

# Einschränkungen für UCCX-Repository

## Inhalt

[Einführung](#)

[Hintergrundinformationen](#)

[Problem](#)

[Lösung](#)

[Schätzung der aktuellen Größe der Repository-Daten](#)

[Bestimmen der Anzahl der Zeilen in den Verzeichnistabellen des Repository](#)

[Bestimmen Sie die Anzahl der Seiten, die von Repository-Ordner tabellen verwendet werden.](#)

[Bestimmen Sie die Anzahl der Seiten, die von den Tabellen für Repository-Elementdateien verwendet werden.](#)

[Berechnen](#)

[Aufforderungen](#)

## Einführung

Dieses Dokument beschreibt ein Problem, das beim Upgrade von Cisco Unified Contact Center Express (UCCX)-Systemen auf Version 8 und höher sowie beim Hochladen einer großen Anzahl von Repository-Elementen auf das System oder beim Uploaden einer großen Anzahl von Repository-Elementen auf das System in Version 8 oder höher auftritt.

## Hintergrundinformationen

UCCX-Versionen 7.x und höher verwenden Microsoft SQL (MSSQL) als Datenbankmodul. MSSQL unterscheidet in Bezug auf die Datenspeicherung nicht zwischen Daten verschiedener Typen. Wenn Daten in einer 3-GB-Datenbank gespeichert werden, speichert MSSQL alle Daten unabhängig vom Typ in einem 3-GB-Block.

Im Gegensatz dazu unterscheidet Informix, die in UCCX Version 8.0 und höher verwendete Datenbank-Engine, beim Speichern auf der Festplatte zwischen Daten verschiedener Typen. Typische Datenbankdaten (z. B. Zeichenfolgen, Zeichen und Integer.) werden in einem Festplattenchunk gespeichert, der der Datenbank gewidmet ist, während BLOB-Daten (Binary Large Object), sofern sie in den Datensätzen der Datenbanktabelle vorhanden sind, in einem separaten Bereich der Festplatte gespeichert werden, der als Leerzeichen bezeichnet wird. Ein Leerzeichen ist eine logische Einheit, die aus einem oder mehreren Festplatten-Chunks besteht, die BLOB-Daten speichern. Informix speichert traditionelle Daten und BLOB-Daten separat, um die Leistung beim Lesen und Schreiben von BLOB-Daten von und auf die Datenbank und den Datenträger zu erhöhen. Wenn eine Datenbank erstellt wird, die BLOB-Daten enthält, muss der Administrator die Größe der Festplatten-Chunks für die Datenbank (um herkömmliche Daten zu speichern) und die Größe des Leerzeichens einzeln angeben.

Für Datenspeicherungsmechanismen platziert MSSQL alle Daten in einer einzigen Gruppe der Größe N, während Informix die Speicherung dieser Daten in zwei Gruppen aufteilt: ein Eimer für die Kontextinformationen über BLOB-Daten der Größe X und ein weiterer Eimer für die BLOB-Objekte der Größe Y.

In UCCX kann der Administrator Repository-Elemente hochladen, die aus Prompts, Documents, Grammars und Scripts bestehen. Der Inhalt dieser Elemente wird in den entsprechenden Datenbanktabellen als BLOB-Daten und Kontextinformationen über sie gespeichert, z. B. Dateiname, Ordner, letzte geänderte Uhrzeit, zuletzt geänderter Benutzer, Länge und Prüfsumme.

Repository-Elemente werden in der UCCX-Datenbank **db\_cra\_repository** gespeichert. In UCCX-Versionen 7.x und früher, die MSSQL verwenden, ist das **db\_cra\_repository** 3 GB groß und enthält sowohl die Kontextinformationen als auch die BLOB-Informationen. In UCCX-Versionen 8.0 und höher, die Informix verwenden, ist der an das **db\_cra\_repository** angehängte Datenspeicher 10,2 MB groß und speichert nur die Kontextinformationen zu den Repository-Elementen. Der Inhalt der Repository-Objekte wird im BLOB-Format in einem Leerzeichen mit der Bezeichnung **uccx\_sbospace** gespeichert. In UCCX-Versionen 8.0 und höher beträgt der **uccx\_sbospace** 3 GB.

Die Ausgabe von **show uccx dbserver disk** auf einem UCCX Version 8.0+-Server zeigt die Unterscheidung zwischen diesen beiden Datenspeichern:

```
admin:show uccx dbserver disk
```

SNO.	DATABASE NAME	TOTAL SIZE (MB)	USED SIZE (MB)	FREE SIZE (MB)	PERCENT FREE
1	rootdbs	358.4	59.5	298.9	83%
2	log_dbs	317.4	307.3	10.1	3%
3	db_cra	512.0	17.3	494.7	96%
4	db_hist	24508.6	6661.2	27847.2	90%
5	db_cra_repository	10.2	3.4	6.9	67%
6	db_trascat	512.0	3.3	508.7	99%
7	temp_uccx	1572.9	0.1	1572.7	99%
8	uccx_sbospace	3145.7	2988.1	157.6	5%
9	uccx_er	204.8	0.1	204.7	99%
10	uccx_ersb	1572.9	1494.1	78.8	5%
11	sadmin	102.4	4.3	98.1	95%

Abhängig von der Mischung von Daten in der MSSQL-Datenbank ist es möglich, dass die Größe der in der MSSQL-Datenbank gespeicherten BLOB-Daten die definierte Größe des in Informix enthaltenen Speicherplatzes überschreitet, wenn eine Migration oder ein Upgrade versucht wird. Ebenso ist es möglich, dass die Kontextinformationen über die in der MSSQL-Datenbank gespeicherten BLOB-Daten die vom Administrator festgelegte Größe für diese Daten im Informix-DatenbankchUNK überschreiten.

In diesem Fall schlägt das Upgrade oder die Migration von UCCX Version 7.x auf UCCX Version 8.x fehl, da entweder das **db\_cra\_repository** oder der **uccx\_sbospace** nicht groß genug sind, um die gleichen Informationen zu speichern, die in MSSQL gespeichert wurden. Dies ist in der Regel ein Problem in einem UCCX-System, das eine große Anzahl von Aufforderungen enthält. Kontextbezogene Prompt- und BLOB-Daten müssen das **db\_cra\_repository** und das **uccx\_sbospace** mit Dokumenten, Grammatiken und Scripts gemeinsam nutzen, aber diese anderen Repository-Typen sind in der Regel klein in Größe und Anzahl.

Betrachten Sie beispielsweise ein UCCX-System der Version 7.x mit Zehntausenden von

Aufforderungen, die jeweils nur wenige Sekunden Audio enthalten. In der UCCX-Version 7.x, die MSSQL verwendet, werden der Inhalt und die Kontextinformationen zur Aufforderung im selben 3-GB-Textbaustein gespeichert. Da es viele kleine Eingabeaufforderungen gibt, speichert die Datenbank möglicherweise 50 MB kontextbezogene Informationen zu den Aufforderungen, aber nur 2 GB BLOB-Daten, die das Audio der Aufforderungen darstellen. Daher belegen die Prompts im Repository etwas mehr als 2 GB der 3-GB-Grenze, die bei der Datenbankerstellung festgelegt wurde.

Wenn Sie versuchen, dieses System auf UCCX Version 8.x und Informix zu migrieren, schlägt die Migration fehl, weil die 50 MB an Kontextinformationen die 10,2 MB-Grenze des **db\_cra\_repository** überschreiten, obwohl der 2 GB Prompt Content weit unter die Grenze des **uccx\_sbospace** passt.

Umgekehrt sollten Sie ein UCCX 7.x-System mit weniger, aber immer noch vielen, langen Eingabeaufforderungen in Betracht ziehen. Bei weniger Eingabeaufforderungen, aber mit einer größeren Anzahl von Aufforderungen ist das Verhältnis von auffordertem Inhalt zu kontextbezogenen Informationen unterschiedlich. In UCCX Version 7.x und MSSQL kann der Prompt-Inhalt bis zu 2,8 GB des **db\_cra\_repository** beanspruchen, und die Kontextinformationen 3 MB. Dieses System wird erfolgreich aktualisiert, da die 3 MB in das **db\_cra\_repository** passt und die 2,8 GB in den zugewiesenen **uccx\_sbospace** passt.

Wenn Sie versuchen, auf UCCX Version 8.x oder höher zu migrieren, überschreiten die Kontextdaten der auf das UCCX Version 7.x oder frühere System hochgeladenen Aufforderungen normalerweise die Größenbeschränkung von **db\_cra\_repository**, bevor der Aufforderungsinhalt die Größenbeschränkung des **uccx\_sbospace** überschreitet. Zusätzlich ist der tatsächliche freie Speicherplatz für benutzerdefinierte Repository-Objekte 6,9 MB, da die Standardkonfiguration 3,4 MB des **db\_cra\_repository** belegt.

## Problem

Wenn Sie versuchen, neue Repository-Elemente (Dokumente, Grammmare, Aufforderungen, Skripte) auf ein UCCX-System hochzuladen, das Version 8 oder höher ausführt, erhalten Sie die folgende Fehlermeldung:

```
The files uploaded are not valid or not structured
according to languages. Please check the help
documentation for more details.
```

Durch die Migration von UCCX-Version 7.0(2) und früher auf Version 8.0 und höher werden das Betriebssystem und die Datenbankengine geändert, auf der die Anwendung ausgeführt wird. Das in UCCX Version 8.0 und höher verwendete Datenbankmodul speichert Daten anders als die in UCCX Version 7.x und höher gespeicherten Daten. Dies hat Auswirkungen auf die Migration von UCCX, da Datenbanken mit großen Datensätzen in UCCX-Version 7.x möglicherweise nicht ordnungsgemäß auf UCCX-Version 8.x migriert werden.

## Lösung

Bevor Sie auf UCCX Version 8.x migrieren, schätzen Sie die Menge des **db\_cra\_repository** und **uccx\_sbospace**, die zum Speichern der aktuellen Repository-Objekte im UCCX Version 7.x-System erforderlich ist, um ein künftiges Wachstum aufzunehmen.

## Schätzung der aktuellen Größe der Repository-Daten

Bestimmen Sie zunächst die Anzahl der Zeilen in jeder der Repository-Tabellen, die Informationen zu Repository-Elementen und -Ordnern enthalten.

### Bestimmen der Anzahl der Zeilen in den Verzeichnistabellen des Repository

Verwenden Sie Microsoft SQL Query Analyzer, um die Anzahl der Zeilen aus den Ordertabellen des Repository mit folgenden Befehlen aufzuzeichnen:

- **SELECT COUNT(\*) FROM documentFoldertbl**
- **SELECT COUNT(\*) FROM grammarsfoldertbl**
- **AUSWÄHLEN VON COUNT(\*) AUS promptfoldertbl**
- **SELECT COUNT(\*) VON scriptsfoldertbl**

### Bestimmen Sie die Anzahl der Seiten, die von Repository-Ordertabellen verwendet werden.

Informix berücksichtigt die Größe der Festplatte in Seiten. Bestimmen Sie die Anzahl der Seiten, die von den Inhalten der Repository-Ordertabellen mit dieser Formel belegt werden, und ersetzen Sie die **Anzahl der Zeilen** durch die Zählungen, die aus den zuvor genannten Befehlen stammen. Berechnen Sie diese Formel für jede Tabelle, und fügen Sie die Anzahl der Seiten hinzu. Es ist nicht möglich, die Anzahl der Seiten genau zu bestimmen, wenn zuerst die Anzahl der Zeilen aus jeder Tabelle hinzugefügt wird, und dann wird das Ergebnis der Formel berechnet.

- **# Pages documentfoldertbl = Anzahl der Zeilen in documentfoldertbl / (2020 / (180 + 4))**
- **# Pages grammarsfoldertbl = Anzahl der Zeilen in grammarsfoldertbl / (2020 / (180 + 4))**
- **# Seiten promptsfoldertbl = Anzahl der Zeilen in promptfoldertbl / (2020 / (180 + 4))**
- **# Pages scriptsfoldertbl = Anzahl der Zeilen in scriptsfoldertbl / (2020 / (180 + 4))**

**# Seiten documentfoldertbl + # Seiten grammarsfoldertbl + # Seiten promptsfoldertbl + # Seiten scriptseitenfoldertbl = Gesamtanzahl Seiten für Ordertabellen**

### Bestimmen Sie die Anzahl der Seiten, die von den Tabellen für Repository-Elementdateien verwendet werden.

Führen Sie die gleiche Berechnung durch, um die Gesamtzahl der Seiten für die Dateitabellen zu ermitteln, die die eigentlichen Repository-Elemente enthalten. Geben Sie diese Befehle mit Microsoft SQL Query Analyzer ein:

- **SELECT COUNT(\*) FROM documentFileBl**
- **SELECT COUNT(\*) FROM grammarsfiletbl**
- **SELECT COUNT(\*) VON promptFileBl**
- **SELECT COUNT(\*) FROM scriptsfildtbl**

Bestimmen Sie die Anzahl der Seiten, die von den Inhalten der Repository-Dateitabellen mit dieser Formel belegt sind, und ersetzen Sie die **Anzahl der Zeilen** durch die Zählungen, die aus den zuvor genannten Befehlen stammen. Berechnen Sie die Formel für jede Tabelle, und fügen Sie die Anzahl der Seiten hinzu.

- # Pages documentsFileBI = **Number of Rows in documentFileBI** / (2020 / (229 + 4))
- # Pages grammarsfiletbl = **Anzahl der Zeilen in grammarsfiletbl** / (2020 / (229 + 4))
- # Pages promptsfiletbl = **Anzahl der Zeilen in promptFileBI** / (2020 / (229 + 4))
- # Pages scriptsfiletbl = **Anzahl der Zeilen in scriptsfiletbl** / (2020 / (229 + 4))

# Pages documentsFileBI + # Pages grammarsfiletbl + # Pages promptsfiletbl + # Pages scriptsfiletbl = **Gesamtanzahl der Seiten für Dateitabellen**

## Berechnen

Führen Sie diese Berechnungen durch, um die aktuelle Größenschätzung der Repository-Daten abzuschließen:

1. Bestimmen Sie die Gesamtzahl der Seiten, die für die Speicherung des aktuellen Repository in Informix erforderlich sind, mit:

**Gesamtanzahl Seiten = Gesamtanzahl Seiten für Dateitabellen + Gesamtanzahl Seiten für Ordertabellen**

2. Stellen Sie fest, welcher Gesamtspeicherplatz (in MB) für die Seiten belegt ist:

**Gesamtanzahl der Seiten x 2 = Gesamtgröße in MB**

Wenn die Berechnungen zeigen, dass die Kontextinformationen zu den aktuell in UCCX Version 7.x hochgeladenen Repository-Elementen und -Ordern 3,4 MB überschreiten, wird empfohlen, das Design der Repository-Elemente neu zu gestalten. Obwohl der freie Speicherplatz für die kontextbezogenen Informationen über Repository-Objekte in **db\_cra\_repository** 6,9 MB beträgt, wird empfohlen, 50% für zukünftiges Wachstum verfügbar zu lassen. Die Wachstumsschätzungen und der maximal zulässige Speicherplatz werden pro Bereitstellung auf Basis der erwarteten Wachstumsfaktoren berechnet.

## Aufforderungen

Da Prompts in der Regel der größte Verbraucher des Repository-Speicherplatz sind, werden im Rest dieses Artikels Methoden zur Reduzierung der Anzahl von Prompts im Repository beschrieben.

Wenn die derzeit im UCCX Version 7.x-Repository hochgeladenen Eingabeaufforderungen einen wesentlichen Teil des gesamten Speicherplatz im Repository belegen, müssen Sie Design, Speicherung und Abruf der Prompt-Anweisung vor der Migration auf UCCX Version 8.x umgestalten. Wenn Sie das Prompt-Design umgestalten möchten, sollten Sie folgende Optionen in Betracht ziehen:

- Reduzieren Sie die Anzahl der Aufforderungsordner. Wie in den vorherigen Berechnungen gezeigt, belegt jeder Prompt-Ordner eine Zeile in der Repository-Datenbank und verbraucht daher Speicherplatz gegen den **db\_cra\_repository**-Grenzwert. Wenn Sie die Anzahl der Ordner reduzieren, können Sie Speicherplatz für Dateien freigeben.
- Geben Sie über Anwendungen/Skripte hinweg Aufforderungen zu gebräuchlichen Wörtern oder Ausdrücken frei, um die Anzahl der im System erforderlichen Aufforderungen zu reduzieren.

- Verwenden Sie die vom System bereitgestellten Aufforderungen zur Eingabe von Gemeinsamkeiten zwischen Aufforderungen, z. B. Zahlen und Währungen.
- Speichern von Aufforderungen auf einem separaten Webserver im Unternehmen und Abrufen und Abspielen der erforderlichen Aufforderungen bei Bedarf mit Voice XML (VXML).

VXML wird verwendet, um Prompts On-Demand von einem externen Ort abzurufen und abzuspielen. Wenn Sie große Mengen von Aufforderungen auf einem separaten Webserver speichern, können Sie:

- Beseitigen Sie die zuvor erwähnten Migrationsprobleme, da Eingabeaufforderungen nicht mehr im **db\_cra\_repository** gespeichert werden.
- Bieten Sie einen besseren Zugriff und eine einfachere Verwaltung.
- Reduzierung der Zeit für Switch-Versionen, Upgrades, Migration und Backup

Obwohl viele Optionen zur Instrumentierung der interaktiven Sprachsteuerung (IVR) in VXML vorhanden sind, werden das UCCX-Skript und die VXML-Anwendung verwendet, um eine Aufforderung von einem externen Webserver abzurufen und an den Anrufer abzuspielen, als Grundlage für die weitere Entwicklung. Wie andere benutzerdefinierte Skripts in UCCX werden die in diesem Abschnitt bereitgestellten Skripts als Leitfaden angeboten und werden nicht vom Cisco Technical Assistance Center (TAC) unterstützt.

**Hinweis:** Eine VXML-Anwendung wird mit dem Schritt **Voice Browser** aus einem UCCX-Skript aufgerufen. Weitere Informationen zum **Sprachbrowser** finden Sie in den [Programmierhandbüchern zu Cisco Unified Contact Center Express](#).

Der Schritt **Sprachbrowser** verwendet ein VXML-Dokument. Dieses Dokument muss als Ergebnis des Schritts **URL-Dokument erstellen** erstellt und auf einem externen Webserver außerhalb von UCCX gehostet werden. Obwohl die VXML-Anwendung geschrieben wurde, um Anrufereingaben über Dual Tone Multi Frequency (DTMF) zu akzeptieren, wurde diese Anwendung nur für die Wiedergabe einer externen Aufforderung entwickelt. Sie kann jedoch erweitert werden, um zusätzliche Funktionen einzuschließen. Es wird davon ausgegangen, dass der Rest des UCCX-Skripts, bevor der Schritt **Sprachbrowser** aufgerufen wird, über die Logik verfügt, die erforderlich ist, um festzustellen, welche Eingabeaufforderung wiedergegeben wird, und eine String-Variable auf den Dateinamen Prompt festgelegt wird.

Da das VXML-Dokument statisch, aber dynamisch dargestellt wird, wird eine Skriptsprache auf Serverseite verwendet, um das VXML-Dokument zu erstellen. Dabei kann es sich um eine beliebige serverseitige Skriptsprache handeln, die den Content-Type-Header der XML **GET-Anforderung**-Antwort festlegen kann. In diesem Beispiel wird PHP verwendet.

Die PHP-Seite wird geschrieben, um einen URL-Parameter in einer **GET-Anforderung** zu akzeptieren, die den Namen der Audioaufforderung darstellt, der abgespielt wird. Die PHP-Seite verknüpft die VXML-Vorlage mit dem in den URL-Parametern für die **GET-Anforderung** übergebenen Dateinamen Prompt, um das vollständige VXML-Dokument zu bilden. Anschließend wird der Content-Type-Header der Antwort auf XML festgelegt, und der Text der Antwort wird als VXML-Inhalt festgelegt.

```
<?php
$wav_filename = $_GET['wav'];

$xml_string = '<?xml version="1.0"?>
<vxml xmlns="http://www.w3.org/2001/vxml" version="2.0">
  <form>
```

```

    <block>
        <prompt bargein="true">
            <audio src="http://<Servername or IP Address>/
                <Path>/'. $wav_filename.' .wav" />
            </prompt>
        </block>
    </form>
</vxml>';
header('Content-type: text/xml');
echo $xml_string;
?>

```

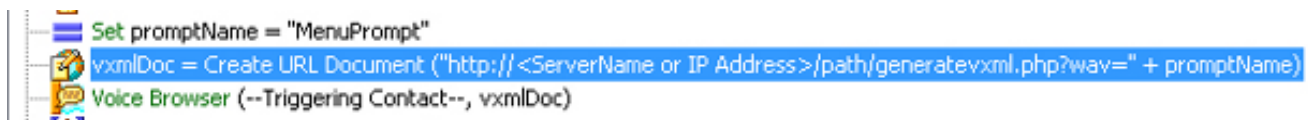
Um ein gut geformtes VXML-Dokument zu erstellen, muss auf die PHP-Beispielseite mit einer **GET Request** zugegriffen werden, die einen Parameter **wav** und einen String-Wert enthält, wobei davon ausgegangen wird, dass die PHP-Beispielseite den Namen **generatevxml.php** hat:

**http://**

Stellen Sie sicher, dass sich die **MenuPrompt.wav** an der Stelle auf dem externen Webserver befindet, der in der in der PHP-Seite enthaltenen VXML-Vorlage angegeben ist.

Verwenden Sie im UCCX-Skript den Schritt **URL-Dokument erstellen**, um eine **GET-Anforderung** von **generatevxml.php** auszuführen, die die Basis-URL von **http://<Servername oder IP-Adresse>/path/generatevxml.php?wav=** mit dem aus der vorherigen Skriptlogik abgeleiteten Forderungsdateinamen verkettet, und legen Sie das Ergebnis in eine Dokumentvariable.

Erstellen Sie einen **Sprachbrowser**-Schritt, der die Dokumentvariable verwendet.



Wenn dieses Skript aufgerufen wird, wird dem Aufrufer die **MenuPrompt.wav-Eingabeaufforderung** abgespielt, vorausgesetzt, sowohl **generatevxml.php** als auch **MenuPrompt.wav** sind auf dem Webserver von UCCX zugänglich.

Wenn VXML-Anwendungen verwendet werden, um externe Aufforderungen zu speichern, sodass auf diese nur bei Bedarf zugegriffen wird, um sie dem Anrufer wiederzugeben, ermöglicht dies eine höhere Effizienz, Verwaltbarkeit und Wartungsfreundlichkeit. Dieses Problem sollte beachtet werden, wenn ein UCCX Version 7.x-System auf ein UCCX Version 8.x-System aktualisiert wird und die Anzahl der Aufforderungen so groß ist, dass der Inhalt der Kontextinformationen größer ist als der Inhalt des **db\_cra\_repository** oder des **uccx\_sbspace**.