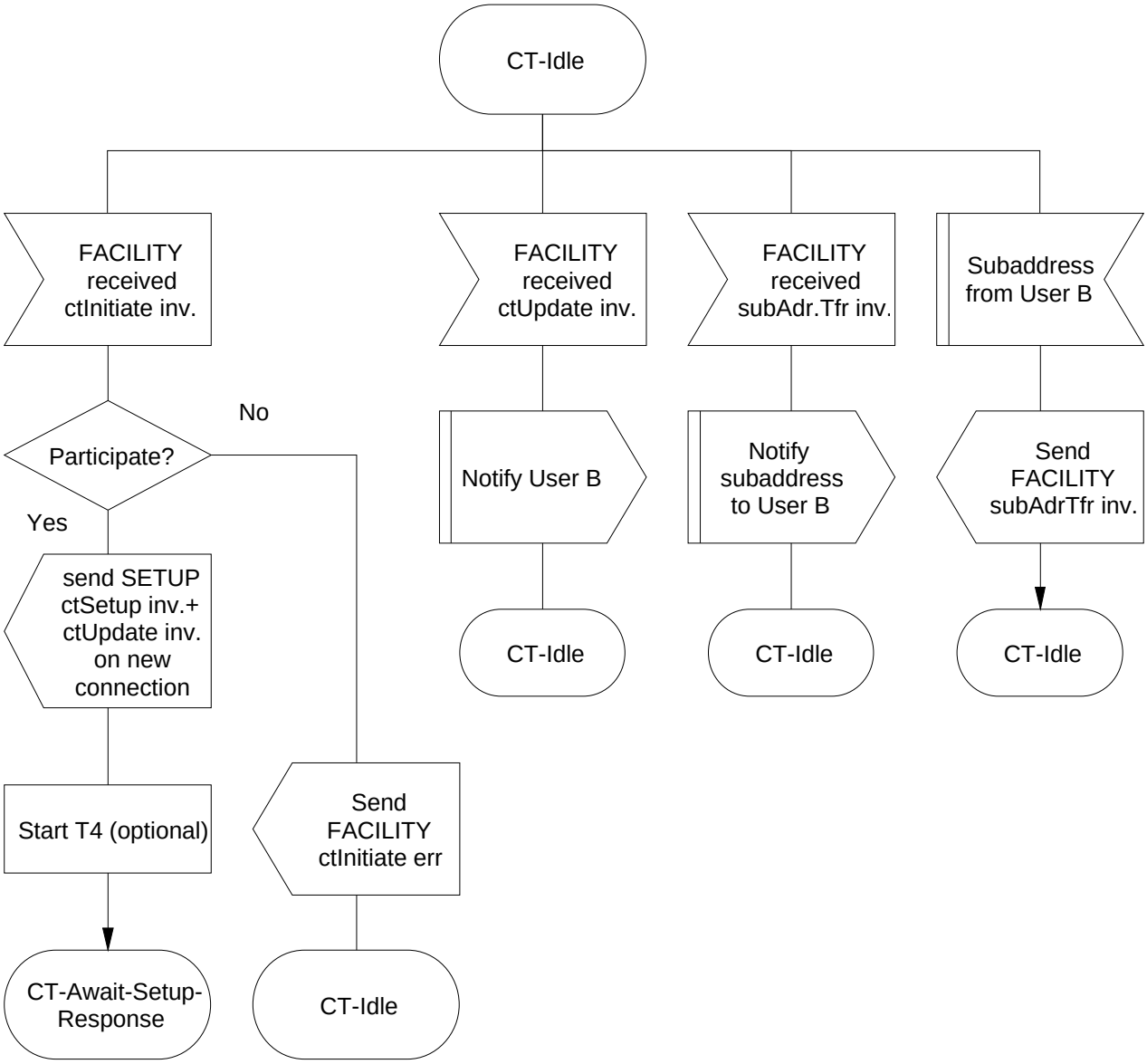


Figure 1 -7 Primary Signalling Entity SDL - TRGSE (sheet 1 of 3)

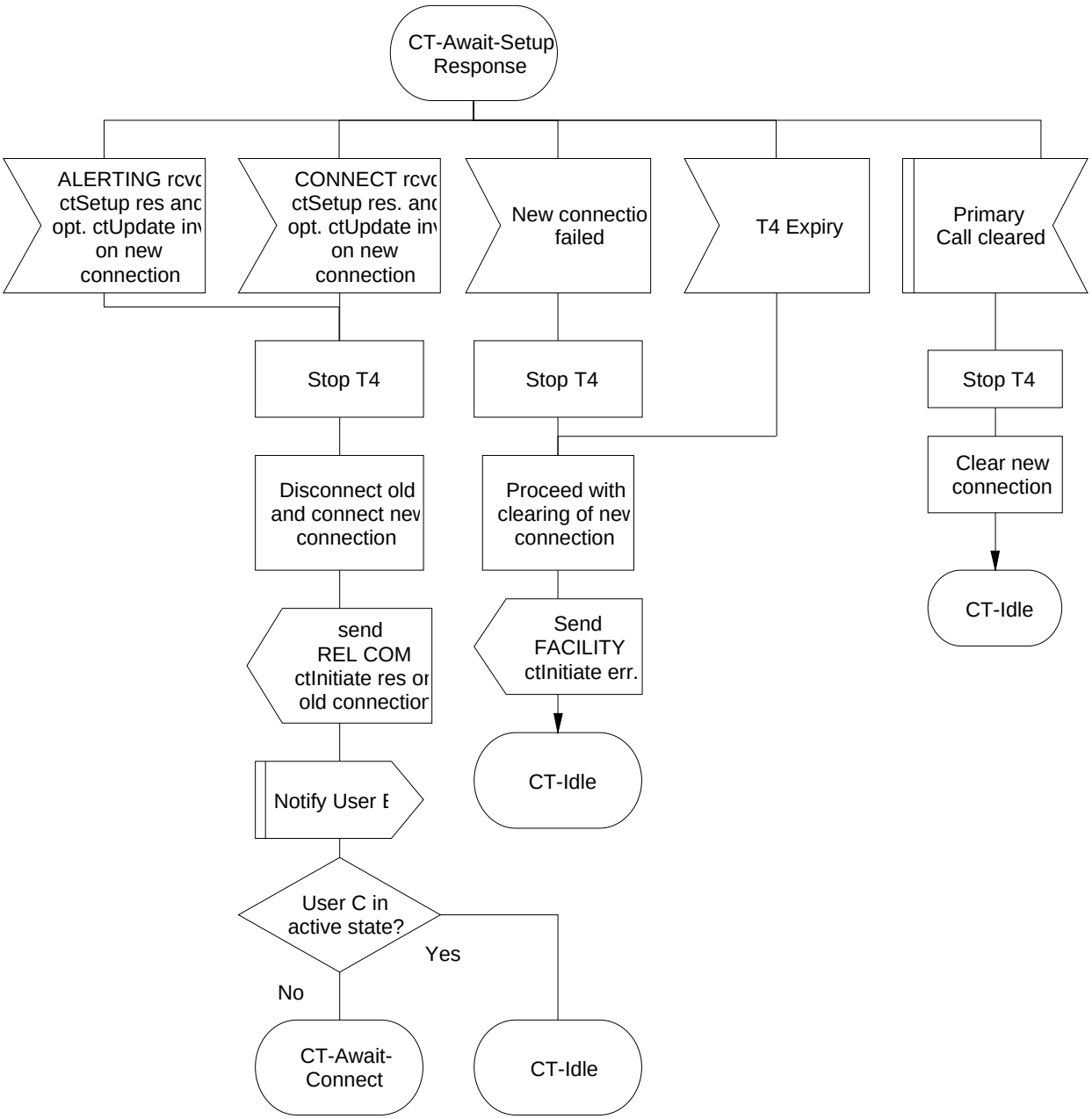


Figure 1 -8 Primary Signalling Entity SDL (sheet 2 of 3)

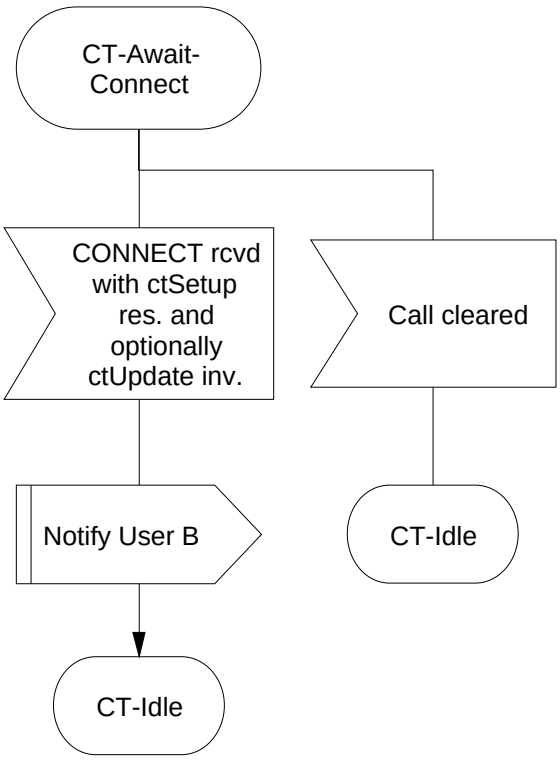


Figure C.2 -9 Primary Signalling Entity SDL (sheet 3of 3)

C.2.10.3.6 Transferred-To Signalling Entity SDLs

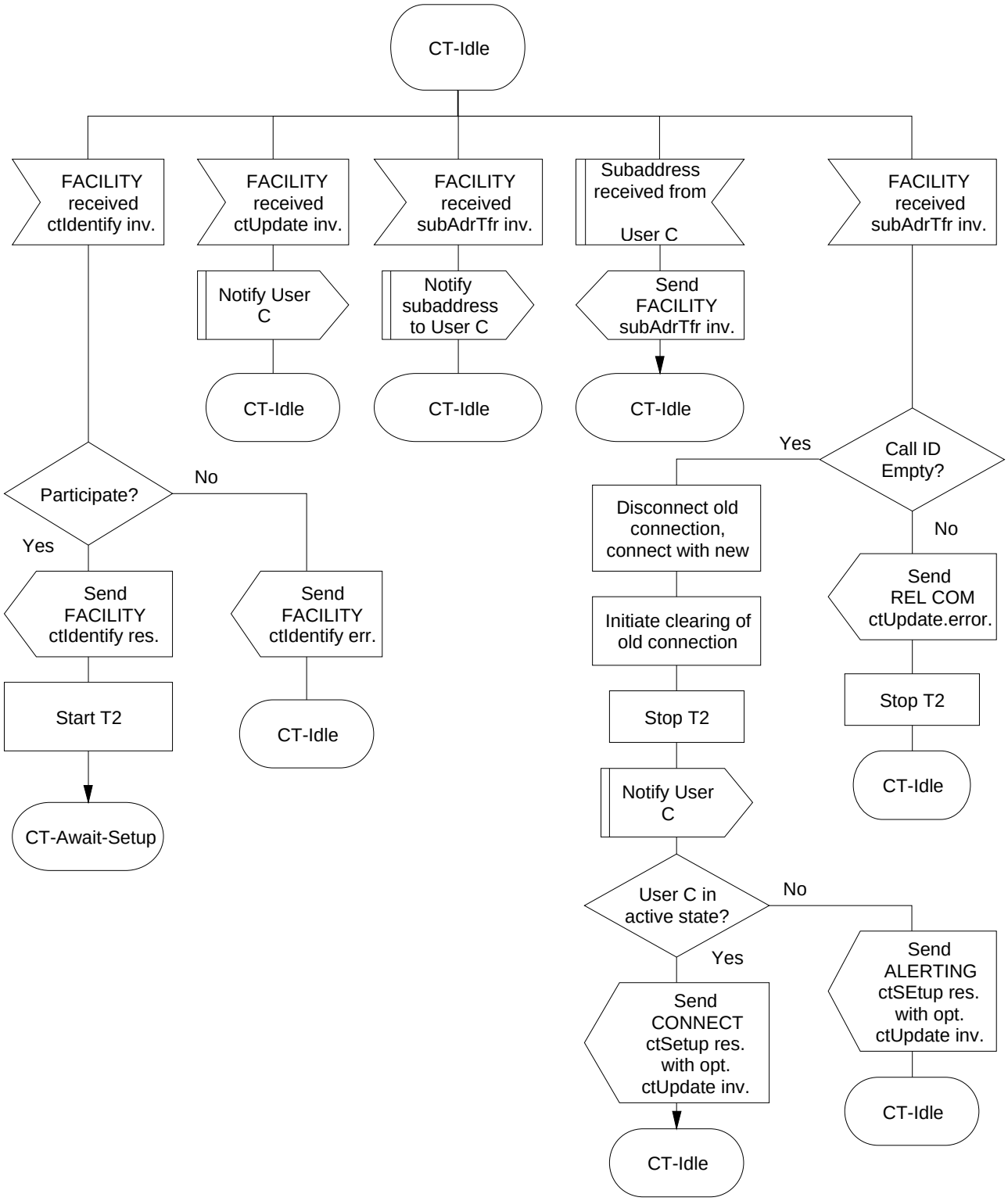


Figure 1 -10 Secondary Signalling Entity SDL (sheet 1 of 2)

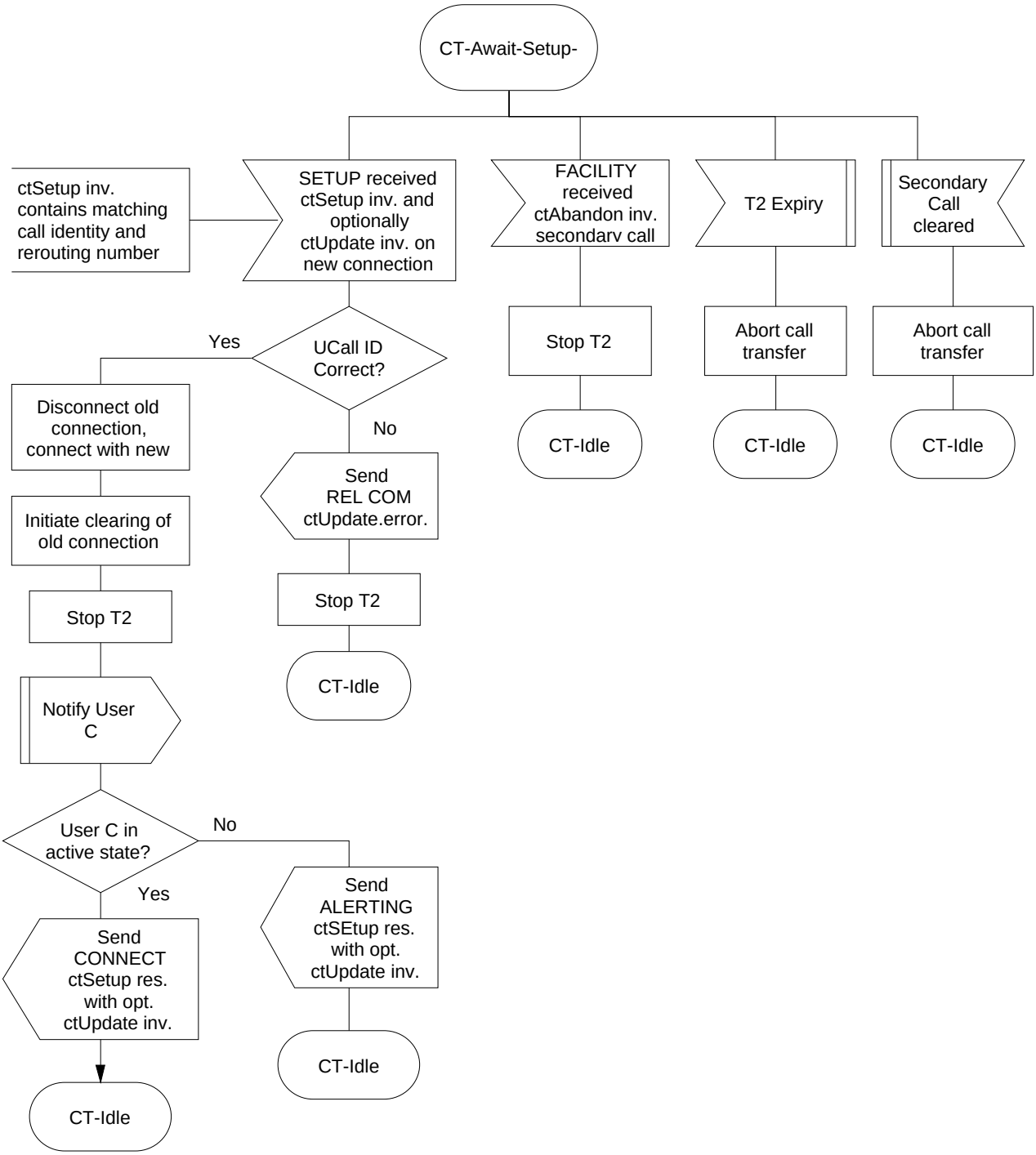


Figure 1 -11 Secondary Signalling Entity SDL (sheet 2 of 2)

Changes to H.225.0

Add to Annex X:

X.2 Operations of SS-CT

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

```
Call-Transfer-Operations
- - Is the same object identifier used ?
            {iso (1)   standard (0)    pss1-call-transfer (13869)
            call-transfer-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-service-extension-definition
              {iso(1) standard (0) pss1-generic-procedures(11582)
               msi-definition(0) }
            Name FROM Name-Operations { iso(1) standard (0) pss1-name(13868)
               name-operations(0) }
            notAvailable, supplementaryServiceInteractionNotAllowed,
              invalidCallState FROM General-Error-List {ccitt(0)
              recommendation(0) q 950 general-error-list(1)}
            PresentedAddressScreened, PresentedNumberScreened, PartyNumber,
              PartySubaddress FROM Addressing-Data-Elements { iso(1)
              standard (0) pss1-generic-procedures (11582)
              addressing-data-elements(9)}
            PSS1InformationElement FROM pss1-generic-parameters-def inition {
              iso(1) standard (0) pss1-generic-procedures (11582)
              pss1-generic-parameters(6)};


CallTransferIdentify     ::=    OPERATION
                         ARGUMENT DummyArg
                         RESULT   CTIdentifyRes
                         ERRORS  {  notAvailable,
                                    invalidCallState,
                                    supplementaryServiceInteractionNotAllowed,
                                    unspecified }

CallTransferAbandon      ::=    OPERATION
                         ARGUMENT DummyArg

CallTransferInitiate     ::=    OPERATION
                         ARGUMENT CTInitiateArg
                         RESULT   DummyRes
                         ERRORS  {  notAvailable,
                                    invalidCallState,
                                    invalidReroutingNumber,
                                    unrecognizedCallIdentity,
                                    establishmentFailure,
                                    supplementaryServiceInteractionNotAllowed,
                                    unspecified }
```

```
CallTransferSetup        ::=    OPERATION
                         ARGUMENT CTSetupArg
                         RESULT   DummyRes
                         ERRORS  {  notAvailable,
                                    invalidCallState,
                                    invalidReroutingNumber,
                                    unrecognizedCallIdentity,
                                    unspecified,
                                    supplementaryServiceInteractionNotAllowed}
--<<<H.323: CallTransferActive is not used>>>
-- CallTransferActive    ::=    OPERATION
--                       ARGUMENT CTActiveArg
--<<<H.323: CallTransferComplete is not used>>>
-- CallTransferComplete  ::=    OPERATION
--                       ARGUMENT CTCompleteArg

CallTransferUpdate       ::=    OPERATION
                         ARGUMENT CTUpdateArg

SubaddressTransfer       ::=    OPERATION
                         ARGUMENT SubaddressTransferArg

DummyArg           ::= CHOICE   {
                         NULL,
                         [1]    IMPLICIT Extension,
                         [2]    IMPLICIT SEQUENCE OF Extension    }

DummyRes           ::= CHOICE   {
                         NULL,
                         [1]    IMPLICIT Extension,
                         [2]    IMPLICIT SEQUENCE OF Extension
                                                    }
--<<<H.323: CTCompleteArg is not used>>>
-- CTCompleteArg   ::= SEQUENCE {
--                       endDesignation           EndDesignation,
--                       redirectionNumber        PresentedNumberScreened,
--                       basicCallInfoElements    PSS1InformationElement OPTIONAL,
                         --Information elements <<< ETSI: Party category and >>>
                         --Progress indicator are conveyed
--                       redirectionName          Name OPTIONAL,
--                       callStatus               CallStatus DEFAULT answered,
--                       argumentExtension        CHOICE {
--                                                  [9] IMPLICIT Extension,
--                                                  [10] IMPLICIT SEQUENCE OF Extension
--                                                } OPTIONAL
--                                }


CTInitiateArg      ::= SEQUENCE {
                       callIdentity      CallIdentity,
                       reroutingNumber   PartyNumber,
                       argumentExtension CHOICE {
                                           [6] IMPLICIT Extension,
                                           [7] IMPLICIT SEQUENCE OF Extension
                                         } OPTIONAL
                                }

CTSetupArg         ::= SEQUENCE {
                       callIdentity      CallIdentity,
                       argumentExtension CHOICE {
                                           [0] IMPLICIT Extension,
                                           [1] IMPLICIT SEQUENCE OF Extension
                                         } OPTIONAL
                                }
```

```
--<<<H.323: CTActiveArg is not used>>>
--CTActiveArg      ::= SEQUENCE {
--                     connectedAddress         PresentedAddressScreened,
--                     basicCallInfoElements    PSS1InformationElement OPTIONAL,
                       -- Information elements <<<ETSI: Party category and>>>
                       -- Progress indicator are conveyed
--                     connectedName            Name OPTIONAL,
--                     argumentExtension        CHOICE {
--                                               [9] IMPLICIT Extension,
--                                               [10] IMPLICIT SEQUENCE OF Extension
--                                              } OPTIONAL
--                               }

CTIdentifyRes      ::= SEQUENCE {
                       callIdentity      CallIdentity,
                       reroutingNumber   PartyNumber,
                       resultExtension   CHOICE {
                                          [6] IMPLICIT Extension,
                                          [7] IMPLICIT SEQUENCE OF Extension
                                         } OPTIONAL
                               }
CTUpdateArg        ::= SEQUENCE {
                       redirectionNumber        PresentedNumberScreened,
                       redirectionName          Name OPTIONAL,
                       basicCallInfoElements    PSS1InformationElement OPTIONAL,
                       -- Information elements <<<ETSI: Party category and>>>
                       -- Progress indicator are conveyed
                       argumentExtension        CHOICE {
                                                 [9] IMPLICIT Extension,
                                                 [10] IMPLICIT SEQUENCE OF Extension
                                                } OPTIONAL
                               }

SubaddressTransferArg ::= SEQUENCE     {
                          redirectionSubaddress PartySubaddress,
                          argumentExtension     CHOICE {
                                                 [0] IMPLICIT Extension,
                                                 [1] IMPLICIT SEQUENCE OF Extension
                                                } OPTIONAL
                                        }

--<<<H.323: CallStatus is not used>>>
--CallStatus       ::= ENUMERATED      {
--                               answered(0),
--                               alerting(1) }

CallIdentity       ::= NumericString (SIZE 1..4)

--<<<H.323: CallStatus is not used>>>
--EndDesignation   ::= ENUMERATED {
--                               primaryEnd(0),
--                               secondaryEnd(1) }

--Unspecified             ERROR PARAMETER Extension
unspecified               Unspecified ::= 1008
callTransferIdentify      CallTransferIdentify   ::= 7
callTransferAbandon       CallTransferAbandon    ::= 8
callTransferInitiate      CallTransferInitiate   ::= 9
callTransferSetup         CallTransferSetup      ::= 10
--callTransferActive      CallTransferActive     ::= 11
--callTransferComplete    CallTransferComplete   ::= 12
callTransferUpdate        CallTransferUpdate     ::= 13
subaddressTransfer        SubaddressTransfer     ::= 14

invalidReroutingNumber    ERROR ::= 1004
            -- used when establishment of the new connection fails
            -- because the reroutingNumber is not a valid PISN address
```

```
unrecognizedCallIdentity ERROR ::= 1005
            -- used when establishment of the new connection fails because it
            -- could not be associated with a SS-CT entity at the Secondary PINX
establishmentFailure     ERROR ::= 1006
            -- used when establishment of the new connection fails and
            -- no other error applies

END  -- of Call-Transfer-Operations
```