GENESYS

CSTA Connector for BroadSoft BroadWorks Deployment Guide

T-Servers 8.1.1

12/30/2021

Table of Contents

CSTA Connector for BroadSoft BroadWorks Deployment Guide

Genesys CSTA Connector for Broadsoft BroadWorks provides a standard CSTA (computer-supported telecommunications applications) Phase III ASN.1 interface for CTI applications such as Genesys T-Server for CSTA Connector. This documentation set should give you most of the information you need in order to install CSTA Connector for BroadSoft Broadworks. See the summary of the highlighted topics below:

About CSTA Connector

Find out about the CSTA Connector:

Overview

New in This Release

Feature Configuration

Find out about the supported features:

Hot Desking

Redundancy Support

Hiding Data in Logs

all topics>>

Configuration Options

Find out about the configuration options:

Connector Section

link-%s Section

License Section

all topics>

Deploying CSTA Connector

Find procedures to configure and install the CSTA Connector that include these topics:

High Availability Deployment

Overview

Welcome to the *CSTA Connector for BroadSoft BroadWorks Deployment Guide*. These topics introduce you to the concepts, terminology, and procedures that are relevant to CSTA Connector for BroadSoft BroadWorks deployment.

About CSTA Connector

Genesys CSTA Connector for BroadSoft BroadWorks is an application designed to provide a CSTA Phase III interface used by T-Server for CSTA Connector. Connector is installed between the BroadWorks XSI Interface and T-Server to allow the translation of CSTA requests and events into BroadWorks XSI requests and events.

Genesys CSTA Connector for BroadSoft BroadWorks is also responsible for authentication, session management, and data type translation between the BroadWorks XSI interface and the CSTA Phase III Interface.

An example of the solution architecture is shown in the following figure:
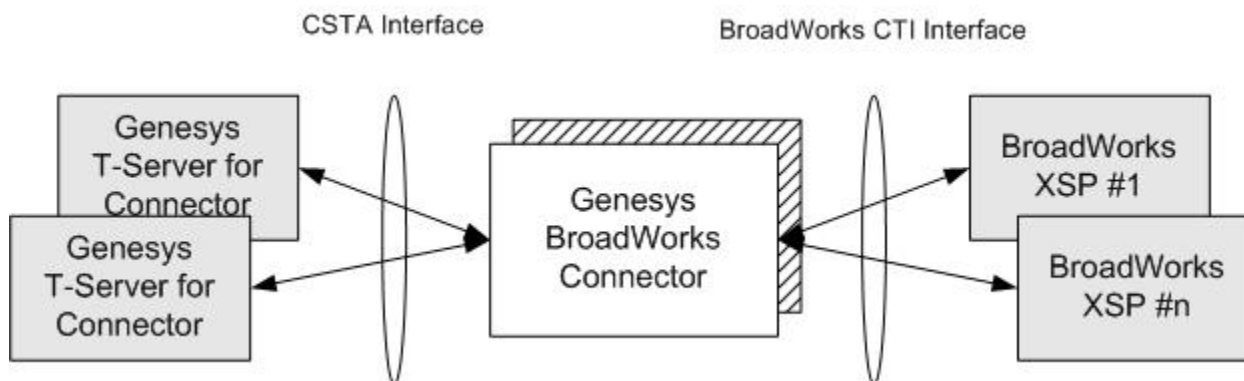
Genesys CSTA Connector for BroadSoft BroadWorks

New in This Release

The following general changes have been implemented in the 8.1 release of CSTA Connector:

- Support for an application-specific LMS file.

- Polling from BroadWorks Connector to XSP.

- Support for the new Broadsoft AS Switchover functionality.

- Support on Red Hat Enterprise Linux 6 32- and 64-bit platforms.

- Limited display of sensitive information: In logs, new options enable sensitive data in logs to be marked for post-processing by the user, such as deletion, replacement, or hiding. See the Genesys 8.1 Security Deployment Guide for details.

- Compliance with FIPS: TLS as implemented by Genesys meets the Federal Information Processing Standards (FIPS).

Feature Configuration

Supported Features

The following is a summary of the functionality supported by CSTA Connector for BroadSoft BroadWorks:

- AS Failover
- Call Distribution Model
- Call Models
- Call Recording
- Configuring the CSTA Connector
- Conference Model
- CTI Connection
- Hiding Data in Logs
- Hot Desking
- Interoperability
- Keep-Alive Feature
- Messaging
- Hot Standby Redundancy Support
- Supported BroadWorks Requests and Events
- Web Application Path

AS Failover

BroadWorks Application Server Failover Support

When the BroadWorks Application Server (AS) is ungracefully shutdown, any existing agent calls are maintained with limited media support and no call control. The redundant AS has no knowledge of the old calls and it does not inform CSTA Connector about those calls. At this point, the call information collected by CSTA Connector before the failover becomes desynchronized with the BroadWorks call information.

This scenario results in *stuck* calls that may prevent new calls from being distributed to the agents, who appear to be busy because of the stuck calls.

Call Synchronization

For call synchronization, BroadWorks introduces an extended usage for the existing `CallSubscriptionEvent` event. The `CallSubscriptionEvent` event always contains the latest known call information for the subscribed user. For AS Failover scenarios, BroadWorks distributes this event as an unsolicited event as soon as BroadWorks detects the user call state desynchronization due to the AS failover. The system behavior in regards of the voice path is described in the Broadsoft BroadWorks documentation.

CSTA Connector, from its side, may induce the `CallSubscriptionEvent` event distribution at any time by issuing an `UpdateSubscriptionRequest` request for a specific user. CSTA Connector invokes the `UpdateSubscriptionRequest` request upon receiving a CSTA Snapshot Device request from the CSTA client. The content of the CSTA Snapshot Device response is created using the information provided in the `CallSubscriptionEvent` event associated with the `UpdateSubscriptionRequest` request.

Previously, the CSTA Snapshot Device has been implemented as BroadWorks' response to the `RetrieveCallIdInfoListRequest` request. The request-response mechanism in BroadWorks may be not synchronized with the real-time CTI Event reporting. BroadSoft recommends using the `CallSubscriptionEvent` event to collect the call data and avoid any potential race condition between the event reporting and receiving responses on the BroadWorks request.

CSTA Connector uses the content of the `CallSubscriptionEvent` event to synchronize Connector's internal call configuration. CSTA Connector notifies its clients (T-Servers) through either a CSTA Event or CSTA Response to inform them about the changes in the call configuration. The LMS message, 37205, is generated for each call party that is not synchronized.

Unsolicited CallSubscriptionEvent

BroadWorks distributes the `CallSubscriptionEvent` event as soon as BroadWorks detects a failover on the subscribed user. CSTA Connector tests the call parties reported in the event against the list of call parties on the given user that are kept in CSTA Connector.

All parties that are not reported in the the `CallSubscriptionEvent` event are cleared from the CSTA Connector memory and the CSTA `Event Connection Cleared` message (cause: `SwitchTerminated`) is sent to all clients (T-Servers).

Note: The `CallSubscriptionEvent` event supplies the party information for a specified agent. Other agents involved in the same call are not affected as they may have received a dedicated `CallSubscriptionEvent` event.

Using UpdateSubscriptionRequest

BroadWorks does not detect call desynchronization and does not distribute an unsolicited `CallSubscriptionEvent` event, if the call originates from an agent or the new call signal alerts the agent. CSTA Connector can force BroadWorks to distribute a `CallSubscriptionEvent` event at any time by sending an `UpdateSubscriptionRequest` request to BroadWorks.

The main purpose of the `UpdateSubscriptionRequest` request is to renew the call event subscription for agents. CSTA Connector issues this request before the subscription expiration to receive further call state updates for agents. CSTA Connector renews the event subscription only once every 30 (+10/-0) minutes. This time is counted starting from the last `UpdateSubscriptionRequest` for a specific agent.

CSTA Connector handles `CallSubscriptionEvent` events associated with the `UpdateSubscriptionRequest` request in the same way as an unsolicited `CallSubscriptionEvent` event. CSTA Connector clients (T-Servers) are informed by using `CSTA Event Connection Cleared` message, if necessary.

A CSTA Connector client may request call synchronization by issuing a `CSTA Snapshot Device` request. CSTA Connector processes it as an `UpdateSubscriptionRequest` request. The call information received in the associated `CallSubscriptionEvent` event is reported back to the client-requestor in the `CSTA Snapshot Device` response. If any unsynchronized calls are discovered then the other CSTA Connector clients are informed using a `CSTA Event Connection Cleared` message.

This functionality is implemented in T-Server for CSTA Connector. T-Server for CSTA Connector uses the `CSTA Snapshot Device` request to validate calls on the agent where the call party remains in the `Alerted` or `Originated` state for longer than average. T-Server for CSTA Connector calculates the average time between the Dialing (or Ringing) and the Established event for specific agent and requests call synchronization, if this time is exceeded.

Note: Call information in `CallSubscriptionEvent` events reflect the current call configuration in the BroadWorks system. For AS Failover scenarios, BroadWorks does not guarantee the full synchronization with an agent end-point. There is a possibility that the BroadWorks and CSTA Connector are synchronized and report that the agent is free, but actually, the agent is still in the same conversation that was established before the AS Failover.

Using Other Subscription Events

BroadWorks virtual subscribers (Routing Points and ACD Queues) have not been included in the BroadWorks failover design, however, behavior for those subscribers after AS failover is similar to the agent subscribers. The main difference is that BroadWorks has no ability to report unsolicited `RoutePointSubscriptionEvent` and `ACDSubcriptionEvent` events.

BroadWorks reports those events when the `UpdateSubscriptionRequest` request is sent for those subscribers. CSTA Connector client may request call synchronization for Routing Points and ACD Queues by issuing a CSTA Snapshot Device request. CSTA Connector processes it as an `UpdateSubscriptionRequest` request. The call information received in the associated `RoutePointSubscriptionEvent` event (or `ACDSubcriptionEvent` event) are reported back to the client-requestor in the CSTA Snapshot Device response. If any unsynchronized calls are discovered, then the other CSTA Connector clients are informed using CSTA Event Connection Cleared.

CSTA Connector clients may request call synchronization for Routing Points and ACD Queues by issuing an CSTA Snapshot Device request. T-Server for CSTA Connector uses this request to validate the calls on the Routing Point and ACD Queues where the call party remains queued longer than the average queuing time. T-Server for CSTA Connector calcullates the average time between the `EventQueued` and `EventDiverted` events for each monitored ACD Queue and Routing Point and requests call synchronization, if this time is exceeded.

Note: The periodic check call feature configured in T-Server may delete established calls on the agent end-points after AS Failover.

AS Log Message

Text	37205|STANDARD|GCTI_EVC_005|Call %s on device %s not in snapshot
Attributes	call-id—CSTA Connector unique call identifier device-id—CSTA Connector DN
Description	A received event indicating that the call no longer exists on the device. CSTA Connector took internal action by removing that call from the CSTA Connector memory and sent a message to the client that the call on the device no longer exists.
Actions	None

Call Distribution Model

The types of call distribution are as follows:

- Automatic (ACD Queue): ACD and the Agent Model
- Host Controlled (Routing Point): CSTA Routing Profile

ACD and the Agent Model

Agent assignment to a BroadWorks Contact Center device is provisioned in BroadWorks configuration and can not be controlled through computer telephony integration (CTI); That means that which ACD Queue the agent takes calls from cannot be controlled through CTI. The call is then automatically distributed to agents in the `Available` state. The agent state can be changed using CTI requests. No state transition restrictions are applied by BroadWorks Connector, this is, a transition from any agent state is permissible to any state at any time.

Note: BroadWorks CTI has a limitation regarding ACD Queue assignments and does not provide a request to enable agents to modify their queue assignment. Agent assignment must be accomplished through BroadWorks configuration.

Agent Model

BroadWorks implements a simple agent model where all agent state changes are permitted. The following table provides a list of Broadsoft agent states along with BroadWorks Connector's mapping into the CSTA agent state model.

Broadsoft Agent States Mapping

BroadWorks Agent State	CSTA Agent State	Comments
Agent Sign-in	agentNotReady	This state is a temporary state while logging in.
Agent Available	agentReady	
Agent Unavailable	agentNotReady	
Agent Wrapping-up	agentWorkingAfterCall	
Agent Signed-out	agentNull	

Agent State Synchronization

BroadWorks CTI issues an `ACDAgentJoinUpdate` event whenever an agent joins or leaves a queue as well as initial synchronization events when the BroadWorks Connector subscribes for these events. The BroadWorks Connector uses this information in these events to allow CSTA client's agent objects to remain synchronized with the switch status.

BroadWorks Connector sends the following CSTA Agent State events to its client whenever it receives an `ACDAgentJoinUpdate` event from the BroadWorks XSP server.

CSTA AgentState Evenst Sent to Clients

CSTA	BroadWorks CTI
`AgentState event:`	`ACDAgentJoinUpdate event:`
• `device`: acd subscriber id	• `subscriberId`: agent subscriber id

• agent ID: subscriber id	• `ACDAgentJoinInfo:`
	    • `ACDUserId`: acd subscriber id
	    • `agentACDState`: `agent_state`

• logged on state: `true`, if signed in	
• acd group: acd subscriber id	
• agent state: `agent_state`	

CSTA Routing Profile

BroadWorks Connector supports the CSTA Routing Profile. A BroadWorks Routing Point subscriber is used as a routing device.

The CSTA Routing profile consists of the following topics:

- Selecting Routing Type
- Treatments
- CSTA PlayEvent Message
- CSTA StopEvent Message
- BroadWorks Event Cause Translation
- Configuring PlayAnnouncement Treatments

Note: Only routing devices support TMakePredictiveCall requests.

Selecting a Routing Type

CSTA Connector provides the following two types of routing functionality:

- Non-supervised routing in CSTA Connector always invokes the BroadWorks `RoutePointBlindTransferRequest` request for routing from BroadWorks Routing Points.

- Supervised routing uses the `RoutePointDistributeCallRequest` request for BroadWorks agents as routing targets.

The `RoutePointDistribute` request offers additional switch functionality for routed calls and customer requests to enable the use of `RoutePointDistributeCallRequest` requests for non-supervised routing.

Configuration Options

enable-rp-distribute

Default Value: `true`
Valid Value: `true, false`
Changes Take Effect: Immediately

Selects `Distribute Call` as the default method for non-supervised routing, if this option is set to `true`. Otherwise, `Blind Transfer` is used.

Treatments

The CSTA `Play Message` request is mapped into a BroadWorks Treatment request as described in the following table:

Mapping of the BroadWorks Treatment Request to the CSTA PlayMessage Request

BroadWorks Request	Parameters	CSTA PlayMessage Parameter
RoutePointPlayBusyRequest		Message ID = 0 (zero)
	duration	Duration
	busyPattern	Extension busyPattern
RoutePointPlayMOHRequest		Message ID = 1
	duration	Duration
	digitMap	Extension collectDigits
RoutePointPlayRingbackRequest		Message ID = 2
	duration	Duration
	digitMap	Extension collectDigits
RoutePointPlaySilenceRequest		Message ID = 3
	duration	Duration
	digitMap	Extension collectDigits
RoutePointPlayTreatmentRequest		Message ID = 4
	duration	Duration

BroadWorks Request	Parameters	CSTA PlayMessage Parameter
	digitMap	Extension collectDigits
	audioUrlList	Extension audioUrl
	videoUrlList	Extension videoUrl

CSTA PlayEvent Message

The CSTA PlayEvent message is generated upon receiving the appropriate *Treatment started* BroadWorks message as described in the following table:

BroadWorks Events and the Associated CSTA PlayEvent Parameters

BroadWorks Events	Parameters	CSTA PlayEvent Parameters
RoutePointBusyStartedEvent		Message ID = 0 (zero)
RoutePointMOHStartedEvent		Message ID = 1
RoutePointRingbackStartedEvent		Message ID = 2
RoutePointSilenceStartedEvent		Message ID = 3
RoutePointTreatmentStartedEvent		Message ID = 4

CSTA StopEvent Message

CSTA StopEvent message is generated upon receiving the appropriate *Treatment completed* BroadWorks message as described in the following table:

BroadWorks Events and the Associated CSTA StopEvent Parameters

BroadWorks Events	Parameters	CSTA StopEvent Parameters
RoutePointBusyCompletedEvent		Message ID = 0 (zero)
	completionReason	Cause
	errorReason	N/A (presented as string in BroadWorks)
	digits	Extension collectDigits
RoutePointMOHCompletedEvent		Message ID = 1
	completionReason	Cause
	errorReason	N/A (presented as string in BroadWorks)
	digits	Extension collectDigits
RoutePointRingbackCompletedEvent		Message ID = 2
	completionReason	Cause
	errorReason	N/A (presented as string in BroadWorks)
	digits	Extension collectDigits
RoutePointSilenceCompletedEvent		Message ID = 3
	completionReason	Cause

BroadWorks Events	Parameters	CSTA StopEvent Parameters
	errorReason	N/A (presented as string in BroadWorks)
	digits	Extension collectDigits
RoutePointTreatmentCompletedEvent		Message ID = 4
	completionReason	Cause
	errorReason	N/A (presented as string in BroadWorks)
	digits	Extension collectDigits

BroadWorks Event Cause Translation

The following table describes the BroadWorks event cause translation for the StopEvent message:

Event Cause Translation for the StopEvent Message

BroadWorks Event Cause	CSTA Event Cause	Comments
Match	CauseCharacterCountReached	
NoMatch	CauseInvalidNumberFormat	
Timeout	CauseTimeout	
EOF	CauseTerminationCharReceived	
Stopped-OutOfQueue	CauseSwitchTerminated	
Stopped-NewPlayCollect	CauseNextMessage	
Stopped-OfferedWithRingback	CauseDistributed	
Error	CauseReserved	There is no specific error cause in CSTA.

Configuring PlayAnnouncement Treatments

Configuring PlayAnnouncement Treatments

Currently, Interaction Routing Designer (IRD) can only pass integers when using the standard `Treatment` block. However, the BroadWorks switch must receive a string value in order to activate a treatment: to specify either a URL or the location of a file stored on the web server.

The BroadWorks Connector looks for the USER_ID value in the treatment request in order to provide the treatment details to the switch. Starting with release 8.1.2, Interaction Routing Designer supports this functionality within the standard `Treatment` block. However, for older versions of IRD, you must use one of the following two approaches to achieve this functionality:

1. the SendRequest Function

2. the Treatment Function

The following topics describe these approaches in more detail:

Using the SendRequest Function

Using the SendRequest Function

To use this approach, add a `Function` block to the routing strategy, select `SendRequest` and provide the following information:

```
SendRequest[RequestApplyTreatment,Cat['83:7|thisdn:', Dest[], '|9:',
ConnID[],'|84.PROMPT.1.INTERRUPTABLE:0|84.PROMPT.1.USER_ID:http://1.1.1.1:111/
test.wav' ]]
```

The clickable image below displays how to use the `SendRequest` function:
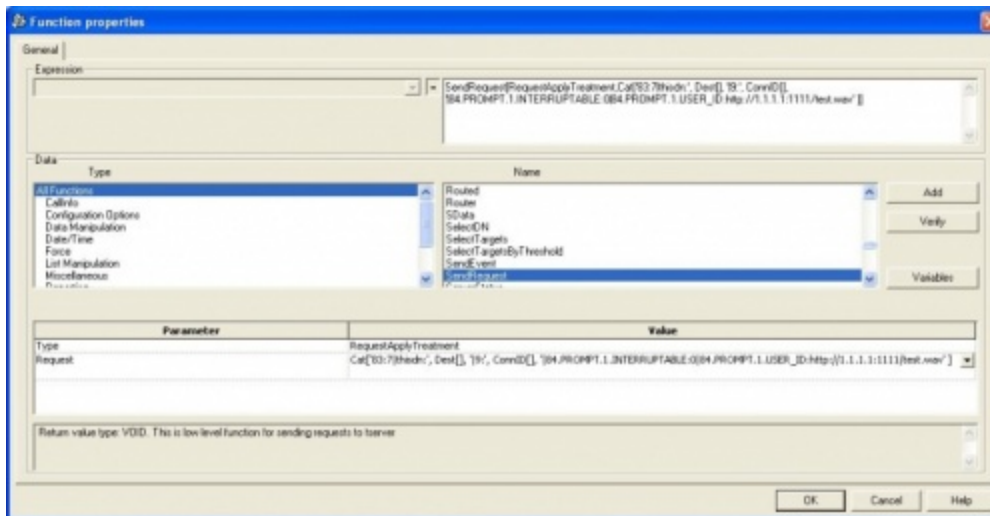
Using the Function Block

Using the Function Block

You can also enter the information typically provided by the `Treatment` block directly into a `Function` block. In the `Function` block, provide the following information:

```
TreatmentPlayAnnouncement['LANGUAGE','','MSGID','','MSGTXT','','PROMPT.1.INTERRUPTABLE',
'0','PROMPT.1.ID','','PROMPT.1.DIGITS','','PROMPT.1.USER_ANN_ID','','PROMPT.1.TEXT','',
'PROMPT.1.USER_ID','http://1.1.1.1:1111/test.wav']
```

The clickable image below displays how to enter information directly into a `Function` block.:
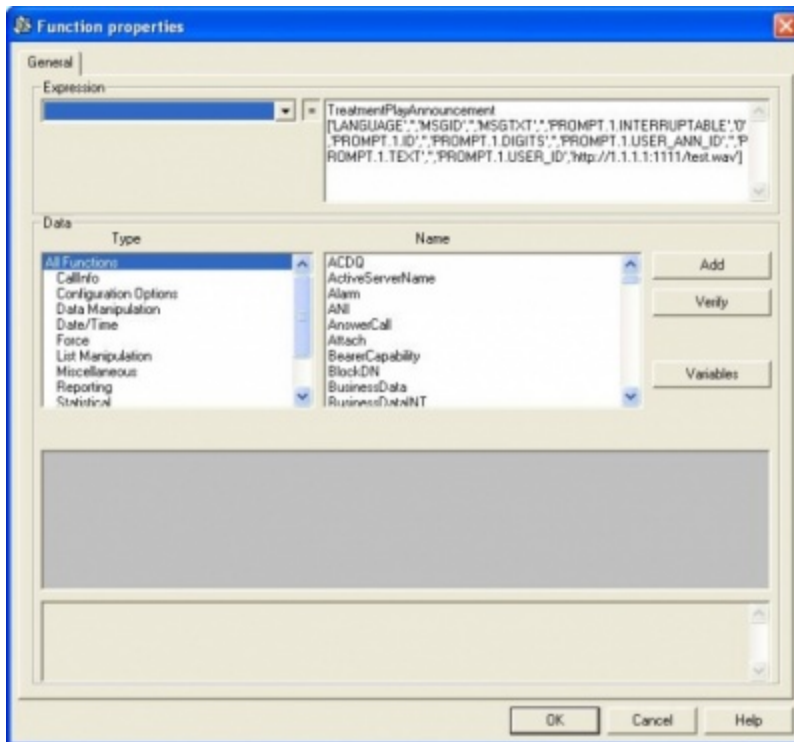
Treatment Request Results

Treatment Request Results

Either of these methods result in the Universal Routing Server (URS) sending the following TApplyTreatment request:

```
message RequestApplyTreatment
    AttributeThisDN '54201'
    AttributeConnID 009a01fd7f522011
    AttributeTreatmentType 7 (TreatmentPlayAnnouncement)
    AttributeTreatmentParms [89] 00 01 03 00..
        'PROMPT'(list) '1'(list) 'INTERRUPTABLE' 0
        'USER_ID' 'http://1.1.1.1:1111/test.wav'
    AttributeReferenceID 12
```

From this request, the BroadWorks Connector takes the information provided by the USER_ID parameter and sends it to the switch in the audioUrlList:

```
12:30:55.704 +++ XMLToSwitch +++
12:30:55.704 <?xml version="1.0" encoding="UTF-8"?>
<xsp:request version="17.0"xmlns:xsp="http://schema.broadsoft.com/XspXMLInterface">
<requestId>2228</requestId>
  <sessionId>10</sessionId>
```

```
  <credentials>UUFBUlRfRW52M19hZG1pbkBhcHBzZXJ2MDE6YWRtaW4=</credentials>
  <xsp:RoutePointPlayTreatmentRequest>
    <uri>/com.broadsoft.cti-actions/v2.0/
routepoint/%routepointId%/calls/%callId%/playtreatment</uri>
    <method>PUT</method>
    <version>17.0</version>
    <params>
      <routepointId>441290554201</routepointId>
      <callId>callhalf-4258123:0</callId>
    </params>
    <xsp:payload>
      <xsi:RoutePointPlayTreatment xmlns:xsi="http://schema.broadsoft.com/xsi">
      <xsi:audioUrlList>
        <xsi:uri>http://1.1.1.1:1111/test.wav</xsi:uri>
      </xsi:audioUrlList>
      <xsi:numberOfPlay>1</xsi:numberOfPlay>
    </xsi:RoutePointPlayTreatment>
    </xsp:payload>
  </xsp:RoutePointPlayTreatmentRequest>
</xsp:request>
```

Call Models

BroadWorks Connector uses the computer-supported telecommunications applications (CSTA) call model for reporting. BroadWorks CTI uses a call model that is similar to the CSTA call model. Genesys BroadWorks Connector translates BroadWorks CTI objects into CSTA objects.

The clickable figure below displays the difference between the BroadWorks Call Model and the CSTA Call Model:
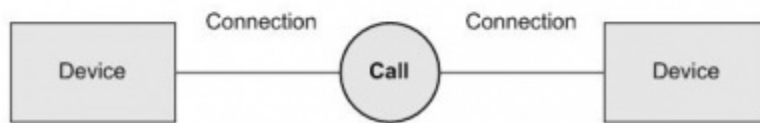
Note: The BroadWorks conference call model does not fit the CSTA call model. See, the BroadWorks Conference Model topic for more details.

Device

The Subscriber ID in BroadWorks uniquely identifies an addressable telephony system user—a subscriber. Subscribers are normally assigned an E.164 number. For compatibility with the CSTA Device Identifier, the E.164 phone number format is used for the Subscriber ID in BroadWorks CTI and CSTA. Refer to the International Telecommunication Union (ITU) E.164 Numbering Plan: http://www.itu.int/rec/T-REC-E.164/en for further details.

BroadWorks Connector translates the public directory number used by BroadWorks to identify subscribers into a private directory number that is used in CSTA switching and computing functions. The CSTA Device Monitoring service is translated to a BroadWorks User Subscription on all available event packages for the specified device type.

BroadWorks Connector supports a subscription for the following BroadWorks resources:

- User—Translates to a virtual or real endpoint.
- Call Center—Translates to a CSTA ACD device.
- Routing Point—Translates to a CSTA Routing device.

Call

A call session in BroadWorks is identified by a BroadWorks External Tracking ID. An External Tracking Id is translated into a CSTA Call ID. All External Tracking ID changes are reflected in the appropriate CSTA event reporting.

Connection

A call in the BroadWorks CTI call model is a *logical connection between a subscriber and a call session* Therefore, a *BroadWorks call* is translated into a CSTA Connection. However, the BroadWorks Call ID is not equivalent to the CSTA Call ID because two or more parties on a call have different BroadWorks Call IDs. The BroadWorks Call ID is actually a party ID. The BroadWorks external tracking ID is equivalent to a CSTA Call ID. Due to this characteristic, the BroadWorks Call ID is mapped to a CSTA dynamic device ID element of the CSTA Connection ID.

Call Recording

Call recording provides the functionality to digitally record calls on a requested connection.

CSTA Connector supports the CSTA `RecordMessage` request that can be invoked on the behalf of the BroadWorks user. For a successful service invocation, `Call Recording On Demand` should be enabled for the BroadWorks user. If a recording request is successful, CSTA Connector distributes the CSTA `Record` event. The CSTA `Stop` event is generated when the device with the active recording leaves the call.

If a conference call recording is requested, CSTA Connector internally initiates a call recording for all BroadWorks calls that are included in the conference call —usually the number of parties in the conference call minus one.

See, BroadWorks Events for more information on the `Record` and `Stop` events.

Known Limitations:

- CSTA Connector for BroadSoft BroadWorks does not support the stop operation for Call Recording.
- CSTA Connector for BroadSoft BroadWorks does not report the unconditional recording for users when `Call Recording Always` is provisioned.
- CSTA Connector for BroadSoft BroadWorks does not support the Query Call Recording functionality on startup.

Related Configuration Options

broadworks-version

Default Value: `17.sp4`
Valid Values: Supported BroadWorks release version.
Changes Take Effect: Immediately
Related Feature: Call Recording

Specifies version "17.sp4" in all CTI requests sent to the BroadWorks XSI server and provides backwards compatibility with previous CSTA Connector releases.

Configuring CSTA Connector

With the 8.1 Management Layer, an application should be configured as type CSTA Connector, and with a pre 8.1 Management Layer, an application should be configured as type High Availability Proxy.

T-Server Specific Connection Configuration

T-Server for Connector initiates a session with a Connector as follows:

- If there is at least one application with the CSTA Connector type in the list of Connections, T-Server reads the Connector's connectivity options (including secure port information) and Connector is handled as a server with normal connectivity options.

- T-Server looks for CSTA Connector or HA Proxy application types (a pre-8.1 Management layer cannot be used with a CSTA Connector type).

- If there is no CSTA Connector application object in the list of Connections, T-Server uses the normal link section configuration.

Device Configuration in the Configuration Layer

The following table displays the device configuration in the Configuration Layer:

Switch Device Type	DN Type	Switch-Specific Type	Association
End Point	Extension	Not applicable	Not applicable
Call Center	ACD Queue	Not applicable	Not applicable
Route Point	• Routing Point • Routing Queue • External Routing Point	• Not applicable • Not applicable • Not applicable	• Not applicable • Not applicable • Not applicable

Switch Terminology

The following table compares the relevant switch terminology with the Genesys terminology:

Genesys Term	Switch Term
ACD Queue	Call Center
Agent ID	User
Extension	End Point
Routing Point	Route Point
Logon	Sign In
Logoff	Sign Out
Ready	Available
NotReady	Unavailable
AfterCallWork	ACD Wrap-up Timer
Account Code	Call Disposition Code
Reason Code	Agent Unavailable Code

Conference Model

In BroadWorks, a conference bridge is logically established by the conference controller device. BroadWorks CTI messaging does not include any messages that inform other conference participants that they are included in the conference. All call sessions remain the same before and after establishing a conference.

In the CSTA model, all conference members (including the conference controller) are connected to the same CSTA call, which is illustrated in the clickable figure below:
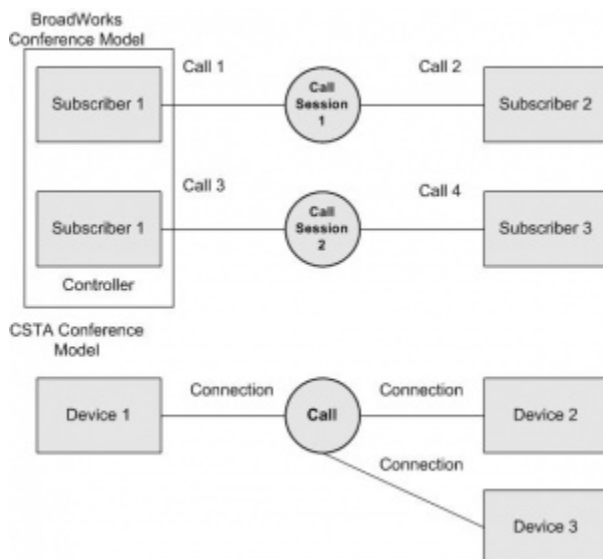
BroadWorks Compared to the CSTA Conference Model

In a conference scenario, BroadWorks Connector uses BroadWorks Events so the conference controller device can link multiple `CallSessions` into a single CSTA Call. In general, the CSTA conference model does not apply additional restrictions on the conference controller device. Due to BroadWorks' specific way of reporting conference calls, the following restrictions are applied at the CSTA-level to the conference controller device (Device 1 in the figure above):

- The conference controller cannot be deleted from the conference call without ending the entire conference call.

- The consultation call from the conference controller is supported, however the `Reconnect Call` and `Transfer Call` CSTA operations are not available for the conference controller.

CTI Connection

Establishing a Connection to the CTI Link

Upon startup, CSTA Connector for BroadSoft BroadWorks opens a Transmission Control Protocol (TCP) connection to the BroadWorks XSP server. All requests (including channel-related requests for event reporting) are sent over the connection. Responses and events are received from the BroadWorks CTI interface through the same connection.

CSTA Connector supports connections to multiple XSP servers. CSTA Connector creates one event channel per transmission control protocol (TCP) connection. See, Redundancy Support for more information.

The figure below displays the initialization of the connection between the CSTA Connector and the BroadWorks XSP server:
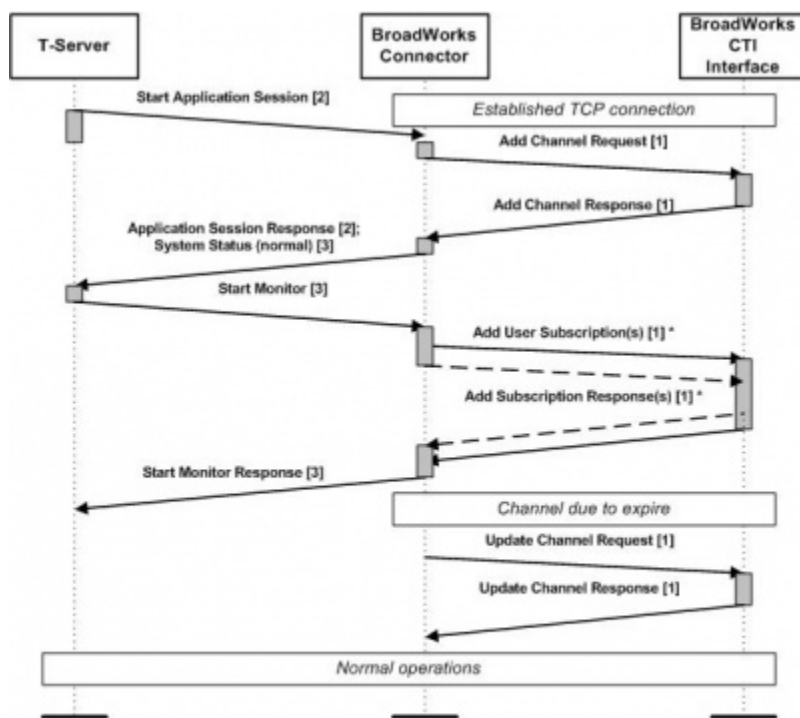
* Start monitor may invoke subscription for multiple event packages

Failure Detection and CTI Link Recovery

CSTA Connector uses the keep-alive functionality on the TCP connection level. In addition, the channel integrity is checked using the `Get Application Controller State` request.

If CSTA Connector detects a channel failure, CSTA clients are sent the system status and/or monitor stop messages indicating a failed CTI link(s). CSTA Connector then restores the TCP socket after the failure and re-creates the Event channel. Resource re-subscription must be re-initiated by the CSTA client.

Hiding Data in Logs

The following topics describe the data hiding functionality for the CSTA Connector:

Hiding Data in Logs Overview

CSTA Connector provides data filtering capabilities based on the Perl-Compatible Regular Expressions (PCRE) library. Sensitive data is defined by a set of regular expressions provisioned in the application configuration though Genesys Management Framework. All data to be hidden is overwritten with an asterisk (*) symbol.

CSTA Connector supports two major modes of hiding sensitive data:

1. **Simple:** Data identified by a matching prefix is hidden up to the end of line.

2. **Complex:** If the regular expression contains one or more *capture* subexpressions, the data matching the subexpressions is hidden while the rest of the text is not. This allows partial hiding of data, such as displaying the last four digits of a credit card number, and preservation of syntactic elements such as parentheses and quotes.

The technical definition of the behaviour of complex patterns is as follows:

*   If the regular expression positively matches, and there are no captured sub-strings, the remainder of the text line after the match is hidden.

*   If the regular expression matches and there are one or more captured sub-strings, all captured sub-strings are hidden.

Multiple Regular Expressions

Multiple regular expressions can be provided that are arranged in ASCII order by their corresponding option name and applied sequentially. Each subsequent expression is applied to a string that could have been already modified by the preceding expression. It is thus essential that the preceding regular expressions do not hide the keyword part of the succeeding expressions. It is recommended that the expressions do not overlap in their matches and that multiple sensitive data chunks are handled in a single expression.

Examples:

*   The non-capture sub-expressions denoted by (?:) are not hidden and can be used for grouping the keyword expressions.
    For example, the expression: User (?:PIN|account) has no capture strings and thus the remainder of the string after the match is hidden.
    If the expression User (PIN|account) is used, the words PIN or account are hidden, while the following data is not, because (PIN|account) forms a capture sub-expression.

*   The non-capture and capture sub-expressions can be combined in one expression with the expected results. For example, the expression:

`[Aa](?:uthentication|ccess)(?: code)?: *"([^"]).{3}"` has two non-capture and one capture strings; the latter, which is the leading characters of the code enclosed between double quotes, is hidden; the last three characters of the code and the closing double quotes are displayed. (The non-capture part matches the words `authentication` or `authorization` with an optionally capitalized initial A and optionally followed by the word code).

Note: The use of optional capture strings can lead to obscure results:

- For example, the expression: `User( name)?`, which matches `User` or `User name`, results in hiding the trailing string in the former case, but results in hiding the word name in the latter case, because `( name)` is a capture sub-expression. In this instance, a non-capture string should be used, such as `User(?: name)?`.

A valid example of an optional capture string is a code that can come in one or two parts:

- For example, the expression: `Code ([A-Z]{3-5} )? ([0-9]{10})`, which matches the word Code followed by a code that is composed of an optional alphanumeric prefix of three to five symbols, and a whitespace, followed by a mandatory 10-digit code. Both the prefix and the digital code are masked.

Limits and Constraints

This feature allows you to use arbitrarily complex expressions that affect the central processing unit (CPU) usage according to the number and the complexity of the expressions.

Note: Test the CPU usage impact of your regular expressions before deploying the solution in the production environment.

HA Considerations

Redundant instances of CSTA Connectors have independent sensitive data settings that are not automatically synchronized. Use the Application Template to set a generic option synchronization mechanism. See, *The Framework Deployment Guide* for more information.

Related Configuration Options

The log-filter-x section contains the sensitive data definitions. The definitions comprise of a set of Perl-Compatible Regular Expression (PCRE) libraries provisioned as arbitrary options in that section. The configuration option names are lexicographically compared to determine the order of operation. Changes to the contents of the section, or any option, take effect immediately.

Note: Previously written logs are not processed retroactively, so the data that was already written remains in the log.

Hot Desking

BroadWorks Connector makes this feature available to T-Server by using the Broadsoft hoteling feature.

Hoteling allows subscribers to associate their service profiles with a different device. Hoteling functionality is delivered through two separate subscribers, *Hoteling Host* and *Hoteling Guest*. To provide Hoteling Host functionality, subscriber accounts need to have the Hoteling Host service assigned to them in the BroadWorks configuration. Subscribers that have Hoteling Guest assigned can be associated with a Hoteling Host subscriber and use the Hoteling Host subscriber's device with their service profiles.

Once a guest subscriber has established a hoteling association with a host, it is assigned the host's endpoint (physical device), leaving the host without one. The guest subscriber's id is used in BroadWorks call reporting—for example: when a call is sent directly to the guest or when an ACD Queue diverts a call to them.

As part of establishing a hoteling session, the CSTA Connector sets unconditional forwarding from the Host subscriber to the Guest subscriber. When the hoteling session is cancelled, the CSTA Connector cancels the forwarding.

T-Server CTI clients initiate a hoteling guest/host association for Hot Desking by sending a `PrivateService` request.

Call Centre Operation

Hoteling is used to support call centre operations by providing a pool of physical devices configured as hoteling hosts that can be used by a potentially larger number of agents configured as hoteling guests. These entities are configured in the Genesys Configuration Layer as Extensions and Agent Logins, respectively. It is important that the Hoteling Guests are only configured as Agent Logins and that no Extensions is configured for the Hoteling Guest.

T-Server CTI clients initiate a hoteling guest/host association by sending an `AgentLogin` request. The hoteling association is reported as the agent being logged in on the host DN. T-Server uses agent substitution to report all subsequent events using the host as the device ID. T-Server CTI clients terminate a hoteling guest/host association by sending an `AgentLogout` request.

Note: The TAgentLogin request does not assign the agent to an ACD Queue as BroadWorks CTI does not support this feature. T-Server CTI clients can still manipulate the agent's ACD Queue state using the TAgentReady and TAgentNotReady requests.

Event Subscription

BroadWorks Connector identifies hoteling guest subscribers by using the `MonitorDeviceType` private extension in the CSTA `StartMonitor` request; they have a `MonitorDeviceType` type of `acdAgent`.

Initially, BroadWorks Connector only subscribes for `HotelingGuest` events on behalf of guest subscribers as follows:

- When BroadWorks Connector receives a `HotelingGuest` event indicating that the guest subscriber is associated with a hoteling host, it subscribes for all event types.

- When BroadWorks Connector receives a `HotelingGuest` event indicating that the guest subscriber is no longer associated with a hoteling host, the Connector deletes all event subscriptions except for `HotelingGuest`.

Initiating a Hoteling Guest/Host Association

The BroadWorks Connector performs the following actions when it initiates a hoteling association:

1. Sends a request to the BroadWorks XSP server to set the host subscriber's agent status to `signed out`, if there is an agent signed in;

2. Sends a request to the BroadWorks XSP server to set unconditional call-forward on the host's subscriber id to the guest's subscriber id;

3. Sends a request to the BroadWorks XSP server to begin the hoteling guest/host association;

4. Sends a request to the BroadWorks XSP server to set the guest subscriber's agent state to `unavailable`.

BroadWorks Connector initiates a hoteling association when it receives a CSTA `SetAgentState` request indicating an agent login. The host subscriber id is taken from the request's device id and the guest subscriber id is taken from the request's agent id.

The following diagram and table shows the sequence of requests and events that are exchanged between BroadWorks Connector and the BroadWorks XSP server:
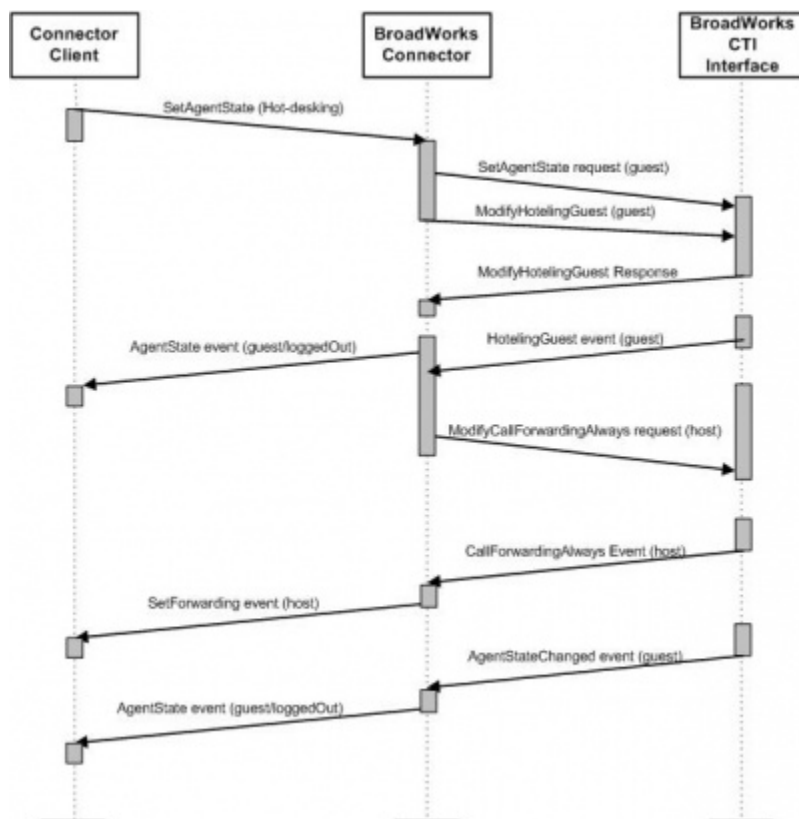
Requests and Events Exchanged Between BroadWorks Connector and the

BroadWorks XSP Server During Initiation

CSTA	BroadWorks CTI	Comments
`SetAgentState` request: • device: host subscriber id • agent state: state • agent ID: guest subscriber id	`SetAgentState` request: • subscriberId: host subscriber id • state: signed out	BroadWorks Connector determines the following: the semantics of the request and if the `SetAgentState` request is an agent login from the current agent state. If the `HotelingGuest` event is matched with a Connector request, then these requests are sent to the XSP in response to receiving the `HotelingGuest` event. **Note:** BroadWorks Connector always initialises the agent state as unavailable to remove any dependency between this request and the initial CSTA `SetAgentState` request.
group: n/a	`ModifyCallForwardingAlways` request: • subscriberId: host subscriber id • address: guest subscriber address • active: true	
	`ModifyHotelingGuest` request: • subscriberId: guest subscriber id • hotelingGuest: see notes • hotelingGuest.isActive: true • hotelingGuest.hostSubscriberId: host subscriber id	The HotelingGuest information is provided in the Hoteling Guest events from the initial events subscription.
`AgentState` event: • device: host subscriber id • agent ID: guest subscriber id • logged on state: true • acd group: n/a • agent state: logged in	`HotelingGuest` event: • subscriberId: guest subscriber id • hotelingGuest: from event • hotelingGuest.isActive: true • hotelingGuest.hostSubscriberId: host subscriber id	BroadWorks Connector may also receive a `HotelGuest` event when an external application modifies the hoteling guest association of a monitored subscriber id.
	`SetAgentState` request: • subscriberId: guest subscriber id	

CSTA	BroadWorks CTI	Comments
	• state: unavailable	
`AgentState event:` • device: subscriber id • agent ID: subscriber id • logged on state: agent_state dependent • acd group: n/a • agent state: agent_state	`AgentStateChanged event:` • subscriberId: subscriber id • agentACDState: agent_state	• BroadWorks Connector's handling of `AgentStateChanged` events is independent of whether the request was sent as part of the initial hoteling sequence, or in response to any subsequent CSTA `SetAgentState` request. • BroadWorks Connector sends the `AgentState` event to the client in response to receiving the `AgentStateChanged` event.
`SetForwarding event` • device: host subscriber id • type: immediate • activate: true • forwardDN: guest subscriber id	`CallForwardingAlways event` • subscriberId: host subscriber id • address: guest subscriber address • active: true	• BroadWorks Connector's handling of `CallForwardingAlways` events is independent of whether the request was sent as part of the initial hoteling sequence, or in response to any other CSTA `SetForwarding` request. • BroadWorks Connector sends the `SetForwarding` event to the Connector client in response to receiving the `CallForwardingAlways` event.

Terminating a Hoteling Guest/Host Association

The BroadWorks Connector performs the following actions when it terminates a hoteling association:

1. Sends a request to the BroadWorks XSP server to set the guest subscriber's agent status to signed out;

2. Sends a request to the BroadWorks XSP server to cancel unconditional call-forward on the host's subscriber id;

3. Sends a request to the BroadWorks XSP server to end the hoteling guest/host association.

The following diagram and table shows the sequence of requests and events that are passed between BroadWorks Connector and the BroadWorks XSP server:
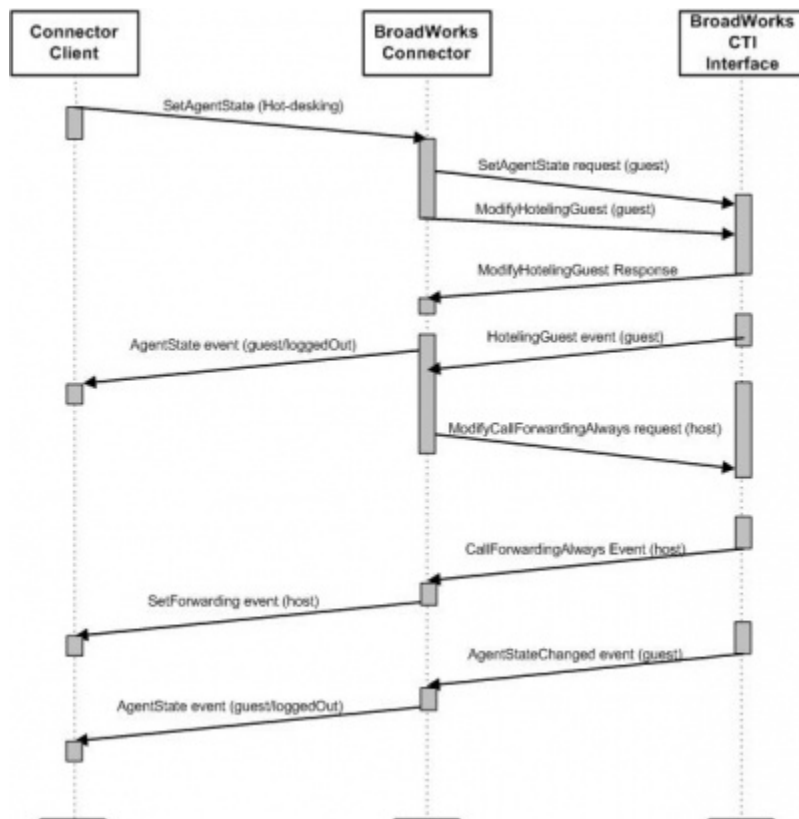
Requests and Events Passed Between BroadWorks Connector and BroadWorks XSP Server During Termination

CSTA	BroadWorks CTI	Comments

`SetAgentState request:`
 	
• device: host subscriber id
• agent state: logged out
• agent ID: guest subscriber id
• group: n/a
	`SetAgentState request:`
	
	• subscriberId: guest subscriber id
	• state: signed out
	
	`ModifyCallForwardingAlways request:`
	
	• subscriberId: host subscriber id
	• address: guest subscriber address
	• active: false
	
	`ModifyHotelingGuest request`
	
	• subscriberId: guest subscriber id
	• hotelingGuest: see notes
	• hotelingGuest.isActive: false
	• hotelingGuest.hostSubscriberId:
		• The `ModifyHotelingGuest` request is sent by BroadWorks Connector to the XSP server in response to receiving a `SetAgentState` request from Connector's client application.
		• The `HotelingGuest` information is provided in the Hoteling Guest event(s).

`AgentState event:` • device: guest subscriber id • agent ID: guest subscriber id • logged on state: false • acd group: n/a • agent state: logged out	`HotelingGuest event` • subscriberId: guest subscriber id • hotelingGuest: from event • hotelingGuest.isActive: false • hotelingGuest.hostSubscriberId:	• BroadWorks Connector may also receive a `HotelGuest` event when an external application modifies the hoteling guest association of a monitored subscriber id. • BroadWorks Connector sends the `AgentState` event to the Connector client to indicate that the hoteling guest/host association has ended. **Note:** The CSTA `AgentState` event uses the guest subscriber id as it has no information about the former host subscriber.
`AgentState event:` • device: subscriber id • agent ID: subscriber id	`AgentStateChanged event:` • subscriberId: guest subscriber id • agentACDState: agent_state	• BroadWorks Connector's handling of `AgentStateChanged` events is independent of whether the request was sent as part of

CSTA	BroadWorks CTI	Comments
• logged on state: true • acd group: n/a • agent state: logged in		the terminating hoteling sequence, or in response to any subsequent CSTA `SetAgentState` requests. • BroadWorks Connector sends the `AgentState` event to the T-Server in response to receiving the `AgentStateChanged` event.
`SetForwarding` event • device: host subscriber id • type: immediate • activate: false • forwardDN: guest subscriber id	`CallForwardingAlways` event • subscriberId: host subscriber id • address: guest subscriber address • active: true	• BroadWorks Connector's handling of `CallForwardingAlways` events is independent of whether the request was sent as part of the initial hoteling sequence, or in response to any other CSTA `SetForwarding` request. • BroadWorks Connector sends the `SetForwarding` event to the Connector client in response to receiving the `CallForwardingAlways` event.

Interoperability

Interoperability testing confirms that T-Server for CSTA Connector connects and communicates with the BroadWorks Connector.

The following section is a series of tables that presents the T-Library functionality supported by T-Server for CSTA Connector and provides the information on the compatibility of the T-Server for CSTA Connector with the BroadWorks Connector.

Note: Refer to the T-Library Functionality topic for the switch functionality supported by the T-Server for CSTA Connector.

The table entries use the following notations:

- **N**—Not supported
- **Y**—Supported
- **I**—Supported, but reserved for Genesys Engineering
- **E**—Event only supported

Supported Services

Call Control Services

Call Control Services

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
TAlternateCall	Alternate Call	TalkRequest	Y	
TAnswerCall	Answer Call	TalkRequest	Y	
TAttachUserData				Internal service, obsolete
TClearCall	Clear Call	ConferenceReleaseRequest	Y	Applicable for conferences only.
TCompleteConference	Conference Call	ConferenceStartRequest	Y	
		ConferenceAddParticipantRequest		Only applicable to the conference controller. (See, BroadSoft BroadWorks Conference Model)
TCompleteTransfer	Transfer Call	ConsultativeTransferRequest	Y	

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
TDeleteAllUserData			Y	Internal service
TDeleteFromConference	Connection Cleared	ReleaseRequest	Y	The conference controller device can not be deleted from the conference. (See, BroadSoft BroadWorks Conference Model)
TDeleteUserData			Y	Internal service
THoldCall	Hold Call	HoldRequest	Y	
		ConferenceHoldRequest		Only applicable to the conference controller. (See, BroadSoft BroadWorks Conference Model)
TInitiateConference	Consultation Call	DialRequest	Y	
TInitiateTransfer	Consultation Call	DialRequest	Y	
TListenDisconnect	Speaker Mute	ConferenceDeafRequest	Y	
TListenReconnect	Speaker Mute	ConferenceUnDeafRequest	Y	
TMakeCall	Make Call	DialRequest	Y	
TMakePredictiveCall	Make Predictive Call	RoutePointOutgoingDialRequest		Available from BroadWorks Route Point only
TMergeCalls,			Y	
—MergeForTransfer		Join Call	N	
—MergeForConference		Transfer Call	N	
TMuteTransfer	Single Step Transfer	MuteTransferRequest	Y	Private extension, MuteTransfer, in CSTA requests.
TQueryAddress,				Internal service
AddressInfo —AddressStatus			Y	
AddressInfo —MessageWaiting Status			Y	
			Y	
—AddressInfoQueue Status —AddressInfo AssociationStatus			N	

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
AddressInfo —CallForwardingStatus			Y	
AddressInfo —AgentStatus			Y	
AddressInfo —NumberOfAgentsIn Queue			Y	
AddressInfo —NumberOfAvailable Agents			Y	
InQueue,			Y	
AddressInfo —NumberOfCallsIn Queue			Y	
AddressInfo —AddressType			Y	
AddressInfo —CallsQuery			Y	
AddressInfo —SendAllCallsStatus			N	
AddressInfo —QueueLoginAudit			Y	
AddressInfo —NumberOfIdleTrunks			N	
AddressInfo —NumberOfTrunksInUse			N	
AddressInfo —DatabaseValue			N	
AddressInfo —DNStatus			Y	
TQueryCall, —CallInfoPartiesQuery	Snapshot Call		Y	
—CallInfoStatusQuery			Y	Internal Service

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
TReconnectCall	Reconnect Call	ReconnectRequest	Y	
TRedirectCall	Deflect Call	BlindTransferRequest	Y	
TReleaseCall	Connection Cleared	ReleaseRequest	Y	
TRetrieveCall	RetrieveCall	TalkRequest	Y	
		ConferenceRetrieveRequest		Only applicable to the conference controller. (See, BroadSoft BroadWorks Conference Model)
TSendDTMF	Generate Digits	TransmitDTMFRequest	Y	
TSetCallAttributes			Y	Internal service
TSingleStepConference	Single Step Conference	MonitorCallRequest	Y	
TSingleStepTransfer	Single Step Transfer	BlindTransferRequest	Y	
TUpdateUserData			Y	Internal service
TMonitorNextCall	MonitorStart (call)	MonitorNextCallRequest	Y	
TCancelMonitoring	MonitorStop (call)	ReleaseRequest	Y	

Capability Exchange Services

Capability Exchange Services

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
TQueryServer			Y	Internal service
TSwitchQuery			N	

Private and Special Services

Private and Special Services

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
Set Account Code	Escape Service		Y	Where supported by CSTA Connector
Associate Guest with host	Escape Service		Y	Where supported by CSTA Connector

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
Disassociate Guest with host	Escape Service		Y	Where supported by CSTA Connector

Registration Services

Registration Services

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
TRegisterAddress	Monitor Start	AddUserSubscriptionRequest	Y	Multiple subscriptions may be created for single subscriber
TRegisterAll			Y	Internal service, restricted
TSetInputMask			Y	Internal service, restricted
TUnregisterAddress	Monitor Stop	DeleteUserSubscriptionRequest	Y	Multiple subscriptions may be deleted for single subscriber
TUnregisterAll			Y	Internal service, restricted

Routing and Treatment Services

Routing and Treatment Services

T-Library	CSTA	BroadWorks CTI	T-Server	Con
TRouteCall,				
—RouteTypeUnknown	Route Select service	RoutePointDistributeCallRequest, RoutePointBlindTransferRequest	Y	
—RouteTypeDefault	Route End service	RoutePointFailCall	Y	
—RouteTypeLabel	Route Select service	RoutePointDistributeCallRequest, RoutePointBlindTransferRequest	Y	Processed as unknown
—RouteTypeOverwriteDNIS	Route Select service	RoutePointDistributeCallRequest, RoutePointBlindTransferRequest	Y	
—RouteTypeDDD	Route Select service	RoutePointDistributeCallRequest,	Y	Processed as

T-Library	CSTA	BroadWorks CTI	T-Server	Con
		RoutePointBlindTransferRequest		unknown
—RouteTypeIDDD	Route Select service	RoutePointDistributeCallRequest, RoutePointBlindTransferRequest	Y	Processed as unknown
—RouteTypeDirect	Route Select service	RoutePointDistributeCallRequest, RoutePointBlindTransferRequest	Y	Processed as unknown
—RouteTypeReject	Route End	RoutePointReleaseCallRequest	Y	
—RouteTypeAnnouncement	Route Select service	RoutePointDistributeCallRequest, RoutePointBlindTransferRequest	Y	Processed as unknown
—RouteTypePostFeature	Route Select service	RoutePointDistributeCallRequest, RoutePointBlindTransferRequest	Y	Processed as unknown
—RouteTypeDirectAgent	Route Select service	RoutePointDistributeCallRequest, RoutePointBlindTransferRequest	Y	Processed as unknown
—RouteTypePriority	Route Select service	RoutePointDistributeCallRequest, RoutePointBlindTransferRequest	Y	Processed as unknown
—RouteTypeDirectPriority	Route Select service	RoutePointDistributeCallRequest, RoutePointBlindTransferRequest	Y	Processed as unknown
—RouteTypeAgentID	Route Select service	RoutePointDistributeCallRequest, RoutePointBlindTransferRequest	Y	Processed as unknown
—RouteTypeCallDisconnect	Route Reject	RoutePointReleaseCallRequest	Y	
TGiveMusicTreatment	Play Message	PlayTreatmentRequest	Y	
TGiveSilenceTreatment	GenerateTelephonyTones	PlaySilenceRequest	Y	
TGiveRingBackTreatment	GenerateTelephonyTones	PlayRingbackRequest	Y	
TApplyTreatment,				
—TreatmentUnknown			N	
—TreatmentIVR			N	
—TreatmentMusic	Play Message	PlayMusicOnHoldRequest	Y	
—TreatmentRingBack	GenerateTelephonyTones	PlayRingbackRequest	Y	
—TreatmentSilence	GenerateTelephonyTones	PlaySilenceRequest	Y	
—TreatmentBusy	GenerateTelephonyTones	PlayBusyRequest	Y	
—TreatmentCollectDigits	Start Data Collection	PlaySilenceRequest	Y	Collect digit added to rec
—Treatment	GenerateTelephonyTones	PlayTreatmentRequest	Y	
PlayAnnouncement,				
—Treatment	Play Message	PlayTreatmentRequest	Y	

T-Library	CSTA	BroadWorks CTI	T-Server	Com
PlayAnnouncementAndDigits,				
—TreatmentVerifyDigits	Start Data Collection		Y	
—Treatment	Start Data Collection		Y	
RecordUserAnnouncement,			N	
—Treatment			N	
DeleteUserAnnouncement,			N	
—TreatmentCancelCall	Route End Service	ReleaseCallRequest	Y	
—TreatmentPlayApplication			N	
—TreatmentSetDefault Route			N	
—TreatmentTextToSpeech			N	
TextToSpeechAndDigits,			N	
—TreatmentFastBusy			N	
—TreatmentRAN			N	

Set Feature Services

Set Feature Services

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
TSetMessageWaitingOn	Set Message Warning		N	
TSetMessageWaitingOff	Set Message Warning		N	
TSetDNDOn	Set Do Not Disturb	ModifyDoNotDisturb	Y	
TSetDNDOff	Set Do Not Disturb	ModifyDoNotDisturb	Y	
TCallSetForwardOn,				
—ForwardModeNone	Set Forwarding	ModifyCallForwardingAlways	Y	Processed as unconditional
—ForwardMode	Set Forwarding	ModifyCallForwardingAlways	Y	
Unconditional,				
—ForwardMode	Set Forwarding	ModifyCallForwardingBusy	Y	
OnBusy,				
—ForwardMode	Set Forwarding	ModifyCallForwardingNoAnswer	Y	
OnNoAnswer,				
—ForwardMode			N	

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
OnBusyAndNoAnswer,				
—ForwardMode			N	
SendAllCalls				
TCallCancelForward,				
—ForwardModeNone	Set Forwarding	ModifyCallForwardingAlways	Y	Processed as unconditional
—ForwardMode Unconditional,	Set Forwarding	ModifyCallForwardingAlways	Y	
—ForwardModeOnBusy	Set Forwarding	ModifyCallForwardingBusy	Y	
—ForwardMode OnNoAnswer,	Set Forwarding	ModifyCallForwardingNoAnswer	Y	
—ForwardMode OnBusyAndNoAnswer,			N	
—ForwardMode			N	
SendAllCalls				
TAgentLogin,				
—AgentWorkMode Unknown	Set Agent State	SetAgentStateRequest(Sign-In),	Y	Full support for emulated agents only, used for hoteling functionality with guest subscribers
—AgentManualIn	Set Agent State	SetAgentStateRequest(Sign-In),	Y	
—AgentAutoIn	Set Agent State	SetAgentStateRequest(Sign-In),	Y	
—AgentAfterCallWork			N	
—AgentAuxWork			N	
—AgentNoCall Disconnect			N	
TAgentLogout	Set Agent State	SetAgentStateRequest(Sign-Out)	Y	Full support for emulated agents only, used for hoteling functionality with guest subscribers
		ModifyHotelingGuest		
TAgentSetReady	Set Agent State	SetAgentStateRequest(Available)		
TAgentSetNotReady,				
—AgentWorkMode Unknown	Set Agent State	SetAgentStateRequest(Unavailable)		
—AgentManualIn	Set Agent State	SetAgentStateRequest(Unavailable)		

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
—AgentAutoIn	Set Agent State	SetAgentStateRequest(Unavailable)		
—AgentAfterCallWork	Set Agent State	SetAgentStateRequest(Wrap-Up)	Y	
—AgentAuxWork	Set Agent State	SetAgentStateRequest(Unavailable)		
—AgentNoCall Disconnect			N	
TAgentSetIdleReason			N	
TSetMuteOff			N	
TSetMuteOn			N	
TListenDisconnect			N	
TListenReconnect			N	

Voice Unit Services

Voice Unit Services

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
TPlayVoiceFile			N	
TLoginMailBox			N	
TLogoutMailBox			N	
TOpenVoiceFile			N	
TCloseVoiceFile			N	
TCollectDigits			N	

Supported Events

Call Control Events

Call Control Events

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
EventAbandoned,	Connection Cleared	CallReleased,	Y	

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
		ACDCallAbandoned,		
		RoutePointCallAbandoned		
EventAgentIdleReasonSet			N	
EventAgentLogin,				
EventAgentLogout	Agent Logged Off	AgentState	Y	
EventAgentNotReady	Agent Logged Off	AgentState	Y	
—AgentWorkMode Unknown	Agent Logged On	AgentState	Y	
—AgentManualIn	Agent Logged On	AgentState	Y	
—AgentAutoIn	Agent Logged On	AgentState	Y	
—AgentAfterCallWork			N	
—AgentAuxWork			N	
—AgentNoCallDisconnect			N	
EventAgentReady		AgentState	Y	
EventDestinationBusy	Failed	CallReleasingEvent	Y	
EventDialing	Originated, Delivered	CallOriginated	Y	
EventDiverted,	Diverted	RoutePointCallForwarded,	Y	
		RoutePointCallFailed,		
		RoutePointCallAnsweredByAgent.,		
		ACDCallForwarded,		
		ACDCallAnsweredByAgent,		
		ACDCallBounced (when "redirect" attribute present)		
EventEstablished	Established	CallAnswered	Y	
EventHeld,	Held	CallHeld,	Y	
		ConferenceHeld		
EventNetworkReached	NetworkReached	CallUpdated	Y	
EventOffHook	ServiceInitiated	CallOriginating	Y	
EventOnHook	ConnectionCleared	CallReleased	Y	
EventPartyAdded	Conferenced	ConferenceStarted	Y	
EventPartyChanged,	Transferred,	CallTransferred	Y	
	Conferenced			
EventPartyDeleted	ConnectionCleared	CallReleased	Y	
EventQueued,	Queued,	ACDCallAdded,	Y	
	Delivered	RoutePointCallAdded		
EventRegistered	Response on Monitor Start Request	Initial Subscription Event	Y	Internal T-Server event for subsequent

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
				registartions
EventReleased	ConnectionCleared	CallReleased		
EventRetrieved,	Retrieved	CallRetrieved,	Y	
		ConferenceRetrieved		
EventRinging	Delivered	CallReceived	Y	

Capability Exchange/Query Event

Capability Exchange/Query Events

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
EventAddressInfo			Y	Internal T-Server event
EventPartyInfo			Y	Internal T-Server Event
EventServerInfo			Y	Internal T-Server Event
EventSwitchInfo			N	

Feature and DN Events

Feature and DN Events

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
EventDNDOff	Do Not Disturb	DoNotDisturb	Y	
EventDNDOn	Do Not Disturb	DoNotDisturb	Y	
EventDNBackInService			TBD	
EventDNOutOfService			TBD	
EventDTMFSent	Digits Generated	Response on TransmitDTMFRequest	Y	Response on Request
EventForwardCancel	Forwarding	CallForwardingAlways,	Y	
		CallForwardingNoAnswer,		

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
		CallForwardingBusy		
EventForwardSet	Forwarding	CallForwardingAlways,	Y	
		CallForwardingNoAnswer,		
		CallForwardingBusy		
EventListenDisconnected		ConferenceCallMadeDeaf	Y	
EventListenReconnected		ConferenceCallMadeUndeaf	Y	
EventMessageWaitingOff			N	
EventMessageWaitingOn			N	

General Events

General Events

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
EventLinkConnected			Y	Internal T-Server event
EventLinkDisconnected			Y	Internal T-Server event
Event Server Connected			Y	Internal T-Server Event
Event Server Disconnected			Y	Internal T-Server Event

ISCC Events

ISCC Events

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
EventRemoteConnectionFailed			Y	Internal T-Server Event
EventRemoteConnectionSuccess			Y	Internal T-Server Event

Miscellaneous Events

Miscellaneous Events

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
EventAttachedData Changed			Y	Internal T-Server Event
EventError			Y	Negative response on request
EventHardwareError			N	
EventResourceAllocated			N	
EventResourceFreed			N	
EventCallInfoChanged			Y	Internal T-Server Event

Private Events

Private Events

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
EventPrivateInfo			Y	

Registration Events

Registration Events

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
EventRegistered	Response on Monitor Start Request	Initial Subscription Event	Y	Internal T-Server event for subsequent registartions
EventRegisteredAll			Y	Internal T-Server Event
Event Unregistered		SubscriptionTerminated	Y	
Event Unregistered All			Y	Internal T-Server Event

Routing and Treatment Events

Routing and Treatment Events

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
EventDigitsCollected,	Stop Event (with collectDigits extension)	RoutePointTreatmentCompleted,	Y	Distributed when collected data is available
		RoutePointMOHCompleted,		
		RoutePointRingbackCompleted,		
		RoutePointBusyCompleted,		
		RoutePointSilenceCompleted		
EventRouteRequest		RoutePointCallAdded	Y	
EventRouteUsed,		RoutePointCallOfferedToAgent,	Y	
		RoutePointCallAnsweredByAgent		
EventTreatmentApplied,		RoutePointTreatmentStarted,	Y	
		RoutePointMOHStarted,		
		RoutePointRingbackStarted,		
		RoutePointBusyStarted,		
		RoutePointSilenceStarted		
EventTreatmentNotApplied		Error Response	Y	
EventTreatmentEnd,		RoutePointTreatmentCompleted,	Y	
		RoutePointMOHCompleted,		
		RoutePointRingbackCompleted,		
		RoutePointBusyCompleted,		
		RoutePointSilenceCompleted		
EventTreatmentRequired			N	

Voice Mail Events

Voice Mail Events

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
EventMailBoxLogin			N	
EventMailBoxLogout			N	
EventVoiceFileClosed			N	

T-Library	CSTA	BroadWorks CTI	T-Server	Comments
EventVoiceFileEndPlay			N	
EventVoiceFileOpened			N	

Keep-Alive Feature

CSTA Connector responds to Keep-Alive requests from the XSP link. CSTA Connector may not always receive timely notification when the XSP link stops functioning. In order for CSTA Connector to detect link failure and initialize alarm and recovery procedures, CSTA Connector usually needs to actively check the link's integrity. This is referred to as Keep-Alive or *KPL* functionality.

CSTA Connector sends BroadWorks KPL request on an established XSP link when the interval from the last message that is received from the XSP link is greater than the value specified by the kpl-interval configuration option. Setting the value of this option to 0 (zero) switches off any *active* KPL functionality.

For each KPL request, CSTA Connector expects to receive a BroadWorks keepaliveResponse message from the XSP link. The XSP link is considered as *unresponsive* when a number of missing KPL responses become greater than the value specified by the kpl-tolerance configuration option. Setting the value of the `kpl-tolerance` option to a value of 0 (zero) marks the link as unresponsive after a single KPL response goes missing.

CSTA Connector then closes the unresponsive link and tries to re-establish it by using the timeout value specified by the `link-recovery-tout` configuration option.

Related Configuration Options

The configuration options can be configured in the main Connector section and in the link-%s section. Options configured in the link-%s section take precedence over options configured in the main Connector section and are applied on that particular link.

The following configuration option enables or disables the Keep-Alive feature:

kpl-interval

kpl-interval

Default Value: 10
Valid Value: Any integer from 0-600
Changes Take Effect: Immediately
Related Feature: Keep-Alive Feature Handling

Specifies a *keep-alive* time interval (in seconds). To check network connectivity, CSTA Connector issues a dedicated KPL request to BroadWorks XSP at the interval specified when there is no other activity on the link. A value of 0 (zero) disables this feature.

This configuration option can be configured in both the Connector and in the link-%s sections. The options configured in the link-%s section take precedence over the options configured in the Connector section and are applied to this particular link. See, the kpl-tolerance option.

kpl-tolerance

kpl-tolerance

Default Value: 0 (zero)
Valid Value: Any integer from 0-10
Changes Take Effect: Immediately
Related Feature: Keep-Alive Feature Handling

Specifies the number of failed keep-alive requests that CSTA Connector permits before considering the XSP link as unresponsive. When an unresponsive link is detected, CSTA Connector closes and reopens this particular XSP link.

This configuration option can be configured in both the Connector and in the link-%s sections. The options configured in the link-%s section take precedence over the options configured in the Connector section and are applied to this particular link. See, the kpl-interval option.

Messaging

Requests

Connector processes incoming CSTA requests according to device types specified in the request. The result of a CSTA request invocation varies according to different types of targets.

Request Groups Requirements

BroadWorks CTI interface offers different subset of requests for different types of resources. See, BroadWorks Requests and *CTI Interface Specification, v 1.6. BroadSoft* for more information on BroadWorks request applicability.

Request Response

BroadWorks Connector uses responses received from BroadWorks CTI to generate appropriate response(s) to send to CSTA clients.

Events

Event Subscription

BroadWorks Connector subscribes to specific types of notifications called Event Packages. Different types of event packages are required for different device types. For example, Routing Points require Route Point Queue event package subscription. Refer to *CTI Interface Specification, v 1.6. BroadSoft* for further details.

Event Processing

BroadWorks CTI events are translated into a common set of CSTA events, which are independant (generally) from the resource type. See, BroadWorks Events for more information.

Hot Standby Redundancy Support

Genesys uses the expression *hot standby* to describe the redundancy type in which a backup server application remains initialized, clients connect to both the primary and backup servers at startup, and the backup server data is synchronized from the primary server. Data synchronization and existing client connections to the backup guarantee higher availability of a component.

The following topics describe the hot standby redundancy support for the CSTA Connector:

Redundancy Support Feature Overview

BroadWorks Connector operates in unison with the generic CSTA Connector T-Server in order to provide a high availability solution. T-Server can be configured to connect to multiple redundant CSTA Connectors. Each CSTA Connector link is active and the T-Server operation continues uninterrupted should it lose a redundant link. T-Server starts attempts to recover a link if it has failed.

In addition, T-Server can be configured to run in the standard Genesys high-availability (HA) mode where a backup T-Server is ready to seamlessly take over operations should the main T-Server fail. The high-availability (HA) deployment configurations is shown in the clickable figure below:
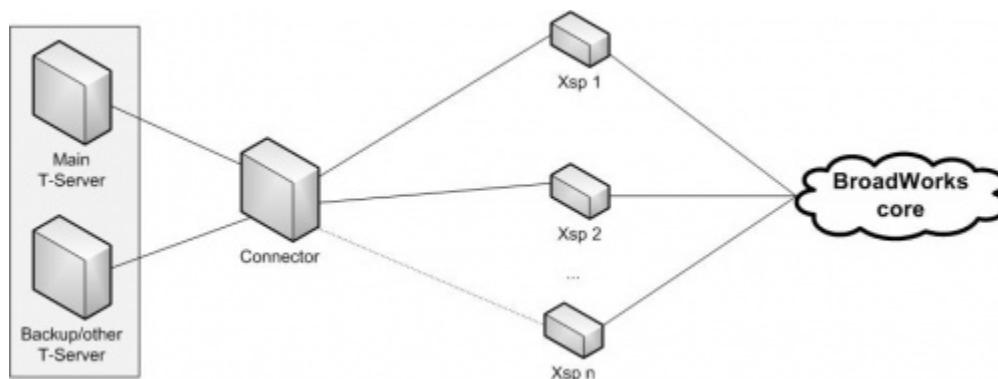
The main features of this solution are:

- CSTA Connector T-Servers that run as a standard Genesys HA pair.
- BroadWorks Connector that operates independently of other Connectors; it does not synchronize the information in the same manner as the T-Server HA pair.
- BroadWorks Connector that maintains the redundant links to the BroadWorks switch independently of the Connector T-Server.
- BroadWorks Connector that supports multiple T-Server connections.

BroadWorks Connector Operation

- BroadWorks Connector supports connections to multiple XSP servers. BroadWorks Connector uses multiple Event Channels, and each Event Channel is given equal weight. BroadWorks CTI balances the event reporting across all the Event Channels. If one of the Event Channels fails, BroadWorks CTI continues to serve the event subscription over the remaining Event Channels, adjusting the load accordingly. BroadWorks Connector initiates its link recovery procedure to re-establish a failed connection.

- BroadWorks Connector supports multiple T-Server connections. T-Servers may or may not start monitors for the same set of devices. BroadWorks Connector filters the events and requests for each connection against their respective start monitor requests.

- BroadWorks Connector responds to System Status requests from T-Server that indicate the overall status of the links, and reports a failure if all links are down.

Setting the DN Properties

The following table indicates how to set the Genesys DN properties for the CSTA Connector for BroadSoft BroadWorks:

Setting the Genesys DN Properties

BroadWorks Subscriber	Genesys DN Type	Switch-Specific Type	Association	Register
User	Extension	1	Not applicable	True
Call Center	ACD Queue	1	Not Applicable	True
Routing Point	Routing Point <ref>In Genesys, a BroadWorks Routing Point can be used as a regular Routing Point or an external Routing Point.</ref>	1	Not applicable	True
Routing Point	External Routing Point <ref>In Genesys, a BroadWorks Routing Point can be used as a regular Routing Point or an external Routing Point.</ref>	1	Not applicable	True

> **Important**
> This information is also part of the configuration for the T-Server for CSTA Connector.

Supported BroadWorks Requests and Events

BroadWorks supports the following requests and events:

- BroadWorks Requests
- BroadWorks Events

BroadWorks Requests

The following tables represent the supported BroadWorks requests:

Call Requests

Call Requests

BroadWorks CTI	CSTA	Comments
BlindTransferRequest	Single Step Transfer Call	
CallRecordRequest	Record Message	Initiates On Demand Call recording.
ConferenceAddParticipantRequest	Conference Call	
ConferenceHoldRequest	Hold Call	
ConferenceReleaseRequest	Clear Call	
ConferenceRetrieveRequest	Retrieve Call	
ConferenceStartRequest	Conference Call	
ConsultativeTransferRequest	Transfer Call	

BroadWorks CTI	CSTA	Comments
DialRequest	Make Call, Consultation Call	
DirectedCallPickupRequest	Directed PickUp Call	
DirectedCallPickupWithBargeInRequest	Join Call	With the BargeIn Extensions attribute.
EscalateToSupervisorRequest	Consultation Call	Agent request
EmergencyCallToSupervisorRequest	Single Step Conference Call	Agent request
GetACDRequest	Snapshot Device	ACD request
GetRoutePointRequest	Snapshot Device	Routing Point request
GetRoutePointStateRequest	GetRoutingMode	Routing Point request
HoldRequest	Hold Call	
ModifyRoutePointStateRequest	SetRoutingMode	Routing Point request
MonitorCallRequest	Join Call	
MonitorNextCallRequest	Make Connection	

BroadWorks CTI	CSTA	Comments
MuteTransferRequest	Consultation Call	Private MuteTransfer Extensions attribute in request.
ReconnectRequest	Reconnect Call	
ReleaseRequest	Clear Connection	
RetrieveCallIdInfoListRequest	Snapshot Device	
RoutePointBounceCallRequest	Route Reject / Re-Route	Routing Point request
RoutePointBlindTransferRequest	Routing Select	Routing Point request
RoutePointDistributeCallRequest	Route Select	Routing Point request (CSTA Extensions attribute)
RoutePointFailCallRequest	Route End	Routing Point request
RoutePointOutgoingDialRequest	Make Predictive Call	Routing Point request
RoutePointReleaseCallRequest	Route Reject	Routing Point request
TalkRequest	Answer Call, Retrieve Call, Alternate Call	

BroadWorks CTI	CSTA	Comments
TransmitDTMFRequest	Generate Digits	

Device Requests

Device Requests

BroadWorks CTI	CSTA	Comments
GetCallForwardingAlwaysRequest	Get Forwarding	
GetCallForwardingBusyRequest	Get Forwarding	
GetCallForwardingNoAnswerRequest	Get Forwarding	
GetDoNotDisturbRequest	Get Do Not Disturb (DND)	
GetHotelingGuestRequest	Escape service	
GetHotelingHostListRequest	N/A	Internal Connector service
ModifyCallForwardingAlwaysRequest	Set Forwarding	
ModifyCallForwardingBusyRequest	Set Forwarding	

BroadWorks CTI	CSTA	Comments
ModifyCallForwardingNoAnswerRequest	Set Forwarding	
ModifyDoNotDisturbRequest	Set Do Not Disturb (DND)	
ModifyHotelingGuestRequest	Set Agent State or Escape Service	
SetAgentStateRequest	Set Agent State	
TagDispositionCodeRequest	Escape service	
TagDispositionCodeToCallRequest	Escape Service	

Media Requests

Media Requests

BroadWorks CTI	CSTA
RoutePointPlayBusyRequest	Play Message
RoutePointPlayMOHRequest	Play Message
RoutePointPlayRingbackRequest	Play Message

BroadWorks CTI	CSTA
RoutePointPlaySilenceRequest	Play Message, StartDataCollection
RoutePointPlayTreatmentRequest	Play Message, StartDataCollection

System Requests

System Requests

BroadWorks CTI	CSTA	Comments
AddChannelRequest	Not required	Internal Connector service
AddEnterpriseGroupSubscriptionRequest	Not required	
AddEnterpriseSubscriptionRequest	Not required	
AddServiceProviderGroupSubscriptionRequest	Not required	
AddServiceProviderSubscriptionRequest	Not required	
AddSystemSubscriptionRequest	Not required	
AddUserSubscriptionRequest	Start Monitor	
DeleteChannelRequest	Not required	Internal Connector

BroadWorks CTI	CSTA	Comments
		service
DeleteSubscriptionRequest	Stop Monitor	Internal Connector service
GetApplicationControllerStateRequest	Not required	Internal Connector KPL service
GetChannelSetRequest	Not required	Internal Connector service
GetSingleSubscriptionRequest	Not required	Internal Connector service
GetSubscriptionsRequest	Not required	Internal Connector service
SetApplicationControllerStateRequest	Route Register	
UpdateChannelRequest	Not required	Internal Connector service
UpdateSubscriptionRequest	Not required	Internal Connector service

BroadWorks Events

The following tables represent the supported BroadWorks events:

Call Events

Call Events

BroadWorks CTI	CSTA	Comments
ACDCallAbandonedEvent	Connection Cleared	
ACDCallAddedEvent	Queued	
ACDCallAnsweredByAgentEvent	Diverted	
ACDCallBouncedEvent	Delivered; Diverted	Translated to a diverted event, if the redirect field is provided.
ACDCallEscapedEvent	Connection Cleared	
ACDCallForwardedEvent	Diverted	
ACDCallOfferedToAgentEvent	Delivered	
ACDCallOverflowedEvent	Diverted	

BroadWorks CTI	CSTA	Comments
ACDCallPromotedEvent	Call Information	
ACDCallReleasedEvent	Connection Cleared	
ACDCallReorderedEvent	Call Information	
ACDCallStrandedEvent	Connection Cleared	
ACDCallTransferredEvent	Diverted	
ACDCallUpdatedEvent	Transferred	
ACDHolidayPolicyAppliedEvent	Diverted	
ACDNightPolicyAppliedEvent	Diverted	
ACDSubscriptionEvent	N/A	The internal CSTA Connector functionality
CallAnsweredEvent	Established	
CallBargedInEvent	Conferenced	
CallCollectingEvent	Service Initiated	With private data

BroadWorks CTI	CSTA	Comments
CallForwardedEvent	Diverted	
CallHeldEvent	Held	
CallMonitoredEvent	Conferenced	
CallOriginatedEvent	Originated	
CallOriginatingEvent	Service Initiated	
CallParkRetrievedEvent	Retrieved	
CallPicked-UpEvent	Diverted	
CallReceivedEvent	Delivered	
CallRecordEvent	Record	
CallRedirectedEvent	Diverted	
CallReleasedEvent	Connection Cleared	
CallReleasingEvent	Failed; Connection Cleared	

BroadWorks CTI	CSTA	Comments
CallRetrievedEvent	Retrieved	
CallSubscriptionEvent	---	The event may be translated into various computer-supported telecommunications applications (CSTA) events.
CallTransferredEvent	Transferred	
CallUpdatedEvent	---	Depends on call topology.
ConferenceHeldEvent	Held	
ConferenceReleasedEvent	Connection Cleared	
ConferenceRetrievedEvent	Retrieved	
ConferenceStartedEvent	Conferenced	Depends on call topology
ConferenceUpdatedEvent	Conferenced	
Not Available	Stop	The CSTA Stop event is distributed when the call is disconnected from the

BroadWorks CTI	CSTA	Comments
		device that requested the recording.
RoutePointCallAbandonedEvent	Connection Cleared	The Route End event is sent if the Routing dialog is opened.
RoutePointCallAddedEvent	Queued; Routing Request	
RoutePointCallAnsweredByAgentEvent	Diverted; Route End	
RoutePointCallBouncedEvent	Queued; Routing Request	
RoutePointCallFailedEvent	Diverted; Route End	The Route End event is sent if the Routing dialog is opened.
RoutePointCallForwardedEvent	Diverted; Route End	The Route End event is sent if the Routing dialog is opened.
RoutePointCallOfferedToAgentEvent	Route Event	
RoutePointCallOverflowedEvent	Diverted; Route End	The Route End event is sent if the Routing dialog is opened.

BroadWorks CTI	CSTA	Comments
RoutePointCallReleasedEvent	Connection Cleared, Route End	The Route End event is sent if the Routing dialog is opened.
RoutePointCallTransferredEvent	Diverted; Route End	
RoutePointCallUpdatedEvent	Transferred	
RoutePointFailedEvent	Connection Cleared; Route End	The Route End event is sent if the Routing dialog is opened.
RoutePointHolidayPolicyAppliedEvent	Connection Cleared; Route End	The Route End event is sent if the Routing dialog is opened.
RoutePointNightPolicyAppliedEvent	Connection Cleared; Route End	The Route End event is sent if the Routing dialog is opened.
RoutePointOutgoingCallAnsweredEvent	Queued; Routing Request	
RoutePointOutgoingCallOriginatedEvent	Originated, Delivered	
RoutePointRecoveredEvent	Routing Mode	
RoutePointSubscriptionEvent	N/A	The internal CSTA

BroadWorks CTI	CSTA	Comments
		Connector functionality
RoutePointWhisperStartedEvent	N/A	

Device Events

Device Events

BroadWorks CTI	CSTA	Comments
ACDAgentJoinUpdateEvent	• Agent Logged On • Agent Logged Off	
AgentSubscriptionEvent	Agent (varies)	A combination of agent events to provide the matching target agent state.
AgentDispositionCodeAddedEvent	Call Information	
AgentStateEvent	Agent (varies)	An appropriate CSTA event to provide the matching target agent state.
CallForwardingAlwaysEvent	Forwarding	
CallForwardingBusyEvent	Forwarding	

BroadWorks CTI	CSTA	Comments
CallForwardingNoAnswerEvent	Forwarding	
DoNotDisturbEvent	Do Not Disturb	
HotelingGuestEvent	• Agent Logged On • Agent Logged Off • Private Event	Distinguished from ACDAgentJoinUpdateEvent by the omission of the ACD Queue group.

Media Events

Media Events

BroadWorks CTI	CSTA
RoutePointBusyCompletedEvent	Stop
RoutePointBusyStartedEvent	TelephonyTones generated
RoutePointMOHCompletedEvent	Stop
RoutePointMOHStartedEvent	TelephonyTones generated
RoutePointRingbackCompletedEvent	Stop
RoutePointRingbackStartedEvent	TelephonyTones generated

BroadWorks CTI	CSTA
RoutePointSilenceCompletedEvent	Stop
RoutePointSilenceStartedEvent	TelephonyTones generated
RoutePointTreatmentCompletedEvent	Stop
RoutePointTreatmentStartedEvent	TelephonyTones generated

System Events

System Events

BroadWorks CTI	CSTA	Comments
ChannelTerminatedEvent	System Status, Monitor Stop	Monitor Stop sent for all subscriptions created over the terminated channel.
SubscriptionTerminatedEvent	Monitor Stop	

Web Application Path

Broadsoft BroadWorks uses vendor-supplied WEB applications to support both CTI request processing and CTI event reporting.

Subscription-Related Requests

Subscription-related requests use the following format:

`/<xsi_event_context>/v2.0/request_specific_parameters",`

where:

`<xsi_event_context>`—represents the application context of the Xsi-Events web application.

Call, Device, and Feature-Related Requests

Call, device, and feature-related requests use the following format:

`/<xsi_action_context>/v2.0/request_specific_parameters",`

where:

`<xsi_action_context>`—represents the application context of the Xsi-Actions web application.

Prior to BroadSoft BroadWorks version 17.sp4, these applications used the following predefined contexts: `/com.broadsoft.cti-events` and `/com.broadsoft.cti-actions` respectively.

Starting with BroadSoft BroadWorks version 17.sp4, the BroadWorks event WEB reporting application can be deployed into any path, therefore any application context can be used.

Related Configuration Options

The following configuration option enables or disables the Web Application Path feature:

xsi-actions-context

xsi-actions-context

Default Value: /com.broadsoft.cti-actions
Valid Values: Any valid context of the Xsi-Actions web application.
Changes Take Effect: Immediately

Represents the application context of the BroadSoft Xsi-Actions web application.

xsi-events-context

xsi-events-context

Default Value: /com.broadsoft.cti-events
Valid Values: Any valid context of the Xsi-Events web application.
Changes Take Effect: Immediately

Represents the application context of the BroadSoft Xsi-Events web application.

Configuration Options

You must configure the configuration objects and options described in the topics below in the Framework Configuration Layer:

CSTA Connector Application-Level Options

The configuration options specific to the CSTA Connector functionality are set in Configuration Manager, in the corresponding sections of the `Options` tab of the CSTA Connector `Application` object. The configuration options are configured in the following sections:

- Connector Section Options
- Link-%s Section Options
- License Section Options
- log-filter-x Section Options

Connector Section

This section must be called Connector.

app-controller

app-controller

Default Value: Mandatory field. No default value.
Valid Value: Any string
Changes Take Effect: Immediately

Required identifier used by BroadWorks routing functionality. Refer to the Route Point Failure Policy in the BroadWorks CTI documentation.

application-name

application-name

Default Value: GenesysConnector
Valid Value: Any string
Changes Take Effect: Immediately

Identifier used in BroadWorks event reporting to represent the Connector application.

broadworks-version

broadworks-version

Default Value: 17.sp4
Valid Values: Supported BroadWorks release version.
Changes Take Effect: Immediately
Related Feature: Call Recording

Specifies version "17.sp4" in all CTI requests sent to the BroadWorks XSI server and provides backwards compatibility with previous CSTA Connector releases.

channel-expire-tout

channel-expire-tout

Default Value: 3600
Valid Values: any positive integer
Changes Take Effect: on event channel creation or renewal

Specifies the expiration timeout, in seconds, when an event reporting channel is created or renewed.

channel-set-name

channel-set-name

Default Value: Mandatory field. No default value
Valid Values: Any string
Changes take effect: When the link is started or restarted

The ChannelSet name specifies a logical name for a CTI link to BroadWorks. Multiple TCP/IP channels opened to BroadWorks CTI server(s) with the same ChannelSet name are treated as a single CTI link. The Connector generates a unique ChannelSet name to ensure that there is no unwanted interaction between multiple Connectors connected to the same switch.

default-domain

default-domain

Default Value: None
Valid Value: Any string
Changes Take Effect: Immediately

> ### Important
> Genesys does not recommend the use of this option.

BroadWorks require a fully qualified device identifier in requests. A default domain name can be appended to a UserID attribute in requests for convenience. In the BroadWorks switch web configuration, the list of assigned domains can be seen by browsing to the Enterprise branch,

selecting the Resources section, and then selecting the Assign Domains sub-section. See, the `dn-prefix` option.

def-treatment-uri

def-treatment-uri

Default Value: None
Valid Value: Any valid URI path
Changes Take Effect: Immediately

The uniform resource identifier (URI) to a default treatment to apply in treatment requests. Refer to the vendor's configuration documentation for further information.

def-whisper-uri

def-whisper-uri

Default Value: None
Valid Value: Any valid URI path
Changes Take Effect: Immediately

The uniform resource identifier (URI) to the default whisper message to apply in routing requests with supervision. Refer to the vendor's configuration documentation for further information.

def-whisper-video-uri

def-whisper-video-uri

Default Value: None
Valid Value: Any valid URI path
Changes Take Effect: Immediately

The uniform resource identifier (URI) to the default whisper video message to apply in routing requests with supervision. Refer to the vendor's configuration documentation for further information.

dn-prefix

dn-prefix

Default Value: Mandatory field. No default value.
Valid Value: Any string
Changes Take Effect: Immediately

Numbers (DNs) are provisioned in BroadWorks and Configuration Manager using a shortened number format.
However, CTI requests require the fully qualified E.164 number. The DN Prefix is prepended to the short DN to provide the full E.164 number.

enable-rp-distribute

enable-rp-distribute

Default Value: `true`
Valid Value: `true`, `false`
Changes Take Effect: Immediately

Specifies the default method for non-supervised routing. If the value of this option is set to `true`, CSTA Connector selects distribute as the default method for non-supervised routing, otherwise blind transfer is used. See the `RPDistributeType` Extensions attribute in the Using the Extensions Attribute section for more information about how this option is used.

force-hoteling-guest

force-hoteling-guest

Default Value: `false`
Valid Value: `true`, `false`
Changes Take Effect: Immediately

Specifies whether CSTA Connector will forcibly associate and/or dissociate a Hoteling Guest association in a `ModifyHotelingGuest` request. Previously, CSTA Connector always set the `Active` flag to `true` when setting a Hoteling Guest association, and set the flag to false when removing a Hoteling Guest association. CSTA Connector was modified so that by default it will no longer pass the `Active` flag in a `ModifyHotelingGuest` request, although backwards compatibility can be maintained with the new option.

link-%s

link-%s

Default Value: No default value.
Valid Value: The `link-` string appended with any string
Changes Take Effect: Immediately

Specifies the section name containing the specific link configuration options for a particular CTI link.

max-outstanding

max-outstanding

Default Value: 16
Valid Value: 1-1000
Changes Take Effect: Immediately

Specifies the maximum number of outstanding unacknowledged requests sent to the switch at any given time.

password

password

Default Value: Mandatory field. No default value.
Valid Value: The password configured on the BroadWorks switch.
Changes Take Effect: When the link is started, or restarted.

Configures the password provisioned in the BroadWorks Administrator, which is required for request authentication on the XSP CTI interface.

restart-period

restart-period

Default Value: 10
Valid Values: Any positive integer between 3-300

Changes Take Effect: When the link is started or restarted.
Specifies the interval (in seconds) that the Connector waits between attempts to reconnect on all configured connections.

rq-gap

rq-gap

Default Value: 0 (zero)
Valid Value: 0-1000
Changes Take Effect: Immediately
Specifies the minimum interval, in milliseconds, between successive requests sent over the link to the PBX. The value can be adjusted to meet the CTI-link load and performance requirements.

subscribe-expire-tout

subscribe-expire-tout

Default Value: 3600
Valid Values: any positive integer
Changes Take Effect: on event subscription creation or renewal

Specifies the expiration timeout, in seconds, when an event subscription is created or renewed.

treatment-uri-pattern

treatment-uri-pattern

Default Value: Mandatory field. No default value.
Valid Value: Any string
Changes Take Effect: Immediately

Provides a pattern to create a uniform resource identifier (URI) for an announcement audio file in BroadWorks. When a value is provided, the Prompt format pattern is replaced with the provided value. When a value is not provided, the pattern is removed from the final URI.

Supported Formats

- `%i`—is replaced by the ID value

- %u—is replaced by the USER_ANN_ID value

- %t—is replaced by the USER_ID value

- %%—a single % character

For example:

- If ID = 1000, the following pattern, http://cti_provider.com/media/tones/%tAnn%i%u.wav is translated into http://cti_provider.com/media/tones/Ann1000.wav.

- If USER_ANN_ID = 999 and USER_ID = BWTenant, the same pattern is translated into http://cti_provider.com/media/tones/BWTenantAnn999.wav.

Restrictions and Limitations

1. The list size in the BroadWorks `RoutePointPlayTreatmentRequest` can have up to four entries. Up to four Prompts can be configured for use with the `treatment-uri-pattern` option. Prompts with more than four numbers are ignored.

2. The Prompt parameters, TEXT and `DIGITS`, are not supported. Prompts containing these parameters are considered as invalid and further Prompt conversion to URI is stopped.

username

username

Default Value: Mandatory field. No default value.
Valid Value: The username configured on the BroadWorks switch.
Changes Take Effect: When the link is started, or restarted.

This option configures the username provisioned in the BroadWorks Administrator, which is required for request authentication on the XSP CTI interface.

xsi-actions-context

xsi-actions-context

Default Value: `/com.broadsoft.cti-actions`
Valid Values: Any valid context of the Xsi-Actions web application.
Changes Take Effect: Immediately

Represents the application context of the BroadSoft Xsi-Actions web application.

xsi-events-context

xsi-events-context

Default Value: `/com.broadsoft.cti-events`
Valid Values: Any valid context of the Xsi-Events web application.
Changes Take Effect: Immediately

Represents the application context of the BroadSoft Xsi-Events web application.

link-%s Section

This section name represents a specific Connector link to the switch and is user-customisable with %s being replaced by any string.

hostname

hostname

Default Value: Mandatory field. No default value.
Valid Value: Any valid host name
Changes Take Effect: When the link is started, or restarted.

Specifies the hostname/IP address of the switch/XSP Server. You must specify a value for this option.

kpl-interval

kpl-interval

Default Value: 10
Valid Value: Any integer from 0-600
Changes Take Effect: Immediately
Related Feature: Keep-Alive Feature Handling

Specifies a *keep-alive* time interval (in seconds). To check network connectivity, CSTA Connector issues a dedicated KPL request to BroadWorks XSP at the interval specified when there is no other activity on the link. A value of 0 (zero) disables this feature.

This configuration option can be configured in both the Connector and in the link-%s sections. The options configured in the link-%s section take precedence over the options configured in the Connector section and are applied to this particular link. See, the kpl-tolerance option.

kpl-tolerance

kpl-tolerance

Default Value: 0 (zero)
Valid Value: Any integer from 0-10

Changes Take Effect: Immediately
Related Feature: Keep-Alive Feature Handling

Specifies the number of failed keep-alive requests that CSTA Connector permits before considering the XSP link as unresponsive. When an unresponsive link is detected, CSTA Connector closes and reopens this particular XSP link.

This configuration option can be configured in both the `Connector` and in the `link-%s` sections. The options configured in the `link-%s` section take precedence over the options configured in the `Connector` section and are applied to this particular link. See, the kpl-interval option.

port

port

Default Value: 8011
Valid Value: Any valid TCP/IP port
Changes Take Effect: When the link is started or restarted.

Specifies the port address of the switch interface. This option must be specified.

priority

priority

Default Value: 0 (zero)
Valid Values: `0-100`
Changes Take Effect: When the link is started or restarted.

Specifies the priority of the connection. Higher numbers have higher priority.

License Section

Connector is a licensed product and can only be operational if a valid license is applied. Licensing uses the FlexLM library in common with T-Server and is configured in Configuration Manager.

num-sdn-licenses

Default Value: max
Valid Value: Any positive integer or max
Changes Take Effect: Immediately

Specifies how many seat-related licenses Connector checks out. A value of 0 (zero) means that Connector does not grant control of seat-related DNs to any client, and it does not look for seat-related DN licenses at all. When set to Max, all available licenses are checked out.

log-filter-x Section

Use the `log-filter-x` section configuration options to hide sensitive data from log(s). The sensitive data is masked by asterisks (*) in the log file. See, Hiding Data in Logs for further details.

Connector also uses the `log` section which is common to T-Server. See, The CSTA T-Server for Connector Deployment Guide wiki for full details of these options.

Any

(Anything can be used for the option name(s) in this section.)

Default Value: None
Valid Value: Perl-compatible regular expression (PCRE)
Changes Take Effect: Immediately

For detailed information about PCRE syntax, please refer to http://pcre.org.

A PCRE with or without string sub-patterns. If the expression contains a parenthesis-delimited `capture` sub-expressions, these sub-string matches are masked. Otherwise, the remainder of the string after the matching expression are masked. To define a sub-string that is a part of matching expression, but is not a part of the sensitive data, a `non-capturing` sub-expression can be used and is denoted by (?:).

For example, the pattern: `User (?:PIN|pin|account)` can be used to mask everything after the full match of the word `User` followed by PIN, `pin` or `account`, but these words themselves are not masked. A case insensitive match can be forced by prefixing the expression with the following PCRE option: (?i).

Examples:

`[Pp]assword:? *`

Masks the following: everything after the word `password`, the possible colon and any white space up to the end of the line. There are no sub-patterns, so everything is masked after a greedy algorithm match of the entire pattern.

For example:

```
User password (syspw) becomes
User password *******
```

`[dD]igits.*"([^"])"`

Masks everything between the double quotes that is preceded by `digits`.

For example:

```
collectedDigits: numberString "773645" becomes
collectedDigits: numberString "******"
```

```
(?i)account.*(\d{4})\d{2}-(\d{3-5})\d+
```

The account consists of two parts:

1. a 6-digit sort code

2. a dash and a variable-length number that is four digits or more

The pattern inhibits the first four digits of the sort code (first sub-pattern), up to the five first digits of the account number (second sub-pattern) but shows the last two digits of the sort code and at least one last number of the account number, or all of the last numbers after the initial five digits that are suppressed.

Matching of the word account is case-insensitive due to the preceding option.