Cisco Webex AI Agent for CCEの設定

内容

はじめに

前提条件

- 要件
- 使用するコンポーネント

Webex AIエージェントの概要

- 背景
- Webex Agent AIアーキテクチャ
- Webex Agent AIコンポーネント
- オンプレミスコンポーネント

CCEでのスクリプト化された自律型AIエージェントの設定

Webexサービスの設定：Control Hub

- Cloud Connectの登録
- CCEの設定
- Cloud Connectの登録
- Cloud Connectプロキシの設定
- クラウド機能管理の状態
- CVP Call Studioアプリケーションの設定 – スクリプトAIエージェント

はじめに

このドキュメントでは、Contact Center Enterprise(CCE)のWebex AI Agent(WAI)を設定する方法について説明します。

前提条件

要件

次の項目に関する知識があることが推奨されます。

- Unified Contact Center Enterprise(UCCE)リリース15.0
- Customer Voice Portal(CVP)リリース15.0
- Customer Voice Portal(CVP)コールスタジオ
- Cisco Virtualized Voice Browser(VVB)

使用するコンポーネント

このドキュメントの情報は、次のソフトウェアのバージョンに基づいています。

- UCCE 15.0

- Webex CCE（日本での対応時期未定）
- CVP 15.0
- Cisco Virtualized Voice Browser(VVB)
- Cloud Connect

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、クリアな（デフォルト）設定で作業を開始しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

Webex AIエージェントの概要

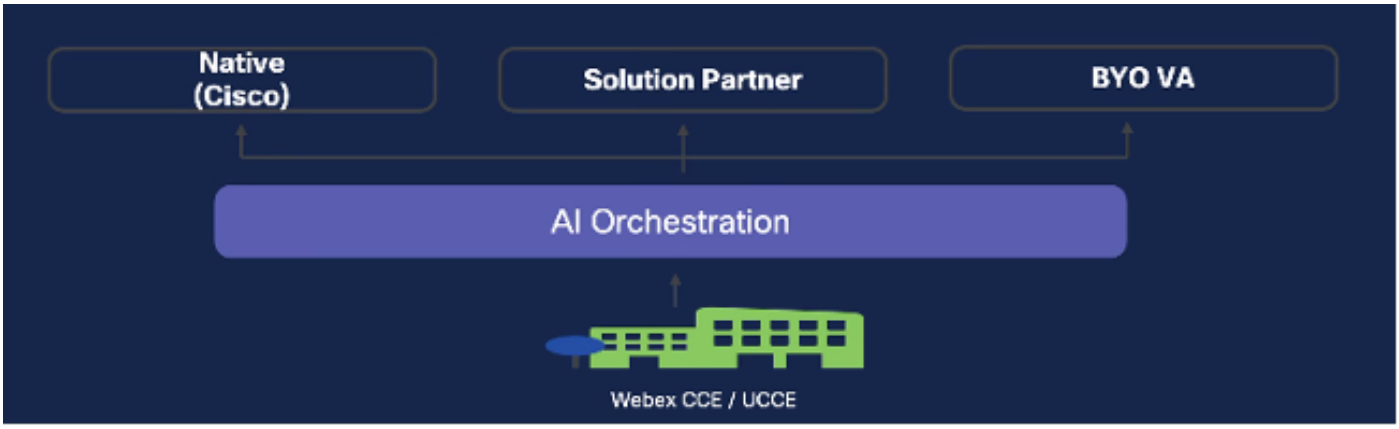
Webex AI Agentは、Cisco Webex CCEまたはCCEに統合された仮想人工知能(AI)アシスタントです。自動インタラクションを使用してサービスとサポートを改善し、ライブエージェントを支援し、分析によって洞察を提供するように設計されています。

背景

AIエージェントは、コンタクトセンターのユーザーと関わり、入力を解釈し、トレーニング方法に基づいて適切な応答を生成し、事前定義されたタスクを実行するように構築されています。このガイドでは、Contact Center Enterprise(CCE)内にAIエージェントを実装するために必要な手順の概要を説明します。

Webex AI Agentは、新しいCCE 15.0リリースの一部として導入されました。

Webex Agent AIアーキテクチャ

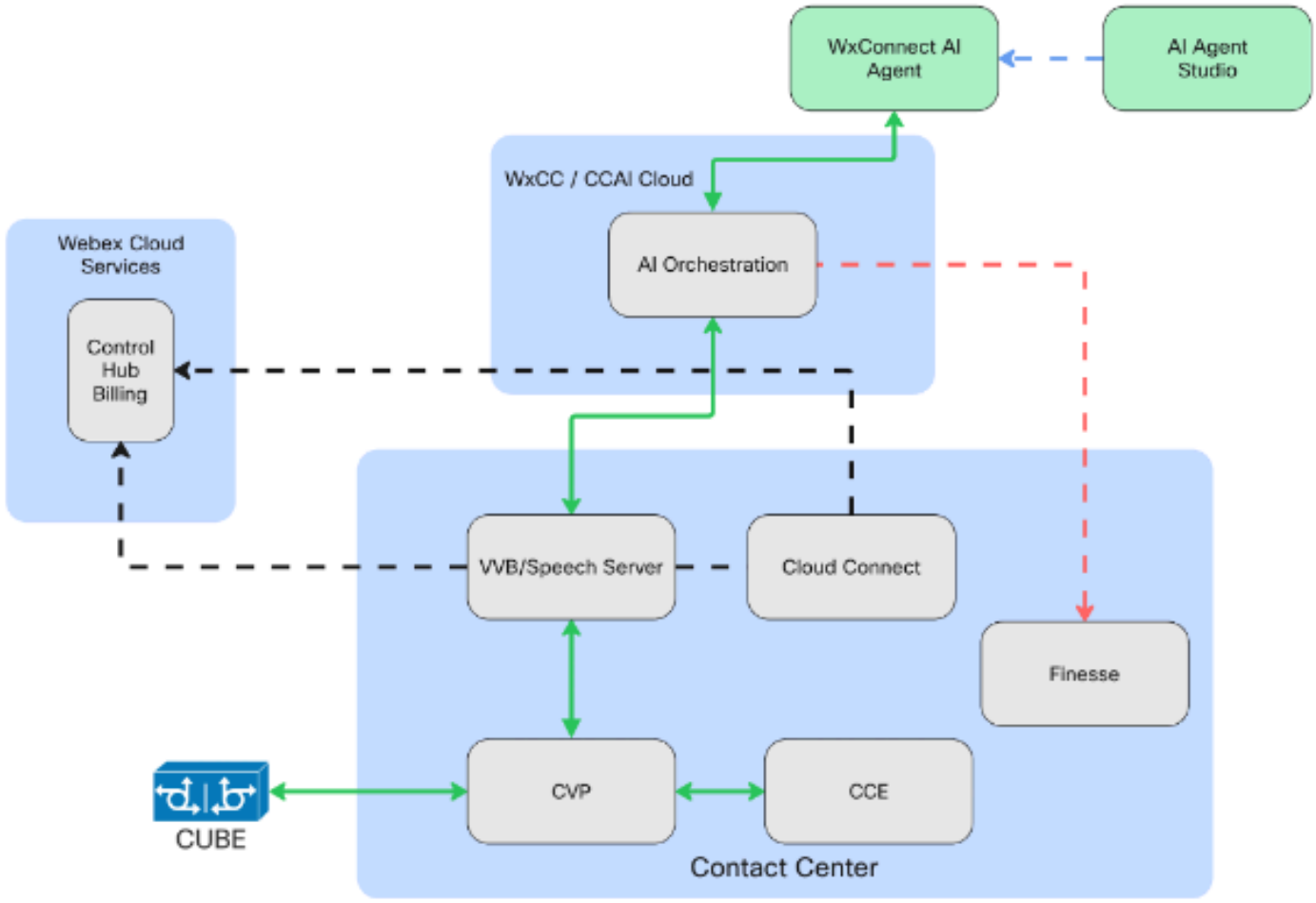


Webex Agent AIアーキテクチャ

ネイティブ：仮想エージェント向けのシスコ社内ソリューション

これは、このラボで設定する仮想エージェントのタイプです。

Webex Agent AIコンポーネント



Webex Agent AIコンポーネント

CCEオンプレミスソリューションとWebex CCE(WxCCE)テナントの両方で、いくつかのコアコンポーネントがWebex AIエージェントソリューションの基盤を形成します。これらのコンポーネントは、オンプレミス環境とクラウド環境の両方にまたがり、連携してインテリジェントなAI主導のインタラクションを実現します。

オンプレミスコンポーネント

- Cisco Customer Voice Portal(CVP):
  CVPは、CCE/WxCCEと統合された自動音声応答(IVR)システムとして機能します。セルフサービスおよびコールルーティングの自動化が可能です。CVPはVVBおよびSpeech Serverと連携して、IVRフロー内にAIエージェントを統合します。

- CVP Call Studio:CVPの音声アプリケーションの設計と構築に使用される開発環境

- Cisco Virtualized Voice Browser(VVB)/Speech Server:
  VVBは、マルチメディアおよびIVR処理を処理します。Speech Serverは、gRPCプロトコルを使用して呼び出し元の要求をAIオーケストレーションサービスに伝達します。

- CCEコア：
  CCEコアには、ペリフェラルゲートウェイ、ルータ、アドミンワークステーション(AW)などの主要コンポーネントが含まれます。 これらは連携して要求を処理し、適切な宛先を決定します。AWは、CCE環境全体の設定と管理に使用されます。

- Cloud Connect:
  Cloud Connectは、オンプレミスのCCEインフラストラクチャとシスコクラウドサービスの間の安全なブリッジとして機能し、シームレスな統合を実現します。

クラウドコンポーネント

- Webex Cloud Services - Control Hub:Webexクラウドソリューションを設定するための中央集中型の管理インターフェイス。Control Hubから、ユーザーはAI Agent Studioをプロビジョニングして起動できます
- AIオーケストレーションサービス：CCEまたはWxCCEシステムからの着信要求を受信し、Ciscoコネクタを介してWebex Connect AI Agentにルーティングするシスコのクラウドベースのオーケストレーションレイヤ
- Webex Connect AI Agent:Webex Connectを搭載したクラウドホスト型AIエージェントで、インタラクションをインテリジェントに処理するように設計されています
- AI Agent Studio:スクリプト化されたAIエージェントのワークフローとスクリプトを設計するために使用されるビジュアルインターフェイスで、カスタマイズされたエンゲージメントエクスペリエンスを実現します。

CCEでのスクリプト化された自律型AIエージェントの設定

スクリプト形式または自律エージェントの作成方法については、このドキュメントを参照してください。

Webexサービスの設定：Control Hub

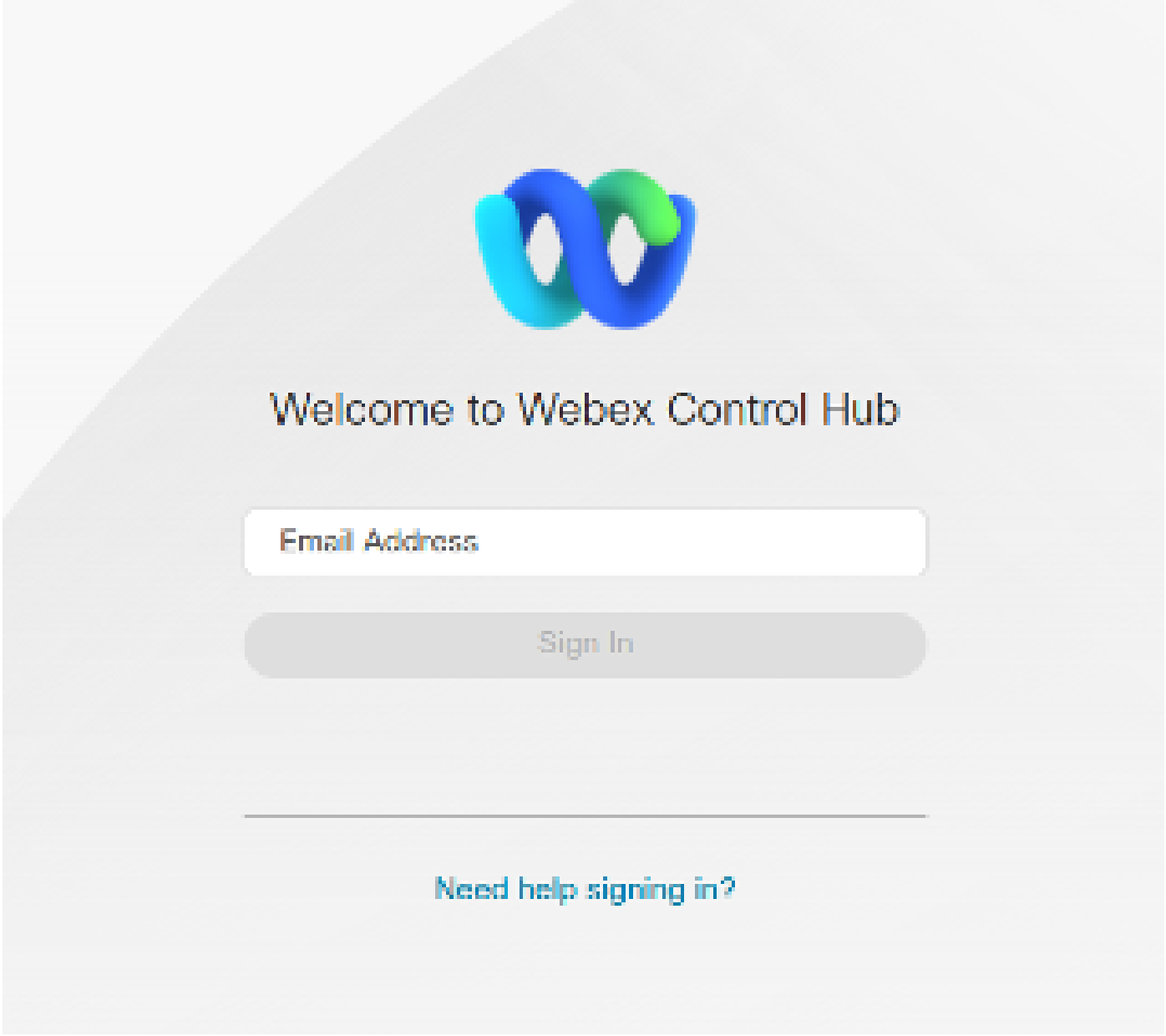
Webex Control Hubは、組織、ユーザ、サービス権限、およびWebex Contact Centerやその他のサービスの設定を管理するための中央インターフェイスとして機能します。1つの顧客組織内ですべてのWebexサービスをプロビジョニングできます。

Control Hub組織内でテストできるWebex AIエージェントの設定作業は次のとおりです。

- Cloud Connectの登録
- AI Agent Builderの起動

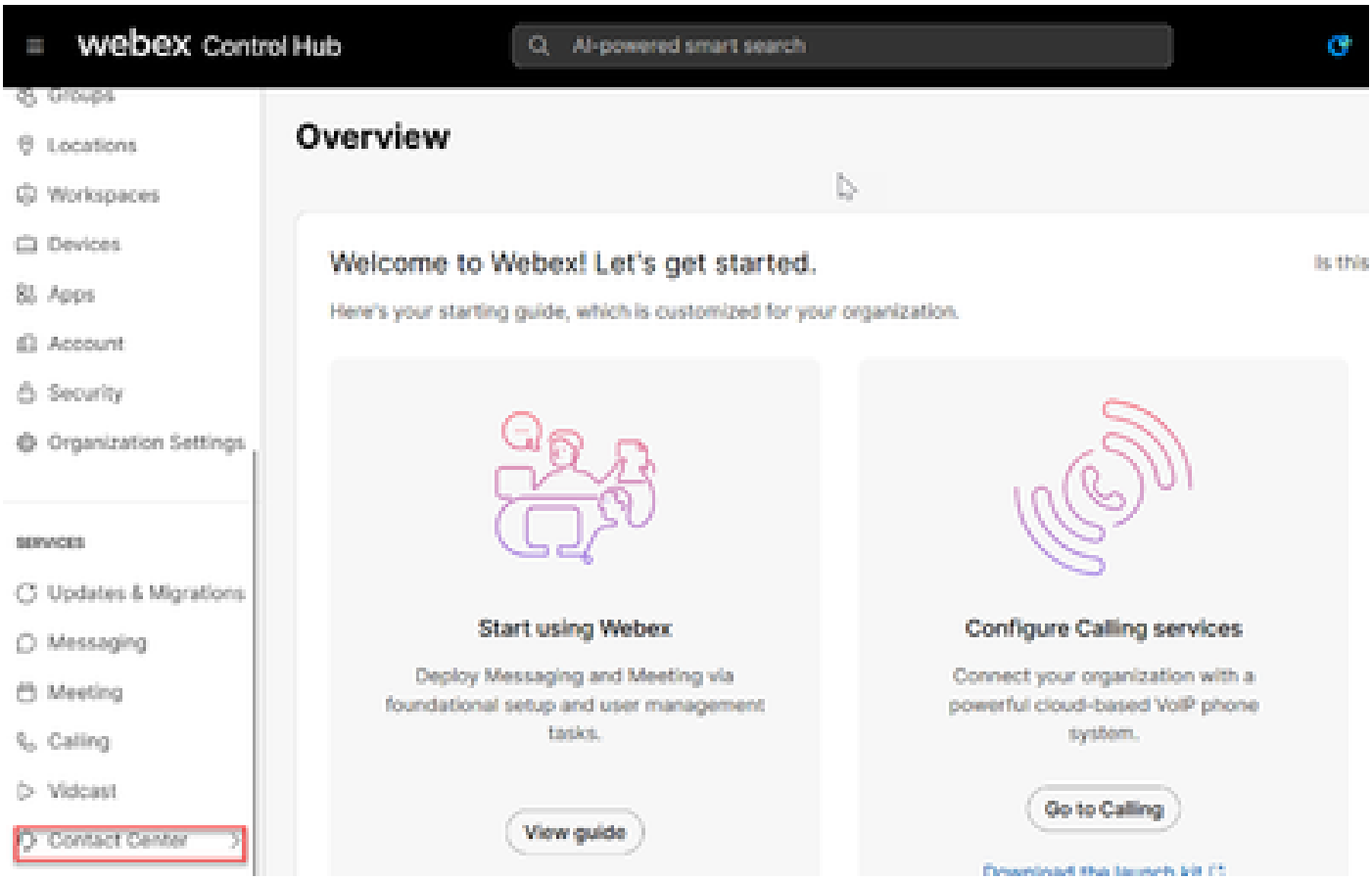
Cloud Connectの登録

ステップ 1：Webブラウザを開き、Control Hub(admin.webex.com)に移動します。



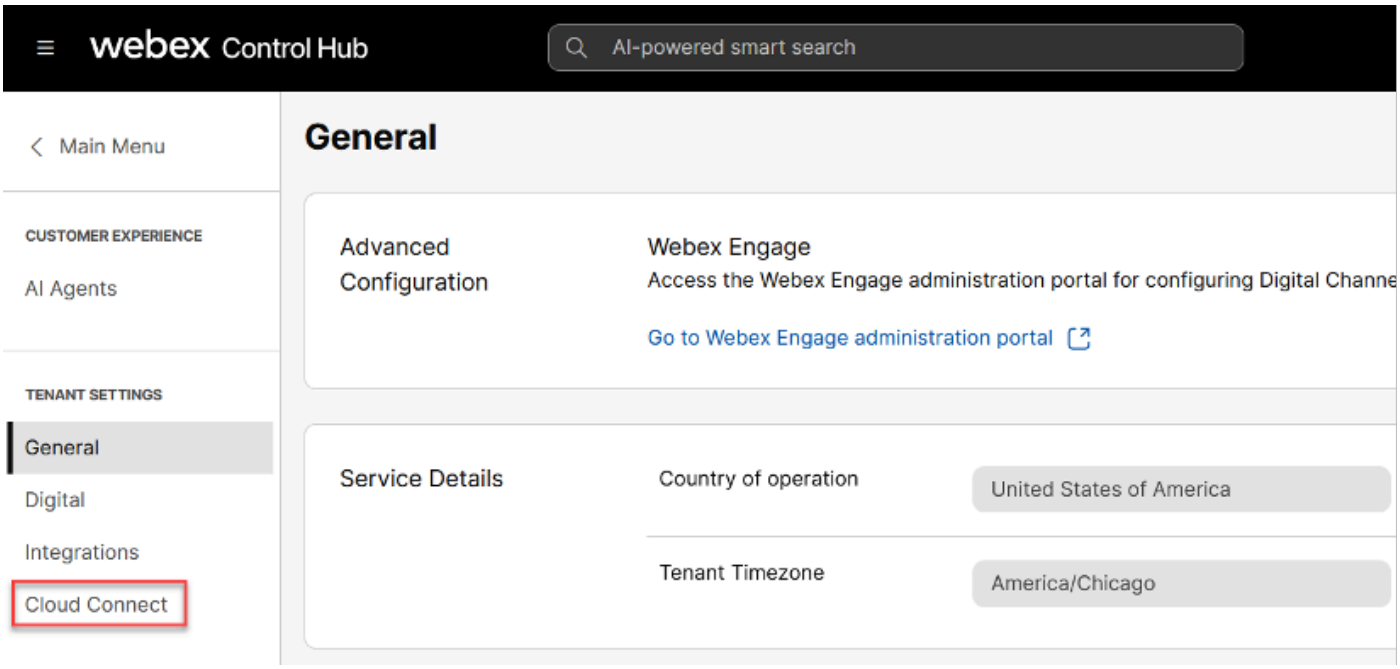
コントロールハブサインイン

ステップ 2：Control Hubで、左側のメニューからContact Centerセクションに移動します。



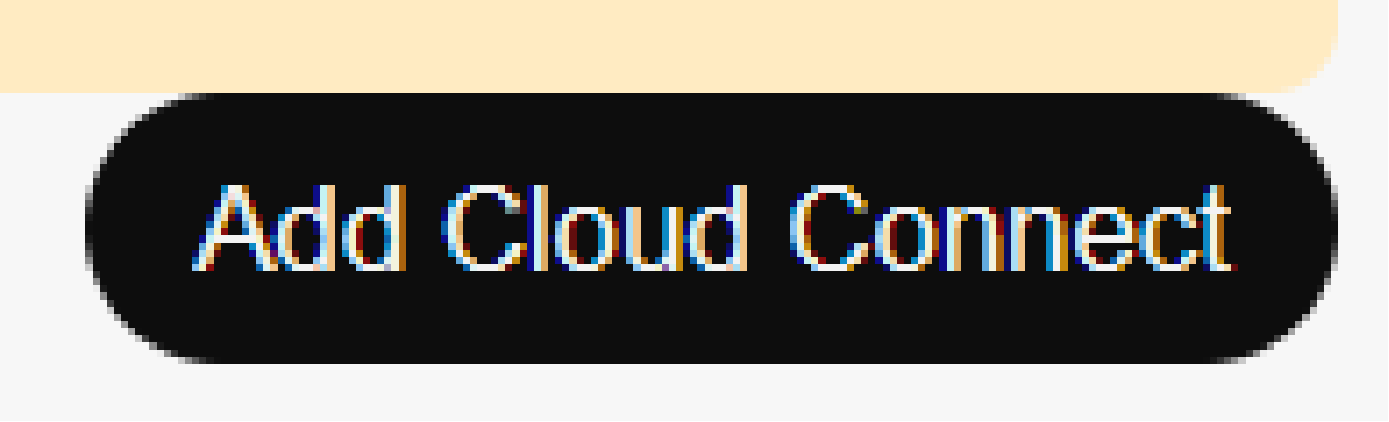
コントロールハブの概要

ステップ 3：Contact Centerページで、Cloud Connectをクリックして登録プロセスを開始します。



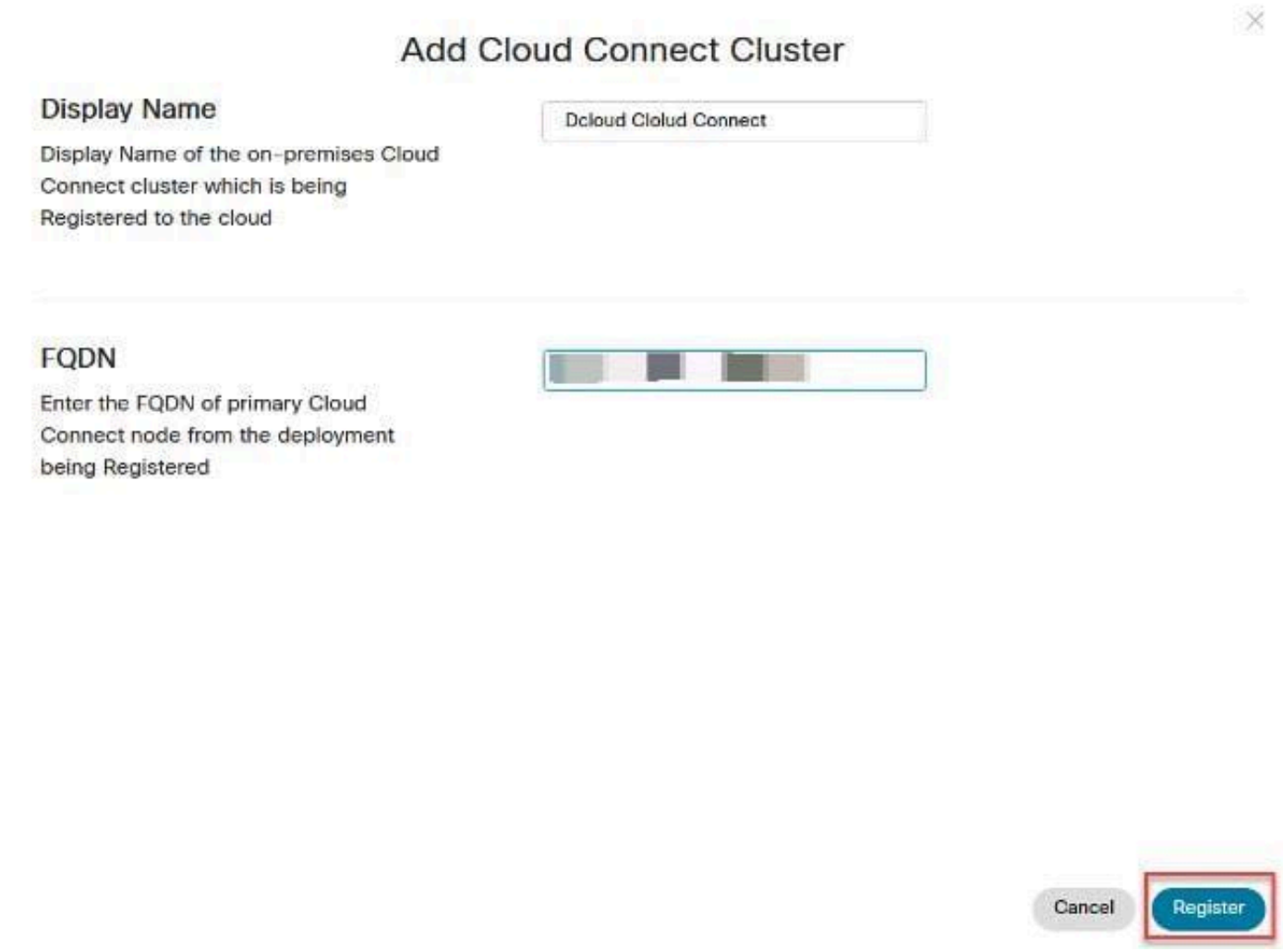
ハブのクラウド接続の制御

ステップ 4：Cloud ConnectページでAdd Cloud Connectをクリックします。



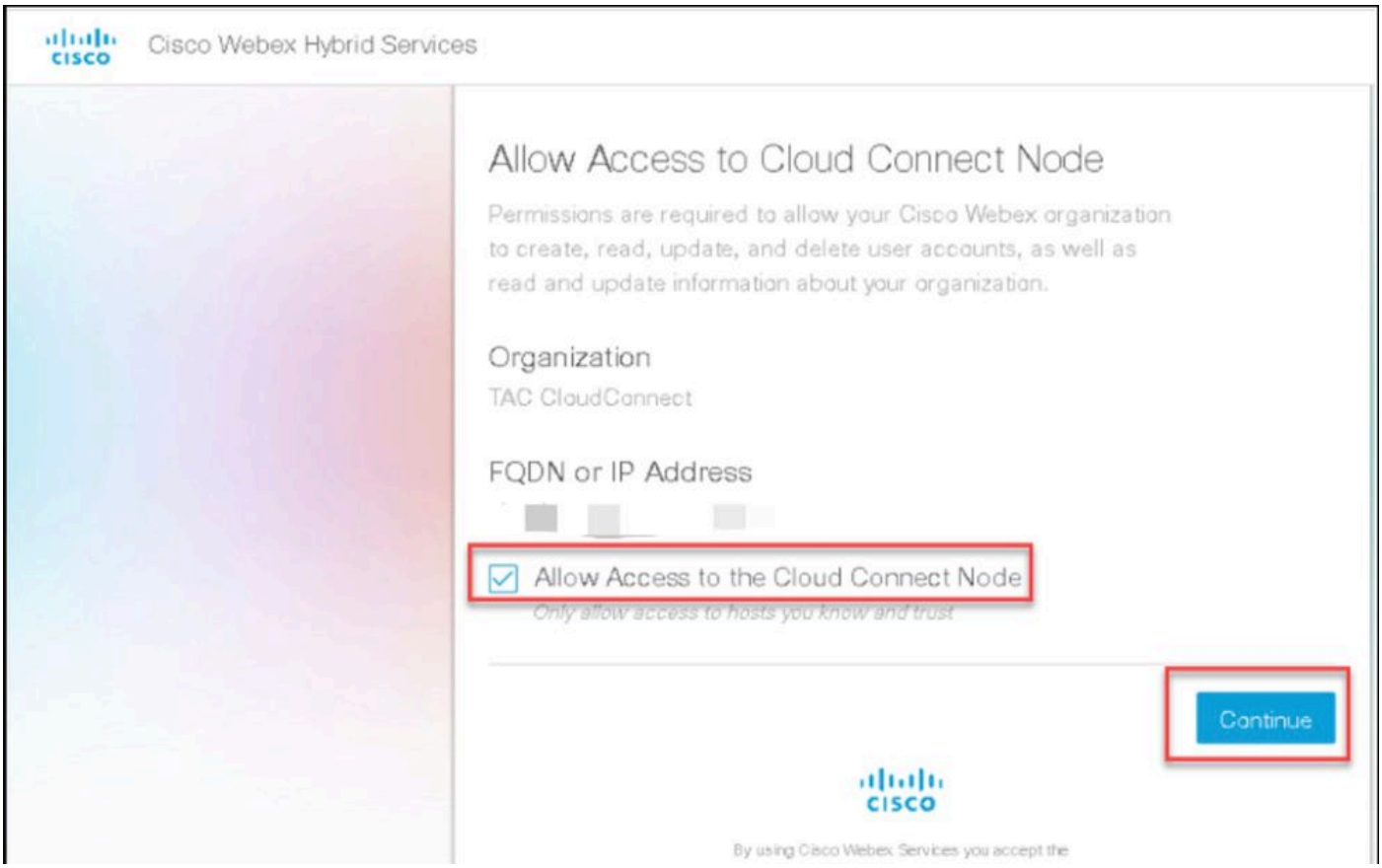
クラウド接続の追加

ステップ 5：NameとFully Qualified Domain Name（FQDN；完全修飾ドメイン名）を入力し、Registerをクリックします。



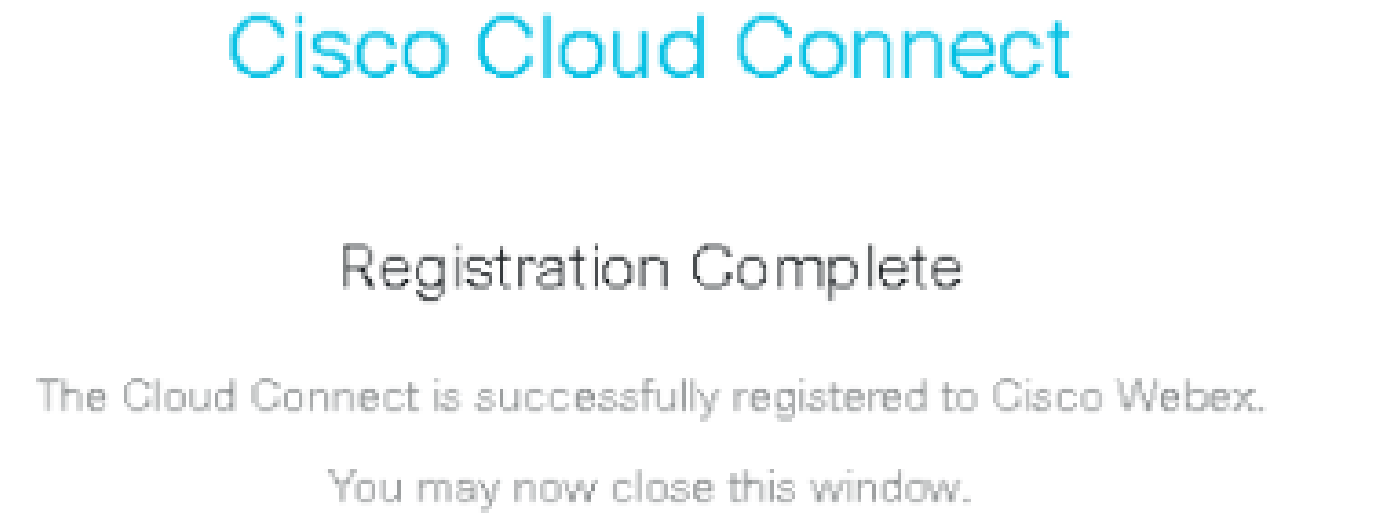
ccクラスタの追加

手順 6：Allow Accessウィンドウで、Allow Access to the Cloud Connect Nodeのボックスにチェックマークを入れてから、Continueをクリックします。



CCへのアクセスの許可

手順 7：登録が完了したことを示す確認メッセージが表示されたら、ウィンドウを閉じることができます。



クラウド接続の登録の完了

CCEの設定

Webex AIエージェントを実装するためのCCEの準備に必要な手順は次のとおりです。

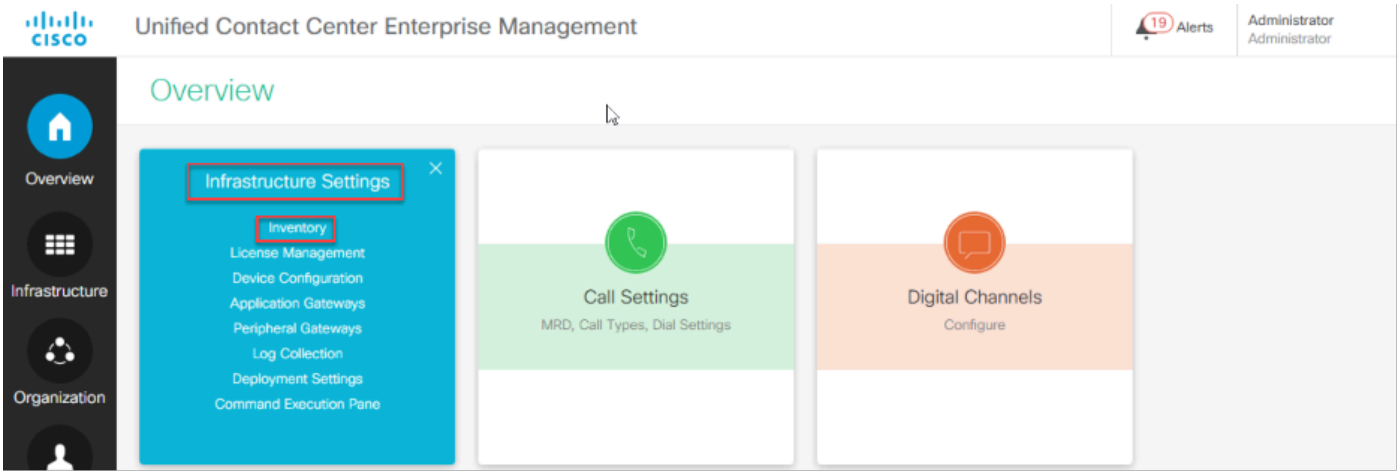
- Cloud Connectの登録
- Cloud Connectプロキシの設定
- Cloud Connect Feature Managementのステータス

Cloud Connectの登録

ステップ 1：Webブラウザを開き、Single Pane of Glass(SPOG)とも呼ばれるPCCE Web Administration Toolに移動します。

ステップ 2：管理者クレデンシャルでログインします。

ステップ 3：Web Adminインターフェイスで、Infrastructure Settingsを選択してから、Inventoryを選択します。



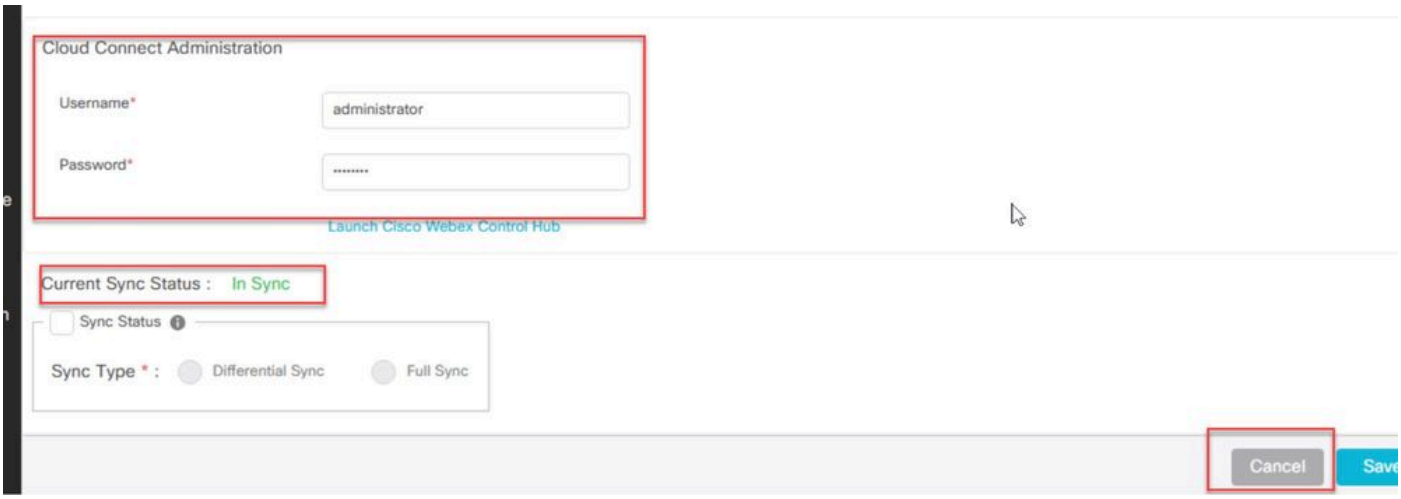
インベントリ

ステップ 4：Inventoryページで、Cloud Connectサーバを見つけます。ステータスが緑色で、アラートが表示されていないことを確認します。

この実習では、1つのCloud Connect Server（パブリッシャ）のみを示します。実稼働環境では、通常、パブリッシャとサブスクライバの2つのサーバが表示されます。

CCパブリッシャ

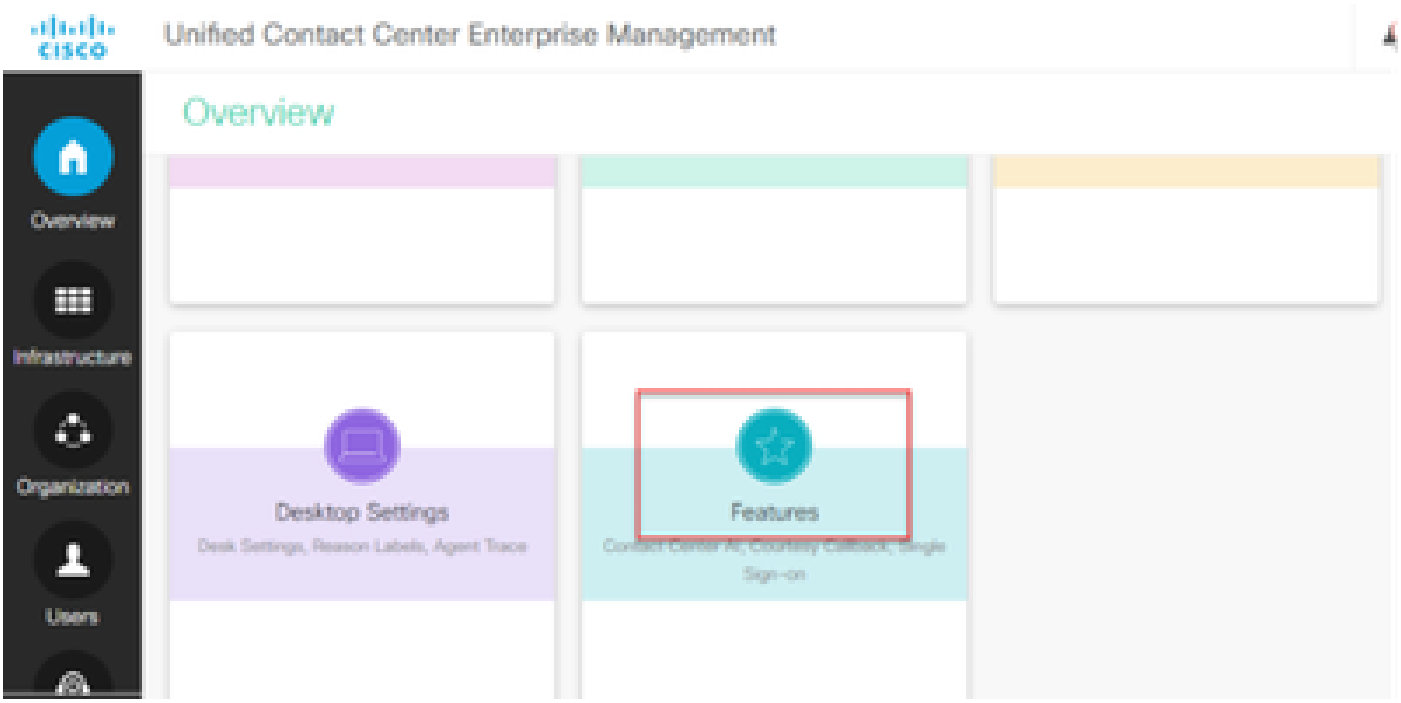
ステップ 5：Cloud Connectサーバをクリックして、管理者クレデンシャルとAWサーバとの同期ステータスを表示します。変更の必要がないため、Cancelをクリックして終了します。



保存します。

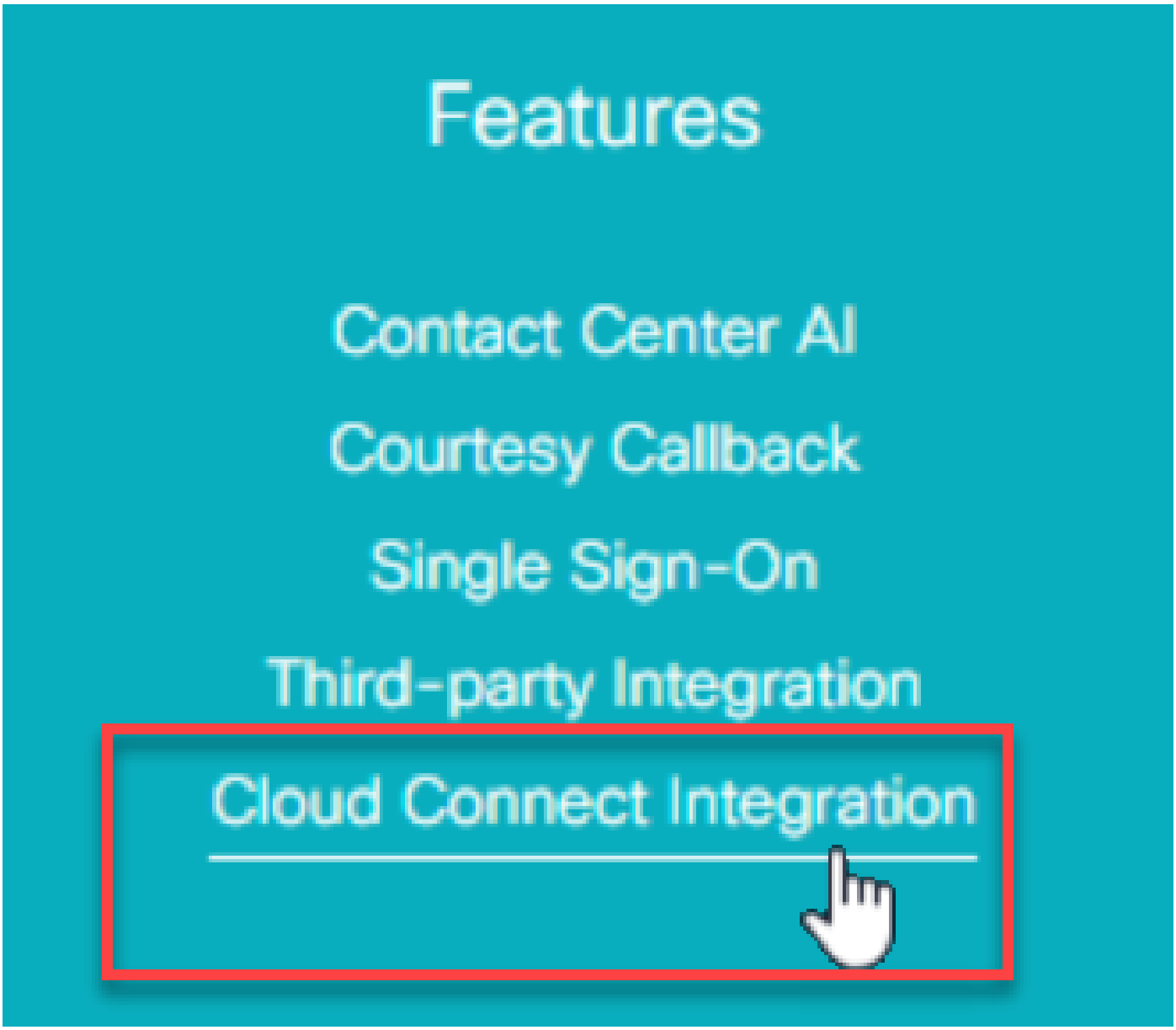
Cloud Connectプロキシの設定

ステップ 1：SPOGで、必要に応じてOverviewページに移動し、スクロールダウンして Featuresカードを見つけて選択します。



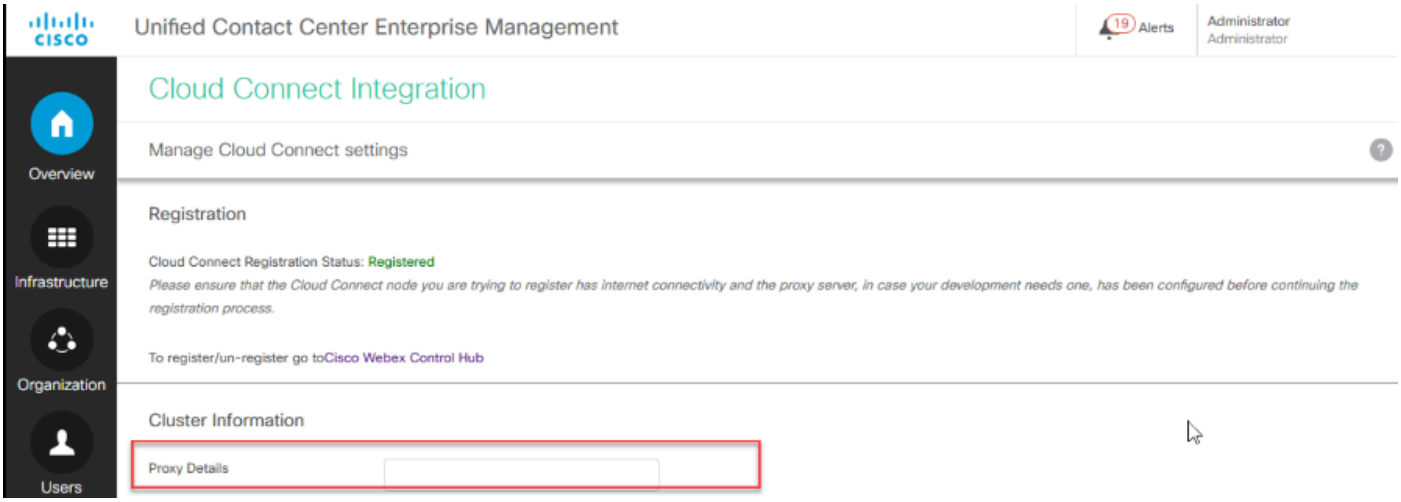
機能

ステップ 2：機能カードで、Cloud Connect Integrationをクリックします。



機能の選択

ステップ3:Cloud Connectの登録ステータスとプロキシ設定の詳細を表示します。



CCの統合

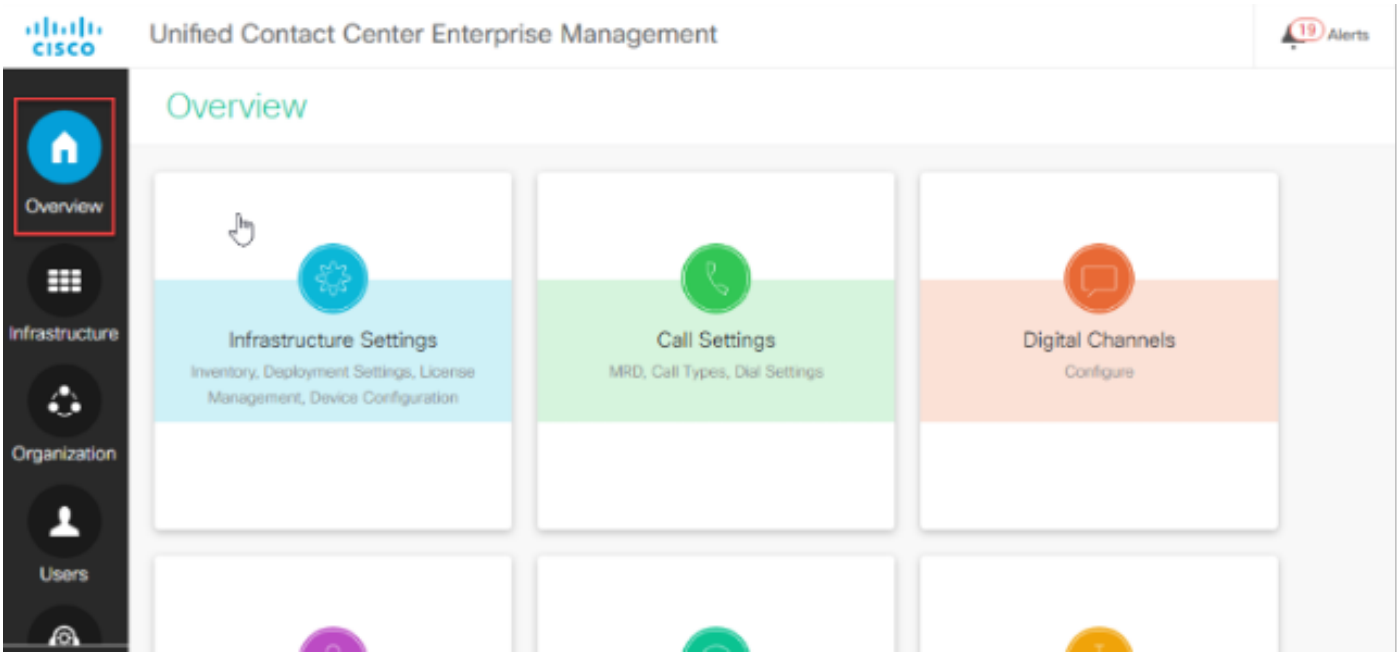
Webexサービスと通信するためにプロキシが必要な環境の場合は、Cloud Connectで使用するプロキシの詳細を入力します。

例： abc.cisco.com:8080

デフォルトでは、HTTPはポート80を使用しますが、プロキシ設定では別のポート番号を指定できます。

この実習では、プロキシは必要ないため、設定されていません。

ステップ 4：「概要」をクリックしてメインメニューに戻ります。



概要

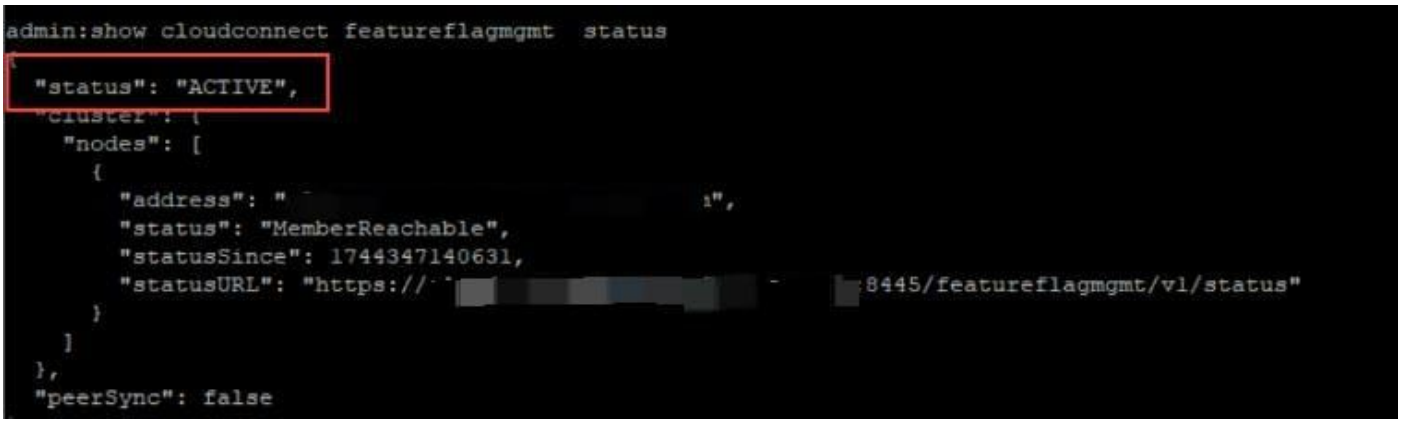
クラウド機能管理の状態

ステップ 1：Cloud Connectコマンドラインインターフェイス(CLI)にログインします。

ステップ 2：Cloud Connect CLIで、次のコマンドを入力します。

show cloudconnect featureflagmgmtのステータス

ステータスがACTIVEであることを確認する必要があります。



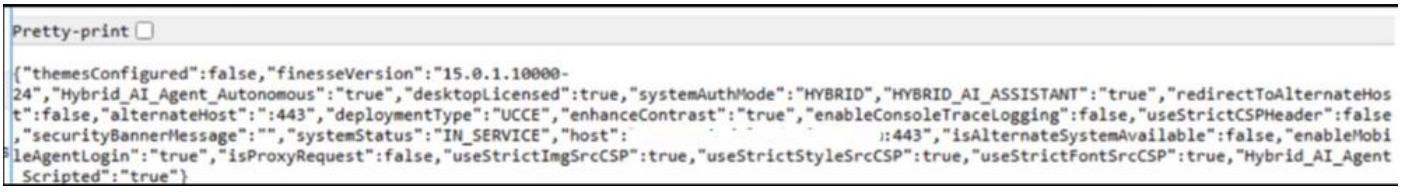
アクティブステータス

注：Cloud Connectでこの機能を有効にするには、SDKキーとしての情報が必要です。必要な情報はすべて、Contact Centerプロダクトマネージャから提供されます

次に、このWebExテナントに対して有効な機能（AIエージェントの実装に必要な機能を含む）を確認するために、Webブラウザで特定のAPIコールを実行できます。

ステップ 3：Webブラウザを開き、前述のAPIを実行します。

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



API出力

ステップ 4：API応答で、Pretty-printボックスにチェックマークを入れて、結果の書式を設定し、読みやすくします。

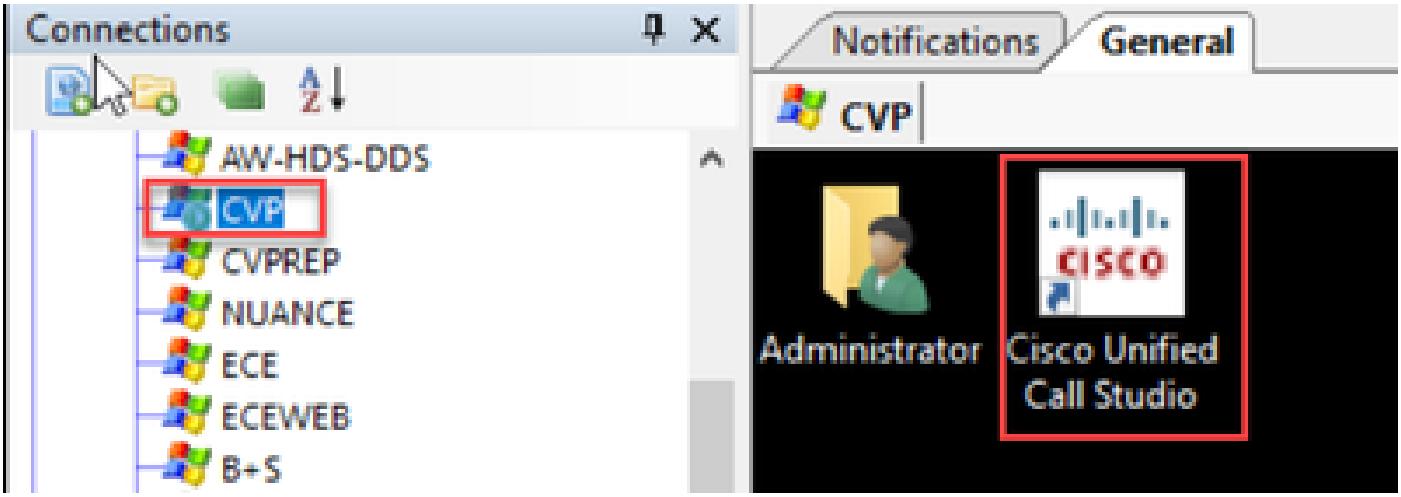
API出力

このラボでは、スクリプト化されたAIエージェントのみを設定したため、このテナントに対して有効にするためにシスコ製品チームが必要とした唯一の機能はHybrid_AI_Agent_Scriptedです。これは有効であることがわかります。

CVP Call Studioアプリケーションの設定 – スクリプトAIエージェント

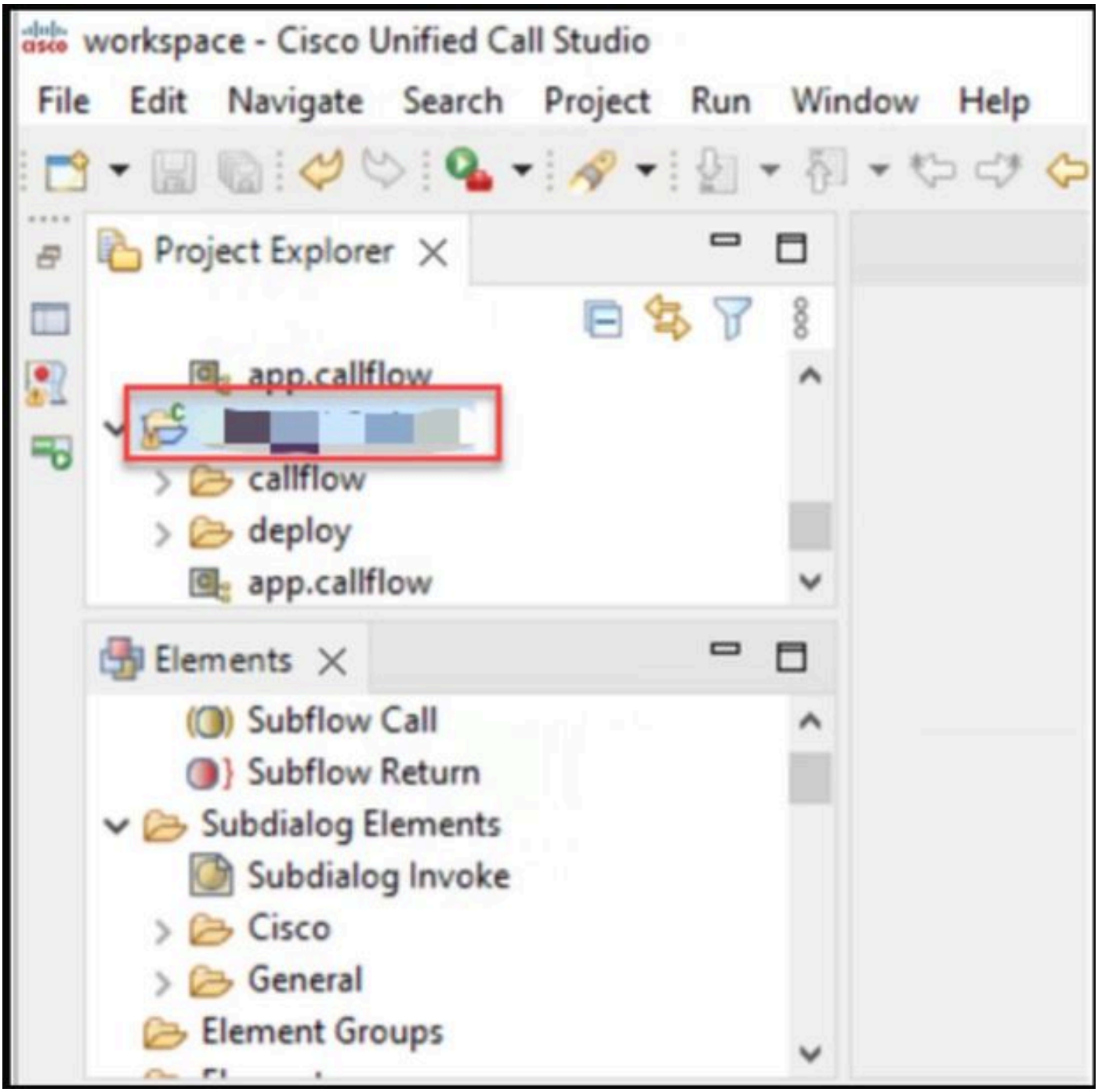
この例では、Call Studioアプリケーションが開発され、Call Studioアプリケーションでの実装のサンプルとして機能します。

ステップ 1：CVPで、デスクトップ上のCall Studioアイコンをクリックしてアプリケーションを起動します。



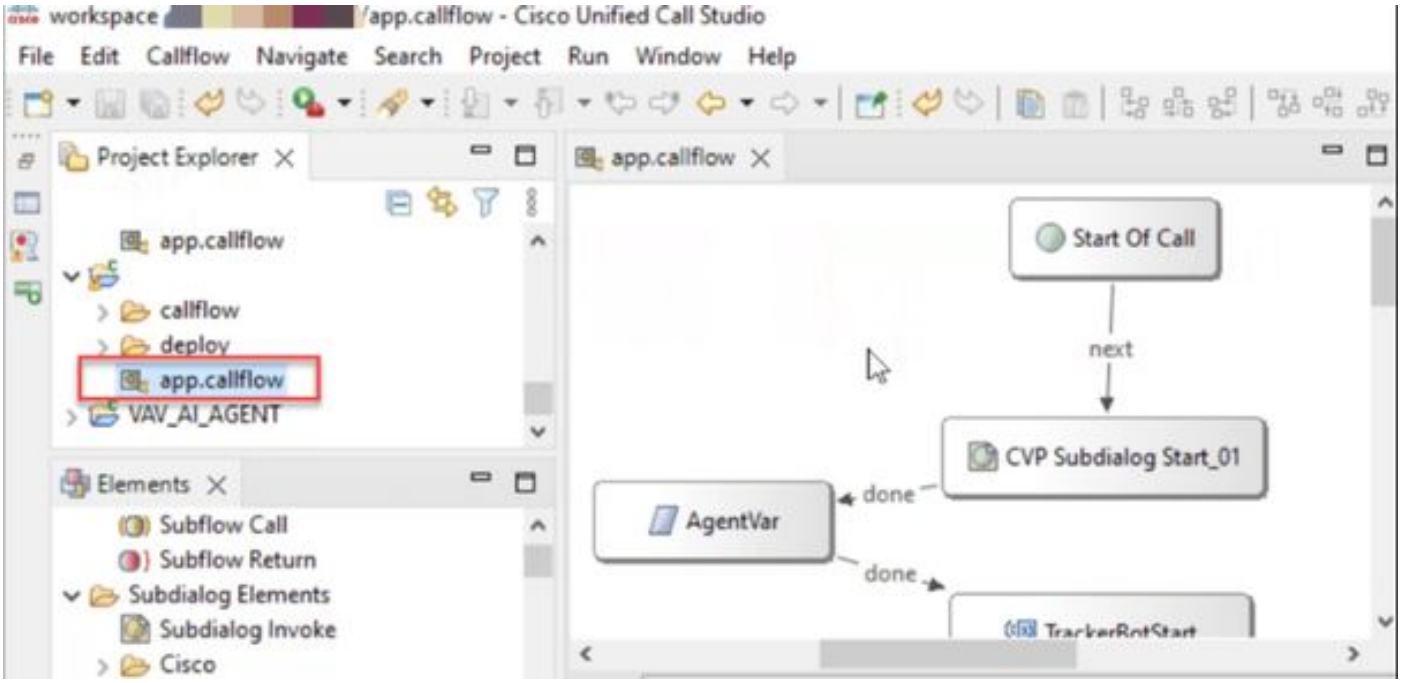
CVP

ステップ 2：CVP Call Studioで、下方向にスクロールして、Sampleアプリケーションプロジェクトを展開します。



アプリケーション

ステップ 3：Sample Applicationフォルダ内でapp.callflowをクリックして、コールフローを開きます。



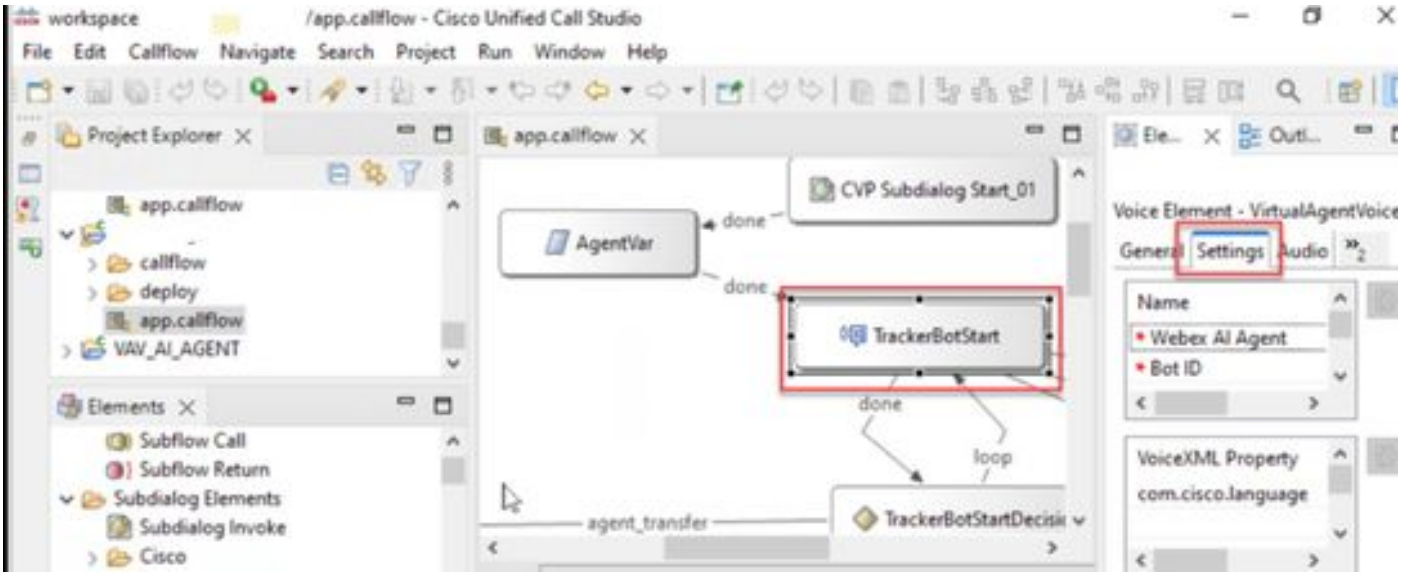
ステップ 4：このアプリケーションのコールフローが表示されます。これは、AIエージェントの相互作用を実証するように設計された単純なフローです。アプリケーションは、VAV(Virtual Agent Voice)要素を使用して、仮想エージェントとの会話を開始します。この要素には、

TrackerBotStartです。

仮想エージェントが会話を終了した後：

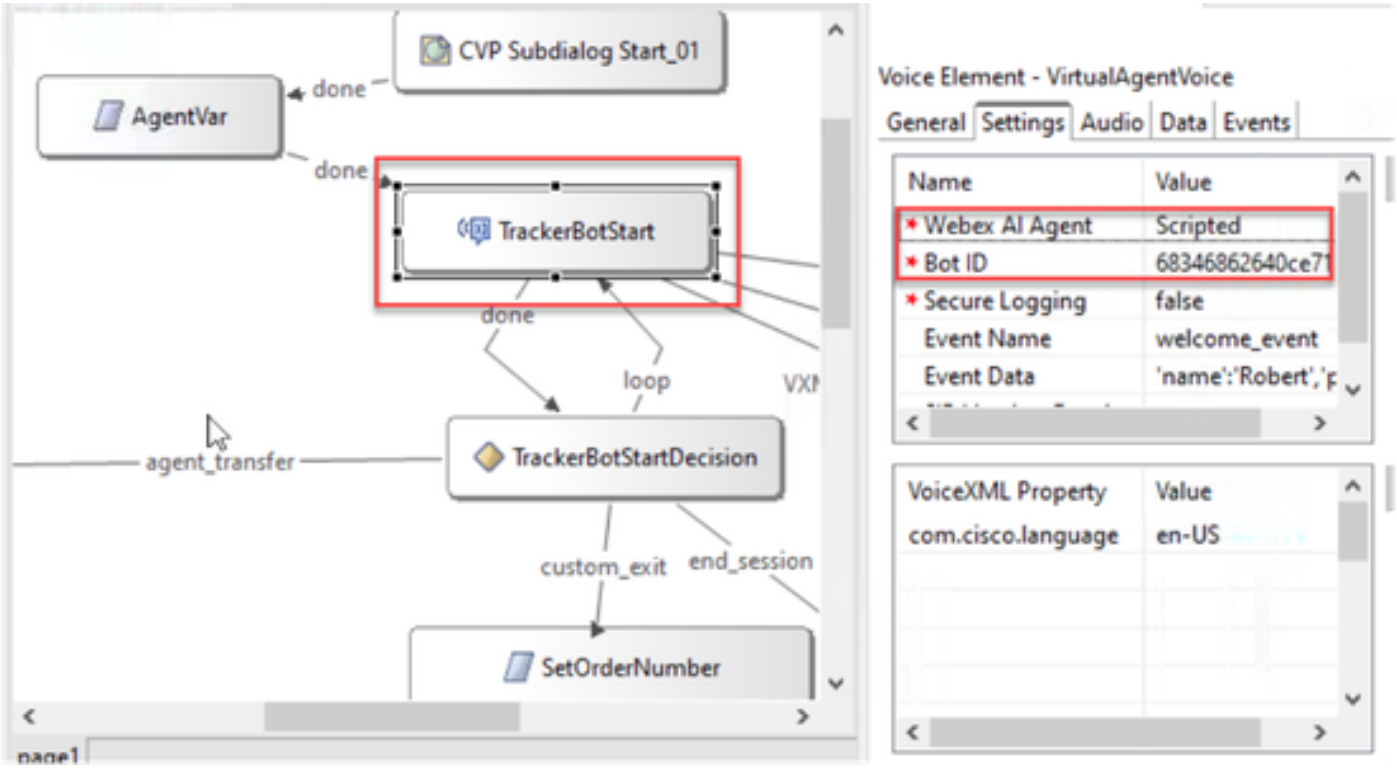
- アプリケーションは、コールをライブエージェントにルーティングするか、
- 結果に応じて、完了するまで仮想エージェントとの会話を続行します。

Virtual Agent Voice要素(TrackerBotStart)をクリックし、Settingsタブに移動して設定を開きます。

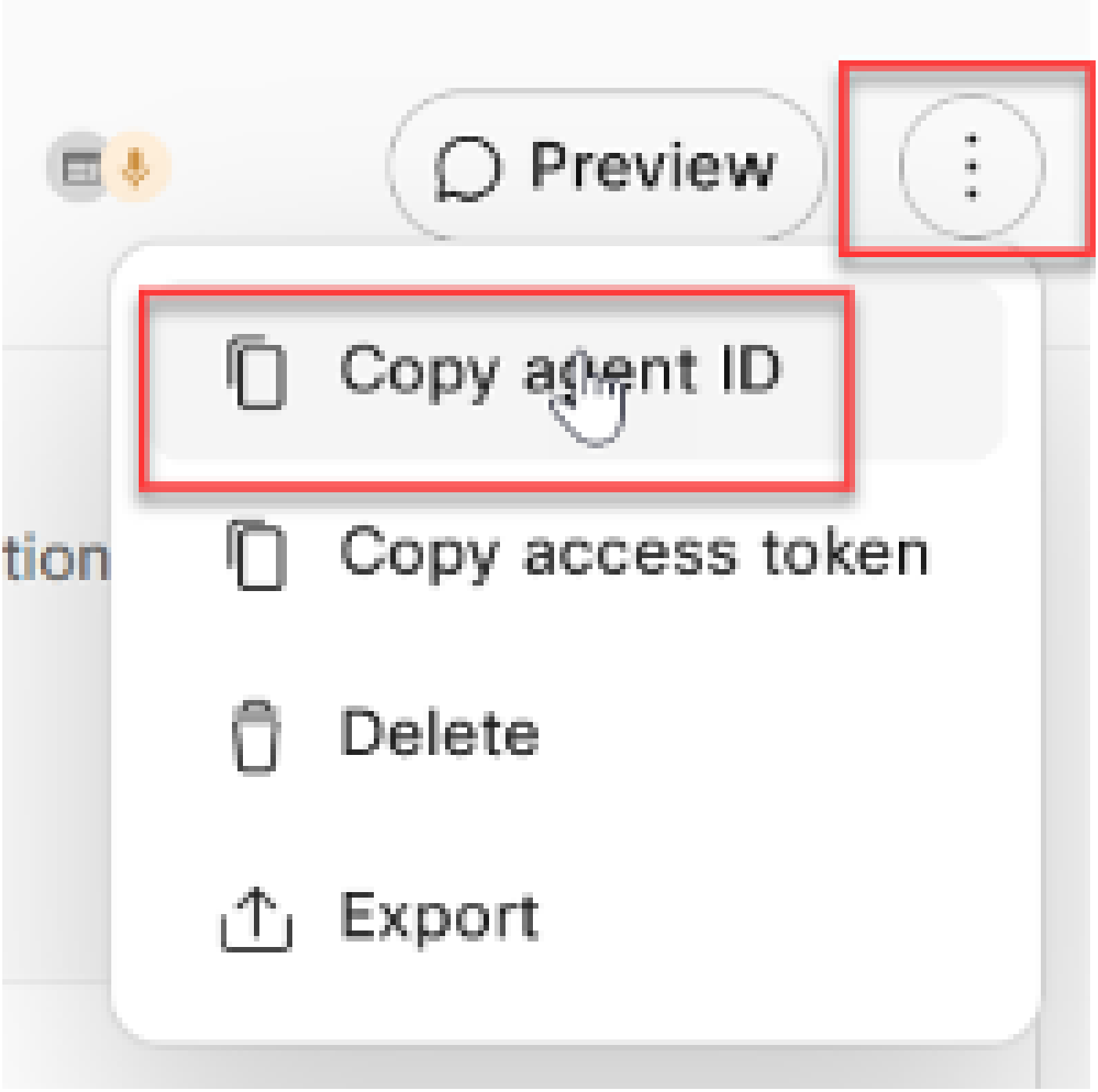


ステップ 5：AIエージェント情報を追加します。Webex AIエージェントタイプの場合は、

Scriptedを選択します。選択すると、Bot IDフィールドが表示されます。

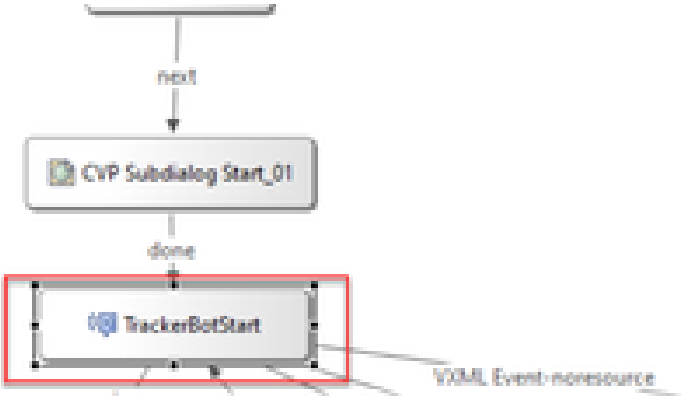


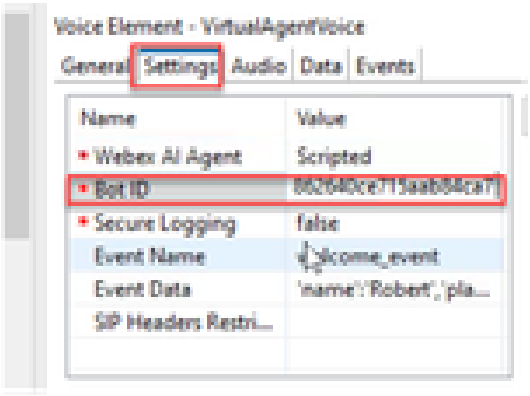
AI Agent StudioからボットIDをコピーし、ボットIDフィールドに貼り付けます。



コピーエージェントID

コピーして貼り付ける必要があるID:68346862640ce715aab84ca7





Call Studioのスクリーンショット

セキュアロギング：この設定は、要素からの機密データの可能性をログに記録するかどうかを制御します。有効（trueに設定）にすると、AIエージェントからのクエリーテキスト、フルフィルメントテキスト、JSONなどの要素出力がマスクされます。ここでは、Secure Loggingはfalseのままにしておきます。

AIエージェントにイベントを渡す必要があります。Event Nameを必ずwelcome_eventに設定してください。

Call Studioのスクリーンショット

イベントデータフィールドに変数と値を追加することで、AIエージェントに情報を送信できます。前述の変数と対応する値をそこに入力してください。

名前：名前

値：自分の名前

氏名：場所

価値：所在地

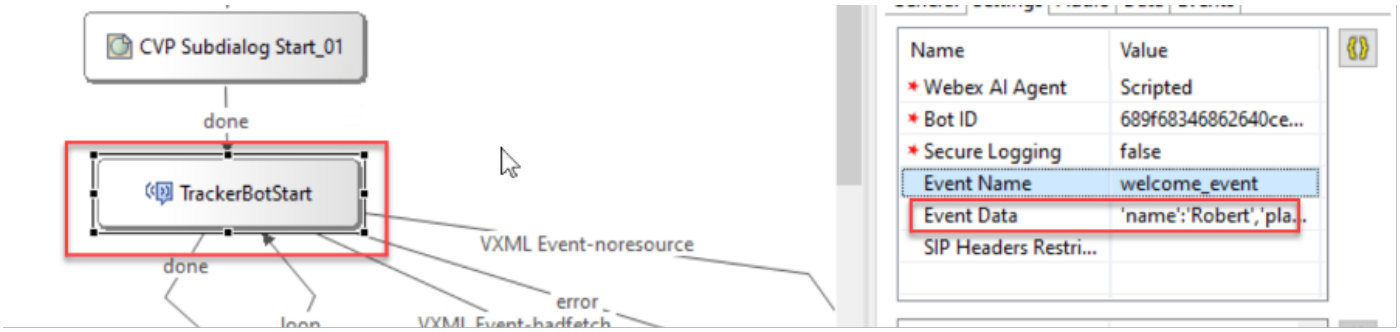
この例の場合、

名前：名前

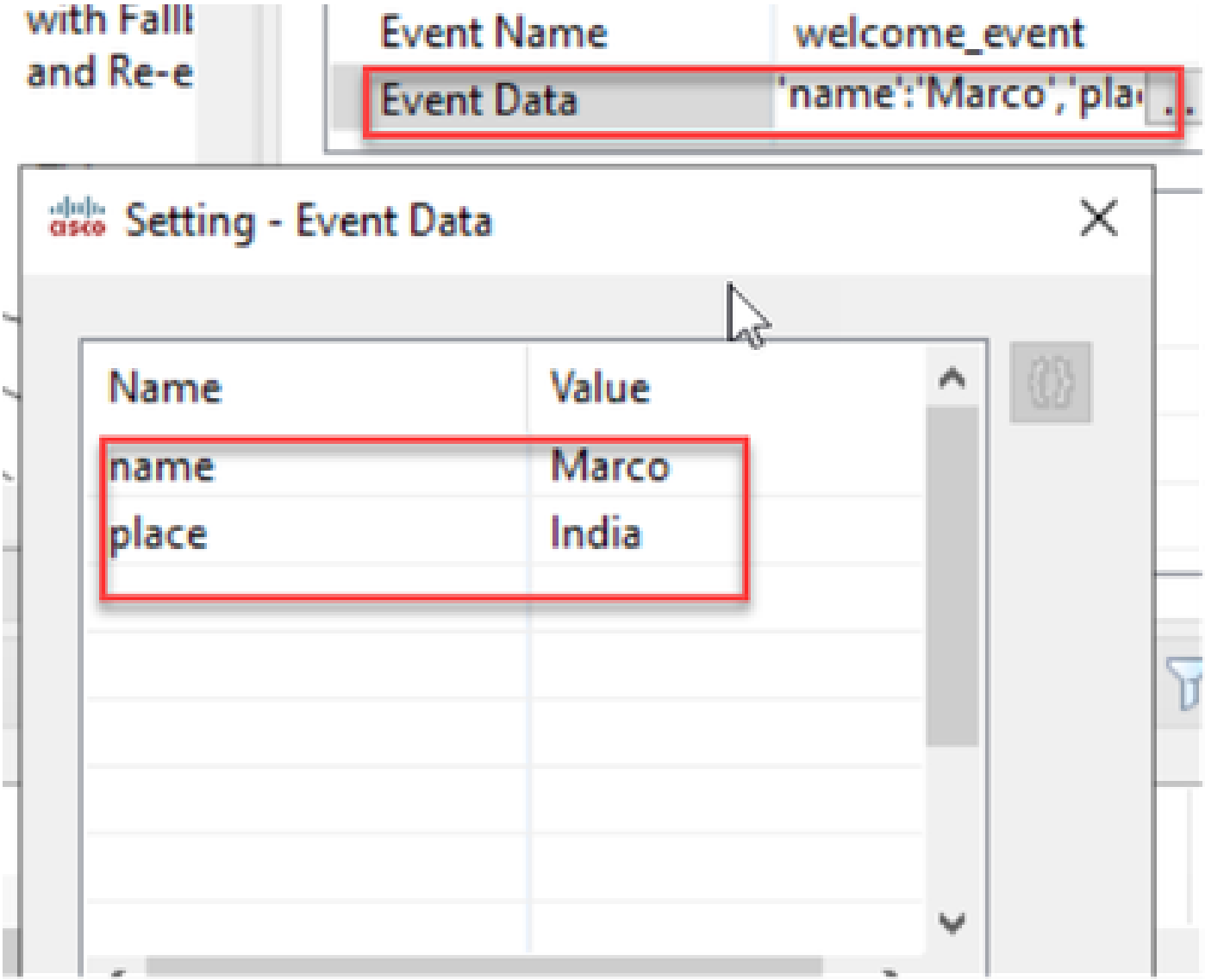
値：Marco

氏名：場所

値：インド



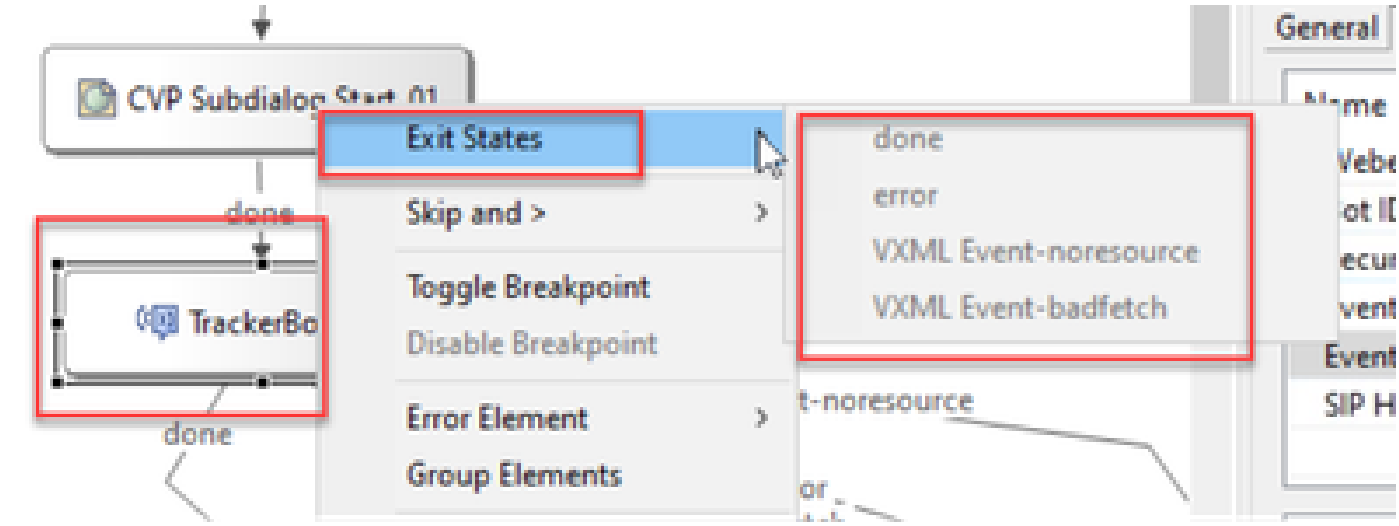
Call Studioのスクリーンショット



Call Studioのスクリーンショット

ステップ 5：次に、リソースが使用できない場合、またはAIエージェントとの統合中にgRPCエラーが発生した場合に、CVPコールフローを正常に処理する方法を確認します。

VAV要素(TrackerBotStart)を右クリックし、終了状態を確認します。

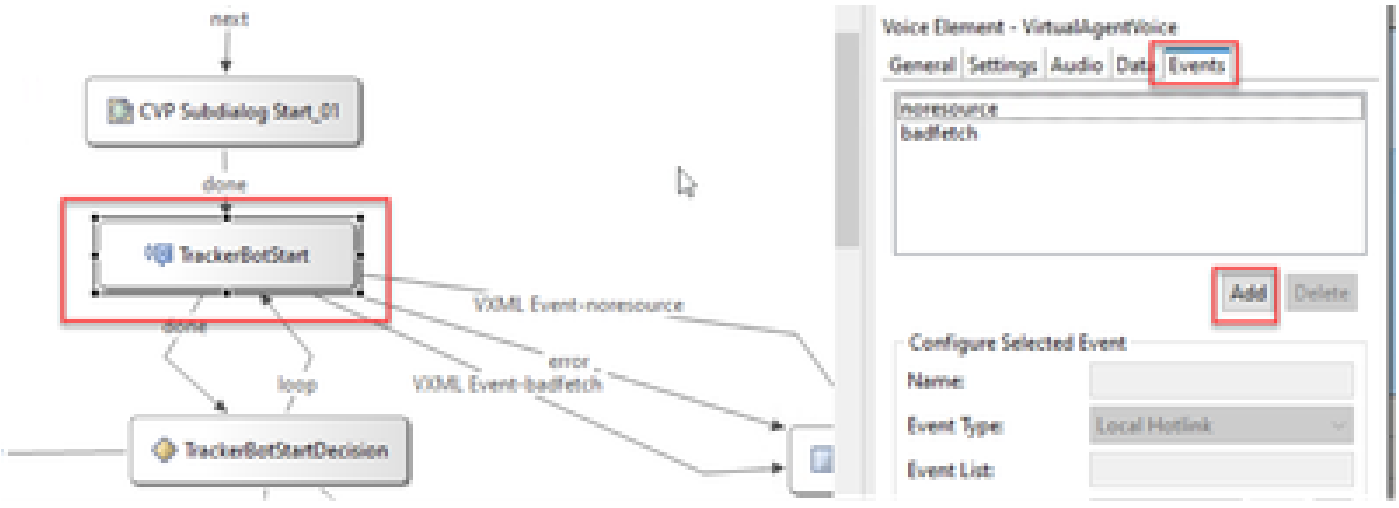


Call Studioのスクリーンショット

デフォルトでは、エラーVXML Event-noresourceおよびVXML Event-badfetchは要素の一部です。これらのエラーのいずれかが発生した場合、アプリケーションはオーディオメッセージを再生してからコールをライブエージェントに転送することでフォールバックするように設計されています。

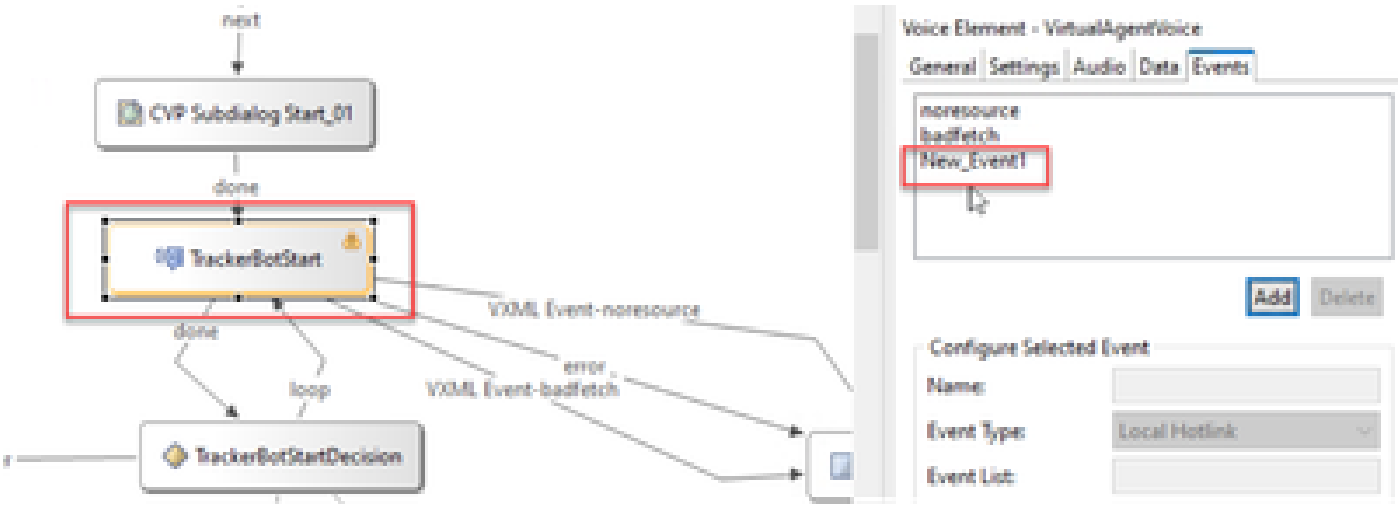
このCall Studioアプリケーションでは、これらのエラーのいずれかがトリガーされた後に音声プロンプトが自動的に再生されるので、発信者にとってスムーズで有益なエクスペリエンスが保証されます。

特定のケースを処理する必要がない限り、追加のイベントを追加する必要はありません。新しいVXMLイベントを追加する場合は、VAV要素をクリックし、[イベント]タブで[追加]をクリックします。



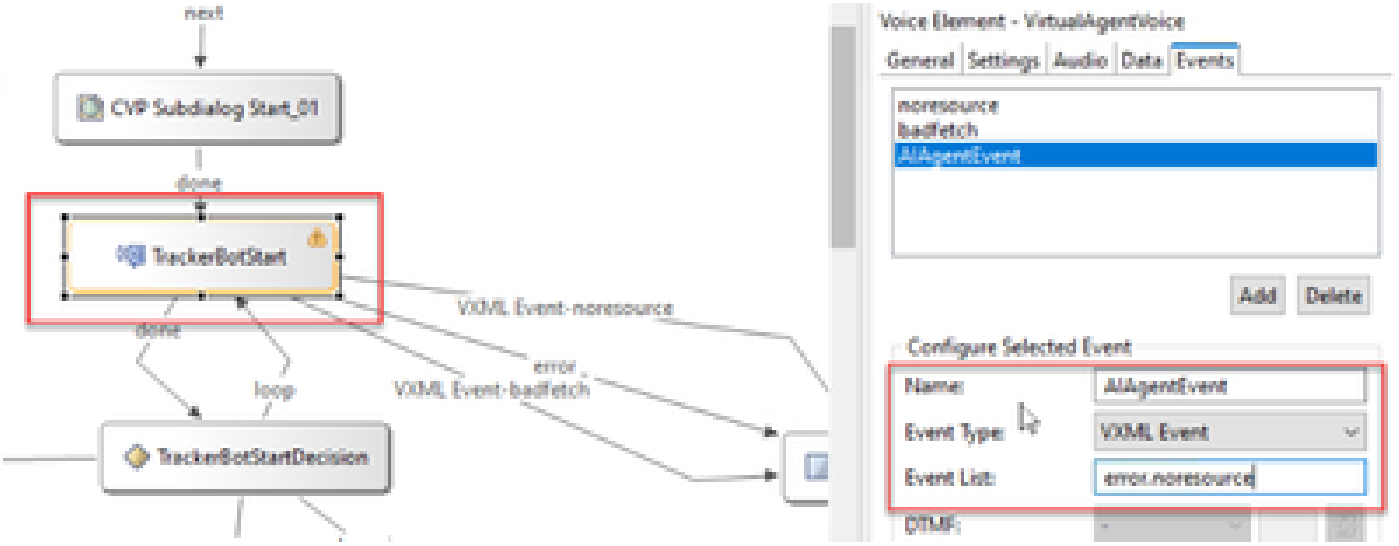
Call Studioのスクリーンショット

New_Event1という名前の新しいイベントが追加されます。名前を変更してイベント設定を構成するには、このフィールドを選択します。



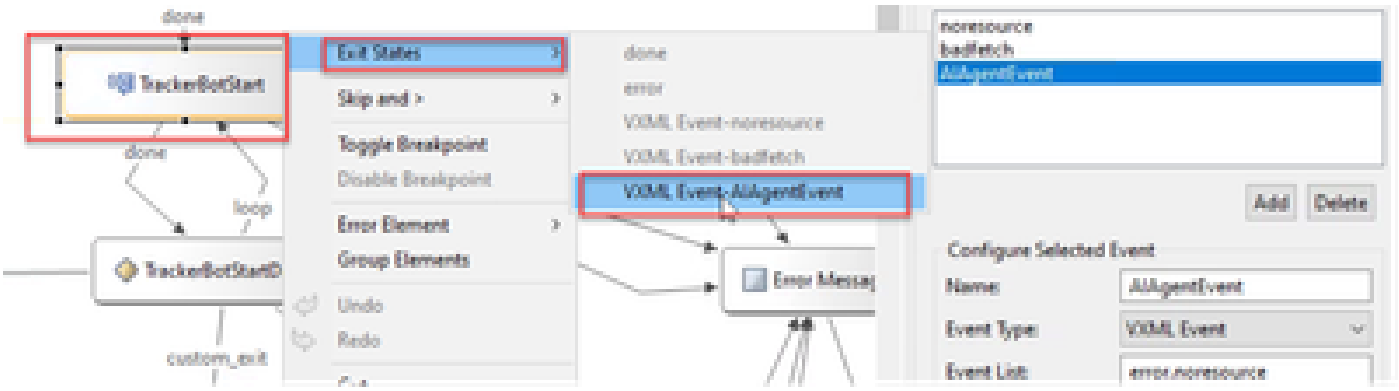
Call Studioのスクリーンショット

名前をAIAgentEventに変更します。Event TypeはVXML Eventに設定したままにします。Event Listでは、error.noresourceを選択できます。



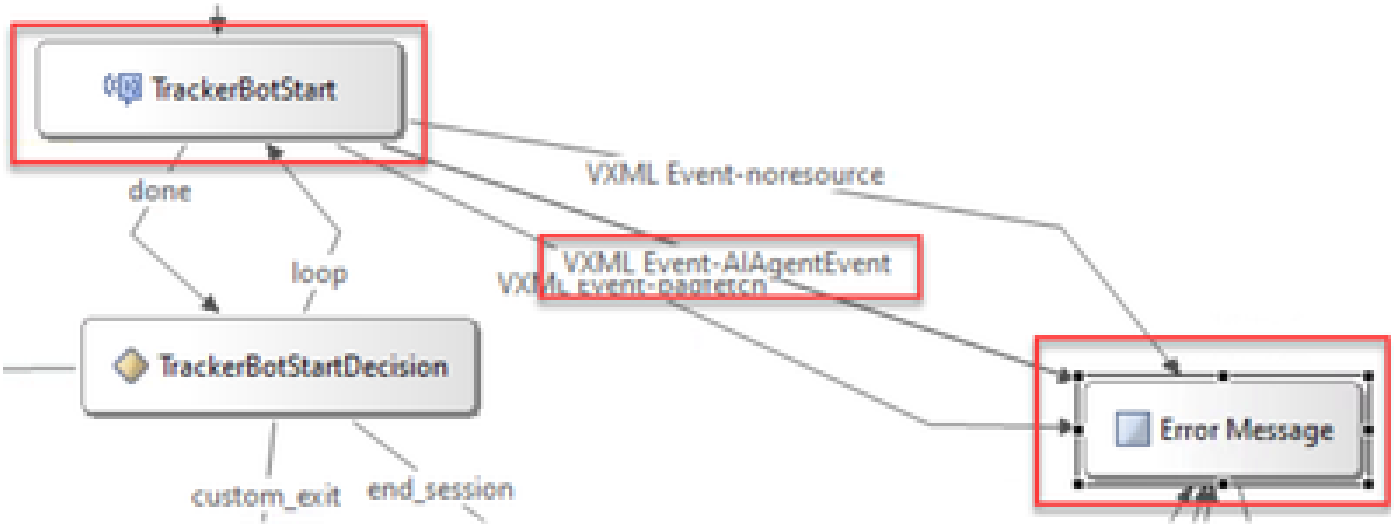
Call Studioのスクリーンショット

ここで、新しい終了状態を、Error Message要素などの他のエラーを処理するオーディオ要素に送ります。そのためには、VAV要素を右クリックし、[終了状態]を選択します。次に、新しいイベント(AIAgentEvent)を選択し、矢印をエラーメッセージ要素までドラッグします。



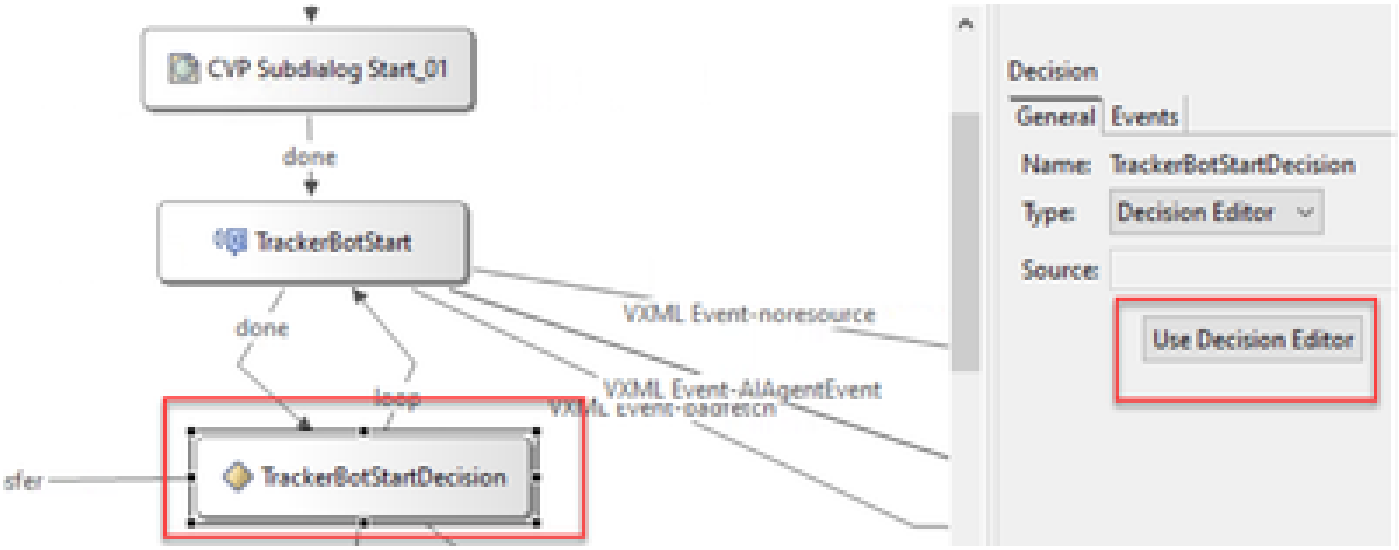
Call Studioのスクリーンショット

最後に次のように表示されます。



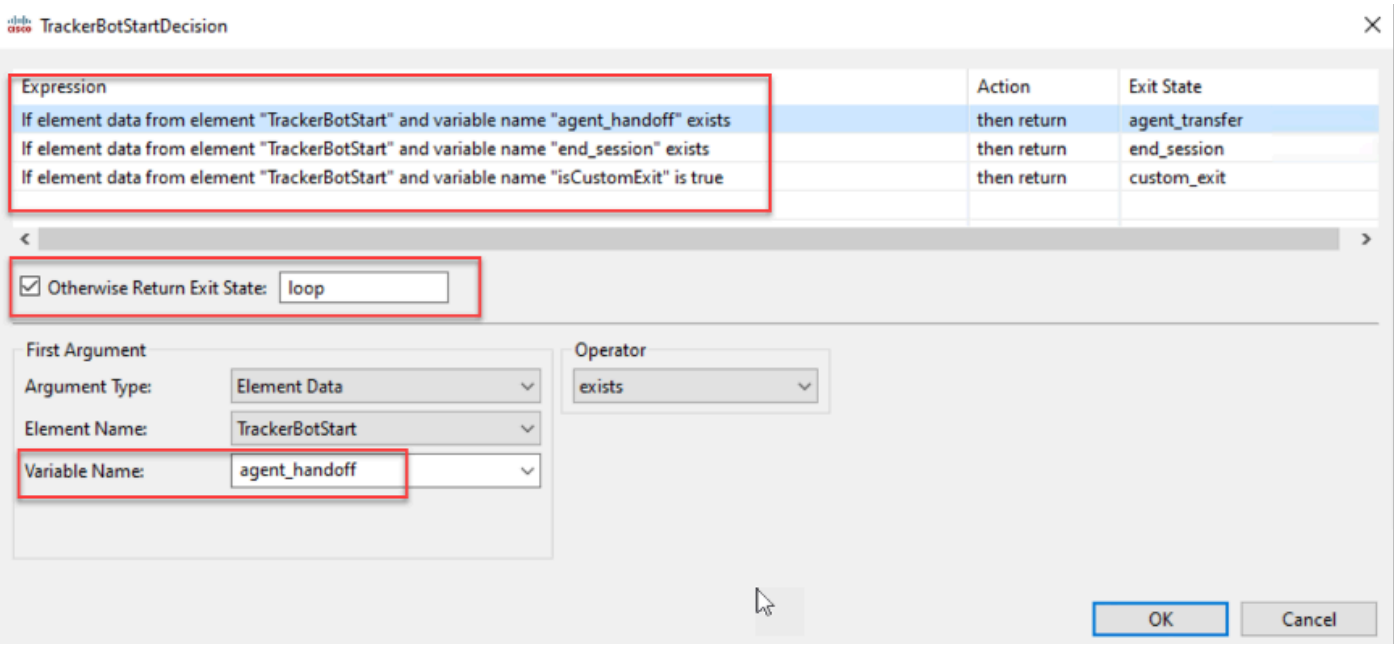
Call Studioのスクリーンショット

手順 6：次に、意思決定ノードについて説明します。Decisionノードをクリックし、Use Decision Editorを選択します。



Call Studioのスクリーンショット

手順 7：実際のエージェントにコールを送信する、セッションを終了する、またはカスタムEXITを使用するオプションが表示されます。これらのアクションは、AIエージェントの結果に依存する3つのデータ要素に依存します。確認したら、OKをクリックします。



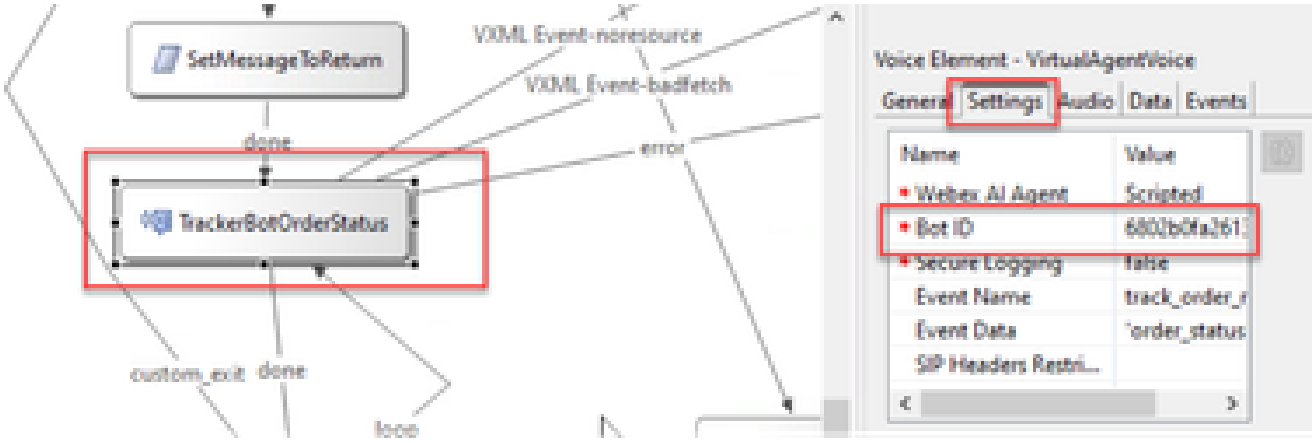
ボットの開始決定

ステップ 8：AIエージェントとのインタラクションが終了すると、このCall Studioアプリケーションが結果を評価します。その評価に基づいて、AIエージェントとの会話は2番目のVAV要素を使用して続行できます。

Call Studioアプリケーションをスクロールダウンすると、TrackerBotOrderStatusという名前の2番目のVAV要素が見つかります。

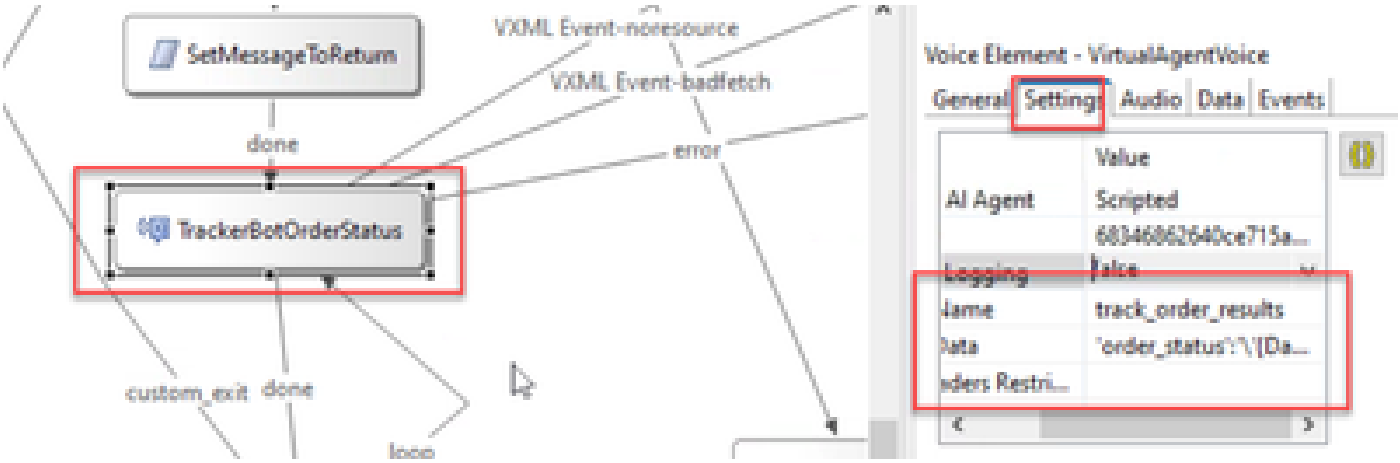
この要素に同じボットIDを追加し、既存のイベント名とイベントデータは設定済みのままにします。

コピーして貼り付ける必要があるID:68346862640ce715aab84ca7



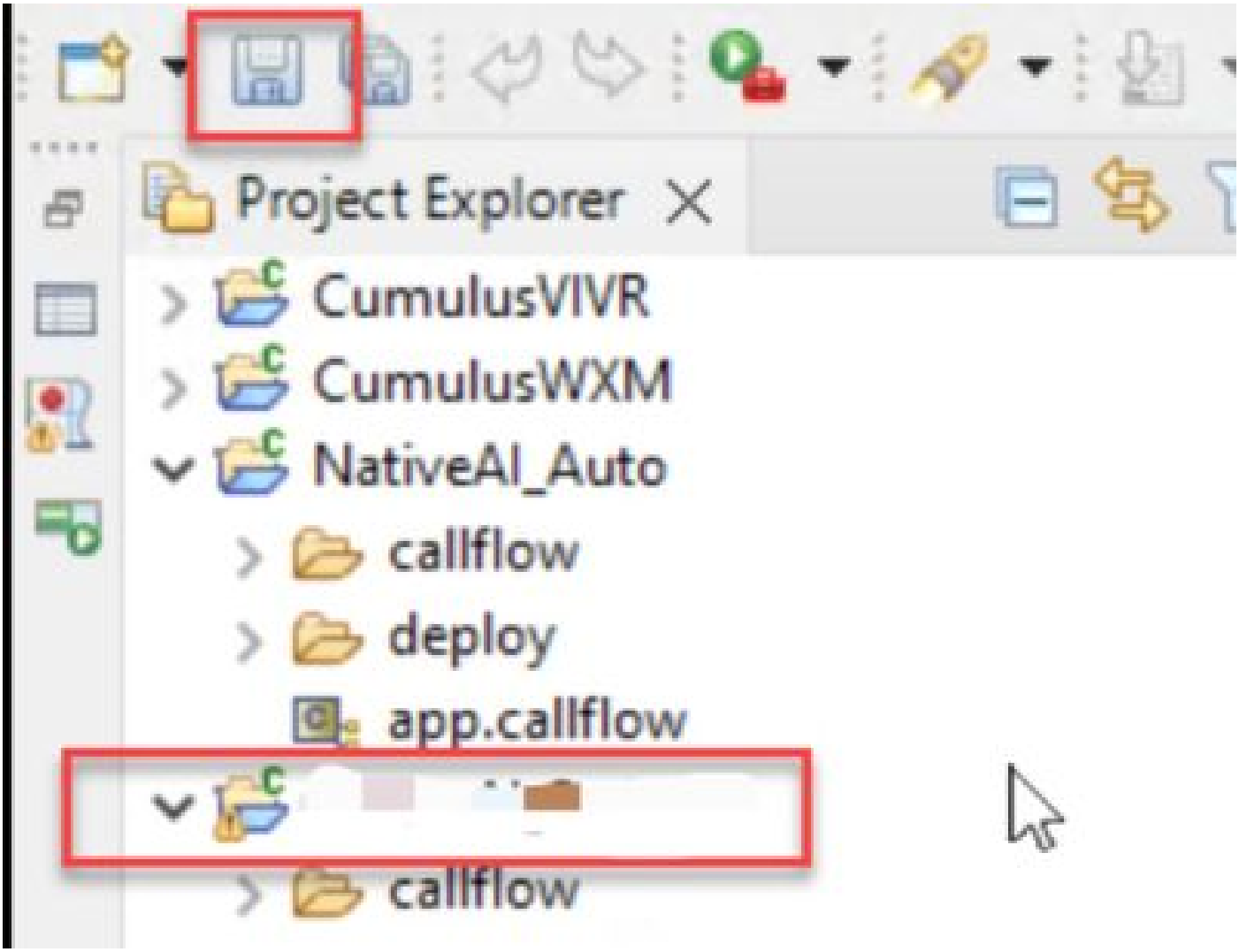
Call Studioのスクリーンショット

Event DataとEvent Nameは変更せずに、設定済みのままにします。

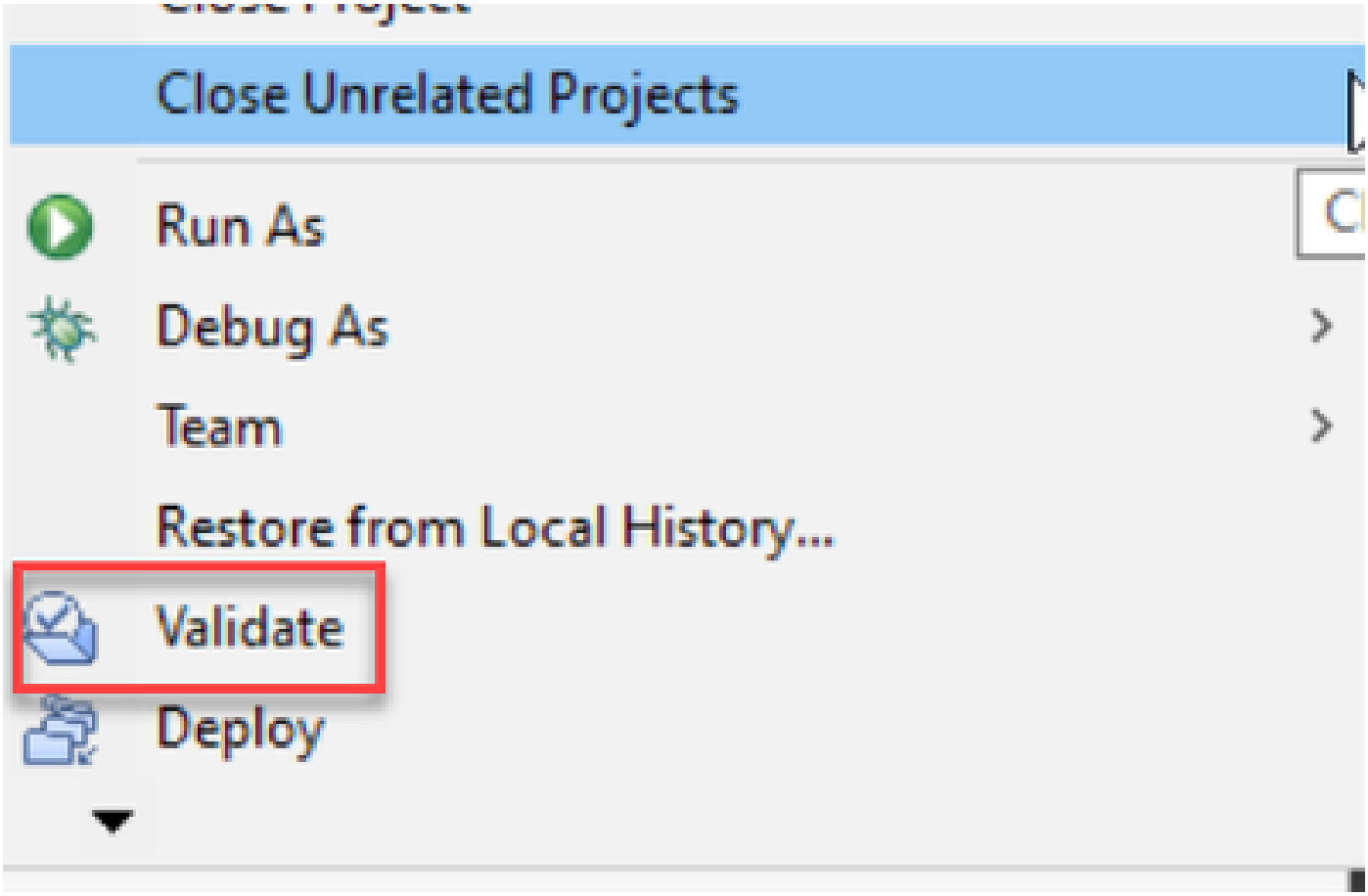


Call Studioのスクリーンショット

ステップ 9：これで、アプリケーションを保存して展開する準備ができました。Save アイコンをクリックして、変更を保存します。

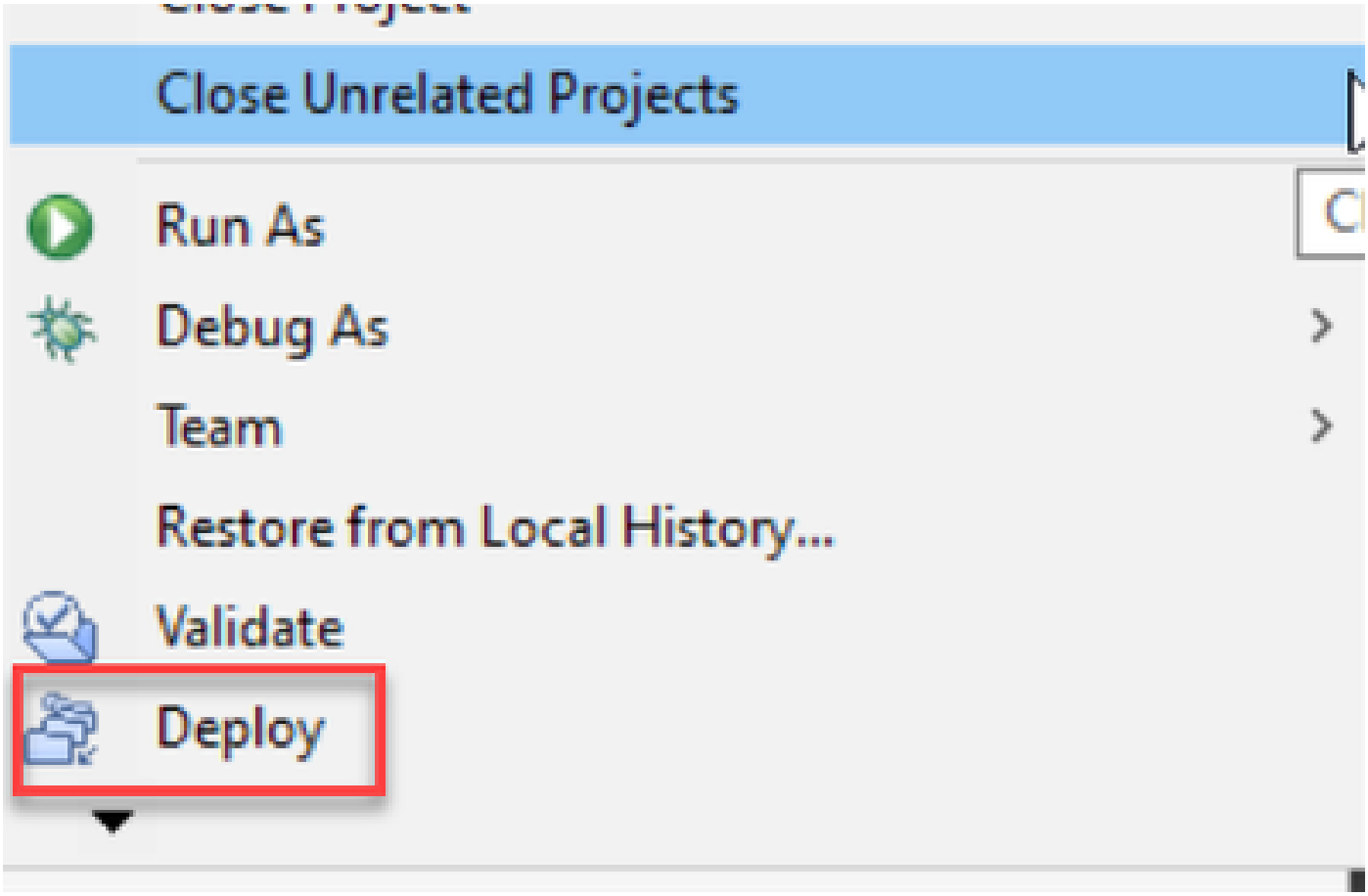


ステップ 10：アプリケーションを検証します。アプリケーションを検証するには、Sampleアプリケーションを右クリックして、Validateを選択します。



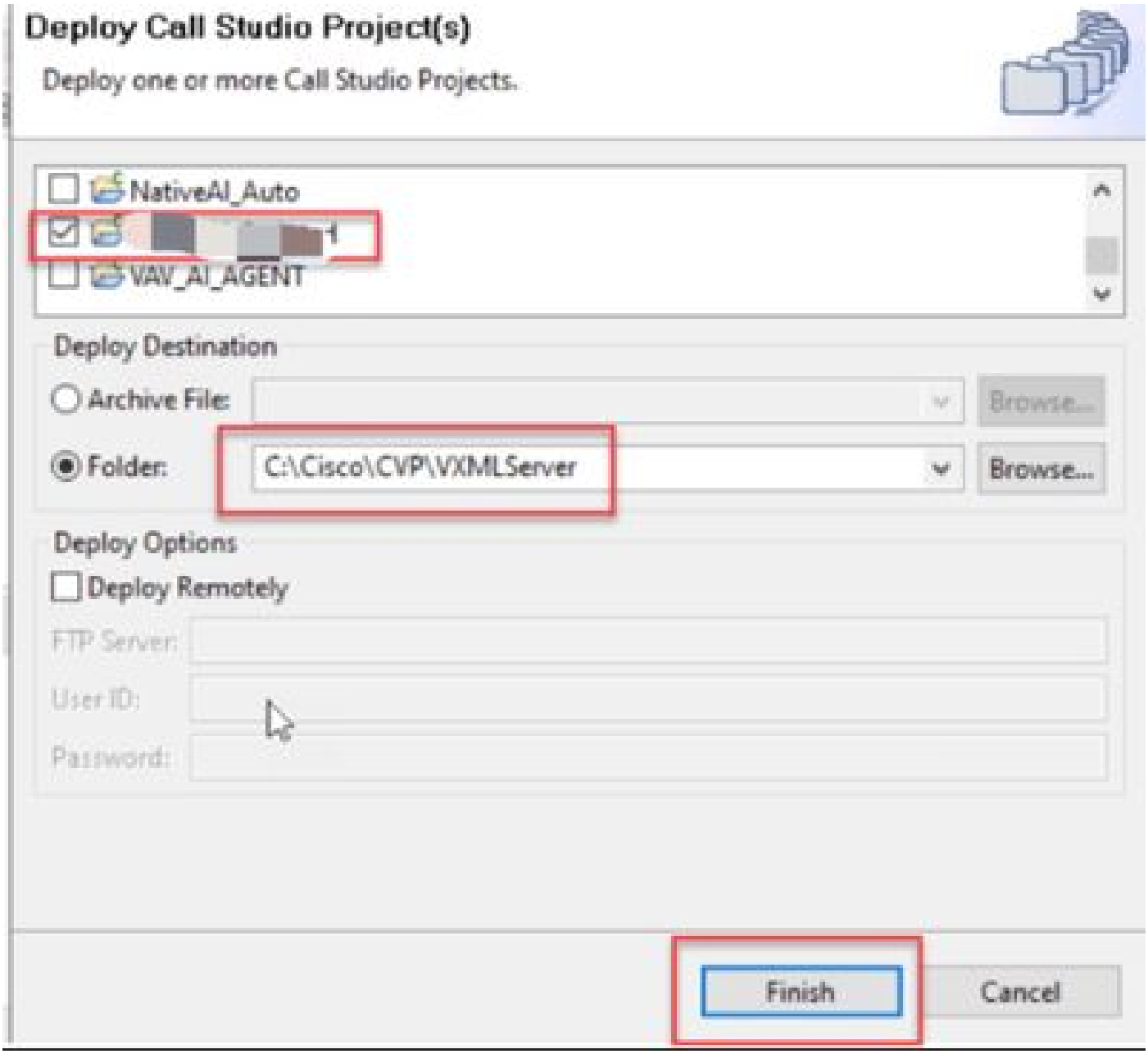
Call Studioのスクリーンショット

ステップ 11アプリケーションを配備します。VXML Serverにアプリケーションを展開するには、Sample Applicationを右クリックして、Deployを選択します。

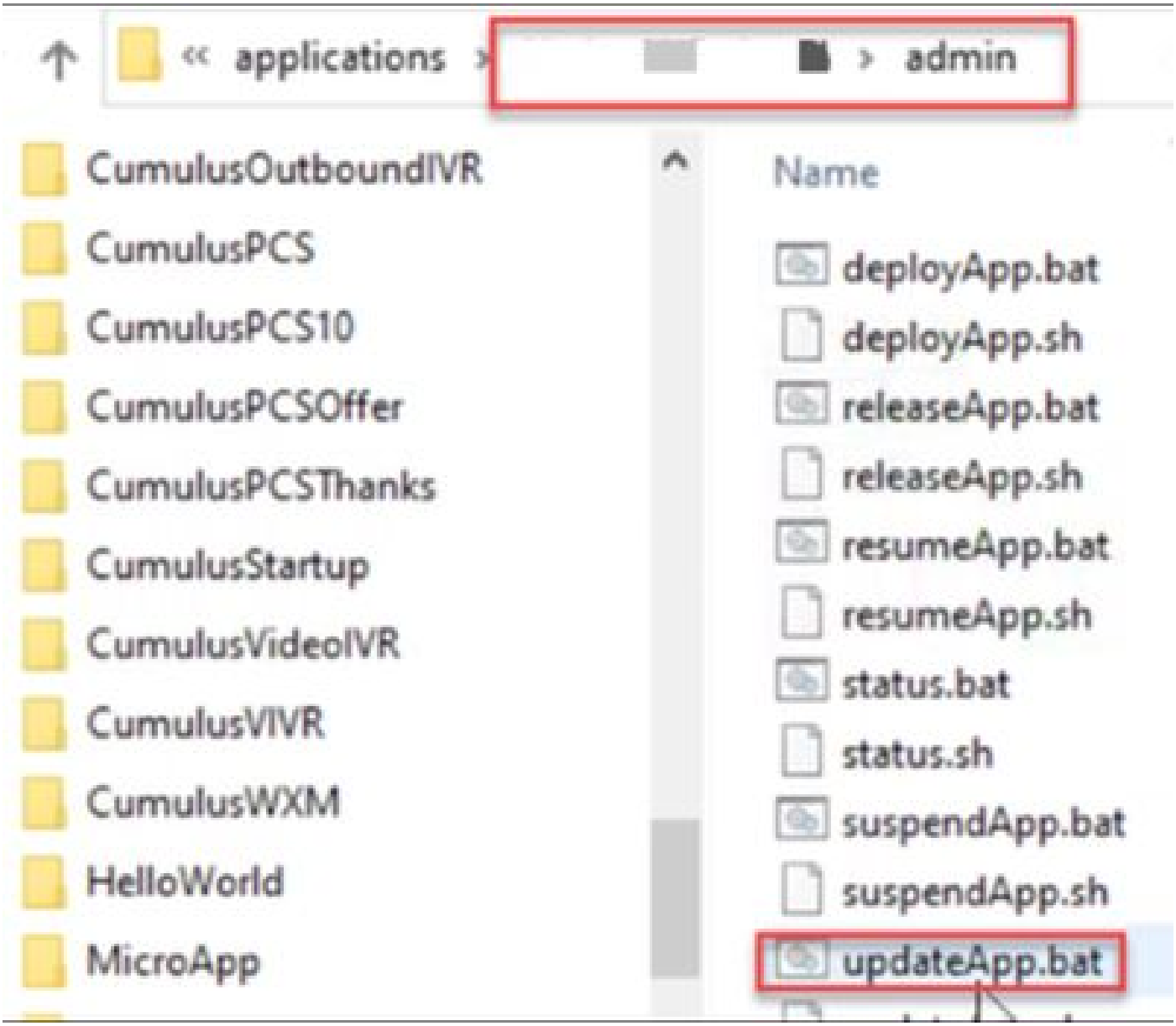


Call Studioのスクリーンショット

ステップ 12次のウィンドウで、Sample Applicationにチェックマークを付け、フォルダをC:\Cisco\CVP\VXMLServerに設定して、デフォルト設定を保持します。続いて、Finishをクリックします。



ステップ 13VXMLサーバでアプリケーションを更新します。
C:\Cisco\CVP\VXMLServer\applications\NativeAI_Scripted\adminに移動し、ダブルクリックするか、コマンドプロンプトで実行して、updateApp.batを実行します。



ステップ 14：開いた新しいコマンドウィンドウで、yesと入力して、アプリケーションを更新することを確認します。

翻訳について
シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。