Configure Cisco Webex AI Agent for CCE

Contents

Introduction

This document describes how to configure Webex AI Agent for Contact Center Enterprise (CCE).

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Unified Contact Center Enterprise (UCCE) Release 15.0
- Customer Voice Portal (CVP) Release 15.0
- Customer Voice Portal (CVP) Call Studio
- Cisco Virtualized Voice Browser (VVB)

Components Used

The information in this document is based on these software versions:

- UCCE 15.0
- Webex CCE

- CVP 15.0

- Cisco Virtualized Voice Browser (VVB)
- Cloud Connect

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Webex AI Agent Overview

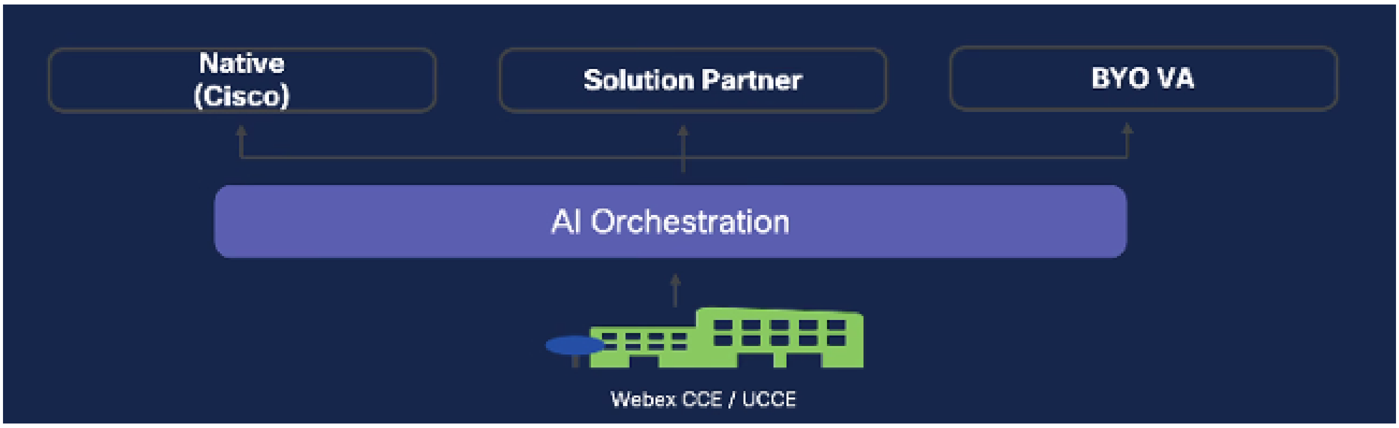
The Webex AI Agent is a virtual artificial intelligence assistant integrated into Cisco Webex CCE or CCE. It is designed to improve service and support by the used of automated interactions, assist live agents, and deliver insights powered by analytics.

Background

The AI Agent is built to engage with contact center users, interpret the input, generate appropriate responses based on how it is trained, and execute predefined tasks. This guide outlines the steps needed to implement an AI Agent within Contact Center Enterprise (CCE).

The Webex AI Agent is introduced as part of the new CCE 15.0 release.

Webex Agent AI Architecture

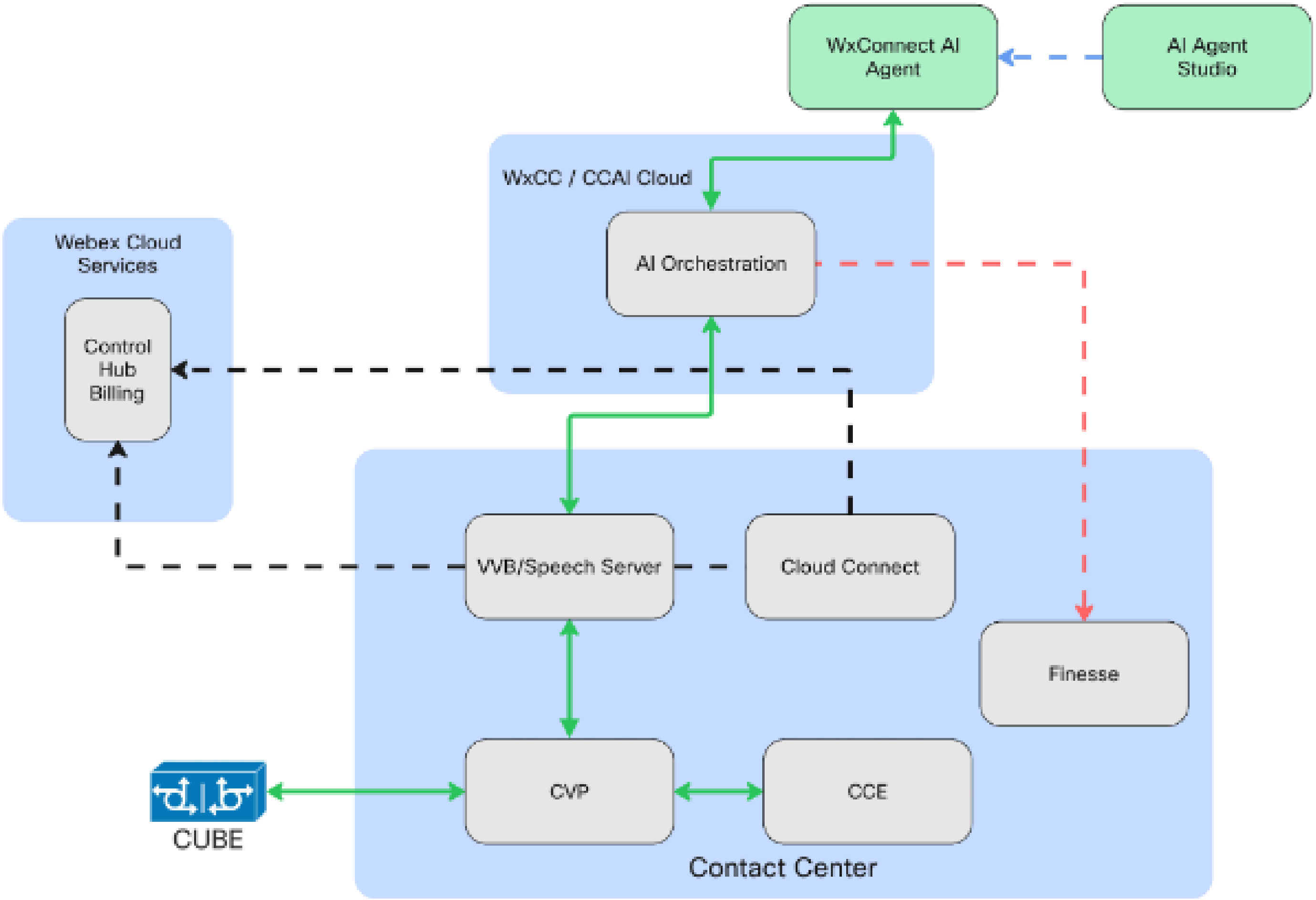


Webex Agent AI Architecture

Native: Cisco in-house solution for Virtual Agent

This is the type of Virtual agent that you configure in this lab.

Webex Agent AI Components



Webex Agent AI Components

In both the CCE on-premises solution and the Webex CCE (WxCCE) tenant, several core components form the foundation of the Webex AI Agent solution. These components span both on-prem and cloud environments and work together to enable intelligent, AI-driven interactions.

On-Premises Components

- **Cisco Customer Voice Portal (CVP):**
  CVP acts as an Interactive Voice Response (IVR) system integrated with CCE/WxCCE. It enables automated self-service and call routing. CVP collaborates with the VVB and Speech Server to integrate the AI Agent within the IVR flow.

- **CVP Call Studio:** a development environment used to design and build voice applications for CVP

- **Cisco Virtualized Voice Browser (VVB) / Speech Server:**
  VVB handles multimedia and IVR processing. The Speech Server communicates caller requests to the AI orchestration services using gRPC protocol.

- **CCE Core:**
  CCE Core includes key components such as Peripheral Gateways, the Router, and Admin Workstation (AW). These work together to process requests and determine the appropriate destination. The AW is used to configure and manage the entire CCE environment.

- **Cloud Connect:**
  Cloud Connect acts as the secure bridge between the on-prem CCE infrastructure and Cisco cloud services, enabling seamless integration.

Cloud Components

- **Webex Cloud Services – Control Hub:** a centralized management interface for configuring Webex cloud solutions. From Control Hub, users can provision and launch the AI Agent Studio
- **AI Orchestration Services:** Cisco's cloud-based orchestration layer that receives incoming requests from the CCE or WxCCE system and routes them to the Webex Connect AI Agent via a Cisco Connector
- **Webex Connect AI Agent:** a cloud-hosted AI Agent powered by Webex Connect, designed to handle interactions intelligently
- **AI Agent Studio:** a visual interface used to design workflows and scripts for the Scripted AI Agent, enabling tailored engagement experiences

Configure Scripted And Autonomous AI Agent on CCE

For information on how to create a Scripted or Autonomous agent, please refer to this document.

Configure Webex Services - Control Hub

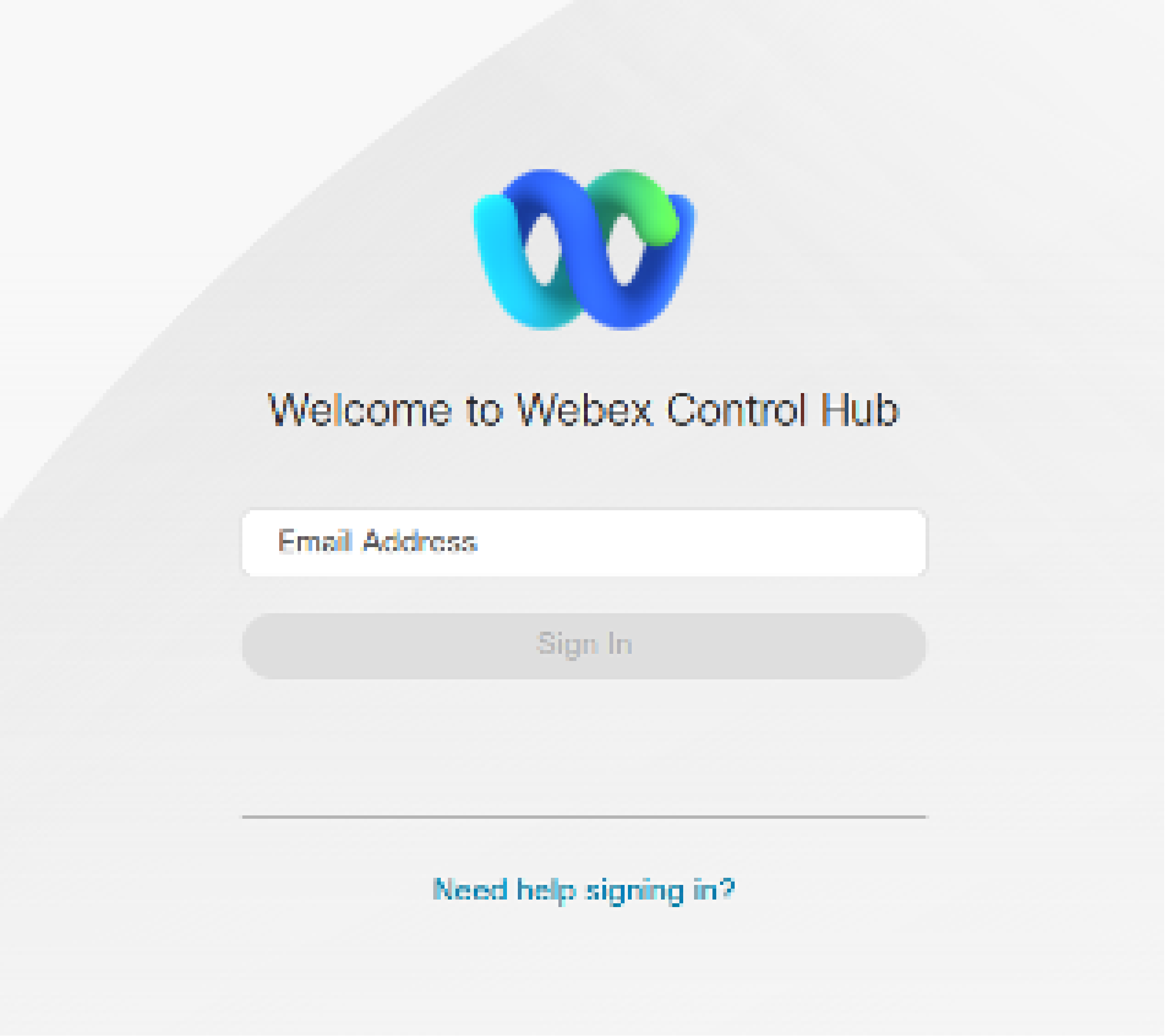
Webex Control Hub serves as the central interface for managing your organization, users, service entitlements, and the configuration of Webex Contact Center and other services. It enables provisioning of all Webex services within a single customer organization.

Here are the Webex AI Agent configuration tasks you can test within your Control Hub organization:

- **Register Cloud Connect**
- **Launch the AI Agent Builder**

Register Cloud Connect

Step 1. Open a web browser and navigate to Control Hub: **admin.webex.com**


Welcome to Webex Control Hub

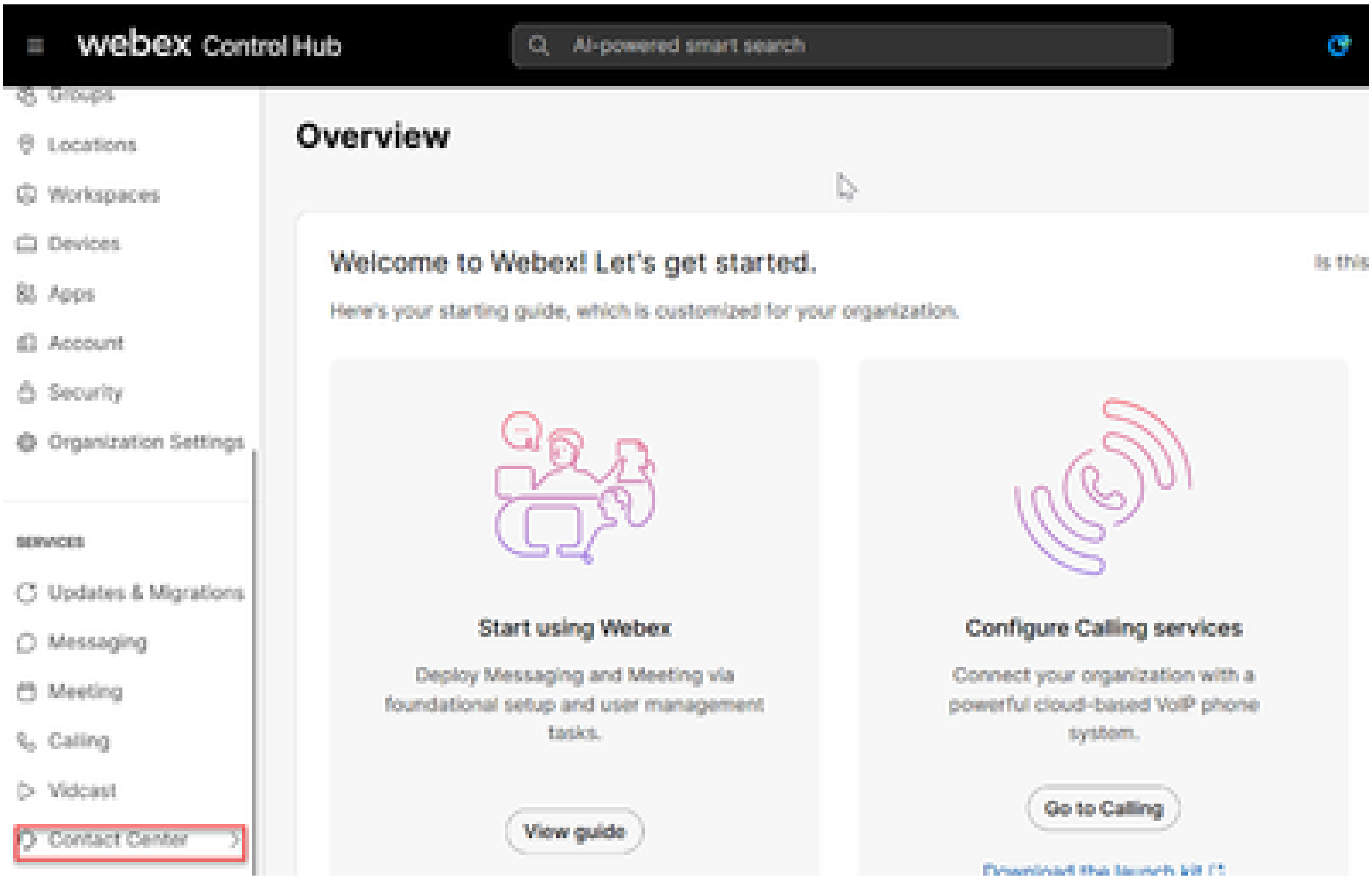
Email Address

Sign In

Need help signing in?

Control hub signin

Step 2. In **Control Hub**, navigate to the **Contact Center** section from the left-hand menu.



Welcome to Webex! Let's get started.

Here's your starting guide, which is customized for your organization.

Start using Webex

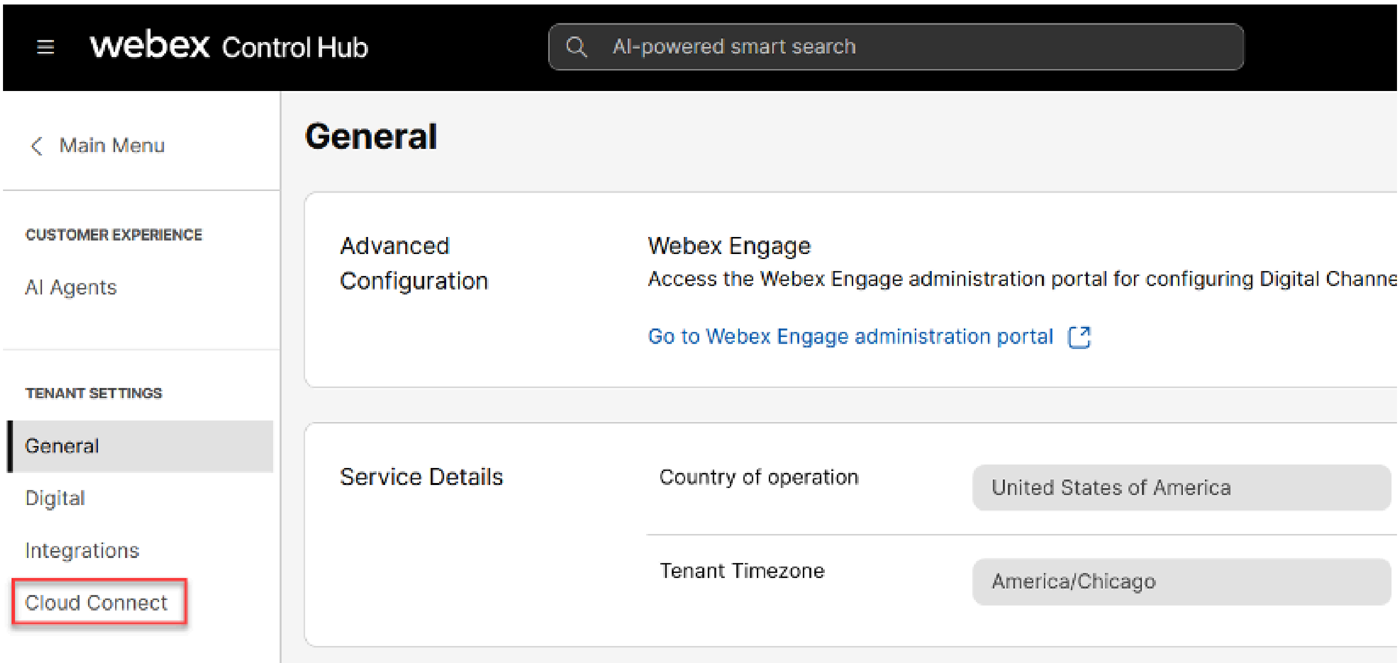
Deploy Messaging and Meeting via foundational setup and user management tasks.

Configure Calling services

Connect your organization with a powerful cloud-based VoIP phone system.

Control hub overview

Step 3. On the **Contact Center** page, click **Cloud Connect** to begin the registration process.



Advanced Configuration

Webex Engage

Access the Webex Engage administration portal for configuring Digital Channe

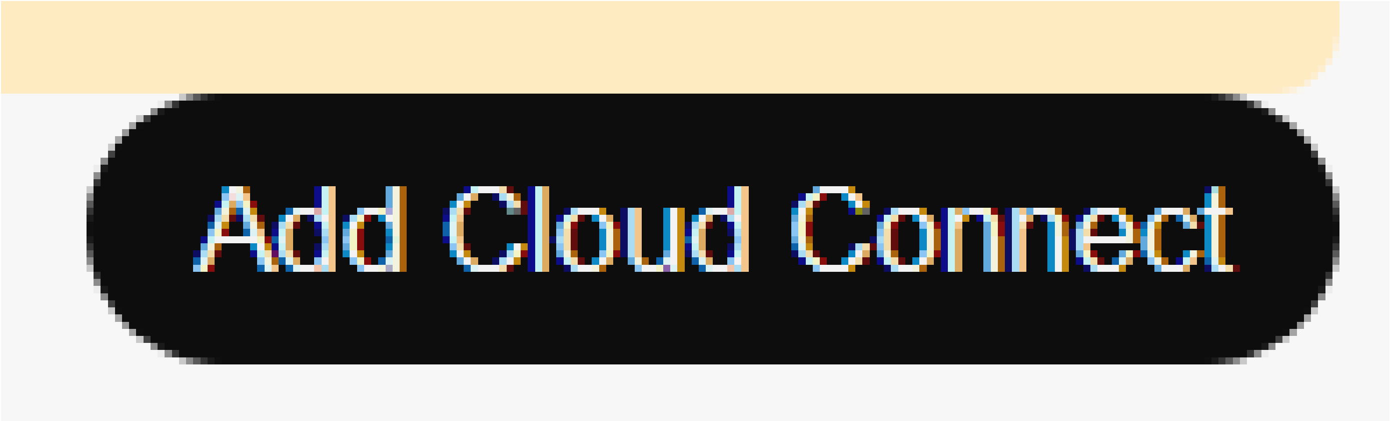
Service Details

Country of operation: United States of America

Tenant Timezone: America/Chicago

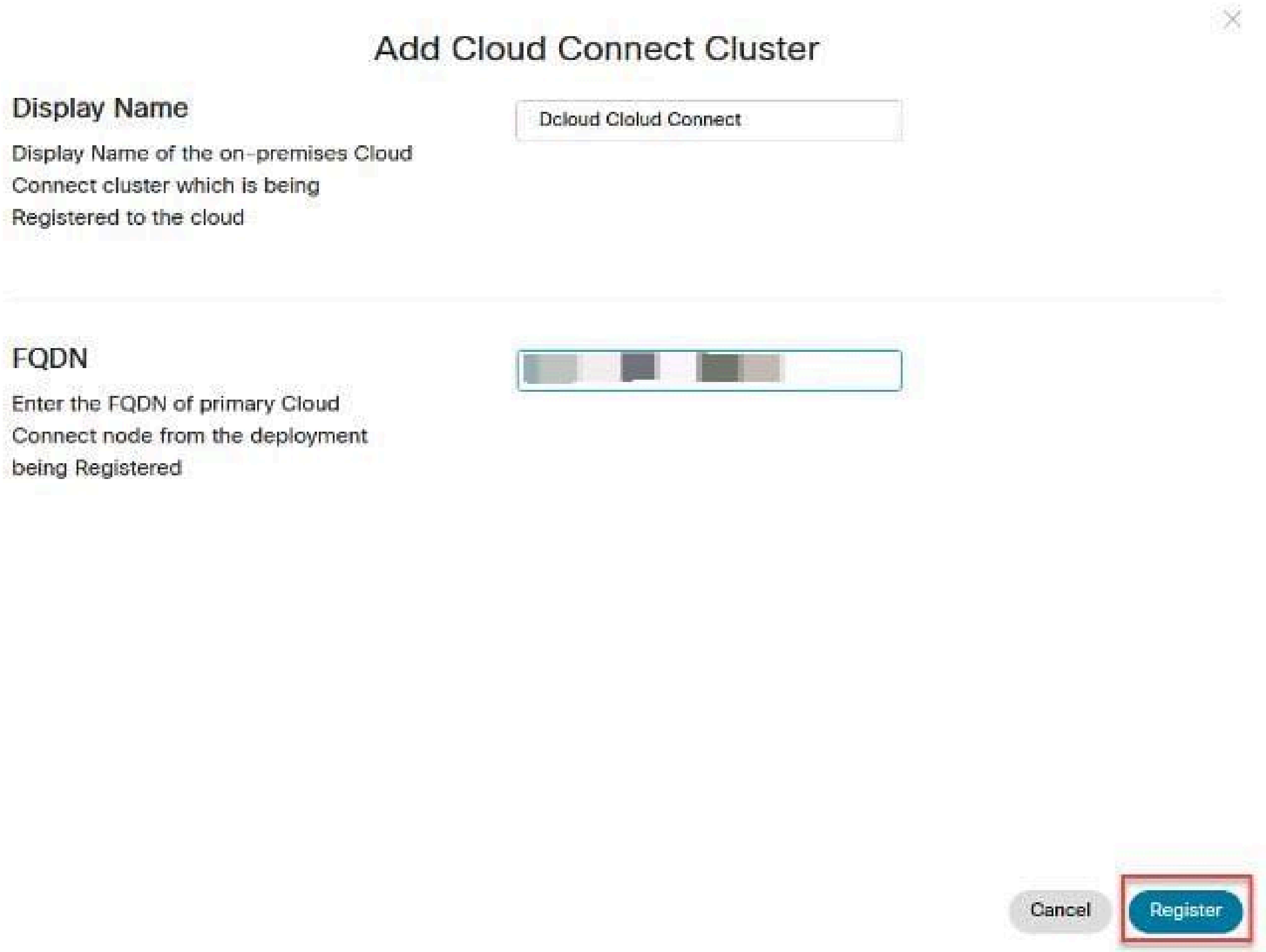
Control hub cloud connect

Step 4. On the **Cloud Connect** page, click **Add Cloud Connect**.



Add cloud connect

Step 5. Enter the **Name** and the Fully Qualified Domain Name **(FQDN)**, then click **Register**.



Add Cloud Connect Cluster

Display Name

Display Name of the on-premises Cloud Connect cluster which is being Registered to the cloud

Dcloud Clolud Connect

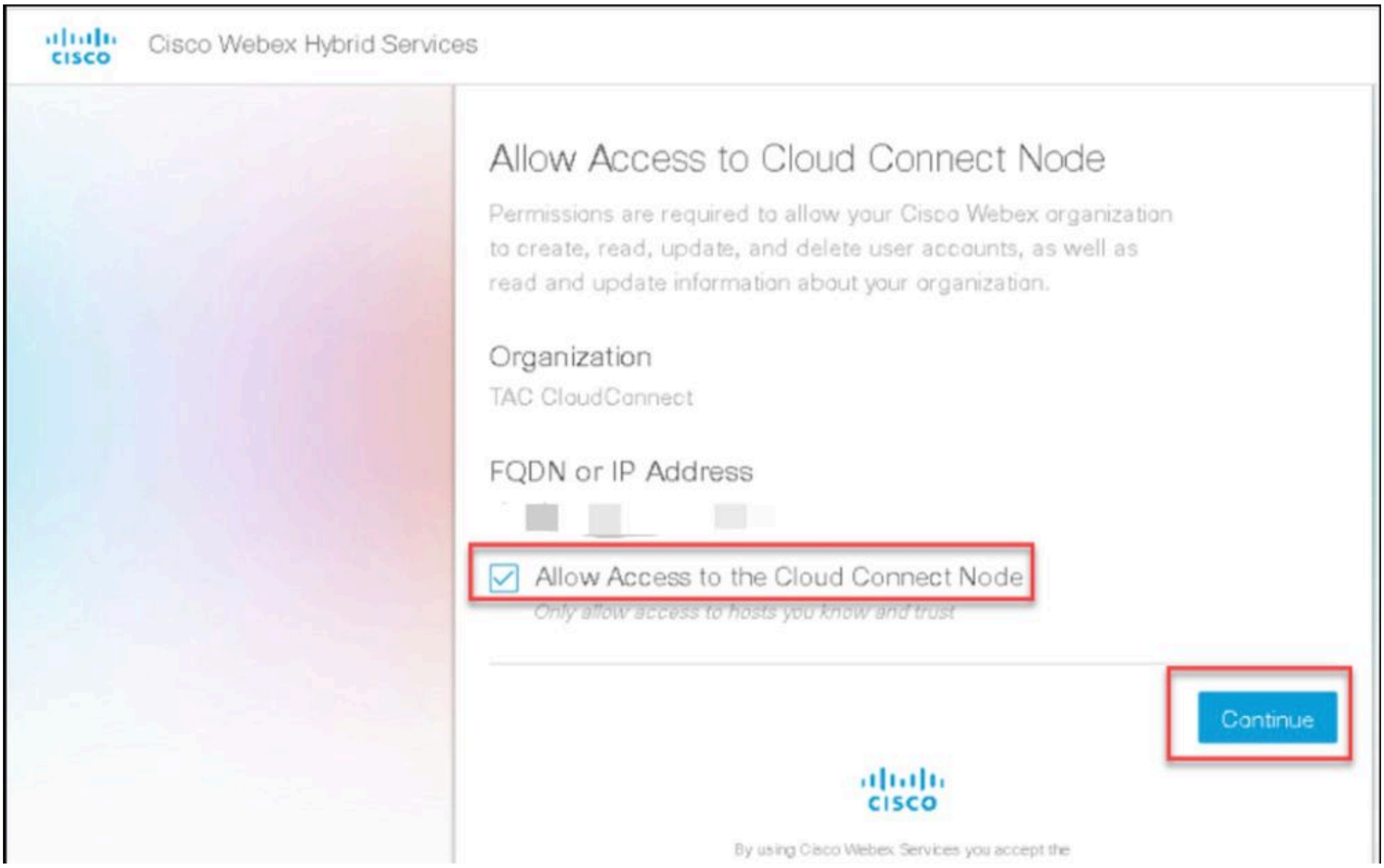
FQDN

Enter the FQDN of primary Cloud Connect node from the deployment being Registered

[redacted]

add cc cluster

Step 6. In the **Allow Access** window, check the box for **Allow Access to the Cloud Connect Node**, then click **Continue**.



Cisco Webex Hybrid Services

Allow Access to Cloud Connect Node

Permissions are required to allow your Cisco Webex organization to create, read, update, and delete user accounts, as well as read and update information about your organization.

Organization

TAC CloudConnect

FQDN or IP Address

[redacted]

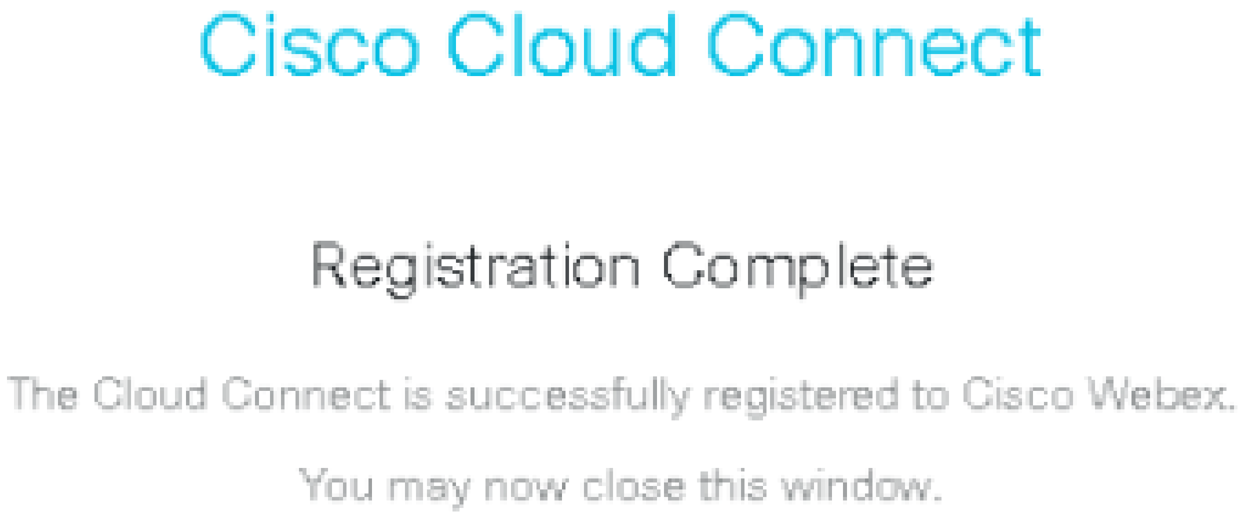
☑ Allow Access to the Cloud Connect Node

Only allow access to hosts you know and trust

By using Cisco Webex Services you accept the

Allow Access to CC

Step 7. Once you see the confirmation that registration is complete, you can close the window.



Cisco Cloud Connect

Registration Complete

The Cloud Connect is successfully registered to Cisco Webex.

You may now close this window.

Cloud connect registeration complete

Configure CCE

The steps required to prepare CCE for Webex AI Agent implementation are:

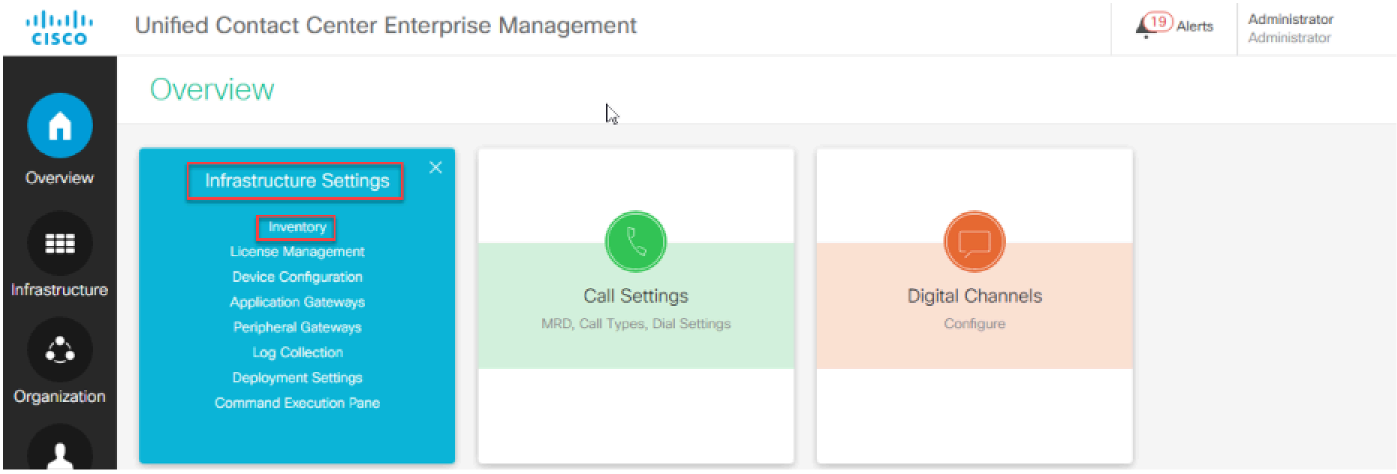
- Cloud Connect registration
- Cloud Connect proxy configuration
- Cloud Connect Feature Management status

Cloud Connect Registration

Step 1. Open a Web browser and navigate to the PCCE Web Administration Tool also known as the Single Pane of Glass (SPOG).

Step 2. Login with the administrator credentials.

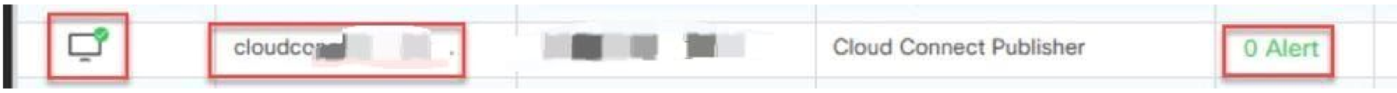
Step 3. In the Web Admin interface, select **Infrastructure Settings**, then choose **Inventory**.



Inventory

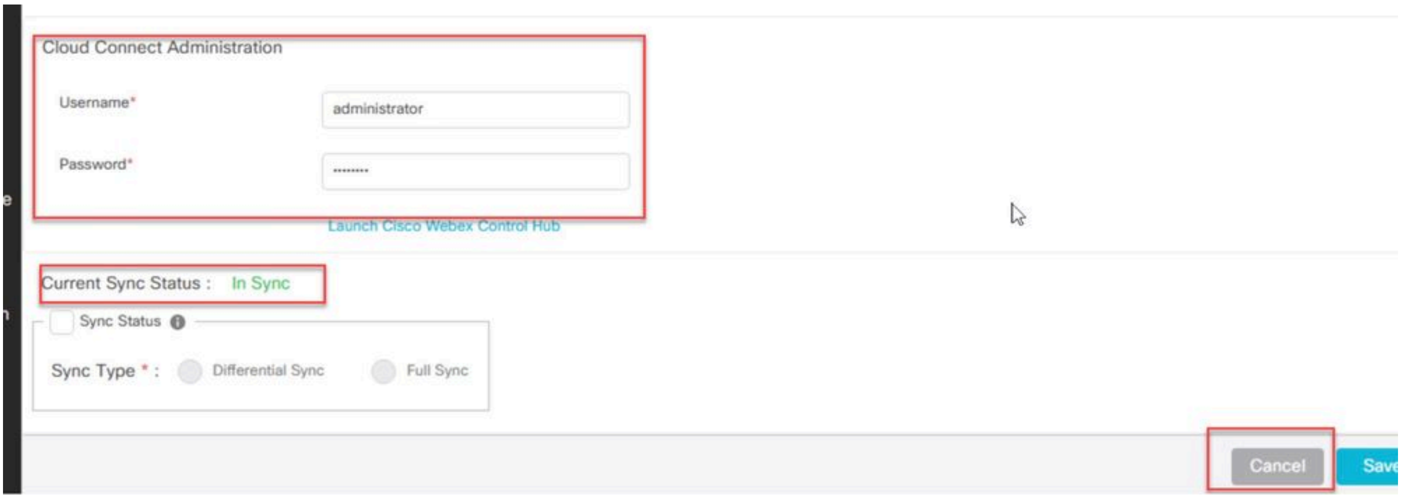
Step 4. On the **Inventory** page, locate the Cloud Connect server. Ensure its status is green and no alerts are present.

In this lab, only one Cloud Connect Server (Publisher) is shown. In a production environment, you typically see two servers: Publisher and Subscriber.



CC Publisher

Step 5. Click on the Cloud Connect server to view the Admin credentials and synchronization status with the AW server. Since no changes are needed, click **Cancel** to exit.



Cloud Connect Administration

Username* administrator

Password* ••••••••

Current Sync Status : In Sync

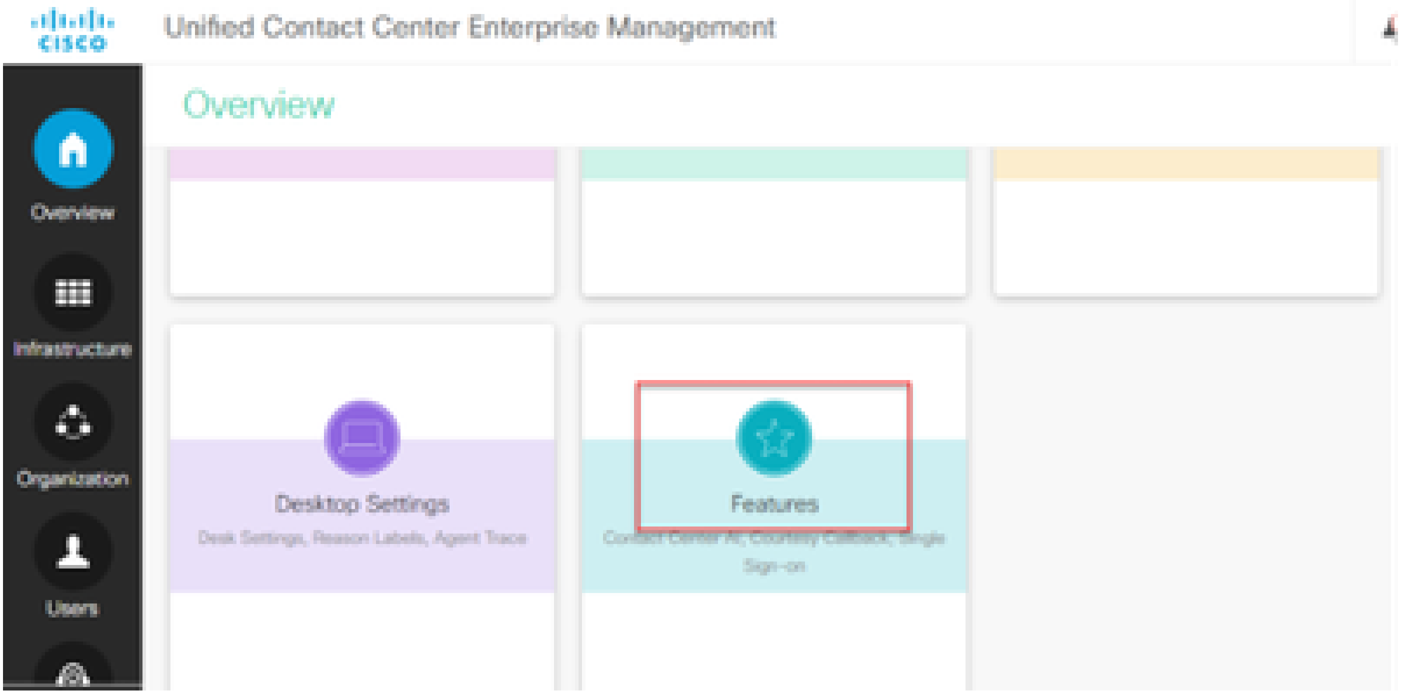
□ Sync Status

Sync Type * : ○ Differential Sync ○ Full Sync

Save

Cloud Connect Proxy Configuration

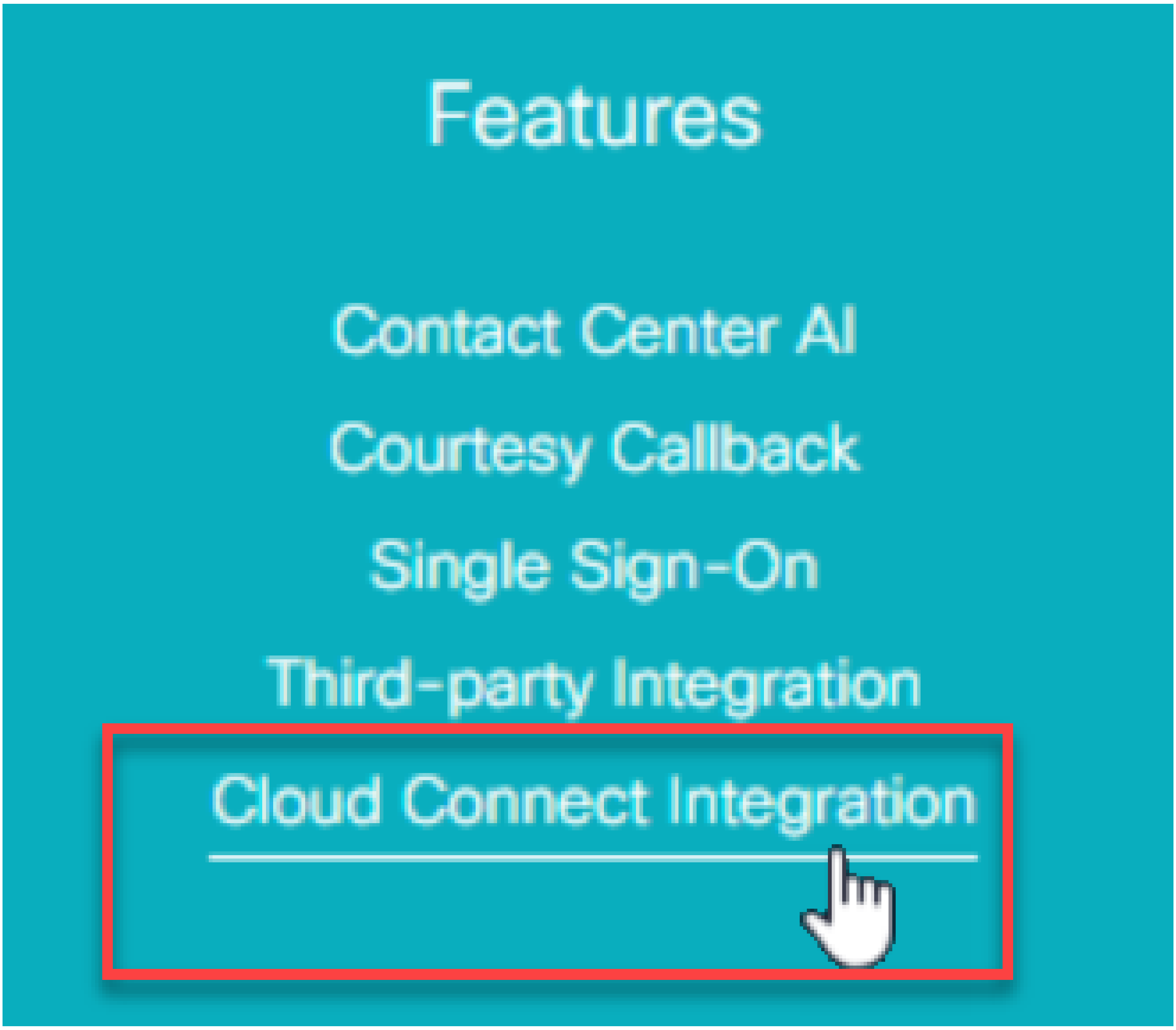
Step 1. In the SPOG, navigate to the Overview page and scroll down if needed to locate and select the **Features** card.



Unified Contact Center Enterprise Management
Overview
Overview
Infrastructure
Organization
Users
Desktop Settings
Desk Settings, Reason Labels, Agent Trace
Features
Contact Center AI, Courtesy Callback, Single Sign-on

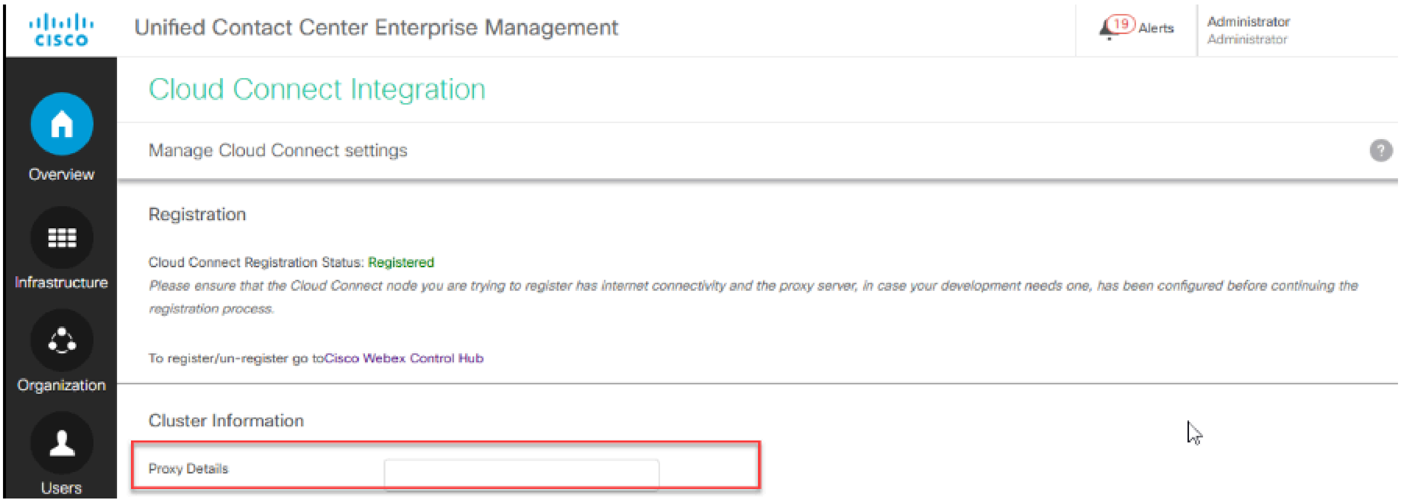
Feature

Step 2. On the Features card, click Cloud Connect Integration.



Feature selection

Step 3. View the registration status of Cloud Connect along with the proxy configuration details.



Cloud Connect Integration

Manage Cloud Connect settings

Registration

Cloud Connect Registration Status: Registered

Please ensure that the Cloud Connect node you are trying to register has internet connectivity and the proxy server, in case your development needs one, has been configured before continuing the registration process.

To register/un-register go toCisco Webex Control Hub

Cluster Information

Proxy Details

CC Integration

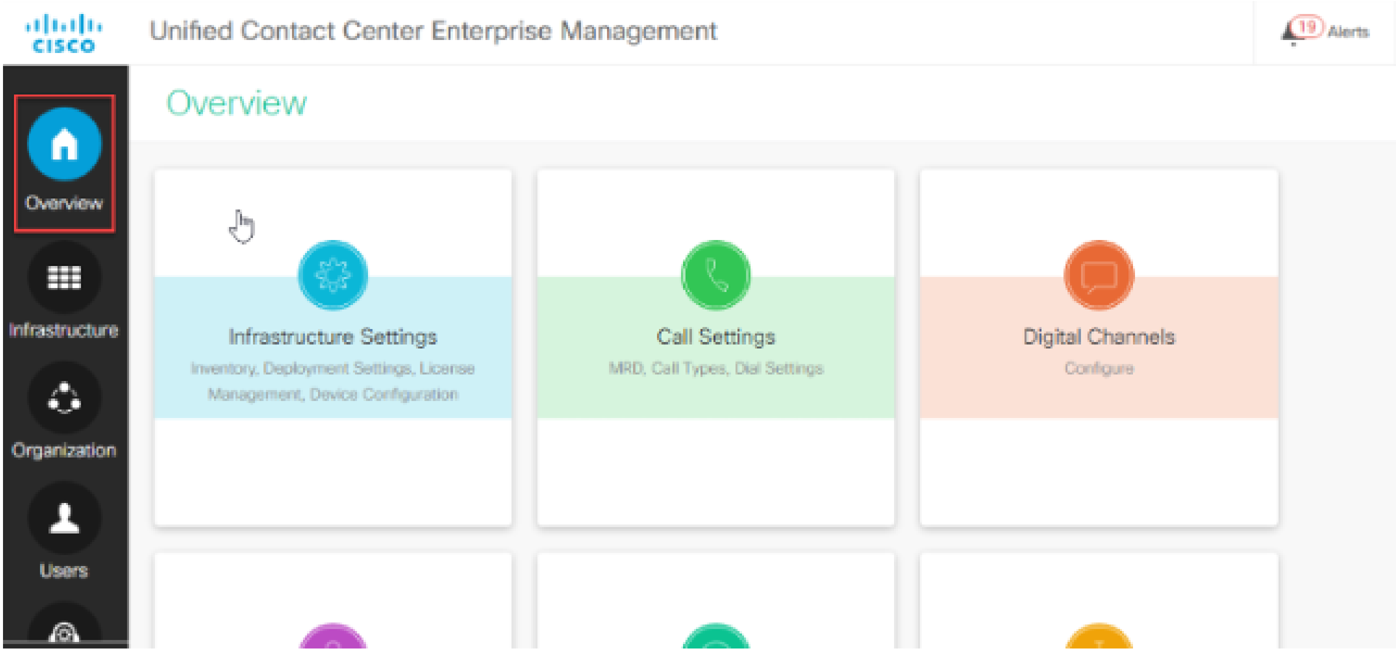
If your environment requires a proxy to communicate with Webex Services, enter the proxy details used by Cloud Connect.

For example: abc.cisco.com:8080

By default, HTTP uses port 80, but you can specify a different port number in the proxy configuration.

In this lab, a proxy is not required, so it is not configured.

Step 4. Click Overview to return to the main menu.


Unified Contact Center Enterprise Management
19 Alerts
Overview
Overview
Infrastructure
Organization
Users
Infrastructure Settings
Inventory, Deployment Settings, License Management, Device Configuration
Call Settings
MRD, Call Types, Dial Settings
Digital Channels
Configure

Overview

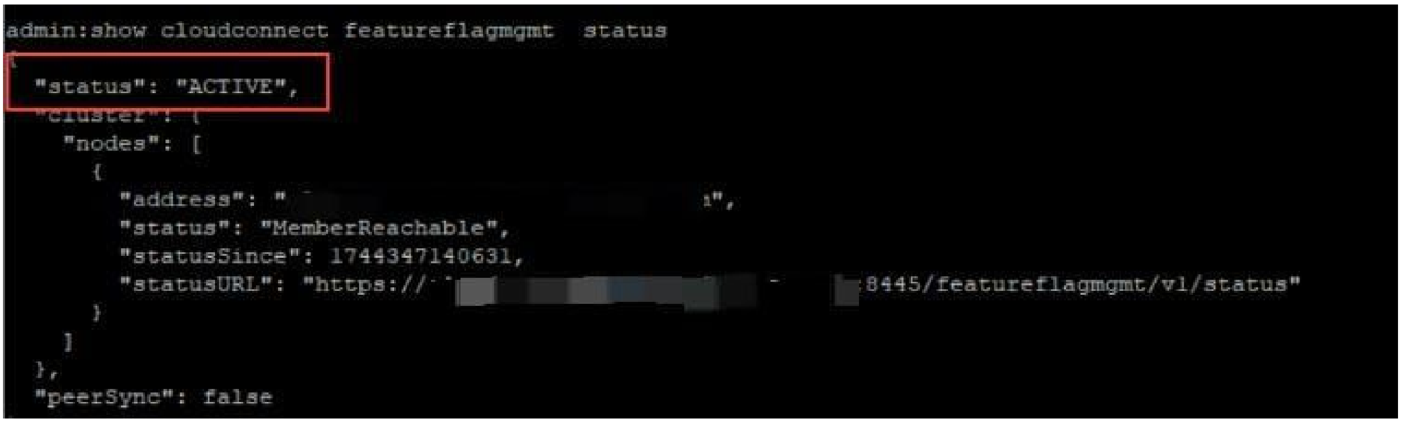
Cloud Feature Management Status

Step 1. Log in to the Cloud Connect Command Line Interface (CLI).

Step 2. In the Cloud Connect CLI, enter the mentioned command:

show cloudconnect featureflagmgmt status

You must see that the status is **ACTIVE**.



```
admin:show cloudconnect featureflagmgmt  status
{
  "status": "ACTIVE",
  "cluster": {
    "nodes": [
      {
        "address": "[redacted]",
        "status": "MemberReachable",
        "statusSince": 1744347140631,
        "statusURL": "https://[redacted]:8445/featureflagmgmt/v1/status"
      }
    ]
  },
  "peerSync": false
```

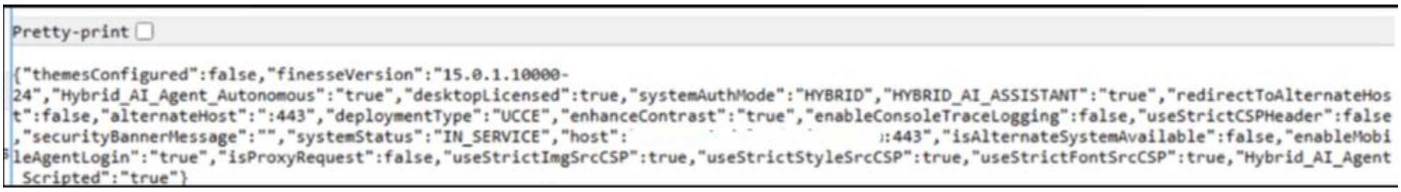
Active status

Note: To enable the Feature in Cloud Connect, Information as the SDK key is required. All the information required is provided by Contact Center Product Manager

Next, to verify the features enabled for this Webex tenant (including those needed for AI Agent implementation), you can run a specific API call in a web browser.

Step 3. Open a Web Browser and execute the mentioned API.

https://finesse1.dcloud.cisco.com/desktop/api/DesktopConfig

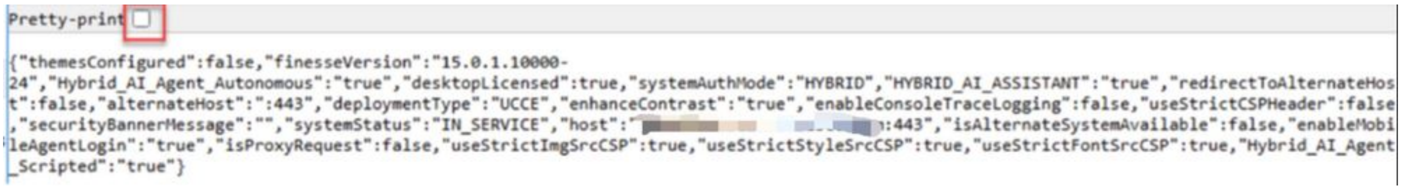


Pretty-print ☐

```
{"themesConfigured":false,"finesseVersion":"15.0.1.10000-
24","Hybrid_AI_Agent_Autonomous":"true","desktopLicensed":true,"systemAuthMode":"HYBRID","HYBRID_AI_ASSISTANT":"true","redirectToAlternateHos
t":false,"alternateHost":":443","deploymentType":"UCCE","enhanceContrast":"true","enableConsoleTraceLogging":false,"useStrictCSPHeader":false
,"securityBannerMessage":"","systemStatus":"IN_SERVICE","host":[redacted]:443","isAlternateSystemAvailable":false,"enableMobi
leAgentLogin":"true","isProxyRequest":false,"useStrictImgSrcCSP":true,"useStrictStyleSrcCSP":true,"useStrictFontSrcCSP":true,"Hybrid_AI_Agent
_Scripted":"true"}
```

API Output

Step 4. In the API response, check the **Pretty-print** box to format the results for easier reading.



Pretty-print ☐

```
{"themesConfigured":false,"finesseVersion":"15.0.1.10000-
24","Hybrid_AI_Agent_Autonomous":"true","desktopLicensed":true,"systemAuthMode":"HYBRID","HYBRID_AI_ASSISTANT":"true","redirectToAlternateHos
t":false,"alternateHost":":443","deploymentType":"UCCE","enhanceContrast":"true","enableConsoleTraceLogging":false,"useStrictCSPHeader":false
,"securityBannerMessage":"","systemStatus":"IN_SERVICE","host":"[redacted]:443","isAlternateSystemAvailable":false,"enableMobi
leAgentLogin":"true","isProxyRequest":false,"useStrictImgSrcCSP":true,"useStrictStyleSrcCSP":true,"useStrictFontSrcCSP":true,"Hybrid_AI_Agent
_Scripted":"true"}
```

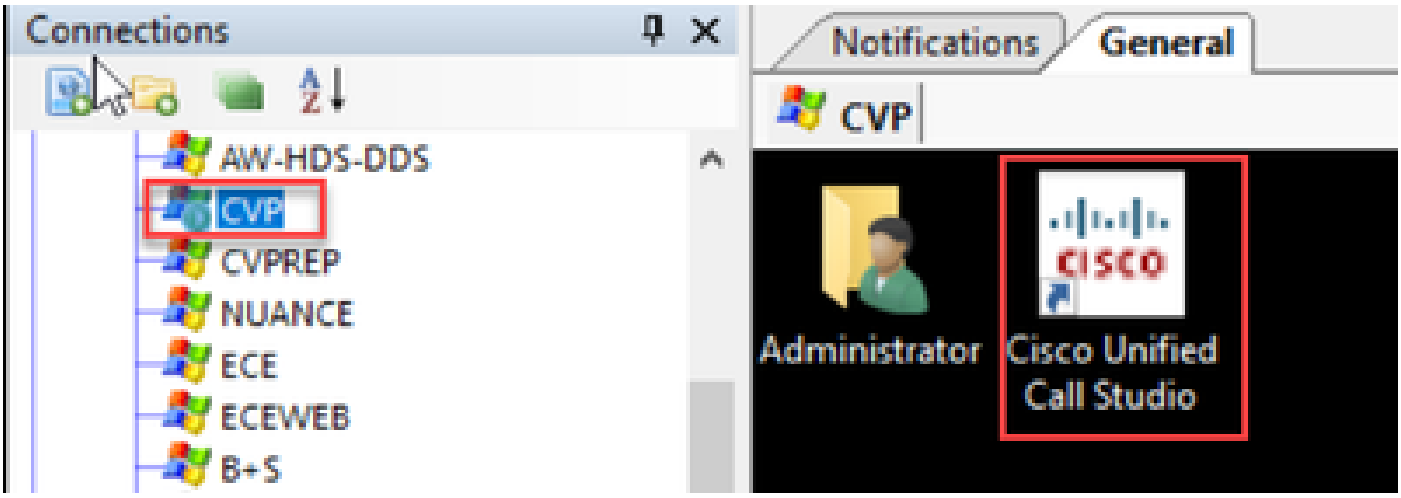
API Output

In this lab, since you have configured only a scripted AI Agent, the only required feature that the Cisco Product team needed to enable for this tenant is **Hybrid_AI_Agent_Scripted**, which you see is enabled.

Configure CVP Call Studio Application - Scripted AI Agent

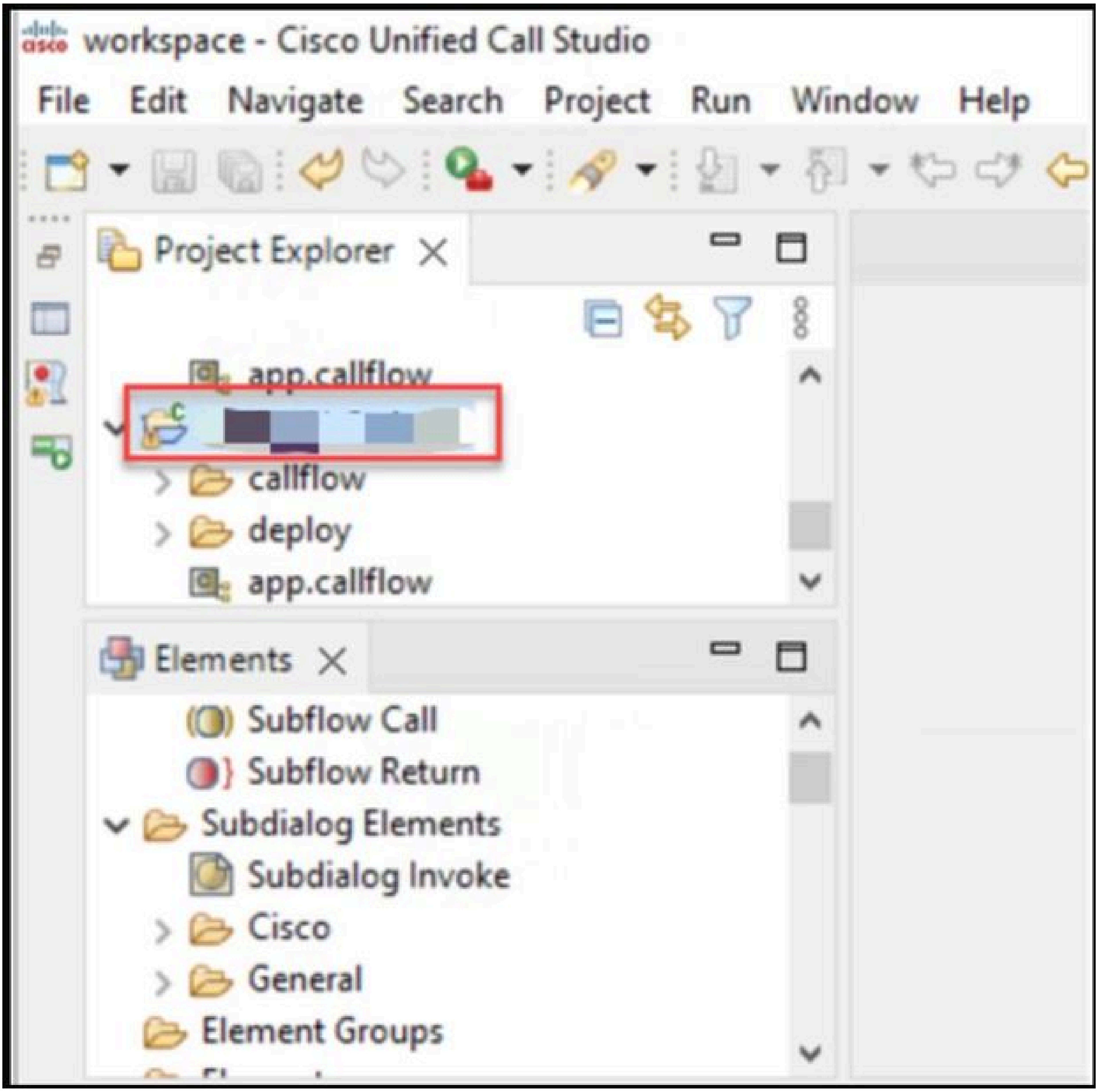
In this example, the Call Studio Application has been developed and it serves as a sample for the implementation in your Call Studio Application.

Step 1. On **CVP**, click the **Call Studio** icon on the desktop to launch the application.



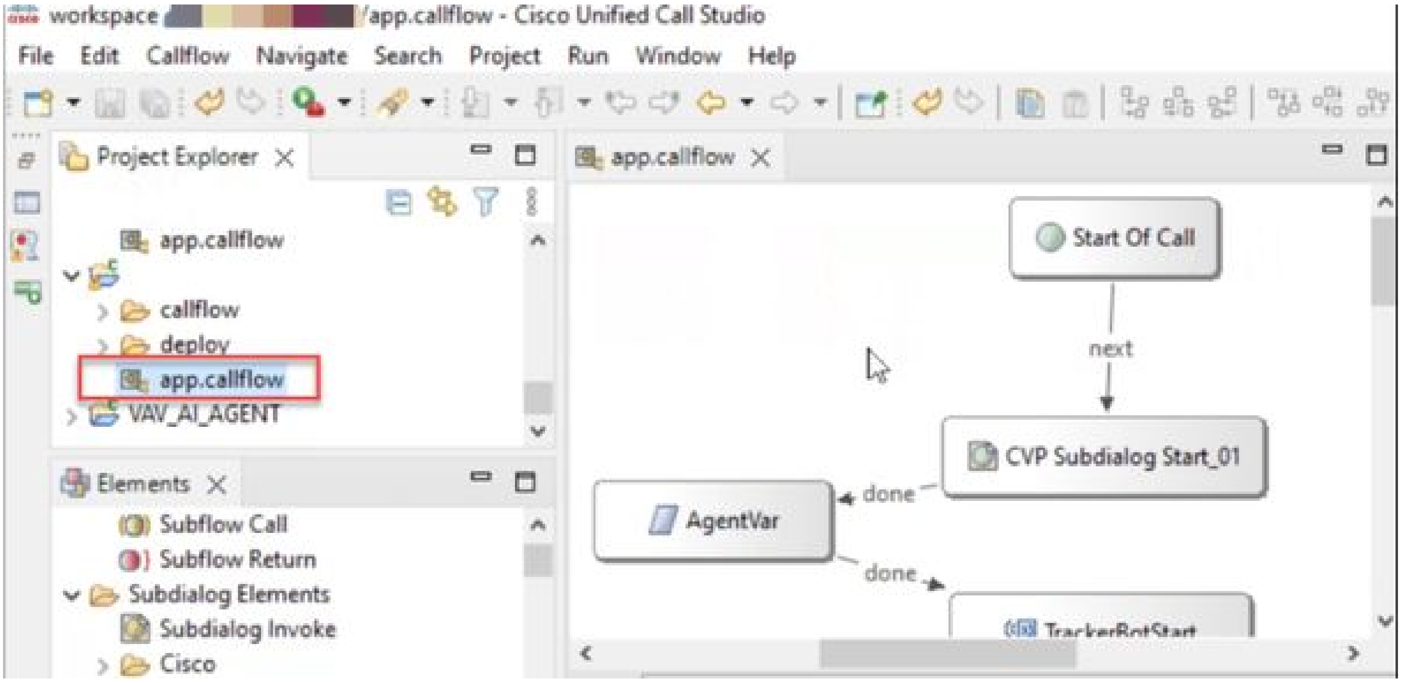
CVP

Step 2. In **CVP Call Studio**, scroll down and expand the **Sample** application project.



Application

Step 3. Within the **Sample Application** folder, click on **app.callflow** to open the call flow.



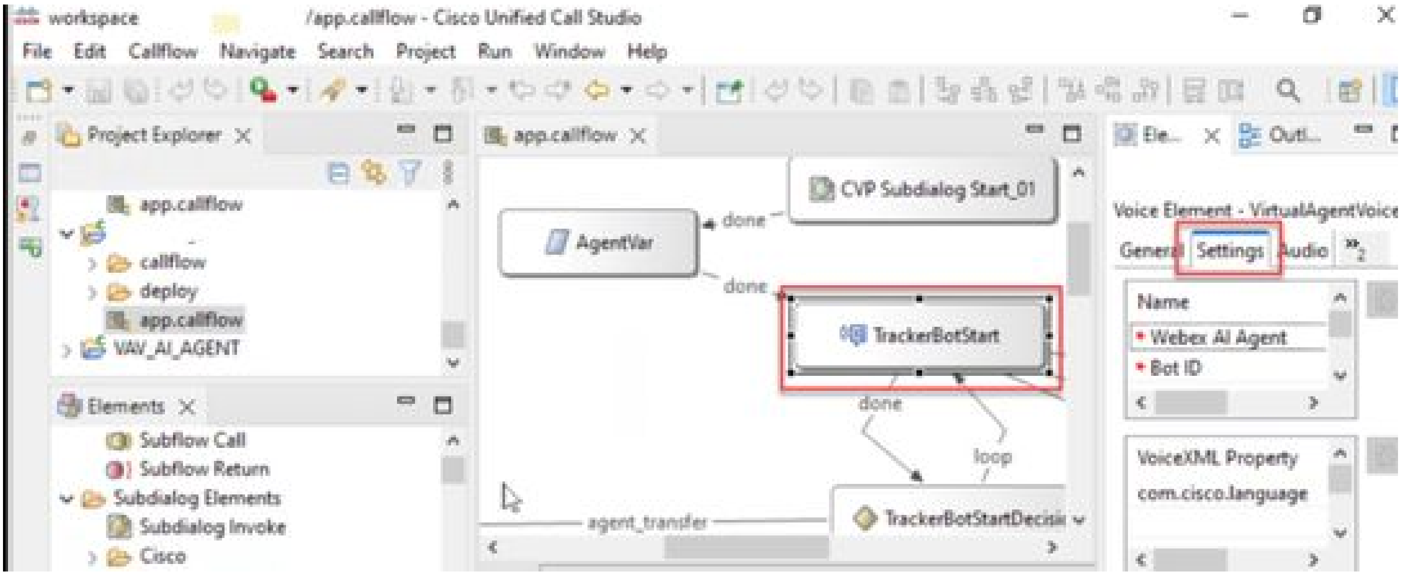
Step 4. You now see the call flow for this application. It is a simple flow designed to demonstrate AI Agent interaction. The application begins the conversation with the virtual agent using the **VAV** (Virtual Agent Voice) element, labeled

TrackerBotStart.

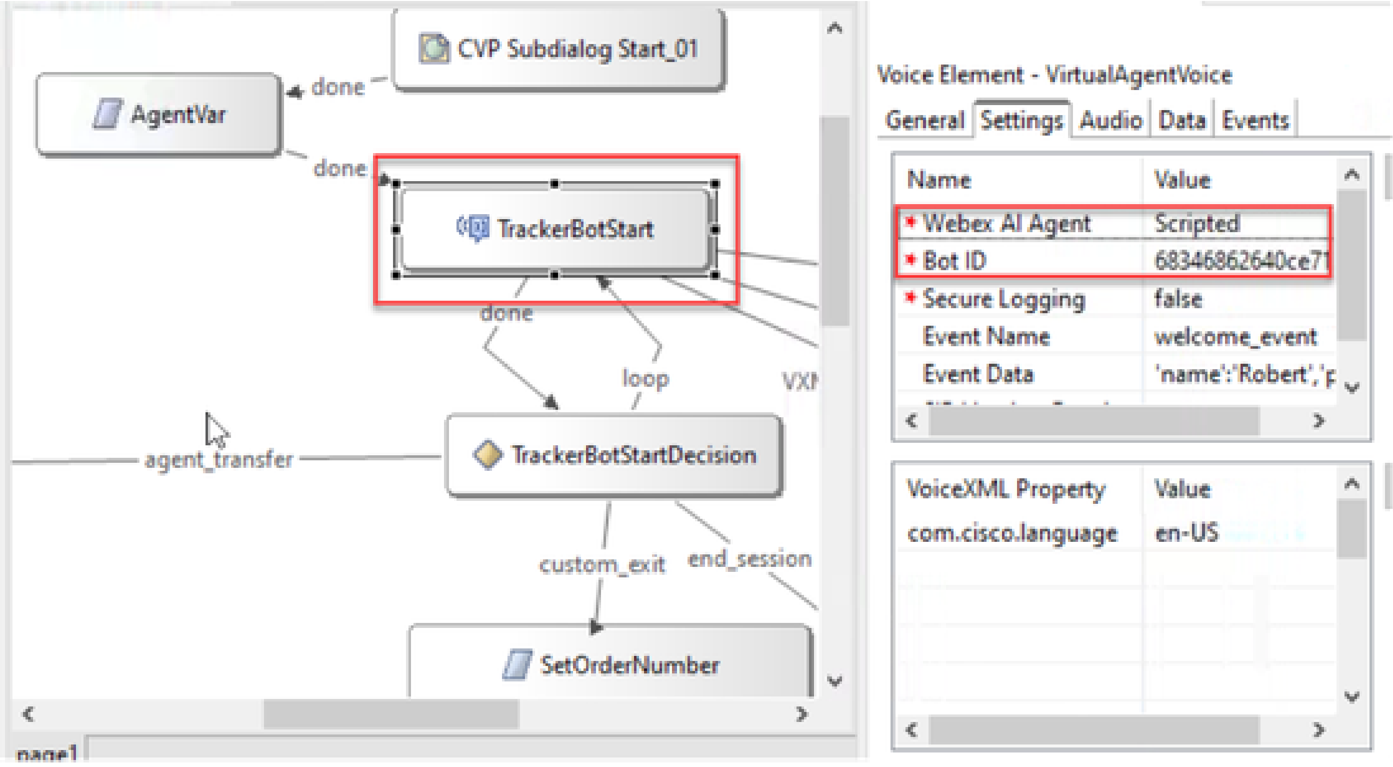
After the virtual agent finishes the conversation:

- The application either routes the call to a live agent, or
- Continue the conversation with the virtual agent until it is complete, depending on the result.

Click the **Virtual Agent Voice** element (**TrackerBotStart**) and go to the **Settings** tab to open its configuration.



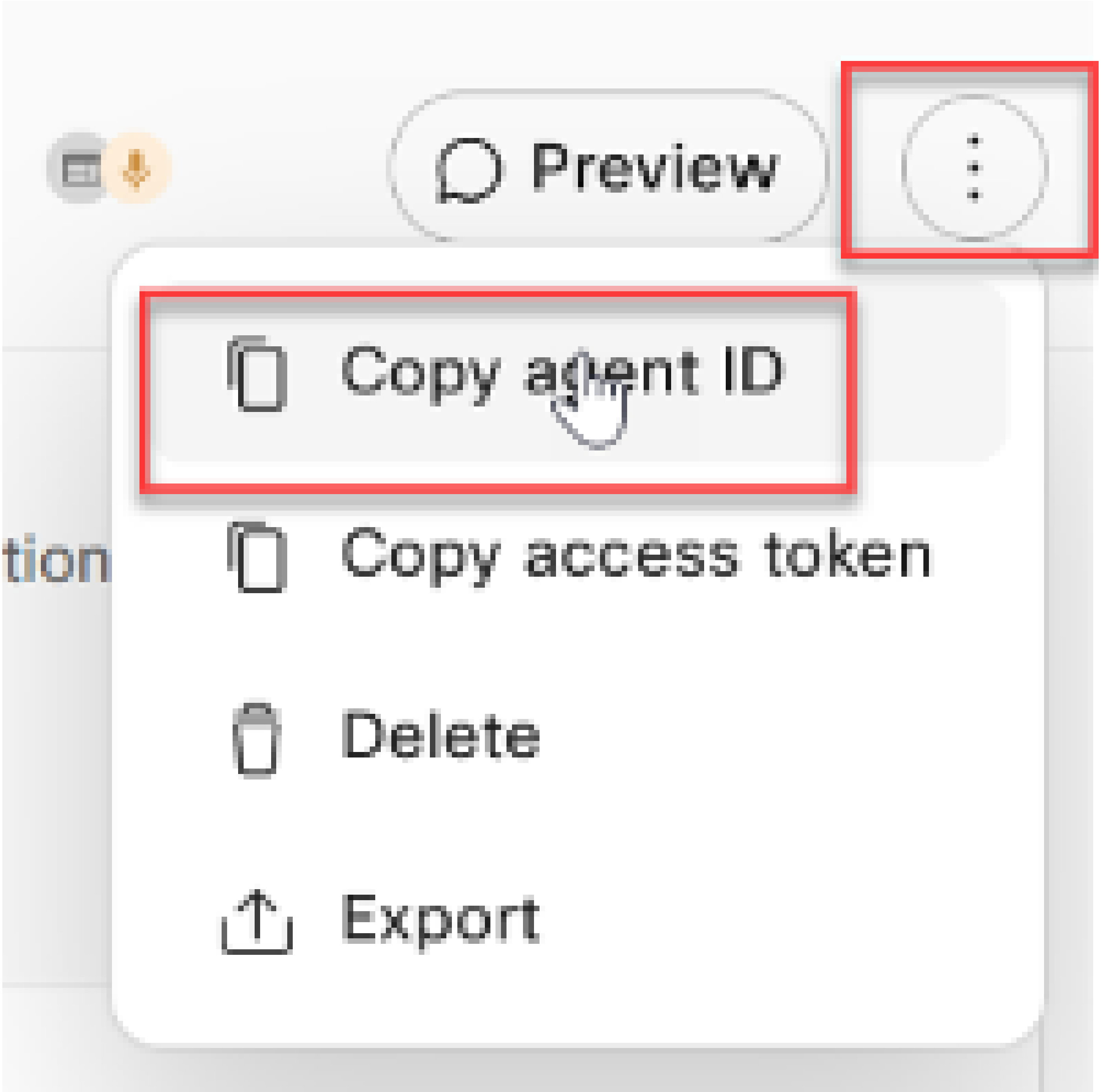
Step 5. Add the AI agent information. For the Webex AI Agent type, choose **Scripted**. Once selected the Bot ID field appears.



Name	Value
* Webex AI Agent	Scripted
* Bot ID	68346862640ce71
* Secure Logging	false
Event Name	welcome_event
Event Data	'name':'Robert','p

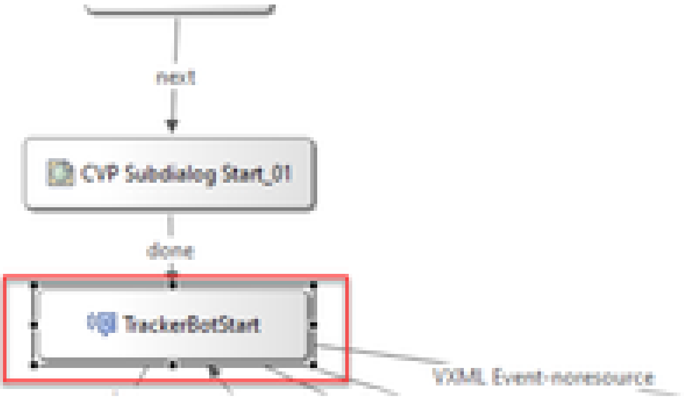
VoiceXML Property	Value
com.cisco.language	en-US

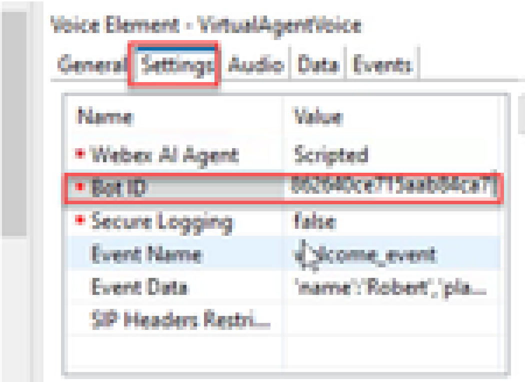
Copy the Bot ID from the AI Agent Studio and paste it into the Bot ID field.



Copy agent id

This is the id that you need to copy and paste: **68346862640ce715aab84ca7**





Voice Element - VirtualAgentVoice

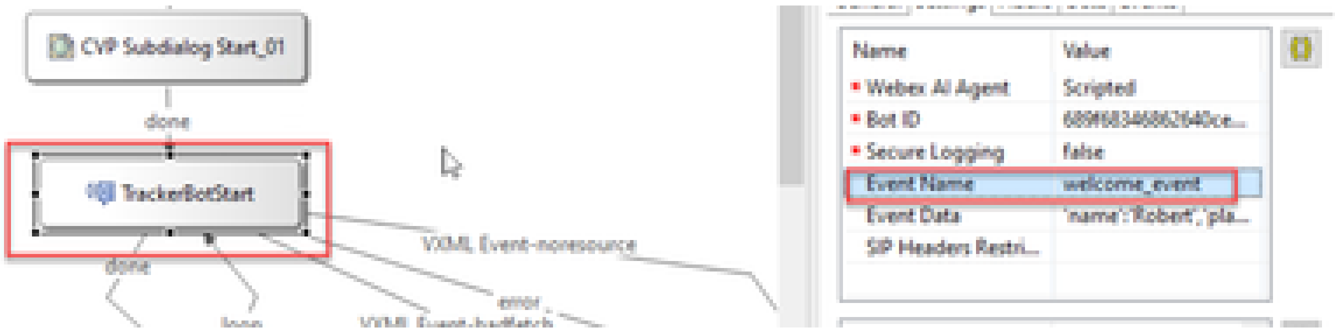
General | Settings | Audio | Data | Events

Name	Value
Webex AI Agent	Scripted
Bot ID	862640ce715aab84ca7
Secure Logging	false
Event Name	welcome_event
Event Data	'name':'Robert','pla...
SIP Headers Restri...	

Call Studio Screenshot

Secure Logging: This setting controls whether potentially sensitive data from the element is logged. If enabled (set to true), the element output—such as query text, fulfillment text, and JSON—from the AI Agent is masked. For now, leave **Secure Logging** set to false.

You are required to pass an event to the AI Agent. Make sure to set the **Event Name** to: **welcome_event**



Name	Value
* Webex AI Agent	Scripted
* Bot ID	689f68346862640ce...
* Secure Logging	false
Event Name	welcome_event
Event Data	'name':'Robert','pla...
SIP Headers Restri...	

Call Studio Screenshot

You can send information to the AI Agent by adding variables and values in the Event Data Field. Please enter the mentioned variables and their corresponding values there.

Name: **name**

Value: ***your name***

Name: **place**

Value: ***your location***

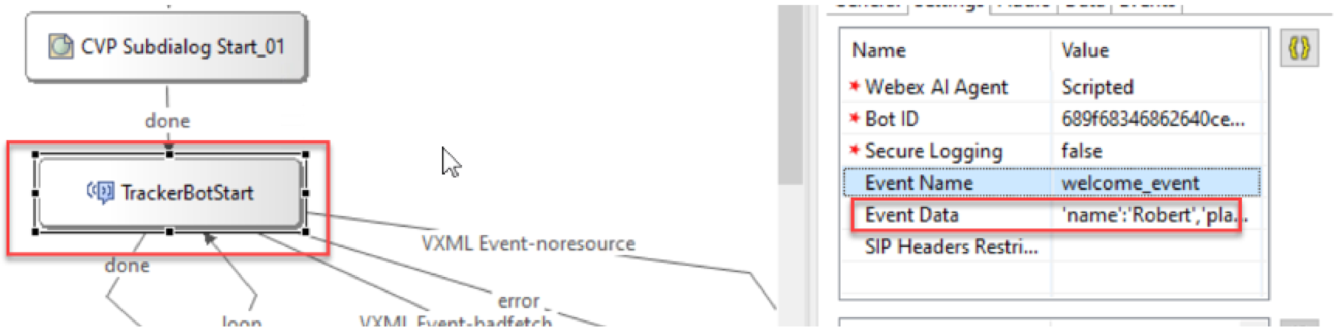
In this example,

Name: **name**

Value: ***Marco***

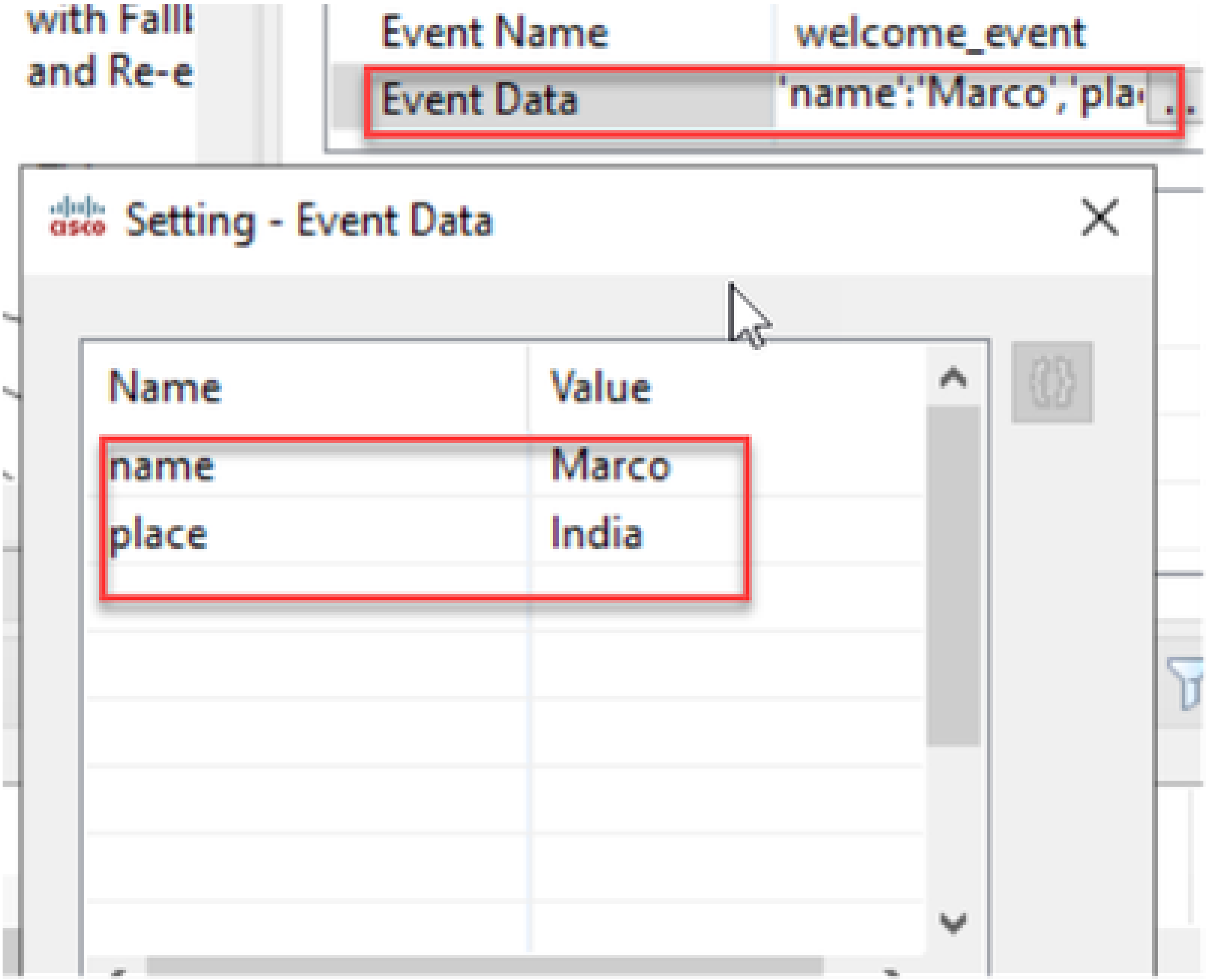
Name: **place**

Value: ***India***



Name	Value
* Webex AI Agent	Scripted
* Bot ID	689f68346862640ce...
* Secure Logging	false
Event Name	welcome_event
Event Data	'name':'Robert','pla...
SIP Headers Restri...	

Call Studio Screenshot



with Fallt
and Re-e

Event Name	welcome_event
Event Data	'name':'Marco','pla

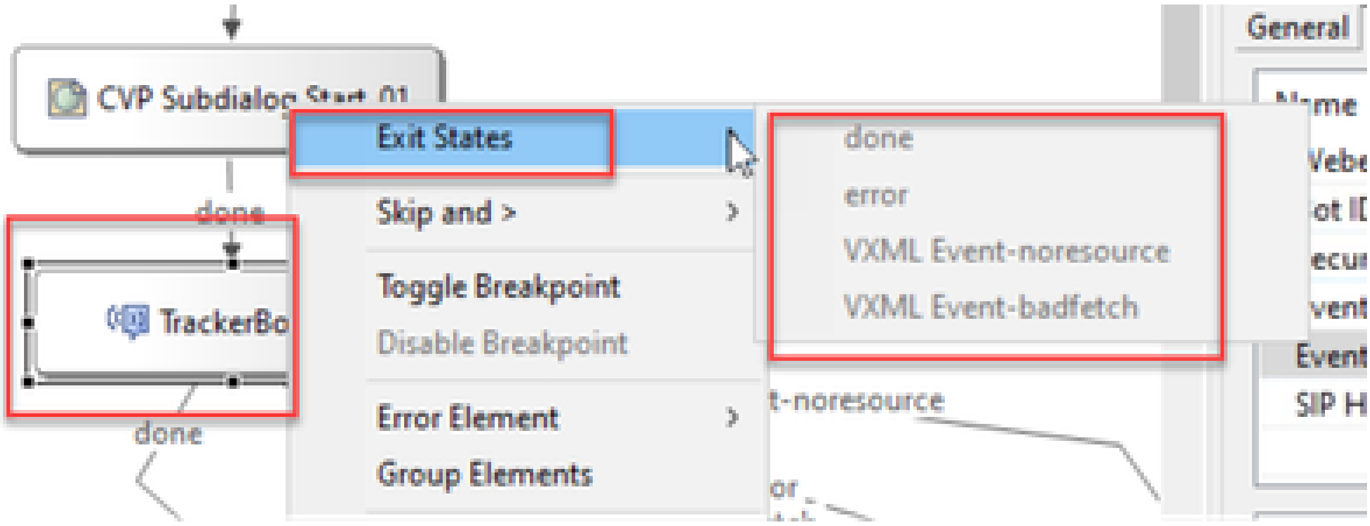
Setting - Event Data

Name	Value
name	Marco
place	India

Call Studio Screenshot

Step 5. Now, we review how to gracefully handle the CVP call flow when resources are unavailable or a gRPC error occurs during integration with the AI Agent.

Right-click the VAV element ((**TrackerBotStart**)) and check the **Exit States**.

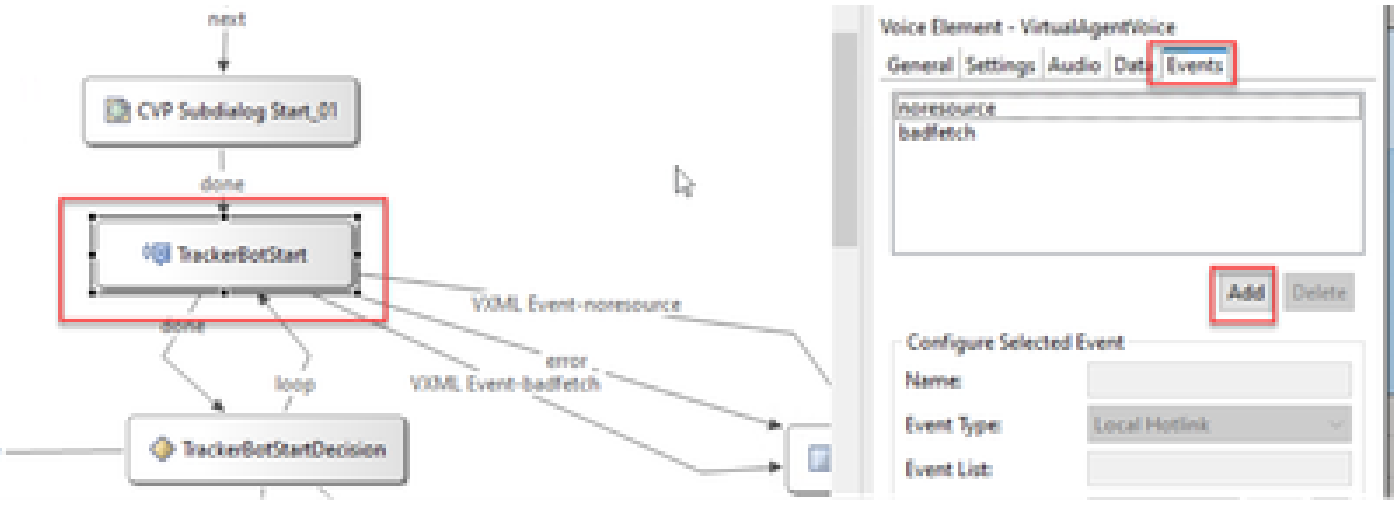


Call Studio Screenshot

By default, the error, VXML Event-noresource and VXML Event-badfetch are part of the element. When any of these errors occur, the application is designed to **fall back** by playing an audio message and then **transferring the call to a live agent**.

In this Call Studio application, notice that an audio prompt is automatically played after one of these errors is triggered, ensuring a smooth and informative experience for the caller.

There is no need to add additional events unless you want to handle a specific case. If you do wish to add a new VXML event, click on the VAV element and on the Event tabs click ADD.

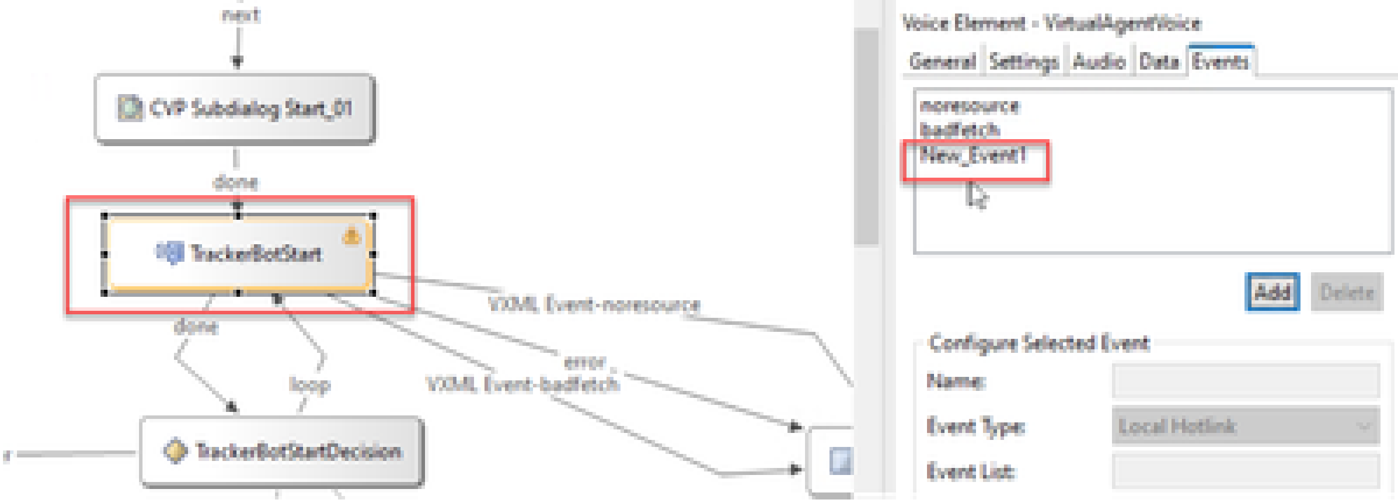


noresource
badfetch

Name:
Event Type: Local Hotlink
Event List:

Call Studio Screenshot

A new event with the name New_Event1 is added. Select it to modify the name and configure the event settings.

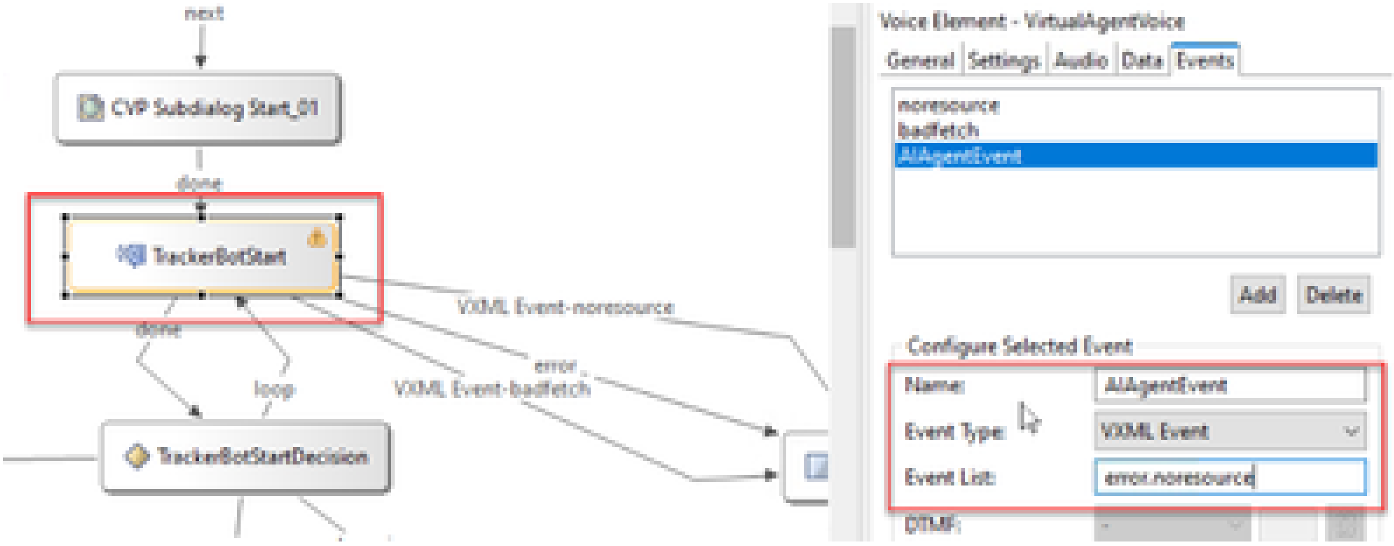


noresource
badfetch
New_Event1

Name:
Event Type: Local Hotlink
Event List:

Call Studio Screenshot

Rename it to **AIAgentEvent**. Leave the **Event Type** set to **VXML Event**. In the **Event List**, you can choose **error.noresource**.



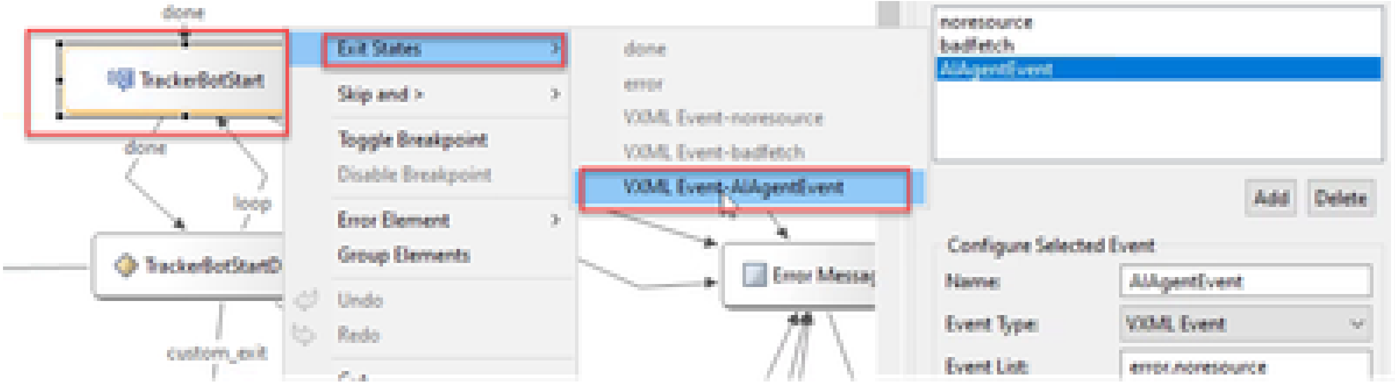
Name: AIAgentEvent

Event Type: VXML Event

Event List: error.noresource

Call Studio Screenshot

Now, direct the new exit state to the audio element that handles other errors, such as the **Error Message** element. To do this, right-click the **VAV element**, select **Exit States**, choose the new event (**AIAgentEvent**), and drag its arrow to the **Error Message** element.



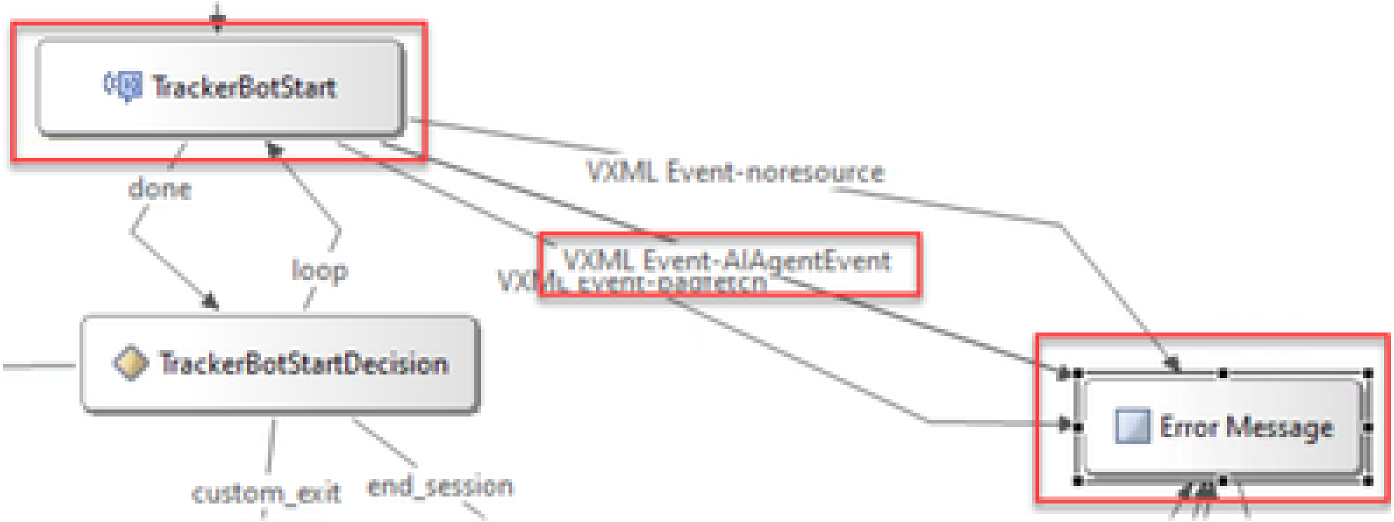
Name: AIAgentEvent

Event Type: VXML Event

Event List: error.noresource

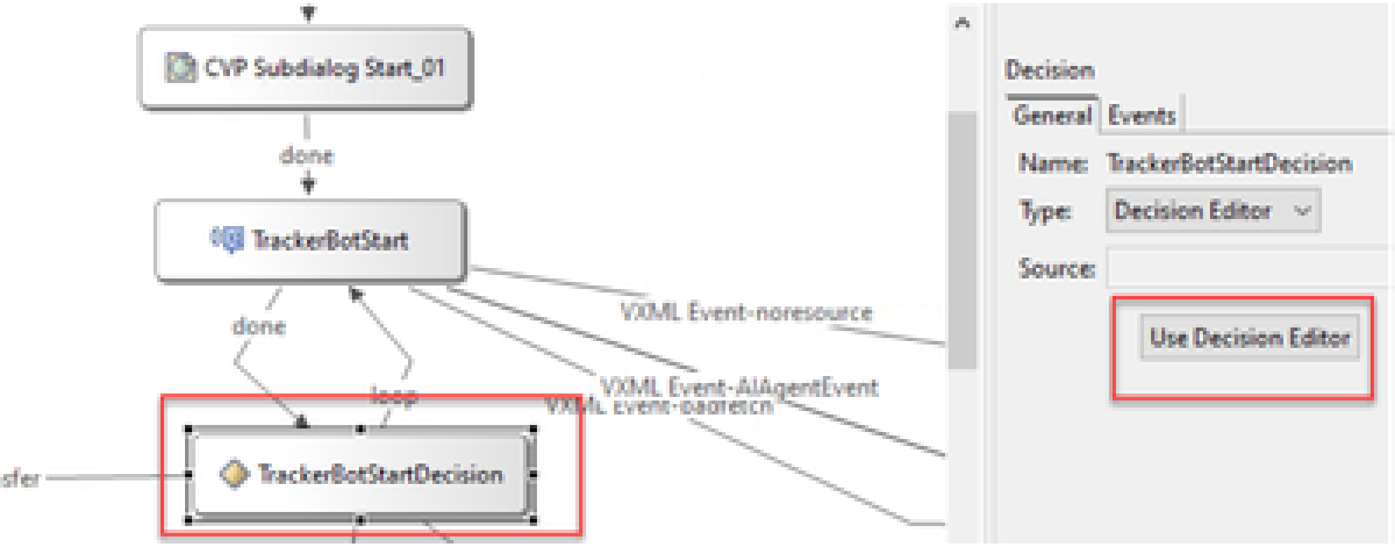
Call Studio Screenshot

At the end you see:



Call Studio Screenshot

Step 6. Now, we review the decision node. Click on the **Decision** node and choose **Use Decision Editor**.

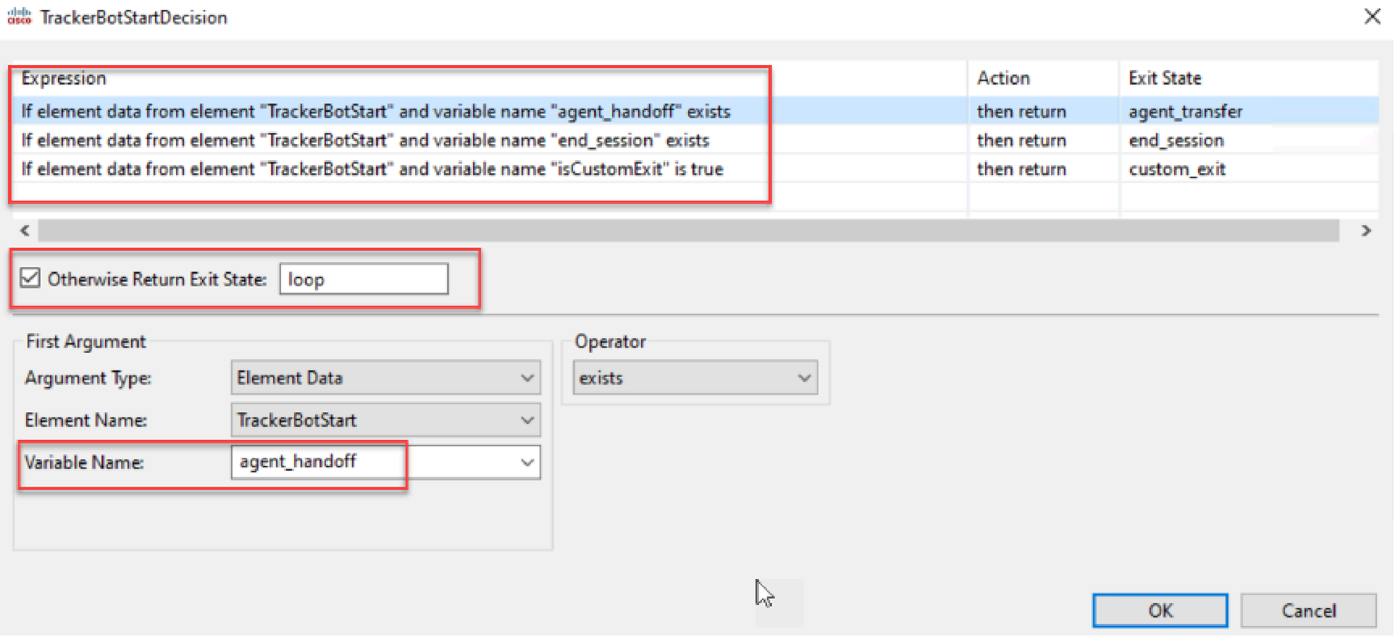


Name: TrackerBotStartDecision

Type: Decision Editor

Call Studio Screenshot

Step 7. You see options to send the call to a real agent, terminate the session, or use a custom exit. These actions rely on three data elements that depend on the result of the AI Agent. Once reviewed, click **OK**.



Expression	Action	Exit State
If element data from element "TrackerBotStart" and variable name "agent_handoff" exists	then return	agent_transfer
If element data from element "TrackerBotStart" and variable name "end_session" exists	then return	end_session
If element data from element "TrackerBotStart" and variable name "isCustomExit" is true	then return	custom_exit

☑ Otherwise Return Exit State: loop

First Argument

Argument Type: Element Data

Element Name: TrackerBotStart

Variable Name: agent_handoff

Operator

exists

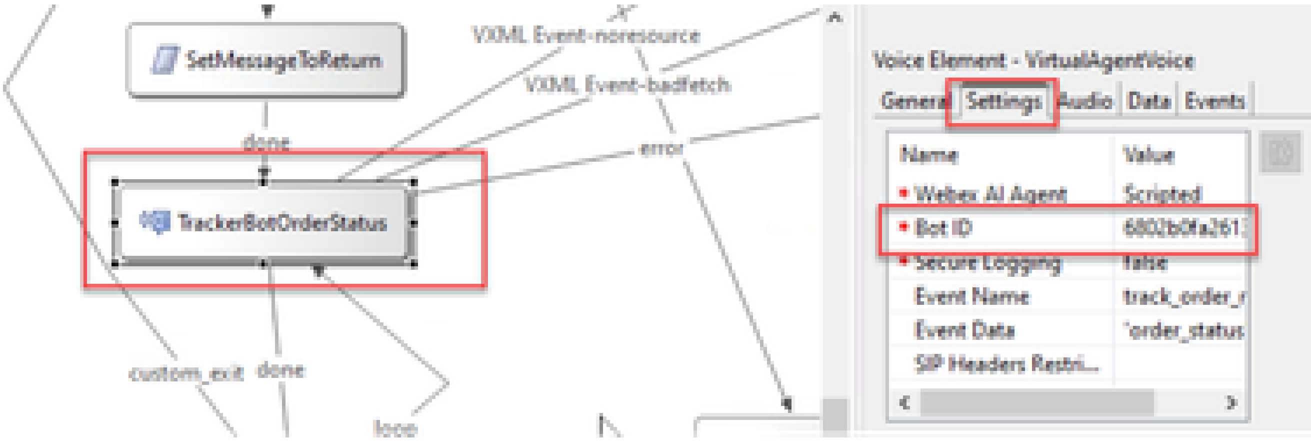
Bot start decision

Step 8. After the interaction with the AI Agent ends, this Call Studio application evaluates the results. Based on that evaluation, the conversation with the AI Agent can continue using a second VAV element.

If you scroll down in the Call Studio application, you find the second VAV element named **TrackerBotOrderStatus**.

Add the **same Bot ID** to this element and keep the existing **Event Name** and **Event Data** as they are already configured.

This is the ID that you need to copy and paste: **68346862640ce715aab84ca7**



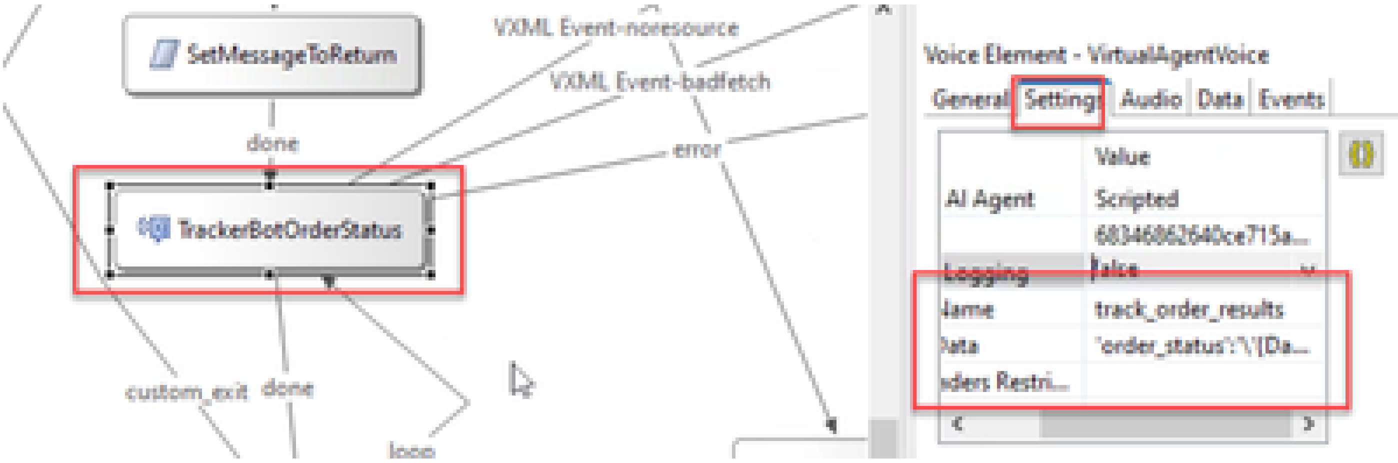
Voice Element - VirtualAgentVoice

General | Settings | Audio | Data | Events

Name	Value
* Webex AI Agent	Scripted
* Bot ID	6802b0fa261
* Secure Logging	false
Event Name	track_order_r
Event Data	'order_status
SIP Headers Restri...	

Call Studio Screenshot

Keep the **Event Data** and **Event Name** unchanged, leaving them as already configured.



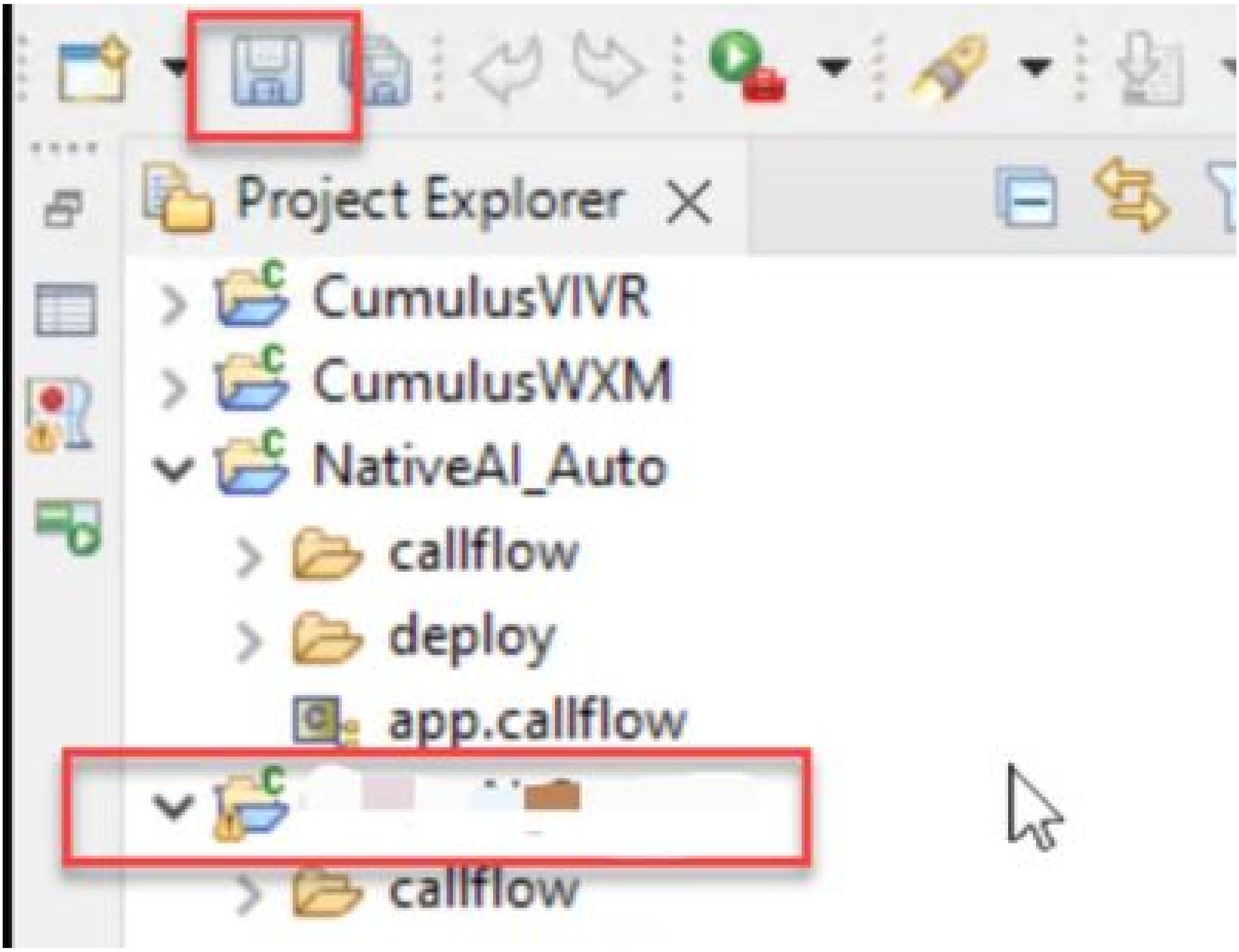
Voice Element - VirtualAgentVoice

General | Settings | Audio | Data | Events

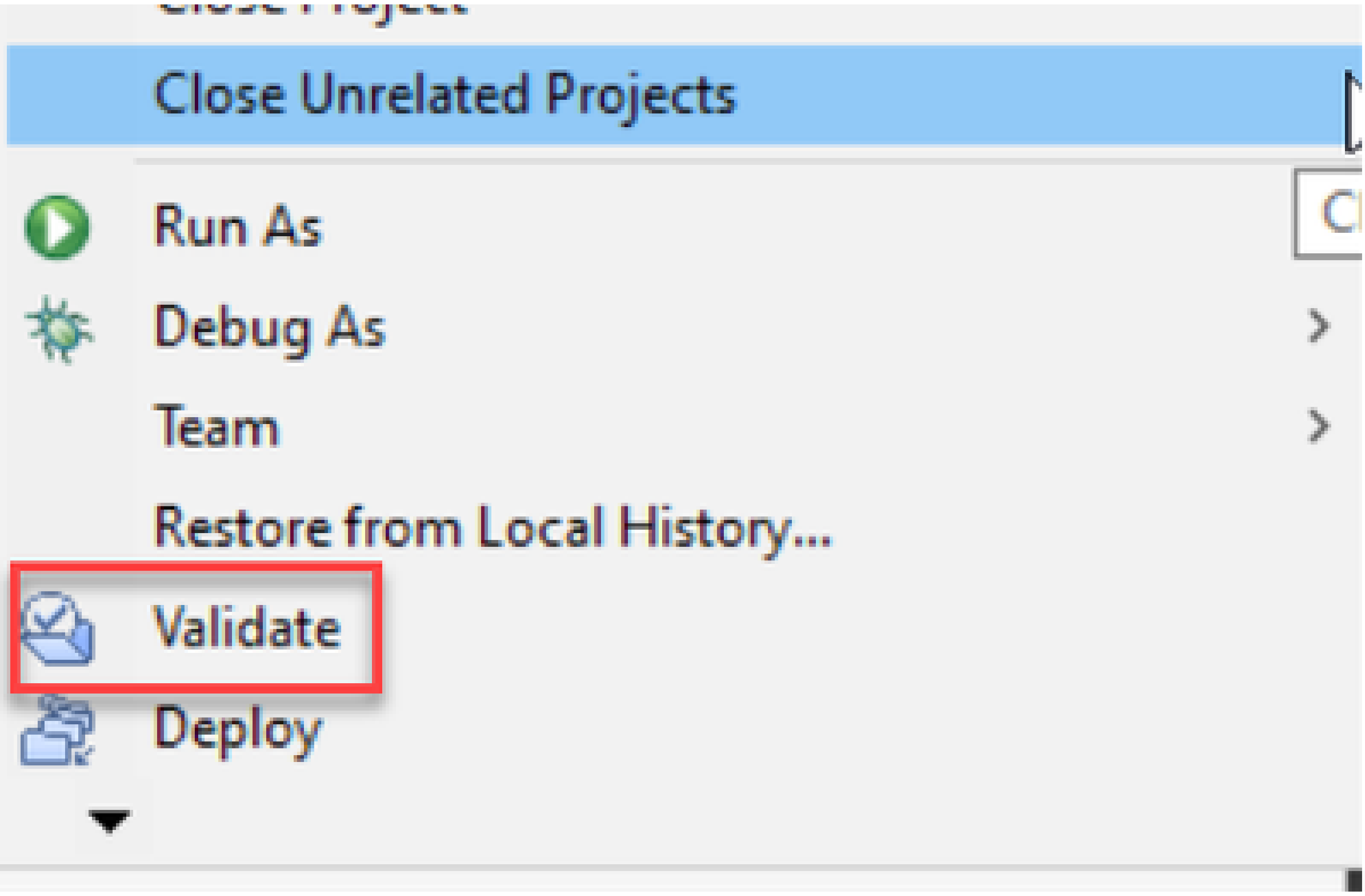
	Value
AI Agent	Scripted
	68346862640ce715a...
Logging	false
Name	track_order_results
Data	'order_status':'\'{Da...
ders Restri...	

Call Studio Screenshot

Step 9. You are now ready to save and deploy the application. Click the **save** icon to save your changes.

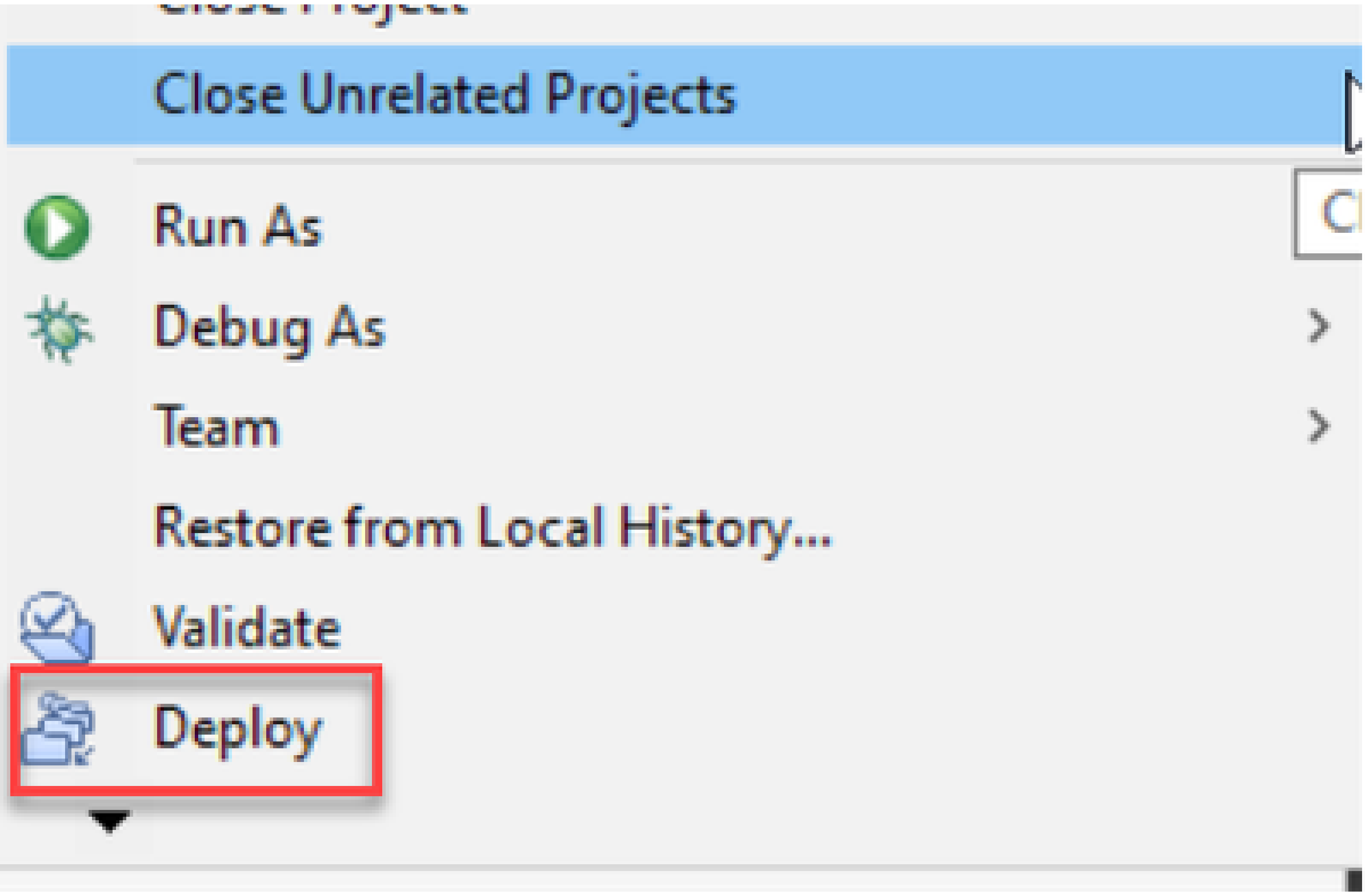


Step 10. Validate the application. To validate the application, right-click on the **Sample application** and select **Validate**.



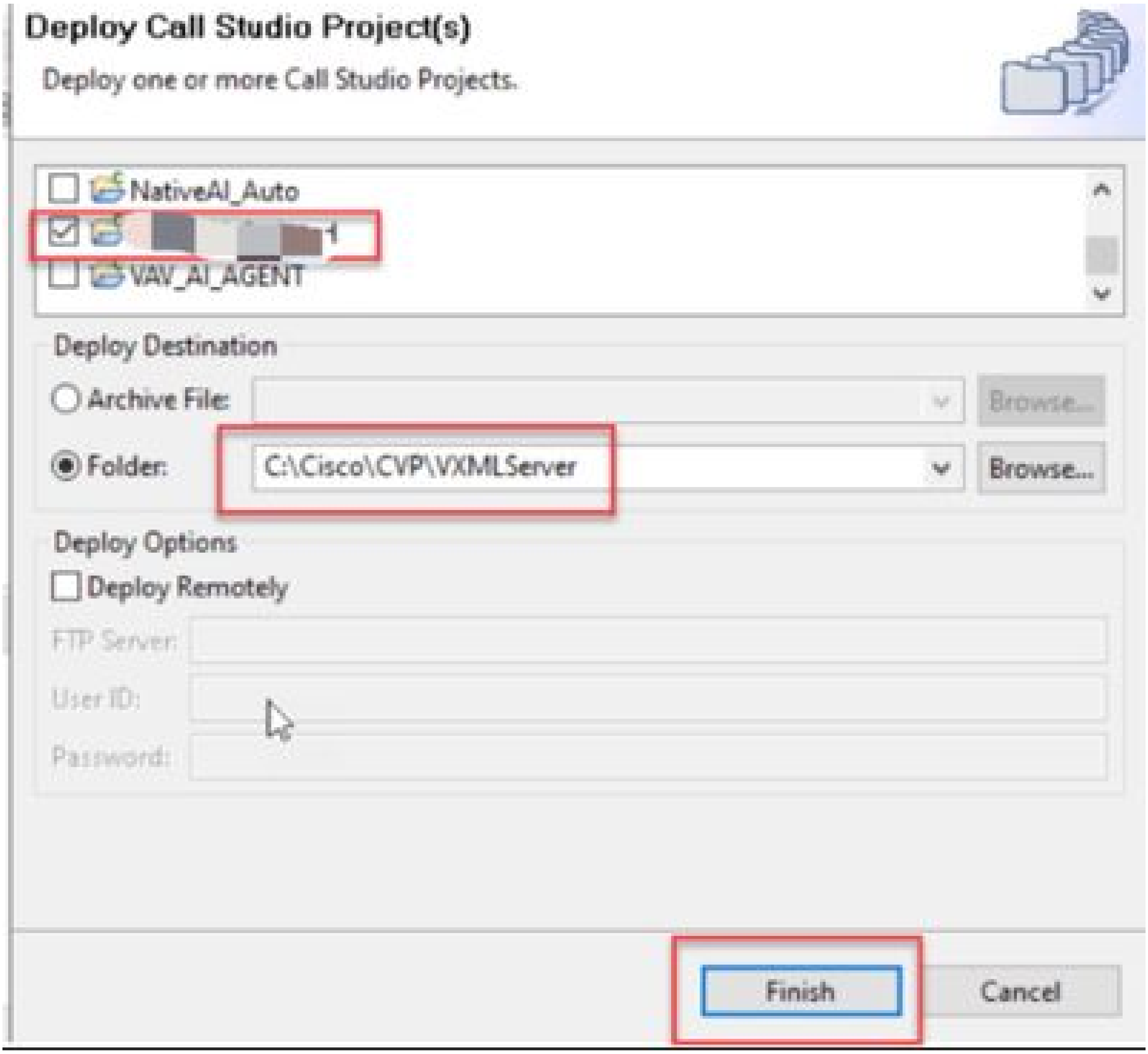
Call Studio Screenshot

Step 11. Deploy the application. To deploy the application on the VXML Server, right-click on the **Sample Application** and select **Deploy**.



Call Studio Screenshot

Step 12. In the next window, keep the default settings with the **Sample Application** checked and the folder set to **C:\Cisco\CVP\VXMLServer**, then click **Finish**.



Deploy Call Studio Project(s)

Deploy one or more Call Studio Projects.

□ NativeAI_Auto

☑ [redacted]

□ VAV_AI_AGENT

Deploy Destination

○ Archive File:

● Folder: C:\Cisco\CVP\VXMLServer

Deploy Options

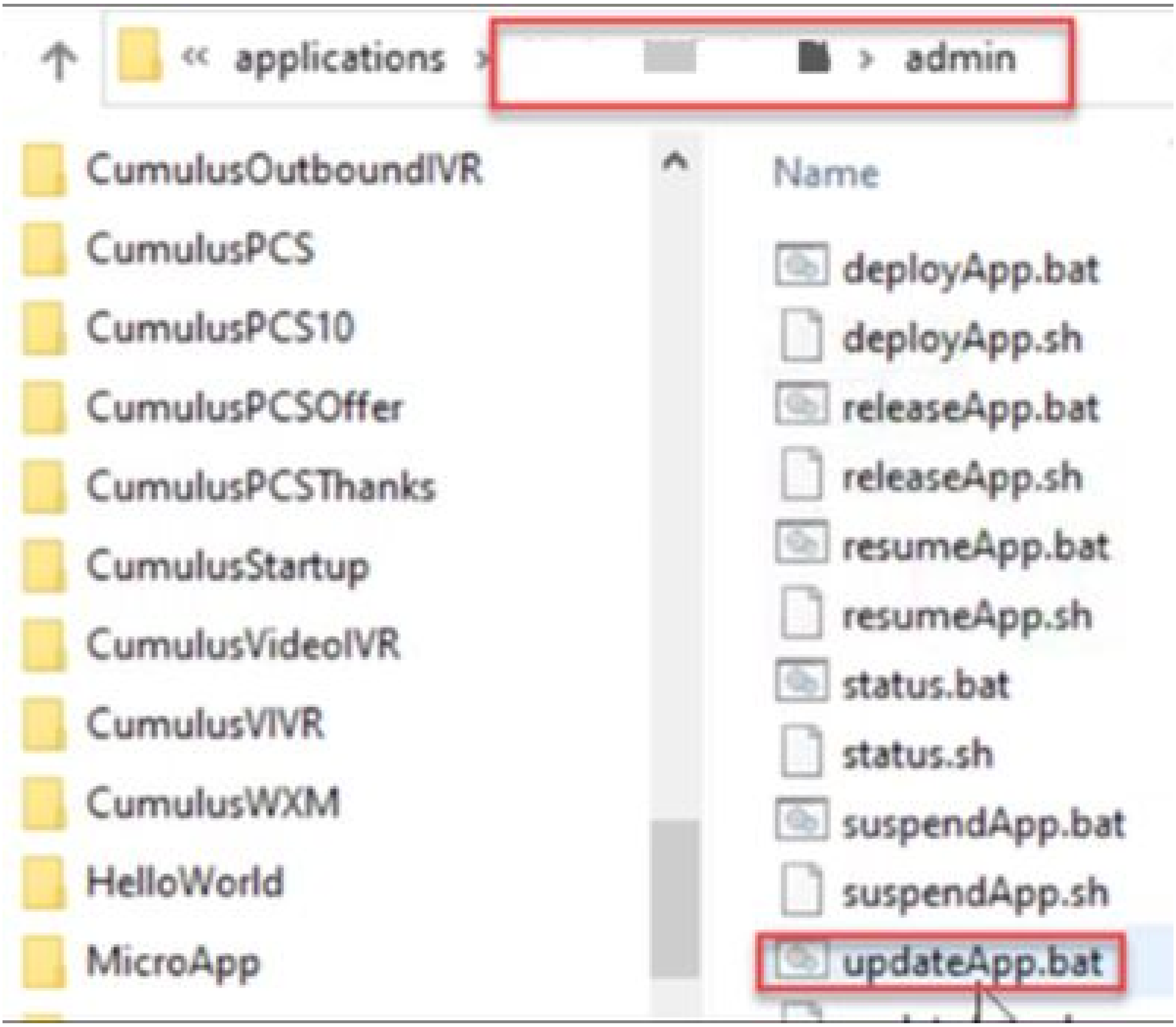
□ Deploy Remotely

FTP Server:

User ID:

Password:

Step 13. On the VXML server update the application. Navigate to **C:\Cisco\CVP\VXMLServer\applications\NativeAI_Scripted\admin,** then, run **updateApp.bat** by double-clicking it or executing it in a command prompt.



Step 14. In the new command window that opens, type **yes** to confirm that you want to update the application.