This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

SIP Endpoint SDK Overview

SIP Endpoint SDK 8.1.2

2/11/2022

Table of Contents

Welcome

> **Warning**
> **This documentation is outdated and is included for historical purposes only.**

The SIP Endpoint SDK enables you to build a SIP endpoint that can seamlessly connect agent desktop applications with the Genesys SIP Server in order to handle audio and video calls. This overview will help you understand its design goals and its architecture, which are now supported on both .NET and Apple's OS X Lion.

Design Goals

Many SIP softphones that are currently available require an agent to interact with their own separate user interface in addition to that of the agent desktop application. For instance, the agent might have to use the SIP phone interface to answer a call, while other actions — such as holding or releasing the call — would have to be done via agent desktop interface.

In contrast to that, the SIP Endpoint SDK is designed to be integrated into an agent desktop so the agent can use a single user interface to control calls. Genesys recommends that this be done in a way that leaves actual control in the hands of a T-Lib–based agent desktop application, which has a fuller feature set and is also fully supported by Genesys.

The SIP Endpoint SDK is also designed to integrate with the Genesys SIP Server. It supports the SIP, SDP, and RTP/RTCP protocols.

In addition to these principal design goals, the SIP Endpoint SDK supports the following features:

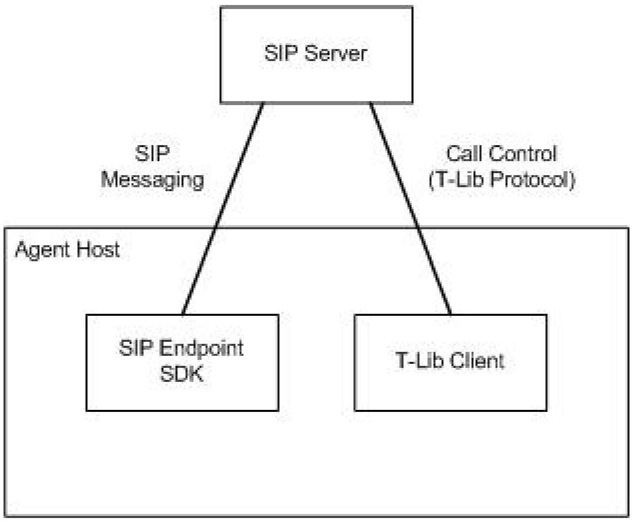
- Traditional call control functionality such as:
    - Establishing inbound and outbound calls
    - Hold and retrieve
    - Transfers and conference calls (Note that SIP Endpoint SDK does not handle transfers and conference calls directly. You must use third-party call control in order to enable these features.)
- Multi-line call handling

Note: Multi-line call handling is available up to the limitations on the number of simultaneous connections supported by the licensing available to Genesys at this time.

Architecture

The SIP Endpoint SDK is a software component that resides on the agent's computer. It uses the SIP protocol to communicate with the Genesys SIP Server and supports both thin (or 3-tier) and thick (or rich) T-Lib clients.

In SIP messaging terms, the SIP Endpoint is a user agent. The contact center agent's computer should also host a Genesys T-Lib client that works with SIP Server and provides Genesys agent-related functionality, as shown below.



As you can see, the SIP Endpoint SDK handles the SIP messaging, and the T-Lib Client uses the T-Lib protocol to handle call control.

SDK Components

The components of the SIP Endpoint SDK are described on the following pages:

- SIP Endpoint SDK for NET Components
- SIP Endpoint SDK for Apple OS Components.

Supported Codecs

SIP Endpoint SDK 8.1.2 supports the following codecs:

- PCMU/8000 (G.711/mu-law)
- PCMA/8000 (G.711/A-law)
- G722/16000 (G.722)
- ISAC/16000 (iSAC/16kHz — internet Speech Audio Codec)
- ISAC/32000 (iSAC/32kHz)
- ILBC/8000 (iLBC — internet Low Bitrate Codec)

- VP8 video

Working with the SIP Endpoint SDK for .NET

The SIP Endpoint SDK for .NET distribution includes the following files, which you can use "as is" in your custom applications:

- Genesyslab.Sip.Endpoint.dll
- Genesyslab.Sip.Endpoint.Provider.Genesys.dll

These files are located in the `\QuickStart\Bin` directory at the root level of the SIP Endpoint SDK directory.

The SIP Endpoint SDK also depends on the following Genesys and third-party libraries, which should be present in the working directory:

- intl.dll
- libgio-2.0-0.dll
- libglib-2.0-0.dll
- libgmodule-2.0-0.dll
- libgobject-2.0-0.dll
- libgthread-2.0-0.dll
- libnice.dll
- zlib1.dll
- Genesyslab.Core.dll
- Genesyslab.Platform.Commons.Collections.dll
- Genesyslab.Platform.Commons.Connection.dll
- Genesyslab.Platform.Commons.dll
- Genesyslab.Platform.Commons.Protocols.dll
- Genesyslab.Platform.Logging.dll
- Genesyslab.Platform.Management.Protocols.dll
- Genesyslab.Sip.Endpoint.dll
- Genesyslab.Sip.Endpoint.Provider.Genesys.dll

Learning More

To continue learning about the SIP Endpoint SDK, we recommend you read the pages describing SIP-Based Third-Party Call Control to understand messaging patterns. After that, you should be ready to use the Deployment Guide to install the SDK on your system.

Once you have installed the SDK, you may want to check the following pages from the Developer's

Guide:

- SIP Endpoint SDK Configuration for .NET and for Apple OS
- SIP Endpoint SDK Disaster Recovery and Geo-Redundancy for .NET

SIP Endpoint SDK for .NET Components

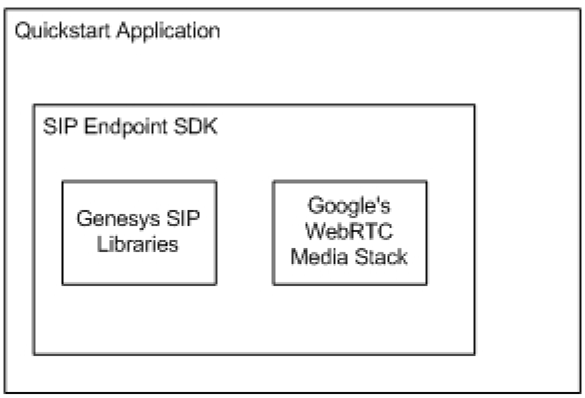
Warning

This documentation is outdated and is included for historical purposes only.

The SIP Endpoint SDK distribution consists of the following main components:

- The SDK, which provides all of the SIP-related functionality. The SDK also provides an interface that you can use to integrate it into different GUI-based applications.
- A sample QuickStart application which is built on the SIP Endpoint SDK.

The SDK itself runs on top of the Genesys SIP libraries and Google's WebRTC media stack, while the QuickStart application runs on top of the SDK, as shown here:



Apple OS Components

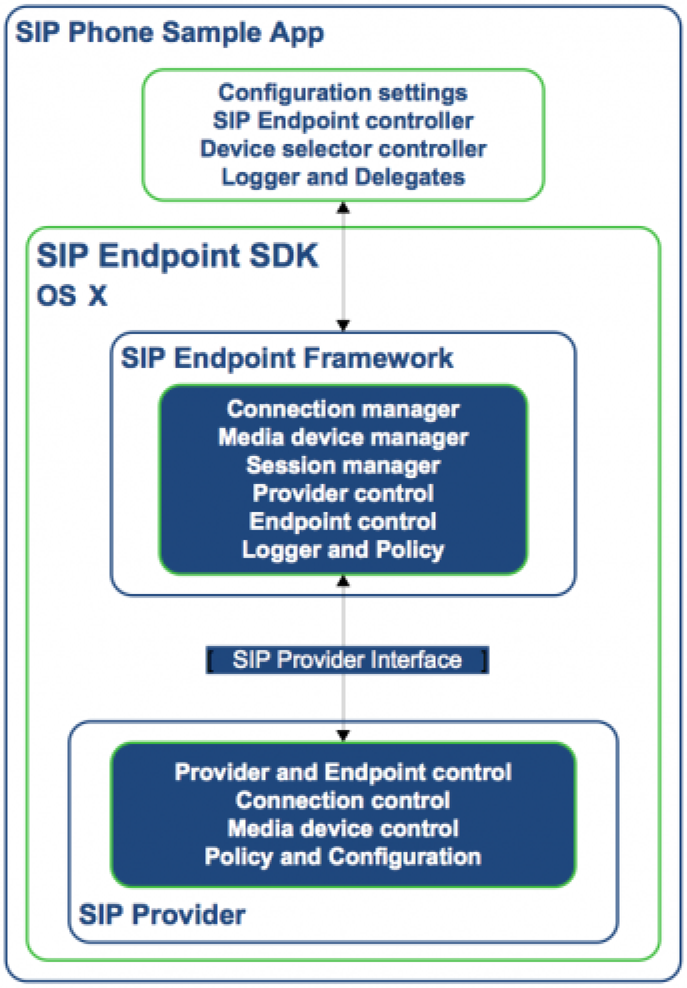
Warning

This documentation is outdated and is included for historical purposes only.

The SIP Endpoint SDK distribution consists of the following main components:

- The SDK itself, which provides all of the SIP-related functionality by way of the SIP Endpoint Framework. The SDK also provides an interface that you can use to integrate it into different GUI-based applications.
- A sample application which is built on the SIP Endpoint SDK.

The SDK itself communicates with a SIP Provider interface, which links the SDK to the SIP Provider, as shown in the following diagram.


SIP Phone Sample App
Configuration settings
SIP Endpoint controller
Device selector controller
Logger and Delegates
SIP Endpoint SDK
OS X
SIP Endpoint Framework
Connection manager
Media device manager
Session manager
Provider control
Endpoint control
Logger and Policy
SIP Provider Interface
Provider and Endpoint control
Connection control
Media device control
Policy and Configuration
SIP Provider

SIP-Based Third-Party Call Control

Warning

This documentation is outdated and is included for historical purposes only.

This section provides sequence diagrams and descriptions of the call scenarios supported by SIP Server and the SIP Endpoint SDK, which are achieved using the T-Lib API. It also demonstrates how the T-Lib API is mapped into SIP messaging.

Note that the SIP Endpoint SDK supports the talk/hold NOTIFY extension.

T-Lib-Initiated MakeCall re-INVITE

Warning

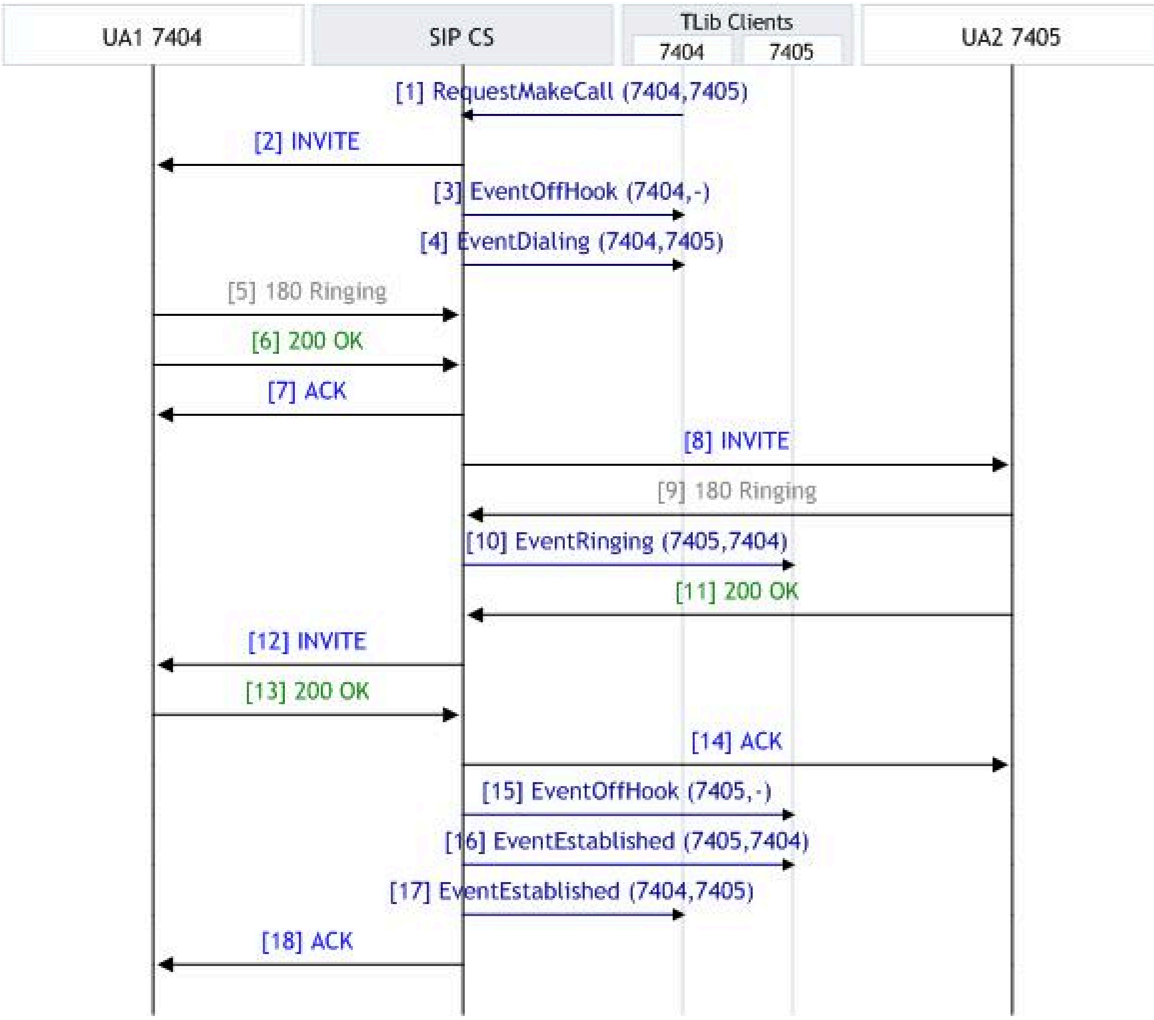
This documentation is outdated and is included for historical purposes only.

Description

UA1 calls UA2. If the UA1 endpoint does not support the REFER method, then the re-INVITE mechanism could be used. The main disadvantages of re-INVITE are:

- The phone may not reflect the real call progress unless early media is activated between both endpoints.
- Re-INVITE makes it next to impossible for third-party switches to track call topology and accumulate accurate call detail records (CDR).

Diagram


UA1 7404
SIP CS
TLib Clients
7404
7405
UA2 7405
[1] RequestMakeCall (7404,7405)
[2] INVITE
[3] EventOffHook (7404,-)
[4] EventDialing (7404,7405)
[5] 180 Ringing
[6] 200 OK
[7] ACK
[8] INVITE
[9] 180 Ringing
[10] EventRinging (7405,7404)
[11] 200 OK
[12] INVITE
[13] 200 OK
[14] ACK
[15] EventOffHook (7405,-)
[16] EventEstablished (7405,7404)
[17] EventEstablished (7404,7405)
[18] ACK

T-Lib-Initiated MakeCall INVITE

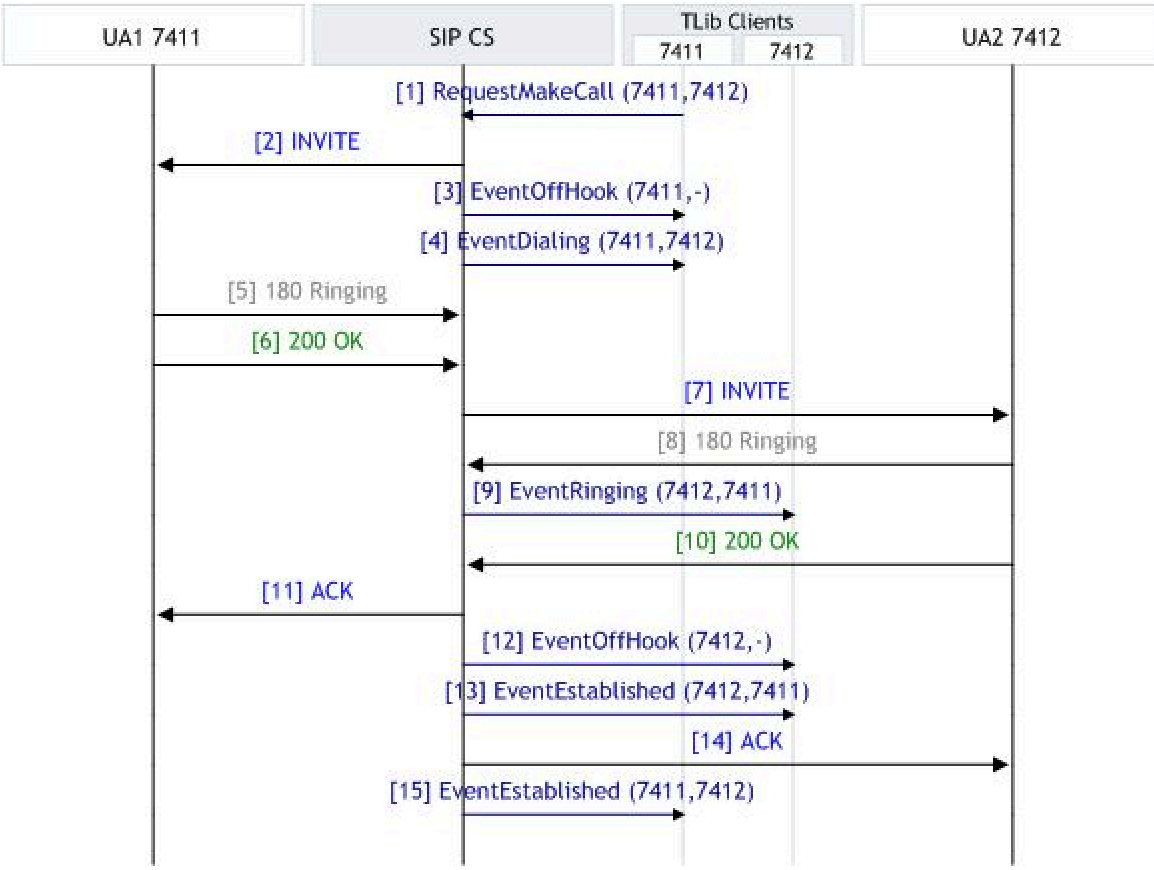
Warning
This documentation is outdated and is included for historical purposes only.

Description

RFC 3725 is a configuration flag that can be set to 2 (the default, which causes the T-Lib-Initiated MakeCall re-INVITE to be used) or 1. When the parameter is set to 1, the call flow will be as given below. Notice in this case that according to the RFC (best current practices for third-party call control) UA1 does not get an ACK until UA2 has responded with `200 OK`. In this case, SIP Server does not send a re-INVITE to UA1.

Note: Usually, the re-INVITE call flow is the preferred use case. This is the default behavior of SIP Server, unless the RFC3725 parameter is set to 1.

Diagram


UA1 7411
SIP CS
TLib Clients
7411
7412
UA2 7412
[1] RequestMakeCall (7411,7412)
[2] INVITE
[3] EventOffHook (7411,-)
[4] EventDialing (7411,7412)
[5] 180 Ringing
[6] 200 OK
[7] INVITE
[8] 180 Ringing
[9] EventRinging (7412,7411)
[10] 200 OK
[11] ACK
[12] EventOffHook (7412,-)
[13] EventEstablished (7412,7411)
[14] ACK
[15] EventEstablished (7411,7412)

T-Lib Termination TReleaseCall

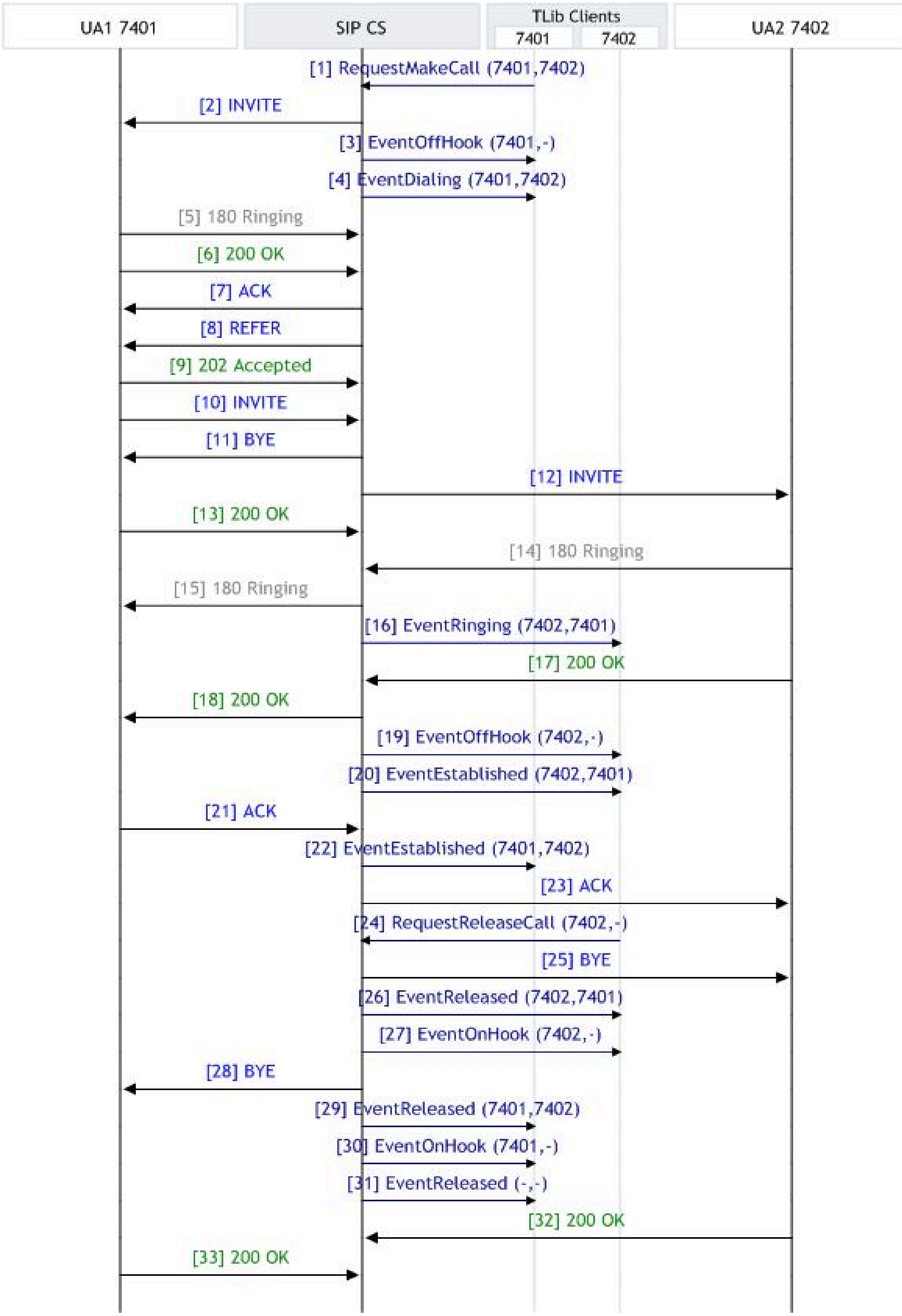
Warning

This documentation is outdated and is included for historical purposes only.

Description

The T-Lib client initiates the termination of the call by invoking TReleaseCall from the agent desktop at UA2. SIP Server terminates the existing call from UA1 to UA2 by sending two BYE requests.

Diagram


UA1 7401
SIP CS
TLib Clients
7401
7402
UA2 7402
[1] RequestMakeCall (7401,7402)
[2] INVITE
[3] EventOffHook (7401,-)
[4] EventDialing (7401,7402)
[5] 180 Ringing
[6] 200 OK
[7] ACK
[8] REFER
[9] 202 Accepted
[10] INVITE
[11] BYE
[12] INVITE
[13] 200 OK
[14] 180 Ringing
[15] 180 Ringing
[16] EventRinging (7402,7401)
[17] 200 OK
[18] 200 OK
[19] EventOffHook (7402,-)
[20] EventEstablished (7402,7401)
[21] ACK
[22] EventEstablished (7401,7402)
[23] ACK
[24] RequestReleaseCall (7402,-)
[25] BYE
[26] EventReleased (7402,7401)
[27] EventOnHook (7402,-)
[28] BYE
[29] EventReleased (7401,7402)
[30] EventOnHook (7401,-)
[31] EventReleased (-,-)
[32] 200 OK
[33] 200 OK

T-Lib-Initiated Hold-Retrieve

Warning

This documentation is outdated and is included for historical purposes only.

Description

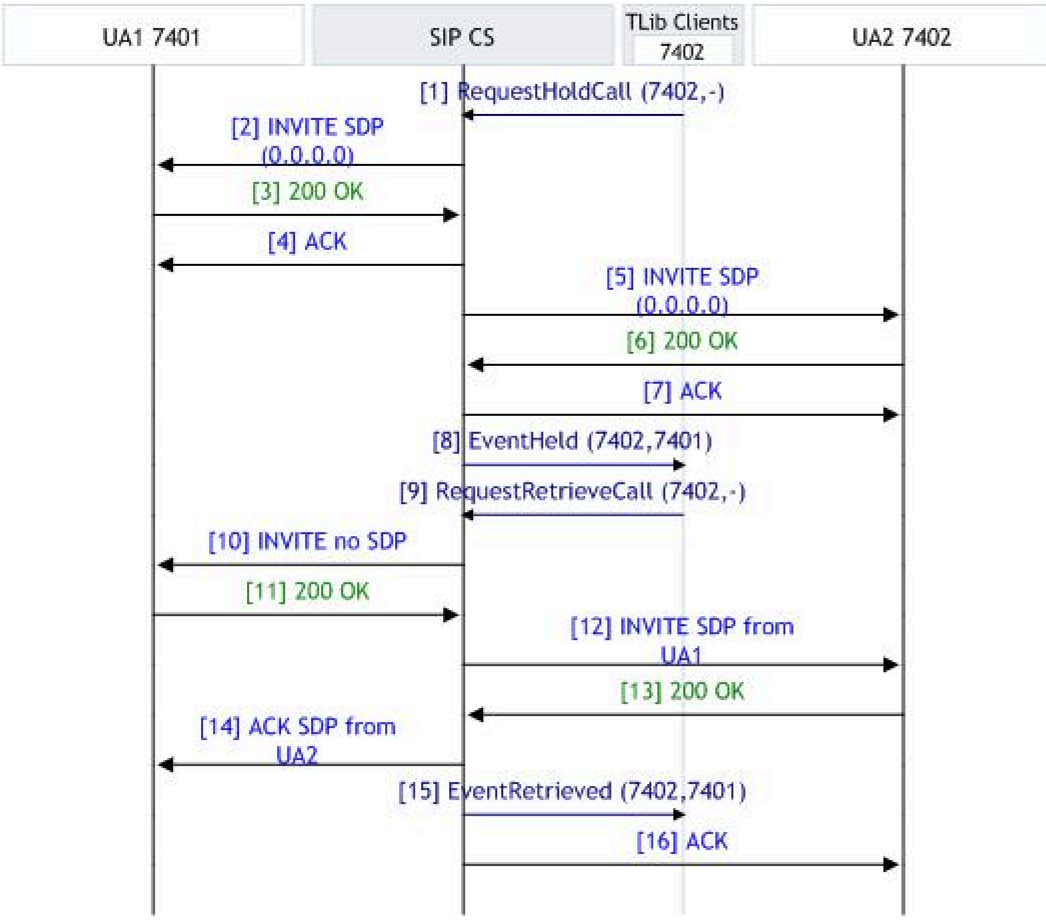
SIP Server's Hold operation can be configured to play music while the customer is placed on hold by an agent. The music-on-hold configuration is global — if it is enabled, all held parties hear music; if it is not enabled, all held parties hear silence.

Hold-with-silence only involves stopping the RTP streams. The Retrieve operation resumes the RTP streams.

There are multiple methods for stopping the RTP streams. SIP Server uses IP address 0.0.0.0 in the `c=` attribute as the only universally accepted method.

Because some endpoints may change the RTP port address when trying to retrieve a call, the re-INVITE with delayed SDP offer-answer negotiation shown in the following diagram is necessary.

Diagram


UA1 7401
SIP CS
TLib Clients
7402
UA2 7402
[1] RequestHoldCall (7402,-)
[2] INVITE SDP
(0.0.0.0)
[3] 200 OK
[4] ACK
[5] INVITE SDP
(0.0.0.0)
[6] 200 OK
[7] ACK
[8] EventHeld (7402,7401)
[9] RequestRetrieveCall (7402,-)
[10] INVITE no SDP
[11] 200 OK
[12] INVITE SDP from
UA1
[13] 200 OK
[14] ACK SDP from
UA2
[15] EventRetrieved (7402,7401)
[16] ACK

T-Lib-Initiated Hold-Retrieve Music on Hold

T-Lib-Init SingleStepTransfer re-INVITE

T-Lib-Initiated SingleStepTransfer REFER

T-Lib-Init SingleStepTransfer User Busy

T-Lib-Initiated Consult Call Single EndPoint

T-Lib-Initiated Consult Transfer re-INVITE

T-Lib-Initiated 3-Way Conf Central Mixing

Answer Call Functionality

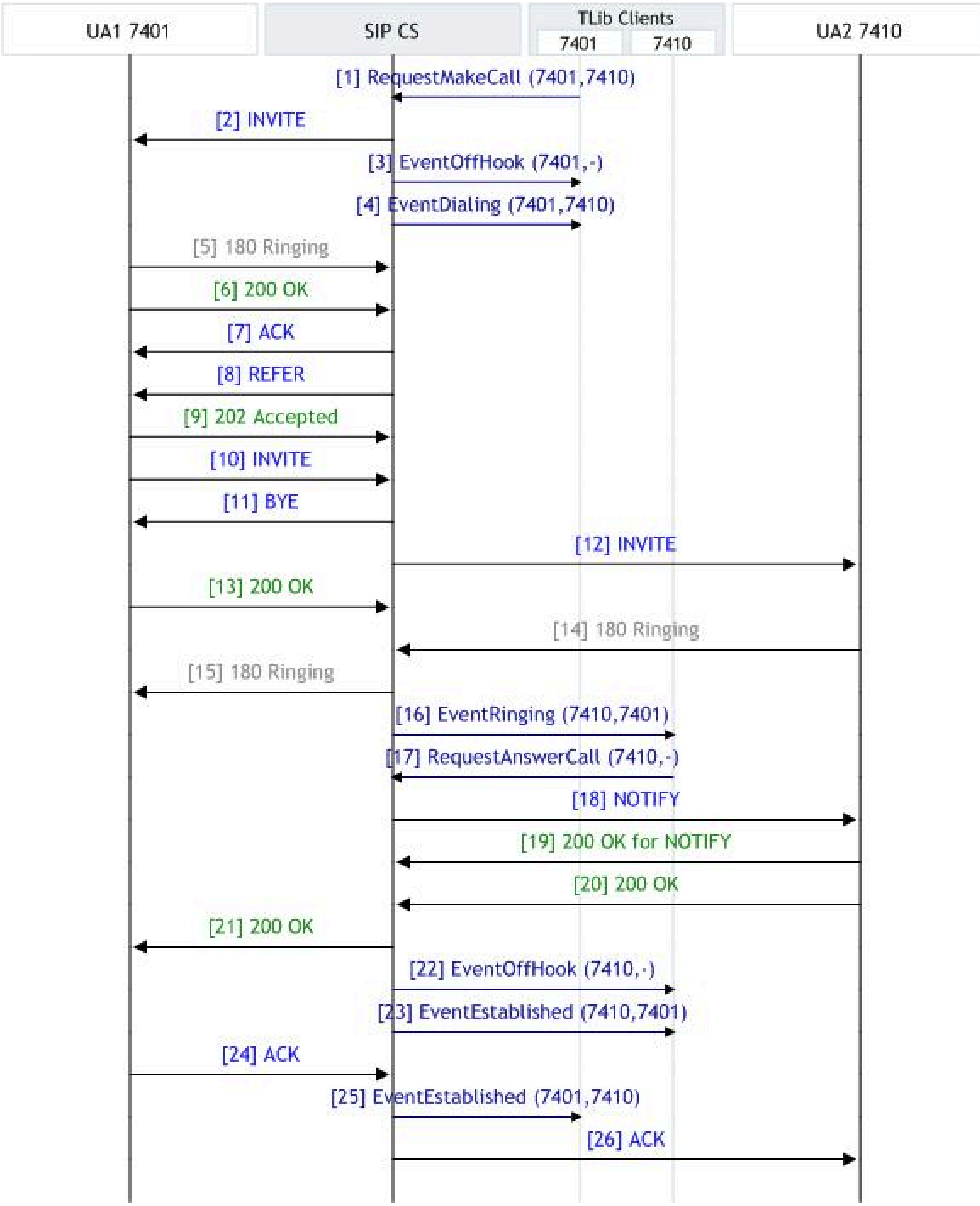
Warning

This documentation is outdated and is included for historical purposes only.

Description

If an endpoint (SIP phone) supports remote CTI control by providing the SIP extensions event package known as "BroadSoft" SIP extensions, SIP Server allows SIP Server clients to answer the call by means of the TAnswerCall request. When responding to the TAnswerCall request, SIP Server will send an event with the header "Event" and a header value of "talk" to the SIP phone, in a ringing state of NOTIFY. This event instructs the SIP phone to answer the call without human interference. Note that SIP phones that support such an event package should send a 180 Ringing message header of "Allow-Events:", which should contain "talk" inside its value.

Diagram


UA1 7401
SIP CS
TLib Clients
7401
7410
UA2 7410
[1] RequestMakeCall (7401,7410)
[2] INVITE
[3] EventOffHook (7401,-)
[4] EventDialing (7401,7410)
[5] 180 Ringing
[6] 200 OK
[7] ACK
[8] REFER
[9] 202 Accepted
[10] INVITE
[11] BYE
[12] INVITE
[13] 200 OK
[14] 180 Ringing
[15] 180 Ringing
[16] EventRinging (7410,7401)
[17] RequestAnswerCall (7410,-)
[18] NOTIFY
[19] 200 OK for NOTIFY
[20] 200 OK
[21] 200 OK
[22] EventOffHook (7410,-)
[23] EventEstablished (7410,7401)
[24] ACK
[25] EventEstablished (7401,7410)
[26] ACK