

Unity Express カスタム スクリプトのクイック スタート ガイド

内容

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[表記法](#)

[概要](#)

[Cisco Unity Express スクリプト エディタの使用](#)

[基本的なスクリプトの作成](#)

[統合方法](#)

[サンプル コールフロー](#)

[サンプル スクリプト](#)

[Cisco Unity Express へのスクリプトのアップロード](#)

[トラブルシューティング](#)

[付録](#)

[付録 1：システム プロンプト](#)

[付録 2：デフォルト トレースのリセット](#)

[関連情報](#)

概要

このドキュメントでは、Cisco Unity Express (CUE) システムについての情報を示します。具体的には、製品のカスタム自動応答アプリケーションを作成する際の手引きを示します。

Cisco Unity Express の詳細については、『[Cisco Unity Express スクリプト エディタ ガイド](#)』を参照してください。このドキュメントは、Cisco Unity Express エディタの使用と Cisco Unity Express 製品内でのスクリプトの管理を開始するための基本的なガイドを必要とする管理者を対象としています。

注：カスタムスクリプトのサポートは、通常のシスコテクニカルサポートでは利用できません。カスタム スクリプトに関する質問、コメント、およびヘルプについては、ask-cue-editor@external.cisco.com 宛に電子メールを送信してください。

前提条件

要件

読者は、コマンドライン インターフェイス (CLI) または GUI を通じて Cisco Unity Express の

管理と設定に精通している必要があります。

Cisco Unity Express スクリプト エディタ (CUEEditor2.1.1.exe)、およびこのドキュメントで使用するサンプル スクリプト (CUE-AA-S6-AASample.zip) は、[Cisco Unity Express 2.1.1 ソフトウェア ダウンロード センター](#)から入手できます。

使用するコンポーネント

このドキュメントの情報は、Cisco Unity Expressバージョン2.1.1に基づいています。この例には、Cisco Unity Express 2.1.1でのみ使用可能な手順が含まれています。ただし、原則の多くは、以前のリリースと同じです。2.1.1 に固有の手順は、このドキュメントで明示的に示されています。

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、初期 (デフォルト) 設定の状態から起動しています。対象のネットワークが稼働中である場合には、どのようなコマンドについても、その潜在的な影響について確実に理解しておく必要があります。

表記法

ドキュメント表記の詳細は、「[シスコ テクニカル ティップスの表記法](#)」を参照してください。

概要

基本的な Cisco Unity Express 製品は、ボイスメールの一部と基本的な自動応答機能付きで出荷されます (Cisco Unity Express 2.1.1 以降には、2 つの自動応答スクリプトが含まれています)。多くのお客様が、通常の自動応答機能は制約が多すぎると感じています。Cisco Unity Express は、カスタム スクリプトを作成できるスクリプト エディタが用意されています。これらのカスタム スクリプトは、Cisco Unity Express にロードされ、通常の自動応答機能に置き換え、または通常の自動応答機能とは別に動作します。

警告：ここに示すスクリプト例は、説明を目的としています。どのようにも変更可能です。この文書の目的は、カスタマイズされたスクリプトの設計および実装プロセスを説明することです。完成した稼働スクリプトの提供を意図するものではありません。ただし、このスクリプトの使用を選択する場合、プロンプトを一つずつ録音します。添付された音声ファイルを再生して、それらがどのように聞こえるかを確認できます。ただし、[電話での管理 \(AVT \) / グリーティング管理システム](#)を使用します。

注：現時点では、Cisco Unity Expressが使用する実際のシステムスクリプトのコピーを取得することはできません。これらのスクリプトには、通常のスクリプト エディタではサポートされていないステップが含まれています。現在の Cisco Unity Express 2.1.1 のシステム スクリプトは、aa.aef、voicebrowser.aef、setmwi.aef、promptmgmt.aef、checkaltgreet.aef、xfermailbox.aef、および aasimple.aef です。

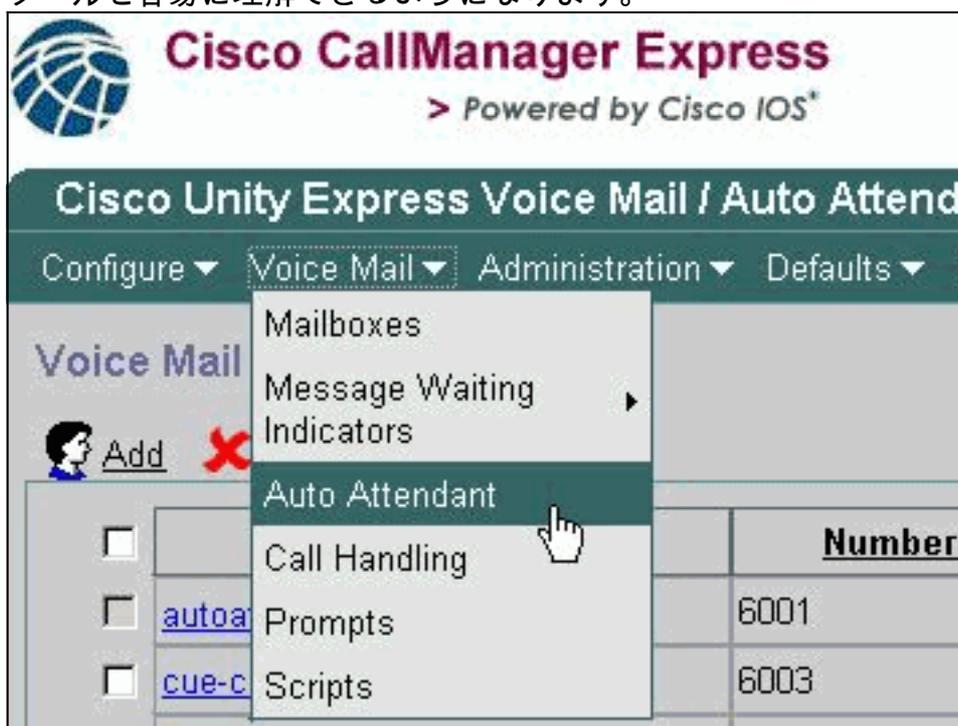
スクリプト エディタは、Cisco Unity Express とは別の PC 上でスクリプトを作成できるツールです。いったん作成したら、スクリプトを Cisco Unity Express にアップロードして設定します。

以下のリストでは、このドキュメントの概念を理解するのに役立つ基本的な用語を説明します。

- **ステップ：**スクリプト作成の基本構成要素です。各ステップは、「if」文、「Goto」など、最も基本的な実行単位です。
- **スクリプト：**シーケンスで実行される1つ以上のステップ。スクリプトは .aef 拡張子が付い

たファイルです。

- **変数**：スクリプト内の変数です。変数は、整数、ブール値、文字列など、さまざまなタイプを使用することができます。
- **パラメータ**：管理者が Cisco Unity Express Web インターフェイスを介して変数にアクセスできるようにする変数のプロパティです。たとえば、デフォルトで値 1000 を割り当てる OperatorExtension 変数がある場合、この値を 2000 に変更する必要がある場合があります。この場合、値の変更が必要になるたびに、Cisco Unity Express にスクリプト全体をロードしなくても済むように、変数は Web インターフェイスを介してアクセスできます。
- **Prompt**：再生可能な.wavファイル。プロンプトは、手動で Cisco Unity Express システムにアップロードされるか、または電話での管理 (AVT) システムを使用して録音されます (リリース 2.1.1 以前には、AVT はグリーンティング管理システム (GMS) と呼ばれていました)。アップロードされるユーザプロンプトは、すべて同じディレクトリに配置されます。これらは、GUI で [Voicemail] > [Prompts] メニュー項目を選択するか、CLI で show ccn prompts コマンドを発行して表示できます。スクリプトでは、ユーザプロンプトは、P[<promptname>] のように指定されます。システムプロンプトは、事前に録音済みであり、使用可能です。これらは SP[<promptname>]SP のように指定されます。[付録 1 に、使用可能なシステムプロンプトを一覧で示します。](#)
- **アプリケーション**：これは、すべてのプロンプトとパラメータが入力されたスクリプトです。デフォルトで、Cisco Unity Express には、ボイスメール、AVT システム、および簡単な自動応答アプリケーションが同梱されています。これらは、インストールの最後に初期化ウィザードを実行する時点ですべて設定されます。
- **トリガー**：トリガーは、特定のアプリケーションを実行する必要があることを Cisco Unity Express に指示します。たとえば、1000 をダイヤルすると、電話システムはコールを Cisco Unity Express にルーティングします。Cisco Unity Express は、コールが番号 1000 に設定されたことを確認すると、その内線番号に対するトリガーを探します。それぞれのアプリケーションが起動します。つまり、番号 1000 に対するコールが、ボイスメール、自動応答、その他に移動する必要があるかどうかわかります。同じアプリケーションに複数のトリガーを指定することができます。ボイスメール、AVT、録音済み自動応答へのトリガーなど、多数のトリガーがデフォルトで追加されます。用語がわかると、Cisco Unity Express Web 管理ツールを容易に理解できるようになります。



[Cisco Unity Express] Web ページで、[Voice Mail] カテゴリの下の [Auto Attendant] オプションが

ら用語で定義されたアプリケーションを参照します。各エントリには、アプリケーションに関連付けられた番号、参照されるスクリプト、スクリプトに関連付けられているパラメータ (アプリケーションがイネーブルであるかどうかを問いません)、および関連するポート番号が含まれています。

現在、コール処理には、ボイスメールに関連付けられた番号、組み込み自動応答、および AVT システムは含まれません。Cisco Unity Express 2.1 以降では、[Administration] > [Call-in Numbers] で、JTAPI (Cisco CallManager の場合) および SIP (Cisco CallManager Express または SRST モードの場合) の両方のすべてのトリガーが追加表示されます。現時点では、同じアプリケーションにポイントするように設定された複数のトリガーがあるかどうかを確認する唯一の方法は、CLI で発行される `show ccn trigger` コマンドを使用する方法です。

[Prompts] オプションは、システムに存在するオーディオ ファイル (.wav) を参照します。

スクリプトの選択は、システムに保存できる .aef ファイルを参照します。現時点では、システムスクリプトの他に、最大 4 つのスクリプトをアップロードできます。

この情報は、CLI から `show ccn` コマンドを使用してすぐに入手できます。この出力は、以下の各コマンドの出力の一部を表します。

```
cue-3745-44a> show ccn application
```

```
Name: customaa
Description: customaa
Script: customaa.aef
ID number: 4
Enabled: yes
Maximum number of sessions: 8
OperatorExtension: 205
MainMenu: MainMenu.wav
ClosedGreeting: ClosedGreeting.wav
InvalidExt: InvalidExt.wav
MaxRetries: 3
MaxExtension: 205
SorryGoodbye: SorryGoodbye.wav
EnterExtension: EnterExtension.wav
namePrompt: namePrompt.wav
MinExtension: 200
...
```

```
cue-3745-44a> show ccn prompts
```

```
Name: MainMenu.wav
Language: de_DE
Last Modified Date: Wed Dec 22 03:34:57 GMT+00:00 2004
Length in Bytes: 121978
Name: EnterExtension.wav
Language: de_DE
Last Modified Date: Wed Dec 22 03:34:26 GMT+00:00 2004
Length in Bytes: 21338
Name: SorryGoodbye.wav
Language: de_DE
Last Modified Date: Wed Dec 22 03:34:02 GMT+00:00 2004
Length in Bytes: 17658
Name: ClosedGreeting.wav
Language: de_DE
Last Modified Date: Wed Dec 22 03:33:00 GMT+00:00 2004
Length in Bytes: 86138
Name: InvalidExt.wav
Language: de_DE
```

```
Last Modified Date:      Wed Dec 22 03:47:07 GMT+00:00 2004
Length in Bytes:        29818
Name:                   namePrompt.wav
Language:               de_DE
Last Modified Date:     Wed Dec 22 03:40:28 GMT+00:00 2004
Length in Bytes:        22618
...
```

```
cue-3745-44a> show ccn trigger
```

```
Name:                   18955
Type:                   SIP
Application:            customaa
Locale:                 systemDefault
Idle Timeout:           10000
Enabled:                yes
Maximum number of sessions: 8
...
```

show ccn application 出力は、アプリケーション名、参照するスクリプト、それがイネーブルになっているかどうか、および処理できる同時コールの数を、それらの値を持つすべてのスクリプトパラメータとともに表示します。

show ccn prompts コマンドは、すべてのコマンド、その言語、サイズ、変更日付を一覧で示します。

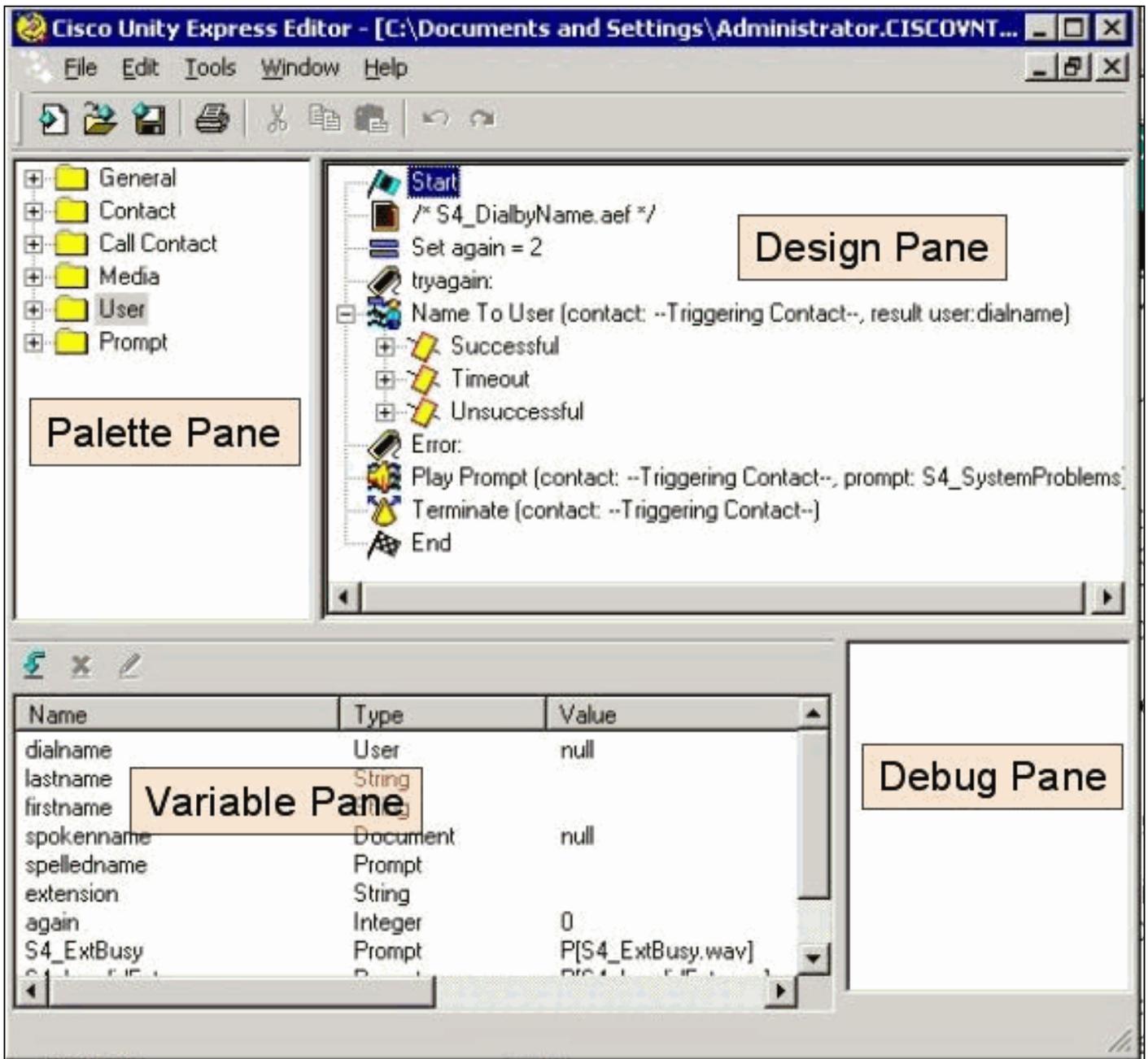
show ccn trigger 情報は、番号（「名前」）、およびコールするアプリケーション、ステータス（イネーブルまたはディセーブル）、およびセッションの数を示します。

[Cisco Unity Express スクリプト エディタの使用](#)

[スクリプト エディタ \(登録ユーザ専用\)](#) を使用して、新しいスクリプトを作成します。スクリプト エディタは、別個の PC にインストールする必要があります。

同じ場所にいくつかのサンプル スクリプトがあります。これらのスクリプトは、スクリプトの特定の操作を実行するのに役立ちます。シスコでは、エディタをインストールし、エディタの操作方法に習熟するために、いくつかのサンプル スクリプトをダウンロードすることを推奨します。

スクリプト エディタを起動すると、下図に示すようにスクリプトを開くことができます:



[Palette] ペインには、主要機能に基づいて分類された各ステップを含むフォルダがあります。このペインから、[Design] ペインにステップをドラッグして、スクリプトを作成します。[Variable] ペインで、変数を作成および編集できます。[Design] ペインにステップをドラッグしたら、ステップを右クリックして [Properties] を選択し、ステップに割り当てられているステップと変数をカスタマイズできます。Cisco Unity Express にスクリプトをアップロードする前に、まず [Tools] > [Validate] オプションを使用して、必ず検証する必要があります。[Debug] ペインにエラー出力が表示されます。

エディタを使用するときに念頭に置いて最も役立つ方法の 1 つが [Help] ヘルプ機能です。ツールバーの [Help] メニューを使用して直接アクセスするか、特定のステップのプロパティを選択するときにアクセスできます。

基本的なスクリプトの作成

[Help] メニューを使用して、いくつかのステップに習熟することができます。スクリプト エディタで、[Palette] ペインから [Design] ペインにいくつかのステップをドラッグし、任意のステップを右クリックして、[Properties] を選択します。[Help] ボタンをクリックして、特定のステップの

ヘルプを表示します。

このドキュメントでは、サンプル スクリプトの作成について説明します。また、ドキュメントでは、ユーザがカスタム自動応答を作成する時点を解決する一般的な問題の一部について説明します。ここでは、コールに回答するスクリプトを作成し、緊急代替グリーティングがあるかどうかを確認し、現在の時刻が通常の営業時間であるかどうかを確認することを目標としています。閉じると、カスタム グリーティングを再生し、オペレータへのコールを送信します。開くと、発信者が、名前によるダイヤルに 1、内線番号によるダイヤルに 2、オペレータに 0 をかけることができるメニューを再生します。内線番号によるダイヤル オプションでは、ユーザが入力できる任意の番号ではなく、指定された範囲にのみコールを転送する必要があります。

警告：このドキュメントで使用する例は、説明のみを目的としています。必要に応じてどの部分も自由に使用できますが、実際の会社の状況と正確に一致するとは限りません。特にエラー処理の分野では、このスクリプトはあまり役に立ちません。

注：www.cisco.comのサンプルスクリプトとは別に、Cisco Unity Express CDには、ここで使用されているスクリプトと同様のスクリプトが含まれています。このスクリプトはaa_sample1.aefと呼ばれます。これは、標準の付属自動応答 (2.1 以前) によく似ています。録音済みスクリプトを出発点として使用する場合は、スクリプト エディタのいくつかのバージョンをインストールするときに検出される aa.aef ファイルの代わりに、このスクリプトを使用してください。

統合方法

現在、Cisco Unity Express は、Cisco CallManager または Cisco CallManager Express への統合をサポートしています。スクリプト自体が変更されませんが、統合方法では、いくつかの若干の違いが生じる可能性があります。Cisco Unity Express から外線に論理的にコールを転送するリダイレクト ステップを実行する場合、この統合方式が必須です。Cisco CallManager Express に統合する場合 (この方法は、現時点では SRST モードでの Cisco CallManager の統合と同じです)、シグナリングは BYE/Also メッセージの形式で SIP を通じて実行されます。これは、リダイレクトがCisco Unity Expressへのコールを切断し、Cisco CallManager Express/SRSTルータに、BYEメッセージの「Also:」部分で指定された内線番号に発信者を接続するように指示することを意味します。その内線番号が無効、または到達不能の場合、発信者をドロップすることができます。これは、Cisco Unity Express には、そのコールを再度要求する方法がなく、転送が失敗したことも知らされないためです。これは、完全なブラインド転送です。コールが応答しない未登録またはビジーの内線番号にリダイレクトされる場合、そのディレクトリ番号に適用される標準の転送ルールが有効です。Cisco Unity Express は、もはやコールの一部ではありません。これは、リダイレクト自体が接続解除と同等であるため、現時点では、ビジー/無効/失敗のリダイレクトを処理するためのコードを追加するときに、利点がありません。想定されるフェールオーバー ケースはありません。

Cisco CallManager の統合では、Cisco Unity Express は、Java Telephony Application Programming Interface (JTAPI) を介して通信します。これにより、ビジー/無効/失敗のリダイレクトのステップを発生させることができます。このため、これらの状況を処理するためにスクリプトにコードを追加します。すべての Cisco CallManager が到達不能になり、システムが SRST モードになると、このドキュメントで前述した SIP シグナリングを使用します。このため、このドキュメントで説明したすべての制限事項が適用されます。

サンプル コールフロー

実稼働スクリプトを開始する前に、完全なコール フローをマッピングすることが不可欠です。この例では、すべての内線番号が 200 ~ 299 の範囲にあると仮定します (重要な点は、これらの番号がすべて 2 で始まることです)。この例では、以下に示すような手順を実行します。

1. 通話に応答します。
2. 代替グリーティングがあるかどうかを確認します。ある場合、それを再生します。
3. 休日かどうかを確認します。休日の場合、休日グリーティングを再生してメインメニューに進みます。
4. 営業時間中か、営業時間後であるかどうかを確認して、それぞれの (オープニング/クロージング) グリーティングを再生します。メインメニューに進みます。
5. メインメニューではプロンプトを再生し、ユーザが内線番号を入力できるようにする必要があります (内線番号がわかっている場合)。ディレクトリでユーザを検索したい場合は 1 を入力し、オペレータに到達したい場合は 0 を入力します。すべての有効なユーザの内線番号は 2 で始まり、内線番号によるダイヤル セクション 2 の分岐を入力します。
6. 名前によるダイヤルは、ユーザを検索し、ユーザに転送します。ゼロを入力すると、ユーザはオペレータに転送されます。
7. 内線番号によるダイヤル オプションは、ただちにあと 2 桁を収集しようとします。入力が少ない場合、メインメニューに戻ります。2 が収集されると、ユーザを検索しようとします。ユーザが存在する場合、ユーザはその内線番号に転送されます。存在しない場合、ユーザはメインメニューに移動します。
8. オペレータ機能への転送では、コールを転送します。何らかの理由で失敗した場合、メインメニューに戻ります。
9. さまざまな問題 (複数の転送の失敗など) に対する追加エラー処理シーケンスでは、後で再試行するようにユーザに指示し、切断します。

エラー処理のために、多くの操作を行うことができます。実際の実装では、完全なコールフロー (エラー処理、およびユーザの操作または選択をすべて含む) をフローチャートに描くと役立ちます。

[サンプル スクリプト](#)

開始

スクリプト エディタにスクリプトを入力します。



すべてのスクリプトは、[Start] ステップで始まります。最初に実行する処理は、コールを受信することです。その後、プロンプト (DP[500]) を再生します (0.5 秒間一時停止します)。状況によっては、公衆電話交換網 (PSTN) への音声カットスルーがすばやく実行されない場合があります。「実際の」グリーティングの前に、任意のスクリプトを再生して、グリーティングへの認識されるクリッピングを回避します。次のステップは、[On Exception Goto] です。このステップを実行すると、スクリプトの実行時に「UndefinedPromptException」タイプの例外を検索します。この特定の例外は、スクリプトが存在しないプロンプトを再生しようとするときに発生します。これは、プロンプトを再生するステップの"Continue On PromptErrors"プロパティが[No]に設定されている場合にのみ発生します。[Yes]に設定されている場合は、プロンプトは再生されず、例外はスローされません。この [On Exception Goto] ステップの使用では、管理者が実行できる必須プロンプト (メインメニューなど) が消去された場合、[Sorry] ラベルに分岐できるセクションに分岐して、後で試行するようにユーザに指示できることを確認します。その後、サブフロー

checkAltGreet.aef を呼び出します。このファイルは、常に存在するシステム スクリプトです。これは AltGreeting.wav ファイルの存在を確認します。存在する場合、それを再生します。存在しない場合、サブフローが返されます。これは、通常メッセージの前に緊急グリーティングを再生するために、GMS (現在は AVT システムと呼ばれています) と併用して使用できるので便利です。この機能の詳細については、『[Unity Expressのグリーティング管理システムと緊急代替グリーティングの設定と使用](#)』を参照してください。代替グリーティングが実際に再生するかどうか (または、ローカル リポジトリにファイルが存在するかどうかを確認する) を確認する方法はありません。スクリプトの開始で、これを使用する必要はありません。アプリケーションに応じて、たとえば、緊急代替グリーティング機能を使用して、休日プロンプトを録音することができます。このサブフローを呼び出す代わりに、休日であるかどうかのみ呼び出します。これは、サイト管理者が、(プロンプトの名前を変更するか、スクリプトの変数に割り当てるために) ブラウザまたは CLI を使用してシステムにアクセスする必要なしに、テレフォニー ユーザ インターフェイスを使用して、休日グリーティングを録音するのに役立ちます。



再生するオープニング プロンプトを決定します。休日の場合、休日グリーティングを再生します。営業時間中の場合 (Cisco Unity Express の Web 管理者のスケジュールを通じて設定可能)、オープニング グリーティング (「thanks for calling Cisco Systems」のような単純な内容を使用できます) を再生するか、さもなければクロージング グリーティングを再生します。再生するグリーティングにかかわらず、メイン メニューに直行することもできます。

注: 「Is Holiday」および「Business Hours」の手順は、2.1.1以降に固有です。Cisco Unity Express の以前のバージョンでは、[Time of Day] ステップが唯一のオプションです。このステップでは、日または日付に基づいて異なるグリーティングを再生することはできません。時刻にのみ基づいて再生されます。

[メインメニュー](#)



[Main Menu] セクションに入ります。エラー処理以降、多くの場合、発信者がメインメニューに戻されるため、カウンタが追加されます。カウンタの「attempts」が、ユーザが設定できる別の変数（デフォルトは3）である「MaxRetries」と比較されます。このカウンタに到達すると、（「Sorry you are having problems.オペレータを待ってください。」）

[Menu] ステップは、基本的なプロンプト（「If you know your party's extension please dial it any time.For spell-by-name press 1, for an operator press 0, to repeat these options, press 9.」）を再生します。この場合、独自の録音プロンプトを再生するため、この手順のMaximum Retriesパラメータを0に変更します。これは、この操作を行うオプションがあるすべての場所で実行されます。これは、0以外の番号を押した場合、タイマーが期限切れに到達するたびに「are you still there?」というシステムプロンプトが聞こえ、Timeout分岐に分岐するためです。1、2、9、または0以外の番号が押された場合、Unsuccessful分岐に到達します。この場合、プロンプト「The extension entered is invalid」を再生し、試行カウンタを増加するステップを途中で停止し、メインメニューに戻ります。

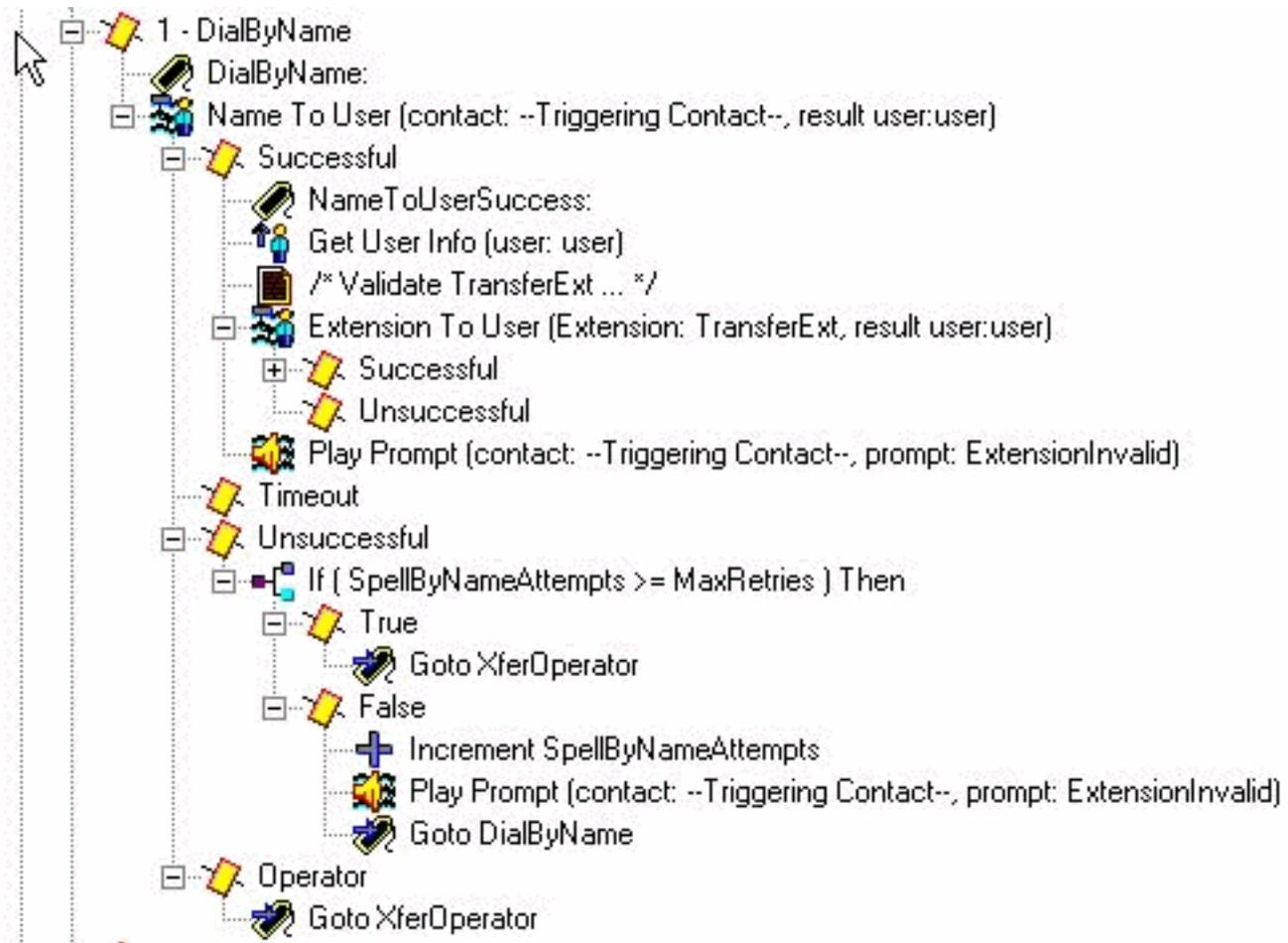
注：メディアステップ（メニューやプロンプトの再生など）に数字を入力するとすぐに音声は停止するようにするには、ステップの[Prompt]タブの[Barge In]フィールドをオンにする必要があります。[Interruptible]フィールド（[General]タブ）は、現在Cisco Unity Expressに適用できない他のステップへの割り込みに使用されます。

単一のエラー処理セクションが追加されます。「Sorry」セクションでは、メッセージ（「We are unable to transfer the call at this time.Please try again later.さようなら」）。その後、コールが接続解除されます。

これらのカスタム変数の代わりに、常にシステムグリーティングを使用することができます。システムプロンプトは常にSP [] 表記を使用し、ユーザプロンプトはP [] 表記を使用します。これを定義するときに、システムプロンプトをプロンプト変数に割り当てることはできません（変数

ペインでは、定義上ユーザの値であるためです)。いったん定義されると、スクリプトで、[Set] ステップ (または [Create Prompt] ステップのいずれか) を通じてシステム プロンプトを変数に割り当てることができます。

名前によるダイヤル (Dial By Name)



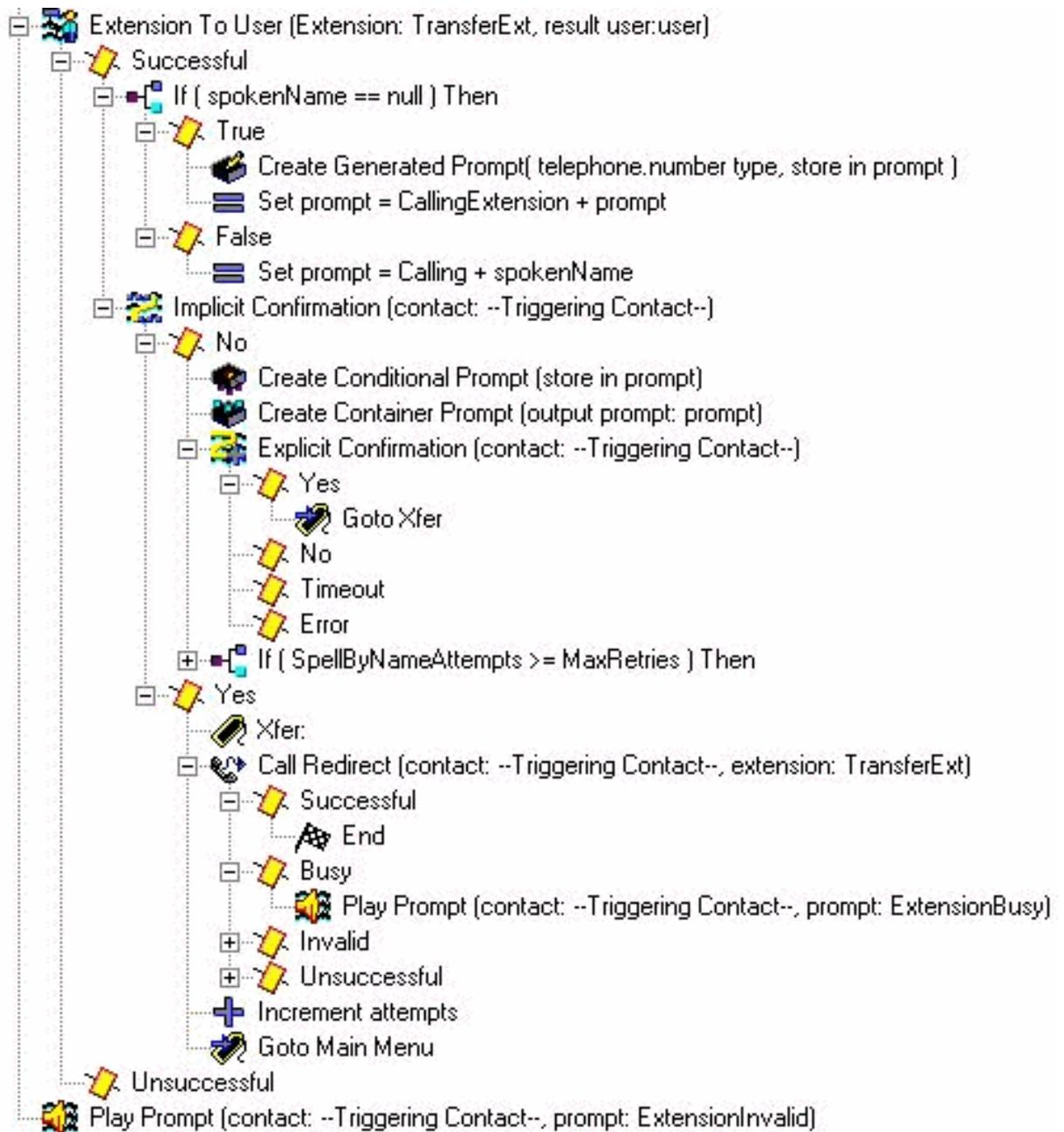
名前によるダイヤル (Dial By Name) を行う 1 つの方法は、Cisco.com からサンプル (S4_DialByName) をダウンロードし、それを Cisco Unity Express に別個のスクリプトとしてロードし、[Call Subflow] ステップを追加して起動します。この場合は、ステップは直接スクリプトに移動し、詳しく説明されます。[NameToUser] ステップでは、そのプロンプトの 1 つ (「 Spell the last name of the person you want to call, followed by the first name for letter Q press 7, and for letter Z press 9」) を設定できます。システム プロンプトをすべて削除することはできません。これらを最小限に抑えるには、まず Cancel キーをディセーブルにする必要があります (デフォルト*)。これが存在する場合、システムは、録音済みプロンプトの末尾にメッセージ 「to start over, press star」 を追加します。次に、Maximum Retries をゼロに変更し、「please try again」と 「are you still there」というメッセージが再生されなようにします。追加する別個のカウンタ変数を持つ回数を常に追跡します。削除できない 1 つのシステム プロンプトが引き続き存在します。これは、複数マッチ用のプロンプト (「More than one name was found.Select from the following...」) です。[Name To User] ステップは、常に Successful、Timeout、Unsuccessful、または Operator (ステップ プロパティで選択した場合) で終了します。Successful および Operator 分岐を除き、[Menu] ステップは途中で停止します。これは、カウンタ試行が増加し、メイン メニューに戻ることを意味します。状況に応じて、すでに選択した操作を繰り返すことができるように、別個のリトライ カウンタを装備し、ステップを追加して、[DialByName] ラベルに戻ることができます。

注：リリース2.1.1で検討が必要な不具合の1つが、Cisco Bug ID [CSCeg81385\(登録ユーザ専用\)](#)です。[NameToUser] ステップの [Maximum Retries] をゼロまたは 1 に設定する場合、端末文字を

使用する必要があります。終端文字を含まない場合、このステップの「Successful」分岐に移動しません。代わりに、一致がある場合でも、常に「Timeout」に移動します。

ユーザが名前にマッピングできる内線番号を入力しない場合、Unsuccessful 分岐は MaxRetries として新しいカウンタ (SpellByNameAttempts) をチェックします。メインメニューに入るために同じ最大リトライ回数が spell-by-name 機能を適用する必要があると仮定します。これらが異なる必要がある場合は、別個の変数を追加できます。

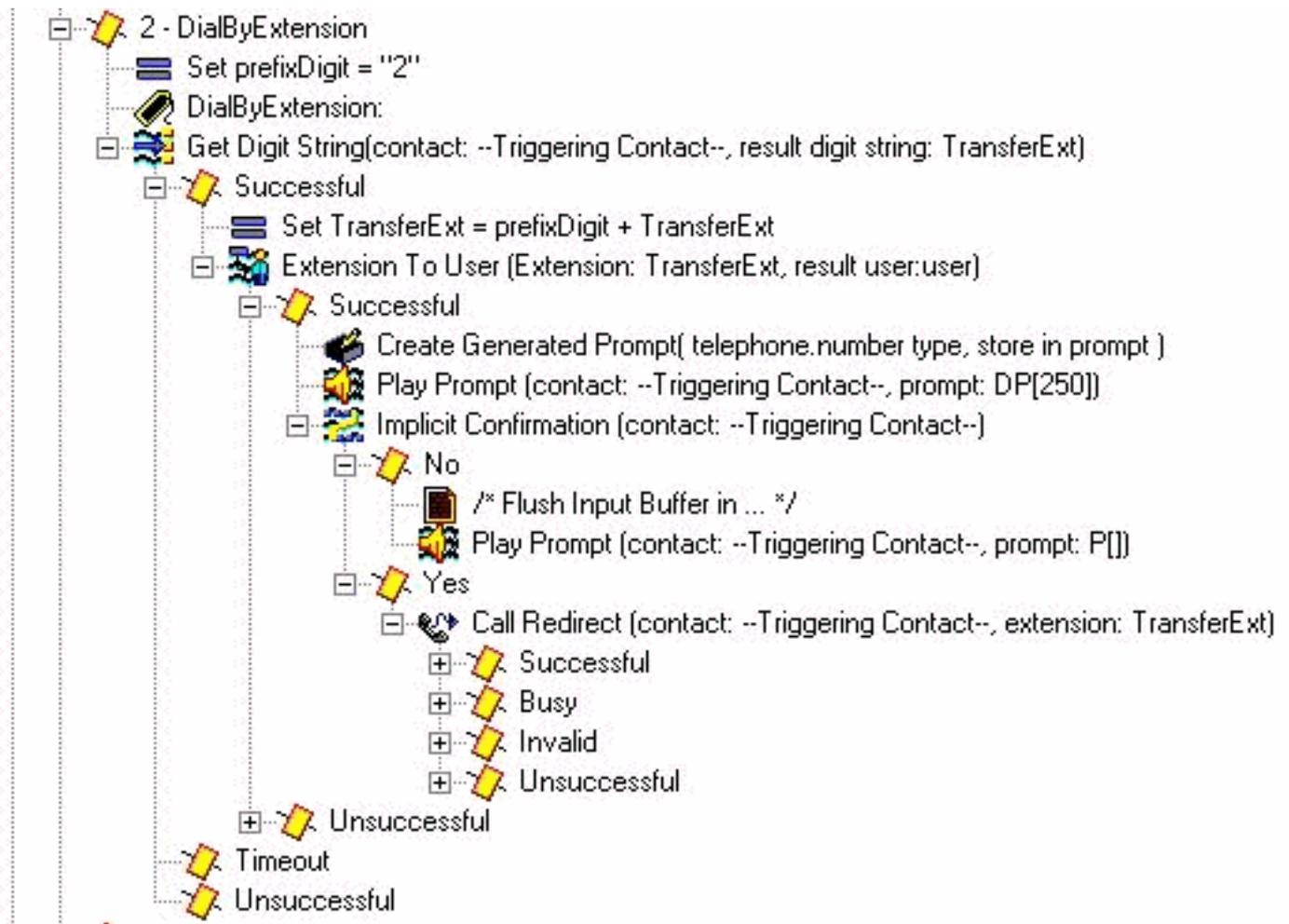
正常な [NameToUser] ステップとは、少なくとも 1 つの桁数が入力され、Cisco Unity Express で定義された単一ユーザの姓が選択されることを意味します (桁数がまったく入力されない場合にのみ、Timeout 分岐に到達します)。内線番号が一意であることを確認するには、内線番号をユーザにマッピング バックできるかどうかをチェックします。これは [ExtensionToUser] ステップで実行できます。この手順はCisco Unity Express 2.1.1で追加されました。以前は、ボイスメールシステムの実際のユーザにマッピングされるように、内線番号を指定する方法はありませんでした。実行可能な最良の方法は、ステップ「If (TransferExt < MinExtension) || (TransferExt > MaxExtension)...」または類似のステップを追加することなどにより、比較するいくつかの変数を設定することでした。



ここで、ユーザに内線番号をマッピング バックした後のスクリプトを表示されます。表示されない場合、内線番号が無効であるというプロンプトが再生されます。通常、内線番号は、何らかの理由で存在しません。Cisco CallManager Express の場合、コールを転送しようとするときにハングアップすることを意味する場合があります。これにより、発信者がすでにユーザを選択しているため、若干の混乱が生じることがあります。ただし、これにより、確認したいエラー状態はすでに明確になっています。内線番号がユーザに正常にマッピングされた場合、まず録音名があるかどうかを確認します (そのユーザを検索した場合、spokenName 変数が入力されています)。次に、プロンプト変数を **Calling + spokenName** に設定します。それ以外の場合、**Calling Extension + ユーザの内線番号のスペル** に設定します。名前 (プロンプト S[name] を再生) をスペルアウトすることもできます。再生する録音名はありません。名前または内線番号などをスペルアウト場合など、システムのスペリングは、常にシステムの音声で行われ、カスタマイズできません。[Implicit Confirmation] ステップは、ユーザが作成したばかりのプロンプトを再生します。ユーザが何も入力しない場合、手順を進めて「Yes」分岐を入力し、コールをリダイレクトします。問題ある場合、Cisco CallManager の統合のケースでは、グリーンディングを再生し、メインメニューに返信されます。発信者が何かを入力した場合、別のプロンプトを生成します。今度は、spokenName またはユーザ名のスペリングされたバージョンのいずれかでプロンプトを開始します ([Conditional Prompt] ステップを作成します)。一時停止し、プロンプト「if this is the name of the person you are calling, press 1」の再生が続きます。ここで、ユーザは YES の場合 1 を押し、または 2 * を押して待機することができます。システムは、[spell-by-name] セクションに返信するか、またはオペレータに返信します。これは、システムを通じて実行した回数によって異なります。

[内線番号によるダイヤル \(Dial By Extension \)](#)

内線番号によるダイヤル (Dial By Extension) セクションでは、最初の桁数 (この例では「2」) が設定されています。prefixDigit変数を2に設定します。この値は、先頭の数字に基づいて簡単に変更できます (または可能な先頭の数字が複数ある場合)。リーディング デイジットは、メニューの別のオプションと同じにすることはできません (この例では、名前によるダイヤルに使用される 1 など)。[Get Digit String] では、桁の固定数 (この例では 2) を収集します。このため、合計桁数を修復する必要があります。有効な内線番号が 2 または 3 で始まる場合、[Goto DialByExtension] ステップの前の [Menu] ステップのオプション 3 に [Set prefixDigit = "3"] ステップを追加できます。



この環境には3桁の内線番号があるため、Get Digit Stringステップの入力長さパラメータは2です。終端文字がなく、キャンセルキーもありません。再試行の最大回数は0です。ユーザーが有効な内線番号を入力しない場合、スクリプトは試行回数を増やしてメインメニューに戻ります。ユーザーがダイヤル番号の途中にいるため、ステップは音声（プロンプトは P[] です）を再生しません。

2桁が正常に収集された場合、すでにダイヤルされている prefixDigit（この例では「2」）を追加します（[Menu] ステップから分岐することができます）。結果として生成された TransferExt を [Extension To User] に渡します。これは、有効なユーザーであるかどうかを確認します。有効なユーザーの場合、スペルアウトされた内線番号を含むプロンプトが作成されます。[Implicit Confirmation] ステップで、内線番号の前に「calling extension」プロンプトを再生します。DialByName で実行したとおり、ユーザーを検索し、音声を再生します。ただし、この場合には必要ありません。桁がダイヤルされない場合、これは発信者が到達しようとする内線番号であり、コールをリダイレクトすると仮定します。そうでない場合、メインメニューに戻ります。[Play Prompt DP[250]] ステップの目的は、[Get Digit String] ステップには終端文字が設定されていないため、その他の入力を削除することです。この考え方は、内線電話番号「200」または「200#」がダイヤルされた場合、これを動作させたいことです。[Implicit Confirmation] ステップの実行中に # が入力されると、このステップの [No] 部分に移動します。場合によっては、[Implicit Confirmation] ステップがまったく不要である可能性があります。ただし、コールをすぐにリダイレクトする必要があります。便宜上、このステップをそのまま残します。

[オペレータ転送](#)

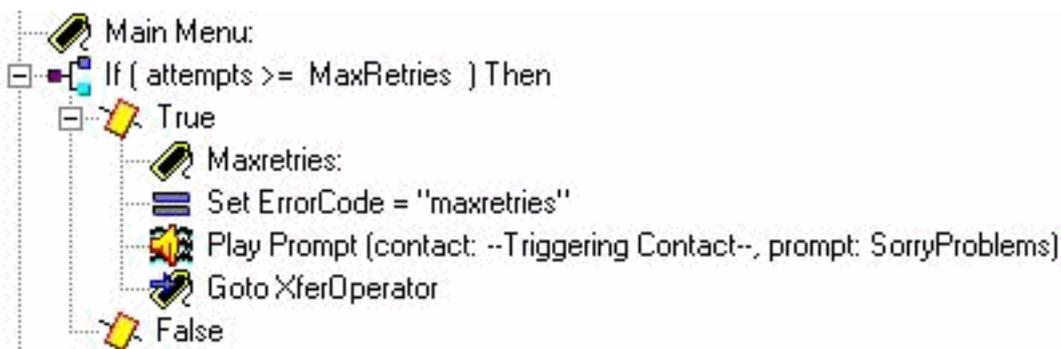


「Transferring call」プロンプトを再生し、設定済みのオペレータの内線番号にリダイレクトします。何らかの理由で失敗した場合、所有している別個のカウンタを増加し、メインメニューに戻ります。この問題の原因は、正常に再転送を行えるように、オペレータに電話機を離れて、再登録する時間を与える何らかの方法が必要になることです。何回か試行した後で（この例では、3にハードコードされています）、[Sorry] ラベルに分岐して、コールを停止します。

高度なエラー処理

多くの場所で、スクリプトが失敗する可能性があります。適切に設計されたスクリプトでのより一般的な分野の1つは、ユーザプロンプトが欠落している場合です。システムが存在しないプロンプトを再生しようとする、さまざまな問題が発生する可能性があります。第一に、プロンプトを再生するステップで、[Continue on Prompt Errors] プロパティが [Yes] に設定されている場合です。次のステップに進み、プロンプトを再生しません。[On Prompt Errors] が [No] に設定されている場合、例外が生成されます。ほとんどの場合、これは、プロンプトが存在しないか、設定されていないこと意味する「UndefinedPromptException」です。[On Exception Goto] ステップがある場合は、この例をキャッチし、システムメッセージまたは、代わりに別の形式の音声を再生できるコードのセクションに分岐できます。[On Exception Goto] ステップは、スクリプト全体（再生する各プロンプトに別個の [On Exception Goto] ステップを定義することはできません）で再生されるすべての音声に再生適用されるので、これを処理する最良の方法は、プロンプトを再生する直前にエラーコード変数を設定することです。このようにして、任意の場所に分岐し、このエラーコードに基づいて操作を行い、元の分岐に戻ることができます。

たとえば、次のようにスクリプトを変更できます。



[Maxretries] ラベルが追加されました。これを使用して、同じ場所に戻ります。[Maxretries] に ErrorCode 変数を設定します。プロンプトの再生ステップで、[Continue on Prompt Errors]が [No]に設定されていることを確認します。この処理を行えるセクションを追加してください。

ErrorCode が「Maxretries」に一致するかどうかを確認し、SorryProblems 変数 (以前に再生を試行した) をシステム メッセージ **Sorry you are having trouble 担当者におつなぎいたしますのでそのまま切らずにお待ちください。**次に、[Maxretries] ラベルに移動して、再生できるようにします。



最後のステップでは、On Exception Goto メッセージを変更して、UndefinedPromptException エラーを検索します。



このステップは、スクリプトのどこでも配置することができます。ただし、通常は冒頭またはその近くに配置されます。プロンプトを再生できる複数のステップがあります。システム プロンプトは、この文書の末尾に一覧で示されています。

この方法は、checkAltGreet.aef サブフローを呼び出す代わりに使用できます。AltGreeting.wav ファイルを再生する必要があります。エラーが存在する場合は、それを処理します。利点は、それを再生するか、または再生しないときにわかることです。

一般的ではありませんが、その他のプロンプトの例外を以下に示します。

- UndefinedPromptException
- PromptException
- UndefinedPromptGenerator
- InvalidPromptArgumentException
- UnsupportedPromptExpression

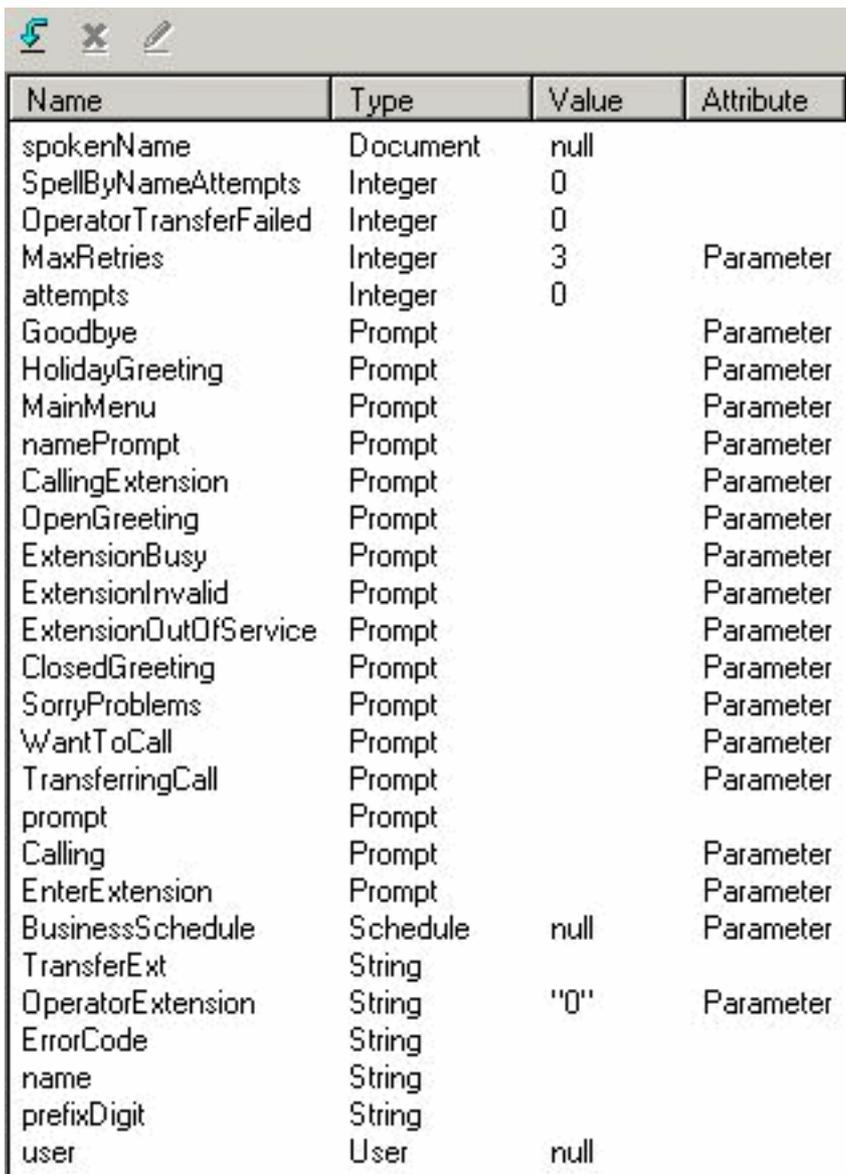
[ボイスメールへのコールの転送](#)

現時点では、コールをボイスメール ボックスに直接リダイレクトすることはできません。ボイスメールに転送する番号に送信する必要があります。これは、call-forward all をボイスメールのパイロット番号に設定するように構成された Cisco CallManager Express (Cisco CallManager の場合も可能です) のダミー番号を指定できます。Cisco Unity Express number/e.164 の番号設定を使用して、その番号をサブスクリバまたはグループに一致させます。[これを実行する方法については、「発信者を Unity Express メールボックスに転送する」を参照してください。](#)

[スクリプトの変数](#)

スクリプト内で使用するすべての変数を使用する前に、これらの変数を作成する必要があります。任意の変数を使用し、後で削除する場合は、スクリプト エディタで [Tools] > [Validate] の順に選択してから検索します。このサンプル スクリプトでは、以下の変数が使用されます。名前、タイプ、値 (変数の初期値)、および属性に注目してください。[Parameter] 属性は、変数が Web ページを介して管理者に公開されていることを意味します。Cisco Unity Express 管理者をカスタ

マイズする任意の操作には、[Parameter] 属性を指定する必要があります。



Name	Type	Value	Attribute
spokenName	Document	null	
SpellByNameAttempts	Integer	0	
OperatorTransferFailed	Integer	0	
MaxRetries	Integer	3	Parameter
attempts	Integer	0	
Goodbye	Prompt		Parameter
HolidayGreeting	Prompt		Parameter
MainMenu	Prompt		Parameter
namePrompt	Prompt		Parameter
CallingExtension	Prompt		Parameter
OpenGreeting	Prompt		Parameter
ExtensionBusy	Prompt		Parameter
ExtensionInvalid	Prompt		Parameter
ExtensionOutOfService	Prompt		Parameter
ClosedGreeting	Prompt		Parameter
SorryProblems	Prompt		Parameter
WantToCall	Prompt		Parameter
TransferringCall	Prompt		Parameter
prompt	Prompt		
Calling	Prompt		Parameter
EnterExtension	Prompt		Parameter
BusinessSchedule	Schedule	null	Parameter
TransferExt	String		
OperatorExtension	String	"0"	Parameter
ErrorCode	String		
name	String		
prefixDigit	String		
user	User	null	

これらのカスタム プロンプトをすべて録音する必要がない場合は、それぞれを編集し、[Parameter] 属性をオフにして、Web ページを介して公開されないようにできます。システム タスクが引き続き動作していることを確認するには、スクリプトの先頭に複数の [Set] ステップを追加します。たとえば、[Calling] を「SP [AA/AACalling]」に設定します。ここに示すプロンプトがすべて対応するシステム スクリプトでありとは限りません。ほとんどの場合、これらを録音することは意味があります。プロンプトを変更しない場合は、いったん録音し、Cisco Unity Express システムに特定のファイル名で存在することを確認し、変数からの [Parameter] 属性を削除します。これにより、プロンプトは固定ファイル名で再生します。ただし、変数は、管理者が誤って変更するおそれがある Web インターフェイスを介して公開されません。

[Cisco Unity Express へのスクリプトのアップロード](#)

通常、新しいアプリケーションとスクリプトを Cisco Unity Express に追加する最初のステップは、プロンプトをアップロードすることです。スクリプト パラメータを設定するには、プロンプトがシステムに存在する必要があるため、これは重要です。そうでない場合、パラメータを空白のままにして、プロンプトをアップロードし、スクリプト パラメータに戻って、編集する必要があります。プロンプトを録音し、アップロードする方法の詳細については、「[電話 \(AVT\) / グリーティング管理システムによる管理](#)」を参照してください。

スクリプトを保存し、試行する準備ができたら、まず Cisco Unity Express にアップロードする必要があります。最も簡単な方法は、スクリプト ファイルをロードするために FTP サーバが必要にならない GUI を使用することです。

次のステップを実行します。

1. 管理者権限を持つアカウントで Cisco Unity Express GUI にログインします。
2. [Voice Mail] > [Auto Attendant] の順に選択します。
3. [Add] をクリックします。
4. [Selected Automated Attendant Script] の横にある [Upload] ボタンをクリックします。
5. [Browse] をクリックします。
6. スクリプト ファイルを検索し、選択して、[OK] を押します。
7. 宛先ファイル名を変更する必要がある場合は変更します。それ以外の場合は [Upload] を押します。
8. Application Name を入力します。これは、任意の小文字とすることができ、このスクリプト ファイルに関連付けられたアプリケーションを参照するために使用されます。
9. [next] をクリックします。
10. これで、パラメータとしてマークされているスクリプトの変数がページに表示されるようになりました。これは、このページ上でスクリプトの変数を設定できることを意味します。まだプロンプト ファイルをアップロードしていない場合は、この最初の手順を確認し、後で情報を入力できます。
11. スクリプト パラメータの入力が完了したら、[Next] をクリックします。
12. アプリケーションが **Enabled** としてマークされ、同時コールの最大数 (デフォルトはライセンスによって許可された最大数) を設定し、コールイン番号を入力します。これは、この特定のアプリケーションを起動 (トリガー) する必要がある場合に呼び出される番号です。これは、主な自動応答番号、または一時的番号を指定できます。これをスキップし、後で入力することもできます。コールイン番号がないと、スクリプトをテストし、起動することはできません。注 : Cisco CallManager と SRST サポートを統合するには、複数のコールイン番号を入力する必要があります。Cisco Unity Express 2.1.1 以前では、この情報は CLI から入力する必要がありました。Cisco Unity Express 2.1 では、[Administration] > [Call-in Numbers] 画面の順に移動して、複数のコールイン番号 (トリガー) を入力できます。
13. [Finished] をクリックします。

スクリプトをアップロードし、アプリケーションを設定したら、コールをアプリケーションにルーティングできることを確認する必要があります。これは Cisco CallManager Express または Cisco CallManager のいずれかを設定する必要があることを意味します。CallManager Express では、Cisco Unity Express で設定されたパターン (トリガー番号またはコールイン番号) に一致する Cisco Unity Express をポイントするダイヤルピアの設定が関与します。Cisco CallManager では、この番号を含む CTI ルート ポイントを追加する必要があります。Cisco Unity Express にログインする JTAPI ユーザとこの CTI ルート ポイントを関連付ける必要があることを忘れないでください。この設定では、Cisco Unity Express のリポートが必要になる場合があります。ルート ポイントを追加し、ユーザと関連付けたら、リポートする必要なしに、好きなだけスクリプトを変更することができます。

スクリプトを更新すると、変数パラメータが追加または変更されていない場合は、Cisco Unity Express エディタに移動し、[Voice Mail] > [Scripts] > [Upload] の順にクリックして、スクリプトを選択できます。これにより、同一のまま残されているその他すべての設定済みパラメータが上書きされます。

プロンプトを作成する場合、実行する最も簡単な方法は、AVT システムを使用することです。コールが録音するファイル名には、日付/時刻スタンプが含まれています。ただし、Cisco Unit

Express 2.1.1 では、CLI で `ccn rename prompt` コマンドを使用することができます。以前は、プロンプトをダウンロードし、新しい名前でも再アップロードして、元の `.wav` ファイルを削除する必要がありました。以下に、いくつかの例を示します。

```
cue-3745-44a> show ccn prompts

...
Name:                UserPrompt_01032005170038.wav
Language:            en_US
Last Modified Date:  Mon Jan 03 17:00:38 EST 2005
Length in Bytes:    35098

cue-3745-44a> ccn rename prompt UserPrompt_01032005170038.wav
MyPrompt.wav language en_US
Warning! Any existing ccn applications that use this prompt will
stop working until they are manually updated with the new prompt name.
Continue? (y/n) y
cue-3745-44a>
```

トラブルシューティング

スクリプトをトラブルシューティングする場合、通常のユーザの入力、およびタイムアウト、予想されるエラー状態の両方について、さまざまなシナリオを調べ、テストする必要があります。

スクリプトをアップロードする前に、スクリプト エディタで [Tools] > [Validate] エントリにアクセスし、「Validation succeeded」メッセージが再生されるかどうか確認します。再生されない場合、デバッグ ペインにすべてのエラーが表示されます。これらのエラーをダブルクリックすると、エラー状態がスクリプトに存在する場所が表示されます。

ほとんどの場合、問題をトラブルシューティングするためには、デフォルト トレースで十分です。この操作を容易に行えるようにするには、デフォルト トレースをそのままにし、フィルタリング機能を使用して必要な操作を表示します。

テストする前に、`clear trace` コマンドを発行します。これにより、メモリ バッファまたはトレース メッセージがクリアされ、コマンド発行後のデータにはフィルタ処理された情報のみが含まれるようになります。

スクリプトを実行する場合、フィルタ処理するために最も重要な操作は EXECUTING_STEP です。この操作は、実行するすべてのステップを示します。以下に、いくつかの例を示します。

```
cue-3745-44a> clear trace

cue-3745-44a> /*** NOW THE TEST CALL IS MADE ***/

cue-3745-44a> show trace buffer long | include EXECUTING_STEP

3119 12/28 17:05:33.955 ACCN APMG 0 EXECUTING_STEP:Executing a step:
Application=App[name=customaa,type=Cisco Script Application,id=4,
desc=customaa,enabled=true,max=8,valid=true,optional=
[cfgVars=[Lcom.cisco.wfapi.util.WFNameValuePair;@14efaa66,script=customaa21.aef]],
Task id=17,000,000,007,Step id=0,Step Class=com.cisco.wfframework.steps.core.
StepStart,Step Description=Start
...

```

この出力例では、いくつかの冗長な情報が途中で削除され、日付/時刻タイムスタンプの先頭が切り取られています。ほとんどの場合、最も重要な情報は各行の末尾にあります。

注：これらの出力行の多くは、スペース上の理由により2行目と3行目に表示されます。

```
5:33.956 Step id=529,Step Class=com.cisco.wfframework.steps.core.StepComment,
Step Description=/* Assumptions: User extensions ... */
5:33.957 Step id=3,Step Class=com.cisco.wf.steps.ivr.AcceptStep,Step
Description=Accept (contact: --Triggering Contact--)
5:33.977 Step id=129,Step Class=com.cisco.wf.steps.ivr.OutputStep,Step
Description=Play Prompt (contact: --Triggering Contact--, prompt: DP[500])
5:34.461 Step id=2,190,Step Class=com.cisco.wfframework.steps.core
.StepOnExceptionGoto,Step Description=On Exception(UndefinedPromptException)
Goto Error
5:34.463 Step id=1,649,Step Class=com.cisco.wfframework.steps.core
.StepComment,Step Description=/* Check for Alternate Greeting. ... */
5:34.464 Step id=1,650,Step Class=com.cisco.wfframework.steps.core
.StepCallSubflow,Step Description=Call Subflow -- checkAltGreet.aef
5:34.467 Step id=0,Step Class=com.cisco.wfframework.steps.core.StepStart,
Step Description=Start
5:34.468 Step id=78,Step Class=com.cisco.wf.steps.ivr.GetContactInfoStep,
Step Description=Get Contact Info (contact: --Triggering Contact--)
5:34.469 Step id=79,Step Class=com.cisco.wfframework.steps.core
.StepCreateHost,Step Description=aType = new com.cisco
.aesop.AltGreetType(language)
5:34.473 Step id=56,Step Class=com.cisco.wfframework.steps.core
.StepHost,Step Description=pExist = aType.isEnabled()
5:34.477 Step id=5,Step Class=com.cisco.wfframework.steps.core
.StepIf,Step Description=If ( pExist == true
5:34.478 Step id=3,Step Class=com.cisco.wfframework.steps.core
.StepEnd,Step Description=End
5:34.480 Step id=510,Step Class=com.cisco.wf.steps.ivr.HolidayStep,
Step Description=Is Holiday (Date: --Current Date--)
5:34.487 Step id=512,Step Class=com.cisco.wf.steps.ivr.BusinessHoursStep,
Step Description=Business Hours (date: --Current Date--,
time: --Current Time--, schedule: BusinessSchedule)
5:34.527 Step id=1,659,Step Class=com.cisco.wfframework.steps.core
.StepLabel,Step Description=Office Open:
5:34.529 Step id=1,660,Step Class=com.cisco.wf.steps.ivr.OutputStep,
Step Description=Play Prompt (contact:
--Triggering Contact--, prompt: OpenGreeting)
5:35.722 Step id=1,669,Step Class=com.cisco.wfframework.steps.core
.StepLabel,Step Description=Main Menu:
5:35.723 Step id=732,Step Class=com.cisco.wfframework.steps.core
.StepIf,Step Description=If ( attempts >= MaxRetries ) Then
5:35.724 Step id=2,195,Step Class=com.cisco.wfframework.steps.core
.StepAssign,Step Description=Set ErrorCode = "mainmenu"
5:35.726 Step id=2,259,Step Class=com.cisco.wf.steps.ivr.MenuStep,
Step Description=Menu (contact: --Triggering Contact--, prompt: MainMenu)
5:35.730 Step id=2,294,Step Class=com.cisco.wf.steps.ivr.ParseInputStep,
Step Description=Get Digit String(contact:
--Triggering Contact--, result digit string: TransferExt)
5:36.197 Step id=2,295,Step Class=com.cisco.wfframework.steps.core
.StepAssign,Step Description=Set TransferExt = "2" + TransferExt
5:36.198 Step id=2,296,Step Class=com.cisco.wf.steps.ivr.ExtensionToAddressStep,
Step Description=Extension To User (Extension: TransferExt, result user:user)
5:36.214 Step id=2,297,Step Class=com.cisco.prompt.steps.CreateGeneratedPromptStep,
Step Description=Create Generated Prompt( telephone.number type, store in prompt )
5:36.229 Step id=2,298,Step Class=com.cisco.wf.steps.ivr.ImplicitConfirmStep,
Step Description=Implicit Confirmation (contact: --Triggering Contact--)
5:41.971 Step id=2,301,Step Class=com.cisco.wf.steps.ivr.RedirectStep,
Step Description=Call Redirect
(contact: --Triggering Contact--, extension: TransferExt)
```

ここで、実行する各ステップを確認できます。式を評価する方法確認したり、ユーザ入力を確認したりすることはできません。スクリプトが目の前にある場合、入力した桁数の知識により、ス

クリプトが分岐する場所以降の手順を実行することができます (この場合、ユーザが電話をかけ、内線番号をダイヤルし、転送されます)。

桁数を表示するには、フィルタを使用します。

```
cue-3745-44a> show trace buffer long | include "process digit"  
3119 12/28 17:05:35.728 ACCN CMTS 0 process digit 2 tag=2  
3119 12/28 17:05:35.943 ACCN CMTS 0 process digit 0  
3119 12/28 17:05:36.195 ACCN CMTS 0 process digit 1
```

この出力から、内線番号 201 にダイヤルされることがわかります。

CUE と CUE スクリプト エディタのバージョンが異なって、GUI または CLI から AA スクリプトをアップロードすると、Upload failed error 両方のバージョンが異なっている場合、CUE と同じバージョンの CUE エディタを使用してスクリプトを作成し直す必要があります。または、CUE を CUE スクリプト エディタのバージョンにアップグレードし、スクリプトを実行します。

付録

付録 1 : システム プロンプト

下表は、使用可能なシステム プロンプトを示します。

注 : Cisco Unity Expressボイスメールアプリケーションで再生されるすべてのプロンプトを、カスタムスクリプトで再利用することはできません。プロンプトの一部は、Cisco Unity Express スクリプト エディタを使用してカスタマイズできないボイスメール スクリプトだけで使用できます。

システム プロンプト	録音された音声
SP[AA/AAWelcome]	「Welcome to Automated Attendant」
SP[AA/AAMain Menu]	「To enter the phone number of the person you are trying to reach press 1, To enter the name of the person you are trying to reach press 2, To transfer to operator press 0」
SP[AA/AAEnter Extn]	「Please enter the phone number and press the # key」
SP[AA/AACallingExtn]	「Calling extension」
SP[AA/AAPhoneReach]	「The phone number you are trying to reach」
SP[AA/AAOutOfServicePhone]	「Is currently out of service」
SP[AA/AANamedDial]	「Spell the last name of the person you want to call followed by the first name.For letter Q, press 7, for Z press 9」
SP[AA/AACalling]	「Calling」

ng]	
SP[AA/AASorry]	「Sorry You are having trouble.Please stay on the line and some will be with you shortly.」
SP[AA/AAWant 2Call]	「If this is the name of the person you are calling, press 1, to start over, press *」
SP[AA/still_ther e]	「Are you still there?」

付録 2 : デフォルト トレースのリセット

CLI を使用して、システム (Cisco Unity Express 2.1.x 以前) をデフォルト トレースにリセットするには、まず **no trace all** コマンドを使用して、すべてのトレースをディセーブルにしてから、これを CLI に貼り付けます。

```

trace ccn engine debug
trace ccn libldap debug
trace ccn subsystemappl debug
trace ccn managerappl debug
trace ccn managerchannel debug
trace ccn subsystemjtapi debug
trace ccn subsystemsip debug
trace ccn stacksip debug
trace ccn subsystemhttp debug
trace ccn vbrowsercore debug
trace ccn subsystemcmt debug
trace ccn libmedia debug
trace ccn managercontact debug
trace ccn stepcall debug
trace ccn stepmedia debug
trace config-ccn sip-subsystem debug
trace config-ccn jtapi-subsystem debug
trace config-ccn sip-trigger debug
trace config-ccn jtapi-trigger debug
trace config-ccn http-trigger debug
trace config-ccn group debug
trace config-ccn application debug
trace config-ccn script debug
trace config-ccn prompt debug
trace config-ccn miscellaneous debug
trace voicemail database query
trace voicemail database results
trace voicemail database transaction
trace voicemail database connection
trace voicemail database execute
trace voicemail mailbox login
trace voicemail mailbox logout
trace voicemail mailbox send
trace voicemail mailbox save
trace voicemail mailbox receive
trace voicemail mailbox delete
trace voicemail message create
trace voicemail message dec
trace voicemail message delete
trace voicemail message get
trace voicemail message inc
trace webinterface initwizard inittrace ccn engine debug

```

```
trace ccn libldap debug
trace ccn subsystemappl debug
trace ccn managerappl debug
trace ccn managerchannel debug
trace ccn subsystemjtapi debug
trace ccn subsystemsip debug
trace ccn stacksip debug
trace ccn subsystemhttp debug
trace ccn vbrowsercore debug
trace ccn subsystemcmt debug
trace ccn libmedia debug
trace ccn managercontact debug
trace ccn stepcall debug
trace ccn stepmedia debug
trace config-ccn sip-subsystem debug
trace config-ccn jtapi-subsystem debug
trace config-ccn sip-trigger debug
trace config-ccn jtapi-trigger debug
trace config-ccn http-trigger debug
trace config-ccn group debug
trace config-ccn application debug
trace config-ccn script debug
trace config-ccn prompt debug
trace config-ccn miscellaneous debug
trace voicemail database query
trace voicemail database results
trace voicemail database transaction
trace voicemail database connection
trace voicemail database execute
trace voicemail mailbox login
trace voicemail mailbox logout
trace voicemail mailbox send
trace voicemail mailbox save
trace voicemail mailbox receive
trace voicemail mailbox delete
trace voicemail message create
trace voicemail message dec
trace voicemail message delete
trace voicemail message get
trace voicemail message inc
trace webinterface initwizard init
```

関連情報

- [音声に関する技術サポート](#)
- [音声とユニファイド コミュニケーションに関する製品サポート](#)
- [Cisco IP Telephony のトラブルシューティング](#)
- [テクニカルサポート - Cisco Systems](#)