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 Simulator Deployment Guide

Using SIP Endpoint Simulator

12/14/2025

Using SIP Endpoint Simulator

Contents

- 1 Using SIP Endpoint Simulator
  - 1.1 User Interface Elements
  - 1.2 Manual Controls and Commands

To use SIP Endpoint Simulator:

1. Right-click the mouse button to select an object (site, DN, and so on) and open the menu with commands available for the selected object.
2. Select the command from the menu and enter the parameters required for the selected command.
3. Click OK to run the command.

User Interface Elements

The main window of the SIP Endpoint Simulator consists of two panes—device view (left) and call view (right)—that represent the same current state information.

The *device view* pane contains the following elements:

- Sites—listed in order of appearance in the configuration file.
- DNs—listed in order of appearance in the configuration file, with SIP sections (and some in-line options) shown in parentheses.
- (optional) SIP dialog groups for trunks and VoIP services.
- T-Library calls (phone-based icons) and SIP dialogs (colored dots) that are active on the DN (in order of creation):
  - T-Library calls are named by the Connection ID.
  - The name of the SIP dialog consists of the internal dialog ID, TLDR result (if applicable), selected audio codec, abbreviated SIP Call-ID, and a reference to the peer's dialog ID (if applicable).

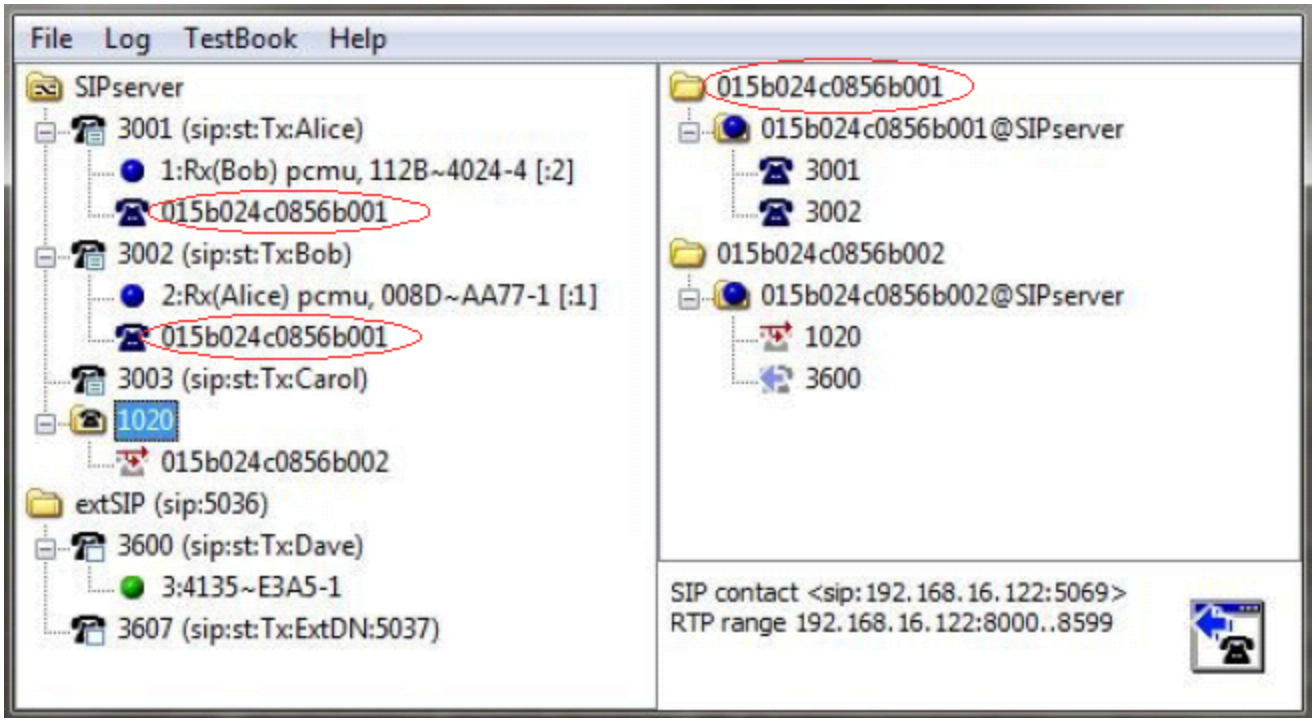
Generally, it is impossible to match a SIP dialog with a corresponding T-Library call (especially when the **dual-dialog-enabled=false** option is used in SIP Server), so SIP Endpoint Simulator does not make an attempt.

The *call view* pane contains the following elements:

- A call thread that is the collection of related calls (main/consultation for a single site, linked calls in multi-site deployments).
- A local call with a label consisting of the Connection ID and the site name.
- A party on that call with the DN name label. Except for the label, this object is essentially the same as "T-Library call" in the device view.

For all elements, the icon usually reflects the current state of that element (although for multiple independent sub-states not all info is visible). You can access the Icons Help from the Help menu.

In the screenshot below, you can see two call threads, each with a single call. The first call—ConnectionID ends with b001—has two parties (3001 and 3002) in the Connected state. The device view pane displays corresponding SIP dialogs connected to each other and the voice path verified (3001 generates the “Alice” tone and hears “Bob” and vice versa). The second call is an inbound call from external DN 3600 (which is not registered on SIP Server, so no T-Library party there) to Routing Point 1020 (no SIP dialog present), in the Routing state.



Manual Controls and Commands

For most elements in both panes, left-clicking the mouse selects the object and right-clicking opens the context-sensitive menu. Operations included in those menus are briefly described in the next sections. As a general rule:

- Operations for SIP dialogs (colored-dot icons) include SIP commands (1pcc).
- Operations for T-Library parties (phone-based icons) include T-Library commands (3pcc).
- Operations for a DN (second-level elements) include both 1pcc and 3pcc commands.

Per standard convention, menu items with the name ending in ellipsis ('...') open a dialog box requesting operation parameters and perform the action only after a value is entered and the Ok button or Enter key are pressed (Cancel button or Esc key aborts the operation). The dialog box prompt shows a very brief description of the value format for easy reference. See below for more complete description of parameters for each operation.

For most operation parameters entered in GUI dialog boxes, SIP Endpoint Simulator keeps a handful of last values for each dialog (but no more than 1000 total entries). That history is persistent, saved between sessions in the `esttt.history` file in the folder where the Simulator executable is located.

Site-level commands

Two sets of operations are defined for the site element (only one of them is usually active at any time), with menu items worded differently depending on the presence of T-Library connection (the

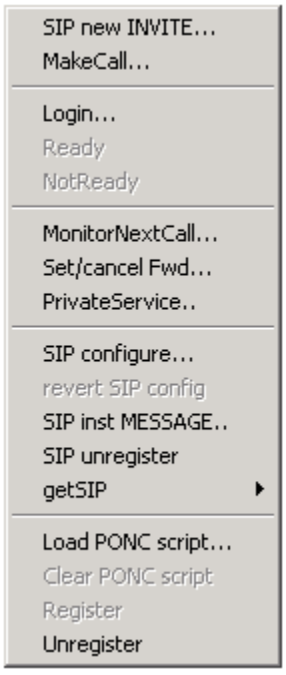
server configuration option) and SIP registration settings (the **sip-register** configuration option):

- **Connect** (T-Library present)—Opens the T-Library connection to the specified T-Server/SIP Server, registers all SIP-enabled DNs with SIP (sending REGISTER requests to a primary and backup SIP proxy) and T-Library (sending TRegisterAddress to T-Server).
- **Disconnect** (T-Library present)—Closes the T-Library connection (which automatically cancels all T-Library registrations) and unregisters all SIP-enabled DNs (sending REGISTER with zero expiration time).
- **(re-)register SIP** (no T-Library, **sip-register=true**)—Registers all SIP-enabled DNs with SIP.
- **SIP unregister** (no T-Library, **sip-register=false**)—Unregisters all SIP-enabled DNs.

No menu is shown for sites without the T-Library connection and when **sip-register=false**, since no actions can be logically performed for them.

DN-level commands for telephony and agent operations

A set of operations included in the DN-level menu depends on the T-Library registration status and the DN type (as reported by T-Server), and presence of the SIP configuration, as follows:



- **SIP new INVITE...**(regular DN, SIP)—Sends a new INVITE request from the DN (that is, makes a 1pcc call). Use the following parameters:
  destination-DN[@*SIP-proxy-host:port*][,sip=(*on-invite-script*)]

  where:
  - *destination-DN*—The destination of the call (user part of the request-URI in the INVITE).
  - *SIP-proxy-host:port*—(Optional) The location of the SIP proxy/gateway (usually it will be SIP Server).

If it is not specified, the value of the **sip-proxy** option is used (which is the most common case).

- *on-invite-script*—The custom SIP script for this particular call, in the same format as the **on-mkcall** configuration option, or the parameters of INVITE message.

  For example:

  `3002` - Sends an INVITE to 3002 through the configured sip-proxy option (SIP Server)

  `3605@pixie:5055` - Sends an INVITE directly to the specified host/port (without SIP Server)

  `3002,sip=(INVITE=(SDP=no))` - Uses the custom SIP command (do not include SDP into INVITE)

  `3002,sip=(SDP=no)` - Uses the custom SIP command, short format

- **MakeCall...** (regular DN, T-Library)—Makes a new 3pcc call (T-Library function TMakeCall). Use the following parameters:
  destination-DN[@*location*][,ty=*type*][,uu=(*user-data*)][,ext=(*extensions*)]

  where:

  - *destination-DN*—The destination of the call (AttributeOtherDN in TMakeCall).
  - *location*—(Optional) The destination location (AttributeLocation).
  - *ty*—(Optional) The MakeCallType (never used with SIP Server).
  - *uu* and *ext*—(Optional) UserData and AttributeExtensions in the request, using standard text representation of TKVList (comma-separated list of "key"="value").

    For example:

    `3002` - Makes a 3pcc call to 3002

    `3301@epi-SIP-sw2` - Makes a call to 3301 at location epi-SIP-sw2 (see the multi.conf sample)

    `4401,uu=(GSW_CALL_TYPE='ENGAGING',GSW_SESSION_DBID=101)` - Makes a call to 4401 and attaches specified user data to the call

- **MakePredictiveCall...** (Routing Point/queue, T-Library)—Makes a 3pcc predictive call (TMakePredictiveCall). Use the following parameters:
  destination-DN[@*location*][,tout=*timeout*][,uu=(*user-data*)][,ext=(*extensions*)]

  where:

  - *destination-DN*—The destination of the call (AttributeOtherDN in TMakeCall).
  - *location*—(Optional) The destination location (AttributeLocation).
  - *timeout*—(Optional) The ring_timeout value (default is 0, which tells T-Server to use the default/configured value).
  - *uu* and *ext*—(Optional) UserData and AttributeExtensions in the request.

- **Login...** (regular DN, T-Library)—Logs an agent in (TAgentLogin). Use the following parameters:
  agent-ID[/*queue*]

  where:

  - *agent-ID*—The agent ID. The parameter string may include $DN that will be replaced with DN digits.
  - *queue*—The ACD queue (optional for SIP Server).

- **Logout** (regular DN, T-Library)—Logs an agent out (TAgentLogout).
- **Ready** (regular DN, T-Library)—Sets an agent to Ready state (TAgentReady).
- **NotReady** (regular DN, T-Library)—Sets an agent to NotReady state (TAgentNotReady).
- **MonitorNextCall...** (regular DN, T-Library)—Enables call monitoring (TMonitorNextCall). Use the following parameters:
  agent-DN[/coach|talk|mute][/agent|call][/all] or *agent-DN*/cancel

  where:

  - *agent-DN*—The DN to be monitored.
  - if **/coach** is specified, sets the monitor mode to **coach**
  - if **/talk** is specified, sets the monitor mode to **connect**
  - if **/mute** is specified, sets the monitor mode to **mute**
    When the monitor mode is not specified, Simulator does not add the MonitorMode key to AttributeExtensions, thus using the T-Server default (which is usually silent monitoring, but may be changed by TServer options)
  - if **/agent** is specified, set monitor scope to **agent**
  - if **/call** is specified, set monitor scope to **call**
  - if **/all** is specified, set monitor type to MonitorAllCalls (default is MonitorOneCall)
  - using the /cancel option in the parameter dialog (for example, "3002/cancel") cancels the call monitoring with the TCancelMonitoring function
- **Set/cancelFwd...** (regular DN, T-Library)—Sets or cancels T-Library call forwarding (either TCallSetForward or TCallCancelForward, depending on the DN presence in parameters). Use the following parameters:
  [*DN-to-forward*] [/*mode*]

  where:

  - *DN-to-forward*—The forwarding destination for TCallSetForward, or empty for TCallCancelForward.
  - /*mode*—The forwarding mode, optional for Set (default is /1), mandatory for Cancel, as follows:
    - /**1** = ForwardModeUnconditional
    - /**2** = ForwardModeOnBusy
    - /**3** = ForwardModeOnNoAnswer
    - /**4** = ForwardModeOnBusyAndNoAnswer
    - /**5** = ForwardModeSendAllCalls
- **PrivateSvc** (all DNs, T-Library)—Passes information and request services for this DN (TPrivateService). Use the following parameters:
  id [/*extensions*]

  Refer to the Framework 8.1 SIP Server Deployment Guide for a list of service IDs and their parameters.

DN-level SIP commands

- **SIP configure...** (SIP-enabled DN)—Modifies the SIP configuration of the particular DN. Parameters must be specified in the same format as in the configuration file, for example:
  on-invite=(ring=(SDP)),play=ExtDN

  Notes:

  - New parameters are added to the existing configuration, replacing options with the same name, but not changing other options. If the intention is to discard old values and apply only the new configuration, start the list with replace keyword, for example: replace,sip=st,play=ExtDN.
  - Most options takes effect for the next call. As an exception (and special feature), the **play** option is applied to all currently active SIP dialogs as well (so the new file/tone starts playing immediately).
  - SIP stack parameters cannot be changed with this command
  - In addition to regular options described, two special options might be used in this command—on-invite-once and on-mkcall-once—have the same effect as their regular counterparts, but are applied to the next call only (and automatically reset after the first use).

- **revert SIP config** (SIP-enabled DN)—Reverts the SIP configuration to the value taken from the configuration file.

- **SIP inst MESSAGE...** (SIP-enabled DN)—Sends the SIP instant MESSAGE (page mode only, RFC 3428). Use the following parameters:
  DN, message-text or *DN,content-type* ="*message-text*"[,x=(*extra-headers*)]

  where:

  - *DN*—The destination of the instant message (with the optional SIP proxy, as in SIP new INVITE).
  - *content-type*—Changes the Content-Type in the MESSAGE request (in the short form text/plain is used).
  - *message-text*—The new lines encoded as '|' , for example: "Hello|just testing|".
  - *extra-headers*—(Optional) A list of additional SIP headers, in the "header=value" format.

- **SIP register** (SIP-enabled DN)—Sends the REGISTER request for the DN, which can be either a new SIP registration or renewal of the existing one.

- **SIP unregister** (SIP-enabled DN)—Sends the REGISTER request with zero expiration.

- **getSIP » device** (SIP-enabled DN)—Opens a window with SIP history of all device-level messages (REGISTER, SUBSCRIBE, and so on). It is also activated by double-clicking the DN element.

- **getSIP » id (last inc)**—Opens SIP history of the last incoming SIP dialog.

- **getSIP » id (last out)**—Opens SIP history of the last outgoing SIP dialog.

Note: By design, the content of the SIP history window is not updated in real time. To get an update, close the window and input the corresponding command again.

Other DN level commands

- **Load PONC script...**(all DNs)—Loads the play-on-new-call (PONC) script to the DN. For Routing Points, the parameter has the same format as it does in the Treatment command described below.
  For example:

  `play=music/ring_back` - Applies the ring-back treatment to a new call on this Route DN

- **Clear PONC script** (all DNs)—Clears the PONC (or Routing) script, stopping any further automatic action for new calls arriving to this device.

- **Register** (all DNs, T-Library)—Registers the DN (TRegisterAddress).

- **Unregister** (all DNs, T-Library)—Unregisters the DN (TUnregisterAddress).

T-Library party commands

A set of operations for T-Library call parties depends on the party state (as reported by T-Server in events). All operations listed in this section have two implicit arguments: ThisDN and ConnectionID.

- **party » Answer** (regular DN, Alerting state)= TAnswerCall
- **party » Hold** (regular DN, Connected state)= THoldCall
- **party » Retrieve** (regular DN, Held state)= TRetrieveCall
- **party » Release** (regular DN, any state)= TReleaseCall
- **party » 1stepTransfer...** (regular DN, Connected state)= TSingleStepTransfer
- **party » 1stepConference...** (regular DN, Connected state)= TSingleStepConference

For the above party commands, use the following parameters:
destination-DN[@*location*][,uu=(*user-data*)][,ext=(*extensions*)]
where:
destination-DN—The destination of the call (AttributeOtherDN in TMakeCall).
location—(Optional) The destination location (AttributeLocation).
user-data and *extensions*—(Optional) UserData and AttributeExtensions in the request, using standard text representation of TKVList (comma-separated list of "key"="value").

- **party » InitTransfer...** (regular DN, Connected state)= TInitiateTransfer

- **party » InitConference...** (regular DN, Connected state)= TinitiateConference
  For SIP Server (and many other T-Servers) these InitTransfer and InitConference commands do the same thing: the currently active call is placed on hold and a new consultation call is initiated (some T-Servers require Initiate/Complete operations to match each other). Parameters for both are in the same format as for the **1stepTransfer/Conference...** above.

- **party » Alternate** (regular DN, Connected/Held states)= TAlternateCall

- **party » Reconnect** (regular DN, Connected/Held states)= TReconnectCall

- **party » TransferComplete**(regular DN, Connected/Held states)= TCompleteTransfer

- **party » ConferenceComplete** (regular DN, Connected/Held states)= TCompleteConference. All four operations—Alternate, Reconnect, TransferComplete, and ConferenceComplete—are only available when two parties are present on the DN, one in Connected and the other in Held state. They might be performed on either party with exactly the same result (SIP Endpoint Simulaator reorders parties in correct order when generating corresponding T-Library request).

- **party » Route...** (Routing Point)—Routes the call to the specified destination. Use the following parameters:
  destination-DN[@*location*][,ty=*type*][,ext=(*extensions*)]

  where:

  - *destination-DN*—The destination of the call (AttributeOtherDN in RequestRouteCall).
  - *location*—(Optional) The destination location (AttributeLocation).
  - *type*—(Optional) The RouteType (default is 0=RouteTypeUnknown).
  - *extensions*—(Optional) AttributeExtensions in the request, using standard text representation of TKVList (comma-separated list of "key"="value").

- **party » Treatment...** (Routing Point)—Applies the treatment to the call on a Routing Point. Use one of the following parameters:
  - play=*filename* [,t=*duration*] - TreatmentMusic (with optional DURATION parameter)
  - annc=(*announcement-prameters*) - TreatmentPlayAnnouncement
  - app=(*play-app-prameters*) - TreatmentPlayApplication
  - collect=(*collect-prameters*) - TreatmentCollectDigits
  - record=*filename* - TreatmentRecordUserAnnouncement

- **party » SendDTMF...** (regular DN, Connected state) = TSendDTMF (parameter = digits to send)

- **party » UserData...** (regular DN, any state) = TUpdateUserData (parameter = TKVList)

TKVList parameter must be specified in its standard text format as a comma-separated list of key-value pairs (the string value must be enclosed in single or double quotes when it starts with digits or includes a comma or parentheses).

- **party » PrivateSvc** (all DNs, any state) = TPrivateService for this party, with parameters: *id* [/*extensions*]

Refer to the Framework 8.1 SIP Server Deployment Guide for a list of service IDs and their parameters.

SIP dialog commands

A set of operations for SIP dialogs depends on the dialog state (according to SIP messages sent or received), as follows:

- **dialog » talk(answer)** (Alerting state)—Answers the call, sending a 200 OK SIP message.
- **dialog » 603 Decline** (Alerting state)—Rejects the call, sending a 603 Decline SIP message.

- **dialog » SIP cancel** (Initiated state)—Abandons the calling attempt, sending a CANCEL message.
- **dialog » SIP hold** (Connected state)—Places the call on hold, sending re-INVITE with the hold SDP (using either RFC 2543 or RFC 3264, depending on the value of **sip-hold-rfc3264**).
- **dialog » SIP retrieve** (Held state)—Retrieves the call from hold, sending re-INVITE with the "talk" SDP.
- **dialog » BYE** (release) (any state)—Releases the call, sending a BYE message.
- **dialog » blindXfer...** (Connected or Held state)—Performs a 1pcc blind transfer to a given destination.
- **dialog » startXfer...** (Connected or Held state)—Initiates a two-step transfer to a given destination. Use the following parameters:
  destination-DN[@*SIP-proxy-host:port*]

  where:
  - *destination-DN*—The destination of the call (user part of the request-URI in the INVITE).
  - *SIP-proxy-host:port*—(Optional) The location of the SIP proxy/gateway (usually it will be SIP Server), if it is not specified, the **sip-proxy** option value is used.
- **dialog » completeXfer** (Connected/Held state)—Completes the two-step transfer. The operation is only available when two SIP dialogs are present on the DN, one in the Connected and the other in the Held state. It can be performed on either SIP dialog with the same result.
- **dialog » customSIP...** (any state)—Executes the Custom SIP script (CSS).
- **dialog » getSIP(call)** (any state), also activated by double-clicking a SIP dialog element—Opens a window with full SIP history of all call-related messages for a particular dialog collected so far. As with device-level SIP history, the content is not updated in real time. To get an update, close the window and invoke the command again.
- **dialog » SendDTMF » NTE/rfc2833...** (SIP, Connected)—Sends DTMF using RFC 2833 telephony events.
- **dialog » SendDTMF » SIP msg INFO...** (SIP, Connected)—Sends DTMF using the INFO message.
- **dialog » SendDTMF » audio tone...** (SIP, Connected)—Sends DTMF as in-band audio tones.