



A MITEL  
PRODUCT  
GUIDE

# Unify OpenScape Contact Center

Unify OpenScape Contact Center V11 R1, Guia de Integração da API URA

Guia de Integração  
09/2024

## Notices

The information contained in this document is believed to be accurate in all respects but is not warranted by Mitel Europe Limited. The information is subject to change without notice and should not be construed in any way as a commitment by Mitel or any of its affiliates or subsidiaries. Mitel and its affiliates and subsidiaries assume no responsibility for any errors or omissions in this document. Revisions of this document or new editions of it may be issued to incorporate such changes. No part of this document can be reproduced or transmitted in any form or by any means - electronic or mechanical - for any purpose without written permission from Mitel Networks Corporation.

## Trademarks

The trademarks, service marks, logos, and graphics (collectively "Trademarks") appearing on Mitel's Internet sites or in its publications are registered and unregistered trademarks of Mitel Networks Corporation (MNC) or its subsidiaries (collectively "Mitel"), Unify Software and Solutions GmbH & Co. KG or its affiliates (collectively "Unify") or others. Use of the Trademarks is prohibited without the express consent from Mitel and/or Unify. Please contact our legal department at [iplegal@mitel.com](mailto:iplegal@mitel.com) for additional information. For a list of the worldwide Mitel and Unify registered trademarks, please refer to the website: <http://www.mitel.com/trademarks>.

© Copyright 2024, Mitel Networks Corporation

All rights reserved

# Índice

<b>1 Sobre este guia</b>	5
1.1 A quem se destina este guia	5
1.2 Convenções de formatação	5
1.3 Feedback da documentação	6
<b>2 Sobre a API URA do OpenScape Contact Center</b>	7
2.1 Visão Geral	7
2.2 Instalando a API URA do OpenScape Contact Center	9
2.3 Funções da API URA para sistemas da URA não monitorados	10
<b>3 A31003-S22A-N100-01-V420Configurando a API URA do OpenScape Contact Center</b>	15
3.1 Configuração de retenção na URA	15
3.2 Configuração de retenção na fila	17
3.3 Configuração de colocação demarcada	19
3.4 Escrevendo um script URA	19
3.4.1 Gravando um script da URA para a configuração de Retenção na URA	20
3.4.2 Gravando um script da URA para a configuração de Retenção na fila	22
3.4.3 Gravando um script da URA para a configuração de FMNQ	24
3.5 Usando um sistema da URA em um ambiente de multilocação	26
<b>4 Usando as funções da API URA do OpenScape Contact Center</b>	27
4.1 Passando parâmetros	27
4.2 CallerDisconnected	28
4.3 CreateCallback	29
4.4 DeleteCallback	32
4.5 Dequeue	33
4.6 Disconnect	34
4.7 Enqueue	35
4.8 EnqueueForAgent	37
4.9 GetBusinessUnit	41
4.10 GetCallTransferable	42
4.11 GetContactData	43
4.12 GetTransitNumber	45
4.13 Initialize	46
4.14 QueryAgentStatus	47
4.15 QueryCallInfo	49
4.16 QueryCallStatus	51
4.17 QueryQueueStatistics	53
4.18 QueryRoutingInfo	55
4.19 QuerySystemStatus	58
4.20 ReleaseTransitNumber	60
4.21 SetBusinessUnit	61
4.22 SetCallInfo	62
4.23 SetCallTransferable	64
4.24 SetContactData	65
4.25 SetDisplay	67
4.26 Transfer	68

## **Índice**

<b>5 Códigos de retorno.....</b>	<b>71</b>
<b>Índice remissivo.....</b>	<b>75</b>

# 1 Sobre este guia

Este guia descreve como integrar um sistema de Resposta de Voz Interativa (URA) com o OpenScape Contact Center usando a Interface de Programação do Aplicativo (API) da URA do OpenScape Contact Center.

## 1.1 A quem se destina este guia

Este guia destina-se a integradores de sistemas que desejem integrar um sistema de Resposta de Voz Interativa (URA) com o OpenScape Contact Center.

## 1.2 Convenções de formatação

Estas são as convenções de formatação usadas neste guia:

**Negrito**

Esta fonte identifica os componentes do OpenScape Contact Center, nomes de janelas e caixas de diálogo, e nomes de itens.

*Itálico*

Esta fonte identifica referências a termos ou expressões que são referências de outros documentos relacionados.

Fonte Monospace

Esta fonte identifica o texto que deve ser digitado ou que o computador apresenta em uma mensagem.

---

**NOTA:** As Notas enfatizam informações que são úteis, mas não essenciais, tais como dicas ou métodos alternativos para se executar uma tarefa.

---

---

**IMPORTANTE:** Dê atenção especial às ações que possam afetar de forma negativa a operação do aplicativo ou resultar em perda de dados.

---

## 1.3 Feedback da documentação

Para relatar algum problema com este documento, chame o Centro de Assistência ao Cliente.

Quando você ligar, tenha em mãos as informações seguintes. Isto ajudará a identificar o documento com o qual você está tendo problemas.

- **Título:** Guia de Integração da API URA
- **Número do pedido:** A31003-S22A-N100-01-V420

## 2 Sobre a API URA do OpenScape Contact Center

Este capítulo fornece uma visão geral da Interface de Programação do Aplicativo (API) da URA do OpenScape Contact Center. A API URA do OpenScape Contact Center é compatível quando estiver conectada com as seguintes plataformas de comunicação:

- OpenScape Voice
- OpenScape 4000 ou HiPath 4000
- OpenScape Business

---

**NOTA:** Quando conectado a uma plataforma de comunicação OpenScape Voice, a utilização da URA monitorada em front-end com ramais de pré-processamento URA não é suportada.

---

### 2.1 Visão Geral

Um sistema da URA interage com o chamador para obter informações adicionais sobre seus requisitos e fornece informações para o chamador enquanto ele está esperando na fila pelo usuário apropriado para tratar de sua chamada.

É possível criar um script da URA personalizado, que coleta informações de seus clientes. É possível, por exemplo, solicitar ao chamador que pressione 1 para Assistência ou 2 para Vendas.

O script da URA pode reunir os requisitos de um cliente a partir das seguintes fontes:

- **Informações em rede (informações públicas da plataforma de comunicação)**
  - Identificação Automática do Número (ANI)—informações sobre o agente chamador
  - Serviço de Identificação do Número Discado (DNIS)—informações sobre o agente chamado
- **Solicitação de chamadas**
  - Sinalização de Multifreqüência de Tom Duplo (DTMF) — respostas do chamador usando um botão de apertar (por exemplo, número da conta ou de identificação pessoal)
  - Reconhecimento de voz — respostas de voz do chamador para obter informações mais específicas (por exemplo, o prompt pode dizer ao cliente para optar por Vendas ou Assistência)

## Sobre a API URA do OpenScape Contact Center

### Visão Geral

- **Recuperação da base de dados de clientes**
  - Perfil do cliente (por exemplo, idioma, nível de habilidade e localização)
  - Histórico do cliente (por exemplo, histórico de crédito, compras anteriores e problemas abertos)
  - Preferências (por exemplo, um usuário específico)

O objetivo do script da URA é identificar uma fila que descreva os requisitos de mudança de uma chamada enquanto ela espera na fila.

É possível criar um script da URA para coletar informações de ANI e DNIS do OpenScape Contact Center, pedir informações específicas ao chamador e, depois, usar essas informações para acessar os dados de uma base de dados hospedada ou local.

Por exemplo, é possível pedir ao chamador para digitar seu número de conta de quatro dígitos, depois usar esse número para acessar as informações da base de dados e identificar filas para as chamadas com base nas informações do DNIS ou dos campos Para e De.

O texto após o diagrama descreve as etapas numeradas.

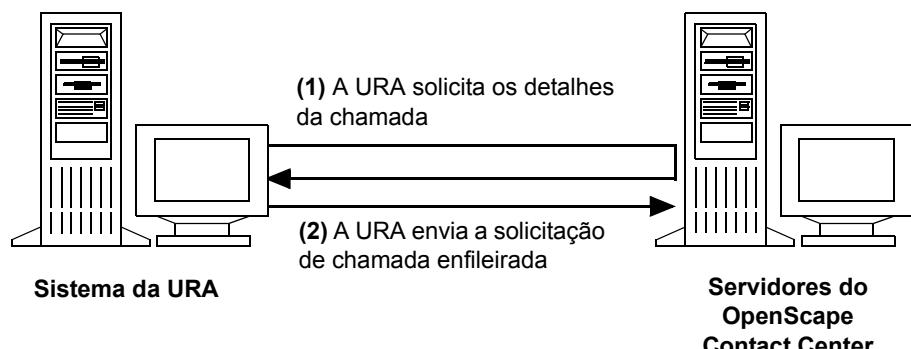


Figura 1 Sistema da URA e interação com o OpenScape Contact Center

1. Quando uma chamada chegar em seu site, ela será atendida pelo sistema da URA. Então, o script da URA coleta as informações de ANI e DNIS ou dos campos Para e De da transação, determina a fila, e envia uma solicitação de chamada enfileirada para o OpenScape Contact Center.

---

**NOTA:** O sistema da URA também pode enviar uma solicitação de roteamento do OpenScape Contact Center para o OpenScape Contact Center, fornecendo as informações coletadas pelo sistema da URA. Então, o Routing Server designa uma fila e outras informações de roteamento com base na opção de roteamento configurada para o sistema, que pode ser usada para enviar uma solicitação de chamada enfileirada.

---

2. Então, o Routing Server procurará o melhor usuário disponível para tratar da chamada.
3. O script da URA pode pedir ao cliente uma série de menus para determinar os requisitos específicos de cada cliente. Quanto mais complexo for seu sistema de menus, mais detalhadas serão as informações que se poderá reunir de seus clientes.

Cada opção exclusiva do menu deve corresponder a uma fila exclusiva, de forma que seja possível usar toda a capacidade de roteamento baseado em habilidades do OpenScape Contact Center.

---

**NOTA:** Para obter mais informações sobre a criação de sistemas de menus da Unidade de Resposta Automática (URA), entre em contato com seu representante da assistência técnica.

---

---

**NOTA:** Para obter mais informações sobre as funções disponíveis da URA, consulte [Capítulo 4, “Usando as funções da API URA do OpenScape Contact Center”](#).

---

## 2.2 Instalando a API URA do OpenScape Contact Center

Para instalar a Interface de Programação do Aplicativo (API) da URA do OpenScape Contact Center, é necessário copiar o arquivo **hppcivr.dll** da pasta padrão da instalação na máquina do servidor principal do OpenScape Contact Center para a máquina da URA e fazer referência ao arquivo na configuração da URA. Além disso, você deve copiar os arquivos a seguir da pasta de instalação padrão na pasta do aplicativo:

- **oscc\_libeay32.dll**
- **oscc\_ssleay32.dll**

Não é necessário fazer nenhuma referência a esses arquivos, pois eles são utilizados internamente pela API.

## Sobre a API URA do OpenScape Contact Center

Funções da API URA para sistemas da URA não monitorados

### 2.3 Funções da API URA para sistemas da URA não monitorados

O OpenScape Contact Center usa funções da API URA para facilitar a integração com recursos da URA não monitorados.

---

**NOTA:** Sistemas da URA não monitorados não são compatíveis quando conectados a um tipo não configurado de plataforma de comunicação.

---

Como esse tipo de sistema da URA não é monitorado pelo OpenScape Contact Center, os seguintes conceitos devem ser levados em consideração:

- O OpenScape Contact Center é notificado com os números ANI e DNIS quando a chamada chegar no sistema da URA.
- É necessária uma coordenação entre o OpenScape Contact Center e o sistema da URA para transferir uma chamada do sistema da URA para o OpenScape Contact Center.
- As chamadas que deixam o sistema da URA e que são desconectadas ou transferidas para fora usam uma API para notificar o OpenScape Contact Center.

O sistema da URA deve fornecer o ANI e o DNIS da chamada ao OpenScape Contact Center. Essas informações podem ser obtidas pela URA através do protocolo de trunking específico da plataforma de comunicação, como por exemplo, ISDN (Rede Digital de Serviços Integrados). A API SetCallInfo configura o ANI e o DNIS localmente antes que essas informações sejam enviadas para o OpenScape Contact Center usando a API QueryCallInfo existente.

A API QueryCallInfo é usada pelo sistema da URA para notificar o OpenScape Contact Center quando uma chamada chegar no sistema da URA. Esse API deve ser chamado logo que possível, depois que o sistema da URA iniciar o processamento da chamada que está chegando. A API QueryCallInfo não possui um parâmetro opcional, TrunkConnectedIVRCall, que deve ser configurado como “1;” para sistemas da URA não monitorados.

As chamadas em um sistema da URA não monitorado que estejam enfileiradas para o OpenScape Contact Center para Retenção na URA ou Retenção na fila devem ser transferidas para o Grupo de controle de roteamento do OpenScape Contact Center usando-se um número de trânsito. Um número de trânsito é um número piloto para o Grupo de controle de roteamento do OpenScape Contact Center. O número de trânsito é necessário para rastrear a chamada da URA não monitorada depois que ela é transferida para o OpenScape Contact Center.

O OpenScape Contact Center mantém um conjunto de números de trânsito, que só são usados para transferir chamadas da URA não monitorada. O sistema da URA consulta o OpenScape Contact Center quanto a um número de trânsito exclusivo, usando a API GetTransitNumber. Após receber o número de trânsito,

o sistema URA deve transferir a chamada imediatamente. Quando a chamada chega ao RCG do OpenScape Contact Center, este a identifica usando o número de trânsito e pode, então, associar as informações das chamadas recebidas anteriormente do sistema da URA com a chamada entrante. O número de trânsito fica, assim, disponível para ser usado com outra chamada da URA não monitorada.

---

**NOTA:** Se ocorrer erro ou time-out, o sistema URA poderá decidir transferir a chamada para um dispositivo não pertencente ao OpenScape Contact Center (como correio de Voz ou XPressions), em cujo caso não será necessário chamar GetTransitNumber.

---

É possível usar o subdiálogo GetTransitNumber de outra forma, dependendo do que a sua plataforma de comunicação estiver usando: Retenção na fila ou Retenção da URA:

- **Retenção na fila** – O GetTransitNumber deve ser chamado imediatamente antes da chamada ser transferida para o RCG do OpenScape Contact Center, o que normalmente acontece logo após a chamada ser enfileirada. A seguir, o OpenScape Contact Center procurará um usuário e direcionará a chamada para o usuário designado.

## Sobre a API URA do OpenScape Contact Center

### Funções da API URA para sistemas da URA não monitorados

- **Retenção na URA** – O GetTransitNumber deve ser chamado imediatamente antes da chamada ser transferida. A chamada ainda é transferida para o RCG do OpenScape Contact Center usando um número de trânsito. Para a Retenção na URA, isso acontece normalmente logo depois que o status da chamada retornada pela API QueryCallStatus for “Pendente” ou “Não atendida”. Depois de transferir a chamada para o RCG do OpenScape Contact Center, este imediatamente desvia a chamada para o usuário designado ou número de time-out designado.

Se for solicitado um número de trânsito e o sistema da URA não puder transferir a chamada imediatamente, deverá ser chamado o API ReleaseTransitNumber. Isto torna o número de trânsito disponível para outra chamada. Antes de tentar transferir novamente a chamada para o OpenScape Contact Center, a API GetTransitNumber deverá ser chamada novamente, para obter um novo número de trânsito.

Se uma chamada em um sistema da URA não monitorada for desconectada (pelo sistema da URA ou por abandono do chamador) ou transferida para outro número que não seja um número de trânsito, a API CallerDisconnected será usada para notificar o OpenScape Contact Center.

---

**NOTA:** É necessário adicionar o API CallerDisconnected ao ramo de desligamento do sistema da URA e estar integrado em outros procedimentos de tratamento de erros, se necessário. Se você não adicionar a API ao ramo de desligamento, suas estatísticas não serão precisas, porque o OpenScape Contact Center não saberá quando a chamada terminou.

---

#### **Exemplo: sequência de etapas do sistema da URA não monitorada (Retenção na URA)**

A seguir encontra-se um exemplo de uma sequência de etapas do sistema da URA para um sistema da URA não monitorado:

1. Atender chamada entrante da URA.
2. Call Initialize(), seguido por SetCallInfo(), seguido por QueryCallInfo().
3. Reproduzir mensagem de boas-vindas/menu principal e processar a chamada até que ela esteja pronta para ser enfileirada no OpenScape Contact Center.
4. Call Enqueue().
5. Call QueryCallStatus() até que o status da chamada seja “Pendente” ou “Não atendida”.
6. Call GetTransitNumber().
7. Transferir imediatamente a chamada usando o número retornado em GetTransitNumber().

8. O OpenScape Contact Center roteará a chamada para o usuário designado ou o ramal de time-out designado.

As etapas exatas para qualquer sistema da URA podem ser diferentes das etapas acima. As coisas importantes a serem observadas são:

- A ordem na qual Initialize(), SetCallInfo() e QueryCallInfo() são chamadas.
- QueryCallInfo() é chamado assim que possível, para que o OpenScape Contact Center possa começar a rastrear a chamada a partir do momento em que ela chegar no sistema da URA.
- GetTransitNumber() é chamado somente quando a chamada estiver prestes a ser transferidas para fora do sistema da URA para um recurso do OpenScape Contact Center (como um usuário, RCG ou ramal de time-out), e a transferência deve ser completada imediatamente após GetTransitNumber() retornar com um número de trânsito.

## **Sobre a API URA do OpenScape Contact Center**

Funções da API URA para sistemas da URA não monitorados

## 3 A31003-S22A-N100-01-V420Configurando a API URA do OpenScape Contact Center

Este capítulo descreve os vários cenários da URA que você poderá usar para melhorar a coleta dos requisitos de seu cliente. Os cenários são:

- Retenção na URA
- Retenção na Fila
- Enfileiramento com colocação demarcada (FMNQ)

---

**NOTA:** Dependendo da experiência exigida pelo chamador, o script da URA poderá usar um ou mais desses cenários.

---

### 3.1 Configuração de retenção na URA

Você pode usar a configuração de Retenção na URA com um sistema da URA para reter uma chamada em fila até que ela seja roteada para um usuário.

---

**NOTA:** Como a função de Retenção na URA deixa o ramal da URA ocupado e evita temporariamente o atendimento de outras chamadas entrantes no sistema URA até que a chamada retida tenha sido colocada na fila de um usuário, deve-se modificar a configuração da Retenção na URA para que ela tenha capacidade de atender mais ramais.

---

O diagrama a seguir mostra o fluxo de chamadas para a configuração de Retenção na URA. Consulte o texto após o fluxograma de chamadas para obter uma descrição das etapas numeradas.

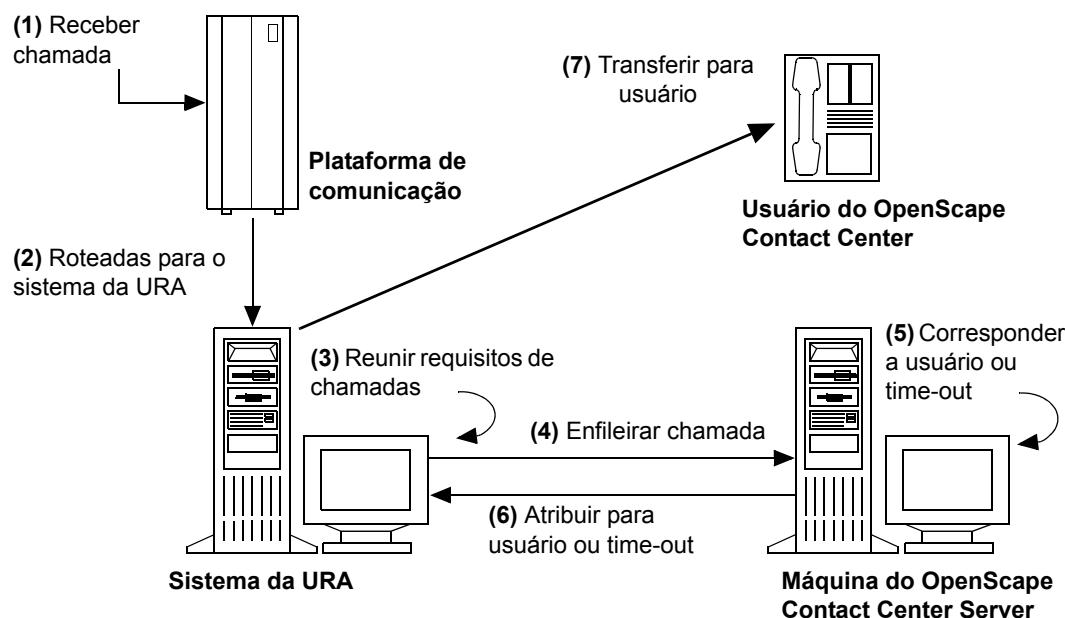


Figura 2 Fluxograma de chamadas para a configuração de Retenção na URA

1. Uma nova chamada chegou.
2. A chamada é roteada imediatamente para um sistema da URA.
3. O sistema URA coleta informações do chamador com relação ao objetivo da chamada e, depois, utiliza essas informações para identificar os parâmetros de roteamento para a chamada, incluindo a fila.
4. O sistema URA envia uma solicitação de chamada enfileirada para o OpenScape Contact Center. Então, o Routing Server procurará o melhor usuário disponível para tratar da chamada.
5. Depois, o OpenScape Contact Center fará uma das coisas a seguir:
  - Atribuirá a chamada ao melhor usuário disponível
  - Reservará a chamada para um usuário específico (opcional)

Se a chamada não puder ser atribuída a um usuário até o fim da última etapa da chamada na fila, ela será roteada para um ramal em time-out.

6. O OpenScape Contact Center informa ao sistema URA o ramal do usuário designado ou o destino de time-out. Se a chamada entrar em time-out, o sistema da URA executa uma das seguintes ações:
  - Transfere a chamada para um número padrão especificado pelo script da URA ou transfere a chamada para o ramal de time-out fornecido pelo Routing Server. O ramal de time-out pode ser um número específico definido pela fila da chamada ou um número padrão global usado por todas as filas que não tenham um número específico definido.
  - Avisa o cliente para obter informações adicionais e utiliza os novos parâmetros de roteamento para enfileirar a chamada novamente. Neste momento, o processo de corresponder a chamada a um usuário é repetido.
7. O sistema da URA transfere a chamada para o usuário designado.

---

**NOTA:** Quando conectado a uma plataforma de comunicação OpenScape Voice, caso a transferência falhe e a chamada seja reconectada ao sistema da URA, o aplicativo da URA precisa processá-la como uma nova chamada.

---

## 3.2 Configuração de retenção na fila

É possível usar a configuração de Retenção na fila com um sistema URA para transferir chamadas para um Grupo ACD/UCD/MLHG do OpenScape Contact Center na plataforma de comunicação, onde as chamadas aguardam até que um dos usuários qualificados fique disponível para atender às chamadas.

Quando um usuário estiver disponível, o OpenScape Contact Center desviará a chamada do grupo ACD/UCD/MLHG do OpenScape Contact Center para o usuário. A configuração de Retenção na Fila libera os ramais da URA para as chamadas entrantes, transferindo-as para a plataforma de comunicação enquanto o Routing Server procura usuários disponíveis.

Esta configuração reduz os requisitos do ramal da URA, para que o sistema da URA possa tratar de um maior número de chamadas entrantes.

O diagrama a seguir mostra o fluxo de chamadas para a configuração de Retenção na fila. Consulte o texto após o fluxograma de chamadas para obter uma descrição das etapas numeradas.

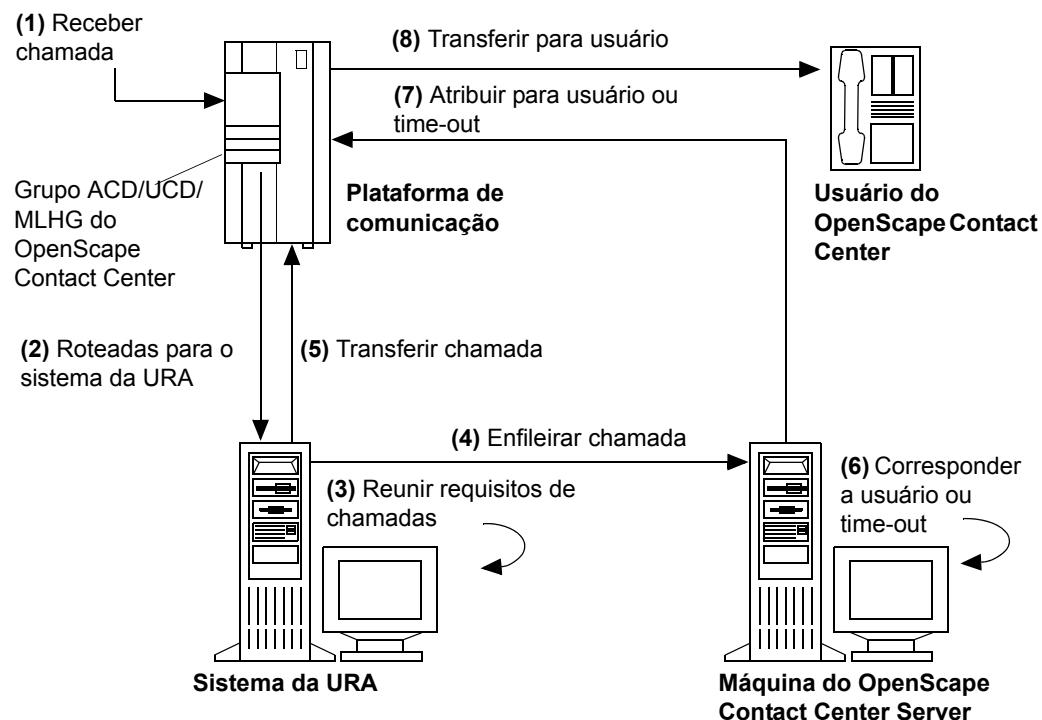


Figura 3

Chamar fluxograma da configuração de Retenção na Fila

1. Uma nova chamada chegou.
2. A chamada é roteada imediatamente para um sistema URA.
3. O sistema URA coleta informações do chamador com relação ao objetivo da chamada e, depois, utiliza essas informações para identificar os parâmetros de roteamento para a chamada, incluindo a fila.
4. O sistema URA envia uma solicitação de chamada enfileirada para o OpenScape Contact Center. Então, o Routing Server procurará o melhor usuário disponível para tratar da chamada.
5. Nesse momento, o sistema URA transfere a chamada para um grupo ACD/UCD/MLHG do OpenScape Contact Center.
6. Depois, o OpenScape Contact Center fará uma das coisas a seguir:
  - Atribuirá a chamada ao melhor usuário disponível
  - Reservará a chamada para um usuário específico (opcional)

Se a chamada não puder ser atribuída a um usuário até o fim da última etapa da chamada na fila, ela será roteadas para um ramal em time-out.

7. A chamada é designada para usuário específico ou destino de time-out.
8. A chamada será transferida para o usuário designado.

---

**NOTA:** Quando conectado a uma plataforma de comunicação OpenScape Voice, caso a transferência falhe e a chamada seja reconectada ao sistema da URA, o aplicativo da URA precisa processá-la como uma nova chamada.

---

### 3.3 Configuração de colocação demarcada

Você poderá definir uma chamada com colocação demarcada para que ela seja “interrompível” ou “não interrompível”. Se a chamada for definida como “interrompível”, o chamador poderá ser deslocado de um sistema da URA para o ramal do melhor usuário disponível, independentemente de estar ou não inserindo informações no sistema da URA. Se a chamada for “não interrompível”, seu roteamento estará temporariamente suspenso até que o chamador:

- insira informações críticas sem o risco de perdê-las
- ouça informações críticas sem interrupção

---

**NOTA:** Os recursos não monitorados da URA não são compatíveis com a FMNQ.

---

### 3.4 Escrevendo um script URA

Um script URA personalizado coleta informações de seus clientes, fornece funções de gerenciamento de chamadas e chama as funções API URA, que acompanham o OpenScape Contact Center.

As funções da API URA são fornecidas em um módulo chamado **hppcivr.dll**, que você deverá incluir ao executar o script da URA. O arquivo deste módulo está localizado na pasta padrão do OpenScape Contact Center, na máquina do servidor principal.

### 3.4.1 Gravando um script da URA para a configuração de Retenção na URA

O fluxograma abaixo é o sugerido para um script da URA para a configuração de retenção da URA:

1. Inicialize a conexão entre o sistema URA e o OpenScape Contact Center.
2. Verifique o status do sistema para certificar-se de que o Routing Server esteja disponível.
3. Colete as informações do ANI e DNIS para a chamada.
4. Identifique a fila utilizando o script da URA. Por exemplo, você pode atribuir uma fila à chamada com base em uma combinação das informações ANI da chamada e da seleção do cliente.
5. Enfileirar a chamada usando a função Enqueue. Assegure que os parâmetros especifiquem que o sistema URA reterá a chamada até que esta seja designada a um usuário, usando a configuração de retenção na URA. Isto notificará o Routing Server que o sistema da URA transferirá a chamada quando esta for atribuída.
6. Verifique regularmente o estado das chamadas enfileiradas (por exemplo, após cada ação ou chamada de função URA) e execute a ação adequada conforme os resultados:
  - Se a chamada for designada, transfira-a para o ramal do usuário.
  - Se ocorreu um time-out, transfira a chamada para o número padrão definido no script da URA ou para o ramal retornado pelo Routing Server.
  - Se houver um erro no Routing Server, transfira a chamada para o número padrão definido no script da URA.

O diagrama a seguir mostra um exemplo de script da URA para a configuração de Retenção na URA.

---

**NOTA:** Quando a chamada estiver enfileirada com sucesso, o script da URA deve verificar regularmente o estado da chamada, utilizando a função QueryCallStatus, como também resolver condições tais como erros de enfileiramento e chamadas em time-out. Nesses casos, o script da URA deverá transferir a chamada para um ramal que não pertença ao OpenScape Contact Center ou enfileirar a chamada com parâmetros diferentes.

---

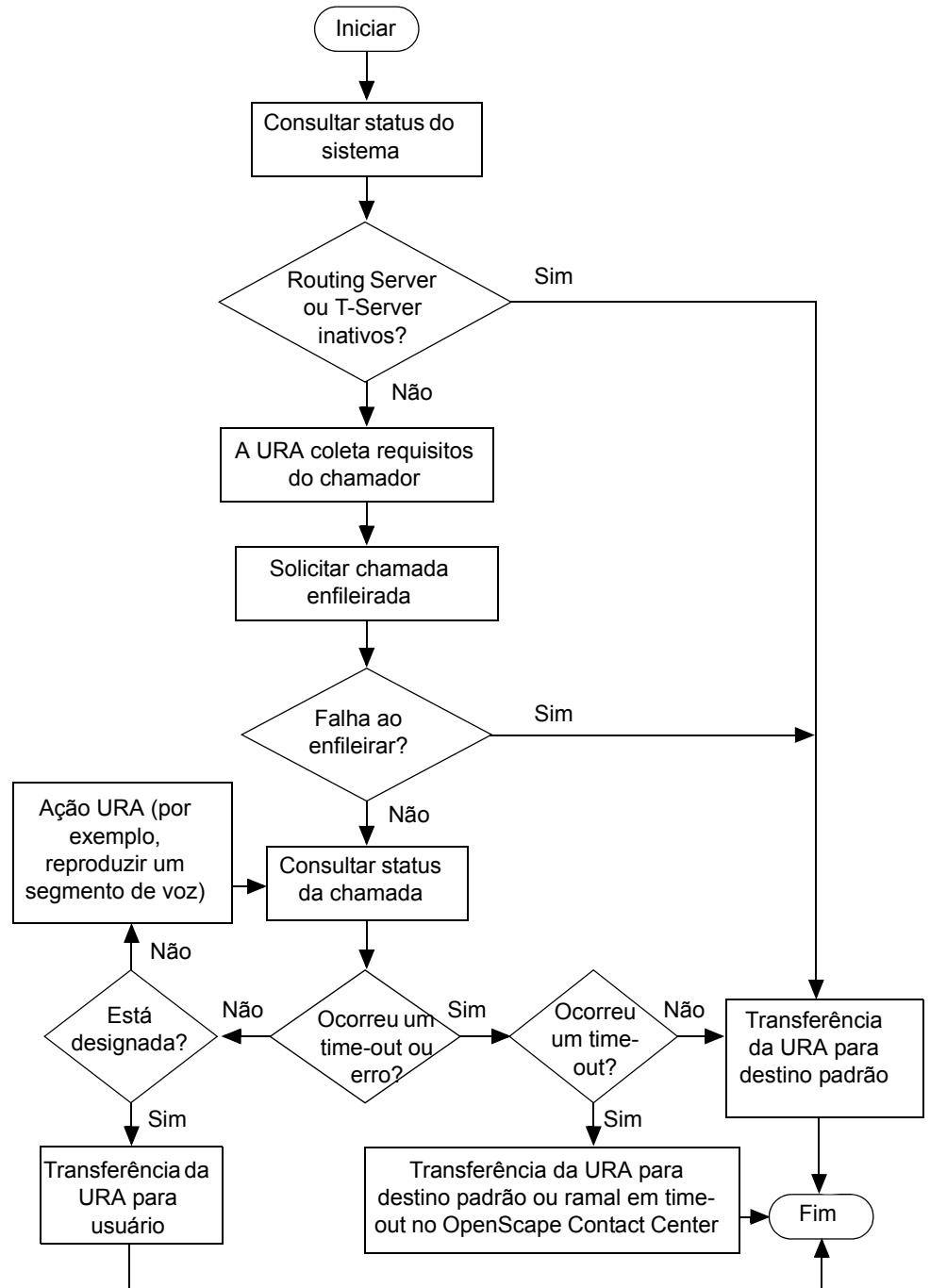


Figura 4

Exemplo de fluxograma para um script de URA na configuração de retenção da URA

### 3.4.2 Gravando um script da URA para a configuração de Retenção na fila

O fluxograma a seguir é uma sugestão para um script da URA personalizado para a configuração de Retenção na Fila:

1. Inicialize a conexão entre o sistema URA e o OpenScape Contact Center.
2. Verifique o status do sistema para certificar-se de que o Routing Server esteja disponível.
3. Colete as informações do ANI e DNIS para a chamada.
4. Identifique a fila utilizando o script da URA. Por exemplo, você pode designar uma fila para a chamada com base em uma combinação das informações ANI da chamada e nas seleções que o cliente fizer.
5. Enfileirar a chamada com a função Enqueue. Assegure que os parâmetros especifiquem que o sistema URA transferirá a chamada para um Grupo ACD/UCD/MLHG do OpenScape Contact Center, usando a configuração de Retenção na fila. Isto notificará o Routing Server de que o OpenScape Contact Center deve transferir a chamada quando esta for designada.
6. Transferir a chamada para um grupo ACD/UCD/MLHG do OpenScape Contact Center. O OpenScape Contact Center retém automaticamente a chamada no Grupo ACD/UCD/MLGH do OpenScape Contact Center até que a chamada seja designada para um usuário.

O diagrama a seguir mostra um exemplo de script da URA para a configuração de Retenção na fila.

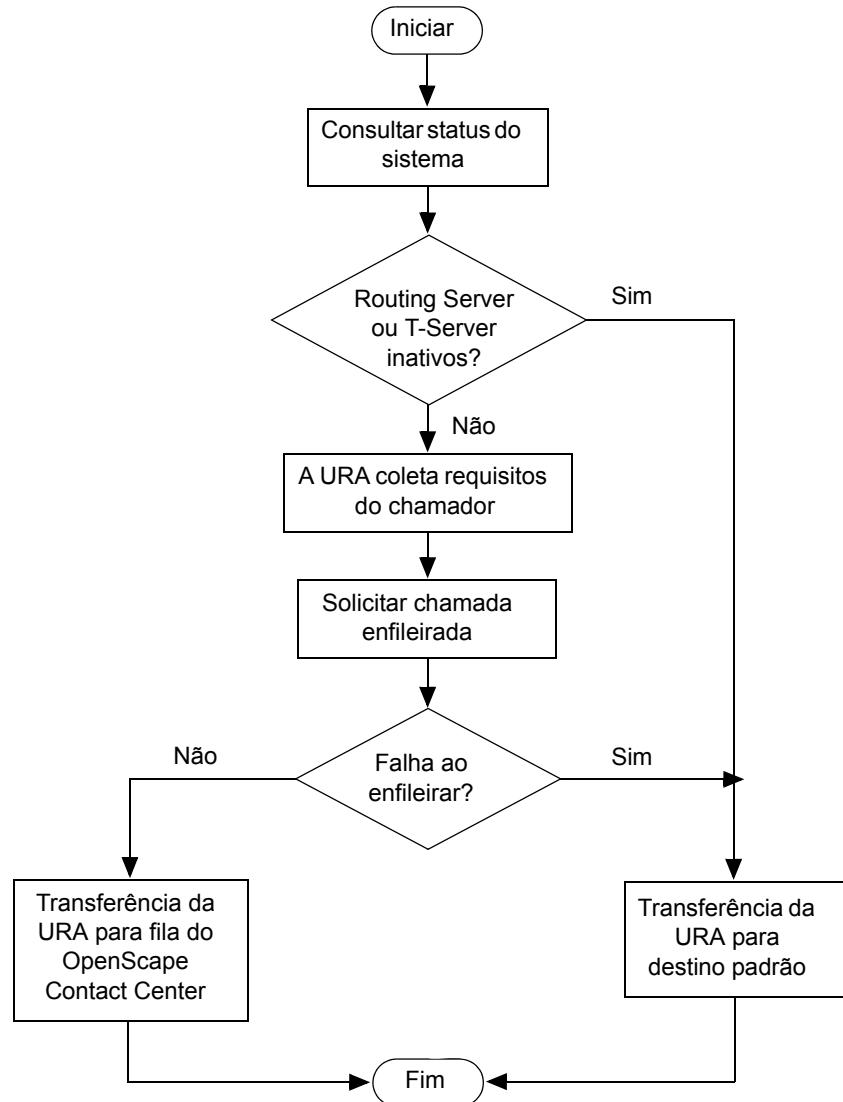


Figura 5

Exemplo de fluxograma de um script da URA para a configuração de Retenção na Fila

### 3.4.3 Gravando um script da URA para a configuração de FMNQ

O fluxograma a seguir é uma sugestão para um script de URA personalizado para a configuração com colocação demarcada:

1. Inicialize a conexão entre o sistema URA e o OpenScape Contact Center.
2. Use a função QueryCallInfo para coletar as informações de ANI e DNIS para a chamada e para obter o valor da flag com colocação demarcada.
3. Se a flag estiver definida para 1, inicie o fluxograma de chamadas com colocação demarcada e verifique se o chamador deseja entrar em uma sessão “interrompível” ou “não interrompível”.
4. Se o chamador deseja entrar em uma sessão “interrompível”, defina a chamada para não-transferível (0).
5. Continue a coletar os requisitos do chamador dentro da sessão “não interrompível”.
6. Quando a sessão “não interrompível” tiver sido concluída, defina a chamada para transferível (1).

---

**NOTA:** Quando estiver conectado a uma plataforma de comunicação OpenScape Voice, será necessário assegurar que a chamada seja transferida para o Grupo de busca de Música de espera.

---

---

**NOTA:** Antes que o script da URA desconecte ou transfira uma chamada com colocação demarcada, você deverá emitir o SetCallTransferable (0) e receber um código de retorno bem sucedido. Se você não receber um código de retorno bem sucedido, isto indicará que a chamada está em processo de ser transferida e será necessário aguardar até que o processo esteja completo.

---

O diagrama a seguir mostra um exemplo de script da URA para a configuração de FMNQ.

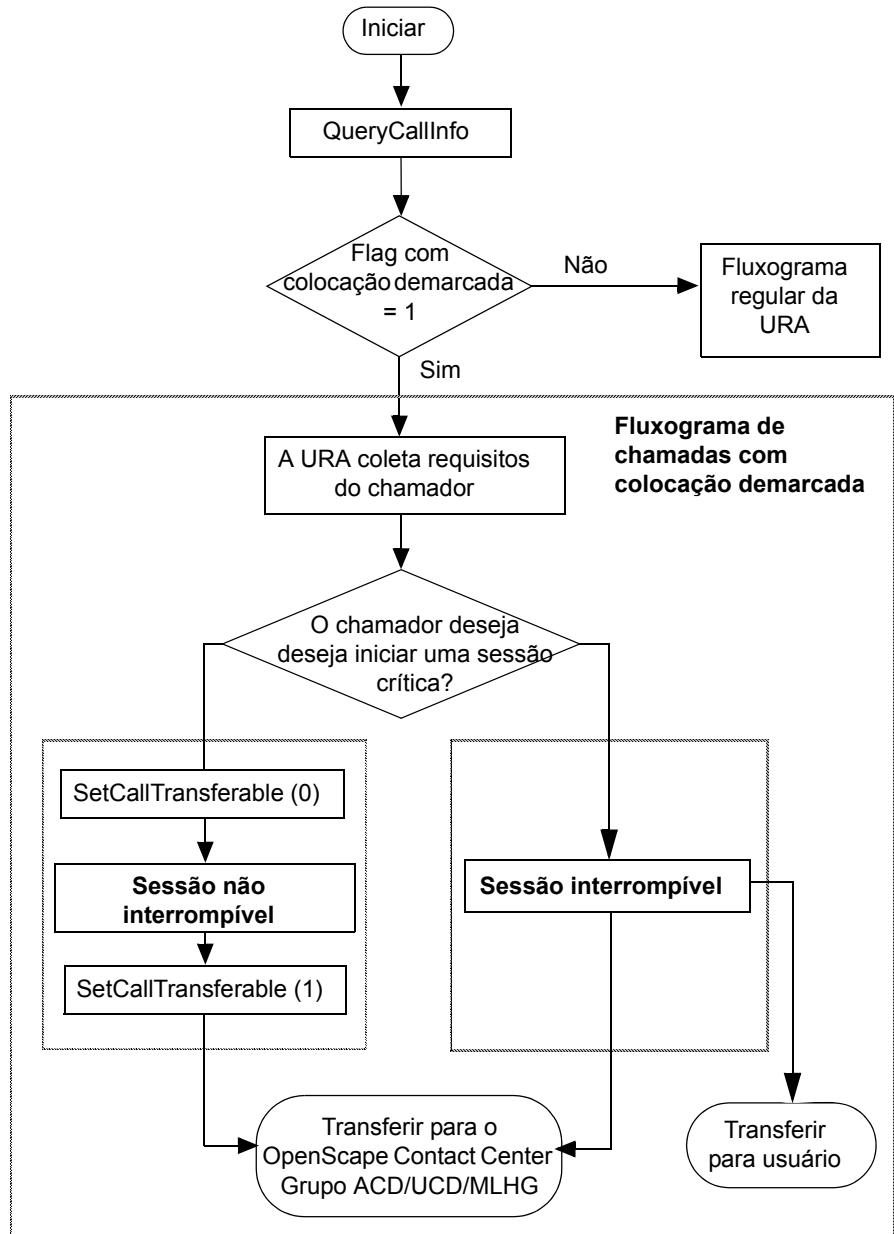


Figura 6

Exemplo de um fluxograma de script da URA para a configuração com colocação demarcada

### 3.5 Usando um sistema da URA em um ambiente de multilocação

Em um sistema multilocação, os recursos da URA são recursos de nível de sistema compartilhados entre várias unidades de negócios.

Se você estiver implementando um aplicativo front-end da URA em um ambiente de multilocação, o sistema URA deve conter a lógica necessária para que ele tome conhecimento das unidades de negócios.

Por exemplo:

- O sistema da URA necessita determinar para qual unidade de negócios uma chamada da URA será roteada.
- Quando um sistema da URA enfileira uma chamada no sistema OpenScape Contact Center, o aplicativo deve enfileirar a chamada em relação à unidade de negócios correta.
- No cenário de Retenção na fila, um sistema da URA necessita conhecer o número piloto correto para cada unidade de negócios, e necessita transferir a chamada para a unidade de negócios correta.

Duas funções da API são fornecidas para utilização em um ambiente de multilocação:

- Função GetBusinessUnit - recupera a unidade de negócios para uma chamada especificada. Para obter mais informações, consulte [Seção 4.9, “GetBusinessUnit”, na página 41](#).
- Função SetBusinessUnit - define a unidade de negócios para uma chamada especificada. Função SetBusinessUnit pode ser chamada somente uma vez para cada chamada da URA. Para obter mais informações, consulte [Seção 4.21, “SetBusinessUnit”, na página 61](#).

---

**NOTA:** Em um ambiente de multilocação, a função SetBusinessUnit deve ser chamado antes de as seguintes funções serem chamados: CreateCallback, Enqueue, EnqueueForAgent, GetBusinessUnit.

---

Um parâmetro BusinessUnitName opcional pode ser especificado na função QueryQueueStatistics para indicar a unidade de negócios à qual uma fila pertence. Em um ambiente que não seja de multilocação, o parâmetro BusinessUnitName é ignorado.

## 4 Usando as funções da API URA do OpenScape Contact Center

Este capítulo descreve como usar as funções da API URA do OpenScape Contact Center e fornece exemplos de como criar scripts da URA.

---

**NOTA:** Um sistema da URA não monitorado é um sistema da URA que não é monitorado pelo OpenScape Contact Center.

---

### 4.1 Passando parâmetros

Ao chamar as funções do OpenScape Contact Center API URA, você poderá passar informações por valor ou por referência. As seções a seguir descrevem esses dois métodos.

#### Para passar parâmetros por valor.

Ao passar parâmetros por valor, você estará passando informações do script da URA para o OpenScape Contact Center. Cada uma das funções da API URA exige que sejam enviadas informações diferentes ao OpenScape Contact Center, para que o aplicativo possa executar a ação solicitada. Por exemplo, o script da URA deverá passar o nome da fila para o OpenScape Contact Center ao chamar a função Enqueue, para que o aplicativo possa enfileirar a chamada.

#### Para passar parâmetros por referência

Ao passar parâmetros por referência, você estará passando informações do OpenScape Contact Center para o script da URA. Ao chamar uma função da API URA, o script da URA especifica um nome de variável para as variáveis passadas por referência. O OpenScape Contact Center retorna as informações necessárias na variável especificada. Por exemplo, o OpenScape Contact Center retorna as informações ANI e DNIS para uma chamada quando o script da URA chamar a função QueryCallInfo.

---

**NOTA:** Os parâmetros que forem passados por referência deverão ser precedidos pelo símbolo “maior que” (>).

---

## 4.2 CallerDisconnected

---

**NOTA:** Esta função somente é usada por scripts de URA para sistemas da URA não monitorados.

---

A função CallerDisconnected permite que sistemas da URA notifiquem o OpenScape Contact Center sobre uma chamada desconectada. A seguir, o T-Server exclui a chamada e libera o número de trânsito (se for o caso) associado com a chamada. O T-Server então envia uma notificação a outros componentes do OpenScape Contact Center.

### Sintaxe

```
CallerDisconnected (CallID, TermType)
```

### Parâmetros

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	A ID da Chamada para a chamada atual.
TermType	Inteiro passado por valor	0, 1 ou 2	0 – Indica que a chamada foi concluída como resultado de um problema interno. 1 – Indica que a chamada foi transferida fora de escopo. 2 - Indica que a chamada foi abandonada.

*Tabela 1 Parâmetros para a função CallerDisconnected*

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

- 0** Bem-sucedida
- 308** QueryCallInfo não foi chamado antes de chamar esta função. Você deve chamar QueryCallInfo para obter uma ID da chamada.
- 310** Essa função só pode ser usada para chamadas URA não monitoradas.
- 910** A versão do API URA DLL não corresponde à versão do OpenScape Contact Center.
- 931** Há uma inconsistência na monitoração do tipo de chamada entre a API URA e o T-Server.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

### Exemplo

```
CallerDisconnected ('1341023126268002', 2)
```

## 4.3 CreateCallback

A função CreateCallback tenta criar um callback no Callback Server. Um sistema da URA pode indicar que um cliente deverá ser chamado novamente quando um usuário ficar disponível. Esta função é compatível com no máximo 1000 bytes de dados do contato.

---

**NOTA:** Será necessário chamar as funções Inicializar e QueryCallInfo antes de chamar esta função. Em um ambiente de multilocação, também é necessário chamar a função SetBusinessUnit antes de chamar esta função. Para obter mais informações, consulte [Seção 4.13, “Initialize”, na página 46](#), [Seção 4.15, “QueryCallInfo”, na página 49](#) e [Seção 4.21, “SetBusinessUnit”, na página 61](#).

---

### Sintaxe

CreateCallback (CallID, QueueName, Description, TimeZoneOffset, ContactName, >CallbackID, Schedule1, Schedule2, Schedule3)

### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por referência	18 caracteres	A ID de chamada retornada de QueryCallInfo deverá ser fornecida como entrada.
QueueName	Seqüência passada por valor	32 caracteres	A fila de callbacks para usar para o callback.
Description (opcional)	Seqüência passada por referência	100 caracteres	A seqüência contém uma breve descrição da chamada, que é exibida no aplicativo OpenScape Contact Center Client Desktop, para que os usuários possam identificar a chamada.
TimeZoneOffset	Seqüência passada por valor	0 a +60*12 ou 0 a -60*12	A diferença em minutos entre o horário local do chamador e UTC (Coordinated Universal Time).
ContactName	Seqüência passada por valor	80 caracteres	O nome do contato.
CallbackID	Seqüência passada por referência	17 caracteres	Uma ID exclusiva para o callback criado.

Tabela 2

Parâmetros para a função CreateCallback

## Usando as funções da API URA do OpenScape Contact Center

### CreateCallback

Nome	Tipo	Intervalo	Descrição
Schedule1	Seqüência passada por valor	47 caracteres para o número de telefone. 81 caracteres para toda a sequência.	Uma sequência indicando o número de telefone no qual o chamador pode ser atingido e o período de tempo (no horário local do chamador) durante o qual o chamador deseja receber callbacks. O formato is: número de telefone;mm/dd/aaaa;hh:mm; mm/dd/aaaa;hh:mm onde a primeira data e horário (usando um relógio de 24 horas) é o horário inicial, e a segunda é o horário final.
Schedule2 (opcional)	Seqüência passada por valor	47 caracteres para o número de telefone. 81 caracteres para toda a sequência.	Uma sequência indicando o número de telefone no qual o chamador pode ser atingido e o período de tempo (no horário local do chamador) durante o qual o chamador deseja receber callbacks. O formato is: número de telefone;mm/dd/aaaa;hh:mm; mm/dd/aaaa;hh:mm onde a primeira data e horário (usando um relógio de 24 horas) é o horário inicial, e a segunda é o horário final.
Schedule3 (opcional)	Seqüência passada por valor	47 caracteres para o número de telefone. 81 caracteres para toda a sequência.	Uma sequência indicando o número de telefone no qual o chamador pode ser atingido e o período de tempo (no horário local do chamador) durante o qual o chamador deseja receber callbacks. O formato is: número de telefone;mm/dd/aaaa;hh:mm; mm/dd/aaaa;hh:mm onde a primeira data e horário (usando um relógio de 24 horas) é o horário inicial, e a segunda é o horário final.

Tabela 2

Parâmetros para a função CreateCallback

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

- 0** Bem-sucedida
- 308** QueryCallInfo não foi chamado antes de chamar esta função. Você deve chamar QueryCallInfo para obter uma ID da chamada.
- 314** O nome da unidade de negócios especificado não estava definido ou não correspondeu ao nome de nenhuma unidade de negócios na base de dados. (Este erro só se aplica quando o OpenScape Contact Center estiver configurado como um sistema multilocação.)
- 1006** Você está tentando criar um callback duplicado
- 1028** Um horário de programação é inválido.
- 1029** As programações de callbacks não possuem um horário mútuo com o horário de funcionamento do central de contatos.

- 1031** Um número de telefone na solicitação está na lista Números Excluídos.
- 1033** Um callback não pode aceitar dados do contato com mais de 1.000 bytes.
- 1040** Todas as programações já expiraram.
- 1045** O nome de fila de callbacks é inválido.
- 1047** A programação contém uma data muito distante no futuro.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

### Exemplo

O exemplo a seguir solicita que o OpenScape Contact Center crie um callback para a chamada atual (identificada pela ID da chamada) usando “CBQueue” como a fila de callbacks.

```
CreateCallback(CallID, 'CBQueue', 'Callback', 0, 'Alan Smith',  
>CBCallID,'5555550199;06/10/2010;11:00;06/11/2010;16:00',  
'5555550199;06/11/2010;12:00;06/11/2010;14:00')
```

A ID do callback é retornada depois que o callback for criado.

### Formato do número de telefone

Os parâmetros do número de telefone devem ser inseridos no formato canônico, da seguinte forma:

```
+[CountryCode] Espaço [(AreaCode) Espaço SubscriberNumber [-Ramal]]
```

O número de telefone também pode ser um endereço discável obtido de um plataforma de comunicação ou o resultado da chamada de uma função TAPI. As sequências a seguir são números de telefone corretos:

- +1 (555) 555-0199
- (555) 555-0199
- +1 555-0199
- +1 (555) 555-0199--1212
- 5555550199

A parte do ramal de um número de telefone é truncada pelo Callback Server antes de ser enviada ao T-Server. Ele só é usado ao exibir o callback nos aplicativos de Client Desktop e Manager, de forma que um usuário possa discar manualmente o ramal, se necessário.

## 4.4 DeleteCallback

A função DeleteCallback é usada para excluir um callback existente que foi criado com a função CreateCallback. Essa função toma como parâmetro a ID do callback retornada pela função CreateCallback.

---

**NOTA:** Será necessário chamar as funções Inicializar e QueryCallInfo antes de chamar esta função. Para obter mais informações, consulte a [Seção 4.13, “Initialize”, na página 46](#) e a [Seção 4.15, “QueryCallInfo”, na página 49](#).

---

### Sintaxe

```
DeleteCallback (CallbackID)
```

### Parâmetros

Nome	Tipo	Intervalo	Descrição
CallbackID	Seqüência passada por valor	17 caracteres	A ID do callback (CallbackID) retornada de uma solicitação CreateCallback bem sucedida, que deverá ser excluída.

*Tabela 3 Parâmetros para a função CreateCallback*

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

**0** Bem-sucedida

**-1021** ID do callback (CallbackID) inválido.

**Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

### Exemplo

O exemplo a seguir solicita que o callback com a ID da chamada de `CBCallID` seja excluído.

```
DeleteCallback(CBCallID)
```

## 4.5 Dequeue

A função Desenfileirar envia uma solicitação ao Routing Server para desenfileirar a chamada especificada. Se você estiver usando a configuração de retenção da URA, poderá usar a função Desenfileirar para desenfileirar uma chamada a fim de que o script da URA possa transferir a chamada para um ramal ou executar outra ação com a chamada.

Se o script da URA permitir que o chamador continue dando entrada a escolhas depois que a chamada for desenfileirada, você poderá usar outras entradas do chamador para alterar o roteamento da chamada. Por exemplo, se os chamadores esperarem por muito tempo na fila, eles poderão decidir deixar uma mensagem de voz; o script da URA pode, então, redirecionar em fila a chamada e transferi-la para um ramal de correio de voz.

---

**NOTA:** Será necessário chamar as funções Inicializar e QueryCallInfo antes de chamar esta função. Para obter mais informações, consulte a [Seção 4.13, “Initialize”, na página 46](#) e a [Seção 4.15, “QueryCallInfo”, na página 49.](#)

---

### Sintaxe

Dequeue (CallID)

### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	A ID da chamada para a chamada que se deseja.

*Tabela 4 Parâmetros para a função Dequeue*

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

- 0** Bem-sucedida
- 304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.
- 306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.
- 308** QueryCallInfo não foi chamado antes de chamar esta função. Você deve chamar QueryCallInfo para obter uma ID da chamada.
- 701** A solicitação expirou antes que pudesse ser concluída. Isso pode ser um indicativo de problemas na rede ou um servidor ocupado.

## Usando as funções da API URA do OpenScape Contact Center

### Disconnect

**-801** Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.

**Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

#### Exemplo

O exemplo a seguir envia uma solicitação para desenfileirar uma chamada.

```
Dequeue ('1341023126268002')
```

## 4.6 Disconnect

A função Disconnect desconecta a chamada especificada no ramal da URA.

---

**NOTA:** Essa função não é compatível com chamadas da URA não monitorada.

---

**NOTA:** Será necessário chamar as funções Inicializar e QueryCallInfo antes de chamar esta função. Para obter mais informações, consulte a [Seção 4.13, “Initialize”](#), na página 46 e a [Seção 4.15, “QueryCallInfo”](#), na página 49.

#### Sintaxe

```
Disconnect (CallID)
```

#### Parâmetros

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	A ID da chamada para a chamada que você deseja desconectar.

*Tabela 5 Parâmetros para a função Disconnect*

#### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

**0** Bem-sucedida

**-304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.

**-306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.

**-311** A função não é compatível com chamadas da URA não monitorada.

- 701** A solicitação da função expirou antes que a função pudesse ter sido completada. Normalmente, isto só ocorre ao consultar o estado de uma chamada. Também pode ser uma indicação de problemas de rede ou que o servidor está ocupado.
- 801** Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.
- 903** Não havia nenhuma chamada associada com a ID da chamada. Certificar-se de que todos os ramais URA estão indicados na base de dados.
- 914** Falha na operação que você está tentando executar.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

#### Exemplo

```
Disconnect ('1341023126268002')
```

## 4.7 Enqueue

A função Enqueue solicita ao Routing Server para enfileirar uma chamada e encontrar o usuário mais disponível para atender à chamada. Esta função passa as informações de roteamento de uma chamada para o Routing Server, inclusive a prioridade inicial da fila e da chamada.

---

**NOTA:** Se você quiser associar os dados dos contatos com a chamada, será necessário configurar esses parâmetros antes de enfileirar a chamada.

---

Use esta função para enfileirar chamadas na configuração de retenção da URA ou de retenção de fila. Após uma chamada ser enfileirada na configuração de Retenção na URA, o script da URA deverá verificar se a chamada foi designada usando a função `QueryCallStatus` e, depois, transferir a chamada para o ramal do usuário. Depois de ter sido enfileirada na configuração de Retenção na Fila, a chamada deverá ser transferida imediatamente ao MLHG grupo ACD/UCD/MLHG do OpenScape Contact Center.

---

**NOTA:** Será necessário chamar as funções `Iniciar` e `QueryCallInfo` antes de chamar esta função. Em um ambiente de multilocação, também é necessário chamar a função `SetBusinessUnit` antes de chamar esta função. Para obter mais informações, veja [Seção 4.13, “Initialize”, na página 46](#), [Seção 4.15, “QueryCallInfo”, na página 49](#) e [Seção 4.21, “SetBusinessUnit”, na página 61](#).

---

#### Sintaxe

```
Enqueue (CallID, IVRHold, QueueName, InitialPriority,  
Description,>EstimatedWait, >CallsinQueue, ANIDNIS)
```

## Usando as funções da API URA do OpenScape Contact Center

### Enqueue

#### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	A ID da Chamada para a chamada que se deseja enfileirar.
IVRHold	Seqüência passada por valor	0 ou 1	0 - Indica que a chamada será transferida para um grupo ACD/UCD/MLHG para a configuração Retenção na Fila. 1 - Indica que a chamada aguardará no ramal da URA pela configuração de Retenção na URA.
QueueName	Seqüência passada por valor	32 caracteres	O nome da fila para a chamada.
InitialPriority	Seqüência passada por valor	1 a 100	Especifica a prioridade de uma chamada, onde 1 é a menor prioridade e 100 é a maior.
Description (opcional)	Seqüência passada por valor	100 caracteres	A seqüência contém uma breve descrição da chamada, que é exibida no aplicativo OpenScape Contact Center Client Desktop, para que os usuários possam identificar a chamada.
EstimatedWait	Inteiro passado por referência	Maior ou igual a 0	O tempo de espera estimado em segundos para esta fila.
CallsInQueue	Inteiro passado por referência	Maior ou igual a 0	O número de chamadas na fila para a fila especificada.
ANIDNIS (opcional)	Seqüência passada por valor	161	Os números ANI e DNIS da chamada. O Statistics Server armazenará esses números na base de dados do OpenScape Contact Center. O ANI e o DNIS devem ser separados por um ponto e vírgula.

*Tabela 6 Parâmetros para a função Enqueue*

#### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

**0** Bem-sucedida

**-304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.

- 305** Esse código indica um erro desconhecido (por exemplo, erro do usuário ou o sistema está instável).
- 306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.
- 314** O nome da unidade de negócios especificado não estava definido ou não correspondeu ao nome de nenhuma unidade de negócios na base de dados. (Este erro só se aplica quando o OpenScape Contact Center estiver configurado como um sistema multilocação.)
- 701** A solicitação expirou antes que pudesse ser concluída. Isso pode ser um indicativo de problemas na rede ou um servidor ocupado.
- 801** Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.
- 903** Não havia nenhuma chamada associada com a ID da chamada. Certificar-se de que todos os ramais URA estão indicados na base de dados.
- 905** A fila especificada não é idêntica a nenhum nome de fila na base de dados.
- 907** O T-Server não está disponível. Este erro indica que o Routing Server não pôde colocar a chamada em fila, porque o T-Server não estava disponível.
- 908** O T-Server não está disponível.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

### Exemplo

O exemplo a seguir solicita enfileirar a chamada com a ID da chamada especificada.

```
Enqueue('1341023126268002', 1, 'Sales', 12, 'Call enqueued from
IVR', >EstWaitTime, >CallInQueue, '9055557900;69030')
```

## 4.8 EnqueueForAgent

A função EnqueueForAgent solicita ao Routing Server que enfileire uma chamada para um usuário e horário especificados. Se o AgentWaitTime expirar, a chamada não ficará mais reservada para um usuário especificado e será enfileirada na fila especificada pelo QueueName. Esta função passa as informações de roteamento de uma chamada para o Routing Server, inclusive a prioridade inicial da fila e da chamada. A função também passa ao Routing Server as informações que aparecem no aplicativo do OpenScape Contact Center Client Desktop, tais como a descrição.

Use esta função para enfileirar chamadas na configuração de retenção da URA ou de retenção de fila. Após uma chamada ser enfileirada na configuração de Retenção na URA, o script da URA deverá verificar se a chamada foi designada usando a função `QueryCallStatus` e, depois, transferir a chamada para o ramal

## Usando as funções da API URA do OpenScape Contact Center

### EnqueueForAgent

do usuário. Depois de ter sido enfileirada na configuração de Retenção na Fila, a chamada deverá ser transferida imediatamente ao MLHG grupo ACD/UCD/MLHG do OpenScape Contact Center.

---

**NOTA:** Será necessário chamar as funções Inicializar e QueryCallInfo antes de chamar esta função. Em um ambiente de multilocação, também é necessário chamar a função SetBusinessUnit antes de chamar esta função. Para obter mais informações, veja [Seção 4.13, “Initialize”, na página 46](#), [Seção 4.15, “QueryCallInfo”, na página 49](#) e [Seção 4.21, “SetBusinessUnit”, na página 61](#).

---

#### Sintaxe

```
EnqueueForAgent (CallID, IVRHold, QueueName, InitialPriority,  
Description, AgentID, AgentWaitTime, >EstimatedWait,  
>CallsinQueue, ANIDNIS)
```

#### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	A ID da Chamada para a chamada que se deseja enfileirar.
IVRHold	Inteiro passado por valor	0 ou 1	0 - Indica que a chamada será transferida para um grupo ACD/UCD/MLHG para a configuração Retenção na Fila. 1 - Indica que a chamada aguardará no ramal da URA pela configuração de Retenção na URA.
QueueName	Seqüência passada por valor	32 caracteres	O nome da fila para a chamada.
InitialPriority	Inteiro passado por valor	1 a 100	Especifica a prioridade de uma chamada, onde 1 é a menor prioridade e 100 é a maior.
Description (opcional)	Seqüência passada por valor	100 caracteres	A seqüência contém uma breve descrição da chamada, que é exibida no aplicativo OpenScape Contact Center Client Desktop, para que os usuários possam identificar a chamada.
AgentID	Seqüência passada por valor	8 caracteres	O usuário para o qual se deseja reservar a chamada.

Tabela 7

Parâmetros para a função EnqueueForAgent

Nome	Tipo	Intervalo	Descrição
AgentWaitTime	Inteiro passado por valor	Maior que 0	O tempo máximo em segundos em que a chamada poderá aguardar por um usuário reservado. Se este tempo expirar, a chamada será enfileirada na fila especificada pelo QueueName.
EstimatedWait	Inteiro passado por valor	Maior ou igual a 0	O tempo de espera estimado em segundos para esta fila.
CallsInQueue	Inteiro passado por valor	Maior ou igual a 0	O número de chamadas na fila para a fila especificada.
ANIDNIS (opcional)	Seqüência passada por valor	161 caracteres	Os números ANI e DNIS da chamada. O Statistics Server armazenará esses números na base de dados do OpenScape Contact Center. O ANI e o DNIS devem ser separados por um ponto e vírgula.

Tabela 7

Parâmetros para a função *EnqueueForAgent*

## Usando as funções da API URA do OpenScape Contact Center

### EnqueueForAgent

#### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

- 0** Bem-sucedida
- 304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.
- 305** Esse código indica um erro desconhecido (por exemplo, erro do usuário ou o sistema está instável).
- 306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.
- 314** O nome da unidade de negócios especificado não estava definido ou não correspondeu ao nome de nenhuma unidade de negócios na base de dados. (Este erro só se aplica quando o OpenScape Contact Center estiver configurado como um sistema multilocação.)
- 701** A solicitação expirou antes que pudesse ser concluída. Isso pode ser um indicativo de problemas na rede ou um servidor ocupado.
- 801** Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.
- 903** Não havia nenhuma chamada associada com a ID da chamada. Certificar-se de que todos os ramais URA estão indicados na base de dados.
- 904** A ID de agente reserva especificada não é idêntica a nenhuma ID dos agentes na base de dados.
- 905** A fila especificada não é idêntica a nenhum nome de fila na base de dados.
- 907** O T-Server não está disponível. Este erro indica que o Routing Server não pôde colocar a chamada em fila, porque o T-Server não estava disponível.
- 908** O T-Server não está disponível.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

#### Exemplo

O exemplo a seguir solicita enfileirar a chamada para o usuário 790006 durante 120 segundos, depois dos quais ela será enfileirada para “CT\_Pilot1”.

```
EnqueueForAgent ('7551024348478002', 0, 'Support', 8, 'Call reserved by IVR for user', '790006', 120, >EstWaitTime, >CallInQueue)
```

## 4.9 GetBusinessUnit

---

**NOTA:** A função GetBusinessUnit só é compatível em um ambiente de multilocação.

---

A função GetBusinessUnit recupera a unidade de negócios para a chamada especificada.

---

**NOTA:** Será necessário chamar as funções Inicializar, QueryCallInfo e SetBusinessUnit antes de chamar esta função. Para obter mais informações, consulte a Seção 4.13, “Initialize”, na página 46, a Seção 4.15, “QueryCallInfo”, na página 49 e a Seção 4.21, “SetBusinessUnit”, na página 61.

---

### Sintaxe

GetBusinessUnit (CallID, >BusinessUnitName)

### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	A CallID da chamada para a qual se deseja recuperar a unidade de negócios.
BusinessUnit Name	Seqüência passada por referência	32 caracteres	O nome da unidade de negócios para a chamada especificada.

*Tabela 8 Parâmetros para a função GetBusinessUnit*

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

- 0** Bem-sucedida
- 306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.
- 400** O sistema tem poucos recursos.
- 914** Falha na operação que você está tentando executar.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

#### Exemplo

O exemplo a seguir recupera o nome da unidade de negócios para a CallID especificada.

```
GetBusinessUnit ('8271023217459002', >BusinessUnitName)
```

## 4.10 GetCallTransferable

A função GetCallTransferable retorna o valor da flag configurado pela função SetCallTransferable e determina se a chamada pode ser transferida pelo OpenScape Contact Center para um usuário designado.

---

**NOTA:** Essa função não é compatível com chamadas da URA não monitorada.

---

---

**NOTA:** Será necessário chamar as funções Inicializar e QueryCallInfo antes de chamar esta função. Para obter mais informações, consulte a [Seção 4.13, “Initialize”, na página 46](#) e a [Seção 4.15, “QueryCallInfo”, na página 49](#).

---

#### Sintaxe

```
GetCallTransferable (CallID, >FlagValue)
```

#### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	A ID da Chamada para a chamada que se deseja transferir.
FlagValue	Inteiro passado por referência	0 ou 1	Indica se uma chamada pode ser transferida para um usuário designado. Se estiver configurado como false (0), o sistema suspenderá a correspondência de um usuário com a chamada até que a flag seja configurada como true (1). Se a chamada for transferida de volta para a fila pela URA, a flag será redefinida automaticamente como verdadeira.

Tabela 9

Parâmetros para a função GetCallTransferable

#### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

**0** Bem-sucedida

- 304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.
- 306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.
- 311** A função não é compatível com chamadas da URA não monitorada.
- 701** A solicitação expirou antes que pudesse ser concluída. Isso pode ser um indicativo de problemas na rede ou um servidor ocupado.
- 801** Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

### Exemplo

O exemplo a seguir retorna se a chamada puder ser transferida pelo OpenScape Contact Center para um usuário designado.

```
GetCallTransferable ('0831023896468002', >FlagValue)
```

## 4.11 GetContactData

A função GetContactData recupera os dados do contato para uma ID de chamada especificada. Se você quiser associar os dados dos contatos com a chamada, será necessário configurar os dados dos contatos antes de enfileirar a chamada.

---

**NOTA:** Será necessário chamar as funções Inicializar e QueryCallInfo antes de chamar esta função. Para obter mais informações, consulte a [Seção 4.13, “Initialize”, na página 46](#) e a [Seção 4.15, “QueryCallInfo”, na página 49](#).

---

### Sintaxe

```
GetContactData (CallID, Key, >Value)
```

### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	O ID da chamada para a chamada para a qual se deseja obter os dados dos contatos.

Tabela 10

Parâmetros para a função GetContactData

## Usando as funções da API URA do OpenScape Contact Center

### GetContactData

Nome	Tipo	Intervalo	Descrição
Key	Seqüência passada por valor	32 caracteres	O nome da chave para o valor dos dados do contato.
Valor	Seqüência passada por referência	128 caracteres	O valor retornado para a chave.

*Tabela 10 Parâmetros para a função GetContactData*

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

- 0** Bem-sucedida
- 304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.
- 306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.
- 307** O nome da chave para o valor dos dados do contato não foi definido antes de chamar esta função.
- 701** A solicitação expirou antes que pudesse ser concluída. Isso pode ser um indicativo de problemas na rede ou um servidor ocupado.
- 801** Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

### Exemplo

O exemplo a seguir recupera os dados do contato para a chave “Nome” para a ID de chamada (CallID) especificada.

```
GetContactData ('8271023217459002', 'Name', >Value)
```

## 4.12 GetTransitNumber

---

**NOTA:** Esta função somente é usada por scripts de URA para sistemas da URA não monitorados.

---

A função GetTransitNumber consulta o OpenScape Contact Center quanto ao número de trânsito. O sistema da URA deve transferir a chamada para o número de trânsito recebido nesta solicitação. O número de trânsito expirará se a chamada não for transferida dentro de 30 segundos. O número de trânsito ficará então disponível para ser usado com outra chamada.

---

**NOTA:** Para Retenção na fila, essa função é chamada após chamar a função Enqueue. Para Retenção na URA, ela é chamada após receber o status Pendente ou Não atendida na função QueryCallStatus.

---

### Sintaxe

GetTransitNumber (CallID, >TransitNumber)

### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	A ID da Chamada para a chamada que se deseja enfileirar.
TransitNumber	Seqüência passada por referência	80 caracteres	Um número piloto que a URA usará para transferir a chamada.

Tabela 11

Parâmetros para a função GetTransitNumber

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

- 0** Bem-sucedida
- 308** QueryCallInfo não foi chamado antes de chamar esta função. Você deve chamar QueryCallInfo para obter uma ID da chamada.
- 310** Essa função só pode ser usada para chamadas URA não monitoradas.
- 910** A versão do API URA DLL não corresponde à versão do OpenScape Contact Center.
- 927** Todos os números de trânsito registrados estão ocupados.

## Usando as funções da API URA do OpenScape Contact Center

### Initialize

- 929** Nenhum número de trânsito foi configurado.
- 931** Há uma inconsistência na monitoração do tipo de chamada entre a API URA e o T-Server.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

#### Exemplo

```
GetTransitNumber ('1341023126268002', >TransitNumber)
```

## 4.13 Initialize

A função Initialize inicializa a conexão com os servidores do OpenScape Contact Center e deve ser chamada antes que quaisquer outras funções da API URA do OpenScape Contact Center seja chamada. Essa função apaga os ANI e DNIS que foram configurados durante a chamada anterior nesse ramal. Portanto, chame essa função no início do seu script da URA, e tenha certeza de que o OpenScape Contact Center esteja funcionando antes de rotear as chamadas. Não chame a função Initialize várias vezes para a mesma chamada.

#### Sintaxe

```
Initialize (LocalPort, AdminServerName, RpcTimeout)
```

#### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
LocalPort	Seqüência passada por valor	1 – 32767	O número da porta TCP/IP a ser utilizado pela API URA. Este deverá ser um número que ainda não esteja em uso.
AdminServer Name	Seqüência passada por valor	128 caracteres	O nome da rede do OpenScape Contact Center Administration Server no site no qual você está trabalhando, como por exemplo, 6000@nomedoservidor. Se o sistema estiver configurado para alta disponibilidade (Alta disponibilidade), é necessário especificar o nome do cluster.
RpcTimeout	Inteiro passado por valor	Maior que 0	O período de tempo em milissegundos entre o momento em que a função faz uma chamada e a retorna. Quando um usuário chama a função, o sistema espera um resultado dentro do período de timeout ou a chamada falha. Use um valor de 10000 ou maior.

Tabela 12

Parâmetros para a função Initialize

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

- 0** Bem-sucedida
- 306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.
- 910** A versão do API URA DLL não corresponde à versão do OpenScape Contact Center.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

### Exemplo

O exemplo a seguir inicializa uma conexão entre o servidor do OpenScape Contact Center chamado “deerhurst” e o sistema da URA. O número do ramal da URA é 6060. Todos os parâmetros são passados por valor.

```
Initialize('6060', '6000@deerhurst', 10000)
```

## 4.14 QueryAgentStatus

A função QueryAgentStatus consulta o status de telefonia de um usuário associado com determinado dispositivo. Essa função retornará o status do usuário como um valor inteiro, dependendo de o usuário estar ativo ou não. É necessário especificar a ID do agente ou o dispositivo. Se a ID do agente for especificada e o usuário estiver ativo, o dispositivo será preenchido e o status será retornado. Se o dispositivo for especificado e houver um usuário ativo no dispositivo especificado, a ID do agente será preenchida e o status será retornado.

---

**NOTA:** Será necessário chamar a função Initialize antes de chamar esta função. Para obter mais informações, consulte [Seção 4.13, “Initialize”, na página 46.](#)

---

### Sintaxe

```
QueryAgentStatus (>AgentID, >Device, >AgentStatus)
```

#### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
AgentID	Seqüência passada por referência	8 caracteres	Uma sequência que identifica o usuário; vazio, se o Dispositivo for fornecido.
Device	Seqüência passada por referência	16 caracteres	O ramal do usuário que se deseja consultar; vazio, se a ID do agente for fornecida.
AgentStatus	Inteiro passado por referência	Maior que 0	Indica o status de um usuário. 1 - Discando 2 - Linha ocupada 3 - Tocando 4 - Conversando 5 - Linha enfileirada 6 - Retenção 7 - Consultando 8 - Fora de serviço 9 - Disponível 10 - Indisponível 11 - Pausa 12 - Inativo 13 - Desconhecido 14 - Pendente 15 - Processando 16 - Pós-processamento

Tabela 13

Parâmetros para a função *QueryAgentStatus*

#### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

**0** Bem-sucedida

**-904** A ID de agente reserva especificada não é idêntica a nenhuma ID dos agentes na base de dados.

**-909** O dispositivo especificado não é idêntico a nenhum dispositivo na base de dados.

**Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

#### Exemplo

O exemplo a seguir consulta o status de um usuário associado com determinado dispositivo e retorna o status do usuário associado com o dispositivo especificado.

```
QueryAgentStatus('188001', >Device, >AgentStatus)
```

## 4.15 QueryCallInfo

A função QueryCallInfo consulta o T-Server quanto à ID da chamada e os números ANI e DNIS associados com o ramal da URA atual e a chamada que está sendo processada. Os números ANI e DNIS são retornados como seqüências. Chame essa função quando uma chamada chega ao ramal da URA para obter as informações de ANI e DNIS para a chamada.

---

**NOTA:** Será necessário chamar a funções Inicializar antes de chamar esta função. Para obter mais informações, consulte [Seção 4.13, “Initialize”, na página 46.](#)

---

### Sintaxe

```
QueryCallInfo (Device, >CallID, >originalANI, >originalDNIS,
>UUID, >Visits, >FMNQ, >InternalANI, >InternalDNIS,
TrunkConnectedIVRCall)
```

### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
Device	Seqüência passada por valor	16 caracteres	O ramal da URA no qual o script da URA está sendo executado.
CallID	Seqüência passada por referência	18 caracteres	A ID da Chamada para a chamada atual.
originalANI	Seqüência passada por referência	80 caracteres	O número ANI associado com a chamada quando ela entrar no sistema.
originalDNIS	Seqüência passada por referência	80 caracteres	É o número DNIS associado com a chamada quando ela entra no sistema.
UUID (opcional)	Seqüência passada por referência	32 caracteres	O UUID associado com a chamada atual.
Visits (opcional)	Inteiro passado por referência	Maior ou igual a 0	O número de vezes que esta chamada foi atendida por uma URA monitorada do OpenScape Contact Center.

Tabela 14

Parâmetros para a função QueryCallInfo

## Usando as funções da API URA do OpenScape Contact Center

### QueryCallInfo

Nome	Tipo	Intervalo	Descrição
FMNQ (opcional)	Inteiro passado por referência	0 ou 1	Indica se este é um cenário FMNQ. 0 - Não 1 - Sim
InternalANI (opcional)	Seqüência passada por referência	80 caracteres	O número do qual originou-se a chamada colocada no grupo ACD/UCD/MLHG da URA.
Internal DNIS (opcional)	Seqüência passada por referência	80 caracteres	O número piloto associado com o MLHG grupo ACD/UCD/MLHG da URA.
TrunkConnected IVRCall	Seqüência passada por valor	Ilimitado.	Indica se a chamada é monitorada ou não monitorada. Esse parâmetro deve ser "1;" para uma chamada de uma URA não monitorada. Qualquer outro valor indica uma chamada de uma URA monitorada.

Tabela 14

Parâmetros para a função QueryCallInfo

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

**0** Bem-sucedida

**-304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.

**-306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.

**-701** A solicitação expirou antes que pudesse ser concluída. Isso pode ser um indicativo de problemas na rede ou um servidor ocupado.

**-801** Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.

### Exemplo

O exemplo a seguir consulta a ID da chamada e os números ANI e DNIS da chamada atual conectada ao ramal URA.

```
QueryCallInfo ('36200', >CallID, >originalANI, >originalDNIS)
```

## 4.16 QueryCallStatus

A função QueryCallStatus consulta o status da chamada associada a uma determinada ID da chamada. O destino para o qual a chamada deve ser transferida é retornado pelo parâmetro Extension com base no status da chamada (Pendente ou Não atendida). Se o status da chamada for Desconectada, então a chamada deve ser desconectada pelo sistema da URA (no cenário de Retenção na URA).

Após uma chamada ter sido enfileirada, chame regularmente a função QueryCallStatus para determinar se a chamada foi designada.

---

**NOTA:** Se o sistema da URA não for monitorado, será possível chamar a função GetTransitNumber para consultar o OpenScape Contact Center para um número de trânsito para o qual a chamada pode ser transferida. Para obter mais informações, consulte [Seção 4.12, “GetTransitNumber”, na página 45](#).

---



---

**NOTA:** Será necessário chamar as funções Inicializar e QueryCallInfo antes de chamar esta função. Para obter mais informações, consulte a [Seção 4.13, “Initialize”, na página 46](#) e a [Seção 4.15, “QueryCallInfo”, na página 49](#).

---

### Sintaxe

QueryCallStatus (CallID, Timeout, >Extension, >PositionInQueue)

### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	A CallID retornada de QueryCallInfo.
Timeout	Seqüência passada por valor	Maior ou igual a 0	O tempo transcorrido (em milissegundos) ao esperar por uma alteração no status da chamada em determinado dispositivo. Ao se especificar zero (0) indica-se uma consulta imediata do status atual.
Extension	Seqüência passada por referência	80 caracteres	O destino para o qual a chamada deve ser transferida.

Tabela 15

Parâmetros para a função QueryCallStatus

## Usando as funções da API URA do OpenScape Contact Center

### QueryCallStatus

Nome	Tipo	Intervalo	Descrição
Position InQueue	Inteiro passado por referência	Maior que 0	A posição da chamada na fila, conforme retornada pelo Routing Server.

Tabela 15

Parâmetros para a função QueryCallStatus

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

- 0** A chamada está em espera.
- 1** A chamada está no estado Enfileirado. Nesse caso, continue a verificar o status da chamada.
- 2** A chamada está no estado Pendente. Neste caso, transfira a chamada para o ramal retornado.
- 3** A chamada está no estado Não atendida. Neste caso, transfira a chamada para o ramal retornado ou para outro ramal em time-out.
- 4** Ocorreu um erro. Neste caso, transfira a chamada para um ramal que não pertença ao OpenScape Contact Center.
- 5** A chamada deve ser desconectada.
- 6** A chamada deve ser transferida para o ramal retornado.
- 304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.
- 306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.
- 308** QueryCallInfo não foi chamado antes de chamar esta função. Você deve chamar QueryCallInfo para obter uma ID da chamada.
- 701** A solicitação expirou antes que pudesse ser concluída. Isso pode ser um indicativo de problemas na rede ou um servidor ocupado.
- 801** Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.
- 903** Não havia nenhuma chamada associada com a ID da chamada. Certificar-se de que todos os ramais URA estão indicados na base de dados.
- 908** O T-Server não está disponível.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

---

**NOTA:** Você pode receber o código de retorno -701 enquanto estiver chamando a função QueryCallStatus por dois motivos, a seguir:

- Se você especificar um time-out diferente de zero, o QueryCallStatus aguarda que os eventos cheguem. Neste caso, o código de retorno -701 só indicará que você aguardou o tempo especificado sem que nada acontecesse.
  - Se você especificar um time-out de 0 e a chamada estiver em um estado Enfileirado, então a posição na fila será enfileirada a partir do Routing Server. Neste caso, o código de retorno -701 indica que há problemas de rede ou um servidor ocupado.
- 

### Exemplo

O exemplo a seguir aguarda no máximo 60.000 milissegundos para monitorar a alteração no status. Se a chamada mudar para o estado Pendente dentro daquele período de tempo, a função retorna o ramal do usuário para o qual o sistema da URA deverá transferir a chamada, usando a variável TRANSFER\_EXT.

```
QueryCallStatus('1341023126268002', 60000, >TRANSFER_EXT)
```

## 4.17 QueryQueueStatistics

A função QueryQueueStatistics determina o número de chamadas na fila, tempo de espera estimado para uma chamada, o tempo médio de espera para a fila, o tempo que a chamada mais antiga ficou na fila e o nível de serviço para a fila especificada.

Esses cálculos presumem que a chamada será enfileirada imediatamente para o Routing Server, independentemente de ter sido enfileirada. você deve chamar QueryCallStatistics antes de chamar Enqueue para assegurar uma resposta precisa do Routing Server.

---

**NOTA:** Será necessário chamar a função Initialize antes de chamar esta função. Para obter mais informações, consulte Seção 4.13, “Initialize”, na página 46.

---

### Sintaxe

```
QueryQueueStatistics (QueueName, >CallsInQueue, >EstimatedWait,  
>AverageWait, >OldestCallInQueue, >ServiceLevel,  
BusinessUnitName)
```

## Usando as funções da API URA do OpenScape Contact Center

### QueryQueueStatistics

#### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
QueueName	Seqüência passada por valor	32 caracteres	A fila para a qual a consulta é realizada.
CallsInQueue	Inteiro passado por referência	Maior ou igual a 0	O número de chamadas na fila para a fila especificada.
EstimatedWait	Inteiro passado por referência	Maior ou igual a 0	O tempo de espera estimado em segundos para a fila especificada.
AverageWait	Inteiro passado por referência	Maior ou igual a 0	O tempo médio de espera em segundos para a fila especificada.
OldestCall InQueue	Inteiro passado por referência	Maior ou igual a 0	O período de tempo em segundos em que a chamada mais antiga encontra-se na fila.
ServiceLevel	Inteiro passado por referência	0 – 100	O nível de serviço para a fila especificada.
BusinessUnit Name (opcional)	Seqüência passada por valor	32 caracteres	O nome da unidade de negócios para a fila especificada. Este parâmetro será obrigatório em um ambiente de multilocação e ignorado em um ambiente que não seja de multilocação.

Tabela 16

Parâmetros para a função *QueryQueueStatistics*

#### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

**0** Bem-sucedida

**-304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.

**-306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.

**-314** O nome da unidade de negócios especificado não estava definido ou não correspondeu ao nome de nenhuma unidade de negócios na base de dados. (Este erro só se aplica quando o OpenScape Contact Center estiver configurado como um sistema multilocação.)

- 701** A solicitação expirou antes que pudesse ser concluída. Isso pode ser um indicativo de problemas na rede ou um servidor ocupado.
- 801** Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.
- 905** A fila especificada não é idêntica a nenhum nome de fila na base de dados.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

#### Exemplo

O exemplo a seguir consulta as estatísticas para a fila “Vendas”.

```
QueryQueueStatistics('Sales', >CallsInQueue, >EstimatedWait,
>AverageWait, >OldestCallInQueue, >ServiceLevel, 'Toronto')
```

## 4.18 QueryRoutingInfo

A função QueryRoutingInfo envia uma solicitação para determinar as informações de roteamento para uma chamada. A consulta baseia-se em informações coletadas e enviadas pelo sistema da URA como parte da solicitação de roteamento do OpenScape Contact Center.

Você deve enfileirar, desconectar ou transferir a chamada com base no resultado da consulta.

Se um componente de Transferir for encontrado em um fluxo de trabalho, então o código retornado será 2. O destino da transferência retornado poderá estar no formato canônico, dependendo de como o componente de Transferir tiver sido configurado. Nesse caso, se você desejar transferir a chamada para esse destino usando somente uma transferência de intermitência, ou outro mecanismo interno de transferência da URA, será necessário traduzir o número apropriadamente. Se você desejar transferir a chamada usando a URA API, então a análise será tratada diretamente pelo OpenScape Contact Center.

---

**NOTA:** Será necessário chamar as funções Inicializar e QueryCallInfo antes de chamar esta função. Para obter mais informações, consulte a [Seção 4.13, “Initialize”, na página 46](#) e a [Seção 4.15, “QueryCallInfo”, na página 49](#).

---

---

**NOTA:** Se você quiser que os valores dos dados de contatos sejam considerados na decisão de roteamento, deverá definir esses valores usando a função SetContactData antes de chamar essa função. Para reter os valores originais dos dados do contato, é necessário copiar e redefinir os valores antes de chamar as funções Enqueue ou EnqueueForAgent. Para obter mais informações, consulte [Seção 4.24, “SetContactData”, na página 65](#).

---

## Usando as funções da API URA do OpenScape Contact Center

### QueryRoutingInfo

#### Sintaxe

```
QueryRoutingInfo (CallID, >Destination, >AgentID,  
>AgentWaitTime, >Description, >InitialPriority, >EstimatedWait,  
>CallsinQueue, ANIDNIS)
```

#### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	A ID da chamada da qual você deseja obter as respectivas informações de roteamento.
Destination	Seqüência passada por referência	80 caracteres	Código de retorno 0 – A fila na qual a chamada deve ser enfileirada. Código de retorno 1 – A chamada deve ser desconectada. Código de retorno 2 – O destino para o qual a chamada deve ser transferida.
AgentID	Seqüência passada por referência	8 caracteres	Retorna a ID do usuário que deverá ser reservado para atender à chamada.  <b>Observação:</b> Válido somente se o código de retorno for 0.
AgentWaitTime	Inteiro passado por referência	Maior que 0	Retorna o tempo máximo em segundos que o chamador deverá aguardar por um usuário reservado antes que a chamada seja liberada e vá para a fila especificada. Isso só é válido se uma ID de agente for retornada.  <b>Observação:</b> Válido somente se o código de retorno for 0.
Descrição	Seqüência passada por referência	100 caracteres	Fornece uma descrição da chamada, que é exibida no aplicativo OpenScape Contact Center Client Desktop, para que os usuários possam identificar a chamada.  <b>Observação:</b> Válido somente se o código de retorno for 0.
InitialPriority	Inteiro passado por referência	1 a 100	Retorna a prioridade da chamada relativa a todas as outras chamadas enfileiradas no Routing Server, incluindo as chamadas associadas à mesma fila da chamada atual. O padrão é 1.  <b>Observação:</b> Válido somente se o código de retorno for 0.

Tabela 17

Parâmetros para a função QueryRoutingInfo

Nome	Tipo	Intervalo	Descrição
EstimatedWait	Inteiro passado por referência	Maior ou igual a 0	Retorna o tempo de espera estimado em segundos para a fila especificada.  <b>Observação:</b> Válido somente se o código de retorno for 0.
CallsInQueue	Inteiro passado por referência	Maior ou igual a 0	Retorna o número de chamadas na fila para a fila especificada.  <b>Observação:</b> Válido somente se o código de retorno for 0.
ANIDNIS (opcional)	Seqüência passada por valor	161 caracteres	Os números ANI e DNIS a serem usados na decisão de roteamento. O ANI e o DNIS devem ser separados por um ponto e vírgula. Se os dados do contato já tiverem sido associados com essa chamada e contiverem valores de ANI e DNIS, esses valores terão prioridade sobre os valores que você especificar aqui.

Tabela 17

Parâmetros para a função QueryRoutingInfo

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

- 0** Bem-sucedida Indica que a chamada deve ser enfileirada com as informações retornadas.
- 1** Bem-sucedida Indica que a chamada deve ser desconectada.
- 2** Bem-sucedida Indica que a chamada deve ser transferida para o destino retornado.
- 304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.
- 306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.
- 400** O sistema tem poucos recursos.
- 801** Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.
- 915** Não pôde recuperar a fila e outras informações de roteamento com base na opção de roteamento, configurada no sistema.
- 916** O Routing Server encontrou um erro na execução do fluxo de trabalho.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

### Exemplo

O exemplo a seguir consulta o Routing Server para obter informações de roteamento.

```
QueryRoutingInfo('1341023126268002', >Destination, >AgentID,  
>AgentWaitTime, >Description, >InitialPriority, >EstimatedWait,  
>CallsInQueue, '9055557900;69030')
```

## 4.19 QuerySystemStatus

A função QuerySystemStatus determina o status do sistema. Esta função deve ser chamada para verificar o status do sistema no início do script da URA e antes de enfileirar-se uma chamada.

---

**NOTA:** Será necessário chamar a função Initialize antes de chamar esta função. Para obter mais informações, consulte [Seção 4.13, “Initialize”, na página 46.](#)

---

### Sintaxe

```
QuerySystemStatus (>OverallStatus, >IndividualServerStates,  
>Entries)
```

### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
OverallStatus	Inteiro passado por referência	0 ou 1	0 - Indica que o sistema não está operacional 1 - Indica que o sistema está operacional.
IndividualServer States	Seqüência passada por referência	250 caracteres	A seqüência que contém cada nome de servidor e seu estado. Por exemplo: server-name1=1; server-name2=2
Entries	Inteiro passado por referência	Maior ou igual a 0	O número de servidores na sequência de status.

*Tabela 18 Parâmetros para a função QuerySystemStatus*

O script da URA deve analisar os estados dos servidores para encontrar o status do Routing Server e do T-Server. Se o estado do Routing Server não for 1 ou o estado do T-Server não for 8, a chamada deverá ser transferida para um número ACD/UCD/MLHG que roteia as chamadas para um grupo ACD/UCD/MLHG de emergência.

### Valores dos status

Os códigos a seguir indicam o status dos vários servidores:

- 0 O servidor não está em funcionamento.

- 1** O servidor está em funcionamento.
- 2** O servidor está inativo. Isto significa que o Administration Server está em funcionamento, mas não recebeu uma indicação do status para o servidor especificado.

Somente o T-Server usa os seguintes códigos:

- 3** O T-Server não está em funcionamento.
- 4** O T-Server está em processo de desligamento.
- 5** O T-Server está fora de serviço porque não pode conectar-se ao provedor de CSTA.
- 6** O T-Server está em processo de iniciar.
- 7** O T-Server está em processo de inicialização.
- 8** O T-Server está em funcionamento.
- 9** A plataforma de comunicação está sobrecarregada e tendo problemas para realizar as solicitações do T-Server.
- 10** O T-Server está em processo de recuperação.

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

- 0** Bem-sucedida
- 304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.
- 306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.
- 701** A solicitação expirou antes que pudesse ser concluída. Isso pode ser um indicativo de problemas na rede ou um servidor ocupado.
- 801** Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.
- 901** Não há nenhuma informação de estado disponível. O Administration Server está fora de operação, mas o Routing Server ainda está disponível. Neste caso, o estado dos outros servidores é desconhecido. Este código será retornado somente pela função QueryCallStatus.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

### Exemplo

O exemplo a seguir consulta o status de cada um dos servidores. A função passa o S\_OVERALLSTATUS e S\_SERVER\_STATES por referência para retornar o status de cada um dos servidores. Da mesma forma, I\_ENTRIES retorna o número de servidores na seqüência S\_SERVER\_STATES.

```
QuerySystemStatus (>S_OVERALLSTATUS, >S_SERVER_STATES,  
>I_ENTRIES)
```

## 4.20 ReleaseTransitNumber

---

**NOTA:** Esta função somente é usada por scripts de URA para sistemas da URA não monitorados.

---

A função ReleaseTransitNumber permite que os aplicativos solicitem que o OpenScape Contact Center libere um número de trânsito para ser usado em outra chamada. Esta função deve ser chamada antes que o número de trânsito expire. Isso significa que o número de trânsito se tornará disponível para outra chamada do OpenScape Contact Center.

O sistema da URA pode usar a função ReleaseTransitNumber para informar ao T-Server que o número de trânsito não será mais necessário. O aplicativo deve chamar a função GetTransitNumber novamente para solicitar um novo número de trânsito antes de tentar transferir a chamada para o OpenScape Contact Center.

### Sintaxe

```
ReleaseTransitNumber (CallID)
```

### Parâmetros

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	A ID da Chamada para a chamada atual.

Tabela 19

Parâmetros para a função ReleaseTransitNumber

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

- 0** Bem-sucedida
- 308** QueryCallInfo não foi chamado antes de chamar esta função. Você deve chamar QueryCallInfo para obter uma ID da chamada.
- 310** Essa função só pode ser usada para chamadas URA não monitoradas.
- 910** A versão do API URA DLL não corresponde à versão do OpenScape Contact Center.
- 928** Nenhum número de trânsito foi alocado para esta ID de chamada.
- 931** Há uma inconsistência na monitoração do tipo de chamada entre a API URA e o T-Server.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

### Exemplo

```
ReleaseTransitNumber ('1341023126268002')
```

## 4.21 SetBusinessUnit

---

**NOTA:** A função SetBusinessUnit só é compatível em um ambiente de multilocação.

---

A função SetBusinessUnit define a unidade de negócios para a chamada especificada.

---

**NOTA:** A função SetBusinessUnit só pode ser chamada uma vez para cada chamada da URA.

---



---

**NOTA:** Será necessário chamar as funções Inicializar e QueryCallInfo antes de chamar esta função. Para obter mais informações, consulte [Seção 4.13, “Initialize”, na página 46](#) e [Seção 4.15, “QueryCallInfo”, na página 49](#).

---

### Sintaxe

SetBusinessUnit (CallID, BusinessUnitName)

### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	O CallID para a chamada a qual se deseja definir a unidade de negócios.
BusinessUnit Name	Seqüência passada por valor	32 caracteres	O nome da unidade de negócios para a chamada especificada.

Tabela 20

Parâmetros para a função SetBusinessUnit

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

- 0** Bem-sucedida
- 304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.
- 306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.
- 308** QueryCallInfo não foi chamado antes de chamar esta função. Você deve chamar QueryCallInfo para obter uma ID da chamada.

## Usando as funções da API URA do OpenScape Contact Center

### SetCallInfo

- 313** A função SetBusinessUnit já foi chamada para esta chamada. (Este erro só se aplica quando o OpenScape Contact Center estiver configurado como um sistema multilocação.)
- 314** O nome da unidade de negócios especificado não estava definido ou não correspondeu ao nome de nenhuma unidade de negócios na base de dados. (Este erro só se aplica quando o OpenScape Contact Center estiver configurado como um sistema multilocação.)
- 400** O sistema tem poucos recursos.
- 801** Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.
- 910** A versão do API URA DLL não corresponde à versão do OpenScape Contact Center.
- 955** A função Multilocação não está licenciada.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

#### Exemplo

O exemplo a seguir define o nome da unidade de negócios para a chamada especificada.

```
SetBusinessUnit ('8271023217459002', 'WirelessDept')
```

## 4.22 SetCallInfo

---

**NOTA:** Esta função somente é usada por scripts de URA para sistemas da URA não monitorados.

---

A função SetCallInfo armazena as informações ANI e DNIS de chamadas de URA não monitorada para serem transferidas ao OpenScape Contact Center. Esta API só pode ser chamada antes de QueryCallInfo e depois de Inicializar

#### Sintaxe

```
SetCallInfo(ANI, DNIS)
```

#### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
ANI	Seqüência passada por valor	80 caracteres	O ANI da chamada.

Tabela 21

Parâmetros para a função SetCallInfo

Nome	Tipo	Intervalo	Descrição
DNIS	Seqüência passada por valor	80 caracteres	O DNIS da chamada.

Tabela 21

Parâmetros para a função SetCallInfo

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

**0** Bem-sucedida

**-304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.

**-306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.

**-312** Sequência errada de operações (por exemplo, SetCallInfo é chamado após QueryCallInfo).

**Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

### Exemplo

```
SetCallInfo('9055551234','12345')
```

## 4.23 SetCallTransferable

---

**NOTA:** Antes que o script da URA desconecte ou transfira uma chamada FMNQ, você deverá chamar a função SetCallTransferable (0) e receber um código de retorno bem sucedido. Se você não receber um código de retorno bem sucedido, isso indicará que a chamada está em processo de ser transferida e será necessário aguardar até que o processo esteja completo e, depois, chamar a função SetCallTransferable (0) novamente.

---

Uma flag que indica se uma chamada pode ser transferida pelo OpenScape Contact Center para um usuário designado.

---

**NOTA:** Essa função não é compatível com chamadas da URA não monitorada.

---

---

**NOTA:** Será necessário chamar as funções Inicializar e QueryCallInfo antes de chamar esta função. Para obter mais informações, consulte a [Seção 4.13, “Initialize”, na página 46](#) e a [Seção 4.15, “QueryCallInfo”, na página 49](#).

---

### Sintaxe

SetCallTransferable (CallID, FlagValue)

### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	As ID da chamada que se deseja transferir ou desconectar.
FlagValue	Inteiro passado por valor	0 ou 1	Indica se uma chamada pode ser transferida para um usuário designado. Se estiver configurado como false (0), o sistema suspenderá a correspondência de um usuário com a chamada até que a flag seja configurada como true (1). Se a chamada for transferida de volta para a fila pelo sistema da URA, a flag será redefinida automaticamente como verdadeiro (1).

Tabela 22

Parâmetros para a função SetCallTransferable

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

- 0** Bem-sucedida
- 304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.
- 306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.
- 311** A função não é compatível com chamadas da URA não monitorada.
- 701** A solicitação expirou antes que pudesse ser concluída. Isso pode ser um indicativo de problemas na rede ou um servidor ocupado.
- 801** Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

### Exemplo

O exemplo a seguir especifica que a chamada poderá ser transferida pelo OpenScape Contact Center para um usuário designado.

```
SetCallTransferable('0741023896297002', 1)
```

## 4.24 SetContactData

A função SetContactData define os dados do contato para uma ID de chamada especificada e adiciona o par chave-valor se ele não existir. Se você quiser que os dados do contato fiquem disponíveis para a chamada, será necessário configurar os dados do contato antes de enfileirar a chamada.

---

**NOTA:** Será necessário chamar as funções Inicializar e QueryCallInfo antes de chamar esta função. Para obter mais informações, consulte a [Seção 4.13, “Initialize”](#), na [página 46](#) e a [Seção 4.15, “QueryCallInfo”](#), na [página 49](#).

---

### Sintaxe

```
SetContactData (CallID, Key, Value)
```

## Usando as funções da API URA do OpenScape Contact Center

### SetContactData

#### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	A ID da chamada (CallID) para a chamada a qual se deseja definir os dados dos contatos.
Key	Seqüência passada por valor	32 caracteres	O nome da chave para o valor dos dados do contato.
Value	Seqüência passada por valor	128 caracteres	O valor a ser definido.

Tabela 23 Parâmetros para a função SetContactData

#### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

- 0** Bem-sucedida
- 304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.
- 306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.
- 308** QueryCallInfo não foi chamado antes de chamar esta função. Você deve chamar QueryCallInfo para obter uma ID da chamada.
- 701** A solicitação expirou antes que pudesse ser concluída. Isso pode ser um indicativo de problemas na rede ou um servidor ocupado.
- 801** Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

#### Exemplo

O exemplo a seguir define os dados de contatos para uma ID de chamada especificada e adiciona o par chave-valor se ele não existir.

```
SetContactData('8271023217459002', 'Name', 'John Doe', 0)
```

## 4.25 SetDisplay

A função SetDisplay define a exibição do telefone para o primeiro usuário que responder à chamada especificada.

---

**NOTA:** Esta função só está disponível em um plataforma de comunicação OpenScape 4000 ou HiPath 4000. Será necessário chamar as funções Inicializar e QueryCallInfo antes de chamar esta função. Para obter mais informações, consulte a [Seção 4.13, “Initialize”, na página 46](#) e a [Seção 4.15, “QueryCallInfo”, na página 49](#).

---

### Sintaxe

SetDisplay (CallID, DisplayInfo)

### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	A ID da chamada, da qual você deseja definir a exibição.
DisplayInfo	Seqüência passada por valor	240 caracteres	O seqüência que é exibida no dispositivo de resposta do usuário.

Tabela 24

Parâmetros para a função SetDisplay

### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

- 0** Bem-sucedida
- 304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.
- 306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.
- 701** A solicitação expirou antes que pudesse ser concluída. Isso pode ser um indicativo de problemas na rede ou um servidor ocupado.
- 801** Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

#### Exemplo

O exemplo a seguir define a exibição do telefone para o primeiro usuário que responder à chamada especificada como “Chamada URA”.

```
SetDisplay('1341023126268002', 'IVR Call')
```

## 4.26 Transfer

A função Transfer executa uma transferência e indica o sucesso ou falha desta. Esta API não aguardará que a parte de destino responda.

---

**NOTA:** Essa função não é compatível com chamadas da URA não monitorada.

---

**NOTA:** Você pode querer utilizar este método ao invés da transferência em intermitência lenta da URA, porque esta função trabalha mais rapidamente. Será necessário chamar as funções Inicializar e QueryCallInfo antes de chamar esta função. Para obter mais informações, consulte a [Seção 4.13, “Initialize”, na página 46](#) e a [Seção 4.15, “QueryCallInfo”, na página 49](#).

#### Sintaxe

```
Transfer (CallID, ToDevice)
```

#### Parâmetros

É necessário inserir os parâmetros na ordem em que eles aparecem na tabela a seguir.

Nome	Tipo	Intervalo	Descrição
CallID	Seqüência passada por valor	18 caracteres	A ID da Chamada para a chamada que se deseja transferir.
ToDevice	Seqüência passada por valor	80 caracteres	A seqüência que identifica o dispositivo para o qual a chamada está sendo transferida.

Tabela 25

Parâmetros para a função Transferência

#### Códigos de retorno

Os códigos mais comuns de retorno por esta função são os seguintes:

**0** Bem-sucedida

**-304** A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.

- 306** Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.
- 311** A função não é compatível com chamadas da URA não monitorada.
- 701** A solicitação da função expirou antes que a função pudesse ter sido completada. Normalmente, isto só ocorre ao consultar o estado de uma chamada. Também pode ser uma indicação de problemas de rede ou que o servidor está ocupado.
- 801** Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.
- 903** Não havia nenhuma chamada associada com a ID da chamada. Certificar-se de que todos os ramais URA estão indicados na base de dados.
- 909** O dispositivo especificado não é idêntico a nenhum dispositivo na base de dados.
- 913** A função passou um “Para Dispositivo” inválido.
- 914** Falha na operação que você está tentando executar.
- 917** Erro na tradução de TAPI.
- Outro** Qualquer outro código indica falha. Para obter mais informações sobre um código específico, consulte [Capítulo 5, “Códigos de retorno”](#).

### Exemplo

O exemplo a seguir transfere a chamada com a ID da chamada especificada para o ramal 5678.

```
Transfer('1341023126268002','5678');
```

### Formato do número de telefone

Os parâmetros do número de telefone devem ser inseridos no formato canônico, da seguinte forma:

+[CountryCode] Espaço [(AreaCode) Espaço SubscriberNumber [--Ramal]

O número de telefone também pode ser um endereço discável obtido de um plataforma de comunicação ou o resultado da chamada de uma função TAPI. As sequências a seguir são números de telefone corretos:

- +1 (555) 555-0199
- (555) 555-0199
- +1 555-0199
- 5555550199

## Usando as funções da API URA do OpenScape Contact Center

### Transfer

## 5 Códigos de retorno

Este capítulo descreve todos os códigos retornados pelas funções da URA. Para obter mais informações sobre como esses códigos se relacionam a funções específicas, consulte [Capítulo 4, “Usando as funções da API URA do OpenScape Contact Center”](#).

Código	Descrição
<b>0</b>	Bem-sucedida  QueryRoutingInfo — Indica que a chamada deve ser enfileirada com as informações retornadas.
<b>1</b>	QueryCallStatus — Indica que a chamada está no estado Enfileirado. Nesse caso, continue a verificar o status da chamada.  QueryRoutingInfo — Indica que a chamada deve ser desconectada.
<b>2</b>	QueryCallStatus — Indica que a chamada está no estado Pendente. Neste caso, transfira a chamada para o ramal retornado.  QueryRoutingInfo — Indica que a chamada deve ser transferida para o destino retornado.
<b>3</b>	QueryCallStatus — Indica que a chamada está no estado Não atendida. Neste caso, transfira a chamada para o ramal retornado ou para outro ramal em time-out.
<b>4</b>	QueryCallStatus — Indica que ocorreu um erro. Neste caso, transfira a chamada para um ramal que não pertença ao OpenScape Contact Center.
<b>5</b>	QueryCallStatus — Indica que a chamada deve ser desconectada.
<b>6</b>	QueryCallStatus — A chamada deve ser transferida para o ramal retornado.
<b>-304</b>	A inicialização não foi chamada antes de chamar esta função. Você deve chamar a inicialização no início do seu script.
<b>-305</b>	Esse código indica um erro desconhecido (por exemplo, erro do usuário ou o sistema está instável).
<b>-306</b>	Um ou mais destes parâmetros passados por valor eram do tipo errado ou continham um valor incorreto. Por exemplo, foi passada uma seqüência em vez de um número inteiro.
<b>-307</b>	O nome da chave para o valor dos dados do contato não foi definido antes de chamar esta função.
<b>-308</b>	QueryCallInfo não foi chamado antes de chamar esta função. Você deve chamar QueryCallInfo para obter uma ID da chamada.
<b>-310</b>	Essa função só pode ser usada para chamadas URA não monitoradas.
<b>-311</b>	A função não é compatível com chamadas da URA não monitorada.
<b>-312</b>	Sequência errada de operações (por exemplo, SetCallInfo é chamado após QueryCallInfo).

Tabela 26

Códigos de retorno

## Códigos de retorno

Código	Descrição
<b>-313</b>	A função SetBusinessUnit já foi chamada para esta chamada. (Este erro só se aplica quando o OpenScape Contact Center estiver configurado como um sistema multilocaçāo.)
<b>-314</b>	O nome da unidade de negócios especificado não estava definido ou não correspondeu ao nome de nenhuma unidade de negócios na base de dados. (Este erro só se aplica quando o OpenScape Contact Center estiver configurado como um sistema multilocaçāo.)
<b>-400</b>	O sistema tem poucos recursos.
<b>-701</b>	A solicitação expirou antes que pudesse ser concluída. Isso pode ser um indicativo de problemas na rede ou um servidor ocupado.
<b>-801</b>	Não foi capaz de conectar ao servidor do OpenScape Contact Center; ou o nome do servidor é inválido ou o servidor está fora de operação.
<b>-901</b>	Não há nenhuma informação de estado disponível. O Administration Server está fora de operação, mas o Routing Server ainda está disponível. Neste caso, o estado dos outros servidores é desconhecido. Este código será retornado somente pela função QueryCallStatus.
<b>-903</b>	Não havia nenhuma chamada associada com a ID da chamada. Certificar-se de que todos os ramais URA estão indicados na base de dados.
<b>-904</b>	A ID de agente reserva especificada não é idêntica a nenhuma ID dos agentes na base de dados.
<b>-905</b>	A fila especificada não é idêntica a nenhum nome de fila na base de dados.
<b>-907</b>	O T-Server não está disponível. Este erro indica que o Routing Server não pôde colocar a chamada em fila, porque o T-Server não estava disponível.
<b>-908</b>	O T-Server não está disponível.
<b>-909</b>	O dispositivo especificado não é idêntico a nenhum dispositivo na base de dados.
<b>-910</b>	A versão do API URA DLL não corresponde à versão do OpenScape Contact Center.
<b>-911</b>	O função continha dados que ficaram corrompidos por motivo desconhecido. Dessa forma, a função não pôde conectar com o Routing Server. A função não foi bem sucedida.
<b>-913</b>	A função passou um “Para Dispositivo” inválido.
<b>-914</b>	Falha na operação que você está tentando executar.
<b>-915</b>	Não pôde recuperar a fila e outras informações de roteamento com base na opção de roteamento, configurada no sistema.
<b>-916</b>	O Routing Server encontrou um erro na execução do fluxo.
<b>-917</b>	Erro na tradução de TAPI.
<b>-927</b>	Todos os números de trânsito registrados estão ocupados.
<b>-928</b>	Nenhum número de trânsito foi alocado para esta ID de chamada.
<b>-929</b>	Nenhum número de trânsito foi configurado.
<b>-931</b>	Há uma inconsistência na monitoração do tipo de chamada entre a API URA e o T-Server.

Tabela 26

Códigos de retorno

Código	Descrição
<b>-955</b>	A função Multilocação não está licenciada.
<b>-1006</b>	Você está tentando criar um callback duplicado
<b>-1021</b>	ID do callback (CallbackID) inválido.
<b>-1028</b>	Um horário de programação é inválido.
<b>-1029</b>	As programações de callbacks não possuem um horário mútuo com o horário de funcionamento do central de contatos.
<b>-1031</b>	Um número de telefone na solicitação está na lista Números Excluídos.
<b>-1033</b>	Um callback não pode aceitar dados do contato com mais de 1.000 bytes.
<b>-1040</b>	Todas as programações já expiraram.
<b>-1045</b>	O nome de fila de callbacks é inválido.
<b>-1047</b>	A programação contém uma data muito distante no futuro.
<b>Outro</b>	Qualquer outro código negativo indica falha. Se outro código for retornado: <ul style="list-style-type: none"> <li>• Certifique-se de que todos os nomes de sites, filas e usuários (ou números de IDs) no script da URA correspondam aos nomes (ou números de IDs) na cópia de produção da base de dados do OpenScape Contact Center.</li> <li>• Certifique-se de que todos os recursos da plataforma de comunicação sejam devidamente inseridos na cópia de produção da base de dados do OpenScape Contact Center.</li> <li>• Entre em contato com o próximo nível de suporte técnico.</li> </ul>

Tabela 26

Códigos de retorno

## **Códigos de retorno**

# Índice remissivo

## A

ambiente de multilocação 26  
 função GetBusinessUnit 41  
 função SetBusinessUnit 61  
 subdiálogo GetBusinessUnit 41  
 subdiálogo SetBusinessUnit 61

## C

códigos de erro 71  
 códigos de retorno 71

## D

documentação  
 convenções de formatação 5  
 dando feedback 6  
 público ao qual se destina 5

## F

FMNQ  
 gravando scripts da URA 24  
 função CallerDisconnected 28  
 função CreateCallback 29  
 função DeleteCallback 32  
 função Desenfileirar 33  
 função Disconnect 34  
 função Enqueue 35  
 função EnqueueForAgent 37  
 função GetBusinessUnit 41  
 função GetCallTransferable 42  
 função GetContactData 43  
 função GetTransitNumber 45  
 função Initialize 46  
 função QueryAgentStatus 47  
 função QueryCallInfo 49  
 função QueryCallStatus 51  
 função QueryQueueStatistics 53  
 função QueryRoutingInfo 55  
 função QuerySystemStatus 58  
 função ReleaseTransitNumber 60  
 função SetBusinessUnit 61  
 função SetCallInfo 62  
 função SetCallTransferable 64  
 função SetContactData 65  
 função SetDisplay 67  
 função Transfer 68  
 funções API URA  
 DeleteCallback 32

funções da API URA

CallerDisconnected 28  
 CreateCallback 29  
 Desenfileirar 33  
 Enqueue 35  
 EnqueueForAgent 37  
 GetBusinessUnit 41  
 GetCallTransferable 42  
 GetContactData 43  
 GetTransitNumber 45  
 Initialize 46  
 QueryAgentStatus 47  
 QueryCallInfo 49  
 QueryCallStatus 51  
 QueryQueueStatistics 53  
 QueryRoutingInfo 55  
 QuerySystemStatus 58  
 ReleaseTransitNumber 60  
 SetBusinessUnit 61  
 SetCallInfo 62  
 SetCallTransferable 64  
 SetContactData 65  
 SetDisplay 67  
 Transfer 68  
 funções de API URA  
 Disconnect 34

## I

instalação 9

## P

para passar parâmetros  
 por referência 27  
 por valor 27

## R

retenção na a URA  
 configuração 15  
 retenção na fila  
 configuração 17  
 gravando scripts da URA 22  
 retenção na URA  
 gravando scripts da URA 20

## S

scripts  
 retenção na URA 20  
 scripts da URA

## Índice remissivo

FMNQ 24  
gravando 19  
retenção na fila 22  
visão geral 7  
scripts de URA  
    para passar parâmetros 27  
sistema da URA  
    interação com o OpenScape Contact Center 8  
sistemas da URA  
    não monitorados 10  
sistemas da URA não monitorados 10  
subdiálogo CreateCallback 29  
subdiálogo DeleteCallback 32  
subdiálogo Dequeue 33  
subdiálogo Enqueue 35  
subdiálogo GetBusinessUnit 41  
subdiálogo GetContactData 43  
subdiálogo GetTransitNumber 45  
subdiálogo Initialize 46  
subdiálogo QueryAgentStatus 47  
subdiálogo QueryCallStatus 51  
subdiálogo QueryQueueStatistics 53  
subdiálogo QueryRoutingInfo 55  
subdiálogo QuerySystemStatus 58  
subdiálogo ReleaseTransitNumber 60  
subdiálogo SetBusinessUnit 61  
subdiálogo SetContactData 65  
subdiálogo SetDisplay 67  
subdiálogos do VoiceXML  
    CreateCallback 29  
    DeleteCallback 32  
    Dequeue 33  
    Enqueue 35  
    GetBusinessUnit 41  
    GetContactData 43  
    GetTransitNumber 45  
    Initialize 46  
    QueryAgentStatus 47  
    QueryCallStatus 51  
    QueryQueueStatistics 53  
    QueryRoutingInfo 55  
    QuerySystemStatus 58  
    ReleaseTransitNumber 60  
    SetBusinessUnit 61  
    SetContactData 65  
    SetDisplay 67

## V

visão geral 7

