Cisco Extended Care 1.0 API Reference Guide

Software Release 1.0
November 20, 2013

Cisco Systems, Inc.
www.cisco.com

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco website at www.cisco.com/go/offices.

Text Part Number: OL-30301-01

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

Cisco Extended Care 1.0 API Reference Guide
© 2013 Cisco Systems, Inc. All rights reserved.

CHAPTER 1 **Introduction** 1-1

1.1 Intended Audience 1-1

1.2 Overview 1-1

1.3 Not Intended for Use In Emergency or for Patient Monitoring 1-1

1.4 What is Included in this Document 1-2

1.5 Cisco Extended Care Software Components 1-2

1.5.1 Cisco Extended Care Application Server Components 1-2

1.6 Getting Started 1-2

1.6.1 Using This Manual 1-2

CHAPTER 2 **The Extended Care Collaboration API** 2-1

API Overview 2-1

2.1 Extended Care Video Call Types 2-1

2.2 Call Control APIs 2-2

2.2.1 Start video call 2-2

2.2.2 End Video Call 2-13

2.2.3 Make video call (Start or Join existing Video Call) 2-14

2.2.4 Leave video call API 2-19

2.2.5 End all video calls API 2-20

2.3 Call Status APIs 2-21

2.3.1 Get call status API 2-21

2.3.2 Get all call status API 2-25

2.4 Camera Preset APIs 2-28

2.4.1 List camera preset API 2-28

2.4.2 List all camera preset API 2-31

2.4.3 Store camera preset API 2-34

2.4.4 Edit camera preset API 2-37

2.4.5 Remove camera preset API 2-40

2.4.6 Activate camera preset API 2-43

2.5 Camera Positioning APIs 2-46

2.5.1 Change camera position API 2-46

2.5.2 Pan camera right API 2-49

2.5.3 Pan camera left API 2-52
2.5.4 Tilt camera up API 2-54
2.5.5 Tilt camera down API 2-57
2.5.6 Zoom in the camera API 2-59
2.5.7 Zoom out the camera API 2-62
2.5.8 Trigger autofocus API 2-64
2.6 Utility APIs 2-67
2.6.1 Get endpoint capabilities 2-67

CHAPTER 3 **The Call Control Application** 3-1
3.1 Call Control Java Client (For REST API calls) 3-2
3.2 IFrame-Based Applications 3-3

APPENDIX A **Common Code Definitions** A-1
A.1 Message Response Success Codes A-1
A.2 Message Response Failure Codes A-1

APPENDIX B **API Availability** B-1

APPENDIX C **Software Compatibility** C-1
Browser Compatibility C-1

GLOSSARY

Introduction

Revised: November 20, 2013, OL-30301-01

1.1 Intended Audience

This document lists the APIs that an external system can use for integrating with Cisco Extended Care for collaboration purposes. Please note that this document lists just the software APIs that an external system should use. Various other specifications are listed in the ***Cisco Extended Care 1.0 Release Notes***.

1.2 Overview

Cisco Extended Care is a personal health and wellness collaboration platform, enabling patient engagement and care team interactions at any time and from anywhere.

The solution provides:

- Enhanced and efficient care coordination
- Secure messaging
- Appointment calendar and scheduling
- Ability to access questionnaires
- Real time video collaboration
- Anywhere access to care team consults
- Personalized content sharing
- Video platform to drive health and wellness awareness and education
- Open APIs to enable third party applications and environments

1.3 Not Intended for Use In Emergency or for Patient Monitoring

Cisco Extended Care is not intended for use in emergency situations. In the event of an emergency, call 911 or your local emergency response system. Cisco Extended Care is not for use in situations involving real-time patient monitoring or alarming.

1.4 What is Included in this Document

The ***Cisco Extended Care 1.0 API Reference Guide*** outlines the Application Programming Interface for Cisco Extended Care.

1.5 Cisco Extended Care Software Components

The Cisco Extended Care software integrates several different software components.

1.5.1 Cisco Extended Care Application Server Components

The Cisco Extended Care Application Server is the data center software that manages all of the connectivity. It is functionally comprised of the following:

- **Cisco Extended Care Application Server –** ties together all the components of Cisco Extended Care. Specifically, the Cisco Extended Care Application Server:
  - maintains the master information about resources
  - manages sessions and appointments
  - Cisco Extended Care Admin, Cisco Extended Care Portal, and Unified Communications (UC) servers
  - interfaces with applications supporting Cisco Extended Care connectors.
- **Cisco Extended Care Portal –** maintains the portal login sessions, provides the UI to the providers and interfaces to the Application Server to validate sessions.
- **Cisco Extended Care Administration** - is the management server used by the installation team to configure, administer and manage the Application Server and the Portal.
- **Cisco Extended Care Patient Portal –** maintains the patient portal login sessions, provides the UI to the providers and interfaces to the Application Server to validate sessions.

1.6 Getting Started

1.6.1 Using This Manual

There are three main concepts described in this document:

- The APIs themselves
- The call control application
- The common code definitions

The Extended Care Collaboration API

Revised: November 20, 2013, OL-30301-01

API Overview

The Extended Care Collaboration APIs provide a standard way to interact with various video endpoints.

General considerations for the Collaboration APIs:

- The APIs are exposed as REST interfaces accessible over HTTP(S).
- The request and response data are represented in xml. Request body could be empty for some APIs, but there will always be a response body.
- Authentication happens through headers.
- URL may have path variables which are typically unique identifiers of a given entity.
- All URL path variables are mandatory.
- Request parameters are generally not used unless specifically mentioned in the API.

Please refer to section Chapter 3, "The Call Control Application"for interaction with software based UC endpoints or to use java based client to access these REST APIs.

2.1 Extended Care Video Call Types

The Extended Care video calls can be broadly categorized into three types. Note that video conferencing does not require Cisco equipment, except for switchable peer to peer as noted in the following list:

- Peer-to-Peer (P2P) - This type of call can have only two endpoints. This is a typical one-to-one call. Please contact Cisco for a list of endpoints supported.
- Multiparty (MP) - This is a conference call where in every endpoint is placed into a bridge and users will see video feeds from more than one endpoints. Note, in addition to the UC endpoints and servers, there should be bridges to support this call type.
- Switchable Peer-to-Peer - This call type is used to make multi-party calls using the inbuilt bridge capability of EX90 endpoints. A maximum of four endpoints can be in a video call. For the call to work, at least one of the first two endpoints in the call must be an EX90.

Note If needed, the ExCare administrator can enforce system level policy to allow only P2P or only MP calls.

The APIs can generally be categorized as follows:

- Call Control APIs
- Call Status APIs
- Camera Preset APIs
- Camera Positioning APIs

2.2 Call Control APIs

2.2.1 Start video call

Start a Video call based on the given conference type among the given list of endpoints. Optionally, camera position / preset input can be given to change camera Pan / Tilt / Zoom if the UC endpoint supports camera positioning. This API is helpful to start a call without any interaction required from each of the endpoint users.

2.2.1.1 Start video call API Definition

URL pattern: https://{IP}:{port}/chpas/devicecontrol/videocall/start

URL Path Variables:None

HTTP Method: POST

HTTP Headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request Body Definition:

Name	Type	Mandatory	Description
CallType	Enum – P2P, MP, SWITCHABLE_P2P	Yes	The type of call to make
VideoCallSessionReq	Object	Yes	Xml representation of the endpoint information
VideoCallSession-Req: EndpointId	String	Yes	The endpoint Id to start the call
VideoCallSession-Req: CameraPresetId	String	No	Activates the given camera preset.

VideoCallSession-Req: CameraPosition	Object	No	Xml representation of Camera position information
CameraPosition: CameraId	Integer	No	The id of the camera.
CameraPosition: Pan	Integer	No	Pan value
CameraPosition: Tilt	Integer	No	Tilt value
CameraPosition: Zoom	Integer	No	Zoom value
CameraPosition: Focus	Integer	No	Focus value

HTTP Response Body Definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success
Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
VideoCallId	String	Yes	UUID to uniquely identify the video call
ConferenceType	String	Yes	The type of conference for this video call.
HasActive ConferenceSessions	Boolean	Yes	Indicates if there is at least one active session in this call.
ConferenceSessions	List	No. Available if at least one session is active.	List of objects representing the sessions in this call.

ConferenceSessions: ConferenceSession	Object	No. Available if at least one session is active.	Xml representation of Call Session Info
ConferenceSession: ConferenceSessionId	Integer	No	ID for a given session of the call.
ConferenceSession: StartTime	Integer	No	Time when this session of the call started. Represented in milliseconds from epoch.
ConferenceSession: EndpointId	String	No	The endpoint id from which this session is started.
ConferenceSession: ConferenceSessionStatus	String	No	Status of this video call session. IDLE, CONNECTED, FAILED, DISCONNECTED.

2.2.1.2 Start video call Sucess Scenarios

2.2.1.2.1 Peer to Peer Call

Sample URL	https://localhost/chpas/devicecontrol/videocall/start
Sample HTTP Request Xml	<MakeVideoCallRequest> <CallType>P2P</CallType> <VideoCallSessions> <VideoCallSessionReq> <EndpointId>EX60-1</EndpointId> </VideoCallSessionReq> <VideoCallSessionReq> <EndpointId>EX60-2</EndpointId> </VideoCallSessionReq> </VideoCallSessions> </MakeVideoCallRequest>
Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage > </Status> <ConferenceCallInfo> <VideoCallId>92562f42-5977-4141-bfe8-fb488ca24154</VideoCal- lId> <ConferenceType>P2P</ConferenceType> <HasActiveConferenceSessions>true</HasActiveConferenceSes- sions> <ConferenceSessions> <ConferenceSession> <ConferenceSessionId>2</ConferenceSessionId> <SessionId>device_ctrl_sessn:EX60-1</SessionId> <StartTime>1366211280203</StartTime> <EndpointId>EX60-1</EndpointId> <ConferenceSession- Status>Connected</ConferenceSessionStatus> </ConferenceSession> <ConferenceSession> <ConferenceSessionId>3</ConferenceSessionId> <SessionId>device_ctrl_sessn:EX60-2</SessionId> <StartTime>1366211280342</StartTime> <EndpointId>EX60-1</EndpointId> <ConferenceSession- Status>Connected</ConferenceSessionStatus> </ConferenceSession> </ConferenceSessions> </ConferenceCallInfo> </Response>

2.2.1.2.2 Multiparty Calls

Sample URL	https://localhost/chpas/devicecontrol/videocall/start
Sample HTTP Request Xml	<MakeVideoCallRequest> <CallType>MP</CallType> <VideoCallSessions> <VideoCallSessionReq> <EndpointId>EX60</EndpointId> </VideoCallSessionReq> <VideoCallSessionReq> <EndpointId>EX90-1</EndpointId> </VideoCallSessionReq> </VideoCallSessions> </MakeVideoCallRequest>

Sample HTTP Response Xml	`<Response>`
	`  <Status>`
	`  <StatusCode>200</StatusCode>`
	`  <StatusMessage>Success</StatusMessage >`
	`  </Status>`
	`<ConferenceCallInfo>`
	`  <VideoCallId>92562f42-5977-4141-bfe8-fb488ca24153</VideoCal-lId>`
	`  <ConferenceType>MP</ConferenceType>`
	`  <HasActiveConferenceSessions>true</HasActiveConferenceSes-sions>`
	`  <ConferenceSessions>`
	`  <ConferenceSession>`
	`  <ConferenceSessionId>2</ConferenceSessionId>`
	`  <SessionId>device_ctrl_sessn:EX60-1</SessionId>`
	`  <StartTime>1366211280203</StartTime>`
	`  <EndpointId>EX60-1</EndpointId>`
	`  <ConferenceSessionStatus>Connected</ConferenceSessionSta-tus>`
	`  </ConferenceSession>`
	`  <ConferenceSession>`
	`  <ConferenceSessionId>3</ConferenceSessionId>`
	`  <SessionId>device_ctrl_sessn:EX60-2</SessionId>`
	`  <StartTime>1366211280342</StartTime>`
	`  <EndpointId>EX60-1</EndpointId>`
	`  <ConferenceSessionStatus>Connected</ConferenceSessionSta-tus>`
	`  </ConferenceSession>`
	`  </ConferenceSessions>`
	`</ConferenceCallInfo>`
	`</Response>`

2.2.1.2.3 Switchable P2P Call (using EX90 as the bridge)

Sample URL	https://localhost/chpas/devicecontrol/videocall/start

Sample HTTP Request Xml	<MakeVideoCallRequest> <CallType>SWITCHABLE_P2P</CallType> <VideoCallSessions> <VideoCallSessionReq> <EndpointId>EX90-1</EndpointId> </VideoCallSessionReq> <VideoCallSessionReq> <EndpointId>EX60-1</EndpointId> </VideoCallSessionReq> </VideoCallSessions> </MakeVideoCallRequest>
Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage > </Status> <ConferenceCallInfo> <VideoCallId>92562f42-5977-4141-bfe8-fb488ca24152</VideoCal- lId> <ConferenceType>SWITCHABLE_P2P</ConferenceType> <HasActive- ConferenceSessions>true</HasActiveConferenceSessions> <ConferenceSessions> <ConferenceSession> <ConferenceSessionId>2</ConferenceSessionId> <SessionId>device_ctrl_sessn:EX90-1</SessionId> <StartTime>1366211280203</StartTime> <EndpointId>EX90-1</EndpointId> <ConferenceSessionStatus>Connected</ConferenceSessionSta- tus> </ConferenceSession> <ConferenceSession> <ConferenceSessionId>3</ConferenceSessionId> <SessionId>device_ctrl_sessn:EX60-1</SessionId> <StartTime>1366211280342</StartTime> <EndpointId>EX60-1</EndpointId> <ConferenceSessionStatus>Connected</ConferenceSessionSta- tus> </ConferenceSession> </ConferenceSessions> </ConferenceCallInfo> </Response>

2.2.1.2.4 Video Call with Camera Positioning (PTZ)

Sample URL	https://localhost/chpas/devicecontrol/videocall/start
Sample HTTP Request Xml	<MakeVideoCallRequest> <CallType>SWITCHABLE_P2P</CallType> <VideoCallSessions> <VideoCallSessionReq> <EndpointId>C20-1</EndpointId> <CameraPosition> <CameraId>1</CameraId> <Pan>500</Pan> <Tilt>10</Tilt> <Zoom>0</Zoom> </CameraPosition> </VideoCallSessionReq> <VideoCallSessionReq> <EndpointId>EX90-1</EndpointId> </VideoCallSessionReq> </VideoCallSessions> </MakeVideoCallRequest>

Sample HTTP Response Xml	`<Response>`
	`  <Status>`
	`  <StatusCode>200</StatusCode>`
	`  <StatusMessage>Success</StatusMessage >`
	`  </Status>`
	`<ConferenceCallInfo>`
	`  <VideoCallId>92562f42-5977-4141-bfe8-fb488ca24152</VideoCal-lId>`
	`  <ConferenceType>SWITCHABLE_P2P</ConferenceType>  <HasActive-ConferenceSessions>true</HasActiveConferenceSessions>`
	`  <ConferenceSessions>`
	`  <ConferenceSession>`
	`  <ConferenceSessionId>2</ConferenceSessionId>`
	`  <SessionId>device_ctrl_sessn:EX90-1</SessionId>`
	`  <StartTime>1366211280203</StartTime>`
	`  <EndpointId>C20-1<EndpointId>  <ConferenceSessionSta-tus>Connected</ConferenceSessionStatus>`
	`  <CameraPositionSetResult>`
	`  <CameraPositionSetResult item="1" status="OK">`
	`  </CameraPositionSetResult>`
	`  </CameraPositionSetResult>`
	`  </ConferenceSession>`
	`  <ConferenceSession>`
	`  <ConferenceSessionId>3</ConferenceSessionId>`
	`  <SessionId>device_ctrl_sessn:EX60-1</SessionId>`
	`  <StartTime>1366211280342</StartTime>`
	`  <EndpointId>EX90-1</EndpointId>  <ConferenceSession-Status>Connected</ConferenceSessionStatus>`
	`  </ConferenceSession>`
	`  </ConferenceSessions>`
	`</ConferenceCallInfo>`
	`</Response>`

2.2.1.2.5 Video Call with Camera Positioning (Preset Id)

Sample URL	https://localhost/chpas/devicecontrol/videocall/start

Sample HTTP Request Xml	<MakeVideoCallRequest> <CallType>SWITCHABLE_P2P</CallType> <VideoCallSessions> <VideoCallSessionReq> <EndpointId>C20-1</EndpointId> <CameraPresetId>5</CameraPresetId> </VideoCallSessionReq> <VideoCallSessionReq> <EndpointId>EX60-1</EndpointId> </VideoCallSessionReq> </VideoCallSessions> </MakeVideoCallRequest>
Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage > </Status> <ConferenceCallInfo> <VideoCallId>92562f42-5977-4141-bfe8-fb488ca24152</VideoCal- lId> <ConferenceType>SWITCHABLE_P2P</ConferenceType> <HasActive- ConferenceSessions>true</HasActiveConferenceSessions> <ConferenceSessions> <ConferenceSession> <ConferenceSessionId>2</ConferenceSessionId> <SessionId>device_ctrl_sessn:EX90-1</SessionId> <StartTime>1366211280203</StartTime> <EndpointId>C20-1<EndpointId> <ConferenceSessionSta- tus>Connected</ConferenceSessionStatus> <CameraPositionSetResult> <PresetActivateResult item=”1” status=”OK”> </PresetActivateResult> </CameraPositionSetResult> </ConferenceSession> <ConferenceSession> <ConferenceSessionId>3</ConferenceSessionId> <SessionId>device_ctrl_sessn:EX60-1</SessionId> <StartTime>1366211280342</StartTime> <EndpointId>EX90-1</EndpointId> <ConferenceSession- Status>Connected</ConferenceSessionStatus> </ConferenceSession> </ConferenceSessions> </ConferenceCallInfo> </Response>

2.2.1.3 Failure Scenarios

Scenario	Response XML
Invalid Call Type	<Response> <Status> <StatusCode>543</StatusCode> <StatusMessage>Unsupported Conference Type</StatusMessage > </Status> </Response>
Invalid Endpoint Id	<Response> <Status> <StatusCode>422</StatusCode> <StatusMessage>INVALID ENDPOINT ID : Invalid-Ep-1</StatusMes-sage > </Status> </Response>
Invalid Camera PTZ parameter	<Response> <Status> <StatusCode>425</StatusCode> <StatusMessage>Invalid Value for parameter {Pan / Tilt / Zoom / Focus} </StatusMessage > </Status> </Response>
Endpoint does not support PTZ	<Response> <Status> <StatusCode>5101</StatusCode> <StatusMessage>Endpoint {endpointId} does not support Pan, Tilt, Zoom and Focus Capabilities</StatusMessage > </Status> </Response>

Invalid Camera PresetId	<Response> <Status> <StatusCode>425</StatusCode> <StatusMessage>Invalid Preset Id</StatusMessage > </Status> </Response>

2.2.2 End Video Call

2.2.2.1 End Video Call API Definition

URL pattern	https://{IP}:{port}/chpas/devicecontrol/videocall/end/{videoCallId}
URL Path Variables	videoCallId – The video call to end
HTTP method	GET
HTTP headers	chp_auth_component: callcontrol-service chp_auth_token: <Auth token provided>
HTTP Request body Definition	None
HTTP Response body Definition	None

2.2.2.2 End Video Call Success Scenarios

2.2.2.2.1 End an ongoing call that provides a valid videoCallId

Sample URL	https://localhost/chpas/devicecon-trol/end/92562f42-5977-4141-bfe8-fb488ca24152
Sample HTTP Request Xml	N/A

Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage > </Status> </Response>

2.2.2.3 End Video Call Failure Scenarios

Scenario	Sample URL	Sample HTTP Response xml
Empty Video Call Id	https://localhost/chpas/devicecontrol/end	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLEMENTED</StatusMessage > </Status> </Response>
Invalid Video Call Id	https://localhost/chpas/devicecontrol/end/`invalid-video-call-1`	<Response> <Status> <StatusCode>5100</StatusCode> <StatusMessage>Video call is not available</StatusMessage > </Status> </Response>

2.2.3 Make video call (Start or Join existing Video Call)

The calling endpoint should provide a unique ID to make a video call. If there is an existing video call by that Id, the endpoint will be added into the call (given that the conference type allows the operation). If there is no existing video call by the given Id, a new video call will be created and the endpoint will be placed into the call. This API shall be used if a call can be started by any one of the users joining into the call. This API also complements the “Start Video Call” API if an additional users need to be placed into an existing call.

2.2.3.1 Make video call API Definition

URL pattern:chpas/devicecontrol/videocall/make/{videoCallId}/{conferenceType}

URL Path Variables:

- videoCallId – The video call to start or join. Note if using this API, the videoCallId has to be generated by the client and passed in. For all other APIs, the videoCallId will be generated by the system.
- conferenceType – The Conference type of the call – P2P, MP or SWITCHABLE_P2P

HTTP method: POST

HTTP headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request Body Definition

Name	Type	Mandatory	Description
VideoCallSessionReq	Object	Yes	Xml representation of the endpoint information
VideoCallSession-Req: EndpointId	String	Yes	The endpoint Id to start or join the call
VideoCallSession-Req: CameraPresetId	String	No	Activates the given camera preset.
VideoCallSession-Req: CameraPosition	Object	No	Xml representation of Cam-era position information
CameraPosition: CameraId	Integer	No	The id of the camera.
CameraPosition: Pan	Integer	No	Pan value
CameraPosition: Tilt	Integer	No	Tilt value
CameraPosition: Zoom	Integer	No	Zoom value
CameraPosition: Focus	Integer	No	Focus value

HTTP Response Body Definition

Name	Type	Mandatory	Description
ConferenceSessionId	Integer	No	ID for a given session of the call.
StartTime	Integer	No	Time when this session of the call started. Represented in milliseconds from epoch.
EndpointId	String	No	The endpoint id from which this session is started.
ConferenceSessionStatus	String	No	Status of this video call session. IDLE, CONNECTED, FAILED, DISCONNECTED.

2.2.3.2 Make video call Success Scenarios

2.2.3.2.1 Start or Join a Video Call

If the videoCallId is new, a new call will be created. Else, user will be put in to the existing call by that Id.

Sample URL	https://localhost/chpas/devicecontrol/video-call/make/92562f42-5977-4141-bfe8-fb488ca24152/MP

Sample HTTP Request Xml	`<VideoCallSessionReq>`
	`  <EndpointId>EX90-1</EndpointId>`
	`</VideoCallSessionReq>`

Sample HTTP Response Xml	`<Response>`
	`  <Status>`
	`  <StatusCode>200</StatusCode>`
	`  <StatusMessage>Success</StatusMessage >`
	`  </Status>`
	`  <ConferenceSession>`
	`  <ConferenceSessionId>2</ConferenceSessionId>`
	`  <SessionId>device_ctrl_sessn:EX90-1</SessionId>`
	`  <StartTime>1366211280203</StartTime>`
	`  <EndpointId>EX90-1</EndpointId>  <ConferenceSession-Status>Connected</ConferenceSessionStatus>  </ConferenceSes-sion>`
	`</Response>`

2.2.3.2.2 Start or Join a video call with Camera Position (Preset Id)

Sample URL	https://localhost/chpas/devicecontrol/video-call/make/92562f42-5977-4141-bfe8-fb488ca24152/MP
Sample HTTP Request Xml	<VideoCallSessionReq> <EndpointId>EX90-1</EndpointId> <CameraPresetId>5</CameraPresetId> </VideoCallSessionReq>
Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage > </Status> <ConferenceSession> <ConferenceSessionId>2</ConferenceSessionId> <SessionId>device_ctrl_sessn:EX90-1</SessionId> <StartTime>1366211280203</StartTime> <EndpointId>EX90-1</EndpointId> <ConferenceSession-Status>Connected</ConferenceSessionStatus> <CameraPositionSetResult> <PresetActivateResult item="1" status="OK"> </PresetActivateResult> </CameraPositionSetResult> </ConferenceSession> </Response>

2.2.3.2.3 Start or join video call – with Camera Position – PTZ

Sample URL	https://localhost/chpas/devicecontrol/video-call/make/92562f42-5977-4141-bfe8-fb488ca24152/MP

Sample HTTP Request Xml	<VideoCallSessionReq> <EndpointId>EX90-1</EndpointId> <CameraPosition> <CameraId>1</CameraId> <Pan>500</Pan> <Tilt>10</Tilt> <Zoom>0</Zoom> </CameraPosition> </VideoCallSessionReq>
Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage > </Status> <ConferenceSession> <ConferenceSessionId>2</ConferenceSessionId> <SessionId>device_ctrl_sessn:EX90-1</SessionId> <StartTime>1366211280203</StartTime> <EndpointId>EX90-1</EndpointId> <ConferenceSession-Status>Connected</ConferenceSessionStatus> <CameraPositionSetResult> <CameraPositionSetResult item="1" status="OK"> </CameraPositionSetResult> </CameraPositionSetResult> </ConferenceSession> </Response>

2.2.3.3 Make video calls Failure Scenario

Scenario	Sample HTTP Request	Sample HTTP Response xml
Invalid Endpoint Id	https://localhost/chpas/devicecontrol/video-call/make/92562f42-5977-4141-bfe8-fb488ca24152/MP <VideoCallSessionReq> <EndpointId>invalid-ep-1</End-pointId> </VideoCallSessionReq>	<Response> <Status> <StatusCode>422</StatusCode> <StatusMessage>INVALID ENDPOINT ID : Invalid-Ep-1</StatusMessage > </Status> </Response>

Invalid Call Type	https://localhost/chpas/devicecontrol/video-call/make/92562f42-5977-4141-bfe8-fb488ca24152/invalid-call-type <VideoCallSessionReq> <EndpointId>EX90-1</EndpointId> </VideoCallSessionReq>	<Response> <Status> <StatusCode>543</StatusCode> <StatusMessage>Unsupported Conference Type</StatusMessage > </Status> </Response>

2.2.4 Leave video call API

This API disconnects the given endpoint from an on-going video call. If the endpoint is the last one in the call, the call will be terminated and cleaned up. This API is used in tandem with the "Make Video Call" API described in the above section.

2.2.4.1 Leave video call API Definition

URL pattern	/chpas/devicecontrol/videocall/leave/{endpointId}
URL Path Variables	endpointId – The endpoint Id to disconnect.
HTTP method	GET
HTTP headers	chp_auth_component: callcontrol-service chp_auth_token: <Auth token provided>
HTTP Request body definition	None
HTTP Response body definition	None

2.2.4.2 Leave video call Success Scenario

2.2.4.2.1 Leave video call

Sample URL	https://localhost/chpas/devicecontrol/leave/EX60-1
Sample HTTP Request Xml	N/A

Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage > </Status> </Response>

2.2.4.3 Leave video call Failure Scenarios

Scenario	Sample HTTP URL	Sample HTTP Response xml
Invalid Endpoint Id	https://localhost/chpas/devicecontrol/leave/invalid-ep-1	<Response> <Status> <StatusCode>422</StatusCode> <StatusMessage>INVALID ENDPOINT ID : Invalid-Ep-1</StatusMessage > </Status> </Response>
Endpoint is not in the call.	https://localhost/chpas/devicecontrol/leave/EX60-1	<Response> <Status> <StatusCode>421</StatusCode> <StatusMessage>No Session Info available for the given endpoint : EX60-1</StatusMessage > </Status> </Response>

2.2.5 End all video calls API

Ends all on going calls and disconnects all endpoints. This can be used as an admin API to clear any open / dangling video calls.

2.2.5.1 End all video calls API Definition

URL pattern	/chpas/devicecontrol/videocall/endall
URL Path Variables	None

HTTP method	GET
HTTP headers	chp_auth_component: callcontrol-service chp_auth_token: <Auth token provided>
HTTP Request body definition	None
HTTP Response body definition	None

2.2.5.2 End all video calls Success Scenario

2.2.5.2.1 End all video calls

Sample URL	https://localhost/chpas/devicecontrol/endall
Sample HTTP Request Xml	N/A
Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage > </Status> </Response>

2.2.5.3 End all video calls Failure Scenarios

None.

2.3 Call Status APIs

Call status APIs are used to retrieve the status of on-going calls.

2.3.1 Get call status API

Get the call status for a particular video call based on the videoCallId, passed to the API.

2.3.1.1 Get call status API Definition

URL pattern:/chpas/devicecontrol/videocall/status/{ videoCallId }

URL path variables:videoCallId – The videoCallId for which to get status.

HTTP method: GET

- HTTP headers:chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition: None.

HTTP Response body definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success
Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
ConferenceCallInfo	Object	Yes	Xml representation of the video call information.
ConferenceCallInfo: VideoCallId	String	Yes	UUID to uniquely identify the video call
ConferenceCallInfo: ConferenceType	String	Yes	The type of conference for this video call.
ConferenceCallInfo: HasActive ConferenceSessions	Boolean	Yes	Indicates if there is at least one active session in this call.
ConferenceCallInfo: ConferenceSessions	List	No. Available if at least one session is alive.	List of objects representing the sessions in this call.
ConferenceSessions: ConferenceSession	Object	No. Available if at least one session is alive.	Xml representation of Call Session Info
ConferenceSession: ConferenceSessionId	Integer	No	ID for a given session of the call.

ConferenceSession: StartTime	Integer	No	Time when this session of the call started. Represented in milliseconds from epoch.
ConferenceSession: EndpointId	String	No	The endpoint id from which this session is started.
ConferenceSession: ConferenceSessionStatus	String	No	Status of this video call session. IDLE, CONNECTED, FAILED, DISCONNECTED.

2.3.1.2 Get call status API Success Scenario

Sample URL	https://localhost/chpas/devicecontrol/videocall/status/1234
Sample HTTP Request Xml	N/A

Sample HTTP Response Xml	`<Response>`
	`<Status>`
	`<StatusCode>200</StatusCode>`
	`<StatusMessage>Success</StatusMessage >`
	`</Status>`
	`<ConferenceCallInfo>`
	`<VideoCallId>1234</VideoCallId>`
	`<ConferenceType>P2P</ConferenceType>`
	`<HasActiveConferenceSessions>true</HasActiveConferenceSessions>`
	`<ConferenceSessions>`
	`<ConferenceSession>`
	`<ConferenceSessionId>128</ConferenceSessionId>`
	`<SessionId>device_ctrl_sessn:jab1</SessionId>`
	`<StartTime>1366623462549</StartTime>`
	`<EndpointId>jab1</EndpointId>`
	`<ConferenceSessionStatus>Connected`
	`</ConferenceSessionStatus>`
	`</ConferenceSession>`
	`<ConferenceSession>`
	`<ConferenceSessionId>129</ConferenceSessionId>`
	`<SessionId>device_ctrl_sessn:jab2</SessionId>`
	`<StartTime>1366623468073</StartTime>`
	`<EndpointId>jab2</EndpointId>`
	`<ConferenceSessionStatus>Connected`
	`</ConferenceSessionStatus>`
	`</ConferenceSession>`
	`</ConferenceSessions>`
	`</ConferenceCallInfo>`
	`</Response>`

Get call status API Failure Scenario

Scenario	Sample HTTP URL	Sample HTTP Response xml
Invalid video-call Id	https://localhost/ chpas/devicecontrol/video-call/status/`invalid-videocallID-1`	<Response> <Status> <StatusCode>5100</StatusCode> <StatusMessage>Video Call ID is invalid or Video call has ended. </StatusMessage> </Status> </Response>

2.3.2 Get all call status API

2.3.2.1 Get all call status API Definition

URL pattern:/chpas/devicecontrol/videocall/status

URL path variables:N/A

HTTP method: GET

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition: None

HTTP Response body definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success
Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
PayloadList	List	No. Available if at least one call is present.	List of on-going video calls.
ConferenceCallInfo	Object	No. Available if at least one call is present.	Xml representation of call information.
ConferenceCallInfo : VideoCallId	String	Yes	UUID to uniquely identify the video call
ConferenceCallInfo: ConferenceType	String	Yes	The type of conference for this video call.

ConferenceCallInfo : HasActive ConferenceSessions	Boolean	Yes	Indicates if there is at least one active session in this call.
ConferenceCallInfo: ConferenceSessions	List	No. Available if at least one session is alive.	List of objects representing the sessions in this call.
ConferenceSessions: ConferenceSession	Object	No. Available if at least one session is alive.	Xml representation of Call Session Info
ConferenceSession: ConferenceSessionId	Integer	No	ID for a given session of the call.
ConferenceSession: StartTime	Integer	No	Time when this session of the call started. Represented in milliseconds from epoch.
ConferenceSession: EndpointId	String	No	The endpoint id from which this session is started.
ConferenceSession: ConferenceSessionStatus	String	No	Status of this video call session. IDLE, CONNECTED, FAILED, DISCONNECTED.

2.3.2.2 Get all call status API Success Scenarios

Sample URL	https://localhost/chpas/devicecontrol/videocall/status
Sample HTTP Request Xml	N/A

Sample HTTP Response Xml	

```
<Response>
  <Status>
    <StatusCode>200</StatusCode>
    <StatusMessage>Success</StatusMessage >
  </Status>
  <PayloadList>
<ConferenceCallInfo>
<VideoCallId>1234</VideoCallId>
<ConferenceType>P2P</ConferenceType>
<HasActiveConferenceSessions>true
           </HasActiveConferenceSessions>
<ConferenceSessions>
<ConferenceSession>
<ConferenceSessionId>130
                  </ConferenceSessionId>
<SessionId>device_ctrl_sessn:jab1
                  </SessionId>
<StartTime>1366624687232</StartTime>
<EndpointId>jab1</EndpointId>
<ConferenceSessionStatus>Active
                  </ConferenceSessionStatus>
</ConferenceSession>
</ConferenceSessions>
</ConferenceCallInfo>
<ConferenceCallInfo>
<VideoCallId>2343</VideoCallId>
<ConferenceType>P2P</ConferenceType>
<HasActiveConferenceSessions>true
           </HasActiveConferenceSessions>
<ConferenceSessions>
<ConferenceSession>
<ConferenceSessionId>131
                  </ConferenceSessionId>
<SessionId>device_ctrl_sessn:jab2
                   </SessionId>
<StartTime>1366624700659</StartTime>
<EndpointId>jab2</EndpointId>
<ConferenceSessionStatus>Active
                  </ConferenceSessionStatus>
</ConferenceSession>
</ConferenceSessions>
</ConferenceCallInfo>
  </PayloadList>
</Response>
```

Get all call status API Failure Scenarios:

None.

2.4 Camera Preset APIs

Camera preset APIs are used to pre-define camera positions. These presets can be used to position the camera during a video call.

2.4.1 List camera preset API

List information about available camera preset.

2.4.1.1 List camera preset API Definition

URL pattern:https://{IP}:{port}/chpas/camera/preset/list/{endpoint_id}/{preset_id}

URL path variables:

- endpoint_id - ID of the endpoint that has a camera and supports camera positioning api's
- presetId - Filter on a specified preset. Acceptable value of PresetId is from 1 to 35

HTTP method: GET

HTTP headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition: None

HTTP Response body definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success
Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
Command	Object	Yes	Holds result of the Camera Preset APIs
Command :PresetListResult	Object	Yes	Holds result for List Camera Preset API

Command :PresetListResult: Preset	Object	Yes	List information about the available preset
Command :PresetListResult: Preset: CameraId	String	Yes	ID of the camera
Command :PresetListResult: Preset: Name	String	Yes	The name of the new camera preset. It will be used in the list returned by the Camera Preset List API call.
Command :PresetListResult: Preset: ListPosition	String	Yes	The position in the list returned by the List Camera Preset API call
Command :PresetListResult: Preset: Pan	String	Yes	Value of Pan for the available preset.
Command :PresetListResult: Preset: Tilt	String	Yes	Value of Tilt for the available preset.
Command :PresetListResult: Preset: Zoom	String	Yes	Value of Zoom for the available preset.
Command :PresetListResult: Preset: Focus	String	Yes	Value of Focus for the available preset.

2.4.1.2 List camera preset API Success Scenario

Sample URL	https://localhost/chpas/devicecontrol/camera/preset/list/C-20/5
Sample HTTP Request Xml	N/A

Sample HTTP Response Xml	`<Response>`
	`  <Status>`
	`  <StatusCode>200</StatusCode>`
	`  <StatusMessage>Success</StatusMessage>`
	`  </Status>`
	`  <Command>`
	`  <PresetListResult status="OK" item="1">`
	`  <Preset item="5">`
	`  <CameraId>1</CameraId>`
	`  <Name>xyz</Name>`
	`  <ListPosition>8</ListPosition>`
	`  <Pan>500</Pan>`
	`  <Tilt>100</Tilt>`
	`  <Zoom>0</Zoom>`
	`  <Focus>4576</Focus>`
	`  </Preset>`
	`  </PresetListResult>`
	`  </Command>`
	`</Response>`

2.4.1.3 List camera preset API Failure Scenario

Scenario	Sample URL	Sample HTTP Response xml
Empty End-point Id	https://{IP}:{port}/chpas/camera/pre-set/list//{ preset_id}	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLE-MENTED</StatusMessage > </Status> </Response>

Invalid End-point Id	https://{IP}:{port}/chpas/camera/preset/list/{invalid_endpoint_id}/{preset_id}	<Response> <Status> <StatusCode>404</StatusCode> <StatusMessage>NOT FOUND: Endpoint {invalid_endpoint_id} not found</Status-Message > </Status> </Response>
Invalid Camera PresetId	https://{IP}:{port}/chpas/camera/preset/list/{endpoint_id}/{invalid_preset_id}	<Response> <Status> <StatusCode>425</StatusCode> <StatusMessage>INVALID_INPUT : {Invalid PresetId}</StatusMessage > </Status> </Response>
Endpoint does not support Camera Preset API	https://{IP}:{port}/chpas/camera/preset/list/{endpoint_id_of_unsupported_endpoint}/{preset_id}	<Response> <Status> <StatusCode>5101</StatusCode> <StatusMessage>Endpoint does not support Camera API : Endpoint[{endpoint_id_of_unsupported_endpoint}] </StatusMessage > </Status> </Response>

2.4.2 List all camera preset API

List information about all available camera presets.

2.4.2.1 List all camera preset API Definition

URL pattern:https://{IP}:{port}/chpas/camera/preset/listAll/{endpoint_id}

URL path variables:

- endpoint_id - ID of the endpoint that has a camera and supports camera positioning api's

HTTP method: GET

HTTP headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition: None

HTTP Response body definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success
Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
Command	Object	Yes	Holds result of the Camera Preset APIs
Command :PresetListResult	Object	Yes	Holds result for List All Camera Preset API
Command :PresetListResult: Preset	List	Yes	List of all the available preset
Command :PresetListResult: Preset:CameraId	String	Yes	ID of the camera for the preset.
Command :PresetListResult: Preset:CameraId	String	Yes	Name of the preset

2.4.2.2 List all camera preset API Success Scenario

Sample URL	https://localhost/chpas/devicecontrol/camera/preset/listAll/C-20
Sample HTTP Request Xml	N/A

Sample HTTP Response Xml	`<Response>`
	`<Status>`
	`  <StatusCode>200</StatusCode>`
	`  <StatusMessage>Success</StatusMessage>`
	`</Status>`
	`<Command>`
	`  <PresetListResult status="OK" item="1">`
	`  <Preset item="20">`
	`  <CameraId>1</CameraId>`
	`  <Name></Name>`
	`  </Preset>`
	`  <Preset item="8">`
	`  <CameraId>1</CameraId>`
	`  <Name></Name>`
	`  </Preset>`
	`  <Preset item="6">`
	`  <CameraId>1</CameraId>`
	`  <Name></Name>`
	`  </Preset>`
	`  <Preset item="4">`
	`  <CameraId>1</CameraId>`
	`  <Name></Name>`
	`  </Preset>`
	`  <Preset item="2">`
	`  <CameraId>1</CameraId>`
	`  <Name></Name>`
	`  </Preset>`
	`  <Preset item="3">`
	`  <CameraId>1</CameraId>`
	`  <Name>pan left</Name>`
	`  </Preset>`
	`  <Preset item="7">`
	`  <CameraId>1</CameraId>`
	`  <Name>pan right</Name>`
	`  </Preset>`
	`  <Preset item="5">`
	`  <CameraId>1</CameraId>`
	`  <Name>Pan left zoom-10</Name>`
	`  </Preset>`
	`  </PresetListResult>`
	`</Command>`
	`</Response>`

2.4.2.3 List all camera API Failure Scenario

Scenario	Sample URL	Sample HTTP Response xml
Empty End-point Id	https://{IP}:{port}/chpas/camera/pre-set/listAll	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLE-MENTED</StatusMessage > </Status> </Response>
Invalid End-point Id	https://{IP}:{port}/chpas/camera/pre-set/listAll/{invalid_endpoint_id}	<Response> <Status> <StatusCode>404</StatusCode> <StatusMessage>NOT FOUND: End-point[{invalid endpointid}] not found</Sta-tusMessage > </Status> </Response>
Endpoint does not support Camera Pre-set API	https://{IP}:{port}/chpas/camera/pre-set/listAll/{endpoint_id_of_unsupported_endpoint}	<Response> <Status> <StatusCode>5101</StatusCode> <StatusMessage>Endpoint does not sup-port Camera API : End-point[{endpoint_id_of_unsupported_endpoint }] </StatusMessage > </Status> </Response>

2.4.3 Store camera preset API

2.4.3.1 Store camera preset API Definition

URL pattern:https://{IP}:{port}/chpas/camera/preset/{endpoint_id}

URL path variables:

- endpoint_id - ID of the endpoint that has a camera and supports camera positioning api's

HTTP method: POST

HTTP headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition:

Name	Type	Mandatory	Description
CameraPresetRequest	Object	Yes	Status information for the API call.
CameraPresetRequest: PresetId	String	Yes	The ID of this camera preset. Acceptable value of PresetId is from 1 to 35.
CameraPresetRequest: CameraId	String	Yes	ID of the camera to store the position of
CameraPresetRequest: ListPosition	String	No	The new camera preset's position in the list returned by the Camera Preset List api call. Acceptable value of List Position is from 1 to 35.
CameraPresetRequest: Name	String	No	The name of the new camera preset. It will be used in the list returned by the Camera Preset List api call. Name can be a String with length of 0 to 255.

HTTP Response body definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success

Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
Command	Object	Yes	Holds result of the Camera Preset APIs
Command : PresetStoreResult	Object	Yes	Holds result for Store Camera Preset API
Command : PresetStoreResult: Reason	Object	Yes	Holds the Id of the Preset Stored by the Store Camera Preset API call.

2.4.3.2 Store camera preset API Success Scenario

Sample URL	https://localhost/chpas/devicecontrol/camera/preset/C-20
Sample HTTP Request Xml	<CameraPresetRequest> <PresetId>14</PresetId> <CameraId>1</CameraId> <ListPosition>1</ListPosition> <Name>Pan Left</Name> </CameraPresetRequest>
Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage> </Status> <Command> <PresetStoreResult status="OK" item="1">> <Reason>PresetId: 14</Reason> </PresetStoreResult> </Command> </Response>

2.4.3.3 Store camera preset API Failure Scenario

Scenario	Sample URL	Sample HTTP Response xml
Empty End-point Id	https://{IP}:{port}/chpas/camera/preset/	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLE-MENTED</StatusMessage > </Status> </Response>
Invalid End-point Id	https://{IP}:{port}/chpas/camera/pre-set/{invalid_endpoint_id}	<Response> <Status> <StatusCode>404</StatusCode> <StatusMessage>NOT FOUND: End-point[{invalid_endpoint_id}] not found</StatusMessage > </Status> </Response>
Endpoint does not support Camera Pre-set API	https://{IP}:{port}/chpas/camera/pre-set/{endpoint_id_of_unsupported_endpoin t}	<Response> <Status> <StatusCode>5101</StatusCode> <StatusMessage>Endpoint does not sup-port Camera API : End-point[{endpointId_of_unsupported_endpoi nt}] </StatusMessage > </Status> </Response>

2.4.4 Edit camera preset API

Edit a stored preset. You can change the name of the camera preset and its position in the list that is returned by the List Camera Preset API call.

2.4.4.1 Edit camera preset API Definition

URL pattern:https://{IP}:{port}/chpas/camera/preset/{endpoint_id}

URL path variables:

- endpoint_id - ID of the endpoint that has a camera and supports camera positioning API.

HTTP method: PUT

HTTP headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition:

Name	Type	Mandatory	Description
CameraPresetRequest	Object	Yes	Status information for the API call.
CameraPresetRequest: PresetId	String	Yes	The ID of this camera preset. Acceptable value of PresetId is from 1 to 35.
CameraPresetRequest: CameraId	String	Yes	ID of the camera you want to edit.
CameraPresetRequest: ListPosition	String	No	The position in the list returned by the List Camera Preset API. Acceptable value of List Position is from 1 to 35.
CameraPresetRequest: Name	String	No	The name of the camera preset. It will be used in the list returned by the List Camera Preset API. Name can be a String with length of 0 to 255.

HTTP response body definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.

Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success
Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
Command	Object	Yes	Holds result of the Camera Preset APIs
Command : PresetEditResult	Object	Yes	Holds result for Edit Camera Preset API

2.4.4.2 Edit camera preset API Success Scenario

Sample URL	https://localhost/chpas/devicecontrol/camera/preset/C-20
Sample HTTP Request Xml	<CameraPresetRequest> <PresetId>14</PresetId> <CameraId>1</CameraId> <ListPosition>1</ListPosition> <Name>Pan Right</Name> </CameraPresetRequest>
Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage> </Status> <Command> <PresetEditResult status="OK" item="1"/> </Command> </Response>

2.4.4.3 Edit camera preset API Failure Scenario

Scenario	Sample URL	Sample HTTP Response xml

Empty End-point Id	https://{IP}:{port}/chpas/camera/preset/	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLE-MENTED</StatusMessage > </Status> </Response>
Invalid End-point Id	https://{IP}:{port}/chpas/camera/pre-set/{invalid_endpoint_id}	<Response> <Status> <StatusCode>404</StatusCode> <StatusMessage>NOT FOUND: End-point[{invalid_endpoint_id}] not found</StatusMessage > </Status> </Response>
Endpoint does not support Camera Pre-set API	https://{IP}:{port}/chpas/camera/pre-set/{endpoint_id_of_unsupported_endpoin t}	<Response> <Status> <StatusCode>5101</StatusCode> <StatusMessage>Endpoint does not sup-port Camera API : End-point[{endpoint_id_of_unsupported_endp oint}] </StatusMessage > </Status> </Response>

2.4.5 Remove camera preset API

Remove a camera preset.

2.4.5.1 Remove camera preset API Definition

URL pattern:https://{IP}:{port}/chpas/camera/preset/{endpoint_id}/{preset_id}

URL path variables:

- endpoint_id - ID of the endpoint that has a camera and supports camera positioning api's
- presetId - ID of the camera preset you want to remove. Acceptable value of preset_id is from 1 to 35.

HTTP method: DELETE

HTTP headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition: None

HTTP Response body definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success
Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
Command	Object	Yes	Holds result of the Camera Preset APIs
Command : PresetRemoveResult	Object	Yes	Holds result for Remove Camera Preset API

2.4.5.2 Remove camera preset API Success Scenario

Sample URL	https://localhost/chpas/devicecontrol/camera/preset/C-20/14
Sample HTTP Request Xml	None
Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage> </Status> <Command> <PresetRemoveResult status="OK" item="1"/> </Command> </Response>

2.4.5.3 Remove camera preset API Failure Scenario

Scenario	Sample URL	Sample HTTP Response xml
Empty End-point Id	https://{IP}:{port}/chpas/camera/preset/	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLE-MENTED</StatusMessage > </Status> </Response>
Invalid End-point Id	https://{IP}:{port}/chpas/camera/pre-set/{invalid_endpoint_id}	<Response> <Status> <StatusCode>404</StatusCode> <StatusMessage>NOT FOUND: End-point[invalid_endpoint_id}] not found</Sta-tusMessage > </Status> </Response>
Empty Pre-setId	https://{IP}:{port}/chpas/camera/pre-set/{endpoint_id}	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLE-MENTED</StatusMessage > </Status> </Response>
Invalid Pre-setId	https://{IP}:{port}/chpas/camera/pre-set/{endpoint_id}/{invalid_preset_id}	<Response> <Status> <StatusCode>425</StatusCode> <StatusMessage>INVALID_INPUT : Invalid Value for parameter PresetId {invalid_presetid}</StatusMessage > </Status> </Response>

Endpoint does not support Camera Pre-set API	https://{IP}:{port}/chpas/camera/pre-set/{endpoint_id_of_unsupported_endpoint}/{preset_id}	<Response> <Status> <StatusCode>5101</StatusCode> <StatusMessage>Endpoint does not support Camera API : Endpoint[{endpoint_id_of_unsupported_endpoint}] </StatusMessage > </Status> </Response>

2.4.6 Activate camera preset API

Activate one of the camera presets.

2.4.6.1 Activate camera preset API Definition

URL pattern:https://{IP}:{port}/chpas/camera/preset/activate/{endpoint_id}/{preset_id}

URL path variables:

- endpoint_id - ID of the endpoint that has a camera and supports camera positioning api's
- presetId - ID of the camera preset you want to activate. Acceptable value of preset_id is from 1 to 35.

HTTP method: GET

HTTP headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition: None

HTTP Response body definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success

Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
Command	Object	Yes	Holds result of the Camera Preset APIs
Command : `PresetActivateResult`	Object	Yes	Holds result for Activate Camera Preset API

2.4.6.2 Activate camera preset API Success Scenario

Sample URL	https://localhost/chpas/devicecontrol/camera/preset/activate/C-20/5
Sample HTTP Request Xml	None
Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage> </Status> <Command> <PresetActivateResult status="OK" item="1"/> </Command> </Response>

2.4.6.3 Activate camera preset API Failure Scenario

Scenario	Sample URL	Sample HTTP Response xml

Empty End-point Id	https://{IP}:{port}/chpas/camera/preset/activate//{preset_id}	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLE-MENTED</StatusMessage > </Status> </Response>
Invalid End-point Id	https://{IP}:{port}/chpas/camera/pre-set/acti-vate/{invalid_endpoint_id}/{preset_id}	<Response> <Status> <StatusCode>404</StatusCode> <StatusMessage>NOT FOUND: End-point[{invalid_endpoint_id}] not found</StatusMessage > </Status> </Response>
Empty Pre-setId	https://{IP}:{port}/chpas/camera/preset/activate /{endpoint_id}	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLE-MENTED</StatusMessage > </Status> </Response>
Invalid Pre-setId	https://{IP}:{port}/chpas/camera/preset/activate /{endpoint_id}/{invalid_preset_id}	<Response> <Status> <StatusCode>425</StatusCode> <StatusMessage>INVALID_INPUT : Invalid Value for parameter PresetId {invalid_presetid}</StatusMessage > </Status> </Response>

Endpoint does not support Camera Pre-set API	https://{IP}:{port}/chpas/camera/pre-set/{endpoint_id_of_unsupported_endpoint}/ activate /{preset_id}	<Response> <Status> <StatusCode>5101</StatusCode> <StatusMessage>Endpoint does not support Camera API : Endpoint[{endpoint_id_of_unsupported_endpoint}] </StatusMessage > </Status> </Response>

2.5 Camera Positioning APIs

Position the camera by defining the pan, tilt, zoom and focus parameters. If the camera is placed in a daisy chain you need to know the CameraId for the camera you want to address.

2.5.1 Change camera position API

Position the camera by defining the pan, tilt, zoom and focus parameters. If the camera is placed in a daisy chain you need to know the CameraId for the camera you want to address.

2.5.1.1 Change camera position API Definition

URL pattern:https://{IP}:{port}/chpas/camera/changePosition/{endpoint_id}

URL path variables:

- endpoint_id - ID of the endpoint that has a camera and supports camera positioning api's

HTTP method: POST

HTTP headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition:

Name	Type	Mandatory	Description
ChangeCameraPosition-Req	Object	Yes	Status information for the API call.

CameraPresetRequest: PresetId	String	Yes	The ID of this camera preset. If not set explicitly, the codec will assign a preset ID automatically. Acceptable value of PresetId is from 1 to 35.
ChangeCameraPosition-Req: CameraId	String	Yes	ID of the camera to store the position of
CameraPresetRequest: Pan	String	No	Set the value for Pan. Acceptable value of pan is from -65535 to 65535.
CameraPresetRequest: Tilt	String	No	Set the value for Tilt. Acceptable value of tilt is from -65535 to 65535.
CameraPresetRequest: Zoom	String	No	Set the value for Zoom. Acceptable value of zoom is from 0 to 65535.
CameraPresetRequest: Focus	String	No	Set the value for Focus. Acceptable value of focus is from 0 to 65535.

HTTP Response body definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success
Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
Command	Object	Yes	Holds result of the Camera Preset APIs

Command : `Camera-PositionSetResult`	Object	Yes	Holds result for Change Camera Position API

2.5.1.2 Change camera position API Success Scenario

Sample URL	https://localhost/chpas/devicecontrol/camera/changePosition/C-20

Sample HTTP Request Xml	`<ChangeCameraPositionReq>`
	`<CameraId>1</CameraId>`
	`<Pan>375</Pan>`
	`<Tilt>100</Tilt>`
	`<Zoom>0</Zoom>`
	`<Focus>0</Focus>`
	`</ChangeCameraPositionReq>`

Sample HTTP Response Xml	`<Response>`
	`<Status>`
	`  <StatusCode>200</StatusCode>`
	`  <StatusMessage>Success</StatusMessage>`
	`</Status>`
	`<Command>`
	`  <CameraPositionSetResult status="OK" item="1"/>`
	`</Command>`
	`</Response>`

2.5.1.3 Change camera position API Failure Scenarios

Scenario	Sample URL	Sample HTTP Response xml
Empty End-point Id	https://{IP}:{port}/chpas/camera/changePosition	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLE-MENTED</StatusMessage > </Status> </Response>

Invalid End-point Id	https://{IP}:{port}/chpas/camera/changePosition/{invalid_endpoint_id}	<Response> <Status> <StatusCode>404</StatusCode> <StatusMessage>NOT FOUND: End-point[{invalid_endpoint_id}] not found</StatusMessage > </Status> </Response>
Endpoint does not support Camera Pre-set API	https://{IP}:{port}/chpas/camera/preset/changePosi-tion/{endpoint_id_of_unsupported_endpoint}	<Response> <Status> <StatusCode>5101</StatusCode> <StatusMessage>Endpoint does not sup-port Camera API : End-point[{endpoint_id_of_unsupported_endpoint}] does not support Pan, Tilt, Zoom and Focus Capabilities</StatusMessage > </Status> </Response>

2.5.2 Pan camera right API

Pan the camera toward the right. If the camera is placed in a daisy chain you need to know the CameraId for the camera that you want to address.

2.5.2.1 Pan camera right API Definition

URL pattern:https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/pan/right/{endpoint_id}

URL path variables:

- endpoint_id - ID of the endpoint that has a camera and supports camera positioning api's

HTTP method: POST

HTTP headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition:

Name	Type	Mandatory	Description
ChangeCameraPosition-Req	Object	Yes	Status information for the API call.
ChangeCameraPosition-Req: CameraId	String	Yes	ID of the camera.

HTTP Response body definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success
Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
Command	Object	Yes	Holds result of the Camera APIs
Command : `CameraPositionSetResult`	Object	Yes	Holds result for pan right API

2.5.2.2 Pan camera right API Success Scenario

Sample URL	https:// {IP}:{port}/chpas/devicecontrol/endpoint/camera/pan/right/C-20

Sample HTTP Request Xml	`<ChangeCameraPositionReq>`
	`<CameraId>1</CameraId>`
	`</ChangeCameraPositionReq>`

Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage> </Status> <Command> <CameraPositionSetResult status="OK" item="1"/> </Command> </Response>

2.5.2.3 Pan camera right API Failure Scenarios

Scenario	Sample URL	Sample HTTP Response xml
Empty Endpoint Id	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/pan/right	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLEMENTED</StatusMessage > </Status> </Response>
Invalid Endpoint Id	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/pan/right/{invalid_endpoint_id}	<Response> <Status> <StatusCode>404</StatusCode> <StatusMessage>NOT FOUND: Endpoint[{invalid_endpoint_id}] not found</StatusMessage > </Status> </Response>

Endpoint does not support Camera Preset API	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/pan/right/{endpoint_id_of_unsupported_endpoint}	<Response> <Status> <StatusCode>5101</StatusCode> <StatusMessage>Endpoint does not support Camera API : Endpoint[{endpoint_id_of_unsupported_endpoint}] does not support Pan, Tilt, Zoom and Focus Capabilities</StatusMessage > </Status> </Response>

2.5.3 Pan camera left API

Pan the camera toward the left. If the camera is placed in a daisy chain you need to know the CameraId for the camera that you want to address.

2.5.3.1 Pan camera left API Definition

URL pattern:https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/pan/left/{endpoint_id}

URL path variables:

- endpoint_id - ID of the endpoint that has a camera and supports camera positioning api's

HTTP method: POST

HTTP headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition:

Name	Type	Mandatory	Description
ChangeCameraPosition-Req	Object	Yes	Status information for the API call.
ChangeCameraPosition-Req: CameraId	String	Yes	ID of the camera.

HTTP Response body definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success
Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
Command	Object	Yes	Holds result of the Camera APIs
Command : `CameraPositionSetResult`	Object	Yes	Holds result for pan left API

2.5.3.2 Pan camera left API Success Scenario

Sample URL	https:// {IP}:{port}/chpas/devicecontrol/endpoint/camera/pan/left/C-20

Sample HTTP Request Xml	`<ChangeCameraPositionReq>`
	`<CameraId>1</CameraId>`
	`</ChangeCameraPositionReq>`

Sample HTTP Response Xml	`<Response>`
	`<Status>`
	`  <StatusCode>200</StatusCode>`
	`  <StatusMessage>Success</StatusMessage>`
	`</Status>`
	`<Command>`
	`  <CameraPositionSetResult status="OK" item="1"/>`
	`</Command>`
	`</Response>`

2.5.3.3 Pan camera left API Failure Scenarios

Scenario	Sample URL	Sample HTTP Response xml
Empty Endpoint Id	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/pan/left	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLEMENTED</StatusMessage > </Status> </Response>
Invalid Endpoint Id	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/pan/left/{invalid_endpoint_id}	<Response> <Status> <StatusCode>404</StatusCode> <StatusMessage>NOT FOUND: Endpoint[{invalid_endpoint_id}] not found</StatusMessage > </Status> </Response>
Endpoint does not support Camera Preset API	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/pan/left/{endpoint_id_of_unsupported_endpoint}	<Response> <Status> <StatusCode>5101</StatusCode> <StatusMessage>Endpoint does not support Camera API : Endpoint[{endpoint_id_of_unsupported_endpoint}] does not support Pan, Tilt, Zoom and Focus Capabilities</StatusMessage > </Status> </Response>

2.5.4 Tilt camera up API

Tilt the camera up. If the camera is placed in a daisy chain you need to know the CameraId for the camera that you want to address.

2.5.4.1 Tilt camera up API Definition

URL pattern:https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/tilt/up/{endpoint_id}

URL path variables:

- endpoint_id - ID of the endpoint that has a camera and supports camera positioning api's

HTTP method: POST

HTTP headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition:

Name	Type	Mandatory	Description
ChangeCameraPosition-Req	Object	Yes	Status information for the API call.
ChangeCameraPosition-Req: CameraId	String	Yes	ID of the camera.

HTTP Response body definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success
Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
Command	Object	Yes	Holds result of the Camera APIs
Command : `CameraPositionSetResult`	Object	Yes	Holds result for tilt up API

2.5.4.2 Tilt camera up API Success Scenario

Sample URL	https:// {IP}:{port}/chpas/devicecontrol/endpoint/camera/tilt/up/C-20
Sample HTTP Request Xml	<ChangeCameraPositionReq> <CameraId>1</CameraId> </ChangeCameraPositionReq>
Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage> </Status> <Command> <CameraPositionSetResult status="OK" item="1"/> </Command> </Response>

2.5.4.3 Tilt camera up API Failure Scenarios

Scenario	Sample URL	Sample HTTP Response xml
Empty End-point Id	https://{IP}:{port}/chpas/devicecon-trol/endpoint/camera/tilt/up/	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLE-MENTED</StatusMessage > </Status> </Response>
Invalid End-point Id	https://{IP}:{port}/chpas/devicecon-trol/endpoint/cam-era/tilt/up/{invalid_endpoint_id}	<Response> <Status> <StatusCode>404</StatusCode> <StatusMessage>NOT FOUND: End-point[{invalid_endpoint_id}] not found</StatusMessage > </Status> </Response>

Endpoint does not support Camera Preset API	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/tilt/up/{endpoint_id_of_unsupported_endpoint}	<Response> <Status> <StatusCode>5101</StatusCode> <StatusMessage>Endpoint does not support Camera API : Endpoint[{endpoint_id_of_unsupported_endpoint}] does not support Pan, Tilt, Zoom and Focus Capabilities</StatusMessage > </Status> </Response>

2.5.5 Tilt camera down API

Tilt the camera down. If the camera is placed in a daisy chain you need to know the CameraId for the camera that you want to address.

2.5.5.1 Tilt camera down API Definition

URL pattern:https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/tilt/down/{endpoint_id}

URL path variables:

- endpoint_id - ID of the endpoint that has a camera and supports camera positioning api's

HTTP method: POST

HTTP headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition:

Name	Type	Mandatory	Description
ChangeCameraPosition-Req	Object	Yes	Status information for the API call.
ChangeCameraPosition-Req: CameraId	String	Yes	ID of the camera.

HTTP Response body definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success
Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
Command	Object	Yes	Holds result of the Camera APIs
Command : `CameraPositionSetResult`	Object	Yes	Holds result for tilt down API

2.5.5.2 Tilt camera down API Success Scenario

Sample URL	https:// {IP}:{port}/chpas/devicecontrol/endpoint/camera/tilt/down/C-20

Sample HTTP Request Xml	`<ChangeCameraPositionReq>`
	`<CameraId>1</CameraId>`
	`</ChangeCameraPositionReq>`

Sample HTTP Response Xml	`<Response>`
	`<Status>`
	`  <StatusCode>200</StatusCode>`
	`  <StatusMessage>Success</StatusMessage>`
	`</Status>`
	`<Command>`
	`  <CameraPositionSetResult status="OK" item="1"/>`
	`</Command>`
	`</Response>`

2.5.5.3 Tilt camera down API Failure Scenarios

Scenario	Sample URL	Sample HTTP Response xml
Empty Endpoint Id	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/tilt/down/	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLEMENTED</StatusMessage > </Status> </Response>
Invalid Endpoint Id	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/tilt/down/{invalid_endpoint_id}	<Response> <Status> <StatusCode>404</StatusCode> <StatusMessage>NOT FOUND: Endpoint[{invalid_endpoint_id}] not found</StatusMessage > </Status> </Response>
Endpoint does not support Camera Preset API	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/tilt/down/{endpoint_id_of_unsupported_endpoint}	<Response> <Status> <StatusCode>5101</StatusCode> <StatusMessage>Endpoint does not support Camera API : Endpoint[{endpoint_id_of_unsupported_endpoint}] does not support Pan, Tilt, Zoom and Focus Capabilities</StatusMessage > </Status> </Response>

2.5.6 Zoom in the camera API

Zoom in the camera. If the camera is placed in a daisy chain you need to know the CameraId for the camera that you want to address.

2.5.6.1 Zoom in the camera API Definition

URL pattern:https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/zoom/in/{endpoint_id}

URL path variables:

- endpoint_id - ID of the endpoint that has a camera and supports camera positioning api's

HTTP method: POST

HTTP headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition:

Name	Type	Mandatory	Description
ChangeCameraPosition-Req	Object	Yes	Status information for the API call.
ChangeCameraPosition-Req: CameraId	String	Yes	ID of the camera.

HTTP Response body definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success
Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
Command	Object	Yes	Holds result of the Camera APIs
Command : `CameraPositionSetResult`	Object	Yes	Holds result for zoom in API

2.5.6.2 Zoom camera in API Success Scenario

Sample URL	https:// {IP}:{port}/chpas/devicecontrol/endpoint/camera/zoom/in/C-20
Sample HTTP Request Xml	<ChangeCameraPositionReq> <CameraId>1</CameraId> </ChangeCameraPositionReq>
Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage> </Status> <Command> <CameraPositionSetResult status="OK" item="1"/> </Command> </Response>

2.5.6.3 Zoom camera in API Failure Scenarios

Scenario	Sample URL	Sample HTTP Response xml
Empty Endpoint Id	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/zoom/in/	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLEMENTED</StatusMessage > </Status> </Response>
Invalid Endpoint Id	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/zoom/in/{invalid_endpoint_id}	<Response> <Status> <StatusCode>404</StatusCode> <StatusMessage>NOT FOUND: Endpoint[{invalid_endpoint_id}] not found</StatusMessage > </Status> </Response>

Endpoint does not support Camera Preset API	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/zoom/in/{endpoint_id_of_unsupported_endpoint}	<Response> <Status> <StatusCode>5101</StatusCode> <StatusMessage>Endpoint does not support Camera API : Endpoint[{endpoint_id_of_unsupported_endpoint}] does not support Pan, Tilt, Zoom and Focus Capabilities</StatusMessage > </Status> </Response>

2.5.7 Zoom out the camera API

Zoom out the camera. If the camera is placed in a daisy chain you need to know the CameraId for the camera that you want to address.

2.5.7.1 Zoom out the camera API Definition

URL pattern:https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/zoom/out/{endpoint_id}

URL path variables:

- endpoint_id - ID of the endpoint that has a camera and supports camera positioning api's

HTTP method: POST

HTTP headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition:

Name	Type	Mandatory	Description
ChangeCameraPosition-Req	Object	Yes	Status information for the API call.
ChangeCameraPosition-Req: CameraId	String	Yes	ID of the camera.

HTTP Response body definition:

Name	Type	Mandatory	Description

Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success
Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
Command	Object	Yes	Holds result of the Camera APIs
Command : `CameraPositionSetResult`	Object	Yes	Holds result for zoom out API

2.5.7.2 Zoom camera out API Success Scenario

Sample URL	https:// {IP}:{port}/chpas/devicecontrol/endpoint/camera/zoom/out/C-20

Sample HTTP Request Xml	`<ChangeCameraPositionReq>`
	`<CameraId>1</CameraId>`
	`</ChangeCameraPositionReq>`

Sample HTTP Response Xml	`<Response>`
	`<Status>`
	`  <StatusCode>200</StatusCode>`
	`  <StatusMessage>Success</StatusMessage>`
	`</Status>`
	`<Command>`
	`  <CameraPositionSetResult status="OK" item="1"/>`
	`</Command>`
	`</Response>`

2.5.7.3 Zoom camera out API Failure Scenarios

Scenario	Sample URL	Sample HTTP Response xml

Empty Endpoint Id	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/zoom/out/	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLEMENTED</StatusMessage > </Status> </Response>
Invalid Endpoint Id	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/zoom/out/{invalid_endpoint_id}	<Response> <Status> <StatusCode>404</StatusCode> <StatusMessage>NOT FOUND: Endpoint[{invalid_endpoint_id}] not found</StatusMessage > </Status> </Response>
Endpoint does not support Camera Preset API	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/zoom/out/{endpoint_id_of_unsupported_endpoint}	<Response> <Status> <StatusCode>5101</StatusCode> <StatusMessage>Endpoint does not support Camera API : Endpoint[{endpoint_id_of_unsupported_endpoint}] does not support Pan, Tilt, Zoom and Focus Capabilities</StatusMessage > </Status> </Response>

2.5.8 Trigger autofocus API

Trigger the autofocus functionality. If the camera is placed in a daisy chain you need to know the CameraId for the camera that you want to address.

2.5.8.1 Trigger autofocus API Definition

URL pattern:https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/triggerAutofocus/{endpoint_id}

URL path variables:

- endpoint_id - ID of the endpoint that has a camera and supports camera positioning apis

HTTP method: POST

HTTP headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition:

Name	Type	Mandatory	Description
ChangeCameraPosition-Req	Object	Yes	Status information for the API call.
ChangeCameraPosition-Req: CameraId	String	Yes	ID of the camera.

HTTP Response body definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success
Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
Command	Object	Yes	Holds result of the Camera APIs
Command : `TriggeAutofocus-Result`	Object	Yes	Holds result for triggeAutofocus API

2.5.8.2 Trigger autofocus API Success Scenario

Sample URL	https:// {IP}:{port}/chpas/devicecontrol/endpoint/camera/triggerAutofocus/C-20

Sample HTTP Request Xml	<ChangeCameraPositionReq> <CameraId>1</CameraId> </ChangeCameraPositionReq>
Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage> </Status> <Command> <TriggerAutofocusResult status="OK" item="1"/> </Command> </Response>

2.5.8.3 Trigger autofocus API Failure Scenarios

Scenario	Sample URL	Sample HTTP Response xml
Empty End-point Id	https://{IP}:{port}/chpas/devicecon-trol/endpoint/camera/triggerAutofocus/	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLE-MENTED</StatusMessage > </Status> </Response>
Invalid End-point Id	https://{IP}:{port}/chpas/devicecon-trol/endpoint/camera/triggerAutofo-cus/{invalid_endpoint_id}	<Response> <Status> <StatusCode>404</StatusCode> <StatusMessage>NOT FOUND: End-point[{invalid_endpoint_id}] not found</StatusMessage > </Status> </Response>

Endpoint does not support Camera Preset API	https://{IP}:{port}/chpas/devicecontrol/endpoint/camera/zoom/in/{endpoint_id_of_unsupported_endpoint}	<Response> <Status> <StatusCode>5101</StatusCode> <StatusMessage>Endpoint does not support Camera API : Endpoint[{endpoint_id_of_unsupported_endpoint}] does not support Pan, Tilt, Zoom and Focus Capabilities</StatusMessage > </Status> </Response>

2.6 Utility APIs

2.6.1 Get endpoint capabilities

Get the endpoint capabilities. If the camera is placed in a daisy chain you need to know the CameraId for the camera you want to address.

2.6.1.1 Get endpoint capabilites API Definition

URL pattern:https://{IP}:{port}/chpas/devicecontrol/endpoint/capabilities/{endpoint_id}

URL path variables:

- endpoint_id - ID of the endpoint that has a camera and supports camera positioning apis

HTTP method: POST

HTTP headers:

- chp_auth_component: callcontrol-service
- chp_auth_token: <Auth token provided>

HTTP Request body definition:

Name	Type	Mandatory	Description
ChangeCameraPosition-Req	Object	Yes	Status information for the API call.
ChangeCameraPosition-Req: CameraId	String	Yes	ID of the camera.

HTTP Response body definition:

Name	Type	Mandatory	Description
Status	Object	Yes	Status information for the API call.
Status: StatusCode	Integer	Yes	Status code for the API call. 200 implies success
Status: StatusMessage	String	Yes	Status message for the API call. If there are any errors, this field provides a description of the error.
Command	Object	Yes	Holds result of the Camera APIs
Command : `EndpointCapabilitiesResult`	Object	Yes	Holds result for getEndpointCapabilities API

2.6.1.2 Get endpoint capabilities API Success Scenario

Sample URL	https:// {IP}:{port}/chpas/devicecontrol/endpoint/capabilities/C-20
Sample HTTP Request Xml	None
Sample HTTP Response Xml	<Response> <Status> <StatusCode>200</StatusCode> <StatusMessage>Success</StatusMessage> </Status> <Command> <EndpointCapabilitiesResult status="OK" item="1"/> </Command> </Response>

2.6.1.3 Zoom camera in API Failure Scenarios

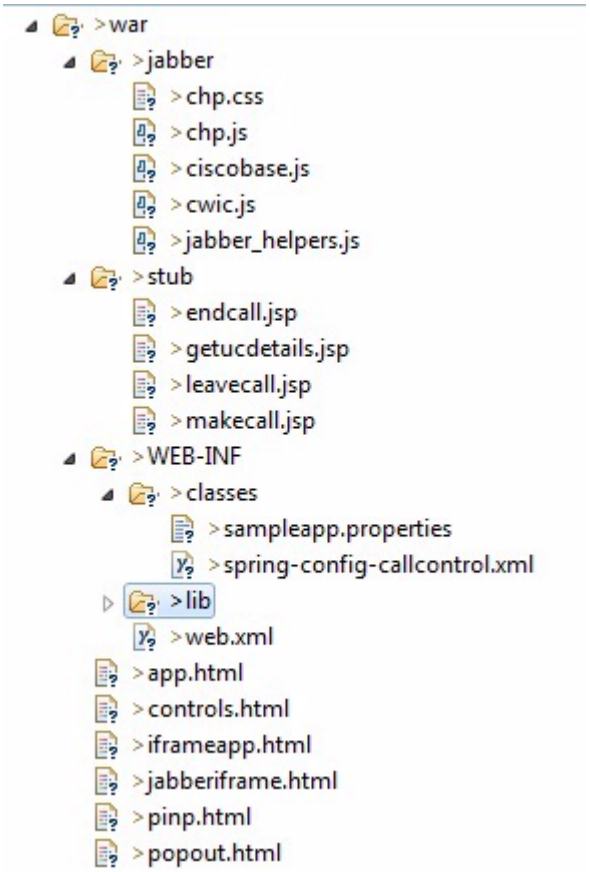
Scenario	Sample URL	Sample HTTP Response xml
Empty Endpoint Id	https://{IP}:{port}/chpas/devicecontrol/endpoint/capabilitiess/	<Response> <Status> <StatusCode>501</StatusCode> <StatusMessage>NOT IMPLEMENTED</StatusMessage > </Status> </Response>
Invalid Endpoint Id	https://{IP}:{port}/chpas/devicecontrol/endpoint/capabilities/{invalid_endpoint_id}	<Response> <Status> <StatusCode>404</StatusCode> <StatusMessage>NOT FOUND: Endpoint[{invalid_endpoint_id}] not found</StatusMessage > </Status> </Response>
Endpoint does not support Camera Preset API	https://{IP}:{port}/chpas/devicecontrol/endpoint/capabilities/{endpoint_id_of_unsupported_endpoint}	<Response> <Status> <StatusCode>5101</StatusCode> <StatusMessage>Endpoint does not support Camera API : Endpoint[{endpoint_id_of_unsupported_endpoint}] does not support Pan, Tilt, Zoom and Focus Capabilities</StatusMessage > </Status> </Response>

The Call Control Application

Revised: November 20, 2013, OL-30301-01

This chapter describes how to use a Cisco-supplied application bundle to access the APIs. The following figure shows the folder hierarchy of the bundle:

Figure 3-1 Application Bundle File Hierarchy

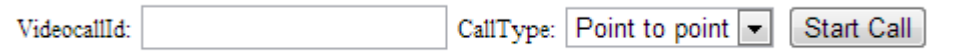


Step 1 Deploy the war distribution in a web container (for ex. Tomcat) and access the following URL in a web browser:

http://<IPAddress>:<port>/callcontrolapp/app.html?password=<deviceControlPassword>&endpointid=<endpointId>

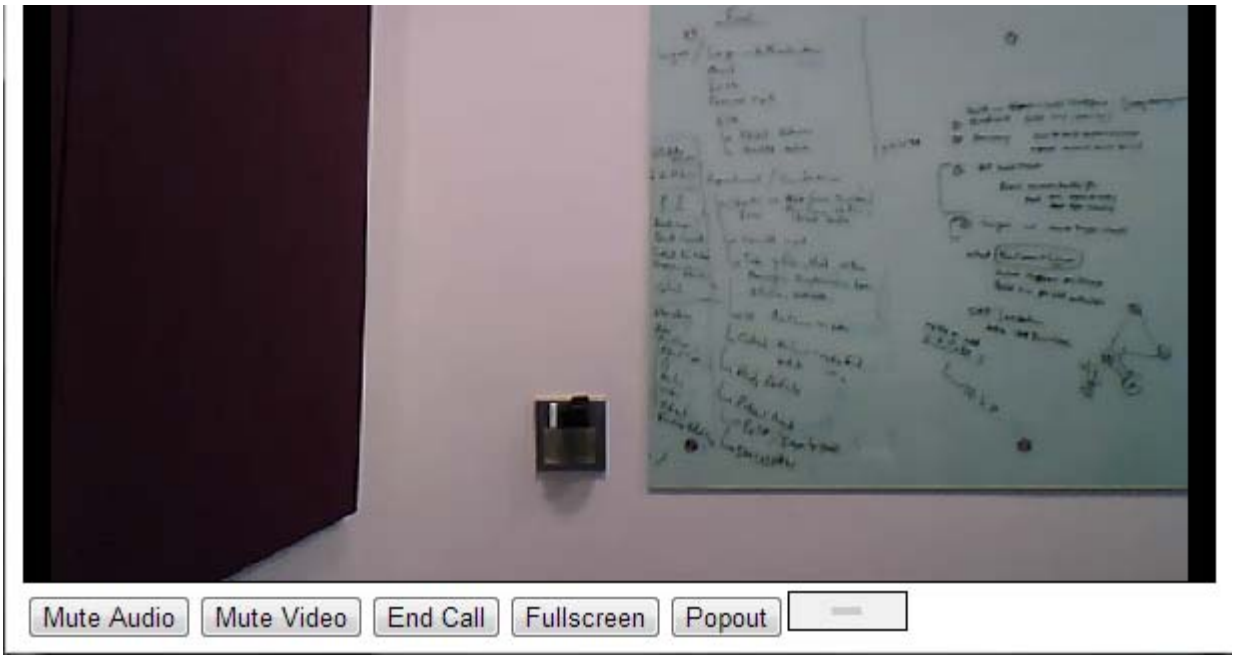
Step 2 Enter the following details in the page loaded and click **Start Call**.

Figure 3-2 Start Call UI.



Step 3 Repeat the above steps on the second endpoint and enter the same videocallid. A video call will be started along with video control buttons as shown below.

Figure 3-3 Video Call Window



3.1 Call Control Java Client (For REST API calls)

If the customer environment is Java based, they can make use of this Java client to call the Collaboration REST APIs. Please refer to the class "CollaborationApiClient" in the Javadoc for the list of methods to invoke the call control APIs. For the javadoc version of this API guide, please contact your Cisco representative.

3.2 IFrame-Based Applications

If the customer does not want to include code to invoke call control APIs, then the application has a provision to include the functionality as an iframe.

Deploy the war distribution in a web container (such asTomcat) and access the following URLs as an iframe from the client application.

Step 1 To start the call, use the following URL:

http://<IP>:<port>/callcontrolapp/jabberiframe.html?method=makeVideoCall&password=<devicecontrolpassword>&endpointid=<endpointid>&videocallid=<videocallid>&calltype=<calltype>

Step 2 To end the call, use the following URL:

http://<IP>:<port>/callcontrolapp/jabberiframe.html?method=endVideoCall&password=<devicecontrolpassword>&endpointid=<endpointid>&videocallid=<videocallid>

Step 3 Upon starting the call, note the following:

- If the endpoint is of type Jabber SDK, the div containing the iframe will be populated with the conference video.
- If it is a hard endpoint, the call will start on the hard endpoint.
- If the endpoint is Jabber Video, then the call will start on the Jabber Video application.

Common Code Definitions

Revised: November 20, 2013, OL-30301-01

A.1 Message Response Success Codes

Table A-1 describes the information that must be gathered to complete the installation.

Table A-1 ISuccess Message

Status Code	Name	Description
200	Successful	Everything is working properly. Look for API specific response data for details.

A.2 Message Response Failure Codes

Each error message has a status code and a status message. The status codes are described below. The status message will provide a description of the context for the error.

Table A-2 Failure Messages

Status Code	Status Message / Name	Description
401	Unauthorized	Invalid authentication parameters.

425	Invalid Input	Generic error code for any invalid input parameters. Example, invalid camera preset id, camera id, etc. Specific error codes mentioned below is preferred over this generic code. This code might also get thrown if any of the mandatory fields are missing.
421	Invalid session id	The given endpoint is not in any call session.
422	Invalid endpoint Id	The endpoint ID provided is not valid – either for the entire system or in the given context.
500	Internal Error	Any unexpected internal error.
501	Not Implemented	The request API is not implemented. Most likely an invalid REST call.
503	Service Unavailable	Unable to handle request due to transient conditions or maintenance activities.
524	Timed Out	Timed out waiting for response from internal components or other delegators.
532	Not Interoperable	One or more endpoints in the list of endpoints in the request are not interoperable with other endpoints.
536	Max participants exceeded	The maximum participants for the given conference type has exceeded. The value, for example, is 2 for a P2P call.
547	Maximum End points exceeded	The conference type cannot accommodate more endpoints into it. Happens typically when a third user is trying to join a P2P call.
552	Check out for the license resource failed	There are not enough licenses available to do the requested operation
5100	Video call is not available	The video call id provided is either invalid or the video call has terminated.

API Availability

Revised: November 20, 2013, OL-30301-01

Refer to Table B-1 to for feature availablity.

Table B-1 APIs and corresponding version availability

API	Extended Care Release
Make video call	1.0 onwards
End video call	1.0 onwards
Start or join video call	1.0 onwards
Join existing video call	1.0 onwards
Leave video call	1.0 onwards
End all video calls	1.0 onwards
Get call status	1.0 onwards
Get all call statuses	1.0 onwards
List camera preset	1.0 onwards
List all camera presets	1.0 onwards
Store camera preset	1.0 onwards
Edit camera preset	1.0 onwards
Remove camera preset	1.0 onwards
Activate camera preset	1.0 onwards
Change camera position	1.0 onwards

Software Compatibility

Revised: November 20, 2013, OL-30301-01

Browser Compatibility

The following table (continued on the next page) lists the browsers and their level of support in Cisco Extended Care. Fully supported means the relevant components have been completely tested on these browsers and are certified to work. Compatible means the relevant browser components have been functionally tested on these browsers at a high level, but exhaustive testing has not been done.

Table C-1 Browser Compatibility by Operating System and User Type

Patient			
Platform	**Browser Name**	**Browser Version**	**Support Level**
Windows 7	Internet Explorer	8	Fully Supported
	Internet Explorer	9	Fully Supported
	Chrome	26	Fully Supported
	Firefox	23.0.1	Fully Supported
Windows 8	Internet Explorer	10	Fully Supported
	Chrome	26	Fully Supported
Mac OS 10.8.4	Safari	6.03	Fully Supported
	Chrome	26	Compatible
iPad iOS 6.1.3	Safari	6.03	Fully Supported
	Chrome	26	Not supported

Provider			
Windows 7	Internet Explorer	8	Fully Supported
	Internet Explorer	9	Fully Supported
	Chrome	26	Fully Supported
	Firefox	23.0.1	Fully Supported
Windows 8	Internet Explorer	10	Fully Supported
Mac OS 10.8.4	Safari	6.0.3	Fully Supported
iPad iOS 6.1.3	Safari	6.03	Fully Supported
	Chrome	26	Not supported
Administrator			
Windows 7	Internet Explorer	8	Fully Supported
	Internet Explorer	9	Fully Supported
	Firefox	23.0.1	Fully Supported

Revised: November 20, 2013, OL-30301-01

A

API Application Programming Interface.

H

HTTP Hypertext Transfer Protocol.

M

MP Multi-party call.

P

P2P Peer to peer call.

R

REST Representational State Transfer.

R

S

Switchable P2P Multi-party call using EX90 as the bridge.

U

UC Unified Communications