

UCCX 저장소 제한 사항

목차

[소개](#)

[배경 정보](#)

[문제](#)

[솔루션](#)

[현재 저장소 데이터 크기 예측](#)

[저장소 폴더 테이블의 행 수 결정](#)

[저장소 폴더 테이블에서 사용하는 페이지 수 결정](#)

[저장소 항목 파일 테이블에서 사용하는 페이지 수 결정](#)

[계산](#)

[프롬프트](#)

소개

이 문서에서는 Cisco UCCX(Unified Contact Center Express) 시스템을 버전 8 이상으로 업그레이드하고 많은 수의 저장소 항목이 시스템에 업로드되거나 버전 8 이상에서 시스템에 많은 수의 저장소 항목을 업로드하려고 할 때 발생하는 문제에 대해 설명합니다.

배경 정보

UCCX 버전 7.x 이상에서는 Microsoft SQL(MSSQL)을 데이터베이스 엔진으로 사용합니다. MSSQL은 데이터 저장의 경우 서로 다른 유형의 데이터를 구분하지 않습니다. 3GB 데이터베이스에 데이터를 저장하면 MSSQL은 유형에 관계없이 모든 데이터를 하나의 3GB 블록에 저장합니다.

반면 UCCX 버전 8.0 이상에서 사용되는 데이터베이스 엔진인 Informix는 디스크에 저장할 때 다른 유형의 데이터를 구별합니다. 일반적인 데이터베이스 데이터(예: 문자열, 문자 및 정수)는 데이터베이스에 대한 디스크 청크에 저장되고 데이터베이스 테이블 레코드에 있는 BLOB(Binary Large Object) 데이터는 sbospace라는 디스크의 별도의 섹션에 저장됩니다. sbospace는 BLOB 데이터를 저장하는 하나 이상의 디스크 청크로 구성된 논리 단위입니다. Informix는 데이터베이스와 디스크에서 BLOB 데이터를 읽고 쓰는 성능을 높이기 위해 기존 데이터와 BLOB 데이터를 별도로 저장합니다. BLOB 데이터를 포함하는 데이터베이스를 만들 때 관리자는 데이터베이스에 대한 디스크 청크의 크기(기존 데이터를 저장하기 위한)와 하위 공간의 크기를 별도로 지정해야 합니다.

데이터 저장 메커니즘의 경우 MSSQL은 모든 데이터를 N 크기의 단일 버킷에 배치하고 Informix는 이 데이터의 저장소를 두 버킷으로 나눕니다. 크기 X의 BLOB 데이터에 대한 컨텍스트 정보를 위한 버킷 하나, 크기 Y인 BLOB 개체 자체의 버킷 하나.

UCCX에서 관리자는 프롬프트, 문서, 문법 및 스크립트로 구성된 저장소 항목을 업로드할 수 있습니다. 이러한 항목의 내용은 해당 데이터베이스 테이블에 파일 이름, 폴더, 마지막 수정 시간, 마지막 수정 사용자, 길이 및 체크섬과 같은 컨텍스트 정보와 BLOB 데이터로 저장됩니다.

리포지토리 항목은 UCCX 데이터베이스 db_cra_repository에 저장됩니다.MSSQL을 사용하는 UCCX 버전 7.x 및 이전 버전에서는 db_cra_repository의 크기가 3GB이며 컨텍스트 정보와 BLOB 정보를 모두 포함합니다.Informix를 사용하는 UCCX 버전 8.0 이상에서는 db_cra_repository에 연결된 데이터 저장소 청크의 크기가 10.2MB이며 저장소 항목에 대한 컨텍스트 정보만 저장합니다.리포지토리 항목의 콘텐츠는 uccx_sbspace라는 하위 공간에 BLOB 형식으로 저장됩니다.UCCX 버전 8.0 이상에서는 uccx_sbspace의 크기가 3GB입니다.

UCCX 버전 8.0+ 서버의 show uccx dbserver disk를 출력하면 다음 두 데이터 저장소 간의 차이가 표시됩니다.

```
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	24508.5	5551.2	27847.3	80%
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_sbspace	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%

MSSQL 데이터베이스의 데이터 조합에 따라 MSSQL 데이터베이스에 저장된 BLOB 데이터 크기가 마이그레이션 또는 업그레이드를 시도할 때 Informix에서 정의된 서브넷 크기를 초과할 수 있습니다.마찬가지로 MSSQL 데이터베이스에 저장된 BLOB 데이터에 대한 컨텍스트 정보가 Informix 데이터베이스 청크에 있는 해당 데이터에 대해 관리자가 지정한 크기를 초과할 수 있습니다.

이 경우 UCCX 버전 7.x에서 UCCX 버전 8.x로의 업그레이드 또는 마이그레이션이 실패합니다. db_cra_repository 또는 uccx_sbspace가 MSSQL에 저장된 동일한 정보를 수용할 만큼 크지 않기 때문입니다.이는 일반적으로 UCCX 시스템에서 많은 수의 프롬프트가 포함된 문제입니다.컨텍스트 프롬프트 및 BLOB 데이터는 db_cra_repository 및 uccx_sbspace를 문서, 문법 및 스크립트와 공유해야 하지만 이러한 다른 저장소 유형은 일반적으로 크기와 수가 작습니다.

예를 들어 수만 개의 프롬프트가 있는 UCCX Version 7.x 시스템은 각각 몇 초 오디오만 있으면 됩니다.MSSQL을 사용하는 UCCX 버전 7.x에서 프롬프트 콘텐츠 및 컨텍스트 정보는 동일한 3GB 청크에 저장됩니다.작은 크기의 프롬프트가 많으므로 데이터베이스에는 프롬프트에 대한 컨텍스트 정보가 50MB에 저장되지만 프롬프트의 오디오를 나타내는 BLOB 데이터는 2GB에 불과합니다.따라서 저장소의 프롬프트는 데이터베이스 생성 시 설정된 3GB 제한의 2GB를 약간 넘습니다.

이 시스템을 UCCX 버전 8.x 및 Informix로 마이그레이션하려고 하면 50MB의 컨텍스트 정보가 db_cra_repology의 10.2MB 제한을 초과하기 때문에 마이그레이션이 실패합니다. 단 2GB의 프롬프트 내용이 uccx_sbspace의 제한에 잘 맞지만 마찬가지입니다.

반대로, UCCX 버전 7.x 시스템의 경우 프롬프트 수가 적지만 긴 경우도 많습니다.프롬프트 수가 적지만 크기가 클수록 컨텍스트 정보에 대한 프롬프트 내용의 비율이 달라집니다.UCCX 버전 7.x 및 MSSQL에서 프롬프트 콘텐츠는 db_cra_repository의 2.8GB와 컨텍스트 정보 3MB가 걸릴 수 있습니다.이 시스템은 3MB가 db_cra_repository에 맞고 2.8GB가 uccx_sbspace 할당에 맞게 업그레이드되므로 성공적으로 업그레이드됩니다.

일반적으로 UCCX 버전 8.x 이상으로 마이그레이션하려는 경우 프롬프트 내용이 uccx_sbspace의

크기 제한을 초과하기 전에 UCCX 버전 7.x 또는 이전 시스템에 업로드된 프롬프트에 대한 컨텍스트 데이터가 **db_cra_repository**의 크기 제한을 초과합니다. 또한 사용자 정의 저장소 항목에 사용 가능한 실제 사용 가능한 공간은 6.9MB이며, 기본 구성은 db_cra_repository의 3.4MB를 소비합니다.

문제

버전 8 이상을 실행하는 UCCX 시스템에 새 리포지토리 항목(문서, 문법, 프롬프트, 스크립트)을 업로드하려고 하면 다음 오류 메시지가 표시됩니다.

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

UCCX 버전 7.0(2) 이하에서 버전 8.0 이상으로 마이그레이션하면 애플리케이션이 실행되는 운영 체제와 데이터베이스 엔진이 변경됩니다. UCCX 버전 8.0 이상에서 사용되는 데이터베이스 엔진은 UCCX 버전 7.x 이상과 다른 방식으로 데이터를 저장합니다. 이는 UCCX 버전 7.x의 대용량 데이터 세트를 포함하는 데이터베이스가 UCCX 버전 8.x로 올바르게 마이그레이션되지 않을 수 있으므로 UCCX를 마이그레이션할 때 영향을 미칩니다.

솔루션

UCCX 버전 8.x로 마이그레이션하기 전에 향후 증가율을 포함하기 위해 현재 저장소 항목을 UCCX 버전 7.x 시스템에 저장하는 데 필요한 **db_cra_repository** 및 **uccx_sbospace**의 양을 추정합니다.

현재 저장소 데이터 크기 예측

먼저 저장소 항목과 폴더 모두에 대한 정보를 저장하는 각 저장소 테이블의 행 수를 결정합니다.

저장소 폴더 테이블의 행 수 결정

Microsoft SQL Query Analyzer를 사용하여 Repository 폴더 테이블의 행 수를 다음 명령으로 기록할 수 있습니다.

- 문서에서 개수(*) 선택 **foldertbl**
- **SELECT COUNT(*) FROM grammarsfoldertbl**
- 프롬프트에서 COUNT(*) 선택 **폴더테이블**
- 스크립트폴더테이블에서 COUNT(*) 선택

저장소 폴더 테이블에서 사용하는 페이지 수 결정

Informix는 페이지 측면에서 디스크 크기를 고려합니다. 이 수식으로 저장소 폴더 테이블의 콘텐츠가 차지하는 페이지 수를 확인하고 앞서 언급한 명령에서 얻은 수에 대해 **행 수**를 대체합니다. 각 테이블에 대해 이 수식을 계산하고 페이지 수를 추가합니다. 각 테이블의 행 수를 먼저 추가한 다음 수식 결과를 계산하면 페이지 수를 정확하게 확인할 수 없습니다.

- # 페이지 문서폴더테이블 = 문서의 **행 수**foldertbl / (2020 / (180 + 4))

- # 페이지 문법폴더테이블 = **grammarsfoldertbl**의 행 수 / (2020 / (180 + 4))
- # 페이지 프롬프트폴더tbl = **프롬프트의 행 수** / (2020 / (180 + 4))
- # 페이지 스크립트폴더테이블 = **scriptsfoldertbl**/(2020 / (180 + 4)) 행 수

페이지 문서폴더tbl + # 페이지 그라마폴더tbl + # 페이지 프롬프트폴더tbl + # 페이지 스크립트폴더tbl = **폴더 테이블 총 페이지 수**

저장소 항목 파일 테이블에서 사용하는 페이지 수 결정

실제 저장소 항목을 포함하는 파일 테이블의 총 페이지 수를 확인하려면 동일한 계산을 완료합니다. Microsoft SQL 쿼리 분석기에 다음 명령을 입력합니다.

- 문서에서 개수(*) 선택 문서파일 테이블
- **SELECT COUNT(*) FROM grammarsfiletbl**
- 프롬프트에서 COUNT(*) 선택
- 스크립트파일테이블에서 COUNT(*) 선택

이 수식으로 저장소 파일 테이블의 콘텐츠가 차지하는 페이지 수를 확인하고 앞서 언급한 명령에서 얻은 수로 **행 수**를 대체합니다. 각 테이블의 수식을 계산하고 페이지 수를 추가합니다.

- # 페이지 문서파일 테이블 = **문서의 행 수** / (2020 / (229 + 4))
- # 페이지 문법파일 테이블 = **grammarsfiletbl**의 행 수 / (2020 / (229 + 4))
- # 페이지 프롬프트파일 테이블 = **프롬프트의 행 수** / (2020 / (229 + 4))
- # 페이지 스크립트파일테이블 = **스크립트파일테이블**/(2020 / (229 + 4))의 행 수

페이지 문서파일 테이블 + # 페이지 그라마파일 테이블 + # 페이지 프롬프트파일 테이블 + # 페이지 스크립트파일테이블 = **파일 테이블 총 페이지 수**

계산

현재 저장소 데이터 크기 예측을 완료하려면 다음 계산을 수행합니다.

1. Informix에 현재 저장소를 저장하는 데 필요한 총 페이지 수를 결정합니다.

총 페이지 수 = 파일 테이블의 총 페이지 수 + 폴더 테이블의 총 페이지 수

2. 페이지에서 사용하는 총 공간(MB)을 결정합니다.

총 페이지 수 x 2 = 총 크기(MB)

UCCX 버전 7.x에서 현재 업로드된 저장소 항목 및 폴더에 대한 컨텍스트 정보가 3.4MB를 초과하는 것으로 계산되면 저장소 항목 설계를 다시 팩터링하는 것이 좋습니다. **db_cra_repository**의 저장소 항목에 대한 컨텍스트 정보에 사용할 수 있는 여유 공간은 6.9MB이지만 향후 증가에 대해 50%를 사용할 수 있도록 유지하는 것이 좋습니다. 성장 예상치와 최대 사용 가능 공간은 예상 성장 요인을 기준으로 구축당 계산됩니다.

프롬프트

프롬프트는 일반적으로 저장소 공간의 가장 큰 소비자이므로 저장소에서 프롬프트 수를 줄이기 위해 사용하는 방법에 대해서는 이 문서의 나머지 부분에서 설명합니다.

UCCX 버전 7.x 저장소에 현재 업로드된 프롬프트가 전체 저장소 스토리지 공간의 상당 부분을 차지하는 경우 UCCX 버전 8.x로 마이그레이션하기 전에 프롬프트 설계, 스토리지 및 검색을 리팩터링합니다. 프롬프트 디자인을 리팩터링하려고 할 때 다음 옵션을 고려하십시오.

- 프롬프트 폴더 수를 줄입니다. 이전 계산에 표시된 것처럼 각 Prompt 폴더는 저장소 데이터베이스의 행을 사용하므로 db_cra_repository 제한에 대한 공간을 사용합니다. 폴더 수를 줄이면 파일의 공간을 확보할 수 있습니다.
- 시스템에서 필요한 프롬프트 수를 줄이기 위해 응용 프로그램/스크립트 간에 공통 단어나 구를 묻는 메시지 공유
- 시스템 제공 프롬프트에서는 숫자 및 통화 등의 프롬프트 간의 공통성을 사용합니다.
- 엔터프라이즈 내의 별도의 웹 서버에 프롬프트를 저장하고 VXML(Voice XML)을 사용하여 필요에 따라 필요한 프롬프트를 검색하고 재생합니다.

VXML은 오프박스 위치에서 요청 시 프롬프트를 검색하고 재생하는 데 사용됩니다. 별도의 웹 서버에 많은 양의 프롬프트를 저장하는 경우 다음을 수행할 수 있습니다.

- 프롬프트는 더 이상 db_cra_repository에 저장되지 않으므로 이전에 언급한 마이그레이션 문제를 제거합니다.
- 더 우수한 액세스 및 관리 용이성 제공
- 스위치 버전, 업그레이드, 마이그레이션 및 백업 시간 단축

VXML에서 IVR(Interactive Voice Response) 사용자 지정을 계획하기 위해 여러 옵션이 있지만 오프박스 웹 서버에서 프롬프트를 검색하고 발신자에게 재생하기 위해 사용되는 UCCX 스크립트 및 VXML 애플리케이션은 추가 개발을 위한 기반으로 사용됩니다. UCCX의 다른 사용자 지정 스크립팅과 마찬가지로, 이 섹션에 제공된 스크립트는 가이드로 제공되며 Cisco TAC(Technical Assistance Center)에서 지원하지 않습니다.

참고: VXML 애플리케이션은 음성 브라우저 단계를 사용하여 UCCX 스크립트에서 호출됩니다. 음성 브라우저 단계에 대한 자세한 내용은 [Cisco Unified Contact Center Express 프로그래밍 가이드](#)를 참조하십시오.

음성 브라우저 단계에서는 VXML 문서를 사용합니다. 이 문서는 Create URL Document(URL 문서 생성) 단계의 결과로 작성되어야 하며 UCCX 외부 웹 서버에서 호스팅되어야 합니다. DTMF(Dual Tone Multi Frequency)를 통해 발신자 입력을 수락하기 위해 VXML 응용 프로그램이 작성되지만 이 응용 프로그램은 오프박스(off-box)에서 호스팅되는 프롬프트만 재생하도록 설계되었습니다. 그러나 추가 기능을 포함하도록 확장할 수 있습니다. 음성 브라우저 단계를 호출하기 전에 UCCX 스크립트의 나머지 부분에는 어떤 프롬프트가 재생되는지 확인하는 데 필요한 논리와 String 변수가 Prompt 파일 이름으로 설정된 것으로 가정합니다.

VXML 문서는 정적 문서이지만 이 문서를 통해 재생되는 프롬프트는 동적이므로 VXML 문서를 만드는 데 서버측 스크립팅 언어가 사용됩니다. XML GET 요청 응답의 콘텐츠 형식 헤더를 설정할 수 있는 서버측 스크립팅 언어일 수 있습니다. 이 예에서는 PHP가 사용됩니다.

PHP 페이지는 재생되는 오디오 프롬프트 이름을 나타내는 GET 요청에서 URL 매개 변수를 수락하기 위해 작성됩니다. PHP 페이지는 전체 VXML 문서를 구성하기 위해 GET Request URL 매개 변수에 전달된 프롬프트 파일 이름과 VXML 템플릿을 연결합니다. 그런 다음 응답의 Content-type 헤더를 XML로 설정하고 응답 본문을 VXML 내용으로 설정합니다.

```
<?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;
?>

```

올바른 형식의 VXML 문서를 만들려면 PHP 페이지의 이름이 generatevxml.php라는 가정을 사용하여 매개 변수 wav와 String 값이 포함된 GET 요청을 사용하여 예제 PHP 페이지에 액세스해야 합니다.

http://

MenuPrompt.wav가 PHP 페이지에 포함된 VXML 템플릿에 지정된 외부 웹 서버의 위치에 있는지 확인합니다.

UCCX 스크립트에서 Create URL Document 단계를 사용하여 http://<Servername 또는 IP Address>/path/generatevxml.php?wav=의 기본 URL을 이전 스크립팅 로직에서 파생된 프롬프트 파일 이름과 연결하여 Generatevxml.php의 GET 요청을 수행하고 결과를 문서 변수에 넣습니다.

문서 변수를 사용하는 음성 브라우저 단계를 만듭니다.

```

Set promptName = "MenuPrompt"
vxmlDoc = Create URL Document ("http://<ServerName or IP Address>/path/generatevxml.php?wav=" + promptName)
Voice Browser (--Triggering Contact--, vxmlDoc)

```

UCCX의 웹 서버에서 generatevxml.php와 MenuPrompt.wav를 모두 액세스할 수 있는 경우 이 스크립트가 호출되면 MenuPrompt.wav 프롬프트가 호출자에게 재생됩니다.

VXML 애플리케이션을 사용하여 프롬프트를 오프박스(off-box)에 저장할 때 발신자에게 재생하기 위해 필요한 경우에만 액세스하도록 할 경우 효율성, 관리 용이성 및 서비스 용이성이 향상됩니다. 이는 UCCX 버전 7.x 시스템이 UCCX 버전 8.x 시스템으로 업그레이드된 경우에 고려해야 할 문제입니다. 프롬프트 수는 컨텍스트 정보의 내용이 db_cra_repository 또는 uccx_sbospace보다 클 수 있습니다.