# Genesys Events and Models Reference

Network T-Server Attended Transfer Call Flows

4/2/2025

Network T-Server Attended Transfer Call Flows

Contents

Standard Network Call Initiation

The Standard Network Call Initiation figure illustrates the standard call setup for a call entering a Genesys environment through a Network T-Server. The diagrams that follow it in this section assume the completion of this stage and present common network attended transfer scenarios and a few error cases.
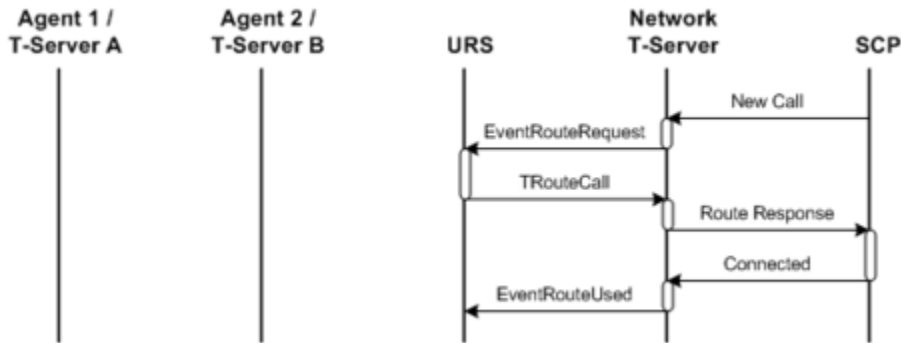
Standard Network Call Initiation

Consultation Leg Initiation, Specific Destination

The following figure illustrates the creation of the consultation leg in a case where the destination DN and its location are provided in the TNetworkConsult() function call.
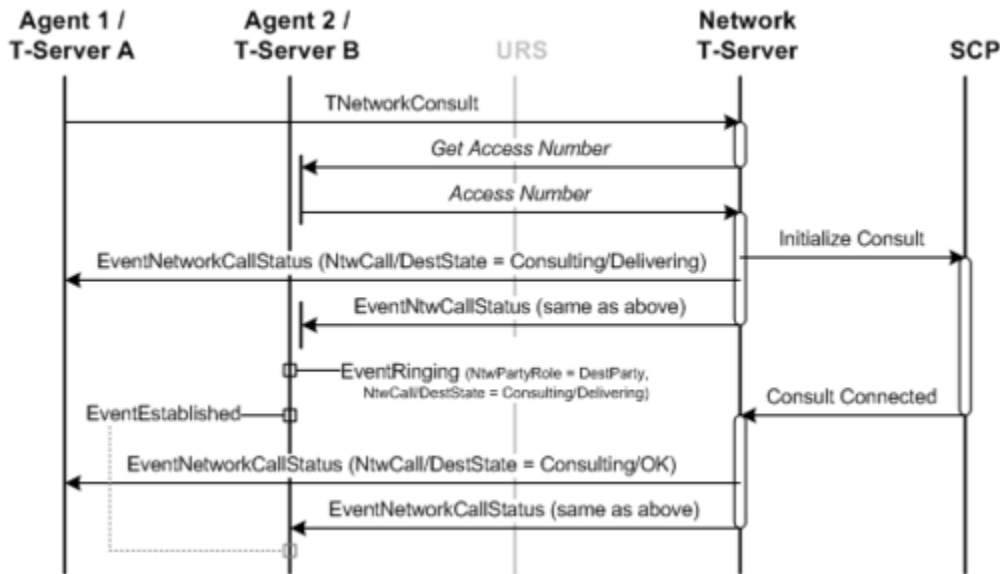
Consultation Leg Initiation, Specific Destination

Failed Consultation: Specific Target

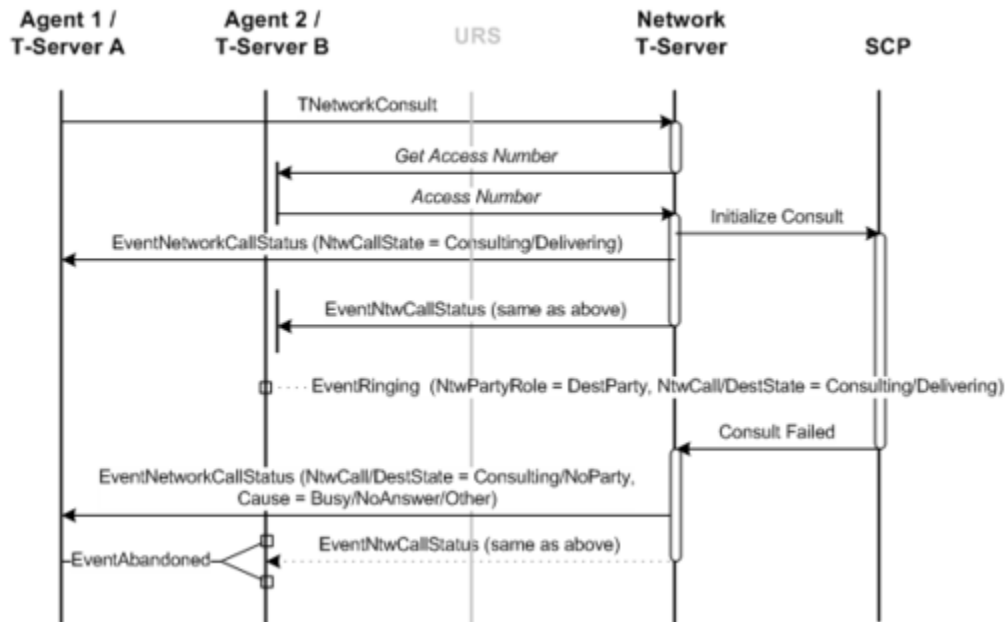In the following figure, the request to consult has failed. Events shown with dotted lines are distributed only when the call is delivered to the intended destination, but not answered (for instance, when State = NoAnswer). Since the consult request in this case is made to a specific DN and location, an explicit reconnect request (or another TNetworkConsult request, if allowed by the given SCP) from Agent 1 is required. Until such a request is made, the call remains in the Consulting/NoParty state.
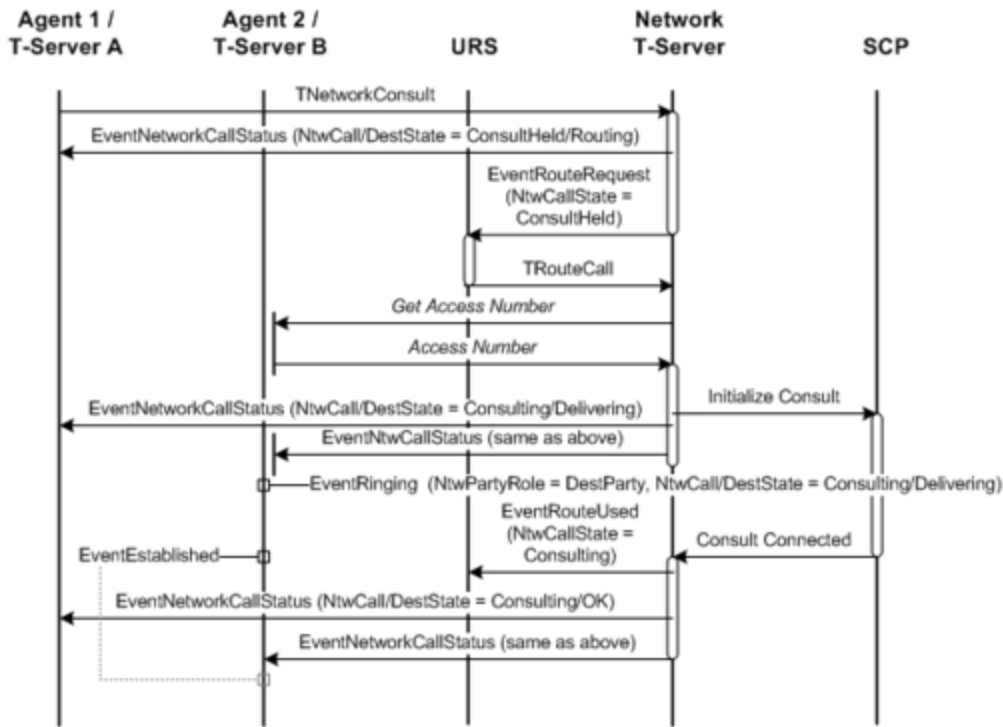

Failed Consultation: Direct Target

Consultation Leg Initiation, URS Selected Destination

The following figure illustrates the creation of the consultation leg in a case where the destination DN and its location are provided by URS. In this case the original caller is still connected to the call (speaking with Agent 1) while URS selects the target. The caller is then placed on hold the moment the consultation leg is created. Depending on the given SCP's capabilities, other scenarios are also possible. For example:

- The original caller might be put on hold immediately (as is the case with conventional local two-step transfers). In such a case, the value for NetworkState for the first event is Consulting/Routing.

- The original caller might be connected to the call throughout the entire consult and delivery phase, and then put on hold only when the destination (Agent 2) answers the call. In this case, the second event has a value of ConsultHeld instead of Consulting.

Consultation Leg Initiation, URS Selected Destination

Failed Consultation: URS Selected Destination

The following figure shows a failed request for consultation. Events shown with dotted lines are distributed only when the call is delivered to the intended destination, but not answered (for instance, when State = NoAnswer). Because URS controls consultation initiation in this case, no explicit reconnection is required. URS continues to offer different route targets until the consultation is connected. (The shaded portion of the diagram may repeat several times.) At this point:

- Reconnect can still occur manually.

- URS can reconnect using a reject or default route instruction.
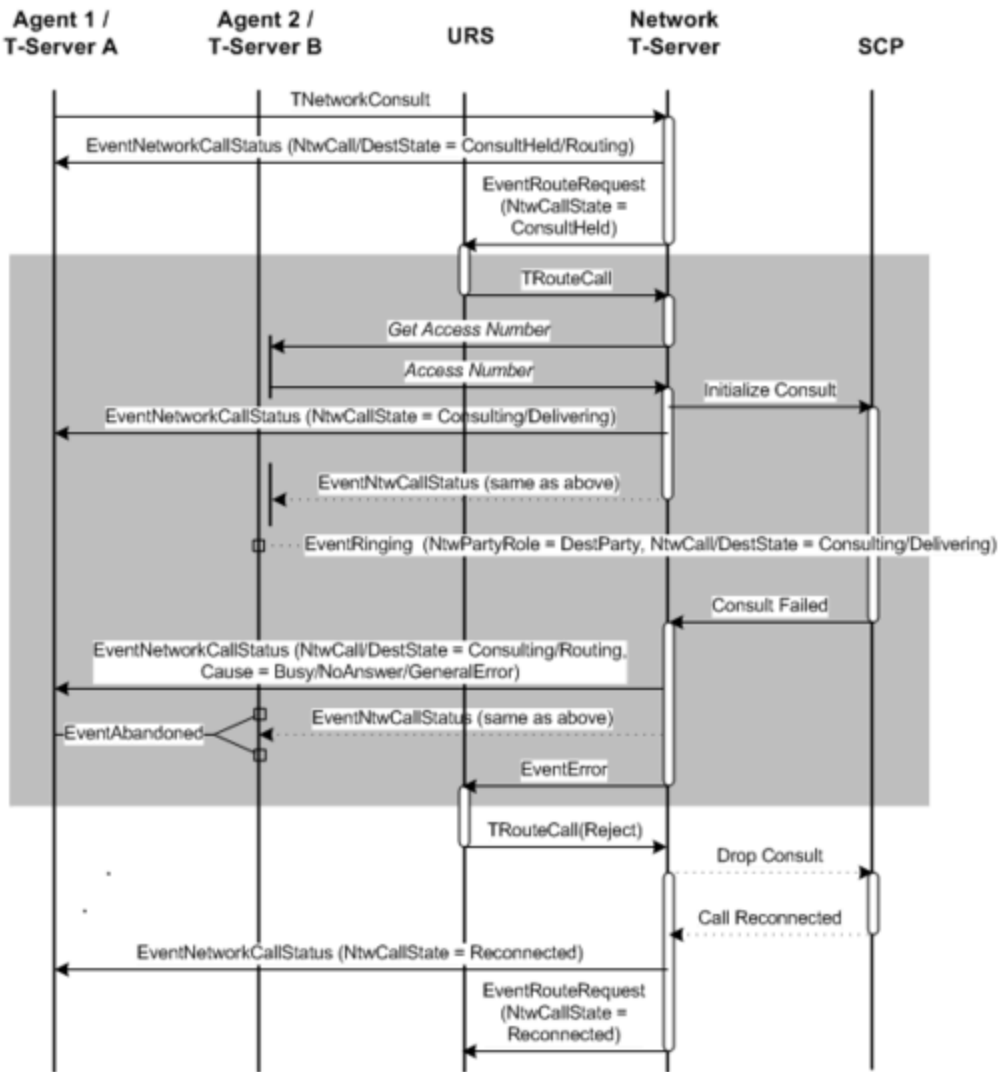
Failed Consultation: URS Selected Target

Transfer/Conference Completion: Explicit

The following figure illustrates the completion of either a network-attended transfer or conference. It assumes that the call had a successful consultation phase for its transfer or conference completion to be valid. This assumption is necessary since EventNetworkCallStatus is a direct response to the TNetworkTransfer feature request. The Network T-Server sends Agent 1 EventNetworkCallStatus regardless of whether the call has already been released.
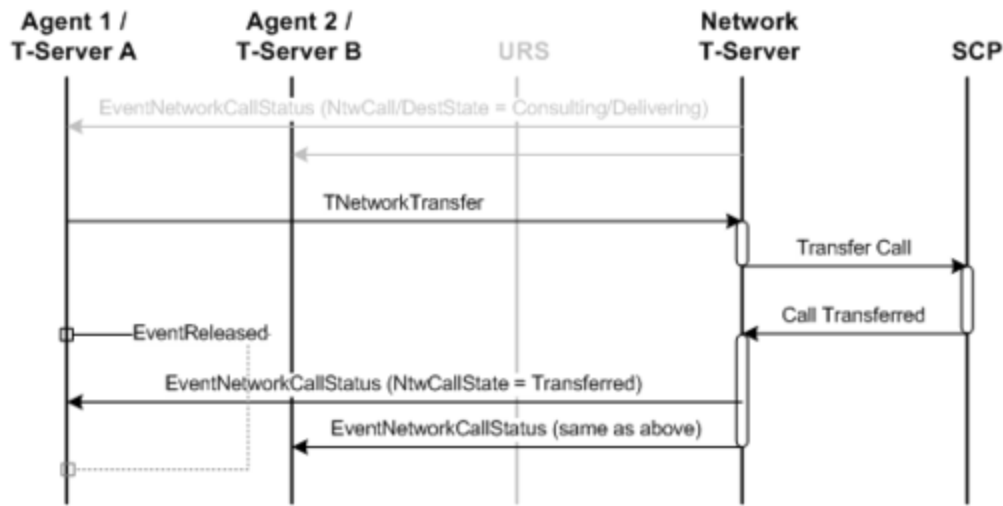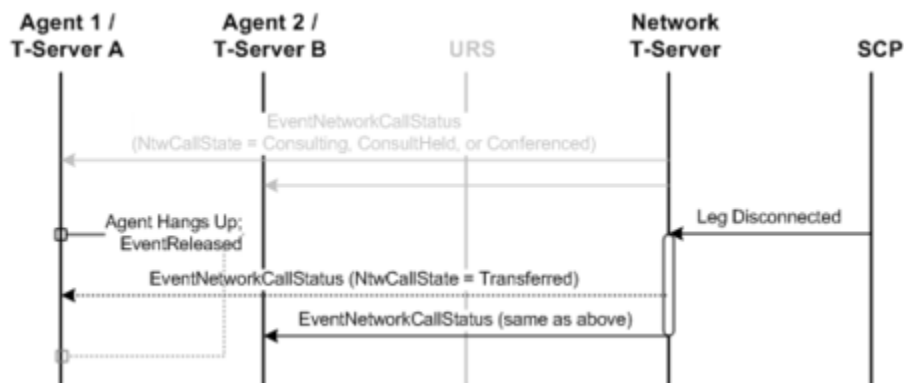
Transfer/Conference Completion: Explicit

Transfer Completion: Implicit

An implicit transfer completion occurs when Agent 1's channel becomes disconnected and the call has a status of `Consulting, ConsultHeld,` or `Conferencing`.

> ### Important
>
> This scenario applies only when the disconnection is with respect to Agent 1. (For details on a disconnection with Agent 2, see Implicit Reconnection (by SCP) and Implicit Reconnection (by Network T-Server). For a disconnection with respect to the caller, see Caller Abandonment.)
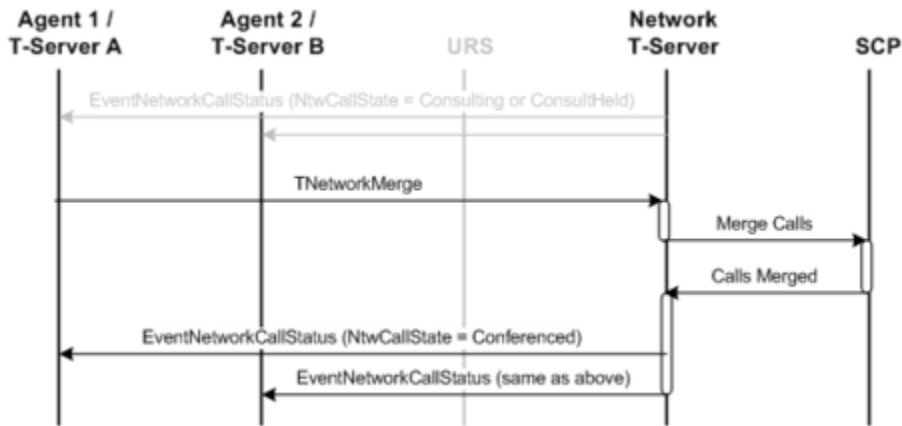
`EventNetworkCallStatus` is sent to Agent 1 only if T-Server A delays the release of `EventReleased` for some reason.

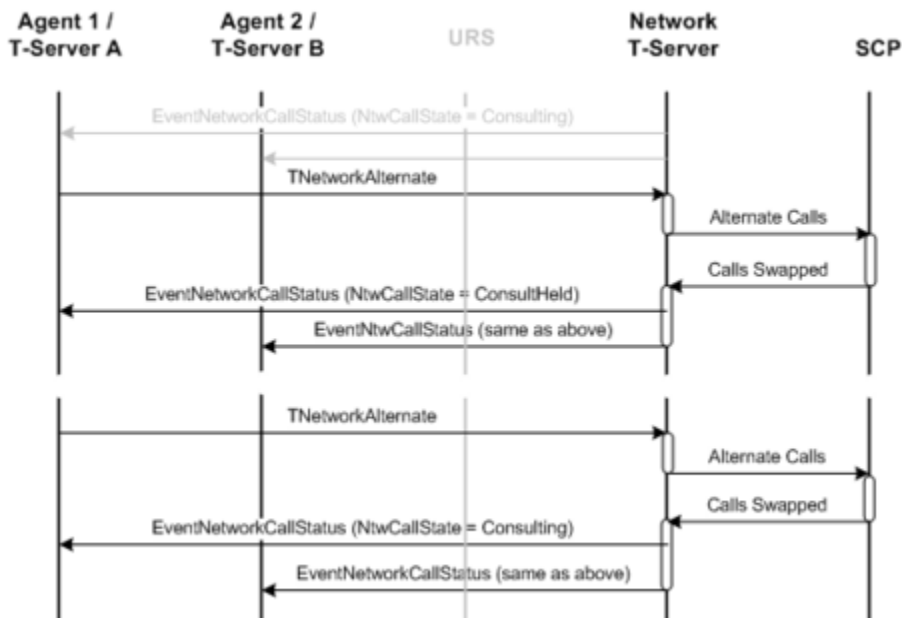Network Attended Transfer Completion: Implicit

Conference Completion

The following figure illustrates a standard conference completion.

Conference Completion

Alternate Call Service

The following figure contains two variations of the service to show different possible network states for the events.
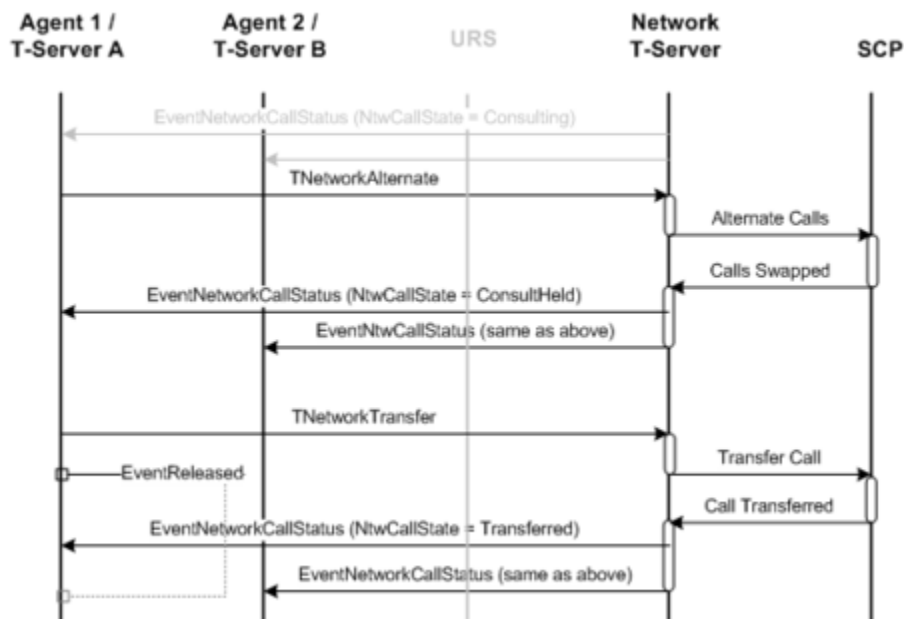
Alternate Call Service with Transfer Completion

The following figure demonstrates the applicability of TNetworkTransfer after the use of the alternate call service.

> ### Important
> TNetworkConference, TNetworkReconnect, and the implicit form of transfer completion are also available after an instance of the alternate call service.


Alternate Call Service with Transfer Completion

Reconnection

The following three figures show the call flows for reconnecting the caller to Agent 1. As with transfer completion, reconnection has both explicit (Network Attended Reconnection: Explicit) and implied (Network Attended Reconnection: Implicit by SCP and Network Attended Reconnection: Implicit by Network T-Server) forms. In fact, the only difference between the implicit form for network transfer and network reconnection is a party's relationship to the original call. If the controlling agent disconnects, the action is considered a transfer. If the destination agent disconnects, it is considered a reconnection.

Explicit Reconnection
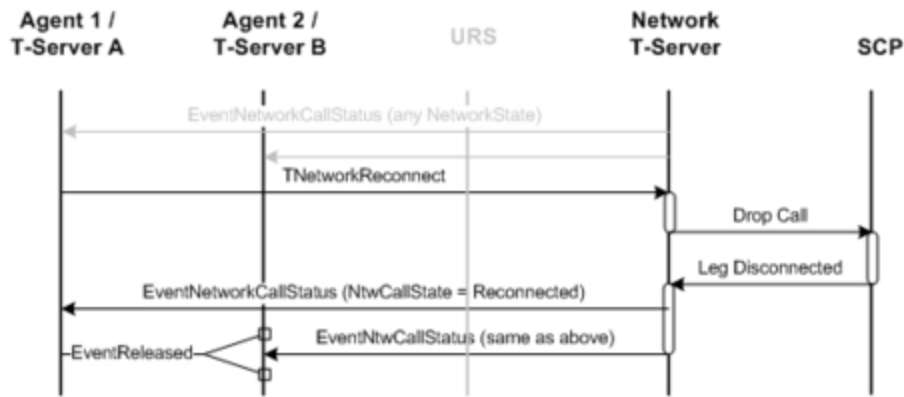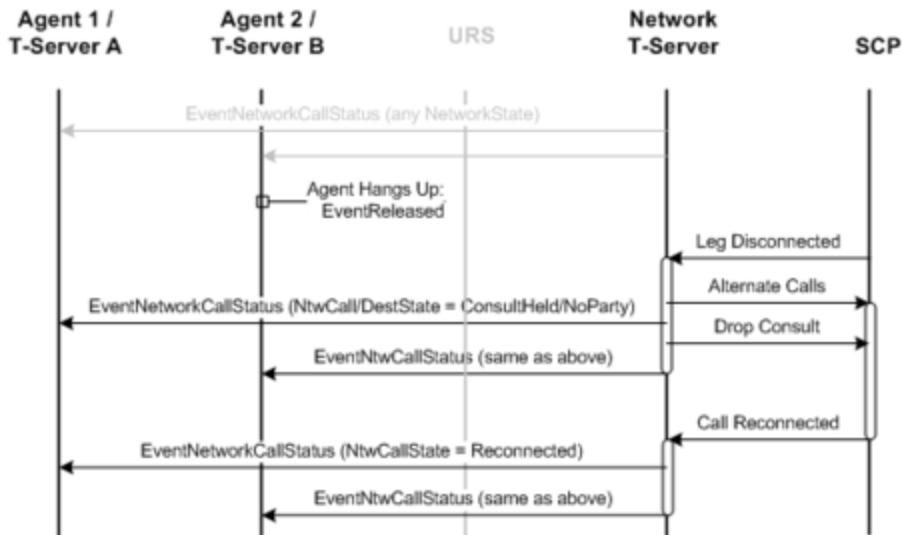

Network Attended Reconnection: Explicit

Implicit Reconnection (by SCP)
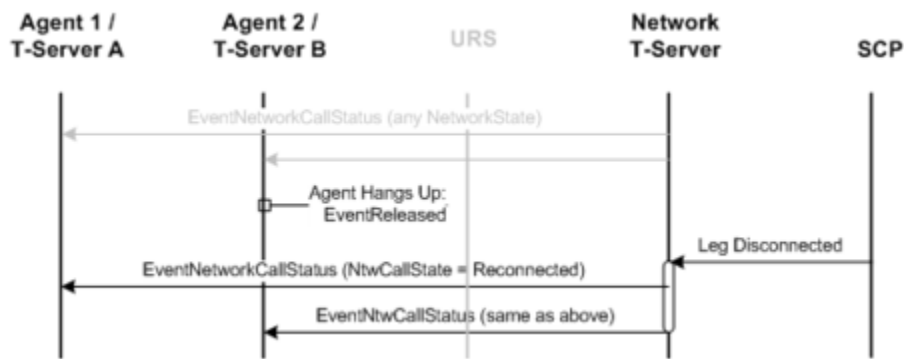
Network Attended Reconnection: Implicit by SCP

## Implicit Reconnection (by Network T-Server)

In some cases, if Agent 2 disconnects while the consultation call is on hold, SCP leaves the consultation leg to be dropped manually. Although this operation then becomes the responsibility of the Network T-Server, the client may get a notification regarding an intermediate state, as shown in the following figure.

Network Attended Reconnection: Implicit by Network T-Server

Caller Abandonment

The following figure illustrates a caller abandonment scenario. If at any time during a multi-party call the origination party disconnects, the SCP may choose to end the call outright, or send a leg disconnected message.
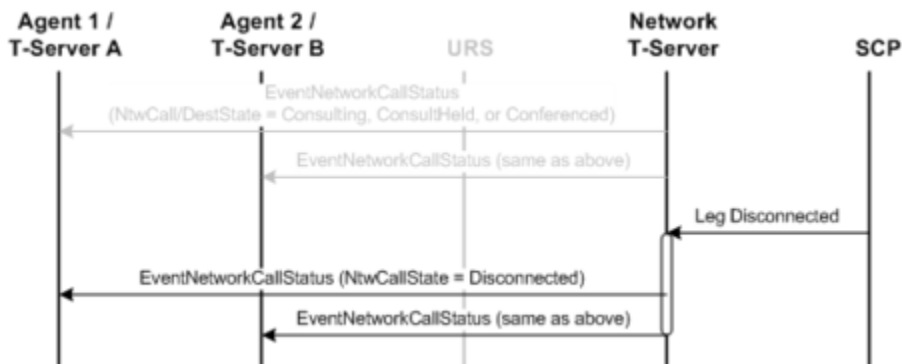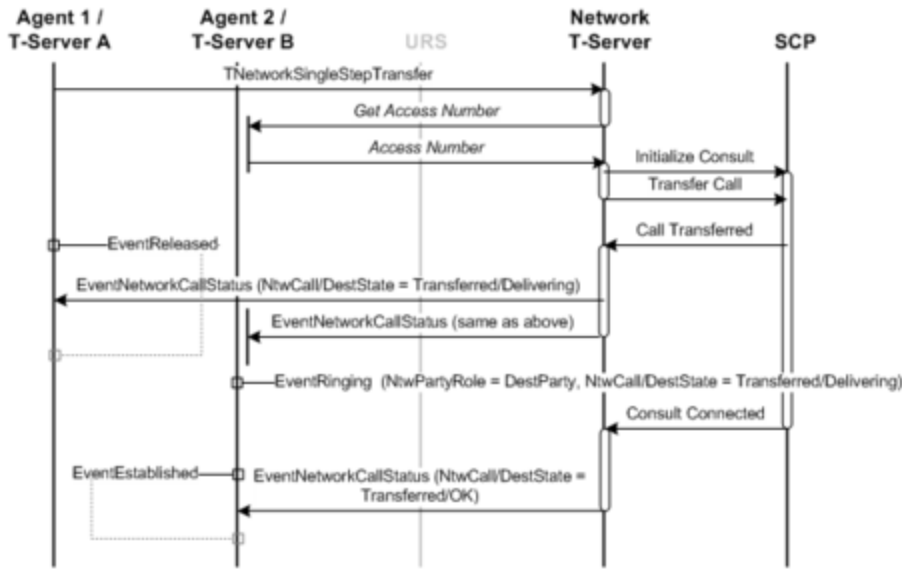

Caller Abandonment

Network Single-Step Transfer

For single-step transfers, the operation is complete immediately after the new call leg is created. Thus, for the external observer, there is no consultation phase.

Network Attended Single-Step Transfer

Premature Disconnection, One Variation

If an agent disconnects the call prior to completion of the consultation leg, a number of things can happen, depending on the current state of the consultation. One variation is detailed in the following figure. In this case, the disconnection occurs prior to the SCP being aware of a pending consultation. The message from the SCP is likely to be `Call Dropped`, as only the caller would remain on the call.
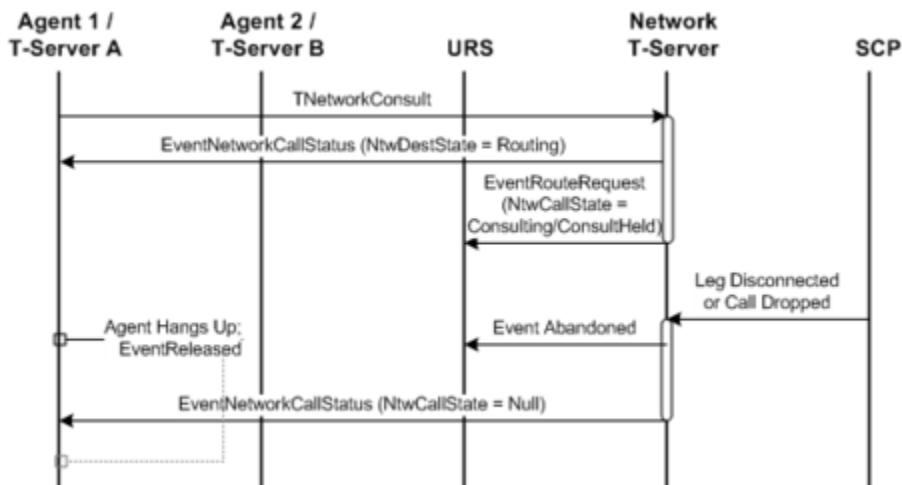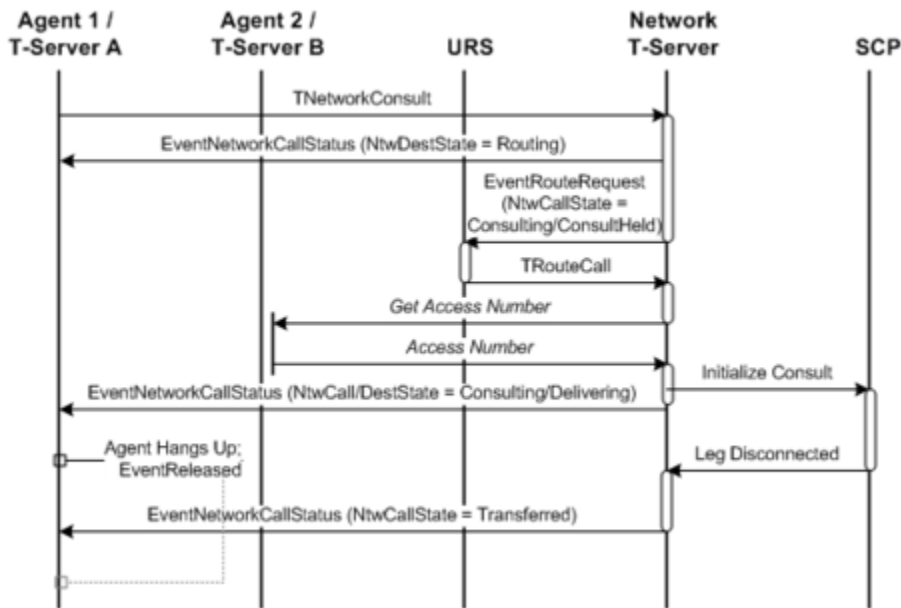

Premature Disconnection: One Variation

Premature Disconnection, a Second Variation

The following figure shows Agent 1 disconnecting after the SCP has been notified of a call, but prior to its connection (or failure). Since the state of the consultation is not known, the call considered to have transferred. The SCP at this point should connect the original caller with the consultation target. The call at this point is treated identically to a normal inbound call that is waiting for its connection status from the SCP. The network attended session is complete, and no further `NetworkCallStatus` events are distributed.

If the delivery results in failure (or a NoAnswer condition), URS should re-route the call to a different target. However, if the call was intended for an explicitly specified target (and URS is not in control of the consultation leg), any call status other than `Connected` results in default routing instructions being returned to the SCP, and the call ending.
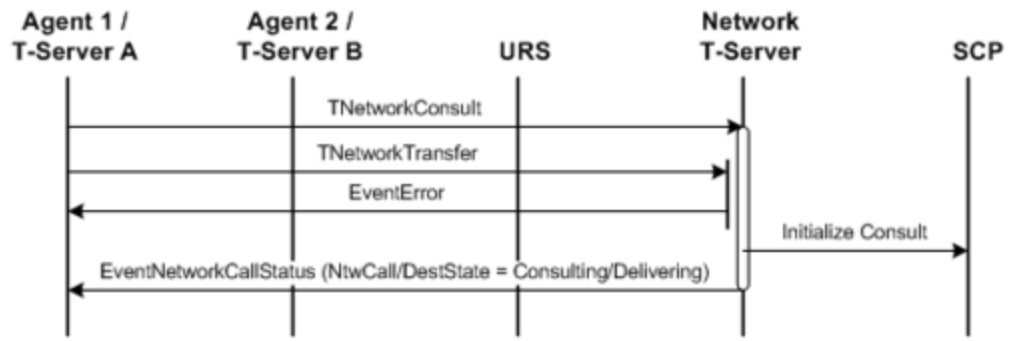

Premature Disconnection: a Second Variation

Transactional Error

A T-Library client is required to wait for a network status message prior to attempting further call control. This message is either the next status message after making its network request or `EventError.` Failure to abide by this results in an error being returned to the requestor.

Transactional Error