



A MITEL
PRODUCT
GUIDE

Unify OpenScape Contact Center

Unify OpenScape Contact Center Enterprise V11 R1,
VoiceXML Integrationshandbuch

Integrationshandbuch

09/2024

Notices

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

Trademarks

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

© Copyright 2024, Mitel Networks Corporation

All rights reserved

Inhalt

1 Informationen zu diesem Handbuch	5
1.1 An wen richtet sich dieses Handbuch?	5
1.2 Formatierungskonventionen	5
1.3 Feedback zur Dokumentation	6
2 Info über die OpenScape Contact Center VoiceXML-Integration	7
2.1 Installieren der OpenScape Contact Center VoiceXML-Integration	7
2.2 Vorteile der VoiceXML-Integration	8
3 Konfigurieren der OpenScape Contact Center VoiceXML-Integration	9
3.1 Konfiguration von Halten im IVR	9
3.2 Konfiguration von Halten in Warteschlange	11
3.3 Schreiben eines IVR-Skripts	13
3.4 VoiceXML-Subdialoge für IVR-Systeme	13
3.4.1 Schreiben eines IVR-Skripts für die Konfiguration Halten im IVR	15
3.4.2 Schreiben eines IVR-Skripts für die Konfiguration Halten in Warteschlange	19
3.5 Verwenden eines IVR-Systems in einer Multi-Tenant-Umgebung	21
4 Verwenden der OpenScape Contact Center VoiceXML-Subdialoge	23
4.1 CreateCallback	23
4.2 DeleteCallback	27
4.3 Dequeue	29
4.4 Enqueue	31
4.5 GetBusinessUnit	34
4.6 GetContactData	36
4.7 GetTransitNumber	38
4.8 Initialize	40
4.9 QueryAgentStatus	42
4.10 QueryCallStatus	45
4.11 QueryQueueStatistics	48
4.12 QueryRoutingInfo	52
4.13 QuerySystemStatus	56
4.14 ReleaseTransitNumber	59
4.15 SetBusinessUnit	61
4.16 SetContactData	63
4.17 SetDisplay	65
4.18 Terminate	67
5 Fehlercodes	69

1 Informationen zu diesem Handbuch

Dieses Handbuch beschreibt die Integration von OpenScape Contact Center mit einem Interactive Voice Response (IVR)-System über die OpenScape Contact Center VoiceXML-Schnittstelle.

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:** VoiceXML Integrationshandbuch
- **Sachnummer:** A31003-S22A-N107-01-0020

2 Info über die OpenScape Contact Center VoiceXML-Integration

VoiceXML (Voice Extensible Markup Language) ist eine offene Schnittstelle zur Integration von IVR-Systemen in OpenScape Contact Center. VoiceXML ist ein Protokollstandard für die Erstellung von Sprache-Benutzerschnittstellen, die auf MFV-Töne oder gesprochene Befehle (Benutzerseite) und aufgezeichnete Nachrichten oder aus Text synthetisierte Sprache (Anwendungsseite) reagieren.

VoiceXML dient zur Erstellung von Audio-Dialogen, deren Funktionsumfang synthetische Sprache, digitalisiertes Audio, Erkennung von Sprach- und MFV-Tasteneingaben, Erfassung gesprochener Eingaben und Telefonie umfasst. Sein wichtigstes Ziel ist es, IVR-Anwendungen um die Vorteile webbasierter Entwicklung und Inhaltsübermittlung zu erweitern.

Die OpenScape Contact Center VoiceXML-Integration wird bei Verbindung zu den folgenden Kommunikationsplattformen unterstützt:

- OpenScape Voice
- OpenScape 4000 oder HiPath 4000

2.1 Installieren der OpenScape Contact Center VoiceXML-Integration

Die OpenScape Contact Center VoiceXML-Integration wird mit dem Webkomponenten-Paket verteilt, das auf der OpenScape Contact Center-DVD enthalten ist.

Eine ausführliche Anleitung zum Konfigurieren und Testen der Webkomponenten-Dateien auf dem Unternehmens-Webserver-Computer zur Unterstützung der OpenScape Contact Center VoiceXML-Integration finden Sie im *Systemverwaltungshandbuch*.

2.2 Vorteile der VoiceXML-Integration

Die Integration von VoiceXML in OpenScape Contact Center bietet eine VoiceXML-Schnittstelle als Alternative zur vorhandenen proprietären IVR-API, um die Integration von IVR-Systemen von Drittanbietern in OpenScape Contact Center zu vereinfachen. Beispielsweise kann ein IVR-System dazu genutzt werden, Anrufern Self-Service-Funktionen anzubieten und Anrufe an einen Benutzer weiterzuleiten. Je nach Eingabeauswahl des Benutzers (beispielsweise bei einer Frage zur Abrechnung) kann das IVR-System die VoiceXML-Schnittstelle nutzen, um den Anruf an OpenScape Contact Center zu übergeben. OpenScape Contact Center leitet den Anruf dann zu dem am besten geeigneten Benutzer im Contact Center weiter.

Diese Integration bietet OpenScape Contact Center-Kunden folgende wichtige Vorteile:

- Flexibilität bei der Nutzung eines einzelnen Unternehmens-Webservers oder bei der Verwendung dedizierter Server zur Unterstützung von Leistungsmerkmalen zur VoiceXML- und Web-Integration
- Fähigkeit der Integration von OpenScape Contact Center in eine heterogene Systemumgebung
- Unterstützung der Integration sowohl proprietärer als auch auf Standards basierender IVR-Systeme

Die folgende Abbildung stellt die Topologie der VoiceXML-Integration dar.

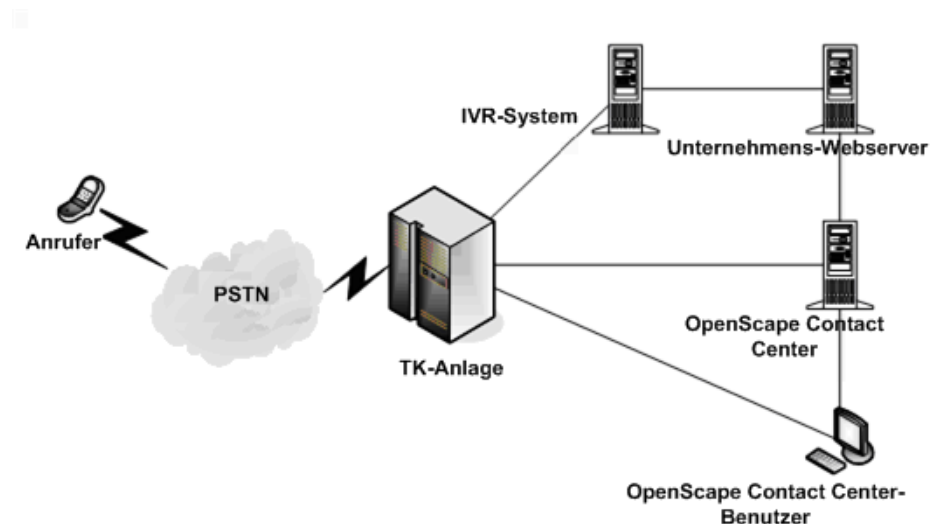


Bild 1

VoiceXML-Integration - Topologie

HINWEIS: Eine Liste der unterstützten Unternehmens-Webserverplattformen finden Sie im *Systemverwaltungshandbuch*.

3 Konfigurieren der OpenScape Contact Center VoiceXML-Integration

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

- Halten im IVR
- Queue Hold

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 Benutzer 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 Benutzer weitergeleitet wurde, müssen Sie die Konfiguration Halten im IVR ändern, um weitere IVR-Nebenstellen aufzunehmen.

Konfigurieren der OpenScape Contact Center VoiceXML-Integration

Konfiguration von Halten im IVR

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.

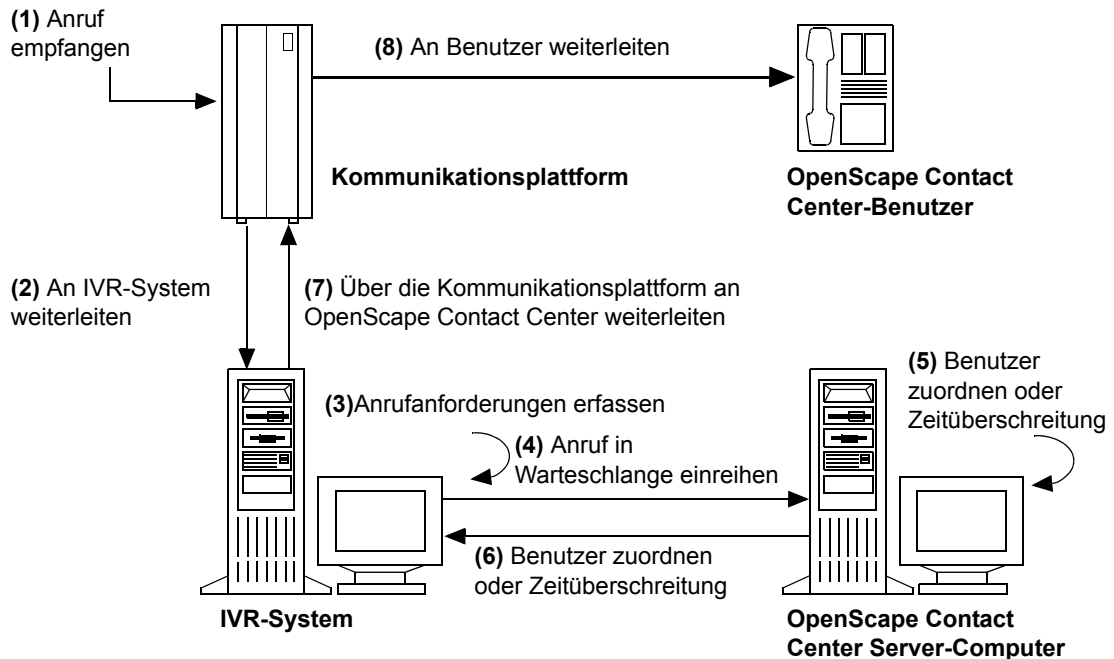


Bild 2 Anrufablauf für die Konfiguration Halten im IVR

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 Benutzer für die Bearbeitung des Anrufs.
5. OpenScape Contact Center führt dann eine der folgenden Aktionen aus:
 - Es ordnet den Anruf dem besten verfügbaren Benutzer zu
 - Es reserviert den Anruf für einen bestimmten Benutzer (optional)Wenn der Anruf bis zum Ende des letzten Anrufschriffs keinem Benutzer zugeordnet werden kann, wird er an eine Zeitüberschreitungs-Nebenstelle weitergeleitet.
6. Das IVR-System fragt den Status des Anrufs ab. Wenn der Anruf die Zeit überschreitet, führt das IVR-System eine der folgenden Aktionen aus:

- Es leitet den Anruf an eine im IVR-Skript angegebene Standardnummer weiter.
 - 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 Benutzer wiederholt.
7. Wenn der Anruf einem Benutzer zugeordnet wurde, leitet das IVR-System den Anruf an die zurückgegebene Transferrnummer weiter.
 8. OpenScape Contact Center leitet den Anruf an den zugeordneten Benutzer weiter.

HINWEIS: Wenn bei Verbindung mit einer OpenScape Voice-Kommunikationsplattform die Weiterleitung fehlschlägt und der Anruf wieder mit dem IVR-System verbunden wird, muss die IVR-Anwendung den Anruf als neuen Anruf verarbeiten.

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/MLHG-Gruppe in der Kommunikationsplattform weiterzuleiten, wo Anrufe warten, bis geeignete Benutzer für die Bearbeitung der Anrufe verfügbar werden.

Sobald ein Benutzer verfügbar wird, leitet OpenScape Contact Center den Anruf aus der OpenScape Contact Center ACD/MLHG-Gruppe an den betreffenden Benutzer 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 Benutzern sucht.

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

Konfigurieren der OpenScape Contact Center VoiceXML-Integration

Konfiguration von Halten in Warteschlange

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.

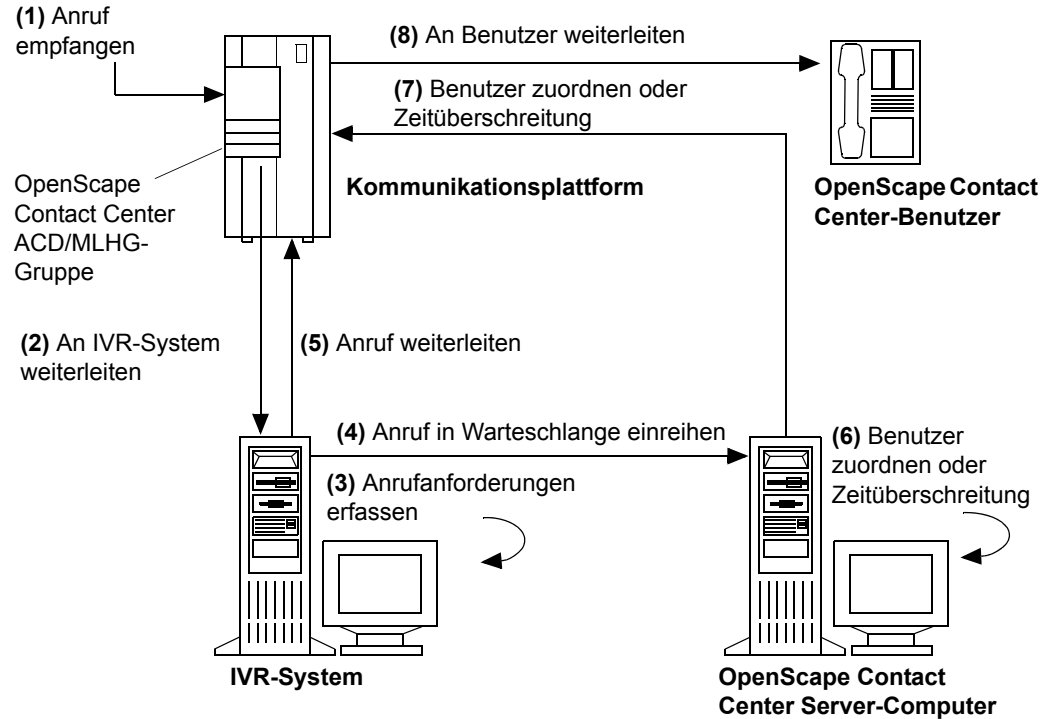


Bild 3 Anrufablauf für die Konfiguration Halten in Warteschlange

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 Benutzer für die Bearbeitung des Anrufs.
5. Zu diesem Zeitpunkt leitet das IVR-System den Anruf an eine Transitnummer und anschließend an eine OpenScape Contact Center ACD/MLHG-Gruppe weiter.
6. OpenScape Contact Center führt dann eine der folgenden Aktionen aus:
 - Es ordnet den Anruf dem besten verfügbaren Benutzer zu
 - Es reserviert den Anruf für einen bestimmten Benutzer (optional)

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

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

HINWEIS: Wenn bei Verbindung mit einer OpenScape Voice-Kommunikationsplattform die Weiterleitung fehlschlägt und der Anruf wieder mit dem IVR-System verbunden wird, muss die IVR-Anwendung den Anruf als neuen Anruf verarbeiten.

3.3 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 VoiceXML-Subdialoge auf.

Die VoiceXML-Subdialoge werden über VoiceXML-Dateien und das **hppcwis**-Servlet (Tomcat) bzw. die Datei **hppcwis.dll** (IIS) bereitgestellt, die auf dem Unternehmens-Webserver installiert sind.

3.4 VoiceXML-Subdialoge für IVR-Systeme

OpenScape Contact Center verwendet OpenScape Contact Center VoiceXML-Subdialoge, um die Integration von IVR-Ressourcen zu erleichtern.

Die folgenden Konzepte müssen 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 den Subdialog Terminate (Beenden), um OpenScape Contact Center über die Gesprächstrennung zu benachrichtigen.

Das IVR-System muss OpenScape Contact Center die ANI- und DNIS-Nummer des Anrufs bereitstellen. Diese Informationen können vom IVR-System über das Trunking-Protokoll der Kommunikationsplattform (zum Beispiel ISDN) abgerufen

Konfigurieren der OpenScape Contact Center VoiceXML-Integration

VoiceXML-Subdialoge für IVR-Systeme

werden. Der Subdialog Initialisierung legt die ANI- und DNIS-Nummer lokal fest, bevor diese Informationen an OpenScape Contact Center gesendet werden. Dieser Subdialog sollte sobald wie möglich aufgerufen werden, nachdem das IVR-System mit der Verarbeitung des ankommenden Anrufs begonnen hat.

Eine IVR-Transitnummer ist eine Pilotnummer zur Integration eines IVR-Systems. Wenn Sie die OpenScape Contact Center VoiceXML-Integration verwenden, ist ein Pool von IVR-Transitnummern erforderlich, um Anrufe vom IVR-System an OpenScape Contact Center weiterzuleiten und nachzuverfolgen. Abhängig vom eingesetzten Typ der Kommunikationsplattform verwendet OpenScape Contact Center eine IVR-Transitnummer, um Anrufe wie folgt weiterzuleiten:

- Auf der OpenScape 4000- oder HiPath 4000-Kommunikationsplattform leitet die IVR-Transitnummer Anrufe an eine OpenScape Contact Center Route-Control-Gruppe weiter.
- Auf der OpenScape Voice-Kommunikationsplattform leitet die IVR-Transitnummer Anrufe an einen OpenScape Contact Center-Sammelanschluss weiter.

Die Transitnummer wird für die Nachverfolgung des IVR-Anrufs nach der Weiterleitung an OpenScape Contact Center benötigt.

OpenScape Contact Center verwaltet einen Pool von Transitnummern, die nur für die Weiterleitung von IVR-Anrufen an OpenScape Contact Center verwendet werden. Das IVR-System fragt bei OpenScape Contact Center unter Verwendung des GetTransitNumber-Subdialogs eine eindeutige Transitnummer ab. Nach dem Empfang der Transitnummer muss das IVR-System den Anruf sofort weiterleiten. Wenn der Anruf die OpenScape Contact Center RCD/ACD-Warteschlange/MLHG erreicht, identifiziert OpenScape Contact Center den Anruf anhand der Transitnummer, und kann dann die zuvor von der IVR empfangenen Anrufinformationen mit dem ankommenden Anruf verbinden. Die Transitnummer steht dann wieder für die Verwendung mit einem anderen 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 den Subdialog 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/ACD-Warteschlange/MLHG aufgerufen werden, also normalerweise kurz nach Einreihung des Anrufs. OpenScape Contact Center sucht dann nach einem Benutzer und leitet den Anruf an den zugeordneten Benutzer 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 RCD/ACD-Warteschlange/MLHG weitergeleitet. Bei Halten im IVR heißt das normalerweise, kurz nachdem der vom QueryCallStatus-Subdialog zurückgegebene Anrufstatus "Wartend" oder "Nicht angenommen" ist. Nach der Weiterleitung des Anrufs an die OpenScape Contact Center RCG/ACD-Warteschlange/MLHG leitet OpenScape Contact Center den Anruf sofort an den zugeordneten Benutzer oder die definierte Zeitüberschreitungs-Nebenstelle weiter.

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

Wenn ein Anruf auf einem IVR-System getrennt (durch das IVR-System oder durch Abbruch des Anrufers) oder extern zu einer Nummer weitergeleitet wird, die keine Transitnummer ist, wird der Subdialog Terminate (Beenden) verwendet, um OpenScape Contact Center zu benachrichtigen.

HINWEIS: Der Subdialog Terminate sollte zu der Aufhängen-Verzweigung des IVR-Systems hinzugefügt und gegebenenfalls in andere Fehlerbehandlungsprozeduren integriert werden. Wenn Sie den Subdialog nicht zu der Aufhängen-Verzweigung hinzufügen, werden Ihre Statistiken ungenau sein, weil OpenScape Contact Center dann nicht erfährt, wann der Anruf beendet wurde.

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. Bereiten Sie die ANI- und DNIS-Informationen für den Anruf vor.
2. Initialisieren Sie die Verbindung zwischen dem IVR-System und OpenScape Contact Center.

Konfigurieren der OpenScape Contact Center VoiceXML-Integration

VoiceXML-Subdialoge für IVR-Systeme

3. Überprüfen Sie den Systemstatus, um sicherzustellen, dass der Routing-Server verfügbar ist.
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 des Enqueue-Subdialogs 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 Benutzer 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 eines IVR-Subdialogs), und ergreifen Sie je nach den Ergebnissen die entsprechende Maßnahme:
 - Wenn der Anruf zugeordnet ist, fordern Sie eine Transitnummer von OpenScape Contact Center an, und leiten Sie den Anruf an die Transitnummer weiter.
 - Wenn, andernfalls, eine Zeitüberschreitung oder ein Fehler auftritt, leiten Sie den Anruf an die im IVR-Skript definierte Standardnummer weiter.

HINWEIS: Nach der erfolgreichen Einreihung eines Anrufs muss das Skript mithilfe des Subdialogs 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.

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

Konfigurieren der OpenScape Contact Center VoiceXML-Integration

VoiceXML-Subdialoge für IVR-Systeme

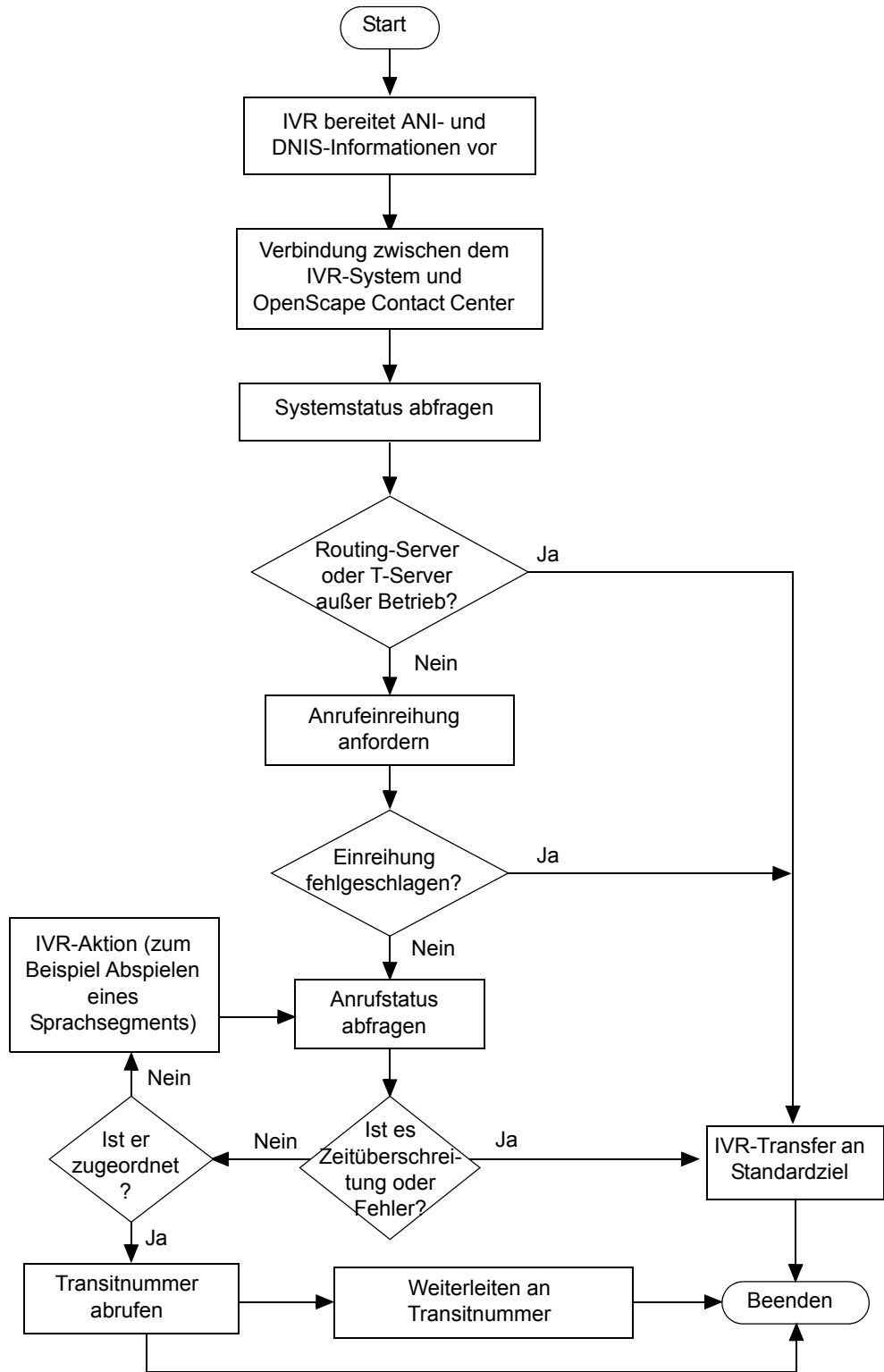


Bild 4

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. Bereiten Sie die ANI- und DNIS-Informationen für den Anruf vor.
2. Initialisieren Sie die Verbindung zwischen dem IVR-System und OpenScape Contact Center.
3. Überprüfen Sie den Systemstatus, um sicherzustellen, dass der Routing-Server verfügbar ist.
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 des Enqueue-Diagrams in die Warteschlange ein. Stellen Sie sicher, dass die Parameter die Konfiguration von Halten in Warteschlange angeben.
6. Fordern Sie eine Transitnummer von OpenScape Contact Center an. Anruf wird an die Transitnummer weitergeleitet und anschließend in eine OpenScape Contact Center ACD/MLHG-Gruppe gesetzt. OpenScape Contact Center hält den Anruf automatisch in der OpenScape Contact Center ACD/MLHG-Gruppe, bis er einem Benutzer zugeordnet wird.

Konfigurieren der OpenScape Contact Center VoiceXML-Integration

VoiceXML-Subdialoge für IVR-Systeme

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

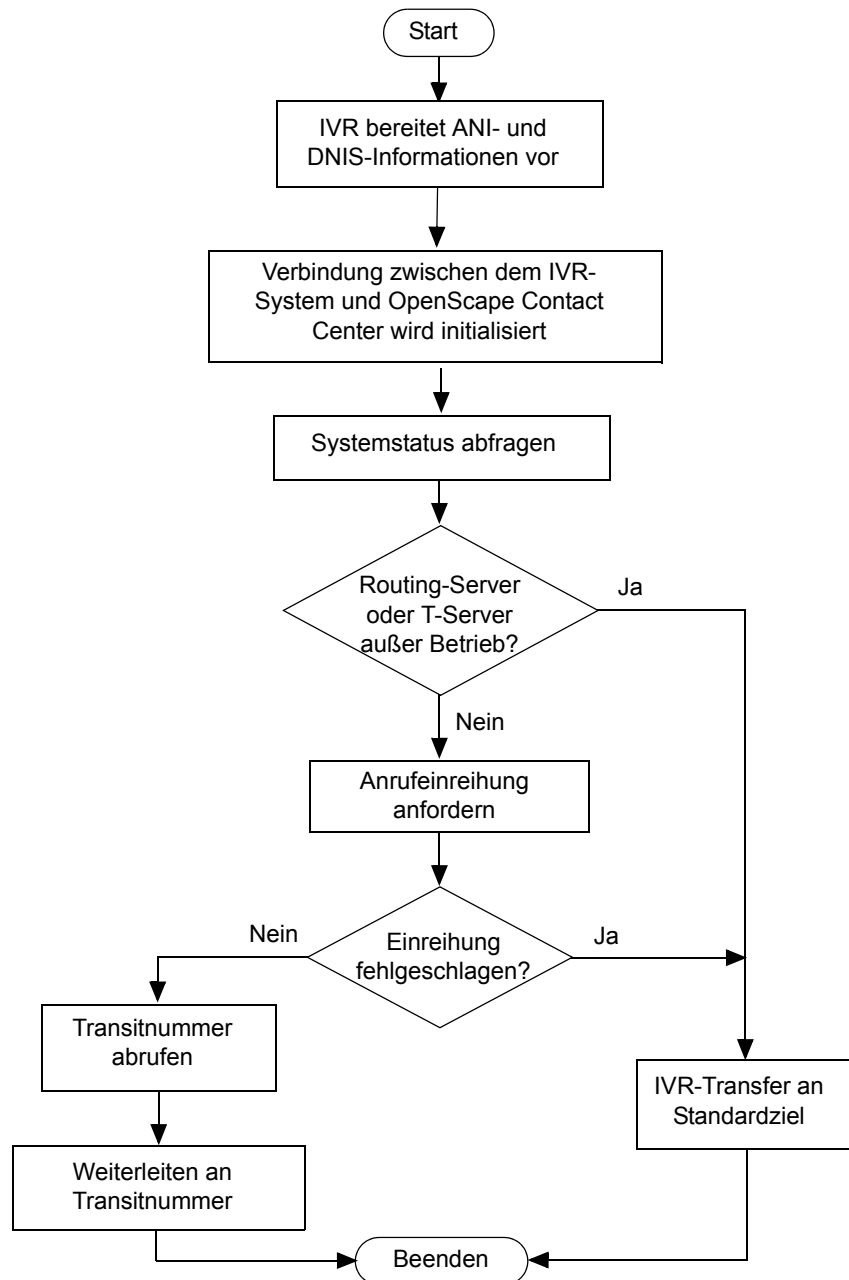


Bild 5

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

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

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 IVR-Front-End-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 VoiceXML-Subdialoge bereitgestellt:

- Subdialog GetBusinessUnit - ruft die Business Unit für einen bestimmten Anruf ab. Weitere Informationen finden Sie in [Section 4.5, "GetBusinessUnit", on page 34](#).
- Subdialog SetBusinessUnit - legt die Business Unit für einen bestimmten Anruf fest. Der Subdialog SetBusinessUnit kann pro IVR-Anruf nur einmal aufgerufen werden. Weitere Informationen finden Sie in [Section 4.15, "SetBusinessUnit", on page 61](#).

HINWEIS: In einer Multi-Tenant-Umgebung muss der Subdialog SetBusinessUnit aufgerufen werden, bevor die folgenden Subdialoge aufgerufen werden: CreateCallback, Enqueue.

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

Konfigurieren der OpenScape Contact Center VoiceXML-Integration

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

4 Verwenden der OpenScape Contact Center VoiceXML-Subdialoge

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

HINWEIS: Ein Fehlercode-Ausgabeparameter ist in allen Subdialogen enthalten.

Der folgende Code wird für alle Subdialoge in den Unterabschnitten **“Syntax“** und **“Beispiel“** verwendet:

```
<script>
  <![CDATA[function GetLink (SubDialogName) {return (http://
  <hostname>/
  <VirtualPath>/VXML/+SubDialogName) }]]>
</script>
```

dabei gilt:

- *hostname* ist der Name des Hostcomputers
- *VirtualPath* ist der Pfad zu dem virtuellen Verzeichnis, das Sie auf dem Unternehmens-Webserver eingerichtet haben.

4.1 CreateCallback

Der Subdialog CreateCallback versucht, einen Callback im Callback-Server einzurichten. Ein IVR-System kann anzeigen, dass ein Kunde zurückgerufen werden soll, wenn ein Agent verfügbar wird. Dieser Subdialog unterstützt bis zu 1000 Byte Kontaktdaten.

HINWEIS: Sie müssen den Subdialog Initialize aufrufen, bevor Sie diesen Subdialog aufrufen. In einer Multi-Tenant-Umgebung müssen Sie zuerst den Subdialog SetBusinessUnit aufrufen, bevor Sie diesen Subdialog aufrufen. Weitere Informationen finden Sie in [Abschnitt 4.8, "Initialize"](#), auf Seite 40 und [Abschnitt 4.15, "SetBusinessUnit"](#), auf Seite 61.

Verwenden der OpenScape Contact Center VoiceXML-Subdialoge

CreateCallback

Syntax

```
<subdialog name="HPPC_CreateCallback"
srcexpr="GetLink('CreateCallback.vxml')">
  <param name="CallID" expr="Call_ID"/>
  <param name="QueueName" expr="m_QueueName"/>
  <param name="Description" expr="'Create callback
description'"/>
  <param name="TimezoneOffset" expr="-300"/>
  <param name="ContactName" expr="'John Doe'"/>
  <param name="Schedule1" expr="'9053268045;05/25/
2007;14:00;05/25/2007;14:30'"/>
  <param name="Schedule2" expr="'9053268046;05/25/
2007;16:00;05/25/2007;16:30'"/>
  <param name="Schedule3" expr="'9053268046;05/26/
2007;16:00;05/26/2007;16:30'"/>
  <filled>
    <assign name="errCode" expr="HPPC_CreateCallback.ErrorCode"/>
    <assign name="ivrCallbackID"
expr="HPPC_CreateCallback.CallbackID"/>
  </filled>
</subdialog>
```

Parameter

Name	Typ	Bereich	Beschreibung
CallID	Eingabe	18 Zeichen	Die Anruf-ID für den Anruf, für den Sie einen Calbback einrichten wollen.
QueueName	Eingabe	32 Zeichen	Die für diesen Callback verwendete Callback-Warteschlange.
Description (optional)	Eingabe	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.
TimeZoneOffset	Eingabe	0 bis +60*12 oder 0 bis -60*12	Der Unterschied in Minuten zwischen der Ortszeit des Anrufers und UTC (Coordinated Universal Time).
ContactName	Eingabe	80 Zeichen	Der Name des Kontakts.
CallbackID	Ausgabe	17 Zeichen	Eine eindeutige ID für den eingerichteten Callback.

Tabelle 1

Parameter für den Subdialog CreateCallback

Name	Typ	Bereich	Beschreibung
Schedule1	Eingabe	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.
Schedule2 (optional)	Eingabe	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)	Eingabe	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 1 Parameter für den Subdialog CreateCallback

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

- 0** Erfolgreich.
- 314** Der angegebene Business Unit-Name wurde nicht festgelegt oder entspricht keinem der Business Unit-Namen in der Datenbank. (Dieser Fehler ist nur relevant, wenn OpenScape Contact Center als Multi-Tenant-System konfiguriert ist.)
- 953** Die Business Unit wurde für diesen Anruf nicht definiert. (Dieser Fehler ist nur relevant, wenn OpenScape Contact Center als Multi-Tenant-System konfiguriert ist.)
- 1006** Sie sind dabei, einen doppelten Callback einzurichten.

Verwenden der OpenScape Contact Center VoiceXML-Subdialoge

CreateCallback

- 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 befindet sich auf der Liste Ausgeschlossene Nummern.
- 1040** Alle Zeitpläne sind bereits abgelaufen.
- 1045** Der Name der Callback-Warteschlange ist ungültig.
- 1047** Der Zeitplan enthält ein Datum, das zu weit in der Zukunft liegt.
- Sons- tige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Fehlercodes"](#).

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.

```
<form id="hppcCreateCallback">
  <subdialog name="HPPC CreateCallback"
    srcexpr="GetLink('CreateCallback.vxml')">
    <param name="CallID" expr="Call_ID"/>
    <param name="QueueName" expr="m_QueueName"/>
    <param name="Description" expr="'Create callback
description'"/>
    <param name="TimezoneOffset" expr="-300"/>
    <param name="ContactName" expr="'John Doe'"/>
    <param name="Schedule1" expr="'9053268045;05/25/
2007;14:00;05/25/2010;14:30'"/>
    <param name="Schedule2" expr="'9053268046;05/25/
2007;16:00;05/25/2010;16:30'"/>
    <param name="Schedule3" expr="'9053268046;05/26/
2007;16:00;05/26/2010;16:30'"/>
    <filled>
      <assign name="errCode"
        expr="HPPC_CreateCallback.ErrorCode"/>
      <assign name="ivrCallbackID"
        expr="HPPC_CreateCallback.CallbackID"/>
    </filled>
  </subdialog>
</block>
<if cond="errCode == 0">
  <prompt cond="PlayWarnings==1">CreateCallback succeed.
  Callback ID is <value expr="ivrCallbackID"/> </prompt>
  <log>HPPC CallbackID=<value expr="ivrCallbackID"/>,
  Callback ID is <value expr="ivrCallbackID"/>:
  CreateCallback succeed.</log>
```

```

<goto next="#hppcCallbackMenu"/>
<else/>
    <prompt>An error has occurred in CreateCallback subdialog.
    The error number is <value expr="errCode"/>. </prompt>
    <goto next="#hppcTerminate"/>
</if>
</block>
</form>

```

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.

4.2 DeleteCallback

Der Subdialog DeleteCallback wird verwendet, um einen vorhandenen Callback zu löschen, der mit dem Subdialog CreateCallback eingerichtet wurde. Dieser Subdialog verwendet als Parameter die vom Subdialog CreateCallback zurückgegebene Callback-ID.

HINWEIS: Sie müssen den Subdialog Initialize aufrufen, bevor Sie diesen Subdialog aufrufen. Weitere Informationen finden Sie in [Abschnitt 4.8, "Initialize"](#), auf Seite 40.

Verwenden der OpenScape Contact Center VoiceXML-Subdialoge

DeleteCallback

Syntax

```
<subdialog name="HPPC_DeleteCallback"
srcexpr="GetLink('DeleteCallback.vxml')">
    <param name="CallbackID" expr="ivrCallbackID"/>
<filled>
    <assign name="errCode" expr="HPPC_DeleteCallback.ErrorCode"/>
</filled>
</subdialog>
```

Parameter

Name	Typ	Bereich	Beschreibung
CallbackID	Eingabe	17 Zeichen	Die zurückgegebene Callback-ID einer erfolgreichen CreateCallback-Anforderung, die gelöscht werden soll.

Tabelle 2 Parameter für den Subdialog DeleteCallback

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

0 Erfolgreich.

-1021 Ungültige Callback-ID.

Sons- Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu
tige einem bestimmten Code finden Sie unter [Kapitel 5, "Fehlercodes"](#).

Beispiel

```
<form id="hppcDeleteCallback">
    <subdialog name="HPPC_DeleteCallback"
srcexpr="GetLink('DeleteCallback.vxml')">
        <param name="CallbackID" expr="ivrCallbackID"/>
    <filled>
        <assign name="errCode"
expr="HPPC_DeleteCallback.ErrorCode"/>
    </filled>
    </subdialog>
    <block>
    <if cond="errCode == 0">
        <prompt cond="PlayWarnings==1">DeleteCallback succeed.</
prompt>
    <goto next="#hppcTerminate"/>
    <else/>
        <prompt>An error has occurred in DeleteCallback subdialog.
The error number is <value expr="errCode"/>. </prompt>
    <goto next="#hppcTerminate"/>
    </if>
    </block>
</form>
```

```

    </if>
  </block>
</form>

```

4.3 Dequeue

Der Subdialog 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 des Subdialogs 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 den Subdialog Initialize aufrufen, bevor Sie diesen Subdialog aufrufen. Weitere Informationen finden Sie in [Abschnitt 4.8, "Initialize"](#), auf Seite 40.

Syntax

```

<subdialog name="HPPC_Dequeue"
srcexpr="GetLink('Dequeue.vxml')">
  <!-- Anruf-ID des aktuellen Anrufs. Obligatorisch-->
  <param name="CallID" expr="Call_ID"/>
<filled>
  <assign name="errCode" expr="HPPC_Dequeue.ErrorCode"/>
</filled>
</subdialog>

```

Parameter

Name	Typ	Bereich	Beschreibung
CallID	Eingabe	18 Zeichen	Die Anruf-ID für den Anruf, den Sie herausnehmen wollen.

Tabelle 3

Parameter für den Subdialog Dequeue

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

- 0** Erfolgreich.
- 306** 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** Die CallID wurde nicht gefunden. Sie müssen die Funktion Initialize aufrufen, um die CallID zu beziehen.
- 914** Der versuchte Vorgang ist fehlgeschlagen.
- 950** VoiceXML ist derzeit nicht aktiviert.
- 951** Der Routing-Server ist nicht verfügbar.
- Sons-
tige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Fehlercodes"](#).

Beispiel

Die folgende Beispielfunktion sendet eine Anforderung zum Herausnehmen eines Anrufs.

```
<form id="hppcDequeue">
  <subdialog name="HPPC_Dequeue"
    srcexpr="GetLink('Dequeue.vxml')">
    <!-- Anruf-ID des aktuellen Anrufs. Obligatorisch-->
    <param name="CallID" expr="Call_ID"/>
    <filled>
      <assign name="errCode" expr="HPPC_Dequeue.ErrorCode"/>
    </filled>
  </subdialog>
  <block>
    <if cond="errCode == 0">
      <prompt>Dequeue succeed.</prompt>
      <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
        <value expr="Call_ID"/>: Dequeue succeed.</log>
      <goto next="#hpcTerminate"/>
    <else/>
      <prompt>Dequeue failed. Error ID is <value expr="errCode"/>
        .</prompt>
      <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
        <value expr="Call_ID"/>. Dequeue failed. Error ID is
        <value expr="errCode"/>.</log>
      <goto next="#hpcTerminate"/>
    </if>
  </block>
</form>
```

4.4 Enqueue

Der Subdialog Enqueue fordert den Routing-Server auf, einen Anruf in die Warteschlange einzureihen und den besten verfügbaren Benutzer für die Bearbeitung des Anrufs zu finden. Dieser Subdialog 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 diesen Subdialog, 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 des Subdialogs 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/MLHG-Gruppe weitergeleitet werden.

HINWEIS: Sie müssen den Subdialog Initialize aufrufen, bevor Sie diesen Subdialog aufrufen. In einer Multi-Tenant-Umgebung müssen Sie zuerst den Subdialog SetBusinessUnit aufrufen, bevor Sie diesen Subdialog aufrufen. Weitere Informationen finden Sie in [Abschnitt 4.8, "Initialize"](#), auf Seite 40 und [Abschnitt 4.15, "SetBusinessUnit"](#), auf Seite 61.

Syntax

```
<subdialog name="HPPC_Enqueue"
srcexpr="GetLink(&apos;Enqueue.vxml&apos;)">
  <param name="CallID" expr="Call_ID"/>
  <param name="IVRHold" expr="ivrHold"/>
  <param name="QueueName" expr="&apos;Sales&apos;"/>
  <param name="InitialPriority" expr="m_InitialPriority"/>
  <param name="Description" expr="m_Description"/>
  <param name="AgentId" expr="m_AgentId"/>
  <param name="AgentWaitTime" expr="m_AgentWaitTime"/>
<filled>
  <assign name=
    "m_EstimatedWait"expr="HPPC_Enqueue.EstimatedWait"/>
  <assign name="m_CallsInQ" expr="HPPC_Enqueue.CallsInQ"/>
  <assign name="errCode" expr="HPPC_Enqueue.ErrorCode"/>
</filled>
</subdialog>
```

Parameter

Name	Typ	Bereich	Beschreibung
CallID	Eingabe	18 Zeichen	Die Anruf-ID für den Anruf, den Sie einreihen wollen.
IVRHold	Eingabe	0 oder 1	0- Gibt an, dass der Anruf für die Konfiguration Halten in Warteschlange an eine ACD/MLHG-Gruppe weitergeleitet wird. 1- Gibt an, dass der Anruf auf der IVR-Nebenstelle auf die Konfiguration Halten im IVR wartet.
QueueName	Eingabe	32 Zeichen	Der Name der Warteschlange für den Anruf.
InitialPriority	Eingabe	1 bis 100	Gibt die Priorität eines Anrufs an, wobei 1 die niedrigste und 100 die höchste Priorität ist.
Description (optional)	Eingabe	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	Ausgabe	Größer oder gleich 0	Die geschätzte Wartezeit (in Sekunden) für diese Warteschlange.
CallsInQueue	Ausgabe	Größer oder gleich 0	Die Anzahl der Anrufe in der angegebenen Warteschlange.
AgentID (optional)	Eingabe	32 Zeichen	Der Benutzer, für den Sie den Anruf reservieren wollen.
AgentWait Time (optional)	Eingabe	32 Zeichen	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.

Tabelle 4 Parameter für den Subdialog Enqueue

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

- 0** Erfolgreich.
- 305** Dieser Code entspricht einem unbekannten Fehler (z. B. Benutzerfehler oder System ist instabil).

- 306** 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** Die CallID wurde nicht gefunden. Sie müssen die Funktion Initialize aufrufen, um die CallID zu beziehen.
- 314** Der angegebene Business Unit-Name wurde nicht festgelegt oder entspricht keinem der Business Unit-Namen in der Datenbank. (Dieser Fehler ist nur relevant, wenn OpenScape Contact Center als Multi-Tenant-System konfiguriert ist.)
- 400** Die Systemressourcen sind niedrig.
- 904** Die angegebene Agenten-ID stimmt nicht mit der ID eines Agenten in der Datenbank überein.
- 905** Die angegebene Warteschlange entspricht keinem der Warteschlangennamen in der Datenbank.
- 914** Der versuchte Vorgang ist fehlgeschlagen.
- 950** VoiceXML ist derzeit nicht aktiviert.
- 951** Der Routing-Server ist nicht verfügbar.
- 953** Die Business Unit wurde für diesen Anruf nicht definiert. (Dieser Fehler ist nur relevant, wenn OpenScape Contact Center als Multi-Tenant-System konfiguriert ist.)
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Fehlercodes"](#).

Beispiel

```
<form id="hppcEnqueue">
  <var name="m_EstimatedWait"/>
  <var name="m_CallsInQ"/>
  <subdialog name="HPPC_Enqueue"
    srcexpr="GetLink(&apos;Enqueue.vxml&apos;)">
    <param name="CallID" expr="Call_ID"/>
    <param name="IVRHold" expr="ivrHold"/>
    <param name="QueueName" expr="&apos;Sales&apos;"/>
    <param name="InitialPriority" expr="m_InitialPriority"/>
    <param name="Description" expr="m_Description"/>
    <param name="AgentId" expr="m_AgentId"/>
    <param name="AgentWaitTime" expr="m_AgentWaitTime"/>
  </subdialog>
  <filled>
    <assign name="
      m_EstimatedWait"expr="HPPC_Enqueue.EstimatedWait"/>
    <assign name="m_CallsInQ" expr="HPPC_Enqueue.CallsInQ"/>
    <assign name="errCode" expr="HPPC_Enqueue.ErrorCode"/>
  </filled>
</form>
```

```
<block>
<if cond="errCode == 0">
  <prompt>Enqueue succeed.</prompt>
  <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
  <value expr="Call_ID"/>: Enqueue succeed.
  EstimatedWaitTime = <value expr="m_EstimatedWait"/>
  CallsInQ = <value expr="m_CallsInQ"/>
</log>
  <goto next="#hppcGetTransitNumber"/>
<else/>
  <prompt>Enqueue failed. Error ID is <value expr="errCode"/>
  >. </prompt>
  <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
  <value expr="Call_ID"/>. Enqueue failed. Error ID is
  <value expr="errCode"/>.</log>
  <goto next="#hppcTerminate"/>
</if>
</block>
</form>
```

4.5 GetBusinessUnit

HINWEIS: Der Subdialog GetBusinessUnit wird nur in einer Multi-Tenant-Umgebung unterstützt.

Der Subdialog GetBusinessUnit ruft die Business Unit für den definierten Anruf ab.

HINWEIS: Sie müssen die Subdialoge Initialize und SetBusinessUnit aufrufen, bevor Sie diese Subdialoge aufrufen. Weitere Informationen finden Sie in [Abschnitt 4.8, "Initialize"](#), auf Seite 40 und [Abschnitt 4.15, "SetBusinessUnit"](#), auf Seite 61.

Syntax

```
<subdialog name="HPPC_GetBusinessUnit"
srcexpr="GetLink('GetBusinessUnit.vxml')">
  <param name="CallID" expr="Call_ID"/>
  <filled>
    <assign name="errCode"
    expr="HPPC_GetBusinessUnit.ErrorCode"/>
```

```

    <assign name="BUName"
      expr="HPPC_GetBusinessUnit.BusinessUnitName" />
  </filled>
</subdialog>

```

Parameter

Name	Typ	Bereich	Beschreibung
CallID	Eingabe	18 Zeichen	Die Anruf-ID des Anrufs, für den Sie die Business Unit abrufen wollen.
BusinessUnit Name	Ausgabe	32 Zeichen	Der Name der Business Unit für den definierten Anruf.

Tabelle 5 Parameter für den Subdialog GetBusinessUnit

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

- 0** Erfolgreich.
- 306** 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** Die Systemressourcen sind niedrig.
- 914** Der versuchte Vorgang ist fehlgeschlagen.
- 950** VoiceXML ist derzeit nicht aktiviert.
- 953** Die Business Unit wurde für diesen Anruf nicht definiert. (Dieser Fehler ist nur relevant, wenn OpenScape Contact Center als Multi-Tenant-System konfiguriert ist.)
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Fehlercodes"](#).

Beispiel

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

```

<form id="hppcGetBusinessUnit">
  <var name="BUName" />
  <subdialog name="HPPC_GetBusinessUnit"
    srcexpr="GetLink('GetBusinessUnit.vxml')">
    <param name="CallID" expr="Call_ID"/>
  </subdialog>
  <filled>
    <assign name="errCode"
      expr="HPPC_GetBusinessUnit.ErrorCode" />
    <assign name="BUName"
      expr="HPPC_GetBusinessUnit.BusinessUnitName" />
  </filled>
</form>

```

```
<block>
<if cond="errCode == 0">
  <prompt cond="PlayWarnings==1">Get Business Unit
  succeed.</prompt>
  <prompt>
  Business Unit Name is <value expr="BUName"/>.
</prompt>
  <log>HPPC Call ID is <value expr="Call_ID"/>: Get Business
  Unit succeed. Business Unit Name is <value expr="BUName"/
  >.</log>
<else/>
  <prompt>An error has occurred in Get Business Unit
  subdialog. The error number is <value expr="errCode"/>. </
  prompt>
  <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
  <value expr="Call_ID"/>. Get Business Unit failed. Error
  ID is <value expr="errCode"/>.</log>
</if>
<goto next="#hppcMainMenu"/>
</block>
</form>
```

4.6 GetContactData

Der Subdialog 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 den Subdialog Initialize aufrufen, bevor Sie diesen Subdialog aufrufen. Weitere Informationen finden Sie in [Abschnitt 4.8, "Initialize"](#), auf Seite 40.

Syntax

```
<subdialog name="HPPC_GetContactData"
srcexpr="GetLink('GetContactData.vxml')">
  <param name="CallID" expr="Call_ID"/>
  <param name="Key" expr="'queuekey'"/>
<filled>
  <assign name="m_value" expr="HPPC_GetContactData.Value"/>
  <assign name="errCode" expr="HPPC_GetContactData.ErrorCode"/>
</filled>
</subdialog>
```

Parameter

Name	Typ	Bereich	Beschreibung
CallID	Eingabe	18 Zeichen	Die Callback-ID für den Anruf, für den Sie die Kontaktdaten abrufen wollen.
Key	Eingabe	32 Zeichen	Der Schlüsselname für den Kontaktdatenwert.
Value	Ausgabe	128 Zeichen	Der für den Schlüssel zurückgegebene Wert.

Tabelle 6 Parameter für den Subdialog GetContactData

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

- 0** Erfolgreich.
- 306** 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.
- 308** Die CallID wurde nicht gefunden. Sie müssen die Funktion Initialize aufrufen, um die CallID zu beziehen.
- 400** Die Systemressourcen sind niedrig.
- 914** Der versuchte Vorgang ist fehlgeschlagen.
- 950** VoiceXML ist derzeit nicht aktiviert.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Fehlercodes"](#).

Beispiel

```
<form id="hppcGetContactData">
  <subdialog name="HPPC_GetContactData"
    srcexpr="GetLink('GetContactData.vxml')">
    <param name="CallID" expr="Call_ID"/>
    <param name="Key" expr="'queuekey'"/>
    <filled>
      <assign name="m_value" expr="HPPC_GetContactData.Value"/>
      <assign name="errCode"
        expr="HPPC_GetContactData.ErrorCode"/>
    </filled>
  </subdialog>
  <block>
    <if cond="errCode == 0">
      <prompt>GetContactData succeed. Contact Data Value is
        <value expr="m_value"/></prompt>
    </if>
  </block>
</form>
```

Verwenden der OpenScape Contact Center VoiceXML-Subdialoge

GetTransitNumber

```
<log>HPPC Call ID is <value expr="Call_ID"/>:
GetContactData succeed.</log>
<goto next="#hppcSetDisplay"/>
<else/>
  <prompt>GetContactData failed. Error ID is <value
    expr="errCode"/>. </prompt>
  <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
    <value expr="Call_ID"/>. GetContactData failed. Error ID
    is <value expr="errCode"/>.</log>
  <goto next="#hppcTerminate"/>
</if>
</block>
</form>
```

4.7 GetTransitNumber

Der Subdialog 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 dieser Subdialog nach dem Aufruf des Subdialogs Enqueue aufgerufen. Bei Halten im IVR wird dieser Subdialog aufgerufen, nachdem der Status "Wartend" oder "Nicht angenommen" im Subdialog QueryCallStatus empfangen wurde.

Syntax

```
<subdialog name="HPPC_GetTransitNumber"
srcexpr="GetLink(&apos;GetTransitNumber.vxml&apos;)">
<!-- Anruf-ID des aktuellen Anrufs. Obligatorisch-->
  <param name="CallID" expr="Call_ID"/>
<filled>
  <assign name="TransferDest"
    expr="HPPC_GetTransitNumber.TransitNumber"/>
  <assign name="errCode"
    expr="HPPC_GetTransitNumber.ErrorCode"/>
</filled>
</subdialog>
```

Parameter

Name	Typ	Bereich	Beschreibung
CallID	Eingabe	18 Zeichen	Die Anruf-ID für den Anruf, den Sie einreihen wollen.
TransitNumber	Ausgabe	80 Zeichen	Eine Pilotnummer, die das IVR-System zum Weiterleiten des Anrufs verwendet.

Tabelle 7 Parameter für den Subdialog GetTransitNumber

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

- 0** Erfolgreich.
- 306** 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** Die CallID wurde nicht gefunden. Sie müssen die Funktion Initialize aufrufen, um die CallID zu beziehen.
- 400** Die Systemressourcen sind niedrig.
- 908** Der T-Server ist nicht verfügbar.
- 914** Der versuchte Vorgang ist fehlgeschlagen.
- 950** VoiceXML ist derzeit nicht aktiviert.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Fehlercodes"](#).

Beispiel

```
<form id="hppcGetTransitNumber">
  <subdialog name="HPPC_GetTransitNumber"
    srcexpr="GetLink(&apos;GetTransitNumber.vxml&apos;)">
    <!-- Anruf-ID des aktuellen Anrufs. Obligatorisch-->
    <param name="CallID" expr="Call_ID"/>
    <filled>
      <assign name="TransferDest"
        expr="HPPC_GetTransitNumber.TransitNumber"/>
      <assign name="errCode"
        expr="HPPC_GetTransitNumber.ErrorCode"/>
    </filled>
  </subdialog>
  <block>
    <if cond="errCode == 0">
      <prompt>GetTransitNumber succeed.</prompt>
      <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
        <value expr="Call_ID"/>: GetTransitNumber succeed.
        TransitNumber = <value expr="TransferDest"/>
```

Verwenden der OpenScape Contact Center VoiceXML-Subdialoge

Initialize

```
</log>
<goto next="#hppcTransfer"/>
<else/>
  <prompt>GetTransitNumber failed. Error ID is <value
    expr="errCode"/>. </prompt>
  <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
    <value expr="Call_ID"/>. GetTransitNumber failed. Error ID
    is <value expr="errCode"/>.</log>
  <goto next="#hppcTerminate"/>
</if>
</block>
</form>
```

4.8 Initialize

Der Subdialog Initialize initialisiert die Verbindung zu den OpenScape Contact Center-Servern und muss aufgerufen werden, bevor einer der anderen OpenScape Contact Center VoiceXML-Subdialoge aufgerufen wird. Dieser Subdialog löscht die ANI- und DNIS-Informationen, die während des vorangegangenen Anrufs auf dieser Nebenstelle eingerichtet wurden. Rufen Sie diesen Subdialog daher am Anfang Ihres IVR-Skripts auf und stellen Sie sicher, dass OpenScape Contact Center betriebsbereit ist, bevor Sie Anrufe weiterleiten.

Syntax

```
<subdialog name="HPPC Initialize"
srcexpr="GetLink(&apos;Initialize.vxml&apos;)">
  <param name="UniqueID" expr="uniqueid"/>
  <param name="ANI" expr="ani"/>
  <param name="DNIS" expr="dnis"/>
  <prompt>Initialize started.</prompt>
  <filled>
    <assign name="errCode" expr="HPPC_Initialize.ErrorCode"/>
    <assign name="Call_ID" expr="HPPC_Initialize.CallID"/>
  </filled>
</subdialog>
```

Parameter

Name	Typ	Bereich	Beschreibung
UniqueID	Eingabe	16 Zeichen	Das Gerät, das mit der IVR-Nebenstelle verbunden ist, auf der das IVR-Skript ausgeführt wird.
ANI (optional)	Eingabe	161 Zeichen	Die ANI-Nummer des Anrufs. Der Statistikserver speichert diese Nummer in der OpenScape Contact Center-Datenbank.
DNIS (optional)	Eingabe	161 Zeichen	Die DNIS-Nummer des Anrufs. Der Statistikserver speichert diese Nummer in der OpenScape Contact Center-Datenbank.
CallID	Ausgabe	18 Zeichen	Die Anruf-ID für den aktuellen Anruf.

Tabelle 8 Parameter für den Subdialog Initialize

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

- 0** Erfolgreich.
- 306** 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** Die CallID wurde nicht gefunden. Sie müssen die Funktion Initialize aufrufen, um die CallID zu beziehen.
- 400** Die Systemressourcen sind niedrig.
- 908** Der T-Server ist nicht verfügbar.
- 914** Der versuchte Vorgang ist fehlgeschlagen.
- 950** VoiceXML ist derzeit nicht aktiviert.

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

Beispiel

```
<form id="hppcInitialize">
  <subdialog name="HPPC_Initialize"
    srcexpr="GetLink(&apos;Initialize.vxml&apos;)">
    <param name="UniqueID" expr="dnis"/>
    <param name="ANI" expr="ani"/>
    <param name="DNIS" expr="dnis"/>
    <prompt>Initialize started.</prompt>
  </filled>
  <assign name="errCode" expr="HPPC_Initialize.ErrorCode"/>
  <assign name="Call_ID" expr="HPPC_Initialize.CallID"/>
</form>
```

```
</filled>
</subdialog>
<block>
<if cond="errCode == 0">
    <prompt>Initialize succeed.</prompt>
    <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
    <value expr="Call_ID"/>: Initialize succeed.</log>
    <goto next="#hppcQueryRoutingInfo"/>
<else/>
    <prompt>Initialize failed. Error ID is <value
    expr="errCode"/>.</prompt>
    <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
    <value expr="Call_ID"/>. Initialize failed. Error ID is
    <value expr="errCode"/>.</log>
    <goto next="#hppcTerminate"/>
</if>
</block>
</form>
```

4.9 QueryAgentStatus

Der Subdialog QueryAgentStatus fragt den Status eines Benutzers ab, der mit einem bestimmten Gerät verbunden ist. Dieser Subdialog gibt den Status des Benutzers als Zeichenfolge zurück, abhängig davon, ob er angemeldet ist oder nicht. Sie müssen entweder die Agenten-ID oder das Gerät angeben. Wenn die Agenten-ID angegeben und der Benutzer 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 den Subdialog Initialize aufrufen, bevor Sie diesen Subdialog aufrufen. Weitere Informationen finden Sie in [Abschnitt 4.8, "Initialize"](#), auf Seite 40.

Syntax

```
<subdialog name="HPPC_QueryAgentStatus"
srcexpr="GetLink('QueryAgentStatus.vxml')">
    <param name="inAgentID" expr="'1234'"/>
    <param name="inDevice" expr="''"/>
<filled>
    <assign name="m_AgentId"
    expr="HPPC_QueryAgentStatus.outAgentID"/>
```

```

<assign name="Device" expr="HPPC_QueryAgentStatus.outDevice"/>

<assign name="AgentStatus"
expr="HPPC_QueryAgentStatus.AgentStatus"/>

<assign name="errCode"
expr="HPPC_QueryAgentStatus.ErrorCode"/>

</filled>

</subdialog>

```

Parameter

Name	Typ	Bereich	Beschreibung
AgentID	Eingabe und Ausgabe	8 Zeichen	Eine Zeichenfolge, die den Benutzer identifiziert; ist leer, wenn das Gerät angegeben wird.
Device	Eingabe und Ausgabe	16 Zeichen	Die Benutzer-Nebenstelle, die Sie abfragen möchten; ist leer, wenn die Agenten-ID angegeben wird.
AgentStatus	Ausgabe	Größer als 0	Gibt den Status eines Benutzers an. 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 9

Parameter für den Subdialog QueryAgentStatus

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

- 0** Erfolgreich.
- 306** 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** Die CallID wurde nicht gefunden. Sie müssen die Funktion Initialize aufrufen, um die CallID zu beziehen.
- 400** Die Systemressourcen sind niedrig.
- 904** Die angegebene Agenten-ID stimmt nicht mit der ID eines Agenten in der Datenbank überein.

Verwenden der OpenScape Contact Center VoiceXML-Subdialoge

QueryAgentStatus

- 908** Der T-Server ist nicht verfügbar.
 - 909** Das angegebene Gerät entspricht keinem Gerät in der Datenbank.
 - 914** Der versuchte Vorgang ist fehlgeschlagen.
 - 950** VoiceXML ist derzeit nicht aktiviert.
- Sons- tige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Fehlercodes"](#).

Beispiel

```
<form id="hppcQueryAgentStatus">
  <var name="AgentStatus"/>
  <var name="Device"/>
  <subdialog name="HPPC_QueryAgentStatus"
    srcexpr="GetLink('QueryAgentStatus.vxml')">
    <param name="inAgentID" expr="'1234'"/>
    <param name="inDevice" expr="'6903'"/>
    <filled>
      <assign name="m_AgentId"
        expr="HPPC_QueryAgentStatus.outAgentID"/>
      <assign name="Device"
        expr="HPPC_QueryAgentStatus.outDevice"/>
      <assign name="AgentStatus"
        expr="HPPC_QueryAgentStatus.AgentStatus"/>
      <assign name="errCode"
        expr="HPPC_QueryAgentStatus.ErrorCode"/>
    </filled>
  </subdialog>
  <block>
    <if cond="errCode == 0">
      <prompt>Query Agent Status succeed </prompt>
      <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
        <value expr="Call_ID"/>: QueryAgentStatus succeed.</log>
      <goto next="#hppcDequeue"/>
    </if>
    <else/>
      <prompt>Query Agent Status failed. Error ID is <value
        expr="errCode"/>. </prompt>
      <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
        <value expr="Call_ID"/>. QueryAgentStatus failed. Error ID
        is <value expr="errCode"/>.</log>
      <goto next="#hppcTerminate"/>
    </else>
  </block>
</form>
```

4.10 QueryCallStatus

Der Subdialog 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 den Subdialog QueryCallStatus nach der Warteschlangeneinreihung eines Anrufs auf, um zu bestimmen, ob der Anruf zugeordnet wurde, sodass das IVR-System den Anruf weiterleiten kann, sobald ein Benutzer vom Routing-Server zugewiesen wurde. Hierzu muss das IVR-System den Subdialog GetTransitNumber aufrufen. Der Anruf kann dann an die Transitnummer weitergeleitet werden. Weitere Informationen finden Sie in [Abschnitt 4.7, "GetTransitNumber"](#), auf Seite 38.

HINWEIS: Sie müssen den Subdialog Initialize aufrufen, bevor Sie diesen Subdialog aufrufen. Weitere Informationen finden Sie in [Abschnitt 4.8, "Initialize"](#), auf Seite 40.

HINWEIS: Bei Verwendung des Subdialogs QueryCallStatus wird die Position in der Warteschlange nur zurückgegeben, wenn der Parameter Timeout auf 0 gesetzt ist. Ist der Parameter Timeout auf einen Wert größer 0 gesetzt, wartet der Subdialog QueryCallStatus auf eine Änderung des Status, fragt aber nicht die Position in der Warteschlange vom Routing-Server ab. Rufen Sie zum Erzielen bestmöglicher Ergebnisse den Subdialog QueryCallStatus auf und geben Sie einen Timeout-Wert von 0 an, damit die Position in der Warteschlange zurückgegeben wird. Rufen Sie anschließend den Subdialog QueryCallStatus erneut auf, geben Sie einen Timeout-Wert größer 0 an und warten Sie auf eine Änderung des Status.

Syntax

```
<subdialog name="HPPC_QueryCallStatus"
srcexpr="GetLink('QueryCallStatus.vxml')">
  <!-- Anruf-ID des aktuellen Anrufs. Obligatorisch-->
  <param name="CallID" expr="Call_ID"/>
  <!-- Timeout in msec. -->
  <param name="Timeout" expr="1000"/>
  <filled>
    <assign name="m_PositionInQ"
      expr="HPPC_QueryCallStatus.PositionInQ"/>
    <assign name="errCode" expr="HPPC_QueryCallStatus.ErrorCode"/>
  </filled>
</subdialog>
```

Verwenden der OpenScape Contact Center VoiceXML-Subdialoge

QueryCallStatus

```
</filled>  
</subdialog>
```

Parameter

Name	Typ	Bereich	Beschreibung
CallID	Eingabe	18 Zeichen	Die von Initialize zurückgegebene Anruf-ID.
TimeOut	Eingabe	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.
PositionInQ	Ausgabe	Größer oder gleich 0	Die vom Routing-Server zurückgegebene Position des Anrufs in der Warteschlange, wenn der Parameter TimeOut auf 0 gesetzt ist.

Tabelle 10 Parameter für den Subdialog QueryCallStatus

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs 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 OpenScape Contact Center-Transitnummer weiter.
- 3** Der Anruf ist im Zustand Nicht angenommen. Leiten Sie in diesem Fall den Anruf an die OpenScape Contact Center-Transitnummer 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 OpenScape Contact Center-Transitnummer weitergeleitet werden.
- 306** 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** Die CallID wurde nicht gefunden. Sie müssen die Funktion Initialize aufrufen, um die CallID zu beziehen.
- 400** Die Systemressourcen sind niedrig.
- 914** Der versuchte Vorgang ist fehlgeschlagen.
- 950** VoiceXML ist derzeit nicht aktiviert.
- 951** Der Routing-Server ist nicht verfügbar.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Fehlercodes"](#).

Beispiel

```

<form id="hppcQueryCallStatus">
  <var name="m_PositionInQ"/>
  <subdialog name="HPPC_QueryCallStatus"
    srcexpr="GetLink('QueryCallStatus.vxml')">
    <!-- Anruf-ID des aktuellen Anrufs. Obligatorisch-->
    <param name="CallID" expr="Call_ID"/>
    <!-- TimeOut in msec. -->
    <param name="TimeOut" expr="1000"/>
    <filled>
      <assign name="m_PositionInQ"
        expr="HPPC_QueryCallStatus.PositionInQ"/>
      <assign name="errCode"
        expr="HPPC_QueryCallStatus.ErrorCode"/>
    </filled>
  </subdialog>
  <block>
    <prompt cond="PlayWarnings==1">QueryCallStatus finished
      with error code <value expr="errCode"/></prompt>
    <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
      <value expr="Call_ID"/>: QueryCallStatus finished.
      TransferDest = <value expr="TransferDest"/>
      PositionInQ = <value expr="m_PositionInQ"/>
      errCode = <value expr="errCode"/>
    </log>
    <if cond="errCode == 1">
      <prompt>Your position in queue is <value
        expr="m_PositionInQ"/>. Your call is in Queue state.
        Continue to wait</prompt>
      <goto next="#hppcSetDisplay"/>
    <elseif cond="errCode == 2"/>
      <prompt>The call is in Pending state and will be
        transferred.</prompt>
      <goto next="#hppcGetTransitNumber"/>
    <elseif cond="errCode == 3"/>
      <prompt>The call is in Unanswered state and will be
        transferred.</prompt>
      <goto next="#hppcGetTransitNumber"/>
    <elseif cond="errCode == 4"/>
      <prompt>An error has occurred and will be transferred to a
        non-OpenScape Contact Center extension.</prompt>
      <goto next="#hppcErrorTransfer"/>
    <elseif cond="errCode == 5"/>

```

```
<prompt>The disconnect component was encountered in
workflow. The call must be terminated.</prompt>

<goto next="#hppcTerminate"/>

<elseif cond="errCode == 6"/>

  <prompt>A transfer component was encountered in
  workflow.</prompt>

  <goto next="#hppcGetTransitNumber"/>

<else/>

  <prompt>An error has occurred in QueryCallStatus
  subdialog. The error number is <value expr="errCode"/>. </
  prompt>

  <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
  <value expr="Call ID"/>. QueryCallStatus failed. Error ID
  is <value expr="errCode"/>.</log>

  <goto next="#hppcMainMenu"/>

</if>

</block>

</form>
```

4.11 QueryQueueStatistics

Der Subdialog 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 den Subdialog Initialize aufrufen, bevor Sie diesen Subdialog aufrufen. Weitere Informationen finden Sie in [Abschnitt 4.8, "Initialize"](#), auf Seite 40.

Syntax

```

<subdialog name="HPPC_QueryQueueStatistics"
srcexpr="GetLink('QueryQueueStatistics.vxml')">
    <!-- Anruf-ID des aktuellen Anrufs. Obligatorisch-->
    <param name="QueueName" expr="m_QueueName"/>
    <param name="BusinessUnitName" expr="m_BusinessUnitName"/>
    <filled>
        <assign name="m_CallsInQ"
expr="GetNumber(HPPC_QueryQueueStatistics.CallsInQ)"/>
        <assign name="m_EstimatedWait"
expr="GetNumber(HPPC_QueryQueueStatistics.EstimatedWait)"/>
        <assign name="m_AverageWait"
expr="GetNumber(HPPC_QueryQueueStatistics.AverageWait)"/>
        <assign name="m_OldestCallInQueue"
expr="GetNumber(HPPC_QueryQueueStatistics.OldestCallInQueue)"/>
        </>
        <assign name="m_ServiceLevel"
expr="GetNumber(HPPC_QueryQueueStatistics.ServiceLevel)"/>
        <assign name="errCode"
expr="GetNumber(HPPC_QueryQueueStatistics.ErrorCode)"/>
    </filled>
</subdialog>

```

Parameter

Name	Typ	Bereich	Beschreibung
QueueName	Eingabe	32 Zeichen	Die Warteschlange, für die die Abfrage durchgeführt wird.
CallsInQueue	Ausgabe	Größer oder gleich 0	Die Anzahl der Anrufe in der angegebenen Warteschlange.
EstimatedWait	Ausgabe	Größer oder gleich 0	Die geschätzte Wartezeit (in Sekunden) für die angegebene Warteschlange.
AverageWait	Ausgabe	Größer oder gleich 0	Die durchschnittliche Wartezeit (in Sekunden) für die angegebene Warteschlange.

Tabelle 11

Parameter für den Subdialog QueryQueueStatistics

Name	Typ	Bereich	Beschreibung
OldestCall InQueue	Ausgabe	Größer oder gleich 0	Die Zeitdauer, die sich der älteste Anruf in der Warteschlange befindet.
ServiceLevel	Ausgabe	0 – 100	Der Service-Level für die angegebene Warteschlange.
BusinessUnit Name (optional)	Eingabe	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 11 Parameter für den Subdialog QueryQueueStatistics

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

- 0** Erfolgreich.
- 306** 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** Der angegebene Business Unit-Name wurde nicht festgelegt oder entspricht keinem der Business Unit-Namen in der Datenbank. (Dieser Fehler ist nur relevant, wenn OpenScape Contact Center als Multi-Tenant-System konfiguriert ist.)
- 905** Die angegebene Warteschlange entspricht keinem der Warteschlangennamen in der Datenbank.
- 914** Der versuchte Vorgang ist fehlgeschlagen.
- 950** VoiceXML ist derzeit nicht aktiviert.
- 951** Der Routing-Server ist nicht verfügbar.
- Sons-tige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Fehlercodes"](#).

Beispiel

```
<form id="hppcQueryQueueStatistics">
  <block>
    <prompt>You query queue statistics for <value
    expr="m_QueueName"/></prompt>
  </block>
  <var name="m_EstimatedWait"/>
  <var name="m_CallsInQ"/>
  <var name="m_AverageWait"/>
  <var name="m_OldestCallInQueue"/>
  <var name="m_ServiceLevel"/>
  <subdialog name="HPPC_QueryQueueStatistics"
  srcexpr="GetLink('QueryQueueStatistics.vxml')">
```

```

<!-- Anruf-ID des aktuellen Anrufs. Obligatorisch-->
<param name="QueueName" expr="m_QueueName"/>
<filled>
  <assign name="m_CallsInQ"
    expr="GetNumber(HPPC_QueryQueueStatistics.CallsInQ)"/>
  <assign name="m_EstimatedWait"
    expr="GetNumber(HPPC_QueryQueueStatistics.EstimatedWait)"/>
  <assign name="m_AverageWait"
    expr="GetNumber(HPPC_QueryQueueStatistics.AverageWait)"/>
  <assign name="m_OldestCallinQueue"
    expr="GetNumber(HPPC_QueryQueueStatistics.OldestCallinQueue)"/>
  <assign name="m_ServiceLevel"
    expr="GetNumber(HPPC_QueryQueueStatistics.ServiceLevel)"/>
  <assign name="errCode"
    expr="GetNumber(HPPC_QueryQueueStatistics.ErrorCode)"/>
</filled>
</subdialog>
<block>
<if cond="errCode == 0">
  <prompt cond="PlayWarnings==1">QueryQueueStatistics
    succeed.</prompt>
  <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
    <value expr="Call_ID"/>: QueryQueueStatistics succeed.
    CallsInQ = <value expr="m_CallsInQ"/>
    EstimatedWait = <value expr="m_EstimatedWait"/>
    AverageWait = <value expr="m_AverageWait"/>
    OldestCallinQueue = <value expr="m_OldestCallinQueue"/>
    ServiceLevel = <value expr="m_ServiceLevel"/>
  </log>
  <prompt>There are <say-as interpret-as="number"><value
    expr="m_CallsInQ"/></say-as> calls in queue. Estimated
    Wait time <value expr="m_EstimatedWait"/> seconds</prompt>
<if cond="m_CallsInQ > 0">
  <prompt>
    Call center is busy, please create a callback.
  </prompt>
  <goto next="#hppcCreateCallback"/>
<else/>
  <prompt>
    Trasferring the call to queue <value expr = "m_QueueName"/>.
  </prompt>

```

```
<goto next="#hppcSetContactData"/>
</if>
<else/>
  <prompt>An error has occurred in QueryQueueStatistics
  subdialog. The error number is <value expr="errCode"/>. </
  prompt>
  <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
  <value expr="Call_ID"/>. QueryQueueStatistics failed.
  Error ID is <value expr="errCode"/>.</log>
  <goto next="#hppcTerminate"/>
</if>
</block>
</form>
```

4.12 QueryRoutingInfo

Der Subdialog QueryRoutingInfo sendet eine Anforderung, um Routing-Informationen für einen Anruf zu ermitteln. Die Abfrage basiert auf Informationen, die vom IVR-System als Teil einer OpenScape Contact Center-Routinganforderung gesammelt und übermittelt werden.

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

Wenn in einem Ablauf eine Komponente Transfer angetroffen wird, so wird der Code 2 zurückgegeben. Das zurückgegebene Transferziel kann je nach 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 den Subdialog Initialize aufrufen, bevor Sie diesen Subdialog aufrufen. Weitere Informationen finden Sie in [Abschnitt 4.8, "Initialize"](#), auf Seite 40.

HINWEIS: Wenn die Kontaktdaten-Werte bei der Routingentscheidung berücksichtigt werden sollen, müssen Sie diese Werte mithilfe des Subdialogs SetContactData vor dem Aufrufen dieses Subdialogs festlegen. Um die ursprünglichen Kontaktdaten-Werte beizubehalten, müssen Sie die Werte vor dem Aufrufen des Subdialogs Enqueue kopieren und zurücksetzen. Weitere Informationen finden Sie in [Abschnitt 4.16, "SetContactData"](#), auf Seite 63.

Syntax

```

<subdialog name="HPPC_QueryRoutingInfo"
srcexpr="GetLink(&apos;QueryRoutingInfo.vxml&apos;)">
    <param name="CallID" expr="Call_ID"/>
<filled>
    <assign name="m_Destination"
expr="HPPC_QueryRoutingInfo.Destination"/>
    <assign name="m_AgentId"
expr="HPPC_QueryRoutingInfo.AgentId"/>
    <assign name="m_AgentWaitTime"
expr="HPPC_QueryRoutingInfo.AgentWaitTime"/>
    <assign name="m_Description"
expr="HPPC_QueryRoutingInfo.Description"/>
    <assign name="m_InitialPriority"
expr="HPPC_QueryRoutingInfo.InitialPriority"/>
    <assign name="m_EstimatedWait"
expr="HPPC_QueryRoutingInfo.EstimatedWait"/>
    <assign name="m_CallsInQ"
expr="HPPC_QueryRoutingInfo.CallsInQ"/>
    <assign name="errCode"
expr="HPPC_QueryRoutingInfo.ErrorCode"/>
</filled>
</subdialog>

```

Parameter

Name	Typ	Bereich	Beschreibung
CallID	Eingabe	18 Zeichen	Die Anruf-ID des Anrufs, für den Sie die entsprechenden Routing-Informationen abrufen wollen.
Destination	Ausgabe	80 Zeichen	Rückgabecode 0 – Die Warteschlange, in die der Anruf eingereicht werden soll. Rückgabecode 1 – Der Anruf sollte getrennt werden. Rückgabecode 2 – Das Ziel, an das der Anruf weitergeleitet werden soll.
AgentID	Ausgabe	8 Zeichen	Gibt die ID des Benutzers zurück, der für die Bearbeitung des Anrufs reserviert werden soll. Wenn keine Agenten-ID zurückgegeben wird, verwenden Sie den Subdialog Enqueue. Hinweis: Nur gültig, wenn der Fehlercode 0 ist.

Tabelle 12

Parameter für den Subdialog QueryRoutingInfo

Name	Typ	Bereich	Beschreibung
AgentWaitTime	Ausgabe	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. Hinweis: Nur gültig, wenn der Fehlercode 0 ist.
Beschreibung	Ausgabe	100 Zeichen	Enthält eine Beschreibung des Anrufs, die in der OpenScape Contact Center Client Desktop-Anwendung angezeigt wird, sodass Benutzer den Anruf identifizieren können. Hinweis: Nur gültig, wenn der Fehlercode 0 ist.
InitialPriority	Ausgabe	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. Hinweis: Nur gültig, wenn der Fehlercode 0 ist.
EstimatedWait	Ausgabe	Größer oder gleich 0	Gibt die geschätzte Wartezeit (in Sekunden) für die angegebene Warteschlange zurück. Hinweis: Nur gültig, wenn der Fehlercode 0 ist.
CallsInQueue	Ausgabe	Größer oder gleich 0	Gibt die Anzahl der Anrufe für die angegebene Warteschlange zurück. Hinweis: Nur gültig, wenn der Fehlercode 0 ist.

Tabelle 12 Parameter für den Subdialog QueryRoutingInfo

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

- 0** Erfolgreich. Gibt an, dass der Anruf mit den zurückgegebenen Informationen in die Warteschlange eingereiht werden muss.
- 1** Erfolgreich. Gibt an, dass der Anruf getrennt werden muss.
- 2** Erfolgreich. Gibt an, dass der Anruf an das zurückgegebene Ziel weitergeleitet werden muss.

- 306** 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** Die CallID wurde nicht gefunden. Sie müssen die Funktion Initialize aufrufen, um die CallID zu beziehen.
- 400** Die Systemressourcen sind niedrig.
- 914** Der versuchte Vorgang ist fehlgeschlagen.
- 950** VoiceXML ist derzeit nicht aktiviert.
- 951** Der Routing-Server ist nicht verfügbar.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Fehlercodes"](#).

Beispiel

Die folgende Beispielfunktion fragt den Routing-Server nach Routing-Informationen ab.

```
<form id="hppcQueryRoutingInfo">
  <var name="m_EstimatedWait"/>
  <var name="m_CallsInQ"/>
  <var name="m_Destination"/>
  <subdialog name="HPPC_QueryRoutingInfo"
    srcexpr="GetLink(&apos;QueryRoutingInfo.vxml&apos;)">
    <!-- Anruf-ID des aktuellen Anrufs. Obligatorisch-->
    <param name="CallID" expr="Call_ID"/>
    <filled>
      <assign name="m_Destination"
        expr="HPPC_QueryRoutingInfo.Destination"/>
      <assign name="m_AgentId"
        expr="HPPC_QueryRoutingInfo.AgentId"/>
      <assign name="m_AgentWaitTime"
        expr="HPPC_QueryRoutingInfo.AgentWaitTime"/>
      <assign name="m_Description"
        expr="HPPC_QueryRoutingInfo.Description"/>
      <assign name="m_InitialPriority"
        expr="HPPC_QueryRoutingInfo.InitialPriority"/>
      <assign name="m_EstimatedWait"
        expr="HPPC_QueryRoutingInfo.EstimatedWait"/>
      <assign name="m_CallsInQ"
        expr="HPPC_QueryRoutingInfo.CallsInQ"/>
      <assign name="errCode"
        expr="HPPC_QueryRoutingInfo.ErrorCode"/>
    </filled>
  </subdialog>
  <block>
    <if cond="errCode == 0">
```

```
<prompt>QueryRoutingInfo succeed.</prompt>
<log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
<value expr="Call_ID"/>: QueryRoutingInfo succeed.
Destination = <value expr="m_Destination"/>
<AgentId = <value expr="m_AgentId"/>
<AgentWaitTime = <value expr="m_AgentWaitTime"/>
<Description = <value expr="m_Description"/>
<InitialPriority = <value expr="m_InitialPriority"/>
<EstimatedWaitTime = <value expr="m_EstimatedWait"/>
<CallsInQ = <value expr="m_CallsInQ"/>
</log>
<goto next="#hppcSetContactData"/>
<else/>
<prompt>QueryRoutingInfo failed. Error ID is <value
expr="errCode"/>. </prompt>
<log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
<value expr="Call_ID"/>. QueryRoutingInfo failed. Error ID
is <value expr="errCode"/>.</log>
<goto next="#hppcTerminate"/>
</if>
</block>
</form>
```

4.13 QuerySystemStatus

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

HINWEIS: Sie müssen den Subdialog Initialize aufrufen, bevor Sie diesen Subdialog aufrufen. Weitere Informationen finden Sie in [Abschnitt 4.8, "Initialize"](#), auf Seite 40.

Syntax

```
<subdialog name="HPPC_QuerySystemStatus"
srcexpr="GetLink('QuerySystemStatus.vxml')">
<filled>
  <assign name="QSSOverallStatus"
expr="HPPC_QuerySystemStatus.OverallStatus"/>
  <assign name="QSSServerState"
expr="HPPC_QuerySystemStatus.ServerState"/>
```

```

<assign name="QSSEntries"
  expr="HPPC_QuerySystemStatus.Entries"/>
<assign name="errCode"
  expr="HPPC_QuerySystemStatus.ErrorCode"/>
</filled>
</subdialog>

```

Parameter

Name	Typ	Bereich	Beschreibung
OverallStatus	Ausgabe	0 oder 1	0 - Gibt an, dass das System nicht betriebsbereit ist 1 - Gibt an, dass das System betriebsbereit ist.
IndividualServer States	Ausgabe	250 Zeichen	Eine Zeichenfolge, die jeden Servernamen und seinen Zustand enthält. Beispiel: server-name1=1; server-name2=2
Entries	Ausgabe	Größer oder gleich 0	Die Anzahl der Server in der Status-Zeichenfolge.

Tabelle 13 Parameter für den Subdialog 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/MLHG-Nummer transferiert werden, die Anrufe an Ausfall-ACD/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.

Verwenden der OpenScape Contact Center VoiceXML-Subdialogs

QuerySystemStatus

- 9 Die Kommunikationsplattform ist überlastet und hat Probleme, die Anfragen des T-Servers auszuführen.
- 10 Der T-Server wird gerade wiederhergestellt.

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

- 0 Erfolgreich.
- 306 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 Die Systemressourcen sind niedrig.
- 950 VoiceXML ist derzeit nicht aktiviert.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Fehlercodes"](#).

Beispiel

```
<form id="hppcQuerySystemStatus">
  <var name="QSSOverallStatus"/>
  <var name="QSSServerState"/>
  <var name="QSSEntries"/>
  <subdialog name="HPPC_QuerySystemStatus"
    srcexpr="GetLink('QuerySystemStatus.vxml')">
    <filled>
      <assign name="QSSOverallStatus"
        expr="HPPC_QuerySystemStatus.OverallStatus"/>
      <assign name="QSSServerState"
        expr="HPPC_QuerySystemStatus.ServerState"/>
      <assign name="QSSEntries"
        expr="HPPC_QuerySystemStatus.Entries"/>
      <assign name="errCode"
        expr="HPPC_QuerySystemStatus.ErrorCode"/>
    </filled>
  </subdialog>
  <block>
    <if cond="errCode == 0">
      <prompt cond="PlayWarnings==1">QuerySystemStatus succeed
      </prompt>
      <prompt> The overall status is <value
        expr="QSSOverallStatus"/>. There are <value
        expr="QSSEntries"/> entries in the returned status data.
        see log for details.</prompt>
    </if>
  </block>
</form>
```

```

<log>HPPC Call ID is <value expr="Call_ID"/>:
QuerySystemStatus succeed. The overall status is <value
expr="QSSOverallStatus"/>. There are <value
expr="QSSEntries"/> entries in the returned status data as
follows <value expr="QSSServerState"/>.</log>

<else/>

<prompt>An error has occurred in QuerySystemStatus
subdialog. The error number is <value expr="errCode"/>. </
prompt>

<log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
<value expr="Call_ID"/>. QuerySystemStatus failed. Error
ID is <value expr="errCode"/>.</log>

</if>

<goto next="#hppcMainMenu"/>

</block>

</form>

```

4.14 ReleaseTransitNumber

Über den Subdialog ReleaseTransitNumber kann eine Anwendung anfordern, dass OpenScape Contact Center eine Transitnummer für die Verwendung durch einen anderen Anruf freigibt. Dieser Subdialog 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 den Subdialog ReleaseTransitNumber verwenden, um den T-Server zu informieren, dass die Transitnummer nicht länger benötigt wird. Die Anwendung muss den Subdialog GetTransitNumber erneut aufrufen, um eine neue Transitnummer anzufordern, bevor versucht wird, den Anruf an OpenScape Contact Center weiterzuleiten.

Syntax

```

<subdialog name="HPPC_ReleaseTransitNumber"
srcexpr="GetLink(&apos;ReleaseTransitNumber.vxml&apos;)">

<!-- Anruf-ID des aktuellen Anrufs. Obligatorisch-->
  <param name="CallID" expr="Call_ID"/>

<filled>
  <assign name="errCode"
    expr="HPPC_ReleaseTransitNumber.ErrorCode"/>
</filled>

</subdialog>

```

Parameter

Name	Typ	Bereich	Beschreibung
CallID	Eingabe	18 Zeichen	Die Anruf-ID für den aktuellen Anruf.

Tabelle 14

Parameter für den Subdialog ReleaseTransitNumber

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

- 0** Erfolgreich.
- 306** 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** Die CallID wurde nicht gefunden. Sie müssen die Funktion Initialize aufrufen, um die CallID zu beziehen.
- 908** Der T-Server ist nicht verfügbar.
- 914** Der versuchte Vorgang ist fehlgeschlagen.
- 950** VoiceXML ist derzeit nicht aktiviert.
- Sons-tige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Fehlercodes"](#).

Beispiel

```
<form id="hppcReleaseTransitNumber">
  <subdialog name="HPPC_ReleaseTransitNumber"
    srcexpr="GetLink(&apos;ReleaseTransitNumber.vxml&apos;)">
    <!-- Anruf-ID des aktuellen Anrufs. Obligatorisch-->
    <param name="CallID" expr="Call_ID"/>
    <filled>
      <assign name="errCode"
        expr="HPPC_ReleaseTransitNumber.ErrorCode"/>
    </filled>
  </subdialog>
  <block>
    <if cond="errCode == 0">
      <prompt>ReleaseTransitNumber succeed.</prompt>
      <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
        <value expr="Call_ID"/>: ReleaseTransitNumber succeed.</log>
      <goto next="#hppcTerminate"/>
    <else/>
      <prompt>ReleaseTransitNumber failed. Error ID is <value
        expr="errCode"/>. </prompt>
```

```

<log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
<value expr="Call_ID"/>. ReleaseTransitNumber failed.
Error ID is <value expr="errCode"/>.</log>

<goto next="#hppcTerminate"/>

</if>

</block>

</form>

```

4.15 SetBusinessUnit

HINWEIS: Der Subdialog SetBusinessUnit wird nur in einer Multi-Tenant-Umgebung unterstützt.

Der Subdialog SetBusinessUnit ruft die Business Unit für den definierten Anruf ab.

HINWEIS: Der Subdialog SetBusinessUnit kann pro IVR-Anruf nur einmal aufgerufen werden.

HINWEIS: Sie müssen den Subdialog Initialize aufrufen, bevor Sie diesen Subdialog aufrufen. Weitere Informationen finden Sie in [Abschnitt 4.8, "Initialize"](#), auf Seite 40.

Syntax

```

<subdialog name="HPPC_SetBusinessUnit"
srcexpr="GetLink('SetBusinessUnit.vxml')">
    <param name="CallID" expr="Call_ID"/>
    <param name="BusinessUnitName" expr="m_BusinessUnitName"/>
    <filled>
        <assign name="errCode"
expr="HPPC_SetBusinessUnit.ErrorCode"/>
    </filled>
</subdialog>

```

Parameter

Name	Typ	Bereich	Beschreibung
CallID	Eingabe	18 Zeichen	Die Anruf-ID für den Anruf, für den Sie die Business Unit festlegen wollen.
BusinessUnit Name	Eingabe	32 Zeichen	Der Name der Business Unit für den definierten Anruf.

Tabelle 15 Parameter für den Subdialog SetBusinessUnit

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

- 0** Erfolgreich.
- 306** 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.
- 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.)
- 908** Der T-Server ist nicht verfügbar.
- 914** Der versuchte VXML-Vorgang ist fehlgeschlagen.
- 950** VoiceXML ist derzeit nicht aktiviert.
- 955** Das Leistungsmerkmal Multi-Tenancy ist nicht lizenziert.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Fehlercodes"](#).

Beispiel

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

```
<form id="hppcSetBusinessUnit">
  <subdialog name="HPPC_SetBusinessUnit"
    srcexpr="GetLink('SetBusinessUnit.vxml')">
    <param name="CallID" expr="Call_ID"/>
    <param name="BusinessUnitName" expr="m_BusinessUnitName"/>
    <filled>
      <assign name="errCode"
        expr="HPPC_SetBusinessUnit.ErrorCode"/>
    </filled>
  </subdialog>
  <block>
    <if cond="errCode == 0">
      <prompt cond="PlayWarnings==1">Set Business Unit
        succeed.</prompt>
    </if>
  </block>
</form>
```

```

<log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
<value expr="Call_ID"/>. Set Business Unit <value expr =
  "m_BusinessUnitName"/> succeed.</log>

<prompt>Your call is assigned to business unit <value expr
  = "m_BusinessUnitName"/>.</prompt>

<else/>

  <prompt>An error has occurred in Set Business Unit
    subdialog. The error number is <value expr="errCode"/>. </
    prompt>

  <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
    <value expr="Call_ID"/>. Set Business Unit failed. Error
    ID is <value expr="errCode"/>.</log>

</if>

<goto next="#hppcMainMenu"/>

</block>

</form>

```

4.16 SetContactData

Der Subdialog 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 Einreihung des Anrufs festlegen.

HINWEIS: Sie müssen den Subdialog Initialize aufrufen, bevor Sie diesen Subdialog aufrufen. Weitere Informationen finden Sie in [Abschnitt 4.8, "Initialize"](#), auf Seite 40.

Syntax

```

<subdialog name="HPPC_SetContactData"
srcexpr="GetLink(&apos;SetContactData.vxml&apos;)">
  <!-- Anruf-ID des aktuellen Anrufs. Obligatorisch-->
  <param name="CallID" expr="Call_ID"/>
  <param name="Key" expr="&apos;Name&apos;"/>
  <param name="Value" expr="&apos;John Doe&apos;"/>
  <filled>
    <assign name="errCode" expr="HPPC_SetContactData.ErrorCode"/>
  </filled>
</subdialog>

```

Parameter

Name	Typ	Bereich	Beschreibung
CallID	Eingabe	18 Zeichen	Die Anruf-ID für den Anruf, für den Sie die Kontaktdaten festlegen wollen.
Key	Eingabe	32 Zeichen	Der Schlüsselname für den Kontaktdatenwert.
Value	Eingabe	128 Zeichen	Der zu setzende Wert.

Tabelle 16 Parameter für den Subdialog SetContactData

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

- 0** Erfolgreich.
- 306** 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** Die CallID wurde nicht gefunden. Sie müssen die Funktion Initialize aufrufen, um die CallID zu beziehen.
- 908** Der T-Server ist nicht verfügbar. Der Versuch des Routing-Servers, Verbindung zum T-Server herzustellen, ist aus einem unbekannten Grund fehlgeschlagen.
- 914** Der versuchte Vorgang ist fehlgeschlagen.
- 950** VoiceXML ist derzeit nicht aktiviert.
- 951** Der Routing-Server ist nicht verfügbar.
- Sonstige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Fehlercodes"](#).

Beispiel

```
<form id="hppcSetContactData">
  <subdialog name="HPPC_SetContactData"
    srcexpr="GetLink (&apos;SetContactData.vxml&apos;)">
    <!-- Anruf-ID des aktuellen Anrufs. Obligatorisch-->
    <param name="CallID" expr="Call_ID"/>
    <param name="Key" expr="&apos;Name&apos;"/>
    <param name="Value" expr="&apos;John Doe&apos;"/>
    <filled>
      <assign name="errCode"
        expr="HPPC_SetContactData.ErrorCode"/>
    </filled>
  </subdialog>
</block>
<if cond="errCode == 0">
  <prompt>SetContactData succeed.</prompt>
```

```

<log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
<value expr="Call_ID"/>. SetContactData succeed.</log>
<goto next="#hppcEnqueue"/>
<else/>
  <prompt>SetContactData failed. Error ID is <value
  expr="errCode"/>. </prompt>
  <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
  <value expr="Call_ID"/>. SetContactData failed. Error ID
  is <value expr="errCode"/>.</log>
  <goto next="#hppcTerminate"/>
</if>
</block>
</form>

```

4.17 SetDisplay

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

HINWEIS: Dieser Subdialog ist nur auf der OpenScape 4000- oder HiPath 4000-Kommunikationsplattform verfügbar. Sie müssen den Subdialog Initialize aufrufen, bevor Sie diesen Subdialog aufrufen. Weitere Informationen finden Sie in [Abschnitt 4.8, "Initialize", auf Seite 40](#).

Syntax

```

<subdialog name="HPPC_SetDisplay"
srcexpr="GetLink('SetDisplay.vxml')">
  <param name="CallID" expr="Call_ID"/>
  <param name="DisplayInfo" expr="'Hello this is John'"/>
<filled>
  <assign name="errCode" expr="HPPC_SetDisplay.ErrorCode"/>
</filled>
</subdialog>

```

Parameter

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

Tabelle 17

Parameter für den Subdialog SetDisplay

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

- 0** Erfolgreich.
- 306** 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** Die CallID wurde nicht gefunden. Sie müssen die Funktion Initialize aufrufen, um die CallID zu beziehen.
- 908** Der T-Server ist nicht verfügbar.
- 914** Der versuchte Vorgang ist fehlgeschlagen.
- 950** VoiceXML ist derzeit nicht aktiviert.
- Sons-tige** Alle sonstigen Codes weisen auf einen Fehler hin. Weitere Informationen zu einem bestimmten Code finden Sie unter [Kapitel 5, "Fehlercodes"](#).

Beispiel

```
<form id="hppcSetDisplay">
  <subdialog name="HPPC_SetDisplay"
    srcexpr="GetLink('SetDisplay.vxml')">
    <param name="CallID" expr="Call_ID"/>
    <param name="DisplayInfo" expr="'Hello this is John'"/>
    <filled>
      <assign name="errCode" expr="HPPC_SetDisplay.ErrorCode"/>
    </filled>
  </subdialog>
  <block>
    <if cond="errCode == 0">
      <prompt>SetDisplay succeed </prompt>
      <log>HPPC Call ID is <value expr="Call_ID"/>: SetDisplay
        succeed.</log>
      <goto next="#hppcGetTransitNumber"/>
    <else/>
      <prompt>SetDisplay failed. Error ID is <value
        expr="errCode"/>. </prompt>
      <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
        <value expr="Call_ID"/>. SetDisplay failed. Error ID is
        <value expr="errCode"/>.</log>
      <goto next="#hppcTerminate"/>
    </if>
  </block>
</form>
```

4.18 Terminate

Der Subdialog Terminate führt das Beenden und Bereinigen durch. Er muss aufgerufen werden, wenn das IVR-System die VoiceXML-Funktionalität in OpenScape Contact Center nicht mehr nutzt.

HINWEIS: Wenn der Subdialog GetTransitNumber aufgerufen wurde, müssen Sie zuerst den Subdialog ReleaseTransitNumber aufrufen, bevor Sie diesen Subdialog aufrufen.

Syntax

```
<subdialog name="HPPC_Terminate"
srcexpr="GetLink(&apos;Terminate.vxml&apos;)">
<!-- Anruf-ID des aktuellen Anrufs. Obligatorisch-->
    <param name="CallID" expr="Call_ID"/>
<filled>
    <assign name="errCode" expr="HPPC_Terminate.ErrorCode"/>
</filled>
</subdialog>
```

Parameter

Name	Typ	Bereich	Beschreibung
CallID	Eingabe	18 Zeichen	Die Anruf-ID für den Anruf, den Sie beenden wollen.
TermType	Eingabe	0, 1 oder 2	0 – Gibt an, dass der Anruf wegen eines internen Problems beendet wurde. 1 – Gibt an, dass der Anruf außerhalb des Einzugsbereichs weitergeleitet wurde. 2 – Gibt an, dass der Anruf abgebrochen wurde.

Tabelle 18 Parameter für den Subdialog Terminate

Fehlercodes

Die gängigsten Rückgabecodes dieses Subdialogs sind:

- 0** Erfolgreich.
- 306** 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** Die CallID wurde nicht gefunden. Sie müssen die Funktion Initialize aufrufen, um die CallID zu beziehen.
- 400** Die Systemressourcen sind niedrig.
- 914** Der versuchte Vorgang ist fehlgeschlagen.

–950 VoiceXML ist derzeit nicht aktiviert.

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

Beispiel

```
<form id="hppcTerminate">
  <subdialog name="HPPC_Terminate"
    srcexpr="GetLink(&apos;Terminate.vxml&apos;)">
    <!-- Anruf-ID des aktuellen Anrufs. Obligatorisch-->
    <param name="CallID" expr="Call_ID"/>
    <filled>
      <assign name="errCode" expr="HPPC_Terminate.ErrorCode"/>
    </filled>
  </subdialog>
  <block>
    <if cond="errCode == 0">
      <prompt>Terminate succeed.</prompt>
      <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
        <value expr="Call_ID"/>: Terminate succeed.</log>
    <else/>
      <prompt>Terminate failed. Error ID is <value
        expr="errCode"/>.</prompt>
      <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
        <value expr="Call_ID"/>. Terminate failed. Error ID is
        <value expr="errCode"/>.</log>
    </if>
  </block>
  <block>
    <prompt>Good bye.</prompt>
    <log>HPPC IVRID=<value expr="ivrCallID"/>, HPPC Call ID is
      <value expr="Call_ID"/>: IVR Call is finished</log>
  </block>
  • </form>
```

5 Fehlercodes

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

Code	Beschreibung
0	Erfolgreich. QueryRoutingInfo — Gibt an, dass der Anruf mit den zurückgegebenen Informationen in die Warteschlange eingereiht werden muss.
1	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.
2	QueryCallStatus – Gibt an, dass der Anruf im Zustand Wartend ist. Leiten Sie in diesem Fall den Anruf an die Transitnummer weiter. QueryRoutingInfo — Gibt an, dass der Anruf an das zurückgegebene Ziel weitergeleitet werden muss.
3	QueryCallStatus – Gibt an, dass der Anruf im Zustand Nicht angenommen ist. Leiten Sie in diesem Fall den Anruf an die Transitnummer oder an eine andere Zeitüberschreitungs-Nebenstelle weiter.
4	QueryCallStatus – Gibt an, dass ein Fehler aufgetreten ist. Leiten Sie in diesem Fall den Anruf an eine Nicht-OpenScape Contact Center Nebenstelle weiter.
5	QueryCallStatus – Gibt an, dass der Anruf getrennt werden muss.
6	QueryCallStatus – Der Anruf muss an die OpenScape Contact Center-Transitnummer weitergeleitet werden.
0009	Die Anfrage konnte aufgrund eines internen Fehlers in einem Webkomponenten-Servlet nicht ausgeführt werden.
1003	Die Anfrage konnte aufgrund eines Socket-Kommunikationsfehlers zwischen dem IIS-Server und dem Web Interaction-Server nicht ausgeführt werden.
–304	Initialize wurde vor dem Starten dieser Funktion nicht aufgerufen. Sie müssen Initialize am Anfang Ihres Skripts aufrufen.
–305	Dieser Code entspricht einem unbekannten Fehler (z. B. Benutzerfehler oder System ist instabil).
–306	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.
–308	Die CallID wurde nicht gefunden. Sie müssen die Funktion Initialize aufrufen, um die CallID zu beziehen.
–311	Die Funktion unterstützt keine nicht überwachten IVR-Anrufe.

Tabelle 19

Fehlercodes

Code	Beschreibung
–312	Falsche Abfolge von Operationen, zum Beispiel SetCallInfo wird aufgerufen nach QueryCallInfo).
–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	Der angegebene Business Unit-Name wurde nicht festgelegt oder entspricht keinem der Business Unit-Namen in der Datenbank. (Dieser Fehler ist nur relevant, wenn OpenScape Contact Center als Multi-Tenant-System konfiguriert ist.)
–400	Die Systemressourcen sind niedrig.
–701	Die Anforderung hat die Zeit überschritten, bevor die Funktion abgeschlossen werden konnte. Dies könnte auf Netzwerkprobleme oder einen ausgelasteten Server hinweisen.
–801	Die Verbindung zu einem der OpenScape Contact Center Server konnte nicht hergestellt werden. 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 der angegebenen Anruf-ID war kein Anruf verbunden. Stellen Sie sicher, dass alle IVR-Nebenstellen in der Datenbank enthalten sind.
–904	Die angegebene Agenten-ID stimmt nicht mit der ID eines Agenten in der Datenbank überein.
–905	Die angegebene Warteschlange entspricht keinem der Warteschlangennamen in 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	Das angegebene Gerät entspricht keinem Gerät in der Datenbank.
–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 Warteschlange und weitere Routing-Informationen, die auf der im System konfigurierten Routing-Option basieren, konnten nicht abgerufen werden.
–916	Der Routing-Server hat einen Fehler in der Ablaufausführung festgestellt.
–917	Fehler in der TAPI-Übersetzung.
–927	Alle angemeldeten Transitnummern sind besetzt.
–928	Dieser CallID wurde keine Transitnummer zugewiesen.
–929	Es wurden keine Transitnummern konfiguriert.

Tabelle 19

Fehlercodes

Code	Beschreibung
–950	VoiceXML ist derzeit nicht aktiviert.
–951	Der Routing-Server ist nicht verfügbar.
–953	Die Business Unit wurde für diesen Anruf nicht definiert. (Dieser Fehler ist nur relevant, wenn OpenScape Contact Center als Multi-Tenant-System konfiguriert ist.)
–955	Das Leistungsmerkmal Multi-Tenancy ist nicht lizenziert.
–1006	Sie sind dabei, einen doppelten Callback einzurichten.
–1021	Ungültige Callback-ID.
–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 befindet sich auf der Liste Ausgeschlossene Nummern.
–1040	Alle Zeitpläne sind bereits abgelaufen.
–1045	Der Name der Callback-Warteschlange ist ungültig.
–1047	Der Zeitplan enthält ein Datum, das zu weit in der Zukunft liegt.
53006	Der Web Interaction-Server kann keine Verbindung zum T-Server herstellen.
53007	Der Web Interaction-Server kann keine Verbindung zum Routing-Server herstellen.
53012	Es ist nicht genügend Arbeitsspeicher zur Durchführung der angeforderten Aktion vorhanden. Wenden Sie sich an Ihren Servicevertreter.
53015	Der Web Interaction-Server hat die Verbindung zum Unternehmens-Webserver verloren. Wenden Sie sich an Ihren Servicevertreter.
53018	Der Web Interaction-Server hat die Verbindung zum Unternehmens-Webserver verloren. Wenden Sie sich an Ihren Servicevertreter.
53028	Der Web Interaction-Server konnte die angeforderte Aktion nicht ausführen. Wenden Sie sich an Ihren Servicevertreter.
53503	Der Web Interaction-Server konnte die angeforderte Aktion nicht ausführen. Wenden Sie sich an Ihren Servicevertreter.
53504	Der Web Interaction-Server ist mit der aktuellen Version der Anwendung nicht kompatibel. Wenden Sie sich an Ihren Servicevertreter.
53508	Der Web Interaction-Server unterstützt die angeforderte Aktion nicht.
53510	VoiceXML ist derzeit nicht aktiviert.
Sonstige	<p>Alle sonstigen negativen Codes weisen auf einen Fehler hin. Wenn ein anderer Code zurückgegeben wird:</p> <ul style="list-style-type: none"> • 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. • Stellen Sie sicher, dass alle Ressourcen der Kommunikationsplattform korrekt in die Betriebsversion der OpenScape Contact Center-Datenbank eingegeben wurden. • Wenden Sie sich an die nächste Support-Stelle.

Tabelle 19

Fehlercodes

Stichwörter

D

Dokumentation

- Feedback geben 6
- Formatierungskonventionen 5
- Zielgruppe 5

F

- Fehlercodes 69
- Funktion CreateCallback 23
- Funktion DeleteCallback 27
- Funktion Dequeue 29
- Funktion Enqueue 31
- Funktion GetBusinessUnit 34
- Funktion GetContactData 36
- Funktion GetTransitNumber 38
- Funktion Initialize 40
- Funktion QueryAgentStatus 42
- Funktion QueryCallStatus 45
- Funktion QueryQueueStatistics 48
- Funktion QueryRoutingInfo 52
- Funktion QuerySystemStatus 56
- Funktion ReleaseTransitNumber 59
- Funktion SetBusinessUnit 61
- Funktion SetContactData 63
- Funktion SetDisplay 65

H

Halten im IVR

- IVR-Skripte schreiben 15
- Konfiguration 9

Halten in Warteschlange

- IVR-Skripte schreiben 19
- Konfiguration 11

I

Installation 7

IVR-API-Funktionen

- CreateCallback 23
- DeleteCallback 27
- Dequeue 29
- Enqueue 31
- GetBusinessUnit 34
- GetContactData 36
- GetTransitNumber 38
- Initialize 40
- QueryAgentStatus 42
- QueryCallStatus 45

QueryQueueStatistics 48

QueryRoutingInfo 52

QuerySystemStatus 56

ReleaseTransitNumber 59

SetBusinessUnit 61

SetContactData 63

SetDisplay 65

IVR-Skripte

Halten im IVR 15

Halten in Warteschlange 19

schreiben 13

M

Multi-Tenant-Umgebung 21

Funktion GetBusinessUnit 34

Funktion SetBusinessUnit 61

Subdialog GetBusinessUnit 34

Subdialog SetBusinessUnit 61

R

Rückgabecodes 69

S

Subdialog CreateCallback 23

Subdialog DeleteCallback 27

Subdialog Dequeue 29

Subdialog Enqueue 31

Subdialog GetBusinessUnit 34

Subdialog GetContactData 36

Subdialog GetTransitNumber 38

Subdialog Initialize 40

Subdialog QueryAgentStatus 42

Subdialog QueryCallStatus 45

Subdialog QueryQueueStatistics 48

Subdialog QueryRoutingInfo 52

Subdialog QuerySystemStatus 56

Subdialog ReleaseTransitNumber 59

Subdialog SetBusinessUnit 61

Subdialog SetContactData 63

Subdialog SetDisplay 65

Subdialog Terminate 67

Subdialoge 13

U

Übersicht 7

V

VoiceXML-Subdialoge

Stichwörter

CreateCallback 23
DeleteCallback 27
Dequeue 29
Enqueue 31
GetBusinessUnit 34
GetContactData 36
GetTransitNumber 38
Initialize 40
QueryAgentStatus 42
QueryCallStatus 45
QueryQueueStatistics 48
QueryRoutingInfo 52
QuerySystemStatus 56
ReleaseTransitNumber 59
SetBusinessUnit 61
SetContactData 63
SetDisplay 65
Terminate 67
Vorteile 8

