

Guia de início rápido de scripts personalizados do Unity Express

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Conventions](#)

[Overview](#)

[Usar o Editor de scripts do Cisco Unity Express](#)

[Criar um script básico](#)

[Métodos de integração](#)

[Exemplo de fluxo de chamada](#)

[Exemplo de script](#)

[Fazer upload de um script para o Cisco Unity Express](#)

[Troubleshoot](#)

[Appendix](#)

[Apêndice 1 - Prompts do sistema](#)

[Apêndice 2 - Redefinir rastreamentos padrão](#)

[Informações Relacionadas](#)

[Introduction](#)

Este documento fornece informações sobre o sistema Cisco Unity Express (CUE). Especificamente, este documento oferece uma prévia para a criação de um aplicativo de autoatendimento personalizado no produto.

Para obter mais detalhes sobre o Cisco Unity Express, consulte o [Guia do Editor de Scripts do Cisco Unity Express](#). Este documento destina-se a administradores que precisam de um guia básico para começar a usar o Cisco Unity Express Editor e manter scripts no produto Cisco Unity Express.

Observação: o suporte personalizado a scripts não está disponível através do Suporte Técnico Cisco regular. Para perguntas, comentários e ajuda com um script personalizado, envie um e-mail para ask-cue-editor@external.cisco.com.

[Prerequisites](#)

[Requirements](#)

O leitor precisa estar familiarizado com a administração e a configuração do Cisco Unity Express através da interface de linha de comando (CLI) ou GUI.

O Editor de scripts do Cisco Unity Express (CUEEditor2.1.1.exe) e o script de exemplo que este documento usa (CUE-AA-S6-AASeemplo.zip) estão disponíveis no [Centro de Download de Software do Cisco Unity Express 2.1.1](#).

[Componentes Utilizados](#)

As informações neste documento são baseadas no Cisco Unity Express versão 2.1.1. O exemplo contém etapas disponíveis somente no Cisco Unity Express 2.1.1. Embora muitos dos princípios sejam os mesmos para versões anteriores. As etapas específicas para 2.1.1 são apontadas explicitamente neste documento.

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, make sure that you understand the potential impact of any command.

[Conventions](#)

For more information on document conventions, refer to the [Cisco Technical Tips Conventions](#).

[Overview](#)

O produto básico do Cisco Unity Express é fornecido com uma peça de correio de voz e um atendimento automático básico (no Cisco Unity Express 2.1.1 e posterior, há dois scripts de atendimento automático). Muitos clientes consideram o atendimento automático padrão muito limitado. O Cisco Unity Express tem um editor de scripts que permite a criação de scripts personalizados. Esses scripts personalizados podem ser carregados no Cisco Unity Express e substituir (ou trabalhar em adição) o atendimento automático normal.

Aviso: o exemplo de script mostrado aqui é para fins de ilustração. Pode ser modificado de qualquer maneira. O objetivo deste documento é ilustrar o processo de projeto e implementação de um script personalizado. Não se destina a fornecer um script de produção acabado. No entanto, se você optar por usar esse script, registre, no mínimo, cada um dos prompts. Você pode ouvir os arquivos de áudio anexados para ouvir o som deles. Mas use a [Administração pelo Sistema de Gerenciamento de Telefone \(AVT\)/Saudações](#).

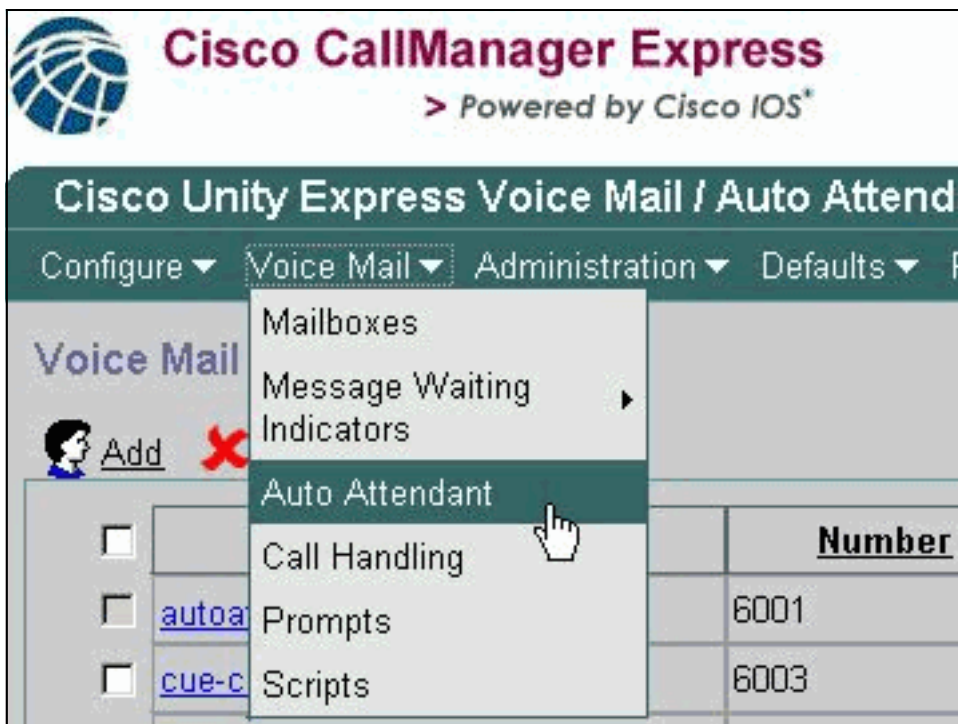
Observação: no momento, não é possível obter uma cópia dos scripts de sistema reais que o Cisco Unity Express usa. Esses scripts têm etapas que o editor de scripts regular não suporta. Os scripts de sistema do Cisco Unity Express 2.1.1 são aa.aef, voicebrowser.aef, setmwi.aef, promptgmt.aef, checkaltgreet.aef, xfermailbox.aef e aasimple.aef.

O editor de scripts é uma ferramenta que permite criar scripts em um PC separado do Cisco Unity Express. Depois de criados, os scripts são carregados e configurados no Cisco Unity Express.

Esta lista explica algumas terminologias básicas para ajudá-lo a entender os conceitos neste documento.

- **Etapas** — O bloco básico de construção para a criação de scripts. Cada etapa é a unidade executável mais básica, como uma instrução "if", um "Goto" e assim por diante.

- **Script** — Uma ou mais etapas executadas em sequência. Um script é um arquivo com uma extensão .aef.
- **Variável** — São variáveis em um script. Variáveis podem ser de tipos diferentes, como Integer, Booleano, String e assim por diante.
- **Parâmetro** — Esta é uma propriedade de uma variável de modo que ela seja exposta ao administrador através da interface da Web do Cisco Unity Express. Por exemplo, se você tiver uma variável OperatorExtension que deseja atribuir o valor 1000 por padrão, às vezes ela precisa ser alterada para 2000. Nesse caso, a variável é exposta através da interface da Web para que o script inteiro não precise ser carregado no Cisco Unity Express novamente cada vez que o valor precisar ser alterado.
- **Prompt** — Um arquivo .wav que pode ser reproduzido. Ele é carregado manualmente no sistema Cisco Unity Express ou gravado por meio do sistema de administração via telefone (AVT) (antes da versão 2.1.1, o AVT era chamado de Sistema de gerenciamento de saudações (GMS)). Todos os prompts de usuário que são carregados são colocados no mesmo diretório. Eles são visíveis na GUI através do item de menu **Correio de voz > Avisos** ou na CLI através do comando **show ccn prompts**. Em um script, os prompts do usuário são especificados como P[<promptname>]. Os prompts do sistema são pré-gravados e podem ser usados. São especificados como SP[<promptname>]. [O Apêndice 1](#) lista os prompts disponíveis do sistema.
- **Aplicativo** — Este é o script com todos os prompts e parâmetros preenchidos. Por padrão, o Cisco Unity Express é enviado com o correio de voz, o sistema AVT e um aplicativo de atendimento automático simples, todos configurados quando você executa o Assistente de inicialização no final de uma instalação.
- **Disparador** — O disparador informa ao Cisco Unity Express que um determinado aplicativo precisa ser executado. Por exemplo, quando você disca 1000, o sistema telefônico (Cisco CallManager ou Cisco CallManager Express) encaminha a chamada para o Cisco Unity Express. Quando o Cisco Unity Express vê que uma chamada é feita para o número 1000, ele procura um gatilho para esse ramal. O respectivo aplicativo é então iniciado. Em outras palavras, ele sabe se uma chamada para 1000 precisa ir para o correio de voz, um atendedor automático ou algo mais. Você pode ter vários acionadores para o mesmo aplicativo. Há vários gatilhos que são adicionados por padrão, como disparadores para o correio de voz, AVT e atendimento automático enlatado. Quando você entende a terminologia, torna a ferramenta de administração da Web Cisco Unity Express mais fácil de



entender.

Na página da Web do Cisco Unity Express, a opção Atendimento automático na categoria Correio de voz se refere aos aplicativos conforme definido na terminologia. Cada entrada contém o número associado ao aplicativo, o script a que ele faz referência, todos os parâmetros associados ao script, se o aplicativo está ou não habilitado e o número de portas associadas.

Atualmente, o Tratamento de chamadas contém apenas os números associados para o correio de voz, o Atendimento automático integrado e o sistema AVT. No Cisco Unity Express 2.1 e posterior, em **Administration > Call-in Numbers** há uma exibição adicional de todos os disparadores, tanto JTAPI (para o Cisco CallManager) como SIP (para o Cisco CallManager Express ou o modo SRST). Atualmente, a única maneira de ver se há vários disparadores configurados que apontam para o mesmo aplicativo é com o comando **show ccn trigger** emitido na CLI.

A opção Prompts se refere aos arquivos de áudio (.wav) existentes no sistema.

A seleção Scripts se refere aos arquivos .aef que você pode armazenar no sistema. Atualmente, você pode carregar até quatro scripts além dos scripts do sistema.

Na CLI, essas informações estão prontamente disponíveis usando os comandos **show ccn**. Esta saída representa partes de cada saída de comando:

```
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
...
```

A saída **show ccn application** exibe o nome do aplicativo, o script ao qual ele se refere, se ele está ou não habilitado e quantas chamadas simultâneas ele pode tratar, juntamente com todos os parâmetros de script com seus valores.

O comando **show ccn prompts** lista todos os comandos, o idioma, o tamanho e a data modificada.

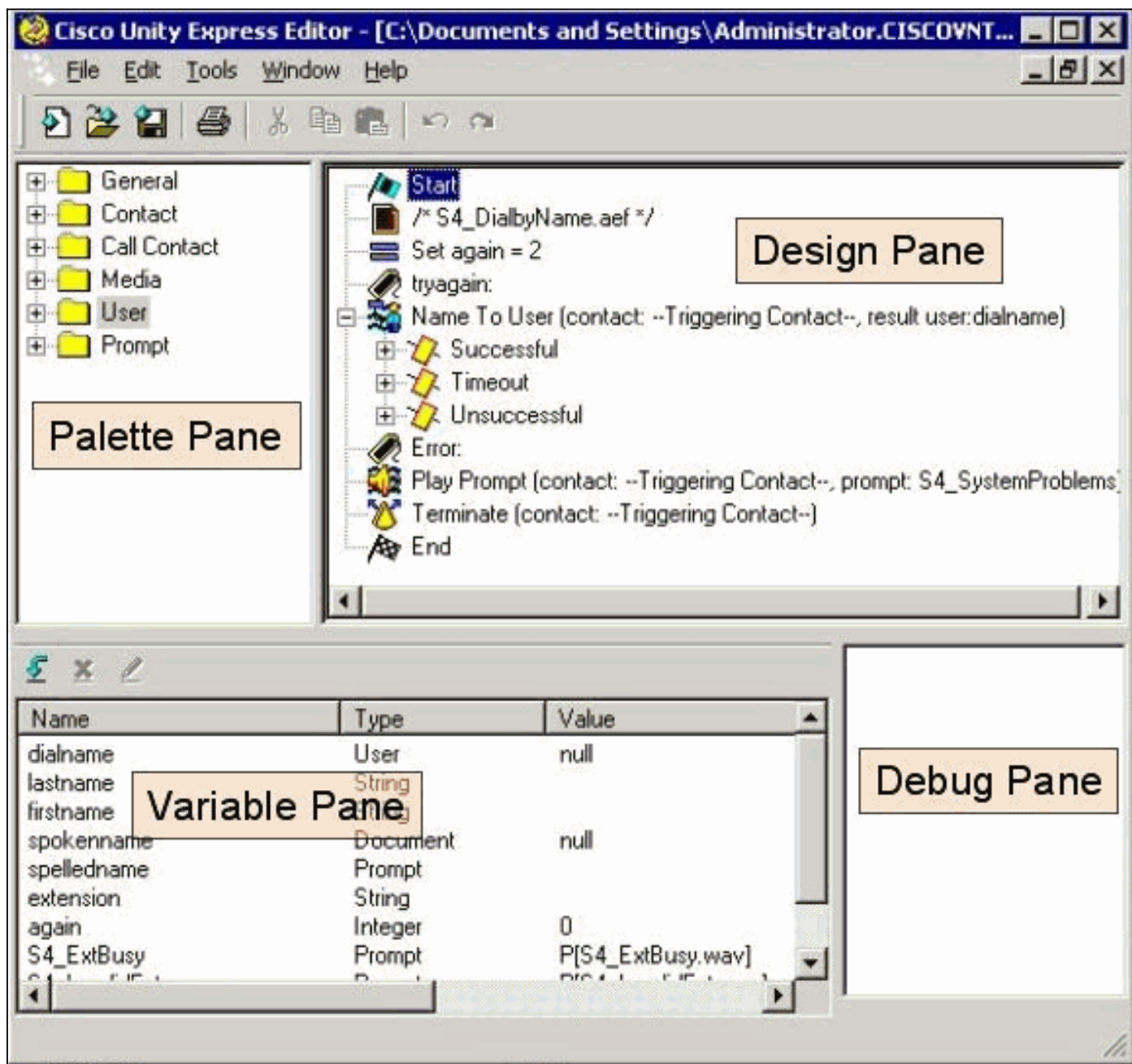
As informações do **acionador show ccn** fornecem o número ("Nome"), bem como o aplicativo que ele chama, o status (habilitado/desabilitado) e o número de sessões.

[Usar o Editor de scripts do Cisco Unity Express](#)

Use o [Editor de scripts](#) (somente clientes [registrados](#)) para criar um novo script. Ele deve ser instalado em um PC separado.

No mesmo local estão vários scripts de exemplo. Esses scripts ajudam a aprender como fazer operações específicas em um script. A Cisco recomenda que você instale o Editor e faça o download de alguns exemplos de scripts para se familiarizar com a maneira como o Editor funciona.

Quando o Editor de scripts for iniciado, você poderá abrir um script conforme mostrado neste gráfico:



O painel Paleta contém pastas com cada etapa, categorizadas com base em sua função principal. A partir daí, arraste as etapas até o painel Design para criar um script. No painel Variável, você pode criar e editar variáveis. Depois de arrastar uma etapa para o painel Design, clique com o botão direito do mouse em uma etapa, selecione **Propriedades** e personalize a etapa e as variáveis atribuídas à etapa. Antes de carregar um script no Cisco Unity Express, você sempre precisa validá-lo primeiro por meio da opção **Ferramentas > Validar**. Qualquer saída de erro é exibida no painel de depuração.

Uma das coisas mais úteis a serem lembradas ao usar o Editor é a função Ajuda. Você pode acessá-lo diretamente pelo menu Ajuda da barra de ferramentas ou ao selecionar uma propriedade de etapa específica.

[Criar um script básico](#)

Você pode usar o menu Ajuda para se familiarizar com algumas das etapas. No Editor de scripts, arraste algumas etapas da paleta para o painel de design e clique com o botão direito do mouse em uma e selecione **Propriedades**. Em seguida, clique no botão **Ajuda** e obtenha ajuda sobre essa etapa específica.

Este documento descreve a criação de um script de exemplo. O documento também discute alguns dos problemas comuns que as pessoas tentam resolver ao criar um atendimento automático personalizado. O objetivo aqui é criar um script que atenda uma chamada, verifique se há uma saudação alternativa de emergência e se a hora atual é o horário comercial normal ou não. Se estiver fechada, reproduza uma saudação personalizada e envie a chamada para o operador. Se estiver aberto, reproduza um menu que permita ao chamador discar 1 para discar por nome, 2 para discar por ramal ou 0 para um operador. A opção de discagem por ramal só deve transferir a chamada para um intervalo especificado, não para qualquer número que você possa digitar.

Aviso: o exemplo que este documento usa é apenas para fins de ilustração. Embora você esteja livre para usar as partes que desejar, provavelmente não corresponde exatamente à situação de sua empresa. Especialmente nas áreas de tratamento de erros, esse roteiro não tem muita utilidade.

Observação: além dos exemplos de scripts em www.cisco.com, o CD do Cisco Unity Express contém um script similar ao usado aqui, chamado `aa_sample1.aef`. É muito semelhante ao atendimento automático regular (anterior à 2.1). Se você planeja usar o script enlatado como um ponto de partida, use este script em vez do arquivo `aa.aef` que é encontrado quando você instala algumas versões do Editor de scripts.

Métodos de integração

O Cisco Unity Express atualmente oferece suporte à integração com o Cisco CallManager ou o Cisco CallManager Express. Embora o script em si não mude, o método de integração pode resultar em pequenas diferenças. Ao executar uma etapa de redirecionamento, que transfere logicamente a chamada do Cisco Unity Express para um ramal externo, esse método de integração é essencial. Quando integrada ao Cisco CallManager Express, que é a mesma para uma integração do Cisco CallManager atualmente no modo SRST, a sinalização é feita através do SIP na forma de uma mensagem BYE/Também. Isso significa que um redirecionamento desliga a chamada para o Cisco Unity Express e diz ao roteador Cisco CallManager Express/SRST para conectar o chamador ao ramal especificado na parte "Também:" da mensagem BYE. Se esse ramal for inválido ou não puder ser alcançado, o chamador poderá ser descartado. Isso porque não há como o Cisco Unity Express recuperar essa chamada e ser notificado de que a transferência falhou. É uma verdadeira transferência cega. Se a chamada for redirecionada para um ramal ou não registrado ou ocupado que não atenda, as regras normais de desvio de chamadas que se aplicam a esse número de diretório estarão em vigor. O Cisco Unity Express não faz mais parte da chamada. Isso também significa que quando você script as etapas de Redirecionamento, atualmente não há grande benefício quando você adiciona código para lidar com redirecionamentos ocupados/inválidos/malsucedidos, já que o próprio redirecionamento é equivalente a uma desconexão. Não há caso de falha possível.

Para integrações do Cisco CallManager, o Cisco Unity Express comunica-se através da JTAPI (Java Telephony Application Programming Interface). Isso permite que etapas de redirecionamento ocupadas/inválidas/malsucedidas ocorram. Portanto, adicione código ao script para tratar dessas situações. Quando todos os Cisco CallManagers se tornam inalcançáveis e o sistema cai no modo SRST, isso usa a sinalização SIP discutida anteriormente neste documento.

Portanto, todas as limitações descritas neste documento se aplicam.

Exemplo de fluxo de chamada

Antes de começar com um script de produção, é imperativo mapear o fluxo de chamadas completo. Para este exemplo, suponha que todas as extensões estejam no intervalo 200-299 (o ponto importante é que todas começam com 2). Para este exemplo, você tem algo como este procedimento:

1. Atenda a chamada.
2. Verifique se há uma saudação alternativa. Se houver, jogue.
3. Verifique se é feriado. Em caso afirmativo, reproduza uma saudação de feriado e vá para o menu principal.
4. Verifique se está durante ou após o horário comercial e reproduza a respectiva saudação (aberta/fechada). Vá para o menu principal.
5. O menu principal precisa reproduzir um prompt e permitir que os usuários digitem um ramal a qualquer momento, se o conhecerem. Eles podem inserir 1 se quiserem procurar um usuário no diretório ou 0 se quiserem acessar o operador. Como todos os ramos de usuário válidos começam com 2, digite 2 ramificações para a seção discagem por ramal.
6. A discagem por nome procura usuários e os transfere para eles. O usuário é transferido para o operador se zero for inserido.
7. A opção dial-by-extension tenta coletar imediatamente mais dois dígitos. Se menos for digitado, ele volta para o menu principal. Se dois forem coletados, ele tentará procurar o usuário. O usuário é transferido para esse ramal se o usuário sair, caso contrário, ele vai para o menu principal.
8. A transferência para a função operador transfere a chamada. Se falhar por algum motivo, ele volta para o menu principal.
9. Uma sequência de tratamento de erros adicional para vários problemas (como várias transferências malsucedidas) que instrui o usuário a tentar novamente mais tarde e desliga.

Muito pode ser feito para tratar erros. Para uma implementação real, provavelmente ajuda a mapear o fluxo de chamadas completo (isso inclui o tratamento de erros e cada interação ou escolha do usuário) em um fluxograma.

Exemplo de script

Iniciar

Insira o script no Editor de scripts.



Todos os scripts começam com a etapa Iniciar. A primeira coisa a fazer é aceitar a chamada. Depois disso, reproduza um prompt (DP[500]) que é uma pausa de 0,5 segundo. Em algumas situações, o cut-through de áudio para a rede telefônica pública comutada (PSTN) não acontece

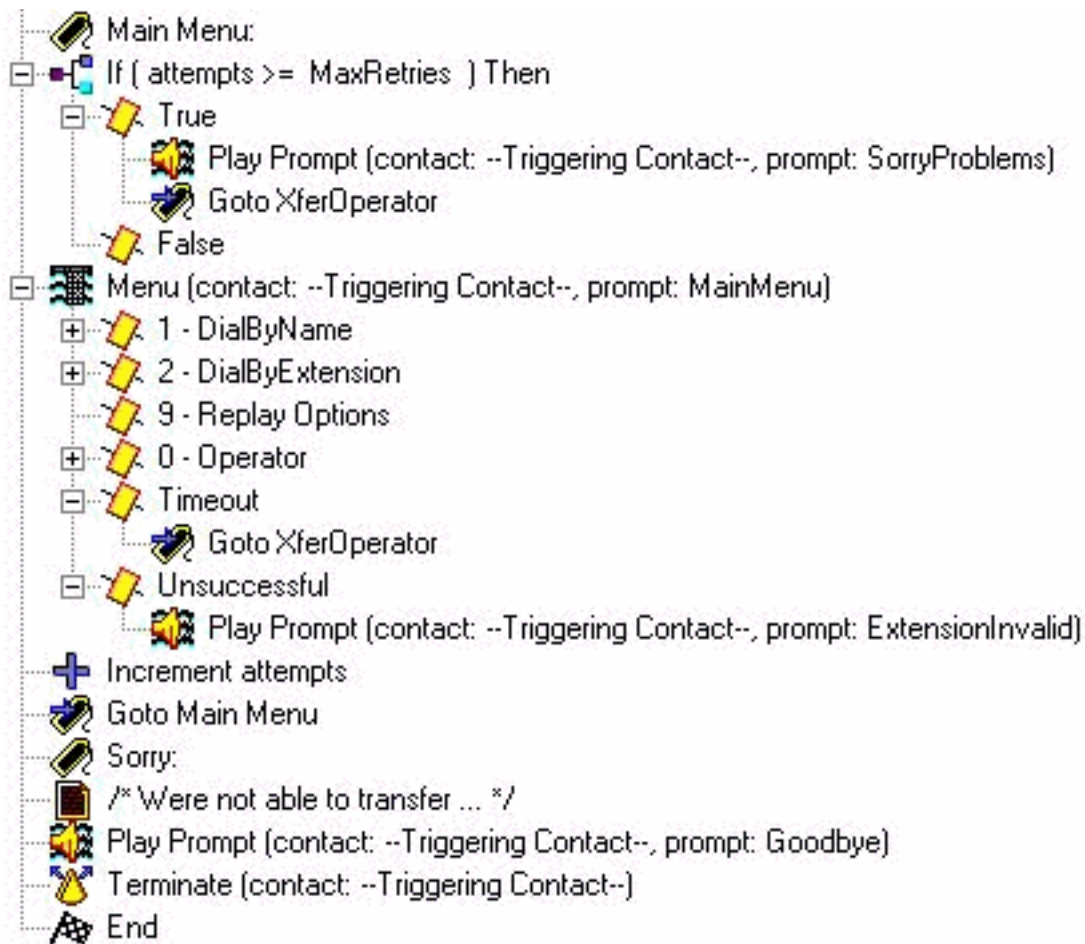
rapidamente. Reproduza algo antes e a saudação "real" para evitar qualquer recorte percebido à saudação. A seguir, uma etapa On Exception Goto. O que isso faz é procurar qualquer exceção do tipo "UndefinedPromptException" no momento da execução do script. Essa exceção específica ocorre quando a etapa tenta reproduzir um prompt que não existe. Isso só ocorre se a etapa que reproduz o prompt tiver a propriedade "Continuar em erros de prompt" definida como Não. Se definida como Sim, nenhum prompt será reproduzido e nenhuma exceção será exibida. O uso desta etapa de Ir para exceção se certificar de que, se um prompt crítico administrável pelo administrador (como o menu principal) for apagado, você poderá ramificar para uma seção na qual poderá ramificar para o rótulo "Desculpe" para informar ao usuário para tentar novamente mais tarde. Depois disso, chame um subfluxo, marque AltGreet.aef. Este é um script do sistema, que está sempre lá. Verifica a existência do arquivo AltGreeting.wav. Se existir, ele o reproduz. Caso contrário, o subfluxo retorna. Isso é útil, pois você pode usá-lo em conjunto com o GMS (agora chamado de sistema AVT) para reproduzir uma saudação de emergência de algum tipo antes da mensagem normal. Para obter mais informações sobre esse recurso, consulte [Configuração e Uso do Sistema de Gerenciamento de Saudações e Saudação Alternativa de Emergência para o Unity Express](#). Não há como saber se a saudação alternativa realmente é reproduzida (ou verifica a existência de um arquivo no repositório local). Você não precisa usar isso no início de um script. Com base em seu aplicativo, você pode usar a funcionalidade de saudação alternativa de emergência para gravar avisos de feriado, por exemplo. Em vez de chamar esse subfluxo, chame-o apenas se for feriado. Isso ajuda os administradores do site a gravar saudações de feriado através da interface do usuário de telefonia sem a necessidade de acessar o sistema por meio de um navegador ou da CLI (para renomear avisos ou atribuí-los a variáveis de script).



Decida qual prompt de abertura reproduzir. Se for feriado, reproduza a saudação de feriado. Se for durante o horário comercial (que pode ser configurado por meio de agendas no administrador da Web do Cisco Unity Express), reproduza a saudação de abertura (que pode ser algo tão simples quanto "obrigado por ligar para a Cisco Systems"), caso contrário, reproduza a saudação de fechado. Independentemente de qual seja a peça, você também vai diretamente para o menu principal.

Observação: as etapas "Is Holiday" (Feriado) e "Business Hours" (Horário comercial) são específicas para a versão 2.1.1 e posterior. Para versões anteriores do Cisco Unity Express, a única opção é usar a etapa "Hora do dia". Isso não permite que você reproduza saudações diferentes com base no dia ou na data. Baseia-se apenas no tempo.

[Menu principal](#)



Digite a seção Main Menu (Menu principal). Como grande parte do tratamento de erros envia o chamador de volta ao menu principal, um contador é adicionado. O contador "tentativas" é comparado a "MaxRetries", outra variável configurável pelo usuário (3 por padrão). Se esse contador for alcançado, peça desculpas e transfira o chamador para o operador ("Desculpe, você está tendo problemas. Aguarde um operador.").

A etapa Menu reproduz um prompt básico ("Se você souber o ramal do seu interlocutor, disque-o a qualquer momento. Para soletrar por nome, pressione 1, para um operador, pressione 0, para repetir essas opções, pressione 9"). Nesse caso, como você reproduz seu próprio prompt gravado, você altera o parâmetro Máximo de Tentativas nesta etapa para 0. Isso é feito em qualquer lugar que você tenha a opção de fazer isso. A razão é que se for algo diferente de 0, você ouvirá o aviso do sistema "você ainda está lá?" cada vez que o temporizador de expiração é alcançado e ele se ramifica para a ramificação Timeout. A ramificação Sem Êxito é alcançada quando alguém pressiona algo diferente de 1, 2, 9 ou 0. Nesse caso, você reproduz um prompt "O ramal inserido é inválido" e sai da etapa onde você incrementa o contador de tentativas e volta para o menu principal.

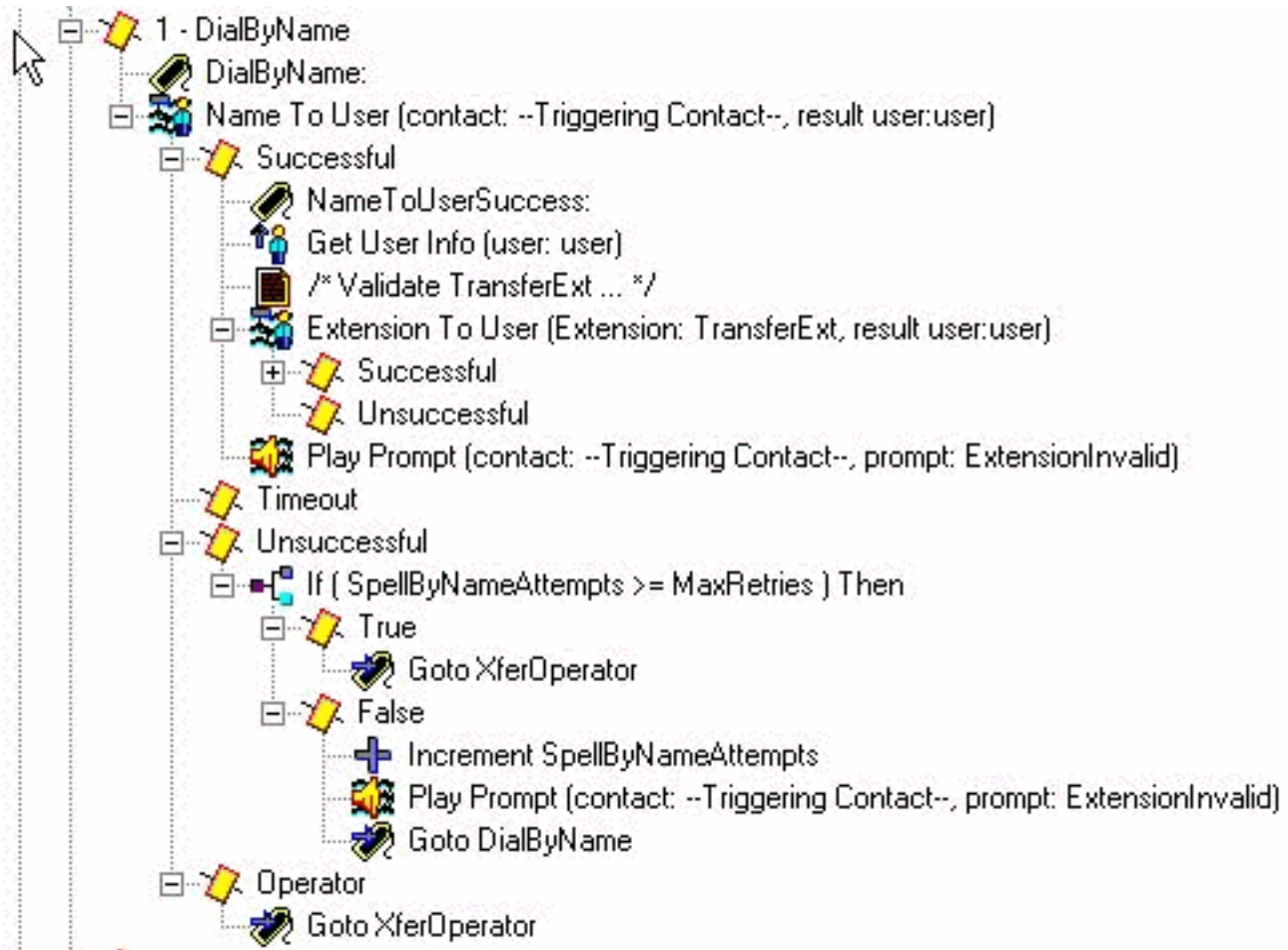
Observação: para garantir que o áudio pare assim que você digitar um dígito nas etapas de mídia (como Menu ou Prompt de Reprodução), você deve verificar o campo **Entrar** na guia Prompt da etapa. O campo Interruptível (na guia Geral) é usado para outras interrupções de etapa que não são atualmente aplicáveis ao Cisco Unity Express.

Uma única seção de tratamento de erros é adicionada. A seção "Desculpe" reproduz uma mensagem ("Não é possível transferir a chamada neste momento. Tente novamente mais tarde. Adeus."). Em seguida, a chamada é desconectada.

Você sempre pode usar uma saudação do sistema em vez dessas variáveis personalizadas. Os

prompts do sistema sempre têm a notação SP[] enquanto os prompts do usuário usam a notação P[]. Não é possível atribuir um prompt do sistema a uma variável de prompt quando você a define (no painel da variável, pois esses são valores de usuário por definição). Uma vez definido, no script, você pode atribuir um prompt do sistema a uma variável através da etapa Set (ou qualquer uma das etapas Create Prompt).

[Discar por nome](#)

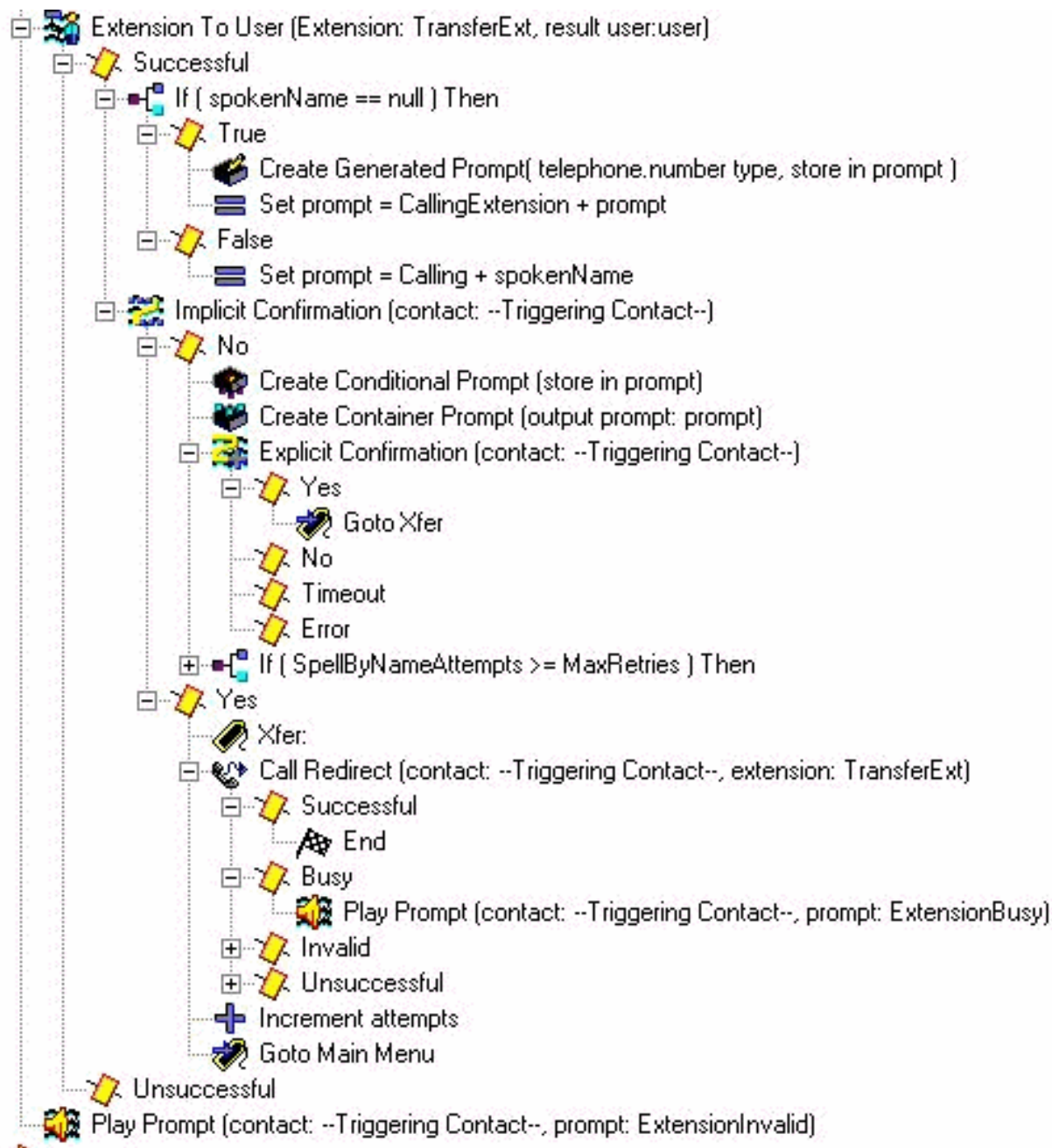


Uma maneira de fazer Discar por nome é baixar a amostra (S4_DialByName) do Cisco.com, carregá-la como um script separado no Cisco Unity Express e, em seguida, adicionar uma etapa de Subfluxo de chamada para chamá-la. Nesse caso, a etapa é colocada diretamente no script e discutida em detalhes. A etapa NameToUser permite configurar apenas um de seus prompts ("Soletre o sobrenome da pessoa que deseja chamar, seguido do nome da letra Q press 7 e, para a letra Z, pressione 9"). Não é possível eliminar todos os prompts do sistema. Para minimizá-los, primeiro é necessário desativar a tecla Cancel (padrão *). Se estiver presente, o sistema adicionará uma mensagem "para iniciar novamente, pressione estrela" no final do prompt gravado. Em segundo lugar, altere o número máximo de tentativas para zero, de modo que as mensagens "tente novamente" e "você ainda está lá" não possam ocorrer. Sempre acompanhe o número de tentativas com uma variável de contador separada adicionada. Ainda há um prompt do sistema que não pode ser eliminado. É o prompt para várias correspondências ("Foi encontrado mais de um nome. Selecione uma das seguintes opções..."). A etapa Nome para o usuário sempre termina como Êxito, Tempo limite, Sem êxito ou Operador (se selecionado nas propriedades da etapa). Exceto para a ramificação Bem-sucedido e Operador, você sai da etapa Menu. Isso significa que você incrementa o contador de tentativas e retorna ao menu principal. Com base na situação, você pode ter um contador de nova tentativa separado e adicionar uma etapa para voltar ao rótulo DialByName para que ele repita a ação já selecionada.

Observação: um defeito na versão 2.1.1 que deve ser considerado é o bug da Cisco ID [CSCeg81385](#) (somente clientes registrados) . Quando Maximum Retries (Máximo de repetições) na etapa NameToUser (Nome para usuário) estiver definido como zero ou um, você deverá usar um caractere de terminação. Sem o caractere de terminação, ele não vai para a ramificação "Êxito" desta etapa. Em vez disso, ele sempre vai para "Timeout", mesmo que haja uma correspondência.

Se o usuário não digitar um ramal que possa ser mapeado para um nome, a ramificação Não bem-sucedida verifica um novo contador (SpellByNameAttempts) em MaxRetries. Suponha que o mesmo número máximo de tentativas para inserir o menu principal precisa ser aplicado à função soletrar por nome. Você pode adicionar uma variável separada se ela precisar ser diferente.

Uma etapa NameToUser bem-sucedida significa que pelo menos um dígito é inserido e o sobrenome de um único usuário, conforme definido no Cisco Unity Express, é selecionado (a ramificação Timeout é alcançada somente se nenhum dígito for inserido). Para ter certeza de que o ramal é exclusivo, verifique se você pode mapear o ramal de volta para o usuário. Isso pode ser realizado pela etapa ExtensionToUser. Esta etapa é adicionada ao Cisco Unity Express 2.1.1. Anteriormente, não havia como, com um ramal, garantir que ele fosse mapeado para um usuário real no sistema de correio de voz. O melhor que você poderia fazer era configurar algumas variáveis para comparar, como com a adição de uma etapa "If (TransferExt < MinExtension) || (TransferExt > MaxExtension)..." ou algo parecido.

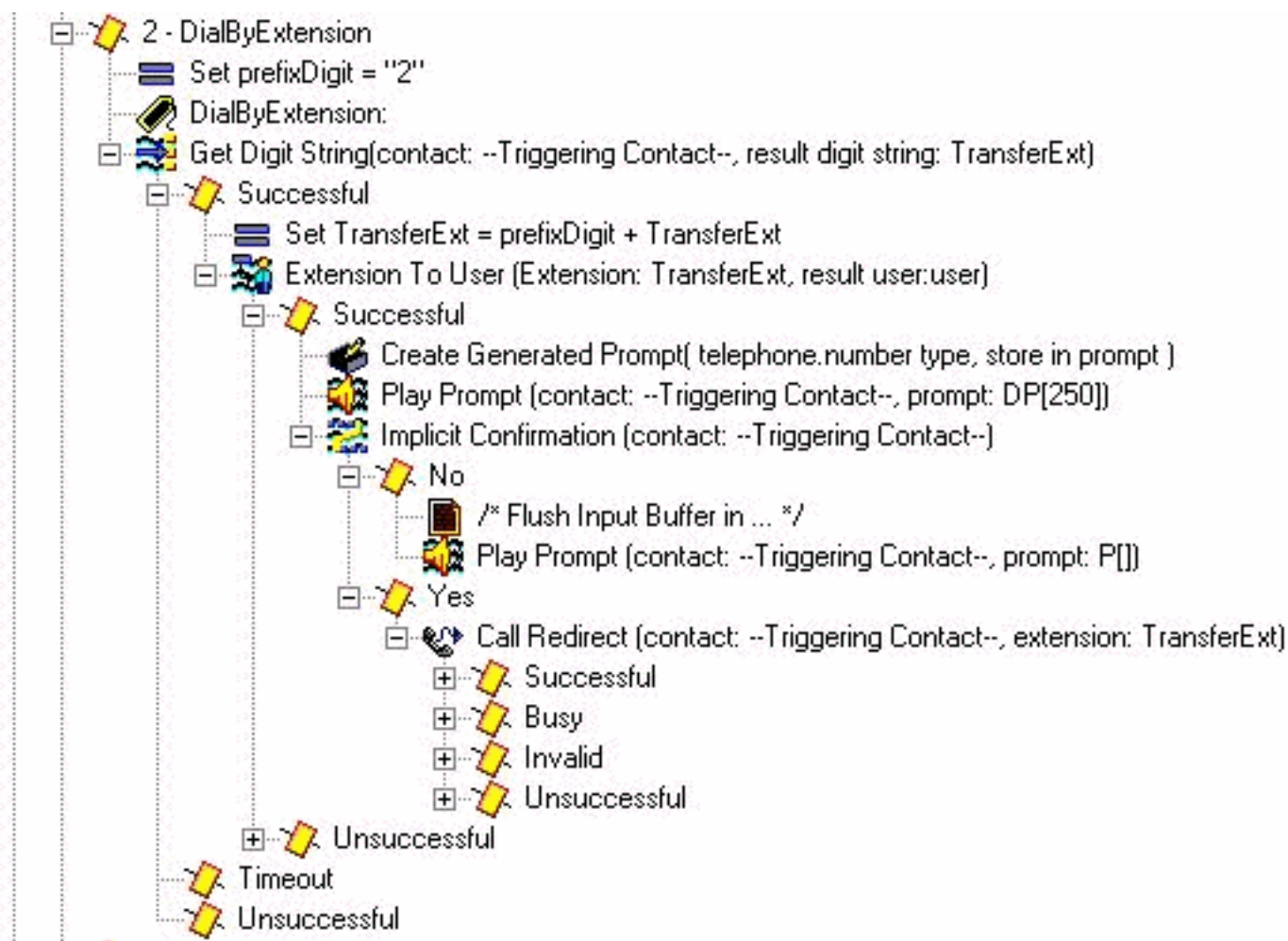


Aqui você vê o script depois de mapear a extensão de volta para o usuário. Se isso não acontecer, reproduza um prompt de que o ramal é inválido. Normalmente não há extensão por algum motivo. No caso do Cisco CallManager Express, isso pode significar que você desliga se tentar transferir a chamada. Isso pode causar uma pequena confusão, uma vez que o chamador já selecionou um usuário. No entanto, essa já é claramente uma condição de erro que você deseja saber mesmo assim. Se o mapeamento da extensão para um usuário for bem-sucedido, primeiro verifique se ele tem um nome gravado (a variável spokenName é preenchida quando você procura esse usuário). Em seguida, defina a variável de prompt como **Calling** mais o spokenName. Caso contrário, defina-o como **Extensão de chamada** mais o ramal soletrado do usuário. Você também pode especificar o nome (reproduza S[name] no prompt). Não há nenhum nome gravado para reproduzir. Qualquer ortografia do sistema, como soletrar um nome ou um ramal, é sempre feita na voz do sistema e não pode ser personalizada. A etapa Confirmação implícita reproduz o prompt que você acabou de criar. Se o usuário não digitar nada, ele vai em frente e entra na ramificação 'Sim' e redireciona a chamada. Se houver um problema, no caso da integração do Cisco CallManager, ele reproduzirá uma saudação e o chamador será enviado de volta ao menu principal. Se o chamador inserir algo, gere outro prompt. Desta vez, comece o prompt com o spokenName ou com uma versão escrita do nome do usuário (etapa Criar prompt condicional). Siga esta ação com uma pausa e um prompt "se este for o nome da pessoa que

you are calling, press 1". Now the user can press 1 for yes, 2, * or wait. The system sends you back to the letter-by-name section or to the operator. This depends on how many times they pass through the system.

Discar por ramal

The extension-by-extension section already has the first digit ("2" in this example) collected. Define the variable prefixDigit as 2. You can easily change this, based on the initial digit (or if there are more than one possible initial digits). No digit to the left can be the same as another option in the menu (like 1, which you use to dial by name in this example). The string Get Digit String also collects a fixed number of digits (two in this case). Therefore, the total number of digits must be corrected. If the valid extensions start with 2 or 3, you will be able to add a step "Set prefixDigit = "3" for the option 3 of the menu followed by a step Goto DialByExtension.

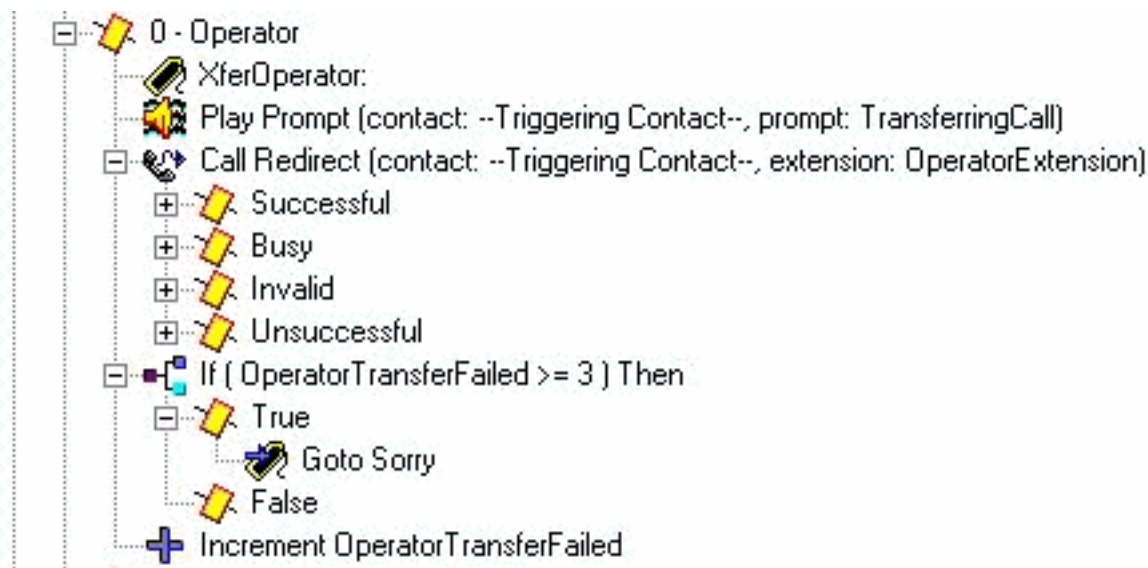


As you have extensions of three digits in this environment, the parameter Get Digit String step Input length is 2. There is no termination character and no cancellation key. The maximum number of new attempts is 0. If a user does not enter a valid extension, the script increments the number of attempts and returns to the main menu. The step does not reproduce any audio (prompt is P[]), because the user is already in the middle of the digits of dialing.

When two digits are collected successfully, precede the prefixDigit ("2" in this case) that is already dialed (which allows you to branch from the Menu step). The TransferExt that results passes then to the Extension To User step, which verifies if this is a valid user. If it is, create a prompt with the extension-by-name. In the Implicit Confirmation step, reproduce the prompt "extension-by-name" followed by the extension. Find the user and reproduce the spoken name, as done in

DialByName. No entanto, neste caso não é necessário. Se nenhum dígito for discado, suponha que esse seja o ramal que o chamador deseja alcançar e redirecione a chamada. Caso contrário, volte para o menu principal. A finalidade da etapa Play Prompt DP[250] é limpar qualquer outra entrada, porque a etapa Get Digit String não tem nenhum caractere de terminação configurado. A ideia é que você queira que isso funcione se alguém discar o ramal "200" ou "200#". Se o # for digitado enquanto a etapa Confirmação implícita for executada, ele vai para a parte Não dessa etapa. Em alguns casos, provavelmente não é necessário ter uma etapa de confirmação implícita, mas sim redirecionar a chamada imediatamente. Para fins de ilustração, essas etapas são deixadas em.

Transferência do operador

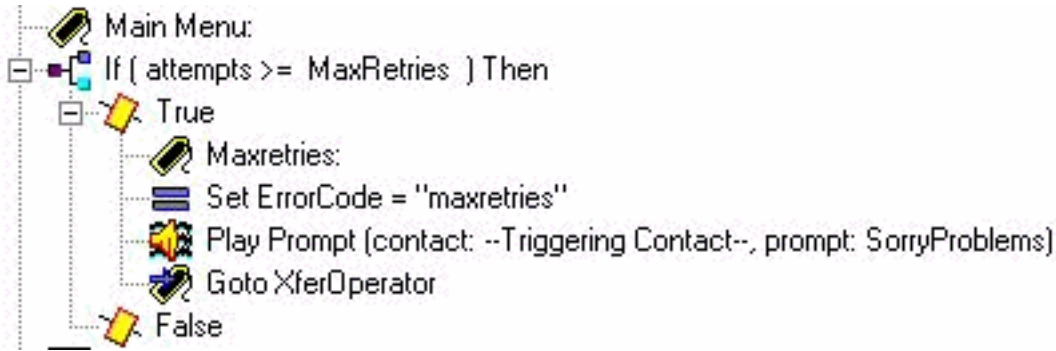


Toque um prompt "Transferindo chamada" e, em seguida, redirecione-a para o ramal do operador configurado. Se ele falhar por algum motivo, você terá um contador separado, o qual será incrementado e voltará para o menu principal. O motivo é que você deseja ter algum método para dar ao operador algum tempo para desligar o telefone ou registrar-se novamente para que você possa tentar transferir para ele novamente com êxito. Depois de várias tentativas (codificado no código 3 neste exemplo), você ramifica para o rótulo "Desculpe", que descarta a chamada.

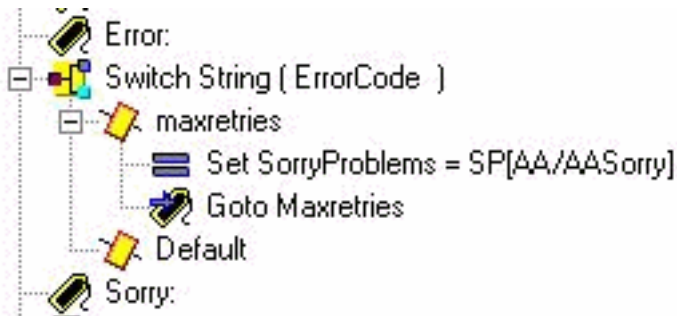
Tratamento de Erros Avançado

Há muitos lugares onde scripts podem falhar. Uma das áreas mais comuns em um script bem projetado é quando os prompts do usuário estão ausentes. Quando o sistema tenta reproduzir um prompt que não existe, várias coisas podem ocorrer. Primeiro, se a etapa que reproduz o prompt tiver "Continue On Prompt Errors" definido como Sim. Ele vai para a próxima etapa e não reproduz o prompt. Se a opção Continuar em erros de prompt estiver definida como Não, uma exceção será gerada. Na maioria das vezes, esse é o "UndefinedPromptException", o que significa que o prompt não foi encontrado ou nunca foi configurado. Quando você tem uma etapa On Exception Goto (Indo para Exceção), ela pode permitir que você capture essa exceção e divida para uma seção de código que pode reproduzir uma mensagem do sistema ou outra forma de áudio. Como a etapa On Exception Goto se aplica a todo o áudio reproduzido em todo o script (você não pode definir uma etapa On Exception Goto separada para cada prompt reproduzido), a melhor maneira de lidar com isso é definir uma variável de código de erro imediatamente antes de reproduzir o prompt. Dessa forma, você pode se dividir em algum lugar, tomar uma ação com base nesse código de erro e, possivelmente, se dividir para trás.

Por exemplo, você pode modificar o script com este:



Você adicionou o rótulo "Maxretries". Isto é usado para voltar ao mesmo lugar. Defina uma variável ErrorCode como **Maxretries**. Na etapa do prompt de reprodução, verifique se a opção Continuar em erros de prompt está definida como **Não**. Agora adicione uma seção que possa lidar com isso. Verifique se ErrorCode corresponde a "Maxretries" e defina a variável DesculpeProblemas (que você tentou reproduzir anteriormente) para a mensagem do sistema **Desculpe por estar com problemas. Por favor, fique na linha e alguns estarão com você em breve**. Em seguida, vá para a etiqueta Maxretries para que ela possa ser reproduzida.



A última etapa é alterar a mensagem On Exception Goto (Invisível na exceção) para procurar o erro UndefinedPromptException:



Esta etapa pode estar em qualquer lugar do script. No entanto, é normalmente colocado no início ou perto dele. Há várias etapas que podem reproduzir prompts. Os prompts do sistema estão listados na parte inferior deste documento.

Esse método pode ser usado em vez de chamar o subfluxo checkAltGreet.aef. Você precisa reproduzir o arquivo AltGreeting.wav. Se houver um erro, trate-o. O benefício é que você sabe quando ele joga ou não.

Embora menos comuns, estas são as outras exceções de prompt:

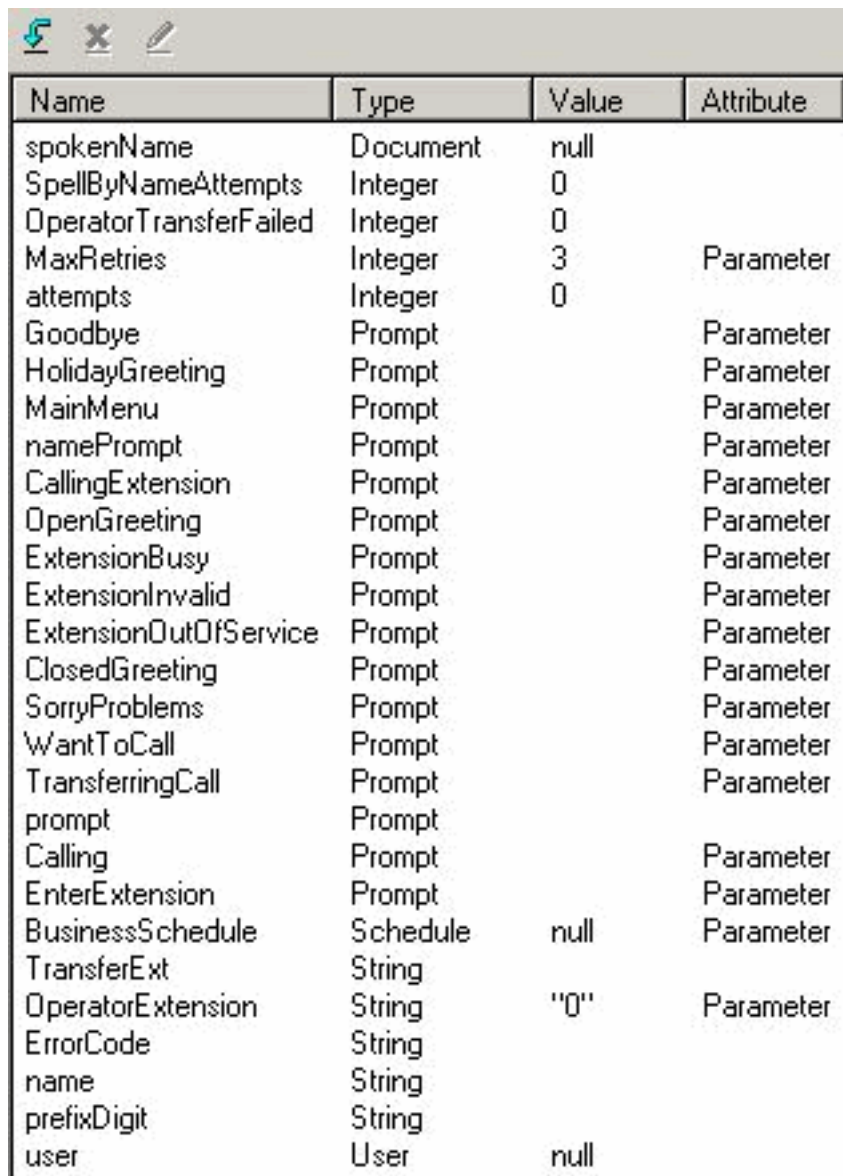
- UndefinedPromptException
- PromptException
- UndefinedPromptGenerator
- InvalidPromptArgumentException
- ExpressãoPromptNão Suportada

[Transferir uma chamada diretamente para o correio de voz](#)

Não é possível redirecionar a chamada diretamente para uma caixa de correio de voz. Ele deve ser enviado a um número que encaminhe para o correio de voz. Esse pode ser um número fictício no Cisco CallManager Express (ou mesmo no Cisco CallManager) configurado para desviar todas as chamadas definido para o número piloto do correio de voz. Use as configurações de número/e.164 do Cisco Unity Express para corresponder esse número a um assinante ou grupo. [Transferir um chamador diretamente para uma caixa de correio do Unity Express](#) explica como isso pode ser feito.

Variáveis de script

Você deve criar todas as variáveis usadas em um script para poder usá-las. Se você usar um e depois excluí-lo, selecione **Ferramentas > Validar** no Editor de scripts para localizá-lo. Para este exemplo de script, estas são as variáveis usadas. Observe o Nome, o Tipo, o Valor (que é o valor da variável inicialmente) e o Atributo. O atributo Parameter (Parâmetro) significa que a variável é exposta ao administrador através da página da Web. Qualquer coisa que você queira que o administrador do Cisco Unity Express personalize precisa ter o atributo Parameter.



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	

Se você não quiser gravar todos esses prompts personalizados, poderá editar cada um e desmarcar o atributo **Parameter** para que não seja exposto na página da Web. Para garantir que o sistema ainda funcione, adicione várias etapas do Set ao início do script. Por exemplo, defina 'Calling' como 'SP[AA/AACalling]'. Nem todos os prompts mostrados aqui têm scripts de sistema

correspondentes. Na maioria dos casos, faz sentido gravá-los você mesmo. Se os prompts não mudarem, você pode gravá-los uma vez, verificar se existem no Cisco Unity Express System com um nome de arquivo especificado e, em seguida, remover o atributo Parameter da variável. Dessa forma, o prompt reproduz um nome de arquivo fixo. Mas, a variável não é exposta através da interface da Web na qual um administrador pode acidentalmente alterá-la.

Fazer upload de um script para o Cisco Unity Express

Geralmente, a primeira etapa para adicionar um novo aplicativo e script ao Cisco Unity Express é carregar os prompts. Isso é importante porque os prompts devem existir no sistema para que você os configure nos parâmetros do script. Caso contrário, você deve deixar os parâmetros em branco, carregar os prompts e, em seguida, voltar e editar os parâmetros do script. Para obter informações adicionais sobre como gravar e carregar avisos, consulte [Administração via Sistema de Gerenciamento de Telefone \(AVT\)/Saudações](#).

Depois de salvar seu script e estar pronto para experimentá-lo, primeiro é necessário carregá-lo no Cisco Unity Express. A maneira mais fácil é usar a GUI, já que isso não exige um servidor FTP para carregar os arquivos de script.

Conclua estes passos:

1. Faça login na GUI do Cisco Unity Express com uma conta com direitos administrativos.
2. Selecione **Correio de voz > Atendimento automático**.
3. Clique em **Add**.
4. Ao lado de "Selected Automated Attendant Script" (Script de atendimento automático selecionado), clique no botão **Upload**.
5. Clique em **Procurar**.
6. Localize o arquivo de script, selecione-o e pressione **OK**.
7. Para alterar o nome do arquivo de destino, faça isso. Caso contrário, pressione **Carregar**.
8. Digite um nome de aplicativo. Isso pode ser qualquer coisa em letras minúsculas e é usado para se referir ao aplicativo associado a esse arquivo de script.
9. Clique em **Next**.
10. A página agora mostra todas as variáveis para o script que estão marcadas como parâmetros. Isso significa que eles podem ser configurados nesta página. Se você ainda não carregou os arquivos de prompt, você pode passar por isso primeiro e depois preencher as informações posteriormente.
11. Clique em **Avançar** quando terminar de inserir parâmetros de script.
12. Verifique se o aplicativo está marcado como **Ativado**, configure o número máximo de chamadas simultâneas (o padrão é o número máximo permitido pela licença) e insira um número de chamada de entrada. Esse é o número chamado para o qual esse aplicativo específico precisa ser chamado (disparador). Esse pode ser o seu número de atendimento automático principal ou temporário. Você também pode pular isso e preenchê-lo mais tarde. Sem um número de chamada de entrada, não é possível testar ou iniciar um script. **Observação:** para integrações do Cisco CallManager com suporte ao SRST, é necessário inserir vários números de chamada de entrada. Antes do Cisco Unity Express 2.1.1, essas informações precisavam ser inseridas através da CLI. No Cisco Unity Express 2.1, vários números de chamada de entrada (disparadores) podem ser inseridos na tela **Administration > Call-in Numbers**.
13. Clique em **Finished (Concluído)**.

Depois de fazer upload do script e configurar o aplicativo, você precisa ter certeza de que pode rotear chamadas para o aplicativo. Isso significa que o Cisco CallManager Express ou o Cisco CallManager devem ser configurados. Com o CallManager Express, isso envolve a configuração de um peer de discagem que aponta para o Cisco Unity Express que corresponde ao padrão (número de disparo ou número de chamada de entrada) configurado no Cisco Unity Express. No Cisco CallManager, um ponto de rota CTI com esse número deve ser adicionado. Não se esqueça de que você deve associar este ponto de rota CTI ao usuário JTAPI no qual o Cisco Unity Express faz logon. Essa configuração pode possivelmente exigir uma reinicialização do Cisco Unity Express. Depois de adicionar o ponto de rota e associar-se ao usuário, você pode alterar o script o quanto quiser sem a necessidade de reinicializar.

Ao atualizar um script, se nenhum parâmetro de variável for adicionado ou alterado, você poderá ir para o editor do Cisco Unity Express, clicar em **Voice Mail > Scripts**, Upload e selecionar o script. Ele é substituído por todos os outros parâmetros configurados que permanecem os mesmos.

Quando você cria avisos, a coisa mais fácil é usar o sistema AVT. Os nomes dos arquivos que ele grava têm carimbos de data/hora incluídos. Mas no Cisco Unity Express 2.1.1 e posterior, é possível usar o comando CLI **ccn rename prompt**. Anteriormente, você tinha que fazer o download do prompt, carregá-lo novamente com um novo nome e excluir o arquivo .wav original. Por exemplo:

```
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>
```

[Troubleshoot](#)

Ao solucionar problemas de um script, você precisa entrar em contato e testar vários cenários, tanto a entrada regular do usuário quanto o tempo limite e as condições de erro antecipadas.

Antes de carregar um script, sempre acesse a entrada **Ferramentas > Validar** no Editor de scripts e certifique-se de que a mensagem "Validação foi bem-sucedida" seja exibida. Caso contrário, todos os erros serão exibidos no painel de depuração. Você pode clicar duas vezes neles e ele mostra onde a condição de erro existe no script.

Na maioria dos casos, os rastreamentos padrão são mais do que suficientes para solucionar problemas. Para simplificar isso, deixe os rastreamentos padrão ativados e utilize o recurso de filtragem para exibir o que você precisa.

Antes de qualquer teste, emita o comando **clear trace**. Isso limpa o buffer de memória das mensagens de rastreamento para que a quantidade de informações filtradas inclua apenas os dados após a emissão do comando.

Para execução de script, o item mais importante a ser filtrado é EXECUTING_STEP. Mostra todas as etapas à medida que são executadas. Por exemplo:

```
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
...
```

Neste exemplo de saída, algumas das informações redundantes são removidas no meio e cortadas no início do carimbo de data/hora. Na maioria das vezes, as informações mais importantes estão no final de cada linha.

Nota: Muitas destas linhas de produção são reduzidas para uma segunda e terceira linhas devido a razões espaciais.

```
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)

```

Agora você pode ver cada etapa enquanto ela é executada. Você não pode ver como as expressões são avaliadas, nem como qualquer entrada do usuário. Se o script na sua frente, você poderá seguir onde o script ramifica, com o conhecimento dos dígitos inseridos (nesse caso, o usuário disca, disca um ramal e é transferido).

Use um filtro para ver os dígitos.

```

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

```

Nessa saída, você vê que a extensão 201 está discada.

Se as versões do CUE e do editor de scripts CUE forem diferentes, e você estiver tentando carregar um script AA via GUI ou CLI, o carregamento falhará com a mensagem de erro `Falha no upload`. Se ambas as versões forem diferentes, você precisará recriar o script usando o editor CUE da versão que é a mesma do CUE. Ou atualize o CUE para a versão do editor de scripts CUE e execute o script.

[Appendix](#)

[Apêndice 1 - Prompts do sistema](#)

Esta tabela mostra os prompts do sistema que você pode usar.

Observação: não é possível reutilizar todos os prompts ouvidos no aplicativo de correio de voz do Cisco Unity Express em um script personalizado. Alguns dos prompts estão disponíveis somente através do script de correio de voz que você não pode personalizar através do Editor de scripts do Cisco Unity Express.

Prompt do sistema	Áudio gravado
-------------------	---------------

SP[AA/AAWelcome]	"Bem-vindo ao atendimento automatizado"
SP[AA/AAMainMenu]	"Para digitar o número de telefone da pessoa que você está tentando acessar, pressione 1. Para digitar o nome da pessoa que você está tentando acessar, pressione 2. Para transferir para o operador, pressione 0"
SP[AA/AAEnterExtn]	"Digite o número de telefone e pressione a tecla #"
SP[AA/AACallingExtn]	"Call extension"
SP[AA/AAPhoneReach]	"O número de telefone que você está tentando acessar"
SP[AA/AAOutOfServicePhone]	"No momento, está fora de serviço"
SP[AA/AANameDial]	"Soletre o sobrenome da pessoa que deseja chamar, seguido do nome. Para a letra Q, prima 7, para Z, prima 9"
SP[AA/AACalling]	"Calling"
SP[AA/AASorry]	"Desculpe, você está tendo problemas. Por favor, fique na linha e alguns estarão com você em breve."
SP[AA/AWant2Call]	"Se este for o nome da pessoa que você está chamando, pressione 1, para recomençar, pressione *"
SP[AA/ainda_lá]	"Você ainda está aí?"

[Apêndice 2 - Redefinir rastreamentos padrão](#)

Para redefinir o sistema (Cisco Unity Express 2.1.x e anterior) para rastreamentos padrão através da CLI, desative primeiro todos os rastreamentos usando o comando **no trace all** e cole-o na CLI:

```

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
```

[Informações Relacionadas](#)

- [Suporte à Tecnologia de Voz](#)
- [Suporte aos produtos de Voz e Comunicações Unificadas](#)
- [Troubleshooting da Telefonia IP Cisco](#)
- [Suporte Técnico - Cisco Systems](#)