

# Integration

## OpenScape Contact Center V8 IVR API Integrationshandbuch

Integrationshandbuch

A31003-S2280-N102-1-20

Communication for the open minded

Siemens Enterprise Communications  
[www.siemens.de/open](http://www.siemens.de/open)

**SIEMENS**

## Communication for the open minded

Siemens Enterprise Communications  
[www.siemens.de/open](http://www.siemens.de/open)

Copyright © Siemens Enterprise  
Communications GmbH & Co. KG 2010  
Hofmannstr. 51, D-80200 München

Siemens Enterprise Communications GmbH & Co. KG  
is a Trademark Licensee of Siemens AG

Bestellnummer: A31003-S2280-N102-1-20

Die Informationen in diesem Dokument enthalten lediglich allgemeine Beschreibungen bzw. Leistungsmerkmale, welche im konkreten Anwendungsfall nicht immer in der beschriebenen Form zutreffen bzw. welche sich durch Weiterentwicklung der Produkte ändern können. Die gewünschten Leistungsmerkmale sind nur dann verbindlich, wenn sie bei Vertragsabschluss ausdrücklich vereinbart werden. Liefermöglichkeiten und technische Änderungen vorbehalten.

OpenScape, OpenStage und HiPath sind eingetragene Warenzeichen der Siemens Enterprise Communications GmbH & Co. KG.

Alle anderen Marken-, Produkt- und Servicennamen sind Warenzeichen oder eingetragene Warenzeichen ihrer jeweiligen Inhaber.

# Inhalt

<b>1 Informationen zu diesem Handbuch</b>	<b>5</b>
1.1 An wen richtet sich dieses Handbuch?	5
1.2 Formatierungskonventionen	5
1.3 Feedback zur Dokumentation	6
<b>2 Info zur OpenScape Contact Center IVR-API</b>	<b>7</b>
2.1 Übersicht	7
2.2 Installieren der OpenScape Contact Center IVR-API	9
2.3 IVR-API-Funktionen für nicht überwachte IVR-Systeme	9
<b>3 Konfigurieren der OpenScape Contact Center IVR-API</b>	<b>13</b>
3.1 Konfiguration von Halten im IVR	13
3.2 Konfiguration von Halten in Warteschlange	15
3.3 Konfiguration von FMNQ	17
3.4 Schreiben eines IVR-Skripts	17
3.4.1 Schreiben eines IVR-Skripts für die Konfiguration Halten im IVR	18
3.4.2 Schreiben eines IVR-Skripts für die Konfiguration Halten in Warteschlange	20
3.4.3 Schreiben eines IVR-Skripts für die Konfiguration FMNQ	22
3.5 Verwenden eines IVR-Systems in einer Multi-Tenant-Umgebung	24
<b>4 Verwenden der OpenScape Contact Center IVR-API-Funktionen</b>	<b>25</b>
4.1 Übermitteln von Parametern	25
4.2 CallerDisconnected	26
4.3 CreateCallback	27
4.4 DeleteCallback	30
4.5 Dequeue	30
4.6 Trennen	32
4.7 Enqueue	34
4.8 EnqueueForAgent	36
4.9 GetBusinessUnit	39
4.10 GetCallTransferable	41
4.11 GetContactData	42
4.12 GetTransitNumber	44
4.13 Initialize	45
4.14 QueryAgentStatus	47
4.15 QueryCallInfo	48
4.16 QueryCallStatus	51
4.17 QueryQueueStatistics	53
4.18 QueryRoutingInfo	56
4.19 QuerySystemStatus	59
4.20 ReleaseTransitNumber	61
4.21 SetBusinessUnit	62
4.22 SetCallInfo	64
4.23 SetCallTransferable	65
4.24 SetContactData	67
4.25 SetDisplay	68
4.26 Transfer	69
<b>5 Rückgabecodes</b>	<b>73</b>

Stichwortverzeichnis ..... 77

# 1 Informationen zu diesem Handbuch

Dieses Handbuch beschreibt, wie Sie OpenScape Contact Center über die OpenScape Contact Center IVR-API-Schnittstelle mit einem Interactive Voice Response (IVR)-System integrieren.

## 1.1 An wen richtet sich dieses Handbuch?

Diese Bedienungsanleitung richtet sich an Systemintegratoren, die ein IVR-System in OpenScape Contact Center integrieren möchten.

## 1.2 Formatierungskonventionen

In diesem Handbuch werden folgende Formatierungskonventionen verwendet:

### **Fettdruck**

In dieser Formatierung erscheinen OpenScape Contact Center Komponenten, Fenster- und Dialogfeldtitel sowie Elementnamen.

### *Kursiv*

In dieser Formatierung erscheinen Verweise auf verwandte Dokumentationen.

### `Nichtproportionale Schrift`

In dieser Schrift erscheint Text, den Sie eingeben müssen oder der vom Computer in einer Meldung angezeigt wird.

---

**HINWEIS:** Hinweise heben Informationen hervor, die nützlich, aber nicht wesentlich sind, zum Beispiel Tipps oder alternative Methoden zum Durchführen einer Aufgabe.

---

---

**WICHTIG:** Wichtige Hinweise machen auf Aktionen aufmerksam, die den Betrieb der Anwendung beeinträchtigen oder zum Verlust von Daten führen können.

---

## 1.3 Feedback zur Dokumentation

Wenn Sie Probleme im Zusammenhang mit diesem Dokument mitteilen möchten, wenden Sie sich bitte an das Kundendienst-Center.

Bitte halten Sie bei Ihrem Anruf folgende Angaben bereit. Dadurch können wir das Dokument, mit dem Sie Schwierigkeiten haben, schneller identifizieren.

- **Titel:** IVR API Integrationshandbuch
- **Sachnummer:** A31003-S2280-N102-1-20

## 2 Info zur OpenScape Contact Center IVR-API

Dieses Kapitel bietet eine Übersicht über die OpenScape Contact Center IVR-API.

### 2.1 Übersicht

Ein IVR-System interagiert mit Anrufern, um zusätzliche Informationen zu ihren Anforderungen abzufragen, und stellt den Anrufern Informationen bereit, während sie in der Warteschlange darauf warten, dass ihr Anruf von einem geeigneten Agenten bearbeitet wird.

Sie können ein benutzerdefiniertes IVR-Skript erstellen, um Informationen von Ihren Kunden einzuholen. Beispielsweise kann der Anrufer aufgefordert werden, die Taste 1 für Service oder die Taste 2 für Vertrieb zu drücken.

Das IVR-Skript kann die Anforderungen eines Kunden aus den folgenden Quellen einholen:

- **Netzwerkinformationen (öffentliche Informationen zur Kommunikationsplattform)**
  - Automatische Rufnummeridentifikation (ANI) – Informationen über den rufenden Teilnehmer
  - Anrufidentifizierung (DNIS) – Informationen über den gerufenen Teilnehmer
- **Call Prompting**
  - Dual-Tone Multi-Frequency (DTMF) (MFV-Signalisierung) – Tasteneingaben vom Anrufer (beispielsweise Konto- oder persönliche ID-Nummer)
  - Spracherkennung – Sprachantworten vom Anrufer auf speziellere Abfragen (zum Beispiel könnte der Kunde aufgefordert werden, entweder Vertrieb oder Service zu wählen)
- **Kundendatenbank**
  - Kundenprofil (beispielsweise Sprache, Fähigkeits-Level und Standort)
  - Kundenhistorie (beispielsweise Kreditwürdigkeit, Abschlüsse in der Vergangenheit und offene Probleme)
  - Kundenpräferenzen (beispielsweise ein bestimmter Agent)

Das IVR-Skript hat die Aufgabe, eine Warteschlange zu identifizieren, die die sich ändernden Anforderungen eines Anrufs während der Wartezeit in der Warteschlange beschreibt.

Sie können ein IVR-Skript erstellen, das ANI- und DNIS-Informationen von OpenScape Contact Center erfasst oder den Anrufer auffordert, bestimmte Informationen einzugeben, und dann diese Informationen nutzen, um auf Daten von einem Host oder einer lokalen Datenbank zuzugreifen.

Beispielsweise könnten Sie einen Anrufer auffordern, seine vierstellige Kontonummer einzugeben, und dann mithilfe dieser Nummer auf Datenbank-Informationen zuzugreifen und basierend auf den DNIS-Informationen oder den Ursprungs-/Zielinformationen Warteschlangen für die Anrufe identifizieren.

Der Text im Anschluss an das Diagramm enthält die nummerierten Schritte.

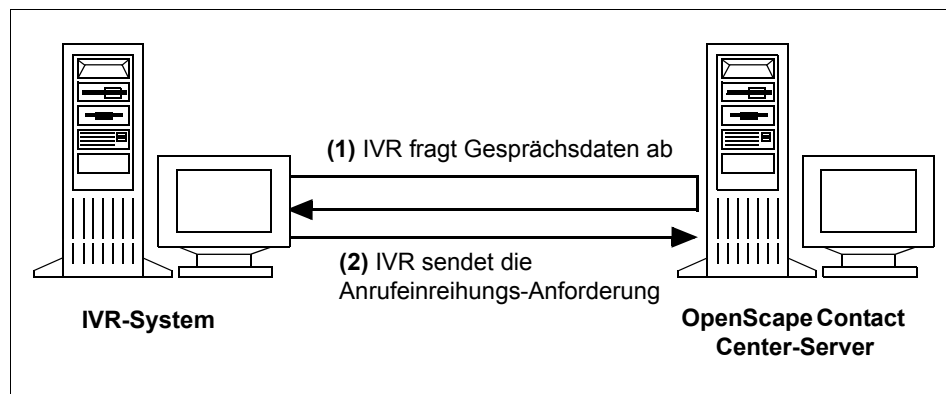


Bild 1 Interaktion zwischen IVR-System und OpenScape Contact Center

1. Wenn ein Anruf bei Ihnen ankommt, wird er vom IVR-System angenommen. Das IVR-Skript sammelt anschließend die ANI- und DNIS- oder An- und Von-Informationen der Transaktion, bestimmt die Warteschlange und sendet eine Anruferreihungs-Anforderung an OpenScape Contact Center.

---

**HINWEIS:** Das IVR-System kann auch eine OpenScape Contact Center-Routinganforderung an OpenScape Contact Center senden, in der vom IVR-System eingeholte Informationen bereitgestellt werden. Der Routing-Server weist dann basierend auf der konfigurierten System-Routingoption eine Warteschlange und andere Routing-Informationen zu, die zum Senden einer Anruferreihungs-Anforderung verwendet werden können.

---

2. Der Routing-Server sucht dann den bestgeeigneten verfügbaren Agenten für die Bearbeitung des Anrufs.
3. Das IVR-Skript kann den Kunden eine Reihe von Menüs präsentieren, um die speziellen Anforderungen jedes Kunden zu ermitteln. Je komplexer Ihr Menüsystem ist, desto detailliertere Informationen können Sie von Ihren Kunden einholen.



Jede Menüauswahl sollte einer eigenen Warteschlange zugeordnet sein, damit Sie die Funktionalität des fähigkeitsbasierten OpenScape Contact Center-Routings optimal nutzen können.

---

**HINWEIS:** Weitere Informationen zur Konzeption von Menüs für IVR-Systeme erhalten Sie von Ihrem Servicevertreter.

---

---

**HINWEIS:** Weitere Informationen zu den verfügbaren IVR-Funktionen finden Sie im [Kapitel 4, "Verwenden der OpenScape Contact Center IVR-API-Funktionen"](#).

---

## 2.2 Installieren der OpenScape Contact Center IVR-API

Zum Installieren der OpenScape Contact Center IVR-API müssen Sie die Datei HPPCIVR.DLL vom Standard-Installationsordner auf dem OpenScape Contact Center Haupt-Server-Computer auf den IVR-Computer kopieren und in der IVR-Konfiguration auf die Datei verweisen.

## 2.3 IVR-API-Funktionen für nicht überwachte IVR-Systeme

OpenScape Contact Center verwendet IVR-API-Funktionen, um die Integrierung von unüberwachten IVR-Ressourcen zu erleichtern.

Da dieser Typ von IVR-System nicht von OpenScape Contact Center überwacht wird, müssen folgende Aspekte berücksichtigt werden:

- OpenScape Contact Center wird mit ANI- und DNIS-Nummern benachrichtigt, wenn der Anruf beim IVR-System ankommt.
- OpenScape Contact Center und das IVR-System müssen koordiniert zusammenarbeiten, um einen Anruf vom IVR-System an OpenScape Contact Center weiterzuleiten.
- Das IVR-System verlassende Anrufe, die entweder getrennt oder extern weitergeleitet werden, verwenden eine neue IVR API, um OpenScape Contact Center zu benachrichtigen.

Das IVR-System muss dem OpenScape Contact Center-System die ANI- und DNIS-Nummer des Anrufs bereitstellen. Diese Informationen können vom IVR-System über das Trunking-Protokoll der Kommunikationsplattform (zum Beispiel ISDN) abgerufen werden. Die neue SetCallInfo-API legt die ANI- und DNIS-Nummer lokal fest, bevor diese Informationen mittels der vorhandenen QueryCallInfo-API an OpenScape Contact Center gesendet werden.

Die QueryCallInfo-API wird vom IVR-System verwendet, um OpenScape Contact Center zu benachrichtigen, wenn ein Anruf im IVR-System ankommt. Diese API sollte sobald wie möglich aufgerufen werden, nachdem das IVR-System mit der Verarbeitung des ankommenden Anrufs begonnen hat. Die QueryCallInfo-API verfügt über einen optionalen Parameter, TrunkConnectedIVRCall, der für nicht überwachte IVR-Systeme auf "1;" gesetzt werden muss.

Anrufe auf einem nicht überwachten IVR-System, die bei OpenScape Contact Center für Halten im IVR oder Halten in Warteschlange eingereiht werden, müssen mithilfe einer Transitnummer an die OpenScape Contact Center Route-Control-Gruppe weitergeleitet werden. Eine Transitnummer ist eine Pilotnummer zur OpenScape Contact Center Route-Control-Gruppe. Die Transitnummer wird für die Nachverfolgung des nicht überwachten IVR-Anrufs nach der Weiterleitung an OpenScape Contact Center benötigt.

OpenScape Contact Center verwaltet einen Pool von Transitnummern, die nur für die Weiterleitung von nicht überwachten IVR-Anrufen verwendet werden. Das IVR-System fragt bei OpenScape Contact Center unter Verwendung der GetTransitNumber-API eine eindeutige Transitnummer ab. Nach dem Empfang der Transitnummer muss das IVR-System den Anruf sofort weiterleiten. Wenn der Anruf die OpenScape Contact Center RCG erreicht, identifiziert OpenScape Contact Center den Anruf anhand der Transitnummer, und kann dann die zuvor vom IVR-System empfangenen Anrufinformationen mit dem ankommenden Anruf verbinden. Die Transitnummer steht dann wieder für die Verwendung mit einem anderen unüberwachten IVR-Anruf zur Verfügung.

---

**HINWEIS:** Bei einem Fehler oder einer Zeitüberschreitung kann das IVR-System den Anruf an ein Nicht-OpenScape-Contact-Center-Gerät (zum Beispiel Voice Mail oder Xpressions) weiterleiten. In diesem Fall muss GetTransitNumber nicht aufgerufen werden.

---

Sie können die API GetTransitNumber unterschiedlich einsetzen, je nachdem ob Ihre Kommunikationsplattform Halten in Warteschlange oder Halten im IVR verwendet:

- **Halten in Warteschlange** – GetTransitNumber muss unmittelbar vor der Weiterleitung des Anrufs an die OpenScape Contact Center RCG aufgerufen werden, also normalerweise kurz nach Einreihung des Anrufs. OpenScape Contact Center sucht dann nach einem Agenten und leitet den Anruf an den zugeordneten Agenten um.
- **Halten im IVR** – GetTransitNumber muss sofort aufgerufen werden, bevor der Anruf weitergeleitet wird. Der Anruf wird auch hier mithilfe einer Transitnummer an die OpenScape Contact Center RCG weitergeleitet. Bei Halten im IVR heißt das normalerweise, kurz nachdem der von der QueryCallStatus-API zurückgegebene Anrufstatus "Wartend" oder "Nicht angenommen" ist. Nach der Weiterleitung des Anrufs an die OpenScape

Contact Center RCG leitet OpenScape Contact Center den Anruf sofort an den zugeordneten Agenten oder die definierte Zeitüberschreitungs-Nebenstelle weiter.

Wenn eine Transitnummer angefordert wird und das IVR-System den Anruf nicht sofort weiterleiten kann, muss die API `ReleaseTransitNumber` aufgerufen werden. Diese macht die Transitnummer für einen anderen Anruf verfügbar. Vor dem Versuch, den Anruf erneut an OpenScape Contact Center weiterzuleiten, muss die `GetTransitNumber`-API erneut aufgerufen werden, um eine neue Transitnummer zu erhalten.

Wenn ein Anruf auf einem nicht überwachten IVR-System getrennt (durch das IVR-System oder durch Abbruch des Anrufers) oder extern zu einer Nummer weitergeleitet wird, die keine Transitnummer ist, wird die neue `CallerDisconnected`-API verwendet, um OpenScape Contact Center zu benachrichtigen.

---

**HINWEIS:** Die API `CallerDisconnected` sollte zu der Aufhängen-Verzweigung des IVR-Systems hinzugefügt und gegebenenfalls in andere Fehlerbehandlungsprozeduren integriert werden. Wenn Sie die API nicht zu der Aufhängen-Verzweigung hinzufügen, werden Ihre Statistiken ungenau sein, weil OpenScape Contact Center dann nicht erfährt, wann der Anruf beendet wurde.

---

#### **Beispiel: Schrittfolge für nicht überwacht IVR-System (Halten im IVR)**

Nachfolgend sehen Sie ein Beispiel für die Schrittfolge eines IVR-Systems für ein nicht überwacht IVR-System:

1. Ankommender IVR-Anruf wird angenommen.
2. Aufruf von `Initialize()`, gefolgt von `SetCallInfo()`, gefolgt von `QueryCallInfo()`.
3. Abspielen der Begrüßungsnachricht/des Hauptmenüs und Verarbeiten des Anrufs, bis er bereit ist, bei OpenScape Contact Center eingereicht zu werden.
4. Aufruf von `Enqueue()`.
5. Aufruf von `QueryCallStatus()`, bis Anrufstatus "Wartend" oder "Nicht angenommen" ist.
6. Aufruf von `GetTransitNumber()`.
7. Sofortige Weiterleitung des Anrufs anhand der von `GetTransitNumber()` zurückgegebenen Nummer.
8. OpenScape Contact Center leitet den Anruf an den zugeordneten Agenten oder die definierte Zeitüberschreitungs-Nebenstelle weiter.

Die genauen Schritte für einzelne IVR-Systeme können von den oben aufgeführten Schritten abweichen. Wichtig ist, dass Folgendes beachtet wird:

## Info zur OpenScape Contact Center IVR-API

### *IVR-API-Funktionen für nicht überwachte IVR-Systeme*

- Die Reihenfolge, in der Initialize(), SetCallInfo() und QueryCallInfo() aufgerufen werden.
- QueryCallInfo() wird sobald wie möglich aufgerufen, sodass OpenScape Contact Center die Verfolgung des Anrufs ab dem Zeitpunkt, zu dem er im IVR-System ankommt, starten kann.
- GetTransitNumber() wird erst aufgerufen, wenn der Anruf kurz vor der Weiterleitung aus dem IVR-System an eine OpenScape Contact Center Ressource (zum Beispiel Agent, RCG oder Zeitüberschreitungs-Nebenstelle) steht, und die Weiterleitung muss sofort erfolgen, nachdem GetTransitNumber() mit einer Transitnummer zurückgegeben wird.

## 3 Konfigurieren der OpenScape Contact Center IVR-API

Dieses Kapitel beschreibt die verschiedenen IVR-Szenarien, die Sie verwenden können, um die Erfassung der Anforderungen Ihrer Kunden zu verbessern. Die Szenarien sind:

- Halten im IVR
- Halten in Warteschlange
- FMNQ (Forget-Me-Not-Queueing)

---

**HINWEIS:** Je nach erforderlicher Erfahrung des Anrufers kann das IVR-Skript eines oder mehrere dieser Szenarien verwenden. Sie erfordern alle InterSoft für Windows 2000.

---

### 3.1 Konfiguration von Halten im IVR

Sie können die Konfiguration Halten im IVR mit einem IVR-System verwenden, um einen eingereichten Anruf zu halten, bis er an einen Agenten weitergeleitet wird.

---

**HINWEIS:** Da die Konfiguration Halten im IVR die IVR-Nebenstelle belegt und vorübergehend verhindert, dass das IVR-System ankommende Anrufe bearbeitet, bis der eingereichte Anruf an einen Agenten weitergeleitet wurde, müssen Sie die Konfiguration Halten im IVR ändern, um weitere IVR-Nebenstellen aufzunehmen.

---

Die nachfolgende Abbildung zeigt den Anrufablauf für die Konfiguration Halten im IVR. Der Text im Anschluss an das Anrufablauf-Diagramm enthält eine Beschreibung der nummerierten Schritte.

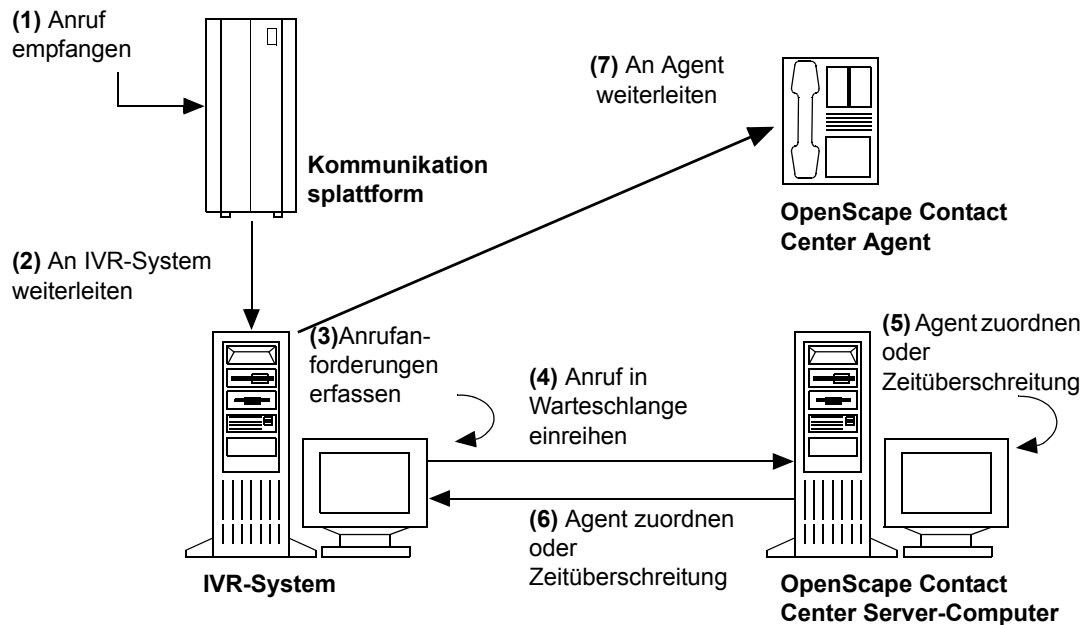


Bild 2 Anrufablauf für die Konfiguration Halten im IVR

1. Ein neuer Anruf geht ein.
2. Der Anruf wird sofort an ein IVR-System weitergeleitet.
3. Das IVR-System holt Informationen vom Anrufer zum Zweck seines Anrufs ein und identifiziert dann anhand dieser Informationen die Routing-Parameter für den Anruf, einschließlich der Warteschlange.
4. Das IVR-System sendet eine Einreihungsanforderung für den Anruf an OpenScape Contact Center. Der Routing-Server sucht dann den bestgeeigneten verfügbaren Agenten für die Bearbeitung des Anrufs.
5. OpenScape Contact Center führt dann eine der folgenden Aktionen aus:
  - Es ordnet den Anruf dem besten verfügbaren Agenten zu
  - Es reserviert den Anruf für einen bestimmten Agenten (optional)Wenn der Anruf bis zum Ende des letzten Anrufschriffs keinem Agenten zugeordnet werden kann, wird er an eine Zeitüberschreitungs-Nebenstelle weitergeleitet.
6. OpenScape Contact Center benachrichtigt das IVR-System über die Nebenstelle des zugeordneten Agenten oder des Zeitüberschreitungsziels. Wenn der Anruf die Zeit überschreitet, führt das IVR-System eine der folgenden Aktionen aus:

- Es leitet den Anruf an eine im IVR-Skript definierte Standardnummer oder eine vom Routing-Server bereitgestellte Zeitüberschreitungs-Nebenstelle weiter. Die Zeitüberschreitungs-Nebenstelle kann entweder eine spezielle von der Warteschlange des Anrufs definierte Nummer oder eine globale Standardnummer sein, die von allen Warteschlangen ohne spezielle definierte Nummer verwendet wird.
  - Es fragt zusätzliche Informationen vom Anrufer ab und verwendet die Routing-Parameter, um den Anruf erneut einzureihen. An dieser Stelle wird der Suchvorgang nach einem geeigneten Agenten wiederholt.
7. Das IVR-System leitet den Anruf an den zugeordneten Agenten weiter.

## 3.2 Konfiguration von Halten in Warteschlange

Sie können die Konfiguration Halten in Warteschlange mit einem IVR-System verwenden, um Anrufe an eine OpenScape Contact Center-ACD/UCD/MLHG-Gruppe in der Kommunikationsplattform weiterzuleiten, wo Anrufe warten, bis geeignete Agenten für die Bearbeitung der Anrufe verfügbar werden.

Sobald ein Agent verfügbar wird, leitet OpenScape Contact Center den Anruf aus der OpenScape Contact Center ACD/UCD/MLHG-Gruppe an den betreffenden Agenten um. Die Konfiguration Halten in Warteschlange entlastet die IVR-Nebenstellen für ankommende Anrufe, indem Anrufe an die Kommunikationsplattform weitergeleitet werden, während der Routing-Server nach verfügbaren Agenten sucht.

Bei dieser Konfiguration ist eine geringere Anzahl von IVR-Nebenstellen erforderlich, sodass Ihr IVR-System eine größere Anzahl ankommender Anrufe bearbeiten kann.

Die folgende Abbildung zeigt den Anrufablauf für die Konfiguration Halten in Warteschlange. Der Text im Anschluss an das Anrufablauf-Diagramm enthält eine Beschreibung der nummerierten Schritte.

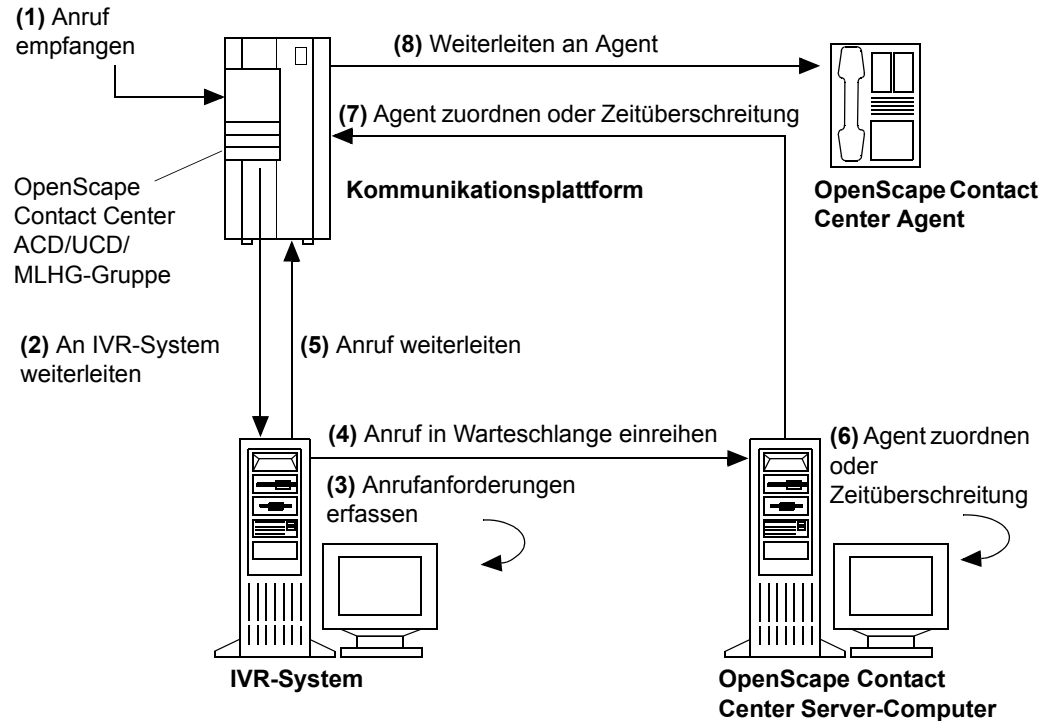


Bild 3 Anrufablauf für die Konfiguration Halten in Warteschlange

1. Ein neuer Anruf geht ein.
2. Der Anruf wird sofort an ein IVR-System weitergeleitet.
3. Das IVR-System holt Informationen vom Anrufer zum Zweck seines Anrufs ein und identifiziert dann anhand dieser Informationen die Routing-Parameter für den Anruf, einschließlich der Warteschlange.
4. Das IVR-System sendet eine Einreihungsanforderung für den Anruf an OpenScape Contact Center. Der Routing-Server sucht dann den bestgeeigneten verfügbaren Agenten für die Bearbeitung des Anrufs.
5. Zu diesem Zeitpunkt leitet das IVR-System den Anruf an eine OpenScape Contact Center ACD/UCD/MLHG-Gruppe weiter.
6. OpenScape Contact Center führt dann eine der folgenden Aktionen aus:
  - Es ordnet den Anruf dem besten verfügbaren Agenten zu
  - Es reserviert den Anruf für einen bestimmten Agenten (optional)

Wenn der Anruf bis zum Ende des letzten Anrufschriffs keinem Agenten zugeordnet werden kann, wird er an eine Zeitüberschreitungs-Nebenstelle weitergeleitet.



7. Der Anruf wird einem bestimmten Agenten oder einem Zeitüberschreitungsziel zugeordnet.
8. Der Anruf wird an den zugeordneten Agenten weitergeleitet.

### 3.3 Konfiguration von FMNQ

Sie können einen FMNQ-Anruf so konfigurieren, dass er entweder "unterbrechbar" oder "nicht unterbrechbar" ist. Wenn der Anruf auf "unterbrechbar" gesetzt ist, können die Anrufer von einem IVR-System zu der Nebenstelle des besten verfügbaren Agenten verschoben werden, und zwar unabhängig davon, ob sie Informationen in das IVR-System eingeben. Wenn der Anruf "nicht unterbrechbar" ist, wird die Weiterleitung des Anrufs vorübergehend ausgesetzt, bis der Anrufer:

- Wichtige Informationen eingibt, ohne dass das Risiko besteht, dass diese Informationen verlorengehen
- Wichtige Informationen ohne Unterbrechung anhört

---

**HINWEIS:** Nicht überwachte IVR-Ressourcen bieten keine Unterstützung für FMNQ.

---

### 3.4 Schreiben eines IVR-Skripts

Ein angepasstes IVR-Skript sammelt Informationen von Ihren Kunden, stellt Funktionen zur Anrufverwaltung bereit und ruft die in OpenScape Contact Center enthaltenen IVR-API-Funktionen auf.

Die IVR-API-Funktionen werden in einer Moduldatei mit dem Namen **HPPCIVR.DLL** bereitgestellt, die beim Ausführen des IVR-Skripts angegeben werden muss. Diese Moduldatei befindet sich im Standardverzeichnis von OpenScape Contact Center auf dem Haupt-Server-Computer.

#### 3.4.1 Schreiben eines IVR-Skripts für die Konfiguration Halten im IVR

Nachfolgend sehen Sie einen Ablauf-Vorschlag für ein IVR-Skript für die Konfiguration Halten im IVR:

1. Initialisieren Sie die Verbindung zwischen dem IVR-System und OpenScape Contact Center.
2. Überprüfen Sie den Systemstatus, um sicherzustellen, dass der Routing-Server verfügbar ist.
3. Holen Sie die ANI- und DNIS-Informationen für den Anruf ein.
4. Identifizieren Sie die Warteschlange mithilfe des IVR-Skripts. Sie können zum Beispiel dem Anruf eine Warteschlange basierend auf einer Kombination der ANI-Informationen des Anrufs und der Auswahl des Kunden zuweisen.
5. Reißen Sie den Anruf mithilfe der Enqueue-Funktion in die Warteschlange ein. Stellen Sie sicher, dass die Parameter definieren, dass das IVR-System den Anruf mithilfe der Konfiguration Halten im IVR hält, bis der Anruf einem Agenten zugeordnet ist. Dadurch wird der Routing-Server benachrichtigt, dass das IVR-System den Anruf weiterleiten wird, sobald er zugeordnet ist.
6. Überprüfen Sie den Zustand des eingereichten Anrufs regelmäßig (zum Beispiel nach jeder Aktion oder jedem Aufruf einer IVR-Funktion), und ergreifen Sie je nach den Ergebnissen die entsprechende Maßnahme:
  - Wenn der Anruf zugeordnet ist, leiten Sie den Anruf an die Nebenstelle des Agenten weiter.
  - Wenn der Anruf die Zeit überschritten hat, leiten Sie den Anruf an die im IVR-Skript definierte Standardnummer oder an die vom Routing-Server zurückgegebene Nebenstelle weiter.
  - Wenn ein Routing-Server-Fehler auftritt, leiten Sie den Anruf an die im IVR-Skript definierte Standardnummer weiter.

Das folgende Diagramm zeigt ein IVR-Beispielskript für die Konfiguration Halten im IVR.

---

**HINWEIS:** Nach der erfolgreichen Einreihung eines Anrufs muss das IVR-Skript mithilfe der Funktion QueryCallStatus regelmäßig den Zustand des Anrufs überprüfen und bestimmte Bedingungen bearbeiten, zum Beispiel Einreihungsfehler und Anrufe mit Zeitüberschreitung. In diesen Fällen sollte das IVR-Skript den Anruf entweder an eine Nicht-OpenScape-Contact-Center-Nebenstelle weiterleiten oder mit anderen Parametern in die Warteschlange einreihen.

---

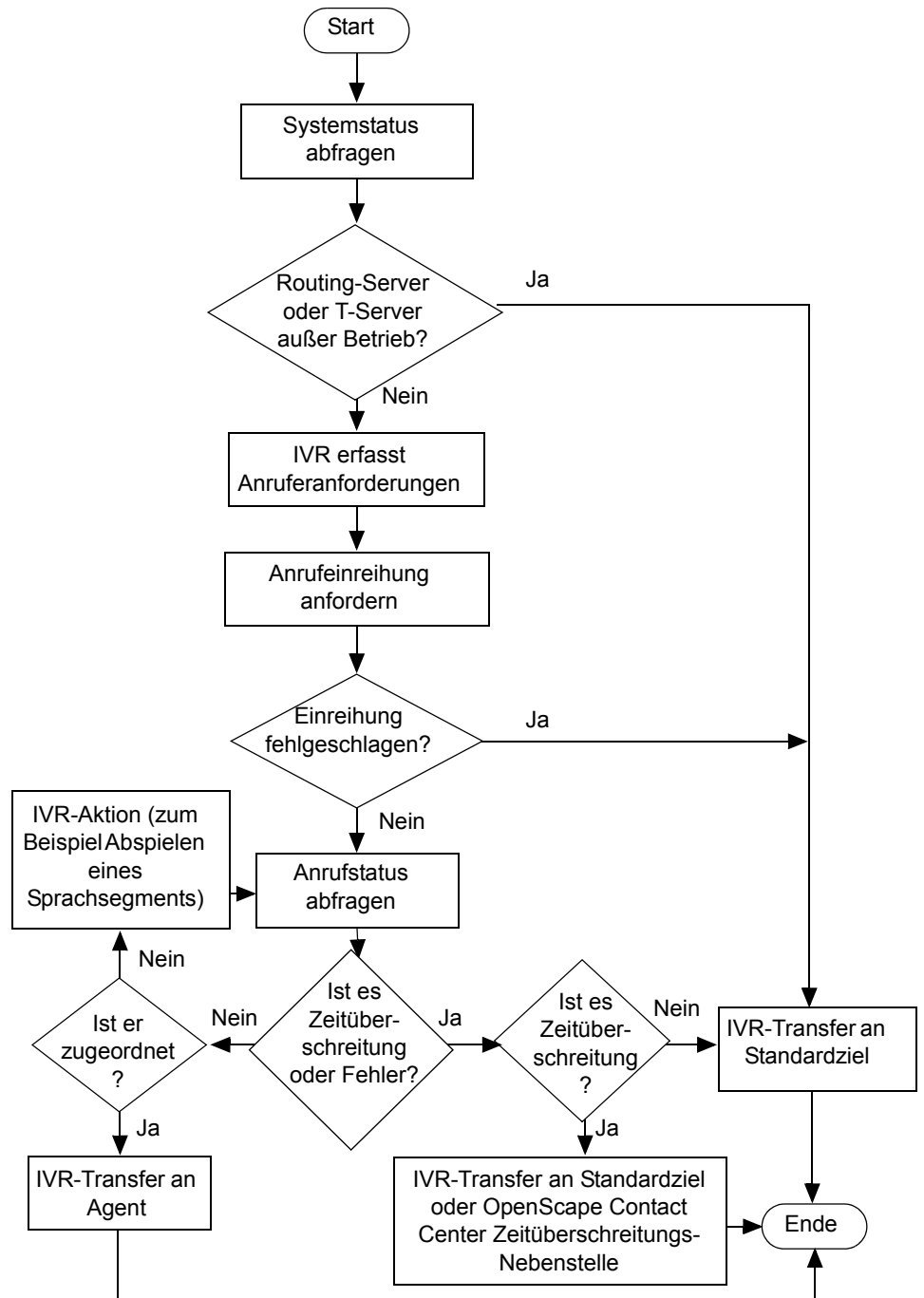


Bild 4

Beispiel-Ablaufdiagramm für ein IVR-Skript in der Konfiguration Halten im IVR

### 3.4.2 Schreiben eines IVR-Skripts für die Konfiguration Halten in Warteschlange

Nachfolgend sehen Sie einen Ablauf-Vorschlag für ein angepasstes IVR-Skript für die Konfiguration Halten in Warteschlange:

1. Initialisieren Sie die Verbindung zwischen dem IVR-System und OpenScape Contact Center.
2. Überprüfen Sie den Systemstatus, um sicherzustellen, dass der Routing-Server verfügbar ist.
3. Holen Sie die ANI- und DNIS-Informationen für den Anruf ein.
4. Identifizieren Sie die Warteschlange mithilfe des IVR-Skripts. Sie können zum Beispiel dem Anruf eine Warteschlange basierend auf einer Kombination der ANI-Informationen des Anrufs und der Auswahl des Kunden zuweisen.
5. Reihen Sie den Anruf mithilfe der Enqueue-Funktion in die Warteschlange ein. Stellen Sie sicher, dass die Parameter definieren, dass das IVR-System den Anruf mithilfe der Konfiguration Halten in Warteschlange an eine OpenScape Contact Center ACD/UCD/MLHG-Gruppe weiterleitet. Dadurch wird der Routing-Server benachrichtigt, dass OpenScape Contact Center den Anruf weiterleiten muss, sobald er zugeordnet ist.
6. Anruf an eine OpenScape Contact Center ACD/UCD/MLHG-Gruppe weiterleiten. OpenScape Contact Center hält den Anruf automatisch in der OpenScape Contact Center ACD/UCD/MLHG-Gruppe, bis er einem Agenten zugeordnet wird.

Das folgende Diagramm zeigt ein IVR-Beispielskript für die Konfiguration Halten in Warteschlange.

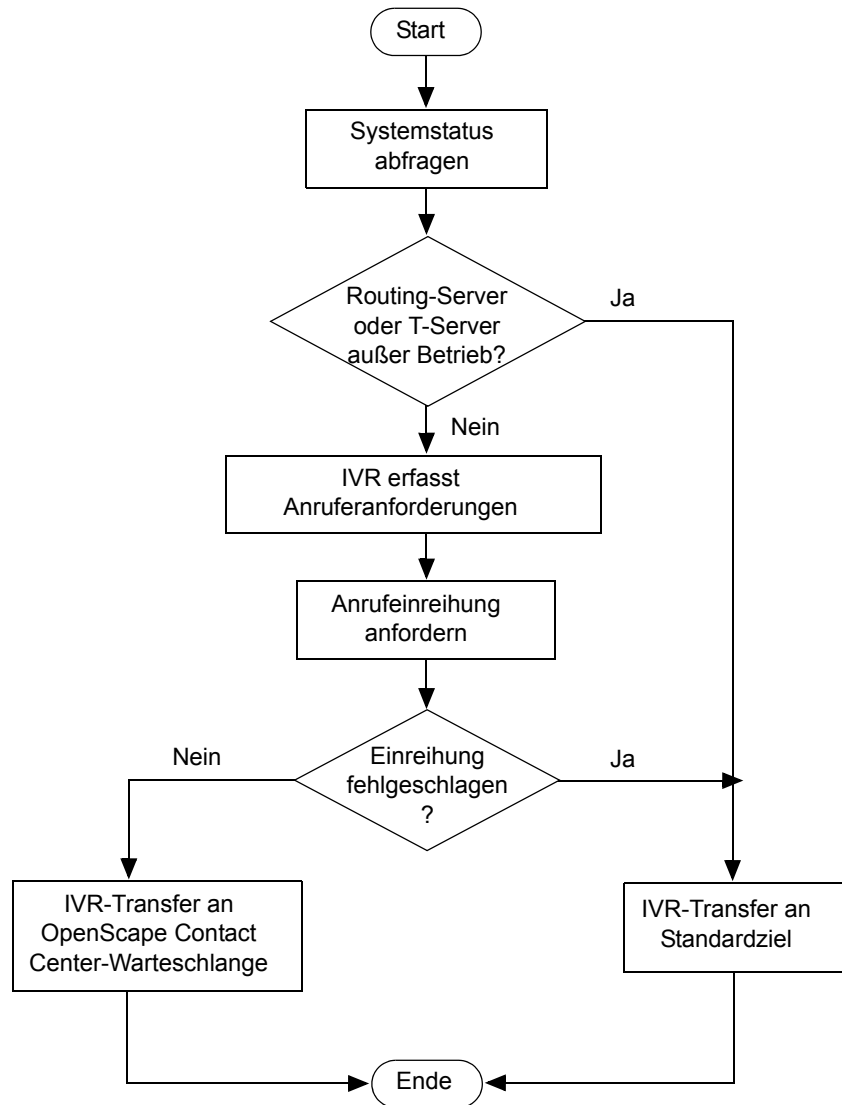


Bild 5

Beispiel-Ablaufdiagramm für ein IVR-Skript in der Konfiguration Halten in Warteschlange

### 3.4.3 Schreiben eines IVR-Skripts für die Konfiguration FMNQ

Nachfolgend sehen Sie einen Ablauf-Vorschlag für ein angepasstes IVR-Skript für die FMNQ-Konfiguration:

1. Initialisieren Sie die Verbindung zwischen dem IVR-System und OpenScape Contact Center.
2. Verwenden Sie die Funktion QueryCallInfo, um die ANI- und DNIS-Informationen für den Anruf zu erfassen und den Wert des FMNQ-Flags abzurufen.
3. Wenn das Flag auf 1 gesetzt ist, starten Sie den FMNQ-Anrufablauf und bestimmen Sie, ob der Anrufer eine "unterbrechbare" oder "nicht unterbrechbare" Sitzung starten möchte.
4. Wenn der Anrufer eine "nicht unterbrechbare" Sitzung starten möchte, setzen Sie den Anruf auf nicht transferierbar (0).
5. Fahren Sie in der "nicht unterbrechbaren" Sitzung mit der Erfassung von Anruferanforderungen fort.
6. Wenn die "nicht unterbrechbare" Sitzung beendet ist, setzen Sie den Anruf auf transferierbar (1).

---

**HINWEIS:** Bei Verbindung zu der OpenScape Voice-Kommunikationsplattform müssen Sie sicherstellen, dass der Anruf an den Wartemusik-Sammelanschluss (MLHG) weitergeleitet wird.

---

---

**HINWEIS:** Bevor das IVR-Skript einen FMNQ-Anruf trennt oder weiterleitet, müssen Sie SetCallTransferable (0) aufrufen und einen erfolgreichen Rückgabecode empfangen. Wenn Sie keinen erfolgreichen Rückgabecode empfangen, weist dies darauf hin, dass der Anruf gerade weitergeleitet wird, und Sie müssen warten, bis der Prozess abgeschlossen ist.

---

Das folgende Diagramm zeigt ein IVR-Beispielskript für die FMNQ-Konfiguration.

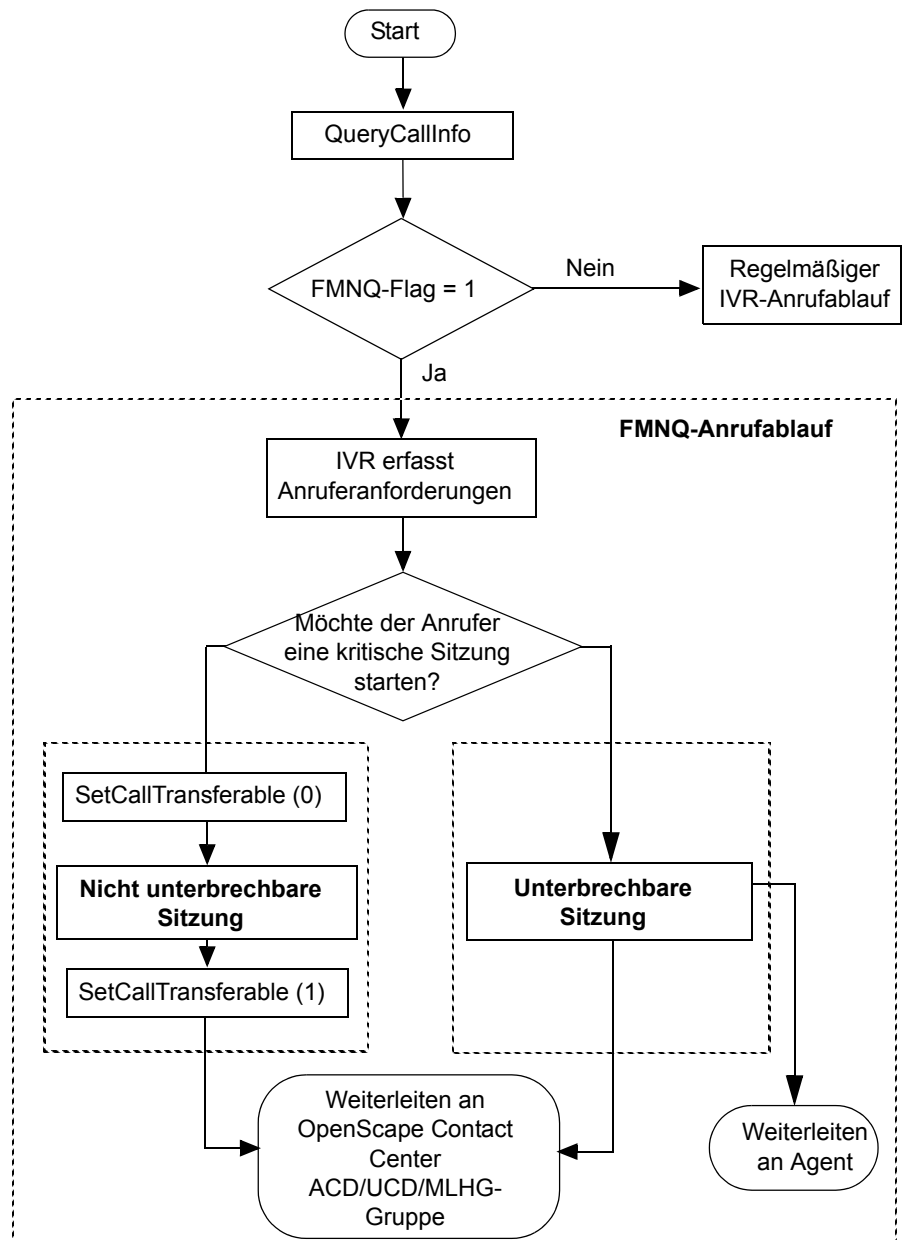


Bild 6

Beispiel-Ablaufdiagramm für ein IVR-Skript in der FMNQ-Konfiguration

## 3.5 Verwenden eines IVR-Systems in einer Multi-Tenant-Umgebung

In einer Multi-Tenant-Umgebung sind IVR-Ressourcen Ressourcen auf Systemebene, die von mehreren Business Units gemeinsam genutzt werden.

Wenn Sie eine Front-End-IVR-Anwendung in einer Multi-Tenant-Umgebung einsetzen, muss das IVR-System die Logik enthalten, mit der sie die Business Units erkennen kann.

Beispiel:

- Das IVR-System muss bestimmen, an welche Business Unit ein IVR-Anruf weitergeleitet wird.
- Wenn ein IVR-System einen Anruf in das OpenScape Contact Center-System einreicht, muss diese Einreihung bei der richtigen Business Unit erfolgen.
- In einem Szenario Halten in Warteschlange muss ein IVR-System die korrekte Pilotnummer für jede Business Unit kennen und muss den Anruf an die richtige Business Unit weiterleiten.

Für die Verwendung in einer Multi-Tenant-Umgebung werden zwei API-Funktionen bereitgestellt:

- Funktion `GetBusinessUnit` - ruft die Business Unit für einen bestimmten Anruf ab. Einzelheiten hierzu finden Sie unter [Abschnitt 4.9, "GetBusinessUnit", auf Seite 39](#).
- Funktion `SetBusinessUnit` - legt die Business Unit für einen bestimmten Anruf fest. Die Funktion `SetBusinessUnit` kann pro IVR-Anruf nur einmal aufgerufen werden. Einzelheiten hierzu finden Sie unter [Abschnitt 4.21, "SetBusinessUnit", auf Seite 62](#).

---

**HINWEIS:** In einer Multi-Tenant-Umgebung muss die Funktion `SetBusinessUnit` aufgerufen werden, bevor die folgenden Funktionen aufgerufen werden: `CreateCallback`, `Enqueue`, `EnqueueForAgent`, `GetBusinessUnit`.

---

Ein optionaler Parameter `BusinessUnitName` kann in der Funktion `QueryQueueStatistics` angegeben werden, um die Business Unit anzuzeigen, zu der eine Warteschlange gehört. In einer Nicht-Multi-Tenant-Umgebung wird der Parameter `BusinessUnitName` ignoriert.



## 4 Verwenden der OpenScape Contact Center IVR-API-Funktionen

Dieses Kapitel beschreibt die Verwendung der OpenScape Contact Center IVR-API-Funktionen und enthält Beispiele für das Erstellen von IVR-Skripten.

---

**HINWEIS:** Ein nicht überwachtes IVR-System wird von OpenScape Contact Center nicht überwacht.

---

### 4.1 Übermitteln von Parametern

Beim Aufrufen von OpenScape Contact Center IVR-API-Funktionen können Sie Informationen nach Wert oder nach Referenz übermitteln. Diese beiden Methoden werden in den folgenden Abschnitten beschrieben.

#### Übermitteln von Parametern nach Wert

Wenn Sie Parameter nach Wert übermitteln, geben Sie Informationen vom IVR-Skript an OpenScape Contact Center weiter. Jede der IVR-API-Funktionen erfordert, dass unterschiedliche Informationen an OpenScape Contact Center gesendet werden, damit die Anwendung die angeforderte Aktion ausführen kann. Beispielsweise muss das IVR-Skript beim Aufrufen der Funktion Enqueue den Warteschlangennamen an OpenScape Contact Center übermitteln, damit die Anwendung den Anruf in die Warteschlange einreihen kann.

#### Übermitteln von Parametern nach Referenz

Wenn Sie Parameter nach Referenz übermitteln, geben Sie Informationen von OpenScape Contact Center an das IVR-Skript weiter. Beim Aufrufen einer IVR-API-Funktion gibt das IVR-Skript einen Variablennamen für Variablen an, die nach Referenz übermittelt werden. OpenScape Contact Center gibt die erforderlichen Informationen in der angegebenen Variablen zurück. Beispielsweise gibt OpenScape Contact Center die ANI- und DNIS-Informationen für einen Anruf zurück, wenn das IVR-Skript die Funktion QueryCallInfo aufruft.

---

**HINWEIS:** Nach Referenz übermittelten Parametern muss das Zeichen "größer als" (>) vorangestellt werden.

---

## 4.2 CallerDisconnected

---

**HINWEIS:** Diese Funktion wird von IVR-Skripten nur für nicht überwachte IVR-Systeme verwendet.

---

Die Funktion CallerDisconnected ermöglicht es IVR-Systemen, OpenScape Contact Center über einen abgebrochenen Anruf zu informieren. Der T-Server löscht dann den Anruf und gibt die mit dem Anruf verbundene Transitnummer (falls zutreffend) frei. Anschließend sendet der T-Server eine Benachrichtigung an andere OpenScape Contact Center Komponenten.

### Syntax

CallerDisconnected (CallID)

### Parameter

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die Anruf-ID für den aktuellen Anruf.

Tabelle 1 Parameter für die Funktion CallerDisconnected

### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 308** QueryCallInfo wurde vor dem Ausführen der Funktion nicht aufgerufen. Sie müssen QueryCallInfo aufrufen, um die Anruf-ID zu erhalten.
- 310** Ein nicht per Amtsleitung verbundener Anruf hat versucht, diese Methode zu verwenden. Diese Methode kann nur für per Amtsleitung verbundene Anrufe verwendet werden.
- 910** Die Version der IVR-API-DLL entspricht nicht der Version von OpenScape Contact Center.
- 931** Bei der Überwachung der Anrufart besteht eine Inkonsistenz zwischen der IVR-API und dem T-Server.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

### Beispiel

CallerDisconnected ('1341023126268002')

## 4.3 CreateCallback

Die Funktion CreateCallback versucht, einen Callback im Callback-Server einzurichten. Das IVR-System kann anzeigen, dass ein Kunde zurückgerufen werden soll, wenn ein Agent verfügbar wird. Diese Funktion unterstützt bis zu 1000 Byte Kontaktdaten.

---

**HINWEIS:** Sie müssen die Funktionen Initialize und QueryCallInfo aufrufen, bevor Sie diese Funktion aufrufen. In einer Multi-Tenant-Umgebung müssen Sie zuerst die Funktion SetBusinessUnit aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize", auf Seite 45](#), [Abschnitt 4.15, "QueryCallInfo", auf Seite 48](#) und [Abschnitt 4.21, "SetBusinessUnit", auf Seite 62](#).

---

### Syntax

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

### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Verweis (by reference)	18 Zeichen	Die von QueryCallInfo zurückgegebene Anruf-ID muss eingegeben werden.
QueueName	Zeichenfolge übergeben nach Wert (by value)	32 Zeichen	Die für diesen Callback verwendete Callback-Warteschlange.
Description (optional)	Zeichenfolge übergeben nach Verweis (by reference)	100 Zeichen	Eine Zeichenfolge mit einer Kurzbeschreibung des Anrufs, die in der OpenScape Contact Center Client Desktop-Anwendung angezeigt wird, sodass die Agenten den Anruf identifizieren können.
TimeZoneOffset	Ganzzahl übergeben nach Wert (by value)	0 bis +60*12 oder 0 bis -60*12	Der Unterschied in Minuten zwischen der Ortszeit des Anrufers und UTC (Coordinated Universal Time).
ContactName	Zeichenfolge übergeben nach Wert (by value)	80 Zeichen	Der Name des Kontakts.

Tabelle 2

Parameter für die Funktion CreateCallback

Name	Typ	Bereich	Beschreibung
CallbackID	Zeichenfolge übergeben nach Verweis (by reference)	17 Zeichen	Eine eindeutige ID für den eingerichteten Callback.
Schedule1	Zeichenfolge übergeben nach Wert (by value)	47 Zeichen für die Telefonnummer. 81 Zeichen für die gesamte Zeichenfolge.	Diese Zeichenfolge gibt neben der Telefonnummer, unter der der Anrufer erreichbar ist, den Zeitraum (gemäß Ortszeit des Anrufers) an, während dem der Anrufer Callbacks empfangen möchte. Das Format ist: phoneNumber;mm/tt/jjjj;hh:mm; mm/tt/jjjj;hh:mm Dabei ist die erste Datum/Uhrzeit-Angabe (im 24-Stunden-Format) die Startzeit, die zweite Angabe die Endezeit.
Schedule2 (optional)	Zeichenfolge übergeben nach Wert (by value)	47 Zeichen für die Telefonnummer. 81 Zeichen für die gesamte Zeichenfolge.	Diese Zeichenfolge gibt neben der Telefonnummer, unter der der Anrufer erreichbar ist, den Zeitraum (gemäß Ortszeit des Anrufers) an, während dem der Anrufer Callbacks empfangen möchte. Das Format ist: Telefonnummer;mm/tt/ jjjj;hh:mm; mm/tt/jjjj;hh:mm Dabei ist die erste Datum/Uhrzeit-Angabe (im 24-Stunden-Format) die Startzeit, die zweite Angabe die Endezeit.
Schedule3 (optional)	Zeichenfolge übergeben nach Wert (by value)	47 Zeichen für die Telefonnummer. 81 Zeichen für die gesamte Zeichenfolge.	Diese Zeichenfolge gibt neben der Telefonnummer, unter der der Anrufer erreichbar ist, den Zeitraum (gemäß Ortszeit des Anrufers) an, während dem der Anrufer Callbacks empfangen möchte. Das Format ist: Telefonnummer;mm/tt/ jjjj;hh:mm; mm/tt/jjjj;hh:mm Dabei ist die erste Datum/Uhrzeit-Angabe (im 24-Stunden-Format) die Startzeit, die zweite Angabe die Endezeit.

Tabelle 2 Parameter für die Funktion CreateCallback

### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 308** QueryCallInfo wurde vor dem Ausführen der Funktion nicht aufgerufen. Sie müssen QueryCallInfo aufrufen, um die Anruf-ID zu erhalten.
- 314** Die Funktion hat einen ungültigen Business Unit-Namen übergeben. Der angegebene Business Unit-Name wurde nicht festgelegt oder entspricht keinem der Business Unit-Namen in der Betriebsversion der Datenbank. (Dieser Fehler ist nur relevant, wenn OpenScape Contact Center als Multi-Tenant-System konfiguriert ist.)

- 1006** Sie sind dabei, einen doppelten Callback einzurichten.
- 1028** Eine Zeit des Zeitplans ist ungültig.
- 1029** Die Callback-Einsatzpläne überschneiden sich nicht mit den Contact Center-Betriebszeiten.
- 1031** Eine Telefonnummer in der Anforderung ist ausgeschlossen.
- 1033** Callback kann keine Kontaktdaten akzeptieren, die länger als 1000 Byte sind.
- 1040** Alle Zeitpläne sind bereits abgelaufen.
- 1045** Der Name der Callback-Warteschlange ist ungültig.
- 1047** Maximales Limit der vorausgeplanten Tage.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

### Beispiel

Im folgenden Beispiel wird angefordert, dass OpenScape Contact Center einen Callback für den aktuellen Anruf (identifiziert anhand der Anruf-ID) einrichtet und dabei "CBQueue" als Callback-Warteschlange verwendet.

```
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')
```

Die Callback-ID wird zurückgegeben, nachdem der Callback eingerichtet wurde.

### Format der Telefonnummer

Der Parameter Telefonnummer muss wie folgt im kanonischen Format eingegeben werden:

+**[Länderkennzahl]** **Leerzeichen** **[(Vorwahl) Leerzeichen]** **Teilnehmernummer** **[--Nebenstelle]**

Die Telefonnummer kann auch eine wählbare Adresse sein, die von einer Kommunikationsplattform oder als Ergebnis des Aufrufs einer TAPI-Funktion abgerufen wird. Die folgenden Zeichenfolgen sind korrekte Telefonnummern:

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

Der Nebenstellen-Abschnitt einer Telefonnummer wird vom Callback-Server abgeschnitten, bevor die Nummer an den T-Server gesendet wird. Er wird nur bei der Anzeige des Callback im Client Desktop und im Manager verwendet, sodass ein Benutzer die Nebenstelle gegebenenfalls manuell wählen kann.

## 4.4 DeleteCallback

Die Funktion DeleteCallback wird verwendet, um einen vorhandenen Callback zu löschen, der mit der Funktion CreateCallback eingerichtet wurde. Diese Funktion verwendet als Parameter die von der Funktion CreateCallback zurückgegebene Callback-ID.

---

**HINWEIS:** Sie müssen die Funktionen Initialize und QueryCallInfo aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize", auf Seite 45](#) und [Abschnitt 4.15, "QueryCallInfo", auf Seite 48](#).

---

### Syntax

DeleteCallback (CallbackID)

### Parameter

Name	Typ	Bereich	Beschreibung
CallbackID	Zeichenfolge übergeben nach Wert (by value)	17 Zeichen	Die zurückgegebene Callback-ID einer erfolgreichen CreateCallback-Anforderung, die gelöscht werden soll.

Tabelle 3 Parameter für die Funktion DeleteCallback

### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

**0** Erfolgreich.

**-1021** Falsche CallbackID.

**Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

### Beispiel

Im folgenden Beispiel wird angefordert, den Callback mit der Callback-ID CBCallID zu löschen.

```
DeleteCallback(CBCallID)
```

## 4.5 Dequeue

Die Funktion Dequeue sendet eine Anforderung an den Routing-Server, einen angegebenen Anruf aus der Warteschlange herauszunehmen. Wenn Sie die Konfiguration Halten im IVR verwenden, können Sie mithilfe der Funktion

Dequeue einen Anruf aus der Warteschlange herausnehmen, damit das IVR-Skript den Anruf an eine Nebenstelle weiterleiten oder eine andere Aktion mit dem Anruf durchführen kann.

Wenn das IVR-Skript dem Anrufer erlaubt, die Eingabe von Auswahloptionen nach der Einreihung des Anrufs fortzusetzen, können Sie weitere Anrufereingaben verwenden, um die Weiterleitung des Anrufs zu ändern. Wenn Anrufer beispielsweise zu lange in der Warteschlange warten, könnten sie sich dafür entscheiden, eine Sprachnachricht zu hinterlassen; das IVR-Skript kann dann den Anruf herausnehmen und an eine VoiceMail-Nebenstelle weiterleiten.

---

**HINWEIS:** Sie müssen die Funktionen Initialize und QueryCallInfo aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize", auf Seite 45](#) und [Abschnitt 4.15, "QueryCallInfo", auf Seite 48](#).

---

## Syntax

Dequeue (CallID, Abandoned)

## Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die Anruf-ID für den Anruf, den Sie herausnehmen wollen.
Abandoned	Ganzzahl übergeben nach Wert (by value)	0 oder 1	Ein Flag, das angibt, ob der Anruf abgebrochen wurde. 0 – der Anruf wurde nicht abgebrochen. 1 – der Anruf wurde abgebrochen.

Tabelle 4 Parameter für die Funktion Dequeue

---

**HINWEIS:** Der Parameter Abandoned wirkt sich nur auf Statistiken aus.

---

## Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 304** Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.

- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 308** QueryCallInfo wurde vor dem Ausführen der Funktion nicht aufgerufen. Sie müssen QueryCallInfo aufrufen, um die Anruf-ID zu erhalten.
- 701** Die Funktionsanforderung hat die Zeit überschritten, bevor die Funktion abgeschlossen werden konnte. Dies könnte auf Netzwerkprobleme oder einen ausgelasteten Server hinweisen.
- 801** Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

#### Beispiel

Die folgende Beispielfunktion sendet eine Anforderung zum Herausnehmen eines Anrufs. Der Anruf wurde nicht abgebrochen (das Abandoned-Flag ist 0), sodass das IVR-Skript den Anruf nun an eine Zeitüberschreitungs-Nebenstelle weiterleiten kann.

```
Dequeue('1341023126268002', 0)
```

## 4.6 Trennen

Die Funktion Disconnect trennt den angegebenen Anruf auf der IVR-Nebenstelle.

---

**HINWEIS:** Diese Funktion unterstützt keine nicht überwachten IVR-Anrufe.

---

---

**HINWEIS:** Sie müssen die Funktionen Initialize und QueryCallInfo aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize"](#), auf Seite 45 und [Abschnitt 4.15, "QueryCallInfo"](#), auf Seite 48.

---

#### Syntax

```
Disconnect (CallID)
```



**Parameter**

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die Anruf-ID für den Anruf, den Sie trennen wollen.

Tabelle 5 Parameter für die Funktion Disconnect

**Rückgabecodes**

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 304** Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.
- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 311** Die Funktion unterstützt keine per Amtsleitung verbundenen (nicht überwachten IVR-) Anrufe.
- 701** Die Funktionsanforderung hat die Zeit überschritten, bevor die Funktion abgeschlossen werden konnte. Dies tritt normalerweise nur auf, wenn der Status eines Anrufs abgefragt wird. Dieser Fehler kann auch auf Netzwerkprobleme oder einen ausgelasteten Server hinweisen.
- 801** Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.
- 903** Mit dem angegebenen Gerät war kein Anruf verbunden. Stellen Sie sicher, dass alle IVR-Nebenstellen in der Betriebsversion der Datenbank enthalten sind.
- 914** Der versuchte Vorgang ist fehlgeschlagen.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

**Beispiel**

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

## 4.7 Enqueue

Die Funktion Enqueue fordert den Routing-Server auf, einen Anruf in die Warteschlange einzureihen und den besten verfügbaren Benutzer für die Bearbeitung des Anrufs zu finden. Diese Funktion übermittelt die Routing-Informationen eines Anrufs an den Routing-Server, einschließlich der Warteschlange und der Start-Priorität des Anrufs.

**HINWEIS:** Wenn Sie die Kontaktdaten dem Anruf zuordnen möchten, müssen Sie diese Parameter vor der Einreihung des Anrufs festlegen.

Verwenden Sie diese Funktion, um Anrufe in den Konfigurationen Halten im IVR oder Halten in Warteschlange einzureihen. Nachdem ein Anruf in der Konfiguration Halten im IVR eingereicht ist, muss das IVR-Skript mithilfe der Funktion QueryCallStatus überprüfen, ob der Anruf zugewiesen wurde, und den Anruf dann an die Nebenstelle des Benutzers weiterleiten. Nachdem ein Anruf in der Konfiguration Halten in Warteschlange eingereicht ist, sollte er sofort an eine OpenScape Contact Center ACD/UCD/MLHG-Gruppe weitergeleitet werden.

**HINWEIS:** Sie müssen die Funktionen Initialize und QueryCallInfo aufrufen, bevor Sie diese Funktion aufrufen. In einer Multi-Tenant-Umgebung müssen Sie zuerst die Funktion SetBusinessUnit aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize"](#), auf [Seite 45](#), [Abschnitt 4.15, "QueryCallInfo"](#), auf [Seite 48](#) und [Abschnitt 4.21, "SetBusinessUnit"](#), auf [Seite 62](#).

### Syntax

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

### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die Anruf-ID für den Anruf, den Sie einreihen wollen.

Tabelle 6                      Parameter für die Funktion Enqueue

Name	Typ	Bereich	Beschreibung
IVRHold	Ganzzahl übergeben nach Wert (by value)	0 oder 1	Ein Flag, das angibt, wie der Anruf bearbeitet wird. 0- Gibt an, dass der Anruf für die Konfiguration Halten in Warteschlange an eine ACD/UCD/MLHG-Gruppe weitergeleitet wird. 1- Gibt an, dass der Anruf auf der IVR-Nebenstelle auf die Konfiguration Halten im IVR wartet.
QueueName	Zeichenfolge übergeben nach Wert (by value)	32 Zeichen	Der Name der Warteschlange für den Anruf.
InitialPriority	Ganzzahl übergeben nach Wert (by value)	1 bis 100	Gibt die Priorität eines Anrufs an, wobei 1 die niedrigste und 100 die höchste Priorität ist.
Description (optional)	Zeichenfolge übergeben nach Wert (by value)	100 Zeichen	Eine Zeichenfolge mit einer Kurzbeschreibung des Anrufs, die in der OpenScape Contact Center Client Desktop-Anwendung angezeigt wird, sodass die Benutzer den Anruf identifizieren können.
EstimatedWait	Ganzzahl, übergeben nach Verweis (by reference)	Größer oder gleich 0	Die geschätzte Wartezeit (in Sekunden) für diese Warteschlange.
CallsInQueue	Ganzzahl, übergeben nach Verweis (by reference)	Größer oder gleich 0	Die Anzahl der Anrufe in der angegebenen Warteschlange.
ANIDNIS (optional)	Zeichenfolge übergeben nach Wert (by value)	161	Die ANI- und DNIS-Nummer des Anrufs. Der Statistikserver speichert diese Nummern in der OpenScape Contact Center-Datenbank. Die ANI- und die DNIS-Nummer müssen durch Strichpunkt getrennt werden.

**Tabelle 6** Rückgabecodes Parameter für die Funktion Enqueue

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 304** Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.
- 305** This code indicates an unknown error (for example, user error or the system is unstable).

- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 314** Die Funktion hat einen ungültigen Business Unit-Namen übergeben. Der angegebene Business Unit-Name wurde nicht festgelegt oder entspricht keinem der Business Unit-Namen in der Betriebsversion der Datenbank. (Dieser Fehler ist nur relevant, wenn OpenScape Contact Center als Multi-Tenant-System konfiguriert ist.)
- 701** Die Funktionsanforderung hat die Zeit überschritten, bevor die Funktion abgeschlossen werden konnte. Dies könnte auf Netzwerkprobleme oder einen ausgelasteten Server hinweisen.
- 801** Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.
- 903** Mit dem angegebenen Gerät war kein Anruf verbunden. Stellen Sie sicher, dass alle IVR-Nebenstellen in der Betriebsversion der Datenbank enthalten sind.
- 905** Die Funktion hat eine ungültige Warteschlange übergeben. Die angegebene Warteschlange entspricht keinem der Warteschlangennamen in der Betriebsversion der Datenbank.
- 907** Der T-Server ist nicht verfügbar. Dieser Fehler weist darauf hin, dass der Routing-Server den Anruf nicht in die Warteschlange einreihen konnte, weil der T-Server nicht verfügbar war.
- 908** Der T-Server ist nicht verfügbar.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

#### Beispiel

Im folgenden Beispiel wird angefordert, den Anruf mit der angegebenen Anruf-ID in die Warteschlange einzureihen.

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

## 4.8 EnqueueForAgent

Die Funktion `EnqueueForAgent` fordert den Routing-Server auf, einen Anruf für einen bestimmten Benutzer und eine bestimmte Zeit einzureihen. Wenn `AgentWaitTime` abgelaufen ist, wird der Anruf nicht länger für einen bestimmten Benutzer reserviert und wird in die durch `QueueName` definierte Warteschlange eingereiht. Diese Funktion übermittelt die Routing-Informationen eines Anrufs an den Routing-Server, einschließlich der Warteschlange und der Start-Priorität des Anrufs. Die Funktion übermittelt auch Informationen an den Routing-Server, die in der OpenScape Contact Center Client Desktop-Anwendung erscheinen, zum Beispiel die Beschreibung.

Verwenden Sie diese Funktion, um Anrufe in den Konfigurationen Halten im IVR oder Halten in Warteschlange einzureihen. Nachdem ein Anruf in der Konfiguration Halten im IVR eingereicht ist, muss das IVR-Skript mithilfe der Funktion QueryCallStatus überprüfen, ob der Anruf zugewiesen wurde, und den Anruf dann an die Nebenstelle des Benutzers weiterleiten. Nachdem ein Anruf in der Konfiguration Halten in Warteschlange eingereicht ist, sollte er sofort an eine OpenScape Contact Center ACD/UCD/MLHG-Gruppe weitergeleitet werden.

**HINWEIS:** Sie müssen die Funktionen Initialize und QueryCallInfo aufrufen, bevor Sie diese Funktion aufrufen. In einer Multi-Tenant-Umgebung müssen Sie zuerst die Funktion SetBusinessUnit aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize"](#), auf [Seite 45](#), [Abschnitt 4.15, "QueryCallInfo"](#), auf [Seite 48](#) und [Abschnitt 4.21, "SetBusinessUnit"](#), auf [Seite 62](#).

### Syntax

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

### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die Anruf-ID für den Anruf, den Sie einreihen wollen.
IVRHold	Ganzzahl übergeben nach Wert (by value)	0 oder 1	Ein Flag, das angibt, wie der Anruf bearbeitet wird. 0 - Gibt an, dass der Anruf für die Konfiguration Halten in Warteschlange an eine ACD/UCD/MLHG-Gruppe weitergeleitet wird. 1 - Gibt an, dass der Anruf auf der IVR-Nebenstelle auf die Konfiguration Halten im IVR wartet.
QueueName	Zeichenfolge übergeben nach Wert (by value)	32 Zeichen	Der Name der Warteschlange für den Anruf.
InitialPriority	Ganzzahl übergeben nach Wert (by value)	1 bis 100	Gibt die Priorität eines Anrufs an, wobei 1 die niedrigste und 100 die höchste Priorität ist.

Tabelle 7

Parameter für die Funktion EnqueueForAgent

Name	Typ	Bereich	Beschreibung
Description (optional)	Zeichenfolge übergeben nach Wert (by value)	100 Zeichen	Eine Zeichenfolge mit einer Kurzbeschreibung des Anrufs, die in der OpenScape Contact Center Client Desktop-Anwendung angezeigt wird, sodass die Benutzer den Anruf identifizieren können.
AgentID	Zeichenfolge übergeben nach Wert (by value)	8 Zeichen	Der Benutzer, für den Sie den Anruf reservieren wollen.
AgentWaitTime	Ganzzahl übergeben nach Wert (by value)	Größer als 0	Die maximale Zeit (in Sekunden), die der Anruf auf einen reservierten Benutzer warten kann. Wenn diese Zeit abgelaufen ist, wird der Anruf in die durch QueueName definierte Warteschlange eingereiht.
EstimatedWait	Ganzzahl übergeben nach Wert (by value)	Größer oder gleich 0	Die geschätzte Wartezeit (in Sekunden) für diese Warteschlange.
CallsInQueue	Ganzzahl übergeben nach Wert (by value)	Größer oder gleich 0	Die Anzahl der Anrufe in der angegebenen Warteschlange.
ANIDNIS (optional)	Zeichenfolge übergeben nach Wert (by value)	161 Zeichen	Die ANI- und DNIS-Nummer des Anrufs. Der Statistikserver speichert diese Nummern in der OpenScape Contact Center-Datenbank. Die ANI- und die DNIS-Nummer müssen durch Strichpunkt getrennt werden.

Tabelle 7 Parameter für die Funktion EnqueueForAgent

### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 304** Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.
- 305** This code indicates an unknown error (for example, user error or the system is unstable).
- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 314** Die Funktion hat einen ungültigen Business Unit-Namen übergeben. Der angegebene Business Unit-Name wurde nicht festgelegt oder entspricht keinem der Business Unit-Namen in der Betriebsversion der Datenbank. (Dieser Fehler ist nur relevant, wenn OpenScape Contact Center als Multi-Tenant-System konfiguriert ist.)

- 701** Die Funktionsanforderung hat die Zeit überschritten, bevor die Funktion abgeschlossen werden konnte. Dies könnte auf Netzwerkprobleme oder einen ausgelasteten Server hinweisen.
- 801** Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.
- 903** Mit dem angegebenen Gerät war kein Anruf verbunden. Stellen Sie sicher, dass alle IVR-Nebenstellen in der Betriebsversion der Datenbank enthalten sind.
- 904** Die Funktion hat eine ungültige Agenten-ID übergeben. Die angegebene Agenten-ID stimmt nicht mit der ID eines Agenten in der Betriebsversion der Datenbank überein.
- 905** Die Funktion hat eine ungültige Warteschlange übergeben. Die angegebene Warteschlange entspricht keinem der Warteschlangennamen in der Betriebsversion der Datenbank.
- 907** Der T-Server ist nicht verfügbar. Dieser Fehler weist darauf hin, dass der Routing-Server den Anruf nicht in die Warteschlange einreihen konnte, weil der T-Server nicht verfügbar war.
- 908** Der T-Server ist nicht verfügbar.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

### Beispiel

Die folgende Beispielfunktion fordert dazu auf, den Anruf für Benutzer 790006 für 120 Sekunden einzureihen; anschließend wird er in die Warteschlange "CT\_Pilot1" eingereiht.

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

## 4.9 GetBusinessUnit

---

**HINWEIS:** Die Funktion GetBusinessUnit wird nur in einer Multi-Tenant-Umgebung unterstützt.

---

Die Funktion GetBusinessUnit ruft die Business Unit für den definierten Anruf ab.

---

**HINWEIS:** Sie müssen die Funktionen Initialize, QueryCallInfo und SetBusinessUnit aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen finden Sie unter [Abschnitt 4.13, "Initialize"](#), auf Seite 45, [Abschnitt 4.15, "QueryCallInfo"](#), auf Seite 48, und [Abschnitt 4.21, "SetBusinessUnit"](#), auf Seite 62.

---

#### Syntax

```
GetBusinessUnit (CallID, >BusinessUnitName)
```

#### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die Anruf-ID des Anrufs, für den Sie die Business Unit abrufen wollen.
BusinessUnit Name	Zeichenfolge übergeben nach Verweis (by reference)	32 Zeichen	Der Name der Business Unit für den definierten Anruf.

*Tabelle 8                      Parameter für die Funktion GetBusinessUnit*

#### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0**            Erfolgreich.
- 306**       Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere  
der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder  
hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine  
ganze Zahl übergeben.
- 400**       Niedrige Systemressourcen.
- 914**       Der versuchte Vorgang ist fehlgeschlagen.
- Sons-  
tige**       Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu  
einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

#### Beispiel

Die folgende Beispielfunktion ruft den Namen der Business Unit für die  
angegebene Anruf-ID ab.

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



## 4.10 GetCallTransferable

Die Funktion GetCallTransferable gibt den Wert des durch die Funktion SetCallTransferable festgelegten Flags zurück und bestimmt, ob der Anruf von OpenScape Contact Center an einen zugewiesenen Benutzer weitergeleitet werden kann.

---

**HINWEIS:** Diese Funktion unterstützt keine nicht überwachten IVR-Anrufe.

---



---

**HINWEIS:** Sie müssen die Funktionen Initialize und QueryCallInfo aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize", auf Seite 45](#) und [Abschnitt 4.15, "QueryCallInfo", auf Seite 48](#).

---

### Syntax

GetCallTransferable (CallID, >FlagValue)

### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die Anruf-ID für den Anruf, den Sie weiterleiten wollen.
FlagValue	Ganzzahl übergeben nach Verweis (by reference)	0 oder 1	Ein Flag, das angibt, ob ein Anruf von OpenScape Contact Center an einen zugeordneten Benutzer weitergeleitet werden kann. Wenn das Flag auf falsch (0) gesetzt ist, setzt OpenScape Contact Center die Zuordnung eines Benutzers für den Anruf aus, bis das Flag auf wahr (1) gesetzt ist. Wenn der Anruf vom IVR-System zurück in die Warteschlange geleitet wird, wird das Flag automatisch auf wahr zurückgesetzt.

Tabelle 9 Parameter für die Funktion GetCallTransferable

### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 304** Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.

- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 311** Die Funktion unterstützt keine per Amtsleitung verbundenen (nicht überwachten IVR-) Anrufe.
- 701** Die Funktionsanforderung hat die Zeit überschritten, bevor die Funktion abgeschlossen werden konnte. Dies könnte auf Netzwerkprobleme oder einen ausgelasteten Server hinweisen.
- 801** Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

#### Beispiel

Die folgende Beispielfunktion gibt zurück, ob der Anruf von OpenScape Contact Center an einen zugeordneten Benutzer weitergeleitet werden kann.

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

## 4.11 GetContactData

Die Funktion GetContactData ruft die Kontaktdaten für eine bestimmte Anruf-ID ab. Wenn Sie die Kontaktdaten dem Anruf zuordnen möchten, müssen Sie die Kontaktdaten vor der Einreihung des Anrufs festlegen.

---

**HINWEIS:** Sie müssen die Funktionen Initialize und QueryCallInfo aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize"](#), auf Seite 45 und [Abschnitt 4.15, "QueryCallInfo"](#), auf Seite 48.

---

#### Syntax

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

#### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die Callback-ID für den Anruf, für den Sie die Kontaktdaten abrufen wollen.

Tabelle 10

Parameter für die Funktion GetContactData

Name	Typ	Bereich	Beschreibung
Key	Zeichenfolge übergeben nach Wert (by value)	32 Zeichen	Der Schlüsselname für den Kontaktdatenwert.
Value	Zeichenfolge übergeben nach Verweis (by reference)	128 Zeichen	Der für den Schlüssel zurückgegebene Wert.

Tabelle 10 Parameter für die Funktion GetContactData

### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 304** Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.
- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 307** Der Schlüsselname für den Kontaktdatenwert wurde vor dem Aufrufen dieser Funktion nicht gesetzt.
- 701** Die Funktionsanforderung hat die Zeit überschritten, bevor die Funktion abgeschlossen werden konnte. Dies könnte auf Netzwerkprobleme oder einen ausgelasteten Server hinweisen.
- 801** Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

### Beispiel

Die folgende Beispielfunktion ruft die Kontaktdaten für den Schlüssel "Name" für die angegebene Anruf-ID ab.

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

## 4.12 GetTransitNumber

---

**HINWEIS:** Diese Funktion wird von IVR-Skripten nur für nicht überwachte IVR-Systeme verwendet.

---

Die Funktion GetTransitNumber fragt eine Transitnummer von OpenScape Contact Center ab. Das IVR-System muss den Anruf an die Transitnummer weiterleiten, die von dieser Anforderung empfangen wurde. Die Transitnummer läuft ab, wenn der Anruf nicht innerhalb von 30 Sekunden weitergeleitet wird. Die Transitnummer steht dann wieder für die Verwendung mit einem anderen Anruf zur Verfügung.

---

**HINWEIS:** Bei Halten in Warteschlange wird diese Funktion nach dem Aufruf der Funktion Enqueue aufgerufen. Bei Halten im IVR wird diese Funktion aufgerufen, nachdem der Status "Wartend" oder "Nicht angenommen" in der Funktion QueryCallStatus empfangen wurde.

---

### Syntax

GetTransitNumber(CallID, >TransitNumber)

### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die Anruf-ID für den Anruf, den Sie einreihen wollen.
TransitNumber	Zeichenfolge übergeben nach Verweis (by reference)	80 Zeichen	Eine Pilotnummer, die das IVR-System zum Weiterleiten des Anrufs verwendet.

Tabelle 11      Parameter für die Funktion GetTransitNumber

### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 308** QueryCallInfo wurde vor dem Ausführen der Funktion nicht aufgerufen. Sie müssen QueryCallInfo aufrufen, um die Anruf-ID zu erhalten.
- 310** Ein nicht per Amtsleitung verbundener Anruf hat versucht, diese Methode zu verwenden. Diese Methode kann nur für per Amtsleitung verbundene Anrufe verwendet werden.
- 910** Die Version der IVR-API-DLL entspricht nicht der Version von OpenScape Contact Center.
- 927** Alle angemeldeten Transitnummern sind besetzt.
- 929** Es wurde keine Transitnummern konfiguriert.
- 931** Bei der Überwachung der Anrufart besteht eine Inkonsistenz zwischen der IVR-API und dem T-Server.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

### Beispiel

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

## 4.13 Initialize

Die Funktion Initialize initialisiert die Verbindung zu den OpenScape Contact Center Servern und muss aufgerufen werden, bevor eine der anderen OpenScape Contact Center IVR-API-Funktionen aufgerufen wird. Diese Funktion löscht die ANI- und DNIS-Informationen, die während des vorangegangenen Anrufs auf dieser Nebenstelle eingerichtet wurden. Rufen Sie diese Funktion daher am Anfang Ihres IVR-Skripts auf und stellen Sie sicher, dass OpenScape Contact Center betriebsbereit ist, bevor Sie Anrufe weiterleiten. Rufen Sie die Funktion Initialize nicht mehrere Male für denselben Anruf auf.

### Syntax

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

#### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
LocalPort	Zeichenfolge übergeben nach Wert (by value)	1 – 32767	Die von der IVR-API verwendete TCP/IP-Portnummer. Diese Nummer darf nicht bereits anderweitig verwendet werden.
AdminServer Name	Zeichenfolge übergeben nach Wert (by value)	128 Zeichen	Der Netzwerkname des OpenScape Contact Center-Administrationsservers an dem Standort, auf dem Sie arbeiten, zum Beispiel 6000@servername. Wenn das System für hohe Verfügbarkeit (Warm Standby) konfiguriert ist, müssen Sie den Cluster-Namen angeben.
RpcTimeout	Ganzzahl übergeben nach Wert (by value)	Größer als 0	Die Zeitdauer (in Millisekunden) zwischen dem Aufruf durch die Funktion und der Rückgabe des Aufrufs. Wenn ein Benutzer die Funktion aufruft, erwartet das System ein Ergebnis innerhalb des Zeitüberschreitungslimits; andernfalls schlägt der Aufruf fehl. Verwenden Sie einen Wert von 10000 oder höher.

Tabelle 12 Parameter für die Funktion Initialize

#### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 910** Die Version der IVR-API-DLL entspricht nicht der Version von OpenScape Contact Center.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

#### Beispiel

Die folgende Beispielfunktion initialisiert eine Verbindung zwischen dem OpenScape Contact Center-Server mit dem Namen "deerhurst" und dem IVR-System. Die Nebenstellenummer des IVR-Systems ist 6060. Alle Parameter werden nach Wert übermittelt.

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

## 4.14 QueryAgentStatus

Die Funktion QueryAgentStatus fragt den Status eines Benutzers ab, der mit einem bestimmten Gerät verbunden ist. Diese Funktion gibt den Status des Benutzers als ganzzahligen Wert zurück, abhängig davon, ob der Benutzer angemeldet ist oder nicht. Sie müssen entweder die Agenten-ID oder das Gerät angeben. Wenn die Agenten-ID angegeben und der Agent angemeldet ist, wird das Gerät eingefügt und der Status zurückgegeben. Wenn das Gerät angegeben und ein Benutzer bei dem angegebenen Gerät angemeldet ist, wird die Agenten-ID eingefügt und der Status zurückgegeben.

---

**HINWEIS:** Sie müssen die Funktion Initialize aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13](#), "Initialize", auf Seite 45.

---

### Syntax

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

### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
AgentID	Zeichenfolge übergeben nach Verweis (by reference)	8 Zeichen	Eine Zeichenfolge, die den Benutzer identifiziert; ist leer, wenn das Gerät angegeben wird.
Device	Zeichenfolge übergeben nach Verweis (by reference)	16 Zeichen	Die Agenten-Nebenstelle, die Sie abfragen möchten; ist leer, wenn die Agenten-ID angegeben wird.

Tabelle 13

Parameter für die Funktion QueryAgentStatus

Name	Typ	Bereich	Beschreibung
AgentStatus	Ganzzahl, übergeben nach Verweis (by reference)	Größer als 0	Dieser Wert gibt den Status eines Benutzers an. Gültige Werte lauten wie folgt: 1 - Wählen 2 - Leitung besetzt 3 - Klingeln 4 - Gespräch 5 - Leitung wartend 6 - Halten 7 - Rückfrage 8 - Außer Betrieb 9 - Verfügbar 10 - Nicht verfügbar 11 - Tätig 12 - Abgemeldet 13 - Unbekannt 14 - Wartend 15 - Verarbeitung 16 - Nachkontaktaktivitäten

Tabelle 13 Parameter für die Funktion QueryAgentStatus

### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 904** Die Funktion hat eine ungültige Agenten-ID übergeben. Die angegebene Agenten-ID stimmt nicht mit der ID eines Agenten in der Betriebsversion der Datenbank überein.
- 909** Die Funktion hat ein ungültiges Gerät übergeben. Das angegebene Gerät entspricht keinem der Geräte in der Betriebsversion der Datenbank.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

### Beispiel

Die folgende Beispielfunktion fragt den Status eines Benutzers ab, der mit einem bestimmten Gerät verbunden ist, und gibt den mit dem Gerät verbundenen Benutzerstatus zurück.

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

## 4.15 QueryCallInfo

Die Funktion QueryCallInfo fragt den T-Server nach der Anruf-ID und der ANI- und DNIS-Nummer ab, die mit der aktuellen IVR-Nebenstelle und dem verarbeiteten Anruf verbunden sind. Sowohl die ANI- als auch die DNIS-Nummer



werden als Zeichenfolge zurückgegeben. Rufen Sie diese Funktion auf, wenn ein Anruf an einer IVR-Nebenstelle ankommt, um die ANI- und DNIS-Informationen für den Anruf zu erhalten.

**HINWEIS:** Sie müssen die Funktion Initialize aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13](#), "Initialize", auf Seite 45.

### Syntax

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

### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
Device	Zeichenfolge übergeben nach Wert (by value)	16 Zeichen	Die IVR-Nebenstelle, auf der das IVR-Skript ausgeführt wird.
CallID	Zeichenfolge übergeben nach Verweis (by reference)	18 Zeichen	Die Anruf-ID für den aktuellen Anruf.
originalANI	Zeichenfolge übergeben nach Verweis (by reference)	80 Zeichen	Die mit dem Anruf verbundene ANI-Nummer, wenn er erstmals in das System eintritt.
originalDNIS	Zeichenfolge übergeben nach Verweis (by reference)	80 Zeichen	Die mit dem Anruf verbundene DNIS-Nummer, wenn er erstmals in das System eintritt.
UUID (optional)	Zeichenfolge übergeben nach Verweis (by reference)	32 Zeichen	Die mit dem aktuellen Anruf verbundene UUID.
Visits (optional)	Ganzzahl übergeben nach Verweis (by reference)	Größer oder gleich 0	Gibt an, wie häufig dieser Anruf von einem von OpenScape Contact Center überwachten IVR-System angenommen wurde.

Tabelle 14

Parameter für die Funktion QueryCallInfo

Name	Typ	Bereich	Beschreibung
FMNQ (optional)	Ganzzahl übergeben nach Verweis (by reference)	0 oder 1	Gibt an, ob es sich hier um ein FMNQ-Szenario handelt. Ja=1 Nein=0.
InternalANI (optional)	Zeichenfolge übergeben nach Verweis (by reference)	80 Zeichen	Die Ursprungsnummer des an die IVR-ACD/UCD/MLHG-Gruppe geleiteten Anrufs.
InternalDNIS (optional)	Zeichenfolge übergeben nach Verweis (by reference)	80 Zeichen	Die mit der IVR-ACD/UCD/MLHG-Gruppe verbundene Pilotnummer.
TrunkConnected IVRCall	Zeichenfolge übergeben nach Wert (by value)	Unbegrenzt.	Gibt an, ob der Anruf überwacht oder nicht überwacht ist. Dieser Parameter muss bei einem nicht überwachten IVR-Anruf "1;" lauten. Jeder andere Wert steht für einen überwachten IVR-Anruf.

Tabelle 14 Parameter für die Funktion QueryCallInfo

### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 304** Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.
- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 701** Die Funktionsanforderung hat die Zeit überschritten, bevor die Funktion abgeschlossen werden konnte. Dies könnte auf Netzwerkprobleme oder einen ausgelasteten Server hinweisen.
- 801** Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.

### Beispiel

Die folgende Beispielfunktion fragt die Anruf-ID und die ANI- und DNIS-Nummer des Anrufs ab, der derzeit mit der IVR-Nebenstelle verbunden ist.

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

## 4.16 QueryCallStatus

Die Funktion QueryCallStatus fragt den Status eines Anrufs ab, der mit einer bestimmten Anruf-ID verbunden ist. Das Ziel, an das der Anruf weitergeleitet werden soll, wird von dem Parameter Extension basierend auf dem Status des Anrufs (Wartend oder Nicht angenommen) zurückgegeben. Wenn der Status des Anrufs Getrennt ist, sollte der Anruf vom IVR-System getrennt werden (im Szenario Halten im IVR).

Rufen Sie nach der Einreihung eines Anrufs regelmäßig die Funktion QueryCallStatus auf, um zu ermitteln, ob der Anruf zugewiesen wurde.

---

**HINWEIS:** Wenn das IVR-System nicht überwacht ist, können Sie die Funktion GetTransitNumber aufrufen, um OpenScape Contact Center nach einer Transitnummer abzufragen, an die der Anruf weitergeleitet werden kann. Weitere Informationen finden Sie in [Abschnitt 4.12, "GetTransitNumber", auf Seite 44](#).

---



---

**HINWEIS:** Sie müssen die Funktionen Initialize und QueryCallInfo aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize", auf Seite 45](#) und [Abschnitt 4.15, "QueryCallInfo", auf Seite 48](#).

---

### Syntax

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

### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die von QueryCallInfo zurückgegebene Anruf-ID.
Timeout	Ganzzahl übergeben nach Wert (by value)	Größer oder gleich 0	Die Zeitdauer (in Millisekunden), die auf eine Änderung des Anrufstatus auf dem angegebenen Gerät gewartet wird. Wenn Sie Null (0) angeben, wird der aktuelle Status sofort abgefragt.
Extension	Zeichenfolge übergeben nach Verweis (by reference)	80 Zeichen	Das Ziel, an das der Anruf weitergeleitet werden soll.

Tabelle 15 Parameter für die Funktion QueryCallStatus

Name	Typ	Bereich	Beschreibung
Position InQueue	Ganzzahl, übergeben nach Verweis (by reference)	Größer als 0	Die vom Routing-Server zurückgegebene Position des Anrufs in der Warteschlange.

Tabelle 15 Parameter für die Funktion QueryCallStatus

### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Der Anruf ist frei.
- 1** Der Anruf ist im Zustand Eingereiht. Setzen Sie in diesem Fall die Überprüfung des Anrufstatus fort.
- 2** Der Anruf ist im Zustand Wartend. Leiten Sie in diesem Fall den Anruf an die zurückgegebene Nebenstelle weiter.
- 3** Der Anruf ist im Zustand Nicht angenommen. Leiten Sie in diesem Fall den Anruf an die zurückgegebene Nebenstelle oder eine andere Zeitüberschreitungs-Nebenstelle weiter.
- 4** Es ist ein Fehler aufgetreten. Leiten Sie in diesem Fall den Anruf an eine Nicht-OpenScape Contact Center Nebenstelle weiter.
- 5** Der Anruf muss getrennt werden.
- 6** Der Anruf muss an die zurückgegebene Nebenstelle weitergeleitet werden.
- 304** Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.
- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 308** QueryCallInfo wurde vor dem Ausführen der Funktion nicht aufgerufen. Sie müssen QueryCallInfo aufrufen, um die Anruf-ID zu erhalten.
- 701** Die Funktionsanforderung hat die Zeit überschritten, bevor die Funktion abgeschlossen werden konnte. Dies könnte auf Netzwerkprobleme oder einen ausgelasteten Server hinweisen.
- 801** Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.
- 903** Mit dem angegebenen Gerät war kein Anruf verbunden. Stellen Sie sicher, dass alle IVR-Nebenstellen in der Betriebsversion der Datenbank enthalten sind.
- 908** Der T-Server ist nicht verfügbar.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

---

**HINWEIS:** Während des Aufrufs der Funktion QueryCallStatus kann aus den folgenden beiden Gründen der Rückgabecode -701 empfangen werden:

- Wenn Sie eine Zeitüberschreitung ungleich 0 angeben, wartet die Funktion QueryCallStatus auf die kommenden Ereignisse. In diesem Fall gibt der Rückgabecode -701 nur an, dass Sie die angegebene Zeit gewartet haben, ohne dass etwas passiert ist.
- Wenn Sie als Zeitüberschreitung 0 angegeben haben und der Anruf in einem eingereichten Zustand ist, fragt der Routing-Server die Position in der Warteschlange ab. In diesem Fall gibt der Rückgabecode -701 an, dass Netzwerkprobleme bestehen oder ein Server ausgelastet ist.

---

### Beispiel

Die folgende Beispielfunktion wartet maximal 60.000 Millisekunden bis zur nächsten Überwachung auf Statusänderung. Wenn der Anruf innerhalb dieses Zeitraums in den zugewiesenen Zustand wechselt, gibt die Funktion unter Verwendung der Funktion TRANSFER\_EXT die Benutzer-Nebenstelle zurück, an die das IVR-System den Anruf weiterleiten soll.

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

## 4.17 QueryQueueStatistics

Die Funktion QueryQueueStatistics bestimmt die Anzahl der Anrufe in der Warteschlange, die geschätzte Wartezeit für einen Anruf, die durchschnittliche Wartezeit für die Warteschlange, die Zeitdauer, die der älteste Anruf in der Warteschlange war und den Service-Level für die angegebene Warteschlange.

Diese Berechnungen setzen voraus, dass der Anruf sofort in die Warteschlange des Routing-Servers eingereicht wird, unabhängig davon, ob der Anruf zuvor eingereicht war. Sie müssen QueryQueueStatistics aufrufen, bevor Sie Enqueue aufrufen, um eine korrekte Antwort vom Routing-Server sicherzustellen.

---

**HINWEIS:** Sie müssen die Funktion Initialize aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize"](#), auf Seite 45.

---

### Syntax

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

#### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
QueueName	Zeichenfolge übergeben nach Wert (by value)	32 Zeichen	Die Warteschlange, für die die Abfrage durchgeführt wird.
CallsInQueue	Ganzzahl, übergeben nach Verweis (by reference)	Größer oder gleich 0	Die Anzahl der Anrufe in der angegebenen Warteschlange.
EstimatedWait	Ganzzahl, übergeben nach Verweis (by reference)	Größer oder gleich 0	Die geschätzte Wartezeit (in Sekunden) für die angegebene Warteschlange.
AverageWait	Ganzzahl, übergeben nach Verweis (by reference)	Größer oder gleich 0	Die durchschnittliche Wartezeit (in Sekunden) für die angegebene Warteschlange.
OldestCall InQueue	Ganzzahl, übergeben nach Verweis (by reference)	Größer oder gleich 0	Die Zeitdauer, die sich der älteste Anruf in der Warteschlange befindet.
ServiceLevel	Ganzzahl, übergeben nach Verweis (by reference)	0 – 100	Der Service-Level für die angegebene Warteschlange.
BusinessUnit Name (optional)	Zeichenfolge übergeben nach Wert (by value)	32 Zeichen	Der Name der Business Unit für die angegebene Warteschlange. Dieser Parameter ist in einer Multi-Tenant-Umgebung erforderlich und wird in einer Nicht-Multi-Tenant-Umgebung ignoriert.

Tabelle 16 Parameter für die Funktion QueryQueueStatistics

#### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 304** Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.

- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 314** Die Funktion hat einen ungültigen Business Unit-Namen übergeben. Der angegebene Business Unit-Name wurde nicht festgelegt oder entspricht keinem der Business Unit-Namen in der Betriebsversion der Datenbank. (Dieser Fehler ist nur relevant, wenn OpenScape Contact Center als Multi-Tenant-System konfiguriert ist.)
- 701** Die Funktionsanforderung hat die Zeit überschritten, bevor die Funktion abgeschlossen werden konnte. Dies könnte auf Netzwerkprobleme oder einen ausgelasteten Server hinweisen.
- 801** Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.
- 905** Die Funktion hat eine ungültige Warteschlange übergeben. Die angegebene Warteschlange entspricht keinem der Warteschlangennamen in der Betriebsversion der Datenbank.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

#### **Beispiel**

Die folgende Beispielfunktion fragt die Statistiken für die Warteschlange "Sales" ab.

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

## 4.18 QueryRoutingInfo

Die Funktion QueryRoutingInfo sendet eine Anforderung zum Zuweisen einer Warteschlange und weitere Routing-Informationen. Die Abfrage basiert auf Informationen, die vom IVR-System als Teil einer OpenScape Contact Center-Routinganforderung gesammelt und übermittelt werden.

Das Ziel, an das der Anruf weitergeleitet werden soll, wird vom Parameter Destination zurückgegeben. Wenn der zurückgegebene Code 1 lautet, sollte der Anruf vom IVR-System getrennt werden.

.Wenn in einem Ablauf eine Komponente Transfer angetroffen wird, kann das zurückgegebene Transferziel je nach der Konfiguration der Komponente Transfer im kanonischen Format angezeigt werden. In diesem Fall müssen Sie, wenn Sie den Anruf an dieses Ziel nur mittels Gabelumschalt-Transfer oder einer anderen internen IVR-Transferfunktion weiterleiten möchten, die Nummer entsprechend übersetzen. Wenn Sie den Anruf mithilfe der IVR-API weiterleiten möchten, erfolgt die Analyse direkt durch OpenScape Contact Center.

---

**HINWEIS:** Sie müssen die Funktionen Initialize und QueryCallInfo aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize", auf Seite 45](#) und [Abschnitt 4.15, "QueryCallInfo", auf Seite 48](#).

---

Sie sollten den Anruf basierend auf den Abfrageergebnissen einreihen, trennen oder weiterleiten.

---

**HINWEIS:** Wenn die Kontaktdaten-Werte bei der Routingentscheidung berücksichtigt werden sollen, müssen Sie diese Werte mithilfe der Funktion SetContactData vor dem Aufrufen dieser Funktion festlegen. Um die ursprünglichen Kontaktdaten-Werte beizubehalten, müssen Sie die Werte vor dem Aufrufen der Funktionen Enqueue oder EnqueueForAgent kopieren und zurücksetzen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.24, "SetContactData", auf Seite 67](#).

---

### Syntax

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



**Parameter**

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die Anruf-ID des Anrufs, für den Sie die entsprechenden Routing-Informationen abrufen wollen.
Ziel	Ganzzahl, übergeben nach Verweis (by reference)	80 Zeichen	Das Ziel, an das der Anruf weitergeleitet werden soll.
AgentID	Zeichenfolge übergeben nach Verweis (by reference)	8 Zeichen	Gibt die ID des Benutzers zurück, der für die Bearbeitung des Anrufs reserviert werden soll.  <b>Hinweis:</b> Nur gültig, wenn der Rückgabecode 0 ist.
AgentWaitTime	Zeichenfolge übergeben nach Verweis (by reference)	Größer als 0	Gibt die maximale Zeit (in Sekunden) zurück, die der Anrufer auf einen reservierten Benutzer warten muss, bevor der Anruf freigegeben und in die angegebene Warteschlange eingereiht wird. Dies gilt nur, wenn eine Agenten-ID zurückgegeben wird.  <b>Hinweis:</b> Nur gültig, wenn der Rückgabecode 0 ist.
Beschreibung	Zeichenfolge übergeben nach Verweis (by reference)	100 Zeichen	Enthält eine Beschreibung des Anrufs, die in der OpenScape Contact Center Client Desktop-Anwendung angezeigt wird, sodass Benutzer den Anruf identifizieren können.  <b>Hinweis:</b> Nur gültig, wenn der Rückgabecode 0 ist.
InitialPriority	Ganzzahl, übergeben nach Verweis (by reference)	1 bis 100	Gibt die Priorität des Anrufs gegenüber allen anderen Anrufen in der Warteschlange des Routing-Servers zurück. Dazu gehören auch Anrufe, die mit derselben Warteschlange wie der aktuelle Anruf verbunden sind. Der Standardwert ist 1.  <b>Hinweis:</b> Nur gültig, wenn der Rückgabecode 0 ist.

Tabelle 17

Parameter für die Funktion QueryRoutingInfo

Name	Typ	Bereich	Beschreibung
EstimatedWait	Zeichenfolge übergeben nach Verweis (by reference)	Größer oder gleich 0	Gibt die geschätzte Wartezeit (in Sekunden) für die angegebene Warteschlange zurück.  <b>Hinweis:</b> Nur gültig, wenn der Rückgabecode 0 ist.
CallsInQueue	Ganzzahl, übergeben nach Verweis (by reference)	Größer oder gleich 0	Gibt die Anzahl der Anrufe für die angegebene Warteschlange zurück.  <b>Hinweis:</b> Nur gültig, wenn der Rückgabecode 0 ist.
ANIDNIS (optional)	Zeichenfolge übergeben nach Wert (by value)	161 Zeichen	Die in der Routingentscheidung verwendete ANI- und DNIS-Nummer. Die ANI- und die DNIS-Nummer müssen durch Strichpunkt getrennt werden. Wenn diesem Anruf bereits Kontaktdaten zugeordnet wurden, die ANI- und DNIS-Werte enthalten, haben diese Werte Vorrang vor den hier von Ihnen angegebenen Werten.

Tabelle 17 Parameter für die Funktion QueryRoutingInfo

### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 1** Der Anruf muss getrennt werden.
- 2** Der Anruf muss an die zurückgegebene Nebenstelle weitergeleitet werden. .
- 304** Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.
- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 400** Niedrige Systemressourcen.
- 801** Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.
- 915** Die Funktion konnte die Warteschlange und weitere Routing-Informationen, die auf der im System konfigurierten Routing-Option basieren, nicht abrufen.
- 916** Der Routing-Server hat einen Fehler in der Ablaufausführung festgestellt.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

### Beispiel

Die folgende Beispielfunktion fragt den Routing-Server nach der Warteschlange und anderen Routing-Informationen ab.

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

## 4.19 QuerySystemStatus

Die Funktion QuerySystemStatus bestimmt den Status des Systems. Diese Funktion sollte aufgerufen werden, um am Anfang des IVR-Skripts und vor der Einreihung eines Anrufs den Systemstatus zu überprüfen.

---

**HINWEIS:** Sie müssen die Funktion Initialize aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize"](#), auf Seite 45.

---

### Syntax

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

### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
OverallStatus	Ganzzahl, übergeben nach Verweis (by reference)	0 oder 1	Gibt an, ob OpenScape Contact Center betriebsbereit und in der Lage ist, Anrufe weiterzuleiten. <b>0</b> - Gibt an, dass die Anwendung <i>nicht</i> betriebsbereit und aktiv ist <b>1</b> - Gibt an, dass die Anwendung betriebsbereit und aktiv ist
IndividualServerStates	Zeichenfolge übergeben nach Verweis (by reference)	250 Zeichen	Eine Zeichenfolge, die jeden Servernamen und seinen Zustand enthält. Beispiel: server-name1=1; server-name2=2
Entries	Ganzzahl, übergeben nach Verweis (by reference)	Größer oder gleich 0	Die Anzahl der Server in der Status-Zeichenfolge.

Tabelle 18 Parameter für die Funktion QuerySystemStatus

Das IVR-Skript muss den Status der einzelnen Server analysieren, um den Status des Routing-Servers und des T-Servers zu ermitteln. Wenn der Status des Routing-Servers nicht 1 oder der Status des T-Servers nicht 8 ist, sollte der Anruf an eine ACD/UCD/MLHG-Nummer transferiert werden, die Anrufe an Ausfall-ACD/UCD/MLHG-Gruppen weiterleitet.

#### Statuswerte

Die folgenden Codes geben den Status der verschiedenen Server an:

- 0** Der Server ist nicht betriebsbereit.
- 1** Der Server ist betriebsbereit.
- 2** Der Server ist inaktiv. Dies bedeutet, dass der Administrationsserver betriebsbereit ist, aber keine Statusinformationen für den betreffenden Server erhalten hat.

Nur der T-Server verwendet die folgenden Codes:

- 3** Der T-Server ist nicht betriebsbereit.
- 4** Der T-Server wird gerade beendet.
- 5** Der T-Server ist außer Betrieb, weil er keine Verbindung zum CSTA-Provider herstellen kann.
- 6** Der T-Server wird gerade hochgefahren.
- 7** Der T-Server wird gerade initialisiert.
- 8** Der T-Server ist betriebsbereit.
- 9** Die Kommunikationsplattform ist überlastet und hat Probleme, die Anfragen des T-Servers auszuführen.
- 10** Der T-Server wird gerade wiederhergestellt.

#### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 304** Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.
- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 701** Die Funktionsanforderung hat die Zeit überschritten, bevor die Funktion abgeschlossen werden konnte. Dies könnte auf Netzwerkprobleme oder einen ausgelasteten Server hinweisen.
- 801** Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.
- 901** Es waren keine Statusinformationen verfügbar. Der Administrationsserver ist nicht betriebsbereit, aber der Routing-Server ist noch verfügbar. In diesem Fall ist der Status der anderen Server unbekannt. Dieser Fehler wird nur von der Funktion QueryCallStatus zurückgegeben.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

**Beispiel**

Die folgende Beispielfunktion fragt den Status jedes einzelnen Servers ab. Die Funktion übermittelt S\_OVERALLSTATUS und S\_SERVER\_STATES nach Referenz, um den Status jedes einzelnen Servers zurückzugeben. Außerdem gibt I\_ENTRIES die Anzahl der Server in der Zeichenfolge S\_SERVER\_STATES zurück.

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

## 4.20 ReleaseTransitNumber

---

**HINWEIS:** Diese Funktion wird von IVR-Skripten nur für nicht überwachte IVR-Systeme verwendet.

---

Über die Funktion ReleaseTransitNumber kann eine Anwendung anfordern, dass OpenScape Contact Center eine Transitnummer für die Verwendung durch einen anderen Anruf freigibt. Diese Funktion muss aufgerufen werden, bevor die Transitnummer abläuft. Dies bedeutet, dass die Transitnummer für einen anderen OpenScape Contact Center-Anruf verfügbar wird.

Das IVR-System kann die Funktion ReleaseTransitNumber verwenden, um den T-Server zu informieren, dass die Transitnummer nicht länger benötigt wird. Die Anwendung muss die Funktion GetTransitNumber erneut aufrufen, um eine neue Transitnummer anzufordern, bevor versucht wird, den Anruf an OpenScape Contact Center weiterzuleiten.

**Syntax**

```
ReleaseTransitNumber (CallID)
```

**Parameter**

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die Anruf-ID für den aktuellen Anruf.

Tabelle 19 Parameter für die Funktion ReleaseTransitNumber

**Rückgabecodes**

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 308** QueryCallInfo wurde vor dem Ausführen der Funktion nicht aufgerufen. Sie müssen QueryCallInfo aufrufen, um die Anruf-ID zu erhalten.

- 310** Ein nicht per Amtsleitung verbundener Anruf hat versucht, diese Methode zu verwenden. Diese Methode kann nur für per Amtsleitung verbundene Anrufe verwendet werden.
- 910** Die Version der IVR-API-DLL entspricht nicht der Version von OpenScape Contact Center.
- 928** Dieser CallID wurde keine Transitnummer zugewiesen.
- 931** Bei der Überwachung der Anrufart besteht eine Inkonsistenz zwischen der IVR-API und dem T-Server.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

#### Beispiel

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

## 4.21 SetBusinessUnit

---

**HINWEIS:** Die Funktion SetBusinessUnit wird nur in einer Multi-Tenant-Umgebung unterstützt.

---

Die Funktion SetBusinessUnit ruft die Business Unit für den definierten Anruf ab.

---

**HINWEIS:** Die Funktion SetBusinessUnit kann pro IVR-Anruf nur einmal aufgerufen werden.

---

---

**HINWEIS:** Sie müssen die Funktionen Initialize und QueryCallInfo aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize"](#), auf Seite 45 und [Abschnitt 4.15, "QueryCallInfo"](#), auf Seite 48.

---

#### Syntax

```
SetBusinessUnit (CallID, BusinessUnitName)
```

**Parameter**

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die Anruf-ID für den Anruf, für den Sie die Business Unit festlegen wollen.
BusinessUnit Name	Zeichenfolge übergeben nach Wert (by value)	32 Zeichen	Der Name der Business Unit für den definierten Anruf.

Tabelle 20

Parameter für die Funktion SetBusinessUnit

**Rückgabecodes**

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 304** Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.
- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 308** QueryCallInfo wurde vor dem Ausführen der Funktion nicht aufgerufen. Sie müssen QueryCallInfo aufrufen, um die Anruf-ID zu erhalten.
- 313** Die Funktion SetBusinessUnit wurde für diesen Anruf bereits aufgerufen. (Dieser Fehler ist nur relevant, wenn OpenScape Contact Center als Multi-Tenant-System konfiguriert ist.)
- 314** Die Funktion hat einen ungültigen Business Unit-Namen übergeben. Der angegebene Business Unit-Name wurde nicht festgelegt oder entspricht keinem der Business Unit-Namen in der Betriebsversion der Datenbank. (Dieser Fehler ist nur relevant, wenn OpenScape Contact Center als Multi-Tenant-System konfiguriert ist.)
- 400** Niedrige Systemressourcen.
- 801** Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.
- 910** Die Version der IVR-API-DLL entspricht nicht der Version von OpenScape Contact Center.
- 955** Multi-Tenancy ist bei OpenScape Contact Center kein lizenziertes Leistungsmerkmal.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

### Beispiel

Die folgende Beispielfunktion stellt den Namen der Business Unit für den definierten Anruf ein.

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

## 4.22 SetCallInfo

---

**HINWEIS:** Diese Funktion wird von IVR-Skripten nur für nicht überwachte IVR-Systeme verwendet.

---

Die Funktion SetCallInfo speichert ANI- und DNIS-Informationen für unüberwachte IVR-Anrufe, die an OpenScape Contact Center weitergeleitet werden. Diese API kann nur vor QueryCallInfo und nach Initialize aufgerufen werden.

### Syntax

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

### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
ANI	Zeichenfolge übergeben nach Wert (by value)	80 Zeichen	Die ANI-Nummer des Anrufs.
DNIS	Zeichenfolge übergeben nach Wert (by value)	80 Zeichen	Die DNIS-Nummer des Anrufs.

Tabelle 21 Parameter für die Funktion SetCallInfo

### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 304** Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.
- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.



–312 Falsche Abfolge von Operationen, zum Beispiel SetCallInfo nach QueryCallInfo().

**Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

#### Beispiel

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

## 4.23 SetCallTransferable

---

**HINWEIS:** Bevor das IVR-Skript einen FMNQ-Anruf trennt oder weiterleitet, müssen Sie die Funktion SetCallTransferable (0) aufrufen und einen erfolgreichen Rückgabecode empfangen. Wenn Sie keinen erfolgreichen Rückgabecode empfangen, zeigt dies an, dass der Anruf gerade weitergeleitet wird. Sie müssen warten, bis der Prozess abgeschlossen ist und dann die Funktion SetCallTransferable (0) erneut aufrufen.

---

Die Funktion SetCallTransferable gibt an, ob der Anruf von OpenScape Contact Center an einen zugeordneten Benutzer weitergeleitet werden kann.

---

**HINWEIS:** Diese Funktion unterstützt keine nicht überwachten IVR-Anrufe.

---

**HINWEIS:** Sie müssen die Funktionen Initialize und QueryCallInfo aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize", auf Seite 45](#) und [Abschnitt 4.15, "QueryCallInfo", auf Seite 48](#).

---

#### Syntax

```
SetCallTransferable (CallID, FlagValue)
```

#### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die Anruf-ID des Anrufs, den Sie weiterleiten oder trennen möchten.

Tabelle 22

Parameter für die Funktion SetCallTransferable

## Verwenden der OpenScape Contact Center IVR-API-Funktionen

### SetCallTransferable

Name	Typ	Bereich	Beschreibung
FlagValue	Ganzzahl übergeben nach Wert (by value)	0 oder 1	Ein Flag, das angibt, ob ein Anruf von OpenScape Contact Center an einen zugeordneten Benutzer weitergeleitet werden kann. Wenn das Flag auf falsch (0) gesetzt ist, setzt OpenScape Contact Center die Zuordnung eines Benutzers für den Anruf aus, bis das Flag auf wahr (1) gesetzt ist. Wenn der Anruf vom IVR-System zurück in die Warteschlange geleitet wird, wird das Flag automatisch auf wahr (1) zurückgesetzt.

Tabelle 22

Parameter für die Funktion SetCallTransferable

### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 304** Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.
- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 311** Die Funktion unterstützt keine per Amtsleitung verbundenen (nicht überwachten IVR-) Anrufe.
- 701** Die Funktionsanforderung hat die Zeit überschritten, bevor die Funktion abgeschlossen werden konnte. Dies könnte auf Netzwerkprobleme oder einen ausgelasteten Server hinweisen.
- 801** Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

### Beispiel

Die folgende Beispielfunktion gibt an, dass der Anruf von OpenScape Contact Center an einen zugeordneten Benutzer weitergeleitet werden kann.

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

## 4.24 SetContactData

Die Funktion SetContactData legt die Kontaktdaten für eine angegebene Anruf-ID fest und fügt das Schlüssel/Wert-Paar hinzu, wenn es nicht existiert. Wenn die Kontaktdaten für den Anruf verfügbar sein sollen, müssen Sie die Kontaktdaten vor der Einreichung des Anrufs festlegen.

---

**HINWEIS:** Sie müssen die Funktionen Initialize und QueryCallInfo aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize", auf Seite 45](#) und [Abschnitt 4.15, "QueryCallInfo", auf Seite 48](#).

---

### Syntax

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

### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die Anruf-ID für den Anruf, für den Sie die Kontaktdaten festlegen wollen.
Key	Zeichenfolge übergeben nach Wert (by value)	32 Zeichen	Der Schlüsselname für den Kontaktdatenwert.
Value	Zeichenfolge übergeben nach Wert (by value)	128 Zeichen	Der zu setzende Wert.

Tabelle 23 Parameter für die Funktion SetContactData

### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 304** Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.
- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 308** QueryCallInfo wurde vor dem Ausführen der Funktion nicht aufgerufen. Sie müssen QueryCallInfo aufrufen, um die Anruf-ID zu erhalten.

- 701** Die Funktionsanforderung hat die Zeit überschritten, bevor die Funktion abgeschlossen werden konnte. Dies könnte auf Netzwerkprobleme oder einen ausgelasteten Server hinweisen.
- 801** Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

#### Beispiel

Die folgende Beispielfunktion legt die Kontaktdaten für eine angegebene Anruf-ID fest und fügt das Schlüssel-Wert-Paar hinzu, wenn es nicht existiert.

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

## 4.25 SetDisplay

Die Funktion SetDisplay stellt das Telefon-Display für den ersten annehmenden Benutzer des angegebenen Anrufs ein.

---

**HINWEIS:** Diese Funktion ist nur auf der HiPath 4000-Kommunikationsplattform verfügbar. Sie müssen die Funktionen Initialize und QueryCallInfo aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize", auf Seite 45](#) und [Abschnitt 4.15, "QueryCallInfo", auf Seite 48](#).

---

#### Syntax

```
SetDisplay (CallID, DisplayInfo)
```

#### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die Anruf-ID des Anrufs, für den Sie das Display einstellen wollen.
DisplayInfo	Zeichenfolge übergeben nach Wert (by value)	240 Zeichen	Die Zeichenfolge, die auf dem Gerät des annehmenden Benutzers angezeigt wird.

Tabelle 24

Parameter für die Funktion SetDisplay

### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0**      Erfolgreich.
- 304**   Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.
- 306**   Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 701**   Die Funktionsanforderung hat die Zeit überschritten, bevor die Funktion abgeschlossen werden konnte. Dies könnte auf Netzwerkprobleme oder einen ausgelasteten Server hinweisen.
- 801**   Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.
- Sonstige**   Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

### Beispiel

Die folgende Beispielfunktion stellt das Telefon-Display für den ersten annehmenden Benutzer des angegebenen Anrufs auf "IVR-Anruf" ein.

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

## 4.26 Transfer

Die Funktion Transfer führt eine Weiterleitung durch und gibt den Erfolg oder Misserfolg der Weiterleitung an. Diese API wartet nicht darauf, dass der Zerteilnehmer antwortet.

---

**HINWEIS:** Diese Funktion unterstützt keine nicht überwachten IVR-Anrufe.

---

---

**HINWEIS:** Sie können diese Methode anstelle der Weiterleitung per IVR-Gabelumschalt-Flash verwenden, weil diese Funktion schneller funktioniert. Sie müssen die Funktionen Initialize und QueryCallInfo aufrufen, bevor Sie diese Funktion aufrufen. Ausführliche Informationen erhalten Sie unter [Abschnitt 4.13, "Initialize"](#), auf Seite 45 und [Abschnitt 4.15, "QueryCallInfo"](#), auf Seite 48.

---

### Syntax

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

#### Parameter

Sie müssen die Parameter in derselben Reihenfolge wie in der folgenden Tabelle eingeben.

Name	Typ	Bereich	Beschreibung
CallID	Zeichenfolge übergeben nach Wert (by value)	18 Zeichen	Die Anruf-ID für den Anruf, den Sie weiterleiten wollen.
ToDevice	Zeichenfolge übergeben nach Wert (by value)	80 Zeichen	Eine Zeichenfolge, die das Gerät identifiziert, an das der Anruf weitergeleitet wird.

Tabelle 25

Parameter für die Funktion Transfer

#### Rückgabecodes

Die gängigsten Rückgabecodes dieser Funktion sind:

- 0** Erfolgreich.
- 304** Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.
- 306** Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
- 311** Die Funktion unterstützt keine per Amtsleitung verbundenen (nicht überwachten IVR-) Anrufe.
- 701** Die Funktionsanforderung hat die Zeit überschritten, bevor die Funktion abgeschlossen werden konnte. Dies tritt normalerweise nur auf, wenn der Status eines Anrufs abgefragt wird. Dieser Fehler kann auch auf Netzwerkprobleme oder einen ausgelasteten Server hinweisen.
- 801** Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.
- 903** Mit dem angegebenen Gerät war kein Anruf verbunden. Stellen Sie sicher, dass alle IVR-Nebenstellen in der Betriebsversion der Datenbank enthalten sind.
- 909** Die Funktion hat ein ungültiges Gerät übergeben. Das angegebene Gerät entspricht keinem der Geräte in der Betriebsversion der Datenbank.
- 913** Die Funktion hat einen ungültigen Parameter "An Gerät" übergeben.
- 914** Der versuchte Vorgang ist fehlgeschlagen.
- 917** API-Umsetzungsfehler.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Rückgabecodes"](#).

#### Beispiel

Die folgende Beispielfunktion leitet den Anruf mit der angegebenen Anruf-ID an Nebenstelle 5678 weiter.

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

#### Format der Telefonnummer

Der Parameter Telefonnummer muss wie folgt im kanonischen Format eingegeben werden:

+ [Länderkennzahl] Leerzeichen [(Vorwahl) Leerzeichen] Teilnehmernummer [--Nebenstelle]

Die Telefonnummer kann auch eine wählbare Adresse sein, die von einer Kommunikationsplattform oder als Ergebnis des Aufrufs einer TAPI-Funktion abgerufen wird. Die folgenden Zeichenfolgen sind korrekte Telefonnummern:

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





## 5 Rückgabecodes

Dieses Kapitel beschreibt alle von den IVR-Funktionen zurückgegebenen Codes. Weitere Informationen dazu, wie diese Codes mit den speziellen Funktionen zusammenhängen, finden Sie unter [Kapitel 4, "Verwenden der OpenScape Contact Center IVR-API-Funktionen"](#).

Code	Beschreibung
<b>0</b>	Erfolgreich.
<b>1</b>	QueryCallStatus – Gibt an, dass der Anruf im Zustand Eingereiht ist. Setzen Sie in diesem Fall die Überprüfung des Anrufstatus fort.  QueryRoutingInfo – Gibt an, dass der Anruf getrennt werden muss.
<b>2</b>	QueryCallStatus – Gibt an, dass der Anruf im Zustand Wartend ist. Leiten Sie in diesem Fall den Anruf an die zurückgegebene Nebenstelle weiter. QueryRoutingInfo – Gibt an, dass der Anruf an die zurückgegebene Nebenstelle weitergeleitet werden muss.
<b>3</b>	QueryCallStatus – Gibt an, dass der Anruf im Zustand Nicht angenommen ist. Leiten Sie in diesem Fall den Anruf an die zurückgegebene Nebenstelle oder eine andere Zeitüberschreitungs-Nebenstelle weiter.
<b>4</b>	QueryCallStatus – Gibt an, dass ein Fehler aufgetreten ist. Leiten Sie in diesem Fall den Anruf an eine Nicht-OpenScape Contact Center Nebenstelle weiter.
<b>5</b>	QueryCallStatus – Gibt an, dass der Anruf getrennt werden muss.
<b>6</b>	QueryCallStatus – Der Anruf muss an die zurückgegebene Nebenstelle weitergeleitet werden.
<b>–304</b>	Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.
<b>–305</b>	This code indicates an unknown error (for example, user error or the system is unstable).
<b>–306</b>	Der Funktionsaufruf enthielt einen ungültigen Parameter. Einer oder mehrere der durch den Wert übergebenen Parameter wiesen den falschen Typ auf oder hatten einen falschen Wert. Beispielsweise wurde statt einer Zeichenfolge eine ganze Zahl übergeben.
<b>–307</b>	Der Schlüsselname für den Kontaktdatenwert wurde vor dem Aufrufen dieser Funktion nicht gesetzt.
<b>–308</b>	QueryCallInfo wurde vor dem Ausführen der Funktion nicht aufgerufen. Sie müssen QueryCallInfo aufrufen, um die Anruf-ID zu erhalten.
<b>–310</b>	Ein nicht per Amtsleitung verbundener Anruf hat versucht, diese Methode zu verwenden. Diese Methode kann nur für per Amtsleitung verbundene Anrufe verwendet werden.
<b>–311</b>	Die Funktion unterstützt keine per Amtsleitung verbundenen (nicht überwachten IVR-) Anrufe.
<b>–312</b>	Falsche Abfolge von Operationen, zum Beispiel SetCallInfo nach QueryCallInfo().

Tabelle 26

Rückgabecodes

Code	Beschreibung
–313	Die Funktion SetBusinessUnit wurde für diesen Anruf bereits aufgerufen. (Dieser Fehler ist nur relevant, wenn OpenScape Contact Center als Multi-Tenant-System konfiguriert ist.)
–314	Die Funktion hat einen ungültigen Business Unit-Namen übergeben. Der angegebene Business Unit-Name wurde nicht festgelegt oder entspricht keinem der Business Unit-Namen in der Betriebsversion der Datenbank. (Dieser Fehler ist nur relevant, wenn OpenScape Contact Center als Multi-Tenant-System konfiguriert ist.)
–400	Niedrige Systemressourcen.
–701	Die Funktionsanforderung hat die Zeit überschritten, bevor die Funktion abgeschlossen werden konnte. Dies könnte auf Netzwerkprobleme oder einen ausgelasteten Server hinweisen.
–801	Die Funktion konnte keine Verbindung zu einem der OpenScape Contact Center Server herstellen. Entweder ist der Servername ungültig, oder der Server ist nicht betriebsbereit.
–901	Es waren keine Statusinformationen verfügbar. Der Administrationsserver ist nicht betriebsbereit, aber der Routing-Server ist noch verfügbar. In diesem Fall ist der Status der anderen Server unbekannt. Dieser Fehler wird nur von der Funktion QueryCallStatus zurückgegeben.
–903	Mit dem angegebenen Gerät war kein Anruf verbunden. Stellen Sie sicher, dass alle IVR-Nebenstellen in der Betriebsversion der Datenbank enthalten sind.
–904	Die Funktion hat eine ungültige Agenten-ID übergeben. Die angegebene Agenten-ID stimmt nicht mit der ID eines Agenten in der Betriebsversion der Datenbank überein.
–905	Die Funktion hat eine ungültige Warteschlange übergeben. Die angegebene Warteschlange entspricht keinem der Warteschlangennamen in der Betriebsversion der Datenbank.
–907	Der T-Server ist nicht verfügbar. Dieser Fehler weist darauf hin, dass der Routing-Server den Anruf nicht in die Warteschlange einreihen konnte, weil der T-Server nicht verfügbar war.
–908	Der T-Server ist nicht verfügbar.
–909	Die Funktion hat ein ungültiges Gerät übergeben. Das angegebene Gerät entspricht keinem der Geräte in der Betriebsversion der Datenbank.
–910	Die Version der IVR-API-DLL entspricht nicht der Version von OpenScape Contact Center.
–911	Die Funktion enthielt Daten, die aus einem unbekannten Grund beschädigt wurden. Die Funktion konnte daher keine Verbindung zum Routing-Server herstellen. Die Funktion ist fehlgeschlagen.
–913	Die Funktion hat einen ungültigen Parameter "An Gerät" übergeben.
–914	Der versuchte Vorgang ist fehlgeschlagen.
–915	Die Funktion konnte die Warteschlange und weitere Routing-Informationen, die auf der im System konfigurierten Routing-Option basieren, nicht abrufen.
–916	Der Routing-Server hat einen Fehler in der Ablaufausführung festgestellt.
–917	API-Umsetzungsfehler.

Tabelle 26

Rückgabecodes

Code	Beschreibung
–927	Alle angemeldeten Transitnummern sind besetzt.
–928	Dieser CallID wurde keine Transitnummer zugewiesen.
–929	Es wurde keine Transitnummern konfiguriert.
–931	Bei der Überwachung der Anrufart besteht eine Inkonsistenz zwischen der IVR-API und dem T-Server.
–955	Multi-Tenancy ist bei OpenScape Contact Center kein lizenziertes Leistungsmerkmal.
–1006	Sie sind dabei, einen doppelten Callback einzurichten.
–1021	Falsche CallbackID.
–1028	Eine Zeit des Zeitplans ist ungültig.
–1029	Die Callback-Einsatzpläne überschneiden sich nicht mit den Contact Center-Betriebszeiten.
–1031	Eine Telefonnummer in der Anforderung ist ausgeschlossen.
–1033	Callback kann keine Kontaktdaten akzeptieren, die länger als 1000 Byte sind.
–1040	Alle Zeitpläne sind bereits abgelaufen.
–1045	Der Name der Callback-Warteschlange ist ungültig.
–1047	Maximales Limit der vorausgeplanten Tage.
<b>Sonstige</b>	<p>Alle sonstigen negativen Codes weisen auf einen Fehler hin. Wenn ein anderer Code zurückgegeben wird:</p> <ul style="list-style-type: none"> <li>• Stellen Sie sicher, dass alle Standort-, Warteschlangen- und Benutzernamen (oder ID-Nummern) in dem IVR-Skript mit den Namen (oder ID-Nummern) in der Betriebsversion der OpenScape Contact Center-Datenbank übereinstimmen.</li> <li>• Stellen Sie sicher, dass alle Ressourcen der Kommunikationsplattform korrekt in die Betriebsversion der OpenScape Contact Center-Datenbank eingegeben wurden.</li> <li>• Wenden Sie sich an die nächste Support-Stelle.</li> </ul>

Tabelle 26

Rückgabecodes



# Stichwortverzeichnis

## D

### Dokumentation

- Feedback geben 6
- Formatierungskonventionen 5
- Zielgruppe 5

## F

### FMNQ

- IVR-Skripte schreiben 22
- Konfiguration 17

- Funktion CallerDisconnected 26
- Funktion CreateCallback 27
- Funktion DeleteCallback 30
- Funktion Dequeue 30
- Funktion Disconnect 32
- Funktion Enqueue 34
- Funktion EnqueueForAgent 36
- Funktion GetBusinessUnit 39
- Funktion GetCallTransferable 41
- Funktion GetContactData 42
- Funktion GetTransitNumber 44
- Funktion Initialize 45
- Funktion QueryAgentStatus 47
- Funktion QueryCallInfo 48
- Funktion QueryCallStatus 51
- Funktion QueryQueueStatistics 53
- Funktion QueryRoutingInfo 56
- Funktion QuerySystemStatus 59
- Funktion ReleaseTransitNumber 61
- Funktion SetBusinessUnit 62
- Funktion SetCallInfo 64
- Funktion SetCallTransferable 65
- Funktion SetContactData 67
- Funktion SetDisplay 68
- Funktion Transfer 69

## H

### Halten im IVR

- IVR-Skripte schreiben 18
- Konfiguration 13

### Halten in Warteschlange

- IVR-Skripte schreiben 20
- Konfiguration 15

## I

### Installation 9

### IVR-API-Funktionen

- CallerDisconnected 26
- CreateCallback 27
- DeleteCallback 30
- Dequeue 30
- Disconnect 32
- Enqueue 34
- EnqueueForAgent 36
- GetBusinessUnit 39
- GetCallTransferable 41
- GetContactData 42
- GetTransitNumber 44
- Initialize 45
- QueryAgentStatus 47
- QueryCallInfo 48
- QueryCallStatus 51
- QueryQueueStatistics 53
- QueryRoutingInfo 56
- QuerySystemStatus 59
- ReleaseTransitNumber 61
- SetBusinessUnit 62
- SetCallInfo 64
- SetCallTransferable 65
- SetContactData 67
- SetDisplay 68
- Transfer 69

### IVR-Skript

- Übermitteln von Parametern 25

### IVR-Skripte

- FMNQ 22
- Halten im IVR 18
- Halten in Warteschlange 20
- schreiben 17
- Übersicht 7

### IVR-Systeme

- Interaktion mit OpenScape Contact Center 8
- nicht überwacht 9

## M

### Multi-Tenant-Umgebung 24

- Funktion GetBusinessUnit 39
- Funktion SetBusinessUnit 62
- Subdialog GetBusinessUnit 39
- Subdialog SetBusinessUnit 62

## N

### nicht überwachte IVR-Systeme 9

## R

### Rückgabecodes 73

## S

- Subdialog CreateCallback 27
- Subdialog DeleteCallback 30

## Stichwortverzeichnis

- Subdialog Dequeue 30
- Subdialog Enqueue 34
- Subdialog GetBusinessUnit 39
- Subdialog GetContactData 42
- Subdialog GetTransitNumber 44
- Subdialog Initialize 45
- Subdialog QueryAgentStatus 47
- Subdialog QueryCallStatus 51
- Subdialog QueryQueueStatistics 53
- Subdialog QueryRoutingInfo 56
- Subdialog QuerySystemStatus 59
- Subdialog ReleaseTransitNumber 61
- Subdialog SetBusinessUnit 62
- Subdialog SetContactData 67
- Subdialog SetDisplay 68

## U

- Übermitteln von Parametern
  - nach Referenz 25
  - nach Wert 25
- Übersicht 7

## V

- VoiceXML-Subdialoge
  - CreateCallback 27
  - DeleteCallback 30
  - Dequeue 30
  - Enqueue 34
  - GetBusinessUnit 39
  - GetContactData 42
  - GetTransitNumber 44
  - Initialize 45
  - QueryAgentStatus 47
  - QueryCallStatus 51
  - QueryQueueStatistics 53
  - QueryRoutingInfo 56
  - QuerySystemStatus 59
  - ReleaseTransitNumber 61
  - SetBusinessUnit 62
  - SetContactData 67
  - SetDisplay 68