



Cisco Unified Contact Center Express Developer Guide, Release 12.5(1)

First Published: 2020-01-31

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

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 NON-INFRINGEMENT 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.

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.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

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

The documentation set for this product strives to use bias-free language. For purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on standards documentation, or language that is used by a referenced third-party product.

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: <https://www.cisco.com/c/en/us/about/legal/trademarks.html>. 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. (1721R)

© 2000–2020 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Preface	xiii
Change History	xiii
About This Guide	xiv
Conventions	xiv
Related Documents	xvi
Documentation and Support	xvi
Documentation Feedback	xvi

CHAPTER 1

Introduction 1

Introduction	1
--------------	----------

CHAPTER 2

System Configuration APIs 5

Advanced Supervisor Capabilities	6
Data Structure for Advanced Supervisor Capabilities	6
Get Advanced Supervisor Capabilities	6
Add, Remove, or Modify Advanced Supervisor Capabilities	7
Data Structure for Outbound Campaign Management for Supervisor	7
Get Outbound Campaigns Managed by Supervisor	7
Add, Remove, or Update Outbound Campaigns Managed by Supervisor	8
Data Structure for Application Management for Supervisor	8
Get Applications Managed by Supervisor	9
Add or Update Applications Managed by Supervisor	9
Agent Statistics	10
Data Structure for Agent Statistics	10
Get Agent Statistics	10
Application	10

- Data Structure for Application 11
- Data Structure for ICMPostRouting 11
- Data Structure for ICMTranslationRouting 12
- Data Structure for ScriptApplication 12
- Get List of Applications 13
- Get Application 14
- Get Application with Script Parameters 16
- Create Application 17
- Modify Application 18
- Delete Application 20
- Assign Prompts to Application 21
- Get Prompts of Application 21
- Area Code 23
 - Data Structure for Area Code 23
 - Get List of Area Codes 23
 - Get Area Code 24
 - Create Area Code 24
 - Modify Area Code 25
 - Delete Area Code 26
- Calendar 26
 - Data Structure for Calendar 26
 - Data Structure for Businessdays 28
 - Data Structure for Holidays 28
 - Data Structure for Specialdays 29
 - Create Calendar 29
 - Get Calendar 32
 - Get Calendar with Associations 35
 - Get List of Calendars 35
 - Get List of Calendars with Associations 39
 - Get List of Calendars with Summary 40
 - Get Calendars Managed by Supervisors 41
 - Modify Calendar 42
 - Modify Calendars Managed by Supervisors 43
 - Delete Calendar 43

Call Control Group	44
Data Structure for Call Control Group	44
Data Structure for Port Group Configuration Type	45
Simple Data Structure Used in Call Control Group	51
Data Structure for Call Control Group Job Status	53
Get List of Call Control Groups	53
Get Call Control Group	54
Create Call Control Group	57
Modify Call Control Group	60
Delete Call Control Group	63
Check Status of POST/PUT/DELETE	63
Campaign	65
Data Structure for Campaign	65
Common Data Structure for IVR-Based and Agent-Based Campaigns	66
Common Data Structure for Outbound IVR - Progressive and Predictive Campaigns	68
Data Structure for Outbound IVR - Predictive Campaign	68
Data Structure for Preview Campaign	69
Common Data Structure for Outbound Agent - Progressive and Predictive Campaigns	69
Data Structure for Outbound Agent - Predictive Campaign	70
Get List of Campaigns	71
Get Campaign	72
Create Campaign	74
Modify Campaign	79
Delete Campaign	84
Modify Campaign State	85
Campaign Contacts	85
Data Structure for Campaign Contacts	85
Get List of Pending Contacts for Campaign	86
Get List of Pending Contacts for Campaign with Specific Call Status	87
Add Contacts to Campaign	88
Get Campaign Contacts Import Status	90
Delete All Contacts from Campaign	91
Channel Parameters	91
Data Structure for Channel Parameters	92

Get Channel Parameters Information	93
Put Channel Parameters Information	93
Channel Provider	94
Data Structure for Channel Provider	94
Get List of Channel Providers	96
Get List of Channel Providers by Type	97
Get a Channel Provider	97
Create a Channel Provider	98
Modify a Channel Provider	99
Delete a Channel Provider	99
Chat Widget	100
Data Structure for Chat Widget	100
Problem Statement CSQ Data Structure	102
Bubble Style Data Structure	102
Bubble Message Data Structure	104
Calendar Details Data Structure	107
Calendar Messages Data Structure	107
Day of Week Data Structure	108
Create Chat Widget	110
Get List of Chat Widgets	113
Get Chat Widget (XML)	113
Get Chat Widget (HTML)	116
Modify Chat Widget	121
Delete Chat Widget	125
Contact Service Queue	125
CSQ Data Structure	126
Email CSQ Data Structure	128
Skill or Resource Group Data Structure	129
Skill Competency Data Structure	130
Skill Details Data Structure	130
Resource Group Lists Data Structure	131
Resource Order Data Structure	131
Get List of CSQs	131
Get CSQ	137

Create CSQ	139
Modify CSQ	140
Delete CSQ	142
Data Source	143
Data Structure for Data Source	143
Get List of Data Sources	144
Get Data Source	144
Create Data Source	145
Modify Data Source	146
Delete Data Source	146
Test Connection for Data Source	147
DB Purge Schedule Configuration	147
Data Structure for DB Purge Schedule Configuration	147
Get Purge Configuration Information	148
Update Purge Configuration Information	148
DB Purge Now Configuration	149
Data Structure for DB Purge Now Configuration	149
Update Purge Now Configuration	149
DB Purge Configuration Status	150
Data Structure for DB Purge Configuration Status	150
Get Purge Configuration Status	150
Dialog Group	151
Data Structure for Dialog Group	151
Get List of Dialog Groups	152
Get Dialog Group	153
Create Dialog Group	153
Modify Dialog Group	154
Delete Dialog Group	154
Email SMTP Server Configuration	155
Datastructure for Email SMTP Server Configuration	155
Get Email SMTP Server Configuration	155
Create Email SMTP Server Configuration	156
Modify Email SMTP Server Configuration	156
Delete Email SMTP Server Configuration	156

Files (Script, Prompt, Grammar, and Document)	157
Data Structure for Files	157
Create a New Language for Prompt	158
Create a New Language for Document	159
Create a New Language for Grammar	160
Create a New Folder	160
Create a New Folder for Script	160
Create a New Folder for Prompt	161
Create a New Folder for Document	161
Create a New Folder for Grammar	162
Upload Files to a Temporary Folder	163
Upload Files to a Temporary Folder for Script	163
Upload Files to a Temporary Folder for Prompt	164
Upload Files to a Temporary Folder for Document	164
Upload Files to a Temporary Folder for Grammar	164
Store Files in Database from a Temporary folder	165
Store Files in Database from a Temporary Folder for Script	165
Store Files in Database from a Temporary Folder for Prompt	165
Store Files in Database from a Temporary Folder for Document	166
Store Files in Database from a Temporary Folder for Grammar	167
Get List for Script Files and Folder	167
Get List for Prompt Files and Folder	168
Get List for Grammar Files and Folder	168
Get List for Document Files and Folder	169
Get Script Files and Folder	169
Get Prompt Files and Folder	170
Get Grammar Files and Folder	171
Get Document Files and Folder	171
Get FreeSpace and UsedSpace for Script/Prompt/Grammar/Document	172
Download Script/Prompt	172
Modify Script Folder Name	173
Modify File	173
Modify Prompt Folder Name or Language Name	174
Modify Grammar Folder Name or Language Name	175

Modify Document Folder Name or Language Name	176
Delete Script Files and Folder	177
Delete Prompt Files and Folder	177
Delete Grammar Files and Folder	178
Delete Document Files and Folder	178
HTTPS Trigger	179
Data Structure for HTTPS Trigger	179
Get List of HTTPS Triggers	180
Get HTTPS Trigger	181
Create HTTPS Trigger	181
Modify HTTPS Trigger	182
Delete HTTPS Trigger	182
Outbound General Configuration	183
Data Structure for Outbound General Configuration	183
Get List of Skills for Outbound General Configuration	185
Modify Outbound General Configuration	186
System Configuration	187
Data Structure for System Configuration	187
Get System Configuration	187
Update System Configuration	187
Prompt Files Path	188
Data Structure for Prompt Files Path	188
GET Prompt Files Path	188
Proxy Configuration	189
Data Structure for Proxy Configuration	189
Get Proxy Configuration	190
Update Proxy Configuration	190
Resource	191
Data Structure for Resource	191
Get List of Resources	193
Get List of Resources by CSQ	194
Get Resource with CSQs	195
Get Modified Information of Resources by CSQ	196
Add Resources	197

Get Resource	199
Modify Resource	200
Delete Resource	201
Schedule Resource	202
Get Schedules	202
Create Schedules	203
Edit Schedules	205
Delete Schedules	207
Resource Bundle of Locale	208
Data Structure for Resource Bundle of Locale	209
Get Resource Bundle of Locale	209
Resource Group	210
Data Structure for Resource Group	210
Get List of Resource Groups	210
Get Resource Group	211
Create Resource Group	211
Modify Resource Group	212
Delete Resource Group	212
Skill	213
Data Structure for Skill	213
Get List of Skills	213
Get Skill	214
Create Skill	214
Modify Skill	215
Delete Skill	215
Sip Gateway Configuration	216
Data Structure for Sip Gateway Configurations	216
Get List of Sip Gateway Configurations	218
Update SIP Gateway Configurations	218
Standalone Cisco Unified Intelligence Center	219
Data Structure for Standalone CUIC	219
Configuration of Standalone CUIC	220
Sample Input XML - Modify CUIC Configuration (PUT)	220
Team	220

Data Structure for Team	221
Get List of Teams	222
Get Team	222
Create Team	223
Modify Team	223
Delete Team	224
Trigger	224
Trigger Data Structure	224
Simple Data Structure and Elements Used in Trigger	227
Complex Data Structure Defined for forwardBusy	228
Get list of triggers	229
Get a Trigger by Directory Number	229
Create a New Trigger	229
Modify a Trigger by ID	230
Delete a Trigger by ID	231
TTS Provider	231
Data Structure for TTS Provider	231
Get list of TTS Providers	232
Get a TTS Provider	232
Create TTS Provider	233
Modify TTS Provider	233
Delete TTS Provider	234
TTS Server	234
Data Structure for TTS Server	234
Get list of TTS Servers	235
Get a TTS Server	236
Create TTS Server	237
Modify TTS Server	237
Delete TTS Server	238
WrapupCategory	238
Data Structure for WrapupCategory	238
Get list of WrapupCategory	239
Get WrapupCategory	241
Create WrapupCategory	241

Modify WrapupCategory 242

Delete WrapupCategory 243

APPENDIX A **Cisco Finesse Desktop and Configuration APIs** 245

APPENDIX B **Cisco Customer Collaboration Platform APIs** 247



Preface

- [Change History](#), on page xiii
- [About This Guide](#), on page xiv
- [Conventions](#), on page xiv
- [Related Documents](#), on page xvi
- [Documentation and Support](#), on page xvi
- [Documentation Feedback](#), on page xvi

Change History

This table lists changes made to this guide. Most recent changes appear at the top.

Change	See	Date
Updated the Unified CCX REST APIs for supporting Experience Management post-call surveys. The code samples have been updated.	System Configuration APIs >> Application >> Data Structure for ScriptApplication System Configuration APIs >> Application >> Get Application System Configuration APIs >> Application >> Get Application with Script Parameters System Configuration APIs >> Application >> Create Application System Configuration APIs >> Application >> Modify Application	May 2020

Change	See	Date
Initial Release of Document for Release 12.5(1)		January 2020
Cisco SocialMiner (SM) has been renamed as Customer Collaboration Platform (CCP).	Appendix B >> Cisco Customer Collaboration Platform APIs	
Added Schedule Resource APIs.	<ul style="list-style-type: none"> • Resource>>Schedule Resource>>Get Schedules • Resource>>Schedule Resource>>Create Schedules • Resource>>Schedule Resource>>Edit Schedules • Resource>>Schedule Resource>>Delete Schedules 	
Added Resource APIs.	<ul style="list-style-type: none"> • Resource>>Add Resource • Resource>>Delete Resource 	
Added Assign Prompts to Application.	Application>>Assign Prompts to Application	
Added Get Prompts of Application.	Application>>Get Prompts of Application	

About This Guide

This document introduces REST Application Program Interface (API) use and conventions for Unified Contact Center Express (Unified CCX) and provides details about each REST API.

Conventions

This manual uses the following conventions.

Convention	Description
boldface font	<p>Boldface font is used to indicate commands, such as user entries, keys, buttons, and folder and submenu names. For example:</p> <ul style="list-style-type: none"> • Choose Edit > Find • Click Finish.

Convention	Description
<i>italic font</i>	<p>Italic font is used to indicate the following:</p> <ul style="list-style-type: none"> • To introduce a new term. Example: A <i>skill group</i> is a collection of agents who share similar skills. • For emphasis. Example: <i>Do not</i> use the numerical naming convention. • An argument for which you must supply values. Example: IF (<i>condition, true-value, false-value</i>) • A book title. Example: See the <i>Cisco Unified Contact Center Express Installation Guide</i>.
window font	<p>Window font, such as Courier, is used for the following:</p> <ul style="list-style-type: none"> • Text as it appears in code or information that the system displays. Example: <code><html><title> Cisco Systems, Inc. </title></html></code> • File names. Example: <code>tserver.properties.</code> • Directory paths. Example: <code>C:\Program Files\Adobe</code>
string	<p>Nonquoted sets of characters (strings) appear in regular font. Do not use quotation marks around a string or the string will include the quotation marks.</p>
[]	<p>Optional elements appear in square brackets.</p>
{ x y z }	<p>Alternative keywords are grouped in braces and separated by vertical bars.</p>
[x y z]	<p>Optional alternative keywords are grouped in brackets and separated by vertical bars.</p>
< >	<p>Angle brackets are used to indicate the following:</p> <ul style="list-style-type: none"> • For arguments where the context does not allow italic, such as ASCII output. • A character string that the user enters but that does not appear on the window such as a password.

Convention	Description
^	The key labeled Control is represented in screen displays by the symbol ^. For example, the screen instruction to hold down the Control key while you press the D key appears as ^D.

Related Documents

Document or Resource	Link
Cisco Unified Contact Center Express Documentation Guide	https://www.cisco.com/en/US/products/sw/custcosw/ps1846/products_documentation_roadmaps_list.html
Cisco Unified CCX documentation	https://www.cisco.com/en/US/products/sw/custcosw/ps1846/tsd_products_support_series_home.html
Cisco Unified Intelligence Center documentation	https://www.cisco.com/en/US/products/ps9755/tsd_products_support_series_home.html
Cisco Finesse documentation	https://www.cisco.com/en/US/products/ps11324/tsd_products_support_series_home.html
Cisco Customer Collaboration Platform documentation Note From Unified CCX Release 12.5(1), CCP documents are available in the Cisco Unified CCX documentation folder.	https://www.cisco.com/en/US/products/sw/custcosw/ps1846/tsd_products_support_series_home.html
Cisco Unified CCX Virtualization Information	https://www.cisco.com/c/dam/en/us/td/docs/voice_ip_comm/uc_system/virtualization/virtualization-cisco-unified-contact-center-express.html
Cisco Unified CCX Compatibility Information	https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-express/products-device-support-tables-list.html

Documentation and Support

To download documentation, submit a service request, and find additional information, see *What's New in Cisco Product Documentation* at <https://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html>.

Documentation Feedback

To provide your feedback for this document, send an email to:

contactcenterproducts_docfeedback@cisco.com



CHAPTER 1

Introduction

- [Introduction, on page 1](#)

Introduction

Cisco Unified Contact Center Express Configuration APIs offer REST APIs for configuration of Unified Contact Center Express.

Each object supports five methods (except where documented otherwise) that can be invoked on the URL that is associated with the object.

The structure of the URL is

`<protocol>://<uccx-server>/adminapi/<object>/<instanceId>`.

URL Structure	Description
<code><protocol></code>	HTTP and HTTPS are the supported protocols.
<code><uccx-server></code>	Routable address (name or IP address) of the Unified CCX Server.
<code><object></code>	One of the supported objects.
<code><instanceId></code>	Optional component needed for GET, DELETE, PUT methods referring to a specific instance of the object.

Supported Operations

Operation	HTTPS Method	URL	Description	Input	Output
LIST	GET	<code><protocol>://<uccx-server>/adminapi/<object></code>	List all the instance of this object	None	List of objects
GET	GET	<code><protocol>://<uccx-server>/adminapi/<object>/<instanceId></code>	Get details of the instances specified by instanceId.	None	Single object instance

Operation	HTTPS Method	URL	Description	Input	Output
CREATE	POST	<protocol>://<uccx-server>/adminapi/<object>	Create a new instance of the object.	New instance data	URL of the new object
MODIFY	PUT	<protocol>://<uccx-server>/adminapi/<object>	Modify the instance specified by instanceId.	Modified instance data	None
DELETE	DELETE	<protocol>://<uccx-server>/adminapi/<object>/<instanceId>	Delete the instance specified by instanceId.	None	None

All methods support Application/XML and Application/JSON as input MIME types.



Note You can modify the attributes that you edit in AppAdmin through Rest APIs.

Security

Users with only administration credentials on Unified CCX can invoke the methods. However, if application management capability is assigned to supervisors, the specific APIs and methods can be invoked by supervisors also. The API uses Basic Authentication mechanism of HTTPS and the credentials (user ID and password) must be sent with every request in the Authorization header.



Note The specific API sections mention methods that can be invoked by supervisors.

If an API operation fails, a detailed error is returned in the HTTPS response message body. The error, in XML format, is as follows:

```
<ApiErrors>
  <ApiError>
    <ErrorType>type</ErrorType>
    <ErrorMessage>message</ErrorMessage>
    <ErrorData>data</ErrorData>
  </ApiError>
</ApiErrors>
```

Unsupported Unicode Noncharacters

REST API does not allow unicode noncharacters in the request because security polices restrict their use.

When you enter unicode noncharacters in the request, an error is returned in the HTTPS response message body. The error, in XML format, is as follows:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
  <ApiErrors>
```

```
<ApiError>
  <ErrorType>InvalidInput</ErrorType>
  <ErrorMessage>The attempted action violates security policies because of invalid
input.</ErrorMessage>
  <ErrorData></ErrorData>
</ApiError>
</ApiErrors>
```

Failure Response Codes

The failure response codes for Unified CCX REST APIs are listed here:

- 207 Multi Status
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found
- 405 Method Not Allowed
- 409 Conflict
- 412 Precondition Failed
- 424 Failed Dependency
- 500 Internal Server Error
- 503 Service Unavailable



CHAPTER 2

System Configuration APIs

- [Advanced Supervisor Capabilities](#), on page 6
- [Agent Statistics](#), on page 10
- [Application](#), on page 10
- [Area Code](#), on page 23
- [Calendar](#) , on page 26
- [Call Control Group](#), on page 44
- [Campaign](#), on page 65
- [Campaign Contacts](#), on page 85
- [Channel Parameters](#), on page 91
- [Channel Provider](#), on page 94
- [Chat Widget](#), on page 100
- [Contact Service Queue](#), on page 125
- [Data Source](#), on page 143
- [DB Purge Schedule Configuration](#), on page 147
- [DB Purge Now Configuration](#), on page 149
- [DB Purge Configuration Status](#), on page 150
- [Dialog Group](#), on page 151
- [Email SMTP Server Configuration](#), on page 155
- [Files \(Script, Prompt, Grammar, and Document\)](#), on page 157
- [HTTPS Trigger](#), on page 179
- [Outbound General Configuration](#), on page 183
- [System Configuration](#), on page 187
- [Prompt Files Path](#), on page 188
- [Proxy Configuration](#), on page 189
- [Resource](#), on page 191
- [Resource Bundle of Locale](#), on page 208
- [Resource Group](#), on page 210
- [Skill](#), on page 213
- [Sip Gateway Configuration](#), on page 216
- [Standalone Cisco Unified Intelligence Center](#) , on page 219
- [Team](#), on page 220
- [Trigger](#), on page 224
- [TTS Provider](#), on page 231

- [TTS Server](#), on page 234
- [WrapupCategory](#), on page 238

Advanced Supervisor Capabilities

The Unified CCX Administrator can now delegate few of the administrative capabilities to Supervisors to manage queues, outbound campaigns, and applications in addition to the existing Supervisor privileges.

Data Structure for Advanced Supervisor Capabilities

Attribute Name	Base Type	Restrictions	Description
capability	string	maxOccurs = 4	Supported list of capabilities: <ul style="list-style-type: none"> • QUEUE_MGMT • CAMPAIGN_MGMT • APPLICATION_MGMT • CALENDAR_MGMT

Get Advanced Supervisor Capabilities

URI	https://<server>/adminapi/resource/{userID}/capabilities
Example URI	https://uccx-server/adminapi/resource/rick/capabilities
HTTPS Method	GET
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 401, 404, 500
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<resource><refurl>https://<uccx-server>/adminapi/resource/rick</refurl></resource>
<capabilities>
  <capabilityList>
    <capability>CAMPAIGN_MGMT</capability>
    <capability>APPLICATION_MGMT</capability>
    <capability>QUEUE_MGMT</capability>
    <capability>CALENDAR_MGMT</capability>
  </capabilityList>
</capabilities>
```


Add, Remove, or Modify Advanced Supervisor Capabilities

URI	https://<server>/adminapi/resource/{userID}/capabilities
Example URI	https://uccx-server/adminapi/resource/rick/capabilities
HTTPS Method	PUT
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 401, 405, 500
Error Codes	—

Sample Input XML—Add/Modify the advanced supervisor capabilities (PUT)

```
<?xml version="1.0" encoding="UTF-8"?>
<resource><refurl>https://<uccx-server>/adminapi/resource/rick</refurl></resource>
<capabilities>
  <capabilityList>
    <capability>CAMPAIGN_MGMT</capability>
    <capability>APPLICATION_MGMT</capability>
    <capability>QUEUE_MGMT</capability>
    <capability>CALENDAR_MGMT</capability>
  </capabilityList>
</capabilities>
```

Data Structure for Outbound Campaign Management for Supervisor

The Unified CCX system allows retrieving and assigning outbound campaigns that can be managed by the Supervisor.

Attribute Name	Base Type	Restrictions	Description
resource	nameUriPair	minOccurs=0	Resource to which the campaigns are assigned.
campaignList	nameUriPair	minOccurs=0	List of campaigns managed by supervisor.

Get Outbound Campaigns Managed by Supervisor

URI	https://<server>/adminapi/resource/{userID}/capabilities/campaigns
Example URI	https://uccx-server/adminapi/resource/rick/capabilities/campaigns
HTTPS Method	GET
Accept	Application/XML, Application/JSON
HTTPS Success Code	200

HTTPS Failure Codes	400, 401, 403, 404, 500
Error Codes	—

Success Response

```
<campaigns>
  <resource name="Rick Barrows">
    <refURL>https://<uccx-server>/adminapi/resource/rick</refURL></resource>
  <campaignList>
    <campaign name="Preview">
      <refURL>https://<uccx-server>/adminapi/campaign/1</refURL></campaign>
    </campaignList>
  </campaigns>
```

Add, Remove, or Update Outbound Campaigns Managed by Supervisor

URI	https://<server>/adminapi/resource/{userID}/capabilities/campaigns
Example URI	https://uccx-server/adminapi/resource/rick/capabilities/campaigns
HTTPS Method	PUT
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 401, 403, 404, 500
Error Codes	—

Sample Input XML—Add or Modify the Outbound Campaign Capability

```
<campaigns>
  <resource name="Rick Barrows">
    <refURL>https://<uccx-server>/adminapi/resource/rick</refURL>
  </resource>
  <campaignList>
    <campaign name="Preview">
      <refURL>https://<uccx-server>/adminapi/campaign/1</refURL>
    </campaign>
  </campaignList>
</campaigns>
```

Data Structure for Application Management for Supervisor

The Unified CCX system allows retrieving and assigning applications that can be managed by the Supervisor.

Attribute Name	Base Type	Restrictions	Description
resource	nameUriPair	minOccurs=0	Resource to which the applications are assigned.
applicationList	nameUriPair	minOccurs=0	List of applications managed by Supervisor.

Get Applications Managed by Supervisor

URI	https://<server>/adminapi/resource/{userID}/capabilities/applications
Example URI	https://uccx-server/adminapi/resource/rick/capabilities/applications
HTTPS Method	GET
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 401, 403, 404, 500
Error Codes	—

Success Response

```
<applications>
  <resource name="Rick Barrows">
    <refURL>https://<ccx-server>/adminapi/resource/rick</refURL></resource>
  <applicationList>
    <application name="CreditApp">
      <refURL>https://<ccx-server>/adminapi/application/CreditApp</refURL>
    </application>
  </applicationList>
</applications>
```

Add or Update Applications Managed by Supervisor

URI	https://<server>/adminapi/resource/{userID}/capabilities/applications
Example URI	https://uccx-server/adminapi/resource/rick/capabilities/applications
HTTPS Method	PUT
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 401, 403, 404, 500
Error Codes	—

Sample Input XML—Add/Modify the application management capabilities (PUT)

```
<applications>
  <resource name="Rick Barrows">
    <refURL>https://<ccx-server>/adminapi/resource/rick</refURL></resource>
  <applicationList>
    <application name="CreditApp">
      <refURL>https://<ccx-server>/adminapi/application/CreditApp</refURL>
    </application>
  </applicationList>
</applications>
```

Agent Statistics

Agentstats provides reports about the resources.

Data Structure for Agent Statistics

Attribute Name	Base Type	Restrictions	Description
loggedIn	unsignedInt	—	Number of resources currently logged in.
ready	unsignedInt	—	Number of resources currently ready.
notReady	unsignedInt	—	Number of resources currently not ready.
talking	unsignedInt	—	Number of resources currently talking.

Get Agent Statistics

HTTPS Method	GET
URI	https://<server>/adminapi/agentstats
Example URI	https://uccx-server/adminapi/agentstats
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<agentStats >
  <loggedIn>0</loggedIn>
  <ready>0</ready>
  <notReady>0</notReady>
  <talking>0</talking>
</agentStats>
```

Application

Application provides information about the application.

Data Structure for Application

Attribute Name	Base Type	Restrictions	Description
ICMPostRouting	ICMPostRouting (choice)	None	Type of script
ICMTranslationRouting	ICMTranslationRouting (choice)	None	Type of script
ScriptApplication	ScriptApplication (choice)	None	Type of script
id	int	0 to 65535 (mandatory,unique)	Numeric identifier of the application
applicationName	applicationName	max length = 30 (mandatory,unique) min length = 1 Application name contains alphanumeric characters, special characters, spaces, underscores (_), and hyphens (-) without leading or trailing white spaces.	Name that uniquely identifies the application
maxsession	int	1 to 2147483647 (mandatory)	Maximum number of sessions
enabled	string	true false (mandatory)	

Data Structure for ICMPostRouting

Attribute	Base Type	Restrictions	Description
initialScript	String	None	Initial script of the Unified CCX postrouting application. Example for system script: <pre><initialScript>aa.aef</initialScript></pre> Examples for user script: <pre><initialScript>SCRIPT[user.aef]</initialScript></pre> <pre><initialScript>SCRIPT[user/user.aef]</initialScript></pre>
timeout	Int	1 to 2147483647	Time in milliseconds that the system waits before it rejects the request for the application.

Attribute	Base Type	Restrictions	Description
defaultScript	String	None	<p>Default script of the Unified CCX postrouting application.</p> <p>Example for system script:</p> <pre><defaultScript>aa.aef</defaultScript></pre> <p>Examples for user script:</p> <pre><defaultScript>SCRIPT[user.aef]</defaultScript></pre> <pre><defaultScript>SCRIPT[user/user.aef]</defaultScript></pre>

Data Structure for ICMTranslationRouting

Attribute	Base Type	Restrictions	Description
timeout	Int	1 to 2147483647	Time in milliseconds that the system waits before it rejects the request for the application.
defaultScript	String	None	<p>Default script of the Unified CCX postrouting application.</p> <p>Example for system script:</p> <pre><defaultScript>aa.aef</defaultScript></pre> <p>Examples for user script:</p> <pre><defaultScript>SCRIPT[user.aef]</defaultScript></pre> <pre><defaultScript>SCRIPT[user/user.aef]</defaultScript></pre>

Data Structure for ScriptApplication

Attribute	Base Type	Restrictions	Description
script	String	None	<p>Script to be selected for the application. It can be system script or user script.</p> <p>Example for system script:</p> <pre><script>SSCRIPT[aa.aef]</script></pre> <p>or</p> <pre><script>aa.aef</script></pre> <p>For user script, specify the script with the relative path.</p> <p>Examples for user script:</p> <pre><script>SCRIPT[user.aef]</script></pre> <pre><script>SCRIPT[user/user.aef]</script></pre>

Attribute	Base Type	Restrictions	Description
defaultScript	String	None	<p>Default script to be executed if there are errors in running the <code>script</code> or <code>surveyScript</code></p> <p>Example for system script:</p> <pre><defaultScript>aa.aef</defaultScript></pre> <p>Examples for user script:</p> <pre><defaultScript>SCRIPT[user.aef]</defaultScript></pre> <pre><defaultScript>SCRIPT[user/user.aef]</defaultScript></pre>
scriptParams	Variable	None	<p>Parameters of the script.</p> <p>A optional flag <code>isDefaultValue</code> in the response of Get Application API with query parameter (<code>allScriptParams</code>) indicates if the value displayed against a parameter is a default value defined in the script.</p>
defaultScriptParams	Variable	None	<p>Parameters of the default script.</p> <p>An optional flag <code>isDefaultValue</code> in the response of Get Application API with query parameter (<code>allScriptParams</code>) indicates if the value displayed against a parameter is a default value defined in the script.</p>
ivrSurveyName	String	minlength=1 maxlength=128	<p>The name of the questionnaire from Cisco Webex Experience Management that is associated with the application. Configure the <code>ivrSurveyName</code> attribute to play an inline IVR survey to the customer after a call.</p> <p>This field is optional. The script application can either have <code>ivrSurveyName</code> or <code>dispatchId</code> attributes.</p>
dispatchId	String	minlength=1 maxlength=24	<p>The ID of the dispatch template from Cisco Webex Experience Management that is associated with the application. Configure the <code>dispatchId</code> attribute to send an offline SMS/Email survey to the customer.</p> <p>This field is optional. The script application can either have <code>ivrSurveyName</code> or <code>dispatchId</code> attributes.</p>

Get List of Applications

HTTPS Method	GET
URI	https://<server>/adminapi/application
Example URI	https://uccx-server/adminapi/application

Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 403, 404, 405, 500
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<applications>
  <application>
    <self>https://uccx-server/adminapi/application/Auto_Sanity_App_ICD</self>
    <id>0</id>
    <applicationName>Auto_Sanity_App_ICD</applicationName>
    <type>Cisco Script Application</type>
    <description>Automated test application</description>
    <maxsession>10</maxsession>
    <enabled>true</enabled>
  </application>
  <application>
    <self>https://uccx-server/adminapi/application/Auto_Sanity_App_AA</self>
    <id>1</id>
    <applicationName>Auto_Sanity_App_AA</applicationName>
    <type>Cisco Script Application</type>
    <description>Automated test application for AA</description>
    <maxsession>10</maxsession>
    <enabled>true</enabled>
  </application>
  <application>
    <self>https://uccx-server/adminapi/application/Auto_Application</self>
    <id>2</id>
    <applicationName>Auto_Application</applicationName>
    <type>Cisco Script Application</type>
    <description>Automated test application</description>
    <maxsession>10</maxsession>
    <enabled>true</enabled>
  </application>
  <application>
    <self>http://uccx-server/adminapi/application/WXM_APP</self>
    <ScriptApplication>
      <script>SCRIPT[icd_wxm.aef]</script>
      <dispatchId>abc12</dispatchId>
    </ScriptApplication>
    <id>0</id>
    <applicationName>WXM_APP</applicationName>
    <type>Cisco Script Application</type>
    <description>WXM_APP</description>
    <maxsession>10</maxsession>
    <enabled>true</enabled>
  </application>
</applications>
```

Get Application

HTTPS Method	GET
URI	https://<server>/adminapi/application/{applicationName}

Example URI	https://uccx-server/adminapi/application/application1
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 403, 404, 405, 500
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<application>
  <self>https://uccx-server/adminapi/application/Auto_Sanity_App_AA</self>
  <ScriptApplication>
    <script>aa.aef</script>
  </ScriptApplication>
  <id>1</id>
  <applicationName>Auto_Sanity_App_AA</applicationName>
  <type>Cisco Script Application</type>
  <description>Automated test application for AA</description>
  <maxsession>10</maxsession>
  <enabled>true</enabled>
</application>
```

Success Response with ivrSurveyName

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
  <application>
    <self>http://10.78.93.251/adminapi/application/WXMIVR_APP</self>
    <ScriptApplication>
      <script>SCRIPT[icd_wxm.aef]</script>
      <ivrSurveyName>survey1</ivrSurveyName>
    </ScriptApplication>
    <id>19</id>
    <applicationName>WXMIVR_APP</applicationName>
    <type>Cisco Script Application</type>
    <description>icd</description>
    <maxsession>10</maxsession>
    <enabled>true</enabled>
  </application>
```

Success Response with dispatchId

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
  <application>
    <self>http://10.78.93.251/adminapi/application/WXM_APP</self>
    <ScriptApplication>
      <script>SCRIPT[icd_wxm.aef]</script>
      <dispatchId>abc12</dispatchId>
    </ScriptApplication>
    <id>0</id>
    <applicationName>WXM_APP</applicationName>
    <type>Cisco Script Application</type>
    <description>WXM_APP</description>
    <maxsession>10</maxsession>
    <enabled>true</enabled>
  </application>
```

Get Application with Script Parameters

HTTPS Method	GET
URI	https://<server>/adminapi/application/{application name}?allScriptParams
Example URI	https://uccx-server/adminapi/application/ICDSystemApp7?allScriptParams
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401,403,404,500
Error Codes	—

Success Response

```

<application>
  <self>https://10.78.93.251/adminapi/application/ICDSystemApp7</self>
  <ScriptApplication>
    <script>SCRIPT[WXM_EmailSurvey.aef]</script>
    <scriptParams>
      <name>CSQ</name>
      <value>"csq1"</value>
      <type>java.lang.String</type>
    </scriptParams>
    <scriptParams>
      <name>customer_Id</name>
      <value>"Michael Littlefoot"</value>
      <type>java.lang.String</type>
    </scriptParams>
    <scriptParams>
      <name>customer_email</name>
      <value>"username@example.com"</value>
      <type>java.lang.String</type>
    </scriptParams>
    <scriptParams>
      <name>DelayWhileQueued</name>
      <value>30</value>
      <type>java.lang.Integer</type>
      <isDefaultValue>true</isDefaultValue>
    </scriptParams>
    <scriptParams>
      <name>WelcomePrompt</name>
      <value>SP[ICD\ICDWelcome.wav]</value>
      <type>com.cisco.prompt.Playable</type>
      <isDefaultValue>true</isDefaultValue>
    </scriptParams>
    <scriptParams>
      <name>QueuePrompt</name>
      <value>SP[ICD\ICDQueue.wav]</value>
      <type>com.cisco.prompt.Playable</type>
      <isDefaultValue>true</isDefaultValue>
    </scriptParams>
    <dispatchId>5e7c75d2cb93e80e60b31a24</dispatchId>
  </ScriptApplication>
  <id>1212</id>
  <applicationName>Complaints</applicationName>

```

```

    <type>Cisco Script Application</type>
    <description>Complaints</description>
    <maxsession>10</maxsession>
    <enabled>true</enabled>
  </application>

```

Create Application

HTTPS Method	POST
URI	https://<server>/adminapi/application
Example URI	https://uccx-server/adminapi/application
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Code	400, 500 401, 405, 412
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Sample Request

```

<application>
  <ScriptApplication>
    <script>SSCRIPT[icd.aef]</script>
    <scriptParams>
      <name>CSQ</name>
      <value>&quot;VoiceCSQ&quot;</value>
      <type>java.lang.String</type>
    </scriptParams>
  </ScriptApplication>
  <id>4</id>
  <applicationName>App</applicationName>
  <type>Cisco Script Application</type>
  <description>App Desc</description>
  <maxsession>10</maxsession>
  <enabled>true</enabled>
</application>

```

Sample Request with ivrSurveyName

```

<application>
  <ScriptApplication>
    <script>SCRIPT[icd_wxm.aef]</script>
    <ivrSurveyName>survey1</ivrSurveyName>
  </ScriptApplication>
  <id>19</id>
  <applicationName>WXMIVR_APP</applicationName>
  <type>Cisco Script Application</type>
  <description>icd</description>
  <maxsession>10</maxsession>

```

```

    <enabled>true</enabled>
</application>

```

Sample Request with dispatchId

```

<application>
  <ScriptApplication>
    <script>SCRIPT[icd_wxm.aef]</script>
    <dispatchId>survey2</dispatchId>
  </ScriptApplication>
  <id>19</id>
  <applicationName>WXM_APP</applicationName>
  <type>Cisco Script Application</type>
  <description>icd</description>
  <maxsession>10</maxsession>
  <enabled>true</enabled>
</application>

```

Sample Request with dispatchId and Script Parameters

```

<application>
  <ScriptApplication>
    <script>SCRIPT[WXM_EmailSurvey.aef]</script>
    <scriptParams>
      <name>CSQ</name>
      <value>"csq1"</value>
      <type>java.lang.String</type>
    </scriptParams>
    <scriptParams>
      <name>customer_Id</name>
      <value>"Michael Littlefoot"</value>
      <type>java.lang.String</type>
    </scriptParams>
    <scriptParams>
      <name>customer_email</name>
      <value>"username@example.com"</value>
      <type>java.lang.String</type>
    </scriptParams>
    <dispatchId>5e7c75d2cb93e80e60b31a24</dispatchId>
  </ScriptApplication>
  <id>1212</id>
  <applicationName>Complaints</applicationName>
  <type>Cisco Script Application</type>
  <description>Complaints</description>
  <maxsession>10</maxsession>
  <enabled>true</enabled>
</application>

```

Modify Application

HTTPS Method	PUT
URI	https://<server>/adminapi/application/{applicationName}
Example URI	https://uccx-server/adminapi/application/application1
Content Type	Application/XML, Application/JSON

HTTPS Success Code	200
HTTPS Failure Codes	400 401, 403, 405, 412, 500
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Sample Request

```
<application>
  <ScriptApplication>
    <script>SCRIPT[ICDSystemTestScriptCS7.aef]</script>
    <scriptParams>
      <name>QueuePrompt</name>
      <value>SP[ICD\ICDQueue.wav]</value>
      <type>com.cisco.prompt.Playable</type>
    </scriptParams>
    <scriptParams>
    </scriptParams>
  </ScriptApplication>
</id>19</id>
<applicationName>ICDSystemApp7</applicationName>
<type>Cisco Script Application</type>
<description>ICDSystemApp7_modified</description>
<maxsession>100</maxsession>
<enabled>true</enabled>
</application>
```

Sample Request with ivrSurveyName

```
<application>
  <ScriptApplication>
    <script>SCRIPT[icd_wxm.aef]</script>
    <ivrSurveyName>survey1</ivrSurveyName>
  </ScriptApplication>
</id>19</id>
<applicationName>WXMIVR_APP</applicationName>
<type>Cisco Script Application</type>
<description>icd</description>
<maxsession>10</maxsession>
<enabled>true</enabled>
</application>
```

Sample Request with dispatchId

```
<application>
  <ScriptApplication>
    <script>SCRIPT[icd_wxm.aef]</script>
    <dispatchId>survey2</dispatchId>
  </ScriptApplication>
</id>19</id>
<applicationName>WXM_APP</applicationName>
<type>Cisco Script Application</type>
<description>icd</description>
<maxsession>10</maxsession>
```

```

    <enabled>true</enabled>
  </application>

```

Sample Request with dispatchId and Script Parameters

```

<application>
  <ScriptApplication>
    <script>SCRIPT[WXM_EmailSurvey.aef]</script>
    <scriptParams>
      <name>CSQ</name>
      <value>"csq1"</value>
      <type>java.lang.String</type>
    </scriptParams>
    <scriptParams>
      <name>customer_Id</name>
      <value>"Michael Littlefoot"</value>
      <type>java.lang.String</type>
    </scriptParams>
    <scriptParams>
      <name>customer_email</name>
      <value>"username@example.com"</value>
      <type>java.lang.String</type>
    </scriptParams>
    <dispatchId>5e7c75d2cb93e80e60b31a24</dispatchId>
  </ScriptApplication>
  <id>1212</id>
  <applicationName>Complaints</applicationName>
  <type>Cisco Script Application</type>
  <description>Complaints</description>
  <maxsession>10</maxsession>
  <enabled>true</enabled>
</application>

```

Delete Application

HTTPS Method	DELETE
URI	https://<server>/adminapi/application/{applicationName}
Example URI	https://uccx-server/adminapi/application/application1
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405, 412
Error Codes	—



Note If the application that you delete is the only application that is associated with supervisors, the respective supervisors will lose the application management capability.

Success Response

HTTPS/1.1 200 OK

Assign Prompts to Application

HTTPS Method	PUT
URI	https://<server>/adminapi/application/<ApplicationName>/prompts
Example URI	https://uccx-server/adminapi/application/CareApplication/prompts
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 401, 403, 404, 405, 500
Error Codes	—

Success Response

HTTP status code 200 OK

Sample input XML—PUT

```
<Prompts>
  <Prompt>
    <Path>nameWithFullPath</Path>
    <Type>file</Type>
  </Prompt>
  <Prompt>
    <Path>nameWithFullPath</Path>
    <Type>folder</Type>
  </Prompt>
</Prompts>
```



Note POST and DELETE methods are not supported. Corresponding error response will be returned.

Get Prompts of Application

HTTPS Method	GET
URI	https://<server>/adminapi/application/<ApplicationName>/prompts https://<server>/adminapi/application/<ApplicationName>/prompts?expandFolders
Example URI	https://uccx-server/adminapi/application/CareApplication/prompts https://uccx-server/adminapi/application/CareApplication/prompts?expandFolders
Accept	Application/XML, Application/JSON

HTTPS Success Code	200
HTTPS Failure Codes	400, 401, 403, 404, 500
Error Codes	—



Note Use "expandFolders" query parameter when you want to get all the files that are in the assigned folders.

Success Response

```
<Prompts>
  <Prompt>
    <Path>nameWithFullPath</Path>
    <Type>file</Type>
  </Prompt>
  <Prompt>
    <Path>nameWithFullPath</Path>
    <Type>folder</Type>
  </Prompt>
</Prompts>
```

Example without "expandFolders" query parameter

```
<Prompts>
  <Prompt>
    <Path>en_US/prompt1.wav</Path>
    <Type>file</Type>
  </Prompt>
  <Prompt>
    <Path>en_AU</Path>
    <Type>folder</Type>
  </Prompt>
</Prompts>
```

Example with "expandFolders" query parameter

```
<Prompts>
  <Prompt>
    <Path>en_US/prompt1.wav</Path>
    <Type>file</Type>
  </Prompt>
  <Prompt>
    <Path>en_AU</Path>
    <Type>folder</Type>
  </Prompt>
  <Prompt>
    <Path>en_AU/prompt2.wav</Path>
    <Type>file</Type>
  </Prompt>
</Prompts>
```


Area Code

The area code determines the geographical location of the phone number you dial, which provides the Greenwich Meridian Time (GMT) zone. The db_cra database contains a mapping of the area codes to the time zones. The U.S. area code mappings are provided along with the product. International customers should provide their own data and add it to the database.

Data Structure for Area Code

Attribute Name	Base Type	Restrictions	Description
areaCode	string	minlength = 1 maxlength = 10 occurrences = 1 The valid characters for this field are “\d+”.	The area code of the call
region	int	minOccurs = 1 maxOccurs = 1	Uses the same data as that of gmtzone
dst	boolean	minoccurs = 0 maxOccurs = 1 Default = False Any value other than “True” is considered as False.	Daylight Savings Time

Get List of Area Codes

HTTPS Method	GET
URI	https://<server>/adminapi/areaCode
Example URI	https://uccx-server/adminapi/areaCode
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<areaCodes>
  <areaCode>
    <self>https://uccx-server/adminapi/areaCode/219</self>
    <areacode>219</areacode>
    <regioncode>158</regioncode>
    <dst>>false</dst>
  </areaCode>
  <areaCode>
    <self>https://uccx-server/adminapi/areaCode/615</self>
    <areacode>615</areacode>
    <regioncode>92</regioncode>
    <dst>>true</dst>
  </areaCode>
  <areaCode>
    <self>https://uccx-server/adminapi/areaCode/607</self>
    <areacode>607</areacode>
    <regioncode>158</regioncode>
    <dst>>true</dst>
  </areaCode>
</areaCodes>
```

Get Area Code

HTTPS Method	GET
URI	https://<server>/adminapi/areaCode/{areacode}
Example URI	https://uccx-server/adminapi/areaCode/760
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<areaCode>
  <self>https://uccx-server/adminapi/areaCode/219</self>
  <areaCode>219</areaCode>
  <regioncode>158</regioncode>
  <dst>>false</dst>
</areaCode>
```

Create Area Code

HTTPS Method	POST
URI	https://<server>/adminapi/areacode
Example URI	https://uccx-server/adminapi/areacode

Content Type	Application/XML, Application/JSON
HTTPS Success code	201
HTTPS Failure Codes	401, 405, 409
Error Codes	ConfigNotFoundException SQLException

Success Response

HTTPS/1.1 201 Created

Sample Input XML—POST

```
<?xml version="1.0" encoding="UTF-8"?>
<AreaCode>
  <self>https://uccx-server/adminapi/areacode</self>
  <areacode>9111</areacode>
  <regioncode>140</regioncode>
  <dst>true</dst>
</AreaCode>
```

Modify Area Code

HTTPS Method	PUT
URI	https://<server>/adminapi/areacode/{areacode}
Example URI	https://uccx-server/adminapi/areacode/760
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—

Success Response

—

Sample Input XML—PUT

```
<?xml version="1.0" encoding="UTF-8"?>
<AreaCode>
  <self>https://uccx-server/adminapi/areaCode</self>
  <areacode>9111</areacode>
  <regioncode>140</regioncode>
  <dst>false</dst>
</AreaCode>
```

Delete Area Code

HTTPS Method	DELETE
URI	https://<server>/adminapi/areacode/{areacode}
Example URI	https://uccx-server/adminapi/areacode/760
Content Type	Application/XML, Application/JSON
HTTPS Success code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	HTTPS/1.1 404 Not Found

Success Response

HTTPS/1.1 200 OK

Calendar

The Calendar Management allows the Unified CCX administrator to define the business calendar hours, holidays, and special days.

Data Structure for Calendar

Attribute Name	Base Type	Restrictions	Description
callId	int	minOccurs = 0 maxOccurs = 1	Unique identifier of the calendar.
name	string	1 to 30 characters length	Name of the calendar.
timeZone	string	Valid time zone	The time zone of the calendar.
description	string	1 to 70 characters length (not mandatory)	The description of the calendar.

Attribute Name	Base Type	Restrictions	Description
calendarType	string	minOccurs = 1 maxOccurs = 1	Indicates the type of the calendar. 1—Full time 2—Fixed Hours 3—Flexible Hours Note Selecting either the Fixed Hours or Flexible Hours makes the businessDays attribute as mandatory.
buisnessDays	ccBusinessDays	minOccurs = 0 maxOccurs = 1	The custom weekly business calendar days that are configured.
specialDays	ccSpecialDays	minOccurs = 0 maxOccurs = 1	The special business calendar days that are configured.
holidays	ccHolidays	minOccurs = 0 maxOccurs = 1	The calendar business holidays that are configured.
associations	associations	minOccurs = 0 maxOccurs = 1	Applications and chat widgets that the calendar is associated with.

Other Restrictions

- There can be a maximum 50 calendars.
- Calendar name should be unique.
- Day of week (businessDay) name should be unique for a calendar.
- Maximum 3 intervals can be configured per day.
- Interval names to be unique within a businessDay.
- Interval time range to not overlap within a businessDay.
- Interval end time to be greater than start time.
- Maximum 40 Special days and 40 Holidays can be configured per calendar.
- Special days and Holidays are optional.
- There should not be any overlap of the dates configured for Specialdays and Holidays.
- Create calendar does not allow older dates.
- Allowed only for admin and supervisor.

- Associations cannot be added or modified by using API.

Data Structure for Businessdays

Primary Attribute	Secondary Attribute	Sub Attribute	Base Type	Restrictions	Description
dayOfWeek	—	—	—	minOccur = 1 maxOccur = 1 Enumerations = MON, TUE, WED, THU,FRI, SAT, SUN	Day of the week.
intervals	interval	name	string	25 characters length	Name of the shift.
		Start time	string	minOccur = 1 maxOccur = 1 Between 00:00 to 23:59 inclusive	Start time of the business hour.
		End time	string	minOccur = 1 maxOccur = 1 Between 00:00 to 23:59 inclusive	End time of business hour.

Data Structure for Holidays

Primary Attribute	Secondary Attribute	Sub Attribute	Base Type	Restrictions	Description
name	—	—	string	minOccur = 1 maxOccur = 1 1 to 50 characters length	Name of the occasion.
date	—	—	string	minOccur = 1 maxOccur = 1	Date of the occasion.

Data Structure for Specialdays

Primary Attribute	Secondary Attribute	Sub Attribute	Base Type	Restrictions	Description
name	—	—	string	minOccur = 1 maxOccur = 1 1 to 50 characters length	Name of the occasion.
date	—	—	string	minOccur = 1 maxOccur = 1	Date of the occasion.
intervals	interval	name	string	25 characters length	Name of the interval.
		Start time	string	minOccur = 1 maxOccur = 1 Between 00:00 to 23:59 inclusive	Start time of the business hour.
		End time	string	minOccur = 1 maxOccur = 1 Between 00:00 to 23:59 inclusive	End time of the business hour.

Create Calendar

HTTPS Method	POST
URI	https://<server>/adminapi/calendar
Example URI	https://uccx-server/adminapi/calendar
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	400,401,403,409,415
Error Codes	500

Success Response

HTTPS/1.1 201 Created

Sample Input XML

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<calendar>
  <name>Calendar1</name>
  <timeZone>Asia/Kolkata</timeZone>
  <description>description</description>
  <calendarType>FLEXIBLEHOURS</calendarType>
  <businessDays>
    <businessDay>
      <dayOfWeek>SUN</dayOfWeek>
      <intervals>
        <interval>
          <name>MorningShift</name>
          <startTime>06:00</startTime>
          <endTime>13:00</endTime>
        </interval>
        <interval>
          <name>AfternoonShift</name>
          <startTime>14:00</startTime>
          <endTime>19:00</endTime>
        </interval>
      </intervals>
    </businessDay>
    <businessDay>
      <dayOfWeek>MON</dayOfWeek>
      <intervals>
        <interval>
          <name>Morning</name>
          <startTime>08:00</startTime>
          <endTime>15:33</endTime>
        </interval>
        <interval>
          <name>Evening</name>
          <startTime>17:22</startTime>
          <endTime>22:33</endTime>
        </interval>
      </intervals>
    </businessDay>
  </businessDays>
  <holidays>
    <holiday>
      <name>NewYear</name>
      <date>2019-01-01</date>
    </holiday>
    <holiday>
      <name>Christmas</name>
      <date>2019-12-25</date>
    </holiday>
  </holidays>
  <specialDays>
    <specialDay>
      <name>ChristmasEve</name>
      <date>2018-24-12</date>
      <intervals>
        <interval>
          <name>Extra</name>
          <startTime>01:22</startTime>
          <endTime>22:33</endTime>
        </interval>
      </intervals>
    </specialDay>
    <specialDay>
      <name>NewYearEVE</name>
      <date>2017-31-12</date>
      <intervals>

```

```

    <interval>
      <name>Special</name>
      <startTime>01:22</startTime>
      <endTime>23:33</endTime>
    </interval>
  </intervals>
</specialDay>
</specialDays>
</calendar>

```

Get Calendar

HTTPS Method	GET
URI	https://<server>/adminapi/calendar/{id}
Example URI	https://uccx-server/adminapi/calendar/1
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400,401,403,404,406,500

Success Response

```

<?xml version="1.0" encoding="UTF-8"?>
<calendar>
  <self>https://uccx-server/adminapi/calendar/1</self>
  <calId>1</calId>
  <name>Calendar1</name>
  <timeZone>Asia/Kolkata</timeZone>
  <description>description</description>
  <calendarType>FLEXIBLEHOURS</calendarType>
  <businessDays>
    <businessDay>
      <dayOfWeek>SUN</dayOfWeek>
      <intervals>
        <interval>
          <name>MorningShift</name>
          <startTime>06:00</startTime>
          <endTime>13:00</endTime>
        </interval>
        <interval>
          <name>AfternoonShift</name>
          <startTime>14:00</startTime>
          <endTime>19:00</endTime>
        </interval>
      </intervals>
    </businessDay>
    <businessDay>
      <dayOfWeek>MON</dayOfWeek>
      <intervals>
        <interval>
          <name>Morning</name>
          <startTime>08:00</startTime>
          <endTime>15:33</endTime>
        </interval>
        <interval>
          <name>Evening</name>
          <startTime>17:22</startTime>
          <endTime>22:33</endTime>
        </interval>
      </intervals>
    </businessDay>
  </businessDays>
  <holidays>
    <holiday>
      <name>NewYear</name>
      <date>2019-01-01</date>
    </holiday>
    <holiday>
      <name>Christmas</name>
      <date>2019-12-25</date>
    </holiday>
  </holidays>
  <specialDays>
    <specialDay>
      <name>ChristmasEve</name>
      <date>2018-24-12</date>
      <intervals>
        <interval>
          <name>Extra</name>
          <startTime>01:22</startTime>
          <endTime>22:33</endTime>
        </interval>
      </intervals>
    </specialDay>
  </specialDays>
  <specialDay>
    <name>NewYearEVE</name>
  </specialDay>

```

```

<date>2017-31-12</date>
<intervals>
  <interval>
    <name>Special</name>
    <startTime>01:22</startTime>
    <endTime>23:33</endTime>
  </interval>
</intervals>
</specialDay>
</specialDays>
</calendar>

```

Get Calendar with Associations

HTTPS Method	GET
URI	https://<server>/adminapi/calendar/{id}?withAssociations
Example URI	https://uccx-server/adminapi/calendar/1?withAssociations
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400,401,403,404,406,500

Success Response

```

<calendar>
  <self>https://uccx-server/adminapi/calendar/19</self>
  <calId>19</calId>
  <name>TestCalendar</name>
  <timeZone>Asia/Kolkata</timeZone>
  <description></description>
  <calendarType>FULLTIME</calendarType>
  <associations>
    <applications>
      <application name="TestApplication">
        <refURL>https://uccx-server/adminapi/application/TestApplication</refURL>
      </application>
      <application name="TestApp2">
        <refURL>https://uccx-server/adminapi/application/TestApp2</refURL>
      </application>
    </applications>
    <chatWidgets>
      <chatWidget name="chatWidget">
        <refURL>http://uccx-server/adminapi/chatWidget/1</refURL>
      </chatWidget>
    </chatWidgets>
  </associations>
</calendar>

```

Get List of Calendars

HTTPS Method	GET
---------------------	-----

URI	https://<server>/adminapi/calendar
Example URI	https://uccx-server/adminapi/calendar
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401,403,404,406,500

Success Response

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<calendars>
  <calendar>
    <self>https://uccx-server/adminapi/calendar/1</self>
    <calId>1</calId>
    <name>Calendar1</name>
    <timeZone>Asia/Kolkata</timeZone>
    <description>description</description>
    <calendarType>FLEXIBLEHOURS</calendarType>
    <businessDays>
      <businessDay>
        <dayOfWeek>SUN</dayOfWeek>
        <intervals>
          <interval>
            <name>MorningShift</name>
            <startTime>06:00</startTime>
            <endTime>13:00</endTime>
          </interval>
          <interval>
            <name>AfternoonShift</name>
            <startTime>14:00</startTime>
            <endTime>19:00</endTime>
          </interval>
        </intervals>
      </businessDay>
      <businessDay>
        <dayOfWeek>MON</dayOfWeek>
        <intervals>
          <interval>
            <name>Morning</name>
            <startTime>08:00</startTime>
            <endTime>15:33</endTime>
          </interval>
          <interval>
            <name>Evening</name>
            <startTime>17:22</startTime>
            <endTime>22:33</endTime>
          </interval>
        </intervals>
      </businessDay>
    </businessDays>
    <holidays>
      <holiday>
        <name>NewYear</name>
        <date>2019-01-01</date>
      </holiday>
      <holiday>
        <name>Christmas</name>
        <date>2019-12-25</date>
      </holiday>
    </holidays>
    <specialDays>
      <specialDay>
        <name>ChristmasEve</name>
        <date>2018-24-12</date>
        <intervals>
          <interval>
            <name>Extra</name>
            <startTime>01:22</startTime>
            <endTime>22:33</endTime>
          </interval>
        </intervals>
      </specialDay>
    </specialDays>
  </calendar>
</calendars>

```



```

    <name>NewYearEVE</name>
    <date>2017-31-12</date>
    <intervals>
      <interval>
        <name>Special</name>
        <startTime>01:22</startTime>
        <endTime>23:33</endTime>
      </interval>
    </intervals>
  </specialDay>
</specialDays>
</calendar>
<calendar>
  .....
  .....
</calendar>
</calendars>

```

Get List of Calendars with Associations

HTTPS Method	GET
URI	https://<server>/adminapi/calendar?withAssociations
Example URI	https://uccx-server/adminapi/calendar?withAssociations
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401,403,404,406,500

Success Response

```

<calendars>
<calendar>
  <self>https://uccx-server/adminapi/calendar/19</self>
  <calId>19</calId>
  <name>TestCalendar</name>
  <timeZone>Asia/Kolkata</timeZone>
  <description></description>
  <calendarType>FULLTIME</calendarType>
  <associations>
    <applications>
      <application name="TestApplication">
        <refURL>https://uccx-server/adminapi/application/TestApplication</refURL>
      </application>
      <application name="TestApp2">
        <refURL>https://uccx-server/adminapi/application/TestApp2</refURL>
      </application>
    </applications>
    <chatWidgets>
      <chatWidget name="chatWidget">
        <refURL>http://uccx-server/adminapi/chatWidget/1</refURL>
      </chatWidget>
    </chatWidgets>
  </associations>
</calendar>
</calendars>

```

Get List of Calendars with Summary

HTTPS Method	POST
URI	https://<server>/adminapi/calendar?detail=brief
Example URI	https://uccx-server/adminapi/calendar?detail=brief
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401,403,406,500
Error Codes	—

Success Response

```

<calendars>
  <calendar>
    <self>https://uccx-server/adminapi/calendar/1</self>
    <calId>1</calId>
    <name>Calendar1</name>
    <timeZone>Asia/Kolkata</timeZone>
    <description>description</description>
    <calendarType>FLEXIBLEHOURS</calendarType>
    <applications>
      <application name="TestApplication">
        <refURL>https://uccx-server/adminapi/application/TestApplication</refURL>
      </application>
      <application name="TestApp2">
        <refURL>https://uccx-server/adminapi/application/TestApp2</refURL>
      </application>
    </applications>
    <holidaysConfigured>true</holidaysConfigured>
    <customBusinessDaysConfigured>>false</customBusinessDaysConfigured>
  </calendar>
  <calendar>
    .....
    .....
  </calendar>
</calendars>

```

Error Response

```

<apiErrors>
  <apiError>
    <errorData>{id}</errorData>
    <errorMessage>{error message}</errorMessage>
    <errorType>{error type}</errorType>
    <i18n>
      <key>{error key}</key>
      <args>
        <arg>{value-1}</arg>
        <arg>{value-2}</arg>
      </args>
    </i18n>
  </apiError>
</apiErrors>

```

Get Calendars Managed by Supervisors

HTTPS Method	GET
URI	https://<server>/adminapi/resource/{id}/capabilities/calendars
Example URI	https://uccx-server/adminapi/resource/1/capabilities/calendars
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200

HTTPS Failure Codes	400,401,403,404,500
Error Codes	—

Success Response

```
<calendars>
  <resource name="Rick Barrows">
    <refURL>https://<uccx-server>/adminapi/resource/rick</refURL>
  </resource>
  <calendarList>
    <calendar name="Calendar1">
      <refURL>Link to Calendar1</refURL>
    </calendar>
    <calendar name="Calendar2">
      <refURL>Link to Calendar2</refURL>
    </calendar>
  </calendarList>
</calendars>
```

Error Response

```
<apiErrors>
  <apiError>
    <errorData>{id}</errorData>
    <errorMessage>{error message}</errorMessage>
    <errorType>{error type}</errorType>
    <i18n>
      <key>{i18n_key}</key>
      <args>
        <arg>{i18n_value}</arg>
      </args>
    </i18n>
  </apiError>
</apiErrors>
```

Modify Calendar

HTTPS Method	PUT
URI	https://< server >/adminapi/calendar/{id}
Example URI	https://uccx-server/adminapi/calendar/1
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400,401,403,404,415,500

Success Response

HTTPS/1.1 200 OK

Modify Calendars Managed by Supervisors

HTTPS Method	PUT
URI	https://<server>/adminapi/resource/{id}/capabilities/calendars
Example URI	https://uccx-server/adminapi/resource/1/capabilities/calendars
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400,401,403,405,500
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Sample Input XML

```
<calendars>
  <resource name="Rick Barrows">
    <refURL>https://<uccx-server>/adminapi/resource/rick</refURL>
  </resource>
  <calendarList>
    <calendar name="businesscalendar1">
      <refURL>https://<uccx-server>/adminapi/calendar/1</refURL>
    </calendar>
  </calendarList>
</calendars>
```

Delete Calendar

HTTPS Method	DELETE
URI	https://< server >/adminapi/calendar/{id} https://< server >/adminapi/calendar/{id}[?forced]
Example URI	https://uccx-server/adminapi/calendar/1 https://uccx-server/adminapi/calendar/1[?forced]
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400,401,403,404,500

Success Response

HTTPS/1.1 200 OK

Error Response

```

<apiErrors>
  <apiError>
    <errorData>Unable to delete calendar as applications [App1, App2] are associated
with this calendar</errorData>
    <errorMessage>Unable to delete calendar as applications [App1, App2] are associated
with this calendar</errorMessage>
    <errorType>400</errorType>
    <i18n>
      <key>adminapi.calendar.application.association.delete.alert</key>
      <args>
        <arg>[App1, App2]</arg>
      </args>
    </i18n>
  </apiError>
</apiErrors>

```

Call Control Group

Call Control Group allows you to control the use of CTI ports. Unified CCX uses Unified CM Telephony call control groups to pool together a series of CTI ports, which the system uses to serve calls as they arrive from the Unified CCX server. You can create multiple Unified CM Telephony call control groups to share and limit the resources to be used by specific applications. You can reserve more ports for higher-priority applications or provide access to fewer ports for applications with less traffic.

Data Structure for Call Control Group

Attribute Name	Base Type	Restrictions	Description
id	int	minOccurs = 0 maxOccurs = 1 nillable = false	Unique identifier for the Call Control Group.
description	string	minOccurs = 1 maxOccurs = 1 nillable = false minLength = 1 maxLength = 30 Description contains alphanumeric characters and can have white space between characters.	Description of the Call Control Group.

Attribute Name	Base Type	Restrictions	Description
numOfCTIPorts	int	minOccurs = 1 maxOccurs = 1 nillable = false minInclusive = 1 maxLength = 65535	Number of CTI ports that are assigned to the Call Control Group.
outboundGroup	boolean	minOccurs = 1 maxOccurs = 1 nillable = false default = false	Checks for the inbound or outbound group that is associated with the Call Control Group.
mediaTerminationSupport	boolean	minOccurs = 1 maxOccurs = 1 nillable = false default = false	Enables auto creation of media termination groups.
deviceNamePrefix	string	minOccurs = 1 maxOccurs = 1 nillable = false minLength = 1 maxLength = 5 Device name prefix contains alphanumeric characters, underscores (_), hyphens (-), and periods (.) without leading or trailing white spaces.	Prefixed to the name that will be given to all of the CTI ports in the group.

Data Structure for Port Group Configuration Type

Attribute Name	Base Type	Restrictions	Description
hostName	string	minOccurs = 0 maxOccurs = 1 nillable = false minLength = 1 maxLength = 255 Hostname contains alphanumeric characters.	IP address or domain name.

Attribute Name	Base Type	Restrictions	Description
listOfCTIPorts	string	minOccurs = 0 maxOccurs = 1 nillable = false	List of CTI ports assigned to the Call Control Group.
startingDirectoryNumber	string	minOccurs = 1 maxOccurs = 1 nillable = false minLength = 1 maxLength = 24 Starting directory number contains numerals and can have asterisk (*) or hash (#) or both as a prefix or a suffix. To support E.164 compliance, Unified CCX allows you to add plus sign (+) before the directory number.	Specified number of ports are created starting from the value that is specified in this field or starting from the next available directory number.
devicePool	CCMCommonObjectName	minOccurs = 1 maxOccurs = 1 nillable = false	Set of common parameters like region, location, SRST reference, etc that can be applied for a device.
dnCallingSearchSpace	CCMCommonObjectName	minOccurs = 0 maxOccurs = 1 nillable = false	Collection of partitions that is searched to determine how to route a dialed number.
location	CCMCommonObjectName	minOccurs = 1 maxOccurs = 1 nillable = false	The Cisco Unified Communications Manager phone location setting specifies the total bandwidth that is available for calls to and from this location.

Attribute Name	Base Type	Restrictions	Description
partition	CCMPartitionName	minOccurs = 0 maxOccurs = 1 nillable = false	The partition to which the Directory Number belongs.
mediaResourceGroupList	CCMCommonObjectName	minOccurs = 0 maxOccurs = 1 nillable = false	Media Resource Group List provides a prioritized grouping of media resource groups. Application chooses the required media resource according to the priority order that is defined in a Media Resource Group List.
voiceMailProfile	CCMCommonObjectName	minOccurs = 0 maxOccurs = 1 nillable = false	List of profiles defined in the Voice Mail Profile Configuration. Voice Mail Profile allows you to define any line-related voice-messaging information that is associated to a directory number.
presenceGroup	CCMCommonObjectName	minOccurs = 1 maxOccurs = 1 nillable = false	Presence group controls the destinations that can be monitored by the watchers in the group.

Attribute Name	Base Type	Restrictions	Description
AARGroup	string	minOccurs = 0 maxOccurs = 1 nillable = false minLength = 1 maxLength = 20 AAR group contains alphanumeric characters, spaces, underscores (_), hyphens (-), and periods (.) without leading or trailing white spaces.	Automated Alternate Routing (AAR) group provides the prefix digits that are used to route calls that are otherwise blocked due to insufficient bandwidth.
userHoldAudioSource	CCMHoldAudioSourceName	minOccurs = 0 maxOccurs = 1 nillable = false	Audio source heard by the caller when the Unified CCX script places the caller on hold.
networkHoldAudioSource	CCMHoldAudioSourceName	minOccurs = 0 maxOccurs = 1 nillable = false	Audio source that is heard by the caller when Unified CCX performs a Consult Transfer.
callPickUpGroup	string	minOccurs = 0 maxOccurs = 1 nillable = false minLength = 1 maxLength = 24 Call pickup group contains numerals (0-9), alphabets (A-D), plus (+), pound (#) and asterisk (*). Note Slash (\) must be precede plus (+).	Call pickup group allows you to answer a call that comes in on a directory number other than your own. When you hear an incoming call ringing on another phone, you can redirect the call to your phone by using this feature.

Attribute Name	Base Type	Restrictions	Description
alertingNameAscii	CCMAAlertingNameASCII	minOccurs = 0 maxOccurs = 1	Name that is displayed during an alert to a shared directory number. Devices that do not support Unicode (internationalized) characters display the Alerting Name ASCII field.

Attribute Name	Base Type	Restrictions	Description
redirectCallingSearchSpace	CCMRedirectCallingSearchSpace	minOccurs = 1 maxOccurs = 1 nillable = false	<p>Collection of partitions that are searched to determine how a redirected call is routed.</p> <p>Redirect Calling Search Space options:</p> <ul style="list-style-type: none"> • DN Calling Search Space—Enables the CTI Port to use its directory number CSS when performing a redirect / consult transfer. • Calling Party—Enables the CTI Port to use the calling party's CSS when performing a redirect / consult transfer. • Redirect Party—Enables the CTI Port to use the CTI Route Point's CSS when performing a redirect / consult transfer.

Attribute Name	Base Type	Restrictions	Description
display	string	minOccurs = 1 maxOccurs = 1 minLength = 0 maxLength = 30 Display contains alphanumeric characters and symbols except tab (\t), new line (\n), double quotation marks ("), brackets (< >, { }, []), percent (%), and ampersand (&).	Username or the directory number that is configured.
externalPhoneMaskNumber	CCMExternalPhoneMaskNumber	minOccurs = 0 maxOccurs = 1	Phone number that is used to send caller ID information when a call is placed from this line.

Simple Data Structure Used in Call Control Group

Attribute Name	Base Type	Restrictions	Description
CCMAlertingNameASCII	string	minLength = 0 maxLength = 30 CCM Alerting Name ASCII can include all special characters except double quotation marks ("), angle brackets (< >), ampersand (&), and percent (%). Do not use the following keywords that are reserved in Unified Communications Manager: Conference, Park Number, Barge, CBarge, Monitor, and Record.	Data type that is used in Unified Communications Manager for AlertingNameASCII.

Attribute Name	Base Type	Restrictions	Description
CCMCommonObjectName	string	minLength = 1 maxLength = 50 CCMCommonObjectName contains alphanumeric characters, spaces, underscores (_), hyphens (-), and periods (.) without leading or trailing white spaces.	Data type that is used in Unified Communications Manager for CommonObjectName.
CCMHoldAudioSourceName	string	minLength = 1 maxLength = 50 CCMHoldAudioSourceName contains numbers from 1 to 51 followed by hyphen (-) and alphanumeric characters with periods (.), hyphens (-), underscores (_) or spaces.	Data type that is used in Unified Communications Manager for HoldAudioSourceName.
CCMPartitionName	string	minLength = 1 maxLength = 50 CCMPartitionName contains alphanumeric characters, spaces, underscores (_), periods (.), and hyphens (-).	Data type that is used in Unified Communications Manager for PartitionName.
CCMRedirectCallingSearchSpace	string	enumeration = Redirect Party, Calling Party, DN Calling Search Space.	Data type that is used in Unified Communications Manager for RedirectCallingSearchSpace.
CCMExternalPhoneMaskNumber	string	minLength = 0 maxLength = 24 CCM external phone mask number contains numerals (0-9), hashes (#), asterisks (*), and X.	Data type that is used in Unified Communications Manager for CCMExternalPhoneMaskNumber.

Data Structure for Call Control Group Job Status

Attribute Name	Base Type	Restrictions	Description
totalPorts	int	minOccurs = 1 maxOccurs = 1 nillable = false	Total number of CTI ports.
updatePorts	int	minOccurs = 1 maxOccurs = 1 nillable = false	Total number of updated ports.
lastUpdatedPort	string	minOccurs = 1 maxOccurs = 1 nillable = false	The last-updated port.
type	string	minOccurs = 1 maxOccurs = 1 nillable = false	Job type—Creation, deletion, modification.
exception	string	minOccurs = 1 maxOccurs = 1 nillable = false	Any exception that occurs while performing POST, PUT, or DELETE.
completed	boolean	minOccurs = 1 maxOccurs = 1 nillable = false	Job is complete.

Get List of Call Control Groups

HTTPS Method	GET
URI	https://<server>/adminapi/callControlGroup
Example URI	https://uccx-server/adminapi/callControlGroup
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 412, 500, 503
Error Codes	—

Success Response**For LAN**

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<callControlGroups>
  <callControlGroup>
    <self type="callControlGroup" href="https://uccx-server/adminapi/callControlGroup/102"
rel="self"/>
    <id>102</id>
    <outboundGroup>>false</outboundGroup>
    <description>description</description>
    <deviceNamePrefix>cti</deviceNamePrefix>
    <numOfCTIPorts>10</numOfCTIPorts>
    <mediaTerminationSupport>>false</mediaTerminationSupport>
  </callControlGroup>
  <callControlGroup>
    <self type="callControlGroup" href="https://uccx-server/adminapi/callControlGroup/107"
rel="self"/>
    <id>107</id>
    <outboundGroup>>false</outboundGroup>
    <deviceNamePrefix>cti</deviceNamePrefix>
    <description>testccg</description>
    <numOfCTIPorts>5</numOfCTIPorts>
    <mediaTerminationSupport>>false</mediaTerminationSupport>
  </callControlGroup>
</callControlGroups>
```

For WAN

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<callControlGroups>
  <callControlGroup>
    <self type="callControlGroup" href="https://uccx-server/adminapi/callControlGroup/4"
rel="self"/>
    <id>4</id>
    <outboundGroup>>false</outboundGroup>
    <description>sdf</description>
    <deviceNamePrefix>cti</deviceNamePrefix>
    <numOfCTIPorts>2</numOfCTIPorts>
    <mediaTerminationSupport>>false</mediaTerminationSupport>
  </callControlGroup>
  <callControlGroup>
    <self type="callControlGroup" href="https://uccx-server/adminapi/callControlGroup/2"
rel="self"/>
    <id>2</id>
    <outboundGroup>>false</outboundGroup>
    <description>TEST</description>
    <deviceNamePrefix>cti</deviceNamePrefix>
    <numOfCTIPorts>1</numOfCTIPorts>
    <mediaTerminationSupport>>false</mediaTerminationSupport>
  </callControlGroup>
</callControlGroups>
```

Get Call Control Group

HTTPS Method	GET
URI	https://<server>/adminapi/callControlGroup/{id}
Example URI	https://uccx-server/adminapi/callControlGroup/12

Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 412, 500, 503
Error Codes	—

Success Response

For LAN

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<callControlGroup>
  <self href="https://uccx-server/adminapi/callControlGroup/107" rel="self"
type="callControlGroup"/>
  <id>107</id>
  <outboundGroup>false</outboundGroup>
  <description>testccg</description>
  <deviceNamePrefix>testc</deviceNamePrefix>
  <numOfCTIPorts>5</numOfCTIPorts>
  <mediaTerminationSupport>false</mediaTerminationSupport>
  <portGroupConfiguration>

<listOfCTIPorts>testc_68900,testc_68907,testc_68908,testc_68909,testc_68910,testc_68911,testc_68912,testc_68913,
testc_68914,testc_68915</listOfCTIPorts>
  <startingDirectoryNumber>68900</startingDirectoryNumber>
  <devicePool>Default</devicePool>
  <dnCallingSearchSpace>None</dnCallingSearchSpace>
  <location>Hub_None</location>
  <partition>None</partition>
  <mediaResourceGroupList>None</mediaResourceGroupList>
  <voiceMailProfile>None</voiceMailProfile>
  <presenceGroup>Standard Presence group</presenceGroup>
  <AARGroup>None</AARGroup>
  <userHoldAudioSource>None</userHoldAudioSource>
  <networkHoldAudioSource>None</networkHoldAudioSource>
  <callPickUpGroup>None</callPickUpGroup>
  <alertingNameAscii></alertingNameAscii>
  <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
  <requireDTMFReception>true</requireDTMFReception>
  <display></display>
  <externalPhoneMaskNumber></externalPhoneMaskNumber>
  </portGroupConfiguration>
</callControlGroup>
```

For WAN—When CTI ports are configured for both nodes

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<callControlGroup>
  <self href="https://uccx-server/adminapi/callControlGroup/12" rel="self"
type="callControlGroup"/>
  <id>2</id>
  <outboundGroup>false</outboundGroup>
  <description>TEST</description>
  <deviceNamePrefix>VISH1</deviceNamePrefix>
  <numOfCTIPorts>1</numOfCTIPorts>
  <mediaTerminationSupport>false</mediaTerminationSupport>
  <portGroupConfiguration>
  <hostName>ccx-94-80</hostName>
  <listOfCTIPorts>VISH1_1000</listOfCTIPorts>
  <startingDirectoryNumber>1000</startingDirectoryNumber>
```

```

<devicePool>Default</devicePool>
<dnCallingSearchSpace>None</dnCallingSearchSpace>
<location>Hub_None</location>
<partition>None</partition>
<mediaResourceGroupList>None</mediaResourceGroupList>
<voiceMailProfile>None</voiceMailProfile>
<presenceGroup>Standard Presence group</presenceGroup>
<AARGroup>None</AARGroup>
<userHoldAudioSource>None</userHoldAudioSource>
<networkHoldAudioSource>None</networkHoldAudioSource>
<callPickUpGroup>None</callPickUpGroup>
<alertingNameAscii></alertingNameAscii>
<redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
<requireDTMFReception>true</requireDTMFReception>
<display></display>
<externalPhoneMaskNumber></externalPhoneMaskNumber>
</portGroupConfiguration>
<portGroupConfiguration>
  <hostName>ccx-94-81</hostName>
  <listOfCTIPorts>VISH1_576</listOfCTIPorts>
  <startingDirectoryNumber>576</startingDirectoryNumber>
  <devicePool>Default</devicePool>
  <dnCallingSearchSpace>None</dnCallingSearchSpace>
  <location>Hub_None</location>
  <partition>None</partition>
  <mediaResourceGroupList>None</mediaResourceGroupList>
  <voiceMailProfile>None</voiceMailProfile>
  <presenceGroup>Standard Presence group</presenceGroup>
  <AARGroup>None</AARGroup>
  <userHoldAudioSource>None</userHoldAudioSource>
  <networkHoldAudioSource>None</networkHoldAudioSource>
  <callPickUpGroup>None</callPickUpGroup>
  <alertingNameAscii></alertingNameAscii>
  <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
  <requireDTMFReception>true</requireDTMFReception>
  <display></display>
  <externalPhoneMaskNumber></externalPhoneMaskNumber>
</portGroupConfiguration>
</callControlGroup>

```

For WAN—When CTI ports are configured for only one node

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<callControlGroup>
  <self href="https://uccx-server/adminapi/callControlGroup/12" rel="self"
type="callControlGroup"/>
  <id>2</id>
  <outboundGroup>>false</outboundGroup>
  <description>TEST</description>
  <deviceNamePrefix>VISH1</deviceNamePrefix>
  <numOfCTIPorts>1</numOfCTIPorts>
  <mediaTerminationSupport>>false</mediaTerminationSupport>
  <portGroupConfiguration>
    <hostName>ccx-94-80</hostName>
    <listOfCTIPorts>VISH1_1000</listOfCTIPorts>
    <startingDirectoryNumber>1000</startingDirectoryNumber>
    <devicePool>Default</devicePool>
    <dnCallingSearchSpace>None</dnCallingSearchSpace>
    <location>Hub_None</location>
    <partition>None</partition>
    <mediaResourceGroupList>None</mediaResourceGroupList>
    <voiceMailProfile>None</voiceMailProfile>
    <presenceGroup>Standard Presence group</presenceGroup>
    <AARGroup>None</AARGroup>
    <userHoldAudioSource>None</userHoldAudioSource>
  </portGroupConfiguration>

```

```

        <networkHoldAudioSource>None</networkHoldAudioSource>
        <callPickUpGroup>None</callPickUpGroup>
        <alertingNameAscii></alertingNameAscii>
        <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
        <requireDTMFReception>true</requireDTMFReception>
        <display></display>
        <externalPhoneMaskNumber></externalPhoneMaskNumber>
    </portGroupConfiguration>
</callControlGroup>

```

Create Call Control Group

HTTPS Method	POST
URI	https://<server>/adminapi/callControlGroup
Example URI	https://uccx-server/adminapi/callControlGroup
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201, 202, 206
HTTPS Failure Codes	400, 401, 404, 405, 412, 415, 500, 503
Error Codes	—

Sample Input XML—Create a New Call Control Group in WAN with All Parameters (POST) to Configure CTI Ports on Both Nodes

```

<?xml version="1.0" encoding="UTF-8"?>
<callControlGroup xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="../../../main/resources/xsd/callControlGroup.xsd">
  <outboundGroup>false</outboundGroup>
  <description>description</description>
  <deviceNamePrefix>c11</deviceNamePrefix>
  <numOfCTIPorts>10</numOfCTIPorts>
  <mediaTerminationSupport>false</mediaTerminationSupport>
  <portGroupConfiguration>
    <hostName>hostName1</hostName>
    <startingDirectoryNumber>100012</startingDirectoryNumber>
    <devicePool>Default</devicePool>
    <dnCallingSearchSpace>None</dnCallingSearchSpace>
    <location>Hub_None</location>
    <partition>None</partition>
    <mediaResourceGroupList>None</mediaResourceGroupList>
    <voiceMailProfile>None</voiceMailProfile>
    <presenceGroup>Standard Presence group</presenceGroup>
    <AARGroup>None</AARGroup>
    <userHoldAudioSource>None</userHoldAudioSource>
    <networkHoldAudioSource>None</networkHoldAudioSource>
    <callPickUpGroup>None</callPickUpGroup>
    <alertingNameAscii>alteringNameAscii</alertingNameAscii>
    <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
    <requireDTMFReception>true</requireDTMFReception>
    <display>display</display>
    <externalPhoneMaskNumber>12345</externalPhoneMaskNumber>
  </portGroupConfiguration>
</portGroupConfiguration>
  <hostName>hostName1</hostName>

```

```

    <startingDirectoryNumber>100012</startingDirectoryNumber>
    <devicePool>Default</devicePool>
    <dnCallingSearchSpace>None</dnCallingSearchSpace>
    <location>Hub_None</location>
    <partition>None</partition>
    <mediaResourceGroupList>None</mediaResourceGroupList>
    <voiceMailProfile>None</voiceMailProfile>
    <presenceGroup>Standard Presence group</presenceGroup>
    <AARGroup>None</AARGroup>
    <userHoldAudioSource>None</userHoldAudioSource>
    <networkHoldAudioSource>None</networkHoldAudioSource>
    <callPickUpGroup>None</callPickUpGroup>
    <alertingNameAscii>alteringNameAscii</alertingNameAscii>
    <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
    <requireDTMFReception>true</requireDTMFReception>
    <display>display</display>
    <externalPhoneMaskNumber>12345</externalPhoneMaskNumber>
  </portGroupConfiguration>
</callControlGroup>

```

Sample Input XML — Create a New Call Control Group in WAN with All Parameters (POST) to Configure CTI Ports on One Node

```

<?xml version="1.0" encoding="UTF-8"?>
<callControlGroup xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="../../../main/resources/xsd/callControlGroup.xsd">
  <outboundGroup>>false</outboundGroup>
  <description>description</description>
  <deviceNamePrefix>c11</deviceNamePrefix>
  <numOfCTIPorts>10</numOfCTIPorts>
  <mediaTerminationSupport>>false</mediaTerminationSupport>
  <portGroupConfiguration>
    <hostName>hostName1</hostName>
    <startingDirectoryNumber>100012</startingDirectoryNumber>
    <devicePool>Default</devicePool>
    <dnCallingSearchSpace>None</dnCallingSearchSpace>
    <location>Hub_None</location>
    <partition>None</partition>
    <mediaResourceGroupList>None</mediaResourceGroupList>
    <voiceMailProfile>None</voiceMailProfile>
    <presenceGroup>Standard Presence group</presenceGroup>
    <AARGroup>None</AARGroup>
    <userHoldAudioSource>None</userHoldAudioSource>
    <networkHoldAudioSource>None</networkHoldAudioSource>
    <callPickUpGroup>None</callPickUpGroup>
    <alertingNameAscii>alteringNameAscii</alertingNameAscii>
    <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
    <requireDTMFReception>true</requireDTMFReception>
    <display>display</display>
    <externalPhoneMaskNumber>12345</externalPhoneMaskNumber>
  </portGroupConfiguration>
</callControlGroup>

```

Sample Input XML—Create a New Call Control Group in WAN with Default Parameters (POST) to Configure CTI Ports on Both Nodes

```

<?xml version="1.0" encoding="UTF-8"?>
<?xml version="1.0" encoding="UTF-8"?>
<callControlGroup xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="../../../main/resources/xsd/callControlGroup.xsd">
  <outboundGroup>>false</outboundGroup>
  <description>description</description>
  <deviceNamePrefix>c1</deviceNamePrefix>

```

```

<numOfCTIPorts>10</numOfCTIPorts>
<mediaTerminationSupport>>false</mediaTerminationSupport>
<portGroupConfiguration>
  <startingDirectoryNumber>999001</startingDirectoryNumber>
  <devicePool>Default</devicePool>
  <location>Hub_None</location>
  <presenceGroup>Standard Presence group</presenceGroup>
  <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
  <requireDTMFReception>>true</requireDTMFReception>
  <hostName>ccx-94-80</hostName>
</portGroupConfiguration>
<portGroupConfiguration>
  <startingDirectoryNumber>999001</startingDirectoryNumber>
  <devicePool>Default</devicePool>
  <location>Hub_None</location>
  <presenceGroup>Standard Presence group</presenceGroup>
  <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
  <requireDTMFReception>>true</requireDTMFReception>
  <hostName>ccx-94-81</hostName>
</portGroupConfiguration>
</callControlGroup>

```

Sample Input XML—Create a New Call Control Group in WAN with Default Parameters (POST) to Configure CTI Ports on One Node

```

<?xml version="1.0" encoding="UTF-8"?>
<?xml version="1.0" encoding="UTF-8"?>
<callControlGroup xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="../../../main/resources/xsd/callControlGroup.xsd">
  <outboundGroup>>false</outboundGroup>
  <description>description</description>
  <deviceNamePrefix>c1</deviceNamePrefix>
  <numOfCTIPorts>10</numOfCTIPorts>
  <mediaTerminationSupport>>false</mediaTerminationSupport>
  <portGroupConfiguration>
    <startingDirectoryNumber>999001</startingDirectoryNumber>
    <devicePool>Default</devicePool>
    <location>Hub_None</location>
    <presenceGroup>Standard Presence group</presenceGroup>
    <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
    <requireDTMFReception>>true</requireDTMFReception>
    <hostName>ccx-94-80</hostName>
  </portGroupConfiguration>
</callControlGroup>

```

Sample Input XML—Create a New Call Control Group in LAN with All Parameters (POST)

```

<?xml version="1.0" encoding="UTF-8"?>
<callControlGroup xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="../../../main/resources/xsd/callControlGroup.xsd">
  <outboundGroup>>false</outboundGroup>
  <description>description</description>
  <deviceNamePrefix>c11</deviceNamePrefix>
  <numOfCTIPorts>10</numOfCTIPorts>
  <mediaTerminationSupport>>false</mediaTerminationSupport>
  <portGroupConfiguration>
    <hostName>hostName1</hostName>
    <startingDirectoryNumber>100012</startingDirectoryNumber>
    <devicePool>Default</devicePool>
    <dnCallingSearchSpace>None</dnCallingSearchSpace>
    <location>Hub_None</location>
    <partition>None</partition>
    <mediaResourceGroupList>None</mediaResourceGroupList>
  </portGroupConfiguration>
</callControlGroup>

```

```

    <voiceMailProfile>None</voiceMailProfile>
    <presenceGroup>Standard Presence group</presenceGroup>
    <AARGroup>None</AARGroup>
    <userHoldAudioSource>None</userHoldAudioSource>
    <networkHoldAudioSource>None</networkHoldAudioSource>
    <callPickUpGroup>None</callPickUpGroup>
    <alertingNameAscii>alteringNameAscii</alertingNameAscii>
    <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
    <requireDTMFReception>true</requireDTMFReception>
    <display>display</display>
    <externalPhoneMaskNumber>12345</externalPhoneMaskNumber>
  </portGroupConfiguration>
</callControlGroup>

```

Sample Input XML—Create a New Call Control Group in LAN with Default Parameters (POST)

```

<?xml version="1.0" encoding="UTF-8"?>
<callControlGroup xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="../../../../../main/resources/xsd/callControlGroup.xsd">
  <outboundGroup>false</outboundGroup>
  <description>description</description>
  <deviceNamePrefix>cl</deviceNamePrefix>
  <numOfCTIPorts>10</numOfCTIPorts>
  <mediaTerminationSupport>false</mediaTerminationSupport>
  <portGroupConfiguration>
    <startingDirectoryNumber>999001</startingDirectoryNumber>
    <devicePool>Default</devicePool>
    <location>Hub_None</location>
    <presenceGroup>Standard Presence group</presenceGroup>
    <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
    <requireDTMFReception>true</requireDTMFReception>
  </portGroupConfiguration>
</callControlGroup>

```

Modify Call Control Group

HTTPS Method	PUT
URI	https://<server>/adminapi/callControlGroup/{id}
Example URI	https://uccx-server/adminapi/callControlGroup/12
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200, 202, 206
HTTPS Failure Codes	400, 401, 404, 405, 412, 500, 503
Error Codes	—

Sample Input XML—Modify Common Parameters of an Existing Call Control Group in WAN when CTI Ports are Configured for Both Nodes (PUT)

```

<?xml version="1.0" encoding="UTF-8"?>
<callControlGroup xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="../../../../../main/resources/xsd/callControlGroup.xsd">
  <id>7</id>
  <outboundGroup>false</outboundGroup>
  <description>description</description>

```

```

<deviceNamePrefix>mod1</deviceNamePrefix>
<numOfCTIPorts>10</numOfCTIPorts>
<mediaTerminationSupport>true</mediaTerminationSupport>
<portGroupConfiguration>
  <startingDirectoryNumber>999001</startingDirectoryNumber>
  <devicePool>Default</devicePool>
  <location>Hub_None</location>
  <presenceGroup>Standard Presence group</presenceGroup>
  <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
  <requireDTMFReception>false</requireDTMFReception>
  <hostName>ccx-94-80</hostName>
</portGroupConfiguration>
<portGroupConfiguration>
  <startingDirectoryNumber>999001</startingDirectoryNumber>
  <devicePool>mod_DP</devicePool>
  <location>Hub_None</location>
  <presenceGroup>Standard Presence group</presenceGroup>
  <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
  <requireDTMFReception>true</requireDTMFReception>
  <hostName>ccx-94-81</hostName>
</portGroupConfiguration>
</callControlGroup>

```

Sample Input XML—Modify Node-Specific Parameters of an Existing Call Control Group for One Node in WAN when CTI Ports are Created for Both Nodes (PUT)

```

<?xml version="1.0" encoding="UTF-8"?>
<callControlGroup xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="../../../main/resources/xsd/callControlGroup.xsd">
  <id>7</id>
  <outboundGroup>false</outboundGroup>
  <description>description</description>
  <deviceNamePrefix>mod1</deviceNamePrefix>
  <numOfCTIPorts>10</numOfCTIPorts>
  <mediaTerminationSupport>true</mediaTerminationSupport>
  <portGroupConfiguration>
    <startingDirectoryNumber>999001</startingDirectoryNumber>
    <devicePool>Default</devicePool>
    <location>Hub_None</location>
    <presenceGroup>Standard Presence group</presenceGroup>
    <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
    <requireDTMFReception>false</requireDTMFReception>
    <hostName>ccx-94-80</hostName>
  </portGroupConfiguration>
</callControlGroup>

```

Sample Input XML—Modify an Existing Call Control Group in WAN with All Parameters (PUT)

```

<?xml version="1.0" encoding="UTF-8"?>
<callControlGroup xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="../../../main/resources/xsd/callControlGroup.xsd">
  <id>5</id>
  <outboundGroup>false</outboundGroup>
  <description>description</description>
  <deviceNamePrefix>c11</deviceNamePrefix>
  <numOfCTIPorts>10</numOfCTIPorts>
  <mediaTerminationSupport>false</mediaTerminationSupport>
  <portGroupConfiguration>
    <hostName>hostName1</hostName>
    <startingDirectoryNumber>100012</startingDirectoryNumber>
    <devicePool>mod_De</devicePool>
    <dnCallingSearchSpace>None</dnCallingSearchSpace>
    <location>Hub_None</location>
  </portGroupConfiguration>
</callControlGroup>

```

```

    <partition>Mod_Part</partition>
    <mediaResourceGroupList>None</mediaResourceGroupList>
    <voiceMailProfile>None</voiceMailProfile>
    <presenceGroup>Standard Presence group</presenceGroup>
    <AARGroup>None</AARGroup>
    <userHoldAudioSource>None</userHoldAudioSource>
    <networkHoldAudioSource>None</networkHoldAudioSource>
    <callPickUpGroup>None</callPickUpGroup>
    <alertingNameAscii>alteringNameAscii</alertingNameAscii>
    <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
    <requireDTMFReception>true</requireDTMFReception>
    <display>display</display>
    <externalPhoneMaskNumber>12345</externalPhoneMaskNumber>
  </portGroupConfiguration>
</portGroupConfiguration>
  <hostName>hostName1</hostName>
  <startingDirectoryNumber>100012</startingDirectoryNumber>
  <devicePool>Default</devicePool>
  <dnCallingSearchSpace>None</dnCallingSearchSpace>
  <location>Hub_None</location>
  <partition>Mod_Part</partition>
  <mediaResourceGroupList>Mod_MRGL</mediaResourceGroupList>
  <voiceMailProfile>None</voiceMailProfile>
  <presenceGroup>Standard Presence group</presenceGroup>
  <AARGroup>None</AARGroup>
  <userHoldAudioSource>None</userHoldAudioSource>
  <networkHoldAudioSource>None</networkHoldAudioSource>
  <callPickUpGroup>None</callPickUpGroup>
  <alertingNameAscii>alteringNameAscii</alertingNameAscii>
  <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
  <requireDTMFReception>true</requireDTMFReception>
  <display>display</display>
  <externalPhoneMaskNumber>12345</externalPhoneMaskNumber>
</portGroupConfiguration>
</callControlGroup>

```

Sample Input XML—Modify an Existing Call Control Group in LAN with All Parameters (PUT)

```

<?xml version="1.0" encoding="UTF-8"?>
<callControlGroup xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="../../../main/resources/xsd/callControlGroup.xsd">
  <id>23</id>
  <outboundGroup>>false</outboundGroup>
  <description>description</description>
  <deviceNamePrefix>c11</deviceNamePrefix>
  <numOfCTIPorts>10</numOfCTIPorts>
  <mediaTerminationSupport>>false</mediaTerminationSupport>
  <portGroupConfiguration>
    <hostName>hostName1</hostName>
    <startingDirectoryNumber>100012</startingDirectoryNumber>
    <devicePool>ModDevPool</devicePool>
    <dnCallingSearchSpace>None</dnCallingSearchSpace>
    <location>Hub_None</location>
    <partition>None</partition>
    <mediaResourceGroupList>None</mediaResourceGroupList>
    <voiceMailProfile>None</voiceMailProfile>
    <presenceGroup>Standard Presence group</presenceGroup>
    <AARGroup>None</AARGroup>
    <userHoldAudioSource>None</userHoldAudioSource>
    <networkHoldAudioSource>None</networkHoldAudioSource>
    <callPickUpGroup>None</callPickUpGroup>
    <alertingNameAscii>alteringNameAscii</alertingNameAscii>
    <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
    <requireDTMFReception>true</requireDTMFReception>
  </portGroupConfiguration>
</callControlGroup>

```



```

        <display>display</display>
        <externalPhoneMaskNumber>12345</externalPhoneMaskNumber>
    </portGroupConfiguration>
</callControlGroup>

```

Sample Input XML—Modify an Existing Call Control Group in LAN with Default Parameters (PUT)

```

<?xml version="1.0" encoding="UTF-8"?>
<?xml version="1.0" encoding="UTF-8"?>
<callControlGroup xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="../../../main/resources/xsd/callControlGroup.xsd">
    <id>24</id>
    <outboundGroup>>false</outboundGroup>
    <description>description</description>
    <deviceNamePrefix>mod</deviceNamePrefix>
    <numOfCTIPorts>10</numOfCTIPorts>
    <mediaTerminationSupport>>false</mediaTerminationSupport>
    <portGroupConfiguration>
        <startingDirectoryNumber>999001</startingDirectoryNumber>
        <devicePool>Default</devicePool>
        <location>Hub_None</location>
        <presenceGroup>Standard Presence group</presenceGroup>
        <redirectCallingSearchSpace>Redirect Party</redirectCallingSearchSpace>
        <requireDTMFReception>>true</requireDTMFReception>
    </portGroupConfiguration>
</callControlGroup>

```

Delete Call Control Group

HTTPS Method	DELETE
URI	https://<server>/adminapi/callControlGroup/{id}
Example URI	https://uccx-server/adminapi/callControlGroup/12
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200, 202, 206
HTTPS Failure Codes	401, 404, 405, 412, 500, 503
Error Codes	—

Check Status of POST/PUT/DELETE

HTTPS Method	GET
URI	https://<server>/adminapi/callControlGroup/{id}/status
Example URI	https://uccx-server:8080/adminapi/callControlGroup/12/status
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201, 206
HTTPS Failure Codes	401, 404, 410, 500, 503

Error Codes	—
--------------------	---



Attention After a POST, PUT or DELETE, perform a GET for the status of the job. If you do not perform a GET request within 5 minutes, a PUT or DELETE request on the same Call Control Group is not allowed.

Success Response

For Create

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccgJobStatus>
  <totalPorts>120</totalPorts>
  <updatePorts>5</updatePorts>
  <lastUpdatedPort>V11_840024</lastUpdatedPort>
  <type>created</type>
  <completed>true</completed>
</ccgJobStatus>
```



Note After the last port is created and the Call Control Group is saved, the completed field will be true.

For Modify

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccgJobStatus>
  <totalPorts>120</totalPorts>
  <updatePorts>5</updatePorts>
  <lastUpdatedPort>V11_840024</lastUpdatedPort>
  <type>modified</type>
  <completed>true</completed>
</ccgJobStatus>
```



Note After the last port is modified and the Call Control Group is saved, the completed field will be true.

For Delete

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccgJobStatus>
  <totalPorts>120</totalPorts>
  <updatePorts>5</updatePorts>
  <lastUpdatedPort>V11_840024</lastUpdatedPort>
  <type>deleted</type>
  <completed>true</completed>
</ccgJobStatus>
```



Note After the last port is deleted and the Call Control Group is saved, the completed field will be true.

Campaign

Campaigns are logical entities that group a set of contacts together in a dialing list. Campaigns deliver outgoing calls to agents. Typical applications include appointment and bill payment reminders.

If you select the IVR-based option for a campaign, the outbound calls are handled by the IVR scripts.

If you select the Agent-based option for a campaign, the outbound calls are handled by the available agents.

Data Structure for Campaign

Attribute Name	Base Type	Restrictions	Description
campaignId	baseIdInt	1 to 2147483647	Unique identifier for the campaign.
campaignName	string	minLength value = 1 maxLength value = 40 Campaign name contains alphanumeric characters, spaces, underscores (_), and hyphens (-) without leading or trailing white spaces.	Unique name for the campaign. Note PUT operation cannot modify the campaign name. If you try to modify the name, REST API throws 400 Bad Request error.
enabled	boolean	default = false	Indicates if the campaign is currently active.
description	string	minLength value = 0 maxLength value = 50	Description of the campaign.
startTime	timeType, string	Between 00:00 to 23:59 inclusive default = 08:00	Campaign start time.
endTime	timeType, string	Between 00:00 to 23:59 inclusive default = 21:00	Campaign end time.
timeZone	string	—	Time zone of the campaign. Note When you create a campaign, the time zone if given is ignored and the server time zone is considered by default. This field is read only and cannot be modified through PUT operation.

Attribute Name	Base Type	Restrictions	Description
campaignType	enum	enum values = AGENT, IVR	Type of campaign to be used for outbound calls. You can specify one of the following two campaign types: <ul style="list-style-type: none"> • Agent-based—All the outbound calls in the campaign are handled by the available agents. • IVR-based—All the outbound calls in the campaign are handled by the IVR scripts.
dialerType	enum	enum values = DIRECT_PREVIEW, PROGRESSIVE, PREDICTIVE	The dialer type options available for a campaign vary depending on the campaign type. <p>For Agent-based campaign type:</p> <ul style="list-style-type: none"> • Direct Preview • Progressive • Predictive <p>For IVR-based campaign type:</p> <ul style="list-style-type: none"> • Progressive • Predictive
pendingContacts	unsignedInt	default = 0	Indicates the number of contacts that are yet to be dialed, contacts that request a callback, and contacts that will be tried again because of unsuccessful prior attempts.
associatedSupervisors	sequence (supervisor : nameUrlPair)	minOccurs = 0 maxOccurs = 1 nillable = false For the supervisor element, minOccurs = 1 maxOccurs = 1 nillable = false	Indicates a campaign to which a supervisor is associated with.

Common Data Structure for IVR-Based and Agent-Based Campaigns

Attribute Name	Base Type	Restrictions	Description
----------------	-----------	--------------	-------------

campaignCallingNum	string	minLength value = 1 maxLength value = 16 The calling number can have up to 15 digits with an optional plus (+) prefix. Matches the E.164 standard.	The campaign calling number is the number that appears to the contact. This number is used by the dialer.
maxDialAttempts	unsignedInt	minInclusive value = 1 maxInclusive value = 3 default = 3	The maximum number of times to dial a contact beyond which the call status is marked as Closed.
callbackTimeLimit	unsignedInt	minInclusive value = 1 maxInclusive value = 60 default = 15	The duration before and after the scheduled callback time during which a callback is attempted.
treatLowVolAsVoice	boolean	default = true	Determines whether a low-volume call is treated as voice or disconnected.
noAnswerRingLimit	unsignedInt	minInclusive value = 1 maxInclusive value = 60 default = 15	The duration for which the Progressive/Predictive dialer allows the phone to ring before disconnecting an unanswered call.
abandonedCallWaitTime	unsignedInt	minInclusive value = 1 maxInclusive value = 10 default = 2	If the customer disconnects the call within the time that is here, then the call is classified as customer abandoned.
retryNoAnswerDelay	unsignedInt	minInclusive value = 0 maxInclusive value = 65535 default = 60	Duration for which the dialer waits before calling back a no-answer call.
retryBusySignalDelay	unsignedInt	minInclusive value = 0 maxInclusive value = 65535 default = 60	Duration for which the dialer waits before calling back a busy telephone number.
retryCustAbandonedDelay	unsignedInt	minInclusive value = 0 maxInclusive value = 65535 default = 0	If a customer abandons a call, the duration after which the dialer calls back the customer.

retryDialerAbandonedDelay	unsignedInt	minInclusive value = 0 maxInclusive value = 65535 default = 0	If the dialer abandons a call, the duration after which the dialer calls back the customer.
---------------------------	-------------	---	---

Common Data Structure for Outbound IVR - Progressive and Predictive Campaigns

Attribute Name	Base Type	Restrictions	Description
applicationTrigger	nameUriPair	—	The JTAPI trigger that is associated with the campaign.
numDedicatedPorts	unsignedInt	—	Number of dedicated IVR ports that you want to reserve for this campaign.
linesPerPort	float	minInclusive value = 1.0 maxInclusive value = 3.0 default = 1.0	Number of lines to be dialed for each port.

Data Structure for Outbound IVR - Predictive Campaign

Attribute Name	Base Type	Restrictions	Description
maxLinesPerPort	float	minInclusive = 1.0 maxInclusive = 3.0 default = 3.0	Maximum number of lines to be dialed for each port.
predictiveCorrectionPace	unsignedInt	minInclusive = 10 maxInclusive = 1000 default = 100	The number of calls that were answered by live voice that the predictive algorithm considers for each iteration.
predictiveGain	float	minExclusive = 0.0 maxInclusive = 1.0 default = 1.0	Controls the size of the lines-per-port corrections.
callAbandonLimit	float	minExclusive = 0.0 maxInclusive = 100.0 default = 3.0	Call abandon percentage.

Data Structure for Preview Campaign

Attribute Name	Base Type	Restrictions	Description
maxDialAttempts	unsignedInt	minInclusive = 1 maxInclusive = 3 default = 3	The maximum number of times a contact is dialed.
cacheSize	unsignedInt	minInclusive = 1 maxInclusive = 100 default = 20	The number of contact records that are retrieved from the database in bulk for dialing.
ansMachineRetry	boolean	default = false	If this attribute is enabled, the phone number is dialed again at a later time if an answering machine is reached the first time.
callbackTimeLimit	unsignedInt	minInclusive = 1 maxInclusive = 60 default = 15	The customer-specified callback time.
missedCallbackAction	string	enum values = NEXT_DAY, RETRY, CLOSE default = NEXT_DAY	The callback action if a callback is missed.
assignedCSQs	string	minLength value = 1	The Contact Service Queues (CSQs) from which agents are selected for outbound calls for the campaign.

Common Data Structure for Outbound Agent - Progressive and Predictive Campaigns

Attribute Name	Base Type	Restrictions	Description
linesPerAgent	float	minInclusive value = 1.0 maxInclusive value = 3.0 default = 1.0	Number of lines to be dialed for each agent.
assignedCSQs	string	minLength value = 1	The Contact Service Queues (CSQs) from which agents are selected for outbound calls for the campaign.

Attribute Name	Base Type	Restrictions	Description
ansMachineTreatment	string	enum values = TRANSFER_TO_IVR, END_CALL default = TRANSFER_TO_IVR	Specifies how to handle an outbound call if an answering machine is reached.
answerMachineTreatmentTrigger	nameUriPair	minOccurs= 0	The JTAPI trigger that is associated with the campaign. Note Application trigger is used only when you choose the transfer to IVR option for an outbound call.
abandonedCallTreatment	string	enum values = TRANSFER_TO_IVR, ABANDON_CALL default = TRANSFER_TO_IVR	Specifies how to handle an outbound call if agents are not available.
abandonedCallTreatmentTrigger	nameUriPair	minOccurs= 0	The JTAPI trigger that is associated with the campaign. Note Application trigger is used only when you choose the transfer to IVR option for an outbound call.
missedCallbackAction	string	enum values = NEXT_DAY, RETRY, CLOSE default = NEXT_DAY	The callback action if a callback is missed.

Data Structure for Outbound Agent - Predictive Campaign

Attribute Name	Base Type	Restrictions	Description
maxLinesPerAgent	float	minInclusive = 1.0 maxInclusive = 3.0 default = 3.0	Maximum number of lines to be dialed for each agent.
predictiveCorrectionPace	unsignedInt	minInclusive = 10 maxInclusive = 1000 default = 100	The number of calls that were answered by live voice that the predictive algorithm considers for each iteration.

Attribute Name	Base Type	Restrictions	Description
predictiveGain	float	minExclusive = 0.0 maxInclusive = 1.0 default = 1.0	Controls the size of the lines-per-port corrections.
callAbandonLimit	float	minExclusive = 0.0 maxInclusive = 100.0 default = 3.0	Call abandon percentage.

Get List of Campaigns

HTTPS Method	GET
URI	https://<server>/adminapi/campaign
Example URI	https://uccx-server/adminapi/campaign
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<campaigns>
  <campaign>
    <self>https://uccx-server/adminapi/campaign/202</self>
    <campaignId>202</campaignId>
    <campaignName>IVRPredictive</campaignName>
    <enabled>>false</enabled>
    <description>This is IVR Predictive campaign</description>
    <startTime>08:00</startTime>
    <endTime>21:00</endTime>
    <timeZone>India Standard Time</timeZone>
    <campaignType>IVR</campaignType>
    <dialerType>PREDICTIVE</dialerType>
    <pendingContacts>0</pendingContacts>
  </campaign>
  <campaign>
    <self>https://uccx-server/adminapi/campaign/201</self>
    <campaignId>201</campaignId>
    <campaignName>IVRProgressive</campaignName>
    <enabled>>false</enabled>
    <description>This is IVR Progressive campaign</description>
    <startTime>08:00</startTime>
    <endTime>21:00</endTime>
    <timeZone>India Standard Time</timeZone>
    <campaignType>IVR</campaignType>
    <dialerType>PROGRESSIVE</dialerType>
  </campaign>
</campaigns>
```

```

    <pendingContacts>0</pendingContacts>
  </campaign>
</campaign>
<campaign>
  <self>https://uccx-server/adminapi/campaign/200</self>
  <campaignId>200</campaignId>
  <campaignName>OBPreview</campaignName>
  <enabled>>false</enabled>
  <description>This is Outbound Preview campaign</description>
  <startTime>08:00</startTime>
  <endTime>21:00</endTime>
  <timeZone>India Standard Time</timeZone>
  <campaignType>AGENT</campaignType>
  <dialerType>DIRECT_PREVIEW</dialerType>
  <pendingContacts>0</pendingContacts>
</campaign>
<campaign>
  <self>https://uccx-server/adminapi/campaign/200</self>
  <campaignId>200</campaignId>
  <campaignName>Agent progressive</campaignName>
  <enabled>>false</enabled>
  <description>This is Outbound Preview campaign</description>
  <startTime>08:00</startTime>
  <endTime>21:00</endTime>
  <timeZone>India Standard Time</timeZone>
  <campaignType>AGENT</campaignType>
  <dialerType>PROGRESSIVE</dialerType>
  <pendingContacts>0</pendingContacts>
</campaign>
<campaign>
  <self>https://uccx-server/adminapi/campaign/200</self>
  <campaignId>200</campaignId>
  <campaignName>Agent predictive</campaignName>
  <enabled>>false</enabled>
  <description>This is Outbound Preview campaign</description>
  <startTime>08:00</startTime>
  <endTime>21:00</endTime>
  <timeZone>India Standard Time</timeZone>
  <campaignType>AGENT</campaignType>
  <dialerType>PREDICTIVE</dialerType>
  <pendingContacts>0</pendingContacts>
</campaign>
</campaigns>

```

Get Campaign

HTTPS Method	GET
URI	https://<server>/adminapi/campaign/{id}
Example URI	https://uccx-server/adminapi/campaign/200
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<campaign>
  <self>https://uccx-server/adminapi/campaign/200</self>
  <campaignId>200</campaignId>
  <campaignName>OBPreview</campaignName>
  <enabled>>false</enabled>
  <description>This is Outbound Preview campaign</description>
  <startTime>08:00</startTime>
  <endTime>21:00</endTime>
  <timeZone>India Standard Time</timeZone>
  <campaignType>AGENT</campaignType>
  <dialerType>DIRECT_PREVIEW</dialerType>
  <pendingContacts>0</pendingContacts>
  <typeSpecificInfo>
    <obPreview>
      <maxDialAttempts>3</maxDialAttempts>
      <cacheSize>20</cacheSize>
      <ansMachineRetry>>false</ansMachineRetry>
      <callbackTimeLimit>15</callbackTimeLimit>
      <missedCallbackAction>NEXT_DAY</missedCallbackAction>
      <assignedCSQs>
        <csq name="1">
          <refURL>https://uccx-server/adminapi/csq/1</refURL>
        </csq>
      </assignedCSQs>
    </obPreview>
  </typeSpecificInfo>
</campaign>
```

If a campaign is associated with a supervisor, the sample response would be as below:

```
<?xml version="1.0" encoding="UTF-8"?>
<campaign>
  <self></self>
  <campaignId>200</campaignId>
  <campaignName>OBPreview</campaignName>
  <enabled>>false</enabled>
  <description>This is Outbound Preview campaign</description>
  <startTime>08:00</startTime>
  <endTime>21:00</endTime>
  <timeZone>India Standard Time</timeZone>
  <campaignType>AGENT</campaignType>
  <dialerType>DIRECT_PREVIEW</dialerType>
  <pendingContacts>0</pendingContacts>
  <typeSpecificInfo>
    <obPreview>
      <maxDialAttempts>3</maxDialAttempts>
      <cacheSize>20</cacheSize>
      <ansMachineRetry>>false</ansMachineRetry>
      <callbackTimeLimit>15</callbackTimeLimit>
      <missedCallbackAction>NEXT_DAY</missedCallbackAction>
      <assignedCSQs>
        <csq name="1">
          <refURL>https://uccx-server/adminapi/csq/1</refURL>
        </csq>
      </assignedCSQs>
    </obPreview>
  </typeSpecificInfo>
  <associatedSupervisors>
    <supervisor name="OBAgent101">
      <refURL>https://uccx-server/adminapi/csq/1</refURL>
    </supervisor>
  </associatedSupervisors>
</campaign>
```

```

    </associatedSupervisors>
  </campaign>

```

Create Campaign

HTTPS Method	POST
URI	https://<server>/adminapi/campaign
Example URI	https://uccx-server/adminapi/campaign
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	400, 401, 405
Error Codes	—



Note When Cisco Finesse is deactivated, new agent predictive and agent progressive campaigns cannot be created.

Success Response

HTTPS/1.1 201 Created

Sample Input XML—Outbound IVR - Progressive Campaign

```
<?xml version="1.0" encoding="UTF-8"?>
<campaign>
  <self>https://uccx-server/adminapi/campaign/201</self>
  <campaignId>201</campaignId>
  <campaignName>IVRProgressive</campaignName>
  <enabled>>false</enabled>
  <description>This is IVR Progressive campaign</description>
  <startTime>08:00</startTime>
  <endTime>21:00</endTime>
  <timeZone>India Standard Time</timeZone>
  <campaignType>IVR</campaignType>
  <dialerType>PROGRESSIVE</dialerType>
  <pendingContacts>0</pendingContacts>
  <typeSpecificInfo>
    <obIvrProgressive>
      <campaignCallingNum>12345</campaignCallingNum>
      <applicationTrigger name="5657656">
        <refURL>https://uccx-server/adminapi/trigger/5657656</refURL>
      </applicationTrigger>
      <maxDialAttempts>3</maxDialAttempts>
      <callbackTimeLimit>15</callbackTimeLimit>
      <numDedicatedPorts>0</numDedicatedPorts>
      <linesPerPort>1.0</linesPerPort>
      <treatLowVolAsVoice>>true</treatLowVolAsVoice>
      <ansMachineTreatment>TRANSFER_TO_IVR</ansMachineTreatment>
      <noAnswerRingLimit>15</noAnswerRingLimit>
      <abandonedCallWaitTime>2</abandonedCallWaitTime>
      <retryNoAnswerDelay>60</retryNoAnswerDelay>
      <retryBusySignalDelay>60</retryBusySignalDelay>
      <retryCustAbandonedDelay>30</retryCustAbandonedDelay>
      <retryDialerAbandonedDelay>60</retryDialerAbandonedDelay>
    </obIvrProgressive>
  </typeSpecificInfo>
</campaign>
```

Sample Input XML—Outbound IVR - Predictive Campaign

```

<?xml version="1.0" encoding="UTF-8"?>
<campaign>
  <self>https://uccx-server/adminapi/campaign/202</self>
  <campaignId>202</campaignId>
  <campaignName>IVRPredictive</campaignName>
  <enabled>>false</enabled>
  <description>This is IVR Predictive campaign</description>
  <startTime>08:00</startTime>
  <endTime>21:00</endTime>
  <timeZone>India Standard Time</timeZone>
  <campaignType>IVR</campaignType>
  <dialerType>PREDICTIVE</dialerType>
  <pendingContacts>0</pendingContacts>
  <typeSpecificInfo>
    <obIvrPredictive>
      <campaignCallingNum>112233</campaignCallingNum>
      <applicationTrigger name="464355">
        <refURL>https://uccx-server/adminapi/trigger/464355</refURL>
      </applicationTrigger>
      <maxDialAttempts>3</maxDialAttempts>
      <callbackTimeLimit>15</callbackTimeLimit>
      <numDedicatedPorts>0</numDedicatedPorts>
      <linesPerPort>1.5</linesPerPort>
      <treatLowVolAsVoice>>true</treatLowVolAsVoice>
      <ansMachineTreatment>TRANSFER_TO_IVR</ansMachineTreatment>
      <noAnswerRingLimit>15</noAnswerRingLimit>
      <abandonedCallWaitTime>2</abandonedCallWaitTime>
      <retryNoAnswerDelay>60</retryNoAnswerDelay>
      <retryBusySignalDelay>60</retryBusySignalDelay>
      <retryCustAbandonedDelay>30</retryCustAbandonedDelay>
      <retryDialerAbandonedDelay>60</retryDialerAbandonedDelay>
      <maxLinesPerPort>3.0</maxLinesPerPort>
      <predictiveCorrectionPace>100</predictiveCorrectionPace>
      <predictiveGain>1.0</predictiveGain>
      <callAbandonLimit>3.0</callAbandonLimit>
    </obIvrPredictive>
  </typeSpecificInfo>
</campaign>

```

Sample Input XML—Preview Campaign

```
<?xml version="1.0" encoding="UTF-8"?>
<campaign>
  <self>https://uccx-server/adminapi/campaign/200</self>
  <campaignId>200</campaignId>
  <campaignName>OBPreview</campaignName>
  <enabled>>false</enabled>
  <description>This is Outbound Preview campaign</description>
  <startTime>08:00</startTime>
  <endTime>21:00</endTime>
  <timeZone>India Standard Time</timeZone>
  <campaignType>AGENT</campaignType>
  <dialerType>DIRECT_PREVIEW</dialerType>
  <pendingContacts>0</pendingContacts>
  <typeSpecificInfo>
    <obPreview>
      <maxDialAttempts>3</maxDialAttempts>
      <cacheSize>20</cacheSize>
      <ansMachineRetry>>false</ansMachineRetry>
      <callbackTimeLimit>15</callbackTimeLimit>
      <missedCallbackAction>NEXT_DAY</missedCallbackAction>
      <assignedCSQs>
        <csq name="1">
          <refURL>https://uccx-server/adminapi/csq/1</refURL>
        </csq>
      </assignedCSQs>
    </obPreview>
  </typeSpecificInfo>
</campaign>
```

Sample Input XML—Outbound Agent - Progressive Campaign

```

<?xml version="1.0" encoding="UTF-8"?>
<campaign>
  <self>https://uccx-server/adminapi/campaign/105</self>
  <campaignId>105</campaignId>
  <campaignName>Agent Progressive</campaignName>
  <enabled>>false</enabled>
  <description>description</description>
  <startTime>08:00</startTime>
  <endTime>21:00</endTime>
  <timeZone>India Standard Time</timeZone>
  <campaignType>AGENT</campaignType>
  <dialerType>PROGRESSIVE</dialerType>
  <pendingContacts>0</pendingContacts>
  <typeSpecificInfo>
    <obAgentProgressive>
      <campaignCallingNum>7777</campaignCallingNum>
      <maxDialAttempts>3</maxDialAttempts>
      <callbackTimeLimit>15</callbackTimeLimit>
      <treatLowVolAsVoice>>true</treatLowVolAsVoice>
      <noAnswerRingLimit>15</noAnswerRingLimit>
      <abandonedCallWaitTime>2</abandonedCallWaitTime>
      <retryNoAnswerDelay>60</retryNoAnswerDelay>
      <retryBusySignalDelay>60</retryBusySignalDelay>
      <retryCustAbandonedDelay>0</retryCustAbandonedDelay>
      <retryDialerAbandonedDelay>0</retryDialerAbandonedDelay>
      <linesPerAgent>1.0</linesPerAgent>
      <assignedCSQs>
        </csq>
        <csq name="37">
          <refURL>https://uccx-server/adminapi/csq/37</refURL>
        </csq>
      </assignedCSQs>
      <answerMachineTreatment>TRANSFER_TO_IVR</answerMachineTreatment>
      <answerMachineTreatmentTrigger name="4600">
        <refURL>https://uccx-server/adminapi/trigger/4600</refURL>
      </answerMachineTreatmentTrigger>
      <abandonedCallTreatment>TRANSFER_TO_IVR</abandonedCallTreatment>
      <abandonedCallTreatmentTrigger name="4800">
        <refURL>https://uccx-server/adminapi/trigger/4800</refURL>
      </abandonedCallTreatmentTrigger>
      <missedCallbackAction>NEXT_DAY</missedCallbackAction>
    </obAgentProgressive>
  </typeSpecificInfo>
</campaign>

```


Sample Input XML—Outbound Agent - Predictive Campaign

```
<?xml version="1.0" encoding="UTF-8"?>
<campaign>
  <self>https://uccx-server/adminapi/campaign/114</self>
  <campaignId>114</campaignId>
  <campaignName>Agent predictive</campaignName>
  <enabled>>false</enabled>
  <description>description</description>
  <startTime>08:00</startTime>
  <endTime>21:00</endTime>
  <timeZone>India Standard Time</timeZone>
  <campaignType>AGENT</campaignType>
  <dialerType>PREDICTIVE</dialerType>
  <pendingContacts>0</pendingContacts>
  <typeSpecificInfo>
    <obAgentPredictive>
      <campaignCallingNum>7777</campaignCallingNum>
      <maxDialAttempts>3</maxDialAttempts>
      <callbackTimeLimit>15</callbackTimeLimit>
      <treatLowVolAsVoice>>true</treatLowVolAsVoice>
      <noAnswerRingLimit>15</noAnswerRingLimit>
      <abandonedCallWaitTime>2</abandonedCallWaitTime>
      <retryNoAnswerDelay>60</retryNoAnswerDelay>
      <retryBusySignalDelay>60</retryBusySignalDelay>
      <retryCustAbandonedDelay>0</retryCustAbandonedDelay>
      <retryDialerAbandonedDelay>0</retryDialerAbandonedDelay>
      <linesPerAgent>2.0</linesPerAgent>
      <assignedCSQs>
        </csq>
        <csq name="37">
          <refURL>https://uccx-server/adminapi/csq/37</refURL>
        </csq>
      </assignedCSQs>
      <answerMachineTreatment>TRANSFER_TO_IVR</answerMachineTreatment>
      <answerMachineTreatmentTrigger name="4600">
        <refURL>https://uccx-server/adminapi/trigger/4600</refURL>
      </answerMachineTreatmentTrigger>
      <abandonedCallTreatment>TRANSFER_TO_IVR</abandonedCallTreatment>
      <abandonedCallTreatmentTrigger name="4800">
        <refURL>https://uccx-server/adminapi/trigger/4800</refURL>
      </abandonedCallTreatmentTrigger>
      <missedCallbackAction>NEXT_DAY</missedCallbackAction>
      <maxLinesPerAgent>2.0</maxLinesPerAgent>
      <predictiveCorrectionPace>100</predictiveCorrectionPace>
      <predictiveGain>1.0</predictiveGain>
      <callAbandonLimit>0.0</callAbandonLimit>
    </obAgentPredictive>
  </typeSpecificInfo>
</campaign>
```

Modify Campaign

HTTPS Method	PUT
URI	https://<server>/adminapi/campaign/{id}
Example URI	https://uccx-server/adminapi/campaign/200
Content Type	Application/XML, Application/JSON

HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Sample Input XML—Outbound IVR - Progressive Campaign

```
<?xml version="1.0" encoding="UTF-8"?>
<campaign>
  <self>https://uccx-server/adminapi/campaign/201</self>
  <campaignId>201</campaignId>
  <campaignName>IVRProgressive</campaignName>
  <enabled>>false</enabled>
  <description>This is IVR Progressive campaign</description>
  <startTime>08:00</startTime>
  <endTime>21:00</endTime>
  <timeZone>India Standard Time</timeZone>
  <campaignType>IVR</campaignType>
  <dialerType>PROGRESSIVE</dialerType>
  <pendingContacts>0</pendingContacts>
  <typeSpecificInfo>
    <obIvrProgressive>
      <campaignCallingNum>12345</campaignCallingNum>
      <applicationTrigger name="5657656">
        <refURL>https://uccx-server/adminapi/trigger/5657656</refURL>
      </applicationTrigger>
      <maxDialAttempts>3</maxDialAttempts>
      <callbackTimeLimit>15</callbackTimeLimit>
      <numDedicatedPorts>0</numDedicatedPorts>
      <linesPerPort>1.0</linesPerPort>
      <treatLowVolAsVoice>>true</treatLowVolAsVoice>
      <ansMachineTreatment>TRANSFER_TO_IVR</ansMachineTreatment>
      <noAnswerRingLimit>15</noAnswerRingLimit>
      <abandonedCallWaitTime>2</abandonedCallWaitTime>
      <retryNoAnswerDelay>60</retryNoAnswerDelay>
      <retryBusySignalDelay>60</retryBusySignalDelay>
      <retryCustAbandonedDelay>30</retryCustAbandonedDelay>
      <retryDialerAbandonedDelay>60</retryDialerAbandonedDelay>
    </obIvrProgressive>
  </typeSpecificInfo>
</campaign>
```

Sample Input XML—Outbound IVR - Predictive Campaign

```

<?xml version="1.0" encoding="UTF-8"?>
<campaign>
  <self>https://uccx-server/adminapi/campaign/202</self>
  <campaignId>202</campaignId>
  <campaignName>IVRPredictive</campaignName>
  <enabled>>false</enabled>
  <description>This is IVR Predictive campaign</description>
  <startTime>08:00</startTime>
  <endTime>21:00</endTime>
  <timeZone>India Standard Time</timeZone>
  <campaignType>IVR</campaignType>
  <dialerType>PREDICTIVE</dialerType>
  <pendingContacts>0</pendingContacts>
  <typeSpecificInfo>
    <obIvrPredictive>
      <campaignCallingNum>112233</campaignCallingNum>
      <applicationTrigger name="464355">
        <refURL>https://uccx-server/adminapi/trigger/464355</refURL>
      </applicationTrigger>
      <maxDialAttempts>3</maxDialAttempts>
      <callbackTimeLimit>15</callbackTimeLimit>
      <numDedicatedPorts>0</numDedicatedPorts>
      <linesPerPort>1.5</linesPerPort>
      <treatLowVolAsVoice>>true</treatLowVolAsVoice>
      <ansMachineTreatment>TRANSFER_TO_IVR</ansMachineTreatment>
      <noAnswerRingLimit>15</noAnswerRingLimit>
      <abandonedCallWaitTime>2</abandonedCallWaitTime>
      <retryNoAnswerDelay>60</retryNoAnswerDelay>
      <retryBusySignalDelay>60</retryBusySignalDelay>
      <retryCustAbandonedDelay>30</retryCustAbandonedDelay>
      <retryDialerAbandonedDelay>60</retryDialerAbandonedDelay>
      <maxLinesPerPort>3.0</maxLinesPerPort>
      <predictiveCorrectionPace>100</predictiveCorrectionPace>
      <predictiveGain>1.0</predictiveGain>
      <callAbandonLimit>3.0</callAbandonLimit>
    </obIvrPredictive>
  </typeSpecificInfo>
</campaign>

```

Sample Input XML—Preview Campaign

```

<?xml version="1.0" encoding="UTF-8"?>
<campaign>
  <self>https://uccx-server/adminapi/campaign/200</self>
  <campaignId>200</campaignId>
  <campaignName>OBPreview</campaignName>
  <enabled>>false</enabled>
  <description>This is Outbound Preview campaign</description>
  <startTime>08:00</startTime>
  <endTime>21:00</endTime>
  <timeZone>India Standard Time</timeZone>
  <campaignType>AGENT</campaignType>
  <dialerType>DIRECT_PREVIEW</dialerType>
  <pendingContacts>0</pendingContacts>
  <typeSpecificInfo>
    <obPreview>
      <maxDialAttempts>3</maxDialAttempts>
      <cacheSize>20</cacheSize>
      <ansMachineRetry>>false</ansMachineRetry>
      <callbackTimeLimit>15</callbackTimeLimit>
      <missedCallbackAction>NEXT_DAY</missedCallbackAction>
      <assignedCSQs>
        <csq name="1">
          <refURL>https://uccx-server/adminapi/csq/1</refURL>
        </csq>
      </assignedCSQs>
    </obPreview>
  </typeSpecificInfo>
</campaign>

```

Sample Input XML—Outbound Agent - Progressive Campaign

```
<?xml version="1.0" encoding="UTF-8"?>
<campaign>
  <self>https://uccx-server/adminapi/campaign/105</self>
  <campaignId>105</campaignId>
  <campaignName>Agent Progressive</campaignName>
  <enabled>>false</enabled>
  <description>description</description>
  <startTime>08:00</startTime>
  <endTime>21:00</endTime>
  <timeZone>India Standard Time</timeZone>
  <campaignType>AGENT</campaignType>
  <dialerType>PROGRESSIVE</dialerType>
  <pendingContacts>0</pendingContacts>
  <typeSpecificInfo>
    <obAgentProgressive>
      <campaignCallingNum>7777</campaignCallingNum>
      <maxDialAttempts>3</maxDialAttempts>
      <callbackTimeLimit>15</callbackTimeLimit>
      <treatLowVolAsVoice>>true</treatLowVolAsVoice>
      <noAnswerRingLimit>15</noAnswerRingLimit>
      <abandonedCallWaitTime>2</abandonedCallWaitTime>
      <retryNoAnswerDelay>60</retryNoAnswerDelay>
      <retryBusySignalDelay>60</retryBusySignalDelay>
      <retryCustAbandonedDelay>0</retryCustAbandonedDelay>
      <retryDialerAbandonedDelay>0</retryDialerAbandonedDelay>
      <linesPerAgent>1.0</linesPerAgent>
      <assignedCSQs>
        </csq>
        <csq name="37">
          <refURL>https://uccx-server/adminapi/csq/37</refURL>
        </csq>
      </assignedCSQs>
      <answerMachineTreatment>TRANSFER_TO_IVR</answerMachineTreatment>
      <answerMachineTreatmentTrigger name="4600">
        <refURL>https://uccx-server/adminapi/trigger/4600</refURL>
      </answerMachineTreatmentTrigger>
      <abandonedCallTreatment>TRANSFER_TO_IVR</abandonedCallTreatment>
      <abandonedCallTreatmentTrigger name="4800">
        <refURL>https://uccx-server/adminapi/trigger/4800</refURL>
      </abandonedCallTreatmentTrigger>
      <missedCallbackAction>NEXT_DAY</missedCallbackAction>
    </obAgentProgressive>
  </typeSpecificInfo>
</campaign>
```

Sample Input XML—Outbound Agent - Predictive Campaign

```
<?xml version="1.0" encoding="UTF-8"?>
<campaign>
  <self>https://uccx-server/adminapi/campaign/114</self>
  <campaignId>114</campaignId>
  <campaignName>Agent predictive</campaignName>
  <enabled>>false</enabled>
  <description>description</description>
  <startTime>08:00</startTime>
  <endTime>21:00</endTime>
  <timeZone>India Standard Time</timeZone>
  <campaignType>AGENT</campaignType>
  <dialerType>PREDICTIVE</dialerType>
  <pendingContacts>0</pendingContacts>
  <typeSpecificInfo>
    <obAgentPredictive>
      <campaignCallingNum>7777</campaignCallingNum>
      <maxDialAttempts>3</maxDialAttempts>
      <callbackTimeLimit>15</callbackTimeLimit>
      <treatLowVolAsVoice>true</treatLowVolAsVoice>
      <noAnswerRingLimit>15</noAnswerRingLimit>
      <abandonedCallWaitTime>2</abandonedCallWaitTime>
      <retryNoAnswerDelay>60</retryNoAnswerDelay>
      <retryBusySignalDelay>60</retryBusySignalDelay>
      <retryCustAbandonedDelay>0</retryCustAbandonedDelay>
      <retryDialerAbandonedDelay>0</retryDialerAbandonedDelay>
      <linesPerAgent>2.0</linesPerAgent>
      <assignedCSQs>
        </csq>
        <csq name="37">
          <refURL>https://uccx-server/adminapi/csq/37</refURL>
        </csq>
      </assignedCSQs>
      <answerMachineTreatment>TRANSFER_TO_IVR</answerMachineTreatment>
      <answerMachineTreatmentTrigger name="4600">
        <refURL>https://uccx-server/adminapi/trigger/4600</refURL>
      </answerMachineTreatmentTrigger>
      <abandonedCallTreatment>TRANSFER_TO_IVR</abandonedCallTreatment>
      <abandonedCallTreatmentTrigger name="4800">
        <refURL>https://uccx-server/adminapi/trigger/4800</refURL>
      </abandonedCallTreatmentTrigger>
      <missedCallbackAction>NEXT_DAY</missedCallbackAction>
      <maxLinesPerAgent>2.0</maxLinesPerAgent>
      <predictiveCorrectionPace>100</predictiveCorrectionPace>
      <predictiveGain>1.0</predictiveGain>
      <callAbandonLimit>0.0</callAbandonLimit>
    </obAgentPredictive>
  </typeSpecificInfo>
</campaign>
```

Delete Campaign

HTTPS Method	DELETE
URI	https://<server>/adminapi/campaign/{id}
Example URI	https://uccx-server/adminapi/campaign/200
Content Type	Application/XML, Application/JSON

HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—



Note If the campaign that you delete is the only campaign that is associated with supervisors, the respective supervisors will lose the outbound campaign management capability.

Success Response

HTTPS/1.1 200 OK

Modify Campaign State

HTTPS Method	PUT
URI	https://<server>/adminapi/campaign/{id}/state/{enable or disable}
Example URI	https://uccx-server/adminapi/campaign/200/state/enable https://uccx-server/adminapi/campaign/200/state/disable
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Code	400, 404, 500
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Campaign Contacts

Campaign contacts are contacts that are associated with the campaigns as campaign members.

Data Structure for Campaign Contacts

Attribute Name	Base Type	Restrictions	Description
campaign	nameUriPair	—	The campaign to which the contacts are associated.

Attribute Name	Base Type	Restrictions	Description
allowDuplicateContacts	boolean	This is an optional attribute. Only one occurrence of this attribute is permitted.	Importing of duplicate contacts is permitted. Duplicate contacts can be in the same contact file or it can be in different contact files.
csvdata	string	<ul style="list-style-type: none"> • The first non-empty line is treated as header line. • The following case-insensitive headers are supported: <ul style="list-style-type: none"> • Account Number • First Name • Last Name • Phone1 • Phone2 • Phone3 • Dial Time • Phone1 is mandatory and must be specified. • Dial Time field is applicable only for UCCX Agent Direct Preview campaigns. • Tomcat has a 3 MB size restriction, so GET operation returns a maximum of 10,000 records. 	List of contacts with relevant information for each contact in a comma-separated text file.

Get List of Pending Contacts for Campaign

HTTPS Method	GET
URI	https://<server>/adminapi/campaign/{id}/contacts
Example URI	https://uccx-server/adminapi/campaign/200/contacts
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405

Error Codes	—
--------------------	---

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<campaignContacts>
  <campaign name="previewCampaign">
    <link href="https://uccx-server/adminapi/campaign/932"
      rel="campaign" type="campaign"/>
  </campaign>
  <csvdata>account number, first name, last name, phone1,
    phone2, phone3, dial time

    , , , 1000, , , 10:45
    , , , 1001, , , 10:45
    , , , 1002, , , 10:45
    , , , 1003, , , 10:45
    , , , 1004, , , 10:45
  </csvdata>
</campaignContacts>

<?xml version="1.0" encoding="UTF-8"?>
<campaignContacts>
  <campaign name="previewCampaign">
    <refURL>https://uccx-server/adminapi/campaign/932</refURL>
  </campaign>
  <csvdata>account number, first name, last name, phone1,
    phone2, phone3, dial time, call result, callback number, call status, call result
    01, call result 02, call result 03, last number dialled, calls made to phone 01, calls made
    to phone 02, calls made to phone 03, retry, num missed call back

    , , , 1000, , , 10:45, 0, , 5, 0, 0, 0, 0, 0, 0, 0, false, 0
    , , , 1001, , , 10:45, 0, , 5, 0, 0, 0, 0, 0, 0, 0, false, 0
    , , , 1002, , , 10:45, 0, , 5, 0, 0, 0, 0, 0, 0, 0, false, 0
    , , , 1003, , , 10:45, 0, , 5, 0, 0, 0, 0, 0, 0, 0, false, 0
    , , , 1004, , , 10:45, 0, , 5, 0, 0, 0, 0, 0, 0, 0, false, 0
  </csvdata>
</campaignContacts>
```

Get List of Pending Contacts for Campaign with Specific Call Status

This API allows you to provide a specific call status to list the pending contacts for a campaign that exists in the system. The following call statuses are supported:

- PENDING
- CALLBACK
- RETRY
- RETRIES_WITH_DELAY
- MAX_CALLS
- UNKNOWN

HTTPS Method	GET
---------------------	-----

URI	https://<server>/adminapi/campaign/{id}/contacts?q={callstatus}
Example URI	https://uccx-server/adminapi/campaign/200/contacts?q=CALLBACK
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<campaignContacts>
  <campaign name="previewCampaign">
    <refURL>https://uccx-server/adminapi/campaign/200</refURL>
  </campaign>
  <csvdata>account number, first name, last name, phone1,
    phone2, phone3, dial time
    , , , 1000, , , 10:45
    , , , 1001, , , 10:45
    , , , 1002, , , 10:45
    , , , 1003, , , 10:45
    , , , 1004, , , 10:45
  </csvdata>
</campaignContacts>

<?xml version="1.0" encoding="UTF-8"?>
<campaignContacts>
  <campaign name="previewCampaign">
    <refURL>https://uccx-server/adminapi/campaign/200</refURL>
  </campaign>
  <csvdata>account number, first name, last name, phone1,
    phone2, phone3, dial time, call result, callback number, call status, call result
    01, call result 02, call result 03, last number dialled, calls made to phone 01, calls made
    to phone 02, calls made to phone 03, retry, num missed call back

    , , , 1000, , , 10:45,0, , 5, 0, 0, 0, 0, 0, 0, 0, 0, 0, false, 0
    , , , 1001, , , 10:45,0, , 5, 0, 0, 0, 0, 0, 0, 0, 0, 0, false, 0
    , , , 1002, , , 10:45,0, , 5, 0, 0, 0, 0, 0, 0, 0, 0, 0, false, 0
    , , , 1003, , , 10:45,0, , 5, 0, 0, 0, 0, 0, 0, 0, 0, 0, false, 0
    , , , 1004, , , 10:45,0, , 5, 0, 0, 0, 0, 0, 0, 0, 0, 0, false, 0
  </csvdata>
</campaignContacts>
```

Add Contacts to Campaign

HTTPS Method	POST
URI	https://<server>/adminapi/campaign/{id}/contacts
Example URI	https://uccx-server/adminapi/campaign/200/contacts
Accept	Application/XML, Application/JSON
Content Type	Application/XML, Application/JSON

HTTPS Success Code	200, 202
HTTPS Failure Codes	400, 401, 403, 404, 405, 500
Error Codes	—

Input XML Example 1

```
<?xml version="1.0" encoding="UTF-8"?>
<campaignContacts xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="campaignContacts.xsd">
  <campaign name="Campaign1">
    <refURL>https://uccx-server/adminapi/campaign/200</refURL>
  </campaign>
  <allowDuplicateContacts>true</allowDuplicateContacts>
  <csvdata>
Account Number, First Name, Last Name ,Phone1, Phone2, Phone3, Dial Time
ABCD1234, John , Doe, 1401111, 1402222, 1403333, 12:35
ABCD1235, Jane , Smith, 1504444, 1505555, 1506666, 12:35
  </csvdata>
</campaignContacts>
```



Note Dial Time field is applicable only for UCCX Agent Direct Preview campaigns.

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<campaignContacts>
  <campaign name="Campaign1">
    <refURL>https://uccx-server/adminapi/campaign/200</refURL>
  </campaign>
  <csvdata/>
</campaignContacts>
```

Input XML Example 2

```

<campaignContacts>
  <campaign name="campaign1">
    <refURL>https://uccx-server/adminapi/campaign/200</refURL>
  </campaign>

  <allowDuplicateContacts>true</allowDuplicateContacts>
  <csvFileData>
    <csvFileRef>43_contacts.csv</csvFileRef>
    <importType>AUTO</importType>
    <contactHeaders>
      <header1>Account Number</header1>
      <header2>First Name</header2>
      <header3>Last Name</header3>
      <header4>Phone1</header4>
      <header5>Phone2</header5>
      <header6>Phone3</header6>
      <header7>Dial Time</header7>
    </contactHeaders>
  </csvFileData>
</campaignContacts>

```

Response Headers

The header "Location" contains the URL for the import status API, which can be executed by client to get the status of import.

Example: `http://<UCCX FQDN>/adminapi/campaign/<campId>/contacts/status`



Attention Do not upload the outbound dialing list using the Load Balancer.

Get Campaign Contacts Import Status

HTTPS Method	GET
URI	https://<server>/adminapi/campaign/{id}/contacts/status
Example URI	https://uccx-server/adminapi/campaign/200/contacts/status
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 403, 404, 405, 500
Error Codes	—

Success Response

```

<statusDetails>
  <self>https://uccx-sa3.cisco.com/adminapi/campaign/6</self>
<campaignId>6</campaignId>

```

```

<importStatus>
  <status importType="AUTO">
    <result>SUCCESS</result>
    <startTime>1536871071356</startTime>
    <numContactsImported>1</numContactsImported>
    <i18n>
      <Key>obss.schedule.import.success</Key>
      <Args>
        <Arg>
        </Arg>
      </Args>
    </i18n>
  </status>
  <status importType="MANUAL">
    <result>SUCCESS</result>
    <i18n>
      <Key>obss.schedule.import.success</Key>
      <Args>
        <Arg>
        </Arg>
      </Args>
    </i18n>
  </status>
</importStatus>
</statusDetails>

```

Delete All Contacts from Campaign

HTTPS Method	DELETE
URI	https://<server>/adminapi/campaign/{id}/contacts
Example URI	https://uccx-server/adminapi/campaign/200/contacts
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Channel Parameters

Channel parameters are parameters that are associated with chat and email contact channels.

Data Structure for Channel Parameters

Attribute Name	Base Type	Restrictions	Description
agentNoAnswerTimeout	int	minInclusive =10 maxInclusive =3600 nillable=false default =15	Specifies the time in seconds for an agent to respond to the chat request after which, the chat request is routed back to the chat queue. This attribute is also used for the chat toaster fade out time. Note When you use Chrome or Firefox, the browser overrides the chat toaster notification to fade out in 20 seconds, even if it is configured to a higher value.
agentJoinTimeout	int	minInclusive =60 maxInclusive =3600 nillable=false default =60	Specifies the time in seconds after which, the customer initiates a chat and, if an agent is not joined, the customer gets a custom error message.
inactivityTimeout	int	minInclusive =60 maxInclusive =3600 nillable=false default =300	Specifies the customer inactivity time in seconds after which, the system ends the chat. This timeout is on the customer side only.
offerChatWhenOnVoice	boolean	nillable="false" default="true"	Enables or disables the option to offer a chat contact to an agent who is on a voice call.
offerVoiceWhenOnChat	boolean	nillable="false" default="true"	Enables or disables the option to offer a voice call to an agent who is on chat.
maxChatSessionsPerAgent	int	minInclusive =1 maxInclusive =5 nillable=false default =1	Specifies the maximum number of chat sessions configured per agent.
maxEmailSessionsPerAgent	int	minInclusive =1 maxInclusive =5 nillable=false default =5	Specifies the maximum number of email sessions configured per agent.

Attribute Name	Base Type	Restrictions	Description
stickyTimeout	int	minInclusive =1 maxInclusive =120 nillable=false default =4	Specify the amount of time (in hours) for which an email message waits in a specific agent CSQ.

Get Channel Parameters Information

HTTPS Method	GET
URI	https://<server>/adminapi/channelParameters
Example URI	https://uccx-server/adminapi/channelParameters
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 402, 405, 412
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<channelParameters>
  <chatParameters>
    <agentNoAnswerTimeout>12</agentNoAnswerTimeout>
    <agentJoinTimeout>60</agentJoinTimeout>
    <inactivityTimeout>300</inactivityTimeout>
    <offerChatWhenOnVoice>false</offerChatWhenOnVoice>
    <offerVoiceWhenOnChat>false</offerVoiceWhenOnChat>
    <maxChatSessionsPerAgent>2</maxChatSessionsPerAgent>
  </chatParameters>
  <emailParameters>
    <maxEmailSessionsPerAgent>3</maxEmailSessionsPerAgent>
    <stickyTimeout>100</stickyTimeout>
  </emailParameters>
</channelParameters>
```

Put Channel Parameters Information

HTTPS Method	PUT
URI	https://<server>/adminapi/channelParameters
Example URI	https://uccx-server/adminapi/channelParameters
Content Type	Application/XML, Application/JSON

HTTPS Success Code	200
HTTPS Failure Codes	401, 402, 405, 412
Error Codes	—

Success Response

HTTPS/1.1 200OK

Sample Input XML—PUT

```
<?xml version="1.0" encoding="UTF-8"?>
<channelParameters>
  <chatParameters>
    <agentNoAnswerTimeout>12</agentNoAnswerTimeout>
    <agentJoinTimeout>60</agentJoinTimeout>
    <inactivityTimeout>360</inactivityTimeout>
    <offerChatWhenOnVoice>false</offerChatWhenOnVoice>
    <offerVoiceWhenOnChat>false</offerVoiceWhenOnChat>
    <maxChatSessionsPerAgent>2</maxChatSessionsPerAgent>
  </chatParameters>
  <emailParameters>
    <maxEmailSessionsPerAgent>3</maxEmailSessionsPerAgent>
  </emailParameters>
</channelParameters>
```

Channel Provider

Channel provider is used to configure the noninteractive channels in Unified CCX. Email is the only non-interactive channel that is supported.

Data Structure for Channel Provider

Attribute Name	Base Type	Restrictions	Description
type	string	minOccurs = 1 maxOccurs = 1 nillable = false Enumeration = email	Type of contact channel.

Attribute Name	Base Type	Restrictions	Description
fqdn	string	minOccurs = 1 maxOccurs = 1 nillable = false Regex = (?=^.{1,255}\$) (^(((?!-)[a-zA-Z0-9-]{1,63} (?<!--))((?!-)[a-zA-Z0-9-] {1,63}(?!-).)+[a-zA-Z]{2,63})\$)	Fully qualified domain name (FQDN) of the channel provider. FQDN is a combination of the channel server hostname and domain name. Do not enter the IP address. Example: <i>host.example.com</i>
protocol	string	minOccurs = 1 maxOccurs = 1 nillable = false Enumeration <ul style="list-style-type: none"> • IMAP for receiving • SMTP for sending 	Protocol that is used to communicate with the channel provider.
port	int	minOccurs = 1 maxOccurs = 1 nillable = false minInclusive = 1 maxInclusive = 65535	Valid 16-bit port number on the channel provider to establish communication.
description	string	minOccurs = 0 maxOccurs = 1 nillable = false minLength = 1 maxLength = 100 Description can contain up to 100 UTF-8 characters.	Description of the channel provider. This field is optional .
proxyType	string	socks or none default = none Regex = [Ss][Oo] [Cc][Kk][Ss][Nn][Oo][Nn][Ee]	Determines if socks proxy should be used to connect to the channel provider server.

Attribute Name	Base Type	Restrictions	Description
mailserverType	string	minOccurs = 0 maxOccurs = 1 default = microsoft Regex = gmail microsoft	Determines the type of the mail server. Note To configure IMAP folder values, see imapFolderConfig in the Cisco Customer Collaboration Platform Developer Guide available at, https://developer.cisco.com/site/socialminer/documentation/ .

Get List of Channel Providers

HTTPS Method	GET
URI	https://<server>/adminapi/channelProvider
Example URI	https://uccx-server/adminapi/channelProvider
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	500
Error Codes	—

Success Response

```
<?xmlversion="1.0"encoding="UTF-8"standalone="yes"?>
<channelProviders>
  <channelProvider>
    <self>https://uccx-server/adminapi/channelProvider/8</self>
    <id>8</id>
    <type>email</type>
    <mailserverType>microsoft</mailserverType>
    <proxyType>none</proxyType>
    <send>
      <fqdn>host.example.com</fqdn>
      <port>465</port>
      <protocol>SMTP</protocol>
    </send>
    <receive>
      <fqdn>host.example.com</fqdn>
      <port>993</port>
      <protocol>IMAP</protocol>
    </receive>
    <description>This is a channel provider.</description>
  </channelProvider>
</channelProviders>
```

Get List of Channel Providers by Type

HTTPS Method	GET
URI	https://<server>/adminapi/channelProvider?type=<type>
Example URI	https://uccx-server/adminapi/channelProvider?type=email
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	500
Error Codes	—

Success Response

```
<?xmlversion="1.0"encoding="UTF-8"standalone="yes"?>
<channelProvider>
  <self>https://uccx-server/adminapi/channelProvider/8</self>
  <id>8</id>
  <type>email</type>
  <mailserverType>microsoft</mailserverType>
  <proxyType>none</proxyType>
  <send>
    <fqdn>host.example.com</fqdn>
    <port>465</port>
    <protocol>SMTP</protocol>
  </send>
  <receive>
    <fqdn>host.example.com</fqdn>
    <port>993</port>
    <protocol>IMAP</protocol>
  </receive>
  <description>This is a channel provider.</description>
</channelProvider>
```

Get a Channel Provider

HTTPS Method	GET
URI	https://<server>/adminapi/channelProvider/<ID>
Example URI	https://uccx-server/adminapi/channelProvider/8
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	404, 500
Error Codes	—

Success Response

```
<?xmlversion="1.0"encoding="UTF-8"standalone="yes"?>
<channelProvider>
  <self>https://uccx-server/adminapi/channelProvider/8</self>
  <id>8</id>
  <type>email</type>
  <mailserverType>microsoft</mailserverType>
  <proxyType>none</proxyType>
  <send>
    <fqdn>host.example.com</fqdn>
    <port>465</port>
    <protocol>SMTP</protocol>
  </send>
  <receive>
    <fqdn>host.example.com</fqdn>
    <port>993</port>
    <protocol>IMAP</protocol>
  </receive>
  <description>This is a channel provider.</description>
</channelProvider>
```

Create a Channel Provider

HTTPS Method	POST
URI	https://<server>/adminapi/channelProvider
Example URI	https://uccx-server/adminapi/channelProvider
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	400, 412, 500
Error Codes	—

Success Response

HTTPS/1.1 201 CREATED

https://uccx-server/adminapi/channelProvider/8

Restriction

You can create only one channel provider of a given type in Unified CCX.

Sample Input XML— POST

```
<?xmlversion="1.0"encoding="UTF-8"standalone="yes"?>
<channelProvider>
  <type>email</type>
  <mailserverType>microsoft</mailserverType>
  <proxyType>none</proxyType>
  <send>
    <fqdn>host.example.com</fqdn>
    <port>465</port>
    <protocol>SMTP</protocol>
```

```

</send>
<receive>
  <fqdn>host.example.com</fqdn>
  <port>993</port>
  <protocol>IMAP</protocol>
</receive>
<description>This is a channel provider.</description>
</channelProvider>

```

Modify a Channel Provider

HTTPS Method	PUT
URI	https://<server>/adminapi/channelProvider/<ID>
Example URI	https://uccx-server/adminapi/channelProvider/2
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 404, 500
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Sample Input XML— POST

```

<?xmlversion="1.0"encoding="UTF-8"standalone="yes"?>
<channelProvider>
  <type>email</type>
  <mailserverType>microsoft</mailserverType>
  <proxyType>none</proxyType>
  <send>
    <fqdn>host.example.com</fqdn>
    <port>465</port>
    <protocol>SMTP</protocol>
  </send>
  <receive>
    <fqdn>host.example.com</fqdn>
    <port>993</port>
    <protocol>IMAP</protocol>
  </receive>
  <description>This is a channel provider.</description>
</channelProvider>

```

Delete a Channel Provider

HTTPS Method	DELETE
URI	https://<server>/adminapi/channelProvider/<ID>
Example URI	https://uccx-server/adminapi/channelProvider/2

Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	404, 412, 500
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Restriction

You can delete a channel provider only if there are no email Contact Service Queues (CSQs) associated with the channel provider.

Chat Widget

Chat Widget is a chat program that is run by Unified CCX.

Data Structure for Chat Widget

Attribute Name	Base Type	Restrictions	Description
id	int	minOccurs = 0 minValue = 1 maxValue = 2147483647	Unique ID for each widget.
name	string	minLength = 1 maxLength = 50 occurrences = 1 Regex = ([^<>~`!@#\$\$%?&*()= [] {} ; / '+ : , "])+	Name of the chat widget.
description	string	minLength = 1 maxLength = 50 occurrences = 1 Regex = ([^<>~`!@#\$\$%?&*()= [] {} ; / '+ : , "])+	Description of the widget that is configured in Unified CCX Administration.

Attribute Name	Base Type	Restrictions	Description
formField	string	minLength = 1 maxLength = 50 minOccurs = 1 maxOccurs = 10 Regex = ([^ <>~`!@#\$\$%? &*()=[]{},;/'+:.,"])+	Fields that are present in the form.
welcomeMessage	string	minLength = 1 maxLength = 90 minOccurs = 0 maxOccurs = 1 Regex = ([^<> ()"])+ Default = "Thank you for contacting us. A customer care representative would assist you soon."	Message that is displayed to the customer when the chat form is submitted.
agentJoinTimeoutMsg	string	minLength = 1 maxLength = 90 minOccurs = 0 maxOccurs = 1 Regex = ([^<> ()"])+ Default = "All our customer care representatives are busy. You may wait or try again later."	Message that is displayed to the customer when a chat request is not handled within the set time.
chatErrorMsg	string	minLength = 1 maxLength = 90 minOccurs = 0 maxOccurs = 1 Regex = ([^<> ()"])+ Default = "Chat service is currently unavailable. Try later."	Message that is displayed to the customer when Unified CCX or chat service is not available to handle chat requests.

Attribute Name	Base Type	Restrictions	Description
problemStatementCSQPair	problemStatementscsq	minOccurs = 1 maxOccurs = Unbounded	Definition of the problem.
type	string	minOccurs = 0 maxOccurs = 1	Indicates the type of widget: • Default : bubble
bubbleStyle	bubbleStyle	minOccurs = 0 maxOccurs = 1	Provides styling information of the chat bubble.
bubbleMessages	bubbleMessages	minOccurs = 0 maxOccurs = 1	Text messages displayed on the chat bubble.
calendarDetails	calendarDetails	minOccurs = 0 maxOccurs = 1	Provides details of calendars and its corresponding messages that are associated with the chat widget.

Problem Statement CSQ Data Structure

Attribute Name	Base Type	Restrictions	Description
problemStatement	string	minOccurs = 1 maxOccurs = 1 minLength = 1 maxLength = 100 Regex = ([^\ "])+	
csq	nameUriPair	minOccurs = 1 maxOccurs = 1	

Bubble Style Data Structure

Attribute Name	Base Type	Restrictions	Description
titleText	string	minOccurs = 1 maxOccurs = 1 minLength = 1 maxLength = 24	Text displayed on the chat web form and chat bubble.
titleTextColor	colorCodeHex	minOccurs = 1 maxOccurs = 1 Regex = #[A-Fa-f0-9]{6}	Color of the title text.

Attribute Name	Base Type	Restrictions	Description
buttonText	string	minOccurs = 1 maxOccurs = 1 minLength = 1 maxLength = 15	Text displayed on the button of the chat web form.
buttonTextColor	colorCodeHex	minOccurs = 1 maxOccurs = 1 Regex = #[A-Fa-f0-9]{6}	Color of the text displayed on the button.
buttonBackgroundColor	colorCodeHex	minOccurs = 1 maxOccurs = 1 Regex = #[A-Fa-f0-9]{6}	Color of the button.
problemStmtCaption	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 30 Default = Choose a problem statement	Label that asks the user to choose a problem statement.
ratingEnabled	boolean	minOccurs = 0 maxOccurs = 1 nillable = false Default = false	Whether post-chat rating is available for the chat.
agentMessageTextColor	colorCodeHex	minOccurs = 1 maxOccurs = 1 Regex = #[A-Fa-f0-9]{6}	Color of the agent message text
agentMessageBackgroundColor	colorCodeHex	minOccurs = 1 maxOccurs = 1 Regex = #[A-Fa-f0-9]{6}	Background color of the agent message in the chat window.
fontType	string	minOccurs = 1 maxOccurs = 1 maxLength = 30	Font family used for the text in the chat web form and chat window.

Bubble Message Data Structure

Attribute Name	Base Type	Restrictions	Description
textForTypingMsg	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 40 Regex = ([^<>\\(\)"()/=!\[\]"#%&*;'+:\^-\@]) Default = Type your message and press enter.	Text displayed to prompt the user to type message.
agentJoinedMsg	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 46 Regex = ([^<>\\(\)"()/=!\[\]"#%&*;'+:\^-\@]) Default = \${agent_alias} has joined the chat.	Text indicating that the agent has joined the chat.
agentLeftMsg	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 46 Regex = ([^<>\\(\)"()/=!\[\]"#%&*;'+:\^-\@]) Default = \${agent_alias} has left the chat.	Text indicating that the agent has left the chat.

Attribute Name	Base Type	Restrictions	Description
transcriptPopupMsg	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 75 Regex = ([^<>\\(\)"()=!\\[]"#%&*;'+:/^-@]) Default = "Chat has ended. Do you want to download the chat transcript?"	Text on the pop-up that notifies the user to download the chat transcript.
transcriptPopupNegativeMsg	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 7 Regex = ([^<>\\(\)"()=!\\[]"#%&*;'+:/^-@]) Default = "No"	Text on the button (on the pop-up) that allows the user to accept the transcript download.
transcriptPopupPositiveMsg	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 7 Regex = ([^<>\\(\)"()=!\\[]"#%&*;'+:/^-@]) Default = "Yes"	Text on the button (on the pop-up) that allows the user to reject the transcript download.

Attribute Name	Base Type	Restrictions	Description
closeChatPopupMsg	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 35 Regex = ([^<>\\(\)"()=!\\[\]"#%&*;'+:\^-\@]) Default = "Do you want to close the chat?"	Text on the pop-up that confirms if the user wants to close the chat.
closeChatPopupNegativeMsg	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 7 Regex = ([^<>\\(\)"()=!\\[\]"#%&*;'+:\^-\@]) Default = "No"	Text on the button (on the pop-up) that allows the user to reject the chat closure.
closeChatPopupPositiveMsg	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 7 Regex = ([^<>\\(\)"()=!\\[\]"#%&*;'+:\^-\@]) Default = "Yes"	Text on the button (on the pop-up) that allows the user to accept the chat closure.

Attribute Name	Base Type	Restrictions	Description
connectivityErrorMsg	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 75 Regex = ([^<>\\(\)"()=!\\[]"#%&*;'+:/^-@])+	Text indicating that the chat was disconnected due to connectivity issues.

Calendar Details Data Structure

Attribute Name	Base Type	Restrictions	Description
calendar	nameUriPair	minOccurs = 0 maxOccurs = 1	Provides details of the calendar that is associated with the Chat Widget.
messages	calendarMessages	minOccurs = 0 maxOccurs = 1	Text messages that are related to Manage Calendar page.

Calendar Messages Data Structure

Attribute Name	Base Type	Restrictions	Description
holidayMsg	string	minOccurs = 0 maxOccurs = 1 minLength value= 1 maxLength value= 120 Regex = ([^<>\\(\)"()=!\\[]"#%&*;'+:/^-@])+	Message that is displayed to customers on a holiday.

Attribute Name	Base Type	Restrictions	Description
offHoursMsg	string	minOccurs = 0 maxOccurs = 1 minLength value= 1 maxLength value= 120 Regex = ([^<>\\(\)"()/=!\\[\]"#%&*;'+:\^-\@])+	Message that is displayed to customers during non-working hours.
businessHoursMsg	string	minOccurs = 0 maxOccurs = 1 minLength value= 1 maxLength value= 50 Regex = ([^<>\\(\)"()/=!\\[\]"#%&*;'+:\^-\@])+	Message that is displayed to customers during business hours.
dayOfWeek	dayOfWeek	minOccurs = 1 maxOccurs = 1	Label for the day of the week.

Day of Week Data Structure

Attribute Name	Base Type	Restrictions	Description
monday	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 15 Regex = ([^<>\\(\)"()/=!\\[\]"#%&*;'+:\^-\@])+	First day of the week.

Attribute Name	Base Type	Restrictions	Description
tuesday	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 15 Regex = ([^<>\\(\)"()=![]"#%&*;'+:/^-@]) ⁺	Second day of the week.
wednesday	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 15 Regex = ([^<>\\(\)"()=![]"#%&*;'+:/^-@]) ⁺	Third day of the week.
thursday	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 15 Regex = ([^<>\\(\)"()=![]"#%&*;'+:/^-@]) ⁺	Fourth day of the week.
friday	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 15 Regex = ([^<>\\(\)"()=![]"#%&*;'+:/^-@]) ⁺	Fifth day of the week.

Attribute Name	Base Type	Restrictions	Description
saturday	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 15 Regex = ([^<>\\(\)"()=!\\[\]"#%& *;'+:\^-\@])+	Sixth day of the week.
sunday	string	minOccurs = 0 maxOccurs = 1 minLength = 1 maxLength = 15 Regex = ([^<>\\(\)"()=!\\[\]"#%& *;'+:\^-\@])+	Seventh day of the week.

Create Chat Widget

HTTPS Method	POST
URI	https://<server>/adminapi/chatWidget
Example URI	https://uccx-server/adminapi/chatWidget
Content Type	Application/XML, Application/JSON
HTTPS Success code	201
HTTPS Failure Codes	400, 401, 404, 405, 412, 500
Error Codes	-

Success Response

HTTPS/1.1 200 Created

Sample Input XML for Chat Bubble— POST

```

<?xml version="1.0" encoding="UTF-8"?>
<chatWidget>
<self href="https://uccx-server:8080/adminapi/chatWidget rel="self" type="chatWidget"/>
  <name>ChatWidget7</name>
  <description>ChatWidget7</description>
  <formField>Name</formField>
  <contextServiceFieldsets>test</contextServiceFieldsets>
  <welcomeMessage>Thank you for contacting us. A customer care representative would assist
you soon.</welcomeMessage>
  <agentJoinTimeoutMsg>All our customer care representatives are busy. You may wait or
try again later.</agentJoinTimeoutMsg>
  <chatErrorMsg>Chat service is currently unavailable. Try later.</chatErrorMsg>
  <problemStatementCSQPair>
  <problemStatement>chatTest</problemStatement>
  <csq name="chatCSQ1">
    <refURL>https://uccx-server/adminapi/csq/1</refURL>
  </csq>
</problemStatementCSQPair>
  <type>bubble</type>
  <calendarDetails>
  <calendar name="calendar1">
    <refURL>https://10.107.45.242/adminapi/calendar/1</refURL>
  </calendar>
  <messages>
  <holidayMsg>Sorry, We are closed today for a business holiday.</holidayMsg>
  <offHoursMsg>We are currently offline, try again during the business
hours.</offHoursMsg>
  <businessHoursMsg>Business Hours (Local Time)</businessHoursMsg>
  <dayOfWeek>
  <monday>Monday</monday>
  <tuesday>Tuesday</tuesday>
  <>wednesday>Wednesday</wednesday>
  <thursday>Thursday</thursday>
  <friday>Friday</friday>
  <saturday>Saturday</saturday>
  <sunday>Sunday</sunday>
  </dayOfWeek>
</messages>
</calendarDetails>
  <bubbleStyle>
  <titleText>chat bubble 1</titleText>
  <titleTextColor>#EBEBEC</titleTextColor>
  <buttonText>chat button 1</buttonText>
  <buttonTextColor>#EBEBEC</buttonTextColor>
  <buttonBackgroundColor>#EBEBEC</buttonBackgroundColor>
  <resumeButtonText>Resume Chat</resumeButtonText>

  <agentMessageTextColor>#EBEBEC</agentMessageTextColor>
  <agentMessageBackgroundColor>#EBEBEC</agentMessageBackgroundColor>
  <fontType>Arial</fontType>
</bubbleStyle>
  <bubbleMessages>
  <textForTypingMsg>Type your message and press enter</textForTypingMsg>
  <agentJoinedMsg>${agent_alias} has joined the chat</agentJoinedMsg>
  <agentTypingMsg>${agent_alias} is typing</agentTypingMsg>
  <agentLeftMsg>${agent_alias} has left the chat</agentLeftMsg>
  <afterChatSessionTranscriptPopupMsg>
  <transcriptPopupMsg>Chat has ended. Do you want to download the chat
transcript?</transcriptPopupMainMsg>
  <transcriptPopupNegativeMsg>No</transcriptPopupNegativeMsg>
  <transcriptPopupPositiveMsg>Yes</transcriptPopupPositiveMsg>
  </afterChatSessionTranscriptPopupMsg>
  <closeChatConfirmationPopupMsg>

```

```

        <closeChatPopupMsg>Do you want to close the chat?</closeChatPopupMainMsg>
        <closeChatPopupNegativeMsg>No</closeChatPopupNegativeMsg>
        <closeChatPopupPositiveMsg>Yes</closeChatPopupPositiveMsg>
    </closeChatConfirmationPopupMsg>
    <connectivityErrorMsg>Chat disconnected due to inactivity timeout or connection
failure.</connectivityErrorMsg>
</bubbleMessages>
<postChatRating>
    <ratingEnabled>true</ratingEnabled>
    <ratingLabel>Rate your chat experience</ratingLabel>
    <ratingButtonText>Submit</ratingButtonText>
</postChatRating>
</chatWidget>

```

Get List of Chat Widgets

HTTPS Method	GET
URI	https://<server>/adminapi/chatWidget
Example URI	https://uccx-server/adminapi/chatWidget
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 412, 500
Error Codes	—

Get Chat Widget (XML)

HTTPS Method	GET
URI	https://<server>/adminapi/chatWidget/<chatWidget id>
Example URI	https://uccx-server/adminapi/chatWidget/2
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—

Success Response

```

<?xml version="1.0" encoding="UTF-8"?>
<chatWidget>
  <self href="https://uccx-server:8080/adminapi/chatWidget/2" rel="self" type="chatWidget"/>

  <id>2</id>
  <name>TestchatWidget0</name>
  <description>TestchatWidget0</description>
  <formField>Name</formField>
  <contextServiceFieldsets>test</contextServiceFieldsets>
  <welcomeMessage>Thank you for contacting us. A customer care representative would assist
you soon.</welcomeMessage>
  <agentJoinTimeoutMsg>All our customer care representatives are busy. You may wait or
try again later.</agentJoinTimeoutMsg>
  <chatErrorMsg>Chat service is currently unavailable. Try later.</chatErrorMsg>
  <problemStatementCSQPair>
    <problemStatementCSQPair>
      <problemStatement>chatTest</problemStatement>
      <csq name="chatCSQ1">
        <link href="https://uccx-server:8080/adminapi/csq/1" rel="used by" type="csq"/>
      </csq>
    </problemStatementCSQPair>
  </problemStatementCSQPair>
  <calendarDetails>
    <calendar name="calendar1">
      <refURL>https://10.107.45.242/adminapi/calendar/1</refURL>
    </calendar>
  </calendarDetails>
  <messages>
    <holidayMsg>Sorry, We are closed today for a business holiday.</holidayMsg>
    <offHoursMsg>We are currently offline, try again during the business
hours.</offHoursMsg>
    <businessHoursMsg>Business Hours (Local Time)</businessHoursMsg>
    <dayOfWeek>
      <monday>Monday</monday>
      <tuesday>Tuesday</tuesday>
      <>wednesday>Wednesday</wednesday>
      <thursday>Thursday</thursday>
      <friday>Friday</friday>
      <saturday>Saturday</saturday>
      <sunday>Sunday</sunday>
    </dayOfWeek>
  </messages>
</calendarDetails>
<type>bubble</type>
<bubbleStyle>
  <titleText>chat bubble 1</titleText>
  <titleTextColor>#FF0000</titleTextColor>
  <buttonText>chat button 1</buttonText>
  <buttonTextColor>#FF0000</buttonTextColor>
  <buttonBackgroundColor>#EBEBEC</buttonBackgroundColor>
  <resumeButtonText>Resume Chat</resumeButtonText>
  <agentMessageTextColor>#FF0000</agentMessageTextColor>
  <agentMessageBackgroundColor>#FF0000</agentMessageBackgroundColor>
  <fontType>Arial</fontType>
</bubbleStyle>
<bubbleMessages>
  <textForTypingMsg>Type your message and press enter</textForTypingMsg>
  <agentJoinedMsg>${agent_alias} has joined the chat</agentJoinedMsg>
  <agentTypingMsg>${agent_alias} is typing</agentTypingMsg>
  <agentLeftMsg>${agent_alias} has left the chat</agentLeftMsg>
  <afterChatSessionTranscriptPopupMsg>
    <transcriptPopupMsg>Chat has ended. Do you want to download the chat
transcript?</transcriptPopupMainMsg>
    <transcriptPopupNegativeMsg>No</transcriptPopupNegativeMsg>
    <transcriptPopupPositiveMsg>Yes</transcriptPopupPositiveMsg>
  </afterChatSessionTranscriptPopupMsg>
</bubbleMessages>

```

```

    <closeChatConfirmationPopupMsg>
      <closeChatPopupMsg>Do you want to close the chat?</closeChatPopupMainMsg>
      <closeChatPopupNegativeMsg>No</closeChatPopupNegativeMsg>
      <closeChatPopupPositiveMsg>Yes</closeChatPopupPositiveMsg>
    </closeChatConfirmationPopupMsg>
    <connectivityErrorMsg>Chat disconnected due to inactivity timeout or connection
failure.</connectivityErrorMsg>
  </bubbleMessages>
</chatWidget>

```

Get Chat Widget (HTML)

HTTPS Method	GET
URI	https://<server>/adminapi/chatWidget/<chatWidget id>
Example URI	https://uccx-server/adminapi/chatWidget/2
Accept	text/html
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405, 412, 500
Error Codes	—

Success Response

```

<!-- Add this script tag without any modification to the target webpage -->
<script type="application/javascript">
  var ciscoBubbleChat = (function () {
    var smHost = 'blr-sm-95-236.cisco.com';
    var widgetId = '2';

    var msgMustAcceptCert = 'Certificate must be accepted to start the conversation.';

    var msgAcceptCertButtonLabel = 'Accept Certificate';
    var msgCloseButtonLabel = 'Close';
    var msgWaitingCertAcceptance = 'Waiting for certificate acceptance.';
    var msgConnectivityIssues = 'We are experiencing connectivity issues. Try later.';

    var appId = 'cisco_bubble_chat';
    var appMargin = 15;
    var appUrl = 'https://' + smHost + '/ccp/ui/BubbleChat.html?host=' + smHost +
'&wid=' + widgetId;
    var connectivityCheckUrl = 'https://' + smHost + '/ccp/ui/ConnectivityCheck.html';

    var messageEventListener;
    var addNoCacheQueryParam;
    return {
      showChatWindow: function (injectedData) {
        var logPrefix = 'CISCO_BUBBLE_CHAT: ';
        if (document.getElementById(appId)) {
          console.log(logPrefix + 'Not loading BubbleChat as it is already
loaded');
          return;
        }

        var validateInjectedData = function(formData) {
          // browser compatible way to check whether it is an object with 10
fields and all the values are strings
          var result = true;
          if (formData && typeof formData === 'object' && formData.constructor
=== Object){
            var counter = 0;
            for (var key in formData) {
              if (!(typeof formData[key] === 'string' || formData[key]
instanceof String)) {
                result = false;
                break;
              }
              counter++;
              if (counter > 10) {
                result = false;
                break;
              }
            }
          } else {
            result = false;
          }
          return result;
        };

        if (injectedData) {
          if (validateInjectedData(injectedData.formData)) {
            appUrl += '&injectedFormData=' +
encodeURIComponent(JSON.stringify(injectedData.formData));
          } else {
            if (typeof injectedData.validationErrorCallback === 'function') {
              injectedData.validationErrorCallback();
            } else {

```



```

        console.log(logPrefix + 'Could not invoke validationErrorCallback
as it is not a function');
    }
}

var iframe = document.createElement('iframe');
iframe.setAttribute('sandbox', 'allow-scripts allow-same-origin allow-forms
allow-popups');
iframe.setAttribute('id', appId);
iframe.setAttribute('style', 'position: fixed; width: 312px; height: 410px;
border: none; bottom: 0px; right: 0; z-index:999;');
document.body.appendChild(iframe);
var frameWindow = iframe.contentWindow ? iframe.contentWindow : iframe;
var frameDoc = frameWindow.document;

// Trigger a page load for iframe inline content loading to work in Firefox

frameDoc.open();
frameDoc.close();

frameDoc.body.innerHTML = '<div id="secure-connectivity-check-container"
style="position: fixed; width: 300px; height: 395px; ' +
    'bottom: 10px; right: 10px; font-family:
Helvetica; font-size: 14px; color: #4F5051;' +
    'box-shadow: 0 0 3px #000; background: #fff;
display: flex; flex-direction: column; display: none;">' +
    '<div style="height: 25%;"></div>' +
    '<div style="height: 25%; display: flex;
align-items: flex-start; justify-content: center; text-align: center;">' +
        '<div style="padding: 0 15% 0 15%;">'
+
        '<div
id="secure-connectivity-check-msg"></div>' +
        '<a id="accept-cert-button"
style="display:none; padding-top: 10px" href="#" onclick="acceptCertificate(); return
void(0);">' +
            msgAcceptCertButtonLabel +
            '</a>' +
        '</div>' +
        '</div>' +
        '<div style="height: 25%; display: flex;
align-items: flex-end; justify-content: center; text-align: center;">' +
            '<div style="padding: 0 15% 0 15%;">'
+
            '<a href="#"
onclick="window.parent.postMessage({messageType: \'unmount\'}, \'*\'); return void(0);">'
+
            msgCloseButtonLabel +
            '</a>' +
        '</div>' +
        '</div>' +
        '<div style="height: 25%;"></div>' +
        '</div>';

frameWindow.acceptCertificate = function () {
    frameDoc.getElementById('secure-connectivity-check-msg').innerHTML =
msgWaitingCertAcceptance;
    frameDoc.getElementById('accept-cert-button').style.display = 'none';
    window.open(addNoCacheQueryParam(connectivityCheckUrl), 'SM_CERT_PAGE');
};

if (!addNoCacheQueryParam) {

```

```

        addNoCacheQueryParam = function (url) {
            return url + (url.indexOf("?") === -1 ? '?' : '&') + 'nocache=' +
new Date().getTime();
        }
    }

    if (!messageEventListener) {
        messageEventListener = function (event) {
            console.log(logPrefix + 'Received event from origin: ' +
event.origin);
            console.log(logPrefix + 'Received event data: ' +
JSON.stringify(event.data));
            switch (event.data.messageType) {
                case 'resize':
                    document.getElementById(appId).style.height = event.data.height
+ appMargin + 'px';
                    console.log(logPrefix + 'Successfully resized');
                    break;
                case 'unmount':
                    document.body.removeChild(document.getElementById(appId));
                    window.removeEventListener('message', messageEventListener);
                    console.log(logPrefix + 'Successfully unmounted BubbleChat and
removed event listener for message');
                    break;
                case 'bubblechat-cert-accepted':
                    document.getElementById(appId).setAttribute('src',
addNoCacheQueryParam(appUrl));
                    console.log(logPrefix + 'Successfully validated certificate
acceptance and loaded BubbleChat');
                    break;
                default:
                    console.log(logPrefix + 'Unknown message type');
            }
        };
    }

    window.addEventListener('message', messageEventListener);
    console.log(logPrefix + 'Event listener for message added');

    // Check HTTPS connectivity and show appropriate screen
    var showConnectivityIssue = function (message, showAcceptCertLink) {
        window.postMessage({ messageType: 'resize', height: 395 }, '*');

        frameDoc.getElementById('secure-connectivity-check-container').style.display = 'block';
        frameDoc.getElementById('secure-connectivity-check-msg').innerHTML =
message;
        frameDoc.getElementById('accept-cert-button').style.display =
showAcceptCertLink ? 'block' : 'none';
    };
    var xhr = new XMLHttpRequest();
    xhr.onreadystatechange = function () {
        if (this.readyState === 4) {
            console.log(logPrefix + 'Connectivity check status: ' + this.status);

            switch (this.status) {
                case 200:
                    iframe.setAttribute('src', addNoCacheQueryParam(appUrl));
                    break;
                case 0:
                    showConnectivityIssue(msgMustAcceptCert, true);
                    break;
                default:
                    showConnectivityIssue(msgConnectivityIssues, false);
            }
        }
    };

```

```

    }
    };
    console.log(logPrefix + 'Checking connectivity to: ' + connectivityCheckUrl);

    xhr.open('GET', addNoCacheQueryParam(connectivityCheckUrl), true);
    xhr.send();
  }
  });
  })();
</script>
<!--
  Use the function 'ciscoBubbleChat.showChatWindow()' as the event handler for initiating
  chat.
  eg: <button onclick="ciscoBubbleChat.showChatWindow()">Start Chat</button>

  Optionally, invisible form data can be submitted, which will be submitted along with
  the fields customer fills in.
  Upto 10 fields can be passed. If more than 10 fields are passed, the invisible form
  data will not be used and
  the provided error callback will be invoked. For injecting form data, an object should
  be passed to
  ciscoBubbleChat.showChatWindow() as an argument. The object should be of the form:
  {
    formData: {
      InjectedField1: 'InjectedValue1',
      InjectedField2: 'InjectedValue2'
      ...
    },
    validationErrorCallback: function(){console.log('business specific logic goes
here');}
  }
  The form data can have any string as field name and value. The submitted invisible form
  data values will be
  shown in the agent desktop, as well as will be updated in ContextService if the specified
  fieldset(s) in the widget
  contains these field names just like the regular visible chat form fields data.
  eg:
  <button onclick="ciscoBubbleChat.showChatWindow({
    formData: {
      AnyFieldName1: 'AnyFieldValue1',
      AnyFieldName2: 'AnyFieldValue2',
      AnyFieldName3: 'AnyFieldValue3',
      AnyFieldName4: 'AnyFieldValue4',
      AnyFieldName5: 'AnyFieldValue5',
      AnyFieldName6: 'AnyFieldValue6',
      AnyFieldName7: 'AnyFieldValue7',
      AnyFieldName8: 'AnyFieldValue8',
      AnyFieldName9: 'AnyFieldValue9',
      AnyFieldName10: 'AnyFieldValue10'
    },
    validationErrorCallback: function(){console.log('error in validating injected
data');}
  })">Click to chat</button>
-->

```

Modify Chat Widget

HTTPS Method	PUT
URI	https://<server>/adminapi/chatWidget/<chatWidget id>

Example URI	https://uccx-server/adminapi/chatWidget/2
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 401, 404, 405
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Sample Input XML— PUT

```

<?xml version="1.0" encoding="UTF-8"?>
<chatWidget>
<self href="https://uccx-server:8080/adminapi/chatWidget/2" rel="self" type="chatWidget"/>

<name>chatWidget7</name>
<description>chatWidget7</description>
<formField>Name</formField>
<contextServiceFieldsets>test</contextServiceFieldsets>
<welcomeMessage>Thank you for contacting us. A customer care representative would assist
you soon.</welcomeMessage>
<agentJoinTimeoutMsg>All customer care representatives are busy. You may wait or try again
later.</agentJoinTimeoutMsg>
<chatErrorMsg>Chat service is currently unavailable. Try later.</chatErrorMsg>
<problemStatementCSQPair>
<problemStatement>chatTest</problemStatement>
<csq name="chatCSQ1">
<link href="https://uccx-server/adminapi/csq/2" rel="" type=""/>
</csq>
</problemStatementCSQPair>
<calendarDetails>
  <calendar name="calendar1">
    <refURL>https://10.107.45.242/adminapi/calendar/1</refURL>
  </calendar>
  <messages>
    <holidayMsg>Sorry, We are closed today for a business holiday.</holidayMsg>
    <offHoursMsg>We are currently offline, try again during the business
hours.</offHoursMsg>
    <businessHoursMsg>Business Hours (Local Time)</businessHoursMsg>
    <dayOfWeek>
      <monday>Monday</monday>
      <tuesday>Tuesday</tuesday>
      <>wednesday>Wednesday</wednesday>
      <thursday>Thursday</thursday>
      <friday>Friday</friday>
      <saturday>Saturday</saturday>
      <sunday>Sunday</sunday>
    </dayOfWeek>
  </messages>
</calendarDetails>
<type>bubble</type>
<bubbleStyle>
  <titleText>chat bubble 1</titleText>
  <titleTextColor>#EBEBEC</titleTextColor>
  <buttonText>chat button 1</buttonText>
  <buttonTextColor>#EBEBEC</buttonTextColor>
  <buttonBackgroundColor>#EBEBEC</buttonBackgroundColor>
  <resumeButtonText>Resume Chat</resumeButtonText>

  <agentMessageTextColor>#EBEBEC</agentMessageTextColor>
  <agentMessageBackgroundColor>#EBEBEC</agentMessageBackgroundColor>
  <fontType>Arial</fontType>
</bubbleStyle>
<bubbleMessages>
  <textForTypingMsg>Type your message and press enter</textForTypingMsg>
  <agentJoinedMsg>${agent_alias} has joined the chat</agentJoinedMsg>
  <agentTypingMsg>${agent_alias} is typing</agentTypingMsg>
  <agentLeftMsg>${agent_alias} has left the chat</agentLeftMsg>
  <afterChatSessionTranscriptPopupMsg>
    <transcriptPopupMsg>Chat has ended. Do you want to download the chat
transcript?</transcriptPopupMainMsg>
    <transcriptPopupNegativeMsg>No</transcriptPopupNegativeMsg>
    <transcriptPopupPositiveMsg>Yes</transcriptPopupPositiveMsg>
  </afterChatSessionTranscriptPopupMsg>
</closeChatConfirmationPopupMsg>

```

```

        <closeChatPopupMsg>Do you want to close the chat?</closeChatPopupMainMsg>
        <closeChatPopupNegativeMsg>No</closeChatPopupNegativeMsg>
        <closeChatPopupPositiveMsg>Yes</closeChatPopupPositiveMsg>
    </closeChatConfirmationPopupMsg>
    <connectivityErrorMsg>Chat disconnected due to inactivity timeout or connection
failure.</connectivityErrorMsg>
</bubbleMessages>
<postChatRating>
    <ratingEnabled>true</ratingEnabled>
    <ratingLabel>Rate your chat experience</ratingLabel>
    <ratingButtonText>Submit</ratingButtonText>
</postChatRating>
</chatWidget>

```

Delete Chat Widget

HTTPS Method	DELETE
URI	https://<server>/adminapi/chatWidget/<chatWidget id>
Example URI	https://uccx-server/adminapi/chatWidget/2
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405, 412, 500
Failure Response	HTTPS/1.1 400 Bad Request
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Contact Service Queue

The Contact Service Queue (CSQ) controls incoming calls by determining where an incoming call is placed in the queue and to which agent the call is sent.

CSQ Data Structure

Attribute Name	Base Type	Restrictions	Description
name	string	minLength = 1 maxLength = 50 minOccurs = 1 maxOccurs = 1 nillable = false Name contains alphanumeric characters, special characters, spaces, underscores (_), and hyphens (-) without leading or trailing white spaces.	Name of the CSQ as set up in Unified CCX Administration
queueType	string	minOccurs = 1 maxOccurs = 1 nillable = false Enumerations = VOICE, EMAIL, CHAT	A type of the CSQ. Possible values: 0—voice CSQ 1—email CSQ 2—chat CSQ
queueAlgorithm	string	minOccurs = 1 maxOccurs = 1 nillable = false Enumerations = FIFO	Criterion that specifies how contacts are queued, as set up in Unified CCX Administration.
autoWork	boolean	minOccurs = 0 maxOccurs = 1 nillable = false default = false	Whether an agent goes to Work State after handling a call from this CSQ: f—No t—Yes
wrapupTime	int	minOccurs = 0 maxOccurs = 1 nillable = false minInclusive = 0 maxInclusive = 7200	Time in seconds that agent is placed in Work state. Possible values: 1—7200 0—disabled

Attribute Name	Base Type	Restrictions	Description
resourcePoolType	string	minOccurs = 1 maxOccurs = 1 nillable = false Enumerations = skillGroup, ResourceGroup	Type of resource pool that is set up in Unified CCX Administration: 1—Resource groups 2—Resource skills
serviceLevel	baseIdInt	minOccurs = 0 maxOccurs = 1 nillable = false default = 5	Goal, in seconds, for the maximum time that a caller spends in the queue before the call is answered by an agent, as set up in Unified CCX Administration.
serviceLevelPercentage	int	minOccurs = 0 maxOccurs = 1 nillable = false default = 70 minInclusive = 1 maxInclusive = 100	Goal for the percentage of calls that meet the service level that is shown in the serviceLevel field, as set up in Unified CCX Administration.
poolSpecificInfo	skillorResourceGroup	minOccurs = 1 maxOccurs = 1	Specific information about the pool, such as Skill Group or Resources Group.
RoutingType	string	minOccurs = 0 maxOccurs = 1 nillable = false Enumerations: VOICE, INTERACTIVE, and NONINTERACTIVE	The Routing Type element is mandatory for the Email CSQ and it should be set to "NONINTERACTIVE". For all other media types including the Voice and Chat, it is not mandatory to specify the routing type.

Mapping Between the CSQ Types and Routing Types That Are Used

CSQ Type	Routing Type
VOICE	VOICE
EMAIL	NONINTERACTIVE
CHAT	INTERACTIVE



Note The support for Routing Type "EMAIL" has been ended from Unified CCX 11.0(1) release onward. If CSQ type is "EMAIL", then the Routing Type can only be specified as "NONINTERACTIVE".

Email CSQ Data Structure

Attribute Name	Base Type	Restrictions	Description
accountUserId	string	minLength = 5 maxLength = 75 minOccurs = 0 maxOccurs = 1 nillable = false Name contains alphanumeric characters, special characters, spaces, dots (.), underscores (_), and hyphens (-) without leading or trailing white spaces. The account user ID cannot have consecutive dots. The domain can have up to four characters.	Email account userid from which emails are retrieved for this CSQ. One email address is associated with only one Email CSQ. This field is mandatory for the Email CSQ.
accountPassword	string	minLength = 1 maxLength = 128 minOccurs = 0 maxOccurs = 1 nillable = false	Password for the account user ID. This field is mandatory for the Email CSQ.
channelProvider	nameUriPair	minOccurs = 0 maxOccurs = 1 nillable = false	Stores information about the source of the contacts. In case of email, it is the mail server. This field is mandatory for the Email CSQ.

Attribute Name	Base Type	Restrictions	Description
pollingInterval	int	minOccurs = 0 maxOccurs = 1 nillable = false minInclusive = 10 maxInclusive = 86400 default = 600	Frequency to fetch emails from the server in seconds. This field is mandatory for the Email CSQ. The pollingInterval is specified in seconds.
folderName	string	minLength = 1 maxLength = 255 minOccurs = 0 maxOccurs = 1 nillable = false default = "Inbox"	The folder that holds the email messages in the email account. This field is mandatory for the Email CSQ.
snapshotAge	int	minOccurs = 0 maxOccurs = 1 nillable = false minInclusive = 10 maxInclusive = 43200 default = 120	Specify the time in minutes from when the emails are to be fetched. This field is mandatory for the Email CSQ. The snapshotAge is specified in minutes.

Skill or Resource Group Data Structure

Attribute Name	Base Type	Restrictions	Description
skillGroup	(choice) skillCompetencyList	minOccurs = 1 maxOccurs = 1	Unique identifier that is used to locate the associated skill group in the Skill Group table.
resourceGroup	(choice) resourceGroupList	minOccurs = 1 maxOccurs = 1	Unique identifier that is used to locate the associated resource group in the Resource Group table.

Skill Competency Data Structure

Attribute Name	Base Type	Restrictions	Description
skillCompetency	skillDetails	minOccurs = 0 maxOccurs = 50	Minimum acceptable skill level for agents with this skill, as set up in the Unified CCX Administration. Values range from 1 (lowest) to 10 (highest).
selectionCriteria	string	minOccurs = 1 maxOccurs = 1 Enumerations = Longest Available, Most Handled Contacts, Shortest Average Handle Time, Most Skilled, Least Skilled, Most Skilled by Weight, Least Skilled by Weight, Most Skilled by Order, Least Skilled by Order.	Resource pool selection model that is set up in the Unified CCX Administration.

Skill Details Data Structure

Attributes	Base Type	Restrictions	Description
	(extension) skillCompetency	minOccurs = 0 maxOccurs = 1	
weight	int	minOccurs = 0 maxOccurs = 1 nillable = false default = 1 minInclusive = 1 maxInclusive = 1000	Skills within a CSQ can be assigned weights. This field is used in the weighted skill calculation of the skill-based resource selection algorithm. Default value is 1.
order	baseIdInt	minOccurs = 0 maxOccurs = 1 nillable = false	Skills within a CSQ can be ordered. This field is used in the order skill calculation of the skill-based resource selection algorithm. Default value is 1.

Resource Group Lists Data Structure

Attribute	Base Type	Restrictions	Description
resourceGroupNameUriPair	nameUriPair	minOccurs = 1 maxOccurs = 1 nillable = false	Name of the resource group, as set up in the Unified CCX Administration.
selectionCriteria	string	minOccurs = 1 maxOccurs = 1 Enumerations=Longest Available, Linear, Circular, Most Handled Contacts, and Shortest Average Handle Time.	Resource pool selection model that is set up in the Unified CCX Administration.
resourceAndOrder	resourceOrder	minOccurs = 0 maxOccurs = unbounded nillable = false	Order in which the resource resides within the resource group.

Resource Order Data Structure

Attributes	Base Type	Restrictions	Description
resource	nameUriPair	minOccurs = 1 maxOccurs = 1 nillable = false	Name of the resource. Resource groups are collections of agents that your CSQ uses to handle incoming calls.
order	baseIdInt	minOccurs = 0 maxOccurs = 1 nillable = false	Order in which the resource resides within the resource group.

Get List of CSQs

HTTPS Method	GET
URI	https://<server>/adminapi/csq
Example URI	https://uccx-server/adminapi/csq
Accept	Application/XML, Application/JSON
HTTPS Success Code	200

HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<csqs>
  <csq>
    <self>https://uccx-server/adminapi/csq/1</self>
    <id>1</id>
    <name>Auto_Sanity_CSQ</name>
    <queueType>VOICE</queueType>
    <queueAlgorithm>FIFO</queueAlgorithm>
    <autoWork>false</autoWork>
    <wrapupTime>0</wrapupTime>
    <resourcePoolType>SKILL_GROUP</resourcePoolType>
    <serviceLevel>5</serviceLevel>
    <serviceLevelPercentage>70</serviceLevelPercentage>
    <poolSpecificInfo>
      <skillGroup>
        <skillCompetency>
          <competencelevel>5</competencelevel>
          <skillNameUriPair name="chat">
            <refURL>https://uccxip/adminapi/Skill/2</refURL>
          </skillNameUriPair>
          <weight>1</weight>
        </skillCompetency>
        <selectionCriteria>Longest Available</selectionCriteria>
      </skillGroup>
    </poolSpecificInfo>
  </csq>
  <csq>
    <self>https://uccx-server/adminapi/csq/2</self>
    <id>2</id>
    <name>icdcsq</name>
    <queueType>VOICE</queueType>
    <queueAlgorithm>FIFO</queueAlgorithm>
    <autoWork>false</autoWork>
    <wrapupTime>0</wrapupTime>
    <resourcePoolType>SKILL_GROUP</resourcePoolType>
    <serviceLevel>5</serviceLevel>
    <serviceLevelPercentage>70</serviceLevelPercentage>
    <poolSpecificInfo>
      <skillGroup>
        <skillCompetency>
          <competencelevel>5</competencelevel>
          <skillNameUriPair name="icd">
            <refURL>https://uccx-server/adminapi/Skill/3</refURL>
          </skillNameUriPair>
          <weight>1</weight>
        </skillCompetency>
        <selectionCriteria>Longest Available</selectionCriteria>
      </skillGroup>
    </poolSpecificInfo>
  </csq>
  <csq>
    <self>https://uccx-server/adminapi/csq/3</self>
    <id>3</id>
    <name>outboundCsq</name>
    <queueType>VOICE</queueType>
    <queueAlgorithm>FIFO</queueAlgorithm>
```

```

<autoWork>>false</autoWork>
<wrapupTime>0</wrapupTime>
<resourcePoolType>SKILL_GROUP</resourcePoolType>
<serviceLevel>5</serviceLevel>
<serviceLevelPercentage>70</serviceLevelPercentage>
<poolSpecificInfo>
  <skillGroup>
    <skillCompetency>
      <competencelevel>5</competencelevel>
      <skillNameUriPair name="ob_skill">
        <refURL>https://uccx-server/adminapi/Skill/4</refURL>
      </skillNameUriPair>
      <weight>1</weight>
    </skillCompetency>
    <selectionCriteria>Longest Available</selectionCriteria>
  </skillGroup>
</poolSpecificInfo>
</csqs>
<csq>
  <self></self>
  <id>60</id>
  <name>EmailCSQ1</name>
  <queueType>EMAIL</queueType>
  <routingType>NONINTERACTIVE</routingType>
  <queueAlgorithm>FIFO</queueAlgorithm>
  <autoWork>>false</autoWork>
  <resourcePoolType>SKILL_GROUP</resourcePoolType>
  <serviceLevel>5</serviceLevel>
  <serviceLevelPercentage>70</serviceLevelPercentage>
  <poolSpecificInfo>
    <skillGroup>
      <skillCompetency>
        <competencelevel>5</competencelevel>
        <skillNameUriPair name="S1">
          <refURL></refURL>
        </skillNameUriPair>
        <weight>1</weight>
      </skillCompetency>
      <selectionCriteria>Longest Available</selectionCriteria>
    </skillGroup>
  </poolSpecificInfo>
  <accountUserId>xyz@cisco.com</accountUserId>
  <accountPassword>accountpassword</accountPassword>
  <reviewQueue name="RevQueue1">
    <refURL></refURL>
  </reviewQueue>
  <channelProvider name="1">
    <refURL>https://uccx-server/adminapi/channelServer/1</refURL>
  </channelProvider>
  <pollingInterval>600</pollingInterval>
  <folderName>Inbox</folderName>
  <snapshotAge>120</snapshotAge>
</csq>
<csq>
  <self href="https://uccx-server/adminapi/csq/23" rel="self" type="csq" />
  <id>23</id>
  <name>csq1weitor</name>
  <queueType>VOICE</queueType> <routingType>VOICE</routingType>
  <queueAlgorithm>FIFO</queueAlgorithm>
  <autoWork>>false</autoWork>
  <wrapupTime>0</wrapupTime>
  <resourcePoolType>SKILL_GROUP</resourcePoolType>
  <serviceLevel>5</serviceLevel>

```

```

<serviceLevelPercentage>70</serviceLevelPercentage>
<poolSpecificInfo>
  <skillGroup>
    <skillCompetency>
      <competencelevel>4</competencelevel>
      <skillNameUriPair name="aa">
        <link href="https://uccx-server/adminapi/Skill/2" rel="selectedSkill"
          type="Skill" />
      </skillNameUriPair>
      <weight>1</weight>
    </skillCompetency>
    <skillCompetency>
      <competencelevel>5</competencelevel>
      <skillNameUriPair name="bb">
        <link href="https://uccx-server/adminapi/Skill/3" rel="selectedSkill"
          type="Skill" />
      </skillNameUriPair>
      <weight>1</weight>
    </skillCompetency>
    <selectionCriteria>Least Skilled by Order</selectionCriteria>
  </skillGroup>
</poolSpecificInfo>
</csq>
</csqs>

```

HTTPS Method	GET
URI	https://<server>/adminapi/csq?detail=full
Example URI	https://uccx-server/adminapi/csq?detail=full
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 401, 403, 405
Error Codes	—

Success Response

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<csqs>
  <csq>
    <self>https://uccx-server/adminapi/csq/1</self>
    <id>1</id>
    <name>chatcsq1</name>
    <queueType>CHAT</queueType>
    <routingType>INTERACTIVE</routingType>
    <queueAlgorithm>FIFO</queueAlgorithm>
    <autoWork>>false</autoWork>
    <resourcePoolType>SKILL_GROUP</resourcePoolType>
    <serviceLevel>5</serviceLevel>
    <serviceLevelPercentage>70</serviceLevelPercentage>
    <poolSpecificInfo>
      <skillGroup>
        <skillCompetency>
          <competencelevel>5</competencelevel>

```



```

    <skillNameUriPair name="chatskill11">
    <refURL>https://uccx-server/adminapi/skill/2</refURL>
    </skillNameUriPair>
    <weight>1</weight>
  </skillCompetency>
  <skillCompetency><competencelevel>5</competencelevel>
    <skillNameUriPair name="chatskill12">
    <refURL>https://uccx-server/adminapi/skill/3</refURL>
    </skillNameUriPair>
    <weight>1</weight>
    </skillCompetency>
    <selectionCriteria>Longest Available</selectionCriteria>
  </skillGroup>
</poolSpecificInfo>
</csq>
<csq>
  <self>https://uccx-server/adminapi/csq/4</self>
  <id>4</id><name>QueueEmail</name>
  <queueType>EMAIL</queueType>
  <routingType>NONINTERACTIVE</routingType>
  <queueAlgorithm>FIFO</queueAlgorithm>
  <autoWork>false</autoWork>
  <resourcePoolType>SKILL_GROUP</resourcePoolType>
  <serviceLevel>5</serviceLevel>
  <serviceLevelPercentage>70</serviceLevelPercentage>
  <poolSpecificInfo><skillGroup><skillCompetency>
    <competencelevel>5</competencelevel>
    <skillNameUriPair name="email1">
    <refURL>https://uccx-server/adminapi/skill/4</refURL>
    </skillNameUriPair>
    <weight>1</weight>
    </skillCompetency><selectionCriteria>Longest Available</selectionCriteria></skillGroup>
  </poolSpecificInfo>
  <accountUserId>abc-def@uccx-dev.com</accountUserId>
  <accountPassword></accountPassword>
  <channelProvider name="2">
    <refURL>https://uccx-server/adminapi/channelProvider/2</refURL>
  </channelProvider><pollingInterval>600</pollingInterval>
  <folderName>Inbox</folderName>
  <snapshotAge>120</snapshotAge>
</csq>
<csq><self>https://uccx-server/adminapi/csq/5</self>
  <id>5</id>
  <name>QueueEmail2</name>
  <queueType>EMAIL</queueType>
  <routingType>NONINTERACTIVE</routingType>
  <queueAlgorithm>FIFO</queueAlgorithm>
  <autoWork>false</autoWork>
  <resourcePoolType>SKILL_GROUP</resourcePoolType>
  <serviceLevel>5</serviceLevel>
  <serviceLevelPercentage>70</serviceLevelPercentage>
  <poolSpecificInfo>
    <skillGroup>
      <skillCompetency><competencelevel>5</competencelevel>
      <skillNameUriPair name="email1">
        <refURL>https://uccx-server/adminapi/skill/4</refURL>
      </skillNameUriPair>
      <weight>1</weight>
      </skillCompetency>
      <selectionCriteria>Longest Available</selectionCriteria>
    </skillGroup>
  </poolSpecificInfo>
  <accountUserId>xyz-def@uccx-dev.com</accountUserId>
  <accountPassword></accountPassword>

```

```

<channelProvider name="2">
  <refURL>https://uccx-server/adminapi/channelProvider/2</refURL>
</channelProvider>
<pollingInterval>600</pollingInterval>
<folderName>Inbox</folderName>
<snapshotAge>120</snapshotAge>
</csq>
<csq>
  <self>https://uccx-server/adminapi/csq/7</self>
  <id>7</id>
  <name>sad</name>
  <queueType>VOICE</queueType>
  <routingType>VOICE</routingType>
  <queueAlgorithm>FIFO</queueAlgorithm>
  <autoWork>false</autoWork>
  <wrapupTime>0</wrapupTime>
  <resourcePoolType>SKILL_GROUP</resourcePoolType>
  <serviceLevel>5</serviceLevel>
  <serviceLevelPercentage>70</serviceLevelPercentage>
  <poolSpecificInfo><skillGroup><skillCompetency>
    <competencelevel>5</competencelevel>
    <skillNameUriPair name="chatskill12">
      <refURL>https://uccx-server/adminapi/skill/3</refURL>
    </skillNameUriPair>
    <weight>1</weight>
  </skillCompetency>
  <selectionCriteria>Longest Available</selectionCriteria>
</skillGroup></poolSpecificInfo>
</csq>
<csq>
  <self>https://uccx-server/adminapi/csq/6</self>
  <id>6</id>
  <name>test_chat</name>
  <queueType>CHAT</queueType>
  <routingType>INTERACTIVE</routingType>
  <queueAlgorithm>FIFO</queueAlgorithm>
  <autoWork>false</autoWork>
  <resourcePoolType>SKILL_GROUP</resourcePoolType>
  <serviceLevel>5</serviceLevel>
  <serviceLevelPercentage>70</serviceLevelPercentage>
  <poolSpecificInfo><skillGroup><skillCompetency>
    <competencelevel>5</competencelevel>
    <skillNameUriPair name="chatskill11">
      <refURL>https://uccx-server/adminapi/skill/2</refURL>
    </skillNameUriPair>
    <weight>1</weight>
  </skillCompetency>
  <selectionCriteria>Longest Available</selectionCriteria>
</skillGroup>
</poolSpecificInfo>
</csq>
</csqs>

```

HTTPS Method	GET
URI	https://<server>/adminapi/csq?detail=<string>
Example URI	https://uccx-server/adminapi/csq?detail=<string>
Accept	Application/XML, Application/JSON

HTTPS Success Code	200
HTTPS Failure Codes	400, 401, 403, 405
Error Codes	—



Note <string> can be any other string except full and fetchRequeueCsqList.

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<results><names><name name="chatcsq1">
<refURL>https://uccx-server/adminapi/csq/1</refURL></name><name name="QueueEmail">
<refURL>https://uccx-server/adminapi/csq/4</refURL></name><name name="QueueEmail2">
<refURL>https://uccx-server/adminapi/csq/5</refURL></name><name name="CSQ1">
<refURL>https://uccx-server/adminapi/csq/7</refURL></name><name name="CSQ2">
<refURL>https://uccx-server/adminapi/csq/6</refURL></name></names></results>
```

HTTPS Method	GET
URI	https://<server>/adminapi/csq?detail=fetchRequeueCsqList&mediaType=chat
Example URI	https://uccx-server/adminapi/csq?detail=fetchRequeueCsqList&mediaType=chat
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 401, 403, 405
Error Codes	—



Note Users with Agent and Supervisor Roles can also access this get list. However, Users with Agent roles have limited access to this URI with only chat and email media types.

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<results><names><name name="chatcsq1">
<refURL>https://uccx-server/adminapi/csq/1</refURL>
</name><name name="chatcsq2">
<refURL>https://uccx-server/adminapi/csq/6</refURL>
</name></names></results>
```

Get CSQ

HTTPS Method	GET
URI	://<server>/adminapi/csq/{id}

Example URI	://uccx-server/adminapi/csq/22
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<csq>
  <self>https://uccx-server/adminapi/csq/3</self>
  <id>3</id>
  <name>outboundCsq</name>
  <queueType>VOICE</queueType>
  <queueAlgorithm>FIFO</queueAlgorithm>
  <autoWork>false</autoWork>
  <wrapupTime>0</wrapupTime>
  <resourcePoolType>SKILL_GROUP</resourcePoolType>
  <serviceLevel>5</serviceLevel>
  <serviceLevelPercentage>70</serviceLevelPercentage>
  <poolSpecificInfo>
    <skillGroup>
      <skillCompetency>
        <competencelevel>5</competencelevel>
        <skillNameUriPair name="ob_skill">
          <refURL>https://uccx-server/adminapi/Skill/4</refURL>
        </skillNameUriPair>
        <weight>1</weight>
      </skillCompetency>
      <selectionCriteria>Longest Available</selectionCriteria>
    </skillGroup>
  </poolSpecificInfo>
</csq>
```

Success Response—New Email CSQ with PLAIN emailAuthType

```
<?xml version="1.0" encoding="UTF-8"?>
<csq>
  <self></self>
  <id>28</id>
  <name>Email_Csq26</name>
  <queueType>EMAIL</queueType>
  <routingType>NONINTERACTIVE</routingType>
  <queueAlgorithm>FIFO</queueAlgorithm>
  <resourcePoolType>SKILL_GROUP</resourcePoolType>
  <poolSpecificInfo>
    <skillGroup>
      <selectionCriteria>Longest Available</selectionCriteria>
    </skillGroup>
  </poolSpecificInfo>
  <accountUserId>abc@cisco.com</accountUserId>
  <accountPassword>accountpassword</accountPassword>
  <channelProvider name="2">
    <refURL></refURL>
  </channelProvider>
  <pollingInterval>600</pollingInterval>
```

```
<folderName>Inbox</folderName>
<snapshotAge>120</snapshotAge>
</csq>
```

Create CSQ

HTTPS Method	POST
URI	https://<server>/adminapi/csq
Example URI	https://uccxserver/adminapi/csq
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	401, 405, 409, 412
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Sample Input XML—Create a CSQ (POST)

```
<csq xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="csq.xsd">
  <self href="" rel="" type="" />
  <name>vcsq1</name>
  <queueType>VOICE</queueType>
  <queueAlgorithm>FIFO</queueAlgorithm>
  <autoWork>true</autoWork>
  <wrapupTime>1</wrapupTime>
  <resourcePoolType>SKILL_GROUP</resourcePoolType>
  <serviceLevel>5</serviceLevel>
  <serviceLevelPercentage>70</serviceLevelPercentage>
  <poolSpecificInfo>
    <skillGroup>
      <skillCompetency>
        <competencelevel>5</competencelevel>
        <skillNameUriPair name="skill1">
          <refURL>URL</refURL>
        </skillNameUriPair>
        <weight>1</weight>
      </skillCompetency>
      <skillCompetency>
        <competencelevel>5</competencelevel>
        <skillNameUriPair name="skill2">
          <refURL>URL</refURL>
        </skillNameUriPair>
        <weight>1</weight>
      </skillCompetency>
      <selectionCriteria>Longest Available</selectionCriteria>
    </skillGroup>
  </poolSpecificInfo>
</csq>
```

Sample Input XML—New Email CSQ with PLAIN emailAuthType

```

<csq xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="csq.xsd">
  <self href="" rel="" type="" />
  <name>newEmail73</name>
  <queueType>EMAIL</queueType>
  <routingType>NONINTERACTIVE</routingType>
  <queueAlgorithm>FIFO</queueAlgorithm>
  <autoWork>true</autoWork>
  <wrapupTime>1</wrapupTime>
  <resourcePoolType>SKILL_GROUP</resourcePoolType>
  <serviceLevel>5</serviceLevel>
  <serviceLevelPercentage>70</serviceLevelPercentage>
  <poolSpecificInfo>
    <skillGroup>
      <skillCompetency>
        <competencelevel>5</competencelevel>
        <skillNameUriPair name="skill1">
          <refURL></refURL>
        </skillNameUriPair>
        <weight>1</weight>
      </skillCompetency>
      <selectionCriteria>Longest Available</selectionCriteria>
    </skillGroup>
  </poolSpecificInfo>
  <accountUserId>abc@cisco.com</accountUserId>
  <accountPassword>abc123</accountPassword>
  <reviewQueue name="test_CSQ_1">
    <refURL></refURL>
  </reviewQueue>
  <channelProvider name="3">
    <refURL>https://uccx-server/adminapi/channelServer/3</refURL>
  </channelProvider>
  <pollingInterval>600</pollingInterval>
  <folderName>Inbox</folderName>
  <snapshotAge>120</snapshotAge>
</csq>

```

Modify CSQ

HTTPS Method	PUT
URI	https://<server>/adminapi/csq/{id}
Example URI	https://uccxserver/adminapi/csq/22
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Sample Input XML—Modify a CSQ (PUT)

```
<csq xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="csq.xsd">
  <self href="" rel="" type="" />
  <name>Modifiedchatcsq</name>
  <queueType>CHAT</queueType>
  <queueAlgorithm>FIFO</queueAlgorithm>
  <resourcePoolType>SKILL_GROUP</resourcePoolType>
  <poolSpecificInfo>
    <skillGroup>
      <skillCompetency>
        <competencelevel>8</competencelevel>
        <skillNameUriPair name="skill1">
          <refURL>URL</refURL>
        </skillNameUriPair>
        <weight>1</weight>
      </skillCompetency>
      <skillCompetency>
        <competencelevel>10</competencelevel>
        <skillNameUriPair name="skill2">
          <refURL>URL</refURL>
        </skillNameUriPair>
        <weight>1</weight>
      </skillCompetency>
      <selectionCriteria>Longest Available</selectionCriteria>
    </skillGroup>
  </poolSpecificInfo>
</csq>
```

Sample Input XML—New Email CSQ with PLAIN emailAuthType

```
<csq xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="csq.xsd">
  <self href="" rel="" type="" />
  <name>modifiedEmailCsq</name>
  <queueType>EMAIL</queueType>
  <routingType>NONINTERACTIVE</routingType>
  <queueAlgorithm>FIFO</queueAlgorithm>
  <autoWork>true</autoWork>
  <wrapupTime>1</wrapupTime>
  <resourcePoolType>SKILL_GROUP</resourcePoolType>
  <serviceLevel>5</serviceLevel>
  <serviceLevelPercentage>70</serviceLevelPercentage>
  <poolSpecificInfo>
    <skillGroup>
      <skillCompetency>
        <competencelevel>5</competencelevel>
        <skillNameUriPair name="skill1">
          <refURL></refURL>
        </skillNameUriPair>
        <weight>1</weight>
      </skillCompetency>
      <selectionCriteria>Longest Available</selectionCriteria>
    </skillGroup>
  </poolSpecificInfo>
  <accountUserId>abc@cisco.com</accountUserId>
  <accountPassword>abc123</accountPassword>
  <reviewQueue name="test_CSQ_1">
    <refURL></refURL>
  </reviewQueue>
  <channelProvider name="3">
    <refURL>https://uccx-server/adminapi/channelServer/3</refURL>
  </channelProvider>
  <pollingInterval>600</pollingInterval>
  <folderName>Inbox</folderName>
  <snapshotAge>120</snapshotAge>
</csq>
```

Delete CSQ

HTTPS Method	DELETE
URI	https://<server>/adminapi/csq/{id}
Example URI	https://uccxserver/adminapi/csq/22
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405, 409
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Data Source

Data sources are databases that are configured to communicate with the Unified CCX system.

Data Structure for Data Source

Attribute Name	Base Type	Restrictions	Description
id	int	minOccurs = 0 maxOccurs = 1 nillable = false	Unique identifier for the data source
name	string	minLength value = 1 maxLength value = 65 minOccurance = 1 maxOccurance = 1 Data source name contains alphanumeric characters, spaces, underscores (_), hyphens (-), plus signs (+), pipes (), colons (:), periods (.), and slashes (/) without leading or trailing white spaces.	Unique name for the data source
userName	string	minLength value = 1 maxLength value = 128 minOccurences = 1 maxOccurences = 1 Special characters like , {, }, <, >, ', \", (,), \, /, %, \+, ?, *, &, and comma are not allowed.	Username that is defined for connecting to the enterprise database
password	string	minLength value = 1 maxLength value = 128 minOccurences = 1 Special characters like &, \", <, >, \, %, and comma are not allowed.	Password that is defined for connecting to the enterprise database
maxConnection	int	minInclusive value = 2 maxInclusive value = 10000 minOccurences = 1	Maximum number of connections that are allowed to connect to the database

Attribute Name	Base Type	Restrictions	Description
driver	string	enumeration <ul style="list-style-type: none"> oracle.jdbc.driver.OracleDriver com.ibm.db2.jcc.DB2Driver net.sourceforge.jtds.jdbc.Driver com.sysbase.jdbc2.jdbc.SybDriver 	Driver for this data source to communicate to the enterprise database
jdbcURL	string	minLength value = 1 maxLength value = 256 minOccurrences = 1	JDBC URL that is used to obtain a connection to the enterprise database

Get List of Data Sources

HTTPS Method	GET
URI	https://<server>/adminapi/dataSource
Example URI	https://uccx-server/adminapi/datasource
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Code	401, 404, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<datasources>
<datasource>
<self>URL</self>
<id>15</id>
<name>oracle1</name>
<userName>System</userName>
<password>MTJDY2JlMTI=</password>
<maxConnection>10</maxConnection>
<driver>oracle.jdbc.driver.OracleDriver</driver>
<jdbcURL>jdbc:oracle:thin://@uccx-server:1521/XE</jdbcURL>
</datasource>
</datasources>
```

Get Data Source

HTTPS Method	GET
URI	https://<server>/adminapi/dataSource/{id}

Example URI	https://uccx-server/adminapi/dataSource/21
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Code	401, 404, 405
Error Codes	HTTPS/1.1 404 Not Found

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<DataSource>
<self>https://uccx-server/adminapi/dataSource/</self>
<id>15</id>
<name>oracle1</name>
<userName>System</userName>
<password>MTJDY2JlMTI=</password><maxConnection>10</maxConnection>
<driver>oracle.jdbc.driver.OracleDriver</driver>
<jdbcURL>jdbc:oracle:thin://@uccx-server:1521/XE</jdbcURL>
</DataSource>
```

Create Data Source

HTTPS Method	POST
URI	https://<server>/adminapi/dataSource
Example URI	https://uccx-server/adminapi/dataSource
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Code	400, 401, 405
Error Codes	HTTPS/1.1 400 Bad Request

Success Response

HTTPS/1.1 201 Created

Sample Input XML—Create DataSource (POST)

```
<?xml version="1.0" encoding="UTF-8"?>
<DataSource>
  <name>oracle1</name>
  <userName>System</userName>
  <password>MTJDY2JlMTI=</password>
  <maxConnection>10</maxConnection>
  <driver>oracle.jdbc.driver.OracleDriver</driver>
  <jdbcURL>jdbc:oracle:thin://@uccx-server:1521/XE</jdbcURL>
</DataSource>
```

Modify Data Source

HTTPS Method	PUT
URI	https://<server>/adminapi/dataSource/{id}
Example URI	https://uccx-server/adminapi/dataSource/21
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Code	400, 401, 404, 405
Error Codes	HTTPS/1.1 400 Bad Request

Success Response

HTTPS/1.1 200 OK

Sample Input XML—Modify DataSource (PUT)

```
<?xml version="1.0" encoding="UTF-8"?>
<DataSource>
  <self>URL</self>
  <id>15</id>
  <name>oracle1</name>
  <userName>System2</userName>
  <password>MTJDY2JlMTI=</password>
  <maxConnection>10</maxConnection>
  <driver>oracle.jdbc.driver.OracleDriver</driver>
  <jdbcURL>jdbc:oracle:thin://@uccx-server:1521/XE</jdbcURL>
</DataSource>
```

Delete Data Source

HTTPS Method	DELETE
URI	https://<server>/adminapi/dataSource/{id}
Example URI	https://uccx-server/adminapi/dataSource/21
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Code	401, 404, 405
Error Codes	HTTPS/1.1 404 Not Found

Success Response

HTTPS/1.1 200 OK

Test Connection for Data Source

HTTPS Method	GET
URI	https://<server>/adminapi/dataSource/{id}/testConnection
Example URI	https://uccx-server/adminapi/dataSource/30/testConnection
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Code	401, 404, 405
Error Codes	<?xml version="1.0" encoding="UTF-8" standalone="yes"?> <DatabaseTestResult><testResult>>false</testResult></DatabaseTestResult>

Success Response

HTTPS/1.1 200 OK

DB Purge Schedule Configuration

DB purge schedule configuration schedules a DB purge with given configurations and triggers automatic purge at a given time stamp.

Data Structure for DB Purge Schedule Configuration

Attribute Name	Base Type	Restrictions	Description
retaininterval	int	minInclusive = 1 maxInclusive = 99	Data older than the specified number of months to be purged.
hrdbdays	int	minInclusive = 1 maxInclusive = 99	Data older than the specified number of days to be purged.
hrdbmaxsize	int	minInclusive = 5 maxInclusive = 95	Percentage of the maximum database size at which an automatic purge is initiated.

Attribute Name	Base Type	Restrictions	Description
scheduledTime	string	nillable= false Regex: [(0[1-9] 1[0-2]):[0] (AM PM)]	Time of the day for the daily purge with the time zone. The time displayed here is based on the primary time zone specified during initial setup of Unified CCX Administration. Note In a High Availability over WAN deployment, the purge schedule happens at the time zone of the primary node.
purge run time	int	minInclusive = 1 maxInclusive = 23	The total duration for which the purge process should run.
initiate automatic purge when extent size exceeds	int	minInclusive = 60 maxInclusive = 90	Percentage of the maximum extents size of any table above which an automatic purge is initiated.

Get Purge Configuration Information

HTTPS Method	GET
URI	https://<server>/adminapi/dbpurge
Example URI	https://uccx-server/adminapi/dbpurge
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<dbpurge>
  <retaininterval>6</retaininterval>
  <hrdbdays>1</hrdbdays>
  <hrdbmaxsize>5</hrdbmaxsize>
  <scheduledTime>04:00 PM</scheduledTime>
</dbpurge>
```

Update Purge Configuration Information

HTTPS Method	PUT
---------------------	-----

URI	https://<server>/adminapi/dbpurge
Example URI	https://uccx-server/adminapi/dbpurge
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Sample Input XML — Modify a DB Purge Configuration (PUT)

```
<?xml version="1.0" encoding="UTF-8"?>
<dbpurge>
  <retaininterval>months</retaininterval>
  <hrdbdays>days</hrdbdays>
  <hrdbmaxsize>dbsize</hrdbmaxsize>
  <scheduledTime>time</scheduledTime>
</dbpurge>
```

DB Purge Now Configuration

DB purge now configuration purges the database manually at any time with the given configuration values.

Data Structure for DB Purge Now Configuration

Attribute Name	Base Type	Restrictions	Description
months	int	minInclusive = 0 maxInclusive = 99	Data older than the specified number of months to be purged.
days	int	minInclusive = 1 maxInclusive = 99	Data older than the specified number of days to be purged.
purge run time	int	minInclusive = 1 maxInclusive = 23	The total duration for which the purge process should run.

Update Purge Now Configuration

HTTPS Method	PUT
URI	https://<server>/adminapi/dbpurgeNow

Example URI	https://uccx-server/adminapi/dbpurgeNow
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Sample Input XML — Modify and initiate a DB Purge now (PUT)

```
<?xml version="1.0" encoding="UTF-8"?>
<dbpurgeNow>
  <months>months</months>
  <days>days</days>
</dbpurgeNow>
```

DB Purge Configuration Status

DB purge configuration status gets the status of manual and schedule purge.

Data Structure for DB Purge Configuration Status

Attribute Name	Base Type	Restrictions	Description
purgeState	string	minOccurs = 1 maxOccurs = 1 nillable = false	State of the purge.
purgeTimeStamp	string	minOccurs = 1 maxOccurs = 1 nillable = false	Time stamp of the purge.
purgeType	string	minOccurs = 1 maxOccurs = 1 nillable = false	Type of the purge.

Get Purge Configuration Status

HTTPS Method	GET
---------------------	-----

URI	https://<server>/adminapi/purgeConfigStatus
Example URI	https://uccx-server/adminapi/purgeConfigStatus
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<purgeConfigStatus>
  <purgeState>COMPLETED_SUCCESSFULLY</purgeState>
  <purgeTimeStamp>Mon Sep 30 12:00:24 PM 2013</purgeTimeStamp>
  <purgeType>SCHEDULED</purgeType>
</purgeConfigStatus>
```

Dialog Group

A dialog group is a pool of dialog channels in which each channel is used to perform dialog interactions with a caller, during which the caller responds to automated prompts by pressing buttons on a touchtone phone.

Data Structure for Dialog Group

Attribute Name	Base Type	Restrictions	Description
id	int	Min Value = 0 Max Value = 65535 Multiplicity = Required	Numeric identifier of the application
dialogGroupDesc	string	Min length = 1 Max length = 80 Multiplicity = Required Dialog Group Description contains alphanumeric characters, special characters, spaces, underscores (_), and hyphens (-) without leading or trailing white spaces.	Descriptor for the dialog group

Attribute Name	Base Type	Restrictions	Description
groupType	string	Multiplicity = Required Enumeration = <ul style="list-style-type: none"> • Cisco Media Termination • MRCP ASR 	Type of the group, corresponding to type of the channels that are managed by the group as defined.
Cisco Media Termination	Cisco Media Termination	Min length = 1 Max length = 65535 Multiplicity = Required	Media termination value
MRCPASR	MRCPASR	Min length = 1 Max length = 20 Multiplicity = Required Nillable = False MRCPASR contains alphanumeric characters, special characters, spaces, underscores (_), and hyphens (-) without leading or trailing white spaces.	Value for MRCPASR

Get List of Dialog Groups

HTTPS Method	GET
URI	https://<server>/adminapi/dialogGroup
Example URI	https://uccx-server/adminapi/dialogGroup
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<dialogGroups>
  <dialogGroup>
    <self>https://uccx-server/adminapi/dialogGroup/0</self>
    <id>0</id>
    <dialogGroupDesc>Default</dialogGroupDesc>
    <groupType>Cisco Media Channel</groupType>
```

```

<CiscoMediaTermination>
  <channels>300</channels>
</CiscoMediaTermination>
</dialogGroup>
<dialogGroup>
  <self>https://uccx-server/adminapi/dialogGroup/1</self>
  <id>1</id>
  <dialogGroupDesc>CMT Auto#8</dialogGroupDesc>
  <groupType>Cisco Media Channel</groupType>
  <CiscoMediaTermination>
    <channels>1</channels>
  </CiscoMediaTermination>
</dialogGroup>
</dialogGroups>

```

Get Dialog Group

HTTPS Method	GET
URI	https://<server>/adminapi/dialogGroup/<dialog group id>
Example URI	https://uccx-server/adminapi/dialogGroup/1
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—

Success Response

```

<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<dialogGroup>
  <self>https://uccx-server/adminapi/dialogGroup/1</self>
  <id>1</id>
  <dialogGroupDesc>CMT Auto#8</dialogGroupDesc>
  <groupType>Cisco Media Channel</groupType>
  <CiscoMediaTermination>
    <channels>1</channels>
  </CiscoMediaTermination>
</dialogGroup>

```

Create Dialog Group

HTTPS Method	POST
URI	https://<server>/adminapi/dialogGroup
Example URI	https://uccx-server/adminapi/dialogGroup
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201

HTTPS Failure Code	401, 405, 409
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Modify Dialog Group

HTTPS Method	PUT
URI	https://<server>/adminapi/dialogGroup
Example URI	https://uccx-server/adminapi/dialogGroup/1
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Delete Dialog Group

HTTPS Method	DELETE
URI	https://<server>/adminapi/dialogGroup/<dialog group id>
Example URI	https://uccx-server/adminapi/dialogGroup/1
Content Type	Application/XML, Application/JSON
HTTPS Success code	200
HTTPS Failure Codes	401, 404, 412
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Email SMTP Server Configuration

The Unified CCX system uses the eMail subsystem to communicate with your email server and enable your applications to create and send email.

Datastructure for Email SMTP Server Configuration

Attribute Name	Base Type	Restrictions	Description
mail_server	xs:string	minOccurs = 1 maxOccurs = 1 Regex: [A-Za-z0-9_.\-]+\d{1,3}\d{1,3}\d{1,3}\d{1,3}	Email SMTP server address
email_address	xs:string	minOccurs = 1 maxOccurs = 1 Regex: ([0-9a-zA-Z][-_&])*[0-9a-zA-Z]@[(-0-9a-zA-Z)[.])[a-zA-Z]+	Sender address, usually CCX administrator email address.

Get Email SMTP Server Configuration

HTTPS Method	GET
URI	https://<server>/adminapi/email
Example URI	https://uccx-server/adminapi/email
Accept	Text/XML, Text/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 402, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<emailControlGroups>
  <emailControlGroup>
    <mail_server>mail.cisco.com</mail_server>
    <email_address>mailid@cisco.com</email_address>
```

```
</emailControlGroup>
</emailControlGroups>
```

Create Email SMTP Server Configuration

HTTPS Method	POST
URI	https://<server>/adminapi/email
Example URI	https://uccx-server/adminapi/email
Content Type	Text/XML
HTTPS Success Code	200, 201
HTTPS Failure Code	401, 402, 405
Error Codes	—

Success Response

https://uccx-server/adminapi/email/0

Modify Email SMTP Server Configuration

HTTPS Method	PUT
URI	https://<server>/adminapi/email/<id>
Example URI	https://uccx-server/adminapi/email/0
Content Type	Text/XML
HTTPS Success Code	200
HTTPS Failure Codes	401, 402, 404, 405
Error Codes	—

Success Response

—

Delete Email SMTP Server Configuration

HTTPS Method	DELETE
URI	https://<server>/adminapi/email/<id>
Example URI	https://uccx-server/adminapi/email/0

Content Type	Text/XML
HTTPS Success code	200
HTTPS Failure Codes	401, 402, 404, 405
Error Codes	—

Success Response

—

Files (Script, Prompt, Grammar, and Document)

Unified CCX applications can make use of many auxiliary files that interact with callers, such as scripts, pre-recorded prompts, grammars, and custom Java classes.

To manage these auxiliary files, you must complete the following tasks:

- Provision telephony and media resources.
- Provision Unified CCX subsystem, if required.
- Provision additional subsystems, if required.
- Configure Cisco script applications.

Depending on your particular Unified CCX implementation, your applications might make use of some or all the file types. Any file you create needs to be available to the Unified CCX engine before a Unified CCX application can use them. This is done through the Unified CCX cluster's Repository datastore, where the script, prompt, grammar, and document files are created, stored, and updated.

Data Structure for Files

Attribute Name	Base Type	Restrictions	Description
Script Prompt Grammar Document AARDocument	xs: string	minOccurs = 1 maxOccurs = unbounded	Auxiliary files.
Folder	xs: string	minOccurs = 0 maxOccurs = unbounded	Path of the current item with respect to the root folder.
File	xs: string	minOccurs = 0 maxOccurs = unbounded	File information.

Attribute Name	Base Type	Restrictions	Description
LanguageDetails	xs: string	minOccurs = 0 maxOccurs = unbounded	Details about the language.
Path	xs: string	nillable = false	Regex: ([^\<%&* '?:"])+
FileName	xs: string	nillable = false	Name of the file. Regex: ([^\<%&* '?:"])+
Details	xs: string	minOccurs = 0 maxOccurs = 1	
ReNameTo	xs: string	minOccurs = 0 maxOccurs = 1	Lets you rename an existing file.
AARFileName	xs: string	minLength = 1 maxLength = 64	
LanguageName	xs: string	nillable = false	Regex: ([^\<%&*() {} '+'?"])+
FolderName	xs: string	nillable = false	Regex: ([^\<%&* '?:"])+
Size	xs: string	nillable = true	The size of the file prefixed with KB. The file size is converted from bytes to KB.
DateModified	xs: string	nillable = true	The date and time when the document was last uploaded or changed along with time zone.
modifiedBy	xs: string	nillable = true	The user ID of the person who performed these modifications.

Create a New Language for Prompt

HTTPS Method	POST
URI	https://<server>/adminapi/prompt
Example URI	https://uccx-server/adminapi/prompt
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201

HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Sample Input XML — Create a Language

```
<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Files.xsd">
  <Prompt>
    <LanguageDetails>
      <path>/</path>
      <LanguageName>FTCustomPromptLanguage1</LanguageName>
    </LanguageDetails>
  </Prompt>
</Files>
```

Create a New Language for Document

HTTPS Method	POST
URI	https://<server>/adminapi/document
Example URI	https://uccx-server/adminapi/document
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Sample Input XML — Create a Language

```
<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Files.xsd">
  <Document>
    <LanguageDetails>
      <path>/</path>
      <LanguageName>FTCustomDocLanguage1</LanguageName>
    </LanguageDetails>
  </Document>
</Files>
```

Create a New Language for Grammar

HTTPS Method	POST
URI	https://<server>/adminapi/grammar
Example URI	https://uccx-server/adminapi/grammar
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Sample Input XML — Create a Language

```
<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Files.xsd">
  <Grammar>
    <LanguageDetails>
      <path>/</path>
      <LanguageName>FTCustomGrammarLanguage1</LanguageName>
    </LanguageDetails>
  </Grammar>
</Files>
```

Create a New Folder

You must create a folder to store the files that the Unified CCX system can use.

Create a New Folder for Script

HTTPS Method	POST
URI	https://<server>/adminapi/script
Example URI	https://uccx-server/adminapi/script
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Sample Input XML — Create a Folder

```
<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Files.xsd">
  <Script>
    <Folder>
      <path>/</path>
      <FolderName>FTCustomScriptF1</FolderName>
    </Folder>
  </Script>
</Files>
```

Create a New Folder for Prompt

HTTPS Method	POST
URI	https://<server>/adminapi/prompt
Example URI	https://uccx-server/adminapi/prompt
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Sample Input XML — Create a Folder

```
<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Files.xsd">
  <Prompt>
    <Folder>
      <path>/FTCustomPromptLanguage1</path>
      <FolderName>FTCustomPromptFolder1</FolderName>
    </Folder>
  </Prompt>
</Files>
```

Create a New Folder for Document

HTTPS Method	POST
URI	https://<server>/adminapi/document
Example URI	https://uccx-server/adminapi/document

Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Sample Input XML — Create a Folder

```
<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Files.xsd">
  <Document>
    <Folder>
      <path>/FTCustomDocLanguage1</path>
      <FolderName>FTCustomDocFolder1</FolderName>
    </Folder>
  </Document>
</Files>
```

Create a New Folder for Grammar

HTTPS Method	POST
URI	https://<server>/adminapi/grammar
Example URI	https://uccx-server/adminapi/grammar
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

HTTPS/1.1 201 No Content

Sample Input XML — Create a Folder

```
<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Files.xsd">
  <Grammar>
    <Folder>
      <path>/FTCustomGrammarLanguage1</path>
      <FolderName>FTCustomGrammarFolder1</FolderName>
    </Folder>
  </Grammar>
</Files>
```

Upload Files to a Temporary Folder

You must upload the file to a temporary location before storing it in the actual folder path.

Procedure

-
- Step 1** From the **REST API Client**, Choose the **Content Type** as multipart/form-data.
 - Step 2** In the *Key-Value* pair fields, enter **file** in the Key field. For the Value field, Click **Choose Files** to navigate to the directory that contains the required file.
 - Step 3** Click the **Send** button to upload the new file to the repository. Display of the **Success Code** confirms the successful upload of the files.

Note If you try to upload invalid files, the upload will be unsuccessful and the **Failure Codes** will be displayed.

Upload Files to a Temporary Folder for Script

HTTPS Method	POST
URI	https://<server>/adminapi/script/uploadFile
Example URI	https://uccx-server/adminapi/script/uploadFile
Content Type	multipart/form-data
HTTPS Success Code	201
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Upload Files to a Temporary Folder for Prompt

HTTPS Method	POST
URI	https://<server>/adminapi/prompt/uploadFile
Example URI	https://uccx-server/adminapi/prompt/uploadFile
Content Type	multipart/form-data
HTTPS Success Code	201
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Upload Files to a Temporary Folder for Document

HTTPS Method	POST
URI	https://<server>/adminapi/document/uploadFile
Example URI	https://uccx-server/adminapi/document/uploadFile
Content Type	multipart/form-data
HTTPS Success Code	201
HTTPS Failure Codes	401, 405, and 500
Failure Response	HTTPS/1.1 400 Bad Request
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Upload Files to a Temporary Folder for Grammar

HTTPS Method	POST
URI	https://<server>/adminapi/grammar/uploadFile
Example URI	https://uccx-server/adminapi/grammar/uploadFile
Content Type	multipart/form-data
HTTPS Success Code	201

HTTPS Failure Codes	401, 405, and 500
Failure Response	HTTPS/1.1 400 Bad Request
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Store Files in Database from a Temporary folder

The files uploaded in the Temporary location is retrieved and stored in the Folder path specified. The Content Type should be either Application/XML or Application/JSON.

Store Files in Database from a Temporary Folder for Script

HTTPS Method	POST
URI	https://<server>/adminapi/script
Example URI	https://uccx-server/adminapi/script
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	401, 405, and 500
Failure Response	HTTPS/1.1 400 Bad Request
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Sample Input XML — Create a Configuration Object for a VXML File

```
<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Files.xsd">
  <Script>
    <File>
      <path>/FTCustomScriptFolder/FTCustomScriptFolder1/</path>
      <FileName>FTCustomScript3.vxml</FileName>
    </File>
  </Script>
</Files>
```

Store Files in Database from a Temporary Folder for Prompt

HTTPS Method	POST
---------------------	------

URI	https://<server>/adminapi/prompt
Example URI	https://uccx-server/adminapi/prompt
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	401, 405, 500
Failure Response	HTTPS/1.1 400 Bad Request
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Sample Input XML — Create a Configuration Object for a VXML File

```
<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Files.xsd">
  <Prompt>
    <File>
      <path>/FTCustomPromptLanguage1/FTCustomPromptFolder1/</path>
      <FileName>FTCustomPrompt3.vxml</FileName>
    </File>
  </Prompt>
</Files>
```

Store Files in Database from a Temporary Folder for Document

HTTPS Method	POST
URI	https://<server>/adminapi/document
Example URI	https://uccx-server/adminapi/document
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Sample Input XML — Create a Configuration Object for a VXML File

```
<?xml version="1.0" encoding="UTF-8"?>
```



```
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Files.xsd">
  <Document>
    <File>
      <path>/FTCustomDocLanguage1/FTCustomDocFolder1/</path>
      <FileName>FTCustomDoc3.vxml</FileName>
    </File>
  </Document>
</Files>
```

Store Files in Database from a Temporary Folder for Grammar

HTTPS Method	POST
URI	https://<server>/adminapi/grammar
Example URI	https://uccx-server/adminapi/grammar
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Sample Input XML — Create a Configuration Object for a VXML File

```
<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Files.xsd">
  <Grammar>
    <File>
      <path>/FTCustomGrammarLanguage1/FTCustomGrammarFolder1/</path>
      <FileName>FTCustomGrammar3.vxml</FileName>
    </File>
  </Grammar>
</Files>
```

Get List for Script Files and Folder

HTTPS Method	GET
URI	https://<server>/adminapi/script
Example URI	https://uccx-server/adminapi/script
Accept	Application/XML, Application/JSON
HTTPS Success Code	200

HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

```
<files>
  <Script>
    <Folder>
      <path>/default/FTCustomScriptF11363861050449</path>
      <FolderName>FTCustomScriptF11363861050449</FolderName>
      <Details><size>0 KB</size><dateModified>03/21/2013 03:48:00 PM India
        Standard Time</dateModified><modifiedBy>Administrator</modifiedBy></Details>
    </Folder>
  </Script>
</files>
```

Get List for Prompt Files and Folder

HTTPS Method	GET
URI	https://<server>/adminapi/prompt
Example URI	https://uccx-server/adminapi/prompt
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

```
<files>
  <Prompt>
    <Folder>
      <path>/en</path>
      <FolderName>en</FolderName>
      <Details><size>0 KB</size><dateModified>03/21/2013 03:47:24 PM India
        Standard Time</dateModified><modifiedBy>Administrator</modifiedBy></Details>
    </Folder>
  </Prompt>
</files>
```

Get List for Grammar Files and Folder

HTTPS Method	GET
URI	https://<server>/adminapi/grammar
Example URI	https://uccx-server/adminapi/grammar
Accept	Application/XML, Application/JSON

HTTPS Success Code	200
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

```
<files>
  <Grammar>
    <Folder>
      <path>/en/</path>
      <FolderName>en</FolderName>
      <Details><size>0 KB</size><dateModified>03/21/2013 03:47:24 PM India
        Standard Time</dateModified><modifiedBy>Administrator</modifiedBy></Details>
    </Folder>
  </Grammar>
</files>
```

Get List for Document Files and Folder

HTTPS Method	GET
URI	https://<server>/adminapi/document
Example URI	https://uccx-server/adminapi/document
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

```
<Files>
  <Document>
    <Folder>
      <path>/default/FTCustomDocF11363861050449/</path>
      <FolderName>FTCustomScriptF11363861050449</FolderName>
      <Details><size>0 KB</size><dateModified>03/21/2013 03:48:00 PM India
        Standard Time</dateModified><modifiedBy>Administrator</modifiedBy></Details>
    </Folder>
  </Document>
</Files>
```

Get Script Files and Folder

HTTPS Method	GET
---------------------	-----

URI	https://<server>/adminapi/script/{FilePath/FolderPath} Note FilePath is the fully qualified name of the file. FolderPath is the fully qualified name of the folder.
Example URI	https://uccx-server/adminapi/script/FolderPath/ https://uccx-server/adminapi/script/FilePath/
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

```
<Files>
  <Script>
    <Folder>
      <path>/FolderName//</path>
      <FolderName>SUBFolderName</FolderName>
      <Details><size>0 KB</size><dateModified>03/21/2013 03:48:01 PM India
        Standard Time</dateModified><modifiedBy>Administrator</modifiedBy></Details>
    </Folder>
  </Script>
</Files>
```

Get Prompt Files and Folder

HTTPS Method	GET
URI	https://<server>/adminapi/prompt/{FilePath/FolderPath} Note FilePath is the fully qualified name of the file. FolderPath is the fully qualified name of the folder.
Example URI	https://uccx-server/adminapi/prompt/FolderPath/ https://uccx-server/adminapi/prompt/FilePath/
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

```
<Files>
  <Prompt>
    <Folder>
      <path>/FolderName//</path>
```

```

    <FolderName>SUBFolderName</FolderName>
    <Details><size>0 KB</size><dateModified>03/21/2013 03:48:01 PM India
    Standard Time</dateModified><modifiedBy>Administrator</modifiedBy></Details>
  </Folder>
</Prompt>
</Files>

```

Get Grammar Files and Folder

HTTPS Method	GET
URI	https://<server>/adminapi/grammar/{FilePath/FolderPath} Note FilePath is the fully qualified name of the file. FolderPath is the fully qualified name of the folder.
Example URI	https://uccx-server/adminapi/grammar/FolderPath/ https://uccx-server/adminapi/grammar/FilePath/
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

```

<Files>
  <Grammar>
    <Folder>
      <path>/FolderName//</path>
      <FolderName>SUBFolderName</FolderName>
      <Details><size>0 KB</size><dateModified>03/21/2013 03:48:01 PM India
      Standard Time</dateModified><modifiedBy>Administrator</modifiedBy></Details>
    </Folder>
  </Grammar>
</Files>

```

Get Document Files and Folder

HTTPS Method	GET
URI	https://<server>/adminapi/document/{FilePath/FolderPath} Note FilePath is the fully qualified name of the file. FolderPath is the fully qualified name of the folder.
Example URI	https://uccx-server/adminapi/document/FolderPath/ https://uccx-server/adminapi/document/FilePath/
Accept	Application/XML, Application/JSON

HTTPS Success Code	200
HTTPS Failure Codes	401, 405, 500
Error Codes	—

Success Response

```

<Files>
  <Document>
    <Folder>
      <path>/FolderName//</path>
      <FolderName>SUBFolderName</FolderName>
      <Details><size>0 KB</size><dateModified>03/21/2013 03:48:01 PM India
        Standard Time</dateModified><modifiedBy>Administrator</modifiedBy></Details>
    </Folder>
  </Document>
</Files>

```

Get FreeSpace and UsedSpace for Script/Prompt/Grammar/Document

HTTPS Method	GET
URI	https://<server>/adminapi/ResourceName}/space/{FolderPath}/ Where "ResourceName" is the "script/prompt/grammar/document" and FolderPath is a fully qualified name of the folder.
Example URI	https://uccx-server:8080/adminapi/prompt/space/default/ https://uccx-server:8080/adminapi/script/space/default/ https://uccx-server:8080/adminapi/document/space/default/ https://uccx-server:8080/adminapi/grammar/space/default/
Accept	Application/XML
HTTPS Success Code	200
HTTPS Failure Codes	401, 402, 405, 500
Error Codes	—

Success Response

200 OK

Download Script/Prompt

HTTPS Method	GET
---------------------	-----

URI	https://<server>/adminapi/ResourceName}/download/{FilePath} Where "ResourceName" is the "script/prompt" and FilePath is a fully qualified name of the file.
Example URI	https://uccx-server:8080/adminapi/prompt/download/default/beep.wav https://uccx-server:8080/adminapi/script/download/default/icd_brkpt.aef
Accept	Application/XML
HTTPS Success Code	200
HTTPS Failure Codes	401, 402, 405, 500
Error Codes	—

Success Response

200 OK

Modify Script Folder Name

HTTPS Method	PUT
URI	https://<server>/adminapi/script/FolderPath
Example URI	https://uccx-server/adminapi/script/folder/
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	404, 500
Error Codes	—

Success Response

—

Modify File

HTTPS Method	PUT
URI	https://<server>/adminapi/{ResourceName}/FilePath
Example URI	https://uccx-server/adminapi/{ResourceName}/CustomScript1.aef Note ResourceName can be script, prompt, grammar, or document.
Content Type	Application/XML, Application/JSON

HTTPS Success Code	200
HTTPS Failure Codes	404, 500
Error Codes	—

Success Response

—

Sample Input XML — Modifying a Script File Name

```
<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="../adminapi/src/main/resources/xsd/Files.xsd">
  <Script>
    <File>
      <path>/</path>
      <FileName>CustomScript1.aef</FileName>
      <reNameTo>CustomScript1Mod.aef</reNameTo>
    </File>
  </Script>
</Files>
```

Sample Input XML — Refreshing a Script File Content

```
<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="../adminapi/src/main/resources/xsd/Files.xsd">
  <Script>
    <File>
      <path>/</path>
      <FileName>CustomScript1.aef</FileName>
      <refresh>true</refresh>
    </File>
  </Script>
</Files>
```



Note If you want to upload the modified script file, upload the file first and use PUT operation to store the uploaded script. If the refresh attribute is set to true, application immediately uses the uploaded script.

Modify Prompt Folder Name or Language Name

HTTPS Method	PUT
URI	https://<server>/adminapi/prompt/FolderPath https://<server>/adminapi/prompt/LanguagePath
Example URI	https://uccx-server/adminapi/prompt/language/ https://uccx-server/adminapi/prompt/folder/
Content Type	Application/XML, Application/JSON

HTTPS Success Code	200
HTTPS Failure Codes	404, 500
Error Codes	—

Success Response

—

Sample Input XML — Modifying a Folder

```
<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Files.xsd">
  <Prompt>
    <Folder>
      <path>/PrmoptLanguage1</path>
      <FolderName>PrmoptFolder1</FolderName>
      <reNameTo>ModPrmoptFolder1</reNameTo>
    </Folder>
  </Prompt>
</Files>
```

Modify Grammar Folder Name or Language Name

HTTPS Method	PUT
URI	https://<server>/adminapi/grammar/FolderPath https://<server>/adminapi/grammar/LanguagePath
Example URI	https://uccx-server/adminapi/grammar/language/ https://uccx-server/adminapi/grammar/folder/
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	404, 500
Error Codes	—

Success Response

—

Sample Input XML — Renaming the Language

```
<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Files.xsd">
  <Grammar>
    <LanguageDetails>
      <path></path>
```

```

    <LanguageName>FTCustomDocLanguage1</LanguageName>
    <reNameTo>FTCustomModDocLanguage1</reNameTo>
  </LanguageDetails>
</Grammar>
</Files>

```

Sample Input XML — Renaming the Folder

```

<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Files.xsd">
  <Grammar>
    <Folder>
      <path>/FTCustomModDocLanguage1</path>
      <FolderName>FTCustomDocumentFolder1</FolderName>
      <reNameTo>FTCustomModDocumentFolder1</reNameTo>
    </Folder>
  </Grammar>
</Files>

```

Modify Document Folder Name or Language Name

HTTPS Method	PUT
URI	https://<server>/adminapi/document/FolderPath https://<server>/adminapi/document/LanguagePath
Example URI	https://uccx-server/adminapi/document/language/ https://uccx-server/adminapi/document/folder/
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	404, 500
Error Codes	—

Success Response

—

Sample Input XML — Renaming the Language

```

<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Files.xsd">
  <Document>
    <LanguageDetails>
      <path>/</path>
      <LanguageName>FTCustomDocLanguage1</LanguageName>
      <reNameTo>FTCustomModDocLanguage1</reNameTo>
    </LanguageDetails>
  </Document>
</Files>

```

Sample Input XML — Renaming the Folder

```
<?xml version="1.0" encoding="UTF-8"?>
<Files xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Files.xsd">
  <Document>
    <Folder>
      <path>/FTCustomModDocLanguage1</path>
      <FolderName>FTCustomDocumentFolder1</FolderName>
      <reNameTo>FTCustomModDocumentFolder1</reNameTo>
    </Folder>
  </Document>
</Files>
```

Delete Script Files and Folder

HTTPS Method	DELETE
URI	https://<server>/adminapi/script/{File/FolderPath}
Example URI	https://uccx-server/adminapi/script/folder/ https://uccx-server/adminapi/script/folder/Test.aef
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	404, 500
Error Codes	—

Success Response

—

Delete Prompt Files and Folder

HTTPS Method	DELETE
URI	https://<server>/adminapi/prompt/{File/FolderPath}
Example URI	https://uccx-server/adminapi/prompt/folder/ https://uccx-server/adminapi/prompt/folder/Test.wav
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	404, 500
Error Codes	—

Success Response

—

Delete Grammar Files and Folder

HTTPS Method	DELETE
URI	https://<server>/adminapi/grammar/{File/FolderPath}
Example URI	https://uccx-server/adminapi/grammar/folder/ https://uccx-server/adminapi/grammar/folder/Test.grxml
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	404, 500
Error Codes	—

Success Response

—

Delete Document Files and Folder

HTTPS Method	DELETE
URI	https://<server>/adminapi/document/{File/FolderPath}
Example URI	https://uccx-server/adminapi/document/folder/ https://uccx-server/adminapi/document/Test.vxml
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	404, 500
Error Codes	—

Success Response

—

HTTPS Trigger

An HTTPS trigger is the relative URL a user enters into the client browser to start the application. You can upload either eXtensible Style Language Transformation (XSLT) templates or Java Server Pages (JSP) templates to serve as your HTTPS trigger.

Data Structure for HTTPS Trigger

Attribute Name	Base Type	Restrictions	Description
name	string	Min length = 2 Max length = 45 HTTPS Trigger name starts with / character and contains alphanumeric and special characters such as -#%& ; = . \ ? / characters without leading and trailing white-space characters.	Name of the HTTPS Trigger
application	nameUriPair	nillable = false minOccurs = 1 minOccurs = 1	Name of application
maxSessions	baseIdInt	nillable = false minOccurs = 1 minOccurs = 1	Maximum number of sessions
idleTimeout	idleTimeout	nillable = false minOccurs = 0 minOccurs = 1 default = 1000	Idle timeout value
locale	string	Min length = 1 Max length = 50 Locale contains alphanumeric characters, special characters, spaces, underscores (_), and hyphens (-) without leading or trailing white spaces.	Defines the locale
enabled	boolean	default = true	Indicates if the HTTPS Trigger is enabled

Sample Input XML

```
<?xml version="1.0" encoding="UTF-8"?>
<httpTrigger>
<self>URL</self>
<name>/test</name>
<application name="app1">
<refURL>URL</refURL>
</application>
<maxSessions>1</maxSessions>
<idleTimeout>1000</idleTimeout>
<locale>en_US</locale>
<enabled>true</enabled>
</httpTrigger>
```

Get List of HTTPS Triggers

HTTPS Method	GET
URI	https://<server>/adminapi/httpTrigger
Example URI	https://uccx-server/adminapi/httpTrigger
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<httpTriggers>
<httpTrigger>
<self>https://uccx-server/adminapi/httpTrigger/redirect_Cisco</self>
<name>/redirect_Cisco</name>
<application name="Auto_App_HTTP">
<refURL>https://uccx-server/adminapi/application/Auto_App_HTTP</refURL>
</application>
<maxSessions>10</maxSessions>
<idleTimeout>2000</idleTimeout>
<locale>en_US</locale>
<enabled>true</enabled>
</httpTrigger>
<httpTrigger>
<self>https://uccx-server/adminapi/httpTrigger/Trigger1355924928134</self>
<name>/Trigger1355924928134</name>
<application name="ForHttpTriggerFT">
<refURL>https://uccx-server/adminapi/application/ForHttpTriggerFT</refURL>
</application>
<maxSessions>1</maxSessions>
<idleTimeout>1</idleTimeout>
<locale>en_US</locale>
<enabled>true</enabled>
</httpTrigger>
<httpTrigger>
<self>https://uccx-server/adminapi/httpTrigger/Trigger</self>
```

```

<name>/Trigger</name>
<application name="ForHttpTriggerFT">
  <refURL>https://uccx-server/adminapi/application/ForHttpTriggerFT</refURL>
</application>
<maxSessions>1</maxSessions>
<idleTimeout>1</idleTimeout>
<locale>en_US</locale>
<enabled>true</enabled>
</httpTrigger>
<httpTrigger>
  <self>https://uccx-server/adminapi/httpTrigger/Trigger1355924936759</self>
  <name>/Trigger1355924936759</name>
  <application name="ForHttpTriggerFT">
    <refURL>https://uccx-server/adminapi/application/ForHttpTriggerFT</refURL>
  </application>
  <maxSessions>15</maxSessions>
  <idleTimeout>1500</idleTimeout>
  <locale>en</locale>
  <enabled>false</enabled>
</httpTrigger>
</httpTriggers>

```

Get HTTPS Trigger

HTTPS Method	GET
URI	https://<server>/adminapi/httpTrigger/{httpTriggerName}
Example URI	https://uccx-server/adminapi/httpTrigger
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—

Success Response

```

<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<httpTrigger>
  <self>https://uccx-server/adminapi/httpTrigger/redirect_Cisco</self>
  <name>/redirect_Cisco</name>
  <application name="Auto_App_HTTP">
    <refURL>https://uccx-server/adminapi/application/Auto_App_HTTP</refURL>
  </application>
  <maxSessions>10</maxSessions>
  <idleTimeout>2000</idleTimeout>
  <locale>en_US</locale>
  <enabled>true</enabled>
</httpTrigger>

```

Create HTTPS Trigger

HTTPS Method	POST
---------------------	------

URI	https://<server>/adminapi/httpTrigger
Example URI	https://uccx-server/adminapi/httpTrigger
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Modify HTTPS Trigger

HTTPS Method	PUT
URI	https://<server>/adminapi/httpTrigger/{httpTriggerName}
Example URI	https://uccx-server/adminapi/httpTrigger/TestHttpTrigger
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Delete HTTPS Trigger

HTTPS Method	DELETE
URI	https://<server>/adminapi/httpTrigger/{httpTriggerName}
Example URI	https://uccx-server/adminapi/httpTrigger/TestHttpTrigger
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Outbound General Configuration

General configuration for Outbound refers to the settings information that is common for all the campaigns.

Data Structure for Outbound General Configuration

Attribute Name	Base Type	Restrictions	Description
startTime	timeTypeOB, string	minOccurs = 1 maxOccurs = 1 Regex: ([01]?[0-9])2[0-3]:[0-5]\d	Outbound campaign start time.
endTime	timeTypeOB, string	minOccurs = 1 maxOccurs = 1 Regex: ([01]?[0-9])2[0-3]:[0-5]\d	Outbound campaign end time.
callTimeout	int	minOccurs = 1 maxOccurs = 1 minInclusive = 5 maxInclusive = 3600	Timeout duration for an agent to respond to the Outbound call.
dialingPrefix	string	minOccurs = 0 maxOccurs = 1 default = "" minLength = 0 maxLength = 5 The dialing prefix can have 4 digits with plus (+) prefix or 5 digits without plus (+) prefix.	Number to prefix to the phone number for dialing outgoing calls.

Attribute Name	Base Type	Restrictions	Description
longDistancePrefix	string	minOccurs = 0 maxOccurs = 1 default = "" minLength = 0 maxLength = 5 The long distance prefix can 4 digits with plus (+) prefix or 5 digits without plus (+) prefix.	Number to prefix to the phone number for dialing long distance.
internationalPrefix	string	minOccurs = 0 maxOccurs = 1 default = "" minLength = 0 maxLength = 5 The international prefix can to 4 digits with plus (+) prefix or 5 digits without plus (+) prefix.	Number to prefix to the international phone numbers.
localAreaCode	string	minOccurs = 0 maxOccurs = 1 default = "" minLength = 0 maxLength = 10 The local area code can have 9 digits with plus (+) prefix or 10 digits without plus (+) prefix.	Area code of the Unified CCX server location.
includeLocalAreaCode	boolean	minOccurs = 0 maxOccurs = 1 default = false	Local area code to be included when you are dialing the phone numbers.
includeLongDistancePrefix	boolean	minOccurs = 0 maxOccurs = 1 default = false	Number to prefix to the local and non-local phone numbers for dialing long distance.

Attribute Name	Base Type	Restrictions	Description
agentAutoAnswer	boolean	minOccurs = 0 maxOccurs = 1 default = true	Unified CCX automatically answers any predictive or progressive campaign call that gets transferred to the agent.
assignedCSQs	csqListOB, int	minOccurs = 0 maxOccurs = 1	CSQs that are used by the Outbound system.
csq	csqDetailsOB, int	minOccurs = 0 maxOccurs = unbounded	CSQs that are assigned for the Outbound campaign with the corresponding percentage allocation.
csqNameUriPair	nameUriPair	—	CSQs that are assigned for the Outbound campaigns.
percentage	int	minInclusive = 1 maxInclusive = 100	The percentage of logged in agents in each of the selected CSQs that are allocated for handling the Outbound calls.

Get List of Skills for Outbound General Configuration

HTTPS Method	GET
URI	https://<server>/adminapi/generalobconfig
Example URI	https://uccx-server/adminapi/generalobconfig
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<generalobconfig xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="../../../main/resources/xsd/generalobconfig.xsd">
  <startTime>08:00</startTime>
  <endTime>21:00</endTime>
  <callTimeout>10</callTimeout>
  <dialingPrefix>1</dialingPrefix>
  <longDistancePrefix>2</longDistancePrefix>
  <internationalPrefix>1</internationalPrefix>
  <localAreaCode>21</localAreaCode>
  <includeLocalAreaCode>>false</includeLocalAreaCode>
  <includeLongDistancePrefix>>false</includeLongDistancePrefix>
  <assignedCSQs>
    <csq>
      <csqNameUriPair name="csq1">
        <refURL> https://uccx-server/adminapi/csq/1</refURL>
      </csqNameUriPair>
      <percentage>10</percentage>
    </csq>
  </assignedCSQs>
</generalobconfig>
```

Modify Outbound General Configuration

HTTPS Method	PUT
URI	https://<server>/adminapi/generalobconfig
Example URI	https://uccx-server/adminapi/generalobconfig
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—

Success Response

HTTPS/1.1 200 OK

System Configuration

Data Structure for System Configuration

Attribute Name	Base Type	Restrictions	Description
notReadyOnSecondaryEnabled	boolean	minOccurs = 0 maxOccurs = 1 nillable = false	notReadyOnSecondaryEnabled: Determines the agent's state change to Not Ready state when the monitored Non ICD lines are used for Incoming or Outgoing calls.

Get System Configuration

HTTPS Method	GET
URI	https://<server>/adminapi/systemConfig
Example URI	https://uccx-Server/adminapi/systemConfig
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 500
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<systemConfig>
  <appConfig>
    <notReadyOnSecondaryEnabled>false</notReadyOnSecondaryEnabled>
  </appConfig>
</systemConfig>
```

Update System Configuration

HTTPS Method	PUT
URI	https://<server>/adminapi/systemConfig
Example URI	https://uccx-Server/adminapi/systemConfig
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200

HTTPS Failure Codes	400, 500
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Sample Input XML — Modify Proxy Configuration (PUT)

```
<?xml version="1.0" encoding="UTF-8"?>
<systemConfig>
  <appConfig>
    <notReadyOnSecondaryEnabled>false</notReadyOnSecondaryEnabled>
  </appConfig>
</systemConfig>
```

Prompt Files Path

Prompt Files Path returns the list of prompts with file path that are uploaded in Unified CCX. This can be invoked by Unified CCX Administrator and Supervisor.

Data Structure for Prompt Files Path

Attribute Name	Base Type	Restrictions	Description
prompts	complexType element containing the complexType element Prompt.	minOccurs = 1 maxOccurs = 1 nillable = false	List of prompts with file path that are uploaded in Unified CCX.
path	xs:string	minOccurs = 1 maxOccurs = 1 nillable = false	Absolute path of the prompt file from the prompt root folder.

GET Prompt Files Path

HTTPS Method	GET
URI	https://<server>/adminapi/files/prompts
Example URI	https://uccx-server/adminapi/files/prompts
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 403, 405, 500

Error Codes	—
--------------------	---

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<Prompts>
  <Prompt>
    <Path>/en_US/Admin_SpokenName.wav</Path>
  </Prompt>
  <Prompt>
    <Path>/en_US/name_cancelled.wav</Path>
  </Prompt>
  <Prompt>
    <Path>/en_AU/Admin_SpokenName.wav</Path>
  </Prompt>
  <Prompt>
    <Path>/en/name_cancelled.wav</Path>
  </Prompt>
</Prompts>
```

Proxy Configuration

Data Structure for Proxy Configuration

Attribute Name	Base Type	Restrictions	Description
httpProxy	string	Host Name cannot be the IP address of the proxy Server. Range: 1 to 65535	<ul style="list-style-type: none"> • hostName: Fully qualified domain name (FQDN) of the HTTPS proxy server. • port: Port number that is used to connect to the HTTPS proxy server.
socksProxy	string	Host Name cannot be the IP address of the proxy Server. Range: 1 to 65535	<ul style="list-style-type: none"> • hostName: Fully qualified domain name (FQDN) of the SOCKS proxy server. • port: Port number that is used to connect to the SOCKS proxy server. • userName: Username of the Socks Proxy server. • password: Password for the Socks Proxy server

Get Proxy Configuration

HTTPS Method	GET
URI	https://<server>/adminapi/proxyConfig
Example URI	https://uccx-Server/adminapi/proxyConfig
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 500
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<proxyConfig>
<httpProxy>
<hostName>proxy.esl.cisco.com</hostName>
<port>80</port>
</httpProxy>
<socksProxy>
<hostName>socks.proxy.com</hostName>
<Port>1080</Port>
<userName>Administrator</userName>
<password>l2Ccbul2</password>
</socksProxy>
</proxyConfig>
```

Update Proxy Configuration

HTTPS Method	PUT
URI	https://<server>/adminapi/proxyConfig
Example URI	https://uccx-Server/adminapi/proxyConfig
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 500
Error Codes	—

Success Response

```
HTTPS/1.1 200 OK
```


Sample Input XML — Modify Proxy Configuration (PUT)

```
<?xml version="1.0" encoding="UTF-8"?>
<proxyConfig>
<httpProxy>
<hostName>proxy.esl.cisco.com</hostName>
<port>80</port>
</httpProxy>
<socksProxy>
<hostName>socks.proxy.com</hostName>
<Port>1080</Port>
<userName>Administrator</userName>
<password>12Ccbu12</password>
</socksProxy>
</proxyConfig>
```

Resource

The Unified CCX system creates a new record in the Resource table when the Unified CCX system retrieves agent information from the Unified CM. A Resource record contains information about the resource (agent). One such record exists for each active and inactive resource.

Data Structure for Resource

Attribute Name	Base Type	Restrictions	Description
resourceId	int	minOccurs = 1 maxOccurs = 1 nillable = false	Numeric identifier of the resource
userID	string	minOccurs = 1 maxOccurs = 1 nillable = false	Numeric identifier of user
fullName	string	minOccurs = 1 maxOccurs = 1 nillable = false	Full name of the user
lastName	string	minOccurs = 1 maxOccurs = 1 nillable = false	Last name of the user

Attribute Name	Base Type	Restrictions	Description
alias	string	minOccurs = 0 maxOccurs = 1 nillable = false minLength = 0 maxLength = 31 Alias contains alphanumeric characters and the special characters hyphens (-), space (), periods (.), and underscores (_), except at the beginning and at the end of the alias name.	Alias name of agent is displayed to the customer when on chat.
extension	string	minOccurs = 1 maxOccurs = 1 nillable = false	Extension of the user
resourceGroup	nameUriPair	minOccurs = 1 maxOccurs = 1 nillable = true	Group of the resource
skillMap	skillMap	minOccurs = 1 maxOccurs = 1 nillable = false	Mapping of skills of the resource
autoAvailable	boolean	—	Availability of resource
type	resourceType	minOccurs = 1 maxOccurs = 1 nillable = false	Type of resource
team	nameUriPair	minOccurs = 1 maxOccurs = 1 nillable = false	Name of the team
primarySupervisorOf	supervisorOf	minOccurs = 1 maxOccurs = 1 nillable = true	Primary supervisor

Attribute Name	Base Type	Restrictions	Description
secondarySupervisorOf	supervisorOf	minOccurs = 1 maxOccurs = 1 nillable = true	Secondary supervisor

Get List of Resources

URI	https://<server>/adminapi/resource
Example URI	https://uccx-server/adminapi/resource
HTTPS Method	GET
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response



Note The response depends on the credentials provided. The following are the use cases that are observed when this API is invoked:

- When an administrator gets the list of resources, the response has the complete list of agents.
- When a supervisor gets the list of resources, the response has the list of agents corresponding to the team of the supervisor. The supervisor can be either a primary or secondary supervisor.
- The query parameter combination of `withCsqs` and `detail=none` is invalid.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<resources>
  <resource>
    <self>https://uccx-server/adminapi/resource/agent1333</self>
    <userID>agent1333</userID>
    <firstName></firstName>
    <lastName>agent1333</lastName>
    <extension>2244333</extension>
    <alias>uccx</alias>
    <skillMap>
      <skillCompetency>
        <competencelevel>5</competencelevel>
        <skillNameUriPair name="OBAgentSkill">
          <refURL>https://uccx-server/adminapi/skill/3</refURL>
        </skillNameUriPair>
      </skillCompetency>
    </skillMap>
  </resource>
</resources>
```

```

    <autoAvailable>true</autoAvailable>
    <type>1</type>
    <team name="Default">
      <refURL>https://uccx-server/adminapi/team/1</refURL>
    </team>
    <primarySupervisorOf/>
    <secondarySupervisorOf/>
  </resource>
</resources>

```

Get List of Resources by CSQ

URI	https://<server>/adminapi/resource?csqid={id}
Example URI	https://<server>/adminapi/resource?csqid=1
HTTPS Method	GET
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	500
Error Codes	—

Success Response

```

<?xml version="1.0" encoding="UTF-8"

standalone="yes"?>
<resources>
  <resource>

<self>https://10.78.95.167/adminapi/resource/agent8000<
/self>
  <userID>agent8000</userID>
  <firstName></firstName>
  <lastName>agent8000</lastName>
  <extension>8000</extension>
  <alias>AgentAlias</alias>
  <resourceGroup name="Auto_Sanity_RG">

<refURL>https://10.78.95.167/adminapi/resourceGroup/2</
refURL>
  </resourceGroup>
  <autoAvailable>true</autoAvailable>
  <type>1</type>
  <team name="Auto_Team">

<refURL>https://10.78.95.167/adminapi/team/2</refURL>
  </team>
  <primarySupervisorOf/>
  <secondarySupervisorOf/>
  </resource>
</resources>

```

Get Resource with CSQs

HTTPS Method	GET
URI	<a href="https://<server>/adminapi/resource?withCsqs">https://<server>/adminapi/resource?withCsqs <a href="https://<server>/adminapi/resource?csqid={id}&withCsqs">https://<server>/adminapi/resource?csqid={id}&withCsqs <a href="https://<server>/adminapi/resource/{id}?withCsqs">https://<server>/adminapi/resource/{id}?withCsqs
Example URI	https://uccx-server/adminapi/resource?withCsqs https://uccx-server/adminapi/resource?csqid=1&withCsqs https://uccx-server/adminapi/resource/1?withCsqs
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 401, 403, 404, 405, 500, 503
Error Codes	—

Success Response

```
<resource>
  <self>
    https://10.78.92.103/adminapi/resource/rick
  </self>
  <userID>
    rick
  </userID>
  <firstName>
    Rick
  </firstName>
  <lastName>
    Barrows
  </lastName>
  <extension>
    1009
  </extension>
  <alias>
  </alias>
  <resourceGroup name="RG1">
    <refURL>
      https://10.78.92.103/adminapi/resourceGroup/2
    </refURL>
  </resourceGroup>
  <skillMap>
    <skillCompetency>
      <competencelevel>
        5
      </competencelevel>
      <skillNameUriPair name="SalesSkill">
        <refURL>
          https://10.78.92.103/adminapi/skill/2
        </refURL>
      </skillNameUriPair>
    </skillCompetency>
  </skillMap>
```

```

<csqMap>
  <csq name="VoiceCSQ1">
    <refURL>
      https://10.78.92.103/adminapi/csq/1
    </refURL>
  </csq>
  <csq name="ChatCSQ1">
    <refURL>
      https://10.78.92.103/adminapi/csq/2
    </refURL>
  </csq>
</csqMap>
<autoAvailable>
  true
</autoAvailable>
<type>
  2
</type>
<team name="Default">
  <refURL>
    https://10.78.92.103/adminapi/team/1
  </refURL>
</team>
<primarySupervisorOf>
  <supervisorOfTeamName name="Team2">
    <refURL>
      https://10.78.92.103/adminapi/team/34
    </refURL>
  </supervisorOfTeamName>
  <supervisorOfTeamName name="Team1">
    <refURL>
      https://10.78.92.103/adminapi/team/32
    </refURL>
  </supervisorOfTeamName>
</primarySupervisorOf>
<secondarySupervisorOf/>
</resource>

```

Get Modified Information of Resources by CSQ

HTTPS Method	GET
URI	https://<server>/adminapi/resource?csqid={Id}&lastReSkillDetails
Example URI	https://<server>/adminapi/resource?csqid=2&lastReSkillDetails
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 401, 403, 404, 405, 500
Error Codes	—

Success Response

```
https://<server fqdn>/adminapi/resource?csqid=&lastReSkillDetails
```

Sample output XML—GET

```

Invoked by Supervisor
<resource>
  ..
  ..
  <lastReSkillDetails>
    <timestamp>1516593630</timestamp>
    <reskiller>Rick Barrows</reskiller>
    <reskillerType>Supervisor</reskillerType>
  </lastReSkillDetails>
</resource>

Invoked by Admin
<resource>
  ..
  ..
  <lastReSkillDetails>
    <timestamp>1516593630</timestamp>
    <reskiller>Admin</reskiller>
    <reskillerType>Administrator</reskillerType>

  </lastReSkillDetails>
</resource>

```



Note PUT, POST, and DELETE methods are not supported. Corresponding error response will be returned.

Add Resources

The add resource operation allows you to add new resources to the CSQ.

HTTPS Method	POST
URI	https://<server>/adminapi/csq/{id}/resources
Example URI	https://uccx-server/adminapi/csq/{id}/resources
Query Parameter	onNoImpact
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	207,400,401,403,404,405,409,424,500,503
Error Codes	—

Sample Input XML—Add Resources (POST)

```

<resources>
  <resource id="Id of the resource">
    </resource>
  <resource id="Id of the resource">
    </resource>
</resources>

```

Output Response

The add resources API will have one of the following output responses:

- Success
- Partial Success
- Failure

Success Response

HTTPS status code 200 OK for a successful POST operation.

Sample Output XML—Add Resources (POST)

```
<?xml version="1.0" encoding="UTF-8"?>
<resources>
  <resource id="Steve">
    <csqs>
      <before>csq1,csq2</before>
      <newlyAdded>csq3,csq4</newlyAdded>
    </csqs>
  </resource>
  <resource id="Michael">
    <csqs>
      <before>csq1,csq2</before>
      <newlyAdded>csq3,csq4</newlyAdded>
    </csqs>
  </resource>
</resources>
```

Partial Success Response

Partial success response when query parameter `onNoImpact` is sent and if there is any impact on a resource:

HTTPS status code 207 Multi-Status.

Sample Output XML—Add Resources (POST)

```
<?xml version="1.0" encoding="UTF-8"?>
<resources>
  <resource id="Steve">
    <status>impact</status>
    <impacts>
      <impact type="ADDS_TO_OTHER_CSQ">
        <csqs>csq1,csq2</csqs>
      </impact>
    </impacts>
  </resource>
  <resource id="Michael">
    <status>success</status>
    <csqs>
      <before>csq1,csq2</before>
      <newlyAdded>csq3,csq4</newlyAdded>
    </csqs>
  </resource>
</resources>
```


Failure Response

Sample Output XML—Add Resources (POST)

```
<?xml version="1.0" encoding="UTF-8"?>
<resources>
  <resource id="sjefferson">
    <status>Error</status>
    <apiError>
      <errorData>sjefferson</errorData>
      <errorMessage>Resource sjefferson not found in Supervisor rickbarrows
team.</errorMessage>
      <errorType>InstanceNotFound</errorType>
      <i18n>
        <key>adminapi.resource.notfound.inSupervisorTeam</key>
        <args>
          <arg>sjefferson</arg>
        </args>
      </i18n>
    </apiError>
  </resource>
  <resource id="jopeter">
    <status>Error</status>
    <apiError>
      <errorData>jopeter</errorData>
      <errorMessage>Resource jopeter already part of CSQ with ID
billingCsq.</errorMessage>
      <errorType>ObjectAlreadyExists</errorType>
      <i18n>
        <key>adminapi.resource.alreadyExists.inCsq</key>
        <args>
          <arg>jopeter</arg>
          <arg>billingCsq</arg>
        </args>
      </i18n>
    </apiError>
  </resource>
</resources>
```

Get Resource

URI	https://<server>/adminapi/resource/{id}
Example URI	https://uccx-server/adminapi/resource/12
HTTPS Method	GET
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—



Note The ID in the URI is the user ID of a particular resource. The resource in the Unified Communications Manager environment is an end user. The user ID in Unified CCX can include all the special characters except the forward slash (/) and the backslash (\).

Success Response

```
<resource>
  <self>https://uccx-server/adminapi/resource/uccxautosup1</self>
  <userID>uccxautosup1</userID>
  <firstName />
  <lastName>uccxautosup1</lastName>
  <extension>6003</extension>
  <skillMap />
  <autoAvailable>true</autoAvailable>
  <type>2</type>
  <team name="Team123">
    <refURL>https://uccx-server/adminapi/team/1</refURL>
  </team>
  <primarySupervisorOf>
    <supervisorOfTeamName name="Team123">
      <refURL>https://uccx-server/adminapi/team/1</refURL>
    </supervisorOfTeamName>
    <supervisorOfTeamName name="tempteam1">
      <refURL>https://uccx-server/adminapi/team/3</refURL>
    </supervisorOfTeamName>
    <supervisorOfTeamName name="tempteam3">
      <refURL>https://uccx-server/adminapi/team/5</refURL>
    </supervisorOfTeamName>
    <supervisorOfTeamName name="chatTeamchatTeamchatTeamchatTeam">
      <refURL>https://uccx-server/adminapi/team/7</refURL>
    </supervisorOfTeamName>
    <supervisorOfTeamName name="TeamTest1213497663301901355924770962">
      <refURL>https://uccx-server/adminapi/team/9</refURL>
    </supervisorOfTeamName>
    <supervisorOfTeamName name="TeamTest1213497663301901355924777915">
      <refURL>https://uccx-server/adminapi/team/10</refURL>
    </supervisorOfTeamName>
    <supervisorOfTeamName name="Team123">
      <refURL>https://uccx-server/adminapi/team/11</refURL>
    </supervisorOfTeamName>
  </primarySupervisorOf>
  <secondarySupervisorOf>
    <supervisorOfTeamName name="tempteam2">
      <refURL>https://uccx-server/adminapi/team/4</refURL>
    </supervisorOfTeamName>
    <supervisorOfTeamName name="tempteam4">
      <refURL>https://uccx-server/adminapi/team/6</refURL>
    </supervisorOfTeamName>
  </secondarySupervisorOf>
</resource>
```

Modify Resource

URI	https://<server>/adminapi/resource/{id}
Example URI	https://uccx-server/adminapi/resource/{id}

HTTPS Method	PUT
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—



Note The ID in the URI is the user ID of a particular resource. The resource in the Unified Communications Manager environment is an end user. The user ID in Unified CCX can include all the special characters except the forward slash (/) and the backslash (\).

Success Response

—

Delete Resource

The delete resource operation removes the agent from a CSQ.

HTTPS Method	DELETE
URI	https://<server>/adminapi/csq/{csqid}/resources/{resourceid}
Example URI	https://ucx-server/adminapi/csq/{id}/resources/{resourceid}
Query Parameter	onNoImpact
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401,403,404,424,500,503
Error Codes	—

Sample output XML— Delete Resource with Impact

HTTPS status code 424 Failed Dependency

When DELETE operation is performed with query parameter `onNoImpact` and there is an impact on other CSQs, HTTP status code 424 is returned with the following response body:

```
<?xml version="1.0" encoding="UTF-8"?>
<resources>
  <resource id="sjefferson">
    <status>IMPACT</status>
    <impacts>
      <impact type="REMOVES_FROM_OTHER_CSQ">
        <csqs>csq1,csq2</csqs>
      </impact>
    </impacts>
  </resource>
</resources>
```

Failure Response

Sample output XML— Delete Resource

```
<?xml version="1.0" encoding="UTF-8"?>
<apiErrors>
  <apiError>
    <errorData>sjefferson</errorData>
    <errorMessage>Resource sjefferson not part of CSQ 1.</errorMessage>
    <errorType>Not Found</errorType>
    <i18n>
      <key>adminapi.resource.notPartOf.Csq</key>
      <args>
        <arg>sjefferson</arg>
        <arg>csq1</arg>
      </args>
    </i18n>
  </apiError>
</apiErrors>
```

Schedule Resource

You can add, edit, remove, and view resources in the CSQ by configuring a schedule for a resource. This schedule can be updated or deleted if required.

Get Schedules

The get schedules operation retrieves the schedules of the agents that are yet to be added or removed for a CSQ.

HTTPS Method	GET
URI	https://<server>/adminapi/csq/{id}/schedules
Example URI	https://uccx-server/adminapi/csq/{id}/schedules
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401,403,404,405,500,503
Error Codes	—

Success Response

Sample output XML— Get Schedules

```
<schedules>
  <schedule id="1">
    <resource id="A1">
      <addAt>1551932880000</addAt>
    </resource>
  </schedule>
  <schedule id="2">
    <resource id="A2">
      <addAt>1551932880000</addAt>
      <removeAt>1651932880000</removeAt>
    </resource>
  </schedule>
  <schedule id="3">
    <resource id="A1">
      <removeAt>1551932880000</removeAt>
    </resource>
  </schedule>
</schedules>
```

Failure Response

Sample output XML— Get Schedules

```
<apiErrors>
  <apiError>
    <errorData>InternalServerError</errorData>
    <errorMessage>Message</errorMessage>
    <errorType>InternalServerError</errorType>
    <i18n>
      <key>adminapi.internal.error</key>
    </i18n>
  </apiError>
</apiErrors>
```

Create Schedules

The create schedules operation saves the schedules of agents that are yet to be added or removed for a CSQ.

HTTPS Method	POST
URI	https://<server>/adminapi/csq/{id}/schedules
Example URI	https://uccx-server/adminapi/csq/{id}/schedules
Query Parameter	onNoImpact
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	207,400,401,403,404,405,409,424,500,503
Error Codes	—

Sample Input XML—Create Schedules

```
<schedules>
  <schedule>
    <resource id="A1">
      <addAt>The auto addition time for this resource to be added to CSQ</addAt>
      <removeAt>The auto removal time for this resource to be removed from CSQ</removeAt>

    </resource>
  </schedule>
  <schedule>
    <resource id="A2">
      <addAt>The auto addition time for this resource to be added to CSQ</addAt>
      <removeAt>The auto removal time for this resource to be removed from CSQ</removeAt>

    </resource>
  </schedule>
</schedules>
```

Output Response

The create schedules API will have one of the following output responses:

- Success
- Partial Success
- Failure

Success Response

```
HTTPS/1.1 200 OK
```

Partial Success Response

The following partial success response is displayed when query parameter `onNoImpact` is sent and if there is any impact on a resource:

```
HTTPS status code 207 Multi-Status.
```

Sample Output XML—Create Schedules

```
<schedules>
  <schedule id="1">
    <resource id="A1">
      <addAt>The auto addition time for this resource to be added to CSQ</addAt>
      <removeAt>The auto removal time for this resource to be removed from CSQ</removeAt>

    </resource>
  </schedule>
  <schedule>
    <resource id="A2">
      <status>impact</status>
      <impacts>
        <impact type="AUTO_REMOVE_UPDATE">
          <removeAt>1525061952</removeAt>
          <csqs>csq1, csq2</csqs>
        </impact>
      </impacts>
    </resource>
  </schedule>
</schedules>
```

Failure Response

Sample Output XML—Create Schedules

```
<apiErrors>
  <apiError>
    <errorData>InternalServerError</errorData>
    <errorMessage>Message</errorMessage>
    <errorType>InternalServerError</errorType>
    <i18n>
      <key>adminapi.internal.error</key>
    </i18n>
  </apiError>
</apiErrors>
```

Edit Schedules

The edit schedule operation allows you to edit the reskill schedule of resources.

HTTPS Method	PATCH
URI	https://<server>/adminapi/csq/{id}/schedules
Example URI	https://uccx-server/adminapi/csq/{id}/schedules
Query Parameter	onNoImpact
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	207,400,401,403,404,405,409,424,500,503
Error Codes	—

Sample Input XML—Edit Schedules (PATCH)

```
<schedules>
  <schedule id="1">
    <resource id="Michael">
      <removeAt>Modified auto removal time for this resource to be removed from
CSQ</removeAt>
    </resource>
  </schedule>
</schedules>
```

Output Response

The edit schedule API will have one of the following output responses:

- Success
- Partial Success
- Failure

Success Response

```
HTTPS/1.1 200 OK
```

Partial Success Response

The following partial success response when query parameter `onNoImpact` is sent and if there is any impact on a resource:

```
HTTPS status code 207 Multi-Status
```


Sample Output XML—Edit Schedules (PATCH)

```
<?xml version="1.0" encoding="UTF-8"?>
  <schedules>
    <schedule id="1">
      <resource id="steve">
        <status>impact</status>
        <impacts>
          <impact type="AUTO_ADD_UPDATE">
            <addAt>1525061950</addAt>
            <csqs>csq1,csq2</csqs>
          </impact>
        </impacts>
      </resource>
    </schedule>
    <schedule id="2">
      <resource id="Jefferson">
        <status>impact</status>
        <impacts>
          <impact type="AUTO_REMOVE_UPDATE">
            <removeAt>1525061952</removeAt>
            <csqs>csq1,csq2</csqs>
          </impact>
        </impacts>
      </resource>
    </schedule>
    <schedule id="3">
      <resource id="Michael">
        <status>Success</status>
      </resource>
    </schedule>
  </schedules>
```

Failure Response**Sample Output XML—Edit Schedules (PATCH)**

```
<?xml version="1.0" encoding="UTF-8"?>
<apiErrors>
  <apiError>
    <errorData></errorData>
    <errorMessage>Resource(s) not found in the system.</errorMessage>
    <errorType>InstanceNotFound</errorType>
    <i18n>
      <key>adminapi.resources.notfound</key>
      <args/>
    </i18n>
  </apiError>
</apiErrors>
```

Delete Schedules

The delete schedules operation removes the schedules of agents from a CSQ.

HTTPS Method	DELETE
URI	https://<server>/adminapi/csq/{id}/schedules
Example URI	https://uccx-server/adminapi/csq/{id}/schedules

Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	207,400,401,403,404,405,500,503
Error Codes	—

Sample Input XML—Delete Schedules

```
<schedules>
  <schedule id="1">
    <resource id="A1"/>
  </schedule>
</schedules>
```

Failure Response

Sample output XML— Delete Schedules

```
<apiErrors>
  <apiError>
    <errorData>InternalServerError</errorData>
    <errorMessage>Message</errorMessage>
    <errorType>InternalServerError</errorType>
    <i18n>
      <key>adminapi.internal.error</key>
    </i18n>
  </apiError>
</apiErrors>
```

Resource Bundle of Locale

The Resource Bundle of Locale returns the resource bundle of the requested locale so that the messages can be viewed in the localized language.

The following admin GET APIs are enhanced to return the i18n details during the error scenarios:

- Get Advanced Supervisor Capabilities
- Get CSQ
- Get List of Resources
- Get List of Resources by CSQ
- Get Resource with CSQs
- Get Resource

Data Structure for Resource Bundle of Locale

Attribute Name	Base Type	Restrictions	Description
locale	string	minOccurs = 1 maxOccurs = 1 nillable = false	Defines the locale.
bundle	bundle	minOccurs = 1 maxOccurs = 1 nillable = false	Defines the Key-Value pair fields.

Get Resource Bundle of Locale

HTTPS Method	GET
URI	https://<server>/adminapi/api/i18n/{locale}
Example URI	https://uccx-server/adminapi/api/i18n/en_US
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 403, 404, 405, 500
Error Codes	—



Note User is allowed to enter partial or full locale name in URI.

- When there is a partial locale name, the system looks for a match of locale name with existing resource locale and returns the first matching locale else it returns a 404 HTTP Error.
- When a locale name is complete, the system searches the requested locale. If found, it returns the matching locale else it returns a 404 HTTP Error.
- When a locale is not mentioned in the URI, the system returns the resource bundle of en_US locale.

Success Response

```
<resourceBundle>
  <locale>en_US</locale>
  <bundle>
    <entry>
      <key>name</key>
      <value>value</value>
    </entry>
  </entry>
```

```

        <key>name</key>
        <value>value</value>
    </entry>
</bundle>
</resourceBundle>

```

Resource Group

Resource groups are collections of agents that the CSQ uses to handle incoming calls. To use resource-group-based CSQs, you must specify a resource group.

Data Structure for Resource Group

Attribute Name	Base Type	Restrictions	Descriptions
id	int	None	Unique identifier for the resource group.
name	string	minLength value = 1 maxLength value = 30 Occurences = 1 Resource Group name contains alphanumeric characters, spaces, underscores (_), and hyphens (-) without leading or trailing white spaces.	Name of the resource group as set up in Unified CCX administration.

Get List of Resource Groups

HTTPS Method	GET
URI	https://<server>/adminapi/resourceGroup
Example URI	https://uccx-server/adminapi/resourceGroup
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	—

Success Response

```

<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<resourceGroups>
  <resourceGroup>
    <self>https://uccx-server/adminapi/resourceGroup/4</self>
  </resourceGroup>
</resourceGroups>

```

```

    <id>4</id>
    <name>Auto_RG</name>
  </resourceGroup>
  <resourceGroup>
    <self>https://uccx-server/adminapi/resourceGroup/7</self>
    <id>7</id>
    <name>rg1</name>
  </resourceGroup>
</resourceGroups>

```

Get Resource Group

HTTPS Method	GET
URI	https://<server>/adminapi/resourceGroup/{id}
Example URI	https://uccx-server/adminapi/resourceGroup/22
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404, 405
Error Codes	HTTPS/1.1 404 Not Found

Success Response

```

<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<resourceGroup>
  <self>https://uccx-server/adminapi/resourceGroup/7</self>
  <id>7</id>
  <name>rg1</name>
</resourceGroup>>

```

Create Resource Group

HTTPS Method	POST
URI	https://<server>/adminapi/resourceGroup
Example URI	https://uccx-server/adminapi/resourceGroup
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	401, 405
Error Codes	HTTPS/1.1 404 Not Found

Success Response

HTTPS/1.1 201 Created

Sample Input XML—Create Resource Group (POST)

```
<?xml version="1.0" encoding="UTF-8"?>
<ResourceGroup>
  <self/>
  <name>Group-02</name>
</ResourceGroup>
```

Modify Resource Group

HTTPS Method	PUT
URI	https://<server>/adminapi/resourceGroup/{id}
Example URI	https://uccx-server/adminapi/resourceGroup/21
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	HTTPS/1.1 400 Bad Request

Success Response

HTTPS/1.1 200 OK

Sample Input XML—Modify ResourceGroup (PUT)

```
<?xml version="1.0" encoding="UTF-8"?>
<ResourceGroup>
  <self>https://uccx-server/adminapi/resourceGroup/</self>
  <id>3</id>
  <name>Group-01</name>
</ResourceGroup>
```

Delete Resource Group

HTTPS Method	DELETE
URI	https://<server>/adminapi/resourceGroup/{id}
Example URI	https://uccx-server/adminapi/resourceGroup/21
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405

Error Codes	HTTPS/1.1 404 Not Found
--------------------	-------------------------

Success Response

HTTPS/1.1 200 OK

Skill

Skills are customer-definable labels that are assigned to agents. All the Unified CCX license packages can route incoming calls to agents who have the necessary skills or sets of skills to handle the call.

Data Structure for Skill

Attribute Name	Base Type	Restrictions	Description
skillId	int	None	Numeric identifier of the skill
skillName	string	minlength = 1 maxlength = 24 occurrences = 1 Skill Name contains alphanumeric characters, special characters, spaces, underscores (_), and hyphens (-) without leading or trailing white spaces.	Name of the skill

Get List of Skills



Note The maximum number of skills is 150.

HTTPS Method	GET
URI	https://<server>/adminapi/skill
Example URI	https://uccx-server/adminapi/skill
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405

Error Codes	—
--------------------	---

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<Skills>
  <Skill>
    <self>https://uccx-server/adminapi/skill/2</self>
    <skillId>2</skillId>
    <skillName>Auto_Sanity_Skill</skillName>
  </Skill>
  <Skill>
    <self>https://uccx-server/adminapi/skill/3</self>
    <skillId>3</skillId>
    <skillName>icd</skillName>
  </Skill>
</Skills>
```

Get Skill

HTTPS Method	GET
URI	https://<server>/adminapi/skill/{id}
Example URI	https://uccx-server/adminapi/skill/12
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<Skill>
  <self>https://uccx-server/adminapi/skill/12</self>
  <skillId>12</skillId>
  <skillName>Auto_Sanity_Skill</skillName>
</Skill>
```

Create Skill

Using this API, you can assign customer-definable label to an agent.

HTTPS Method	POST
URI	https://<server>/adminapi/skill
Example URI	https://uccx-server/adminapi/skill
Content Type	Application/XML, Application/JSON

HTTPS Success Code	201
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Sample Input XML—Create Skill (POST)

```
<Skill>
<self href="https://uccx-server/adminapi/skill/1" rel="self" type="skill"/>
<skillId>1</skillId>
<skillName>skill1</skillName>
</Skill>
```

Modify Skill

HTTPS Method	PUT
URI	https://<server>/adminapi/skill/{id}
Example URI	https://uccx-server/adminapi/skill/12
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Sample Input XML—Modify Skill (PUT)

```
<Skill>
<self href="https://uccx-server/adminapi/skill/1" rel="self" type="skill"/>
<skillId>1</skillId>
<skillName>skillNewName</skillName>
</Skill>
```

Delete Skill

When you delete a skill:

- the skill is removed automatically.
- If the skill is associated with any agents or CSQs, you are directed to another web page where you can see a list of the associated CSQs and agents, and you are prompted to confirm whether you want to delete it or not.

HTTPS Method	DELETE
URI	https://<server>/adminapi/skill {skillId}
Example URI	https://uccx-server/adminapi/skill/1
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Sip Gateway Configuration

SIP Gateways provide media termination and signal translation between the PSTN and the IP networks using the SIP signaling protocol for voice and video traffic. You can add or modify the SIP Gateway parameters using REST API that enable the Outbound subsystem of the Unified CCX to communicate with the SIP gateway.

Call Progress Analysis is a feature of the SIP gateway which determines whether the reached end point is an answering machine, live voice, fax, or beep tone and so on. The SIP gateway performs call progressive analysis of the call and informs the outcome of the call to the Unified CCX

Data Structure for Sip Gateway Configurations

Attribute Name	Base Type	Restrictions	Description
IpAddressorHostName	string	Regex: (([a-zA-Z0-9] [a-zA-Z0-9] [a-zA-Z0-9-]{0,61}[a-zA-Z0-9]) (\.([a-zA-Z0-9] [a-zA-Z0-9] [a-zA-Z0-9-]{0,61}[a-zA-Z0-9]))*)	IP Address or hostname of the SIP Gateway. Note The IP address validation for the valid range and the host name validation for the existence of at least one alphabetic character is done using Java code.

Attribute Name	Base Type	Restrictions	Description
GatewayPort	int	minOccurs = 1 maxOccurs = 1 nillable = false minInclusive = 1 maxInclusive = 65535	SIP port used by the gateway to communicate with Unified CCX.
LocalCCXPort	int	Single port for communication is 5065	Port number used on the Unified CCX server to communicate with the gateway. Default value is as follows: <ul style="list-style-type: none"> • Fresh install: 5065 • Upgrade: Previously configured value. If not configured, use 5065.
transportProtocol	string	default = UDP	Protocol required to send SIP messages.
CpaAnalysisPeriod	int	minOccurs = 1 maxOccurs = 1 nillable = false minInclusive = 1000 maxInclusive = 10000	CPA analysis period.
CpaMinSilencePeriod	int	minOccurs = 1 maxOccurs = 1 nillable = false minInclusive = 100 maxInclusive = 1000	Minimum silence period.
CpaMaxTimeAnalysis	int	minOccurs = 1 maxOccurs = 1 nillable = false minInclusive = 1000 maxInclusive = 10000	Maximum time analysis period.

Attribute Name	Base Type	Restrictions	Description
CpaMinValidSpeechTime	int	minOccurs = 1 maxOccurs = 1 nillable = false minInclusive = 50 maxInclusive = 500	Minimum valid speech time.
CpaMaxTermToneAnalysis	int	minOccurs = 1 maxOccurs = 1 nillable = false minInclusive = 1000 maxInclusive = 60000	Maximum term tone analysis period.

Get List of Sip Gateway Configurations

HTTPS Method	GET
URI	https://<server>/adminapi/obsipgatewayconfiguration
Example URI	https://uccx-server/adminapi/obsipgatewayconfiguration
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<obSipGatewayConfiguration>
  <IpAddressorHostName>uccx-server</IpAddressorHostName>
  <GatewayPort>1024</GatewayPort>
  <LocalCCXPort>5678</LocalCCXPort>
  <transportProtocol>UDP</transportProtocol>
  <CpaMinSilencePeriod>1000</CpaMinSilencePeriod>
  <CpaAnalysisPeriod>2010</CpaAnalysisPeriod>
  <CpaMaxTimeAnalysis>3000</CpaMaxTimeAnalysis>
  <CpaMinValidSpeechTime>112</CpaMinValidSpeechTime>
  <CpaMaxTermToneAnalysis>15000</CpaMaxTermToneAnalysis>
</obSipGatewayConfiguration>
```

Update SIP Gateway Configurations

HTTPS Method	PUT
---------------------	-----

URI	https://<server>/adminapi/obsipgatewayconfiguration
Example URI	https://uccx-server/adminapi/obsipgatewayconfiguration
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Sample Input XML — Modify SIP Gateway Configuration (PUT)

```
<?xml version="1.0" encoding="UTF-8"?>
<obSipGatewayConfiguration>
  <IpAddressorHostName>uccx-server</IpAddressorHostName>
  <GatewayPort>1024</GatewayPort>
  <LocalCCXPort>5678</LocalCCXPort>
  <transportProtocol>UDP</transportProtocol>
  <CpaMinSilencePeriod>1000</CpaMinSilencePeriod>
  <CpaAnalysisPeriod>2010</CpaAnalysisPeriod>
  <CpaMaxTimeAnalysis>3000</CpaMaxTimeAnalysis>
  <CpaMinValidSpeechTime>112</CpaMinValidSpeechTime>
  <CpaMaxTermToneAnalysis>15000</CpaMaxTermToneAnalysis>
</obSipGatewayConfiguration>
```

Standalone Cisco Unified Intelligence Center

Unified CCX 11.0(1) provides support for a standalone Cisco Unified Intelligence Center system in addition to the on-box Cisco Unified Intelligence Center.

A standalone Unified Intelligence Center can be installed with standard, enhanced and premium licenses. The version of the standalone Cisco Unified Intelligence Center should be the same as the Cisco Unified Intelligence Center that is embedded in Unified CCX. The standalone Cisco Unified Intelligence Center supports multiple data source including Unified CCX.

Data Structure for Standalone CUIC

Attribute Name	Base Type	Restrictions	Description
FQDN	string	Regex : [A-Za-z0-9]([A-Z0-9a-z_\-\.])+[a-z]{2} minLength = 1 maxLength = 255	Fully Qualified Domain Name of the standalone CUIC server.

Attribute Name	Base Type	Restrictions	Description
Datasource Name	string	minLength = 1	Datasource name to which the CUIC server is pointing.
Username	string	minLength = 1	Username of the CUIC server.
Password	string	minLength = 1	Password of the CUIC server.

Configuration of Standalone CUIC

HTTPS Method	URI	Example URI	Description
GET	https://<server>/adminapi/standalonecuicconfiguration	https://ucce-server/adminapi/standalonecuicconfiguration	Get details of the CUIC Server.
PUT	https://<server>/adminapi/standalonecuicconfiguration	https://ucce-server/adminapi/standalonecuicconfiguration	Create or modify the CUIC Server Information.

Sample Input XML - Modify CUIC Configuration (PUT)

```
<?xml version="1.0" encoding="UTF-8"?>
<standaloneCuicConfiguration xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="standaloneCuicConfiguration.xsd">
  <fqdn>host.cisco.com</fqdn>
  <datasourceName>ucce realtime</datasourceName>
  <userName>userName</userName>
  <password>password</password>
</standaloneCuicConfiguration>
```

Team

A team is a group of agents who report to the same supervisor. A team can have one primary supervisor and optional secondary supervisors. A supervisor can also monitor CSQs that are assigned to the team that is being supervised.

Data Structure for Team

Attribute Name	Base Type	Restrictions	Description
teamName	string	minOccurs = 1 maxOccurs = 1 nillable = false minLength value = 1 maxLength value = 49 Team name contains alphanumeric characters, special characters, spaces, underscores (_), and hyphens (-) without leading or trailing white spaces.	Name of the team
primarySupervisor	nameUriPair	minOccurs = 0 maxOccurs = 1 nillable = false	Name of the primary supervisor of the team
secondarySupervisors	nameUriPair	minOccurs = 0 maxOccurs = 1 nillable = false	Name of the secondary supervisor of the team
resources	nameUriPair	minOccurs = 0 maxOccurs = 1 nillable = false	Name of the resource
csqs	nameUriPair	minOccurs = 0 maxOccurs = 1 nillable = false	Name of the CSQ

Attribute Name	Base Type	Restrictions	Description
nonAcdBusyOverride	boolean	minOccurs = 0 maxOccurs = 1 nillable = false	Determines whether the team settings override global settings. Note This attribute is available only if Change Agent State to Not Ready when Agent Busy on Non ACD Line is configured at a team level. To configure this functionality at a team level, you must install UCCX 12.5(1) SU1 ES01.

Get List of Teams

HTTPS Method	GET
URI	https://<server>/adminapi/team
Example URI	https://uccx-server/adminapi/team
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	HTTPS/1.1 404 Not Found

Success Response

—

Get Team

HTTPS Method	POST
URI	https://<server>/adminapi/team/{id}
Example URI	https://uccx-server/adminapi/team/12
Accept	Application/XML, Application/JSON

HTTPS Success Code	201
HTTPS Failure Codes	401, 405
Error Codes	HTTPS/1.1 404 Not Found

Success Response

—

Create Team

HTTPS Method	POST
URI	https://<server>/adminapi/team
Example URI	https://uccx-server/adminapi/team
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

HTTPS/1.1 201 Created

Sample Input XML—Create Team (PUT)

Modify Team

HTTPS Method	PUT
URI	https://<server>/adminapi/team/{id}
Example URI	https://uccx-server/adminapi/team/12
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

—

Delete Team

HTTPS Method	DELETE
URI	https://<server>/adminapi/team/{id}
Example URI	https://uccx-server/adminapi/team/12
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 405
Error Codes	—

Success Response

—

Trigger

You must configure Unified CM Telephony triggers to invoke application scripts in response to incoming contacts. A Unified CM Telephony trigger responds to calls that arrive on a specific route point by selecting telephony and media resources to serve the call and invoking an application script to handle the call.

Trigger Data Structure

Attribute Name	Base Type	Restrictions	Description
directoryNumber	directoryNumber	minOccurs = 1 maxOccurs = 1	A unique phone number.
locale	RestrictedString	minOccurs = 0 maxOccurs = 1	Defines the locale.
application	nameUriPair	minOccurs = 1 maxOccurs = 1	Application name used to associate with the trigger.
deviceName	deviceName	minOccurs = 1 maxOccurs = 1	A unique identifier for the device.
description	description	minOccurs = 1 maxOccurs = 1	A descriptive name for the CTI route point.

Attribute Name	Base Type	Restrictions	Description
callControlGroup	nameUriPair	minOccurs = 1 maxOccurs = 1	The call control group used to associate with the trigger.
triggerEnabled	boolean	minOccurs = 1 maxOccurs = 1 default = true	Used to enable or disable the trigger.
maxNumOfSessions	baseIdInt	minOccurs = 0 maxOccurs = 1	The maximum number of simultaneous calls that this trigger can handle.
idleTimeout	baseIdInt	minOccurs = 0 maxOccurs = 1 default = "5000"	The number of milliseconds (ms) the system should wait before rejecting the Unified CM Telephony request for this trigger .
overrideMediaTermination	overrideMediaTermination	minOccurs = 0 maxOccurs = 1	Used to enable or disable the media termination.
alertingNameAscii	CCMAlertingNameASCII	minOccurs = 0 maxOccurs = 1 default = ""	Displays the ASCII name alerts for the following scenarios: <ul style="list-style-type: none"> • if the device is not capable of handling the Unicode strings • if the locale on endpoint devices do not match • if the Unicode string is not specified
devicePool	CCMCommonObjectName	minOccurs = 0 maxOccurs = 1 default = "default"	The device pool to which you want to assign the route point.
location	CCMCommonObjectName	minOccurs = 0 maxOccurs = 1 default = "Hub_none"	The total bandwidth that is available for calls to/from a location.
partition	CCMPartitionName	minOccurs = 0 maxOccurs = 1 default = none	The partition to which the Directory Number belongs.

Attribute Name	Base Type	Restrictions	Description
voiceMailProfile	cucmString	minOccurs = 0 maxOccurs = 1	A list of profiles defined in the Voice Mail Profile Configuration.
callingSearchSpace	CCMCommonObjectName	minOccurs = 0 maxOccurs = 1	A collection of partitions that are searched for numbers that are called from the directory number.
callingSearchSpaceForRedirect	enum	minOccurs = 0 maxOccurs = 1 default = "default" enum values = default, calling.party, redirecting.party	A collection of partitions that are searched to determine how a redirected call is routed. Redirect Calling Search Space options: <ul style="list-style-type: none"> • DN Calling Search Space—Enables the CTI Port to use its directory number CSS when performing a redirect / consult transfer. • Calling Party—Enables the CTI Port to use the calling party's CSS when performing a redirect / consult transfer. • Redirect Party—Enables the CTI Port to use the CTI Route Point's CSS when performing a redirect / consult transfer.
presenceGroup	CCMCommonObjectName	minOccurs = 0 maxOccurs = 1 default = "Standard Presence group"	A list of groups to integrate the device with the <i>iPass</i> server.
forwardBusy	forwardBusy	minOccurs = 0 maxOccurs = 1	Used to configure call forward and pickup settings.

Simple Data Structure and Elements Used in Trigger

Attribute Name	Base Type	Restrictions	Description
directoryNumber	string	minLength = 0 maxLength = 50 To support E.164 compliance, Unified CCX allows to add "+" preceding to an agent extension or a route point directory number. Including "+", the directory number includes numeric characters and the following special characters: X # [] - *	A unique phone number.
deviceName	string	minLength = 1 maxLength = 15 Description type cannot contain special characters <, >, ?, %, (,), \, /, +, ' and cannot have a leading or trailing spaces.	A unique identifier for this device.
description	string	minLength = 1 maxLength = 50 Description type cannot contain special characters ", \, &, <, >, / and cannot have a leading space.	A descriptive name for the CTI route point.
cucmString	string	minLength = 1 maxLength = 50 Can contain numbers from 0-9, characters a-z, A-Z, .- _ and empty spaces.	Unicode string representing route point information.
display	string	minOccurs = 1 maxOccurs = 30 default = "" Can contain a maximum of 30 alphanumeric characters except \,], ", %, <, >, [, &, { and }.	Used to display the user name or the directory number.

Attribute Name	Base Type	Restrictions	Description
externalPhoneMaskNumber	string	minOccurs = 0 maxOccurs = 1 maxLength = 24 All numeric character 0-9 and character #, +, and X are allowed.	externalPhoneMaskNumber
overrideMediaTermination	sequence	minOccurs = 0 maxOccurs = unbounded	Used to enable or disable the media termination.
CCMAAlertingNameASCII	string	minLength = 0 maxLength = 30 All characters are allowed except double quotation marks ("), brackets (<,>), ampersand (&), and percent (%). Do not use the following keywords that are reserved in Unified Communications Manager: Conference, Park Number, Barge, CBarge, Monitor, and Record.	Displays the ASCII name alerts for the following scenarios: <ul style="list-style-type: none"> • The device is not capable of handling the Unicode strings. • The locales on endpoint devices do not match. • The Unicode string is not specified.

Complex Data Structure Defined for forwardBusy

Attribute Name	Base Type	Restrictions	Description
forwardBusyVoiceMail	boolean	default = false minOccurs = 0 maxOccurs = 1	Used to enable or disable settings in the Destination box and Calling Search Space.
forwardBusyDestination	directoryNumber	minOccurs = 0 maxOccurs = 1	Used to set a phone number, including an outside destination.
forwardBusyCallingSearchSpace	string	default = none minOccurs = 1 maxOccurs = 1	Used to apply the call forward setting on all devices that are using the same directory number.

Get list of triggers

HTTPS Method	GET
URI	https://<server>/adminapi/trigger
Example URI	https://uccx-server/adminapi/trigger
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 403, 404, 405
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Get a Trigger by Directory Number

HTTPS Method	GET
URI	https://<server>/adminapi/trigger/<trigger number>
Example URI	https://uccx-server/adminapi/trigger/876
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 403, 404, 405
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Create a New Trigger

HTTPS Method	POST
URI	https://<server>/adminapi/trigger
Example URI	https://uccx-server/adminapi/trigger
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201

HTTPS Failure Codes	400, 401, 405, 409
Error Codes	—

Success Response

Retrieves the unique URI pointing to the trigger.

Sample Input XML—Create a New Trigger with Default Parameters (POST)

```
<?xml version="1.0" encoding="UTF-8"?>
<trigger>
  <self rel="self" href="https://uccx-server/adminapi/trigger/3000" type="trigger"/>
  <directoryNumber>3000</directoryNumber>
  <locale>en_AU</locale>
  <application name="Auto_OB_Application">
    <refURL>https://uccx-server/adminapi/application/Auto_OB_Application</refURL>
  </application>
  <deviceName>abcd</deviceName>
  <description>abcd</description>
  <callControlGroup name="2">
    <refURL>https://uccx-server/adminapi/callControlGroup/2</refURL>
  </callControlGroup>
  <triggerEnabled>true</triggerEnabled>
  <maxNumOfSessions>20</maxNumOfSessions>
  <idleTimeout>5000</idleTimeout>
  <overrideMediaTermination>
    <dialogGroup name="0">
      <refURL>https://uccx-server/adminapi/dialogGroup/0</refURL>
    </dialogGroup>
  </overrideMediaTermination>
  <alertingNameAscii></alertingNameAscii>
  <devicePool>Default</devicePool>
  <location>Hub_None</location>
  <partition>None</partition>
  <voiceMailProfile>None</voiceMailProfile>
  <callingSearchSpace>None</callingSearchSpace>
  <callingSearchSpaceForRedirect>default</callingSearchSpaceForRedirect>
  <presenceGroup>Standard Presence group</presenceGroup>
  <forwardBusy>
    <forwardBusyVoiceMail>>false</forwardBusyVoiceMail>
    <forwardBusyDestination>
    </forwardBusyDestination>
    <forwardBusyCallingSearchSpace>None</forwardBusyCallingSearchSpace>
  </forwardBusy>
  <display></display>
  <externalPhoneMaskNumber></externalPhoneMaskNumber>
</trigger>
```

Modify a Trigger by ID

HTTPS Method	PUT
URI	https://<server>/adminapi/trigger/<trigger id>
Example URI	https://uccx-server/adminapi/trigger/876
Content Type	Application/XML, Application/JSON

HTTPS Success Code	200
HTTPS Failure Codes	400, 404
Error Codes	—

Success Response

HTTPS/1.1 200 OK

Delete a Trigger by ID

HTTPS Method	DELETE
URI	https://<server>/adminapi/trigger/<trigger id>
Example URI	https://uccx-server/adminapi/trigger/876
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 404
Error Codes	—

Success Response

HTTPS/1.1 200 OK

TTS Provider

TTS Provider provides the information about the text to speech service provider.

Data Structure for TTS Provider

Attribute Name	Base Type	Restrictions	Description
ttsProviderName	string	Regex: [A-Za-z0-9.-_s]+	Name of the TTS provider.
noOflicenses	int	minOccurs = 0 maxOccurs = 1 nillable = false	The number of TTS licenses.

Get list of TTS Providers

HTTPS Method	GET
URI	https://<server>/adminapi/ttsProvider
Example URI	https://uccx-server:8080/adminapi/ttsProvider
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 402, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ttsProviders>
  <ttsProvider>
    <ttsProviderId>47</ttsProviderId>
    <ttsProviderName>Scansoft Realspeak 4.0</ttsProviderName>
    <noOfLicenses>0</noOfLicenses>
  </ttsProvider>
  <ttsProvider>
    <ttsProviderId>51</ttsProviderId>
    <ttsProviderName>Nuance Vocalizer 4.0</ttsProviderName>
    <noOfLicenses>0</noOfLicenses>
  </ttsProvider>
</ttsProviders>
```

Get a TTS Provider

HTTPS Method	GET
URI	https://<server>/adminapi/ttsProvider/ttsProviderName
Example URI	https://uccx-server:8080/adminapi/ttsProvider/Scansoft Realspeak 4.0
Accept	Application/XML
HTTPS Success Code	200
HTTPS Failure Codes	401, 402, 404, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<ttsProvider>
  <ttsProviderId>44</ttsProviderId>
  <ttsProviderName>Scansoft Realspeak 4.0</ttsProviderName>
  <noOfLicenses>0</noOfLicenses>
</ttsProvider>
```

Create TTS Provider

HTTPS Method	POST
URI	https://<server>/adminapi/ttsProvider
Example URI	https://uccx-server:8080/adminapi/ttsProvider
Content Type	Application/XML
HTTPS Success Code	200, 201
HTTPS Failure Codes	401, 402, 405
Error Codes	—

Success Response

```
https://uccx-server/adminapi/ttsProvider/Nuance%20Vocalizer%204%2E0
```

Sample Input XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ttsProvider>
  <ttsProviderName>Scansoft Realspeak 4.0</ttsProviderName>
</ttsProvider>
```

Modify TTS Provider

HTTPS Method	PUT
URI	https://<server>/adminapi/ttsProvider/ttsProviderName
Example URI	https://uccx-server:8080/adminapi/ttsProvider/Scansoft Realspeak 4.0
Content Type	Application/XML
HTTPS Success Code	200, 201
HTTPS Failure Codes	401, 402, 404, 405
Error Codes	—

Sample Input XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ttsProvider>
  <ttsProviderName>Scansoft Realspeak 4.0</ttsProviderName>
  <noOfLicenses>50</noOfLicenses>
</ttsProvider>
```

Delete TTS Provider

HTTPS Method	DELETE
URI	https://<server>/adminapi/ttsProvider/ttsProviderName
Example URI	https://uccx-server:8080/adminapi/ttsProvider/Scansoft Realspeak 4.0
Content Type	Application/XML
HTTPS Success Code	200
HTTPS Failure Codes	401, 402, 404, 405
Error Codes	—

TTS Server

TTS Server provides the information about the text to speech server.

Data Structure for TTS Server

Attribute Name	Base Type	Restrictions	Description
ttsServerName	string	Regex: [A-Za-z0-9_-]+ \d{1,3}\d{1,3}\d{1,3}\d{1,3}	IP Address or hostname of the TTS server.
providerName	providerType	Regex: [A-Za-z0-9-_\s]+	Name of the TTS provider.
portNumber	int	minInclusive = 0 maxInclusive = 65535	Port number used for the TTS.
locales	localeType	minOccurs = 1 maxOccurs = unbounded nillable = false	The type of locale used for the TTS.

Attribute Name	Base Type	Restrictions	Description
language	string	minOccurs = 1 maxOccurs = 1 nillable = false	The type of language used for the TTS.
gender	enumeration	minOccurs = 1 maxOccurs = 1 nillable = false Enumeraton: (Male, Female and Neutral)	The type of gender used for the TTS.
defaultGender	enumeration	minOccurs = 1 maxOccurs = 1 nillable = false Enumeraton: (Male, Female and Neutral)	The default gender used for the TTS.

Get list of TTS Servers

HTTPSs Method	GET
URI	https://<server>/adminapi/ttsServer
Example URI	https://uccx-server:8080/adminapi/ttsServer
Accept	Application/XML, Application/JSON
HTTPSs Success Code	200
HTTPSs Failure Codes	401, 402, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ttsServers>
  <ttsServer>
    <ttsServerId>48</ttsServerId>
    <ttsServerName>localhost</ttsServerName>
    <providerName>Scansoft Realspeak 4.0</providerName>
    <portNumber>4900</portNumber>
    <locales>
      <language>en_US</language>
      <gender>Female</gender>
      <defGender>Female</defGender>
    </locales>
  </ttsServer>
  <ttsServer>
    <ttsServerId>49</ttsServerId>
    <ttsServerName>localhost2</ttsServerName>
    <providerName>Scansoft Realspeak 4.0</providerName>
    <portNumber>4900</portNumber>
    <locales>
      <language>en_US</language>
      <gender>Female</gender>
      <defGender>Female</defGender>
    </locales>
  </ttsServer>
</ttsServers>
```

Get a TTS Server

HTTPSs Method	GET
URI	https://<server>/adminapi/ttsServer/ttsProviderName
Example URI	https://uccx-server:8080/adminapi/ttsServer/localhost
Accept	Application/XML
HTTPSs Success Code	200
HTTPSs Failure Codes	401, 402, 404, 405
Error Codes	—

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<ttsServer>
  <ttsServerId>48</ttsServerId>
  <ttsServerName>localhost</ttsServerName>
  <providerName>Scansoft Realspeak 4.0</providerName>
  <portNumber>4900</portNumber>
  <locales>
    <language>en_US</language>
    <gender>Female</gender>
    <defGender>Female</defGender>
  </locales>
</ttsServer>
```

Create TTS Server

HTTPS Method	POST
URI	https://<server>/adminapi/ttsServer
Example URI	https://uccx-server:8080/adminapi/ttsServer
Content Type	Application/XML
HTTPS Success Code	200, 201
HTTPS Failure Codes	401, 402, 405
Error Codes	—

Success Response

```
https://uccx-server/adminapi/ttsServer/localhost
```

Sample Input XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ttsServer>
  <ttsServerName>localhost</ttsServerName>
  <providerName>Scansoft Realspeak 4.0</providerName>
  <portNumber>4900</portNumber>
  <locales>
    <language>en_US</language>
    <gender>Male</gender>
    <defGender>Female</defGender>
  </locales>
</ttsServer>
```

Modify TTS Server

HTTPS Method	PUT
URI	https://<server>/adminapi/ttsServer/localhost
Example URI	https://uccx-server:8080/adminapi/ttsServer/localhost
Content Type	Application/XML
HTTPS Success Code	200, 201
HTTPS Failure Codes	401, 402, 404, 405
Error Codes	—

Sample Input XML

```
<?xml version="1.0" encoding="UTF-8"?>
<ttsServer>
  <ttsServerName>localhost</ttsServerName>
  <providerName>Scansoft Realspeak 4.0</providerName>
  <portNumber>4900</portNumber>
  <locales>
    <language>en_US</language>
    <gender>Female</gender>
    <defGender>Female</defGender>
  </locales>
  <locales>
    <language>en_SZ</language>
    <gender>Male</gender>
    <defGender>Male</defGender>
  </locales>
</ttsServer>
```

Delete TTS Server

HTTPS Method	DELETE
URI	https://<server>/adminapi/ttsServer/localhost
Example URI	https://uccx-server:8080/adminapi/ttsServer/localhost
Content Type	Application/XML
HTTPS Success Code	200
HTTPS Failure Codes	401, 402, 404, 405
Error Codes	—

WrapupCategory

WrapupCategory provides information about the reasons that agents can apply to the chats and email handled. A Wrap-Up reason indicates why a customer called the contact center.

Data Structure for WrapupCategory

Attribute Name	Base Type	Restrictions	Description
id	int	minOccurs = 0 maxOccurs = 1	Unique identifier of the WrapupCategory.
name	string	Should have 1 to 40 characters length	Name of the WrapupCategory.
type	string	Currently only 'nonvoice' is permitted	Type of the WrapupCategory.

Attribute Name	Base Type	Restrictions	Description
global	boolean	none	Flag indicating whether the WrapupCategory is Global or CSQ level .
wrapupReasons	ComplexType element containing the ComplexType element 'reason' with elements 'id' of type 'int' and 'text' of type 'string'	Reason should be minOccurs = 1 maxOccurs = 25 Text should have 1 to 40 chars length and there should be no duplicates	All the Reasons associated with this Category .
associatedCSQs	ComplexType element containing the element 'csq' which is of the type 'nameUriPair'	Associated CSQs should be there only if the type is not global.	All the CSQs associated with this category .

Other Restrictions

- There can be a maximum number of 25 categories.
- There can be maximum 10 global categories.
- The category name should be unique.
- CSQs associated should be of chat/email type and should be existing.
- A Wrap-Up Category can be mapped to any number of CSQs.
- A CSQ can be associated with a maximum number of 10 Wrap-Up categories.
- Special characters like <, >, ~, ` , !, @, #, \$, %, ^, &, *, (,), =, |, [,], {, }, ;, /, '+, :, " , and comma are not allowed in the attributes, WrapupCategory name and wrapupReasons text.

Get list of WrapupCategory

HTTPS Method	GET
URI	https://<server>/adminapi/wrapupcategory

Example URI	https://uccx-server:8080/adminapi/wrapupcategory
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401
Error Codes	500

Success Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<wrapupCategories>
  <wrapupCategory>
    <id>1</id>
    <name>CategoryName1</name>
    <type>nonvoice</type>
    <global>true</global>
    <wrapupReasons>
      <reason>
        <id>1</id>
        <text>Reason1</text>
      </reason>
      <reason>
        <id>2</id>
        <text>Reason2</text>
      </reason>
    </wrapupReasons>
    <associatedCSQs/>
  </wrapupCategory>
  <wrapupCategory>
    <id>2</id>
    <name>CategoryName2</name>
    <type>nonvoice</type>
    <global>>false</global>
    <wrapupReasons>
      <reason>
        <id>3</id>
        <text>Reason3</text>
      </reason>
      <reason>
        <id>4</id>
        <text>Reason4</text>
      </reason>
    </wrapupReasons>
    <associatedCSQs>
      <csq name="1">
        <refURL>https://10.100.10.100/adminapi/csq/1</refURL>
      </csq>
      <csq name="2">
        <refURL>https://10.100.10.100/adminapi/csq/2</refURL>
      </csq>
    </associatedCSQs>
  </wrapupCategory>
</wrapupCategories>
```

Get WrapupCategory

HTTPS Method	GET
URI	https://<server>/adminapi/wrapupcategory/{id}
Example URI	https://uccx-server:8080/adminapi/wrapupcategory/1
Accept	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	401, 404
Error Codes	500

Success Response

```
<?xml version="1.0" encoding="UTF-8"?>
<wrapupCategory>
  <id>1</id>
  <name>CategoryName1</name>
  <type>nonvoice</type>
  <global>true</global>
  <wrapupReasons>
    <reason>
      <id>1</id>
      <text>Reason1</text>
    </reason>
    <reason>
      <id>2</id>
      <text>Reason2</text>
    </reason>
  </wrapupReasons>
  <associatedCSQs/>
</wrapupCategory>
```

Create WrapupCategory

HTTPS Method	POST
URI	https://<server>/adminapi/wrapupcategory
Example URI	https://uccx-server:8080/adminapi/wrapupcategory
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	400, 401, 409, 412
Error Codes	500

Success Response

HTTPS/1.1 201 Created

Sample Input XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<wrapupCategory>
  <name>CategoryName1</name>
  <type>nonvoice</type>
  <global>true</global>
  <wrapupReasons>
    <reason>
      <id>1</id>
      <text>Reason1</text>
    </reason>
    <reason>
      <id>2</id>
      <text>Reason2</text>
    </reason>
  </wrapupReasons>
  <associatedCSQs/>
</wrapupCategory>
```

Modify WrapupCategory

HTTPS Method	PUT
URI	https://< server >/adminapi/wrapupcategory/{id}
Example URI	https://uccx-server/adminapi/wrapupcategory/1
Content Type	Application/XML, Application/JSON
HTTPS Success Code	200
HTTPS Failure Codes	400, 401, 409, 412
Error Codes	500

Success Response

HTTPS/1.1 200 OK

Sample Input XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<wrapupCategory>
  <name>CategoryName1</name>
  <type>nonvoice</type>
  <global>true</global>
  <wrapupReasons>
    <reason>
      <id>1</id>
      <text>Reason1</text>
    </reason>
    <reason>
      <id>2</id>
      <text>Reason2</text>
    </reason>
  </wrapupReasons>
  <associatedCSQs/>
</wrapupCategory>
```

Delete WrapupCategory

HTTPS Method	DELETE
URI	https://< server >/adminapi/wrapupcategory/{id}
Example URI	https://uccx-server/adminapi/wrapupcategory/1
Content Type	Application/XML, Application/JSON
HTTPS Success Code	201
HTTPS Failure Codes	404
Error Codes	500

Success Response

HTTPS/1.1 200 OK



APPENDIX **A**

Cisco Finesse Desktop and Configuration APIs

The Cisco Web Services Developer Guide includes the Cisco Finesse desktop and configuration APIs, a list of API parameters, errors, and information about gadget development.

This guide is available on the Finesse page of the Cisco DevNet and located at <https://developer.cisco.com/site/finesse/docs>.



APPENDIX **B**

Cisco Customer Collaboration Platform APIs

The Cisco Customer Collaboration Platform Developers Guide includes the Customer Collaboration Platform APIs.

This guide is available on the Customer Collaboration Platform page of the Cisco DevNet and located at <https://developer.cisco.com/site/socialminer/overview/>.

