

IBM DB2 Cube Views



안내 및 참조서

버전 8.2

IBM DB2 Cube Views



안내 및 참조서

버전 8.2

주:

이 정보와 이 정보가 지원하는 제품을 사용하기 전에, 331 페이지의 『주의사항』의 정보를 읽으십시오.

제 2 판(2004년 9월)

이 문서에는 IBM의 소유권에 대한 정보가 들어 있습니다. 이것은 라이선스 계약에 따라 제공되었으며 저작권 법의 보호를 받습니다. 본 서적에 포함된 정보는 어떠한 제품 보증을 포함하지 않으며 본 매뉴얼에서 제공되는 어떠한 사항도 이와 같이 해석되지 않습니다.

IBM 책은 온라인 또는 한국 IBM 담당자를 통해 주문할 수 있습니다.

- 책을 온라인으로 주문하려면, www.ibm.com/shop/publications/order에서 IBM Publications Center로 가십시오.
- 한국 IBM 담당자를 찾으려면, www.ibm.com/planetwide에서 IBM Directory of Worldwide Contacts로 가십시오.

미국 또는 캐나다에서 DB2 마케팅 및 판매처로부터 DB2 책을 주문하려면, 1800-IBM-4YOU(425-4968)로 문의하십시오.

IBM에 정보를 보내는 경우, IBM은 귀하의 권리를 침해하지 않는 범위 내에서 적절하다고 생각하는 방식으로 귀하가 제공한 정보를 사용하거나 배포할 수 있습니다.

목차

이 책의 정보 v
 이 책의 독자 v
 이 책에 사용된 규칙. v
 온라인 정보 vi

제 1 장 DB2 Cube Views 설치, 이주 및 구성 . . . 1
 DB2 Cube Views 향상 1
 DB2 Cube Views 소개 4
 설치 요구사항 5
 DB2 Cube Views 설치 7
 DB2 Cube Views XML 파일을 V8.1에서 V8.2로
 이주 7
 DB2 Cube Views에서 데이터베이스 설정 9
 CVSAMPLE 샘플 데이터베이스 설정 9
 데이터베이스 연결 문제점 해결 10

**제 2 장 DB2 Cube Views 메타데이터 오브젝트 정
 보 13**
 관계형 테이블에 맵핑하는 메타데이터 오브젝트 . . . 14
 관계형 테이블에 맵핑하는 큐브 모델의 예 16
 일반 메타데이터 오브젝트 등록 정보 22
 큐브 모델 24
 사실 오브젝트 25
 차원. 25
 계층 구조. 26
 레벨. 30
 치수. 33
 속성. 35
 속성 관계 36
 조인. 38
 큐브. 39
 큐브 사실 오브젝트. 40
 큐브 차원 40
 큐브 계층 구조 41
 큐브 레벨. 41
 메타데이터 오브젝트 규칙. 42

**제 3 장 DB2 Cube Views 메타데이터 오브젝트
 모델 설계 49**
 OLAP 센터 시작 및 새로 고침 49
 OLAP 센터 및 API 버전 호환성 50
 DB2 Cube Views 사용 권한 및 특권 50
 DB2 Cube Views 메타데이터 오브젝트 작성 53

DB2 Cube Views와 OLAP 도구 간에 메타데이
 터 교환. 53
 빠른 시작 마법사로 큐브 모델 작성 55
 완전한 큐브 모델 작성 56
 큐브 모델에 기존 차원 추가. 60
 조인 작성 61
 큐브 작성 62
 큐브 모델에서 차원 제거 65
 데이터베이스에서 메타데이터 오브젝트 삭제 65

**제 4 장 DB2 Cube Views 비즈니스 모델링 시나리
 오 67**
 웨어하우스에 시간 초과된 재고의 플로우 및 값 계산 68
 광고 비용과 판매액의 상관관계. 73
 이익 및 상점의 이익 마진 계산. 78
 인터넷 주문 숫자 카운팅 82
 판매액 수치 순위 84
 사실 테이블에 저장된 시간 데이터를 사용하여 Time
 차원 작성 88

제 5 장 DB2 Cube Views 큐브 모델 최적화 . . . 91
 요약 테이블 92
 함수 종속성 및 제한조건이 있는 요약 테이블 98
 최적화 프로세스에 대한 개요 101
 최적화 수행 시 메타데이터 설계 고려사항. 104
 큐브의 최적화 슬라이스 107
 후보 최적화 슬라이스에서 쿼리 분석. 116
 최적화 수행 시 제한조건 정의. 118
 최적화 어드바이저의 매개변수. 124
 큐브 모델 최적화 126
 요약 테이블을 작성하는 SQL 스크립트에 대한 예 127
 쿼리 결과 테스트 130
 요약 테이블 문제점 해결 131
 요약 테이블 유지보수 133
 요약 테이블 삭제 134

**제 6 장 DB2 Cube Views 및 페더레이티드 데이
 터 소스 137**
 페더레이티드 시스템에 대한 개요. 137
 DB2 Cube Views를 사용한 리모트 데이터 소스의
 최적화 개요. 139
 DB2 Cube Views에서 페더레이티드 시스템 사용 140
 리모트 데이터 소스 정의 141

DB2 Cube Views의 리모트 테이블에 별칭 정의	142	DB2 Cube Views 메타데이터 테이블 및 XML 스	
DB2 Cube Views의 별칭에서 정보용 제한조건		키마 파일	178
정의	142	DB2 Cube Views 구성 파일	179
리모트 데이터 소스에서 쿼리 성능 문제점 해결	143	메타데이터 오브젝트 형식	180
제 7 장 DB2 Cube Views API	145	제 8 장 샘플 파일	183
DB2 Cube Views API 개요	145	db2mdapiclient 유틸리티에 대한 개요	183
DB2 Cube Views API: DB2 스토어드 프로시저		db2mdapiclient 명령: 메타데이터 오브젝트 조작	184
및 XML 구문 분석	146	샘플 데이터베이스 파일	185
DB2 Cube Views 스토어드 프로시저	148	API 샘플 파일	186
DB2 Cube Views API 매개변수	150		
DB2 Cube Views API 메타데이터 조작 매개변		부록, 메시지	193
수	150	SQLSTATE, API 및 기타 서버 메시지	193
입력 및 출력 매개변수	150	API SQL 상태	193
DB2 Cube Views 메타데이터 조작	151	일반	195
검색 조작: 설명	151	External API 오류	222
권고 조작: 권고	152	최적화	275
수정 조작: 변경, 작성, 삭제, импорт 및 이름 바		OLAP 센터 메시지	286
꾸기	158	10000-10600	286
관리 조작: 유효성 확인 및 변환	163	DB2 및 DB2 Cube Views의 상태 메시지	328
메타데이터 조작의 샘플 입력 및 출력 매개변수	165		
조작 매개변수	167	주의사항	331
조작 피연산자	171	상표	333
메시지 구조	172	용어	335
조작 단계 시퀀스	173	색인	339
로그 및 추적	173	IBM에 문의	343
DB2 Cube Views API의 런타임 추적	173	제품 정보	343
DB2 Cube Views API에 대한 로그 파일	174	문서에 관한 주석	344
오류 로그	175		
로그 및 추적 시나리오	175		
코드 페이지 지원	176		

이 책의 정보

이 책에서는 다음 DB2 Cube Views 주제에 관한 정보를 제공합니다.

- DB2 Cube Views 시작 방법
- 메타데이터의 임포트와 익스포트 및 큐브 모델과 큐브의 작성에 사용할 수 있는 OLAP 센터(그래픽 사용자 인터페이스)
- OLAP 쿼리의 성능 향상에 도움이 되는 최적화
- DB2 Universal Database™(DB2®) 카탈로그에 저장할 수 있는 메타데이터 오브젝트
- SQL을 사용하여 데이터에 액세스하는 응용프로그램을 작성할 수 있는 API(Application programming interface)
- 메타데이터 오브젝트에서 차원 및 복합 치수를 빌드하는 방법에 대한 예(이러한 차원 및 치수를 사용하여 일반 비즈니스 시나리오를 모델화할 수 있습니다).

이 책의 독자

DB2 Cube Views를 사용하여 OLAP 및 데이터베이스 도구에서 다차원 메타데이터를 캡처하고 해당 메타데이터를 DB2 카탈로그에 저장할 수 있습니다. 그 다음 해당 메타데이터를 사용하여 온라인 분석 처리(OLAP) 큐브 모델 및 큐브를 작성할 수 있습니다. 이때 큐브는 큐브 모델의 서브세트입니다.

또한 DB2 Cube Views는 큐브 모델에 발행된 쿼리의 성능 향상에 도움이 되도록 제공 SQL 스크립트를 제공하여 요약 테이블을 빌드하는 최적화 어드바이저를 제공합니다.

OLAP 메타데이터 및 DB2 Universal Database(DB2 UDB)에 대해 작업하는 데이터베이스 관리자인 경우 이 책을 읽으십시오. 독자는 다음 사항을 잘 알고 있어야 합니다.

- DB2 카탈로그 및 요약 테이블
- OLAP 개념(예: 큐브, 차원, 계층 구조 및 치수)
- API 개념과 CLI, ODBC, JDBC, XML 및 DB2 스토어드 프로시저

이 책에 사용된 규칙

명령 구문은 다음 규칙을 실행하는 텍스트 형식으로 표시됩니다.

- 명령은 굵은체의 소문자로 표시됩니다.
- 변수는 기울임꼴로 표시되고 목록의 명령 바로 뒤에서 설명됩니다.

- 둘 이상의 대안 매개변수 중 하나를 입력할 수 있는 경우 사용 가능한 매개변수는 수직 막대로 구분되어 표시되며 디폴트 매개변수에는 밑줄이 표시됩니다.
- 선택적 변수 및 매개변수는 대괄호로 닫힙니다.

예를 들어 **copy** 명령 구문은 다음 형식으로 표시될 수 있습니다.

copy filename [filetype]

*filename*은 필수 파일 이름입니다.

*[filetype]*은 선택적 파일 유형입니다.

온라인 정보

이 섹션에서는 이 제품과 관련된 웹 주소를 제공합니다.

www.ibm.com/redbooks

IBM[®] Redbooks[™] 웹 사이트

Redbooks 웹 사이트에서 다음 Redbook의 하드카피 및 CD 버전을 검색, 다운로드 또는 주문하거나 보십시오.

- *DB2 UDB's High Function Business Intelligence in e-business*, SG24-6546-00
- *Up and Running with DB2 UDB ESE Partitioning for Performance in an e-Business Intelligence World*, SG24-6917-00
- *Database Performance Tuning on AIX[®]*, SG24-5511-01
- *DB2 UDB V7.1 Performance Tuning Guide*, SG24-6012-00

www.ibm.com/software/data/db2/db2md

IBM DB2 Cube Views 웹 사이트

www.ibm.com/software/data/

IBM Data Management 웹 사이트

www.ibm.com/software/data/db2/udb/winos2unix/support/

DB2 Universal Database 및 DB2 Connect[™] 온라인 지원 웹 사이트

www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/download.d2w/report

DB2 유지보수 - DB2 UDB의 FixPack 웹 사이트

www.ibm.com/software/data/developer

DB2 개발 프로그램 도메인 웹 사이트

www.ibm.com/software/data/db2/library

DB2 제품 및 서비스 기술 라이브러리 웹 사이트

[www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/v8pubs.d2w/
en_main](http://www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/v8pubs.d2w/en_main)

DB2 서적 웹 사이트

제 1 장 DB2 Cube Views 설치, 이주 및 구성

이 섹션에서는 다음 주제를 설명합니다.

DB2 Cube Views 개선된 기능

이 릴리스에는 DB2 Cube Views V8.1 이후 DB2 Cube Views 메타데이터에 적용된 여러 변경사항이 들어 있습니다.

DB2 Cube Views 소개

DB2 Cube Views는 OLAP 처리를 수행하도록 DB2 UDB 기능을 향상시키는 DB2 Universal Database의 추가 기능입니다. DB2 Cube Views를 사용하여 OLAP 솔루션의 전개 및 관리를 간소화(streamline)하고 OLAP 도구 및 응용프로그램의 성능을 향상시킬 수 있습니다.

설치 요구사항

DB2 Cube Views를 설치하기 전에 모든 요구사항을 충족시켰는지 확인하십시오.

DB2 Cube Views 설치

설치 런치패드를 사용하여 DB2 Cube Views를 Windows, AIX, Linux 및 Solaris 운영 체제 플랫폼에 설치하십시오.

DB2 Cube Views 이주

스크립트를 직접 실행하거나 OLAP 센터를 사용하여 DB2 Cube Views 사용 가능 데이터베이스를 버전 8.1에서 버전 8.2로 이주시킬 수 있습니다.

DB2 Cube Views에서 데이터베이스 설정

DB2 Cube Views에서 사용할 새 데이터베이스를 설정할 수 있습니다.

CVSAMPLE 샘플 데이터베이스 설정

DB2 Cube Views는 CVSAMPLE 샘플 데이터베이스를 작성하는 데 사용할 수 있는 샘플 데이터를 제공합니다.

데이터베이스 연결 문제점 해결

OLAP 센터를 사용하여 데이터베이스에 연결할 수 없는 경우 DB2 Cube Views 버전이 DB2 카탈로그의 메타데이터 테이블 버전과 일치하는지 점검하십시오.

DB2 Cube Views 향상

이 릴리스에는 DB2 Cube Views V8.1 이후 DB2 Cube Views 메타데이터에 적용된 여러 변경사항이 들어 있습니다.

요약 테이블을 향상시키는 제한조건 및 함수 종속성 정보

DB2 Cube Views는 제한조건 및 함수 종속성을 사용하여 크기가 더 작으며 쿼리 적용 범위가 더 나은 요약 테이블을 생성합니다.

제한조건 정보

DB2 Cube Views에서 큐브 모델 제한조건은 사실 및 차원 테이블 사이 또는 snowflake 차원 테이블 사이에 정의됩니다. 요약 테이블에 차원 테이블의 기본 키가 들어 있으면 DB2 UDB는 쿼리가 차원의 다른 컬럼을 참조해도 해당 쿼리를 요약 테이블에 다시 라우트할 수 있습니다.

함수 종속성 정보

DB2 UDB에는 SQL 쿼리를 해석하는 함수 종속성 정보에 기반한 인텔리전트 라우팅 성능이 있습니다.

모든 컬럼이 동일한 테이블에 존재하는 경우 함수 종속성을 통해 하나 이상의 컬럼이 하나 이상의 다른 컬럼에 기능적으로 종속되도록 지정할 수 있습니다.

SQL 쿼리에 요약 테이블의 컬럼 및 요약 테이블이 구성된 테이블의 컬럼이 들어 있으며 이때의 두 컬럼 사이에 함수 종속성이 있는 경우 DB2 옵티마이저는 마지막 결과 세트를 얻기 위해 두 테이블을 함께 조인하여 쿼리를 해석할 수 있습니다.

DB2 Cube Views는 결과적으로 요약 테이블의 범위를 더 좁아지도록 선택된 차원 및 레벨의 키 컬럼만 포함하는 요약 테이블을 권장할 수 있습니다.

요약 테이블을 향상시키는 사용자 지정 최적화 슬라이스

현재 이 DB2 Cube Views 릴리스에는 사용할 쿼리 유형 및 이들이 참조할 큐브 영역을 지정하는 메커니즘이 들어 있습니다. 최적화 어드바이저는 이 추가 메타데이터를 사용하여 해당 큐브 영역의 최적화에 초점을 맞출 수 있습니다. 어떤 쿼리 유형을 사용해도 성능을 향상시킬 수 있지만 보고서 쿼리가 성능을 가장 많이 향상시킬 수 있습니다.

메타데이터 API에서 지원하는 다중 XML 스키마 버전

DB2 Cube Views 메타데이터 API는 새 DESCRIBE 및 TRANSLATE 조작을 포함하여 XML 스키마 버전 번호 8.2.0.1.0만 지원합니다.

메타데이터 API에서 전송하거나 이 API로 전송되는 모든 XML 문서에 버전 번호가 있어야 합니다. 버전 번호에 따라 스토어드 프로시저는 클라이언트가 사용하는 XML 스키마를 식별할 수 있습니다.

속성 및 치수의 널(NULL) 입력 가능성

메타데이터 API 속성 및 치수가 널(NULL) 입력 가능이라는 읽기 전용 등록 정보를 포함하도록 확장됩니다. 이 등록 정보는 예, 아니오 또는 알 수 없음 값으로 구성된 값 세트입니다.

널(NULL) 입력 가능의 예

속성 또는 치수가 널(NULL) 값을 가질 수 있습니다.

널(NULL) 입력 가능의 아니오

속성 또는 치수가 널(NULL) 값을 가질 수 없습니다.

널(NULL) 입력 가능의 알 수 없음

널(NULL) 입력 가능은 API를 통해 또는 속성이나 치수가 V8.1에서 V8.2로 이주된 경우 이주 유틸리티를 통해 판별될 수 없습니다.

레벨을 사용하는 계층 구조 모델링

이전 DB2 Cube Views 릴리스의 경우 계층 구조는 속성 및 속성 관계 오브젝트로 구성되어 있는 정렬된 속성 목록에서 모델화됩니다. 현재 이 릴리스의 경우 계층 구조는 각 레벨이 하나 이상의 속성을 참조하는 레벨 오브젝트의 정렬된 목록에서 모델화됩니다.

이 릴리스에는 사용자가 새 레벨 오브젝트를 작성 및 수정할 수 있는 새 레벨 마법사 및 레벨 등록 정보 창이 들어 있습니다.

레벨 오브젝트의 함수 종속성

레벨을 작성할 때 DB2 Cube Views가 함수 종속성 오브젝트를 작성하도록 요청할 수 있습니다.

함수 종속성은 레벨 오브젝트의 디폴트 속성 및 관련 속성이 레벨의 키 속성을 통해 기능적으로 판별됨을 표시합니다. 최적화 어드바이저는 이러한 함수 종속성을 사용하여 권장 요약 테이블의 크기를 최소화할 수 있습니다.

함수 종속성을 작성할 수 없으면 경고 메시지가 리턴됩니다. 예를 들어 함수 종속성은 둘 이상의 테이블에 있는 컬럼을 지정할 수 없습니다.

제한사항: 기본 컬럼이 레벨 오브젝트의 함수 종속성에 정의된 관계를 준수하지 않는 경우 쿼리 결과는 올바르지 않을 수 있습니다. DB2 UDB가 함수 종속성에 관하여 데이터의 유효성을 확인하지 않기 때문에 테이블 컬럼의 데이터가 사용자가 지정한 방식을 통해 기능적으로 종속되는지 확인해야 합니다.

함수 종속성에 대한 자세한 정보는 30 페이지의 『레벨』을 참조하십시오.

최적화 어드바이저의 자동화된 조작

이 릴리스에는 새 API 권고 조작이 들어 있습니다. 이 조작은 큐브 모델의 쿼리 성능을 향상시키기 위해 빌드해야 하는 요약 테이블을 권장합니다. 이 조작에는 요약 테이블에서 최적화 어드바이저가 실행될 수 있는 시간 및 사용할 수 있는 디스크 스페이스 크기를 제한하는 인수가 있습니다.

갱신된 CVSAMPLE 샘플 데이터베이스

향상된 CVSAMPLE 샘플 데이터베이스가 새로 제공됩니다. CVSAMPLE 데이터베이스는 보다 강화된 snowflake 스키마입니다. 이 릴리스의 메타데이터 변경사항을 반영하는 DB2 Cube Views 메타데이터도 제공됩니다. CVSAMPLE 데이터베이스 설정에 대한 정보는 9 페이지의 『CVSAMPLE 샘플 데이터베이스 설정』을 참조하십시오.

DB2 Cube Views 소개

DB2 Cube Views는 OLAP 처리를 수행하도록 DB2 UDB 기능을 향상시키는 DB2 Universal Database의 추가 기능입니다. DB2 Cube Views를 사용하여 OLAP 솔루션의 전개 및 관리를 간소화(streamline)하고 OLAP 도구 및 응용프로그램의 성능을 향상시킬 수 있습니다.

DB2 Cube Views를 사용하여 관계형 테이블의 차원 구조를 설명하고 OLAP 구문을 작성할 수 있습니다. 구조 정보 및 OLAP 구문을 DB2 데이터베이스의 다차원 메타데이터로 저장할 수 있습니다.

DB2 UDB의 새 다차원 메타데이터는 다음의 두 가지 주요 혜택을 제공합니다.

비즈니스 인텔리전스 도구 및 응용프로그램 간 다차원 메타데이터의 플로우 향상

웨어하우징 및 비즈니스 인텔리전스 도구의 사용자는 제공되는 그래픽 인터페이스인 OLAP 센터를 사용하여 DB2 데이터베이스의 일부로 다차원 메타데이터를 저장하여 모든 도구 및 응용프로그램에서 이를 사용할 수 있도록 합니다.

OLAP 스타일 쿼리의 성능 향상

다차원 메타데이터에 기반하여 OLAP 센터에서 최적화 어드바이저의 권장사항을 통해 DB2 요약 테이블을 작성할 수 있습니다. 요약 테이블에는 OLAP 구조에 맵핑할 사전 계산된 데이터가 들어 있습니다. OLAP 구조가 동일한 웨어하우징 또는 비즈니스 인텔리전스 응용프로그램에서 생성된 쿼리의 성능이 향상됩니다.

DB2 Cube Views는 요약 테이블, 서로 다른 인덱스 스킴, OLAP 스타일 연산자 및 집계 함수와 같은 DB2 기능을 사용합니다. 다음 구성요소가 제공됩니다.

다차원 메타데이터 오브젝트

메타데이터 오브젝트 세트를 작성하여 관계형 데이터 및 OLAP 구조를 차원으로

로 모델화할 수 있습니다. DB2 Cube Views는 DB2 카탈로그에서 사용자가 작성하는 각 메타데이터 오브젝트를 저장합니다.

OLAP 센터

OLAP 센터에서 큐브 모델, 큐브 및 OLAP 도구에서 사용될 기타 메타데이터 오브젝트를 작성, 조작, импорт 또는 익스포트할 수 있습니다. OLAP 센터에서는 메타데이터 오브젝트에 대한 작업을 수행하는 데 도움이 되는 사용하기 쉬운 마법사 및 창을 제공합니다. 예를 들어 최적화 어드바이저는 메타데이터 오브젝트를 분석하여 OLAP 스타일 SQL 쿼리에서 집계된 데이터를 저장하고 색인화하는 요약 테이블을 빌드하는 방법을 권장합니다. OLAP 센터를 시작하려면 OLAP 센터 시작을 참조하십시오. OLAP 센터를 시작한 후 최적화 어드바이저 마법사를 사용하려면 큐브 모델 최적화를 참조하십시오.

다차원 서비스

DB2 Cube Views는 OLAP 도구 및 응용프로그램 개발자에게 SQL 기반 및 XML 기반 API(Application Program Interface)를 제공합니다. CLI, ODBC 또는 JDBC 연결을 사용하거나 DB2 UDB의 Embedded SQL을 사용하는 경우 응용프로그램 및 도구는 단일 스토어드 프로시저를 사용하여 메타데이터 오브젝트를 작성, 수정 및 검색할 수 있습니다.

샘플 데이터

제품 사용법 학습에 도움이 되도록 샘플 응용프로그램 및 데이터베이스를 사용할 수 있습니다.

DB2 카탈로그 및 OLAP 도구 사이에서 메타데이터 오브젝트를 교환할 수도 있습니다. DB2 카탈로그에서 메타데이터 오브젝트를 импорт 또는 익스포트하는 데 특정 OLAP 및 데이터베이스 도구에서 메타데이터 브릿지 유틸리티를 사용할 수 있습니다. 메타데이터 브릿지가 제공되는지 판별하려면 특정 OLAP 또는 데이터베이스 도구에 관한 문서를 참조하십시오.

설치 요구사항

DB2 Cube Views를 설치하기 전에 모든 요구사항을 충족시켰는지 확인하십시오.

시스템 요구사항

연결하려는 각 DB2 UDB 서버 및 클라이언트에 적절한 구성요소를 설치해야 합니다. 서버 및 클라이언트에 다차원 서비스가 필요합니다. 클라이언트에 OLAP 센터를 설치할 수도 있습니다.

제한사항: 모든 클라이언트 및 서버 구성요소에 동일한 버전의 DB2 Cube Views를 설치해야 합니다. OLAP 센터, 버전 8.1 클라이언트는 DB2 UDB, 버전 8.2 서버에 연결될 수 있지만 작성, 변경 또는 삭제 작업을 수행할 수 없습니다.

다음 서버, 클라이언트 및 하드웨어 구성요소가 있어야 합니다.

- 서버 구성요소:

Microsoft® Windows®

Windows NT® 4, Windows 2000® 32비트, Windows XP Professional 32비트, Windows Server 2003 32비트 또는 Windows Server 2003 64비트

AIX AIX 버전 4.3.3 32비트, AIX 5L™ 32비트 또는 AIX 5L 64비트

Linux®:

Linux Red Hat™ 8(커널 2.4.18, glibc 2.2.93-5) 32비트, Linux SuSE 8.0(커널 2.4.18, glibc 2.2.5) 32비트, Linux SLES 8 SP3(커널 2.4.21, glibc 2.2.5) 32비트 또는 Linux RHEL 3 Update 2(커널 2.4.21-9, glibc 2.3.2.) 32비트.

DB2 UDB에서 지원하는 분산 및 커널 레벨에 관한 최신 정보를 보려면 www.ibm.com/db2/linux/validate로 이동하십시오.

Sun Solaris™ 운영 체제

Solaris 8 32비트, Solaris 8 64비트, Solaris 9 32비트 또는 Solaris 9 64비트

HP-UX

Intel™ Itanium용 HP-UX 11i v2 64비트

- 클라이언트 구성요소: Windows NT 4, Windows 2000 32비트, Windows XP 32비트, Windows XP 64비트, Windows Server 2003 32비트 또는 Windows Server 2003 64비트

- 하드웨어 구성요소:

- 500MB 디스크 스페이스
- 256MB RAM

DB2 Cube Views 설치 시 전제조건

DB2 Cube Views를 설치하기 전에 다음 구성요소를 설치하십시오.

DB2 정보 센터

OLAP 센터의 온라인 도움말 및 *DB2 Cube Views* 안내 및 참조서의 온라인 버전에 액세스하려면 DB2 정보 센터를 설치해야 합니다. DB2 정보 센터 CD로 DB2 정보 센터를 설치하십시오.

DB2 Universal Database, 버전 8.2

DB2 Cube Views를 설치하기 전에 DB2 UDB, 버전 8.2를 설치해야 합니다.

주: DB2 UDB, 버전 8.2를 설치하기 전에 DB2 UDB의 이전 버전 모듈을 제거하십시오.

DB2 Cube Views 설치

설치 런치패드를 사용하여 DB2 Cube Views를 Windows, AIX, Linux 및 Solaris 운영 체제 플랫폼에 설치하십시오.

Windows에 DB2 Cube Views 설치

DB2 Cube Views를 설치하기 전에 다음을 확인하십시오.

- DB2 정보 센터를 설치했습니다.
- DB2 Universal Database, 버전 8.2를 완전히 설치했습니다.
- 모든 설치 요구사항을 충족합니다.

Windows에 DB2 Cube Views를 설치하려면 다음을 수행하십시오.

1. DB2 Cube Views CD를 넣으십시오. 설치 프로그램이 자동으로 시작됩니다.
2. 런치패드에서 최신 설치, 디스크 및 메모리 요구사항의 릴리스 정보를 누르십시오. 또한 추가 지시사항이 있는지 DB2 Cube Views CD의 루트에 있는 readme.txt 파일을 점검하십시오.
3. 제품 설치를 눌러 설치를 시작하고 프롬프트를 따르십시오.

AIX, Linux 또는 Solaris 운영 체제에 DB2 Cube Views 설치

DB2 Cube Views를 설치하기 전에 다음을 확인하십시오.

- DB2 Universal Database, 버전 8.2를 완전히 설치했습니다.
- 모든 설치 요구사항을 충족합니다.

AIX, Linux 또는 Solaris 운영 체제에 DB2 Cube Views를 설치하려면 다음을 수행하십시오.

1. DB2 Cube Views CD를 넣으십시오.
2. UNIX 운영 체제의 디렉토리로 전환한 후 db2setup 파일을 실행하십시오.
3. 런치패드에서 최신 설치, 디스크 및 메모리 요구사항의 릴리스 정보를 누르십시오. 또한 추가 지시사항이 있는지 DB2 Cube Views CD의 루트에 있는 readme.txt 파일을 점검하십시오.
4. 제품 설치를 눌러 설치를 시작하고 프롬프트를 따르십시오.

DB2 Cube Views XML 파일을 V8.1에서 V8.2로 이주

스크립트를 직접 실행하거나 OLAP 센터를 사용하여 DB2 Cube Views 사용 가능 데이터베이스를 버전 8.1에서 버전 8.2로 이주시킬 수 있습니다.

DB2 스크립트로 이주

권장사항: 이 이주 스크립트를 실행하기 전에 먼저 버전 8.1 XML 메타데이터를 백업한 후 익스포트하십시오.

sqllib\misc 디렉토리의 db2mdmigrate.sql 파일을 사용하여 메타데이터를 V8.1에서 V8.2로 이주하는 데 필요한 추가 SQL 및 새 메타데이터 테이블을 작성하십시오. 이 파일을 사용하려면 다음을 수행하십시오.

1. 다음 명령을 입력하여 이주하려는 데이터베이스에 연결하십시오.

```
db2 connect to database_name
```

2. DB2 명령 창에서 다음 명령을 입력하여 db2mdmigrate.sql 파일을 실행하십시오.

```
db2 -tvf db2mdmigrate.sql
```

이 스크립트에는 오류 조절이 없습니다. 스크립트는 DDL 및 SQL문 세트를 통해 진행됩니다. 임의의 명령문에 실패하거나 스크립트를 완료하기 전에 이를 중단한 경우 이주가 부분적으로만 수행되며 DB2 Cube Views가 제대로 작동하지 않을 수 있습니다.

3. 오류를 발견하면, 다음 태스크를 수행하십시오.

- a. 모든 DB2INFO.* 테이블을 삭제하십시오.

- b. DB2INFO.MD_MESSAGE 스토어드 프로시저를 삭제하십시오.

- c. sqllib\misc 디렉토리의 db2mdapi.sql 파일을 사용하여 메타데이터 테이블을 다시 작성하십시오.

OLAP 센터로 이주

OLAP 센터를 DB2 UDB 데이터베이스에 연결하면 OLAP 센터가 DB2 UDB의 현재 버전을 자동으로 발견하고 메타데이터 테이블이 이주되어야 하는지를 판별합니다. OLAP 센터가 이주가 필요하다고 판별한 경우 오류 메시지를 표시하여 사용자에게 알립니다.

OLAP 센터의 이주 권장을 승인하면 OLAP 센터가 데이터베이스에 연결되어 메타데이터 테이블을 V8.2로 이주합니다. 이주를 거부하면 OLAP 센터가 데이터베이스에 연결되지 않습니다.

이주 진행 도중 오류가 발생하면 OLAP 센터가 트랜잭션을 롤백하고 데이터베이스가 이주되지 않습니다.

DB2 Cube Views에서 데이터베이스 설정

DB2 Cube Views에서 사용할 새 데이터베이스를 설정할 수 있습니다.

데이터베이스 설정에는 다음이 포함됩니다.

- 데이터베이스가 있는 DB2 Cube Views 스토어드 프로시저 등록
- DB2 Cube Views에서 메타데이터 카탈로그 테이블 작성

OLAP 센터를 사용하여 DB2 Cube Views에 구성되지 않은 데이터베이스에 처음 로그인하는 경우 OLAP 센터가 사용자에게 맞게 데이터베이스를 설정합니다. 또는 db2mdapi.sql 파일을 사용하여 데이터베이스를 설정할 수 있습니다.

중요사항: db2mdapi.sql 파일을 변경하지 마십시오. 그렇지 않으면 예기치 못한 결과가 발생합니다.

db2mdapi.sql 파일을 사용하여 데이터베이스를 설정하려면 다음을 수행하십시오.

1. DB2 명령 창을 열고 데이터베이스에 연결하십시오.
2. SQLLIB\misc 디렉토리로 변경한 후 DB2 명령 창에서 다음 명령을 입력하십시오.

```
db2 -tvf db2mdapi.sql
```

데이터베이스에서 db2mdapi.sql 스크립트를 한번만 실행하십시오. 오류가 발생하면 문제점을 수정한 후 다음 태스크를 수행하십시오.

- a. 모든 DB2INFO.* 테이블을 삭제하십시오.
- b. DB2INFO.MD_MESSAGE 스토어드 프로시저를 삭제하십시오.
- c. sqllib\misc 디렉토리의 db2mdapi.sql 파일을 사용하여 메타데이터 테이블을 다시 작성하십시오.

CVSAMPLE 샘플 데이터베이스 설정

DB2 Cube Views는 CVSAMPLE 샘플 데이터베이스를 작성하는 데 사용할 수 있는 샘플 데이터를 제공합니다.

샘플 데이터에는 가상 음료 판매 회사에 관한 데이터가 들어 있는 테이블 세트가 있습니다. 샘플 데이터 테이블을 설명하는 메타데이터 오브젝트 세트도 있습니다. 제공된 CVSAMPLE 데이터는 버전 8.1에 제공된 MDSAMPLE 데이터베이스의 향상 및 확장된 버전입니다. *DB2 Cube Views* 안내 및 참조서에 있는 대부분의 예는 CVSAMPLE 데이터베이스 및 이에 대응하는 큐브 모델에 기반합니다.

DB2 명령 창을 열고 다음 명령을 입력하여 샘플 CVSAMPLE 데이터베이스를 작성한 후 데이터로 채우십시오.

1. CVSAMPLE 샘플 데이터베이스를 작성하십시오.

```
db2 create db cvsample
```

2. 데이터베이스에 연결하십시오.

```
db2 connect to cvsample
```

3. db2mdapi.sql 스크립트를 실행하여 DB2 Cube Views에 데이터베이스를 설정하십시오. SQLLIB\misc 디렉토리로 변경한 후 다음 DB2 명령을 입력하십시오.

```
db2 -tvf db2mdapi.sql
```

4. SQLLIB\samples\olap\cvsample 디렉토리로 변경한 후 다음 DB2 명령을 입력하여 CVSAMPLE 테이블을 작성하십시오.

```
db2 -tvf CVSampleTables.sql
```

CVSAMPLE 데이터베이스를 작성한 경우 비즈니스 인텔리전스 응용프로그램에서 익스포트된 XML 파일의 정의를 임포트하여 DB2 Cube Views 메타데이터 오브젝트를 작성할 수 있습니다.

예를 들어 다음 프로시저는 CVSAMPLE 데이터베이스의 전체 설명으로 DB2 Cube Views 카탈로그 테이블을 채웁니다.

CVSAMPLE 메타데이터를 임포트하려면 다음을 수행하십시오.

1. OLAP 센터를 시작한 후 CVSAMPLE 데이터베이스에 연결하십시오.
2. **OLAP** 센터 -> **임포트**를 누르십시오.
3. SQLLIB/samples/olap/cvsample 디렉토리에 있는 CVSampleMetadata.xml 파일을 찾아보십시오. 다음을 누르십시오

OLAP 센터에서 메타데이터 오브젝트를 찾아볼 수 있습니다. OLAP 센터 사용에 대한 정보는 53 페이지의 『DB2 Cube Views 메타데이터 오브젝트 작성』을 참조하십시오.

데이터베이스 연결 문제점 해결

OLAP 센터를 사용하여 데이터베이스에 연결할 수 없는 경우 DB2 Cube Views 버전이 DB2 카탈로그의 메타데이터 테이블 버전과 일치하는지 점검하십시오.

다음 테이블은 DB2 Cube Views와 DB2 카탈로그의 메타데이터 테이블 버전과 서로 다른 경우 OLAP 센터의 동작을 표시합니다.

DB2 Cube Views 버전	DB2 카탈로그의 메타데이터 테이블 버전	OLAP 센터의 동작
설치되지 않음	적용할 수 없음	연결에 실패하고 OLAP 센터가 오류 메시지를 표시함
버전 8.1	없음	연결에 실패하고 OLAP 센터가 오류 메시지를 표시함

DB2 Cube Views 버전	DB2 카탈로그의 메타데이터 테이블 버전	OLAP 센터의 동작
버전 8.1	버전 8.1	연결에 실패하고 OLAP 센터가 오류 메시지를 표시함
버전 8.2	없음	OLAP 센터가 DB2 Cube Views, 버전 8.2에서 사용할 데이터베이스를 구성할 수 있음
버전 8.2	버전 8.1	OLAP 센터가 DB2 Cube Views, 버전 8.2에서 사용할 데이터베이스를 이주할 수 있음
버전 8.2	버전 8.2	제대로 연결되었습니다.

제 2 장 DB2 Cube Views 메타데이터 오브젝트 정보

DB2 Cube Views 메타데이터 오브젝트는 OLAP 구조로 관계형 테이블을 설명하지만, 이러한 메타데이터 오브젝트는 기존의 OLAP 오브젝트와는 차이가 있습니다. 메타데이터 오브젝트는 기본 테이블에 데이터에 대한 메타데이터를 저장하고, 관련 데이터가 있는 위치와 기본 데이터 내의 관계를 설명합니다.

DB2 Cube Views는 데이터를 이해하는 새 관점을 제공하는 메타데이터 오브젝트의 관계형 데이터에 관한 정보를 저장합니다. DB2 Cube Views는 DB2 카탈로그를 확장하여 테이블 및 컬럼에 대한 정보를 저장할 수 있도록 할 뿐만 아니라, 테이블 및 컬럼이 OLAP 오브젝트에 관련되는 방식과 해당 메타데이터 오브젝트 간의 관계에 대한 정보를 포함할 수 있도록 합니다.

일부 메타데이터 오브젝트는 데이터를 집계하거나 관계형 테이블의 특정 컬럼에 직접 대응됨으로써 관계형 데이터를 직접 액세스하기 위한 기본으로 작동합니다. 기타 메타데이터 오브젝트는 기본 메타데이터 오브젝트 간의 관계를 설명하고 기본 메타데이터 오브젝트를 함께 링크합니다. 모든 메타데이터 오브젝트는 서로의 관계에 의해 큐브 모델이라고 하는 메타데이터 오브젝트로 그룹화될 수 있습니다. 기본적으로 큐브 모델은 관계형 테이블의 특정 그룹 및 구성을 표현합니다.

DB2 Cube Views는 다음 메타데이터 오브젝트를 관리하고 DB2 카탈로그에 저장합니다.

- 큐브 모델
- 사실 오브젝트
- 차원
- 계층 구조
- 레벨
- 치수
- 속성
- 속성 관계
- 조인
- 큐브
- 큐브 사실 오브젝트
- 큐브 차원
- 큐브 계층 구조
- 큐브 레벨

관계형 테이블에 맵핑하는 메타데이터 오브젝트

큐브 모델을 다양한 방식으로 구성할 수 있지만, 관계형 스타 스키마 또는 snowflake 스키마를 표현하기 위해 작성되는 경우가 많습니다. 큐브 모델에는 관계형 데이터의 관계를 설명하는 메타데이터 오브젝트가 들어 있습니다.

스타 스키마는 중심에 사실 테이블이 있고 그 사실 테이블에 조인된 하나 이상의 차원 테이블이 있습니다. snowflake 스키마는 여러 테이블로 정의된 하나 이상의 차원과 같이 스타 스키마의 확장입니다. 단순 스타 스키마를 기반으로 하는 큐브 모델은 중심의 사실 오브젝트 주변에 작성됩니다. 사실 오브젝트에는 차원의 사실 테이블로부터 데이터를 집계하는 방법을 설명하는 치수 세트가 포함되어 있습니다. 치수는 관계형 테이블에 있는 컬럼의 데이터 계산을 설명하며 조인되어 사실 오브젝트를 작성합니다. 그림 1은 치수 및 사실 오브젝트가 관계형 데이터와 연관되는 방법을 보여줍니다.

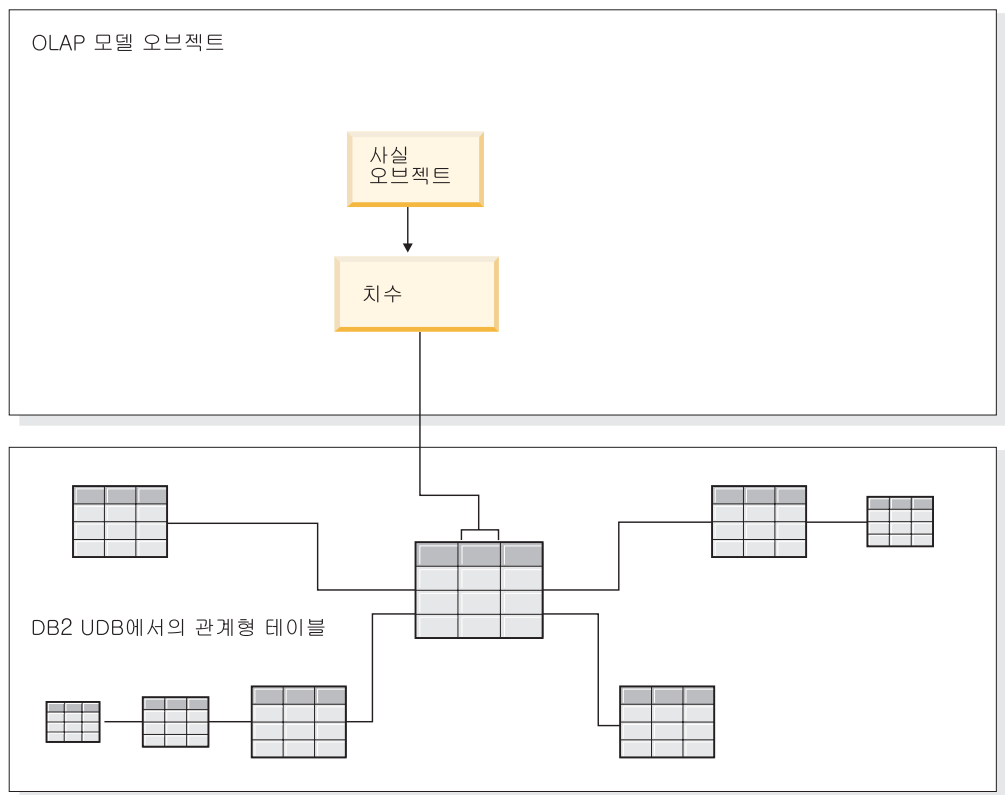


그림 1. 사실 오브젝트 및 치수가 관계형 데이터와 연관되는 방법

차원 테이블이 스타 스키마의 사실 테이블에 연결되는 것과 같이 차원은 큐브 모델의 사실 오브젝트에 연결됩니다. 관계형 테이블에서의 데이터 컬럼은 차원을 작성하기 위해 구성되는 속성에 의해 표현됩니다.

15 페이지의 그림 2는 관계형 테이블에서 차원이 작성되는 방법을 표시합니다. 계층 구조는 차원 내의 레벨이 상호 연관되고 구성되는 방법에 대한 정보를 저장합니다. 계층

구조는 차원 간을 계산하고 탐색하는 방법을 제공합니다. 각 차원에는 관련 속성 세트가 들어 있는 레벨을 포함하는 해당 차원에 대응하는 계층 구조가 있습니다. 큐브 모델에서 각각의 차원은 다중 계층 구조를 가질 수 있습니다.

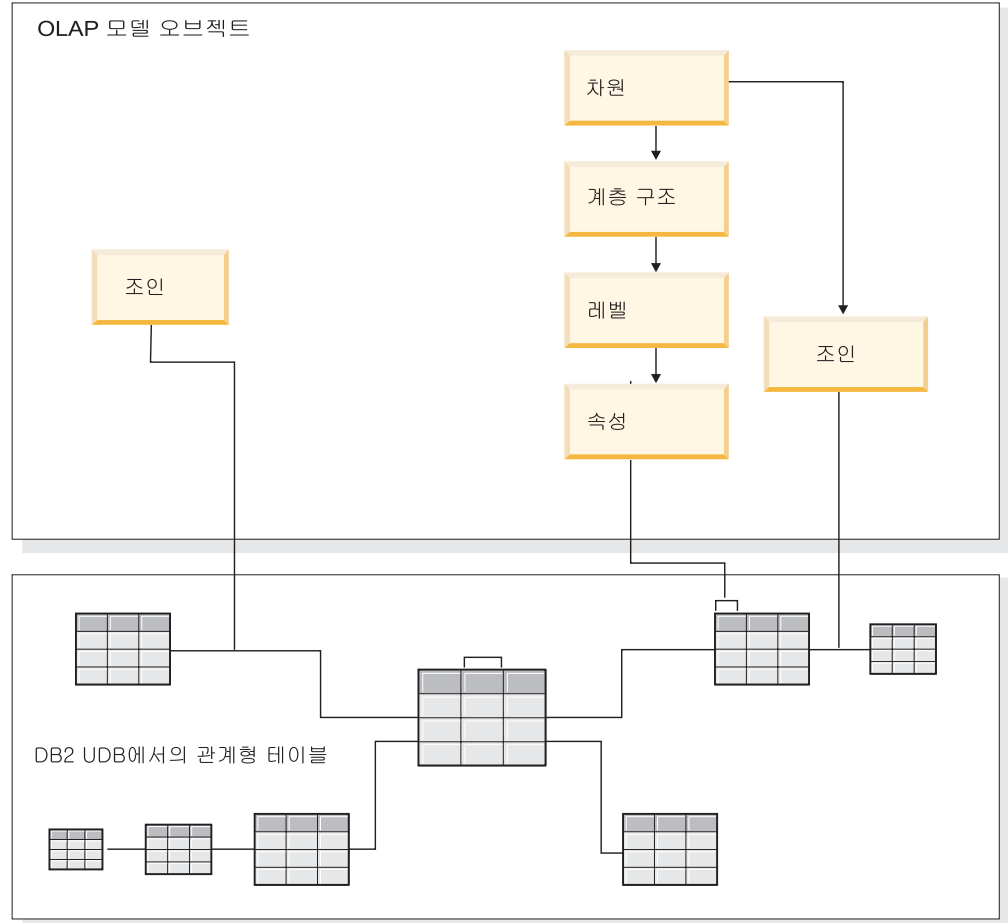


그림 2. 차원 관계형 테이블에서 차원을 작성하는 방법

모든 차원은 스타 스키마 또는 snowflake 스키마를 기반으로 하는 큐브 모델의 사실 오브젝트에 연결됩니다. 조인은 테이블을 연결하여 사실 오브젝트 또는 차원을 작성할 수 있습니다. 큐브 모델에서 조인은 사실 오브젝트를 차원에 연결할 수 있습니다. 차원은 대응하는 계층 구조, 레벨, 속성 및 관련 조인을 참조합니다. 사실 오브젝트는 해당 치수, 속성 및 관련 조인을 참조합니다. 16 페이지의 그림 3은 메타데이터 오브젝트가 큐브 모델에 맞게 조정되어 관계형 snowflake 스키마에 맵핑되는 방법을 표시합니다.

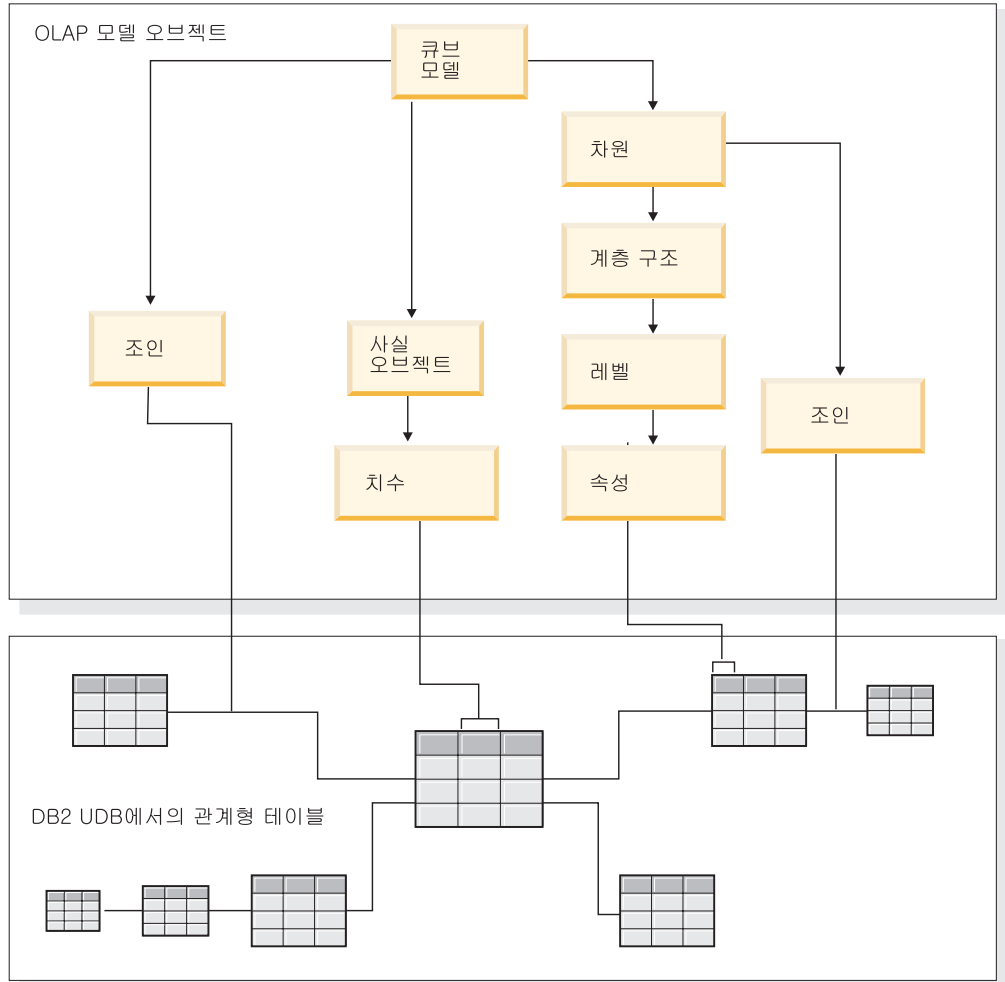


그림 3. 큐브 모델. 메타데이터 오브젝트가 적절하게 조정되어 관계형 snowflake 스키마에 맵핑되는 방법

큐브 모델의 구성요소를 다시 사용하여 특정 응용프로그램에 대한 보다 정밀한 큐브를 작성할 수 있습니다. 큐브는 가장 정밀한 메타데이터 오브젝트이며 OLAP 개념 큐브에 가장 근접한 오브젝트입니다. 큐브는 큐브 모델의 특정 인스턴스 또는 서브세트입니다. 큐브에는 상위 큐브 모델에서 파생된 유사하지만 보다 제한적인 특정 메타데이터 오브젝트 세트(큐브 차원, 큐브 계층 구조 및 큐브 사실 오브젝트)가 있습니다. 큐브에서는 큐브 차원마다 하나의 큐브 계층 구조만 정의할 수 있지만, 차원에서는 큐브 모델에 대해 다수의 계층 구조를 정의할 수 있습니다. 큐브 및 큐브 모델 간의 이러한 구조적 차이 때문에 단일 SQL문으로 대부분의 큐브를 검색할 수 있습니다.

관계형 테이블에 맵핑하는 큐브 모델의 예

큐브 모델 및 이에 대응하는 메타데이터 세트는 snowflake 스키마에 기반한 CVSAMPLE 데이터베이스에 빌드됩니다.

그림 4에서는 Store, Location, Time, Product, Line 및 Family 차원 테이블이 있는 Sales 사실 테이블의 snowflake 스키마를 표시합니다. Market 차원에는 기본 차원 테이블로 Store 및 아우트리거 차원 테이블로 Location과 같은 두 개의 차원 테이블이 있습니다. Product 차원에는 기본 차원 테이블로 Product 및 아우트리거 차원 테이블로 Line 및 Family와 같이 모두 세 개의 차원 테이블이 있습니다.

각 기본 차원 테이블(Store, Time 및 Product)의 기본 키는 Sales 사실 테이블에 대응하는 외부 키에 조인됩니다. 예를 들어 Store.StoreID = Sales.StoreID이고 Time.TimeID = Sales.TimeID이며 Product.ProductID = Sales.ProductID입니다.

snowflake 차원에서 각 차원 테이블의 기본 키는 다른 차원 테이블에 대응하는 외부 키에 조인됩니다. 예를 들어 Location.PostalcodeID = Store.PostalcodeID이고 Family.FamilyID = Line.FamilyID이며 Line.LineID = Product.LineID입니다.

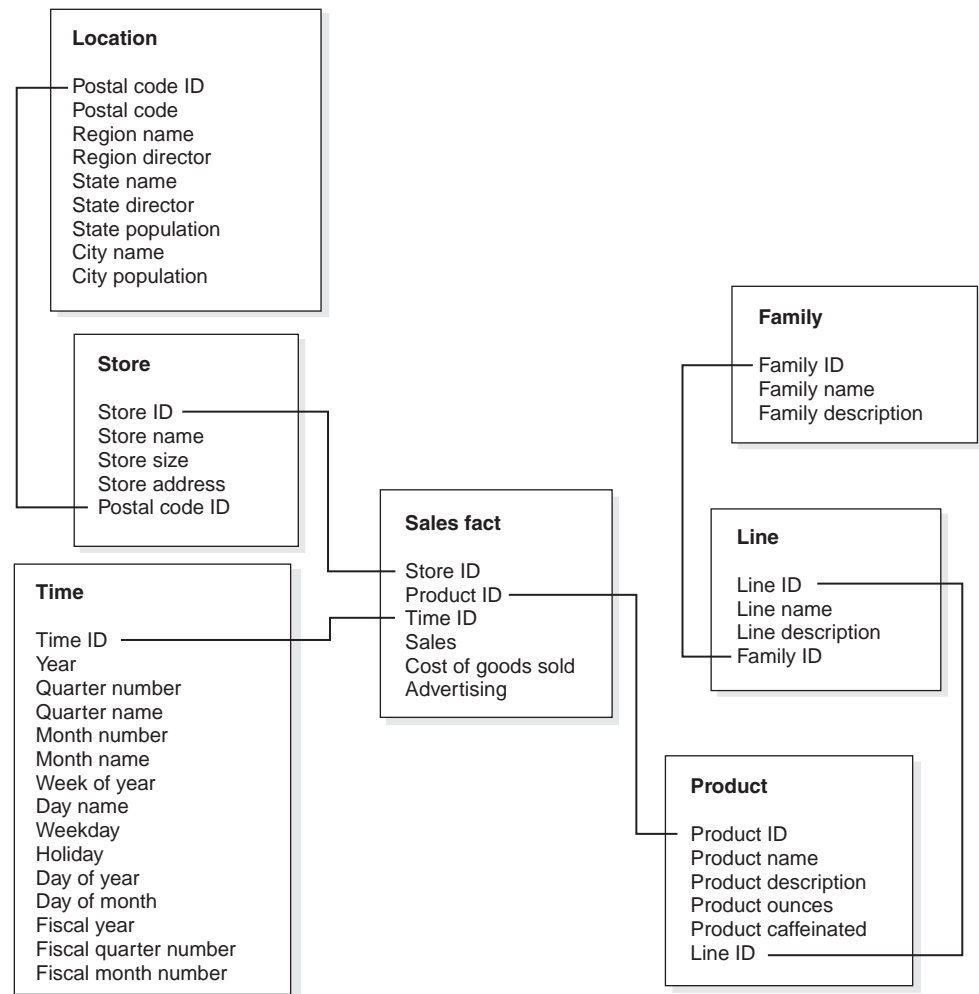


그림 4. snowflake 스키마. CVSAMPLE snowflake 스키마에 기반한 예

CVSAMPLE snowflake 스키마에 기반한 큐브 모델은 Sales 사실 테이블에서 집계된 관계형 데이터를 설명하는 Sales 사실 오브젝트에 빌드됩니다. 차수는 Sales 테이블에

있는 컬럼의 데이터를 계산하는 방법을 설명합니다. 또한 사실 오브젝트에는 차원을 사실 오브젝트에 조인하는 데 사용되는 사실 테이블의 외부 키에 대응하는 속성이 들어 있습니다. 이 예에서 Sales 사실 오브젝트에는 Sales, Cost of goods sold, Advertising, Total expense, Advertising-sales correlation, Profit 및 Profit margin과 같은 일곱 개의 치수가 들어 있습니다. Sales 사실 오브젝트에는 StoreID(Salesfact), TimeID(Salesfact) 및 ProductID(Salesfact)와 같은 세 개의 속성이 들어 있습니다.

차원 테이블이 스타 스키마의 사실 테이블에 연결되는 것과 같이 차원은 큐브 모델의 사실 오브젝트에 연결됩니다. 관계형 테이블에서의 데이터 컬럼은 차원에서 참조하는 속성 오브젝트에 의해 표현됩니다.

Product 차원에서는 다음 속성을 참조합니다.

- Family ID
- Family 이름
- Family 설명
- Line ID
- Line 이름
- Line description
- Product ID
- Product name
- Product description
- Product ounces
- Product caffeinated

Time 차원에서는 다음 속성을 참조합니다.

- Year
- Quarter name
- Quarter number
- Month name
- Month number
- Time ID
- Day of month
- Day name
- Day of week
- Holiday
- Weekday
- Fiscal year

- Fiscal quarter name
- Fiscal quarter number
- Fiscal month.

Market 차원에서는 다음 속성을 참조합니다.

- Region name
- Region director
- State name
- State director
- State population
- City name
- City population
- Postal code ID
- Postal code
- Store ID
- Store name
- Store size
- Store address.

각 차원을 사실 오브젝트에 연결하도록 조인이 작성됩니다. 이 예에서 세 개의 조인은 Product, Time 및 Store입니다. 20 페이지의 그림 5에서는 이 예에서 설명한 큐브 모델을 표시합니다.

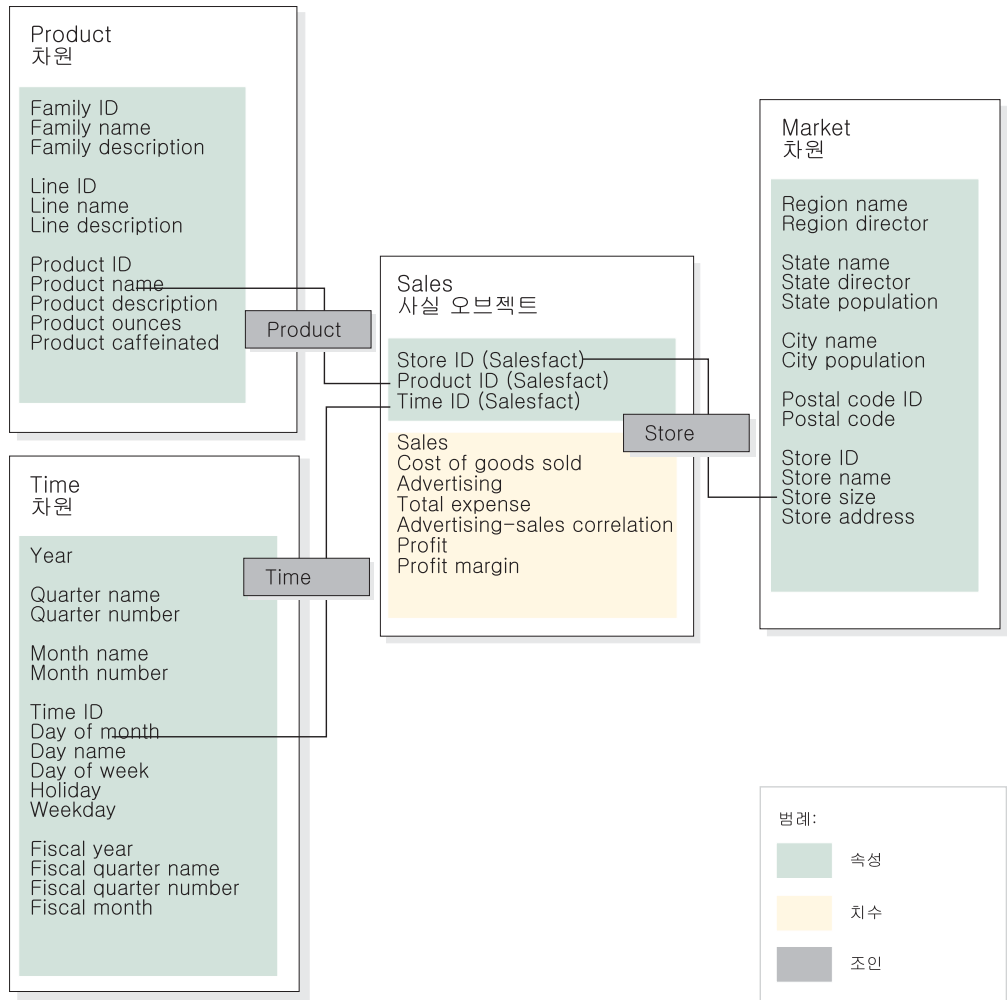


그림 5. 큐브 모델. CVSAMPLE 스타 스키마에 기반한 큐브 모델의 예

계층 구조는 차원 내의 레벨로 묶인 속성이 상호 연관되고 구성되는 방법에 대한 정보를 저장합니다. 메타데이터 오브젝트로서, 계층 구조는 차원 간을 계산하고 탐색하는 방법을 제공합니다. 각각의 차원에는 관련 속성을 묶는 레벨과 대응되는 계층 구조가 있습니다. 큐브 모델에서 각각의 차원은 다중 계층 구조를 가질 수 있습니다.

Product 계층 구조에는 21 페이지의 그림 6에 표시된 대로 Product 차원의 모든 속성이 들어 있습니다. Product 차원의 속성은 세 개의 레벨로 그룹화됩니다. Family 레벨은 Product 계층 구조의 최상위 레벨입니다. Family 레벨에는 레벨 키 속성으로 Family ID, 디폴트 속성으로 Family 이름 및 관련 속성으로 Family 설명이 들어 있습니다. 두 번째 레벨인 Line 레벨에는 레벨 키 속성으로 Line ID, 디폴트 속성으로 Line name 및 관련 속성으로 Line description 이 들어 있습니다. 맨 아래 레벨인 Product 레벨에는 레벨 키 속성으로 Product ID, 디폴트 속성으로 Product name 및 관련 속성으로 Product description, Product ounces 및 Product caffeinated가 들어 있습니다.

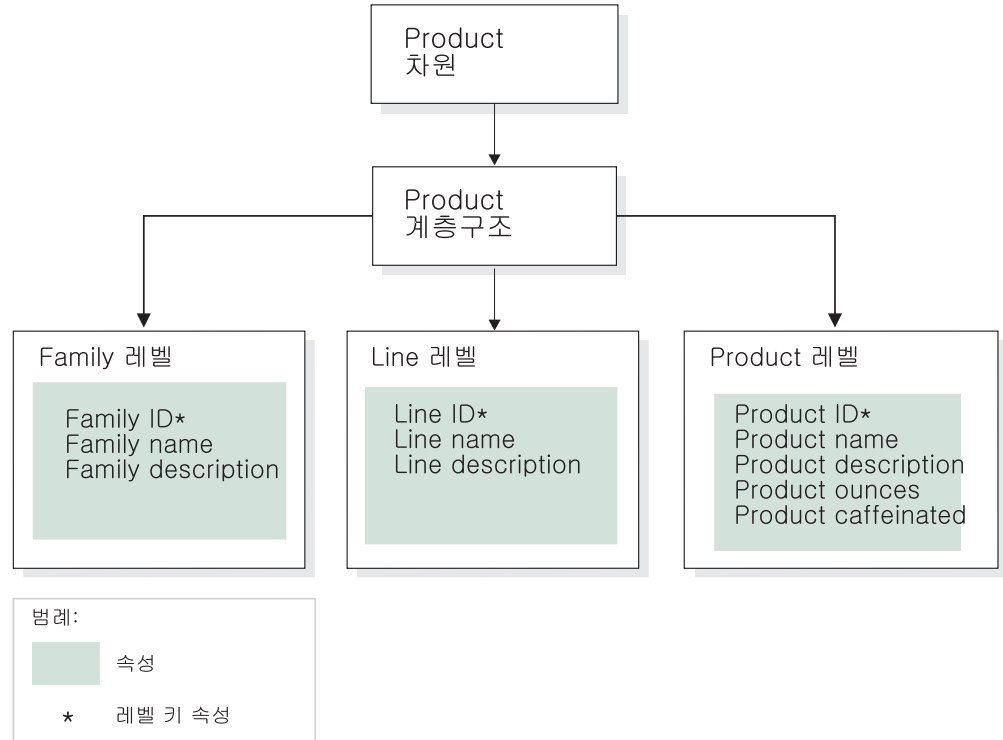


그림 6. 차원. CVSAMPLE 데이터베이스의 Product 차원에 기반한 차원의 예

큐브 모델에 하나 이상의 큐브를 빌드할 수도 있습니다. CVSAMPLE 데이터베이스에는 두 개의 큐브가 있지만 General Sales 큐브는 여기에서 설명됩니다. General Sales 큐브는 22 페이지의 그림 7에 표시됩니다. 큐브 사실 오브젝트는 큐브 모델 사실 오브젝트에서 치수(Sales, Cost of goods sold, Advertising 및 Total expense)의 서브세트를 참조합니다. 큐브에는 세 개의 큐브 차원이 있으며 각 큐브 차원은 큐브 모델의 세 차원 중 하나를 참조합니다. Product 큐브 차원에는 Family, Line 및 Product 레벨을 참조하는 큐브 레벨을 가진 Product 큐브 계층 구조가 있습니다. Market 큐브 차원에는 Region, State, City, Postal code 및 Store 레벨을 참조하는 큐브 레벨을 가진 Market 큐브 계층 구조가 있습니다. Time 큐브 차원에는 Year, Quarter 및 Month. 을 포함하여 레벨의 서브세트를 참조하는 큐브 레벨을 가진 Time 큐브 계층 구조가 있습니다. 큐브에는 큐브 차원당 정의된 하나의 큐브 계층 구조만 있습니다. (큐브는 큐브 차원당 하나의 큐브 계층 구조만 가질 수 있습니다.)

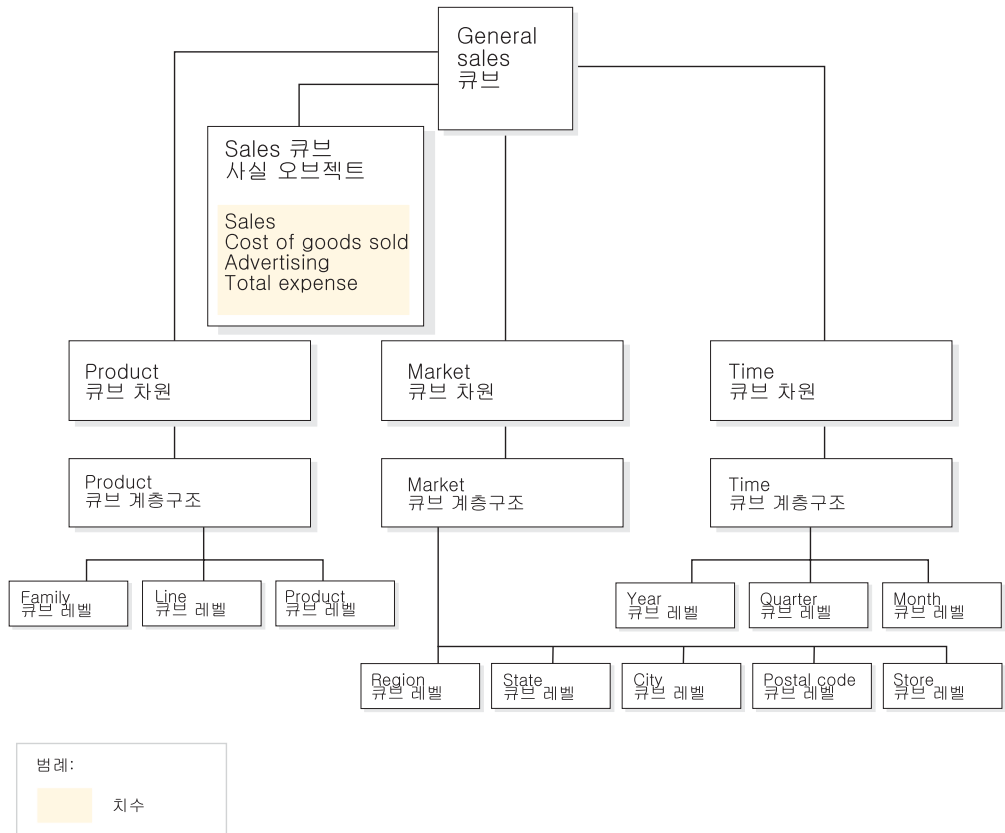


그림 7. 큐브 CVSAMPLE 데이터베이스의 General Sales 큐브에 기반한 큐브의 예

일반 메타데이터 오브젝트 등록 정보

각각의 메타데이터 오브젝트에는 일반 등록 정보 및 메타데이터 오브젝트 특정 등록 정보 세트가 있습니다. 일반 등록 정보는 메타데이터 오브젝트 인스턴스를 식별하고, 메타데이터 오브젝트 인스턴스의 사용법 또는 역할을 설명하며, 메타데이터 오브젝트 인스턴스의 변경사항을 추적하기 위해 사용됩니다. 메타데이터 오브젝트는 기타 DB2 오브젝트의 이름을 지정하는 방식과 동일하게 스키마를 사용하여 이름이 지정됩니다. 메타데이터 오브젝트에 대해 디폴트 사용자 이름 스키마를 사용하지 않으려면 원하는 스키마 이름으로 메타데이터 오브젝트를 완전히 규정하십시오.

다음 테이블은 모든 메타데이터 오브젝트에 대해 존재하는 일반 등록 정보를 설명합니다.

표 1. 일반 메타데이터 오브젝트 등록 정보

등록 정보	설명
이름	메타데이터 오브젝트의 이름입니다.
스키마	메타데이터 오브젝트를 소유하는 스키마입니다.

표 1. 일반 메타데이터 오브젝트 등록 정보 (계속)

등록 정보	설명
비즈니스 이름	사용자에게 제시되는 이름입니다. 이 이름은 그래픽 사용자 인터페이스에서 사용자에게 보다 의미있는 이름으로 사용될 수 있습니다.
주석	메타데이터 오브젝트의 특성 및 사용에 관한 텍스트 설명 또는 주석입니다.
작성 시간	메타데이터 오브젝트가 작성된 시간입니다.
작성자	메타데이터 오브젝트를 정의한 사용자(스키마)입니다.
수정 시간	메타데이터 오브젝트를 최종적으로 수정한 시간입니다.
수정자	수정을 수행한 사용자(스키마)입니다.

각각의 메타데이터 오브젝트에는 등록 정보의 일반 세트와 함께, 특정 등록 정보 세트가 있습니다. 이 특정 등록 정보는 메타데이터 오브젝트를 정의하는 특성 및 구성요소를 설명합니다. 각각의 메타데이터 오브젝트에 특정된 등록 정보에 대한 정보는 해당 메타데이터 오브젝트의 주제를 참조하십시오.

메타데이터 오브젝트 이름 지정 규칙

DB2 UDB는 이름 오브젝트에 일반 및 분리와 같은 서로 다른 두 개의 이름 지정 규칙을 제공합니다. 메타데이터 오브젝트의 경우 오브젝트 이름을 지정하고 DB2 테이블 및 컬럼을 참조할 때 분리 규칙을 사용합니다. 분리 규칙에서는 자국어 문자와 같은 특수 문자, 스페이스 및 대소문자 혼용 이름이 허용됩니다. 전체 문자 세트는 메타데이터 오브젝트가 저장된 데이터베이스의 코드 페이지에서 판별됩니다.

다음 규칙은 메타데이터 오브젝트에 적용됩니다.

표 2. 메타데이터 오브젝트의 이름 지정 규칙

오브젝트	규칙
스키마	<ul style="list-style-type: none"> 길이: 1 - 30바이트 제한된 이름: 스키마 이름이 <i>SESSION</i>이거나 <i>SYS</i>로 시작해서는 안됩니다. 오직 대문자로 된 이름이 제한됩니다.
메타데이터 오브젝트의 이름	<ul style="list-style-type: none"> 길이: 1 - 128바이트 다른 제한사항은 없습니다.
메타데이터 오브젝트의 비즈니스 이름	<ul style="list-style-type: none"> 길이: 1 - 128바이트 다른 제한사항은 없음
메타데이터 오브젝트의 주석	<ul style="list-style-type: none"> 길이: 0 - 254바이트 다른 제한사항은 없음
참조 컬럼에 사용하는 테이블 스키마	<ul style="list-style-type: none"> 길이: 1 - 128바이트 다른 제한사항은 없음
참조 컬럼에 사용되는 테이블 이름	<ul style="list-style-type: none"> 길이: 1 - 128바이트 다른 제한사항은 없음

표 2. 메타데이터 오브젝트의 이름 지정 규칙 (계속)

오브젝트	규칙
참조 컬럼에 사용되는 컬럼 이름	<ul style="list-style-type: none"> 길이: 1 - 128바이트 다른 제한사항은 없음

큐브 모델

DB2 Cube Views 큐브 모델은 논리적 스타 스키마 또는 snowflake 스키마를 표시한 것으로 중심 사실 오브젝트와 관련된 차원 오브젝트를 그룹화합니다.

각각의 차원은 다중 계층 구조를 가질 수 있습니다. 사실 오브젝트 및 차원에 의해 사용되는 테이블을 조인하는 방법에 대한 구조적 정보는 큐브 모델에 의해 참조됩니다. 또한 SQL 쿼리를 구성하고 OLAP 데이터를 검색할 수 있는 충분한 정보가 큐브 모델에 저장됩니다. 큐브 모델을 이해하고 특정 차원의 여러 뷰를 표시할 수 있는 기타 보고 및 OLAP 도구를 사용하면 큐브 모델 사용에 도움이 됩니다.

큐브 모델은 복잡한 관계 세트를 정의하며 관련 사실 오브젝트 및 차원을 응용프로그램에 선택적으로 노출시키는 데 사용할 수 있습니다. 차원을 중심의 사실 오브젝트에 연결하는 각각의 조인 오브젝트는 대응하는 차원과 함께 세트로 저장됩니다. 큐브 모델 구성요소의 서브세트를 많은 큐브들이 다양한 분석 목적으로 사용할 수 있습니다.

큐브 모델 마법사를 사용하여 OLAP 센터에서 빈 큐브 모델을 작성할 수 있습니다. 빈 큐브 모델에는 사실 오브젝트 또는 차원이 없습니다. OLAP 센터의 마법사로 사실 오브젝트 및 하나 이상의 차원을 작성하여 큐브 모델을 완성할 수 있습니다. 빠른 시작 마법사를 사용하여 완전한 큐브 모델을 작성할 수도 있습니다. DB2 Cube Views는 최적화 어드바이저를 열 때 해당 큐브 모델의 유효성을 확인합니다. 큐브 모델을 최적화하려면 큐브 모델에는 다음 필수 구성요소가 있어야 합니다.

- 사실 오브젝트
- 최소 한 개의 차원
- 하나 이상의 차원에 정의된 계층 구조
- 기존의 사실 오브젝트 및 차원 간의 조인
- 기존 테이블 컬럼을 참조하는 속성

큐브 모델에 고유한 등록 정보는 다음 테이블에 설명되어 있습니다.

표 3. 큐브 모델 등록 정보

등록 정보	설명
사실 오브젝트	큐브 모델에서 사용되는 사실 오브젝트입니다.
(차원, 조인) 세트	큐브 모델에 사용되는 차원 및 이와 대응되는 조인입니다.

사실 오브젝트

사실 오브젝트는 큐브 모델에서 스타 스키마의 중심으로 사용되며 특정 응용프로그램에 관심이 있는 관련 치수를 묶습니다.

사실 오브젝트는 사실 대 차원 조인에서 사용되는 속성 및 다중 데이터베이스 테이블에서 추가적인 치수의 맵핑에 사용되는 속성 및 조인을 참조합니다. 다중 관계형 사실 테이블을 특정 속성에 대해 조인하여 추가로 관련 치수를 맵핑할 수 있습니다. 따라서 치수 세트 외에도 사실 오브젝트는 속성 세트 및 조인 세트를 저장합니다.

OLAP 센터의 사실 마법사를 사용하여 사실 오브젝트를 작성할 수 있습니다. 사실 마법사에서 하나 이상의 사실 테이블 및 모든 필수 조인, 치수 및 치수에 대한 집계를 지정하십시오.

사실 오브젝트의 특정 등록 정보는 다음 테이블에 설명되어 있습니다.

표 4. 사실 오브젝트 등록 정보

등록 정보	설명
치수 세트	사실 오브젝트에서 모든 관련 치수의 세트입니다.
속성 세트	사실 오브젝트에서 사용되는 모든 속성의 세트입니다.
조인 세트	지정된 치수 및 속성을 모두 조인하는 데 필요한 모든 조인의 세트입니다.

차원

차원은 치수의 한 측면을 함께 설명하는 관련 속성 세트를 범주화하는 방법을 제공합니다. 차원은 Region, Product 또는 Time과 같은 논리적 범주에 따라 사실 오브젝트에서 데이터를 체계화하기 위해 큐브 모델에서 사용됩니다.

차원은 0개 이상의 계층 구조를 참조합니다. 계층 구조에서는 레벨로 그룹화되어 참조되는 속성의 관계 및 구조를 설명하며 차원을 탐색하는 탐색 및 계산 방법을 제공합니다.

이들 속성을 그룹화하기 위해 필요한 관련 속성 및 조인은 차원의 등록 정보에 정의되어 있습니다.

차원은 차원의 시간 지향성을 설명하는 유형도 가지고 있습니다. 예를 들어, 시간이라고 하는 차원은 Year, Quarter 및 Month와 같은 레벨을 포함합니다. Region, State 및 City와 같은 레벨이 들어 있는 Market이라는 다른 차원은 일반 유형입니다. 응용프로그램은 유형 정보를 사용하여 시간 관련 기능을 지능적으로 적절하게 수행합니다.

OLAP 센터의 차원 마법사를 사용하면 큐브 모델의 컨텍스트에서 또는 큐브 모델을 참조하지 않고 새 차원을 작성할 수 있습니다. 차원 추가 마법사를 통해 기존 차원을 큐브 모델에 추가하여 큐브 모델에서 차원을 공유할 수 있습니다.

차원의 특정 등록 정보는 다음 테이블에 설명되어 있습니다.

표 5. 차원 등록 정보

등록 정보	설명
속성 세트	차원에서 사용되는 모든 속성의 세트입니다.
조인 세트	지정된 속성을 모두 조인하는 데 필요한 모든 조인의 세트입니다. 차원 테이블의 조인에 필요한 조인만 여기에 지정됩니다.
계층 구조 세트	차원에 적용되는 계층 구조의 세트입니다.
레벨 세트	차원에 의해 참조되는 레벨의 세트입니다.
유형	일반 또는 시간의 차원 유형입니다.

계층 구조

계층 구조는 큐브 모델의 차원에서 묶는 속성 세트 간의 관계를 정의합니다. 이러한 관계는 차원을 이동하기 위한 탐색 및 계산 수단을 제공합니다. 큐브 모델의 차원에 대해 다중 계층 구조를 정의할 수 있습니다.

계층 구조 유형은 계층 구조 내 레벨 간의 관계를 설명합니다. 다음의 네 가지 계층 구조 유형이 지원됩니다.

균형 깊이가 일관적인 의미있는 레벨과 분기를 가진 계층 구조입니다. 각 레벨의 논리적 상위는 해당 속성 바로 위의 레벨에 있습니다. 균형 계층 구조는 Year, Quarter 및 Month와 같이 각 레벨의 의미와 깊이가 일관적인 시간을 표현할 수 있습니다. 각각의 레벨이 같은 유형의 정보를 표현하고 논리적으로 동등하기 때문에 이들은 일관적입니다. 그림 8은 균형된 Time 계층 구조의 예를 보여줍니다.

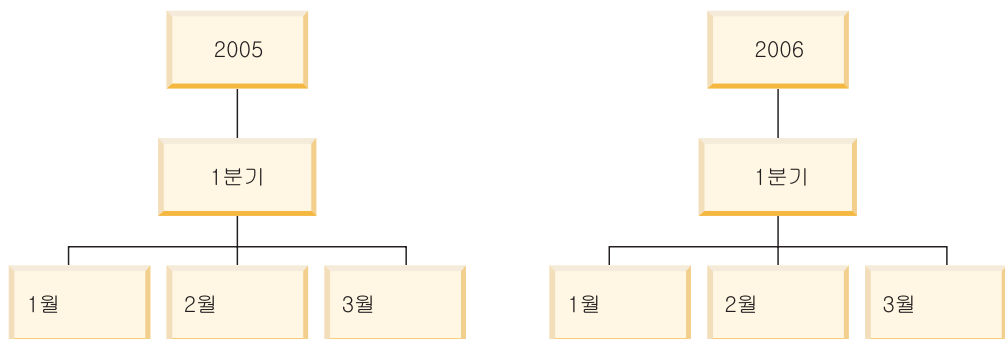


그림 8. 균형 계층 구조 균형 계층 구조의 예

불균형 일관적인 상위-하위 관계를 갖지만 논리적으로 일관적이지 않은 레벨을 갖는 레벨의 계층 구조입니다. 계층 구조 분기의 깊이도 일관적이지 않을 수 있습니다. 불균형 계층 구조는 조직 차트를 표현할 수 있습니다. 예를 들어, 27 페이지의 그림 9는 계층 구조의 최상위 레벨에 CEO(Chief Executive Officer)를 표시하며 COO(Chief Operating Officer) 및 ES(Executive Secretary)를 포함하

여 아래로 분기될 수 있는 최소한 두 명의 사람을 표시합니다. COO에는 추가로 분기될 수 있는 사람들이 많이 있지만 ES는 그렇지 않습니다. 계층 구조의 양쪽 분기에 대한 상위-하위 관계는 일관적입니다. 그러나, 양쪽 분기의 레벨은 논리적으로 동등하지 않습니다. 예를 들어, ES는 COO와 논리적으로 동등하지 않습니다.

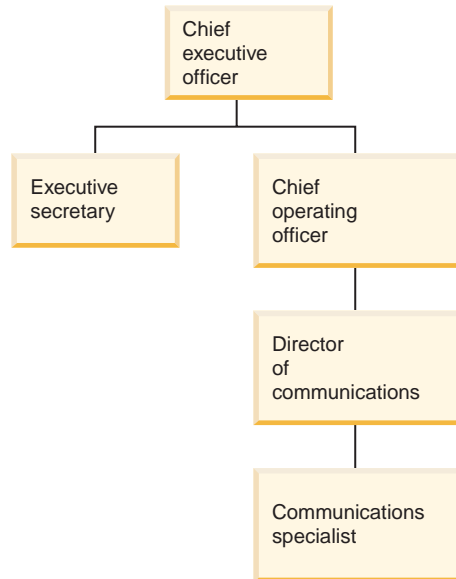


그림 9. 불균형 계층 구조 불균형 계층 구조의 예

불완전 각 레벨은 일관적인 의미를 갖지만 Quarter 레벨의 최소 한 구성원 속성이 채워져 있지 않아서 분기의 깊이가 일관적이지 않은 계층 구조입니다. 불완전 계층 구조는 지리적 계층 구조를 표현할 수 있습니다. 여기서 도시 또는 Country와 같은 각 레벨의 의미는 일관적으로 사용되지만 계층 구조의 깊이는 다양합니다. 28 페이지의 그림 10은 Continent, Country, Province/State 및 City 레벨이 정의되어 있는 지리적 계층 구조를 보여줍니다. 한 분기는 북아메리카를 대륙으로, 미국을 국가로, 캘리포니아를 지방/주로, 그리고 샌프란시스코를 시로 갖습니다. 그러나 한 구성원이 일부 레벨에서 항목을 갖지 않는 경우에는 불완전 계층 구조가 됩니다. 예를 들어, 다른 분기가 유럽을 대륙으로, 그리스를 국가로, 아테네를 시로 갖지만, Province/State 레벨은 이 예에서 비즈니스 모델에 대해 그리스에 적용되지 않으므로 이 레벨에 대한 항목을 갖지 않을 수 있습니다. 이 예에서 그리스 및 미국 분기는 상이한 깊이로 내려가므로 불완전 계층 구조를 형성합니다.



그림 10. 불완전 계층 구조 불완전 계층 구조의 예

네트워크

레벨의 순서가 지정되어 있지는 않지만 레벨이 의미론적 의미를 갖는 계층 구조입니다. 예를 들어, 그림 11은 색상, 크기 및 패키지 유형과 같은 Product 속성을 설명하는 네트워크 계층 구조를 표시합니다. 레벨에 고유한 상위-하위 관계가 없으므로 레벨의 순서는 중요하지 않습니다. 부품 회사는 색상으로 흰색, 크기로 소형, 그리고 패키지 유형으로 수축 포장과 같은 구성원 항목을 가질 수 있습니다. 두 번째 구성원 항목은 색상으로 빨간색, 크기로 대형, 패키지 유형으로 상자가 될 수 있습니다.

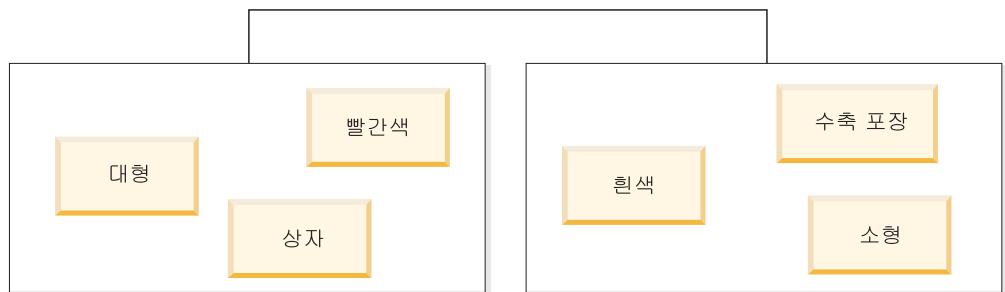


그림 11. 네트워크 계층 구조 네트워크 계층 구조의 예

계층 구조는 계층 구조에 대한 전개 메커니즘도 지정합니다. 전개 메커니즘은 계층 구조의 레벨에서 데이터를 해석하는 방법을 정의합니다. 다음의 두 가지 전개 메커니즘이 지원됩니다.

표준 계층 구조의 레벨 정의 관계를 사용합니다. 여기서 계층 구조의 각 레벨은 전

개에서 하나의 항목으로 사용됩니다. 예를 들어, Time 차원에 대한 균형 계층 구조는 Year, Quarter 및 Month와 같은 각각의 정의된 레벨에 의해 구성될 수 있습니다. 표준 전개는 네 개의 모든 계층 구조 유형과 함께 사용 가능합니다. 표 6은 Time 차원에 대한 균형 계층 구조 속성 중 일부를 표준 전개로 구성하는 방법을 보여줍니다.

표 6. 표준 전개. Time 차원에 대한 균형 계층 구조의 표준 전개

Year	Quarter	Month
2003	1/4분기	1
2003	1/4분기	2
2003	1/4분기	3
2004	1/4분기	1
2004	1/4분기	2
2004	1/4분기	3

반복 계층 구조의 레벨 간에 고유한 상위-하위 관계를 사용합니다. 반복 전개를 사용하는 불균형 계층 구조는 상위-하위 레벨 쌍으로 표현됩니다. 예를 들어, 표 7에서는 27 페이지의 그림 9에 표시된 조직 차트를 설명하는 불균형 계층 구조에 대한 레벨 쌍을 보여줍니다. 상위-하위 레벨 쌍에는 CEO 및 executive secretary, CEO 및 operating officer, chief operating officer 및 director of communications, director of communications 및 communications specialist의 쌍이 포함됩니다. 반복 전개는 불균형 계층 구조에서만 사용 가능합니다.

표 7. 반복 전개. 조직 차원에 대한 불균형 계층 구조의 반복 전개

상위 속성	하위 속성
CEO	Executive secretary
CEO	Chief operating officer
Chief operating officer	Director of communications
Director of communications	Communications specialist

계층 구조 마법사를 사용하여 OLAP 센터에서 계층 구조를 작성할 수 있습니다. 차원을 작성한 이후에 차원에 대한 계층 구조를 정의할 수 있습니다.

계층 구조 오브젝트의 등록 정보는 다음 테이블에 설명되어 있습니다.

표 8. 계층 구조 등록 정보

등록 정보	설명
정렬된 레벨 세트	계층 구조 레벨의 내림차순 목록입니다.
유형	계층 구조 유형은 균형, 불균형, 불완전, 네트워크가 될 수 있습니다.
전개	계층 구조 전개는 표준, 반복이 될 수 있습니다.

레벨

레벨은 계층 구조 정렬에 하나의 논리적 단계로 함께 작동하는 속성 세트로 구성됩니다. 레벨에는 서로 관련되어 레벨에서 하나 이상의 역할로 기능할 수 있는 속성이 하나 이상 들어 있습니다. 레벨의 속성 간 관계는 보통 함수 종속성으로 정의됩니다.

레벨은 다음과 같이 세 방법 중 하나로 속성을 사용할 수 있습니다.

레벨 키 속성

값이 각 레벨 인스턴스를 고유하게 식별하는 하나 이상의 속성입니다. 레벨 키 속성을 정의하는 가장 단순한 방법은 City 레벨의 경우 City ID와 같이 고유한 ID 컬럼을 사용하는 것입니다. City name 속성은 도시 이름이 여러 지방 및 국가에서 반드시 고유하지 않으므로 레벨 키 속성이 될 수 없습니다. 하지만 Country name, State name 및 City name 속성 세트가 도시를 고유하게 정의할 수 있으므로 이 세 속성 세트를 레벨 키 속성으로 포함할 수 있습니다. DB2 Cube Views는 사용자가 정의한 레벨에 기반하여 최적화 성능을 향상시키기 위해 함수 종속성을 작성하므로 레벨 키 속성이 해당 레벨을 고유하게 정의하도록 해야 합니다.

디폴트 속성

데이터의 각 레벨 인스턴스에 유용한 이름을 제공하는 보고 응용프로그램을 통해 값이 표시될 수 있는 속성입니다. 디폴트 속성은 필수이고 레벨 키 속성을 통해 기능적으로 판별되어야 합니다. 예를 들어 레벨 키 속성이 도시 ID로 정의되면 도시 ID 컬럼의 값은 보고서에서 유용하지 않습니다. 표시되는 데이터가 유용하도록 보고서에 표시된 도시 이름을 디폴트 속성으로 정의할 수 있습니다. 레벨 키 속성 세트의 속성을 디폴트 속성으로 사용할 수 있습니다. 예를 들어 City 레벨의 레벨 키 속성을 Country name, State name 및 City name 세트로 정의하면 디폴트 속성을 시 이름으로 정의할 수 있습니다.

관련 속성

레벨 키 속성으로 정의된 레벨 인스턴스에 관한 추가 정보를 제공하는 0개 이상의 속성 컬렉션입니다. 모든 관련 속성은 레벨 키 속성을 통해 기능적으로 판별되어야 합니다. 예를 들어 City 레벨에는 City mayor, City population, City location 및 City description과 같은 관련 속성이 있을 수 있습니다.

OLAP 센터의 레벨 마법사를 사용하면 새 레벨을 독립적으로 작성하거나 계층 구조의 컨텍스트에서 작성할 수 있습니다. 차원에서 큐브 모델 간에 레벨을 공유할 수 있습니다. 레벨을 작성할 때 DB2 Cube Views는 디폴트로 레벨의 속성에 함수 종속성을 작성하므로 디폴트 속성 및 관련 속성은 레벨의 레벨 키 속성과 기능적으로 종속됩니다. 최적화 어드바이저는 함수 종속성을 사용하여 최상의 요약 테이블을 권장하고 DB2 옵티마이저는 함수 종속성을 사용하여 SQL 쿼리를 제대로 라우트합니다.

레벨을 사용하여 계층 구조의 모델화하는 방법으로 여러 가지가 있습니다. 수행하는 모델링 기술이 최적이거나 최적이지 아닌 경우 모두 레벨 키 속성이 디폴트 속성 및 관련 속성을 기능적으로 판별하도록 각 레벨을 정의하는 것이 중요합니다. 레벨의 레벨 키 속성은 해당 레벨의 값을 고유하게 식별해야 합니다. 각 레벨의 속성 사이에 함수 종속성을 작성하도록 정의해야 합니다. 함수 종속성을 통해 최적화 어드바이저는 요약 테이블을 권장하고 DB2 옵티마이저는 SQL 쿼리를 제대로 라우트합니다. 함수 종속성을 사용하면 최적화 어드바이저 마법사가 쿼리 결과를 더 신속하게 리턴할 수 있도록 크기가 더 작은 요약 테이블을 권장할 수 있습니다.

최적의 모델링

최적으로 각 차원의 관계형 데이터는 차원의 각 레벨에 ID 컬럼이 들어 있는 단일 차원 테이블에 저장되고 각 ID 컬럼은 레벨의 값을 고유하게 식별합니다. 예를 들어 다음 컬럼이 들어 있는 영역 차원에 단일 차원 테이블이 있을 수 있습니다.

표 9. 차원 테이블 최적의 모델링

최적의 영역 차원 테이블에 있는 컬럼

City ID (기본 키)

City name

City mayor

State ID

State name

State governor

Country ID

Country name

Country, State 및 City 레벨이 들어 있는 계층 구조를 작성할 수 있습니다. 각 레벨에서 표 10에 표시된 대로 레벨 키 속성 및 모든 관련 속성으로 정의된 ID 컬럼 사이에 함수 종속성을 정의할 수 있습니다.

표 10. 계층 구조 최적의 모델링

레벨	레벨 키 속성	레벨 관련 속성
Country	Country ID	Country name
State	State ID	State name
		State governor
City	City ID	City name
		City mayor

함수 종속성은 다음과 같은 속성 쌍 사이에 작성됩니다.

- Country ID -> Country name
- State ID -> State name, State governor

- City ID -> City name, City mayor

DB2 Cube Views는 City ID가 기본 키이고 해당 키의 제한조건이 이미 정의되어 있어야 하므로 City ID 및 관련 속성에 함수 종속성을 작성하지 않습니다.

일반적인 모델링

계층 구조의 각 레벨에 고유한 ID 데이터 컬럼이 없는 경우 유효한 함수 종속성을 작성하도록 각 레벨의 레벨 키 속성을 정의할 때 주의해야 합니다. 예를 들어 다음 컬럼이 들어 있는 영역 차원에 단일 차원 테이블이 있을 수 있습니다.

표 11. 차원 테이블의 일반적인 모델링

일반적인 영역 차원 테이블의 컬럼

City ID (Primary key)

City name

City mayor

State name

State governor

Country name

최적의 모델링 예와 같이 Country, State 및 City 레벨이 들어 있는 계층 구조를 작성할 수 있습니다. 하지만 레벨의 각 행이 고유하게 정의되도록 레벨 키 속성 정의에 주의를 기울여야 합니다. 예를 들어 레스터(Leicester)라는 도시 이름은 미국 및 영국에 모두 있으므로 도시 이름이 City 레벨을 고유하게 정의하지 않습니다. City 레벨을 고유하게 정의하는 유일한 방법은 표 12에 표시된 대로 Country name, State name 및 City name 속성의 조합을 사용하는 것입니다.

표 12. 계층 구조의 일반적인 모델링

레벨	레벨 키 속성	레벨 관련 속성
Country	Country name	
State	Country name State name	State governor
City	Country name State name City name	City mayor

함수 종속성은 다음과 같은 두 개의 속성 조합 사이에 작성됩니다.

- Country name, State name 및 City name -> City mayor
- Country name 및 State name -> State governor

레벨 오브젝트의 등록 정보는 다음 테이블에 설명되어 있습니다.

표 13. 레벨 등록 정보

등록 정보	설명
레벨 키 속성 세트	함께 레벨을 고유하게 정의하는 하나 이상의 속성 세트입니다.
하나의 디폴트 속성	레벨 키 속성을 통해 기능적으로 판별되며 보고 응용 프로그램에서 유용한 데이터를 표시하는 데 사용할 수 있는 하나의 필수 디폴트 속성입니다.
관련 속성 세트	레벨에 관한 추가 정보가 들어 있으며 레벨 키 속성을 통해 기능적으로 판별되는 0개 이상의 속성 세트입니다.
함수 종속성	레벨에 이에 대응하는 DB2 UDB 함수 종속성이 있는지를 지정하는 BOOLEAN(Yes 또는 No)입니다.
함수 종속성의 이름	함수 종속성이 예로 설정된 경우 이 등록 정보에는 DB2 UDB 함수 종속성의 이름(최대 길이가 18바이트임)이 들어 있습니다. 함수 종속성이 아니므로 설정된 경우 이 등록 정보는 무시됩니다.

치수

치수는 치수 엔티티를 정의하며 사실 오브젝트에서 사용됩니다.

치수는 큐브 모델에 있는 차원 세트의 컨텍스트 내에서 의미를 갖습니다. 예를 들어, 300의 세입은 자체적으로는 의미가 없습니다. 지역 및 시간과 같은 차원의 컨텍스트에 세입 치수를 적용할 때 치수가 의미를 갖게 됩니다. 뉴욕의 1월 세입은 300입니다. 치수의 일반적인 예는 세입, 비용 및 이익입니다.

치수는 두 개의 등록 정보(SQL 표현식 목록 및 집계 목록)를 조합하여 정의합니다. SQL 표현식을 작성하기 위해 테이블 컬럼, 속성 및 치수는 템플릿에 맵핑됩니다. 그런 후 SQL 표현식 결과는 치수의 첫 번째 집계 함수에 대한 입력으로 사용됩니다. 치수에 둘 이상의 집계가 있는 경우, 집계 함수는 나열된 순서대로 수행됩니다. 여기서 각각의 후속 집계는 이전 집계의 결과를 입력으로 사용합니다. 치수에 대한 SQL 표현식이 다른 치수만 참조하는 경우에 집계 함수는 선택적입니다. 참조되는 치수가 집계를 제공하므로 집계 함수는 선택적입니다.

치수의 SQL 표현식은 두 가지 등록 정보(템플릿 및 컬럼, 속성, 치수의 목록)를 조합하여 작성됩니다. 템플릿은 토큰 표기법을 사용합니다. 여기서, {\$n}은 토큰이며 n은 목록의 특정 컬럼 또는 속성 또는 치수를 나타냅니다. 컬럼, 속성 및 치수의 목록은 정렬되어 있으며 목록에서 컬럼, 속성 또는 치수의 위치는 토큰 n 값과 대응됩니다.

SQL 표현식은 첫 번째 집계의 입력으로 사용됩니다. 각각의 집계는 대응되는 차원 목록에 적용되는 함수를 지정합니다. 집계 함수는 기본 데이터베이스에서 지원되는 임의의 집계 함수일 수 있습니다. 다음의 집계 함수가 DB2 Cube Views에서 지원됩니다.

- AVG
- CORRELATION
- COUNT
- COUNT_BIG
- COVARIANCE
- MAX
- MIN
- REGRESSION 함수(모두 9가지 유형)
- STTDEV
- SUM
- VARIANCE

각각의 차원은 치수 오브젝트에 의해 한 번만 집계될 수 있습니다. 치수는 빈 차원 목록을 가진 한 개의 집계를 가져야 하고, 다른 집계들은 각각 명시적 차원 목록을 가져야 합니다. 빈 차원 목록에 대한 집계는 다른 집계에 의해 특별히 사용되지 않는 큐브 모델의 모든 차원에 적용됩니다.

컬럼에 직접 맵핑되는 단순 치수의 예는 세입입니다. 세입 치수는 큐브 모델에 대해 Product, Market 및 Time의 세 차원으로 작성될 수 있습니다. 세입에는 SQL 표현식 템플릿 `template = "${$1}"`이 있습니다. 이 템플릿은 `list = "Column Fact.Rev"`인 컬럼, 속성, 치수의 단일 항목 목록에 지정된 컬럼으로의 단순 맵핑을 표현합니다. 집계 목록은 (SUM, <NULL>)입니다. 여기서 SUM은 집계 함수이며, <NULL>은 차원의 빈 목록입니다. SQL 표현식은 SUM 집계 함수의 입력으로 사용되며, SQL 표현식 결과는 SUM(Fact.Rev)가 됩니다.

보다 복잡한 치수인 이익은 SQL 표현식 템플릿 `templante = "${$1} - {$$2}"`를 가질 수 있습니다. 여기서, 속성, 컬럼 및 치수의 목록은 `list = "Measure Revenue, Column Fact.Cost"`입니다. 토큰을 올바른 참조로 바꾸면 SQL 표현식이 "Revenue - Fact.Cost"가 됩니다. 세입 치수 참조를 자체 컬럼 참조로 확장하면 SQL 표현식은 "Fact.Rev - Fact.Cost"가 됩니다. Profit 치수의 집계 목록은 (SUM, <NULL>)입니다. 이익 SQL 표현식은 SUM 집계 함수의 입력으로 사용되므로, Profit 치수의 SQL은 SUM(Fact.Rev - Fact.Cost)입니다.

치수에 둘 이상의 매개변수를 필요로 하는 CORRELATION과 같은 집계 함수가 있는 경우, 치수는 둘 이상의 SQL 표현식을 갖게 됩니다.

치수는 SQL 데이터 유형을 기반으로 하는 데이터 유형도 갖습니다. DB2 Cube Views는 자동으로 치수의 데이터 유형을 결정합니다. 스키마에 의해 완전히 규정된 경우 각각의 이름은 치수 및 속성에서 고유해야 합니다.

OLAP 센터는 메타데이터 오브젝트 정의의 많은 복잡한 부분을 숨기고 있습니다. OLAP 센터에서는 치수의 SQL 표현식 목록 또는 집계 목록을 명시적으로 정의할 필요가 없습니다. 컬럼, 속성 또는 기타 치수에 직접 맵핑되는 치수를 작성하려면 사실 마법사 또는 사실 등록 정보 창에서 치수를 작성할 때 소스를 선택하십시오. 계산된 치수를 작성하고자 하는 경우, SQL 표현식 빌더 창을 사용하여 소스 표현식을 작성할 수 있습니다. SQL 표현식 빌더는 사용 가능한 컬럼, 속성 및 치수의 목록, 연산자, 함수 및 상수를 제공합니다. 치수 등록 정보 창에서는 치수에 대한 소스 데이터의 데이터 유형 및 소스 데이터가 집계된 이후 치수의 데이터 유형을 볼 수 있습니다.

다음 테이블은 치수를 정의하는 특정 등록 정보를 설명합니다. OLAP 센터는 치수를 작성할 때 이들 각각을 정의합니다.

표 14. 치수 등록 정보

등록 정보	설명
SQL 표현식의 목록(템플릿, [(컬럼, 속성 및 치수의 목록)])	치수의 첫 번째 집계 함수에 대한 입력으로 사용되는 SQL 표현식의 목록입니다. 각각의 SQL 표현식에는 템플릿 및 컬럼, 속성, 치수의 순차 목록이 있습니다.
집계의 목록(함수, 차원의 목록)	치수를 계산하는 방법을 지정하는 집계의 목록입니다. 각각의 집계에는 SQL 집계 함수 및 이 함수를 적용할 차원의 선택적 목록이 있습니다.
데이터 유형(스키마, 이름, 길이, 스케일, 널(NULL) 입력 가능)	치수의 데이터 유형을 결정합니다. 데이터 유형은 SQL 데이터 유형을 기반으로 하며 데이터 유형 스키마, 이름, 길이, 스케일 및 널(NULL) 입력 가능으로 구성됩니다. OLAP 센터는 SYSIBM 이외의 스키마인 경우에만 스키마를 표시합니다.

속성

속성은 데이터베이스 테이블 컬럼의 기본 추상 개념을 표현합니다. 속성에는 SQL 표현식이 들어 있습니다. 이때 SQL 표현식은 테이블 컬럼에 대한 단순 맵핑이거나 더 복잡한 표현식일 수 있습니다. 더 복잡한 표현식에서는 여러 컬럼 또는 속성을 결합할 수 있으며 필요한 경우 사용자 정의 함수(UDF)를 포함하여 모든 SQL 함수를 사용할 수 있습니다.

DB2 Cube Views의 OLAP 센터는 속성 오브젝트 정의의 많은 복잡한 부분을 숨기고 있습니다. OLAP 센터에서는 속성의 표현식 템플릿 또는 매개변수 목록을 명시적으로 정의할 필요가 없습니다. 컬럼에 직접 맵핑되는 속성을 작성하려면 차원 마법사 또는 차원 등록 정보 창에서 속성을 작성할 때 소스 컬럼을 선택하십시오. 계산된 속성을 작성하고자 하는 경우, SQL 표현식 빌더 창을 사용하여 소스 표현식을 작성할 수 있습니다. SQL 표현식 빌더는 사용 가능한 속성, 컬럼, 연산자, 함수 및 상수의 목록을 제공합니다.

OLAP 센터를 사용하지 않고 속성을 작성하고자 하는 경우, 두 등록 정보(컬럼/속성의 목록 및 템플릿 포함)를 조합하여 속성의 SQL 표현식 정의를 작성해야 합니다. 템플

리트는 토큰 표기법을 사용합니다. 여기서 $\{\$\$n\}$ 는 토큰이며 n은 목록의 특정 컬럼 또는 속성을 나타냅니다. 컬럼 및 속성의 목록은 정렬되어 있으며 목록에서 컬럼 또는 속성의 위치는 토큰 n 값과 대응됩니다. 예를 들어, 템플리트 `template = "\{\$\$1\} || ' ' || \{\$\$2\}"`를 `list = "Column CUSTOMER.FIRSTNAME, Attribute LastName"`과 같은 대응되는 목록과 함께 사용함으로써 고객의 이름 및 성을 병합할 수 있습니다. 이 경우에 이름과 성 사이에는 스페이스가 있습니다. 올바른 목록 참조로 템플리트 토큰을 대체하면 SQL 표현식은 `"Customer.FirstName || ' ' || LastName"`입니다. 속성 참조는 컬럼 참조로 추가 확장됨으로써 SQL 표현식 `"Customer.FirstName || ' ' || Customer.LastName"`이 작성됩니다.

기타 속성이 속성을 정의하는 SQL 표현식에 사용되는 경우, 기타 속성은 속성 참조 루프를 형성할 수 없습니다. 예를 들어, 속성 A가 속성 B를 참조하는 경우에 속성 B는 속성 A를 참조할 수 없습니다.

스키마에 의해 완전히 규정된 경우에 각각의 이름은 데이터베이스에 있는 기타 모든 속성 및 치수의 이름으로부터 고유해야 합니다.

다음의 테이블은 속성을 정의하는 특정 등록 정보를 설명합니다. OLAP 센터는 속성 오브젝트를 작성할 때 이들 각각을 정의합니다.

표 15. 속성 등록 정보

등록 정보	설명
SQL 표현식 템플리트	속성을 정의하는 SQL 표현식입니다. 템플리트는 $\{\$\$n\}$ 표기법을 사용하여 컬럼 및 속성을 참조합니다. 여기서, n은 컬럼 및 속성의 목록과 대응되는 서수입니다.
SQL 표현식에 대한 컬럼 및 속성의 목록	속성을 구성하는 모든 컬럼 및 속성의 순차 목록입니다. 이 컬럼 및 속성은 SQL 표현식 템플리트에 지정된 대로 적용이 됩니다.
데이터 유형(스키마, 이름, 길이, 스케일, 널 (NULL) 입력 가능)	속성의 데이터 유형을 판별합니다. 데이터 유형은 SQL 데이터 유형을 기반으로 하며 데이터 유형 스키마, 이름, 길이, 스케일 및 널(NULL) 입력 가능으로 구성됩니다. OLAP 센터는 SYSIBM 이외의 스키마인 경우에만 스키마를 표시합니다.

속성 관계

속성 관계는 일반적인 속성의 관계를 설명하지만, 속성 관계가 큐브 모델의 일부는 아닙니다.

관계는 다음의 등록 정보로 구성되어 있습니다.

- 왼쪽 및 오른쪽 속성
- 유형
- 카디널리티

- 가능한 함수 종속성

유형은 왼쪽 속성과 관련하여 오른쪽 속성의 역할이 무엇인지를 설명합니다. 설명 및 연관의 두 가지 가능한 유형이 있습니다.

설명 오른쪽 속성이 왼쪽 속성의 디스크립터임을 지정합니다. 예를 들어, ProductName 오른쪽 속성은 ProductCode 왼쪽 속성을 설명합니다.

연관 오른쪽 속성은 왼쪽 속성과 연관되어 있지만 왼쪽 속성의 디스크립터는 아님을 지정합니다. 예를 들어, CityPopulation 오른쪽 속성은 CityID와 연관되어 있지만 이의 디스크립터는 아닙니다.

카디널리티는 왼쪽 및 오른쪽 속성의 인스턴스가 관련되는 방식을 설명합니다. 속성 관계에 다음과 같은 카디널리티를 사용할 수 있습니다.

일대일 각각의 오른쪽 속성 인스턴스에 대해 최대 하나의 왼쪽 속성 인스턴스가 있으며 각각의 왼쪽 속성 인스턴스에 대해 최대 하나의 오른쪽 속성 인스턴스가 있습니다.

일대다 각각의 오른쪽 속성 인스턴스에 대해 최대 하나의 왼쪽 속성 인스턴스가 있으며 각각의 왼쪽 속성 인스턴스에 대해 임의의 수의 오른쪽 속성 인스턴스가 있습니다.

다대일 각각의 오른쪽 속성 인스턴스에 대해 임의의 수의 왼쪽 속성 인스턴스가 있으며 각각의 왼쪽 속성 인스턴스에 대해 최대 하나의 오른쪽 속성 인스턴스가 있습니다.

다대다 각각의 오른쪽 속성 인스턴스에 대해 임의의 수의 왼쪽 속성 인스턴스가 있으며 각각의 왼쪽 속성 인스턴스에 대해 임의의 수의 오른쪽 속성 인스턴스가 있습니다.

함수 종속성 등록 정보는 속성 관계가 두 속성 간의 함수 관계를 정의하는지 여부를 알려줍니다. 속성 관계를 함수 종속성으로 지정한다는 것은 모든 왼쪽 속성의 인스턴스가 오른쪽 속성의 인스턴스를 결정한다는 것을 의미합니다. DB2 Cube Views는 함수 종속성 등록 정보를 설정하는 방법에 상관 없이 속성 관계에서 설명하는 속성 사이에서 함수 종속성을 작성하지 않습니다.

CountryID 및 Country, StateID 및 State, CityID 및 City, CityID 및 City_Population 간에 기능 종속성을 나타내는 여러 속성 관계를 정의할 수 있습니다.

OLAP 센터에서 명시적으로 속성 관계 오브젝트를 작성할 수 있습니다. 속성 관계 마법사를 사용하면 명시적으로 속성 관계를 작성할 수 있습니다. 관계형 오브젝트 보기에 서 속성 관계 마법사를 열고 모든 오브젝트 정의 등록 정보를 지정하십시오.

속성 관계 오브젝트를 정의하는 특정 등록 정보는 다음 테이블에 설명되어 있습니다.

표 16. 속성 관계 등록 정보

등록 정보	설명
왼쪽 속성	관계에서 사용되는 왼쪽 속성입니다.
오른쪽 속성	관계에서 사용되는 오른쪽 속성입니다.
유형	속성 관계에 의해 설명되는 관계의 유형입니다. 유형(설명, 연관)은 속성이 사용하는 역할을 판별하는 데 사용됩니다.
카디널리티	조인에서 예상되는 카디널리티(일대일, 일대다, 다대일, 다대다)
함수 종속성	속성 관계가 역시 함수 종속성인지 여부를 판별합니다(예, 아니오). 주: DB2 Cube Views는 이 등록 정보에 지정한 값(Yes 또는 No)과 상관 없이 속성 관계에서의 함수 종속성을 작성하지 않습니다.

조인

조인은 두 개의 관계형 테이블 사이의 조인을 설명하는 메타데이터 오브젝트입니다. 조인은 조인되는 테이블의 컬럼을 참조하는 속성을 참조합니다.

가장 간단한 조인 형식은 두 개의 속성(즉, 첫 번째 테이블의 컬럼에 맵핑되는 속성과 두 번째 테이블의 컬럼에 맵핑되는 속성)을 참조합니다. 컬럼의 비교 방식을 나타내는 연산자도 지정합니다.

조인은 첫 번째 테이블에서 둘 이상의 컬럼을 두 번째 테이블에 있는 동일한 수의 컬럼에 조인하는 복합 조인을 모델링하는 데에도 사용할 수 있습니다. 복합 조인은 속성 쌍을 사용하여 해당 컬럼을 함께 맵핑합니다. 각 속성 쌍에는 컬럼 쌍을 비교하는 방법을 나타내는 연산자가 있습니다.

조인에는 유형과 카디널리티도 있습니다. 조인 유형은 관계형 조인 유형에 맵핑합니다. 조인을 차원에 사용하면 차원 테이블들을 조인할 수 있으며 큐브 모델에 사용하면 큐브 모델의 차원을 사실 오브젝트로 조인할 수 있으며 사실 오브젝트 내에서 사용하면 여러 사실 테이블들을 함께 조인할 수 있습니다. OLAP 센터의 조인 마법사를 사용하여 조인을 작성할 수 있습니다.

조인을 정의하는 특정 등록 정보는 다음 테이블에 설명되어 있습니다.

표 17. 조인 등록 정보

등록 정보	설명
(왼쪽 속성, 오른쪽 속성, 연산자) 목록	왼쪽 속성: 조인의 왼쪽에 있는 속성입니다. 오른쪽 속성: 조인의 오른쪽에 있는 속성입니다. 연산자: 조인에서 예상되는 연산자(=, <, >, <>, >=, <=)입니다.
유형	예상되는 조인 유형(내부, 전체 외부, 왼쪽 외부, 오른쪽 외부)

표 17. 조인 등록 정보 (계속)

등록 정보	설명
카디널리티	조인에서 예상되는 카디널리티(일대일, 일대다, 다대일, 다대다)

큐브

큐브는 단일 SQL문을 사용하여 때때로 전달될 수 있는 OLAP 큐브에 대한 정확한 정의입니다. 큐브 모델에서 전달되는 큐브에는 큐브 모델의 메타데이터 오브젝트를 기초로 메타데이터 오브젝트의 서브세트가 들어 있습니다.

큐브 사실 오브젝트 및 큐브 차원의 목록은 참조되는 큐브 모델의 해당 내용에 대한 서브세트입니다. 큐브 차원은 큐브 차원 당 하나의 큐브 계층 구조만을 허용하기 때문에 큐브는 여러 계층 구조를 사용하지 않는 도구와 응용프로그램에 적합합니다.

큐브는 큐브 모델을 최적화하여 대부분의 활동 및 가장 중요한 큐브 모델의 영역을 지정할 때 사용할 수 있습니다. 가장 자주 쿼리되는 특정 큐브 영역을 정의하는 최적화 슬라이스를 지정할 수 있습니다.

OLAP 센터의 큐브 마법사를 사용하여 큐브를 작성할 수 있습니다. 연관된 큐브를 작성하려면 완전한 큐브 모델을 가지고 있어야 합니다. 큐브의 등록 정보는 다음 테이블에 설명되어 있습니다.

표 18. 큐브 등록 정보

등록 정보	설명
큐브 모델	큐브가 파생된 큐브 모델입니다.
큐브 사실 오브젝트	큐브에서 사용되는 큐브 사실 오브젝트입니다. 큐브 사실 오브젝트는 큐브 모델의 사실 오브젝트에서 파생됩니다.
큐브 차원의 목록	큐브 차원에서 사용되는 큐브 차원의 순차 목록입니다. 각 큐브 차원은 큐브 모델의 해당 차원에서 파생됩니다. 하나의 큐브 계층 구조는 각각의 큐브 차원과 연관됩니다.
최적화 슬라이스의 목록	0개 이상의 최적화 슬라이스 세트입니다. 각 최적화 슬라이스에는 다음이 들어 있습니다. <ul style="list-style-type: none"> • 유형은 드릴 다운, 보고서, MOLAP 추출, 하이브리드 추출 또는 드릴 쓰루일 수 있습니다. • 큐브의 각 큐브 차원마다 하나의 최적화 레벨이 존재하는 최적화 레벨 세트가 들어 있습니다. 최적화 레벨은 다음을 참조합니다. <ul style="list-style-type: none"> - 큐브 차원 및 해당 큐브 계층 구조 - 큐브 레벨, 임의 또는 모두

큐브 사실 오브젝트

큐브 사실 오브젝트에는 순차 목록으로 된 특정 사실 오브젝트의 치수 서브세트가 있습니다. 큐브 사실 오브젝트는 큐브 모델의 사실 오브젝트에 대한 범위를 정할 수 있는 유연성을 큐브에 제공합니다. 조인 및 속성과 같은 구조적 정보는 상위 사실 오브젝트에서 참조됩니다.

OLAP 센터에서 큐브 마법사를 사용하여 큐브 모델의 컨텍스트에서 큐브를 작성합니다. 큐브 사실 오브젝트가 연관된 큐브 모델의 사실 오브젝트에서 파생되는 것을 OLAP 센터가 알고 있으므로 명시적으로 큐브 사실 오브젝트를 정의할 필요는 없습니다. 큐브 모델 사실 오브젝트에서 큐브에서 사용할 치수를 선택합니다.

큐브 사실 오브젝트를 정의하는 특정 등록 정보는 다음 테이블에 설명되어 있습니다.

표 19. 큐브 사실 오브젝트 등록 정보

등록 정보	설명
사실 오브젝트	큐브 사실 오브젝트가 파생된 사실 오브젝트입니다.
치수 목록	큐브에서 사용되는 치수의 순차 목록입니다. 모든 치수는 큐브 사실 오브젝트가 파생된 사실 오브젝트의 일부여야 합니다.

큐브 차원

큐브 차원은 큐브에서 사용할 차원의 범위를 정하기 위해 사용됩니다. 큐브 차원은 자신이 파생된 차원 및 해당 큐브에 대한 관련 큐브 계층 구조를 참조합니다.

하나의 큐브 계층 구조만 큐브 차원에 적용될 수 있습니다. 큐브 차원에 적용되는 조인 및 속성은 차원 정의에서 참조됩니다.

OLAP 센터에서 큐브 마법사를 사용하여 큐브 모델의 컨텍스트에서 큐브를 작성합니다. 큐브에서 보유하고자 하는 큐브 모델 차원을 선택합니다. 큐브 차원으로 포함하는 각각의 차원에 대해, 큐브 계층 구조에 포함할 레벨을 선택할 수 있습니다.

큐브 차원 오브젝트를 정의하는 특정 등록 정보는 다음 테이블에 설명되어 있습니다.

표 20. 큐브 차원 등록 정보

등록 정보	설명
차원	큐브 차원이 파생된 차원입니다.
큐브 계층 구조	큐브 차원에 적용되는 큐브 계층 구조입니다.

큐브 계층 구조

큐브 계층 구조는 계층 구조의 서브셋이며 큐브에서 사용됩니다. 큐브 계층 구조는 자신이 파생된 계층 구조(상위 계층 구조)를 참조하며 상위 계층구조의 상위 레벨인 큐브 레벨 세트를 가질 수 있습니다.

큐브 차원은 하나의 큐브 계층 구조만 가질 수 있습니다. 일반적으로 큐브 계층 구조는 자신이 파생된 계층 구조와 동일한 계층 구조 유형 및 전개 메커니즘을 갖습니다. 계층 구조가 네트워크 유형인 경우, 큐브 계층 구조는 누락된 구성원이 없으면 균형 상태, 누락된 구성원이 있으면 불완전 상태가 됩니다.

OLAP 센터에서 큐브 마법사를 사용하여 큐브 모델의 컨텍스트에서 큐브를 작성합니다. 큐브에서 포함할 큐브 모델 차원을 선택합니다. 큐브 차원으로 포함하는 각각의 차원에 대해, 큐브 계층 구조에 포함할 큐브 레벨을 선택할 수 있습니다.

큐브 계층 구조를 정의하는 특정 등록 정보는 다음 테이블에 설명되어 있습니다.

표 21. 큐브 계층 구조 등록 정보

등록 정보	설명
계층 구조	큐브 계층 구조가 파생된 계층 구조입니다.
정렬된 큐브 레벨 세트	상위 계층 구조에 포함된 레벨의 서브셋인 정렬된 하나 이상의 큐브 레벨 세트입니다. 큐브 레벨의 순서는 상위 계층 구조에서와 동일해야 합니다.

큐브 레벨

큐브 레벨은 레벨의 서브셋이며 큐브에서 사용됩니다. 큐브 레벨은 자신이 파생된 레벨(상위 레벨)을 참조하고 상위 레벨에 정의된 레벨 키 속성 및 디폴트 속성을 상속합니다.

큐브 레벨에는 상위 레벨이 가지는 관련 속성의 서브셋인 속성 세트가 있을 수 있습니다.

큐브 레벨의 등록 정보는 다음 테이블에 설명되어 있습니다.

표 22. 큐브 레벨 등록 정보

등록 정보	설명
속성 세트	상위 레벨이 가지는 관련 속성의 서브셋인 0개 이상의 관련 속성 세트입니다.

메타데이터 오브젝트 규칙

메타데이터 오브젝트에 적용되는 세 가지 유형의 규칙은 기본 규칙, 큐브 모델 완전성 규칙 및 최적화 규칙입니다. 이러한 규칙 세트는 각 오브젝트가 큐브 모델 컨텍스트의 안과 밖에서 유효하게 하고 효과적인 SQL 쿼리를 작성하고 최적화할 수 있게 합니다.

기본 규칙

기본 규칙은 사용하는 컨텍스트 밖에서 메타데이터 오브젝트의 유효성을 정의합니다. 모든 메타데이터 오브젝트는 자체적인 규칙 세트를 갖습니다. 모든 기본 규칙을 준수하는 경우, 메타데이터 오브젝트는 유효합니다.

완전성 규칙

완전성 규칙은 큐브 모델에만 적용되고 기본 규칙을 확장하여 큐브 모델이 다른 메타데이터 오브젝트에 제대로 링크되고 유효한 SQL 쿼리가 작성될 수 있도록 합니다.

최적화 규칙

최적화 규칙은 기본 규칙 및 큐브 모델 완전성 규칙을 보다 더 확장합니다. 이러한 규칙은 메타데이터 오브젝트에 대해 작성된 SQL 쿼리가 제대로 최적화될 수 있도록 합니다.

큐브 모델 규칙

기본 규칙:

- 차원-조인 쌍은 차원 및 조인을 모두 갖고 있어야 합니다.
- 차원 대 사실 조인의 한쪽에 있는 모든 속성은 차원의 속성 목록에 존재해야 하며, 조인의 다른 쪽에 있는 모든 속성은 사실 오브젝트의 속성 목록에 존재해야 합니다.
- 큐브 모델은 큐브 모델의 사실 오브젝트에서의 치수의 집계에 의해 참조되는 모든 명시적 차원을 참조해야 합니다. 큐브 모델에 차원이 있는 경우 차원 목록이 빈 집계는 큐브 모델의 차원 하나 이상과 일치해야 합니다. 차원이 동일한 치수의 기타 집계에서 참조되지 않는지 확인하십시오. 하지만 큐브 모델에 차원이 없는 경우 모든 치수에는 차원 목록이 빈 집계만 있어야 합니다.

완전성 규칙:

- 큐브 모델은 하나의 사실 오브젝트를 참조해야 합니다.
- 큐브 모델은 최소한 하나의 차원을 참조해야 합니다.

최적화 규칙:

- 사실 및 차원을 조인하기 위해 사용되는 조인은 일대일 또는 다대일의 카디널리티를 가져야 하며, 차원의 기본 테이블로 사실 테이블을 조인해야 합니다.
- 큐브 모델의 차원 하나 이상에 계층 구조가 하나 이상 있어야 합니다.

사실 오브젝트 규칙

기본 규칙:

- 사실 오브젝트는 최소한 하나의 치수를 참조해야 합니다.
- 사실 오브젝트에 의해 참조되는 모든 속성 및 치수는 사실 오브젝트 내에서 조인될 수 있어야 합니다. 사실 오브젝트의 조인만 고려됩니다.
- 정확히 하나의 조인만 사실 오브젝트 내의 임의의 두 테이블 간에 정의될 수 있습니다.
- 조인 루프는 사실 오브젝트 내에서 허용되지 않습니다.
- 사실 오브젝트에 의해 참조되는 조인은 사실 오브젝트의 속성을 참조해야 합니다.

차원 규칙

기본 규칙:

- 차원은 최소한 하나의 속성을 참조해야 합니다.
- 차원에 의해 참조되는 모든 속성은 차원 내에서 조인될 수 있어야 합니다. 차원의 조인만 고려됩니다.
- 조인 루프는 차원 내에서 허용되지 않습니다.
- 정확히 하나의 조인만 차원 내의 임의의 두 테이블 간에 정의될 수 있습니다.
- 차원에 의해 참조되는 조인은 차원의 속성을 참조해야 합니다.
- 차원에 의해 참조되는 레벨은 차원의 속성을 참조해야 합니다.
- 차원에 의해 참조되는 계층 구조는 차원의 레벨을 참조해야 합니다.

최적화 규칙:

- 차원은 조인이 일대일 또는 다대일 카디널리티로 접속하는 하나의 기본 테이블을 갖고 있어야 합니다.

레벨 규칙

기본 규칙:

- 레벨은 최소한 하나의 레벨 키 속성을 참조해야 합니다.
- 레벨은 최소한 하나의 디폴트 속성을 참조해야 합니다.

계층 구조 규칙

기본 규칙:

- 계층 구조는 최소한 하나의 레벨을 참조해야 합니다.
- 정확히 두 개의 레벨이 반복 전개에 존재해야 합니다.
- 표준 전개는 모든 유형의 계층 구조에 대해 사용이 가능하며 반복 전개는 불균형 계층 구조에 대해서만 사용이 가능합니다.

치수 규칙

기본 규칙:

- 각각의 SQL 표현식 템플리트는 영(0)개 이상의 속성, 컬럼 및 치수 매개변수를 가질 수 있습니다.
- SQL 표현식 템플리트에 대한 매개변수로 사용되는 속성 및 치수는 종속성 루프를 형성할 수 없습니다.
- 치수의 SQL 템플리트는 빈 문자열일 수 없습니다.
- SQL 템플리트는 집계 함수를 사용할 수 없습니다.
- 최소한 하나의 치수가 참조되며 치수들만 참조되는 경우, 집계의 정의는 선택적입니다.
- 집계 존재하는 경우, SQL 템플리트의 수는 첫 번째 집계 함수의 매개변수 수와 일치해야 합니다.
- 다중 SQL 템플리트의 치수는 집계 스크립트에서 최소한 하나의 단계를 정의해야 합니다.
- 치수가 다중 SQL 템플리트를 정의하는 두 번째 치수를 참조하는 경우, 참조하는 치수는 집계 스크립트를 가질 수 없습니다.
- 다중 매개변수 집계 함수는 첫 번째 집계에서만 사용이 가능합니다.
- 치수가 하나 이상의 집계를 정의하는 경우, 하나의 집계는 차원의 빈 목록을 지정해야 합니다.
- 치수는 집계 내에서 또는 집계들 간에 각 차원을 한 번만 참조할 수 있습니다.
- SQL 템플리트에서 토큰 표시기는 번호 1로 시작하고 연속적으로 번호를 계속 부여해야 합니다.
- SQL 표현식 내에서 모든 컬럼, 속성 및 치수가 최소한 한 번은 참조되어야 합니다.

속성 규칙

기본 규칙:

- 각각의 SQL 템플리트는 영(0)개 이상의 다음 매개변수 속성 및 컬럼을 갖습니다.
- SQL 표현식 템플리트에 대한 매개변수로 사용되는 속성은 종속성 루프를 형성할 수 없습니다.
- 속성의 SQL 템플리트는 빈 문자열일 수 없습니다.
- SQL 템플리트는 집계 함수를 가질 수 없습니다.
- SQL 템플리트에서 토큰 표시기는 번호 1로 시작하고 연속적으로 번호를 계속 부여해야 합니다.
- SQL 표현식에서 모든 컬럼 및 속성은 최소한 한 번 이상 참조되어야 합니다.

속성 관계 규칙

기본 규칙:

- 속성 관계는 두 속성을 참조해야 합니다.
- 속성 관계는 다대다 카디널리티의 함수 종속성으로 정의될 수 없습니다.

조인 규칙

기본 규칙:

- 조인은 최소한 하나의 3요소(왼쪽 속성, 오른쪽 속성, 연산자)를 참조해야 합니다.
- 각각의 조인 3요소에 대해 유효한 조작이 정의되어야 합니다. 왼쪽 및 오른쪽 속성의 데이터 유형은 상호 간 및 지정된 조작과 호환 가능해야 합니다.
- 모든 왼쪽 속성은 단일 테이블의 하나 이상의 컬럼으로 해석되어야 합니다.
- 모든 오른쪽 속성은 단일 테이블의 하나 이상의 컬럼으로 해석되어야 합니다.

최적화 규칙:

- 조인에 참여하는 컬럼에 대해 제한조건을 정의해야 합니다. 동일한 컬럼 세트가 등호의 양쪽에 사용되는 경우, 컬럼 세트와 일치하는 기본 키를 정의해야 합니다. 등호의 양쪽에서 상이한 컬럼 세트가 사용되는 경우, 한쪽에는 일치하는 기본 키가 있고 다른 쪽에는 일치하는 외부 키가 있어야 하며 기본 키를 참조해야 합니다.
- 조인 카디널리티는 일대일, 다대일 또는 일대다가 되어야 합니다. 등호의 양쪽에 동일한 컬럼 세트가 있는 조인에서 카디널리티는 일대일이어야 합니다. 기타 모든 조인의 경우, 카디널리티는 기본 키가 정의된 쪽에서는 일(1)이어야 하며 외부 키가 정의된 쪽에서는 다(N)이어야 합니다. 외부 키 쪽에도 기본 키가 정의되어 있으면 해당 쪽의 카디널리티에 일(1)을 사용해야 합니다.
- 조인에서 사용되는 모든 속성은 널(null)이 아닌 SQL 표현식으로 해석되어야 합니다.
- 조인 유형은 내부 조인이어야 합니다.

큐브 규칙

기본 규칙:

- 큐브는 하나의 큐브 모델을 참조해야 합니다.
- 큐브는 하나의 큐브 사실 오브젝트를 참조해야 합니다.
- 큐브는 최소한 하나의 큐브 차원을 참조해야 합니다.
- 큐브 사실 오브젝트는 참조된 큐브 모델에서 사용되는 사실 오브젝트에서 파생된 것이어야 합니다.
- 모든 큐브 차원은 참조된 큐브 모델에서 사용되는 차원에서 파생된 것이어야 합니다.

- 큐브에 0개 이상의 최적화 슬라이스가 있어야 합니다.
- 최적화 슬라이스에 큐브의 각 큐브 차원마다 하나의 최적화 계층이 있어야 합니다.
- 최적화 슬라이스에 하나 이상의 최적화 레벨이 있어야 합니다.
- 최적화 레벨에 하나의 큐브 차원 참조 및 하나의 계층 구조 참조가 있어야 합니다.
- 최적화 레벨에 모두, 임의 또는 큐브 레벨 참조가 있어야 합니다.
- 최적화 레벨 오브젝트에서 참조된 큐브 차원 및 큐브 계층 구조는 큐브 오브젝트의 오브젝트에서 파생되어야 합니다. 큐브 계층 구조가 큐브 차원에 속해야 합니다.
- 최적화 레벨 오브젝트에서 큐브 레벨 참조가 모두 또는 임의가 아닌 경우 큐브 레벨이 큐브 계층 구조에 속해야 합니다.
- 최적화 레벨의 큐브 차원 또는 큐브 계층 구조 참조는 최적화 슬라이스에서 반복되지 않아야 합니다.
- 하이브리드 추출 유형의 최적화 슬라이스 오브젝트가 있는 경우 MOLAP 추출 유형의 다른 최적화 슬라이스가 있을 수 없고 그 반대의 경우도 마찬가지입니다.
- 각 큐브마다 MOLAP 추출 유형의 최적화 슬라이스가 최대로 하나 있습니다.
- 각 큐브마다 하이브리드 추출 유형의 최적화 슬라이스가 최대로 하나 있습니다.
- 하이브리드 추출 유형의 최적화 슬라이스 오브젝트가 있는 경우 드릴 쓰루 유형의 다른 최적화 슬라이스가 있을 수 있습니다.

큐브 사실 오브젝트 규칙

기본 규칙:

- 큐브 사실 오브젝트는 하나의 사실 오브젝트를 참조해야 합니다.
- 큐브 사실 오브젝트는 최소한 하나의 치수를 참조해야 합니다.
- 큐브 사실 오브젝트에 의해 참조되는 모든 치수는 이에 대응되는 사실 오브젝트에서도 참조되어야 합니다.

큐브 차원 규칙

기본 규칙:

- 큐브 차원은 하나의 차원을 참조해야 합니다.
- 큐브 차원은 하나의 큐브 계층 구조를 참조해야 합니다.
- 참조된 큐브 계층 구조는 참조된 차원에서 사용되는 계층 구조로부터 파생된 것이어야 합니다.

큐브 레벨 규칙

기본 규칙:

- 큐브 레벨은 하나의 레벨을 참조해야 합니다.
- 큐브 레벨은 최소한 하나의 관련 속성을 참조해야 합니다.

- 모든 참조된 속성은 레벨의 관련 속성이어야 합니다.

큐브 계층 구조 규칙

기본 규칙:

- 큐브 계층 구조는 하나의 계층 구조를 참조해야 합니다.
- 큐브 계층 구조는 최소한 하나의 큐브 레벨을 참조해야 합니다.
- 큐브 계층 구조에서 참조하는 큐브 레벨은 대응하는 계층 구조의 일부인 레벨에서 파생되어야 합니다.
- 큐브 계층 구조의 큐브 레벨은 참조된 계층 구조의 해당되는 레벨에서와 동일한 순서대로 나열되어야 합니다. 큐브 레벨 순서는 네트워크 계층 구조에 적용되지 않습니다.

제 3 장 DB2 Cube Views 메타데이터 오브젝트 모델 설계

이 섹션에서는 다음 주제를 설명합니다.

OLAP 센터 시작

OLAP 센터를 시작하면 메타데이터 오브젝트를 관리할 수 있습니다.

OLAP 센터 및 API 호환성

DB2 Cube Views 버전 8.2는 OLAP 센터 및 메타데이터 API에 대해 제한된 역호환성 지원 기능을 갖고 있습니다.

DB2 Cube Views의 권한 및 특권

OLAP 센터를 사용하여 태스크를 수행하려면, 운영 체제의 필수 권한 및 특권 확보에 도움을 받기 위해 DB2 데이터베이스 관리자(DBA)에게 문의해야 합니다.

DB2 Cube Views 메타데이터 오브젝트 작성

OLAP 센터를 사용하여 DB2 Cube Views 메타데이터 오브젝트를 작성할 수 있습니다.

큐브 모델에서 차원 제거

차원이 더 이상 필요하지 않은 경우 큐브 모델에서 해당 자원을 제거할 수 있습니다. 차원이 다른 큐브 모델에 사용되는 경우 차원을 삭제하지 않아도 차원을 제거할 수 있습니다.

데이터베이스에서 메타데이터 오브젝트 삭제

이 데이터베이스에 있는 큐브 모델의 메타데이터 오브젝트를 더 이상 사용하지 않는 경우 메타데이터 오브젝트를 삭제할 수 있습니다.

OLAP 센터 시작 및 새로 고침

OLAP 센터를 시작하면 메타데이터 오브젝트를 관리할 수 있습니다.

OLAP 센터를 시작하려면 다음을 수행하십시오.

1. 시작 -> 프로그램 -> **IBM DB2** -> 비즈니스 인텔리전스 도구 -> **OLAP** 센터를 누르십시오. 데이터베이스 연결 창이 열립니다.
2. 데이터베이스 연결 창에서 메타데이터를 관리하려는 데이터베이스에 로그인하십시오.

OLAP 센터가 데이터베이스의 메타데이터 오브젝트 시간에 스냅샷을 표시합니다. DB2 Cube Views가 관리하는 메타데이터 오브젝트의 무결성을 항상 확인하지만 다른 OLAP 센터 사용자 또는 API 응용프로그램이 데이터베이스에 메타데이터 오브젝트를 작성하면 OLAP 센터 창의 콘텐츠는 자동으로 갱신되지 않습니다. 다른 사용자 또는 API 응

용프로그램이 메타데이터 오브젝트를 변경한 경우 보기 -> 새로 고침을 선택하여 새 데이터베이스 상태를 확인할 수 있습니다. 동시에 동일한 메타데이터 오브젝트에서 작업하는 여러 사용자가 데이터베이스의 최신 데이터를 보지 못할 수 있으므로 오류가 발생할 수 있습니다. 여러 사용자가 동일한 메타데이터 오브젝트 세트에서 동시에 작업하지 말아야 합니다.

OLAP 센터 및 API 버전 호환성

DB2 Cube Views 버전 8.2는 OLAP 센터 및 메타데이터 API에 대해 제한된 역호환성 지원 기능을 갖고 있습니다.

DB2 Cube Views API는 다음을 지원합니다.

- 버전 8.2 API는 버전 8.1 클라이언트의 설명 요청을 지원합니다. 설명은 이전 릴리스에서 지원되는 유일한 버전 8.1 조작입니다.

DB2 Cube Views OLAP 센터는 다음을 지원합니다.

- 버전 8.2 OLAP 센터는 버전 8.1 API 또는 버전 8.1 메타데이터 테이블을 지원하지 않습니다.
- 버전 8.1 OLAP 센터는 버전 8.2 API 또는 버전 8.2 메타데이터 테이블을 지원하지 않습니다.
- 버전 8.1 OLAP 센터는 버전 8.2 DB2 데이터베이스에 연결될 수 없습니다.
- OLAP 센터는 버전 8.1 XML의 임포트를 지원합니다. 버전 8.1 XML 파일을 임포트하면 OLAP 센터는 메타데이터를 임포트하기 전에 변환 조작을 사용하여 XML을 버전 8.2로 이주시킵니다.
- OLAP 센터는 버전 8.2 및 버전 8.1 XML 모두를 익스포트할 수 있습니다.

DB2 Cube Views db2mdapiclient는 다음을 지원합니다.

- 버전 8.1 db2mdapiclient는 버전 8.2 XML 사용 시 버전 8.2 API를 지원합니다.
- 버전 8.2 db2mdapiclient는 버전 8.1 XML 사용 시 버전 8.1 API를 지원합니다.

DB2 Cube Views 사용 권한 및 특권

OLAP 센터를 사용하여 태스크를 수행하려면, 운영 체제의 필수 권한 및 특권 확보에 도움을 받기 위해 DB2 데이터베이스 관리자(DBA)에게 문의해야 합니다.

달리 지정하지 않는 한, SELECT, INSERT, UPDATE 및 DELETE 특권에 대한 다음 테이블의 모든 참조는 DB2 Cube Views가 설치된 데이터베이스의 DB2INFO 스키마의 테이블에 적용됩니다.

Windows 권한 및 특권

Windows 운영 체제의 경우, OLAP 센터 태스크에 대한 권한은 데이터베이스에 연결하기 위해 사용되는 사용자 ID에 적용됩니다.

표 23. Windows 일반 권한 및 특권

태스크	권한 및 특권
모든 OLAP 센터 태스크	<ul style="list-style-type: none"> DB2 Cube Views가 설치된 데이터베이스의 경우 SYSADM 또는 DBADM DB2 Cube Views가 설치된 데이터베이스 내의 스토어드 프로시저 DB2INFO.MD_MESSAGE의 경우 EXECUTE(DB2 UDB V8 전용) 목표 데이터베이스의 경우 CONNECT
메타데이터 오브젝트 작성	SELECT, INSERT
메타데이터 오브젝트 수정	SELECT, INSERT, UPDATE, DELETE
메타데이터 오브젝트 삭제	SELECT, DELETE
XML 파일로 메타데이터 오브젝트 익스포트	SELECT
XML 파일에서 DB2 UDB로 메타데이터 오브젝트 임포트	SELECT, INSERT, UPDATE
최적화 어드바이저 실행	시스템 테이블 및 기본 테이블의 경우 SELECT
최적화 어드바이저에 의해 생성된 DB2 명령 실행	<ul style="list-style-type: none"> 스키마 DB2INFO에서 CREATEIN, DROPIN 기본 테이블에서 SELECT 및 ALTER(또는 CONTROL)

일부 설정 및 설치 태스크는 OLAP 센터를 사용하여 수행될 수 있습니다.

표 24. Windows 설정 및 설치 권한과 특권

태스크	권한 및 특권
모든 설정 및 설치 태스크	<ul style="list-style-type: none"> DB2 Cube Views가 설치되어 있는 데이터베이스의 경우 SYSADM(메타데이터 오브젝트에 대한 스키마 작성용) DB2 Cube Views가 설치될 데이터베이스의 경우 CONNECT, CREATETAB, IMPLICIT_SCHEMA DB2 Cube Views가 설치되어 있는 데이터베이스 내의 스키마 DB2INFO의 경우 CREATEIN, DROPIN DB2 Cube Views가 설치되어 있는 데이터베이스의 DB2INFO 스키마에 있는 모든 테이블의 경우 SELECT, INSERT, UPDATE, DELETE, CONTROL

AIX 권한 및 특권

AIX에서는 두 가지 상이한 유형의 사용자 ID가 있으며, 각각 자신의 고유한 권한 세트를 갖습니다. 한 가지 유형의 사용자 ID를 설정하여 데이터베이스에 연결하고 DB2INFO.MD_MESSAGE 스토어드 프로시저를 실행해야 합니다. 이 사용자 ID 유형은 아래에서 실행 사용자 ID라고 합니다.

다른 유형의 사용자 ID를 설정하여 기타 모든 OLAP 센터 태스크를 실행해야 합니다. 이 사용자 ID 유형은 아래에서 실행 사용자 ID라고 합니다. 이 사용자 ID는 FENCED 스토어드 프로시저를 실행하기 위해 AIX에서 설정되는 특수 사용자 ID입니다. FENCED 스토어드 프로시저는 이 사용자 ID에서 실행되며, 이 사용자 ID는 sqllib/adm에 있는 .fenced 파일의 소유자로 설정됩니다. 이 사용자 ID는 스토어드 프로시저가 사용할 수 있는 시스템 자원을 제어합니다.

표 25. 연결 사용자 ID에 대한 AIX 일반 권한 및 특권

태스크	권한 및 특권
모든 OLAP 센터 태스크	<ul style="list-style-type: none"> DB2 Cube Views가 설치된 데이터베이스의 경우 SYSADM 또는 DBADM DB2 Cube Views가 설치된 데이터베이스 내의 스토어드 프로시저 DB2INFO.MD_MESSAGE의 경우 EXECUTE(DB2 UDB V8 전용) 목표 데이터베이스의 경우 CONNECT

표 26. 실행 사용자 ID에 대한 AIX 권한 및 특권

태스크	권한 및 특권
메타데이터 오브젝트 작성	SELECT, INSERT
메타데이터 오브젝트 수정	SELECT, INSERT, UPDATE, DELETE
메타데이터 오브젝트 삭제	SELECT, DELETE
XML 파일로 메타데이터 오브젝트 익스포트	SELECT
XML 파일에서 DB2 UDB로 메타데이터 오브젝트 임포트	SELECT, INSERT, UPDATE, DELETE
최적화 어드바이저 실행	시스템 테이블 및 기본 테이블의 경우 SELECT
최적화 어드바이저에 의해 생성된 DB2 명령 실행	<ul style="list-style-type: none"> 스키마 DB2INFO에서 CREATEIN, DROPIN 기본 테이블에서 SELECT 및 ALTER(또는 CONTROL)

일부 설정 및 설치 태스크는 OLAP 센터를 사용하여 수행될 수 있습니다. 이 태스크에서는 연결 사용자 ID에 대한 추가 권한이 필요합니다.

표 27. 연결 사용자 ID에 대한 AIX 설정 및 설치 권한과 특권

태스크	권한 및 특권
모든 설정 및 설치 태스크	<ul style="list-style-type: none"> DB2 Cube Views가 설치되어 있는 데이터베이스의 경우 SYSADM(메타데이터 오브젝트에 대한 스키마 작성용) DB2 Cube Views가 설치될 데이터베이스의 경우 CONNECT, CREATETAB, IMPLICIT_SCHEMA DB2 Cube Views가 설치되어 있는 데이터베이스 내의 스키마 DB2INFO의 경우 CREATEIN, DROPIN DB2 Cube Views가 설치되어 있는 데이터베이스의 DB2INFO 스키마에 있는 모든 테이블의 경우 SELECT, INSERT, UPDATE, DELETE, CONTRO

DB2 Cube Views 메타데이터 오브젝트 작성

OLAP 센터를 사용하여 DB2 Cube Views 메타데이터 오브젝트를 작성할 수 있습니다.

DB2 Cube Views와 OLAP 도구 간에 메타데이터 교환

OLAP 센터를 사용하여 DB2 UDB 데이터베이스에 연결하고 메타데이터 오브젝트를 임포트하거나 익스포트하십시오.

DB2 데이터베이스에 연결

DB2 데이터베이스에 연결하거나 연결을 끊을 수 있습니다.

OLAP 센터를 사용하여 DB2 데이터베이스에 연결하기 전에 DB2 Cube Views 메타데이터와 함께 사용할 데이터베이스를 설정해야 합니다. 연결 시도 중인 데이터베이스가 DB2 Cube Views 메타데이터와 함께 사용할 수 있도록 설정되지 않은 경우 연결을 시도할 때 메시지가 나타납니다. 연결하는 데 사용하는 사용자 ID에게 적절한 권한 및 특권이 있는 경우 OLAP 센터는 설정 태스크를 수행할 수 있습니다.

참고: 데이터베이스에 이미 연결되었으며 다른 데이터베이스에 연결하는 경우 원래의 데이터베이스에서 연결이 끊어집니다. 원래의 데이터베이스에 있는 모든 메타데이터 오브젝트가 OLAP 센터 오브젝트 트리에 더 이상 표시되지 않습니다.

DB2 데이터베이스에 연결하려면 다음을 수행하십시오.

1. **OLAP** 센터 -> 연결을 눌러 데이터베이스 연결 창을 열고 필요한 정보를 지정하십시오.

- 데이터베이스 이름 필드에서 연결하려는 데이터베이스를 선택하십시오.
 - 사용자 이름 필드에 지정한 데이터베이스의 사용자 ID를 입력하십시오.
 - 암호 필드에 지정한 사용자 ID의 암호를 입력하십시오.
2. 확인을 누르십시오 연결한 DB2 데이터베이스의 메타데이터 오브젝트가 OLAP 센터 오브젝트 트리에 표시됩니다.

DB2 데이터베이스에서 연결을 끊으려면 다음을 수행하십시오.

OLAP 센터 기본 창에서 **OLAP 센터** -> 연결 끊기를 누르십시오. 모든 메타데이터 오브젝트가 OLAP 센터 오브젝트 트리에서 제거됩니다.

메타데이터 임포트

DB2 Cube Views를 사용하여 메타데이터 오브젝트를 관리할 수 있도록 메타데이터 오브젝트를 DB2 UDB에 임포트할 수 있습니다.

DB2 UDB로 메타데이터 오브젝트를 임포트하기 전에 DB2 Cube Views XML 파일이 있는지 확인하십시오. 벤더 응용프로그램으로 메타데이터 브릿지를 사용하거나, 기존 DB2 UDB 데이터베이스에서 메타데이터를 익스포트하거나, XML 편집기를 사용하여 DB2 Cube Views XML 파일을 작성할 수 있습니다.

버전 8.1 및 버전 8.2 DB2 Cube Views XML 파일을 임포트할 수 있습니다. 임포트 마법사는 오브젝트를 작성할 때 버전 8.1 메타데이터 오브젝트를 버전 8.2 메타데이터 오브젝트로 변환시킬 수 있습니다.

메타데이터를 임포트하려면 다음을 수행하십시오.

1. **OLAP 센터** -> 임포트를 눌러 임포트 마법사를 여십시오. 임포트 마법사가 열립니다.
2. 소스 페이지에서 임포트하려는 메타데이터 정보가 있는 XML 파일을 지정하십시오.
3. 임포트 옵션 페이지에서 임포트될 메타데이터 오브젝트의 목록을 검토하고 임포트 옵션을 지정하십시오. 오브젝트가 현재 존재하는지 또는 오브젝트를 임포트한 후에 작성할 것인지를 확인할 수 있습니다.
4. 요약 페이지에서 메타데이터 임포트 시 결정한 사항을 검토하십시오.
5. 완료 버튼을 눌러 메타데이터 오브젝트를 임포트하십시오. 임포트한 오브젝트가 OLAP 센터 오브젝트 트리에 표시됩니다.

메타데이터 익스포트

비즈니스 상대 응용프로그램에서 메타데이터 오브젝트를 사용할 수 있도록 DB2 Cube Views 메타데이터 오브젝트를 익스포트할 수 있습니다.

DB2 Cube Views XML 파일로 메타데이터 오브젝트를 익스포트하기 전에 익스포트할 메타데이터 오브젝트를 작성해야 합니다.

메타데이터를 익스포트하려면 다음을 수행하십시오.

1. **OLAP 센터** -> 익스포트를 눌러 익스포트 창을 여십시오.
2. 익스포트할 메타데이터의 버전을 선택하십시오. 벤더 도구에서 브릿지를 사용하는 경우 벤더 도구가 지원하는 메타데이터의 버전을 점검하십시오.
3. 모든 메타데이터 오브젝트를 선택하여 데이터베이스의 모든 메타데이터 오브젝트를 익스포트하거나, 선택한 메타데이터 오브젝트를 선택하여 익스포트할 특정 메타데이터 오브젝트 세트를 지정하십시오. 한 개의 큐브 모델이나 같은 상위 큐브 모델을 가진 하나 이상의 큐브를 선택할 수 있습니다. 하나 이상의 큐브를 선택하면 상위 큐브 모델이 자동으로 익스포트됩니다.
4. 익스포트할 XML 파일의 이름을 지정하십시오. 파일이 이미 존재하는 경우 이를 겹쳐줍니다.
5. 확인을 누르십시오. 지정한 메타데이터 오브젝트에 대한 정보가 포함된 DB2 Cube Views XML 파일이 작성됩니다. DB2 Cube Views XML 파일을 비즈니스 상대 응용프로그램에서 사용할 수 있습니다.

빠른 시작 마법사로 큐브 모델 작성

관계형 스키마를 기반으로 큐브 모델 및 대응되는 사실 오브젝트, 치수, 차원, 속성 및 조인을 모두 한 번에 작성할 수 있습니다.

큐브 모델 및 이에 대응되는 메타데이터를 작성하기 전에, 데이터베이스의 테이블에 대한 참조 무결성 제한조건을 정의해야 합니다.

빠른 시작 마법사는 스키마에서 논리적으로 추론할 수 있는 메타데이터 오브젝트를 작성합니다. 사실 테이블 및 치수 컬럼을 지정하면, 마법사는 대응되는 차원, 조인 및 속성을 감지하게 됩니다. 빠른 시작 마법사를 완료한 이후에는 필요에 따라 메타데이터 오브젝트를 추가, 삭제하고 수정할 수 있습니다.

큐브 모델 및 해당 메타데이터를 작성하려면 다음을 수행하십시오.

1. OLAP 센터 오브젝트 트리에서 큐브 모델 폴더 또는 데이터베이스를 마우스 오른쪽 단추로 누르고 **큐브 모델 작성 - 빠른 시작**을 눌러 빠른 시작 마법사를 여십시오. 빠른 시작 마법사가 열립니다. 오브젝트 트리에 큐브 모델 폴더가 없으면, 보기 > **OLAP 오브젝트** 표시를 눌러 OLAP 오브젝트 보기 안에 있는지 확인하십시오.

소개 페이지는 빠른 시작 마법사가 수행할 수 있는 작업을 설명합니다.

2. 사실 테이블 페이지에서 큐브 모델에 대한 스키마 및 사실 테이블을 선택하십시오. 선택하는 사실 테이블은 큐브 모델의 사실 오브젝트로 직접 맵핑됩니다. 지정된 사실 테이블에서 외부 키로 사용되는 컬럼은 큐브 모델의 속성 오브젝트로 맵핑됩니다. 마법사는 내재된 차원(컬럼이 사실 테이블에 존재하는 차원)을 발견할 수 없습니다. 빠른 시작 마법사를 완료한 후 내재된 차원을 큐브 모델에 추가할 수 있습니다.

3. 치수 페이지에서 치수에 직접 맵핑하고자 하는 지정된 사실 테이블에서 컬럼을 선택하십시오. 기본 키 및 외부 키는 치수로 맵핑될 수 없으며, 나열되지 않습니다. 디폴트 집계 함수는 숫자 컬럼을 기반으로 하는 치수에 대해서는 SUM이며, 문자 컬럼을 기반으로 하는 치수에 대해서는 COUNT입니다. 이 마법사에서는 계산된 치수를 작성할 수 없으며, 빠른 시작 마법사를 완료한 후 계산된 치수를 작성할 수 있습니다.
4. 요약 페이지에서 작성될 메타데이터 오브젝트를 보십시오. **완료**를 눌러서 큐브 모델 및 해당 메타데이터 오브젝트를 작성하십시오. **완료**를 누르면, 빠른 시작 마법사가 DB2 카탈로그에 다음 메타데이터 오브젝트를 작성합니다.
 - 기타 모든 메타데이터 오브젝트를 포함하는 큐브 모델
 - 지정된 사실 테이블과 대응되는 사실 오브젝트
 - 지정된 사실 테이블 컬럼과 대응되는 치수
 - 사실 테이블에 조인된 각 차원 테이블에 대응되는 차원. 아우트리거 테이블은 차원 테이블에 조인된 테이블이며 적절한 차원 테이블에 포함되어 있습니다.
 - 차원 및 아우트리거 테이블의 각 컬럼에 대응되며 사실 테이블의 임의의 외부 키에 대응되는 속성
 - 차원 테이블 및 임의의 대응되는 아우트리거 테이블을 조인하는 차원 오브젝트 내의 조인 및 사실-차원 조인의 역할을 하는 조인 오브젝트

큐브 모델에 계층 구조, 계산된 치수 및 큐브를 추가할 수 있습니다.

완전한 큐브 모델 작성

완전한 큐브 모델을 작성하기 위해 빈 큐브 모델을 작성한 후 큐브 모델의 각 차원에 사실 오브젝트, 차원, 계층 구조 및 레벨을 추가해야 합니다. 큐브 모델은 관계형 데이터에 발행된 쿼리를 최적화할 수 있도록 스타 스키마 또는 snowflake 스키마의 관계형 데이터 간 관계를 정의합니다.

빈 큐브 모델을 작성하거나 아래의 단계에 따라 큐브 모델을 완성하는 데 필요한 모든 오브젝트를 작성할 수 있습니다. 큐브 모델을 완성한 후에 큐브 모델 오브젝트의 모든 등록 정보 또는 등록 정보 서브세트를 포함할 수 있는 큐브 오브젝트를 작성할 수 있습니다.

DB2 Cube Views 큐브 모델은 논리적 스타 스키마 또는 snowflake 스키마를 표현한 것입니다. 큐브 모델은 중심의 사실 오브젝트 주변에 관련 차원 오브젝트를 묶은 것입니다. 각 차원에는 큐브 모델의 유연성을 높이는 여러 계층 구조가 있을 수 있습니다. 사실 및 차원 오브젝트에 의해 사용되는 테이블을 조인하는 방법에 대한 구조적 정보가 큐브 모델에 저장됩니다. 또한 큐브 모델은 OLAP 데이터를 검색할 수 있는 충분한 정보를 저장합니다. 큐브 모델을 이해하고 특정 차원의 여러 뷰를 지원할 수 있는 기타 보고 및 OLAP 도구를 사용하면 차원에 정의된 다중 계층 구조에서 도움이 될 수 있습니다.

완전한 큐브 모델을 작성하려면 다음을 수행하십시오.

1. 큐브 모델을 작성하십시오.
2. 사실 오브젝트를 작성하십시오.
3. 차원을 작성하십시오.
4. 차원에 대한 계층 구조를 작성하십시오.
5. 각 계층 구조에 레벨 작성.

큐브 모델 작성

큐브 모델을 작성하여 중심의 사실 오브젝트 주변에 관련 차원 오브젝트를 묶습니다.

사실 및 차원 오브젝트에 의해 사용되는 테이블을 조인하는 방법에 대한 구조적 정보가 큐브 모델에 저장됩니다. 큐브 모델에서 오브젝트의 서브셋을 사용하여 큐브를 작성하면 관련 사실 및 차원을 응용프로그램에 선택적으로 노출시킬 수 있습니다.

큐브 모델 마법사는 빈 큐브 모델을 작성하는 단계를 안내합니다. 마법사를 사용하여 큐브 모델을 작성한 후에 사실 오브젝트와 차원을 해당 조인 오브젝트와 함께 추가하십시오. 차원은 상이한 큐브 모델들 간에 공유가 가능합니다.

큐브 모델을 작성하려면 다음을 수행하십시오.

1. OLAP 센터 오브젝트 트리에서 큐브 모델 폴더 또는 데이터베이스를 마우스 오른쪽 단추로 누르고 큐브 모델 작성을 눌러 큐브 모델 마법사를 여십시오. 오브젝트 트리에 큐브 모델 폴더가 없으면, 보기 > **OLAP** 오브젝트 표시를 눌러 OLAP 오브젝트 보기 안에 있는지 확인하십시오.
2. 이름 페이지에서 큐브 모델에 대한 이름, 비즈니스 이름, 스키마 및 주석을 지정하십시오.
3. 완료를 눌러 큐브 모델을 작성하십시오.

사실 오브젝트 작성

기존 큐브 모델의 일부분으로 사실 오브젝트를 작성합니다. 사실 오브젝트는 지정된 응용프로그램에 관련된 치수를 그룹화합니다. 사실 오브젝트는 큐브 모델에서 스타 스키마의 중심으로 사용됩니다. 간단한 치수 및 계산된 치수를 작성할 수 있습니다.

사실 오브젝트를 작성하려면 먼저 큐브 모델을 작성하십시오.

사실 오브젝트를 작성하려면 다음을 수행하십시오.

1. 큐브 모델을 마우스 오른쪽 단추로 누른 후 사실 작성을 눌러 사실 마법사를 여십시오. 사실 작성 옵션을 사용할 수 없는 경우 선택된 큐브 모델에 정의된 사실 오브젝트가 이미 정의되어 있는 것입니다.

- 이름 페이지에서 사실 오브젝트에 대한 이름, 비즈니스 이름 및 스키마를 지정하십시오. 사실 오브젝트에 대한 주석도 입력할 수 있습니다. 사실 마법사에서 작성한 모든 계산 및 미계산 치수는 사실 오브젝트에 지정한 동일한 스키마 이름을 갖습니다.
- 테이블 페이지의 사용 가능한 테이블 목록에서 하나 이상의 사실 테이블을 선택하십시오.
- 조인 페이지에서 선택한 테이블 간의 조인을 지정하십시오. 기존의 조인을 선택하거나 조인 작성을 눌러 새 조인을 작성할 수 있는 마법사를 열 수 있습니다. 선택한 모든 테이블은 루프 없이 조인해야 합니다. 각 테이블 쌍 사이에 하나의 조인만을 지정할 수 있습니다. 테이블 페이지에서 하나의 테이블만을 선택한 경우 조인을 지정할 필요가 없습니다.
- 치수 페이지에서 간단한 치수, 계산된 치수 또는 둘 모두를 작성하십시오.

간단한 치수

컬럼에 직접 맵핑되는 치수입니다. 컬럼을 선택된 치수 목록으로 이동하면 컬럼에 직접 맵핑되는 간단한 치수가 작성됩니다.

계산된 치수

표현식을 참조하고 하나 이상의 컬럼, 속성 또는 기타 치수에서 작성된 치수입니다. 계산된 치수를 작성하려면 계산된 치수 작성을 누르십시오.

- 집계 페이지에서 각 치수에 대한 간단한 집계 함수를 지정하십시오. 디폴트 집계 함수는 숫자 데이터 유형의 치수일 경우 SUM입니다. 문자 데이터 유형의 치수일 경우 디폴트 집계 함수는 COUNT입니다. 집계를 변경하려면 테이블에서 집계를 누르고 다른 함수를 선택하거나 아무 것도 선택하지 마십시오.
- 완료 버튼을 눌러 사실 오브젝트를 작성하십시오.

차원 작성

차원을 큐브 모델의 일부분으로 작성하거나 나중에 사용할 차원을 작성할 수 있습니다.

큐브 모델에 대한 차원을 작성하려면 먼저 사실 오브젝트를 작성하십시오.

차원 오브젝트는 관련 속성 세트와 치수의 일부 측면을 설명하는 속성들 간의 조인을 정의합니다. snowflake 스키마 사용과 같이 속성을 그룹화하는 데 조인이 필요한 경우 필요한 조인과 조인에 사용된 속성이 차원 정의에 나열됩니다. 지역이라고 하는 차원에는 State, Country, City 및 Population과 같은 속성이 포함될 수 있습니다. 차원은 차원의 탐색 및 계산을 작동시키는 데 사용할 수 있는 계층 구조를 참조합니다. 차원에는 차원의 성격을 설명하는 유형이 있습니다.

차원을 작성하려면 다음을 수행하십시오.

- 큐브 모델을 마우스 오른쪽 단추로 누른 후 차원 작성을 눌러 차원 마법사를 여십시오.

2. 이름 페이지에서 차원에 대한 이름, 비즈니스 이름, 스키마 및 주석을 지정하십시오. 차원 마법사에서 작성한 모든 속성은 차원에 대해 지정한 동일한 스키마 이름을 갖습니다.
3. 테이블 페이지에서 차원을 작성할 소스 테이블을 선택하십시오. 최소한 하나의 테이블을 선택해야 합니다.
4. 조인 페이지에서 선택한 테이블 간의 조인을 지정하십시오. 기존의 조인을 선택하거나 조인 작성을 눌러 조인을 작성할 수 있는 마법사를 열 수 있습니다. 선택한 모든 테이블은 루프 없이 조인해야 합니다. 두 테이블로 이루어진 각 세트 사이에 하나의 조인만을 지정할 수 있습니다. 테이블 페이지에서 하나의 테이블만을 선택한 경우 조인을 지정할 필요가 없습니다.
5. 속성 페이지에서 차원에 포함시킬 속성을 지정하십시오. 최소한 하나의 속성을 지정해야 합니다. 목록에서 속성을 선택하거나 계산된 속성 작성을 눌러 표현식을 작성할 수 있는 마법사를 열 수 있습니다. 오브젝트 트리는 소스 테이블을 사용 가능한 컬럼 및 속성과 함께 표시합니다. 테이블의 컬럼 중 하나에 맵핑되는 속성이 데이터 베이스에 이미 있는 경우 이 속성이 표시됩니다. 컬럼이 속성에 맵핑되지 않는 경우 컬럼 자체가 표시됩니다. 컬럼이 선택되어 선택된 속성 목록으로 이동된 경우 컬럼에 맵핑되는 속성이 작성됩니다. 차원의 계층 구조에서 사용하려는 모든 속성을 선택하십시오. OLAP 센터는 지정된 조인에 사용되는 ID 속성을 차원에 자동으로 추가합니다.
6. 유형 페이지에서 작성하려는 차원의 유형을 지정하십시오. 차원이 Time 차원임을 지정하려면 시간을 선택하고 차원이 Time 차원이 아님을 지정하려면 일반을 선택하십시오.
7. 사실-차원 조인 페이지에서 조인 오브젝트를 지정하여 차원을 사실 오브젝트에 조인하십시오. 또한 조인 마법사를 사용하여 조인을 작성할 수 있습니다. 이 페이지는 차원을 큐브 모델의 일부로 작성하는 경우에만 표시됩니다.
8. 완료 버튼을 눌러 차원을 작성하십시오.

차원에 대한 계층 구조 작성

계층 구조는 제공된 큐브 모델 차원 내에서 둘 이상의 레벨 간의 관계를 정의합니다. 하나의 레벨만 사용하는 계층 구조를 정의할 수도 있습니다. 이러한 관계를 정의하면 지정된 차원을 가로지르는 탐색 및 계산 도구가 제공됩니다. 예를 들어, CalendarYear 계층 구조에는 Year, Quarter 및 Month와 같은 레벨이 포함될 수 있습니다.

차원에 대한 계층 구조를 작성하려면 먼저 차원을 작성하십시오.

큐브 모델에 있는 차원에 대해 여러 계층 구조를 정의할 수 있습니다. 레벨 간의 관계는 계층 구조 유형에 의해 결정됩니다.

차원에 대한 계층 구조를 작성하려면 다음을 수행하십시오.

1. 차원 폴더를 펼쳐서 계층 구조 마법사를 연 후 기존의 큐브 차원을 보십시오. 차원을 마우스 오른쪽 단추로 누르고 계층 구조 작성을 누르십시오
2. 이름 페이지에서 계층 구조에 대한 이름, 비즈니스 이름, 스키마 및 모든 주석을 지정하십시오.
3. 레벨 페이지에서 계층 구조의 레벨 및 계층 구조 유형을 지정하십시오.
 - a. 사용 가능 레벨 목록에 레벨이 없는 경우 레벨 작성을 누르고 레벨 작성 마법사를 사용하여 레벨을 작성하십시오.
 - b. 원하는 레벨을 지정하십시오. 최소한 하나의 레벨을 선택해야 합니다.
 - c. 계층 구조에 대한 유형 및 전개를 지정하십시오. 반복 계층 구조 유형을 지정할 경우 두 레벨만을 선택할 수 있습니다.

최소한 하나의 레벨을 선택한 후에 샘플 표시를 눌러 계층 구조의 샘플 데이터를 볼 수 있습니다.
4. 완료 버튼을 눌러 큐브 모델에 새 계층 구조를 추가하십시오.

각 계층 구조에 대한 레벨 작성

레벨에서 관련 속성 세트 간 관계를 정의합니다. 가능한 경우 DB2 Cube Views가 레벨 속성 간 관계를 정의하는 함수 종속성을 작성합니다.

1. OLAP 센터 오브젝트 트리에서 큐브 모델의 차원 폴더 아래에 있는 차원을 확장하여 레벨 마법사를 여십시오. 레벨 폴더를 마우스 오른쪽 단추로 누르고 레벨 작성을 누르십시오.
2. 이름 페이지에서 큐브 모델에 대한 이름, 비즈니스 이름, 스키마 및 주석을 지정하십시오.
3. 레벨 키 속성 페이지에서 고유하게 레벨을 정의한 속성을 하나 이상 선택하십시오.
4. 디폴트 속성 페이지에서 레벨 데이터를 설명하는 속성을 하나 선택하십시오.
5. 관련 속성 페이지에서 레벨에 관한 자세한 정보를 제공하는 속성을 0개 이상 선택하십시오.
6. 함수 종속성을 사용한 최적화 페이지에서 레벨 키 속성과 디폴트 및 관련 속성 사이에 적절한 함수 종속성을 작성할지를 지정하십시오. 레벨 키 속성이 기능적으로 디폴트 속성 및 관련 속성 모두를 판별하는지가 확실한 경우에만 이 선택란을 선택해야 합니다.
7. 완료 버튼을 눌러 레벨을 작성하십시오

큐브 모델에 기존 차원 추가

큐브 모델에 기존 차원을 추가할 수 있습니다. 다른 큐브 모델의 차원을 사용하거나 다른 큐브 모델의 차원을 사용했지만 나중에 이를 제거한 경우 차원이 이미 존재할 수 있습니다.

차원을 추가하기 전에 큐브 모델 및 사실 오브젝트를 작성하십시오. 또한 큐브 모델의 일부가 아닌 기존 차원을 갖고 있어야 합니다.

기존 차원을 추가하려면 다음을 수행하십시오.

1. 큐브 모델을 마우스 오른쪽 단추로 누른 후 차원 추가를 눌러 차원 추가 마법사를 여십시오.
2. 차원 페이지에서 하나 이상의 기존 차원을 선택하여 큐브 모델을 추가하십시오. 최소한 하나의 차원을 선택하십시오.
3. 사실-차원 조인 페이지에서 조인 오브젝트를 지정하여 추가된 차원을 큐브 모델의 사실 오브젝트에 조인하십시오.

차원에 적절한 기존 조인이 하나 있는 경우 해당 조인이 이에 대응하는 차원에 표시됩니다. 디폴트 조인이 적절한지, 조인의 한 측면에서 속성이 적절한 차원 테이블의 컬럼을 참조하는지 및 조인의 다른 측면에서 속성이 사실 테이블의 컬럼을 참조하는지를 확인하십시오.

적절한 기존 조인이 없거나 적절한 기존 조인이 둘 이상인 경우 조인을 지정해야 합니다. 다른 조인을 지정하거나 특정 차원에 대한 조인을 작성하려면, 차원을 선택한 후 조인 지정을 누르십시오.

4. 완료를 눌러 큐브 모델에 차원을 추가하십시오.

기존 차원의 사실-차원 조인 지정

기존 차원의 사실-차원 조인을 큐브 모델에 지정할 수 있습니다.

- 기존 조인을 지정하려면 다음을 수행하십시오.
 1. 후보 조인의 목록에서 조인을 선택하십시오.
 2. 확인을 누르십시오.
- 조인을 작성하려면 다음을 수행하십시오.
 1. 조인 작성을 누르십시오. 조인 마법사가 열립니다.
 2. 원하는 조인을 작성하십시오. 새 조인이 기존 후보 조인의 목록에 나타나게 됩니다.
 3. 후보 조인의 목록에서 작성한 조인을 선택하십시오.
 4. 확인을 누르십시오.

조인 작성

조인을 작성하여 사실 오브젝트의 일부 또는 차원의 일부로 차원을 사실 오브젝트에 조인할 수 있습니다.

조인 오브젝트는 두 개의 관계형 테이블을 함께 조인합니다. 그런 다음, 조인은 조인되는 테이블의 컬럼을 참조하는 속성을 참조합니다. 조인에는 유형과 카디널리티도 있습니다.

가장 간단한 조인 형식은 두 개의 속성(즉, 첫 번째 테이블의 컬럼에 맵핑되는 속성과 두 번째 테이블의 컬럼에 맵핑되는 속성)을 컬럼 비교 방법을 나타내는 연산자와 함께 참조합니다.

조인 오브젝트는 첫 번째 테이블에서 둘 이상의 컬럼을 두 번째 테이블에 있는 동일한 수의 컬럼에 조인하는 복합 조인을 모델링하는 데에도 사용할 수 있습니다. 복합 조인은 속성 쌍을 사용하여 해당 컬럼을 함께 맵핑합니다. 각 속성 쌍에는 컬럼 쌍을 비교하는 방법을 나타내는 연산자가 있습니다.

조인 오브젝트는 큐브 모델의 차원을 사실 오브젝트로 조인하기 위해 큐브 모델에 주로 사용됩니다. 또한 조인을 사용하여 snowflake 스키마의 차원 테이블을 함께 조인할 수 있습니다. 또는 종종 사실 오브젝트에서 여러 사실 테이블을 함께 조인하는 데 사용됩니다.

조인을 작성하려면 다음을 수행하십시오.

1. 조인 마법사를 여십시오.
2. 이름 페이지에서 조인에 대한 이름, 비즈니스 이름, 스키마 및 모든 주석을 지정하십시오.

팁: 조인되는 두 테이블의 이름이 모두 포함된 이름을 조인에 사용하십시오. 예를 들어, SalesFact 테이블 및 Product 테이블을 조인하는 경우, 조인의 이름 (SalesFact-Product)을 지정하십시오.

3. 조인 페이지에서 하나 이상의 속성 쌍을 추가하고 조인 유형 및 조인의 카디널리티(cardinality)를 선택하십시오. 왼쪽 속성 및 오른쪽 속성을 선택한 다음 추가를 눌러 속성 쌍을 작성하십시오. 속성 쌍이 속성 쌍 테이블에 나타납니다. 디폴트 조인 연산자는 =입니다. 테이블에서 현재 연산자를 누르고 새 연산자를 선택하여 연산자를 변경할 수 있습니다. 디폴트 조인 유형은 내부이며 디폴트 카디널리티는 일대일입니다.
4. 완료를 눌러 조인을 작성하십시오

큐브 작성

큐브를 작성하여 중요한 큐브 모델의 영역을 지정할 수 있습니다. 또한 큐브를 사용하여 비즈니스 대상 응용프로그램의 데이터 서브세트를 정의할 수 있습니다.

큐브를 작성하기 전에 큐브 모델을 작성하거나 импорт하십시오. 큐브 모델은 사실 오브젝트 및 최소한 하나의 차원을 갖고 있어야 하며, 각각의 차원은 계층 구조를 갖고 있어야 합니다.

큐브는 단일 SQL문을 사용하여 쿼리할 수 있는 OLAP 큐브에 대한 정확한 정의입니다. 큐브는 기존의 큐브 모델에서 파생된 것입니다. 큐브 사실 및 큐브 차원 목록은 참조된 큐브 모델에 있는 사실 및 차원의 서브세트입니다. 큐브 차원은 큐브 차원 당 하나의 큐브 계층 구조만을 허용하기 때문에 큐브는 여러 계층 구조를 사용하지 않는 도구와 응용프로그램에 적합합니다.

큐브를 작성하려면 다음을 수행하십시오.

1. 큐브 모델을 마우스 오른쪽 단추로 누른 후 큐브 작성을 눌러 큐브 마법사를 여십시오.
2. 이름 페이지에서 큐브에 대한 이름, 비즈니스 이름, 스키마 및 주석을 지정하십시오.
3. 치수 페이지에서 큐브에 포함시킬 치수를 선택하십시오. 적어도 하나의 치수를 선택해야 합니다.
4. 차원 페이지에서 큐브에 포함시킬 큐브 차원을 선택하십시오. 적어도 하나의 큐브 차원을 선택해야 합니다. 큐브 차원을 선택하고 선택된 큐브 차원 옆의 누름 단추를 눌러 창을 열면 큐브 차원에 대한 계층 구조 정보를 지정할 수 있습니다.
5. 쿼리 유형 지정 페이지에서 이 큐브의 사용 방법을 지정하십시오. 고급 설정을 선택한 경우 지정을 눌러 이 큐브에서 가장 자주 쿼리하려는 특정 최적화 슬라이스를 지정하십시오.
6. 완료를 눌러 큐브를 작성하십시오.

큐브 계층 구조 및 큐브 차원에 대한 큐브 레벨 지정

각 큐브 차원의 큐브 계층 구조에 포함시키도록 큐브 모델 및 관련 속성을 지정하여 다른 응용프로그램의 사용자 정의 큐브를 작성할 수 있습니다. 큐브 레벨의 서브세트 및 관련 속성을 지정하여 큐브 계층 구조에서 참조할 수 있습니다.

큐브 마법사로 연 큐브 계층 구조의 속성 선택 창에서 큐브 차원의 큐브 계층 구조를 지정하려면 다음을 수행하십시오.

1. 가능한 계층 구조 목록에서 큐브 계층 구조를 선택하십시오.
2. 레벨 및 속성 목록에서 필요한 큐브 레벨 또는 관련 속성 옆의 선택란을 선택했는지 확인하십시오. 레벨 키 속성 및 디폴트 속성이 레벨 선택 시 포함됩니다.
3. 확인을 눌러 큐브 레벨 및 관련 속성을 큐브 계층 구조에 추가하고 큐브 마법사로 리턴하십시오.

큐브 최적화 슬라이스 지정

선택사항이지만 큐브 모델의 가장 중요한 영역에 초점을 맞춘 요약 테이블을 제공하도록 최적화 어드바이저를 안내하는 데 큰 도움이 되는 최적화 슬라이스를 큐브에 지정할 수 있습니다.

최적화 슬라이스는 가장 자주 쿼리하는 큐브 영역을 표시합니다. 최적화 슬라이스 유형은 해당 영역의 쿼리 방법을 표시합니다. 예를 들어 쿼리의 50%에 월이 있는 경우 임

의-월-임의(Any-Month-Any) 슬라이스에서 최적화 슬라이스를 정의할 수 있습니다. 최적화 어드바이저는 사용자가 제공하는 최적화 슬라이스를 사용하여 사용자 필요에 가장 근접한 요약 테이블을 권장할 수 있습니다.

큐브에 최적화 슬라이스를 지정하려면 다음을 수행하십시오.

1. 큐브 마법사 또는 큐브 등록 정보 창에서 쿼리 유형 페이지를 열고 고급 설정을 선택한 후 지정을 누르십시오.
2. 큐브에 최적화 슬라이스를 하나 이상 정의하십시오.

슬라이스 추가

새로 작성을 눌러 슬라이스를 추가하십시오. 새 슬라이스가 대화식 그래픽에 최적화 슬라이스 목록 테이블의 행으로 나타납니다. 슬라이스를 추가한 후 슬라이스를 수정하여 가장 자주 쿼리하려는 각 큐브 차원의 큐브 레벨 및 쿼리 유형을 지정하십시오.

슬라이스 수정

대화식 그래픽 또는 그래픽 아래의 최적화 슬라이스 목록 테이블에서 슬라이스를 수정할 수 있습니다.

테이블의 슬라이스를 수정하려면 슬라이스를 표시하는 테이블의 행을 선택하십시오. 수정하려는 유형 또는 큐브 차원을 누르고 옵션을 선택하십시오. 대화식 그래픽에서 슬라이스를 수정하려면 큐브 차원의 노드를 레벨로 끌어 레벨을 변경하고 슬라이스를 마우스 오른쪽 단추로 누른 후 쿼리 유형을 선택하여 유형을 변경하십시오.

각 큐브 차원에서 다음 옵션 중 하나를 선택할 수 있습니다.

- 슬라이스가 있어야 하는 큐브 레벨의 환경설정이 없는 경우 임의를 선택하십시오. 최적화 어드바이저가 최적화할 큐브 레벨을 판별합니다.
- 모든 제품 또는 모든 영역에서와 같이 최고 집계 레벨에서 데이터를 자주 쿼리하는 경우 모두를 선택하십시오.
- 대부분의 쿼리가 사용하는 큐브 레벨을 알고 있는 경우 큐브 차원에 정의된 특정 큐브 레벨을 선택하십시오. 예를 들어 임의-월-임의(Any-Month-Any) 큐브 레벨에서 최적화 슬라이스를 정의할 수 있습니다. 여기서 Market 큐브 차원에서는 **Any**, Time 큐브 차원에서는 **Month**, Product 큐브 차원에서는 **Any**를 선택합니다.

권장사항: 최상의 결과를 얻으려면 하나의 큐브에 적은 수(세 개 이하)의 쏘점이 맞춰진 슬라이스를 정의하십시오.

슬라이스 제거

테이블 또는 대화식 그래픽에서 슬라이스를 선택한 후 제거를 눌러 슬라이스를 삭제할 수 있습니다.

큐브 모델에서 차원 제거

차원이 더 이상 필요하지 않은 경우 큐브 모델에서 해당 자원을 제거할 수 있습니다. 차원이 다른 큐브 모델에 사용되는 경우 차원을 삭제하지 않아도 차원을 제거할 수 있습니다.

차원을 큐브 모델에서 제거하는 경우에는 다음 조치가 적용됩니다.

- 차원은 선택한 큐브 모델에서만 제거되며, 해당 차원을 참조하는 모든 기타 큐브 모델에서는 계속 유지됩니다.
- 큐브 모델의 이 차원을 기반으로 하는 모든 큐브 차원은 대응되는 큐브에서 제거됩니다.
- 차원은 데이터베이스에서 삭제되지 않습니다.
- 차원은 모든 차원 폴더에서 사용할 수 있습니다.

큐브 모델에서 차원을 제거하려면 다음을 수행하십시오.

1. OLAP 센터 오브젝트 트리에서 큐브 모델 폴더를 펼쳐서 기존의 큐브 모델을 보십시오. 오브젝트 트리에 큐브 모델 폴더가 포함되지 않은 경우 보기 -> **OLAP 오브젝트 표시**를 눌러 OLAP 오브젝트 보기 안에 있는지 확인하십시오.
2. 큐브 모델 노드를 펼쳐서 큐브 모델에 포함된 오브젝트 범주를 보십시오.
3. 차원 폴더를 펼쳐서 기존의 차원을 보십시오.
4. 차원을 마우스 오른쪽 단추로 누르고 제거를 누르십시오.

데이터베이스에서 메타데이터 오브젝트 삭제

이 데이터베이스에 있는 큐브 모델의 메타데이터 오브젝트를 더 이상 사용하지 않는 경우 메타데이터 오브젝트를 삭제할 수 있습니다.

OLAP 오브젝트 보기나 관계 오브젝트 보기에서 대부분의 오브젝트를 삭제할 수 있습니다. 다른 오브젝트가 참조하지 않는 오브젝트만을 삭제할 수 있습니다.

데이터베이스에서 메타데이터 오브젝트를 삭제하려면 다음을 수행하십시오.

OLAP 센터 오브젝트 트리에서 하나 이상의 오브젝트를 선택하고 선택한 오브젝트를 마우스 오른쪽 단추로 누른 후 삭제를 누르십시오. 메뉴에 삭제 메뉴 항목이 포함되지 않은 경우 선택된 오브젝트를 삭제할 수 없습니다. 상위 오브젝트를 삭제하면 모든 하위 오브젝트가 삭제됩니다. 예를 들어, 차원을 삭제하면 모든 해당 계층 구조도 삭제됩니다.

제 4 장 DB2 Cube Views 비즈니스 모델링 시나리오

이 섹션에서는 다음 주제를 설명합니다.

웨어하우스에 시간 초과된 재고의 플로우 및 값 계산

소매 사업인 XYZ Retail은 재고가 판매를 위해 특정 상점으로 보내지기 전에, 웨어하우스에 이를 보관합니다. XYZ Retail은 웨어하우스에 시간 초과된 재고 상태에 대한 데이터를 유지보수하며 이 데이터를 분석하려고 합니다.

광고 비용과 판매액의 상관관계

자동차 판매점이 광고에 대한 지출 증가를 고려하고 있습니다. 올바른 결론을 내기 위해 판매점은 우선 광고 지출과 판매액 사이의 기록상의 관계를 분석하려고 합니다. 판매점은 광고의 레벨을 다양화하는 것이 판매액에 영향을 미치는지 여부 및 특히 광고의 증가와 판매액 증가가 밀접한 연관이 있는지를 판별하려고 합니다.

이익 및 상점의 이익 마진 계산

장난감 상점의 총 지배인은 제품 유형 및 시기와 같은 여러 요소가 이익 및 이익 마진에 어떻게 영향을 주는지 분석하려고 합니다.

인터넷 주문 숫자 카운팅

소매 회사는 몇 년 전에 인터넷 판매를 추가하여 사업을 확장했습니다. 이제 회사는 인터넷 판매의 효과를 분석하고자 합니다. 회사가 요구하는 첫 번째 계산 중 하나는 인터넷을 통해 완료된 주문의 수입입니다.

판매액 수치 순위

사무실 공급 상점 체인은 지난 몇 년에 걸쳐 급속하게 확장되었습니다. 비즈니스 실무자들은 성과가 가장 낮은 상점 일부를 닫아 비용을 줄이고 이익을 늘리는 것에 대해 고려 중입니다.

사실 테이블에 저장된 시간 데이터를 사용하여 **Time** 차원 작성

소매 사업인 XYZ Retail은 DB2 Cube Views에서 판매액 트랜잭션 데이터를 모델링하여 데이터를 보다 효과적으로 분석할 수 있습니다. 그러나, 데이터의 트랜잭션 특성으로 인해, 사용 가능한 유일한 시간 정보는 각 트랜잭션과 연관된 날짜입니다.

웨어하우스에 시간 초과된 재고의 플로우 및 값 계산

소매 사업인 XYZ Retail은 재고가 판매를 위해 특정 상점으로 보내지기 전에, 웨어하우스에 이를 보관합니다. XYZ Retail은 웨어하우스에 시간 초과된 재고 상태에 대한 데이터를 유지보수하며 이 데이터를 분석하려고 합니다.

특히, 회사는 웨어하우스의 다음 두 측면을 조사하려고 합니다.

- 웨어하우스로 들어오고 나가는 상품의 플로우
- 특정 시간에 웨어하우스의 상품 값

첫 번째 측면은 상품의 플로우가 시간 초과된 데이터 조사를 수반한다는 것입니다. 두 번째 측면은 상품의 값이 특정 시점에서 웨어하우스의 스냅샷을 취한다는 것입니다.

시나리오 세부사항

XYZ Retail에는 QUANTITY_IN, QUANTITY_OUT, CURRENT_QUANTITY, PRODUCT_VALUE, PRODUCT_ID 및 TIME_ID와 같은 웨어하우스 관련 컬럼이 있는 사실 테이블이 있습니다. 이 데이터는 매주 테이블에 입력됩니다. 또한 데이터베이스에는 Product 테이블 및 Time 테이블도 있습니다. 예를 들어, 샘플 사실 테이블 데이터 세트가 표 28에 표시되어 있습니다.

표 28. 샘플 사실 테이블 데이터

PRODUCT_ID	TIME_ID	QUANTITY_IN	QUANTITY_OUT	CURRENT_QUANTITY	PRODUCT_VALUE
1234	1	5	0	5	5
1234	2	20	10	15	5
1234	3	10	20	5	5

한 제품 유형이 웨어하우스로 여러 번 들어오고 나갈 수 있으므로, 세 개의 각 샘플 데이터 항목에 대해 PRODUCT_ID 값은 동일합니다.

XYZ Retail의 DBA는 다음과 같은 세 가지의 다른 치수를 작성해야 합니다.

Flow In

상품의 플로우를 웨어하우스로 모델화합니다.

Flow Out

웨어하우스의 Flow Out을 모델화합니다.

Current Value

특정 시간에 상품의 값을 모델화합니다.

첫 번째 두 치수인 Flow In 및 Flow Out을 작성하기 위해, DBA는 QUANTITY_IN 및 QUANTITY_OUT 컬럼으로 각각 맵핑되고 모든 차원에 걸쳐 데이터를 합산하는 치수를 작성합니다. 데이터가 모든 차원에 걸쳐 SUM 함수만을 사용하여 집계되므로 이

는 전체 추가 치수로 알려져 있습니다. 예를 들어, 표 29에서는 1234라는 PRODUCT_ID 를 갖는 제품에 대해 3개월 동안의 QUANTITY_IN 및 QUANTITY_OUT 컬럼의 샘플 데이터 세트를 표시합니다 Flow In 및 Flow Out 치수는 이러한 월별 값을 합산하여 분기 동안 웨어하우스로 들어오고 나가는 전체 수량을 계산합니다.

표 29. PRUDUCT_ID 1234의 Flow In 및 Flow Out 전체 추가 치수에 대한 샘플 데이터 계산

	1월	2월	3월	1/4분기
QUANTITY_IN	5	20	10	35
QUANTITY_OUT	0	10	20	30

전체 추가 치수는 작성되는 치수 중 가장 단순하고 일반적이어서, 보다 복잡한 치수에 대한 블록을 빌드하는 데 자주 사용됩니다. 숫자 소스 데이터에 근거하는 치수의 경우, OLAP 센터는 전체 추가 치수를 디폴트로 작성합니다.

세 번째 치수인 Current Value을 작성하기 위해, DBA는 PRODUCT_VALUE와 CURRENT_QUANTITY를 곱하여 값을 계산하는 계산된 치수를 작성합니다. 예를 들어, PRODUCT_ID=1234인 제품의 값이 5일 경우, 샘플 데이터의 Current Value 치수는 표 30에 표시됩니다.

표 30. PRODUCT_ID 1234의 Current Value 치수에 대한 샘플 데이터 계산

	1월	2월	3월
CURRENT_QUANTITY	5	10	20
Current Value	25	50	100

그런 다음, 이 데이터는 차원에 걸쳐 집계되어야 합니다. 그러나 특정 시점에서 이 값이 계산되고 있으므로, Time 차원에 걸쳐 합산하는 것은 합당하지 않습니다. 대신, 집계는 Product 차원에 걸쳐 데이터를 합산하고 시간 초과된 데이터의 평균을 찾습니다. 일부 집계만 SUM 함수를 수반하므로 이는 세미 추가 치수로 알려져 있습니다.

월을 분기에 추가하는 것이 이치에 맞지 않으므로, 스냅샷 데이터를 계산하는 치수 및 월별 재고 데이터와 같은 특정 시점을 나타내는 데이터는 세미 추가 치수입니다. 전체 분기 동안 제품이 웨어하우스에 남아 있는 경우, 그 제품은 분기의 각 3개월 동안의 웨어하우스 재고의 CURRENT_QUANTITY 스냅샷 데이터에 포함됩니다. CURRENT_QUANTITY 데이터가 시간 초과되어 합산되는 경우, 3개월 동안 웨어하우스에 있던 제품은 세 번 계산됩니다. 70 페이지의 표 31에서 표시한 대로, 1/4분기의 값 25에는 웨어하우스의 활동에 대한 유의값이 없습니다 테이블은 웨어하우스에 25개의 제품이 있지 않았으므로 25 제품의 값을 계산하는 것은 의미가 없다는 것을 보여줍니다.

표 31. PRODUCT_ID 1234의 Time 차원에 대한 SUM 함수를 사용하는 CURRENT_QUANTITY 컬럼의 샘플 데이터 계산

	1월	2월	3월	1/4분기
SUM(CURRENT_QUANTITY)	5	15	5	25

모든 차원에 걸쳐 SUM 함수를 사용하는 대신에, Time 차원에 대해 AVG, MIN 및 MAX와 같은 기타 집계 함수를 수행할 수 있습니다. 예를 들어, 1월, 2월 및 3월에 대한 동일한 샘플 데이터 세트를 사용하여 표 32에 표시된 대로 Time 차원에 대한 두 번째 집계 함수를 사용하여 해당 분기의 의미있는 값을 작성할 수 있습니다. Current Value 치수는 해당 분기에 웨어하우스에 저장된 상품의 평균 총계 값 또는 해당 분기의 특정 시점에서의 최대 또는 최소값을 나타냅니다.

표 32. PRODUCT_ID 1234의 Time 차원에 대한 AVG, MAX 및 MIN 함수를 사용하는 CURRENT_QUANTITY 컬럼의 샘플 데이터 계산

	1월	2월	3월	1/4분기
AVG(CURRENT_QUANTITY)	5	15	5	8.3
MAX(CURRENT_QUANTITY)	5	15	5	15
MIN(CURRENT_QUANTITY)	5	15	5	5

치수 작성 단계

다음 단계는 사용자가 OLAP 센터 사실 등록 정보 창을 사용하여 기존 사실 오브젝트에서 Flow In, Flow Out 및 Current Value 치수를 작성하는 방법을 설명합니다.

1. 사실 등록 정보 창을 열려면, 사실 오브젝트 트리에서 사실 오브젝트를 마우스 오른쪽 단추로 누르고 치수 편집을 누르십시오. 사실 등록 정보 창이 열립니다.
2. 다음과 같이 FLOW IN 치수를 작성하십시오.
 - a. 치수 페이지에서, 계산된 치수 작성을 눌러 FLOW IN 치수를 작성하십시오. SQL 표현식 빌더 창이 열립니다.
 - b. SQL 표현식 빌더 창에서, 이름 필드에 FLOW IN을 입력하십시오.
 - c. FLOW IN 표현식을 작성하려면, 다음 단계를 완료하십시오.
 - 데이터 목록에서 사실 테이블 및 컬럼 폴더를 펼치십시오
 - QUANTITY_IN 컬럼을 더블 클릭하여 이를 표현식에 추가하십시오
 - 확인을 눌러 SQL 표현식 빌더 창을 닫으십시오. 집계 페이지에서 디폴트 집계 함수인 SUM을 변경할 필요는 없습니다. 데이터 소스가 숫자이고 치수가 기존 치수 뿐만 아니라 컬럼도 참조하므로 SUM 함수는 FLOW IN 치수에 대한 기본값입니다.
3. 다음과 같이 Flow Out 치수를 작성하십시오.

- a. 치수 페이지에서, 계산된 치수 작성을 눌러 Flow Out 치수를 작성하십시오 SQL 표현식 빌더 창이 열립니다.
 - b. SQL 표현식 빌더 창에서, 이름 필드에 FLOW OUT을 입력하십시오.
 - c. Flow Out 표현식을 작성하려면, 다음 단계를 완료하십시오.
 - 데이터 목록에서 사실 테이블 및 컬럼 폴더를 펼치십시오
 - **QUANTITY_OUT** 컬럼을 더블 클릭하십시오
 - d. 확인을 눌러 SQL 표현식 빌더 창을 닫으십시오. 집계 페이지에서 디폴트 집계 함수인 SUM을 변경할 필요는 없습니다. 데이터 소스가 숫자이고 치수가 기존 치수 뿐만 아니라 컬럼도 참조하므로, SUM 함수는 Flow Out 치수에 대한 기본값입니다.
4. 다음과 같이 Current Value 치수를 작성하십시오.
- a. 치수 페이지에서, 계산된 치수 작성을 눌러 Current Value 치수를 작성하십시오 SQL 표현식 빌더 창이 열립니다.
 - b. SQL 표현식 빌더 창에서, 이름 필드에 CURRENT VALUE를 입력하십시오.
 - c. Current Value 표현식을 작성하려면, 다음 단계를 완료하십시오.
 - 데이터 목록에서 사실 테이블 및 컬럼 폴더를 펼치십시오
 - 데이터 목록에서 **PRODUCT_VALUE** 컬럼을 더블 클릭하십시오
 - 연산자 목록에서 * 연산자를 더블 클릭하십시오
 - 데이터 목록에서 **CURRENT_QUANTITY** 컬럼을 더블 클릭하십시오
- 72 페이지의 그림 12에서는 SQL 표현식 빌더 창에서 작성할 수 있는 Current Value 표현식을 표시합니다

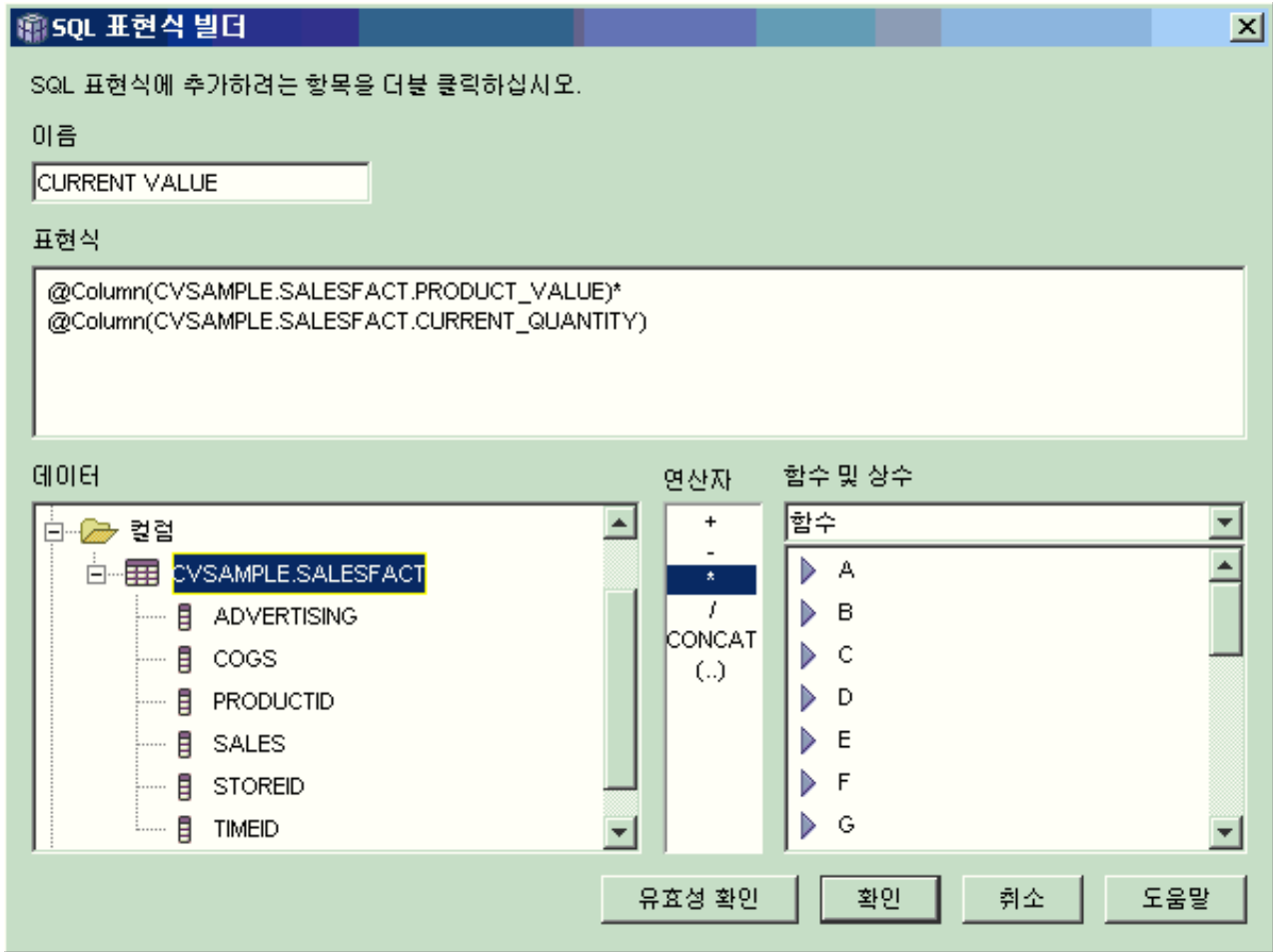



그림 12. SQL 표현식 빌더 창에서 Current Value 표현식을 완료하십시오

- d. 확인을 눌러 SQL 표현식 빌더 창을 닫으십시오.
- e. 집계 페이지에서, Current Value 치수에 대한 집계를 누르고 목록에서 집계 스크립트를 누르십시오. 집계 스크립트 빌더 창이 열립니다. 디폴트 집계 스크립트에는 모든 차원에 사용되는 SUM 함수가 있습니다.
- f. 필요한 경우, **Time**을 선택한 후  를 눌러 Time 차원을 아래로 이동하여 스크립트에서 마지막으로 나열되는 차원이 되도록 하십시오. 그런 다음, Time 차원을 선택하고 컬럼 함수 목록에서 **AVG** 함수를 더블 클릭하십시오 73 페이지의 그림 13에서 표시된 대로, 집계 스크립트가 Time을 제외한 모든 차원에 걸쳐 데이터(평균)를 합산합니다.

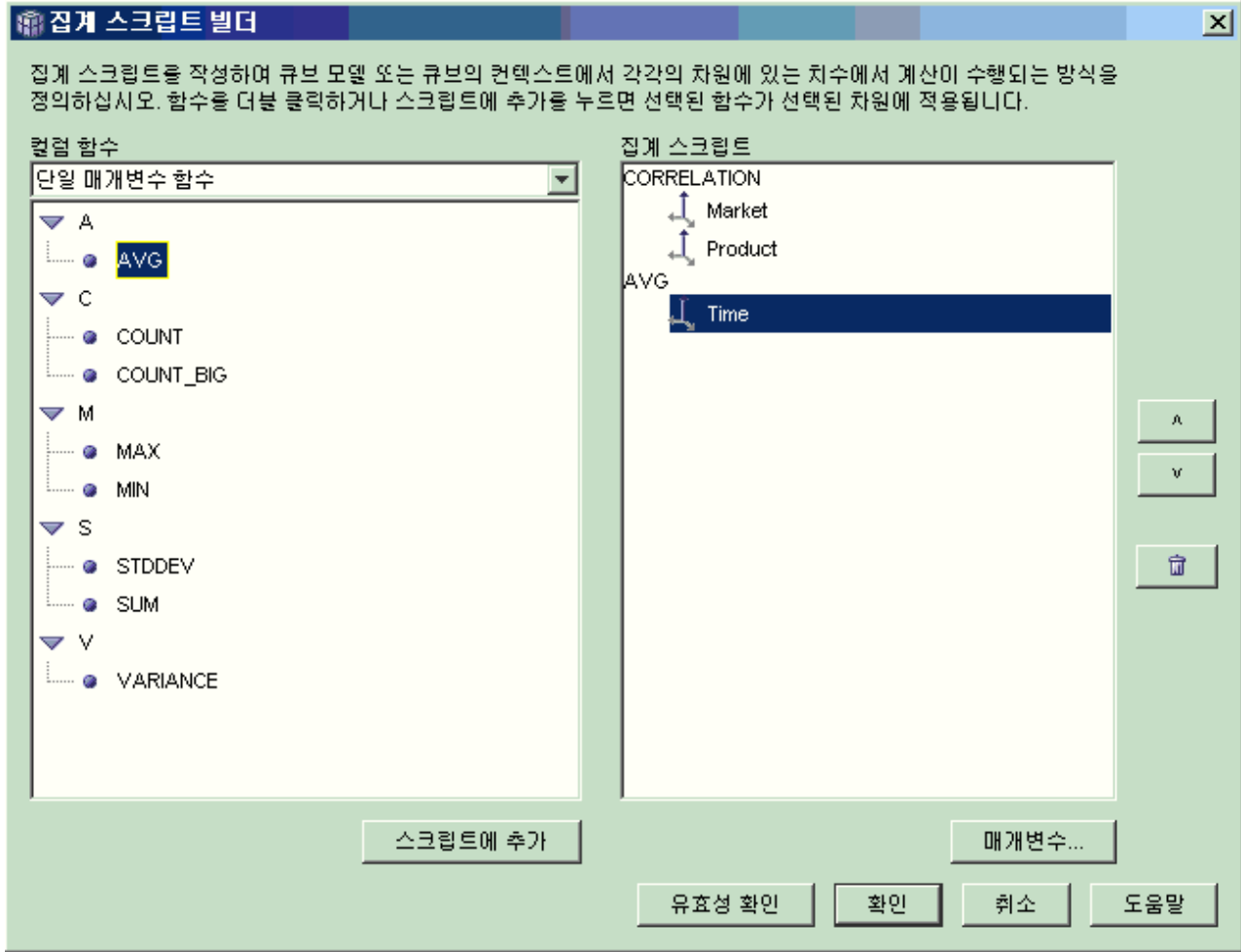


그림 13. Current Value 치수에 대한 집계 스크립트

- g. 집계 스크립트 빌더 창에서, 유효성 확인을 눌러 집계 스크립트가 유효한지 확인하십시오 확인을 눌러 집계 스크립트를 저장하고 창을 닫으십시오
- 5. 확인을 눌러 사실 오브젝트에 변경사항을 저장하고 사실 등록 정보 창을 닫으십시오

이제 사용자에게 웨어하우스의 재고에 대한 세 가지의 계산된 치수가 있습니다. 이러한 치수를 사용하여 웨어하우스로 들어오고 나가는 제품 플로우의 패턴을 분석할 수 있습니다.

광고 비용과 판매액의 상관관계

자동차 판매점이 광고에 대한 지출 증가를 고려하고 있습니다. 올바른 결론을 내기 위해 판매점은 우선 광고 지출과 판매액 사이의 기록상의 관계를 분석하려고 합니다. 판매점은 광고의 레벨을 다양화하는 것이 판매액에 영향을 미치는지 여부 및 특히 광고의 증가와 판매액 증가가 밀접한 연관이 있는지를 판별하려고 합니다.

시나리오 세부사항

판매점의 데이터베이스에는 Sales 및 Ad Costs 컬럼이 있는 사실 테이블이 있습니다. 그 데이터베이스에는 몇몇 기타 차원 테이블도 있습니다. DBA는 DB2 CORRELATION 함수를 사용하는 치수를 작성하여 비용 및 판매액 사이의 상관 계산을 수행할 수 있습니다. CORRELATION 함수는 두 개의 입력 매개변수를 요구하는 다중 매개변수 함수입니다. 이 경우, DBA는 Sales 및 Ad Costs 컬럼을 두 개의 입력 매개변수로 사용합니다.

DBA는 먼저 다중 매개변수 집계 함수를 집계 스크립트에 적용해야 합니다. 다중 매개변수 함수는 전체 차원에 걸쳐 적용될 수 있거나, Time 차원을 제외한 모든 차원에 먼저 적용된 후 MAX 함수와 같은 두 번째 함수가 시간 함수에 적용될 수 있습니다. DBA가 치수에 대한 SQL 표현식을 정의하므로 이는 직접 Ad Costs 컬럼으로 맵핑됩니다. 다중 매개변수 함수에서 사용되는 두 매개변수 중 SQL 표현식이 우선 순위입니다. DBA는 두 번째 매개변수를 Sales 컬럼에 직접 맵핑하는 SQL 표현식으로 정의합니다. CORRELATION 함수가 유일한 집계 함수로 정의되어 있으므로, 치수는 모든 차원에 걸쳐 광고 비용 및 판매액 결과 사이의 통계적 상관을 계산할 수 있습니다.

치수 작성 단계

다음 단계는 OLAP 센터 사실 등록 정보 창을 사용하여 기존 사실 오브젝트의 Advertising-Sales Correlation 치수를 작성하는 방법을 설명합니다.

1. OLAP 센터 오브젝트 트리에서 사실 오브젝트를 마우스 오른쪽 단추로 누른 후 치수 편집을 눌러 사실 등록 정보 창을 여십시오.
2. 계산된 치수 작성 누름 단추를 누르십시오. SQL 표현식 빌더 창이 열립니다.
3. SQL 표현식 빌더 창에서, 이름 필드에 ADVERTISING-SALES CORRELATION을 입력하십시오.
4. 집계 스크립트에서 다중 매개변수 CORRELATION 함수의 첫 번째 매개변수로 사용되기도 하는 치수의 표현식을 정의하십시오. 표현식을 정의하려면, 데이터 목록의 치수 폴더를 펼치고 **AD COSTS** 치수를 더블 클릭하여 이를 표현식 목록에 추가하십시오. 75 페이지의 그림 14에서는 SQL 표현식 빌더 창에서 작성하는 표현식을 표시합니다.

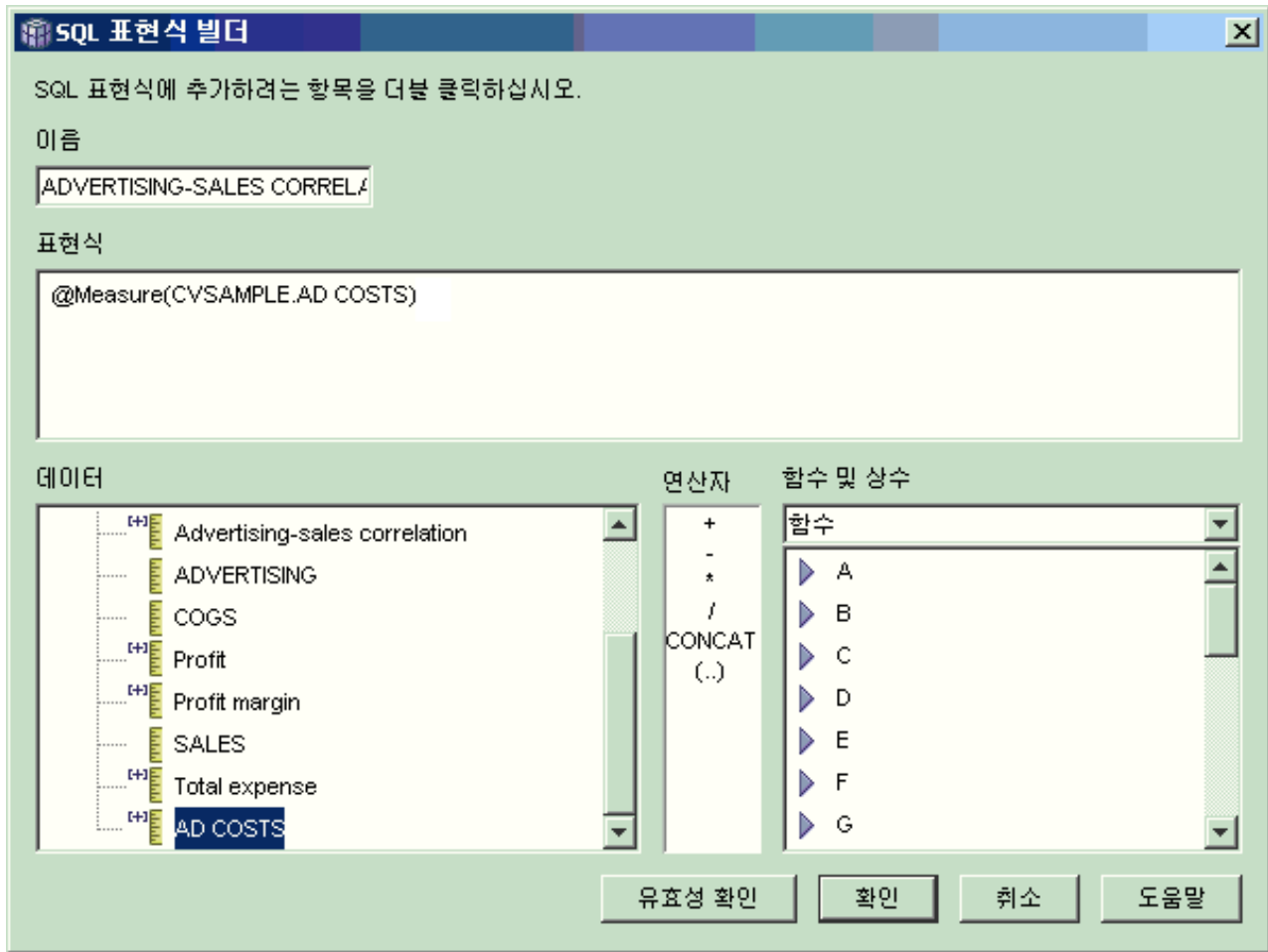


그림 14. SQL 표현식 빌더 창에서 advertising-sales correlation 상관 표현식을 완료하십시오.

5. 집계 페이지에서, **ADVERTISING-SALES CORRELATION** 치수에 대한 집계 함수를 누르고 76 페이지의 그림 15에 표시된 대로 집계 스크립트를 선택하십시오. 집계 스크립트 빌더 창이 열립니다.



그림 15. 사실 등록 정보 창의 집계 페이지

6. 컬럼 함수 필드에서, 다중 매개변수 함수를 선택하십시오. 다중 매개변수 함수 목록에서, **CORRELATION** 함수를 선택하고 스크립트에 추가를 누르십시오. 함수 매개변수 창이 열립니다.
7. 기존 치수 사용을 선택한 후 77 페이지의 그림 16에 표시된 목록에서 **SALES**를 선택하십시오.

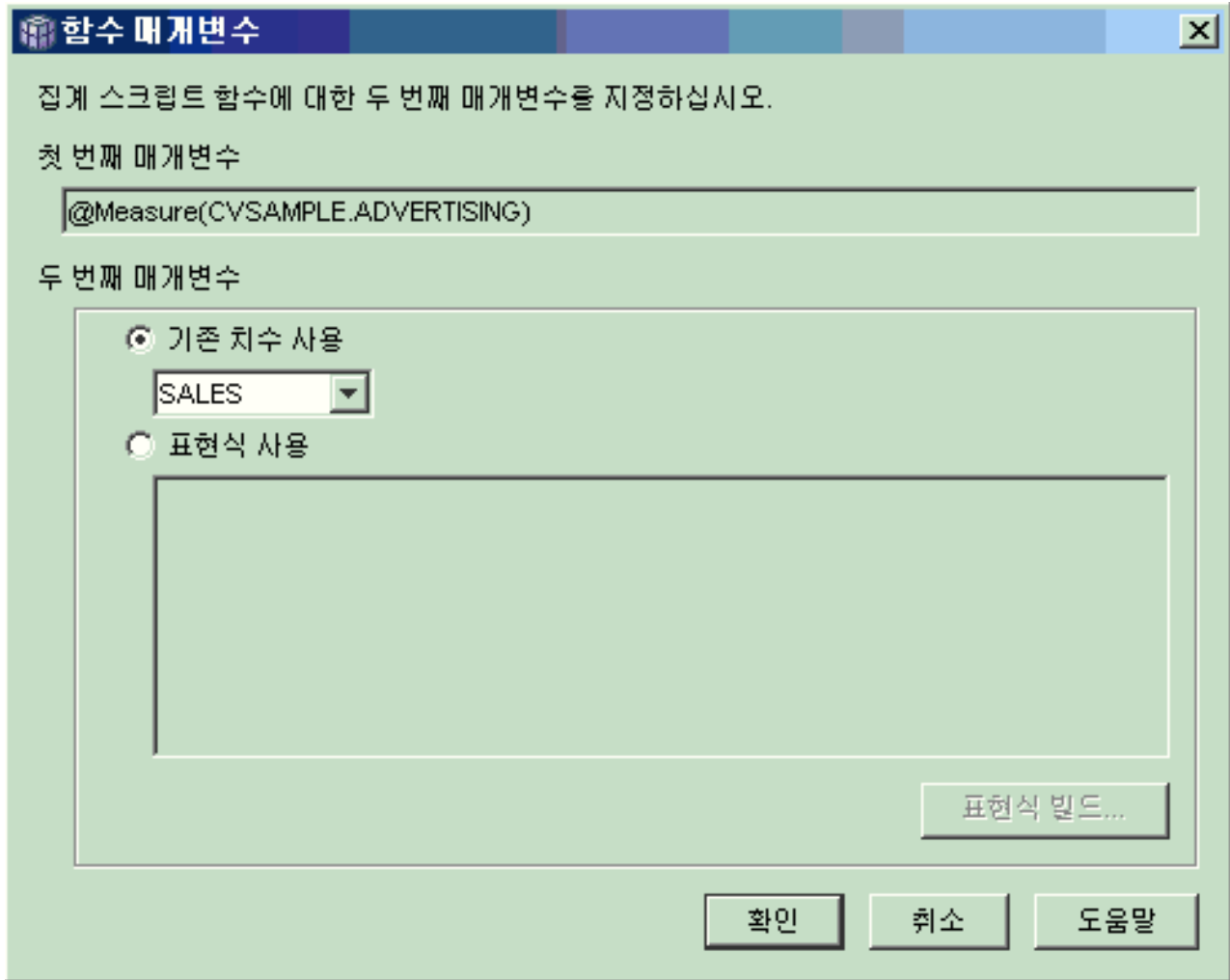


그림 16. 함수 매개변수 창에서 두 번째 매개변수로 지정된 Sales 치수

8. 확인을 눌러 선택사항을 저장하고 함수 매개변수 창을 닫으십시오.
9. 78 페이지의 그림 17에서는 CORRELATION 함수가 스크립트의 차원 목록 위쪽에 있음을 표시합니다

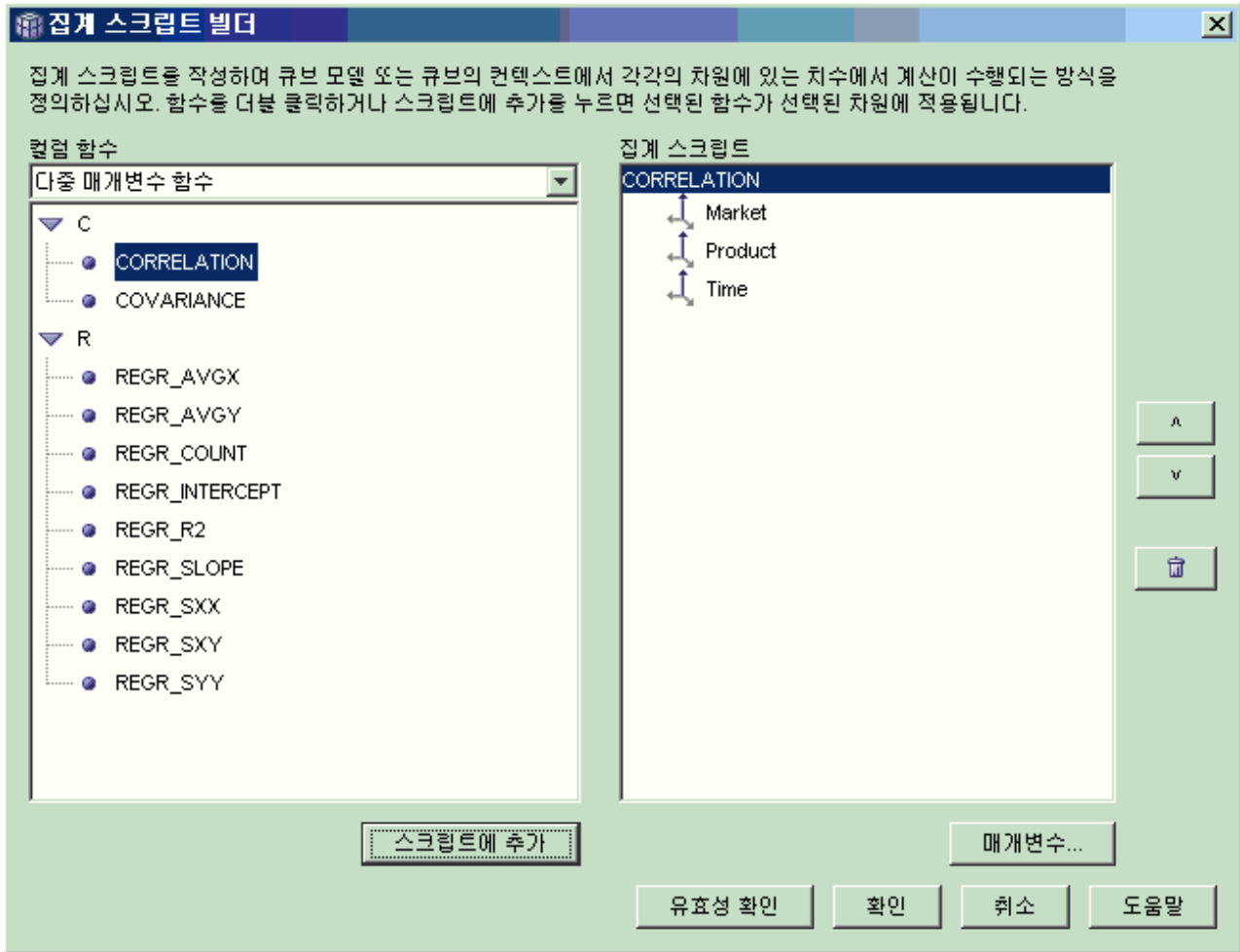


그림 17. Advertising-Sales Correlation 치수에 대한 집계 스크립트

10. 집계 스크립트 빌더 창에서, 유효성 확인 누름 단추를 눌러 집계 스크립트가 유효한지 확인하십시오
11. 확인을 눌러 집계 스크립트를 저장하고 창을 닫으십시오
12. 확인을 눌러 사실 오브젝트에 변경사항을 저장하고 사실 등록 정보 창을 닫으십시오

이제 데이터베이스에 두 가지의 데이터 유형을 상관시키는 치수가 있습니다. 이 치수를 사용함으로써 기록된 경향의 결과에 근거하여 이후의 광고 지출을 결정할 수 있습니다.

이익 및 상점의 이익 마진 계산

장난감 상점의 총 지배인은 제품 유형 및 시기와 같은 여러 요소가 이익 및 이익 마진에 어떻게 영향을 주는지 분석하려고 합니다.

고급 분석을 완료하기 전에, 장난감 상점의 DBA는 profit 및 profit margin 치수를 먼저 작성해야 합니다. 그런 다음, DBA는 profit 및 profit margin 치수와 다른 인수들을 상관시키고 비교하는 추가 치수를 작성할 수 있습니다.

시나리오 세부사항

장난감 상점의 데이터베이스에는 몇몇 차원 테이블 각각에 해당되는 외부 키 컬럼 뿐만 아니라, Sales, Costs of Goods Sold(COGS) 및 Expense 컬럼이 있는 사실 테이블이 있습니다. DBA는 Sales, COGS 및 Expense 컬럼에 각각 맵핑되는 Sales, COGS 및 Expense 치수를 이미 작성했습니다. Profit 및 Profit Margin 치수는 이러한 기존 치수로부터 전적으로 작성될 수 있습니다.

Profit 치수를 작성하려면, DBA는 SQL 표현식의 SALES-(COGS+EXPENSE)를 계산하는 치수를 작성하고 모든 차원에 걸쳐 계산된 데이터를 합산합니다. 기존 치수 또는 컬럼 또는 둘의 조합을 참조하여 Profit 치수를 작성할 수 있습니다.

Profit 치수가 작성된 후에, DBA는 Profit Margin 치수를 작성할 수 있습니다. Profit Margin 치수는 백분율((Profit / Sales)*100)로 표현되는 기존 두 치수의 비율이며 해당 집계 함수가 필요하지 않습니다. 치수는 이미 집계된 데이터의 다른 치수를 참조하기만 하므로 집계 함수가 필요하지 않습니다. DBA가 다른 치수를 참조하기만 하는 혼합 치수를 사용하여 비율을 계산할 경우, DBA는 추가 집계를 정의할 필요가 없습니다. SUM과 같은 대부분의 집계 함수는 비율에는 적용되지 않습니다. 예를 들어, 장난감 상점이 전체 4분기 동안 40%, 32%, 28%, 37%의 이익 마진을 남긴 경우, 전체 비율을 합산하여 1년에 137%의 이익 마진을 남긴다는 것은 이치에 맞지 않습니다.

치수 작성 단계

다음 단계는 사용자가 OLAP 센터 사실 등록 정보 창을 사용하여 기존 사실 오브젝트에서 profit 및 profit margin 치수를 작성하는 방법을 설명합니다.

1. 사실 등록 정보 창을 열려면, 사실 오브젝트 트리에서 사실 오브젝트를 마우스 오른쪽 단추로 누르고 치수 편집을 누르십시오. 사실 등록 정보 창이 열립니다.
2. 다음과 같이 profit 치수를 작성하십시오.
 - a. 치수 페이지에서, 계산된 치수 작성 누름 단추를 누르십시오. SQL 표현식 빌더 창이 열립니다.
 - b. SQL 표현식 빌더 창에서, 이름 필드에 PROFIT을 입력하십시오.
 - c. Profit 표현식을 작성하려면, 치수 폴더를 데이터 목록에서 펼친 후 다음 단계를 완료하십시오.
 - 데이터 목록에서 **SALES** 치수를 더블 클릭하여 이를 표현식에 추가하십시오
 - 연산자 목록에서 - 연산자를 더블 클릭하십시오
 - 데이터 목록에서 **COGS** 치수를 더블 클릭하십시오

- 연산자 목록에서 + 연산자를 더블 클릭하십시오
- 데이터 목록에서 **EXPENSE** 치수를 더블 클릭하십시오
- 표현식 필드에서 @Measure(MDSAMPLE.COGS)+@Measure(MDSAMPLE.EXPENSE) 부분을 강조표시한 후 연산자 목록에서 (..)을 더블 클릭하여 선택한 표현식 부분을 괄호로 묶으십시오.

그림 18에서는 SQL 표현식 빌더 창에서 작성할 수 있는 Profit 표현식을 표시합니다

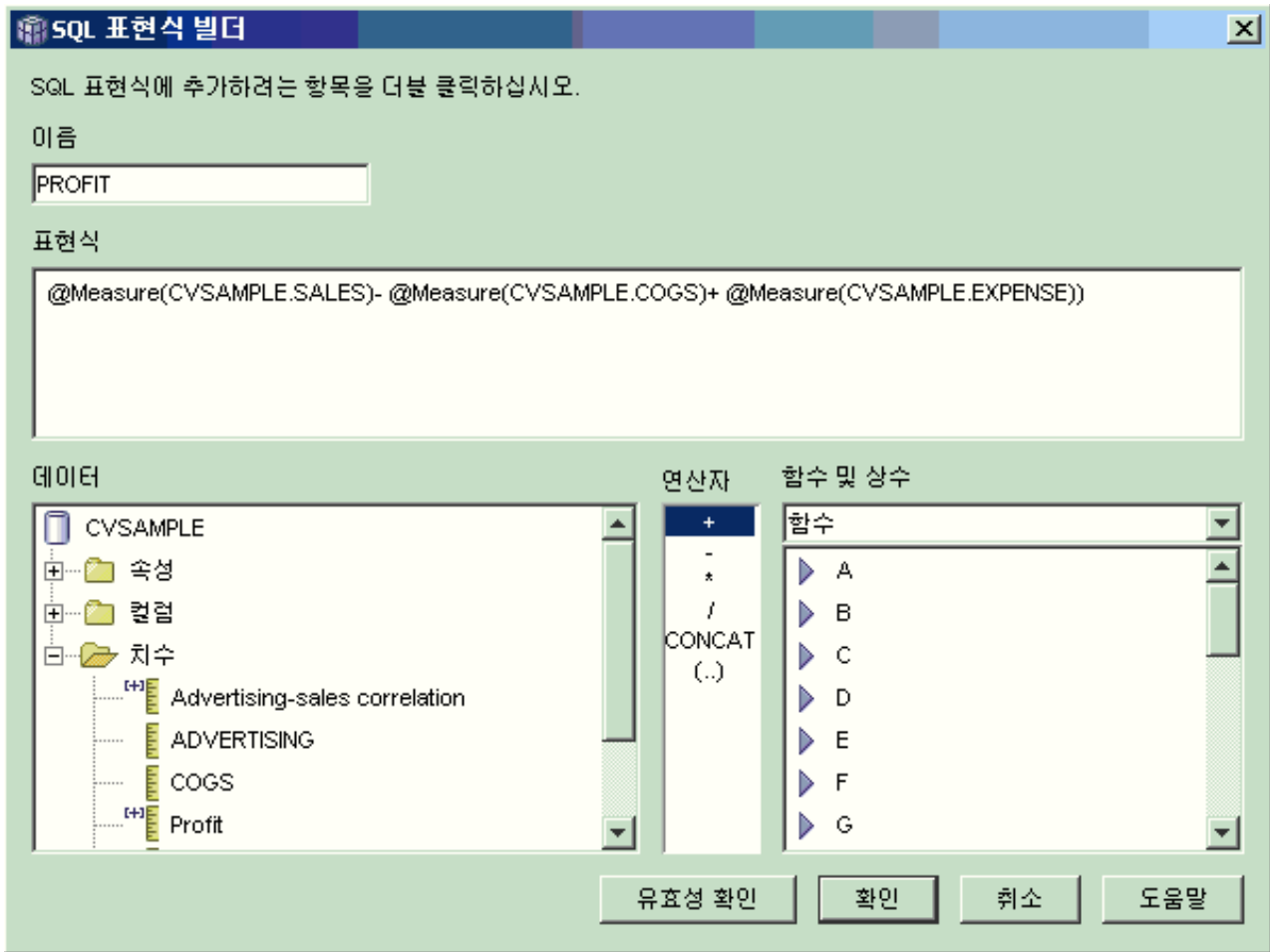


그림 18. SQL 표현식 빌더 창에서 profit 표현식을 완료하십시오.

- d. 확인을 눌러 profit 치수를 작성하고 SQL 표현식 빌더 창을 닫으십시오
 - e. 집계 페이지에서, profit 집계를 누르고 SUM 기능을 선택하십시오. profit 치수가 완료됩니다.
3. 다음과 같이 Profit Margin 치수를 작성하십시오.
 - a. 치수 페이지에서, 계산된 치수 작성을 누르십시오 SQL 표현식 빌더 창이 열립니다.

- b. SQL 표현식 빌더 창에서, 이름 필드에 PROFIT MARGIN을 입력하십시오.
- c. Profit Margin 표현식을 작성하려면, 치수 폴더를 데이터 목록에서 펼친 후 다음 단계를 완료하십시오
 - 데이터 목록에서 **PROFIT** 치수를 더블 클릭하여 이를 표현식에 추가하십시오
 - 연산자 목록에서 / 연산자를 더블 클릭하십시오
 - 데이터 목록에서 **SALES** 치수를 더블 클릭하십시오
 - 표현식 필드에 입력하여 괄호 안의 전체 표현식을 닫으십시오
 - 표현식의 끝에 커서를 위치시키고 연산자 목록에서 * 연산자를 더블 클릭하십시오.
 - 표현식 필드에서 표현식의 끝에 100을 입력하십시오.

그림 19에서는 SQL 표현식 빌더 창에서 작성할 수 있는 Profit margin 표현식을 표시합니다.

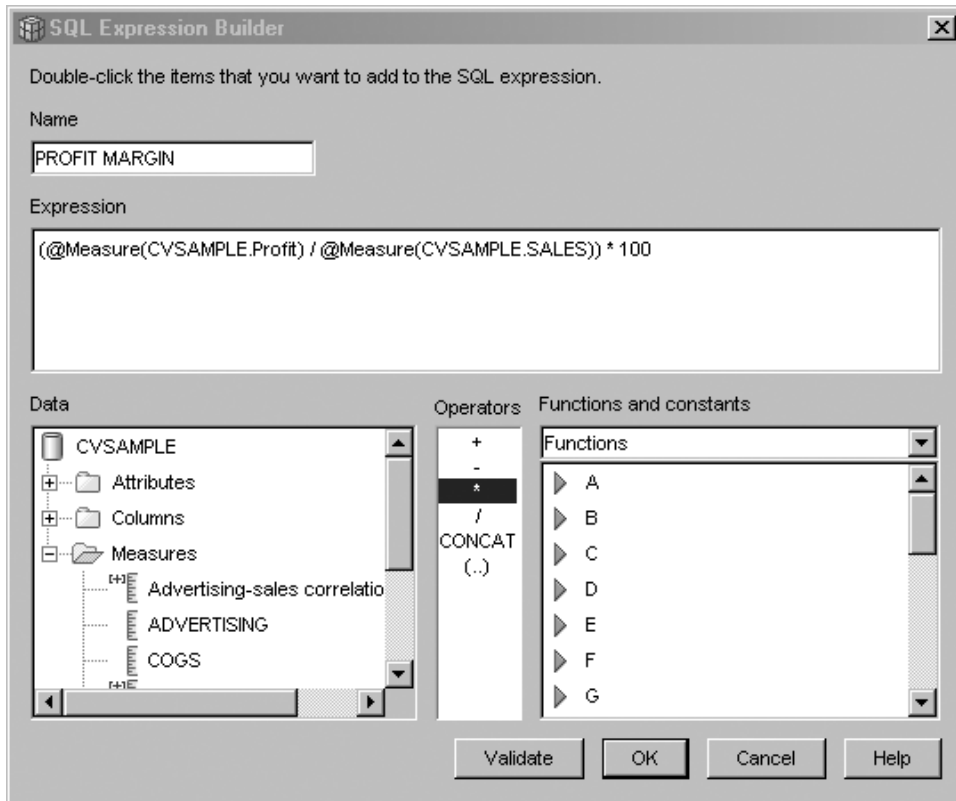


그림 19. SQL 표현식 빌더 창에서 profit margin 표현식을 완료하십시오.

- d. 확인을 눌러 Profit Margin 치수를 작성하고 SQL 표현식 빌더 창을 닫으십시오

집계 페이지에서, OLAP 센터가 혼합 치수에 대한 디폴트값으로 집계 함수를 NONE으로 설정하므로 집계 함수를 변경할 필요가 없습니다.

4. 확인을 눌러 사실 등록 정보 창을 닫고 사실 오브젝트에 추가한 두 개의 새 치수를 저장하십시오

DBA가 이러한 두 치수를 작성한 후에, 이 중요한 치수에 대해 추가 분석이 완료될 수 있습니다.

인터넷 주문 숫자 카운팅

소매 회사는 몇 년 전에 인터넷 판매를 추가하여 사업을 확장했습니다. 이제 회사는 인터넷 판매의 효과를 분석하고자 합니다. 회사가 요구하는 첫 번째 계산 중 하나는 인터넷을 통해 완료된 주문의 수입입니다.

시나리오 세부사항

회사의 데이터베이스에는 ORDER_ID, PRODUCT_ID, QUANTITY 및 TIME_ID 컬럼이 있는 인터넷 주문의 사실 테이블이 있습니다. PRODUCT_ID 컬럼에는 해당 주문에서 판매된 각 제품이 있으며, QUANTITY 컬럼은 주문에서 구매된 제품의 수량을 저장합니다. 하나 이상의 제품 주문에는 주문에서 판매된 고유한 제품의 수와 같은 수의 행 항목이 있습니다. 예를 들어, 표 33에서는 세 개의 주문을 표시하는데, 여기서 주문 1에는 세 개의 제품(Product As, Product O 및 Product G가 하나씩)이 있습니다

표 33. 사실 테이블 콘텐츠의 부분

ORDER_ID	PRODUCT_ID	QUANTITY
1	A	3
1	O	1
1	G	1
2	L	1
2	Q	2
3	P	5

DBA는 주문 계수 치수를 작성하여 ORDER_ID 컬럼의 각 고유 항목을 카운트할 수 있습니다. 주문 계수 치수는 DISTINCT 키워드를 사용하여 SQL 표현식에서 정의되며 집계에 대한 COUNT 함수는 모든 차원에 해당됩니다. 치수의 SQL 표현식은 구별 순서 목록을 작성하며, 이는 집계 작업 동안 카운트됩니다. 치수는 어떠한 합산 작업도 포함되지 않으므로 이를 비추가 치수라고 합니다.

비추가 치수는 카운트하려는 문자 데이터 또는 기타 데이터가 있을 때에도 유용합니다. 예를 들어, 비추가 치수를 사용하여 제품을 배송하려는 곳의 우편 번호의 수를 카운트할 수 있습니다.

이 예에서, DBA는 ORDER_ID 컬럼으로 직접 맵핑되는 주문 ID를 정의하려 했습니다. 그러나, 동일한 방법으로 ORDER_ID 컬럼을 사용하도록 선택할 수 있습니다. 디폴트 집계는 SQL 표현식에 컬럼이 사용되는지 치수가 사용되는지에 따라 다릅니다. 그러나 두 경우 모두 『치수 작성 단계』에서 설명한 대로 디폴트 집계를 COUNT 함수로 변경해야 합니다.

치수 작성 단계

다음 단계는 OLAP 센터 사실 등록 정보 창을 사용하여 기존 사실 오브젝트의 주문 계수 치수를 작성하는 방법을 설명합니다.

1. 사실 등록 정보 창을 열려면, 사실 오브젝트 트리에서 사실 오브젝트를 마우스 오른쪽 단추로 누르고 치수 편집을 누르십시오. 사실 등록 정보 창이 열립니다.
2. 치수 페이지에서, 계산된 치수 작성 누름 단추를 누르십시오 SQL 표현식 빌더 창이 열립니다.
3. SQL 표현식 빌더 창에서, 이름 필드에 ORDER COUNT를 입력하십시오.
4. Order count 표현식을 작성하려면, 치수 폴더를 데이터 목록에서 펼친 후 다음 단계를 완료하십시오
 - 함수 및 상수 필드에서, 기타를 선택하십시오 기타 함수 및 상수 목록에서, **DISTINCT** 키워드를 더블 클릭하십시오
 - 데이터 목록에서 **ORDER ID** 치수를 더블 클릭하십시오

84 페이지의 그림 20에서는 SQL 표현식 빌더 창에서 작성할 수 있는 Order Count 표현식을 표시합니다

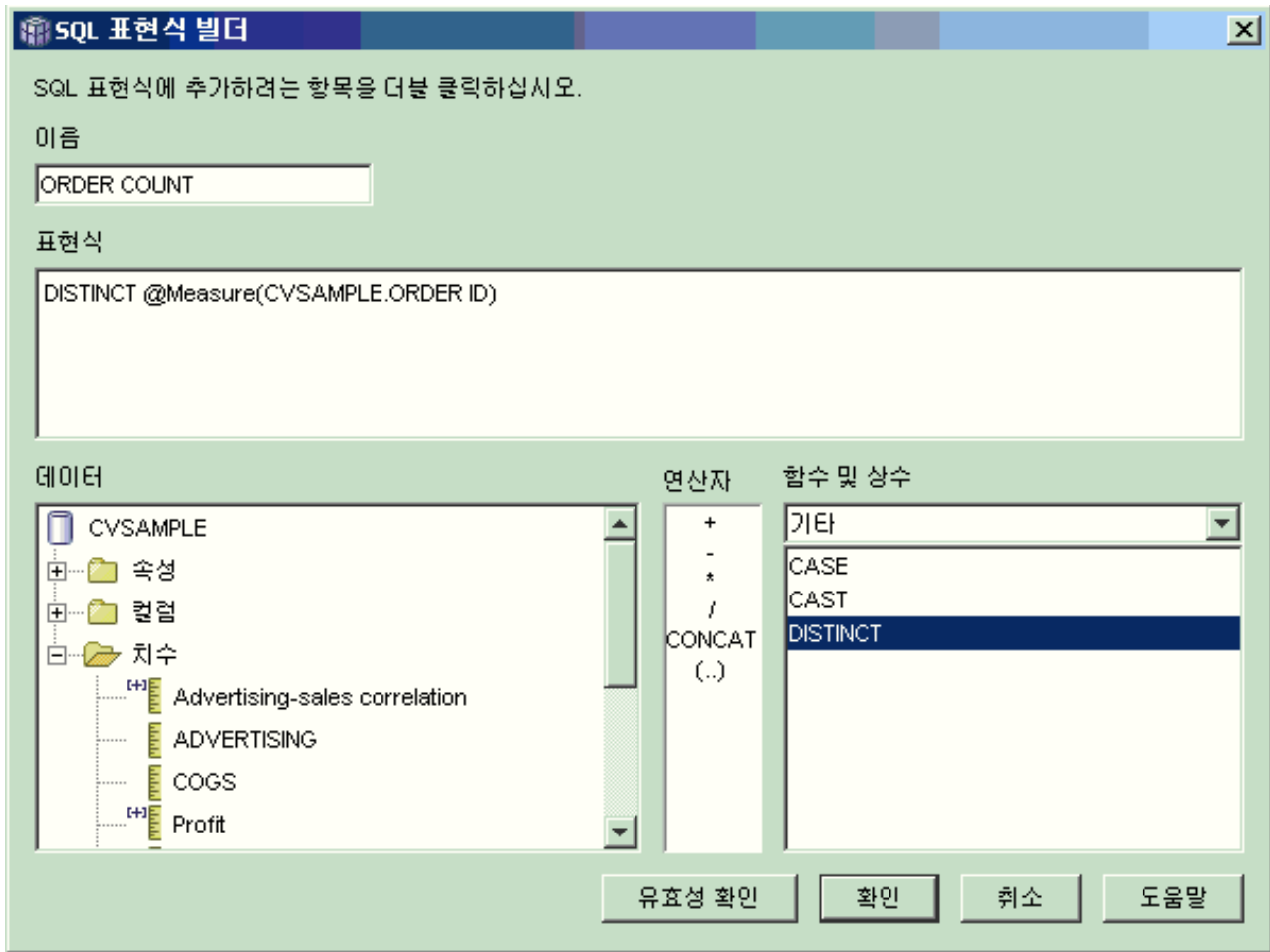


그림 20. SQL 표현식 빌더 창에서 order count 표현식을 완료하십시오.

5. 확인을 눌러 SQL 표현식 빌더 창을 닫으십시오.
6. 집계 페이지에서, 주문 계수 치수에 대한 집계를 누르고 **COUNT** 함수를 선택하십시오.
7. 확인을 눌러 사실 오브젝트에 변경사항을 저장하고 사실 등록 정보 창을 닫으십시오.

이제 사용자는 구별 주문 ID 행 값의 수를 카운트하는 치수가 있습니다. 기타 치수와 함께 이 치수를 결합하여 사용자의 데이터를 보다 유용하게 분석할 수 있습니다.

판매액 수치 순위

사무실 공급 상점 체인은 지난 몇 년에 걸쳐 급속하게 확장되었습니다. 비즈니스 실무자들은 성과가 가장 낮은 상점 일부를 닫아 비용을 줄이고 이익을 늘리는 것에 대해 고려 중입니다.

상점의 판매액 기록은 상점 폐쇄를 결정하는 중요한 인수입니다. 분석자는 판매 수치의 순위를 정하고 분석을 완료하기 위해 차원에 걸쳐 드릴 다운할 수 있기를 원합니다.

시나리오 세부사항

사무실 공급 상점의 데이터베이스에는 기타 컬럼 및 Sales 컬럼이 있는 사실 테이블이 있습니다. 그 데이터베이스에는 몇몇 차원 테이블도 있습니다. DBA는 RANK 함수를 사용하는 Sales 순위 치수를 작성할 수 있으며, 이 함수는 DB2 Universal Database(DB2 UDB)가 제공한 OLAP 함수입니다.

DB2 Cube Views는 DB2 UDB가 제공하는 다음의 OLAP 함수를 지원합니다.

RANK

행을 순서화하고 각 행에 순위를 지정합니다. 순위는 바로 앞 행의 숫자에 1을 더해 정의되며 이는 순서와는 구별됩니다. 중복 행 값을 가지고 있어서 두 개 이상의 행에 대한 상대적인 순서를 판별할 수 없을 경우, 동일한 순위 번호가 지정됩니다. 중복 행 값이 있을 경우 순위 지정 결과 숫자에 갭이 있습니다. 86 페이지의 표 34에서는 일련의 샘플 행 값에 대한 RANK 함수로부터의 순위 지정 결과의 예를 표시합니다.

RANK 함수의 일반적인 구문은 다음과 같습니다.

RANK () OVER (ORDER BY *sort-key-expression expression-order*)

여기서 *sort-key-expression*은 순위가 매겨지는 데이터 세트이며 *expression-order*는 키워드인 **ASC** 또는 **DESC**입니다 DB2 Cube Views는 *sort-key-expression*이 컬럼 또는 속성이 아닌 기존 치수이길 원합니다. 또한, DB2 Cube Views는 DB2 UDB가 이 함수와 함께 제공하는 PARTITION BY 절을 지원하지 않습니다. RANK 함수에 대한 자세한 정보는 DB2 UDB 정보 센터를 참조하십시오.

DENSERANK

행을 순서화하고 각 행에 순위를 지정합니다. 행의 순위는 바로 앞 행의 숫자에 1을 더한 것으로 정의됩니다. 따라서 순위 지정 결과는 순차적이며 순위 숫자에 갭이 없습니다. 86 페이지의 표 34에서는 일련의 샘플 행 값에 대한 DENSERANK 함수로부터의 순위 지정 결과의 예를 표시합니다.

DENSERANK 함수의 일반적인 구문은 다음과 같습니다.

DENSERANK () OVER (ORDER BY *sort-key-expression expression-order*)

여기서 *sort-key-expression*은 순위가 매겨지는 데이터 세트이며 *expression-order*는 키워드인 **ASC** 또는 **DESC**입니다 DB2 Cube Views는 *sort-key-expression*이 컬럼 또는 속성이 아닌 기존 치수이길 원합니다. 또한, DB2 Cube Views는 DB2 UDB가 이 함수와 함께 제공하는 PARTITION BY 절을 지원하지 않습니다. DENSERANK 함수에 대한 자세한 정보는 DB2 UDB 정보 센터를 참조하십시오.

ROWNUMBER

맨 첫 행을 1로 시작하여 순서화함으로써 행의 순차 행 번호를 계산합니다. ORDER BY 절이 지정되지 않은 경우, 행에 번호가 임의의 순서로 지정됩니다.

ROWNUMBER 함수의 일반적인 구문은 다음과 같습니다.

ROWNUMBER () OVER ([ORDER BY *sort-key-expression* *expression-order*])

여기서 *sort-key-expression*은 순위가 매겨지는 데이터 세트이며 *expression-order*는 키워드인 ASC 또는 DESC입니다 DB2 Cube Views는 컬럼 또는 속성이 아닌 기존 치수가 이 함수의 데이터 소스로 사용되기를 원합니다. 또한, DB2 Cube Views는 DB2 UDB가 이 함수와 함께 제공하는 PARTITION BY 절을 지원하지 않습니다. ROWNUMBER 함수에 대한 자세한 정보는 DB2 UDB 정보 센터를 참조하십시오.

이러한 OLAP 함수는 SQL 표현식 빌더 함수 및 상수 목록에 나열되지 않습니다.

표 34. RANK 및 DENSERANK 함수를 사용하는 일련의 값 샘플에 대한 순위 결과

행 값	순서화	RANK 함수로부터의 순위 결과	DENSERANK 함수로부터의 순위 결과
100	1	1	1
35	2	2	2
23	3	3	3
8	4	4	4
8	4	4	5
6	5	6	6

치수 작성 단계

다음 단계는 OLAP 센터 사실 등록 정보 창을 사용하여 기존 사실 오브젝트의 Sales Rank 치수를 작성하는 방법을 설명합니다.

1. 사실 등록 정보 창을 열려면, 사실 오브젝트 트리에서 사실 오브젝트를 마우스 오른쪽 단추로 누르고 치수 편집을 누르십시오. 사실 등록 정보 창이 열립니다.
2. 치수 페이지에서, 계산된 치수 작성을 눌러 Sales Rank 치수를 작성하십시오 SQL 표현식 빌더 창이 열립니다.
3. SQL 표현식 빌더 창에서, 이름 필드에 SALES RANK를 입력하십시오.
4. Sales Rank 표현식을 작성하려면, 다음 단계를 완료하십시오.
 - 표현식 필드에 다음과 같은 함수 구문을 입력하십시오. RANK () OVER (ORDER BY measure DESC).
 - 데이터 목록에서 치수 폴더를 펼치십시오
 - 표현식 필드의 함수 구문에 Measure라는 단어를 강조표시한 후, SALES 치수를 더블 클릭하여 표현식에 SALES 치수를 추가하십시오

최종 표현식이 그림 21에 표시됩니다

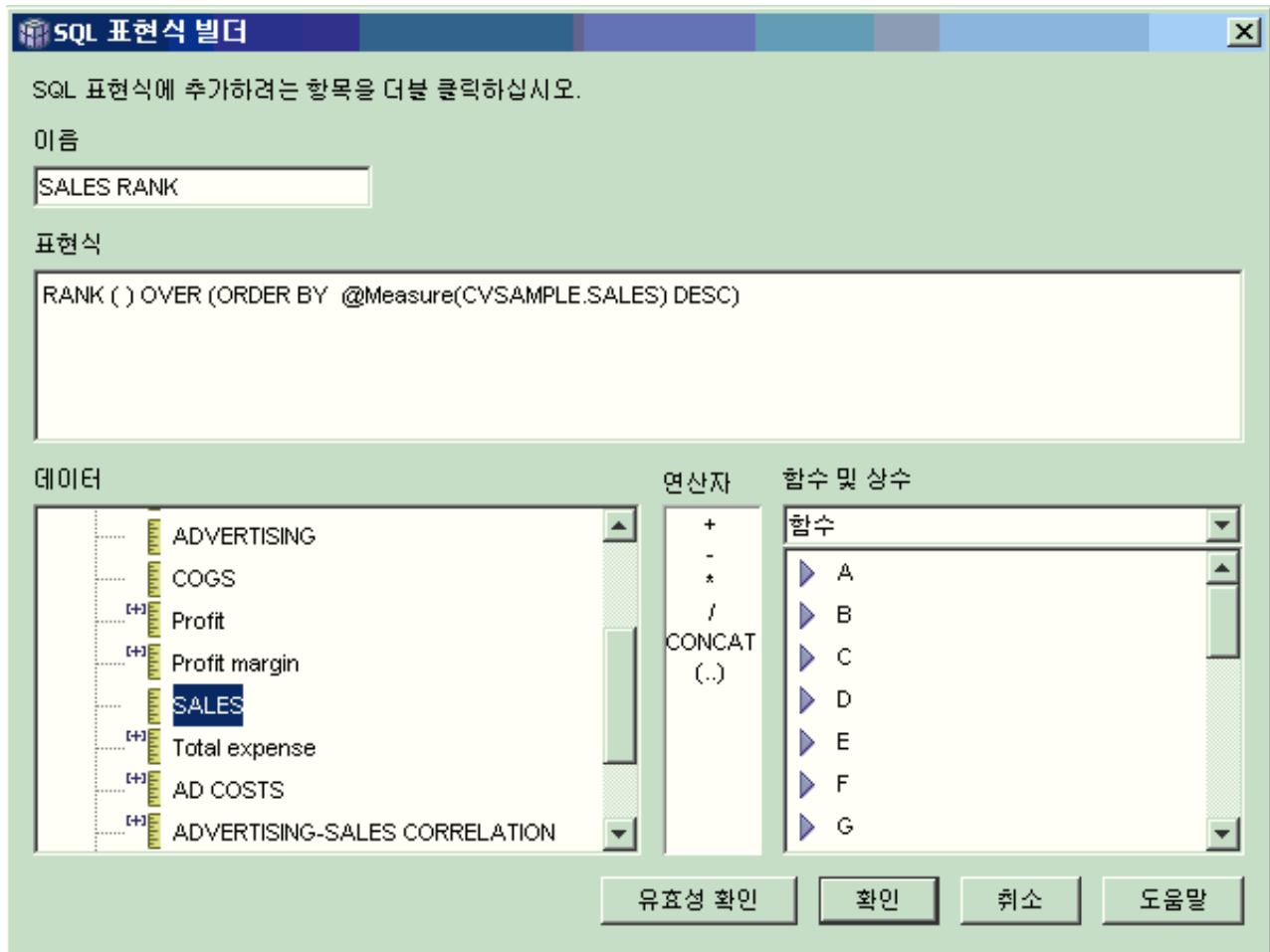


그림 21. SQL 표현식 빌더 창에서 Sales Rank 표현식을 완료하십시오

- 유효성 확인을 눌러 표현식이 유효한지 확인하십시오 확인을 눌러 SQL 표현식 빌더 창을 닫으십시오.

집계 페이지에서 디폴트 집계, 없음을 변경할 필요는 없습니다. 데이터 소스가 숫자이고 기존 치수만을 참조하므로, 없음 옵션은 Sales Rank 치수에 대한 기본값입니다.

치수의 표현식에 RANK 함수를 사용하여 Sales 컬럼을 내림차순으로 순서화할 경우, 분석자는 다른 차원에 걸쳐 드릴 다운하여 특정 제품 라인에 대해 작년에 가장 저조한 판매액을 기록한 상점을 판별할 수 있거나 데이터베이스에 저장된 다른 차원의 데이터와 관련된 상점을 판별할 수 있습니다.

사실 테이블에 저장된 시간 데이터를 사용하여 Time 차원 작성

소매 사업인 XYZ Retail은 DB2 Cube Views에서 Sales 트랜잭션 데이터를 모델링하여 데이터를 보다 효과적으로 분석할 수 있습니다. 그러나, 데이터의 트랜잭션 특성으로 인해, 사용 가능한 유일한 시간 정보는 각 트랜잭션과 연관된 날짜입니다.

Time 차원에서 모델화되는 시간 정보는 분기별 판매 경향 분석 및 각 주별 평균 재고 값 계산과 같은 여러 일반 계산에 컨텍스트를 추가하는 데 사용됩니다.

특정일에 트랜잭션이 없는 경우 데이터에 결함이 생겨 정확하게 집계하고 표시하는 데 문제가 발생할 수 있으므로 많은 DBA는 시간 데이터를 트랜잭션 시간소인 또는 날짜로 저장하지 않습니다. 일반적으로, Time 테이블에서 Time 데이터를 모델화하는 것이 좋습니다. 그러나 XYZ Retail의 DBA는 매일 최소 하나의 트랜잭션이 있을 것을 확신하고 데이터에 대한 현재 구조를 보존하기로 합니다.

시나리오 세부사항

XYZ Retail에는 Sales, Costs, Quantity Sold 및 Date를 포함하는 각 트랜잭션에 대한 측정 가능 정보가 있는 사실 테이블이 있습니다. 데이터베이스에는 Region Dimension 테이블 및 Product 차원 테이블도 있습니다. 문제점은 Time 데이터가 개별 차원 테이블에 저장되어 있지 않고 사실 테이블에 포함되어 있다는 것입니다. DBA는 사실 오브젝트의 Date 데이터에 근거하여 차원 오브젝트를 작성해야 합니다.

사실 테이블의 Date 데이터 단일 컬럼에 근거하여 Time 차원을 작성하려면 다음과 같은 두 가지 고유한 요구사항을 수행해야 합니다.

- 유효한 큐브 모델의 모든 차원 오브젝트는 사실 오브젝트로 조인되어야 하며 Time 차원 오브젝트 및 사실 오브젝트는 동일한 사실 테이블에 근거를 두어야 하므로, 사실 테이블을 자체적으로 조인하는 자체 조인을 사용하여 Time 차원 오브젝트를 사실 오브젝트로 조인해야 합니다.
- DBA는 Week, Month, Quarter 및 Year과 같은 의미있는 레벨로 Date 데이터를 집계하는 계산된 속성을 빌드해야 합니다.

자체 조인은 테이블을 자체적으로 조인하는 조인의 한 유형이며, 이 경우의 테이블은 사실 테이블입니다. 자체 조인은 하나 이상의 컬럼을 조인해야 하며 이로 인해 사실 테이블의 모든 행을 고유하게 식별할 수 있습니다. 기본 키를 선택하는 것이 가장 좋습니다. 그러나, 기본 키가 정의되어 있지 않은 경우, 사실 테이블을 차원 테이블과 조인하는 데 사용할 컬럼 세트를 선택하십시오. 큐브 모델을 최적화하려면, 기본 키가 정의되어 있어야 합니다. 조인 카디널리티(cardinality)는 1:1이어야 하며 조인 유형은 내부이어야 합니다.

그림 22에서는 사실 오브젝트, 사실 테이블에 근거한 차원 및 사실-차원 조인이 동일한 사실 테이블로 맵핑되는 방법을 표시합니다.

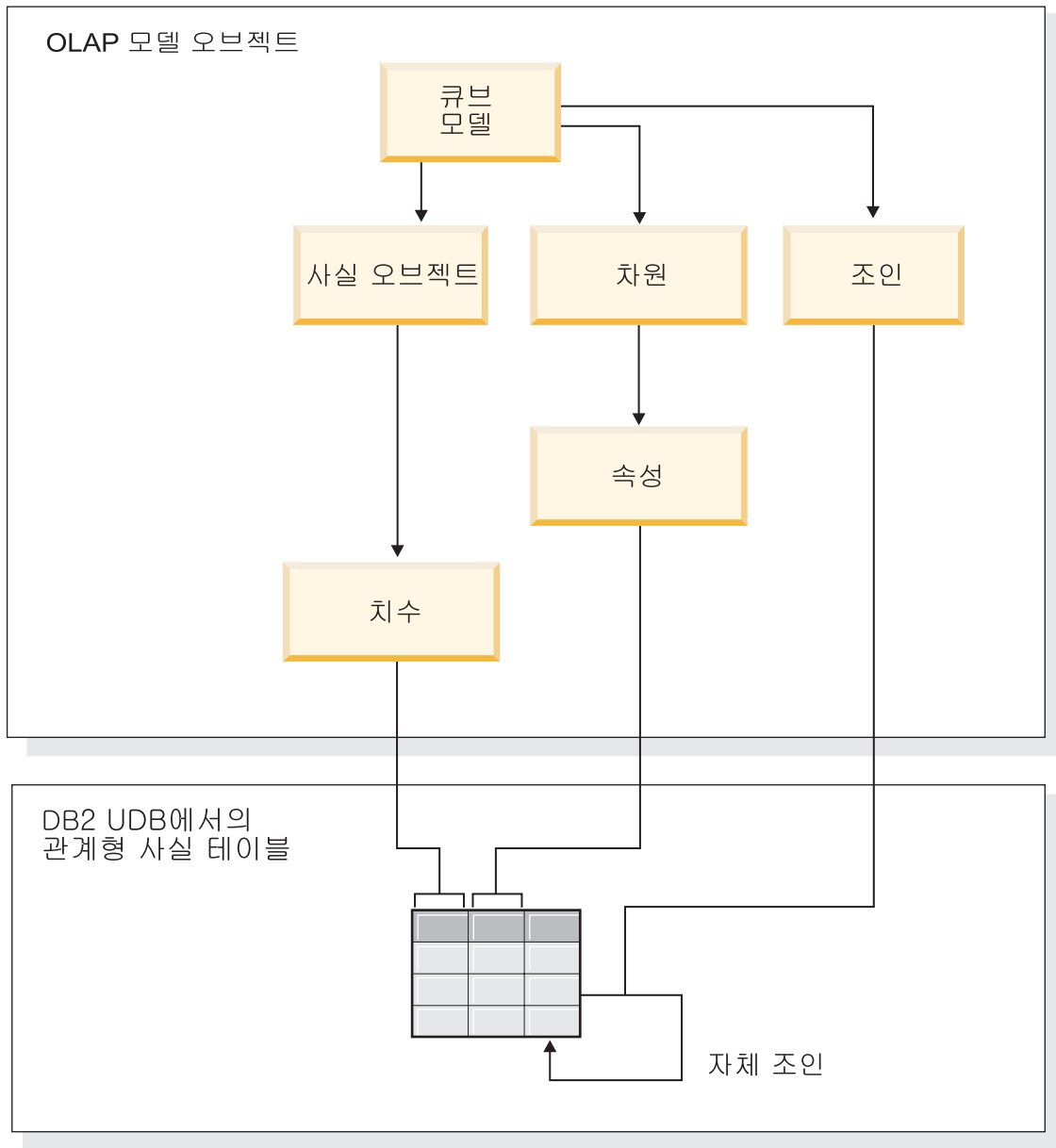


그림 22. 자체 조인이 테이블을 자체적으로 조인하는 방법

속성 및 차원 작성 단계

다음 단계는 사용자가 OLAP 센터 차원 마법사를 사용하여 사실 테이블에 근거하여 계산된 속성 및 Time 차원을 작성하는 방법을 설명합니다.

1. 차원 마법사를 열려면, OLAP 센터 오브젝트 트리의 큐브 모델을 마우스 오른쪽 단추로 누른 후 차원 작성을 누르십시오. 차원 마법사가 열립니다.

2. 이름 페이지에서, 이름 필드에 Time을 입력하십시오. 선택적으로 비즈니스 이름을 변경하고 주석을 입력할 수도 있습니다. 다음을 누르십시오
3. 큐브 모델의 사실 테이블을 선택하십시오. 다음을 누르십시오 차원에 하나의 테이블만 있으므로 차원 조인을 지정할 필요는 없습니다. 차원 조인 페이지에서, 다음을 누르십시오.
4. 차원 속성 페이지에서, 시간소인 컬럼을 선택하십시오
5. 선택사항: 시간소인 데이터를 보다 큰 청크(예:Month, Quarter 및 Year)로 집계하는 추가로 계산된 속성을 작성하십시오. 계산된 속성을 작성하려면, 계산된 속성 작성 누름 단추를 눌러 SQL 표현식 빌더를 열고, 소스 시간소인 컬럼을 Month, Quarter 및 Year으로 계산하는 각 속성의 표현식을 정의하십시오 계산된 각각의 속성을 정의한 후, 유효성 확인 누름 단추를 눌러 표현식이 유효한지 확인한 다음, 확인을 눌러 SQL 표현식 빌더를 닫고 차원 마법사로 리턴하십시오 선택한 후 다음을 누르고 원하는 속성 모두를 작성하십시오
6. 차원 입력 페이지에서, 시간을 선택하십시오 다음을 누르십시오
7. 사실 차원 조인 페이지에서, 조인 작성을 누르십시오. 열린 조인 마법사에서, 자체 조인을 작성하십시오. 이름을 입력하고 다음을 누르십시오. 기본 키와 같은 사실 테이블의 모든 행을 왼쪽 및 오른쪽 속성 둘 다에 대해 고유하게 정의하는 컬럼 또는 컬럼 세트를 선택하십시오. 한 번에 한 쌍을 선택하고 추가를 눌러 조인에 속성 쌍을 추가하십시오 내부 조인 유형 및 1:1 카디널리티(cardinality)를 선택하십시오. 필요한 속성 쌍을 추가한 후, 완료를 누르십시오 조인 마법사가 닫힙니다.
8. 사실 차원 조인 페이지에서, 완료를 누르십시오

큐브 모델에서 정의한 Time 차원을 사용하여, XYZ Retail은 해당 데이터 분석에 의미있는 새 레벨을 추가할 수 있습니다. 재고를 포함한 시간 관련 분석을 수행할 수 있습니다.

제 5 장 DB2 Cube Views 큐브 모델 최적화

이 섹션에서는 다음 주제를 설명합니다.

요약 테이블

DB2 Cube Views는 DB2 요약 테이블을 사용하여 큐브 모델에 발행된 쿼리 성능을 향상시킵니다. 요약 테이블은 특히 요약 데이터가 들어 있는 구체화된 쿼리 테이블(MQT)의 특수 유형입니다.

함수 종속성 및 제한조건이 있는 요약 테이블

최적화 어드바이저는 함수 종속성 및 제한조건과 같은 데이터 간 관계 정보를 이용하여 DB2 옵티마이저가 효과적으로 쿼리에 응답하는 데 필요한 집계된 치수 및 레벨 속성이 들어 있는 요약 테이블을 권장합니다.

최적화 프로세스에 대한 개요

DB2 Cube Views로 스타 스키마 또는 snowflake 스키마를 최적화하면 OLAP 유형의 SQL 쿼리 성능을 향상시킬 수 있습니다. 최적화 프로세스에는 최적화 어드바이저가 권장하는 요약 테이블의 작성, 구현 및 유지보수가 포함됩니다.

최적화 수행 시 메타데이터 설계 고려사항

레벨과 계층 구조, 치수, 큐브 및 최적화 슬라이스를 포함하는 메타데이터 오브젝트의 설계 방법은 최적화 어드바이저 마법사가 권장하는 요약 테이블에 영향을 미칩니다.

큐브의 최적화 슬라이스

최적화 슬라이스는 선택사항이지만 큐브 모델의 가장 중요한 영역에 초점을 맞춘 요약 테이블을 제공하도록 최적화 어드바이저를 안내하는 데 큰 도움이 됩니다.

최적화 슬라이스에서 쿼리 분석

최적화 슬라이스는 쿼리 성능을 향상시키는 데 유용한 도구지만 사용자 쿼리를 정확히 반영하는 경우에만 효과적입니다.

최적화 수행 시 제한조건 정의

제한조건은 최적화 어드바이저 및 DB2 옵티마이저에 유용한 정보를 제공합니다. 스타 스키마 또는 snowflake 스키마에 있는 외부 키 및 기본 키의 정보용 또는 강제실행 제한조건을 정의해야 합니다.

최적화 어드바이저의 매개변수

Optimization Advisor 마법사에 제공하는 각 매개변수 정보는 마법사가 권장하는 요약 테이블 및 사용자가 얻는 성능 향상에 영향을 미칩니다. 정확한 정보를 제공하고 비용과 성능 요구사항을 고려하여 신중히 결정하도록 하십시오.

큐브 모델 최적화

큐브 모델에 대해 다시 수행된 쿼리를 최적화함으로써 OLAP 스타일의 SQL 쿼리를 실행하는 제품의 성능을 향상시킬 수 있습니다.

요약 테이블을 작성하는 SQL 스크립트 예

최적화 어드바이저 마법사가 SQL 스크립트를 제공하여 권장 요약 테이블을 작성합니다. SQL 스크립트에는 하나 이상의 요약 테이블을 빌드하는 데 필요한 SQL 명령이 들어 있습니다.

쿼리 결과 테스트

최적화 어드바이저로 요약 테이블을 작성하기 전과 후에 DB2 Universal Database의 db2batch 벤치마크 도구를 사용하여 쿼리 성능 결과를 벤치마크할 수 있습니다.

요약 테이블 문제점 해결

요약 테이블을 작성한 후 쿼리 성능이 향상되지 않으면 DB2EXPLAIN 기능을 사용하여 쿼리 라우팅 문제점을 해결할 수 있습니다.

요약 테이블 유지보수

기본 테이블의 데이터를 변경하는 경우 요약 테이블을 갱신해야 합니다. 지금 새로 고침 또는 지연 새로 고침과 같은 서로 다른 두 가지 방법으로 요약 테이블을 갱신할 수 있습니다.

요약 테이블 삭제

DB2 Cube Views는 큐브 모델 삭제 시 연관된 요약 테이블을 삭제하지 않습니다. 기타 목적으로 요약 테이블을 사용하지 않는 경우 테이블을 삭제하여 디스크 여유 공간을 늘릴 수 있습니다.

요약 테이블

DB2 Cube Views는 DB2 요약 테이블을 사용하여 큐브 모델 및 큐브에 발행된 쿼리의 성능을 향상시킵니다. 요약 테이블은 특히 요약 데이터가 들어 있는 구체화된 쿼리 테이블(MQT)의 특수 유형입니다.

최적화 어드바이저는 데이터를 요약하는 MQT를 항상 권장하므로 DB2 Cube Views 문서에서 용어 요약 테이블을 사용하여 권장된 MQT를 설명합니다.

쿼리에서 비용이 많이 드는 계산 및 조인을 미리 완료하여 해당 데이터를 요약 테이블에 저장할 수 있습니다. 사전 계산된 데이터를 사용할 수 있는 쿼리를 실행할 때 DB2 UDB는 쿼리를 요약 테이블에 다시 라우트합니다. 쿼리가 사전 계산된 계산과 정확히 일치할 필요는 없습니다. SUM 및 COUNT와 같이 단순한 분석을 사용하는 경우 DB2 UDB는 사전 계산된 데이터에서 결과를 동적으로 집계할 수 있습니다. 하나의 요약 테이블에서 다수의 다른 쿼리를 만족시킬 수 있습니다. 요약 테이블을 사용하면 일반적으로

로 사용되는 데이터에 액세스하거나 하나 이상의 차원 또는 테이블에서 집계된 데이터와 관련된 쿼리의 성능을 많이 향상시킬 수 있습니다.

그림 23에서는 Sales 오브젝트 및 Time, Market, Product 차원이 있는 snowflake 스키마에 기반한 큐브 모델을 표시합니다. 사실 오브젝트에는 치수 및 속성이 있습니다. 각 차원에는 속성 세트가 있고 사실-차원 조인을 통해 사실 오브젝트에 조인됩니다.

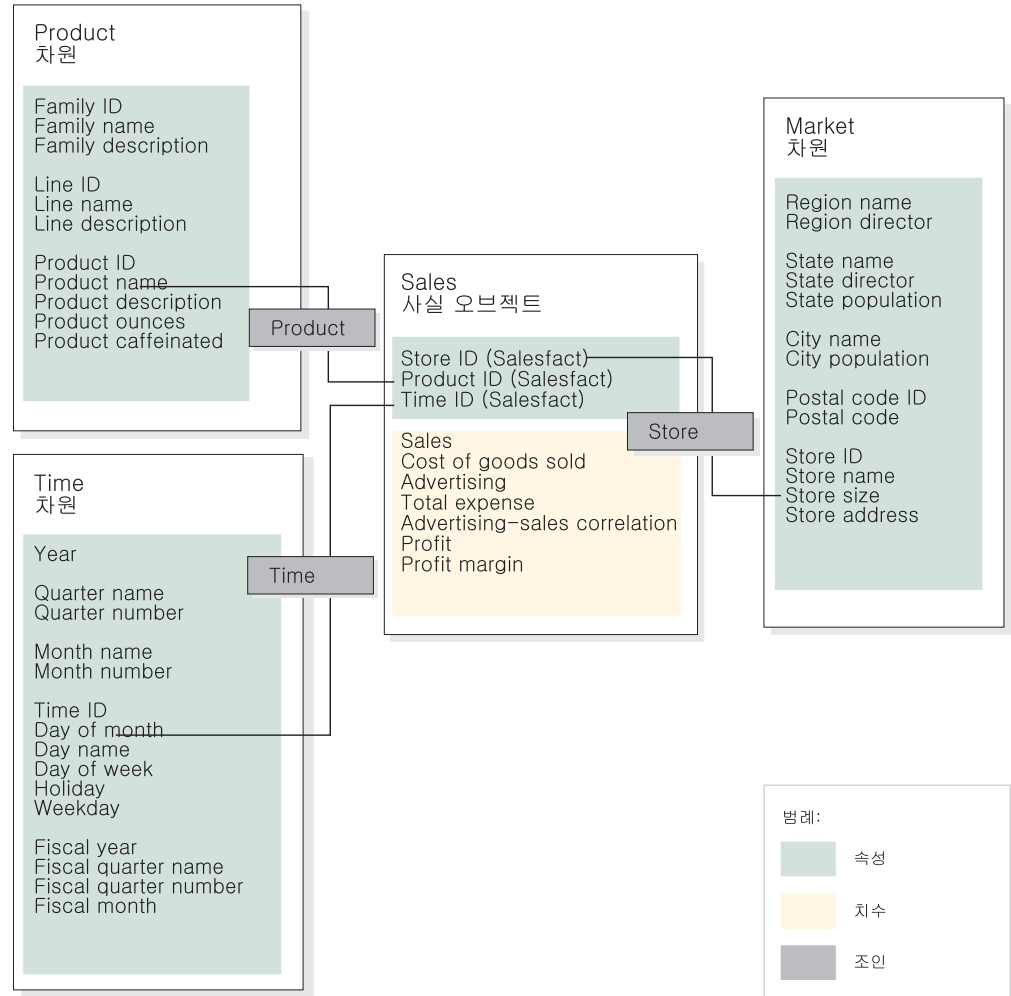


그림 23. 큐브 모델. Sales 사실오브젝트 및 Time, Product, Market 차원이 있는 큐브 모델

큐브 모델의 각 계층 구조가 94 페이지의 그림 24에 표시됩니다. 계층 구조의 맨 아래에서 굵은 선으로 연결된 강조표시된 상자는 기본 테이블에 실제로 존재하는 데이터를 표현합니다. Sales 데이터는 Day 레벨, Store 레벨 및 Product 레벨에 저장됩니다. 계층 구조에서 기본 레벨 상위의 데이터가 집계되어야 합니다. 특정 월에서 Sales 데이터의 기본 테이블을 쿼리하는 경우 DB2 UDB는 월별 판매액 수치를 리턴하도록 일별 Sales 데이터를 동적으로 추가해야 합니다. 예를 들어 다음 쿼리를 사용하여 2004년 각 지역에서 각 제품 라인에 대한 Sales 데이터를 월별로 볼 수 있습니다.

```

SELECT LINE_ID, REGION_NAME, MONTH_NUMBER, SUM(SALES)
FROM TIME, STORE, LOCATION, PRODUCT, LINE, SALESFACT
WHERE SALESFACT.STOREID = STORE.STOREID
      AND STORE.POSTALCODEID = LOCATION.POSTALCODEID
      AND SALESFACT.PRODUCTID = PRODUCT.PRODUCTID
      AND PRODUCT.LINEID = LINE.LINEID
      AND SALESFACT.TIMEID = TIME.TIMEID
      AND YEAR = '2004'
GROUP BY LINEID, MONTH_NUMBER;

```

Line-Region-Month 슬라이스를 연결하는 얇은 선(그림 24에 있음)은 쿼리가 액세스하는 슬라이스입니다. Line-Region-Month은 큐브 모델의 슬라이스이며 각 계층 구조에서 하나의 레벨을 포함합니다. 특정 슬라이스 또는 그 상위의 쿼리를 만족시키도록 요약 테이블을 정의할 수 있습니다. 쿼리에서 액세스하는 Line-Region-Month 슬라이스의 요약 테이블을 빌드할 수 있습니다. 일부 추가 집계 계층이 있는 요약 테이블을 통해 All Time, Year, Quarter, All Markets, All Products 및 Family를 포함하여 해당 슬라이스 또는 그 상위데이터에 액세스하는 기타 모든 쿼리를 만족시킬 수 있습니다. 하지만 해당 슬라이스 아래에 있는 더 자세한 데이터(예: Day 또는 City)를 쿼리하는 경우 DB2 UDB는 이러한 더 자세한 쿼리에서 요약 테이블을 사용할 수 없습니다.

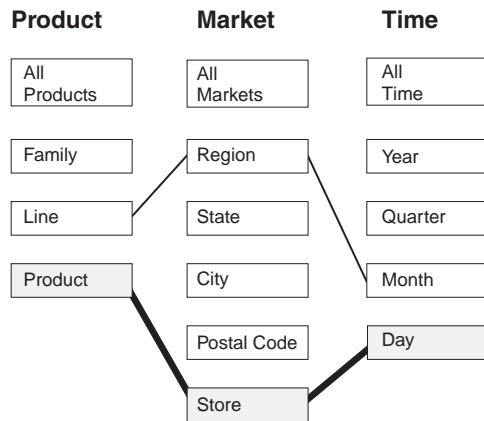


그림 24. 쿼리 슬라이스 Product, Market 및 Time 계층 구조. Line-Region-Month 슬라이스 및 기본 데이터가 Product-Store-Day 슬라이스에 존재함을 표시합니다.

95 페이지의 그림 25에서 점선은 Line-State-Month 슬라이스를 정의합니다. Line-State-Month 슬라이스에 빌드된 요약 테이블은 해당 슬라이스 또는 그 상위의 데이터에 액세스하는 쿼리를 만족시킬 수 있습니다. Line-State-Month 슬라이스에 빌드된 요약 테이블이 만족시킬 수 있는 모든 데이터는 위쪽의 강조표시된 상자 세트에 들어 있습니다.

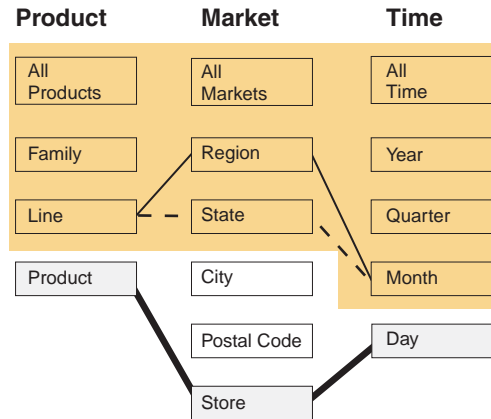


그림 25. 요약 테이블 슬라이스 Product, Market 및 Time 계층 구조. 강조표시된 데이터를 Line-State-Month 슬라이스에 빌드된 요약 테이블로 만족시킬 수 있습니다.

DB2 SQL 컴파일러의 재기록기(rewriter)에서 기존 요약 테이블에 관한 정보를 알고 있으므로 쿼리를 자동으로 재작성하여 기본 테이블 대신 요약 테이블에서 읽을 수 있습니다. 요약 테이블이 보통 기본 테이블보다 훨씬 더 작고 사전 집계된 데이터를 포함하기 때문에 재작성된 쿼리가 일반적으로 훨씬 더 빠릅니다. 사용자는 기본 테이블에 대해 계속 쿼리를 계속 작성합니다. DB2 UDB는 특정 쿼리에 대해 요약 테이블을 사용할 시기를 결정하며, 대신에 96 페이지의 그림 26에서처럼 요약 테이블에 액세스하는 사용자의 쿼리를 다시 작성합니다. 재작성된 쿼리는 사전 집계된 데이터가 들어 있는 요약 테이블에 액세스합니다. 종종 요약 테이블이 기본 테이블보다 훨씬 더 작으므로 훨씬 더 빠르고 기본 테이블과 동일한 결과를 리턴합니다.

DB2 EXPLAIN 기능을 사용하여 쿼리가 다시 라우팅되었는지 여부 및 가능한 경우에는 라우팅된 테이블을 확인할 수 있습니다.

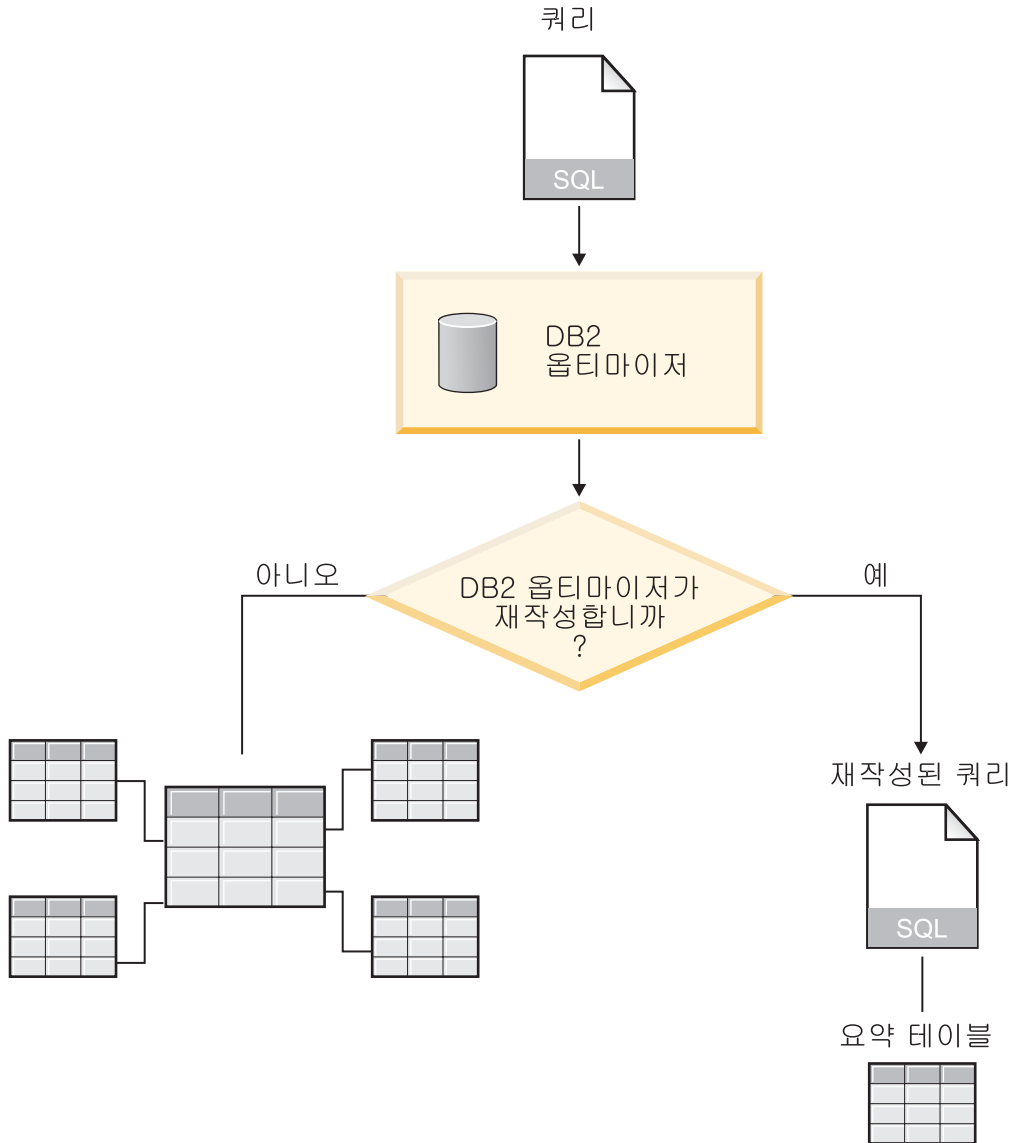


그림 26. 쿼리 재작성. 쿼리를 재작성하는 DB2 UDB 프로세스

2004년 각 지역에서 각 제품 라인에 대한 Sales 데이터를 월별로 보는 쿼리를 재작성하여 Line-Region-Month 슬라이스에 빌드된 요약 테이블을 사용할 수 있습니다. 원래 쿼리는 다음과 같습니다.

```

SELECT LINE_ID, REGION_NAME, MONTH_NUMBER, SUM(SALES)
FROM TIME, STORE, LOCATION, PRODUCT, LINE, SALESFACT
WHERE SALESFACT.STOREID = STORE.STOREID
  AND STORE.POSTALCODEID = LOCATION.POSTALCODEID
  AND SALESFACT.PRODUCTID = PRODUCT.PRODUCTID
  AND PRODUCT.LINEID = LINE.LINEID
  AND SALESFACT.TIMEID = TIME.TIMEID
  AND YEAR = '2004'
GROUP BY LINEID, MONTH_NUMBER;

```

재작성된 쿼리는 다음과 같습니다.

```

SELECT LINE_ID, REGION_NAME, MONTH_NUMBER, SUM(SALES)
FROM SUMMARYTABLE1
WHERE YEAR = '2004'
GROUP BY LINE_ID, REGION_NAME, MONTH_NUMBER;

```

데이터가 사전 집계되고 많은 조인 테이블이 사전 계산되어서 DB2 UDB가 대형 사실 테이블을 포함하는 여섯 개의 테이블 대신 하나의 소형 테이블에 액세스하므로 DB2 UDB에서 수행하는 재작성된 쿼리는 더 간단하고 더 빨라집니다. 요약 테이블을 통해 많은 혜택을 얻을 수 있습니다. 대형 사실 테이블이 있는 스키마에서 더 두드러집니다. 예를 들어 행이 10억 개인 사실 테이블은 행이 1백만 개인 요약 테이블로 사전 집계되므로 이 집계와 관련된 계산은 쿼리가 발행되는 횟수가 아닌 한 번만 수행됩니다. 대형 기본 테이블에 액세스하는 것보다 더 작은 요약 테이블에 1000번 액세스하는 것이 훨씬 더 빠릅니다.

이 예의 그림 27에서는 Line-State-Month 슬라이스의 요약 테이블을 표시합니다. 상점 보다는 주의 수가 더 적기 때문에 요약 테이블의 행을 줄이려면 DB2 UDB가 하위 레벨인 Store가 아닌 상위 레벨인 Region에서 데이터를 계산해야 합니다. 이러한 레벨의 데이터는 이미 집계되어 있으므로 Month 및 Line별 Sales 데이터를 리턴하는 데 DB2 UDB가 추가 계산을 수행하지 않아도 됩니다. 이 쿼리는 사전에 해당 쿼리에서 사용된 테이블을 조인하는 요약 테이블의 데이터로 완전히 만족되므로 쿼리가 발행될 때 조인을 수행하지 않아도 됩니다. 추가 복합 쿼리의 경우 성능이 더 많이 향상될 수 있습니다.

Region name	State name	Line ID	Year	Quarter number	Quarter name	Month number	Sales	Cost of goods	Advertising	Total expense	Profit
West	Idaho	054	2004	1	Qtr 1	2	9700	2500	700	3200	6500
East	Maine	102	2004	2	Qtr 2	5	3000	500	200	700	2300
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

그림 27. 요약 테이블. Line-Region-Month 슬라이스에 작성된 요약 테이블에 대한 예

중중 쿼리는 요약 테이블에 들어 있는 속성과 관련된 속성에 액세스할 수 있습니다. DB2 옵티마이저는 함수 종속성 및 제한조건을 사용하여 요약 테이블을 해당 차원 테이블과 동적으로 조인할 수 있습니다.

최적화 어드바이저가 요약 테이블을 권장하면 큐브 모델에 모든 치수가 포함됩니다. 이 예에서 Sales 사실오브젝트에는 Sales, Cost of goods, Advertising, Total expense 및 Profit과 같은 다섯 개의 치수만 들어 있으며 이들은 모두 요약 테이블에 포함됩니다. 큐브 모델에 치수를 50개 정의하면 50개 치수 모두가 요약 테이블에 포함됩니다. DB2

Cube Views가 레벨의 속성 간 함수 종속성을 정의하므로 최적화 어드바이저는 레벨에 정의된 모든 관련 속성을 요약 테이블에 포함시키지 않아도 됩니다.

합수 종속성 및 제한조건이 있는 요약 테이블

최적화 어드바이저는 함수 종속성 및 제한조건과 같은 데이터 간 관계 정보를 이용하여 DB2 옵티마이저가 효과적으로 쿼리에 응답하는 데 필요한 집계된 치수 및 레벨 속성이 들어 있는 요약 테이블을 권장합니다.

DB2 Cube Views는 가능한 경우 레벨의 속성 사이에 함수 종속성을 정의합니다. 레벨을 정의할 때 해당 레벨의 레벨 키 속성 및 기타 속성(디폴트 속성 및 관련 속성) 사이에 관계를 정의합니다. 관계는 레벨 키 속성이 레벨에 있는 기타 속성을 판별하는 데 함께 사용될 수 있음을 표시합니다. DB2 Cube Views는 속성 사이에 함수 종속성을 정의하여 레벨의 속성 사이의 관계를 자세히 기록합니다. DB2 UDB 및 DB2 Cube Views는 함수 종속성에서 정의한 관계를 사용하여 데이터의 최적화를 지능적으로 수행할 수 있습니다.

레벨 속성의 기본 데이터가 함수 종속성을 통해 설명된 방식으로 기능적으로 종속되어 있는지를 확인해야 합니다. DB2 UDB는 함수 종속성의 유효성을 확인하지 않습니다.

합수 종속성이 레벨 키 속성 및 레벨의 관련 속성 사이에 존재하면 최적화 어드바이저는 요약 테이블의 관련 속성 없이 레벨 키 속성을 포함할 수 있습니다. DB2 옵티마이저가 마지막 결과 세트를 작성하도록 쿼리가 발행될 때 요약 테이블을 차원 테이블과 조인하므로 레벨의 관련 속성과 관련된 쿼리는 요약 테이블로 계속 라우트될 수 있습니다.

예를 들어 92 페이지의 『요약 테이블』에 설명된 쿼리와 유사한 쿼리를 사용하여 2004년 각 지역에서 각 Product 라인에 대한 Sales 데이터를 월별로 볼 수 있습니다. 다음 쿼리는 Line ID 대신 Line name, Month number 대신 Month name으로 결과를 그룹화한다는 점에서 다릅니다. 결과 세트는 사용하기 쉽지만 데이터에 액세스하는 함수 종속성 및 제한조건에 따라 다릅니다.

```
SELECT LINE_NAME, REGION_NAME, MONTH_NAME, SUM(SALES)
FROM TIME, STORE, LOCATION, PRODUCT, LINE, SALESFACT
WHERE SALESFACT.STOREID = STORE.STOREID
  AND STORE.POSTALCODEID = LOCATION.POSTALCODEID
  AND SALESFACT.PRODUCTID = PRODUCT.PRODUCTID
  AND PRODUCT.LINEID = LINE.LINEID
  AND SALESFACT.TIMEID = TIME.TIMEID
  AND YEAR = '2004'
GROUP BY LINE_NAME, REGION_NAME, MONTH_NAME;
```

DB2 옵티마이저는 100 페이지의 그림 28에 표시된 대로 요약 테이블을 해당 차원 테이블에 조인하도록 쿼리를 다시 작성합니다.

Line 레벨에는 다음 속성이 들어 있습니다.

- 레벨 키 속성으로 Line ID
- 디폴트 속성으로 Line name
- 관련 속성으로 Line description

Line ID는 Line 테이블의 기본 키이므로 Line ID 컬럼에 제한조건이 있습니다. Line ID 컬럼에 제한조건이 있으므로 DB2 Cube Views는 이 레벨에 함수 종속성을 작성하지 않습니다. DB2 옵티마이저는 제한조건을 사용하여 요약 테이블을 Line 테이블에 조인한 후 쿼리 결과 세트의 Line name 데이터에 액세스합니다.

Month 레벨에는 다음 속성이 들어 있습니다.

- 레벨 키 속성으로 Year 및 Month 숫자
- 디폴트 속성으로 Month name

Month name이 기능적으로 Year 및 Month number의 조합에 종속되도록 지정하는 Month 레벨에 함수 종속성이 있습니다. Year 및 Month number 컬럼은 Time 테이블의 기본 키 또는 외부 키 부분이 아니므로 이 컬럼에는 제한조건이 없습니다. DB2 옵티마이저는 Month name 및 Year과 Month number의 조합 사이에 있는 함수 종속성을 사용하여 요약 테이블을 Time 테이블에 조인한 후 쿼리 결과 세트의 Month name 데이터에 액세스합니다.

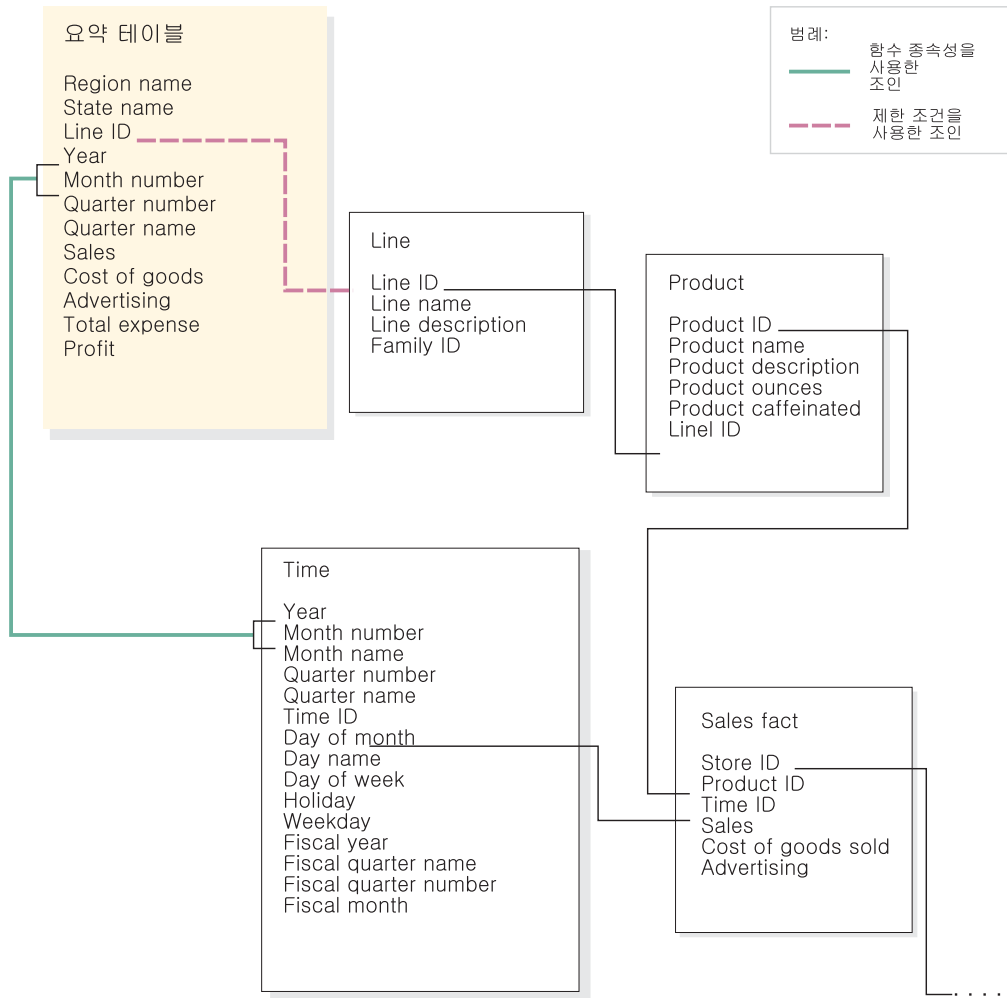


그림 28. 조인. DB2 옵티마이저는 쿼리가 발행될 때 요약 테이블을 해당 차원 테이블과 동적으로 조인합니다.

DB2 옵티마이저는 다음과 같이 쿼리를 다시 작성합니다.

```
SELECT Q4.LINE_NAME, Q4.REGION_NAME, Q4.MONTH_NAME, SUM(Q4.SALES)
FROM (
  SELECT DISTINCT Q1.LINE_NAME, Q3.REGION_NAME, Q2.MONTH_NAME,
    Q3.SALES, Q2.YEAR, Q2.MONTH_NUMBER
  FROM LINE AS Q1, TIME AS Q2, SUMMARYTABLE1 AS Q3
  WHERE (Q3.YEAR=2004)
    AND Q3.LINEID=Q1.LINEID)
    AND (Q3.MONTH_NUMBER=Q2.MONTH_NUMBER)
    AND (2004=Q2.YEAR)
) AS Q4
GROUP BY Q4.LINE_NAME, Q4.REGION_NAME, Q4.MONTH_NAME
```

결과 쿼리는 원래 쿼리에 있는 여섯 개의 테이블 대신 세 개의 테이블만 조인하며 다시 작성된 쿼리는 보통 크기가 크고 속도가 느린 대형 사실 테이블에 액세스하지 않아도 됩니다. 다시 작성된 쿼리의 실행 속도는 요약 테이블에 이미 사전 집계된 차수 데이터가 들어 있으므로 훨씬 더 빠릅니다.

최적화 프로세스에 대한 개요

DB2 Cube Views로 스타 스키마 또는 snowflake 스키마를 최적화하면 OLAP 유형의 SQL 쿼리 성능을 향상시킬 수 있습니다. 최적화 프로세스에는 최적화 어드바이저가 권장하는 요약 테이블의 작성, 구현 및 유지보수가 포함됩니다.

최적화 어드바이저는 요약 테이블을 권장하여 큐브 모델을 최적화하는 데 도움이 될 수 있습니다. DB2 UDB 요약 테이블에는 하나 이상의 테이블에서 사전 계산되었으며 쿼리에서 사용 가능한 결과가 포함되므로 쿼리 성능이 향상될 수 있습니다. 어려운 테이블 조인 및 복잡한 계산을 사전에 계산하여 요약 테이블에 저장할 수 있으므로 이러한 집계를 사용하는 향후의 쿼리는 훨씬 빨리 실행될 수 있습니다. 요약 테이블에 대한 정보는 92 페이지의 『요약 테이블』을 참조하십시오.

최적화 어드바이저는 메타데이터 및 마법사에 제공한 정보를 분석하고 적합한 요약 테이블을 제시합니다. 최적화 어드바이저의 실행 이후에는 권장 요약 테이블 세트를 작성할 수 있는 SQL 파일을 갖게 됩니다. 요약 테이블 작성을 위해 SQL을 실행하기 전에 SQL을 수정하는 옵션이 있습니다.

최적화 어드바이저의 실행은 최적화 프로세스의 한 단계일 뿐입니다. 최적화를 시작하기 전에 다음을 포함하는 일부 문제에 대해 고려하십시오(단, 이에 한정되지는 않음).

- 기본 테이블에서 DB2 제한조건의 효과적인 사용법
- 최적화 유효성 확인 규칙을 따르도록 큐브 모델을 정의하는 방법
- 최적화할 쿼리의 유형
- 제공하려는 스페이스 크기
- 요약 테이블에 들어 있는 데이터가 현재 상태가 되도록 요약 테이블을 유지보수하는 방법

최적화하기 전에, 기본 테이블에 대한 제한조건을 정의해야 합니다. 필수 제한조건 유형에 관한 정보는 118 페이지의 『최적화 수행 시 제한조건 정의』를 참조하십시오.

최적화 프로세스의 많은 부분은 반복적이며 성능상의 효과를 미세히 조정하고 유지하기 위해 반복 작업이 필요합니다. 102 페이지의 그림 29는 최적화 프로세스의 기본 단계들에 대한 개요를 표시합니다

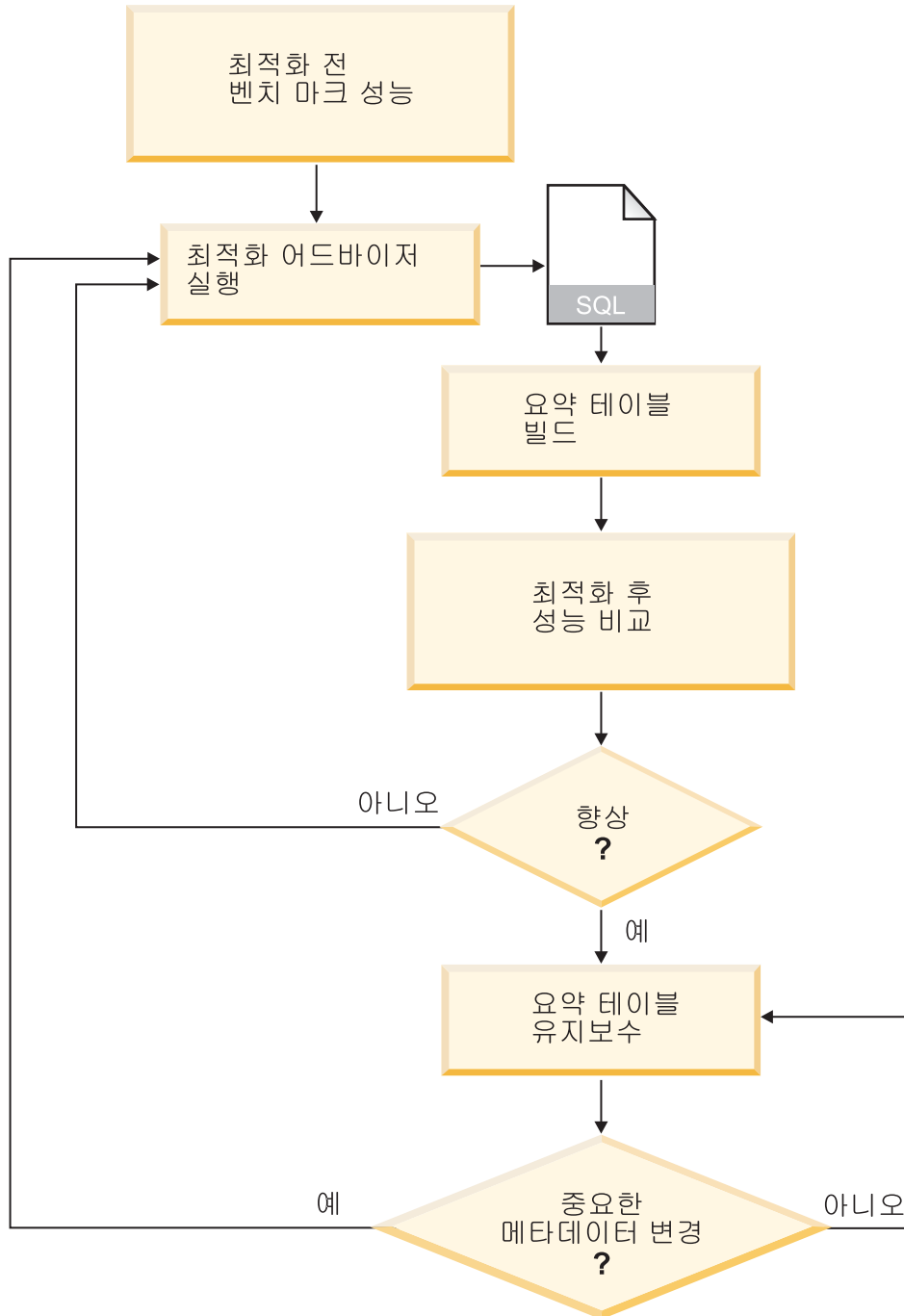


그림 29. 최적화 프로세스 최적화 프로세스의 기본 단계에 대한 개요

최적화 프로세스에는 다음의 일반 태스크가 포함됩니다.

- 치수 성능

DB2 Cube Views 최적화 어드바이저를 실행하기 전에, 일반 쿼리의 특정 세트에 대해 현재 성능을 측정해야 합니다. 성능 측정은 나중에 최적화의 성공 여부를 분석할 수 있는 벤치 마크를 제공하는 선택적 단계입니다. DB2 UDB에서 제공하는 db2batch 벤치마크 도구를 사용하여 벤치 마크를 작성할 수 있습니다. db2batch 사용에 대한

정보는 130 페이지의 『쿼리 결과 테스트』를 참조하십시오. 샘플 쿼리를 실행하여 성능 벤치 마크를 완료하지만 최적화 어드바이저가 메타데이터 기반이고 발행될 특정 쿼리를 알지 않고도 권장사항을 작성하기 때문에 샘플 쿼리가 필요하지 않습니다.

- **최적화 어드바이저 마법사 실행**

최적화하려는 쿼리 유형, 디스크 스페이스와 시간 제한, 갱신 메소드 및 테이블 스페이스 위치를 포함하여 마법사에 여러 중요한 매개변수를 제공합니다. 이러한 매개변수 선택 항목에 대한 정보는 124 페이지의 『최적화 어드바이저의 매개변수』를 참조하십시오. 최적화 어드바이저는 사용자가 제공하는 정보, 메타데이터, DB2 통계 및 사용자가 허용하는 모든 데이터 샘플링에 기반하여 권장사항을 작성합니다. 최적화 어드바이저는 사용자가 지정하는 매개변수를 고려하여 두 개의 SQL 파일을 생성합니다. SQL 파일에는 권장 요약 테이블 세트를 빌드하기 위한 SQL 명령이 들어 있습니다. 기타 SQL 파일에는 권장 요약 테이블을 갱신하기 위한 SQL 명령이 들어 있습니다.

- **요약 테이블 작성**

마법사 완료 후 요약 테이블을 즉시 작성하거나 조작을 일반적인 데이터베이스 유지보수 스케줄에 추가할 수 있습니다. 요약 테이블의 작성에는 많은 시간과 처리 자원이 필요할 수 있습니다. 요약 테이블이 작성된 이후 최적화된 큐브 모델에 대한 쿼리의 성능이 향상되었는지 확인하십시오. 최적화 이전에 실행한 동일 세트의 샘플 쿼리를 실행하고 성능 결과를 비교하십시오. 성능상 효과가 현저히 나타나지 않으면, 최적화 어드바이저를 다시 실행하고 디스크 스페이스 또는 시간 또는 둘 모두를 추가로 할당하거나 다른 설정값을 변경하십시오. 성능 결과 확인 및 분석 방법에 대한 정보는 130 페이지의 『쿼리 결과 테스트』를 참조하십시오.

- **요약 테이블 유지보수**

요약 테이블이 작성된 이후에는 데이터와의 동기화를 적절하게 유지하기 위해 주기적으로 테이블을 유지보수해야 합니다. 최적화 어드바이저를 실행할 때 지금 새로 고침 또는 지연 새로 고침 갱신 옵션을 선택합니다.

즉시 새로 고침 옵션

즉시 갱신 옵션을 선택하는 경우, DB2 UDB는 동기화된 요약 테이블 및 기본 테이블을 유지하며 기초가 되는 테이블이 변경될 때 요약 테이블을 증분식으로 갱신합니다. DB2 UDB는 SUM 및 COUNT와 같은 단순 집계계의 경우에 증분 유지보수를 지원합니다. 기타 집계계의 경우 최적화 어드바이저는 사용자가 선택한 새로 고침 옵션과 상관 없이 지연 새로 고침 옵션을 사용하는 요약 테이블을 권장합니다.

지연 새로 고침 옵션

지연 옵션을 선택하는 경우 요약 테이블을 재빌드하여 이를 갱신합니다. 요

약 테이블 갱신을 수행할 시기를 결정할 수 있습니다. 기본 테이블을 전반적으로 변경하는 경우 갱신을 지연하면 점진적인 갱신보다 더 효과적일 수 있습니다.

이러한 옵션 사이에서 선택하면, 유지보수에 할당할 수 있는 자원 및 데이터 동기화의 정확도 사이에서 적정선을 찾아야 합니다. 지금 및 지연 갱신 옵션에 관한 자세한 정보는 133 페이지의 『요약 테이블 유지보수』를 참조하십시오.

• 주기적 재평가

주기적으로 요약 테이블을 재평가하여 요구사항이 계속 충족되는지 확인해야 합니다. 큐브 모델을 추가하거나 갱신하여 메타데이터를 많이 변경하는 경우 최적화 어드바이저를 다시 실행하여 다음과 같이 새 요약 테이블 세트를 작성해야 할 수 있습니다.

- 새 차원 또는 치수와 같은 메타데이터 오브젝트를 추가하는 경우 새 오브젝트의 데이터에 액세스하는 쿼리는 기존 요약 테이블을 사용할 수 없습니다. 하지만 새 오브젝트를 사용하지 않는 쿼리는 요약 테이블을 계속 사용합니다.
- 메타데이터 오브젝트를 갱신하여 이전에 최적화되지 않은 데이터를 포함시키는 경우 갱신된 오브젝트에 액세스하는 쿼리는 요약 테이블을 사용할 수 없습니다.
- 하나 이상의 오브젝트를 삭제하는 경우 요약 테이블의 효과는 변경되지 않지만 디스크 공간을 더 이상 사용되지 않는 집계에 사용하고 있습니다.

메타데이터를 만히 변경하는 경우 이외에 정기적으로 수행되는 쿼리 유형이 최적화하려는 유형이 아닌 유형으로 변경되는 경우 마법사를 다시 실행해야 할 수도 있습니다.

마법사를 실행하여 새 테이블을 작성할 때마다 벤치 마크 작성 및 요약 테이블의 성능 분석을 포함하여 전체 최적화 프로세스를 다시 완료해야 합니다.

별도의 다른 목적으로 사용되지 않는 경우, 큐브 모델을 삭제할 때 연관된 요약 테이블도 함께 삭제할 수 있습니다. DB2 Cube Views는 연관된 큐브 모델을 삭제해도 요약 테이블을 삭제하지는 않습니다. 요약 테이블 삭제 방법에 대한 정보는 134 페이지의 『요약 테이블 삭제』를 참조하십시오.

최적화 수행 시 메타데이터 설계 고려사항

레벨과 계층 구조, 치수, 큐브 및 최적화 슬라이스를 포함하는 메타데이터 오브젝트의 설계 방법은 최적화 어드바이저 마법사가 권장하는 요약 테이블에 영향을 미칩니다.

일반적으로 사용자는 데이터 구조에 따라 사실 오브젝트, 차원 및 조인의 기본 구조를 정의하려 합니다. 큐브 모델에 있는 이러한 상위 레벨 오브젝트의 기본 구조에는 선택 항목이 거의 없으므로 이러한 오브젝트를 변경하여 권장 요약 테이블의 성능을 향상시

키기는 어렵습니다. 하지만 레벨, 계층 구조, 치수, 큐브 및 최적화 슬라이스를 정의하는 방법을 선택하면 추가 유연성을 얻을 수 있습니다.

레벨 및 계층 구조

가능하면 30 페이지의 『레벨』에 설명된 최적의 모델링 메소드를 사용하여 레벨을 정의하십시오. 최적의 모델링 메소드를 사용하면 권장 요약 테이블이 사용하는 디스크 스페이스 크기 및 권장 요약 테이블을 새로 고치는 데 사용하는 임시 스페이스 크기를 줄일 수 있습니다.

치수

제한된 디스크 스페이스가 문제가 되는 경우 큐브에 정의한 치수가 많으면 요약 테이블 크기가 커지므로 큐브에서 중요한 치수만 포함하고 정기적으로 사용되지 않거나 비즈니스 값이 없는 모든 치수를 삭제하도록 선택할 수 있습니다.

큐브에 포함시킬 치수 유형(분산 또는 비분산)이 다음과 같이 최적화에 영향을 줄 수도 있습니다.

- 분산 치수는 한 레벨에서 다음 레벨로 항상 집계될 수 있습니다. 예를 들어 분기의 SUM(판매액)은 월별 Sales 데이터를 집계하여 계산될 수 있습니다.
- 표준 편차와 같은 비분산 치수는 항상 기본 데이터에서 직접 계산되어야 하므로 한 레벨에서 다음 레벨로 집계될 수 없습니다.

일반적으로 큐브 모델에 분산 치수만 들어 있는 경우 최적화 어드바이저 및 DB2 옵티마이저에는 더 많은 유연성 및 옵션이 있습니다. 비분산 치수가 들어 있는 큐브 모델을 최적화할 수는 있지만 큐브에서 불필요한 비분산 치수를 생략하면 더 나은 최적화 결과를 얻을 수 있습니다.

큐브

비즈니스의 요구와 일치하는 큐브를 설계하십시오. 큐브 모델 데이터의 중요한 영역에 초점을 맞춘 큐브를 빌드하면 최적화 수행 시 큐브가 보다 효과적으로 사용됩니다.

사용자가 자주 발행하는 하나 이상의 쿼리 특성을 알고 있는 경우 최적화 슬라이스의 해당 정보를 최적화 어드바이저에 지정할 수 있습니다. 최적화 어드바이저는 지정된 큐브 영역에 발행되는 쿼리의 성능을 향상시키는 요약 테이블을 권장합니다. 최적화 슬라이스는 각 큐브 차원에서 특정 레벨 세트 또는 임의 레벨을 지정하여 정의되고 쿼리 유형은 드릴 다운, 보고서, MOLAP 추출, 하이브리드 추출 또는 드릴 쓰루와 같은 슬라이스로 예상됩니다.

OLAP 센터에서 드릴 다운과 같은 하나의 쿼리 유형에서 일반적으로 하나의 큐브를 사용하도록 지정할 수 있습니다. 큐브에 하나의 쿼리 유형을 지정하는 경우 OLAP 센터는 모든 큐브 차원에 사용자가 지정한 유형 및 임의 옵션으로

최적화 슬라이스를 정의합니다. 최적화 슬라이스에 관한 자세한 정보는 107 페이지의 『큐브의 최적화 슬라이스』를 참조하십시오.

다음 목록에서는 OLAP 센터의 한 큐브에 지정할 수 있는 쿼리 유형을 설명합니다.

드릴 다운 쿼리

드릴 다운 쿼리는 보통 큐브 모델의 맨 위에 초점을 맞춘 데이터 서브 세트에 액세스합니다. 쿼리는 큐브 모델의 모든 레벨로 이동할 수 있습니다. 사용자가 한 차원으로 깊이 드릴하면 사용자는 보통 다른 차원의 훨씬 더 높은 위치에 머무릅니다. 드릴 다운 쿼리로 최적화하는 경우 큐브 모델의 상위 레벨에 머무르는 쿼리가 가장 많이 향상됩니다. 관계형 OLAP(ROLAP) 스프레드시트 응용프로그램은 보통 드릴 다운 쿼리를 수행하는 데 사용됩니다. 예를 들어 스프레드시트 응용프로그램 사용자는 2004년의 모든 지역 및 모든 제품의 수익에 액세스하여 시작할 수 있습니다. 그 다음 모든 지역의 분기별 수익 및 각 국가를 쿼리하여 더 심도 있게 데이터를 이해할 수 있습니다.

결과가 처리될 때까지 대기해야 하는 사용자가 실시간으로 이러한 유형의 쿼리를 발행하므로 일반적으로 해당 쿼리 유형의 성능이 매우 중요합니다.

보고서 쿼리

보고서 쿼리도 마찬가지로 큐브 모델의 모든 부분에 액세스할 수 있습니다. 보고서 쿼리는 종종 일괄처리로 발행됩니다. 사용자가 각 개별 쿼리에 대한 즉각적인 응답을 기다릴 가능성이 적으므로 보통 드릴 다운 쿼리만큼 보고서 쿼리의 성능은 중요하지 않습니다.

MOLAP 추출 쿼리

MOLAP 추출 쿼리는 기본 큐브 레벨 또는 큐브에 정의된 최적화 슬라이스에 액세스합니다. 큐브는 MOLAP 데이터 스토어로 데이터를 로드하는 데 사용됩니다. 큐브 최적화 슬라이스는 후속 처리를 위해 데이터를 MOLAP 응용프로그램에 로드하는 추출 슬라이스로 논리적으로 맵핑됩니다.

OLAP 센터의 큐브에 MOLAP 추출 유형을 지정하는 경우 또는 모든 큐브 차원에서 지정된 임의 옵션으로 최적화 슬라이스를 지정하는 경우 최적화 어드바이저는 기본 큐브 레벨에서 추출되도록 데이터를 최적화하는 요약 테이블을 작성합니다. 기본 큐브 레벨이 사용자가 추출한 슬라이스로 맵핑되는지 확인하십시오.

고급 설정

OLAP 센터의 큐브에 고급 설정을 지정하는 경우 자주 쿼리되는 특정 큐브 영역에 최적화 슬라이스를 지정할 수 있습니다.

사용자가 자주 발행되는 하나 이상의 쿼리 특성을 알고 있는 경우 최적화 슬라이스의 해당 정보를 최적화 어드바이저에 지정할 수 있습니다. 최적화 어드바이저는 요약 테이블을 권장할 때 해당 슬라이스를 고려합니다. 최적화 슬라이스는 각 큐브 차원에서 특정 레벨 세트 또는 임의 레벨을 지정하여 정의되고 해당 슬라이스에서 쿼리 유형은 드릴 다운, 보고서, MOLAP 추출, 하이브리드 추출 또는 드릴 쓰루로 예상됩니다. 최적화 슬라이스에 관한 정보는 『큐브의 최적화 슬라이스』를 참조하십시오.

큐브의 최적화 슬라이스

최적화 슬라이스는 선택사항이지만 큐브 모델의 가장 중요한 영역에 초점을 맞춘 요약 테이블을 제공하도록 최적화 어드바이저를 안내하는 데 큰 도움이 됩니다.

큐브에는 종종 다수의 큐브 차원이 있을 수 있습니다. 하나 이상의 최적화 슬라이스를 지정하면 가장 많이 쿼리되는 큐브 지역을 지정할 수 있습니다. 최적화 슬라이스는 하나 이상의 레벨 세트에서 정의되며 해당 슬라이스에서 쿼리 유형은 및 드릴 다운, 보고서, MOLAP 추출, 하이브리드 추출 또는 드릴 쓰루로 예상됩니다. 최적화 슬라이스를 정의할 때 어떤 유형의 쿼리에서도 혜택을 얻을 수 있지만 보고서 쿼리로 가장 많은 혜택을 얻을 수 있습니다.

최적화 슬라이스를 작성할 때 큐브 차원 당 하나의 옵션을 지정해야 합니다. 각 큐브 차원에 옵션을 지정할 때 다음 지침을 사용하십시오.

- 지정된 레벨이 중요하거나 자주 쿼리됨을 알고 있는 경우 Time 큐브 차원의 Month 와 같이 큐브 차원에 특정 레벨을 지정하십시오.
- 큐브 차원의 가장 높은 집계가 중요하거나 자주 쿼리되는 경우 큐브 차원에 모두를 지정하십시오.
- 해당 큐브 차원의 다른 레벨보다 훨씬 더 중요한 레벨이 없거나 해당 큐브 차원에서 다수의 레벨이 쿼리되거나 해당 큐브 차원의 각 레벨이 쿼리되는 빈도를 알지 못할 경우 큐브 차원에 임의를 지정하십시오.

예를 들어 10개의 큐브 차원이 있는 하나의 큐브에서 두 개의 큐브 차원에만 중요한 레벨이 있을 수 있으므로 두 개의 해당 큐브 차원에 특정 레벨을 지정하고 나머지 모든 큐브 차원에 임의를 지정할 수 있습니다.

다음 섹션에서는 각 쿼리 유형의 최적화 슬라이스 예 및 최적화 어드바이저가 권장할 수 있는 요약 테이블을 설명합니다.

드릴 다운 최적화 슬라이스

드릴 다운 최적화 슬라이스는 사용자가 일반적으로 큐브 차원에 정의된 레벨에 드릴 다운함을 의미합니다. 따라서 최적화 어드바이저는 하나 이상의 권장 요약 테이블에 이러한 레벨을 포함시켜야 합니다. 유형이 드릴 다운이므로 최적화 어드바이저는 큐브 차원 중 일부에서 얇은 심도의 쿼리 및 깊은 심도의 쿼리를 모두 최적화할 수 있습니다.

권장사항: 레벨이 특히 중요하다는 것을 알고 있을 때에만 큐브 차원에 특정 레벨을 지정하십시오. 일반적으로 대부분의 큐브 차원에 임의 옵션을 지정하고 해당 레벨이 집중적인 쿼리 활동과 관련된 경우에만 특정 레벨을 선택하십시오.

109 페이지의 그림 30에서는 최적화 어드바이저가 요약 테이블을 권장할 수 있는 슬라이스 및 드릴 다운 최적화 슬라이스에 대한 예를 표시합니다. 최적화 슬라이스는 Procut 큐브 차원 및 Market 큐브 차원의 임의 레벨과 Time 큐브 차원의 Month 레벨에서 정의됩니다. 이 최적화 슬라이스는 사용자가 일반적으로 Month 레벨에 드릴 다운되지만 Product 또는 Market 큐브 차원에 특정 드릴 다운 패턴을 가지고 있지 않음을 의미합니다.

가능한 요약 테이블 권장사항에는 두 개의 집계 레벨이 들어 있습니다. 한 집계 레벨은 Line-State-Month 슬라이스에서 정의되며 다른 집계 레벨은 Family-Region-Month 슬라이스에서 정의됩니다. 두 집계 레벨은 모두 Time 차원의 Month 레벨에서 집계를 포함하지만 두 개의 기타 차원에서는 다른 집계 레벨을 제공합니다. 얇은 심도의 드릴 다운 쿼리를 바로 만족시킬 수 있어서 DB2 UDB가 집계할 필요가 없기 때문에 더 높은 슬라이스의 성능은 해당 쿼리에서 더 많이 향상됩니다. 더 낮은 집계 레벨의 경우 깊은 심도의 드릴 다운 쿼리에서 성능이 향상됩니다.

드릴 다운 슬라이스



가능한 권장사항

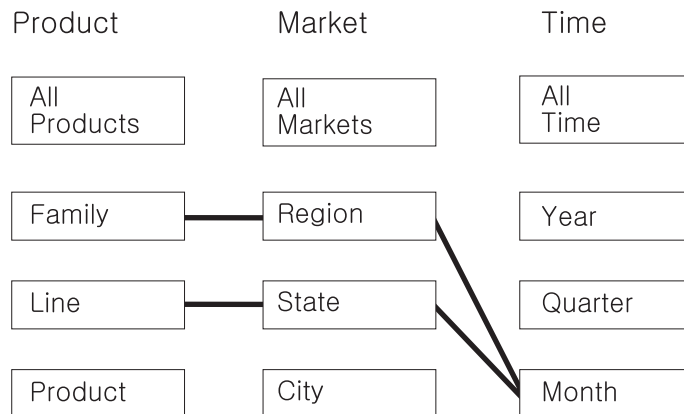


그림 30. 드릴 다운. 드릴 다운 최적화 슬라이스 및 가능한 요약 테이블 권장사항

보고서 최적화 슬라이스

보고서 최적화 슬라이스는 사용자가 일반적으로 큐브 차원에 정의된 레벨에서 보고서를 작성함을 의미합니다. 따라서 최적화 어드바이저는 하나 이상의 권장 요약 테이블에 이러한 레벨을 포함시켜야 합니다.

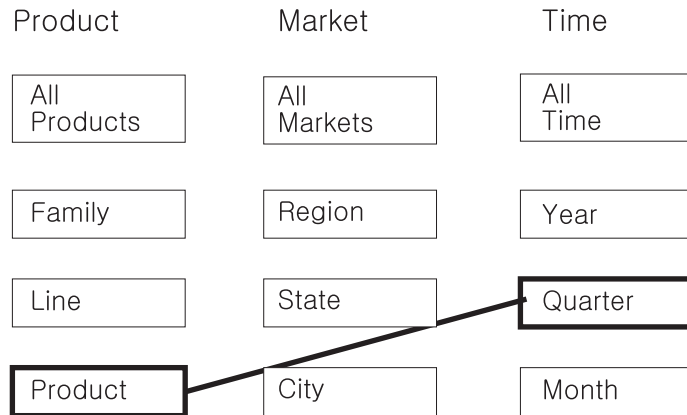
권장사항: 레벨이 특히 중요하다라는 것을 알고 있을 때에만 큐브 차원에 특정 레벨을 지정하십시오. 일반적으로 대부분의 큐브 차원에 임의 옵션을 지정하고 해당 레벨이 집중적인 쿼리 활동과 관련된 경우에만 특정 레벨을 선택하십시오.

111 페이지의 그림 31에서는 최적화 어드바이저가 요약 테이블을 권장할 수 있는 슬라이스 및 보고서 최적화 슬라이스에 대한 예를 표시합니다.

최적화 슬라이스는 Product 큐브 차원에서 Product 레벨, Market 큐브 차원에서 임의 레벨 및 Time 큐브 차원에서 Quarter 레벨로 정의됩니다. 이 최적화 슬라이스는 Product 및 Quarter 레벨을 포함하지만 Market 큐브 차원을 포함하지 않거나 Market 큐브 차원에 여러 레벨을 포함시킬 수 있거나 사용자가 포함시킨 Market 영역의 레벨을 알지 못하는 보고서를 사용자가 작성함을 의미합니다. 예를 들어 사용자는 마지막 네 분기에서 각 제품군에 Sales 데이터를 표시하는 보고서를 자주 작성할 수 있지만 보고서는 Region, State 또는 City 별로 Sales 데이터를 다양하게 표시합니다.

가능한 요약 테이블 권장사항에는 두 개의 집계 레벨이 들어 있습니다. 한 집계 레벨은 Product-City-Quarter 슬라이스에서 정의되며 다른 집계 레벨은 Product-Region-Quarter 슬라이스에서 정의됩니다. 두 집계 레벨은 모두 해당 슬라이스에 지정된 Product 및 Quarter 레벨을 포함합니다. 데이터 샘플링 및 기타 메타데이터에 기반하여 최적화 어드바이저는 두 개의 집계 레벨(하나는 Market 큐브 차원의 City 레벨을 포함하며 다른 하나는 Market 큐브 차원의 Region 레벨을 포함함)을 다루는 요약 테이블을 작성하려고 합니다.

보고서 슬라이스



가능한 권장사항

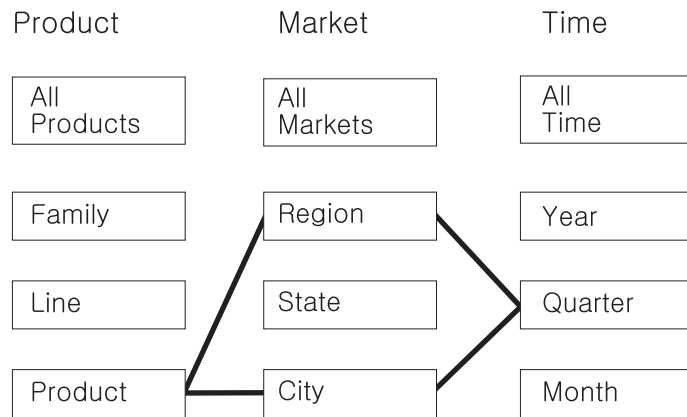


그림 31. 보고서. 보고서 최적화 슬라이스 및 가능한 요약 테이블 권장사항

MOLAP 추출 최적화 슬라이스

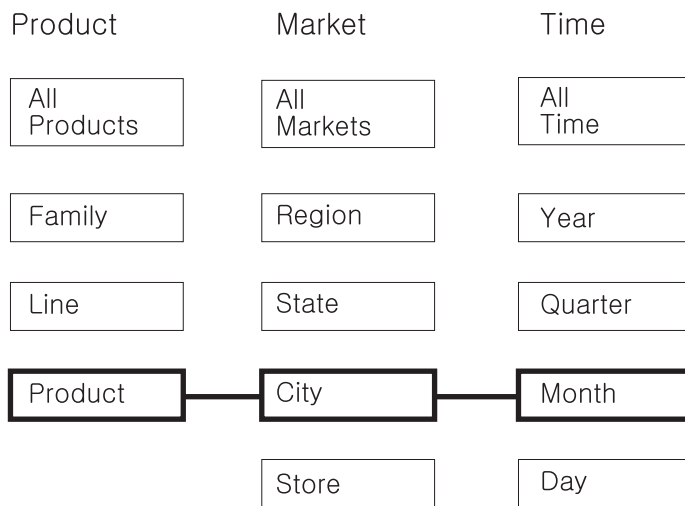
MOLAP 추출 최적화 슬라이스는 사용자가 일반적으로 지정된 슬라이스의 데이터를 벤더 제품의 MOLAP 큐브로 추출함을 의미합니다. 따라서 최적화 어드바이저는 지정된 슬라이스로 발행된 쿼리를 신속하게 처리하도록 요약 테이블을 권장해야 합니다.

권장사항: 최적화 슬라이스가 MOLAP 큐브에 추출되는 데이터의 레벨과 일치하도록 각 큐브 차원에 특정 레벨을 지정하십시오.

큐브당 하나의 MOLAP 추출 최적화 슬라이스만 정의할 수 있습니다. MOLAP 추출 최적화 슬라이스가 들어 있는 큐브에 하이브리드 추출 최적화 슬라이스를 정의할 수 있습니다.

그림 32에서는 최적화 어드바이저가 요약 테이블을 권장할 수 있는 슬라이스 및 MOLAP 추출 최적화 슬라이스에 대한 예를 표시합니다. 최적화 슬라이스는 Product 큐브 차원에서 Product 레벨, Market 큐브 차원에서 City 레벨 및 Time 큐브 차원에서 Month 레벨로 정의됩니다. 이 최적화 슬라이스는 Product-City-Month 레벨의 데이터를 MOLAP 큐브에 추출함을 의미합니다. 가능한 권장사항에는 Product-City-Month 최적화 슬라이스에서 지정한 MOLAP 추출 쿼리를 직접 만족시키는 하나의 요약 테이블이 들어 있습니다.

MOLAP 추출 슬라이스



가능한 권장사항

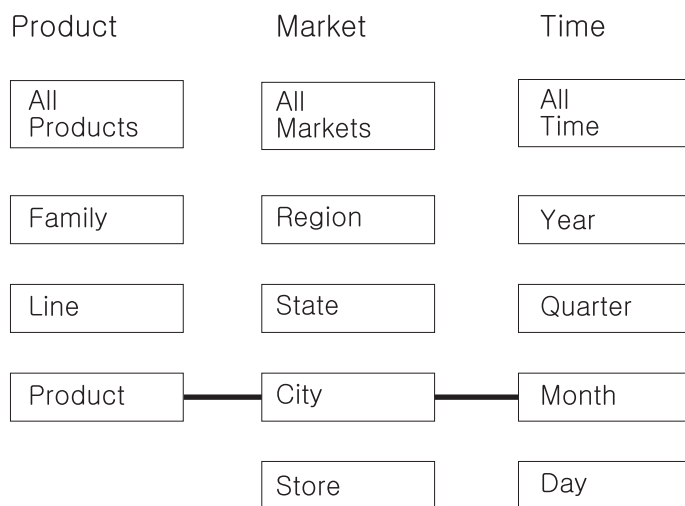


그림 32. MOLAP 추출. MOLAP 추출 최적화 슬라이스 및 가능한 요약 테이블 권장사항

하이브리드 추출 최적화 슬라이스

하이브리드 추출 최적화 슬라이스는 사용자가 일반적으로 지정된 슬라이스의 데이터를 벤더 제품의 하이브리드 OLAP(HOLAP) 큐브에 추출함을 의미합니다. 따라서 최적화 어드바이저는 지정된 슬라이스를 권장 요약 테이블에 포함시켜야 합니다.

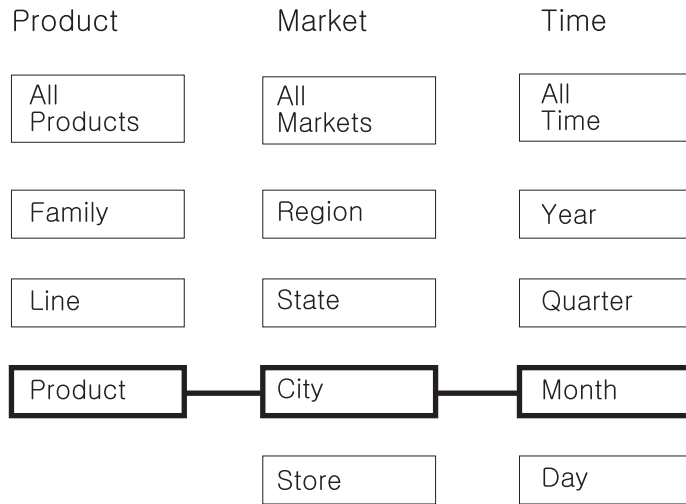
권장사항: 최적화 슬라이스가 HOLAP 큐브에 추출하는 데이터의 레벨과 일치하도록 각 큐브 차원에 특정 레벨을 지정하십시오.

큐브당 하나의 하이브리드 추출 최적화 슬라이스만 정의할 수 있습니다. 하이브리드 추출 최적화 슬라이스가 들어 있는 큐브에 MOLAP 추출 최적화 슬라이스를 정의할 수 없습니다. 하이브리드 추출 최적화 슬라이스가 들어 있는 동일한 큐브에 0개 이상의 드릴 쓰루 최적화 슬라이스를 정의할 수 있습니다. 최적화 어드바이저는 지정된 하이브리드 추출 최적화 슬라이스 아래에 드릴 쓰루 쿼리가 있을 수 있음을 예상하고 지정된 슬라이스의 하이브리드 추출 쿼리 및 지정된 슬라이스 아래의 드릴 쓰루 쿼리에서 최적화하려고 합니다.

114 페이지의 그림 33에서는 최적화 어드바이저가 요약 테이블을 권장할 수 있는 슬라이스 및 하이브리드 추출 최적화 슬라이스에 대한 예를 표시합니다. 최적화 슬라이스는 Product 큐브 차원의 Product 레벨, Market 큐브 차원의 City 레벨 및 Time 큐브 차원의 Month 레벨에서 정의됩니다. 이 최적화 슬라이스는 Product-City-Month 레벨의 데이터를 HOLAP 큐브에 추출함을 의미합니다.

가능한 요약 테이블 권장사항에는 두 개의 집계 레벨이 들어 있습니다. Product-City-Month 슬라이스의 집계 레벨은 최적화 슬라이스에서 지정한 HOLAP 추출 쿼리를 직접 만족시킵니다. Line-State-Day 슬라이스의 집계 레벨은 Time 큐브 차원의 가능한 드릴 쓰루 쿼리를 만족시키기 위해 하이브리드 추출 슬라이스 아래에 있는 Day 레벨을 포함합니다. 최적화 어드바이저가 기타 메타데이터를 분석하고 데이터 샘플링을 수행하여 이 권장 요약 테이블을 개발했습니다.

하이브리드 추출 슬라이스



가능한 권장사항

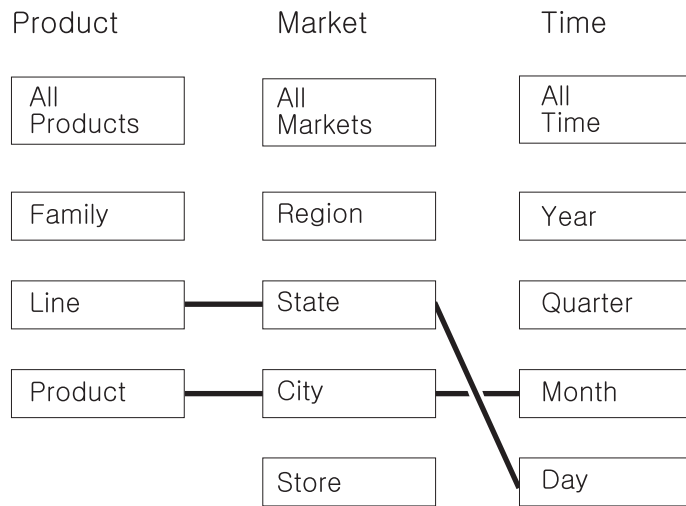


그림 33. 하이브리드 추출. 하이브리드 추출 최적화 슬라이스 및 가능한 요약 테이블 권장사항

드릴 쓰루 최적화 슬라이스

드릴 쓰루 최적화 슬라이스에는 해당 슬라이스에 대응하는 큐브에 정의된 하이브리드 추출 최적화 슬라이스가 있어야 합니다. 드릴 쓰루 최적화 슬라이스는 사용자가 일반적으로 벤더 제품의 OLAP(HOLAP) 큐브에서 지정된 슬라이스에 드릴 쓰루함을 의미합니다. 따라서 최적화 어드바이저는 지정된 레벨 또는 그 아래에 있는 슬라이스를 권장 요약 테이블에 포함시켜야 합니다.

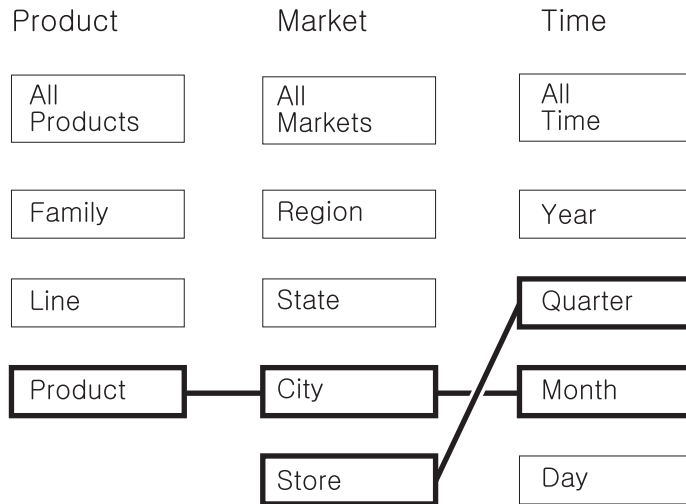
권장사항: 레벨이 특히 중요하다라는 것을 알고 있을 때에만 드릴 쓰루 최적화 슬라이스 유형의 큐브 차원에 특정 레벨을 지정하십시오. 일반적으로 드

릴 쓰루 최적화 슬라이스의 경우 대부분의 큐브 차원에 임의 옵션을 지정하고 해당 레벨이 집중적인 쿼리 활동과 관련된 경우에만 특정 레벨을 선택하십시오.

116 페이지의 그림 34에서는 최적화 어드바이저가 요약 테이블을 권장할 수 있는 슬라이스 및 드릴 쓰루 최적화 슬라이스와 이에 대응하는 하이브리드 추출 최적화 슬라이스에 대한 예를 표시합니다. 하이브리드 추출 최적화 슬라이스는 Product 큐브 차원에서 Product 레벨, Market 큐브 차원에서 City 레벨 및 Time 큐브 차원에서 Month 레벨로 정의됩니다. 이 최적화 슬라이스는 Product-City-Month 레벨의 데이터를 HOLAP 큐브에 추출함을 의미합니다. 드릴 쓰루 최적화 슬라이스는 Product 차원에서 임의 레벨, Market 큐브 차원에서 Store 레벨 및 Time 큐브 차원에서 Quarter 레벨로 정의됩니다. 이 최적화 슬라이스는 일반적으로 HOLAP 큐브의 드릴 쓰루 쿼리에 Store 및 Quarter 레벨이 포함되어 해당 드릴 쓰루 쿼리가 Product 큐브 차원에서 특정 레벨을 참조할 수 있거나 참조하지 않을 수 있음을 의미합니다.

가능한 요약 테이블 권장사항에는 두 개의 집계 레벨이 들어 있습니다. Product-City-Month 슬라이스의 요약 테이블은 최적화 슬라이스에서 지정한 HOLAP 추출 쿼리를 직접 만족시킵니다. Family-Store-Quarter 슬라이스의 집계 레벨에는 드릴 쓰루 최적화 슬라이스에서 지정한 Store 레벨 및 Quarter 레벨이 들어 있습니다. 최적화 어드바이저가 기타 메타데이터를 분석하고 데이터 샘플링을 수행하여 이 요약 테이블에서 Product 큐브 차원의 Family 레벨을 포함하도록 권장합니다.

드릴 쓰루 슬라이스로 하이브리드 추출



가능한 권장사항

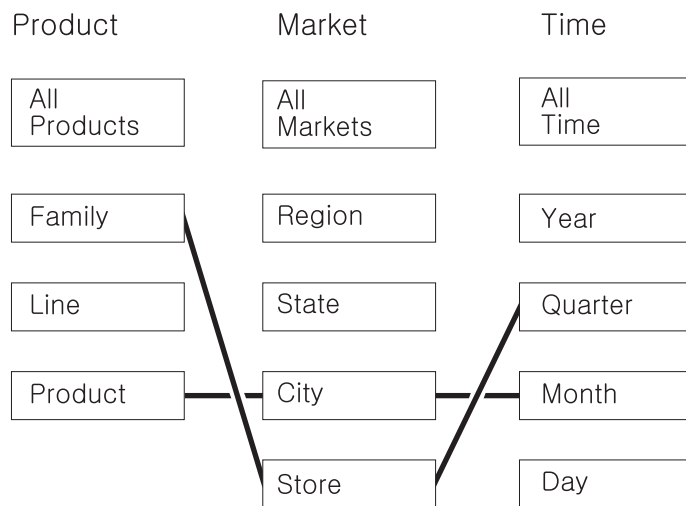


그림 34. 드릴 쓰루 하이브리드 추출 최적화 슬라이스와 드릴 쓰루 최적화 슬라이스 및 가능한 요약 테이블 권장사항

후보 최적화 슬라이스에서 쿼리 분석

최적화 슬라이스는 쿼리 성능을 향상시키는 데 유용한 도구지만 발행된 SQL 쿼리를 정확히 반영하는 경우에만 효과적입니다.

사용자의 쿼리 패턴을 분석하는 경우 다음 정보를 고려하십시오.

- 사용자가 발행할 쿼리 유형
- 쿼리에서 자주 액세스하는 계층 구조 레벨
- 사용자가 쿼리를 발행하는 제품

사용자의 쿼리 실행기록을 검토하여 최적화 슬라이스를 작성해야 하는 큐브 영역을 판별할 수 있습니다. 또한 사용자를 조사하여 나중에 쿼리 요구를 변경할 것인지를 판별할 수 있습니다. 최고 우선순위인 영역을 최적화하려고 합니다.

권장사항: 하나의 큐브에 최적화 슬라이스를 넷 이상 지정하지 마십시오.

특정 상황에서 정의하는 최적화 슬라이스의 예

각 시나리오는 CVSAMPLE 데이터베이스의 Daily Sales 큐브에 기반을 두고 있으며 이 데이터베이스에는 각각 다음 큐브 계층 구조를 갖는 Product, Market 및 Time과 같은 세 개의 큐브 차원이 있습니다.

표 35. CVSAMPLE. CVSAMPLE 데이터베이스에 있는 Daily Sales 큐브의 큐브 차원 및 이에 대응하는 큐브 계층 구조입니다.

Product 큐브 차원	Market 큐브 차원	Time 큐브 차원
All products	All markets	All time
Family	Region	Year
Line	State	Quarter
Product	City	Month
	Postal code	Day
	Store	

다음 테이블에 설명된 상황은 CVSAMPLE 데이터베이스에 기반을 두고 있는 특정 쿼리 로드에서 정의할 수 있는 최적화 슬라이스의 예입니다.

표 36. 최적화 슬라이스 예

상황	시나리오	정의할 최적화 슬라이스
사용자는 특정 쿼리 유형을 발행하는 쿼리 제품을 사용함	사용자가 드릴 다운 쿼리를 발행하는 제품을 주로 사용하는 점을 알고 있습니다. 더 자주 쿼리되는 큐브 영역에 관한 정보가 없습니다.	사용자에게 중요한 큐브 영역에 관한 특정 정보를 알지 못하므로 특정 최적화 슬라이스에서는 혜택을 많이 제공하지 않습니다. 이러한 일반 상황에서 큐브 마법사로 큐브를 작성할 때 또는 큐브 작성 후 큐브 등록 정보를 수정하여 전체 큐브를 드릴 다운 쿼리로 지정할 수 있습니다.
대부분의 쿼리 유형은 하나이고 이들 쿼리는 하나의 레벨 그룹화에 촛점이 맞추어져 있습니다.	사용자가 주로 보고서 쿼리를 발행하고 대부분의 쿼리가 Market 차원의 주 레벨을 참조한다는 점을 알고 있습니다.	큐브에서 특히 중요한 영역에 관한 특정 정보를 알고 있으면 최적화 슬라이스는 많은 도움이 됩니다. 보고서 유형인 임의-주-임의(Any-State-Any) 최적화 슬라이스를 정의할 수 있습니다. 이 슬라이스의 경우 Product 큐브 차원에서는 임의 레벨, Market 큐브 차원에서는 주 레벨, Time 큐브 차원에서는 임의 레벨을 참조합니다.

표 36. 최적화 슬라이스 예 (계속)

상황	시나리오	정의할 최적화 슬라이스
대부분의 쿼리 유형은 하나이고 이들 쿼리는 일부 레벨 그룹화에 축점이 맞추어져 있습니다.	사용자가 주로 보고서 쿼리를 발행한다는 점을 알고 있습니다. 절반의 쿼리가 Market 큐브 차원에서는 주 레벨을 참조하고 나머지 절반의 쿼리는 큐브 차원의 다른 레벨 전반에서 임의 대로 분산됩니다.	<p>다음 두 최적화 슬라이스를 정의하여 이 쿼리 분산을 정확히 표시할 수 있습니다.</p> <ul style="list-style-type: none"> Any-State-Any 레벨의 보고서 최적화 슬라이스를 정의하여 State 레벨을 참조하는 쿼리를 표시하십시오. 이 슬라이스의 경우 Product 큐브 차원에서는 임의 레벨, Market 큐브 차원에서는 주 레벨, Time 큐브 차원에서는 임의 레벨을 참조합니다. 임의-임의-임의(Any-Any-Any) 레벨의 보고서 최적화 슬라이스를 정의하여 모든 큐브 차원에서 모든 레벨을 참조할 수 있는 기타 쿼리를 표시하십시오. 두 번째 슬라이스를 지정하여 Market 차원의 기타 레벨에 액세스하는 쿼리가 상당히 많음을 명확히 표시합니다.
대부분의 쿼리 유형은 하나이고 이들 쿼리는 다수 레벨 그룹화에 축점이 맞추어져 있습니다.	사용자가 주로 보고서 쿼리를 발행한다는 점을 알고 있습니다. 쿼리 활동이 무작위는 아니지만 약 15개의 큐브 영역에 걸쳐 분산되어 있습니다.	<p>최적화 슬라이스는 쿼리 활동이 많은 영역을 지정함을 의미하며 이 상황의 경우 쿼리가 너무 많이 분산되어 있으므로 특정 최적화 슬라이스 세트를 지정할 수 없습니다. 최적화 슬라이스를 지정하는 대신 큐브 마법사로 큐브를 작성할 때 또는 큐브 작성 후에 큐브 등록 정보를 수정하여 전체 큐브를 보고서 쿼리로 지정하십시오.</p>

최적화 수행 시 제한조건 정의

제한조건은 최적화 어드바이저 및 DB2 옵티마이저에 유용한 정보를 제공합니다. 스타 스키마 또는 snowflake 스키마에 있는 외부 키 및 기본 키의 정보용 또는 강제 실행 제한조건을 정의해야 합니다.

최적화 어드바이저를 사용하기 전에 기본 테이블에 대한 제한조건을 정의해야 합니다. 제한조건은 큐브 모델이 최적화에 유효하도록 42 페이지의 『메타데이터 오브젝트 규칙』에 설명된 최적화 규칙, 큐브 모델 완전성 규칙 및 기본 규칙을 지원해야 합니다. 기본적으로 규칙은 큐브 모델의 메타데이터 오브젝트를 함께 조인하는 방법을 정의합니다.

정의해야 하는 외부 키 제한조건에서 정보용 제한조건을 사용할 수 있습니다. 정보용 제한조건은 DB2 Universal Database, 버전 8에 제공되는 새 제한조건 유형입니다. 정보용 제한조건은 유지보수 비용을 증가시키지 않고도 쿼리 성능을 향상시키는 방법을 제공합니다. 이러한 제한조건은 DB2 SQL 컴파일러에 의해 사용될 수 있지만 데이터베이스 관리 프로그램에 필요한 것은 아닙니다. 이러한 유형의 제한조건을 통해 DB2 UDB는 관계를 요구하지 않고도 데이터의 관계를 알 수 있습니다. 기본 키 제한조건의 경우 DB2 UDB에서 제공하는 데이터베이스 강제 실행 제한조건을 사용해야 합니다.

각 조인에는 이에 대응하는 정의된 제한조건이 필요합니다. 예를 들어 snowflake 스키마에서 사용되는 사실-차원 조인 및 차원-차원 조인과 관련된 컬럼에는 제한조건이 있어야 합니다.

120 페이지의 그림 35에 표시된 snowflake 스키마에 기반하여 큐브 모델을 최적화하려면 각 사실-차원 조인에 제한조건을 정의해야 합니다. 세 가지의 사실 대 차원 조인은 다음과 같습니다.

- Store.StoreID 및 Sales.StoreID 사이
- Time.TimeID 및 Sales.TimeID 사이
- Product.ProductID 및 Sales.ProductID 사이

몇 개의 규칙이 이러한 조인 각각에 적용됩니다. 외부 키 제한조건에서만 정보용 제한조건을 사용할 수 있습니다.

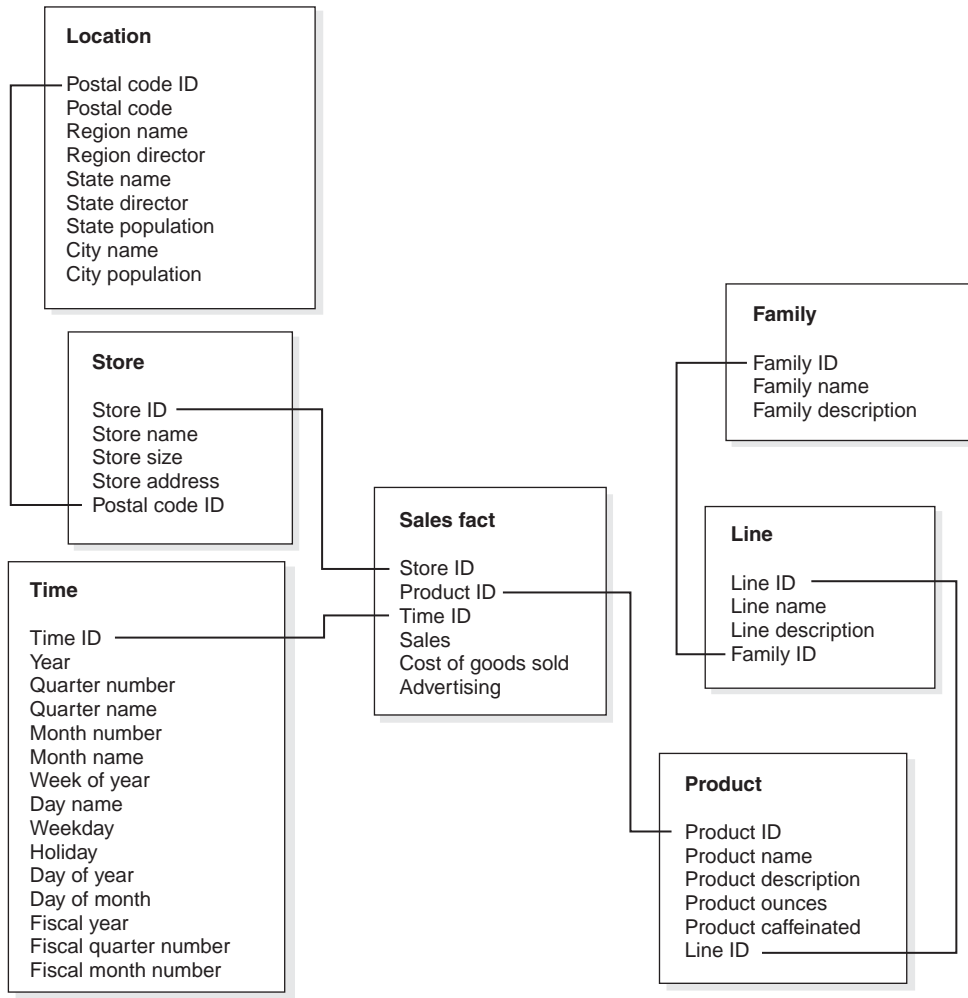


그림 35. snowflake 스키마. CVSAMPLE 데이터베이스의 snowflake 스키마에서 사용된 관계형 테이블입니다.

Store 및 Sales 테이블 간 조인의 경우 다음 제한조건을 정의해야 합니다.

- StoreID는 Store 테이블의 기본 키입니다.
- Store.StoreID 및 Sales.StoreID는 모두 널이 아닌 컬럼입니다.
- Sales.StoreID는 Store.StoreID를 참조하는 외부 키입니다. 외부 키 제한조건은 정보용 제한조건으로 정의될 수 있습니다.
- Sales.StoreID가 Sales 테이블의 기본 키가 아닌 경우 조인 카디널리티(cardinality)는 일대다(Store.StoreID : Sales.StoreID)입니다. Sales.StoreID가 Sales 테이블의 기본 키인 경우 조인 카디널리티(cardinality)는 일대일입니다.
- 조인 유형은 내부 조인입니다.

Time 및 Sales 테이블 간 조인의 경우 다음 제한조건을 정의해야 합니다.

- TimeID는 Time 테이블의 기본 키입니다.
- Time.TimeID 및 Sales.TimeID는 모두 널이 아닌 컬럼입니다.

- Sales.TimeID는 Time.TimeID를 참조하는 외부 키입니다. 외부 키 제한조건은 정보용 제한조건으로 정의될 수 있습니다.
- Sales.TimeID가 Sales 테이블의 기본 키가 아닌 경우 조인 카디널리티(cardinality)는 일대다(Time.TimeID : Sales.TimeID)입니다. Sales.TimeID가 Sales 테이블의 기본 키인 경우 조인 카디널리티(cardinality)는 일대일입니다.
- 조인 유형은 내부 조인입니다.

Product 및 Sales 테이블 간 조인의 경우 다음 제한조건을 정의해야 합니다.

- ProductID는 Product 테이블의 기본 키입니다.
- Product.ProductID 및 Sales.ProductID는 모두 널이 아닌 컬럼입니다.
- Sales.ProductID는 Product.ProductID를 참조하는 외부 키입니다. 외부 키 제한조건은 정보용 제한조건으로 정의될 수 있습니다.
- Sales.ProductID가 Sales 테이블의 기본 키가 아닌 경우 조인 카디널리티(cardinality)는 일대다(Product.ProductID : Sales.ProductID)입니다. Sales.ProductID가 Sales 테이블의 기본 키인 경우 조인 카디널리티(cardinality)는 일대일입니다.
- 조인 유형은 내부 조인입니다.

snowflake 스키마에서 각 차원에는 하나 이상의 추가 차원이 조인할 수 있는 기본 차원 테이블이 들어 있습니다. 기본 차원 테이블은 사실 테이블에 조인할 수 있는 유일한 테이블입니다. 기본 테이블에 직접 조인되는 각 아우트리저 테이블의 조인 카디널리티(cardinality)는 다대일(기본 테이블이 다수임) 또는 일대일이어야 합니다. 보통 기본 차원 테이블에는 이러한 조인 카디널리티(cardinality) 규칙 때문에 모든 차원 테이블의 더 자세한 세부사항 정보 레벨이 있습니다. 차원 테이블 세트가 일대일 조인 카디널리티(cardinality)만 사용하는 경우 모든 테이블에는 동일한 세부사항 레벨이 들어 있습니다.

이 큐브 모델은 snowflake 스키마에 기반하므로 차원 테이블 간 조인에 추가 제한조건을 정의해야 합니다. 세 가지의 치수 대 차원 조인은 다음과 같습니다.

- Store 테이블 및 Location 테이블 사이
- Product 테이블 및 Line 테이블 사이
- Line 테이블 및 Family 테이블 사이

몇 개의 규칙이 이러한 조인 각각에 적용됩니다. 외부 키 제한조건에서만 정보용 제한조건을 사용할 수 있습니다.

Product 및 Location 테이블 간 조인의 경우 다음 제한조건을 정의해야 합니다.

- PostcodeID는 Location 테이블의 기본 키입니다.
- Location.PostcodeID 및 Store.PostalCodeID는 모두 널이 아닌 컬럼입니다.

- Store.PostalCodeID는 Location.PostalCodeID를 참조하는 외부 키입니다. 외부 키 제한조건은 정보용 제한조건으로 정의될 수 있습니다.
- Store.PostalCodeID가 Product 테이블의 기본 키 또는 고유 키가 아니므로 조인 카디널리티(cardinality)는 일대다(Location.PostalCodeID : Store.PostalCodeID)입니다.
- 조인 유형은 내부 조인입니다.

Product 및 Line 테이블 간 조인의 경우 다음 제한조건을 정의해야 합니다.

- LineID는 Line 테이블의 기본 키입니다.
- Line.LineID 및 Product.LineID는 모두 널이 아닌 컬럼입니다.
- Product.LineID는 Line.LineID를 참조하는 외부 키입니다. 외부 키 제한조건은 정보용 제한조건으로 정의될 수 있습니다.
- Product.LineID가 Store 테이블의 기본 키 또는 고유 키가 아니므로 조인 카디널리티(cardinality)는 일대다(Line.LineID : Product.LineID)입니다.
- 조인 유형은 내부 조인입니다.

Line 및 Family 테이블 간 조인의 경우 다음 제한조건을 정의해야 합니다.

- FamilyID는 Family 테이블의 기본 키입니다.
- Family.FamilyID 및 Line.FamilyID는 모두 널이 아닌 컬럼입니다.
- Line.FamilyID는 Family.FamilyID를 참조하는 외부 키입니다. 외부 키 제한조건은 정보용 제한조건으로 정의될 수 있습니다.
- Line.FamilyID가 Line 테이블의 기본 키 또는 고유 키가 아니므로 조인 카디널리티(cardinality)는 일대다(Family.FamilyID : Line.FamilyID)입니다.
- 조인 유형은 내부 조인입니다.

123 페이지의 그림 36에서는 snowflake 스키마 차원에서 유효한 차원 테이블 세트를 표시합니다. 기본 차원 테이블은 Customer에 직접 조인된 City와 CustomerGroup 및 City에 조인된 CityInfo를 포함하여 세 개의 추가 아우트리거 테이블로 구성된 Customer 테이블입니다. 도시 또는 고객 그룹에 다수의 고객이 있을 수 있으므로 조인 카디널리티의 의미가 유효하며 도시당 하나의 도시 정보 세트가 존재합니다. 이는 최적화 유효성 확인 규칙을 준수하므로 최적화에 유효한 차원입니다. 차원에는 하나의 테이블만 있으며 기본 테이블에 직접 조인되는 City 및 CustomerGroup 테이블은 다대일 카디널리티와 조인됩니다. CityInfo 테이블은 유효한 일대일 카디널리티와도 조인됩니다. Customer 테이블에는 네 개의 차원 테이블 중 가장 자세한 세부사항 정보 레벨이 들어 있습니다.

최적화에 대한 유효한 차원

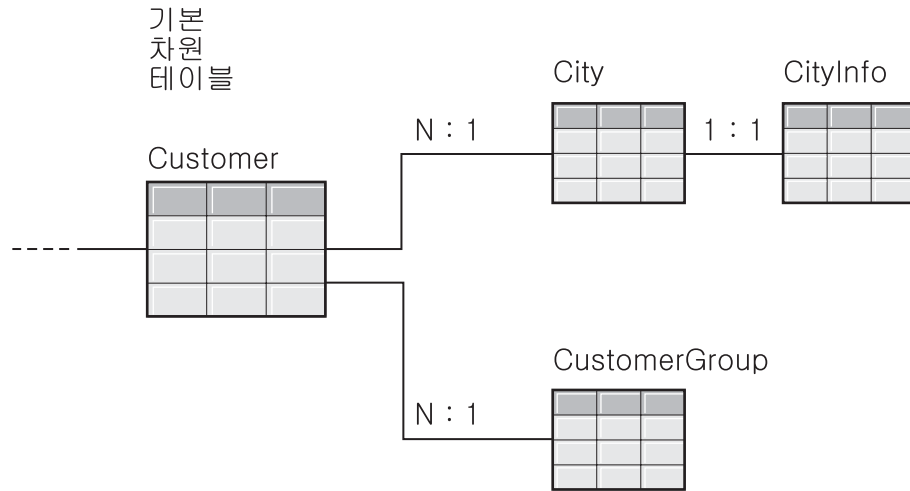


그림 36. 유효한 차원. 최적화될 수 있는 하나의 차원에서 사용되는 차원 테이블 세트

124 페이지의 그림 37에서는 snowflake 스키마 차원의 유효하지 않은 차원 테이블 세트를 표시합니다. 정의되어 있는 카디널리티 관계 때문에 이러한 테이블은 큐브 모델에서 최적화될 기본 차원 테이블이 될 수 없습니다. 카디널리티의 의미가 유효해도 이러한 테이블이 기본 차원 테이블로 사실 테이블과 조인된 경우 사실 테이블의 데이터가 곱해져서 팬 트랩(fan trap)으로 알려질 수 있습니다.

예를 들어 Customer가 기본 차원 테이블인 경우 Region 및 SalesRep 간 일대다 조인 카디널리티 때문에 해당 차원이 최적화에 유효하지 않게 됩니다. 각 도에 다섯 명의 영업 담당자가 있는 경우 SalesRep 및 Region 테이블이 조인되어 있으면 각 Region에는 다섯 항목이 들어 있습니다. 이러한 테이블이 시 및 Customer 테이블과 조인되어 결국 사실 테이블과 조인된 경우 다섯 개의 추가 행이 각 City, Customer 및 사실 테이블의 기존 행에 추가됩니다. 동일한 사실 행을 다섯 번 반복하면 치수가 잘못 계산될 수 있습니다. 차원의 기타 테이블마다 유사한 문제점이 있습니다. City 테이블은 City 및 Customer 및 Region와 SalesRep 간 일대다 조인 때문에 기본 차원 테이블이 될 수 없습니다. Region 테이블은 차원의 모든 조인이 Region 테이블에 일대다로 조인되어 있으므로 사실 테이블과 조인할 수 없습니다. SalesRep 테이블도 Region 및 City 테이블과 City 및 Customer 테이블 간 일대다로 조인되어 있으므로 기본 차원 테이블이 될 수 없습니다.

최적화에 대한 유효하지 않은 차원

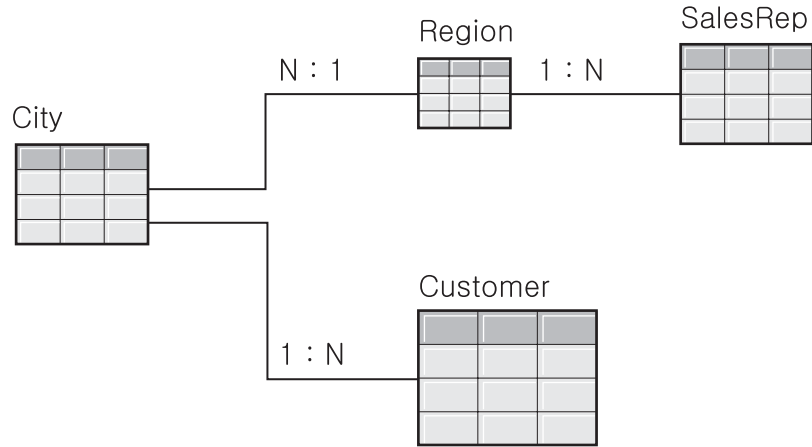


그림 37. 유효하지 않은 차원. 최적화될 수 없는 하나의 차원에서 사용되는 차원 테이블 세트

최적화 어드바이저의 매개변수

Optimization Advisor 마법사에 제공하는 각 매개변수 정보는 마법사가 권장하는 요약 테이블 및 사용자가 얻는 성능 향상에 영향을 미칩니다. 정확한 정보를 제공하고 비용과 성능 요구사항을 고려하여 신중히 결정하도록 하십시오.

예상된 쿼리 활동

큐브 모델에서 각 큐브를 최적화하려는 쿼리 유형을 지정하십시오. 쿼리 유형은 DB2 관계형 데이터가 일반적으로 액세스되는 시기 및 방법을 설명합니다. 이 정보는 최적화 어드바이저가 가장 자주 쿼리되는 큐브 모델의 부분을 이해하는 데 도움을 줍니다. 소수의 특정 슬라이스가 가장 자주 쿼리된다는 점을 알고 있는 경우 큐브에 최적화 슬라이스를 지정할 수도 있습니다.

큐브 작성 시 각 큐브에 쿼리 유형을 정의합니다. 최적화 어드바이저를 사용하여 각 큐브에 지정하려는 유형을 검토하고 필요한 사항을 변경할 수 있습니다.

디스크 스페이스 제한사항

요약 테이블에서 사용될 수 있는 대략적인 디스크 스페이스 크기를 지정합니다. 최적화 어드바이저는 요약 테이블을 빌드하기 전에는 해당 요약 테이블의 정확한 크기를 알 수 없으므로 가능한 지정된 디스크 스페이스 크기와 유사한 요약 테이블을 권장합니다. 빌드되는 요약 테이블은 사용자가 지정하는 것보다 크거나 적은 스페이스를 사용할 수 있습니다.

사용자가 지정하는 디스크 스페이스 크기는 최적화 결과와 직접 관련됩니다. 디스크 스페이스를 늘리면 성능이 향상된 쿼리 수 및 향상 정도가 모두 증가될 수 있습니다. 디스크 스페이스 크기를 선택할 때 다음 요인을 고려해야 합니다.

- 필요한 쿼리 성능 레벨
- 최적화할 큐브 모델 수
- 각 큐브 모델의 중요도
- 각 큐브 모델의 사용 빈도
- 디스크 스페이스의 사용 가능성 및 사용 비용

보통 큐브 모델에서 참조하는 관계형 테이블이 현재 사용하는 스페이스의 1 - 10%와 같이 적절한 디스크 스페이스 크기를 허용하면 성능이 크게 향상될 수 있습니다. 표 37에서는 요약 테이블에서 사용하는 디스크 스페이스 크기 및 예상된 쿼리 성능 향상 사이의 관계를 표시합니다. 제공할 스페이스 크기를 결정할 때 모든 메타데이터 및 기본 테이블의 컨텍스트에 있는 각 큐브 모델을 고려하십시오.

표 37. 디스크 스페이스 사용되는 디스크 스페이스의 백분율 및 이에 대응하는 예상된 성능 향상

요약 테이블에서 사용하는 기본 테이블 디스크 스페이스의 백분율	관련 쿼리에서의 예상된 향상
1% 미만	낮음
5%	중간
50%	높음
무제한	최고

시간 제한사항

사용자가 지정하는 시간은 최적화 어드바이저가 권장사항을 판별하는 데 사용할 수 있는 최대값입니다. 최적화 어드바이저의 실행 시간이 길수록 더 나은 결과를 얻을 수 있습니다. 다음 테이블에서는 최적화 어드바이저에 제공해야 하는 시간 크기에 대한 일부 근사값 지침을 제공합니다. 성능 결과가 다양하므로 표 38에 지정된 시간보다 더 많은 시간을 허용해야 할 수 있습니다.

표 38. 시간 제한사항. 최적화 어드바이저에 제공할 시간 크기에 대한 지침

데이터베이스 최적화 시나리오	근사값 시간 제한
데이터 샘플을 수행하지 않는 경우	5 - 30분
크기가 10GB 미만인 소형 데이터베이스에서 데이터 1시간 이하 샘플링을 수행하는 경우	
크기가 10GB를 초과하는 대형 데이터베이스에서 데이터 1시간 이상 샘플링을 수행하는 경우	

데이터 샘플링

데이터 샘플링은 최적화 어드바이저가 큐브 모델의 데이터를 검사하는 방법입니다. 이를 통해 가장 효과적인 권장사항 세트를 작성할 수 있도록 최적화 어드바이저에 추가 정보를 제공합니다. 데이터 샘플링으로 작성된 권장사항은 지정된 디스크 스페이스와 보다 정확하게 일치합니다. 데이터 샘플링을 수행하지 않은 경우 최적화 어드바이저는 메타데이터 및 DB2 통계만 분석하여 권장사항을 판별합니다.

큐브 모델 최적화

큐브 모델에 대해 다시 수행된 쿼리를 최적화함으로써 OLAP 스타일의 SQL 쿼리를 실행하는 제품의 성능을 향상시킬 수 있습니다.

전제조건: 큐브 모델에 사용되는 기본 테이블에 대해 DB2 제한조건이 지정되어 있어야 합니다. 각각의 사실 테이블 및 차원 테이블 간에, 그리고 snowflake 스키마의 각 차원 테이블 간에 제한조건이 지정되어 있어야 합니다. 제한조건은 널(null)이 아닌 컬럼에 지정되어 있어야 합니다. 제한조건 정의에 대한 자세한 정보는 118 페이지의 『최적화 수행 시 제한조건 정의』를 참조하십시오.

큐브 모델을 최적화할 때, 최적화 어드바이저 마법사는 큐브 모델에 대한 권장 요약 테이블 세트를 빌드할 수 있는 SQL을 작성합니다. 요약 테이블은 쿼리 성능의 향상을 위해 일반 액세스 데이터를 집계합니다.

큐브 모델을 최적화하려면 다음을 다음을 수행하십시오.

1. OLAP 센터 오브젝트 트리에서 큐브 모델을 마우스 오른쪽 단추로 누르고 **최적화 어드바이저**를 눌러 최적화 어드바이저 마법사를 여십시오.
2. 쿼리 유형 페이지에서 각 큐브를 최적화하려는 쿼리 유형을 검토하십시오. 쿼리 유형을 수정하거나 큐브의 최적화 슬라이스를 지정할 수 있습니다. 큐브 쿼리 유형은 최적화 결과를 향상시키는 데 사용됩니다. 최적화 슬라이스에 관한 정보는 107 페이지의 『큐브의 최적화 슬라이스』를 참조하십시오.
3. 요약 테이블 페이지에서는 요약 테이블의 즉시 또는 지연 갱신을 지정하십시오. 갱신 옵션에 대한 정보는 133 페이지의 『요약 테이블 유지보수』를 참조하십시오. 요약 테이블 및 요약 테이블 인덱스를 저장할 테이블 스페이스를 지정하십시오.
4. 제한사항 페이지에서 빌드할 요약 테이블 및 인덱스에 허용할 디스크 스페이스 크기를 지정하십시오. 데이터 샘플링을 허용하고자 하는 경우에 지정하십시오. 또한 최적화 어드바이저의 권장사항 결정에 허용할 최대 시간을 지정하십시오. 스페이스, 정보 및 시간을 보다 많이 지정할수록 성능 결과가 현저하게 향상됩니다. 최적화 어드바이저 마법사에서 매개변수를 지정하는 방법에 관한 정보는 124 페이지의 『최적화 어드바이저의 매개변수』를 참조하십시오.

이는 최적화 어드바이저 마법사의 마지막 입력 매개변수 페이지입니다. 요약 테이블의 작성 및 새로 고침 권장사항을 판별하는 최적화 어드바이저의 진행을 모니터링할 수 있도록(필요한 경우 중지함) 다음을 눌러 진행 창을 여십시오. 최적화 어드바이저의 시간 제한사항을 수정할 수 있습니다. 중지를 누른 경우 지정된 시간이 지나면 최적화 어드바이저가 판별한 요약 테이블 권장사항을 리턴합니다.

5. SQL 스크립트 페이지에서 요약 테이블 작성 SQL 스크립트 필드 및 요약 테이블 새로 고침 SQL 스크립트 필드에 각각 고유한 파일 이름을 입력하십시오.
6. 완료를 눌러 권장 SQL 스크립트를 사용자가 지정한 파일 이름으로 저장하십시오.
7. SQL 스크립트를 실행하십시오. 대형 요약 테이블을 작성하는 경우, 요약 테이블의 작성을 완료하기 위해 상당한 시간이 소요될 수 있습니다. DB2 명령 센터 또는 명령 창을 사용하여 SQL 스크립트를 실행할 수 있습니다. DB2 명령 창에서 SQL 스크립트를 실행하려면 다음을 수행하십시오.
 - a. SQL 스크립트를 저장한 디렉토리로 변경하십시오.
 - b. 최적화한 큐브 모델의 데이터베이스에 연결하십시오. 예를 들어, db2 connect to CVSAMPLE를 입력하십시오.
 - c. 다음 명령을 입력하십시오.

```
db2 -tvf filename
```

여기서, *filename*은 요약 테이블 작성 SQL 스크립트의 이름입니다.

요약 테이블을 작성하는 SQL 스크립트에 대한 예

최적화 어드바이저 마법사는 권장 요약 테이블을 작성하는 SQL 스크립트를 제공합니다. SQL 스크립트에는 하나 이상의 요약 테이블을 빌드하는 필수 SQL 명령이 들어 있습니다.

128 페이지의 그림 38에서는 하나의 요약 테이블을 작성하는 샘플 SQL 스크립트의 일부와 메타데이터 오브젝트를 SQL에 맵핑하는 방법을 표시합니다. 샘플 SQL 스크립트에서 요약 테이블은 DB2INFO.MQT0000000021T01입니다. 여기서 21은 큐브 모델 ID이고 T01은 요약 테이블 ID입니다. 큐브 모델 ID 길이는 10자리까지 가능합니다. 요약 테이블 ID는 큐브 모델에 있는 요약 테이블을 식별합니다. 요약 테이블 ID는 하나의 큐브 모델에서 최대 99개의 요약 테이블에 허용됩니다. 최적화 어드바이저 마법사가 정의하는 요약 테이블 이름을 변경하지 마십시오. 테이블 이름을 변경하면 DB2 Cube Views는 큐브 모델에 작성하는 요약 테이블을 식별할 수 없습니다.

```

DROP TABLE DB2INFO.MQT0000000021T01;
UPDATE COMMAND OPTIONS USING c OFF;
CREATE SUMMARY TABLE DB2INFO.MQT0000000021T01 AS
(SELECT
SUM(T1."SALES") AS "SALES",
SUM(T1."COGS") AS "COGS",
SUM(T1."ADVERTISING") AS "ADVERTISING",
SUM(T1."COGS" + T1."ADVERTISING") AS "Total expense",
SUM(T1."SALES" - (T1."COGS" + T1."ADVERTISING")) AS "Profit",
T5."REGION_NAME" AS "REGION_NAME",
T5."STATE_NAME" AS "STATE_NAME",
T6."LINEID" AS "LINEID (LINE)",
T4."YEAR" AS "YEAR",
T4."QUARTER_NUMBER" AS "QUARTER_NUMBER",
"Qtr." CONCAT (cast (T4."QUARTER_NUMBER" AS CHAR(1))) AS "QUARTER_NAME",
T4."MONTH_NUMBER" AS "MONTH_NUMBER"
FROM "CVSAMPLE"."SALESFACT" AS T1,
"CVSAMPLE"."STORE" AS T2,
"CVSAMPLE"."PRODUCT" AS T3,
"CVSAMPLE"."TIME" AS T4,
"CVSAMPLE"."LOCATION" AS T5,
"CVSAMPLE"."LINE" AS T6
WHERE
T1."STOREID"=T2."STOREID" AND T1."PRODUCTID"=T3."PRODUCTID"
AND T1."TIMEID"=T4."TIMEID" AND T2."POSTALCODEID"=T5."POSTALCODEID"
AND T3."LINEID"=T6."LINEID"
GROUP BY
T5."REGION_NAME",
T5."STATE_NAME",
T6."LINEID",
T4."YEAR",
T4."QUARTER_NUMBER",
"Qtr." CONCAT (cast (T4."QUARTER_NUMBER" AS CHAR(1))),
T4."MONTH_NUMBER")
DATA INITIALLY DEFERRED
REFRESH DEFERRED
ENABLE QUERY OPTIMIZATION
MAINTAINED BY SYSTEM
NOT LOGGED INITIALLY;
COMMENT ON TABLE DB2INFO.MQT0000000021T01 IS 'AST created for cube model CVSAMPLE.Sales Model';
COMMIT;
ALTER TABLE DB2INFO.MQT0000000021T01 ACTIVATE NOT LOGGED INITIALLY;
REFRESH TABLE DB2INFO.MQT0000000021T01;
CREATE INDEX DB2INFO.IDX0000000021T0101 ON DB2INFO.MQT0000000021T01("STATE_NAME");
CREATE INDEX DB2INFO.IDX0000000021T0102 ON DB2INFO.MQT0000000021T01("MONTH_NUMBER");
CREATE INDEX DB2INFO.IDX0000000021T0101C ON DB2INFO.MQT0000000021T01("LINEID (LINE)") CLUSTER;
COMMIT;
REORG TABLE DB2INFO.MQT0000000021T01;
RUNSTATS ON TABLE DB2INFO.MQT0000000021T01 AND INDEXES ALL;
COMMIT;

```

치수로 맵핑

속성으로 맵핑

사실 및 차원 테이블

사실 및 차원 테이블 조인.
적용가능하면, 눈송이 스키마에서
차원 대 차원 테이블을 조인

큐브 모델의 슬라이스로 맵핑

색인 작성

그림 38. 샘플 SQL 스크립트

둘 이상의 요약 테이블이 큐브 모델에 권장되면 요약 테이블을 작성하는 SQL 스크립트는 각 요약 테이블에 이 명령문 세트를 포함시킵니다.

다음 섹션은 샘플 요약 테이블 작성 SQL 스크립트의 명령문을 설명합니다.

DROP TABLE문

작성할 각 요약 테이블은 해당 이름의 테이블이 아직 존재하지 않도록 먼저 삭제됩니다. 그림 38에서 DB2INFO.MQT0000000021T01 테이블은 DROP TABLE DB2INFO.MQT0000000021T01;문으로 삭제됩니다.

CREATE TABLE문

스크립트는 CREATE TABLE문을 사용하여 요약 테이블을 작성합니다. 이 명령문은 스크립트에서 가장 많은 부분을 차지하고 여기에는 SELECT, FROM, WHERE 및 GROUP BY절이 있는 SELECT문과 갱신 메소드 정의가 들어 있습니다. 요약 테이블

은 적절한 컬럼에서 작성되지만 여기에는 데이터가 들어 있지 않습니다. 요약 테이블을 새로 고치는 SQL 스크립트는 사실 및 차원 테이블에서 데이터를 집계한 후 요약 테이블을 채웁니다.

테이블 이름은 다음과 같이 CREATE TABLE문의 첫 번째 행인 CREATE SUMMARY TABLE DB2INFO.MQT0000000021T01에서 정의됩니다.

128 페이지의 그림 38에 표시된 SELECT절에는 SUM으로 시작하는 다섯 개의 행이 있습니다. 이러한 행은 각각 큐브 모델의 치수 중 하나에 매핑됩니다. 예를 들어, SUM(T1."SALES"-(T1."COGS"+T1."ADVERTISING')) AS "Profit"은 집계 함수 SUM으로 계산된 Profit 치수에 매핑됩니다. 요약 테이블을 작성 중인 큐브 모델에는 Sales, COGS, Advertising, Total expense, Profit과 같은 치수가 들어 있습니다. 계산을 수행하지 않는 컬럼을 선택하는 다음 일곱 개의 행은 속성으로 매핑됩니다. 예를 들어, T5."REGION_NAME" AS "REGION_NAME"은 Region name 속성으로 매핑됩니다. 요약 테이블에는 큐브 모델의 Region name, State name, Line ID, Year, Quarter number, Quarter name 및 Month name과 같은 속성이 들어 있습니다.

FROM절에 있는 테이블은 큐브 모델에서 사용되는 사실 및 차원 테이블입니다. 이 예에서는 SalesFact, Store, Product, Time, Location 및 Line 테이블을 사용합니다.

WHERE절이 사실 및 차원 테이블 사이에 조인을 정의하며 각 조인은 큐브 모델의 조인 오브젝트로 매핑됩니다. 최적화되고 있는 큐브 모델은 snowflake 스키마에 기반하므로 차원-차원 조인에도 WHERE절이 들어 있습니다.

GROUP BY절은 큐브 모델에 정의된 슬라이스로 매핑됩니다. 128 페이지의 그림 38에서는 특정 슬라이스로 매핑되는 하나의 그룹화 세트를 표시합니다. 그룹화에는 슬라이스를 정의하는 다음 메타데이터 유형을 포함할 수 있습니다.

- 해당 슬라이스 레벨에 있는 계층 구조의 레벨 키 속성
- 해당 슬라이스 레벨 상위에 있는 레벨 키 속성
- 레벨 키 속성에 기능적으로 종속되지 않는 관련 속성

요약 테이블이 다중 레벨을 포함할 수 있도록 이 SQL 스크립트 섹션에 GROUPING SETS가 들어 있을 수 있습니다. 큐브 모델에 비분산 치수가 들어 있는 경우 이 SQL 스크립트 섹션에 ROLLUP이 들어 있을 수 있습니다.

표시된 예에서 최적화되고 있는 큐브 모델에는 Market[Region, State, City, Postal code, Store], Product [Family, Line, Product], Time [Year, Quarter, Month, Day] 및 er, Month, Day], and Fiscal Time [Fiscal year, Fiscal quarter, Fiscal month]과 같은 계층 구조가 있습니다. 계층 구조에서 레벨이 그룹화 세트에 포함하지 않으면 슬라이스는 All Time, All Regions 또는 All Products와 같은 최상위 레벨입니다. GROUP BY절에 있는 슬라이스는 State-Line-Month 슬라이스이므로 Region name, Year, Quarter number 및 Quarter name속성이 들어 있습니다. Region Name이 State 레벨

보다 상위에 있고 Year, Quarter 숫자 및 Quarter 이름은 Month 레벨 상위에 있는 속성입니다. 슬라이스의 각 레벨 속성은 SELECT절이 맵핑하는 속성입니다.

CREATE TABLE문의 마지막은 갱신 메소드 정의 부분입니다. 128 페이지의 그림 38에서 CREATE TABLE문의 마지막 세 행은 요약 테이블을 지연 새로 고침으로 설정합니다.

```
DATA INITIALLY DEFERRED
REFRESH DEFERRED
ENABLE QUERY OPTIMIZAITON
MAINTAINED BY SYSTEM
NOT LOGGED INITIALLY;
```

요약 테이블을 지금 새로 고침으로 정의하려는 경우 명령문은 다음과 같습니다.

```
DATA INITIALLY DEFERRED
REFRESH IMMEDIATE
ENABLE QUERY OPTIMIZATION
MAINTAINED BY SYSTEM
NOT LOGGED INITIALLY;
```

CREATE INDEX문

최적화 어드바이저 마법사는 요약 테이블에 하나 이상의 인덱스를 권장하므로 인덱스는 요약 테이블이 작성된 후 작성됩니다. 128 페이지의 그림 38에서 클러스터된 인덱스 및 클러스터되지 않은 인덱스 모두 작성됩니다. 인덱스가 작성된 후 REORG문을 사용하여 클러스터링 인덱스에 따라 테이블을 재구성합니다. 이를 통해 종종 테이블에서 읽기 성능을 향상시킬 수 있습니다.

RUNSTATS문

요약 테이블의 모든 권장사항이 작성되면 RUNSTATS문은 DB2 옵티마이저가 쿼리를 다시 라우팅할 때 요약 테이블 및 인덱스를 고려하는 데 사용하는 DB2 옵티마이저 통계를 갱신합니다.

쿼리 결과 테스트

최적화 어드바이저로 요약 테이블을 작성하기 전과 후에 DB2 Universal Database의 db2batch 벤치마크 도구를 사용하여 쿼리 성능 결과를 벤치마크할 수 있습니다.

쿼리 성능을 테스트하려면 다음을 수행하십시오.

1. 세미콜론으로 분리된 테스트하려는 쿼리가 있는 입력 파일을 작성하십시오.
2. 명령행에서 다음 명령을 입력하십시오.

```
db2batch -d dbname -f file_name -cli
```

여기서 *dbname*은 쿼리를 실행할 데이터베이스를 지정하고 *file_name*은 SQL 쿼리가 있는 입력 파일을 지정하며 *-cli*는 CLI 모드에서의 실행을 지정합니다.

db2batch 도구는 성능 결과를 요약하고 산술 및 기하 평균 모두를 제공합니다. 구분 및 옵션의 경우 명령행에서 db2batch -h를 입력하십시오. db2batch 벤치마크 도구 및 벤치마크 테스트 작성에 대한 자세한 정보는 DB2 정보 센터를 참조하십시오.

권장 요약 테이블을 작성한 후 성능 결과에 만족한 경우 추가 성능 분석을 수행하지 않아도 됩니다.

쿼리 성능이 예상 만큼 향상되지 못한 경우 최적화 어드바이저 마법사를 다시 실행하여 추가 디스크 스페이스 및 시간을 허용할 수 있으며 이전에 사용하지 않은 데이터 샘플링을 사용할 수 있습니다. 추가 디스크 스페이스를 허용하면 성능에 큰 영향을 미칩니다. 요약 테이블에 공간을 많이 제공할수록 더 많이 향상됩니다. 마법사가 데이터 샘플링을 수행하도록 하는 경우 마법사는 더 나은 권장사항을 작성할 수 있습니다. 마찬가지로 마법사가 권장사항을 작성하는 데 드는 시간을 더 많이 허용할수록 권장사항이 보다 나아질 수 있습니다.

쿼리 성능이 전혀 향상되지 않거나 거의 향상되지 않아서 성능 결과에 만족하지 못한 경우 또는 쿼리가 일정 시간 동안 만족스럽게 수행한 후 성능 저하가 나타나는 경우 『요약 테이블 문제점 해결』을 참조하십시오.

요약 테이블 문제점 해결

요약 테이블을 작성한 후 쿼리 성능이 향상되지 않으면 DB2EXPLAIN 기능을 사용하여 쿼리 라우팅 문제점을 해결할 수 있습니다.

DB2EXPLAIN을 사용하여 DB2 UDB가 요약 테이블을 사용하는지 검증하기 전에 다음을 수행해야 합니다.

- 기본 테이블 및 요약 테이블의 통계가 최신 상태인지 검증하십시오.
- 승인없이 수행되고 있는 쿼리를 알지 못하는 경우 이를 식별하십시오. DB2 SQL 스냅샷 모니터를 사용하여 저속 쿼리를 캡처할 수 있습니다.

예상처럼 쿼리가 수행되지 않는 이유를 판별하려면 다음을 수행하십시오.

1. Explain 테이블을 작성하십시오. 데이터베이스에 Explain 테이블을 설정하려면 데이터베이스에 연결하고 \SQLLIB\misc 디렉토리에서 다음 명령을 실행하십시오.

```
db2 -tvf explain.ddl
```
2. Explain 기능을 실행하십시오. Explain 모드가 켜지면 SQL 쿼리가 실행되지 않으며 Explain 명령의 정보 요청만 처리됩니다. 다음 일련의 SQL 명령을 실행하여 Explain 모드를 켜고 요약 테이블이 지연 새로 고침된 경우 DB2 UDB가 이들을 고려하도록 새로 고침 유효 기간을 설정하십시오. 그 다음 쿼리를 실행하고 Explain 모드를 끈 후 Explain 테이블을 쿼리하여 쿼리가 다시 라우트되었는지 확인하십시오.

```

set current explain mode explain

set current refresh age any

SELECT SUM(SALES) FROM MDSAMPLE.SALESFACT

set current explain mode no

SELECT EXPLAIN_TIME, EXPLAIN_LEVEL AS "LEV",
       QUERYNO, STATEMENT_TEXT
FROM EXPLAIN_STATEMENT
WHERE STATEMENT_TEXT LIKE '%SALESFACT%'
ORDER BY EXPLAIN_TIME

```

3. Explain 정보를 보고 재작성한 쿼리가 요약 테이블에 다시 라우트되었는지 점검하십시오. 예를 들어 다음 샘플과 같은 보고서를 볼 수 있습니다.

```

2002-06-30-23.22.12.325002 O 11 SELECT SUM(SALES)
FROM MDSAMPLE.SALESFACT
2002-06-30-23.22.12.325002 P 11 SELECT Q3.$C0
FROM (SELECT SUM(Q2.$C0) FROM (SELECT Q1.SALESFACT_SALES
FROM DB2INFO.MQT0000000021T01 AS Q1) AS Q2) AS Q3

```

하나의 쿼리를 실행하면 두 행이 나타납니다. O이 표시된 행은 DB2 UDB에 전송된 원래 쿼리입니다. P가 표시된 행은 DB2 옵티마이저가 재작성한 쿼리입니다. DB2 옵티마이저가 DB2INFO.MQT0000000021T01 요약 테이블에서 데이터를 선택한 이 예에서 재작성된 쿼리를 볼 수 있습니다.

쿼리가 요약 테이블에 다시 라우트되지만 예상 대로 수행되지 않은 경우 다른 옵션을 사용하여 최적화 어드바이저 마법사를 다시 실행해야 합니다.

쿼리가 요약 테이블에 다시 라우트되지 않은 경우 이유를 판별하고 해당 조치를 취하십시오. 쿼리가 요약 테이블로 다시 라우트되지 않은 이유로 다음이 해당됩니다.

요약 테이블이 존재하지 않음

우선 요약 테이블이 존재하는지 확인하십시오. 존재하지 않는 경우 최적화 어드바이저 마법사를 실행하여 요약 테이블 작성 SQL 스크립트를 생성하십시오. 그 다음 스크립트를 실행하여 요약 테이블을 작성하십시오.

지연 새로 고침 요약 테이블이 만기됨

요약 테이블이 존재하고 지연 새로 고침으로 설정한 경우 새로 고침 유효 기간을 갱신할 필요가 있습니다. (DFT_REFRESH_AGE) = 99999999999999 (ANY)를 설정하여 테이블의 새로 고침 유효 기간을 가능한 길게 설정하며 세션과 독립적으로 설정할 수 있습니다.

쿼리가 요약 테이블에 없는 데이터에 액세스함

쿼리가 요약 테이블에 없는 데이터에 액세스하는 경우 DB2 옵티마이저가 쿼리를 다시 라우트하지 않습니다. 요약 테이블을 작성한 후 새 치수를 추가한 경우 새 치수는 요약 테이블에 존재하지 않습니다. 새 치수를 쿼리하

려는 경우 요약 테이블에 쿼리를 충족시키는 모든 데이터가 들어 있지 않으므로 DB2 옵티마이저가 쿼리를 요약 테이블에 다시 라우트할 수 없습니다.

또한 요약 테이블이 빌드된 큐브 모델 슬라이스 아래에 있는 데이터를 쿼리하려는 경우 요약 테이블을 사용할 수 없습니다. 예를 들어 쿼리가 City 레벨에서 집계된 데이터를 요청하지만 요약 테이블에 State 레벨(City 레벨의 상위 레벨임)에서 집계된 데이터가 들어 있는 경우 쿼리는 요약 테이블을 사용할 수 없습니다.

쿼리에 다시 라우트할 수 없는 구문이 포함됨

DB2 옵티마이저가 일부 복합 쿼리 구문을 사용하는 쿼리를 다시 라우트할 수 없습니다. DB2 옵티마이저가 쿼리를 다시 라우트하지 못하게 방해하는 일부 복합 구문은 다음과 같은 재귀 및 물리적 등록 정보 함수입니다.

- NODENUMBER
- 외부 조인
- 통합
- XMLAGG
- OVER절로 지정한 집계 함수인 창 집계 함수

요약 테이블 유지보수

기본 테이블의 데이터를 변경하는 경우 요약 테이블을 갱신해야 합니다. 지금 새로 고침 또는 지연 새로 고침과 같은 서로 다른 두 가지 방법으로 요약 테이블을 갱신할 수 있습니다.

최적화 어드바이저 마법사를 실행할 때 지금 새로 고침 또는 지연 새로 고침 요약 테이블을 작성하도록 선택합니다. 선택한 항목에 따라 테이블 및 새로 고침 요약 테이블 SQL 스크립트의 갱신 설정에 영향을 줍니다. 두 옵션 모두 보통의 데이터베이스 유지보수 스케줄 일부로 새로 고침 요약 테이블 스크립트를 실행해야 합니다. 새로 고침 스크립트 실행에는 많은 시간과 처리 자원이 필요할 수 있습니다. 유지보수 일괄처리 창에서 갱신을 완료하는 데 충분한 시간을 할당했는지 확인하십시오.

즉시 새로 고침

지금 새로 고침 요약 테이블은 기본 테이블과 거의 동기화된 상태로 보존됩니다. DB2 UDB는 기본 테이블의 변경된 부분에 대응하는 요약 테이블 부분만 변경하여 요약 테이블을 점진적으로 갱신할 수 있도록 기본 테이블의 변경사항을 추적합니다. 요약 테이블 데이터를 기본 테이블과 동기화된 상태로 보존하는 것이 중요한 경우 지금 새로 고침 옵션을 사용하십시오. 예를 들어 기본 테

이블이 매주 Sales 데이터로 갱신되고 사용자는 갱신된 Sales 데이터를 반영하는 매주 보고서를 완료하는 경우 지금 새로 고침을 선택하는 것이 적절할 수 있습니다.

일반적으로 기본 테이블 전체에서 많은 사항을 변경하는 경우 지금 새로 고침을 선택하면 DB2 UDB가 변경사항을 트래킹하는 데 많은 오버헤드를 요구하고 다시 변경사항을 집계하는 데 갱신 명령문을 개별적으로 수행하므로 이 옵션은 최적의 선택사항이 아닐 수 있습니다.

INSERT, UPDATE 및 DELETE와 일반 정규 SQL문을 사용하여 기본 테이블을 갱신하는 경우 DB2 UDB는 기본 테이블을 변경한 후 영향을 받는 요약 테이블을 자동으로 동기화합니다. 하지만 DB2 LOAD 또는 IMPORT 명령을 사용하여 기본 테이블을 갱신하는 경우 갱신을 완료한 후 새로 고침 스크립트를 실행하여 직접 동기화를 트리거해야 합니다.

모든 상황에서 지금 갱신을 사용할 수 없으며 필요한 경우 최적화 어드바이저 마법사가 지연 옵션을 권장할 수도 있습니다.

지연 새로 고침

지연 새로 고침 요약 테이블의 갱신 횟수는 사용자가 요약 테이블을 기본 테이블과 직접 동기화시켜야 하므로 지금 새로 고침보다 더 적습니다. 요약 테이블은 작성 시 데이터의 스냅샷에 기반합니다. 갱신할 때마다 현재 데이터에 기반한 요약 테이블이 재작성되지만 요약 테이블이 마지막으로 작성된 이후 데이터를 변경한 방법은 알 수 없습니다.

대응하는 기본 테이블 전체에서 많은 사항을 변경하는 경우 또는 데이터에 액세스해야 하는 간격보다 더 빠르게 데이터를 갱신하는 경우 지연 새로 고침을 선택하는 것이 좋습니다. 예를 들어 Sales 데이터가 매주 갱신되지만 분기별로만 보고서를 작성해야 하는 경우 지연 새로 고침 옵션을 사용하여 보고서를 실행하기 전에 각 분기마다 요약 테이블을 재빌드할 수 있습니다.

요약 테이블 삭제

DB2 Cube Views는 큐브 모델 삭제 시 연관된 요약 테이블을 삭제하지 않습니다. 기타 목적으로 요약 테이블을 사용하지 않는 경우 테이블을 삭제하여 디스크 여유 공간을 늘릴 수 있습니다.

요약 테이블은 테이블 유형으로 제어 센터 또는 명령행을 사용하여 보통 DB2 프로시저를 통해 삭제할 수 있습니다. 연관된 인덱스도 요약 테이블과 함께 삭제됩니다.

요약 테이블은 DB2INFO 스키마에 정의되어 있습니다. 요약 테이블 이름에 큐브 모델 ID가 들어 있습니다. 예를 들어 요약 테이블의 이름을 DB2INFO.MQT0000000021T01로 지정할 수 있습니다. 여기서 21은 큐브 모델 ID이고 T01은 큐브 모델의 요약 테이블을 고유하게 식별합니다. 큐브 모델 ID 길이는 10자리까지 가능합니다.

명령행에서 요약 테이블을 삭제하려면 `DROP TABLE table_name`을 입력하십시오.

제 6 장 DB2 Cube Views 및 페더레이티드 데이터 소스

이 섹션에서는 다음 주제를 설명합니다.

페더레이티드 시스템에 대한 개요

정보 통합의 엔터프라이즈 레벨 솔루션으로 IBM DB2 Information Integrator를 사용할 수 있습니다. DB2 Information Integrator는 공통 플랫폼에서 데이터 관리 시스템 및 페더레이티드 시스템과 기타 여러 기술을 결합하는 기술 콜렉션입니다.

DB2 Cube Views를 사용한 리모트 데이터 소스의 최적화 개요

DB2 Cube Views를 통해 페더레이티드 스타 스키마 또는 snowflake 스키마를 최적화하여 쿼리 성능을 크게 향상시킬 수 있습니다.

DB2 Cube Views에서 페더레이티드 시스템 사용

DB2 Cube Views에서 페더레이티드 시스템을 사용 가능하게 하려면 리모트 데이터 소스를 정의하고 페더레이티드 서버에서 별칭을 정의하며 해당 별칭에서 정보용 제한조건을 정의해야 합니다.

페더레이티드 데이터 소스의 쿼리를 다시 라우팅하는 문제점 해결

리모트 데이터 소스에 발행된 쿼리가 최적화 이후에 예상한 만큼 향상되지 않은 경우 페더레이티드 시스템이 DB2 Cube Views에 제대로 설정되었는지 확인하십시오.

페더레이티드 시스템에 대한 개요

정보 통합의 엔터프라이즈 레벨 솔루션으로 IBM DB2 Information Integrator를 사용할 수 있습니다. DB2 Information Integrator는 공통 플랫폼에서 데이터 관리 시스템 및 페더레이티드 시스템과 기타 여러 기술을 결합하는 기술 콜렉션입니다.

DB2 페더레이티드 시스템은 DBMS(distributed database management system)의 특수 유형입니다. 정보 통합 기술에 페더레이티드 시스템 측면을 사용하여 다양한 데이터 소스의 다양한 데이터 유형에 액세스할 수 있습니다. 페더레이티드 시스템은 다음 구성 요소로 구성됩니다.

- 페더레이티드 서버로 작동하는 DB2 인스턴스
- 페더레이티드 데이터베이스 역할을 하는 데이터베이스
- 하나 이상의 데이터 소스
- 데이터베이스 및 데이터 소스에 액세스하는 클라이언트(사용자 및 응용프로그램)

페더레이티드 시스템을 사용하여 단일 SQL문에서 분산 요청(DR)을 다중 데이터 소스에 전송할 수 있습니다. 예를 들어 단일 SQL문에서 DB2 Universal Database 테이블, Oracle 테이블 및 XML 태그 파일에 있는 데이터를 조인할 수 있습니다.

페더레이티드 서버

페더레이티드 시스템의 DB2 서버를 페더레이티드 서버라고 합니다. 페더레이티드 서버로 작동하도록 DB2 인스턴스를 임의의 수로 구성할 수 있습니다. 기존 DB2 인스턴스를 페더레이티드 서버로 사용하거나 특별히 페더레이티드 시스템에 새 DB2 인스턴스를 작성할 수 있습니다.

페더레이티드 시스템을 관리하는 DB2 인스턴스가 일반 사용자 및 클라이언트 응용프로그램의 요청에 응답하므로 이를 서버로 취급합니다. 페더레이티드 서버는 종종 수신하는 요청의 일부를 처리하기 위해 데이터 소스에 전송합니다. 푸시다운 조작용 리모트로 처리되는 조작용입니다. 페더레이티드 시스템을 관리하는 DB2 인스턴스가 데이터 소스에 요청을 푸시다운할 때 클라이언트 역할을 하는 경우에도 이를 페더레이티드 서버라고 합니다.

페더레이티드 서버는 래퍼를 통해 데이터 소스와 상호작용합니다. 페더레이티드 서버는 래퍼 모듈이라는 라이브러리에 저장된 루틴을 사용하여 래퍼를 구현합니다. 이러한 루틴을 통해 페더레이티드 서버는 데이터 소스에 연결하고 데이터 소스에서 데이터를 반복적으로 수신하는 것과 같은 조작용을 수행할 수 있습니다. 보통 DB2 페더레이티드 인스턴스 소유자는 CREATE WRAPPER문을 사용하여 페더레이티드 데이터베이스에 래퍼를 등록합니다.

페더레이티드 데이터베이스

일반 사용자 및 클라이언트 응용프로그램의 경우 데이터 소스는 단일 집합 DB2 UDB 데이터베이스로 표시됩니다. 사용자 및 응용프로그램은 페더레이티드 서버가 관리하는 페더레이티드 데이터베이스와 연결됩니다. 페더레이티드 데이터베이스에는 시스템 카탈로그가 들어 있습니다. 페더레이티드 데이터베이스 시스템 카탈로그에는 데이터 소스 및 해당 특성을 식별하는 항목이 들어 있습니다. 페더레이티드 서버는 데이터 소스 래퍼 및 페더레이티드 데이터베이스 시스템 카탈로그에 저장된 정보를 참고하여 SQL문을 처리하는 최상의 계획을 판별합니다.

데이터 소스

페더레이티드 시스템에서 데이터 소스는 관련된 DBMS 인스턴스(예: DB2, Informix, Oracle 또는 Sybase) 또는 관련이 없는 데이터 소스(예: BLAST 검색 알고리즘 또는 XML 태그 파일)일 수 있습니다.

데이터 소스에 액세스하는 데 사용되는 메소드 또는 프로토콜은 데이터 소스 유형에 따라 다릅니다. 예를 들어 DRDA는 z/OS 및 OS/390용 DB2와 같은 DB2 계열 데이터 소스에 액세스하는 데 사용됩니다.

클라이언트

클라이언트로는 사용자가 설정한 페더레이티드 데이터베이스 및 데이터 소스에 액세스하는 사용자 및 응용프로그램을 포함할 수 있습니다. DB2 Cube Views는 페더레이티드 데이터베이스 및 데이터 소스에 액세스하는 클라이언트 역할을 할 수 있는 응용프로그램의 한 가지 예입니다.

DB2 Cube Views를 사용한 리모트 데이터 소스의 최적화 개요

DB2 Cube Views를 통해 페더레이티드 스타 스키마 또는 snowflake 스키마를 최적화하여 쿼리 성능을 크게 향상시킬 수 있습니다.

다음과 같은 혜택이 있습니다.

- 통합 DB2 플랫폼으로 여러 IBM 및 벤더 제품에 액세스함
- Linux, UNIX 및 Windows용 DB2 UDB의 다양한 기능을 z/OS용 DB2 UDB 및 iSeries용 DB2 UDB에 확장시킴
- 페더레이티드 데이터베이스 및 데이터 소스의 쿼리 성능을 향상시킴

페더레이티드 시스템에서 사실 및 차원 테이블은 하나의 리모트 서버 또는 여러 리모트 서버에 있을 수 있습니다. DB2 Cube Views의 스타 스키마 또는 snowflake 스키마를 로컬로 표시해야 합니다. 리모트 테이블, 리모트 테이블의 복제 사본 또는 별칭 및 복제 테이블의 조합을 참조하는 별칭이 있는 스타 스키마 또는 snowflake 스키마를 로컬로 표시할 수 있습니다. 로컬 서버에서 일부 방법으로 스타 스키마 또는 snowflake 스키마의 모든 사실 테이블 및 차원 테이블을 로컬로 표시해야 합니다.

스타 스키마 또는 snowflake 스키마가 로컬 서버에 있으면 DB2 Cube Views를 사용하여 해당 스타 스키마 또는 snowflake 스키마에 기반한 큐브 모델을 빌드한 후 최적화 어드바이저를 사용하여 해당 큐브 모델을 최적화할 수 있습니다. 별칭이 있는 일부 테이블을 로컬로 표시하는 경우 최적화 어드바이저 마법사가 데이터 샘플링을 사용할 수 있도록 지정하면 더 나은 요약 테이블 권장사항을 제공하지만 권장사항을 작성하는데 추가 시간이 필요합니다. 최적화 어드바이저가 모든 상황에서 데이터 샘플링을 사용할 수는 없지만 가능한 경우 데이터 샘플링을 허용하는 것이 좋습니다.

권장 요약 테이블을 로컬 서버에 작성하여 리모트 스타 스키마 또는 snowflake 스키마에 지정된 쿼리에 대해 다음 세 가지 방법 중 하나로 응답할 수 있습니다.

쿼리를 로컬 요약 테이블에 라우트함

요약 테이블의 데이터로 쿼리에 응답할 수 있는 경우 DB2 옵티마이저는 쿼리를 직접 로컬 요약 테이블로 라우트하므로 리모트 테이블과 상호작용하지 않아도 됩니다. 이 라우팅 시나리오는 성능을 가장 많이 향상시킵니다.

중중 요약 테이블만으로 쿼리에 응답할 수 없지만 요약 테이블을 하나 이상의 차원 테이블과 조인하여 응답할 수 있습니다. 이 경우 DB2 옵티마이저가 로컬

테이블에서 조인을 완료할 수 있도록 일부 또는 모든 차원 테이블(사실 테이블은 제외함)을 페더레이티드 서버에 공동 배치하는 작업을 고려할 수 있습니다.

쿼리를 리모트 데이터 소스에 푸시다운함

요약 테이블 및 공동 배치된 차원 테이블의 데이터로 쿼리에 응답할 수 없을 경우 DB2 옵티마이저는 쿼리를 리모트 서버에 푸시다운하려고 합니다. 리모트 서버는 쿼리를 완료하고 결과 세트를 로컬 서버에 리턴합니다.

결과 세트가 작으면 성능을 많이 향상시킬 수 있습니다. 이 메소드는 리모트 스타 스키마 또는 snowflake 스키마의 모든 테이블이 하나의 리모트 서버에만 있는 경우 가능합니다.

쿼리를 리모트 데이터 소스에서 검색된 데이터에서 로컬로 실행함

요약 테이블 및 공동 배치된 차원 테이블의 데이터로 쿼리에 응답할 수 없어서 리모트 데이터 소스에 푸시다운할 수 없는 경우 쿼리를 만족시키는 데 필요한 테이블을 검색하여 리모트 서버에서 로컬 서버로 복사하면 쿼리가 로컬로 완료됩니다. 예를 들어 스타 스키마가 리모트 zSeries 서버에 있고 쿼리가 z/OS용 DB2가 아닌 DB2 UDB에서 사용할 수 있는 기능을 사용하는 경우 리모트 서버에서 쿼리를 완료할 수 없습니다.

쿼리에 응답하는 데 필요한 사실 및 차원 테이블이 클 경우 이 라우팅 시나리오의 성능을 향상시키지 못할 수 있습니다.

DB2 Cube Views에서 페더레이티드 시스템 사용

DB2 Cube Views에서 페더레이티드 시스템을 사용 가능하게 하려면 리모트 데이터 소스를 정의하고 페더레이티드 서버에서 별칭을 정의하며 해당 별칭에서 정보용 제한조건을 정의해야 합니다.

전제조건: DB2 Information Integrator 지원이 사용 가능한지 확인하십시오. 명령행에서 페더레이티드 시스템 지원을 사용 가능하게 하려면 `db2 update dbm cfg using federated yes`를 입력하십시오. 페더레이티드 시스템 지원을 사용 가능하게 한 후 DB2 UDB를 중지 및 시작하십시오.

DB2 Cube Views에서 리모트 데이터 소스를 사용 가능하게 하려면 다음을 수행하십시오.

1. 리모트 데이터 소스를 정의하십시오.
2. 리모트 테이블의 별칭을 정의하십시오.
3. 별칭에서 정보용 제한조건을 정의하십시오.

리모트 데이터 소스 정의

DB2 Cube Views에서 페더레이티드 서버를 정의 및 구성하려면 액세스하려는 리모트 데이터 소스 및 오브젝트에 관한 정보를 페더레이티드 서버에 제공해야 합니다.

리모트 데이터 소스를 정의하려면 다음을 수행하십시오.

1. 페더레이티드 서버가 연결할 리모트 데이터 소스를 알 수 있도록 페더레이티드 서버 데이터베이스 디렉토리의 리모트 데이터베이스 및 이에 대응하는 노드를 카탈로그화하십시오. 구성 지원 프로그램의 데이터베이스 추가 마법사를 사용하여 리모트 데이터베이스 및 이에 대응하는 노드를 카탈로그화할 수 있습니다.
2. 페더레이티드 서버에서 로컬 데이터베이스에 연결하십시오. 기존의 로컬 데이터베이스가 사전에 없는 경우 로컬 데이터베이스를 작성한 후 해당 데이터베이스에 연결하십시오.

3. CREATE WRAPPER 명령을 발행하여 랩퍼를 등록하십시오. 예를 들어 리모트 데이터 소스가 z/OS와 같이 DB2 계열에 속하는 경우 다음 명령을 발행하십시오.

```
CREATE WRAPPER drda LIBRARY 'libdb2drda.a'
```

데이터 소스에 필요한 라이브러리를 알고 있어야 합니다.

4. 액세스하려는 리모트 데이터 소스에서 사용하는 각 서버의 서버 정의를 등록하십시오. 예를 들어 DB2 계열 서버의 drda 랩퍼를 작성하십시오. 다음 명령을 발행하여 리모트 z/OS 서버를 등록하십시오.

```
CREATE SERVER server_name  
  TYPE DB2/ZOS  
  VERSION 8.1  
  WRAPPER DRDA  
  AUTHORIZATION "userid" PASSWORD "password"  
  OPTIONS (DBNAME 'database_name')
```

팁: SQL 스크립트 파일에서 설명한 대로 이 명령을 실행할 수 있습니다. DB2 명령 창에서 이 명령을 직접 발행하려면 큰따옴표로 묶은 명령을 입력하고 사용자가 제공한 사용자 ID 및 암호 앞뒤에 이스케이프(escape) 큰따옴표(\")를 사용하십시오.

```
DB2 "CREATE SERVER server_name  
  TYPE DB2/ZOS  
  VERSION 8.1  
  WRAPPER drda  
  AUTHORIZATION \"userid\" PASSWORD \"password\"  
  OPTIONS (DBNAME 'database_name')"
```

5. 정의한 각 서버의 경우 쿼리 워크로드를 설정하여 항상 페더레이티드 서버에서 리모트 데이터 소스로 항상 푸시다운하십시오. 리모트 데이터 소스의 데이터 크기가 크고 리모트 데이터 소스에서 페더레이티드 서버로 데이터 복사를 시도하는 데 페더레이티드 서버의 DB2 UDB를 사용하지 않습니다. 서버를 최대 푸시다운으로 설정하면 쿼리 결과 세트만이 페더레이티드 서버에 복사되도록 DB2 UDB는 항상 리

모트 데이터 소스에 처음으로 쿼리를 푸시다운하려고 합니다. 이 설정은 DB2 Cube Views 최적화를 통해 성능을 향상시키려고 할 때 필요합니다. 예:

```
CREATE SERVER OPTION DB2_MAXIMAL_PUSHDOWN
FOR SERVER server_name
SETTING 'Y'
```

6. 랩퍼가 서버에 자동 연결될 수 있도록 사용자 매핑을 작성하십시오. 예:

```
CREATE USER MAPPING FOR USER
SERVER server_name
OPTIONS(REMOTE_AUTHID 'user_name', REMOTE_PASSWORD 'password')
```

DB2 Cube Views의 리모트 테이블에 별칭 정의

페더레이티드 서버에서 작성 중인 스타 스키마 또는 snowflake 스키마의 일부인 각 리모트 테이블의 별칭을 정의하십시오. DB2 Cube Views에서는 리모트 테이블에서 작업할 별칭이 필요합니다.

리모트 데이터 소스에 테이블의 별칭을 정의하려면 다음을 수행하십시오.

CREATE NICKNAME문을 사용하여 액세스하려는 각 리모트 테이블에 별칭을 정의하십시오. 예:

```
CREATE NICKNAME local_schema_name.local_nickname
FOR remote_server_name.remote_schema_name.remote_table_name
```

팁: 리모트 서버에 정의한 것과 동일한 로컬 별칭의 스키마 이름을 사용하십시오. 예:

```
CREATE NICKNAME CVSAMPLE.FAMILY FOR remote_server_name.CVSAMPLE.FAMILY
```

DB2 Cube Views의 별칭에서 정보용 제한조건 정의

정보용 제한조건을 정의하여 성능을 향상시킬 수 있는 데이터 간 관계를 문서화하십시오. 페더레이티드 서버에서 스타 스키마 또는 snowflake 스키마의 테이블 간 외부 키의 정보용 제한조건을 정의해야 합니다.

DB2 Cube Views 최적화 어드바이저 마법사에는 성능을 향상시키는 요약 테이블을 권장하는 정보용 제한조건이 필요합니다. 또한 DB2 옵티마이저는 제한조건을 사용하여 쿼리를 보다 효과적으로 처리하고 이를 기존의 요약 테이블에 적절히 라우팅합니다.

DB2 Information Integrator는 리모트 데이터 소스에 있는 기존의 기본 키 제한조건과 일치하는 페더레이티드 서버의 기본 키 제한조건을 자동으로 정의합니다. 페더레이티드 서버에서 스타 스키마 또는 snowflake 스키마를 빌드하는 데 사용하는 외부 키의 정보용 제한조건을 작성해야 합니다.

외부 키를 정의하려면 다음을 수행하십시오.

ALTER NICKNAME문을 사용하여 별칭에 정보 제한조건을 추가하십시오. 예:


```

ALTER NICKNAME local_schema_name.local_nickname
ADD FOREIGN KEY(column_name)
REFERENCES local_schema_name.local_nickname (column_name)
ON DELETE RESTRICT
NOT ENFORCED
ENABLE QUERY OPTIMIZATION

```

DB2 Cube Views에서 리모트 데이터 소스를 사용 가능하게 했습니다.

그 다음 DB2 Cube Views를 사용하여 페더레이티드 서버의 스타 스키마 또는 snowflake 스키마에 완전한 큐브 모델을 빌드하십시오. 완전한 큐브 모델을 빌드한 후 최적화 어드바이저 마법사를 사용하여 큐브 모델을 최적화할 수 있습니다.

리모트 데이터 소스에서 쿼리 성능 문제점 해결

리모트 데이터 소스에 발행된 쿼리가 최적화 이후에 예상한 만큼 향상되지 않은 경우 페더레이티드 시스템이 DB2 Cube Views에 제대로 설정되었는지 확인하십시오.

리모트 데이터 소스로 발행된 쿼리 성능을 향상시키기 전에 다음 단계를 완료해야 합니다.

1. DB2 Cube Views에서 페더레이티드 시스템을 사용하십시오.
2. 42 페이지의 『메타데이터 오브젝트 규칙』에 설명된 기본 규칙, 큐브 모델 완전성 규칙 및 최적화 규칙을 충족시키는 완전한 큐브 모델을 작성하십시오.
3. 큐브 모델을 최적화하십시오.
4. 쿼리 성능이 향상되지 않는 경우 131 페이지의 『요약 테이블 문제점 해결』에 설명된 단계를 점검하십시오.

위의 단계를 완료한 후에도 여전히 쿼리 성능이 향상되지 않으면 다음을 고려하십시오.

- 적용 가능한 모든 제한조건이 정의되었는지 확인하십시오.
- 141 페이지의 『리모트 데이터 소스 정의』의 설명 대로 DB2_MAXIMAL_PUSHDOWN 설정이 예로 설정되었는지 확인하십시오.
- 페더레이티드 서버의 쿼리와 관련된 차원 테이블의 공동 배치를 고려하십시오. 차원 테이블의 복제 사본이 페더레이티드 서버에 존재하도록 차원을 공동 배치하면 성능이 향상될 수 있습니다.

제 7 장 DB2 Cube Views API

DB2 Cube Views API 개요

DB2 Cube Views는 DB2 Cube Views에 저장된 메타데이터에 프로그램을 통해 액세스하는 API(Application Programming Interface)를 제공합니다. API를 사용하는 응용프로그램은 DB2 Cube Views 메타데이터 오브젝트를 사용하여 관계형 테이블 및 조인과 상호작용하지 않아도 메타데이터와 상호작용할 수 있습니다.

DB2 Cube Views API는 DB2 데이터베이스의 시스템 카탈로그 테이블에 저장된 메타데이터에 액세스합니다. API를 사용하는 응용프로그램은 데이터 웨어하우스에 있는 다차원 구조 및 OLAP 구조를 모델화하는 메타데이터 오브젝트를 작성 및 수정할 수 있습니다.

그림 39에서는 API를 통해 데이터 및 메타데이터가 교환되는 방법을 표시합니다.

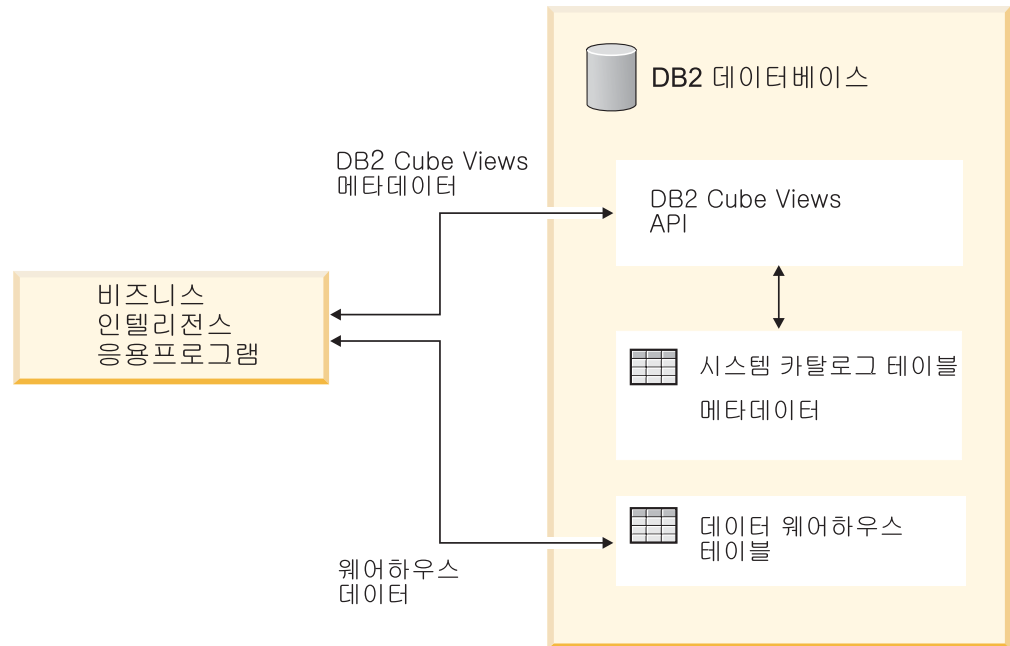


그림 39. DB2 Cube Views API를 통한 데이터 교환

API는 DB2 데이터베이스에 등록된 단일 스토어드 프로시저입니다. 이 스토어드 프로시저는 복합 메타데이터 및 메타데이터 조작을 표현하는 데 사용하는 입력 및 출력 매개변수를 승인합니다. API 매개변수 형식은 XML 스키마로 정의됩니다.

API는 다음 기술을 사용하여 DB2 Cube Views와 비즈니스 인텔리전스 응용프로그램 사이에서 메타데이터를 교환합니다.

- ODBC가 있는 SQL
- DB2 CLI 및 JDBC
- XML

DB2 Cube Views API: DB2 스토어드 프로시저 및 XML 구문 분석

DB2 Cube Views API는 DB2 Cube Views 스토어드 프로시저 및 XML 구문 분석을 통해 비즈니스 인텔리전스 응용프로그램과 DB2 데이터베이스 사이에서 메타데이터를 교환합니다.

DB2 스토어드 프로시저로 프로그램을 작성하는 방법에 관한 정보는 *DB2 응용프로그램 개발 안내서*를 참조하십시오. DB2 Cube Views API로 프로그램을 작성하기 전에 다음 개념을 이해해야 합니다.

트랜잭션

DB2 UDB는 트랜잭션의 다중 사용자 메타데이터 액세스를 지원합니다. 자세한 정보는 *DB2 응용프로그램 개발 안내서* 참조하십시오. DB2 Cube Views API로 수행되는 모든 데이터베이스 조치는 호출 응용프로그램의 데이터베이스 트랜잭션에 속합니다. 따라서 API가 데이터베이스 작업 단위를 관리하는 md_message 스토어드 프로시저를 호출한 후 응용프로그램이 COMMIT 또는 ROLLBACK을 실행할 수 있습니다.

메모리 관리

매개변수는 CLOB 구조 양식의 md_message 스토어드 프로시저와 응용프로그램 사이에서 교환됩니다. md_message 스토어드 프로시저를 호출하는 응용프로그램은 스토어드 프로시저를 카탈로그하는 데 사용되는 구조와 동일한 크기의 CLOB 매개변수 구조를 사전에 할당해야 합니다. API는 DB2 UDB의 CLOB 최대 크기를 2GB까지 지원합니다. 디폴트 CLOB 크기는 1MB입니다.

시스템 구성

대형 매개변수의 교환을 지원하려면 다음 DB2 UDB 설정값을 변경해야 할 수도 있습니다.

- md_message 스토어드 프로시저를 호출하는 데이터베이스 클라이언트 응용프로그램은 더 큰 힙(heap) 및 스택 크기를 사용하여 링크되어야 할 수도 있습니다.
- query_heap_sz 설정값을 사용하여 데이터베이스의 DB2 쿼리 힙 크기를 증가시켜야 할 수도 있습니다.

XML 구문 분석

API를 사용하는 응용프로그램은 md_message 스토어드 프로시저에서 리턴하는 출력 매개변수의 구문을 분석해야 합니다. API를 사용하려는 개발자는 다양한 XML 구문 분석기를 사용할 수 있습니다.

오류 조절

API에서는 오류 정보를 다음과 같이 세 가지 양식으로 생성합니다.

- 스토어드 프로시저에서 호출 응용프로그램에 리턴하는 SQLCODE 및 SQLSTATE 정보
- **response** API 매개변수를 사용하여 호출 응용프로그램에 전달되는 XML 구조
- API를 실행하는 데이터베이스 서버에 있는 오류 및 런타임 로그 파일

XML 유효성 확인, 구문 분석 또는 태그 수행으로 오류가 발생하면 **response** 매개변수는 조작 태그 대신 <error> 태그와 함께 호출 응용프로그램에 리턴됩니다. <error> XML 태그에는 API에서 발견한 문제점을 설명하는 메시지 및 리턴 코드와 함께 <status> 태그가 들어 있습니다.

XML 처리와 무관한 API에서 오류가 발생하지만 이 오류가 메타데이터 조작의 실행과 관련된 경우 **response** 매개변수의 콘텐츠가 리턴됩니다.

다음 예에서는 <error> 태그의 정보 유형을 표시합니다. 이 예에 표시된 매개변수 구조의 설명에서는 XML 태그의 제한된 수만 사용합니다. 대부분의 매개변수에는 여기에 표시된 수보다 더 많은 태그가 있으며 매개변수 콘텐츠의 유효성은 XML 스키마를 통해 확인됩니다.

```
<olap:response xmlns:olap="http://www.ibm.com/olap"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" version="8.1.2.1.0">
<error>
<status id="3100" text="The system failed to parse XML for
  &quot;INPUT PARAMETER&quot; (line:&quot;3&quot;;,
  char:&quot;26&quot;;, message:&quot;Unknown element
  'dropa'&quot;). " type="error"/>
</error>
</olap:response>
```

DB2 Cube Views 스토어드 프로시저가 호출되면 스토어드 프로시저가 실제로 실행되었는지와 상관 없이 DB2 UDB는 SQLCODE 및 SQLSTATE를 호출 응용프로그램에 리턴합니다. DB2 Cube Views 스토어드 프로시저를 실행할 수 있는 경우 스토어드 프로시저는 호출 응용프로그램에 전송된 XML 데이터의 일부로 상태 메시지를 리턴합니다.

DB2 Cube Views 스토어드 프로시저

이 스토어드 프로시저는 `md_message`라고 하며 DB2 Cube Views 매개변수 형식으로 표현되는 매개변수를 처리합니다.

이 프로시저는 입력 매개변수에서 조작 및 메타데이터 정보를 추출한 후 요청한 메타데이터 조작을 수행합니다. 프로시저는 조작에 따라 요청한 조작 및 메타데이터 정보의 실행 상태(성공 또는 실패)가 들어 있는 출력 매개변수를 생성합니다.

스토어드 프로시저는 UNIX 시스템의 `.fenced` 파일에 지정된 사용자로 실행됩니다. 분리(`fenced`) 사용자에게 `db2md_config.xml` 파일에 지정된 로그 파일에 대한 WRITE 액세스가 있어야 합니다. 그렇지 않으면 스토어드 프로시저는 아무 것도 로그할 수 없습니다.

DB2 Cube Views 스토어드 프로시저는 DB2 UDB 스토어드 프로시저로 구현됩니다. DB2 UDB에서 임의의 프로그래밍 인터페이스를 사용하는 모든 응용프로그램에서 이를 사용할 수 있습니다. 스토어드 프로시저의 이름에서는 대소문자를 구분하지 않지만 스토어드 프로시저 매개변수의 이름 및 컨텐츠에서는 대소문자를 구분합니다. `md_message` 및 프로토타입의 구문은 다음과 같습니다.

```
Syntax:      call md_message (request, metadata, response)
Prototype:  md_message (request IN  CLOB(1M),
                       metadata INOUT CLOB(1M),
                       response OUT  CLOB(1M))
```

request, **metadata** 및 **response** 매개변수는 DB2 UDB 데이터 유형인 문자 대형 오브젝트(CLOB) 유형입니다. 응용프로그램은 수행할 조작의 설명으로 **request** 매개변수를 채우고 선택적으로 조작을 수행할 메타데이터로 **metadata** 매개변수를 채울 수 있습니다. 여기서 입력 매개변수를 구현하면 `md_message`는 **response** 매개변수의 조작 상태 및 **metadata** 매개변수의 요청 메타데이터를 리턴합니다. **metadata** 매개변수는 메타데이터의 입력 및 출력 모두에 사용됩니다. DB2 UDB는 데이터베이스 서버의 `md_message` 스토어드 프로시저 및 비즈니스 인텔리전스 응용프로그램 사이에서 매개변수 구조를 전송합니다.

CLOB 인수의 크기는 다양합니다. 디폴트 크기는 1MB입니다. CLOB 인수 크기는 `sqllib/misc/db2mdapi.sql` 스크립트에서 지정됩니다. CLOB 매개변수 크기가 최대 2GB 인 스토어드 프로시저를 다시 카탈로그할 수 있습니다. 매개변수의 크기를 늘리면 스토어드 프로시저를 시작할 때 출력 매개변수 버퍼가 카탈로그된 크기로 사전 할당되므로 런타임에서 스토어드 프로시저가 추가 메모리를 사용합니다. 크기가 너무 작으면 입력 및 출력 매개변수의 데이터가 절단됩니다.

디폴트 CLOB 크기를 변경하려면 스토어드 프로시저 API를 증가된 크기 한계로 다시 등록하십시오.

1. sqllib/misc/db2mdapi.sql 파일을 편집하고 CREATE PROCEDURE문의 크기 한계를 늘리십시오.

```
CREATE PROCEDURE
  DB2INFO.MD_MESSAGE(IN   request CLOB(1M),
                    INOUT metadata CLOB(10M),
                    OUT   response CLOB(10M))
```

2. db2mdapi.sql 파일을 다시 실행하여 스토어드 프로시저 API를 다시 등록하십시오. 스크립트에서 이미 존재하는 테이블을 작성하려고 할 때 발생하는 모든 오류를 무시하십시오.

C++에서 스토어드 프로시저 API를 호출하는 방법에 관한 정보는 sqllib/samples/olap/client/db2mdapiclient.cpp 파일의 C++ 소스 코드 샘플을 참조하십시오.

다음 예에서는 Embedded SQL 응용프로그램에서 DB2 Cube Views 스토어드 프로시저를 호출하는 방법을 표시합니다.

```
// Standard declarations
// ...

// Include the Communication Area to access error details
EXEC SQL INCLUDE SQLCA;

// SQL declarations of host variables that will be used for calling the
// DB2 Cube Views stored procedure
EXEC SQL BEGIN DECLARE SECTION;

// Allocate CLOB for the request parameter
SQL TYPE IS CLOB(1M)      request;

// Allocate CLOB for the metadata parameter
SQL TYPE IS CLOB(1M)     metadata;

// Allocate CLOB for the response parameter
SQL TYPE IS CLOB(1M)    response;

EXEC SQL END DECLARE SECTION;

// Connect to database and other application initializations
// ...

// Populate the request parameter structure with the operation
strcpy(request.data, "<request><describe> ... </describe></request>");

// string length with end-of-string
request.length = strlen(request.data) + 1;

// Populate the metadata parameter structure with the metadata
strcpy(metadata.data, "");

// string length with EOS
metadata.length = strlen(metadata.data) + 1;

// Call DB2 Cube Views stored procedure
EXEC SQL CALL "DB2INFO.MD_MESSAGE"(:request,:metadata,:response);
```

```

// Check that the stored procedure has returned without errors
if (sqlca.sqlcode)
{
// error checking using sqlaintp()
}

// Process response parameter structure to determine success of operation
// ...

// Process metadata parameter structure to extract requested metadata
// ...

// Disconnect from database and other application terminations
// ...

```

DB2 Cube Views API 매개변수

DB2 Cube Views API 메타데이터 조작 매개변수

DB2 Cube Views의 API는 검색, 수정 및 관리와 같은 세 가지 유형의 메타데이터 기능을 제공합니다. 각 기능 유형에는 하나 이상의 조작이 들어 있으며 각 조작에는 매개변수 세트가 있습니다.

매개변수 형식은 메타데이터 조작 및 오브젝트를 표시하고 이를 비즈니스 인텔리전스 응용프로그램과 DB2 Cube Views 사이에서 교환하는 표준을 정의합니다. 매개변수 형식으로 XML을 사용하여 DB2 Cube Views 메타데이터 조작 및 오브젝트를 표시합니다. XML 스키마는 매개변수 형식을 정의합니다.

입력 및 출력 매개변수

스토어드 프로시저 `md_message`의 각 조작에는 두 개의 입력 및 두 개의 출력 매개변수가 있습니다.

입력 매개변수는 요청 매개변수 및 메타데이터 매개변수입니다. 출력 매개변수는 응답 매개변수 및 메타데이터 매개변수입니다.

입력	매개변수 콘텐츠
요청	스토어드 프로시저에서 요청한 조작이 들어 있습니다. 요청에는 요청 동작 및 범위에 영향을 미치는 옵션이 포함될 수 있습니다.
메타데이터	요청 매개변수에 설명된 조작에서 사용할 메타데이터 오브젝트가 들어 있습니다. 설명 조작과 같은 일부 요청 조작에서는 입력 메타데이터가 필요하지 않습니다.
출력	
응답	메타데이터 오브젝트를 제외한 스토어드 프로시저에서 수행한 모든 조작 결과가 들어 있습니다.

입력	매개변수 콘텐츠
메타데이터	요청 입력 매개변수에 설명된 조작에서 요청한 메타데이터 오브젝트가 들어 있습니다.
	작성 조작과 같은 일부 조작에서는 출력 메타데이터를 리턴하지 않습니다.

그림 40에서는 요청된 응용프로그램의 입력 매개변수가 DB2 Cube Views API를 지나 출력 매개변수로 전송되는 방법을 표시합니다. API는 DB2 UDB가 읽을 수 있도록 표시되는 메타데이터 오브젝트를 작성합니다.

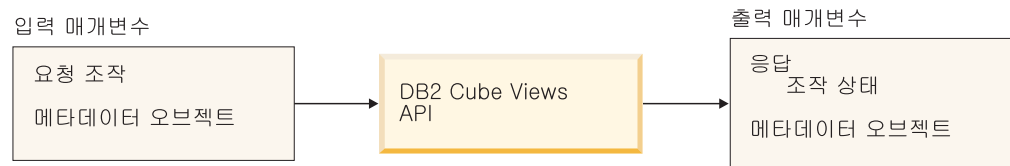


그림 40. API 매개변수

DB2 Cube Views 메타데이터 조작

검색 조작: 설명

DB2 Cube Views에는 하나의 검색 조작(설명)이 들어 있습니다.

설명 조작은 XML 버전 8.1.2.1.0 및 8.2.0.1.0 모두를 지원합니다. 기타 모든 조작에는 XML 버전 8.2.0.1.0이 필요합니다. XML 스키마 파일 버전에 관한 자세한 정보는 178 페이지의 『DB2 Cube Views 메타데이터 테이블 및 XML 스키마 파일』을 참조하십시오.

설명

이 조작은 메타데이터 오브젝트 정보를 검색합니다. 이 조작은 지정된 *objectType*의 메타데이터 오브젝트 하나 이상에 관한 정보를 리턴합니다(예: 단일 차원 오브젝트, 차원 오브젝트 세트, 모든 오브젝트 유형이 들어 있는 오브젝트 세트). 메타데이터 오브젝트는 메타데이터 매개변수에서 리턴됩니다. 검색한 메타데이터 오브젝트 표시에 사용하는 형식은 180 페이지의 『메타데이터 오브젝트 형식』에 설명되어 있습니다. 또한 이 조작은 반복 모드에서 *objectType* 스펙을 충족시키는 오브젝트 세트에서 시작하는 오브젝트 연관 경로를 통과할 때 기타 모든 메타데이터 오브젝트에 관한 정보를 리턴합니다.

152 페이지의 표 39에서는 설명 요청 매개변수의 구성요소를 나열합니다. 152 페이지의 표 40에서는 설명 응답 매개변수의 구성요소를 나열합니다.

표 39. 설명 요청 매개변수

이름	유형	데이터 유형	올바른 값	설명
objectType	인수	XML 문자열	all cubeModel cube cubeFacts cubeDimension cubeHierachy cubeLevel dimension facts hierarchy attributeRelationship level join measure attribute	검색 가능한 DB2 Cube Views 메타 데이터 오브젝트 유형입니다. 이 매개 변수에 유효한 값 중 하나를 지정할 수 있습니다.
restriction [선택사항]	인수	XML 문자열	167 페이지의 『조작 매개변수』를 참조하십시오.	메타데이터 검색 범위를 제한합니다. 이는 SQL의 술어와 유사합니다.
recurse	인수	XML 문자열	yes no	메타데이터 오브젝트의 반복 검색을 사용 가능하게 하거나 사용 불가능하게 합니다.

표 40. 설명 응답 매개변수

이름	유형	데이터 유형	올바른 값	설명
상태	상태 메시지	DB2 Cube Views 메시지 구조	172 페이지의 『메시지 구조』 섹션을 참조하십시오.	설명 조작의 상태를 표시하는 메시지입니다.
object	검색 결과	XML	171 페이지의 『조작 피연산자』를 참조하십시오.	요청된 메타데이터 데이터 오브젝트. 조작 도중 오류가 발생하면 이 값이 빈 문자열이 됩니다.

권고 조작: 권고

이 조작은 지정된 큐브 모델의 쿼리 성능을 향상시키기 위해 빌드해야 하는 요약 테이블을 권장합니다. 이 조작에는 요약 테이블에서 어드바이저가 실행될 수 있는 시간 및 사용할 수 있는 디스크 스페이스 크기를 제한하는 인수가 있습니다.

표 41. 권고 요청 매개변수

이름	유형	데이터 유형	올바른 값	설명
cubeModelRef	인수	XML 요소		최적화할 큐브 모델입니다.
tablespaceName [선택사항]	인수	XML 문자열		요약 테이블의 테이블 스페이스입니다. 테이블 스페이스가 제공되지 않으면 생성된 SQL은 테이블 스페이스를 지정하지 않습니다. 이 경우 DB2 UDB는 테이블을 디폴트 테이블 스페이스에 배치합니다.

표 41. 권고 요청 매개변수 (계속)

이름	유형	데이터 유형	올바른 값	설명
indexspaceName [선택사항]	인수	XML 문자열		요약 테이블 인덱스의 테이블 스페이스입니다. 테이블 스페이스가 제공되지 않으면 생성된 SQL은 테이블 스페이스를 지정하지 않습니다. 이 경우 DB2 UDB는 인덱스를 디폴트 테이블 스페이스에 배치합니다.
diskspaceLimit [선택사항]	인수	XML 음이 아닌 정수		요약 테이블 및 해당 인덱스에서 사용 가능한 디스크 스페이스(바이트 단위)입니다. 0을 지정하면 어드바이저가 사용할 디스크 스페이스 크기를 결정함을 의미합니다. 어드바이저는 사용자가 추가 디스크 스페이스를 제공할 때 더 나은 권장사항을 제시합니다. 0을 지정하면 스페이스가 무제한임을 의미합니다. 디폴트값은 0입니다.
timeLimit [선택사항]	인수	XML 음이 아닌 정수		쿼리를 권장하는 데 사용해야 하는 시간의 크기(초 단위)입니다. 어드바이저는 사용자가 추가 시간을 제공할 때 더 나은 권장사항을 제시합니다. 0을 지정하면 시간 크기가 무제한임을 의미하며 어드바이저는 시간을 더 추가해도 권장사항이 더 나아지지 않음을 판별하면 중지됩니다. 디폴트값은 0입니다.
sampling [선택사항]	인수	XML 문자열	yes 또는 no	큐브 모델 기본 테이블의 데이터 샘플링을 수행해야 하는지를 지정합니다. sampling을 사용하면 일반적으로 권장사항이 향상되지만 어드바이저의 실행 시간을 늘리지는 않습니다. sampling을 허용하지 않으면 어드바이저는 데이터베이스 통계에 기반한 권장사항만 작성합니다. 어드바이저는 데이터 샘플링을 사용하여 데이터베이스 통계 및 샘플링을 모두 이용합니다. 원래 데이터의 소형 Replica가 있으면 DB2 UDB 통계를 변경하여 테이블을 원래 테이블 크기로 표시할 수 있습니다. 샘플링이 지정되지 않으면 어드바이저는 샘플링 없이 원래 테이블에 작성한 것과 동일한 권장사항을 작성합니다.

표 41. 권고 요청 매개변수 (계속)

이름	유형	데이터 유형	올바른 값	설명
refresh [선택사항]	인수	XML 문자열	deferred immediate	또는 기본 테이블이 변경된 경우 기본 테이블과 요약 테이블을 동기화시키기 위해 DB2 UDB가 요약 테이블을 지금 새로 고침할 것인지를 지정합니다. 지연 새로 고침의 경우 요약 테이블을 새로 고치는 시기를 지정해야 함을 의미합니다. 지금 새로 고침의 경우 DB2는 기본 테이블의 갱신과 동시에 요약 테이블을 갱신합니다. 기본 테이블이 데이터 로드를 사용하여 갱신되면 직접 새로 고침을 지정해야 합니다. 지연 새로 고침 요약 테이블은 완전히 다시 빌드되지만 지금 새로 고침 요약 테이블이 점진적으로 갱신됩니다. 지금 새로 고침을 사용하는 경우 제한사항이 많으므로 어드바이저는 사용자가 지금을 지정해도 지연 새로 고침을 사용하도록 선택할 수 있습니다.

다음 예는 권고 조작을 표시합니다.

```
<olap:request xmlns:olap="http://www.ibm.com/olap" ... >
<advise
  tablespaceName="TS_MQTTABLE"
  indexspaceName="TS_MQTINDEX"
  diskSpaceLimit="75000"
  timeLimit="300"
  sampling="yes"
  refresh="immediate">
<cubeModelRef name="SalesModel" schema="MDSAMPLE"/>
</advise>
</olap:request>
```

표 42. 권고 응답 매개변수

이름	유형	데이터 유형	올바른 값	설명
status	상태 메시지	DB2 Cube Views 메시지 구조		권고 조작의 상태를 표시하는 메시지입니다.
info	메시지 목록	메시지 구조의 목록		권고 요청에 지정한 내용에서 권고 조작을 수행한 결과로 얻은 가능한 변경사항 및 발생한 치명적이지 않은 조건을 표시하는 경고 메시지입니다. 예를 들어 권고 조작은 권장 MQT에 분산되지 않는 치수 때문에 REFRESH IMMEDIATE MQT의 해당 스펙을 겹쳐쓸 수 있습니다. 정보 메시지는 모델의 일부가 최적화되지 않는 조건을 표시하거나 집계 및 인덱스를 선택한 이유를 포함하여 수행된 최적화에 관한 정보를 표시합니다.

표 42. 권고 응답 매개변수 (계속)

이름	유형	데이터 유형	올바른 값	설명
권장사항		XML 요소		모든 권장 요약 테이블 및 인덱스에 필요한 전체 디스크 스페이스의 추정된 크기입니다.
sql		XML 요소		요약 테이블을 작성하고 채운 후 여기에 인덱스를 작성하는 SQL입니다. 사용자가 이 SQL을 실행해야 합니다.
refreshSql		XML 요소		갱신된 기본 테이블과 동기화하기 위해 요약 테이블을 새로 고치는 SQL입니다. 사용자가 이 SQL을 실행해야 합니다. 지금 새로 고침 요약 테이블의 경우 DB2 UDB는 종종 기본 테이블이 갱신될 때 요약 테이블을 자동으로 갱신합니다. 이 조치는 기본 테이블이 변경되는 방식에 따라 다릅니다.

다음 예는 권고 응답 조작을 표시합니다.

```

<olap:response xmlns:olap="http://www.ibm.com/olap" ... >
<advise>
<status id="0" text="Operation completed successfully.
No errors were encountered." type="informational"/>
<info>
<message id="7401" text="The DB2INFO.MQT0000000041T01 summary table
is recommended.
It is estimated to have 100 rows, 55KB table size and 5KB index size."/>
<message id="7401" text="The DB2INFO.MQT0000000041T02 summary table
is recommended.
It is estimated to have 8 rows, 4KB table size and 1KB index size."/>
<message id="7406" text="The PART dimension does not have any
hierarchies that can be optimized by the Optimization Advisor.
The recommendations will not optimize for any attributes
from this dimension."/>
</info>
<recommendation diskpace="65108"/>
<sql>
<![CDATA[
-- *****
-- * Script to create/refresh summary tables.
-- *
-- * Cube model schema: MDSAMPLE
-- * Cube model name: SalesModel
-- * Diskspace limit: 75000
-- * Time limit: 300
-- * Sampling: Yes
-- * Refresh type: Refresh immediate
-- * Tablespace name: TS_MQTTABLE
-- * Indexspace name: TS_MQTINDEX
-- *****

DROP TABLE DB2INFO.MQT0000000041T01;

DROP TABLE DB2INFO.MQT0000000041T02;

```

```

UPDATE COMMAND OPTIONS USING c OFF;

CREATE SUMMARY TABLE DB2INFO.MQT0000000041T01 AS
(SELECT
SUM(T2."COGS") AS "COGS",
SUM(T2."MARKETING"+T2."PAYROLL") AS "EXPENSE",
SUM(T2."MARKETING") AS "MARKETING",
SUM(T2."PAYROLL") AS "PAYROLL",
SUM(T2."SALES"-(T2."COGS"+(T2."MARKETING"+T2."PAYROLL"))) AS "PROFIT",
SUM(T2."SALES") AS "SALES",
COUNT(*) AS "COUNT",
T5."REGION" AS "REGION",
T5."DIRECTOR" AS "DIRECTOR",
T6."FAMILY" AS "FAMILY",
T6."FAMILYNAME" AS "FAMILYNAME",
T3."SKU" AS "SKU",
T3."CAFFEINATED" AS "CAFFEINATED",
T3."OUNCES" AS "OUNCES",
T3."PKGTYPE" AS "PKGTYPE",
T3."SKUNAME" AS "SKUNAME",
T4."YEAR" AS "YEAR"

FROM
"MDSAMPLE"."MARKET" AS T1,
"MDSAMPLE"."SALESFACT" AS T2,
"MDSAMPLE"."PRODUCT" AS T3,
"MDSAMPLE"."TIME" AS T4,
"MDSAMPLE"."REGION" AS T5,
"MDSAMPLE"."FAMILY" AS T6

WHERE
T1."STATEID"=T2."STATEID" AND
T3."PRODUCTID"=T2."PRODUCTID" AND
T4."TIMEID"=T2."TIMEID" AND
T1."REGIONID"=T5."REGIONID" AND
T3."FAMILYID"=T6."FAMILYID"

GROUP BY
T5."REGION",
T5."DIRECTOR",
T6."FAMILY",
T6."FAMILYNAME",
T3."SKU",
T3."CAFFEINATED",
T3."OUNCES",
T3."PKGTYPE",
T3."SKUNAME",
T4."YEAR")

DATA INITIALLY DEFERRED
REFRESH IMMEDIATE
IN "TS_MQTTABLE"
INDEX IN "TS_MQTINDEX"
NOT LOGGED INITIALLY;

COMMENT ON TABLE DB2INFO.MQT0000000041T01 IS 'AST created for
cube mode1 MDSAMPLE.SalesModel';

```

```

REFRESH TABLE DB2INFO.MQT0000000041T01;

CREATE INDEX DB2INFO.IDX0000000041T0101 ON DB2INFO.MQT0000000041T01("FAMILY",
"SKU");
CREATE INDEX DB2INFO.IDX0000000041T0102 ON DB2INFO.MQT0000000041T01("REGION");

RUNSTATS ON TABLE DB2INFO.MQT0000000041T01 AND INDEXES ALL;

CREATE SUMMARY TABLE DB2INFO.MQT0000000041T02 AS
(SELECT
SUM(T2."COGS") AS "COGS",
SUM(T2."MARKETING"+T2."PAYROLL") AS "EXPENSE",
SUM(T2."MARKETING") AS "MARKETING",
SUM(T2."PAYROLL") AS "PAYROLL",
SUM(T2."SALES"-(T2."COGS"+(T2."MARKETING"+T2."PAYROLL"))) AS "PROFIT",
SUM(T2."SALES") AS "SALES",
COUNT(*) AS "COUNT",
T1."YEAR" AS "YEAR",
T1."QUARTER" AS "QUARTER"

FROM
"MDSAMPLE"."TIME" AS T1,
"MDSAMPLE"."SALESFACT" AS T2

WHERE
T1."TIMEID"=T2."TIMEID"

GROUP BY
T1."YEAR",
T1."QUARTER")

DATA INITIALLY DEFERRED
REFRESH IMMEDIATE
IN "TS_MQTTABLE"
INDEX IN "TS_MQTINDEX"
NOT LOGGED INITIALLY;

COMMENT ON TABLE DB2INFO.MQT0000000041T02 IS 'AST created for
cube model MDSAMPLE.SalesModel';

REFRESH TABLE DB2INFO.MQT0000000041T02;

CREATE INDEX DB2INFO.IDX0000000041T02C ON DB2INFO.MQT0000000041T02("QUARTER")
CLUSTER;

REORG TABLE DB2INFO.MQT0000000041T02;

RUNSTATS ON TABLE DB2INFO.MQT0000000041T02 AND INDEXES ALL;

COMMIT;

]]>
</sql>
<refreshSql>
<![CDATA[
-- *****
-- * Script to create/refresh summary tables.

```

```

-- *
-- * Cube model schema: MDSAMPLE
-- * Cube model name: SalesModel
-- * Diskspace limit: 75000
-- * Time limit: 300
-- * Sampling: Yes
-- * Refresh type: Refresh immediate
-- * Tablespace name: TS_MQTTABLE
-- * Indexspace name: TS_MQTINDEX
-- *****

UPDATE COMMAND OPTIONS USING c OFF;

REFRESH TABLE DB2INFO.MQT0000000041T01;
REFRESH TABLE DB2INFO.MQT0000000041T02;
COMMIT;

]]>
</refreshSql>
</advise>
</olap:response>

```

수정 조작: 변경, 작성, 삭제, импорт 및 이름 바꾸기

DB2 Cube Views API는 다음과 같이 다섯 개의 수정 조작을 포함합니다.

- 변경
- 작성
- 삭제
- импорт
- 이름 바꾸기

스토어드 프로시저에서 수정 조작을 구현하는 경우 오브젝트가 완전한지 및 참조용으로 유효한지도 확인합니다.

작성

작성 조작은 메타데이터 오브젝트를 작성합니다. 하나 이상의 메타데이터 오브젝트 피연산자를 승인하고 오브젝트가 조작에 전송된 순서 대로 DB2 Cube Views에서 해당 오브젝트를 작성합니다. 이 조작에 전송된 오브젝트의 시퀀스에는 다른 오브젝트 유형이 들어 있을 수 있습니다.

이 조작에 전송된 오브젝트는 다른 오브젝트를 선택적으로 참조할 수 있습니다. 오브젝트 사이에 참조가 존재하는 경우 이들은 오브젝트의 정렬 순서로 정렬되어야 합니다. 예를 들어 MyObject 오브젝트가 YourObject 오브젝트를 참조하는 경우 YourObject는 MyObject에 앞서 조작에 전송되어야 합니다. 메타데이터 오브젝트가 서로를 참조할 수 있는 방법에 관한 자세한 정보는 180 페이지의 『메타데이터 오브젝트 형식』을 참조하십시오.

작성 조작용은 각 오브젝트의 유효성을 확인합니다. 작성된 오브젝트가 이미 존재하는 경우 또는 작성되는 오브젝트에서 참조하는 오브젝트가 아직 존재하지 않는 경우 오류가 리턴됩니다. 입력 오브젝트가 존재하지 않는 스키마를 지정하는 경우 사용자가 데이터베이스에 충분한 권한을 가지고 있으면 해당 조작용을 통해 스키마를 작성합니다.

표 43. 작성 요청 매개변수

이름	유형	데이터 유형	올바른 값	설명
object	피연산자	XML 요소	171 페이지의 『조작 피연산자』를 참조하십시오.	작성될 수 있는 오브젝트입니다.

표 44. 작성 응답 매개변수

이름	유형	데이터 유형	올바른 값	설명
status	상태 메시지	DB2 Cube Views 메시지 구조	172 페이지의 『메시지 구조』 섹션을 참조하십시오.	작성 조작용의 상태를 표시하는 메시지입니다.

변경

변경 조작용은 메타데이터 오브젝트 정보를 갱신합니다. 하나 이상의 메타데이터 오브젝트 피연산자를 승인하고 메타데이터 카탈로그 테이블의 해당 오브젝트 사본을 갱신합니다. 오브젝트는 조작용에 전송된 순서 대로 갱신됩니다. 이 조작용에 전송된 오브젝트의 시퀀스에는 다른 오브젝트 유형이 들어 있을 수 있습니다.

이 조작용에 전송된 오브젝트는 다른 오브젝트를 선택적으로 참조할 수 있습니다. 오브젝트 사이에 참조가 존재하는 경우 이들은 오브젝트의 정렬 순서로 정렬되어야 합니다. 메타데이터 오브젝트가 서로를 참조할 수 있는 방법에 관한 자세한 정보는 180 페이지의 『메타데이터 오브젝트 형식』을 참조하십시오.

이 조작용으로는 오브젝트의 이름 또는 스키마를 갱신할 수 없습니다. 오브젝트 이름은 이름 바꾸기 조작용으로 변경될 수 있습니다.

변경 조작용은 각 오브젝트의 유효성을 확인합니다. 갱신된 오브젝트가 존재하지 않는 경우 오류가 리턴됩니다.

표 45. 변경 요청 매개변수

이름	유형	데이터 유형	올바른 값	설명
object	피연산자	XML 요소	171 페이지의 『조작 피연산자』를 참조하십시오.	갱신될 수 있는 오브젝트입니다.

표 46. 변경 응답 매개변수

이름	유형	데이터 유형	올바른 값	설명
status	상태 메시지	DB2 Cube Views 메시지 구조	172 페이지의 『메시지 구조』 섹션을 참조하십시오.	변경 조작용의 상태를 표시하는 메시지입니다.

이름 바꾸기

이름 바꾸기 조작은 현재 스키마 및 이름으로 식별되는 단일 DB2 Cube Views 메타데이터 오브젝트의 이름을 바꿉니다. 오브젝트 이름만 변경될 수 있습니다. 오브젝트의 스키마는 변경될 수 없습니다. 이름 바꾸기 조작은 오브젝트가 현재 다른 메타데이터 오브젝트에서 참조되는 경우에도 해당 오브젝트의 이름을 바꿉니다.

표 47. 이름 바꾸기 요청 매개변수

이름	유형	데이터 유형	올바른 값	설명
objectType	인수	XML 문자열	cubeModel cube cubeFacts cubeDimension cubeHierachy cubeLevel dimension facts hierarchy attributeRelationship level join measure attribute	이름이 바뀌는 DB2 Cube Views 메타데이터 오브젝트의 유형입니다. 이 매개변수에 유효한 값 중 하나를 지정할 수 있습니다.
currentRef	피연산자	DB2 Cube Views 메타데이터 오브젝트 참조	171 페이지의 『조작 피연산자』를 참조하십시오.	이름이 바뀌는 메타데이터 오브젝트의 현재 스키마 및 이름입니다.
newRef	피연산자	DB2 Cube Views 메타데이터 오브젝트 참조	171 페이지의 『조작 피연산자』를 참조하십시오.	이름이 바뀐 메타데이터 오브젝트의 새 스키마 및 이름입니다.

표 48. 이름 바꾸기 응답 매개변수

이름	유형	데이터 유형	올바른 값	설명
status	상태 메시지	DB2 Cube Views 메시지 구조	172 페이지의 『메시지 구조』 섹션을 참조하십시오.	이름 바꾸기 조작의 상태를 표시하는 메시지입니다.

삭제

삭제 조작은 DB2 Cube Views에서 메타데이터 오브젝트를 삭제합니다. 이 조작은 지정된 *objectType* 및 *restriction* 구성요소에 따라 하나 이상의 메타데이터 오브젝트를 삭제합니다. 삭제된 오브젝트가 현재 다른 메타데이터 오브젝트에서 참조되는 경우 오류가 리턴됩니다.

표 49. 삭제 요청 매개변수

이름	유형	데이터 유형	올바른 값	설명
objectType	인수	XML 문자열	167 페이지의 『조작 매개변수』를 참조하십시오.	삭제할 메타데이터 오브젝트의 유형입니다. 이 매개변수에 유효한 값 중 하나를 지정할 수 있습니다.
restriction [선택사항]	인수	XML 문자열	167 페이지의 『조작 매개변수』를 참조하십시오.	메타데이터 삭제 범위를 제한합니다. 이는 SQL의 술어와 유사합니다.

표 50. 삭제 응답 매개변수

이름	유형	데이터 유형	올바른 값	설명
status	상태 메시지	DB2 Cube Views 메시지 구조	172 페이지의 『메시지 구조』 섹션을 참조하십시오.	삭제 조작의 상태를 표시하는 메시지입니다.

임포트

임포트 조작은 메타데이터 오브젝트를 작성하거나 메타데이터 카탈로그에 있는 메타데이터 오브젝트 존재를 보고합니다. 이 조작은 미리 존재하는 메타데이터 오브젝트의 존재를 처리하는 방식을 제외하고 작성 조작과 유사하게 작동합니다.

임포트 조작에서 선택적 모드를 정의할 수 있습니다. 이러한 서로 다른 모드에 따라 카탈로그에 이미 존재하는 오브젝트와 이름이 동일한 오브젝트를 임포트하려고 할 때 취할 조치를 판별합니다.

실행할 모드에 따라 작성된 오브젝트가 이미 존재하는 경우 또는 작성된 오브젝트에서 참조하는 오브젝트가 아직 존재하지 않는 경우 오류가 리턴됩니다. 입력 오브젝트가 존재하지 않는 스키마를 지정하는 경우 사용자가 데이터베이스에 충분한 권한을 가지고 있으면 임포트 조작을 통해 스키마를 작성합니다.

임포트 조작은 각 오브젝트의 유효성을 확인합니다.

다양한 조작 모드에 관한 자세한 설명은 167 페이지의 『조작 매개변수』를 참조하십시오.

표 51. 임포트 요청 매개변수

이름	유형	데이터 유형	올바른 값	설명
mode	인수	XML 문자열	create new - ignore collisions create new - replace collisions create new - abort on collision report new - report collisions	임포트된 기존 오브젝트 및 새 오브젝트에서 취할 조치를 정의합니다. 167 페이지의 『조작 매개변수』의 <i>mode</i> 에 관한 설명을 참조하십시오.
object	피연산자	XML 요소	171 페이지의 『조작 피연산자』를 참조하십시오.	임포트될 오브젝트입니다.

표 52. 임포트 응답 매개변수

이름	유형	데이터 유형	올바른 값	설명
status	상태 메시지	DB2 Cube Views 메시지 구조	172 페이지의 『메시지 구조』 섹션을 참조하십시오.	임포트 조작의 상태를 표시하는 메시지입니다.
newList	참조 목록	XML 요소	167 페이지의 『조작 매개변수』의 <i>mode</i> 에 관한 설명을 참조하십시오.	새 오브젝트를 참조하는 이름 스키마 쌍의 목록입니다.
collisionList	참조 목록	XML 요소	167 페이지의 『조작 매개변수』의 <i>mode</i> 에 관한 설명을 참조하십시오.	다른 오브젝트와 충돌하는 오브젝트를 참조하는 이름 스키마 쌍의 목록입니다.

API가 수정 조작 수행 시 함수 종속성을 처리하는 방법

DB2 Cube Views 메타데이터 API는 특정 규칙에 따라 레벨 오브젝트의 함수 종속성을 관리합니다. 표 53에서는 API가 각 레벨 수정 요청 시 취하는 조치를 표시합니다.

표 53. 함수 종속성에 대한 API 조치

요청	조치
레벨 작성	가능한 API는 이에 대응하는 DB2 함수 종속성을 작성합니다. 함수 종속성을 작성할 수 없는 경우 API는 함수 종속성이 없는 레벨을 작성하고 경고 메시지를 리턴합니다.
레벨 삭제	API가 레벨 및 이와 연관된 DB2 함수 종속성을 삭제합니다. API가 레벨 또는 함수 종속성을 삭제하려는 도중 오류가 발생하는 경우 API는 오류 메시지를 리턴하며 레벨 오브젝트는 삭제되지 않습니다.
함수 종속성이 없는 레벨 변경	레벨을 변경할 때 함수 종속성을 작성하지 않으면 조치를 취하지 않습니다. 레벨을 변경할 때 함수 종속성을 작성하면 가능한 API는 이에 대응하는 DB2 함수 종속성을 작성합니다. 함수 종속성을 작성할 수 없는 경우 API는 함수 종속성이 없는 레벨을 변경하며 경고 메시지를 리턴합니다.

표 53. 함수 종속성에 대한 API 조치 (계속)

요청	조치
함수 종속성이 있는 레벨 변경	레벨에 함수 종속성이 있고 이 레벨을 변경할 때 함수 종속성을 삭제하려는 경우 API는 레벨을 변경하고 이와 연관된 DB2 함수 종속성을 삭제합니다. API가 레벨을 변경하려는 도중 오류가 발생하는 경우 API는 오류 메시지를 리턴하며 레벨 오브젝트는 변경되지 않습니다. 레벨을 변경하여 함수 종속성을 유지보수하려는 경우 API는 함수 종속성을 삭제한 후 이를 재작성합니다. 함수 종속성을 삭제할 수 없는 경우 API는 오류 메시지를 리턴하지만 계속해서 레벨을 변경하고 이름이 다른 새 함수 종속성을 재작성합니다.
레벨(새로 작성: 충돌 모 드 무시) импорт	가능한 API는 이에 대응하는 DB2 함수 종속성을 작성합니다. 함수 종속성을 작성할 수 없는 경우 API는 함수 종속성이 없는 레벨을 작성하고 경고 메시지를 리턴합니다.
레벨(새로 작성: 충돌 모 드 대체) импорт	레벨(새로 작성: 충돌 모 드 대체)와 동일합니다.

관리 조작: 유효성 확인 및 변환

DB2 Cube Views에는 두 개의 관리 조작(유효성 확인 및 변환)이 들어 있습니다. DB2 Cube Views 버전 8.1에서 DB2 Cube Views 버전 8.2로 이주할 때에만 변환 조작을 사용합니다. 변환 조작은 버전 8.1 메타데이터 XML을 버전 8.2 메타데이터 XML로 맵핑합니다.

유효성 확인

유효성 확인 조작은 하나 이상의 메타데이터 오브젝트에서 유효성을 확인합니다. 오브젝트가 DB2 Cube Views 오브젝트 규칙을 준수하는지에 따라 유효성이 정의됩니다. 이 조작으로 유효성이 확인되는 오브젝트에서 *objectType* 인수 및 *restriction* 매개변수를 지정하십시오. *mode* 매개변수를 사용하여 유효성 확인 범위를 지정하십시오.

유효성 확인 조작에서는 다음을 점검합니다.

- 메타데이터 오브젝트 정보의 완전성
- 메타데이터 오브젝트 간의 참조 무결성
- 참조되는 관계형 테이블, 보기, 별명 및 별칭 컬럼의 존재 여부
- 속성 및 치수와 같은 메타데이터 오브젝트에 저장된 SQL 표현식의 정확성

유효성 확인 조작은 유효하지 않은 메타데이터 오브젝트를 찾으면 중지됩니다. 위반을 찾으면 이 조작은 유효성 확인 위반을 설명하는 메시지를 리턴합니다. 기타 조작(작성, 변경 및 импорт)도 내재적으로 메타데이터 오브젝트의 유효성을 확인합니다. 변환 조작은 메타데이터 오브젝트의 유효성을 확인하지 않습니다.

표 54. 유효성 확인 요청 매개변수

이름	유형	데이터 유형	올바른 값	설명
objectType	인수	XML 문자열	167 페이지의 『조작 매개변수』를 참조하십시오.	유효성을 확인하고 있는 DB2 Cube Views 메타데이터 오브젝트의 유형입니다. 이 매개변수에 유효한 값 중 하나를 지정할 수 있습니다.
restriction (선택적)	인수 [선택사항]	XML 문자열	167 페이지의 『조작 매개변수』를 참조하십시오.	메타데이터 유효성 확인 범위를 제한합니다. 이는 SQL의 술어와 유사합니다.
mode	인수	XML 문자열	base cubeModel completeness 최적화	수행할 유효성 확인 조치의 범위를 정의합니다.

표 55. 유효성 확인 응답 매개변수

이름	유형	데이터 유형	올바른 값	설명
status	상태 메시지	DB2 Cube Views 메시지 구조	172 페이지의 『메시지 구조』 섹션을 참조하십시오.	유효성 확인 조작의 상태를 표시하는 메시지를 제공합니다.
info	메시지 목록	메시지 구조의 목록	172 페이지의 『메시지 구조』 섹션을 참조하십시오.	유효성 확인 조작으로 생성되는 경고 및 오류를 설명하는 메시지 목록입니다.

변환

변환 조작은 XML 스키마의 유효성을 확인하여 들어오는 메타데이터 XML의 구문이 올바른지를 점검하지만 컬럼 또는 기타 참조의 유효성을 확인하지는 않습니다. 변환 조작을 사용하여 임의의 DB2 Cube Views 메타데이터를 변환할 수 있습니다. 변환 조작을 수행하려면 XML 문서가 필요합니다. XML 문서에 없는 오브젝트를 참조하면 오류가 발생할 수 있습니다.

변환 조작은 다음 테이블에 표시된 대로 오브젝트를 맵핑합니다. 버전 8.1의 모든 오브젝트는 별도로 언급되지 않으면 버전 8.2의 동일한 오브젝트로 맵핑됩니다.

표 56. 버전 8.1 오브젝트가 버전 8.2 오브젝트로 맵핑되는 방법

버전 8.1 오브젝트	버전 8.2 오브젝트
속성	<ul style="list-style-type: none"> 속성은 새 등록 정보 널(NULL) 입력 기능을 갖도록 확장됩니다. 기존 속성은 알 수 없음과 같은 널(NULL) 입력 기능을 갖게 됩니다.
조인	변경 없음
속성 관계	<ul style="list-style-type: none"> 계층 구조 및 큐브 계층 구조에서 더 이상 참조하지 않습니다. 모든 속성 관계는 상위 항목이 삭제된 오브젝트가 됩니다. 나머지 모든 오브젝트는 이러한 속성 관계를 참조하지 않습니다. <p>OLAP 센터에서 큐브 모델 또는 큐브를 익스포트할 때 익스포트한 XML 파일의 속성 관계를 더 이상 참조하지 않습니다. OLAP 센터에서 속성 관계를 익스포트하려면 모든 메타데이터를 익스포트해야 합니다.</p>

표 56. 버전 8.1 오브젝트가 버전 8.2 오브젝트로 맵핑되는 방법 (계속)

버전 8.1 오브젝트	버전 8.2 오브젝트
계층 구조 + 속성 + 속성 관계	<ul style="list-style-type: none"> 버전 8.1 계층 구조에서 참조하는 속성마다 정확히 하나의 레벨 오브젝트만 작성됩니다. 레벨 오브젝트는 버전 8.1 소스 속성과 동일한 이름(스키마, 이름 등)을 사용합니다. 레벨의 디폴트 속성은 소스 속성입니다. 레벨의 키는 버전 8.1 계층 구조 중 하나에 기반을 둔 소스 속성의 상위에 있는 상위 구성원 속성을 더한 소스 속성으로 구성됩니다. 소스 속성이 둘 이상의 버전 8.1 계층 구조에서 참조되면 상위 구성원이 가장 적은 계층 구조가 선택되고(연결된 경우 임의의 계층 구조를 선택함) 상위 구성원 속성이 버전 8.2 레벨의 키 속성으로 위부터 아래의 소스에 순차적으로 추가됩니다. 왼쪽 속성이 소스 속성과 동등한 계층 구조에서 참조하는 모든 버전 8.1 속성 관계는 해당 레벨에서 관련 속성을 작성하는 데 사용됩니다. 왼쪽 속성과 일치하는 각 속성 관계의 경우 오른쪽에 있는 속성은 관련 속성으로 레벨에 추가됩니다. 관련 속성 목록이 작성되면 중복된 사항은 제거됩니다.
큐브 계층 구조 + 속성 + 속성 관계	<ul style="list-style-type: none"> 각 버전 8.1 큐브 계층 구조에서 참조하는 속성마다 고유한 큐브 레벨 오브젝트가 작성됩니다. 큐브 레벨 이름은 속성 이름 및 큐브 이름에 기반하여 지정됩니다. 이름이 Sales인 큐브가 이름이 Region인 속성을 가진 경우 큐브 레벨 이름은 Region(Sales)로 지정됩니다. 각 큐브 레벨은 이에 대응하는 레벨 오브젝트를 참조합니다. 예를 들어 Region (Sales)는 Region 레벨을 지정합니다. 큐브 레벨의 관련 속성은 지정된 큐브 계층 구조의 속성 관계 오브젝트에 기반합니다. 왼쪽 속성이 상위 레벨의 소스 속성과 동등한 모든 속성 관계의 경우 오른쪽에 있는 속성이 관련 속성으로 큐브 레벨에 추가됩니다.
계층 구조	버전 8.1의 각 계층 구조는 레벨 세트(속성 및 속성 관계가 없음)가 정렬된 버전 8.2 계층 구조로 맵핑됩니다.
큐브 계층 구조	버전 8.1의 각 계층 구조는 큐브 레벨 세트(속성 및 속성 관계가 없음)가 정렬된 버전 8.2 큐브 계층 구조로 맵핑됩니다.
차원	연관된 레벨이 추가됩니다.
큐브 차원	변경 없음
치수	속성과 같이 치수는 새 등록 정보 널(NULL) 입력 기능을 갖도록 확장됩니다. 기존 치수는 알 수 없음과 같은 널(NULL) 입력 기능을 갖게 됩니다.
사실	변경 없음
큐브 사실	변경 없음
큐브 모델	변경 없음
큐브	변환 중 변경되지는 않지만 버전 8.2에서 한 큐브에 사용 정보가 들어 있을 수 있습니다.

메타데이터 조작의 샘플 입력 및 출력 매개변수

다음 샘플에서는 세 가지 유형의 메타데이터 조작에서 매개변수를 구조화하는 방법을 표시합니다. 이 예에서는 XML 구조 부분이 제외되지만 이 부분은 생략 부호(.)로 표시됩니다..

검색 조작

다음 샘플에서는 설명이라고 하는 검색 조작을 구조화하는 방법을 표시합니다. 설명 조작에 대한 자세한 정보는 151 페이지의 『검색 조작: 설명』을 참조하십시오. 다음 예에서 메타데이터 매개변수는 입력 시 빈 상태지만 출력 시 데이터로 채워져 있습니다.

요청 및 메타데이터

```
<olap:request xmlns:olap="http://www.ibm.com/olap" ... >
  <describe objectType="cube" recurse="no">
    <restriction>
      <predicate property="schema" operator="=" value="myschema"/>
    </restriction>
  </describe>
</olap:request>

<olap:metadata xmlns:olap="http://www.ibm.com/olap" ... />
```

응답 및 메타데이터

```
<olap:response xmlns:olap="http://www.ibm.com/olap" ... >
  <describe>
    <status id="0" text="Operation completed
      successfully." type="informational"/>
  </describe>
</olap:response>

<olap:metadata xmlns:olap="http://www.ibm.com/olap" ... >
  <cube name="cube1" schema="myschema" ... > ... </cube>
  ...
  <cube name="cubeN" schema="myschema" ... > ... </cube>
</olap:metadata>
```

수정 조작

다음 샘플은 작성이라고 하는 수정 조작을 구조화하는 방법을 표시합니다. 작성 조작 및 기타 수정 조작에 대한 자세한 정보는 158 페이지의 『수정 조작: 변경, 작성, 삭제, импорт 및 이름 바꾸기』를 참조하십시오. 메타데이터 매개변수는 입력 시 데이터가 채워진 상태지만 출력 시 빈 상태입니다.

요청 및 메타데이터

```
<olap:request xmlns:olap="http://www.ibm.com/olap" ... >
  <create/>
</olap:request>

<olap:metadata xmlns:olap="http://www.ibm.com/olap" ... >
  <attribute name="LocationID" ... > ... </attribute>
  <attribute name="Country" ... > ... </attribute>
  <attribute name="State" ... > ... </attribute>
  <attribute name="City" ... > ... </attribute>
  <dimension name="Location" ... type="regular">
    <attributeRef name="LocationID" ... </attributeRef>
    <attributeRef name="Country" ... </attributeRef>
    <attributeRef name="State" ... </attributeRef>
```



```

        <attributeRef name="City" ... </attributeRef>
        ...
    </dimension>
</olap:metadata>

```

응답 및 메타데이터

```

<olap:response xmlns:olap="http://www.ibm.com/olap" ... >
  <create>
    <status id="0" text="Operation completed
      successfully." type="informational"/>
  </create>
</olap:response>

```

```

b<olap:metadata xmlns:olap="http://www.ibm.com/olap" ... >

```

관리 조작

다음 예에서는 유효성 확인이라고 하는 관리 조작을 구조화하는 방법을 표시합니다. 유효성 확인 조작에 대한 자세한 정보는 163 페이지의 『관리 조작: 유효성 확인 및 변환』을 참조하십시오.

요청 및 메타데이터

```

<olap:request xmlns:olap="http://www.ibm.com/olap" ... >
  <validate objectType="cube" mode="base">
    <restriction>
      <predicate property="schema" operator="=" value="myschema"/>
    </restriction>
  </describe>
</olap:request>

```

```

<olap:metadata xmlns:olap="http://www.ibm.com/olap" ... />

```

응답 및 메타데이터

```

<olap:response xmlns:olap="http://www.ibm.com/olap" ... >
  <validate>
    <status id="1" text="...Additional information
      returned." type="informational"/>
    <info><message id="6299" text="At least one
      database view was found during validation."
      type="warning"/></info>
  </validate>
</olap:response>

```

```

<olap:metadata xmlns:olap="http://www.ibm.com/olap" ... >

```

db2mdapiclient 유틸리티에서 사용할 수 있는 추가 조작 XML 샘플 파일은 SQLLIB\samples\olap\xml\input 디렉토리에 있습니다.

조작 매개변수

각 메타데이터 조작에서 다양한 매개변수를 사용할 수 있습니다. 이러한 매개변수는 조작 동작을 특정 응용프로그램에 맞게 조정합니다.

DB2 Cube Views에서는 메타데이터 조작 시 5개의 매개변수를 제공합니다.

- objectType
- recurse
- restriction
- mode(임포트 조작용)
- mode(유효성 확인 조작용)

objectType 매개변수

이 매개변수는 요청된 조작과 관련된 메타데이터 오브젝트의 유형을 지정합니다. 다음 오브젝트 유형은 DB2 Cube Views 메타데이터 오브젝트 모델과 직접 대응합니다.

- all
- cubeModel
- cube
- cubeDimension
- cubeFacts
- cubeHierarchy
- cubeLevel
- dimension
- facts
- hierarchy
- attributeRelationship
- level
- join
- measure
- attribute

recurse 매개변수

이 매개변수는 조작을 반복적으로 수행하는지를 제어합니다. 반복되지 않는 모드의 경우 objectType 매개변수 및 **restriction** 매개변수 스펙과 직접 일치하는 메타데이터 오브젝트에서만 조작은 해당 조치를 수행합니다. 조작이 메타데이터 오브젝트의 반복되지 않는 모드 세트에서 시작되어 오브젝트의 반복되지 않는 모드 세트에서 시작하는 오브젝트 연관 경로를 통과할 때 기타 모든 메타데이터 오브젝트에서 해당 조치를 추가로 수행합니다. **recurse** 매개변수에는 예 옵션 및 아니오 옵션이 들어 있습니다.

예를 들어 반복되지 않는 조작성 차원 목록을 리턴하지만 반복 조작성 차원 목록 및 해당 차원에서 참조하는 기타 모든 오브젝트(서로 다른 유형임)와 해당 차원 오브젝트에서 참조하는 오브젝트를 리턴할 수 있습니다.

restriction 매개변수

이 매개변수는 메타데이터 조작성 범위가 제한 또는 한정되도록 지정합니다. 이 매개변수는 SQL 쿼리의 술어와 유사합니다. 제한사항은 DB2 Cube Views XML 스키마에서 정의한 <restriction> 및 <predicate> 태그를 사용하여 XML에 표시됩니다.

제한사항은 모든 메타데이터 오브젝트에 공통된 오브젝트 등록 정보 및 메타데이터 오브젝트 간 관계에 기반을 둘 수 있습니다.

등록 정보 기반의 술어에는 다음 속성이 들어 있습니다.

등록 정보

술어 태그와 연관되며 이름 속성 또는 스키마 속성을 지정해야 합니다.

연산자 술어 태그와 연관되며 등호(=)를 지정해야 합니다.

값 술어 태그와 연관되며 등록 정보 속성에서 지정한 등록 정보와 비교할 문자열 표시 값입니다.

restriction 매개변수를 조작 단계의 전체 시퀀스와 관련시키는 방법에 관한 설명은 173 페이지의 『조작 단계 시퀀스』를 참조하십시오.

이 예에서는 조작 범위를 ABC 스키마의 오브젝트로 제한합니다.

```
<restriction>  
  <predicate property="schema" operator="=" value="ABC">  
</restriction>
```

mode(임포트용) 매개변수

이 매개변수는 임포트 조작에서 사용할 모드를 설정합니다. 다음 테이블은 사용 가능한 모드를 설명합니다.

입력으로 임포트 조작에 전송된 오브젝트가 메타데이터 카탈로그에 이미 존재하는 경우 충돌이 발생합니다.

표 57. импорт 모드

모드	설명	리턴된 참조 목록
새로 작성: 충돌 무시	<ul style="list-style-type: none"> 충돌하지 않는 입력 오브젝트가 작성됩니다. 충돌하는 입력 오브젝트는 작성되지 않습니다. 사전에 존재하는 오브젝트는 변경되지 않습니다. 충돌로 오류가 생성되지 않습니다. 	<p>newList</p> <p>제대로 작성된 오브젝트의 이름 및 스키마 쌍이 들어 있습니다.</p> <p>collisionList</p> <p>충돌하는 오브젝트(무시되어 작성되지 않음)의 이름 및 스키마 쌍이 들어 있습니다.</p>
새로 작성: 충돌 대체	<ul style="list-style-type: none"> 충돌하지 않는 입력 오브젝트가 작성됩니다. 충돌하는 입력 오브젝트가 사전에 존재하는 오브젝트를 대체합니다. 사전에 존재하는 오브젝트가 입력 오브젝트로 대체됩니다. 충돌로 오류가 생성되지 않습니다. 	<p>newList</p> <p>제대로 작성된 오브젝트의 이름 및 스키마 쌍이 들어 있습니다.</p> <p>collisionList</p> <p>충돌하는 오브젝트(대체됨)의 이름 및 스키마 쌍이 들어 있습니다.</p>
새로 작성: 충돌 시 중단	<ul style="list-style-type: none"> 입력 오브젝트는 조작 전체에 충돌이 없을 때만 작성됩니다. 충돌 시 조작의 일부로 오브젝트가 작성되지 않습니다. 사전에 존재하는 오브젝트는 변경되지 않습니다. 충돌로 오류가 생성됩니다. 	<p>newList</p> <p>제대로 작성된 오브젝트 또는 충돌하지 않은 오브젝트(작성되지 않음)의 이름 및 스키마 쌍이 들어 있습니다.</p> <p>collisionList</p> <p>충돌하는 오브젝트(작성되지 않음)의 이름 및 스키마 쌍이 들어 있습니다.</p>
새로 작성: 충돌 보고	<ul style="list-style-type: none"> 오브젝트가 작성되지 않습니다. 입력 오브젝트의 충돌 상태를 보고합니다. 사전에 존재하는 오브젝트는 변경되지 않습니다. 충돌로 오류가 생성되지 않습니다. 	<p>newList</p> <p>충돌하지 않는 오브젝트(작성 및 작성되지 않음)의 이름 및 스키마 쌍이 들어 있습니다.</p> <p>collisionList</p> <p>충돌하는 오브젝트(작성되지 않음)의 이름 및 스키마 쌍이 들어 있습니다.</p>

newList 및 collisionList 참조 목록의 태그는 사전 정의된 순서를 준수해야 합니다. 다음 목록은 IMPORT 조작의 참조 유형 순서를 표시합니다.

1. <attributeRef>
2. <joinRef>
3. <attributeRelationshipRef>
4. <levelRef>
5. <cubeLevelRef>

6. <hierarchyRef>
7. <cubeHierarchyRef>
8. <dimensionRef>
9. <cubeDimensionRef>
10. <measureRef>
11. <factsRef>
12. <cubeFactsRef>
13. <cubeModelRef>
14. <cubeRef>

mode(유효성 확인용) 매개변수

이 매개변수는 유효성 확인 조작에서 사용할 모드를 설정합니다. 다음 테이블은 유효성 확인 조작에 사용 가능한 모드를 설명합니다. 다음 테이블의 규칙은 DB2 Cube Views 오브젝트 규칙 범주를 참조합니다.

표 58. 유효성 확인 모드

모드	설명
기본	기본 규칙을 준수하는지 점검
cubeModel 완전성	<ul style="list-style-type: none"> • 큐브 모델 완전성 규칙을 준수하는지 점검 • 기본 규칙을 준수하는지 점검
최적화	<ul style="list-style-type: none"> • 최적화 규칙을 준수하는지 점검 • 큐브 모델 완전성 규칙을 준수하는지 점검 • 기본 규칙을 준수하는지 점검

조작 피연산자

조작에 요청을 동반하는 메타데이터 오브젝트 또는 해당 참조가 필요한 경우 이러한 오브젝트 또는 참조를 조작의 "피연산자"라고 합니다.

요청 또는 메타데이터 매개변수를 사용하여 메타데이터 조작에 전송되는 피연산자는 다음과 같습니다.

object 이 피연산자에는 작동하는 메타데이터 오브젝트가 들어 있습니다. 메타데이터 오브젝트 표시에 사용하는 형식은 180 페이지의 『메타데이터 오브젝트 형식』에 설명되어 있습니다.

currentRef

이 피연산자는 이름 바꾸기 조작에 사용되며 이 조작에는 메타데이터 오브젝트의 스키마 및 이름이 들어 있습니다.

newRef

currentRef 피연산자와 유사한 이 피연산자는 이름 바꾸기 조작에 사용하고 이 조작에는 메타데이터 오브젝트의 스키마 및 이름이 들어 있습니다.

메시지 구조

DB2 Cube Views API는 특정 구조를 갖는 정보용, 경고 및 오류 메시지를 리턴합니다.

다음 테이블에서 DB2 Cube Views 메시지의 구성요소를 설명합니다.

표 59. 메시지 구성요소

구성요소	설명
ID	메시지의 고유한 정수 ID입니다.
유형	메시지는 다음 세 가지 유형 중 하나일 수 있습니다. <ul style="list-style-type: none">• 정보용• 경고• 오류
텍스트	메시지 텍스트가 들어 있는 문자열입니다.
토큰	메시지의 텍스트 문자열로 대체되는 값입니다. 메시지는 임의 수의 토큰을 포함할 수 있습니다. 다음 XML 요소가 메시지에 토큰으로 나타날 수 있습니다. <ul style="list-style-type: none">• attributeRef• joinRef• attributeRelationshipRef• levelRef• cubeLevelRef• hierarchyRef• cubeHierarchyRef• dimensionRef• cubeDimensionRef• measureRef• factsRef• cubeFactsRef• cubeModelRef• cubeRef• column• text

다음은 토큰이 없는 메시지의 예입니다.

```
<status id="0" text="Operation completed successfully."type="informational"/>
```

다음은 토큰이 있는 메시지의 예입니다.

```
<status id="6331" text="The left attribute for
the &quot;MDOBJ_ID_ATTRIBUTERELATIONSHIP.MDSAMPLE.State_PopGroup&quot;
attribute relationship is not a part of the
&quot;MDOBJ_HIERARCHY.MDSAMPLE.RegionState&quot; hierarchy."
type="error">
<tokens>
<attributeRelationshipRef name="State_PopGroup" schema="MDSAMPLE"/>
<text value="MDOBJ_HIERARCHY.MDSAMPLE.RegionState"/>
</tokens>
</status><
```

조작 단계 시퀀스

세 개의 조작 인수만 조작 범위를 판별합니다.

다음 세 개의 인수는 조작에 적용되는 순서 대로 나열됩니다.

1. objectType
2. restriction
3. recurse

다음 예에서는 의도한 제한사항과 명확히 일치하지 않는 일부 오브젝트를 리턴할 수도 있지만 이를 조작의 recurse 단계의 일부로 리턴하는 방법을 표시합니다.

예: myschema 스키마에 속하는 큐브를 반복적으로 설명합니다.

조작 인수:

```
objectType = "cube"
restriction = <restriction>
    <predicate property="schema" operator="=" value="myschema"/>
</restriction>
recurse = "yes"
```

설명 조작은 해당 범위를 큐브 오브젝트로 제한하여 시작합니다. 이러한 큐브 오브젝트 중 myschema 스키마에 속한 오브젝트만 선택됩니다. 선택된 큐브 오브젝트 각각의 경우 이들이 참조하는 오브젝트가 선택됩니다. 이 오브젝트가 서로 다른 유형이며 내재적으로 다른 스키마입니다. 그 다음 선택된 모든 오브젝트가 큐브 요청에 대한 응답의 일부로 리턴됩니다.

로그 및 추적

DB2 Cube Views API의 런타임 추적

API는 세 가지의 추적 우선순위를 지원합니다. 구성 파일을 사용하면 관리자가 로그할 추적 레벨을 설정할 수 있습니다. 런타임 추적은 디폴트로 꺼져 있습니다. 디폴트 추적 파일 이름은 db2mdtrace.log입니다.

보통 추적을 실행하지 않아도 됩니다. API에 오류가 발생하여 IBM 고객만족센터에서 추적 파일을 제공하도록 요청한 경우 추적이 필요할 수도 있습니다.

다음 테이블은 다른 추적 레벨을 설명합니다.

표 60. 추적 레벨

레벨	설명	예
없음	<ul style="list-style-type: none"> 추적 꺼짐 	적용할 수 없음
높음	<ul style="list-style-type: none"> 내부 및 외부 API 입력과 종료점만 추적함 구성요소 간의 플로우를 추적함 함수 인수를 포함할 수 있음 	<ul style="list-style-type: none"> 구문 분석을 시작 및 종료함 작성, 설명, 삭제 조작 등을 시작 및 종료함
중간	<ul style="list-style-type: none"> 내부 및 외부 API의 복합 함수 사이에서 제어 플로우를 추적함 구성요소 간의 플로우를 추적함 상위 레벨 추적 포인트를 포함함 	작성 조작 시 수행된 복합 함수 호출을 표시함
낮음	<ul style="list-style-type: none"> 내부 API의 단순 함수 또는 ATOMIC 함수를 추적함 상위 레벨 및 중간 레벨 추적 포인트를 포함함 대부분의 추적 포인트에서 이 레벨을 사용함 	메타데이터 오브젝트에서 가져오기 메소드 또는 설정 메소드의 호출을 표시합니다.

없음 (None) 이외의 값으로 설정된 레벨에서 추적을 켜면 오류 로그 및 추적 로그에 설정된 레벨 및 심각도 설정에 따라 이들 로그에 API에서 발생한 오류가 기록될 수 있습니다.

DB2 Cube Views API에 대한 로그 파일

API 로그 파일은 DB2 인스턴스 레벨에서 생성됩니다. 오류 로그 파일의 이름은 db2mderror.log이고, 추적 로그 파일의 이름은 db2mdtrace.log입니다.

DB2 Cube Views API를 실행하는 지정된 DB2 인스턴스의 경우 API의 로그 파일은 DB2 진단 데이터 디렉토리(DB2DIAG라고도 함)에 생성됩니다. 이 DB2 진단 디렉토리는 일반적으로 다음 디렉토리입니다.

Windows의 경우

DB2 인스턴스 경로 디렉토리(예: c:\sqllib\

AIX의 경우

DB2 인스턴스 경로/db2dump(예: ~my inst /sqllib/db2dump)

DB2DIAG db2 dbm cfg 설정을 사용하여 디폴트 DB2DIAG 경로를 변경할 수 있습니다.

db2idrop 유틸리티는 DB2 인스턴스와 연관된 로그 파일을 정리합니다. DB2DIAG의 디폴트값을 사용하지 않으면 db2idrop 유틸리티는 DB2 Cube Views API의 로그 파일을 정리할 수 없습니다. db2idrop 유틸리티가 정리할 수 없는 로그 파일은 직접 정

리해야 합니다. DB2 Cube Views API가 구성 파일을 로드할 때 발생하는 오류는 db2mdapi.log 파일에 기록됩니다. db2mdapi.log 파일은 기타 API 로그와 마찬가지로 DB2DIAG 경로에 있습니다.

오류 로그

API는 오류 심각도를 세 가지로 구별합니다. 디폴트 심각도 설정이 중간이고 디폴트 오류 로그 파일 이름은 mdererror.log입니다. 구성 파일을 읽는 중 오류가 발생하면 이 오류는 db2mdapi.log 파일에 기록됩니다.

다음 테이블은 오류 심각도 레벨을 설명합니다.

표 61. 오류 심각도 레벨

심각도	설명	예
없음	<ul style="list-style-type: none"> 오류 및 경고 모두 무시 	적용할 수 없음
높음(대부분의 오류가 이 심각도임)	<ul style="list-style-type: none"> 중요하고 복구 불가능한 오류만 기록함 로그에 호출 스택을 덤프함 	<ul style="list-style-type: none"> 내부 코드 오류
중간	<ul style="list-style-type: none"> 사용자가 복구 가능한 오류를 기록함 높은 심각도 오류도 기록함 로그에 호출 스택을 덤프함 	<ul style="list-style-type: none"> 일반 사용자의 실수임(예: 중복 오브젝트를 작성하려고 함) 메타데이터 유효성 확인 오류 메모리 부족. 메모리를 늘리거나 사용을 줄일 수 있습니다.
낮음	<ul style="list-style-type: none"> 경고 상황을 기록함 높은 심각도 및 중간 심각도 오류도 로그함 낮음 심각도 오류는 호출 스택을 덤프하지 않음 	<ul style="list-style-type: none"> 내부 오류의 경고 정보 메시지

API가 심각도가 높은 오류 또는 중간 오류 로그로 구성된 상태에서 심각도가 높은 오류 또는 중간 오류가 발생하면 API는 오류가 발생하는 API 지점에서 시작하는 호출 스택을 생성합니다. 이 호출 스택은 중간 레벨 추적과 유사하지만 데이터는 추적 로그가 아닌 오류 로그에 전송됩니다.

로그 및 추적 시나리오

다음 로그 및 추적 시나리오는 오류를 캡처할 수 있는 방법을 설명합니다.

시나리오 1(중간 추적 레벨 및 높은 오류 심각도): 심각도가 높은 오류가 발생하면 이는 오류 및 추적 로그 모두에 나타납니다.

```
<log>
  <trace level="medium" logFile="db2mdtrace.log" bufferSize="0" />
  <error level="high" logFile="db2mderror.log" bufferSize="0" />
</log>
```

시나리오 2(중간 추적 레벨 및 낮은 오류 심각도): 심각도가 낮은 오류가 발생하면 추적 로그는 중간 또는 높은 레벨의 항목만 허용하므로 이 오류는 오류 로그에만 나타납니다.

```
<log>  
  <trace level="medium" logfile="db2mdtrace.log" bufferSize="0" />  
  <error level="low" logfile="db2mderror.log" bufferSize="0" />  
</log>
```

환경 변수가 누락되거나 로그 파일에 액세스하지 못한 경우와 관련된 오류는 스토어드 프로시저 호출의 SQLSTATE를 통해 데이터베이스 클라이언트 응용프로그램에 리턴됩니다. 구성 파일을 처리하는 중 오류가 발생하면 이 오류는 db2mdapi.log 파일에 기록됩니다. 사용자 지정 로그 파일을 여는 중 오류가 발생해도 캡처되는 오류는 없습니다.

코드 페이지 지원

DB2 Cube Views는 DB2 클라이언트 코드 페이지(응용프로그램 코드 페이지) 및 DB2 데이터베이스 코드 페이지와 같은 두 개의 코드 페이지를 사용합니다. DB2 클라이언트 코드 페이지의 판별 방법에 관한 정보는 DB2 관리 안내서: 계획의 "지원되는 지역 코드 및 코드 페이지"를 참조하십시오. DB2 Cube Views API 스토어드 프로시저는 DB2 데이터베이스 코드 페이지에서 실행됩니다. DB2 데이터베이스 코드 페이지는 데이터베이스 작성 시 설정됩니다. DB2 클라이언트 코드 페이지와 DB2 데이터베이스 코드 페이지는 서로 다를 수 있습니다. CLI는 스토어드 프로시저 문자 대형 프로젝트 (CLOB) 매개변수를 클라이언트 코드 페이지에서 스토어드 프로시저의 데이터베이스 코드 페이지로 변환합니다.

다음 그림은 CLI(call-level interface)를 통해 클라이언트가 서버와 통신하는 방법을 표시합니다. CLI는 클라이언트 코드 페이지를 데이터베이스 코드 페이지로 변환합니다.

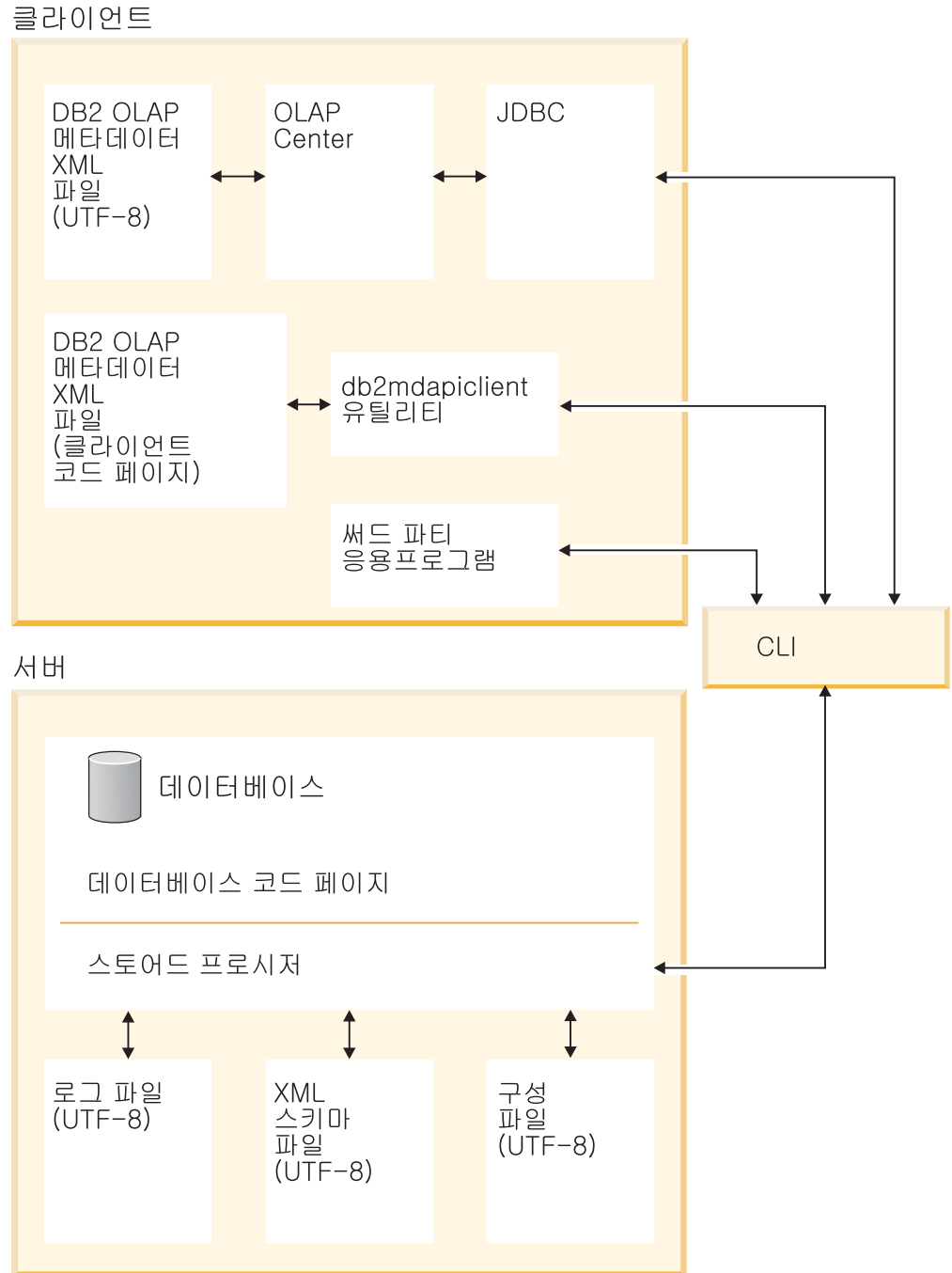


그림 41. 코드 페이지 또는 UTF-8 형식을 사용하는 다른 클라이언트에서 DB2 CLI를 통해 데이터베이스 서버로 데이터가 플로우하는 방법

CLI는 응용프로그램 코드 페이지와 데이터베이스 코드 페이지 간 변환을 관리합니다. DB2 Cube Views 클라이언트에서 API로 전송되는 데이터를 입력으로 간주합니다. API에서 DB2 Cube Views 클라이언트로 전송되는 데이터는 출력으로 간주합니다. 입력 및 출력 데이터는 DB2 클라이언트 코드 페이지에 인코드됩니다.

DB2 Cube Views의 구성요소에는 다음 코드 페이지 스펙이 있습니다. OLAP 센터:

- UTF-8로 인코딩된 DB2 Cube Views XML 파일만 승인 및 생성함
- 입력 DB2 Cube Views XML 파일이 UTF-8 이외의 인코딩을 지정할 때 오류를 리턴함
- 파일에 인코딩 스펙이 없음을 파일이 UTF-8로 인코딩된 것으로 해석함
- 익스포트 함수를 사용하여 UTF-8의 명시 인코딩 스펙이 있는 DB2 Cube Views XML 파일을 작성함

db2mdapiclient:

- -I 옵션이 지정되지 않은 경우 입력 DB2 Cube Views XML 파일을 UTF-8로 인코딩된 파일로 해석하므로 파일에 나열된 명시 인코딩 스펙을 무시합니다.
- -I 옵션이 지정되지 않은 경우 UTF-8로 인코딩된 DB2 Cube Views XML 파일을 생성하며 이러한 파일의 명시 인코딩 스펙을 포함하지 않습니다.

스토어드 프로시저 API:

- CLOB 매개변수를 DB2 클라이언트 코드 페이지에서 인코딩되는 것으로 해석함
- 입력 DB2 Cube Views XML 파일의 명시 인코딩 스펙을 무시함
- 명시 인코딩 스펙이 없는 DB2 Cube Views XML 파일을 생성함
- DB2 데이터베이스 코드 페이지를 사용하여 입력 및 출력 XML 파일을 처리함
- 임베디드(embedded) DB2 메시지가 들어 있는 UTF-8을 사용하여 인코딩된 API 로그 파일을 생성합니다. API 로그 파일은 XML이 아닙니다.
- UTF-8의 인코딩을 명시적으로 기록하는 로그 파일을 작성하지 않음
- API가 사용하는 XML 스키마 파일을 UTF-8로 인코딩함
- XML API 구성 파일을 UTF-8로 인코딩함

썬드 파티 응용프로그램의 경우 DB2 Cube Views API를 직접 호출하는 기타 응용프로그램은 DB2 클라이언트 코드 페이지에 인코딩된 XML 파일을 매개변수로 전송 및 승인해야 합니다.

DB2 Cube Views 메타데이터 테이블 및 XML 스키마 파일

버전 번호가 올바르지 않으면 API의 특정 기능이 지원되지 않습니다.

메타데이터 테이블 버전

API가 DB2 Cube Views의 현재 메타데이터 테이블 세트를 통해 DB2 데이터베이스에 연결될 때만 API가 작동합니다. DB2 Cube Views 버전 8.2의 현재 버전은 8.2.0.1.0입니다. DB2 Cube Views 버전 8.1의 버전 번호는 8.1.2.0입니다. DB2 Cube Views의 메타데이터 카탈로그 테이블 버전 번호는 SYSINFOVERSION 테이블에 저장되어 있습니다.

메타데이터 테이블 버전에 대한 자세한 정보는 183 페이지의 『db2mdapiclient 유틸리티에 대한 개요』를 참조하십시오.

DB2 Cube Views XML 스키마 파일 또는 XSD 파일이 DB2 Cube Views API에서 사용됩니다. XML 스키마 파일은 API DB2INFO.MD_MESSAGE 스토어드 프로시저에서 사용됩니다.

XML 스키마 파일 버전

메타데이터 API에서 전송하거나 이 API로 전송되는 모든 XML 문서에 버전 번호가 있어야 합니다. 이 버전 번호에 따라 스토어드 프로시저가 클라이언트가 사용하는 XML 스키마를 알 수 있습니다. XSD 스키마 파일은 특정 조작에 수용 가능한 버전 번호를 지정합니다.

DB2 Cube Views 버전 8.2의 현재 버전은 8.2.0.1.0입니다.

다음 스키마 파일에는 스토어드 프로시저 md_message()에 관한 정보가 들어 있습니다.

db2md_parameter.xsd 파일

이 파일에는 요청 및 응답 조작에 관한 정보가 들어 있습니다. 이 파일은 버전 8.1.2.1.0 및 8.2.0.1.0에서 작동합니다.

db2md_metadata.xsd 파일

이 파일은 db2md_types.xsd 파일을 포함하며 메타데이터 요소만 지정합니다. 이 파일에는 버전 정보가 없습니다.

db2md_types.xsd 파일

이 파일에는 모든 메타데이터 오브젝트에 대한 정보가 들어 있습니다. 이 파일은 버전 8.1.2.1.0 및 8.2.0.1.0에서 작동합니다.

API는 이전 버전의 스키마 파일(버전 8.1.2.1.0)을 지원할 수 있지만 설명 및 변환 조작만 지원합니다. API는 버전 8.1.2.1.0의 요청된 조작이 설명 또는 변환 조작이 아닌 경우 오류를 리턴합니다.

DB2 Cube Views 구성 파일

API는 DB2 인스턴스의 레벨에서 구성될 수 있습니다. db2md_config.xml 구성 파일의 매개변수를 변경할 수 있습니다.

DB2 Cube Views를 설치하는 모든 경우 db2_installation_path/cfg 디렉토리에 디폴트 구성 파일이 들어 있습니다. 예를 들어 Windows의 경우 디폴트 구성 파일은 c:\sqllib\cfg 디렉토리에 있을 수 있고 AIX의 경우 디폴트 구성 파일은 /usr/opt/db2_08_01/cfg 디렉토리에 있을 수 있습니다.

DB2 Cube Views를 실행하는 모든 DB2 인스턴스의 경우 the db2_instance_path 디렉토리에 db2md_config.xml 파일의 실제 사본이 들어 있습니다. 예를 들어 Windows의 경우 실제 사본은 c:\sqllib\my_inst 디렉토리에 있을 수 있고 AIX의 경우 실제 사본은 ~my_inst/sqllib 디렉토리에 있을 수 있습니다.

db2icrt 유틸리티는 디폴트 구성 파일을 db2_instance_path 디렉토리에 복사하고 새 인스턴스를 작성합니다. DB2 Cube Views를 설치하기 전에 작성된 DB2 인스턴스의 경우 설치 프로그램이 파일을 제대로 복사하지 않은 경우 사용자가 직접 구성 파일을 인스턴스 디렉토리에 복사할 수 있습니다. API가 인스턴스 디렉토리에서 구성 파일을 찾을 수 없는 경우 API는 디폴트 구성 파일을 인스턴스 디렉토리에 복사하려고 합니다.

구성 파일 db2md_config.xml은 오류 로그 및 런타임 추적을 설정하는 데 사용됩니다. 관리자가 구성 파일의 콘텐츠를 수정하여 추적 레벨, 로그할 오류의 심각도 및 로그 시 사용할 버퍼 크기(바이트 단위)를 지정할 수 있습니다.

db2md_config.xml 구성 파일의 콘텐츠 구조는 db2md_config.xsd XML 스키마 파일에서 정의됩니다. 다음 예는 구성 파일의 콘텐츠를 표시합니다.

```
<olap:config xmlns:olap="http://www.ibm.com/olap
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance
  xmlns:xsd="http://www.w3.org/2001/XMLSchema
  xsi:schemaLocation="http://www.ibm.com/olap db2md_config.xsd">
  <log>
    <trace level="none" logFile="db2mdtrace.log" bufferSize="0"/>
    <error level="medium" logFile="db2mderror.log" bufferSize="0"/>
  </log>
</olap:config>
```

메타데이터 오브젝트 형식

DB2 Cube Views XML 스키마는 DB2 Cube Views 메타데이터 오브젝트 모델의 오브젝트로 직접 맵핑되는 기본 XML 요소를 정의합니다. 이때 복합 메타데이터 구조는 이러한 기본 요소의 시퀀스로 표현됩니다. 복합 메타데이터 구조에 있는 오브젝트 간 연관관은 기본 요소 간 이름 참조를 통해 캡처됩니다.

이름 참조에 관한 예는 큐브 요소가 차원 요소에 대한 참조를 포함할 수 있는 방식입니다. 다음 예에서는 다음과 같이 XML 스키마가 정의한 대로 큐브 오브젝트에 제공되는 데이터 유형을 표시합니다. 이 예에서는 텍스트 설명만 표시되지만 응용프로그램에서는 XML 표현의 정보가 사용됩니다.

```
cube
->cube model reference
->cube dimension references
->cube facts reference
->view
```

큐브 오브젝트의 경우 다른 오브젝트 유형에 대한 모든 참조는 큐브를 표현하는 기본 요소에 포함됩니다. 반복되지 않는 검색 조작을 사용하면 큐브 오브젝트의 시퀀스 및

큐브 오브젝트만 표시됩니다. 반복 검색 조작을 사용하면 식별된 큐브에서 참조하는 다른 유형의 기타 오브젝트에 관한 정보와 함께 큐브 오브젝트에 관한 정보가 나타납니다.

오브젝트의 정렬 순서는 DB2 Cube Views XML 스키마에서 정의됩니다. 단일 조작 범위에서 동일한 오브젝트 유형(예: 큐브 오브젝트)은 함께 그룹화됩니다. 이러한 그룹에서 요소의 정렬 순서는 동일한 오브젝트 유형 간 참조에 따라 결정됩니다. 오브젝트를 참조하기 전에 먼저 참조되는 오브젝트가 표시되어야 합니다. 이러한 그룹의 정렬 순서는 다음과 같습니다.

1. attribute
2. join
3. attributeRelationship
4. level
5. cubeLevel
6. hierarchy
7. cubeHierarchy
8. dimension
9. cubeDimension
10. measure
11. facts
12. cubeFacts
13. cubeModel
14. cube

오브젝트 유형 그룹의 정렬 순서는 오브젝트 사이의 연관과 무관합니다. 속성 및 조인이 다른 오브젝트 유형과 연관될 때 다른 역할을 한다는 사실은 연관 형식에서의 정렬 순서에 영향을 주지 않습니다.

메타데이터 정보의 유형 및 정렬 순서를 표시하는 XML 파일 예는 CVSAMPLE에 포함된 XML 메타데이터 파일을 참조하십시오.

제 8 장 샘플 파일

db2mdapiclient 유틸리티에 대한 개요

db2mdapiclient 유틸리티는 DB2 Cube Views 다차원 서비스의 응용프로그램을 코드화하는 데 사용할 샘플 소스 코드로 제공됩니다. 이 유틸리티를 사용하여 메타데이터 오브젝트의 임포트, 익스포트 및 최적화를 수행할 수 있습니다.

주: DB2 Cube Views 버전 8.2에서 제공하는 유틸리티가 확장되었지만 `sqllib\samples\olap\client\db2mdapiclient.cpp` 파일의 샘플 소스 코드와 정확히 일치하지는 않습니다.

db2mdapiclient 유틸리티를 사용하여 DB2 Cube Views 스토어드 프로시저인 MD_MESSAGE()에서 지원하는 모든 조작을 수행할 수 있습니다. 이 스토어드 프로시저는 다음 테이블에서 설명됩니다.

표 62. 조작. db2mdapiclient 유틸리티가 수행할 수 있는 다차원 서비스 조작

태스크	조작
파일로 파일로 메타데이터 오브젝트 익스포트	DESCRIBE
파일에서 메타데이터 오브젝트 임포트	CREATE 또는 IMPORT
기존 메타데이터 오브젝트 변경	ALTER 또는 RENAME
기존 메타데이터 오브젝트 삭제	DROP
기존 메타데이터 오브젝트가 유효한지 확인	VALIDATE
큐브 모델 최적화	ADVISE
버전 8.1에서 버전 8.2로 XML 메타데이터 이주	TRANSLATE

db2mdapiclient 유틸리티는 파일을 사용하여 MD_MESSAGE() 스토어드 프로시저에서 전송 및 수신되는 XML을 보유합니다.

임포트의 경우 db2mdapiclient 유틸리티는 보통 DB2 Cube Views 브릿지에서 생성되거나 OLAP 센터에서 익스포트된 XML 파일을 사용합니다. 익스포트의 경우 db2mdapiclient 유틸리티는 DB2 Cube Views 브릿지 유틸리티가 데이터베이스 또는 OLAP 도구에 메타데이터를 추가하는 데 사용할 수 있는 XML 파일을 생성합니다. 이때 입력 및 출력 XML 파일에 사용되는 문자 인코딩이 중요합니다. 문자 인코딩에 대한 자세한 정보는 176 페이지의 『코드 페이지 지원』을 참조하십시오.

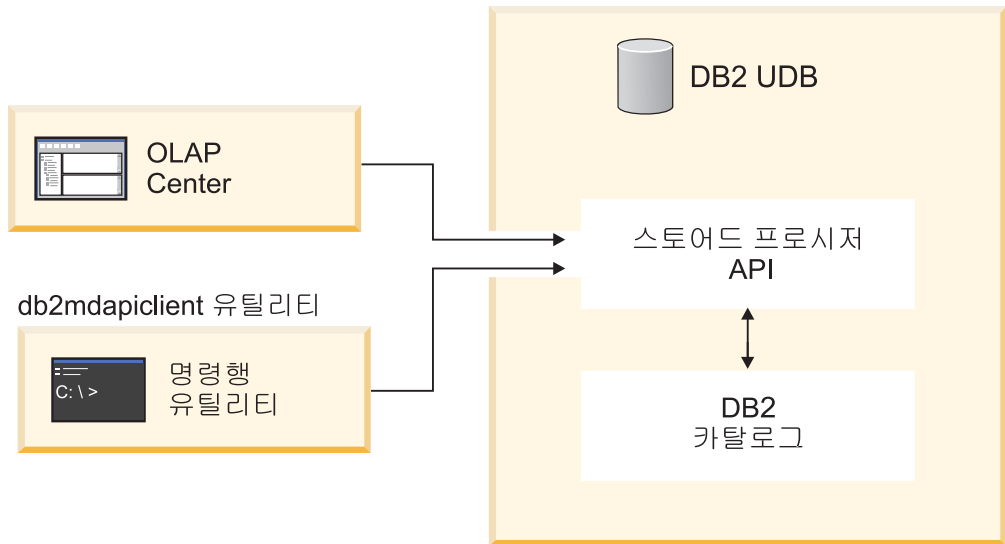


그림 42. 메타데이터 전송. db2mdapiclient 유틸리티 및 OLAP 센터가 다차원 서비스를 통해 메타데이터를 전송함

db2mdapiclient 명령: 메타데이터 오브젝트 조작

명령행에서 db2mdapiclient 유틸리티를 사용할 수 있습니다.

명령행에서 db2mdapiclient를 입력하면 db2mdapiclient 명령의 매개변수 목록을 볼 수 있습니다. **db2mdapiclient** 명령의 구문은 다음과 같습니다.

```
db2mdapiclient -d dbname [-u user] [-p password]
-i request.xml -o response.xml
[-m inputmetadata.xml] [-n outputmetadata.xml] [-a parameter_buffer_size]
[-b metadata_buffer_size] [-v] [-l] [-h]
```

-d *dbname*은 데이터베이스의 이름을 지정합니다.

-u *user*는 지정된 데이터베이스에 연결할 사용자 ID를 지정합니다.

-p *password*는 지정된 데이터베이스에 연결할 암호를 지정합니다.

-i *request.xml*은 수행할 조작이 들어 있는 필수 입력 파일을 지정합니다.

-o *response.xml*은 MD_MESSAGE() 스토어드 프로시저의 응답 XML이 들어 있는 필수 출력 파일을 지정합니다. MD_MESSAGE() 스토어드 프로시저의 세 번째 인수에서 이 응답 XML을 리턴합니다.

-m *inputmetadata.xml*은 DB2 Cube Views 메타데이터 오브젝트 XML이 들어 있는 입력 파일입니다. 이 옵션은 작성 또는 импорт 조작에 필요합니다.

-n *outputmetadata.xml*은 필요한 경우 MD_MESSAGE() 스토어드 프로시저의 두 번째 인수에서 리턴하는 응답 메타데이터 오브젝트 XML이 들어 있는 선택적 출력 파일입니다.

-a *parameter_buffer_size*는 매개변수의 버퍼 크기를 지정합니다. 디폴트값은 1048576바이트입니다.

-b *metadata_buffer_size*는 메타데이터 오브젝트 정보의 버퍼 크기를 지정합니다. 디폴트값은 1048576바이트입니다.

-v는 명령 처리 도중 추가 정보를 인쇄하도록 지정합니다.

-l는 UTF-8 지원을 사용하지 않으며 입력 및 출력 파일이 로컬 코드 페이지에 있도록 지정합니다.

-h는 사용 정보를 표시합니다.

다음 다이어그램은 MD_MESSAGE() 스토어드 프로시저가 두 개의 입력 파일 및 두 개의 출력 파일과 연관되는 방법을 표시합니다.

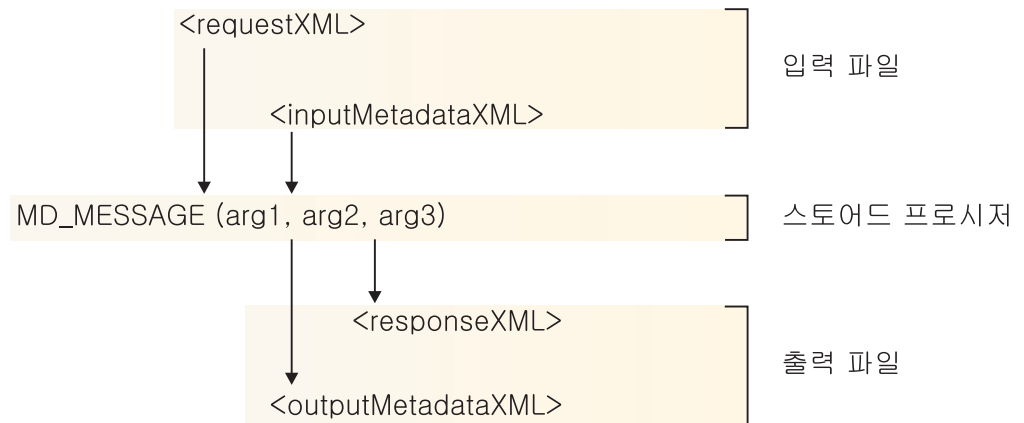


그림 43. 스토어드 프로시저가 db2mdapiclient 유틸리티에서 두 개의 입력 및 출력 파일을 처리하는 방법

예를 들어 Windows용 CVSAMPLE 데이터베이스의 DB2 Cube Views 메타데이터 오브젝트를 임포트하려면 `..\SQLLIB\samples\olap\xml\input` 디렉토리로 변경한 후 다음 명령을 입력하십시오.

```
db2mdapiclient -d CVSAMPLE -u db2admin -p mypasswd -i create.xml  
-o myresponse.xml -m ..\..\CVSAMPLE\CVSampleMetadata.xml.xml
```

제공된 샘플 파일의 설명은 186 페이지의 『API 샘플 파일』을 참조하십시오. 다차원 서비스 조작에 관한 자세한 정보는 145 페이지의 『DB2 Cube Views API 개요』를 참조하십시오.

샘플 데이터베이스 파일

CVSAMPLE 데이터베이스와 관련된 다음의 모든 파일은 `\SQLLIB\samples\olap\cvsample\` 디렉토리에 있습니다.

CVSampleMetadata.xml

CVSAMPLE 메타데이터가 들어 있는 XML 파일입니다. 이 파일을 사용하여 OLAP 센터 및 db2mdapiclient 유틸리티를 통해 CVSAMPLE 메타데이터를 임포트하십시오.

CVSampleTables.sql

CVSAMPLE 테이블을 채우는 데 사용하는 SQL 스크립트입니다.

FAMILY.txt, LINE.txt, LOCATION.txt, PRODUCT.txt, SALESFACT.txt, STORE.txt, TIME.txt

CVSAMPLE 테이블 데이터가 들어 있는 텍스트 파일 세트입니다.

CVSampleExplain.sql

DB2가 쿼리를 요약 테이블을 다시 라우팅하는지 판별하는 데 사용할 수 있는 SQL 스크립트입니다.

또한 \SQLLIB\samples\olap\xml\input 디렉토리에 CVSAMPLE 데이터베이스와 관련된 파일이 들어 있습니다.

Create.xml

작성 조작이 있는 XML 파일입니다. 이 파일을 사용하여 db2mdapiclient 유틸리티를 통해 샘플을 로드하십시오.

API 샘플 파일

CVSAMPLE 데이터베이스의 샘플 API 파일은 DB2 Cube Views에서 제공됩니다. 샘플 파일을 사용하여 db2mdapiclient 유틸리티로 샘플 시나리오를 수행할 수 있습니다. db2mdapiclient 유틸리티는 다차원 서비스 스토어드 프로시저 인터페이스의 썬 랩퍼(thin wrapper)입니다. 이 유틸리티를 샘플 소스 코드로 제공하여 API에서 응용프로그램을 코드화하는 방법을 표시합니다. 소스 코드는 \SQLLIB\samples\olap\client\db2mdapiclient.cpp에 있습니다.

각 시나리오에 나열된 샘플 파일의 콘텐츠를 MD_MESSAGE() 스토어드 프로시저에 매개변수로 전송합니다. 종종 스토어드 프로시저에 전송되는 메타데이터 매개변수는 입력 시 무시되거나 출력 시 리턴되지 않습니다. 이는 다음 시나리오에서 <empty>로 지정됩니다. 입력 메타데이터가 필요하지 않은 경우 db2mdapiclient 유틸리티 명령에서 -m 옵션을 지정하지 마십시오. 출력 메타데이터가 필요하지 않은 경우 db2mdapiclient 유틸리티 명령에서 -n 옵션을 지정하지 마십시오.

db2mdapiclient 유틸리티에서 MD_MESSAGE() 스토어드 프로시저를 사용하는 방법에 관한 자세한 정보는 183 페이지의 『db2mdapiclient 유틸리티에 대한 개요』를 참조하십시오. MD_MESSAGE() 스토어드 프로시저를 사용하는 방법에 관한 자세한 정보는 148 페이지의 『DB2 Cube Views 스토어드 프로시저』를 참조하십시오.

모든 샘플 파일은 \SQLLIB\samples\olap\xml\ 디렉토리에 있습니다. 샘플 파일을 사용하여 다음 샘플 시나리오를 수행할 수 있습니다.

DROP

이러한 파일을 사용하여 메타데이터 카탈로그의 모든 메타데이터 오브젝트를 삭제하십시오. 이 샘플에서는 메타데이터 카탈로그가 비어 있지 않은 것으로 가정합니다. 메타데이터 카탈로그가 빈 경우 조작 시 오브젝트가 없음을 알리는 경고 메시지가 수신됩니다.

입력 매개변수가 들어 있는 파일

- 요청: input\Drop.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -m 옵션을 지정하지 마십시오.

출력 매개변수가 들어 있는 파일

- 응답: output\Drop_response.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -n 옵션을 지정하지 마십시오.

CREATE

이러한 파일을 사용하여 메타데이터 카탈로그에서 메타데이터 오브젝트를 작성하십시오. 이 샘플에서는 메타데이터 카탈로그가 빈 것으로 가정합니다.

입력 매개변수가 들어 있는 파일

- 요청: input\Create.xml
- 메타데이터: input\CVSampleTestMetadata.xml

출력 매개변수가 들어 있는 파일

- 응답: output\Create_response.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -n 옵션을 지정하지 마십시오.

DESCRIBE

이러한 파일을 사용하여 메타데이터 카탈로그의 모든 메타데이터 오브젝트를 설명하십시오.

입력 매개변수가 들어 있는 파일

- 요청: input\Describe.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -m 옵션을 지정하지 마십시오.

출력 매개변수가 들어 있는 파일

- 응답: output\Describe_response.xml
- 메타데이터: <metadata XML file>

DESCRIBE(제한됨)

이러한 파일을 사용하여 CVSSample.Daily Sales 큐브를 반복적으로 설명하십시오. 이 샘플에서는 CVSSAMPLE의 메타데이터 CVSSampleMetadata.xml을 사전에 임포트한 것으로 가정합니다.

입력 매개변수가 들어 있는 파일

- 요청: input\CVSSampleDescribe_restricted.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -m 옵션을 지정하지 마십시오.

출력 매개변수가 들어 있는 파일

- 응답: output\DescribeRestricted_response.xml
- 메타데이터: <metadata XML file>

ALTER

이러한 파일을 사용하여 메타데이터 카탈로그의 메타데이터 오브젝트를 변경하십시오. 이 샘플에서는 CVSSAMPLE의 메타데이터 CVSSampleMetadata.xml을 사전에 임포트한 것으로 가정합니다.

입력 매개변수가 들어 있는 파일

- 요청: input\Alter.xml
- 메타데이터: input\CVSSampleAlter.xml

출력 매개변수가 들어 있는 파일

- 응답: output\Alter_response.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -n 옵션을 지정하지 마십시오.

RENAME

이러한 파일을 사용하여 CVSSAMPLE.Sales Model 큐브 모델 이름을 바꾸십시오. 큐브 모델 이름은 CVSSAMPLE.Sales Model(2004)로 바뀝니다. 이 샘플에서는 CVSSAMPLE의 메타데이터 CVSSampleMetadata.xml을 사전에 임포트한 것으로 가정합니다.

입력 매개변수가 들어 있는 파일

- 요청: input\CVSSampleRename.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -m 옵션을 지정하지 마십시오.

출력 매개변수가 들어 있는 파일

- 응답: output\Rename_response.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -n 옵션을 지정하지 마십시오.

VALIDATE

이러한 파일을 사용하여 최적화 유효성 확인 모드를 통해 메타데이터 카탈로그의 모든 메타데이터 오브젝트 유효성을 확인하십시오.

입력 매개변수가 들어 있는 파일

- 요청: input\Validate.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -m 옵션을 지정하지 마십시오.

출력 매개변수가 들어 있는 파일

- 응답: output\Validate_response.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -n 옵션을 지정하지 마십시오.

VALIDATE(제한됨)

이러한 파일을 사용하여 최적화 유효성 확인 모드를 통해 CVSSAMPLE.Daily Sales 큐브 유효성을 확인하십시오. 이 샘플에서는 CVSSAMPLE의 메타데이터 CVSSampleMetadata.xml을 사전에 임포트한 것으로 가정합니다.

입력 매개변수가 들어 있는 파일

- 요청: input\CVSSampleValidate_restricted.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -m 옵션을 지정하지 마십시오.

출력 매개변수가 들어 있는 파일

- 응답: output\ValidateRestricted_response.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -n 옵션을 지정하지 마십시오.

새로 작성 - 충돌 무시 모드의 IMPORT

이러한 파일을 사용하여 새로 작성 - 충돌 무시 임포트 모드를 통해 메타데이터 오브젝트를 메타데이터 카탈로그에 임포트하십시오. 이 샘플에서는 메타데이터 카탈로그가 빈 것으로 가정합니다.

입력 매개변수가 들어 있는 파일

- 요청: input\Import_model.xml
- 메타데이터: CVSampleTestMetadata.xml

출력 매개변수가 들어 있는 파일

- 응답: output\Import_model_response.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -n 옵션을 지정하지 마십시오.

새로 작성 - 충돌 대체 모드의 **IMPORT**

이러한 파일을 사용하여 새로 작성 - 충돌 대체 임포트 모드를 통해 메타데이터 오브젝트를 메타데이터 카탈로그에 임포트하십시오. 이 샘플에서는 새로 작성 - 충돌 무시 모드 시나리오에서 사전에 **IMPORT**를 완료한 것으로 가정합니다.

입력 매개변수가 들어 있는 파일

- 요청: input\Import_mode2.xml
- 메타데이터: CVSampleTestMetadata.xml

출력 매개변수가 들어 있는 파일

- 응답: output\Import_mode2_response.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -n 옵션을 지정하지 마십시오.

새로 작성 - 충돌 중단 모드의 **IMPORT**

이러한 파일을 사용하여 새로 작성 - 충돌 중단 임포트 모드를 통해 메타데이터 오브젝트를 메타데이터 카탈로그에 임포트하십시오. 이 샘플에서는 새로 작성 - 충돌 대체 모드 시나리오에서 사전에 **IMPORT**를 완료한 것으로 가정합니다.

입력 매개변수가 들어 있는 파일

- 요청: input\Import_mode3.xml
- 메타데이터: CVSampleTestMetadata.xml

출력 매개변수가 들어 있는 파일

- 응답: output\Import_mode3_response.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -n 옵션을 지정하지 마십시오.

새로 작성 - 충돌 보고 모드의 **IMPORT**

이러한 파일을 사용하여 새로 작성 - 충돌 보고 임포트 모드를 통해 메타데이터 오브젝트를 메타데이터 카탈로그에 임포트하십시오. 이 샘플에서는 새로 작성 - 충돌 중단 모드 시나리오에서 사전에 IMPORT를 완료한 것으로 가정합니다.

입력 매개변수가 들어 있는 파일

- 요청: input\Import_mode4.xml
- 메타데이터: CVSampleTestMetadata.xml

출력 매개변수가 들어 있는 파일

- 응답: output\Import_mode4_response.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -n 옵션을 지정하지 마십시오.

TRANSLATE

이러한 파일을 사용하여 Cube Views 버전 8.1 메타데이터를 Cube Views 버전 8.2 메타데이터로 변환하십시오.

입력 매개변수가 들어 있는 파일

- 요청: input\Translate.xml
- 메타데이터: CVSampleTranslateMetadata.xml

출력 매개변수가 들어 있는 파일

- 응답: output\Translate_response.xml
- 메타데이터: <metadata XML file>

ADVISE

이러한 파일을 사용하여 큐브 모델 CVSAMPLE.Sales Model에서 최적화 어드바이저를 실행하십시오. 이 샘플에서는 CVSAMPLE의 메타데이터 CVSampleMetadata.xml을 사전에 임포트하지만 이전의 이름 바꾸기 시나리오를 실행하지 않은 것으로 가정합니다.

입력 매개변수가 들어 있는 파일

- 요청: input\CVSampleAdvise.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -m 옵션을 지정하지 마십시오.

출력 매개변수가 들어 있는 파일

- 응답: output\Advise_response.xml
- 메타데이터: <empty>

db2mdapiclient 유틸리티에서 -n 옵션을 지정하지 마십시오.

부록. 메시지

다음 메시지는 DB2 Cube Views의 서버, API 및 OLAP 센터에서 출력됩니다.

소켓 오류: 데이터베이스 연결을 여러 번 열고 닫으면 소켓 오류가 발생할 수 있습니다. 드문 일이지만 Windows 2000 Advanced Server의 파티션된 환경에서 DB2 Universal Database Enterprise Server Edition, 버전 8.1.2의 DB2 Cube Views를 실행할 때 소켓 오류가 발생할 수 있습니다. 단일 Windows 세션에서 다음 단계를 한번 넘게 반복해도 오류가 발생할 수 있습니다.

1. DB2 데이터베이스에 대한 연결을 여십시오.
2. DB2 Cube Views 스토어드 프로시저를 호출하여 메타데이터 조작을 수행하십시오.
3. 데이터베이스 연결을 닫으십시오.

일시적인 해결책으로 Windows 워크스테이션을 재시작하여 소켓을 다시 활성화합니다.

SQLSTATE, API 및 기타 서버 메시지

API SQL 상태

01HQ1: 출력 XML 및 서버 로그를 참조하십시오.

설명

스토어드 프로시저에 대한 호출이 완료되었지만 요청된 메타데이터 조작 중 하나를 실행하는 중 오류가 발생했습니다.

사용자 조치

스토어드 프로시저의 출력 매개변수 컨텐츠에서 정보를 확인하십시오. 서버 로그의 항목을 점검하여 추가 정보를 얻을 수도 있습니다.

38Q00: 자세한 정보는 서버 로그를 참조하십시오.

설명

스토어드 프로시저에 대한 호출에 실패했습니다. 요청된 메타데이터 작업이 실행되지 않았습니다. 출력 매개변수를 통해 스토어드 프로시저에서 정보가 리턴되지 않았습니다.

사용자 조치

자세한 정보는 서버 로그의 항목을 점검하십시오.

38Q01: 설치 경로를 알 수 없습니다.

설명

DB2 설치 디렉토리를 스토어드 프로시저 프로세스에서 판별할 수 없으므로 스토어드 프로시저에 대한 호출에 실패했습니다. 요청된 메타데이터 조작이 실행되지 않았습니다. 출력 매개변수를 통해 스토어드 프로시저에서 정보가 리턴되지 않았습니다.

사용자 조치

Windows 운영 체제를 사용하는 경우 DB2PATH 환경 변수가 디폴트 값에 의해 또는 사용자 조치에 의해 올바른 값으로 설정되었는지 확인하십시오. 데이터베이스 관리 프로그램을 재시작한 다음 스토어드 프로시저에 대한 호출을 다시 발행하십시오. 추가 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

38Q02: 서버 로그 파일을 열 수 없습니다.

설명

스토어드 프로시저 프로세스에 의한 작성을 위해 스토어드 프로시저에서 사용한 하나 이상의 로그 파일을 열 수 없으므로 스토어드 프로시저에 대한 호출에 실패했습니다. 요청된 메타데이터 조작이 실행되지 않았습니다. 출력 매개변수를 통해 스토어드 프로시저에서 정보가 리턴되지 않았습니다.

사용자 조치

해당 파일 시스템에서 읽고 쓰기 위해 스토어드 프로시저 구성 파일(예: olap_config.xml)에 지정된 로그 파일을 작성하거나 열 수 있는지 확인하십시오. 로그 파일이 없으면 스토어드 프로시저가 이러한 파일을 작성하려고 시도합니다. AIX에서 로그 파일이 분리된(fenced) 데이터베이스 사용자 ID로 읽고 쓸 수 있는지 확인하십시오.

38Q03: 메타데이터 입력 매개변수가 누락되었습니다.

설명

요청된 메타데이터 조작에서 메타데이터가 스토어드 프로시저에 대한 입력으로 전송되어야 하지만 입력 메타데이터 매개변수를 통해 제공된 메타데이터가 없기 때문에 스토어드 프로시저에 대한 호출에 실패했습니다. 출력 매개변수를 통해 스토어드 프로시저에서 정보가 리턴되지 않았습니다.

사용자 조치

다음에 스토어드 프로시저에 대한 호출을 작성할 때 요청된 메타데이터 조작에 대해 입력 메타데이터 스토어드 프로시저 매개변수를 사용하여 필수 메타데이터를 제공하십시오.

38Q04: [error_type] 오류: 응답 출력 버퍼가 너무 작습니다.

설명

조작 응답의 출력 매개변수 버퍼가 너무 작아 리턴되는 CLOB 구조를 수용할 수 없으므로 스토어드 프로시저에 대한 호출에 실패했습니다. 출력 매개변수를 통해 스토어드 프로시저에서 정보가 리턴되지 않았습니다.

사용자 조치

출력 응답 매개변수에 큰 크기를 사용하여 스토어드 프로시저를 다시 카탈로그화하십시오.

일반

성공 코드

0: 조작이 완료되었습니다. 오류가 발생하지 않았습니다.:

설명

요청된 메타데이터 조작이 완료되었습니다. 조작 실행 중 오류가 발생하지 않았습니다.

사용자 조치

이 정보는 사용자를 위한 정보용입니다. 조치가 필요하지 않습니다.

1: 조작이 완료되었습니다. 추가 정보가 리턴되었습니다.:

설명

요청된 메타데이터 조작이 완료되었습니다. 조작에서 경고 또는 오류 상태를 설명할 수 있는 추가 정보를 리턴했습니다.

사용자 조치

리턴된 추가 정보에서 INFO 요소를 점검하십시오.

2: 조작이 완료되었습니다. 메타데이터에 대한 변경이 없습니다.:

설명

요청된 메타데이터 조작이 완료되었습니다. 조작으로 데이터베이스 카탈로그의 메타데이터에 대한 변경사항이 작성되지 않았습니다.

사용자 조치

메타데이터 카탈로그에서 메타데이터를 변경하려는 경우 다른 모드를 사용하여 메타데이터 조작 요청을 다시 발행하십시오.

일반 외부 오류/경고 코드

100: 조작에 메모리를 할당할 수 없습니다. 메모리가 사용 가능한지 확인하십시오.:

설명

요청된 메타데이터 조작 실행 중 스토어드 프로시저가 필수 메모리 세그먼트를 할당할 수 없습니다.

사용자 조치

분리(fenced) 스토어드 프로시저 프로세스에 사용 가능한 메모리를 늘리십시오.

101: object name 오브젝트 처리 중에 내부 오류가 발생했습니다.:

설명

요청된 메타데이터 조작 실행 중에 내부 오류가 발생했습니다.

사용자 조치

추가 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

102: 출력 버퍼 크기 buffer_size가 너무 작습니다. 최소 size로 버퍼 크기를 변경하십시오.:

설명

스토어드 프로시저에 사용 가능한 출력 매개변수 버퍼가 너무 작아 스토어드 프로시저에서 생성한 CLOB 구조를 수용할 수 없습니다.

사용자 조치

가능하면 OUT 및 INOUT 매개변수에 큰 크기를 사용하여 스토어드 프로시저를 다시 카탈로그화하십시오. 그렇지 않으면 스토어드 프로시저에서 작은 정보를 리턴하도록 쿼리를 제한해야 합니다.

103: 이 제품에 유효한 사용권이 없습니다.:

설명

이 제품 설치에 대해 유효한 제품 사용권이 없으므로 메타데이터 조작을 수행할 수 없습니다.

사용자 조치

시스템에 유효한 제품 사용권을 설치하거나 IBM 고객만족센터 또는 IBM 영업센터에 문의하여 새 제품 사용권을 구매하십시오.

104: 내부 오류가 발생했습니다. 다음 토큰이 리턴되었습니다. token0, token1, token2, token3.:

설명

요청된 메타데이터 조작 실행 중에 내부 오류가 발생했습니다.

사용자 조치

추가 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

599: 작업이 실행되지 않았습니다.:

설명

이 조작을 실행하기 전에 오류가 발생했습니다. 따라서 조작이 실행되지 않았습니다.

사용자 조치

동일한 스토어드 프로시저 호출 시 실행된 이전 메타데이터 조작의 결과를 점검하십시오. 서버 로그의 항목을 점검하여 추가 정보를 얻을 수도 있습니다. 이전 조작의 실패를 가져온 문제점을 수정한 후 스토어드 프로시저를 다시 호출하고 동일한 메타데이터 조작을 요청하십시오.

일반 내부 오류/경고 코드

600: 입력 parameter_name 매개변수가 올바르지 않습니다. 메시지는 message입니다. 매개변수를 점검하고 다시 시도하십시오.:

설명

스토어드 프로시저 내부의 메소드에 입력으로 전송된 매개변수 중 하나가 유효하지 않습니다.

사용자 조치

추가 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

601: 입력 parameter_name 매개변수가 널(NULL)입니다.:

설명

스토어드 프로시저 내부의 메소드에 입력으로 전송된 매개변수 중 하나가 유효하지 않은 값 널(NULL)입니다.

사용자 조치

추가 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

602: 값의 value parameter_name 매개변수가 유효한 범위 range_value1 및 range_value2 안에 있지 않습니다.:

설명

스토어드 프로시저 내부의 메소드에 입력으로 전송된 매개변수 중 하나가 유효한 범위 밖의 값을 갖습니다.

사용자 조치

추가 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

603: 유니코드 문자열 string이 잘못되었거나 유효하지 않습니다. 메모리 문제일 수 있습니다.:

설명

스토어드 프로시저의 유니코드 문자열이 올바르지 않거나 유효하지 않습니다. 이것은 시스템의 메모리 문제나 스토어드 프로시저의 문제일 수 있습니다. 또한 스토어드 프로시저가 로드하는 ICU 라이브러리의 버전이 올바르지 않기 때문일 수도 있습니다.

사용자 조치

스토어드 프로시저에서 처리 중인 데이터 볼륨을 수용할 수 있는 메모리가 있는지 확인하십시오. 스토어드 프로시저의 현재 버전과 함께 사용하기 위한 ICU 라이브러리의 버전이 로드 중인지 확인하십시오. 설정이 올바른지 알아보기 위해 환경에 설정된 런타임 라이브러리 검색 경로를 점검해야 할 수도 있습니다.

추가 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

604: string 문자열의 내용을 유니코드에서 code_page의 사용 중인 코드 페이지로 변환하는 데 실패했습니다.:

설명

유니코드 문자열 오브젝트를 다른 인코딩을 사용하는 문자열로 변환하는 데 실패했습니다. 이것은 시스템의 메모리 문제나 스토어드 프로시저의 문제일 수 있습니다. 또한 시스템의 코드 페이지 변환 문제이거나 스토어드 프로시저에서 잘못된 ICU 라이브러리 버전을 로드했을 수도 있습니다.

사용자 조치

필수적인 ICU 코드 페이지 변환 파일이 데이터베이스 서버 시스템에 설치되어 있는지 확인하십시오. 스토어드 프로시저에서 처리 중인 데이터 볼륨을 수용할 수 있는 적절한 메모리가 있는지 확인하십시오. 현재 스토어드 프로시저 버전과 함께 사용할 ICU 라이브러리가 로드되어 있는지 확인하십시오. 설정이 올바른지 알아보기 위해 환경에 설정된 런타임 라이브러리 검색 경로를 점검해야 할 수도 있습니다.

추가 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

605: memory_buffer의 할당된 메모리를 사용 가능하게 해야 합니다.:

설명

스토어드 프로시저 내부의 메소드가 다른 내부 메소드에서 비워야 하는 메모리 버퍼를 리턴했습니다.

사용자 조치

스토어드 프로시저의 내부에 있는 메소드가 리턴된 메모리 버퍼를 비워야 합니다. 추가 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

606: UChar의 XMLCh에서 UChar로의 변환에 실패했습니다.:

설명

XMLCh 문자와 UChar 문자 간 변환에 실패했습니다.

사용자 조치

추가 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

607: size 크기의 입력 버퍼는 너무 작습니다. 버퍼 크기를 최소 new_size(으)로 변경하십시오.:

설명

스토어드 프로시저 내부의 메모리 버퍼가 너무 작아서 필수 메시지의 텍스트를 모두 수용할 수 없습니다.

사용자 조치

추가 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

608: stored_procedure_name 유형은 현재 컨텍스트에서 올바르지 않습니다.:

설명

스토어드 프로시저 처리 중에 예기치 않은 유형이 발생했습니다.

사용자 조치

시도한 메타데이터 조작의 상태 ID 및 텍스트에 관한 도움을 받으려면 IBM 고객만족센터에 문의하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일을 제공하십시오.

609: data_type 데이터 유형은 현재 컨텍스트에서 올바르지 않습니다.:

설명

스토어드 프로시저 처리 중에 예기치 않은 데이터 유형이 발견되었습니다.

사용자 조치

시도한 메타데이터 조작의 상태 ID 및 텍스트에 관한 추가적인 도움을 받으려면 IBM 고객만족센터에 문의하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일을 제공하십시오.

MDOBJECT.LIB 오류

1000: object_name 오브젝트를 복제하는 데 실패했습니다.:

설명

스토어드 프로시저 내부의 클래스 오브젝트를 클론하는 도중 오류가 발생했습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

1001: 계층 구조에 셋 이상의 레벨이 있으므로 hierarchy_name 계층 구조에 대한 전개를 반복으로 설정할 수 없습니다.:

설명

계층 구조에 레벨이 너무 많기 때문에 반복 전개될 수 없습니다.

사용자 조치

반복 전개로 변경하기 전에 계층 구조를 두 개 이하의 레벨로 수정하십시오.

1002: 호출된 function_name 함수는 지원되지 않습니다.:

설명

스토어드 프로시저 클래스 중 하나에서 스토어드 프로시저 내부의 가상 메소드가 구현되지 않았습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

1003: 컨테이너가 예기치 않게 비어 있습니다.:

설명

스토어드 프로시저 내부의 컨테이너 구조가 예기치 않게 비어 있습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

1004: object_name 오브젝트를 컨테이너에서 찾을 수 없습니다.:

설명

스토어드 프로시저의 내부 컨테이너 구조 중 하나에서 검색된 오브젝트가 예기치 않게 누락되었습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

1005: element_name 컨테이너에 중복 요소가 이미 있습니다.:

설명

스토어드 프로시저의 내부 컨테이너 구조 중 하나에 사전에 있어서는 안되는 오브젝트가 존재합니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

1006: 목록 작업 중 예외가 발생했습니다.:

설명

스토어드 프로시저의 목록 구조 중 하나에서 조작을 수행하는 동안 예기치 않은 예외가 발생했습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

1007: 컨테이너에서 내부 오류가 발생했습니다. 오류 코드는 error and number 이고 메시지는 message입니다.:

설명

스토어드 프로시저의 내부 컨테이너 구조 중 하나에서 조작을 실행하는 도중 오류가 발생했습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

1008: 복사 작업이 모든 등록 정보를 완전히 복사하지 않았습니다. value 값을 가진 등록 정보 property_name의 복사 조작에 실패했습니다.:

설명

스토어드 프로시저의 내부 오브젝트 중 하나에서 복사 조작을 실행하는 도중 오류가 발생했습니다. 내부 오브젝트 등록 정보 중 하나를 복사하는 데 실패했습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

1009: type1의 오브젝트 유형이 유효하지 않습니다. type2이(가) 예상되었습니다.:

설명

스토어드 프로시저 처리 도중 예기치 않은 오브젝트 유형을 발견했습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

1010: parameter_name 매개변수에 전체 ID가 없습니다.:

설명

스토어드 프로시저 내부의 메소드에 입력으로 전송되는 매개변수 중 하나가 불완전한 메타데이터 오브젝트 ID입니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

1011: object_name 오브젝트에 전체 ID가 없습니다.:

설명

스토어드 프로시저의 메타데이터 오브젝트 ID가 예기치 않게 불완전합니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

1012: parameter_name 매개변수가 오브젝트와 동일합니다.:

설명

스토어드 프로시저 내부의 오브젝트 메소드에 입력으로 전송되는 매개변수 중 하나가 예기치 않게 메소드를 소유한 오브젝트와 동일한 오브젝트입니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

1013: 예기치 않은 널(NULL) 포인터가 발견되었습니다.:

설명

스토어드 프로시저 처리 도중 예기치 않은 널(NULL) 포인터를 발견했습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

1014: 컨테이너 커서가 컨테이너의 끝에 도달했습니다.:

설명

스토어드 프로시저 내부의 컨테이너 구조 중 하나에서 커서가 예기치 않게 컨테이너 끝에 도달했습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

1015: object_name 오브젝트가 유효하지 않습니다. 이유: ID=ID, Message=message.:

설명

스토어드 프로시저 내부의 메타데이터 오브젝트가 유효하지 않습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

1016: object_name 오브젝트에 유형이 list_type이고 예기치 않게 비어 있는 목록이 있습니다.:

설명

스토어드 프로시저 내부의 컨테이너 구조가 비어 있습니다.

사용자 조치

추가 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

MDOPERATION.LIB 오류

2001: 생성된 쿼리 query에 필수 컬럼 objectType이 없습니다.:

설명

스토어드 프로시저에서 생성한 SQL 쿼리가 필요한 컬럼에서 누락되었습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

MDPARSER.LIB 오류

3001: operation 중 구문 분석기에 의해 XML 예외가 발견되었습니다. 메시지는 message입니다.:

설명

스토어드 프로시저에 전송된 XML을 구문 분석하는 동안 스토어드 프로시저에서 예기치 않은 예외가 발생했습니다.

사용자 조치

스토어드 프로시저에 전송된 XML이 제대로 형성되어 이 제품에서 발행된 XML 스키마에 유효한지 확인하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

3002: operation에서 예기치 않은 구문 분석기 예외가 발견되었습니다.:

설명

스토어드 프로시저에 전송된 XML을 구문 분석하는 동안 스토어드 프로시저에서 예기치 않은 예외가 발생했습니다.

사용자 조치

스토어드 프로시저에 전송된 XML이 제대로 형성되어 이 제품에서 발행된 XML 스키마에 유효한지 확인하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한

메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

3003: operation 중 구문 분석기에 의해 SAXParseException이 발견되었습니다. 메시지는 message입니다.:

설명

스토어드 프로시저에 전송된 XML을 구문 분석하는 동안 스토어드 프로시저에서 예기치 않은 예외가 발생했습니다.

사용자 조치

스토어드 프로시저에 전송된 XML이 제대로 형성되어 이 제품에서 발행된 XML 스키마에 유효한지 확인하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

3004: 시스템이 operation 구문 분석기 오류 메시지를 가져오는 데 실패했습니다.:

설명

스토어드 프로시저에 전송된 XML을 구문 분석하는 동안 스토어드 프로시저에서 예기치 않은 오류가 발생했습니다. XML 구문 분석기에서 오류 메시지를 검색할 수 없습니다.

사용자 조치

스토어드 프로시저에 전송된 XML이 제대로 형성되어 이 제품에서 발행된 XML 스키마에 유효한지 확인하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

3100: 시스템이 매개변수 유형 (행: line, 문자: character, 메시지: message)의 XML을 구문 분석하는 데 실패했습니다.:

설명

스토어드 프로시저는 입력 XML을 구문 분석할 수 없습니다. 입력 XML이 잘못 형성되거나 이 제품에서 발행되는 XML 스키마에서 유효하지 않을 수 있습니다.

사용자 조치

스토어드 프로시저에 전송된 XML이 제대로 형성되어 이 제품에서 발행된 XML 스키마에 유효한지 확인하십시오.

3101: 알 수 없는 메타데이터 오브젝트를 발견했습니다. parser_message.:

설명

메타데이터 오브젝트의 알 수 없는 유형이 스토어드 프로시저에 전송된 XML에 있습니다. 입력 XML은 스토어드 프로시저에서 처리될 수 없습니다.

사용자 조치

스토어드 프로시저에 전송된 XML이 제대로 형성되어 이 제품에서 발행된 XML 스키마에 유효한지 확인하십시오.

3102: 알 수 없는 XML 속성이 발견되었습니다(attribute_name, attribute_value):

설명

XML 속성의 알 수 없는 유형이 스토어드 프로시저에 전송된 XML에 있습니다. 입력 XML은 스토어드 프로시저에서 처리될 수 없습니다.

사용자 조치

스토어드 프로시저에 전송된 XML이 제대로 형성되어 이 제품에서 발행된 XML 스키마에 유효한지 확인하십시오.

3103: 이름이 name이고 value인 속성에 대해 핸들러가 유효하지 않은 열거 값을 발견했습니다.:

설명

유효하지 않은 열거 값이 스토어드 프로시저에 전송된 XML에 있습니다. 입력 XML은 스토어드 프로시저에서 처리될 수 없습니다.

사용자 조치

스토어드 프로시저에 전송된 XML이 제대로 형성되어 이 제품에서 발행된 XML 스키마에 유효한지 확인하십시오.

3500: 이름이 name인 속성이나 요소에는 데이터가 필요합니다.:

설명

스토어드 프로시저는 스토어드 프로시저에서 리턴해야 하는 XML에 표시된 XML 속성 또는 요소에 값을 설정하는 데 실패했습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

3501: 속성 attribute_name1 및 attribute_name2에는 데이터가 필요합니다.:

설명

스토어드 프로시저는 스토어드 프로시저에서 리턴해야 하는 XML에 표시된 XML 속성 또는 요소에 값을 설정하는 데 실패했습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

3502: 이름이 name이고 값이 value인 속성에 대해 포맷터가 유효하지 않은 열거 값을 발견했습니다.:

설명

스토어드 프로시저에서 리턴해야 하는 XML을 포맷팅하는 동안 스토어드 프로시저에서 유효하지 않은 열거 값을 발견했습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

MDDATABASE.LIB 오류

4000: 데이터베이스 연결에 실패했습니다. 데이터베이스 이름은 database_name이고 사용자 이름은 user_name입니다.:

설명

스토어드 프로시저가 데이터베이스를 자신과 연결하는 데 실패했습니다.

사용자 조치

스토어드 프로시저가 사용하는 사용자 ID에 데이터베이스와 연결하는 데 적절한 특권이 있는지 확인하십시오.

4001: 연결이 이미 존재하기 때문에 데이터베이스 연결이 실행되지 않았습니다.:

설명

스토어드 프로시저가 중복된 데이터베이스와의 내부 연결을 발견했습니다. 요청된 메타데이터 조작은 수행될 수 없습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4002: 데이터베이스 작업에 실패했습니다.:

설명

스토어드 프로시저가 데이터베이스에 발행한 SQL문을 실행하는 동안 오류가 발생했습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4003: CLI 호출 call_name의 실행에 실패했습니다.:

설명

표시된 CLI 호출을 실행하는 도중 오류가 발생했습니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4004: 리턴된 데이터가 잘렸습니다.:

설명

실패한 데이터베이스 조작 중에 리턴된 진단 정보가 절단되었습니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4005: 데이터베이스로부터 경고를 받았습니다. SQLSTATE=code, Message=message.:

설명

스토어드 프로시저가 발행한 CLI 호출로 경고 정보가 리턴됩니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오.

4008: 알 수 없는 DB2 데이터 유형이 발견되었습니다.:

설명

데이터베이스 요청을 실행하는 도중 스토어드 프로시저에서 알 수 없는 데이터 유형을 발견했습니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4009: 유효한 세이브포인트 이름이 생성되지 않았습니다.:

설명

스토어드 프로시저에서 유효한 데이터베이스 트랜잭션 세이브포인트 이름을 생성할 수 없습니다. 스토어드 프로시저가 해당 데이터베이스 응용프로그램 ID를 사용하여 세이브포인트 이름을 형성합니다.

사용자 조치

스토어드 프로시저에 호출을 다시 발행하십시오. 호출을 다시 발행하면 스토어드 프로시저의 새 데이터베이스 응용프로그램 ID가 생성되어 유효한 세이브포인트 이름을 생성할 수 있습니다. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4010: DB2 세이브포인트를 설정하려는 데 실패했습니다.:

설명

스토어드 프로시저에서 데이터베이스 트랜잭션 세이브포인트를 설정할 수 없습니다. 스토어드 프로시저의 이 인스턴스에서 사용한 이름과 동일한 이름의 세이브포인트가 현재 트랜잭션에 이미 존재할 수도 있습니다.

사용자 조치

가능하면 스토어드 프로시저에 호출을 다시 발행하기 전에 현재 트랜잭션의 세이브포인트를 릴리스하십시오. 새 트랜잭션을 사용하여 스토어드 프로시저에 호출을 다시 발행할 수도 있습니다.

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4011: 이 실행 지점 이전에 세이브포인트가 설정되지 않았습니다.:

설명

스토어드 프로시저의 한 지점에서 트랜잭션 세이브포인트가 예기치 않게 누락되었습니다. 누락된 세이브포인트가 스토어드 프로시저에서 설정될 수 없거나 스토어드 프로시저 외부에서 수행된 데이터베이스 조치를 통해 세이브포인트가 릴리스되었을 수 있습니다.

사용자 조치

스토어드 프로시저에 호출을 다시 발행하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4012: 유효하지 않은 세이브포인트 문자열 스토리지가 있습니다.:

설명

스토어드 프로시저 프로세스에 사용 가능한 메모리가 부족하므로 스토어드 프로시저 내부의 데이터 구조에 데이터베이스 트랜잭션 세이브포인트 이름이 제대로 저장되지 않았습니다.

사용자 조치

스토어드 프로시저에 호출을 다시 발행하십시오. 문제가 지속되면 분리(fenced) 스토어드 프로시저 프로세스에 사용 가능한 메모리를 늘리십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4013: 세이브포인트를 지우는 데 실패했습니다.:

설명

스토어드 프로시저에서 데이터베이스 트랜잭션 세이브포인트를 지울 수 없습니다. 스토어드 프로시저에서 누락된 세이브포인트를 설정할 수 없거나 스토어드 프로시저 외부에서 수행된 데이터베이스 조치를 통해 세이브포인트가 릴리스되었을 수 있습니다.

사용자 조치

스토어드 프로시저에 호출을 다시 발행하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4014: DB2 AUTOCOMMIT 설정을 판별하려는 데 실패했습니다.:

설명

스토어드 프로시저가 DB2 AUTOCOMMIT 기능이 사용 가능한지를 판별하려는 데 실패했습니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4015: DB2 AUTOCOMMIT OFF를 설정하려는 데 실패했습니다.:

설명

스토어드 프로시저가 DB2 AUTOCOMMIT 기능을 사용하지 않으려는 데 실패했습니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4016: CLI 호출 SQLFetch()에서 데이터가 리턴되지 않았습니다.:

설명

CLI 함수 SQLFetch()에서 스토어드 프로시저로 리턴된 데이터가 없습니다. 이는 허용 가능하지만 스토어드 프로시저는 이 오류가 변경되지 않은 스토어드 프로시저를 통해 전파되도록 허용해서는 안 됩니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4017: Object_name 오브젝트의 구문이 올바르지 않았습니다.:

설명

스토어드 프로시저 내부의 데이터베이스 오브젝트가 제대로 초기화되지 않았습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4018: 데이터베이스 연결 끊기에 실패했습니다.:

설명

스토어드 프로시저가 데이터베이스의 연결을 끊는 데 실패했습니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4019: DB2 SQL 오류 - SQLCODE sqlcode, SQLSTATE sqlstate, SQLMESG sqlmesg:

설명

스토어드 프로시저가 데이터베이스에 발행한 SQL문을 실행하는 도중 오류가 발생했습니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4020: DB2 SQL 오류 - 세부사항이 제공되지 않습니다.:

설명

스토어드 프로시저가 데이터베이스에 발행한 SQL문을 실행하는 도중 발생한 오류의 진단 정보를 사용할 수 없습니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4021: DB2 SQL 오류 - 세부사항이 제공되지 않습니다.:

설명

스토어드 프로시저가 데이터베이스에 발행한 SQL문을 실행하는 도중 발생한 다른 오류의 진단 정보를 수집하려는 도중 오류가 발생했습니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4022: DB2 handle_name 핸들의 할당에 실패했습니다.:

설명

스토어드 프로시저에서 DB2 핸들을 할당하려는 도중 오류가 발생했습니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4023: DB2 handle_name 핸들을 사용 가능하게 하는 데 실패했습니다.:

설명

스토어드 프로시저에서 DB2 핸들의 여유 공간을 확보하려는 도중 오류가 발생했습니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4028: 트랜잭션이 중지되지 않았습니다.:

설명

스토어드 프로시저의 트랜잭션을 종료하려는 도중 오류가 발생했습니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4029: 기본 오브젝트 테이블에서 동일한 이름과 스키마를 공유하는 중복 행이 발견되었습니다.:

설명

메타데이터 카탈로그 테이블 중 하나에서 동일한 이름 및 스키마를 공유하는 중복 행이 예기치 않게 발견되었습니다. 이러한 공유는 스토어드 프로시저의 내부 오류를 표시합니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4030: DBINFO 구조가 초기화되지 않았습니다. DBINFO 옵션을 사용하여 데이터베이스에서 스토어드 프로시저가 작성되었는지 확인하십시오.:

설명

스토어드 프로시저가 데이터베이스 클라이언트에서 DBINFO 구조를 수신하지 않았습니다.

사용자 조치

DBINFO 옵션을 사용하여 스토어드 프로시저를 적절한 데이터베이스에 카탈로그화했는지 확인하십시오.

4031: 스키마를 DB2INFO로 설정하는 데 실패했습니다.:

설명

스토어드 프로시저가 현재 스키마로 DB2INFO를 설정하는 데 실패했습니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 스토어드 프로시저에 호출을 다시 발행하십시오.

4032: 기본 오브젝트 테이블에 있는 오브젝트와 삽입되는 오브젝트 간 충돌로 인해 작업에 실패했습니다.:

설명

메타데이터 카탈로그 테이블 중 하나에 중복 메타데이터 오브젝트 항목이 있기 때문에 스토어드 프로시저의 SQL INSERT문에 실패했습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4033: 컬럼 함수의 피연산자에 컬럼 함수가 들어 있습니다.:

설명

다른 컬럼 함수에 중첩된 컬럼 함수가 스토어드 프로시저가 발행한 SQL문 중 하나에서 발견되었습니다. SQL문에서 컬럼 함수는 중첩될 수 없습니다.

사용자 조치

스토어드 프로시저에서 생성한 SQL문에 중첩된 컬럼 함수가 더 이상 존재하지 않도록 입력 속성 또는 치수 오브젝트의 SQL 표현식 템플릿을 수정하십시오.

4034: DB2 ISOLATION LEVEL 설정이 판별되지 않았습니다.:

설명

스토어드 프로시저가 데이터베이스 트랜잭션 분리 레벨을 판별하려는 시도에 실패했습니다. 분리 레벨을 판별할 수 없습니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4035: DB2 SOLATION LEVEL을 READ STABILITY으로 설정하는 데 실패했습니다.:

설명

스토어드 프로시저가 데이터베이스 트랜잭션 분리 레벨을 설정하려는 시도에 실패했습니다. 스토어드 프로시저에 읽기 안정성의 분리 레벨이 필요합니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4036: 현재 설치된 DB2 Universal Database의 버전을 판별할 수 없습니다.:

설명

스토어드 프로시저가 데이터베이스 관리 프로그램의 버전 레벨을 판별하려는 시도에 실패했습니다.

사용자 조치

클라이언트 및 서버의 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4037: DB2 Universal Database, 버전 version_number, FixPak number0(가) 현재 설치되어있지만 이 버전의 DB2 Cube Views와 호환되지 않습니다. DB2 Universal Database 버전을 업그레이드하십시오.:

설명

현재 설치된 버전 DB2 Universal Database가 DB2 Cube Views의 이 버전과 호환되지 않습니다. DB2 Cube Views 버전 레벨 이상이 되도록 DB2 Universal Database 버전을 업그레이드하십시오.

사용자 조치

DB2 Universal Database 및 DB2 Cube Views의 호환 가능한 버전을 동일한 서버에 설치했는지 확인하십시오. 자세한 정보는 설치 및 설정 문서를 참조하십시오.

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4038: SQL문이 너무 길거나 복잡하여 처리할 수 없습니다.:

설명

길이 또는 복잡도의 시스템 한계를 초과하거나 관련된 제한조건 또는 트리거가 너무 많으므로 스토어드 프로시저가 발행한 명령문을 처리할 수 없습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

4039: 필수 SYSINFOVERSION 메타데이터 테이블이 DB2 카탈로그 테이블에 존재하지 않습니다. 메타데이터를 이주하십시오.:

설명

필수 SYSINFOVERSION 메타데이터 테이블이 DB2 카탈로그 테이블에 없습니다. 메타데이터를 이주하십시오.

사용자 조치

메타데이터를 이주시켜 DB2 카탈로그 테이블을 갱신하십시오. 이주에 관한 정보는 *DB2 Cube Views 안내 및 참조서* 책을 참조하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트에 관한 추가적인 도움을 받으려면 IBM 고객만족센터에 문의하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일을 제공하십시오.

4040: 메타데이터 테이블 버전이 현재 DB2 Cube Views API 버전과 일치하지 않습니다. 메타데이터 테이블 버전은 version_number1이며 현재 API 버전은 version_number2입니다. 메타데이터 테이블을 현재 API 버전으로 이주해야 합니다.:

설명

SYSINFOVERSION 테이블의 VERSION 값이 조작을 수행하려고 시도 중인 DB2 Cube Views API의 버전과 일치하지 않습니다. 메타데이터 테이블을 이주시키고 올바른 VERSION 값이 SYSINFOVERSION 테이블에 있는지 확인해야 합니다.

사용자 조치

DB2 카탈로그의 메타데이터 테이블을 이주시키십시오. 이주에 관한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트에 관한 추가적인 도움을 받으려면 IBM 고객만족센터에 문의하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일을 제공하십시오.

4041: 메타데이터 테이블 버전을 지정하지 않았습니다. 메타데이터 테이블을 현재 API 버전으로 이주해야 합니다.:

설명

SYSINFOVERSION 테이블의 VERSION 값이 존재하지 않습니다. db2mdmigrate.sql 스크립트를 실행하여 DB2 카탈로그의 메타데이터 테이블을 이주시켜야 합니다.

사용자 조치

기존 메타데이터 테이블을 이주시키십시오. 이주에 관한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트에 관한 추가적인 도움을 받으려면 IBM 고객만족센터에 문의하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일을 제공하십시오.

4042: 다중 또는 중복 메타데이터 테이블 버전을 지정했습니다. 하나의 올바른 버전만이 지정될 수 있도록 메타데이터를 이주하십시오.:

설명

메타데이터 테이블의 버전이 둘 이상이거나 중복 버전이 SYSINFOVERSION 카탈로그 테이블에 지정되었습니다. SYSINFOVERSION 테이블에 올바른 버전이 하나만 지정되어 있어야 합니다.

사용자 조치

추가 지원은 IBM 고객만족센터에 문의하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일을 제공하십시오.

MDUTILITY.LIB 오류

5000: 유틸리티가 string 문자열을 구문 분석하는 데 실패했습니다.:

설명

내부 문자열 값을 구문 분석하는 도중 스토어드 프로시저 내부의 메소드에 오류가 발생했습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

5001: 유틸리티 숫자 형식이 제대로 초기화되지 않았습니다. 사용 가능한 메모리가 충분한지 확인하십시오.:

설명

ICU 숫자 포맷터가 스토어드 프로시저에 제대로 초기화되지 않았습니다. 스토어드 프로시저 프로세스에서 사용 가능한 메모리 자원이 부적합한 것이 원인일 수 있습니다. 또한 스토어드 프로시저에서 로드한 ICU 라이브러리의 버전이 잘못된 것이 원인일 수 있습니다.

사용자 조치

분리(fenced) 스토어드 프로시저 프로세스에서 사용 가능한 메모리를 늘리고 스토어드 프로시저에 호출을 다시 발행하십시오. 현재 버전의 스토어드 프로시저에 사용하려는 ICU 라이브러리 버전을 로드 중인지 확인하십시오. 환경에 설정된 런타임 라이브러리 검색 경로를 점검하여 올바른 설정인지를 판별해야 할 수도 있습니다.

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

5002: 유틸리티 자원 번들이 초기화되지 않았습니다. 오류 코드=code. 번들이 존재 하며 path_name 경로에 있는지 확인하십시오.:

설명

ICU 자원 번들이 스토어드 프로시저에 제대로 초기화되지 않았습니다. 스토어드 프로시저에서 적합한 메모리 자원을 사용할 수 없거나 스토어드 프로시저에서 잘못된 버전의 ICU 라이브러리를 로드했거나 스토어드 프로시저에서 잘못된 자원 번들을 로드했기 때문에 제대로 초기화되지 않았을 수 있습니다.

사용자 조치

분리(fenced) 스토어드 프로시저 프로세스에서 사용 가능한 메모리를 늘리고 스토어드 프로시저에 호출을 다시 발행하십시오. 현재 버전의 스토어드 프로시저에 사용하려는 ICU 라이브러리 버전을 로드 중인지 확인하십시오. 환경에 설정된 런타임 라이브러리 검색 경로를 점검하여 올바른 설정인지를 판별해야 할 수도 있습니다. 스토어드 프로시저의 자원 번들이 올바른 버전으로 데이터베이스 서버 시스템에 설치되었는지 확인하십시오.

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

5003: 환경 변수 variable_name의 데이터 경로를 찾을 수 없습니다. 환경 변수가 올바르게 설정되었는지 점검하십시오.:

설명

스토어드 프로시저에서 사용하는 DB2 환경 변수가 설정되지 않았습니다.

사용자 조치

DB2를 시스템에 제대로 설치했는지 확인하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

5004: 목표 스트림이 닫혔습니다.:

설명

스토어드 프로시저 내부에서 사용하는 데이터 스트림이 예기치 않게 닫혔습니다. 데이터베이스 시스템에서 사용 가능한 filehandle이 충분하지 않을 수 있습니다.

사용자 조치

운영 체제에서 사용 가능한 filehandle이 충분한지 확인하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

5005: 목표에서 디폴트 인코딩을 사용하여 문자를 쓰고 있습니다.:

설명

스토어드 프로시저가 스토어드 프로시저에 문서화된 디폴트 인코딩을 사용하여 데이터베이스 서버 파일 시스템의 파일에 데이터를 작성합니다.

사용자 조치

스토어드 프로시저에서 작성한 파일을 읽는 응용프로그램은 스토어드 프로시저의 디폴트 인코딩에 인코딩된 데이터를 해석할 수 있어야 합니다.

5006: string의 입력 로그 문자열이 기록되지 않았습니까. 문자열의 레벨은 string_level이며 로그 레벨은 log_level입니다.:

설명

현재 로그 레벨에서는 표시된 메시지를 스토어드 프로시저에 설정된 로그 파일 중 하나에 작성할 수 없습니다.

사용자 조치

표시된 메시지를 스토어드 프로시저의 로그 파일 중 하나에 작성해야 하는 경우 로그 레벨을 수정하십시오.

5007: 오류 코드 code의 메시지 텍스트를 찾을 수 없습니다.:

설명

표시된 오류 코드의 텍스트를 스토어드 프로시저의 자원 번들 파일에서 찾지 못했습니다. 잘못된 버전의 자원 번들 파일을 사용했을 수도 있습니다.

사용자 조치

스토어드 프로시저의 자원 번들이 올바른 버전으로 데이터베이스 서버 시스템에 설치되었는지 확인하십시오. 후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

5008: 전역 정적 MsgBase 오브젝트의 operation 조작에 액세스하는 도중 실패했습니다.:

설명

내부 메시지 오브젝트에 액세스하려는 도중 스토어드 프로시저에서 오류가 발생했습니다.

사용자 조치

후속 지원은 IBM 고객만족센터에 문의하십시오. 시도한 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

External API 오류

6000-6199

6000: OLAPMSG()가 실패했습니다. 오류 코드는 code입니다.:

설명

실행 중 스토어드 프로시저가 실패했습니다.

사용자 조치

리턴 코드에 따라 문제점을 수정하고 스토어드 프로시저에 대한 호출을 다시 발행하거나 IBM 고객만족센터에 문의하여 추가 지원을 받으십시오. IBM 고객만족센터에 문의하는 경우 시도된 메타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

6001: object_name 오브젝트에 대해 인스턴스화된 SQL 템플릿이 value 값에 대해 유효하지 않습니다. 이유 ID: ID, 메시지: message.:

설명

인스턴스화된 SQL 템플릿은 복합 속성이나 복합 치수에 관련된 모든 속성 및 치수에 대한 SQL 표현식 템플릿을 결합하여 구성할 수 있는 SQL문 조각입니다. 지정된 오브젝트에 대해 인스턴스화된 SQL 템플릿에 문제점이 발견되었습니다.

사용자 조치

지정된 이유 ID 및 메시지를 참조하십시오. 이유 메시지에서 제시한 변경사항을 적용한 후 스토어드 프로시저에 대한 호출을 다시 발행하십시오.

6002: object1 오브젝트가 object2 오브젝트를 참조하지만 object2 오브젝트가 데이터베이스에 없습니다.:

설명

데이터베이스 오브젝트는 다른 오브젝트가 데이터베이스에 있는 경우에만 다른 오브젝트를 참조할 수 있습니다.

사용자 조치

데이터베이스에서 참조할 오브젝트를 작성한 다음 메타데이터 조작 요청을 다시 발행하십시오. 또는 누락된 오브젝트에 대한 참조를 제거한 다음 메타데이터 조작 요청을 다시 발행하십시오.

6003: 지정된 경로에 있는 log_name 로그를 열 수 없습니다. 지정된 경로가 존재하며 파일에 쓰기 액세스 권한이 있는지 확인하십시오.:

설명

스토어드 프로시저에서 사용한 하나 이상의 로그 파일을 열 수 없습니다.

사용자 조치

스토어드 프로시저 구성 파일에 지정된 경로가 있는지 확인하십시오. 데이터베이스 서버에서 스토어드 프로시저를 실행하는 사용자 ID가 필수 로그 파일 작성, 읽기 및 쓰기 권한을 가지고 있는지 확인하십시오.

6005: 이 작업에 대해 입력 메타데이터 매개변수가 예기치 않게 비어 있습니다. 누락된 메타데이터 매개변수가 이 작업에 필요합니다.:

설명

요청된 메타데이터 조작에서는 입력으로 메타데이터가 제공되어야 합니다. 메타데이터 교환을 위한 스토어드 프로시저 매개변수가 비어 있습니다.

사용자 조치

메타데이터 조작 요청을 필수 메타데이터로 다시 발행하십시오.

6006: 검색 기준과 일치하는 오브젝트를 찾을 수 없습니다. search_criteria.:

설명

메타데이터 조작으로 지정된 검색 기준에 맞는 메타데이터 오브젝트를 찾지 못했습니다. 메타데이터 카탈로그 콘텐츠에 대한 변경이 없습니다.

사용자 조치

메타데이터 카탈로그의 콘텐츠를 변경하려는 경우 메타데이터 조작을 새 검색 기준으로 다시 발행하십시오.

6007: 카탈로그에 있는 오브젝트와 임포트 되고 있는 오브젝트 사이에 충돌이 있습니다. 메타데이터를 변경하지 않았습니다.:

설명

임포트 중인 오브젝트와 메타데이터 카탈로그에 있는 오브젝트 사이에 충돌이 발견되었습니다. 지정한 임포트 모드 때문에 메타데이터 카탈로그의 오브젝트에 대한 변경사항이 없습니다.

사용자 조치

메타데이터 카탈로그의 콘텐츠를 변경하려는 경우 메타데이터 조작을 다른 임포트 모드를 사용하여 다시 발행하십시오.

6008: 중복 ID(가) metadata_input 내에서 object ID로 존재합니다.:

설명

이 메타데이터 조작에 대한 메타데이터 입력에서 중복 메타데이터 오브젝트가 발견되었습니다. 메타데이터 조작에 대한 입력으로 중복 오브젝트는 허용되지 않습니다.

사용자 조치

입력 메타데이터에서 중복 메타데이터 오브젝트를 제거하고 메타데이터 조작을 다시 발행하십시오.

6009: 입력 object_name 오브젝트와 같은 ID를 공유하는 오브젝트가 이미 메타데이터 카탈로그에 존재합니다.:

설명

동일한 ID를 가진 메타데이터 오브젝트가 메타데이터 카탈로그에 있기 때문에 메타데이터 조작을 수행할 수 없습니다.

사용자 조치

실패한 메타데이터 조작을 다시 발행하기 전에 작성 중인 오브젝트와 동일한 ID를 공유하는 메타데이터 카탈로그에서 오브젝트를 삭제하십시오. 또는 작성 중인 새 오브젝트의 등록 정보와 일치하도록 기존 오브젝트를 변경할 수도 있습니다. 그렇지 않으면 수행 중인 메타데이터 조작에서 이 오류를 발생시키는 새 오브젝트를 제외해야 합니다.

6010: 입력 object_name 오브젝트의 object_name 오브젝트 참조가 이미 존재합니다.:

설명

지정된 오브젝트 사이의 참조가 메타데이터 카탈로그에 이미 정의되어 있습니다. 중복 참조는 허용되지 않습니다.

사용자 조치

메타데이터 조작 요청에서 중복 참조 중 하나를 제거하십시오.

6011: 이름 바꾸기 조작을 사용하여 object_name 오브젝트의 스키마는 이름바꾸기 조작을 사용하여 변경할 수 없습니다.:

설명

이름 바꾸기 조작을 사용하여 메타데이터 오브젝트의 스키마를 변경할 수 없습니다.

사용자 조치

이름 바꾸기 중인 조작에 지정된 스키마가 일정한 상태인지 확인하고 아니면 변경 조작을 사용하지십시오.

6013: 클라이언트가 사용한 XML 스키마의 버전 version1은 서버의 API에서 지원되지 않습니다. 서버의 API는 XML 스키마의 버전 version2를 지원합니다.:

설명

클라이언트가 사용하고 입력 매개변수 문자열에 임베드된 XML 스키마의 버전을 서버의 스토어드 프로시저 버전에서 지원하지 않습니다.

사용자 조치

클라이언트 응용프로그램과 스토어드 프로시저가 이 제품과 함께 발행된 XML 스키마의 동일한 버전을 사용하는지 확인하십시오.

6014: object_name 오브젝트의 SQL 템플릿을 명확히 할 수 없습니다. 이유 ID: ID, 메시지: message.:

설명

스토어드 프로시저는 복합 속성이나 복합 치수에 관련된 모든 속성 및 치수에 대한 SQL 표현식 템플릿을 결합하여 속성 및 치수에 대한 SQL 템플릿을 구성할 수 있습니다. SQL 템플릿을 구성하는 중 스토어드 프로시저에 문제가 발생했습니다.

사용자 조치

지정된 이유 ID 및 메시지를 참조하십시오. 이유 메시지에서 제시한 변경사항을 적용한 후 스토어드 프로시저에 대한 호출을 다시 발행하십시오.

6015: 데이터베이스 사용자 ID에 활성 데이터베이스에서 데이터베이스 스키마를 작성할 수 있는 권한이 없습니다.:

설명

데이터베이스 서버의 스토어드 프로시저 프로세스를 소유하는 사용자 ID가 활성 데이터베이스에서 데이터베이스 스키마를 작성할 권한을 가지고 있지 않습니다. 각 고유 메타데이터 오브젝트 스키마에 대해 데이터베이스 스키마가 작성됩니다.

사용자 조치

클라이언트와 서버에서 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 스토어드 프로시저 프로세스를 소유하는 사용자 ID에 활성 데이터베이스의 스키마를 작성할 권한을 부여하십시오. 스토어드 프로시저에 대한 호출을 다시 발행하십시오.

6016: 데이터베이스 사용자 ID는 활성 데이터베이스에서 필수 조치를 수행할 권한이 없습니다. 데이터베이스 서버에서 다음 오류 메시지가 리턴되었습니다. message.:

설명

데이터베이스 서버의 스토어드 프로시저 프로세스를 소유하는 사용자 ID가 활성 데이터베이스의 필수 조치를 수행할 권한을 가지고 있지 않습니다.

사용자 조치

클라이언트와 서버에서 데이터베이스 관리 프로그램 로그 파일을 점검하십시오. 스토어드 프로시저 프로세스를 소유하는 사용자 ID에 필수 권한을 부여하십시오. 스토어드 프로시저에 대한 호출을 다시 발행하십시오.

6017: object_name 오브젝트는 메타데이터 카탈로그에 존재하지 않습니다.:

설명

요청된 조작에서는 지정된 오브젝트가 메타데이터 카탈로그에 있어야 합니다.

사용자 조치

메타데이터 조작 요청을 다시 발행하기 전에 메타데이터 카탈로그에서 지정된 오브젝트를 작성하십시오.

6018: 필수 테이블이 데이터베이스에 존재하지 않습니다. 데이터베이스 서버에서 다음 오류 메시지가 리턴되었습니다. message.:

설명

요청 조작에서 필요로 하는 테이블이 데이터베이스에 없습니다.

사용자 조치

누락된 테이블이 사용자 테이블인 경우 테이블을 작성하고 메타데이터 조작 요청을 다시 발행하십시오. 누락된 테이블이 메타데이터 카탈로그 테이블이거나 데이터베이스 시스템 테이블인 경우 IBM 고객만족센터에 문의하여 추가 지원을 받으십시오. 시도한 메

타데이터 조작의 상태 ID 및 텍스트를 제공하십시오. 가능하면 데이터베이스 서버의 스토어드 프로시저 로그 파일도 제공하십시오.

6019: API 작업은 XML의 version_number 버전을 사용하며, 메타데이터는 XML의 버전 version_number를 사용합니다. API 작업 및 메타데이터 XML의 버전이 일치해야 합니다.:

설명

API 조작 XML 버전이 메타데이터 XML 버전과 일치해야 합니다.

사용자 조치

API 조작 XML과 메타데이터 XML이 동일한 버전 번호를 사용하고 있는지 확인하십시오.

6020: operation_name API 조작이 메타데이터 XML의 버전 version_number를 지원하지 않습니다.:

설명

API 조작 XML 버전은 8.2.0.1.0이어야 하고, TRANSLATE 조작의 경우 메타데이터 XML 버전은 8.1.2.1.0이어야 합니다.

사용자 조치

메타데이터 규칙, 메타데이터 유효성 확인 및 쿼리 최적화에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6021: DESCRIBE 작업에 대해서만 version_number XML을 사용할 수 있습니다.:

설명

DESCRIBE 조작에서만 지정된 XML 버전을 사용할 수 있습니다.

사용자 조치

메타데이터 규칙, 메타데이터 유효성 확인 및 쿼리 최적화에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

유효성 확인 오류

6200: object_name 오브젝트가 완전하지 않습니다. 필수 등록 정보가 설정되었는지 확인하십시오.:

설명

지정된 입력 오브젝트에 일부 필수 등록 정보 세트가 없습니다.

사용자 조치

해당 오브젝트에 필수 등록 정보를 설정하고 메타데이터 조작 요청을 다시 발행하십시오.

6201: hierarchy_name 계층 구조는 반복 유형이지만 2개의 속성만 가지고 있는 것이 아니기 때문에 유효하지 않습니다.:

설명

반복 계층 구조는 정확히 두 개의 속성을 참조해야 합니다. 식별 계층 구조가 이 규칙을 위반합니다.

사용자 조치

식별된 계층 구조가 정확히 두 개의 속성을 참조하도록 수정하십시오. 메타데이터 규칙에 대한 자세한 정보는 *설정 및 사용자 안내서*를 참조하십시오.

6202: object_name 오브젝트에는 최소한 한 개의 SQL 템플릿이 있어야 합니다.:

설명

제품 문서에 나와 있는 메타데이터 오브젝트 정의에 따라 식별된 치수에 하나 이상의 SQL 템플릿을 정의해야 합니다.

사용자 조치

하나 이상의 SQL 템플릿을 정의하도록 식별된 치수를 수정하십시오. 메타데이터 규칙에 대한 자세한 정보는 *설정 및 사용자 안내서*를 참조하십시오.

6206: attribute_name 속성에는 한 개의 SQL 템플릿만 있어야 합니다.:

설명

제품 문서에 나와 있는 메타데이터 오브젝트 정의에 따라 식별된 치수에 하나의 SQL 템플릿만 정의해야 합니다.

사용자 조치

하나의 SQL 템플릿만 정의하도록 식별된 치수를 수정하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

6207: attribute_name 속성은 조인의 일부이지만 컬럼 참조가 없습니다.:

설명

식별된 속성 오브젝트는 메타데이터 조인 오브젝트가 유효하게 참조하도록 데이터베이스 컬럼을 참조해야 합니다.

사용자 조치

데이터베이스 컬럼을 참조하도록 식별된 속성 오브젝트를 수정하거나 데이터베이스 컬럼을 참조하는 다른 속성 오브젝트를 참조하도록 관련 조인 오브젝트를 수정하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6208: attribute_name 속성은 조인의 일부이며 table_name과 동일한 테이블을 가리켜야 합니다.:

설명

첫 번째 식별된 속성 오브젝트는 다른 식별된 오브젝트와 동일한 데이터베이스 테이블을 참조해야 합니다.

사용자 조치

다른 식별된 오브젝트와 동일한 데이터베이스 테이블을 참조하도록 첫 번째 식별된 속성을 수정하거나 동일한 데이터베이스 테이블을 참조하는 다른 속성 오브젝트를 참조하도록 관련 조인 오브젝트를 수정하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6209: object_name 오브젝트의 스키마가 최대 길이를 초과했습니다.:

설명

식별된 오브젝트의 스키마가 최대 길이를 초과합니다.

사용자 조치

식별된 오브젝트의 스키마를 줄이십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6210: object_name 오브젝트의 이름이 최대 길이를 초과했습니다.:

설명

식별된 오브젝트의 이름이 최대 길이를 초과합니다.

사용자 조치

식별된 오브젝트의 이름을 줄이십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6211: object_name 오브젝트의 테이블 이름이 최대 길이를 초과했습니다.:

설명

식별된 오브젝트의 테이블 이름이 최대 길이를 초과합니다.

사용자 조치

테이블의 이름을 줄이십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

6212: object_name 오브젝트의 비즈니스 이름이 최대 길이를 초과했습니다.:

설명

식별된 오브젝트의 비즈니스 이름이 최대 길이를 초과합니다.

사용자 조치

비즈니스 이름을 줄이십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

6213: object_name 오브젝트의 주석이 최대 길이를 초과했습니다.:

설명

식별된 오브젝트의 주석이 최대 길이를 초과합니다.

사용자 조치

식별된 오브젝트의 주석을 줄이십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

6214: object_name 오브젝트의 스키마는 SYS로 시작할 수 없습니다.:

설명

메타데이터 오브젝트의 스키마는 문자열 SYS로 시작할 수 없습니다.

사용자 조치

메타데이터 오브젝트에 대해 SYS로 시작하지 않는 스키마를 사용하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

6215: object_name 오브젝트의 스키마는 SESSION이 될 수 없습니다.:

설명

메타데이터 오브젝트의 스키마는 문자열 SESSION이 될 수 없습니다.

사용자 조치

메타데이터 오브젝트에 대해 문자열 SESSION이 아닌 스키마를 사용하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

6216: object_name 오브젝트의 이름 및 스키마가 완료되지 않았습니다. 이유 ID=ID, 메시지=message.:

설명

식별된 오브젝트의 이름 또는 스키마 또는 둘 모두가 누락되었거나 유효하지 않습니다.

사용자 조치

식별된 오브젝트의 이름과 스키마 모두에 유효한 문자열을 제공하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6217: hierarchy_name 큐브 계층 구조에서 참조하지 않는 레벨을 참조하므로 **hierarchy_name** 계층 구조는 유효하지 않습니다.:

설명

큐브 계층 구조가 상위 계층 구조가 참조하지 않는 레벨을 참조합니다.

사용자 조치

상위 계층 구조에서도 참조하는 레벨만 참조하도록 큐브 계층 구조를 변경하십시오. 또는 큐브 계층 구조가 참조하는 것과 동일한 레벨을 참조하도록 계층 구조를 변경할 수도 있습니다. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

경고 코드

6250: level_name 레벨 키 속성이 하나의 테이블 컬럼으로 매핑되지 않으므로 API는 **level_name** 레벨 오브젝트에 대한 함수 종속성을 작성할 수 없습니다.:

설명

레벨 키 속성이 단일 테이블 컬럼으로 매핑되지 않기 때문에 API가 레벨 오브젝트에 대한 함수 종속성을 작성할 수 없습니다.

사용자 조치

함수 종속성에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6251: 레벨 키 속성이 차원 테이블에 관한 기존의 고유한 제한조건에 대응하므로 API는 level_name 레벨 오브젝트에 대해 함수 종속성을 작성하지 않습니다.:

설명

레벨 키 속성이 차원 테이블의 고유한 기존 제한조건과 일치하기 때문에 API가 레벨 오브젝트에 대한 함수 종속성을 작성하지 않습니다. 함수 종속성이 필요하지 않습니다.

사용자 조치

함수 종속성에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6252: level_name 레벨 키 속성이 널(NULL) 입력 가능하므로 API는 **level_name** 레벨 오브젝트에 대해 함수 종속성을 작성할 수 없습니다.:

설명

레벨 키 속성 중 하나가 널(NULL) 입력 가능한 경우 API가 레벨 오브젝트에 대한 함수 종속성을 작성할 수 없습니다.

사용자 조치

함수 종속성에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6253: 속성이 둘 이상의 테이블에 걸쳐 있으므로 API는 level_name 레벨 오브젝트에 대해 함수 종속성을 작성할 수 없습니다.:

설명

레벨 키 속성, 디폴트 속성 및 관련 속성이 둘 이상의 테이블에 걸쳐 있는 경우 API가 레벨 오브젝트에 대한 함수 종속성을 작성할 수 없습니다.

사용자 조치

함수 종속성에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6254: 해당 속성이 레벨 키 속성으로 이미 포함되어 있으므로 level_name 레벨 오브젝트에 대한 함수 종속성은 **attribute_name** 디폴트 또는 관련 속성을 제외합니다.:

설명

레벨 오브젝트에 대한 함수 종속성이 레벨 키 속성으로 이미 포함되어 있는 관련 속성을 제외합니다.

사용자 조치

함수 종속성에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6255: 디폴트 또는 관련 속성이 한 테이블 컬럼에 매핑되지 않기 때문에 level_name 레벨 오브젝트에 대한 함수 종속성이 **attribute_name** 디폴트 또는 관련 속성을 제외시킵니다.:

설명

해당 속성이 한 테이블 컬럼에 매핑되지 않기 때문에 레벨 오브젝트에 대한 함수 종속성이 디폴트 또는 관련 속성을 제외합니다.

사용자 조치

함수 종속성에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6256: 모든 디폴트 및 관련 속성이 제외되었으므로 API는 level_name 레벨 오브젝트에 대해 함수 종속성을 작성할 수 없습니다.:

설명

최소 하나의 디폴트 속성 또는 관련 속성이 레벨 오브젝트와 일치하는 함수 종속성을 작성하는 데 필요합니다.

사용자 조치

함수 종속성에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6257: API가 함수 종속성 작성 시도 중에 오류가 발생했으므로 level_name 레벨 오브젝트에 대해 함수 종속성을 작성할 수 없습니다.:

설명

API가 함수 종속성을 작성하는 SQL문을 실행하려는 중에 오류가 발생했습니다.

사용자 조치

함수 종속성에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오. 서버 로그의 해당 항목을 선택하여 자세한 정보를 찾아 볼 수도 있습니다.

6258: 연관된 dependency_name 함수 종속성을 삭제하려는 중 오류가 발생했기 때문에 API가 level_name 레벨 오브젝트를 변경하거나 삭제할 수 없습니다.:

설명

API가 함수 종속성을 삭제하는 SQL 문을 발행하려는 중 오류가 발생했기 때문에 DB2 Cube Views가 레벨 오브젝트를 변경하거나 삭제할 수 없습니다. 함수 종속성을 작성하거나 삭제할 권한이 있는지 확인하십시오.

사용자 조치

함수 종속성에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오. 자세한 정보는 서버 로그의 항목을 점검하십시오.

6299: 유효성 확인 중에 최소한 한 개의 데이터베이스 뷰가 발견되었습니다. 뷰의 컬럼이 포함하는 조인의 제한조건 관련 유효성 확인 점검은 수행되지 않았습니다. 기타 모든 유효성 확인 점검은 수행되었습니다.:

설명

뷰의 컬럼을 포함하는 조인에 대해 제한조건 관련 유효성 확인이 수행되지 않았습니다. 다른 모든 요청된 조인에 대해서는 제한조건 관련 유효성 확인이 수행되었으며 요청된 모든 오브젝트에 대해 나머지 모든 유효성 확인도 수행되었습니다.

사용자 조치

메타데이터 규칙, 메타데이터 유효성 확인 및 쿼리 최적화에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

규칙 관련 유효성 확인 오류

6300: model_name 큐브 모델이 하나 이상의 사실을 참조하지 않습니다.:

설명

식별된 큐브 모델 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 큐브 모델은 하나 이상의 사실을 참조해야 합니다.

사용자 조치

하나 이상의 사실을 참조하도록 식별된 큐브 모델을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

6301: model_name 큐브 모델이 0개 이상의 차원을 참조하지 않습니다.:

설명

식별된 큐브 모델 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 큐브 모델은 0개 이상의 차원을 참조해야 합니다.

사용자 조치

0개 이상의 차원을 참조하도록 식별된 큐브 모델을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

6302: 차원-조인 쌍 중 하나의 model_name 큐브 모델에서 차원 및/또는 조인이 누락되었습니다.:

설명

식별된 큐브 모델 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 큐브 모델의 차원-조인 쌍은 차원과 조인 모두를 참조해야 합니다.

사용자 조치

모든 차원-조인 쌍이 차원과 조인을 모두 참조하도록 식별된 큐브 모델을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

6303: 큐브 모델 model_name에 의해 참조된 join_name 조인이 유효하지 않습니다. 한쪽의 모든 속성은 facts_name 사실에 의해 참조해야 하며, 다른 쪽의 모든 속성은 큐브 모델의 차원 중 하나에 의해 참조되어야 합니다.:

설명

식별된 큐브 모델 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 큐브 모델의 조인의 한쪽은 큐브 모델 사실의 속성을 참조하고 다른 쪽은 큐브 모델 차원 중 하나의 속성을 참조해야 합니다.

사용자 조치

조인 한 쪽의 모든 속성이 큐브 모델 사실에서 나오고 다른 쪽의 모든 속성은 큐브 모델 차원에서 나오도록 식별된 큐브 모델에 유효하지 않은 조인을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6304: measure_name치수에 있는 집계 중 하나가 **model_name** 큐브 모델이 직접 참조하지 않는 **dimension_name** 차원을 직접 참조합니다.:

설명

식별된 큐브 모델 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 큐브 모델에서 사용한 치수의 집계는 동일한 큐브 모델에서 사용한 차원만 참조해야 합니다.

사용자 조치

식별된 큐브 모델에서 사용한 차원만 참조하도록 식별된 치수의 집계를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6305: measure_name 치수내에서 비어 있는 차원-목록 집계가 **model_name** 큐브 모델에서 이전에 일치하지 않은 최소한 하나의 차원과 일치하지 않습니다.:

설명

식별된 큐브 모델 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 큐브 모델에서 사용하는 치수의 빈-치수-목록 집계는 각 큐브 모델에서 일치하지 않는 하나 이상의 치수와 일치해야 합니다.

사용자 조치

빈-차원-목록이 식별된 큐브 모델에서 이전에 일치하지 않은 하나 이상의 차원과 일치하도록 식별된 치수의 집계를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6306: model_name 큐브 모델은 차원 오브젝트를 참조하지 않으므로 **measure_name** 치수에는 비어 있는 차원-목록 집계만 포함되어야 합니다.:

설명

식별된 큐브 모델 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 큐브 모델이 차원을 참조하지 않는 경우 큐브 모델의 치수에는 빈-차원-목록 집계만 포함되어야 합니다.

사용자 조치

빈-차원-목록 집계만 포함하도록 식별된 치수를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6307: cube_name 큐브가 하나의 큐브 사실 오브젝트를 참조하지 않습니다.:

설명

식별된 큐브 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 큐브는 하나의 큐브 사실 오브젝트를 참조해야 합니다.

사용자 조치

하나의 큐브 사실 오브젝트를 참조하도록 식별된 큐브를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6308: cube_name 큐브가 최소한 하나 이상의 큐브 차원 오브젝트를 참조하지 않습니다.:

설명

식별된 큐브 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 큐브는 최소한 하나의 차원 오브젝트를 참조해야 합니다.

사용자 조치

하나 이상의 큐브 차원 오브젝트를 참조하도록 식별된 큐브를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6309: cube_name 큐브에 의해 참조된 cube_facts_name 큐브 사실은 model_name 큐브 모델에 의해 참조된 사실 오브젝트에서 파생된 것이 아닙니다.:

설명

식별된 큐브 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 식별된 큐브에서 사용하는 큐브 사실은 식별된 큐브 모델에서 사용하는 사실에서 파생되어야 합니다.

사용자 조치

지정된 규칙이 위반되지 않도록 하나 이상의 식별된 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6310: cube_name 큐브에 의해 참조된 **cube_dimension_name** 큐브 차원은 **model_name** 큐브 모델에 의해 참조된 차원 오브젝트 중 하나에서 파생된 것이 아닙니다.:

설명

식별된 큐브 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 식별된 큐브에서 사용하는 큐브 차원은 식별된 큐브 모델에서 사용하는 차원 중 하나에서 파생되어야 합니다.

사용자 조치

지정된 규칙이 위반되지 않도록 하나 이상의 식별된 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6311: facts_name 사실 오브젝트가 치수를 참조하지 않습니다.:

설명

식별된 사실 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 사실 오브젝트는 최소한 하나의 치수를 참조해야 합니다.

사용자 조치

하나 이상의 차원을 참조하도록 식별된 사실을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6312: 사실 오브젝트 facts_name이 참조한 속성 및 치수의 일부가 사실 오브젝트 조인을 사용하여 조인될 수 없습니다.:

설명

식별된 사실 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 사실 오브젝트의 속성 및 치수는 사실의 조인 오브젝트를 사용하여 모두 조인 가능해야 합니다.

사용자 조치

사실 오브젝트의 추가 조인 오브젝트를 참조하여 식별된 사실 오브젝트가 참조한 모든 속성 및 치수를 조인 가능하도록 하십시오. 또는 사실의 현재 조인에서 조인 가능하지 않은 사실 오브젝트 또는 치수에서 해당 속성을 제거하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6313: facts_name 사실 오브젝트에 두 테이블 간의 여러 조인이 여러 개 있습니다.:

설명

식별된 사실 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 사실 오브젝트는 동일한 두 테이블 사이에 여러 조인을 가질 수 없습니다.

사용자 조치

두 테이블 사이에 하나의 조인만 갖도록 식별된 사실 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6314: facts_name 사실 오브젝트에 조인 루프가 있습니다.:

설명

식별된 사실 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 식별된 사실 오브젝트의 조인이 경로 루프를 형성합니다. 이것은 허용되지 않습니다.

사용자 조치

루프를 일으키는 조인 중 하나를 식별된 사실 오브젝트에서 제거하거나 루프가 없도록 루프를 일으키는 조인 중 하나를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6315: join_name 조인은 facts_name 사실 오브젝트에서 이 속성만을 참조하지 않습니다.:

설명

식별된 사실 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 사실 오브젝트의 조인은 해당 사실 오브젝트의 속성만 참조해야 합니다.

사용자 조치

식별된 사실 오브젝트의 속성만 참조하도록 식별된 조인을 변경하거나 식별된 조인 오브젝트가 참조하는 누락된 속성을 사실 오브젝트에 추가하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6316: cube_facts_name 큐브 사실 오브젝트가 사실 오브젝트를 참조하지 않거나 둘 이상의 사실 오브젝트를 참조합니다.:

설명

식별된 큐브 사실 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 큐브 사실 오브젝트는 하나의 사실 오브젝트를 참조해야 합니다.

사용자 조치

하나의 사실 오브젝트를 참조하도록 식별된 큐브 사실 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6317: cube_facts_name 큐브 사실 오브젝트가 치수를 참조하지 않습니다.:

설명

식별된 큐브 사실 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 큐브 사실 오브젝트는 최소한 하나의 치수를 참조해야 합니다.

사용자 조치

하나 이상의 차원을 참조하도록 식별된 큐브 사실 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6318: cube_facts_name 큐브 사실 오브젝트에 의해 참조된 **measure_name** 치수는 **facts_name** 사실 오브젝트의 일부가 아닙니다.:

설명

식별된 큐브 사실 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 큐브 사실 오브젝트는 큐브 사실 오브젝트가 파생된 사실 오브젝트에 의해 참조되는 치수를 참조해야 합니다.

사용자 조치

식별된 사실 오브젝트에 식별된 치수를 추가하거나 식별된 큐브 사실 오브젝트에서 식별된 치수를 제거하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6319: dimension_name 차원이 속성을 참조하지 않습니다. 차원은 최소한 하나의 속성을 참조해야 합니다.:

설명

식별된 차원 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 차원 오브젝트는 최소한 하나의 속성을 참조해야 합니다.

사용자 조치

하나 이상의 속성을 참조하도록 식별된 차원 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6320: dimension_name 차원에서 참조한 속성의 일부를 차원 조인을 사용하여 조인할 수 없습니다.:

설명

식별된 차원 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 차원 오브젝트의 속성은 차원의 조인 오브젝트를 사용하여 모두 조인 가능해야 합니다.

사용자 조치

차원 오브젝트의 추가 조인 오브젝트를 참조하여 식별된 차원 오브젝트가 참조한 모든 속성을 조인 가능하도록 하십시오. 또는 차원 오브젝트에서 차원의 현재 조인에 의해 조인할 수 없는 속성을 제거하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6321: dimension_name 차원에 조인 루프가 있습니다.:

설명

식별된 차원 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 식별된 차원 오브젝트의 조인이 경로 루프를 형성합니다. 이것은 허용되지 않습니다.

사용자 조치

식별된 차원 오브젝트에서 루프를 일으키는 조인 중 하나를 제거하거나 루프가 없도록 루프를 일으키는 조인 중 하나를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6322: dimension_name 차원에 두 테이블 간의 조인이 여러 개 있습니다.:

설명

식별된 차원 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 차원 오브젝트는 동일한 두 테이블 사이에 여러 조인을 가질 수 없습니다.

사용자 조치

두 테이블 사이에 하나의 조인만 갖도록 식별된 차원 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 제품 문서를 참조하십시오.

6323: hierarchy_name 계층 구조가 계층 구조 상위 차원 hierarchy_name이 참조하지 않은 레벨을 참조합니다.:

설명

차원의 계층 구조는 해당 차원 오브젝트의 레벨만 참조할 수 있습니다. 이 계층 구조는 상위 차원이 참조하지 않는 레벨을 참조합니다.

사용자 조치

상위 차원 오브젝트의 레벨만 참조하도록 계층 구조를 변경하거나 계층 구조가 차원 오브젝트를 참조하는 레벨을 추가하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6325: 차원의 조인은 해당 차원의 속성만을 참조해야 합니다. join_name 조인이 dimension_name 차원에 존재하지 않는 속성을 참조합니다.:

설명

차원의 조인이 해당 차원의 속성만 참조해야 한다는 메타데이터 오브젝트 규칙을 차원 및 조인이 위반합니다.

사용자 조치

상위 차원의 속성만 참조하도록 조인을 변경하거나 조인이 해당 차원을 참조하는 속성을 추가하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6326: cube_dimension_name 큐브 차원이 차원을 참조하지 않습니다.:

설명

식별된 큐브 차원 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 큐브 차원 오브젝트는 차원을 참조해야 합니다.

사용자 조치

차원을 참조하도록 식별된 큐브 차원 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6327: cube_dimension_name 큐브 차원이 큐브 계층 구조를 참조하지 않습니다.:

설명

식별된 큐브 차원 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 큐브 차원 오브젝트는 큐브 계층 구조를 참조해야 합니다.

사용자 조치

큐브 계층 구조를 참조하도록 식별된 큐브 차원 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6328: cube_dimension_name 큐브 차원에 의해 참조된 **cube_hierarchy_name** 큐브 계층 구조는 **dimension_name** 차원에 의해 참조된 계층 구조에서 파생되지 않았습니다.:

설명

식별된 큐브 차원 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 식별된 큐브 차원에서 사용하는 큐브 계층 구조는 식별된 차원에서 사용하는 계층 구조 중 하나에서 파생되어야 합니다.

사용자 조치

지정된 규칙이 위반되지 않도록 하나 이상의 식별된 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6329: hierarchy_name 계층 구조가 레벨을 참조하지 않습니다.:

설명

계층 구조 오브젝트는 최소한 하나의 레벨을 참조해야 합니다.

사용자 조치

하나 이상의 레벨을 참조하도록 식별된 계층 구조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6330: 반복 전개를 사용하는 hierarchy_name 계층 구조가 정확히 두 개의 레벨을 참조하지 않습니다.:

설명

식별된 계층 구조 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 반복 전개를 사용하는 계층 구조 오브젝트는 두 개의 레벨을 참조해야 합니다.

사용자 조치

두 개의 레벨을 참조하도록 식별된 계층 구조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6332: hierarchy_name 계층 구조의 유형이 전개와 호환되지 않습니다.:

설명

식별된 계층 구조 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 계층 구조 유형 및 전개의 호환성은 제품 문서에서 설명합니다.

사용자 조치

유형이 해당 전개와 호환되도록 식별된 계층 구조를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6334: hierarchy_name 큐브 계층 구조는 정확히 하나의 계층 구조를 참조해야 합니다.:

설명

이 큐브 계층 구조는 큐브 계층 구조가 정확히 한 계층 구조만 참조해야 한다는 메타데이터 오브젝트 규칙을 위반합니다.

사용자 조치

하나의 계층 구조만 참조하도록 식별된 큐브 계층 구조를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6335: hierarchy_name 큐브 계층 구조는 최소한 하나의 큐브 레벨을 참조해야 합니다.:

설명

식별된 큐브 계층 구조 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 큐브 계층 구조 계층은 최소한 하나의 큐브 레벨을 참조해야 합니다.

사용자 조치

하나 이상의 큐브 레벨을 참조하도록 식별된 큐브 계층 구조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6336: hierarchy_name1 큐브 계층 구조는 큐브 레벨을 참조하지만 대응하는 **hierarchy_name2** 계층 구조가 해당 **level_name** 레벨을 참조하지 않습니다. **level_name** 레벨을 **hierarchy_name2** 계층 구조에 추가하거나 **hierarchy_name1** 큐브 계층 구조에서 해당 큐브 레벨을 제거해야 합니다.:

설명

이 큐브 계층 구조는 큐브 계층 구조가 해당 계층이 참조한 레벨에서 파생된 큐브 레벨을 참조해야 한다는 메타데이터 오브젝트 규칙을 위반합니다.

사용자 조치

레벨을 계층 구조에 추가하거나 큐브 계층 구조에서 레벨을 제거하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6337: hierarchy_name 큐브 계층 구조의 큐브 레벨 순서가 **hierarchy_name** 계층 구조의 해당 레벨 순서와 일치하지 않습니다.:

설명

식별된 큐브 계층 구조 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 큐브 계층 구조에서 큐브 레벨의 상대적 순서는 큐브 계층 구조가 파생된 계층 구조의 동일한 레벨에 있는 상대적 순서와 동일해야 합니다.

사용자 조치

식별된 오브젝트 모두에 있는 속성의 상대적 순서가 일치하도록 식별된 오브젝트 중 하나를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6340: measure_name 치수에 대한 SQL 표현식 템플릿 중 하나가 속성, 치수 또는 컬럼이 아닌 매개변수를 사용합니다.:

설명

식별된 치수 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 치수 오브젝트의 SQL 표현식 템플릿은 속성, 치수 또는 컬럼인 매개변수를 사용해야 합니다.

사용자 조치

SQL 표현식 템플릿이 속성, 치수 또는 컬럼을 매개변수로 사용하도록 식별된 치수를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6341: SQL 표현식 템플릿 내에서 **measure_name** 치수에 대한 매개변수로 사용된 속성 또는 치수 사이에 종속성 루프가 존재합니다.:

설명

식별된 치수 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 치수의 SQL 표현식 템플릿에 대한 매개변수로 사용되는 속성 및 치수는 종속성 루프를 형성하지 않아야 합니다.

사용자 조치

SQL 표현식 템플릿이 매개변수에 관한 종속성 루프를 포함하지 않도록 식별된 치수를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6342: measure_name 치수에 SQL 표현식 템플릿 중 하나에 대해 정의된 비어 있는 문자열이 있습니다.:

설명

식별된 치수 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 치수의 SQL 표현식 템플릿은 빈 문자열일 수 없습니다.

사용자 조치

SQL 표현식 템플릿이 빈 문자열이 되지 않도록 식별된 치수를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6343: measure_name 치수의 SQL 표현식 템플릿에 집계 함수가 들어 있습니다.:

설명

식별된 치수 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 치수의 SQL 표현식 템플릿은 집계 포함할 수 없습니다.

사용자 조치

SQL 표현식 템플릿에 집계 함수가 포함되지 않도록 식별된 치수를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6344: measure_name 치수에 집계가 누락되었거나 치수 이외의 오브젝트를 잘못 참조하고 있습니다.:

설명

식별된 치수 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 해당 치수가 하나 이상의 다른 치수를 참조하고 치수만 참조하는 경우 치수에 집계는 필요하지 않습니다.

사용자 조치

집계를 추가하거나 식별된 치수가 하나 이상의 다른 치수를 참조하고 치수만 참조하도록 하여 식별된 치수를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6345: measure_name 치수의 SQL 표현식 템플릿 수가 첫 번째 집계 함수에 사용된 매개변수의 수와 일치하지 않습니다.:

설명

식별된 치수 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 집계가 존재하는 경우 치수의 SQL 템플릿의 수는 첫 번째 집계 함수의 매개변수 수와 일치해야 합니다.

사용자 조치

첫 번째 집계 함수의 매개변수 수가 치수의 SQL 표현식 템플릿 수와 일치하도록 식별된 치수를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6346: 여러 SQL 표현식 템플릿이 있는 measure_name 치수가 집계 스크립트에서 최소한 하나의 단계를 정의하지 않습니다.:

설명

식별된 치수 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 다중 SQL 표현식 템플릿의 치수는 집계 스크립트에서 최소한 하나의 단계를 정의해야 합니다.

사용자 조치

집계 스크립트가 하나 이상의 단계를 갖도록 식별된 치수를 수정하십시오. 또는 나머지 SQL 표현식 템플릿이 다른 치수만 참조한 경우 치수의 SQL 표현식 템플릿 중 하나를 제거하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6347: measure_name1 치수가 집계 스크립트를 정의했습니다. 그러나 참조된 치수 **measure_name2**가 SQL 표현식에 대해 여러 템플릿을 정의하기 때문에 집계 스크립트를 정의하지 않았어야 합니다.:

설명

식별된 치수 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 치수 A가 여러 SQL 템플릿을 정의하는 치수 B를 참조하는 경우 치수 A에는 집계 스크립트가 있으면 안 됩니다. 이 규칙은 치수 참조 트리의 모든 레벨에 적용됩니다.

사용자 조치

문제를 일으키는 치수에서 집계 스크립트를 제거하거나 참조된 치수에서 SQL 표현식 템플릿 중 하나를 제거하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6348: measure_name 치수에 첫 번째 집계로 사용되지 않은 여러 매개변수 집계 함수가 들어 있습니다.:

설명

식별된 치수 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 다중 매개변수 집계 함수는 치수의 첫 번째 집계로만 사용이 가능합니다.

사용자 조치

다중 매개변수 집계 함수를 식별된 치수가 사용하는 첫 번째 집계로 작성하거나 식별된 치수에서 다중 매개변수 집계를 제거하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6349: measure_name 치수가 정확히 하나의 빈-치수-목록 집계만 가지고 있지 않습니다.:

설명

식별된 치수 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 치수가 하나 이상의 집계를 정의하면, 하나의 집계는 차원의 빈 목록을 가져야 합니다.

사용자 조치

하나의 빈 치수 목록을 갖도록 식별된 치수를 변경하거나 집계를 정의하지 않도록 식별된 치수를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6350: dimension_name 차원이 **measure_name** 치수에서 여러 번 참조되었습니다.:

설명

식별된 치수 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 치수는 경우 집계 내에서는나 집계 사이에 두 번 이상 차원을 참조할 수 없습니다.

사용자 조치

식별된 차원을 한 번만 참조하도록 식별된 치수를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6351: object_name 오브젝트의 **SQL 표현식 템플릿**에서 숫자 **number**에 대한 토큰 표시기가 누락되었습니다. 토큰 표시기는 숫자 **1**로 시작하여 연속으로 번호를 매겨야 합니다.:

설명

식별된 치수 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 치수의 SQL 표현식 템플릿에서 토큰 표시기는 1로 시작하며 연속적으로 번호를 부여해야 합니다.

사용자 조치

SQL 표현식 템플릿에 대한 토큰 표시기에 1부터 연속으로 번호가 지정되도록 식별된 치수를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6352: measure_name 치수에 제공된 참조 **reference**를 사용하지 않는 **SQL 표현식 템플릿**가 있습니다.:

설명

식별된 치수 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 치수의 SQL 표현식 템플릿은 제공되는 모든 컬럼, 속성 및 치수 참조를 사용해야 합니다. 각 참조는 두 번 이상 사용할 수 있습니다.

사용자 조치

제공된 모든 컬럼, 속성 및 치수 참조를 사용하도록 식별된 치수의 SQL 표현식 템플릿을 변경해야 합니다. 또는 식별된 치수의 SQL 표현식 템플릿이 사용하지 않는 컬

럼, 속성 및 치수 참조를 제거하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

6353: attribute_name 속성에 대한 SQL 표현식 템플릿 중 하나가 속성이나 컬럼이 아닌 매개변수를 사용합니다.:

설명

식별된 속성 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 속성 오브젝트의 SQL 표현식 템플릿은 속성 또는 컬럼인 매개변수를 사용해야 합니다.

사용자 조치

SQL 표현식 템플릿이 속성 또는 컬럼을 매개변수로 사용하도록 식별된 속성을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

6354: SQL 표현식 템플릿에서 attribute_name 속성에 대한 매개변수로 사용된 속성 사이에 종속성 루프가 존재합니다.:

설명

식별된 속성 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 속성의 SQL 표현식 템플릿에 대한 매개변수로 사용되는 속성은 종속성 루프를 형성하지 않아야 합니다.

사용자 조치

SQL 표현식 템플릿이 매개변수에 관한 종속성 루프를 포함하지 않도록 식별된 속성을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

6355: attribute_name 속성이 SQL 표현식 템플릿 중 하나에 빈 문자열을 정의했습니다.:

설명

식별된 속성 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 속성의 SQL 표현식 템플릿은 빈 문자열일 수 없습니다.

사용자 조치

SQL 표현식 템플릿이 빈 문자열이 아니도록 식별된 속성을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

6356: attribute_name 속성의 SQL 표현식 템플릿에 집계 함수가 포함되어 있습니다.:

설명

식별된 속성 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 속성의 SQL 표현식 템플리트는 집계 포함할 수 없습니다.

사용자 조치

SQL 표현식 템플리트에 집계 함수가 포함되지 않도록 식별된 속성을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6358: attribute_name 속성에 제공된 참조 reference를 사용하지 않는 SQL 표현식 템플리트가 있습니다.:

설명

식별된 속성 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 속성의 SQL 표현식 템플리트는 제공되는 모든 컬럼 및 속성 참조를 사용해야 합니다. 각 참조는 두 번 이상 사용할 수 있습니다.

사용자 조치

제공된 모든 컬럼 및 속성 참조를 사용하도록 식별된 속성의 SQL 표현식 템플리트를 변경하십시오. 또는 식별된 속성의 SQL 표현식 템플리트가 사용하지 않는 컬럼 및 속성 참조를 제거하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6359: relationship_name 속성 관계가 두 개의 서로 다른 속성을 참조하지 않습니다.:

설명

식별된 속성 관계 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 속성 관계 오브젝트는 두 구별 속성을 참조해야 합니다.

사용자 조치

두 개의 고유 속성을 참조하도록 식별된 속성 관계 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6360: relationship_name 속성 관계가 잘못 정의되어 있습니다. 카디널리티 등록 정보가 N:N으로 설정되어 있지만 함수 종속성 등록 정보는 YES로 설정되어 있습니다.:

설명

식별된 속성 관계 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 속성 관계의 함수 종속성 등록 정보가 YES로 설정된 경우 속성 관계의 카디널리티 등록 정보를 N:N으로 설정할 수 없습니다.

사용자 조치

카디널리티가 N:N으로 설정되지 않거나 함수 종속성 등록 정보가 NO로 설정되도록 식별된 속성 관계를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6361: join_name 조인이 하나 이상의 3요소를 참조하지 않습니다. 3요소에는 왼쪽 속성, 오른쪽 속성 및 연산자를 포함합니다.:

설명

식별된 조인 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 조인 오브젝트는 왼쪽 속성, 오른쪽 속성 및 연산자를 포함하는 최소한 하나의 3요소를 참조해야 합니다.

사용자 조치

하나 이상의 3요소를 참조하도록 식별된 조인 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6362: join_name 조인에 있는 왼쪽 속성이 모두 단일 테이블의 컬럼으로 해석되지 않습니다.:

설명

식별된 조인 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 조인의 왼쪽 속성은 모두 단일 데이터베이스 테이블의 컬럼으로 해석되어야 합니다.

사용자 조치

왼쪽 속성이 모두 단일 테이블의 컬럼으로 해석되도록 식별된 조인 오브젝트를 변경하십시오. 또는 모두 메타데이터 규칙을 준수하도록 식별된 조인 오브젝트의 왼쪽 속성을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6363: join_name 조인에 있는 오른쪽 속성이 모두 단일 테이블의 컬럼으로 해석되지 않습니다.:

설명

식별된 조인 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 조인의 오른쪽 속성은 모두 단일 데이터베이스 테이블의 컬럼으로 해석되어야 합니다.

사용자 조치

오른쪽 속성이 모두 단일 테이블의 컬럼으로 해석되도록 식별된 조인 오브젝트를 변경하십시오. 또는 모두 메타데이터 규칙을 준수하도록 식별된 조인 오브젝트의 오른쪽 속성을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6364: join_name 조인에 대한 3요소 중 하나 이상이 유효한 작업을 정의하지 않습니다. 왼쪽 및 오른쪽 속성의 데이터 유형이 상호 간 또는 연산자와 호환되지 않을 수 있습니다.:

설명

식별된 조인 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 조인 오브젝트의 각 3요소는 유효한 조작을 정의해야 합니다. 오른쪽 및 왼쪽 속성의 데이터 유형은 지정된 조작을 수행할 수 있도록 상호 호환되어야 합니다.

사용자 조치

각 3요소가 유효한 조작을 정의하도록 식별된 조인 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6365: model_name 큐브 모델이 하나의 사실 오브젝트만을 참조하지 않습니다.:

설명

식별된 큐브 모델 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 전체 큐브 모델은 하나의 사실 오브젝트를 참조해야 합니다.

사용자 조치

하나의 사실 오브젝트를 참조하도록 식별된 큐브 모델 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6366: model_name 큐브 모델이 하나 이상의 차원을 참조하지 않습니다.:

설명

식별된 큐브 모델 오브젝트가 메타데이터 오브젝트 규칙을 위반했습니다. 전체 큐브 모델은 최소한 하나의 차원 오브젝트를 참조해야 합니다.

사용자 조치

하나 이상의 차원 오브젝트를 참조하도록 식별된 큐브 모델을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6367: join_name 사실-차원의 카디널리티가 1:1이나 N:1로 설정되지 않았습니다.:

설명

사실-차원 조인의 카디널리티가 1:1 또는 N:1이 아니기 때문에 큐브 모델이 최적화 어드바이저 권장사항을 활용하지 못합니다. 최적화가 수행되지 않습니다.

사용자 조치

큐브 모델이 최적화 어드바이저 권장사항을 활용하도록 하려면 사실에서 차원 오브젝트에 이르는 각 조인의 카디널리티가 1:1 또는 N:1로 설정되어야 합니다. 사실 속성에 대한 조인의 카디널리티가 1이나 N으로 설정되어야 하며 차원 속성의 카디널리티는 1이어야 합니다. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조를 참조하십시오.

6368: join_name 사실-차원 조인이 facts_name 사실 오브젝트의 테이블을 dimension_name 차원의 기본 테이블로 조인하지 않습니다.:

설명

식별된 조인 오브젝트가 최적화 규칙을 위반했습니다. 차원의 조인에서 구성된 조인 네트워크 때문에 이 테이블에서 전송되는 모든 조인의 카디널리티가 N:1 또는 1:1인 하나 이상의 테이블(기본 테이블)이 있어야 합니다. 큐브 모델의 경우 사실과 차원 오브젝트의 조인에는 이 차원의 기본 테이블이 있어야 합니다.

사용자 조치

큐브 모델 오브젝트에서는 모든 사실-차원 조인이 사실 오브젝트에서 각 차원의 기본 테이블까지인지 확인하십시오. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6369: 차원의 조인에서 구성된 조인 네트워크에서 표시한 대로 dimension_name 차원에 기본 테이블이 없습니다.:

설명

식별된 차원 오브젝트가 최적화 규칙을 위반했습니다. 차원의 조인에서 구성된 조인 네트워크 때문에 이 테이블에서 전송되는 모든 조인의 카디널리티가 N:1 또는 1:1인 하나 이상의 테이블이 있어야 합니다. 차원의 이러한 기본 테이블이 없는 경우 최적화가 수행되지 않습니다.

사용자 조치

차원에서 사용된 조인 오브젝트의 카디널리티를 점검하십시오. 최적화가 수행되도록 하려면 최적화 규칙에서 설명한 대로 차원에 기본 테이블이 있어야 합니다. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6370: join_name 조인에 참조 제한조건이 정의되지 않은 컬럼이 들어 있습니다.:

설명

식별된 조인 오브젝트가 최적화 규칙을 위반했습니다. 조인에 참여하는 컬럼에 대한 제한조건이 정의되어야 합니다. 조인이 자체 조인 즉, 등호의 양쪽에 사용되는 동일한 컬럼 세트인 경우, 컬럼 세트와 일치하는 기본 키를 정의해야 합니다. 다른 모든 경우에는 한 쪽의 컬럼 세트가 조인의 다른 쪽과 다른 경우 기본 키가 조인의 다른 쪽 컬럼과 일치해야 하고, 외부 키는 컬럼의 다른 세트와 일치하고 기본 키를 참조해야 합니다. 누락된 제한조건 때문에 최적화가 수행되지 않습니다.

사용자 조치

조인에 참여하는 컬럼에 대한 제한조건을 작성하십시오. 성능을 고려하여 표준 제한사항을 원하지 않는 경우 쿼리 최적화를 사용 가능하게 하여 정보용 제한조건을 작성하십시오. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6371: join_name 자체 조인에 포함된 컬럼을 사용하여 기본 키가 정의되어 있지 않습니다.:

설명

식별된 조인 오브젝트가 최적화 규칙을 위반했습니다. 조인에 참여하는 컬럼에 대한 제한조건을 정의해야 합니다. 조인이 자체 조인, 즉 등호의 양쪽에 사용되는 동일한 컬럼 세트인 경우, 컬럼 세트와 일치하는 기본 키를 정의해야 합니다. 누락된 제한조건 때문에 최적화가 수행되지 않습니다.

사용자 조치

테이블에서 기본 키를 정의한 경우 자체 조인의 속성을 테이블의 기본 키 컬럼을 표시하는 속성으로 설정하십시오. 아니면 자체 조인에 참여하는 컬럼에서 기본 키를 작성하십시오. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6372: join_name 조인의 한 쪽 컬럼을 사용하여 기본 키가 정의되어 있지 않습니다.:

설명

식별된 조인 오브젝트가 최적화 규칙을 위반했습니다. 조인에 참여하는 컬럼에 대한 제한조건을 정의해야 합니다. 한 쪽 컬럼 세트가 조인의 다른 쪽과 다른 경우 기본 키가 조인의 다른 쪽 컬럼과 일치해야 하고, 외부 키는 컬럼의 다른 세트와 일치하고 기본 키를 참조해야 합니다. 누락된 제한조건 때문에 최적화가 수행되지 않습니다.

사용자 조치

조인의 한 쪽 컬럼에서 기본 키를 작성하십시오. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6373: join_name 조인의 한 쪽 컬럼을 사용하여 외부 키가 정의되어 있지 않습니다.:

설명

식별된 조인 오브젝트가 최적화 규칙을 위반했습니다. 조인에 참여하는 컬럼에 대한 제한조건을 정의해야 합니다. 한 쪽 컬럼 세트가 조인의 다른 쪽과 다른 경우 기본 키가 조인의 다른 쪽 컬럼과 일치해야 하고, 외부 키는 컬럼의 다른 세트와 일치하고 기본 키를 참조해야 합니다. 누락된 제한조건 때문에 최적화가 수행되지 않습니다.

사용자 조치

조인의 기본 키 컬럼과 조인의 다른 쪽 컬럼 사이의 외부 키 제한조건을 작성하십시오. 성능을 고려하여 표준 제한사항을 원하지 않는 경우 쿼리 최적화를 사용 가능하게 하여 정보용 제한조건을 작성하십시오. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6374: join_name 조인의 한 쪽 컬럼을 사용하는 외부 키가 조인의 다른 쪽 컬럼을 사용하는 기본 키를 참조하지 않습니다.:

설명

식별된 조인 오브젝트가 최적화 규칙을 위반했습니다. 조인에 참여하는 컬럼에 대한 제한조건을 정의해야 합니다. 한 쪽 컬럼 세트가 조인의 다른 쪽과 다른 경우 기본 키가 조인의 한 쪽 컬럼과 일치해야 하고, 외부 키는 컬럼의 다른 세트와 일치하고 기본 키를 참조해야 합니다. 누락된 제한조건 때문에 최적화가 수행되지 않습니다.

사용자 조치

조인의 기본 키 컬럼과 조인의 다른 쪽 컬럼 사이의 외부 키 제한조건을 작성하십시오. 성능을 고려하여 인해 표준 제한조건을 원하지 않는 경우, 쿼리 최적화가 사용 가능한 정보용 제한조건을 작성하십시오. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6375: join_name 조인의 카디널리티가 1:1, N:1 또는 1:N으로 설정되지 않았습니다.:

설명

식별된 조인 오브젝트가 최적화 규칙을 위반했습니다. 조인 카디널리티가 M:N인 경우 최적화를 수행할 수 없습니다.

사용자 조치

조인이 기반으로 하는 제한조건에 따라 조인 카디널리티를 1:1 1:N 또는 N:1로 설정하십시오. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6376: join_name 자체 조인의 카디널리티가 1:1로 설정되지 않았습니다.:

설명

식별된 조인 오브젝트가 최적화 규칙을 위반했습니다. 자체 조인의 카디널리티가 1:1로 설정되지 않는 경우 최적화를 수행할 수 없습니다.

사용자 조치

자체 조인의 카디널리티를 1:1로 설정하십시오. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6377: join_name 조인의 카디널리티가 기본 키가 정의되어 있는 쪽에 대해 1로 설정되지 않았습니다.:

설명

식별된 조인 오브젝트가 최적화 규칙을 위반했습니다. 기본 키가 정의되어 있는 쪽에는 조인 카디널리티가 1이 되어야 하고, 외부 키가 정의되어 있는 쪽에는 N이 되어야 합니다. 외부 키 쪽에도 기본 키가 정의되어 있으면, 카디널리티로 1을 사용해야 합니다. 이에 해당하지 않으면 최적화를 수행할 수 없습니다.

사용자 조치

기본 키가 정의되어 있는 쪽의 경우 조인 카디널리티가 1로 설정되어야 합니다. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6378: join_name 조인의 카디널리티가 외부 키가 정의되어 있는 쪽에 대해 N으로 설정되지 않았습니다.:

설명

식별된 조인 오브젝트가 최적화 규칙을 위반했습니다. 기본 키가 정의되어 있는 쪽에는 조인 카디널리티가 1이 되어야 하고, 외부 키가 정의되어 있는 쪽에는 N이 되어야 합니다. 외부 키 쪽에도 기본 키가 정의되어 있으면, 카디널리티로 1을 사용해야 합니다. 이에 해당하지 않으면 최적화를 수행할 수 없습니다.

사용자 조치

외부 키가 정의되어 있는 쪽의 경우 조인 카디널리티가 N으로 설정되어야 합니다. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6379: join_name 조인의 카디널리티가 기본 키 및 외부 키가 모두 정의되어 있는 쪽에 대해 1로 설정되지 않았습니다.:

설명

식별된 조인 오브젝트가 최적화 규칙을 위반했습니다. 기본 키가 정의되어 있는 쪽에는 조인 카디널리티가 1이 되어야 하고, 기본 키 및 외부 키가 모두 정의되어 있는 쪽에도 1이 되어야 합니다. 이에 해당하지 않으면 최적화를 수행할 수 없습니다.

사용자 조치

조인 카디널리티는 1:1로 설정되어야 합니다. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6380: join_name 조인에서 참조하는 **attribute_name** 속성이 널(null)이 아닌 SQL 표현식으로 해석되지 않습니다.:

설명

식별된 조인 오브젝트가 최적화 규칙을 위반했습니다. 조인에서 사용되는 모든 속성은 널(null)이 아닌 SQL 표현식으로 해석되어야 합니다. 널(null)이 아닌 SQL 표현식으로 해석되는 속성을 조인에서 참조하는 경우 최적화를 수행할 수 없습니다.

사용자 조치

널(null)이 아닌 속성에 대한 참조를 조인에서 제거하십시오. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6381: join_name 조인에 **INNER JOIN** 유형이 없습니다.:

설명

식별된 조인 오브젝트가 최적화 규칙을 위반했습니다. 조인 유형은 내부 조이어야 합니다. 이에 해당하지 않으면 최적화를 수행할 수 없습니다.

사용자 조치

단일 컬럼으로 해석되는 속성만 참조하도록 조인을 변경하십시오. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6382: join_name 조인의 **attribute_name** 속성 참조가 제한 조건에 포함되기 위한 요구사항인 단일 계산 결과 컬럼으로 해석되지 않습니다.:

설명

식별된 조인 오브젝트가 최적화 규칙을 위반했습니다. DB2 제한조건은 조인에서 참조하는 속성에 적용되어야 합니다. 제한조건은 컬럼에만 적용될 수 있고 따라서, 조인에서 참조하는 속성은 테이블의 단일 컬럼으로 해결되어야 합니다. 이에 해당하지 않으면 최적화를 수행할 수 없습니다.

사용자 조치

단일 컬럼으로 해석되는 속성만 참조하도록 조인을 변경하십시오. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6385: model_name 큐브 모델은 계층 구조를 가진 하나 이상의 차원을 참조해야 합니다.:

설명

식별된 큐브 모델이 계층 구조가 있는 하나 이상의 차원을 참조해야 한다는 최적화 규칙을 위반합니다.

사용자 조치

차원이 하나 이상의 계층 구조를 참조하도록 식별된 큐브 모델의 차원을 변경하십시오. 최적화 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6386: 각 최적화 슬라이스에는 cube_name 큐브의 큐브 차원마다 정확히 하나의 최적화 레벨을 정의해야 합니다.:

설명

큐브가 최적화 슬라이스에 큐브의 큐브 차원 당 정의된 최적화 레벨이 하나만 있어야 한다는 규칙을 위반합니다.

사용자 조치

큐브 오브젝트에서 큐브 차원 당 하나의 최적화 레벨만 참조하도록 최적화 슬라이스를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6387: 최적화 레벨은 cube_name 큐브의 한 개 큐브 차원을 참조해야 합니다.:

설명

이 큐브는 최적화 레벨이 큐브에 속하는 큐브 차원을 하나만 참조해야 한다는 메타데이터 오브젝트 규칙을 만족시키지 않습니다.

사용자 조치

큐브 오브젝트에서 하나의 큐브 차원을 참조하도록 최적화 레벨을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6388: 최적화 레벨은 cube_name 큐브의 한 개 큐브 계층 구조를 참조해야 합니다.:

설명

이 큐브는 최적화 레벨이 큐브에 속하는 큐브 계층 구조를 하나만 참조해야 한다는 메타데이터 오브젝트 규칙을 만족시키지 않습니다.

사용자 조치

큐브 오브젝트에서 하나의 큐브 차원 및 하나의 큐브 계층 구조를 참조하도록 최적화 레벨을 변경하십시오.

6389: cube_name 큐브의 최적화 레벨을 allLevel, anyLevel 또는 큐브 레벨 참조로 설정해야 합니다.:

설명

이 큐브는 최적화 레벨에 allLevel, anyLevel 또는 큐브 레벨 참조가 있어야 한다는 메타데이터 오브젝트 규칙을 만족시키지 않습니다.

사용자 조치

큐브 오브젝트에서 allLevel, anyLevel 또는 큐브 레벨 참조를 참조하도록 식별된 최적화 레벨을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6390: 최적화 레벨에서 참조된 큐브 차원 및 큐브 계층 구조는 cube_name 큐브의 오브젝트에서 파생되어야 합니다. 큐브 계층 구조가 큐브 차원에 속해야 합니다.:

설명

이 큐브는 참조되는 큐브 차원 및 큐브 계층 구조가 해당 최적화 레벨 큐브의 오브젝트에서 파생되어야 한다는 메타데이터 오브젝트 규칙을 만족시키지 않습니다. 큐브 계층 구조가 큐브 차원에 속해야 합니다.

사용자 조치

참조되는 큐브 차원 및 큐브 계층 구조가 해당 큐브의 오브젝트에서 파생되도록 최적화 레벨을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6391: 최적화 레벨에서 큐브 레벨 참조가 anyLevel 또는 allLevel이 아니면 level_name 큐브 레벨은 cube_name 큐브의 hierarchy_name 큐브 계층 구조에 속해야 합니다.:

설명

이 큐브는 최적화 레벨에서 큐브 레벨 참조가 anyLevel 또는 allLevel이 아닐 경우 큐브 레벨이 해당 큐브 계층 구조에 속해야 한다는 메타데이터 오브젝트 규칙을 만족시키지 않습니다.

사용자 조치

참조되는 큐브 레벨이 해당 큐브 계층 구조에 속하도록 최적화 레벨을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6392: cube_name 큐브에 MOLAP 추출 최적화 슬라이스와 하이브리드 추출 최적화 슬라이스가 둘 다 존재할 수 없습니다.:

설명

이 큐브는 큐브에 MOLAP 추출 최적화 슬라이스와 하이브리드 추출 최적화 슬라이스가 동시에 포함될 수 없다는 메타데이터 오브젝트 규칙을 만족시키지 않습니다.

사용자 조치

큐브에 MOLAP 추출 최적화 슬라이스와 하이브리드 추출 최적화 슬라이스가 동시에 포함되지 않도록 최적화 슬라이스 중 하나를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6393: cube_name 큐브에 둘 이상의 MOLAP 추출 최적화 슬라이스가 존재할 수 없습니다.:

설명

이 큐브는 큐브에 최대 하나의 MOLAP 추출 유형 최적화 슬라이스가 있어야 한다는 메타데이터 오브젝트 규칙을 만족시키지 않습니다.

사용자 조치

0 또는 1개의 MOLAP 추출 유형 최적화 슬라이스가 포함되도록 큐브를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6394: cube_name 큐브에 둘 이상의 하이브리드 추출 최적화 슬라이스가 존재할 수 없습니다.:

설명

이 큐브는 큐브에 최대 하나의 하이브리드 추출 유형 최적화 슬라이스가 있어야 한다는 메타데이터 오브젝트 규칙을 만족시키지 않습니다.

사용자 조치

0 또는 1개의 하이브리드 추출 유형 최적화 슬라이스가 포함되도록 큐브를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6395: cube_name 큐브에 하이브리드 추출 최적화 슬라이스가 있는 경우에만 드릴 쓰루 최적화 슬라이스가 존재할 수 있습니다.:

설명

이 큐브는 큐브에 하이브리드 추출 최적화 슬라이스가 있는 경우에만 드릴 쓰루 최적화 슬라이스가 존재할 수 있다는 메타데이터 오브젝트 규칙을 만족시키지 않습니다.

사용자 조치

큐브에 하이브리드 추출 최적화 슬라이스가 없는 경우 최적화 슬라이스 유형을 드릴 쓰루에서 다른 유형으로 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6396: dimension_name 차원의 **level_name** 레벨이 참조하는 **attribute_name** 속성은 차원 속성 목록에 포함되어야 합니다.:

설명

이 차원은 차원의 레벨에서 참조하는 모든 속성이 해당 차원의 속성 목록에 포함되어야 한다는 메타데이터 오브젝트 규칙을 만족시키지 않습니다.

사용자 조치

식별된 속성을 참조하지 않도록 레벨을 변경하거나 식별된 속성을 차원에 추가하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6397: level_name 레벨은 최소한 하나의 레벨 키 속성을 참조해야 합니다.:

설명

이 큐브는 레벨이 최소 하나의 레벨 키 속성을 참조해야 한다는 메타데이터 오브젝트 규칙을 만족시키지 않습니다.

사용자 조치

최소 하나의 레벨 키 속성을 참조하도록 레벨을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6398: level_name 레벨은 레벨 키 속성 세트에 **attribute_name** 속성을 두 번 이상 포함할 수 없습니다.:

설명

이 레벨은 레벨 키 속성 세트가 중복 속성을 포함할 수 없다는 메타데이터 오브젝트 규칙을 만족시키지 않습니다.

사용자 조치

중복 레벨 키 속성을 포함하지 않도록 레벨을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6399: level_name 레벨은 최소한 하나의 디폴트 속성을 참조해야 합니다.:

설명

이 레벨은 레벨에 하나의 디폴트 속성만 있어야 한다는 메타데이터 오브젝트 규칙을 만족시키지 않습니다.

사용자 조치

하나의 디폴트 속성만 포함하도록 레벨을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6400-6499

6401: level_name 레벨에서 한 속성을 관련 속성과 디폴트 속성으로 동시에 사용할 수 없습니다.:

설명

이 레벨은 한 속성을 디폴트 속성과 관련 속성으로 동시에 사용할 수 없다는 메타데이터 오브젝트 규칙을 준수하지 않습니다.

사용자 조치

디폴트 속성이 관련 속성으로 사용되지 않도록 레벨을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6402: level_name 레벨에서 한 속성을 관련 속성과 디폴트 속성으로 동시에 사용할 수 없습니다.:

설명

이 레벨은 관련 속성 세트에 중복 속성을 포함할 수 없다는 메타데이터 오브젝트 규칙을 준수하지 않습니다.

사용자 조치

중복 관련 속성이 포함되지 않도록 레벨을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6403: level_name 레벨은 관련 속성 세트에 attribute_name 속성을 한 번 이상 포함할 수 없습니다.:

설명

이 큐브 레벨은 큐브 레벨이 한 레벨만 참조해야 한다는 메타데이터 오브젝트 규칙을 준수하지 않습니다.

사용자 조치

한 레벨만 참조하도록 큐브를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6404: level_name 큐브 레벨은 정확히 하나의 레벨만 참조해야 합니다.:

설명

이 큐브 레벨은 큐브 레벨의 모든 관련 속성이 해당 레벨의 관련 속성이기도 해야 한다는 메타데이터 오브젝트 규칙을 준수하지 않습니다.

사용자 조치

상위 레벨에서도 참조하는 속성을 참조하도록 큐브 레벨을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6405: level_name 큐브 레벨은 관련 속성 세트에 attribute_name 속성을 한 번 이상 포함할 수 없습니다.:

설명

이 큐브 레벨은 관련 속성 세트에 중복 속성을 포함할 수 없다는 메타데이터 오브젝트 규칙을 준수하지 않습니다.

사용자 조치

중복 관련 속성이 포함되지 않도록 큐브 레벨을 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

참조 제한조건 관련 오류

6500: attribute_name 속성 또는 치수에 대한 SQL 템플릿이 다른 속성, 치수 또는 컬럼을 계속 참조하기 때문에 이 조사를 수행할 수 없습니다. 이 조사를 실행하기 전에 이러한 참조를 삭제해야 합니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조사를 수행할 수 없습니다. 식별된 오브젝트에 대한 SQL 표현식 템플릿에 다른 속성, 치수 또는 컬럼에 대한 참조가 포함되어 있습니다. 이러한 속성, 치수 또는 컬럼은 이 조사를 실행하기 전에 식별된 오브젝트에서 제거해야 합니다.

사용자 조치

식별된 오브젝트를 삭제하기 전에 SQL 표현식 템플릿에서 속성, 치수 또는 컬럼을 더 이상 참조하지 못하도록 식별된 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6501: 다른 속성 또는 치수에서 attribute_name 속성 또는 치수를 참조하기 때문에 이 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 다른 속성 또는 치수에서 식별된 속성 또는 치수를 참조합니다. 따라서 식별된 속성 또는 치수를 삭제할 수 없습니다.

사용자 조치

식별된 속성 또는 치수를 삭제하기 전에 식별된 속성 또는 치수를 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6502: 치수로 정의되어 있는 집계에서 dimension_name 차원을 참조하기 때문에 이 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 치수의 집계에서 식별된 차원을 참조합니다. 따라서 식별된 차원을 삭제할 수 없습니다.

사용자 조치

식별된 차원을 삭제하기 전에 식별된 차원을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6503: object_name 오브젝트 작업을 수행할 수 없습니다. 큐브 계층 구조는 해당 큐브 계층 구조가 파생되는 데 사용된 계층 구조가 이미 참조하고 있는 속성을 참조해야 합니다 .:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 다음 중 한 가지 상황으로 인해 문제가 발생했습니다.

- 계층 구조에서 속성을 제거하려는 시도를 했지만 관련 큐브 계층 구조에서 제거 중인 속성을 아직 사용하고 있습니다.

- 큐브 계층 구조로 속성을 추가하려는 시도를 했지만 관련 계층 구조에서 추가 중인 속성을 아직 사용하지 않습니다.

사용자 조치

다음 조치 중 하나를 수행하십시오.

- 관련 계층 구조에서 동일한 속성을 제거하기 전에 큐브 계층 구조에서 속성을 제거하십시오.
- 관련 큐브 계층 구조에 동일한 속성을 추가하기 전에 계층 구조에 속성을 추가하십시오.

메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6504: object_name 오브젝트 조작을 수행할 수 없습니다. 큐브 계층 구조는 해당 큐브 계층 구조가 파생되는 데 사용된 계층 구조가 이미 참조한 속성 관계를 참조해야 합니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 다음 중 한 가지 상황으로 인해 문제가 발생했습니다.

- 계층 구조에서 속성 관계를 제거하려는 시도를 했지만 관련 큐브 계층 구조에서 제거 중인 속성 관계를 아직 사용하고 있습니다.
- 큐브 계층 구조에 속성 관계를 추가하려는 시도를 했지만 관련 계층 구조에서 추가 중인 속성 관계를 아직 사용하지 않습니다.

사용자 조치

다음 조치 중 하나를 수행하십시오.

- 관련 계층 구조에서 동일한 속성 관계를 제거하기 전에 큐브 계층 구조에서 속성 관계를 제거하십시오.
- 관련 큐브 계층 구조에 동일한 속성 관계를 추가하기 전에 계층 구조에 속성 관계를 추가하십시오.

메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6505: 큐브 계층 구조에서 hierarchy_name 계층 구조를 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 큐브 계층 구조에서 식별된 계층 구조를 참조합니다. 따라서 식별된 계층 구조를 삭제할 수 없습니다.

사용자 조치

식별된 계층 구조를 삭제하기 전에 식별된 계층 구조를 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6506: object_name 오브젝트에 대한 작업을 수행할 수 없습니다. 큐브 사실은 해당 큐브 사실이 파생되는 데 사용된 사실이 이미 참조한 치수를 참조해야 합니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 다음 중 한 가지 상황으로 인해 문제가 발생했습니다.

- 사실에서 치수를 제거하려는 시도를 했지만 관련 큐브 사실에서 제거 중인 치수를 이미 사용하고 있습니다.
- 큐브 사실에서 치수를 추가하려는 시도를 했지만 관련 사실에서 추가 중인 치수를 아직 사용하지 않습니다.

사용자 조치

다음 조치 중 하나를 수행하십시오.

- 관련 사실에서 동일한 치수를 제거하기 전에 큐브 사실에서 치수를 제거하십시오.
- 관련 큐브 사실에 동일한 치수를 추가하기 전에 사실에 치수를 추가하십시오.

메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6507: 큐브 사실 오브젝트에서 facts_name 사실을 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 큐브 사실에서 식별된 사실을 참조합니다. 따라서 식별된 사실을 삭제할 수 없습니다.

사용자 조치

식별된 사실을 삭제하기 전에 식별된 사실을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6508: 차원에서 hierarchy_name 계층 구조를 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 차원에서 식별된 계층 구조를 참조합니다. 따라서 식별된 계층 구조를 삭제할 수 없습니다.

사용자 조치

식별된 계층 구조를 삭제하기 전에 식별된 계층 구조를 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6509: 사실 오브젝트에서 join_name 조인을 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 사실에서 식별된 조인을 참조합니다. 따라서 식별된 조인을 삭제할 수 없습니다.

사용자 조치

식별된 조인을 삭제하기 전에 식별된 조인을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6510: 큐브에서 cube_dimension_name 큐브 차원을 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 큐브에서 식별된 큐브 차원을 참조합니다. 따라서 식별된 큐브 차원을 삭제할 수 없습니다.

사용자 조치

식별된 큐브 차원을 삭제하기 전에 식별된 큐브 차원을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

6511: object_name 오브젝트 작업을 수행할 수 없습니다. 큐브의 큐브 차원은 해당 큐브가 파생된 큐브 모델에서 참조하는 차원에서 파생되어야 합니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 다음 중 한 가지 상황으로 인해 문제가 발생했습니다.

- 큐브 모델에서 차원을 제거하려는 시도를 했지만 관련 큐브의 큐브 차원에서 제거 중인 속성을 아직 사용하고 있습니다.
- 큐브 차원에 치수를 추가하려는 시도를 했지만 관련 큐브 모델에서 추가 중인 큐브 치수에 대한 차원을 아직 사용하지 않습니다.

사용자 조치

다음 조치 중 하나를 수행하십시오.

- 관련 큐브 모델에서 관련 치수를 제거하기 전에 큐브에서 큐브 차원을 제거하십시오.
- 관련 큐브에 관련 큐브 차원을 추가하기 전에 큐브 모델에 차원을 추가하십시오.

메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

6512: 큐브 차원에서 dimension_name을 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 큐브 차원에서 식별된 차원을 참조합니다. 따라서 식별된 차원을 삭제할 수 없습니다.

사용자 조치

식별된 차원을 삭제하기 전에 식별된 차원을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내 및 참조서*를 참조하십시오.

6513: object_name 오브젝트 작업을 수행할 수 없습니다. 큐브 차원의 큐브 계층 구조는 이 큐브 차원이 파생되는 데 사용된 것과 동일한 차원에서 참조한 계층 구조에서 파생되어야 합니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 다음 중 한 가지 상황으로 인해 문제가 발생했습니다.

- 차원에서 계층 구조를 제거하려는 시도를 했지만 관련 큐브 차원의 큐브 계층 구조에서 제거 중인 계층 구조를 아직 사용하고 있습니다.
- 큐브 차원에 큐브 계층 구조를 추가하려는 시도를 했지만 관련 차원에서 추가 중인 큐브 계층 구조에 대한 계층 구조를 아직 사용하지 않습니다.

사용자 조치

다음 조치 중 하나를 수행하십시오.

- 관련 차원에서 관련 계층 구조를 제거하기 전에 큐브 차원에서 큐브 계층 구조를 제거하십시오.
- 관련 큐브 차원에 큐브 계층 구조를 추가하기 전에 차원에 계층 구조를 추가하십시오.

메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6514: 큐브 차원에서 cube_hierarchy_name 큐브 계층 구조를 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 큐브 차원에서 식별된 큐브 계층 구조를 참조합니다. 따라서 식별된 큐브 계층 구조를 삭제할 수 없습니다.

사용자 조치

식별된 큐브 계층 구조를 삭제하기 전에 식별된 큐브 계층 구조를 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6515: object_name 오브젝트 작업을 수행할 수 없습니다. 큐브 차원의 큐브 계층 구조는 이 큐브 차원이 파생되는 데 사용된 것과 동일한 차원에서 참조한 계층 구조에서 파생되어야 합니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 다음 중 한 가지 상황으로 인해 문제가 발생했습니다.

- 차원에서 계층 구조를 제거하려는 시도를 했지만 관련 큐브 차원의 큐브 계층 구조에서 제거 중인 계층 구조를 아직 사용하고 있습니다.
- 큐브 차원에 큐브 계층 구조를 추가하려는 시도를 했지만 관련 차원에서 추가 중인 큐브 계층 구조에 대한 계층 구조를 아직 사용하지 않습니다.

사용자 조치

다음 조치 중 하나를 수행하십시오.

- 관련 차원에서 관련 계층 구조를 제거하기 전에 큐브 차원에서 큐브 계층 구조를 제거하십시오.
- 관련 큐브 차원에 큐브 계층 구조를 추가하기 전에 차원에 계층 구조를 추가하십시오.

메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6516: 차원에서 join_name 조인을 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 차원에서 식별된 조인을 참조합니다. 따라서 식별된 조인을 삭제할 수 없습니다.

사용자 조치

식별된 조인을 삭제하기 전에 식별된 조인을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6517: 차원에서 attribute_name 속성을 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 차원에서 식별된 속성을 참조합니다. 따라서 식별된 속성을 삭제할 수 없습니다.

사용자 조치

식별된 속성을 삭제하기 전에 식별된 속성을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6518: 계층 구조에서 attribute_name 속성을 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 계층 구조에서 식별된 속성을 참조합니다. 따라서 식별된 속성을 삭제할 수 없습니다.

사용자 조치

식별된 속성을 삭제하기 전에 식별된 속성을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6519: 계층 구조에서 relationship_name 속성 관계를 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 계층 구조에서 식별된 속성 관계를 참조합니다. 따라서 식별된 속성 관계를 삭제할 수 없습니다.

사용자 조치

식별된 속성 관계를 삭제하기 전에 식별된 속성 관계를 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6520: 큐브 모델에서 dimension_name 차원을 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 큐브 모델에서 식별된 차원을 참조합니다. 따라서 식별된 차원 관계를 삭제할 수 없습니다.

사용자 조치

식별된 차원을 삭제하기 전에 식별된 차원을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6521: 큐브 모델에서 join_name 조인을 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 큐브 모델에서 식별된 조인을 참조합니다. 따라서 식별된 조인을 삭제할 수 없습니다.

사용자 조치

식별된 조인을 삭제하기 전에 식별된 조인을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내* 및 참조서를 참조하십시오.

6522: 사실 오브젝트에서 object_name을 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 사실에서 식별된 오브젝트를 참조합니다. 따라서 식별된 오브젝트를 삭제할 수 없습니다.

사용자 조치

식별된 오브젝트를 삭제하기 전에 식별된 오브젝트를 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내* 및 참조서를 참조하십시오.

6523: 속성 관계에서 attribute_name 왼쪽 속성을 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 속성 관계에서 식별된 속성을 참조합니다. 따라서 식별된 속성을 삭제할 수 없습니다.

사용자 조치

식별된 속성을 삭제하기 전에 식별된 속성을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내* 및 참조서를 참조하십시오.

6524: 속성 관계에서 attribute_name 오른쪽 속성을 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 속성 관계에서 식별된 속성을 참조합니다. 따라서 식별된 속성을 삭제할 수 없습니다.

사용자 조치

식별된 속성을 삭제하기 전에 식별된 속성을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내* 및 참조서를 참조하십시오.

6525: 조인에서 attribute_name 오른쪽 속성을 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 조인에서 식별된 속성을 참조합니다. 따라서 식별된 속성을 삭제할 수 없습니다.

사용자 조치

식별된 속성을 삭제하기 전에 식별된 속성을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내* 및 참조서를 참조하십시오.

6526: 조인에서 attribute_name 왼쪽 속성을 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 조인에서 식별된 속성을 참조합니다. 따라서 식별된 속성을 삭제할 수 없습니다.

사용자 조치

식별된 속성을 삭제하기 전에 식별된 속성을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views 안내* 및 참조서를 참조하십시오.

6527: 큐브에서 model_name 큐브 모델을 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 큐브에서 식별된 큐브 모델을 참조합니다. 따라서 식별된 큐브 모델을 삭제할 수 없습니다.

사용자 조치

식별된 큐브 모델을 삭제하기 전에 식별된 큐브 모델을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6528: 큐브에서 cube_facts_name 큐브 사실 오브젝트를 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 큐브에서 식별된 큐브 사실을 참조합니다. 따라서 식별된 큐브 사실을 삭제할 수 없습니다.

사용자 조치

식별된 큐브 사실을 삭제하기 전에 식별된 큐브 사실을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6529: 큐브 모델에서 facts_name 사실 오브젝트를 참조하기 때문에 작업을 수행할 수 없습니다.:

설명

메타데이터 카탈로그의 메타데이터 오브젝트 사이에 존재하는 참조 제한조건을 위반하기 때문에 요청한 조작을 수행할 수 없습니다. 현재 큐브 모델에서 식별된 사실을 참조합니다. 따라서 식별된 사실을 삭제할 수 없습니다.

사용자 조치

식별된 사실을 삭제하기 전에 식별된 사실을 더 이상 참조하지 못하도록 참조 오브젝트를 변경하십시오. 메타데이터 규칙 및 메타데이터 오브젝트 간의 참조 제한조건에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6530: level_name 레벨을 차원이 참조하므로 작업을 수행할 수 없습니다.:

설명

요청한 조작이 메타데이터 카탈로그의 메타데이터 오브젝트 사이에 있는 참조 제한조건을 위반하므로 이 조작을 수행할 수 없습니다. 레벨을 참조하고 있는 차원이 있으므로 지금 레벨을 삭제할 수 없습니다.

사용자 조치

레벨을 삭제하기 전에 더 이상 이 레벨을 참조하지 않도록 참조 중인 오브젝트를 변경하십시오. 메타데이터 규칙 및 메타데이터 오브젝트 간의 참조 제한조건에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6531: level_name 레벨을 계층 구조가 참조하므로 작업을 수행할 수 없습니다.:

설명

요청한 조작이 메타데이터 카탈로그의 메타데이터 오브젝트 사이에 있는 참조 제한조건을 위반하므로 이 조작을 수행할 수 없습니다. 레벨을 참조하고 있는 계층 구조가 있으므로 지금 레벨을 삭제할 수 없습니다.

사용자 조치

레벨을 삭제하기 전에 더 이상 이 레벨을 참조하지 않도록 참조 중인 오브젝트를 변경하십시오. 메타데이터 규칙 및 메타데이터 오브젝트 간의 참조 제한조건에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6532: level_name 큐브 레벨을 큐브 계층 구조가 참조하므로 작업을 수행할 수 없습니다.:

설명

요청한 조작이 메타데이터 카탈로그의 메타데이터 오브젝트 사이에 있는 참조 제한조건을 위반하므로 이 조작을 수행할 수 없습니다. 레벨을 참조하고 있는 큐브 계층 구조가 있으므로 지금 큐브 레벨을 삭제할 수 없습니다.

사용자 조치

레벨을 삭제하기 전에 더 이상 큐브 레벨을 참조하지 않도록 참조 중인 오브젝트를 변경하십시오. 메타데이터 규칙 및 메타데이터 오브젝트 간의 참조 제한조건에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6533: cube_name 큐브에 대해 작업을 수행할 수 없습니다. 큐브 차원과 큐브 계층 구조는 큐브에서 파생되어야 합니다. 큐브 계층 구조가 큐브 차원에 속해야 합니다.:

설명

요청한 조작이 메타데이터 카탈로그의 메타데이터 오브젝트 사이에 있는 참조 제한조건을 위반하므로 이 조작을 수행할 수 없습니다. 큐브의 큐브 차원을 현재 최적화 슬라이스의 큐브 차원 및 큐브 계층 구조가 참조하므로 지금 큐브 최적화 슬라이스를 삭제할 수 없습니다.

사용자 조치

큐브의 최적화 슬라이스를 삭제하기 전에 더 이상 큐브의 최적화 슬라이스를 참조하지 않도록 참조 중인 오브젝트를 변경하십시오. 메타데이터 규칙 및 메타데이터 오브젝트 간의 참조 제한조건에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

6534: level_name 레벨에서 작업을 수행할 수 없습니다. 해당 큐브 레벨이 관련 속성을 참조하므로 이 관련 속성을 레벨에서 제거할 수 없습니다.:

설명

요청한 조작이 메타데이터 카탈로그의 메타데이터 오브젝트 사이에 있는 참조 제한조건을 위반하므로 이 조작을 수행할 수 없습니다. 관련 속성을 해당 큐브 레벨에서 참조하므로 이 관련 속성을 레벨에서 삭제할 수 없습니다.

사용자 조치

관련 속성을 삭제하려면 큐브 레벨이 삭제할 레벨의 관련 속성을 참조하지 않도록 참조 중인 큐브 레벨 오브젝트를 수정하십시오. 메타데이터 규칙 및 메타데이터 오브젝트 간의 참조 제한조건에 대한 자세한 정보는 *DB2 Cube Views* 안내 및 참조서를 참조하십시오.

최적화

최적화 오류 코드

7001: model_name 큐브 모델에 대해 큐브가 정의되지 않았습니다.:

설명

큐브 모델에 정의된 큐브가 없습니다.

사용자 조치

큐브 모델의 데이터를 MOLAP 큐브로 읽어 들이는 추출 쿼리를 최적화하려면 MOLAP 큐브를 나타내는 큐브를 정의해야 합니다. 큐브를 한 개 이상 정의하지 않으면 추출 쿼리를 최적화할 수 없습니다.

7002: model_name 큐브 모델이 존재하지 않습니다.:

설명

지정한 이름을 가진 큐브 모델이 정의되지 않았습니다.

사용자 조치

올바른 큐브 모델 및 스키마 이름이 지정되었는지 확인하십시오. 이름 및 스키마에는 대 소문자가 구분됩니다. 기존 큐브 모델의 목록을 보려면 OLAP 센터를 사용하십시오.

7003: table_space_name 테이블 스페이스를 찾을 수 없습니다.:

설명

이 이름을 가진 테이블 스페이스가 정의되지 않았습니다.

사용자 조치

올바른 테이블 스페이스 이름이 지정되었는지 확인하십시오.

7004: 최적화 어드바이저가 권장사항을 판별할 수 없습니다.:

설명

이 큐브 모델을 최적화하는 데 사용할 수 있는 디스크 스페이스의 특정 한계를 지정했습니다. 어드바이저가 그 이하로 또는 동일하게 지정한 디스크 스페이스 크기를 사용하도록 하는 권장사항을 생성할 수 없습니다.

사용자 조치

디스크 스페이스 한계를 더 크게 지정하고 최적화 어드바이저 마법사를 다시 실행하십시오.

7005: table_space_name 테이블 스페이스는 요약 테이블에 저장하는 데 사용할 수 없습니다.:

설명

테이블 스페이스에 테이블 데이터를 저장하는 데 필요한 올바른 데이터 스토리지 유형이 없습니다. 테이블 스페이스의 유형은 일반이어야 합니다. LONG, 사용자 임시, 시스템 임시 테이블 스페이스는 요약 테이블을 저장하는 데 사용할 수 없습니다.

사용자 조치

일반 테이블 스페이스를 지정하여 요약 테이블을 저장하십시오.

7006: table_space_name 테이블 스페이스는 인덱스를 저장하는 데 사용할 수 없습니다.:

설명

인덱스 데이터를 저장하는 데 필요한 올바른 데이터 스토리지 유형이 지정된 테이블 스페이스에 없습니다. 테이블 스페이스의 유형은 일반 또는 LONG 유형이어야 합니다. 사용자 임시 및 시스템 임시 테이블 스페이스는 인덱스를 저장하는 데 사용할 수 없습니다.

사용자 조치

일반 또는 LONG 테이블 스페이스를 지정하여 인덱스를 저장하십시오.

7007: model_name 큐브 모델의 최적화 유효성 확인에 실패했습니다.:

설명

큐브 모델 및 연관 메타데이터 오브젝트가 최적화에 필요한 메타데이터 오브젝트 규칙을 한 가지 이상 위반했습니다. 최적화가 수행되지 않습니다. performed.

사용자 조치

큐브 모델 및 연관 메타데이터 오브젝트가 최적화에 대한 메타데이터 오브젝트 규칙을 위반하면 최적화를 수행할 수 없습니다. 최적화 규칙에 대한 자세한 정보는 *설정 및 사용자 안내서*를 참조하십시오.

7008: 큐브 모델에 계층 구조가 최적화될 수 있는 차원이 없습니다.:

설명

최적화 어드바이저가 계층 구조가 최적화될 수 있는 차원을 찾을 수 없으므로 최적화가 수행되지 않습니다.

사용자 조치

큐브 모델에 계층 구조가 반복되지 않는 차원이 하나 이상 있는지 확인하십시오.

7009: 둘 이상의 MOLAP 추출 또는 하이브리드 추출을 지정하거나 한 큐브에 MOLAP 추출 및 하이브리드 추출 유형의 최적화 슬라이스를 모두 지정할 수 없습니다. 둘 이상의 최적화 슬라이스 추출 유형이 cube_name 큐브에 지정되었습니다.:

설명

큐브에 둘 이상의 최적화 슬라이스 추출 유형이 지정되었으므로 지정된 큐브가 최적화되지 않습니다. 큐브마다 MOLAP 추출 또는 하이브리드 추출 유형의 최적화 슬라이스를 하나만 지정할 수 있습니다.

사용자 조치

큐브마다 MOLAP 추출 또는 하이브리드 추출 유형의 최적화 슬라이스가 하나뿐인지 확인하십시오.

7010: 하나 이상의 큐브 차원에서 하이브리드 추출 최적화 슬라이스 아래의 큐브 레벨로 Drillthrough 최적화 슬라이스를 정의해야 합니다.:

설명

Drillthrough 최적화 슬라이스는 하나 이상의 큐브 차원에서 하이브리드 추출 최적화 슬라이스 아래의 큐브 레벨로 정의해야 합니다.

사용자 조치

Drillthrough 최적화 슬라이스가 하나 이상의 큐브 차원에서 하이브리드 추출 최적화 슬라이스 아래의 큐브 레벨로 정의되어 있는지 확인하십시오.

7011: 최적화 어드바이저가 중지되어 할당된 시간 내에 권장사항을 판별할 수 없습니다.:

설명

권장사항을 판별하기 전에 최적화 어드바이저가 권고 중지 조작에 의해 중지되었습니다.

사용자 조치

최적화 어드바이저가 권장사항을 판별할 수 있도록 좀 더 오래 실행하도록 하십시오.

7012: 유효하지 않은 ID 값:

설명

지정된 ID 값이 유효하지 않습니다.

7013: 뷰, 별명, 별칭 또는 기타 샘플링을 지원하지 않는 데이터베이스 오브젝트를 기반으로 하는 사실 오브젝트에서 DB2 테이블 샘플링을 시도했습니다.:

설명

DB2 Cube Views가 뷰, 별명 또는 별칭을 기준으로 하는 사실 오브젝트에서 데이터 샘플링을 시도했습니다. 뷰, 별명 및 별칭에 대해 DB2 테이블 샘플링이 지원되지 않습니다.

사용자 조치

DB2 Cube Views는 테이블에서만 정의되고 뷰, 별명 또는 별칭에서는 정의되지 않는 사실 오브젝트에서 테이블 샘플링을 수행할 수 있습니다. 최적화 어드바이저에 대해 샘플링 옵션을 끄십시오.

7014: 최적화 어드바이저가 권장사항을 판별할 수 없습니다.:

설명

최적화 어드바이저가 지정된 매개변수로 큐브 모델에 대한 권장사항을 판별할 수 없습니다.

사용자 조치

최적화 어드바이저가 요약 테이블을 권장할 수 없는 이유에 대한 자세한 정보를 보려면 정보 및 경고 메시지를 점검하십시오.

최적화 경고

7200: 큐브 모델에 비배분 치수가 하나 이상 있으므로 권장 요약 테이블에서는 지연 새로 고침을 사용합니다.:

설명

요약 테이블에서 지금 새로 고침 옵션을 선택했습니다. 하지만 큐브 모델에 비분산 치수가 정의된 경우 요약 테이블을 지금 새로 고침할 수 없습니다. 분산 치수는 임의의 중간 값에서 집계할 수 있는 SUM 및 COUNT와 같은 단순 집계 함수를 사용합니다. 비분산 치수는 STDDEV와 같은 보다 복잡 집계 함수를 사용하며 이들 함수는 기본 테이블에서 집계됩니다.

사용자 조치

요약 테이블을 기본 테이블과 동시에 유지보수하지 않아도 되는 경우 조치는 필요하지 않습니다. 요약 테이블을 기본 테이블과 동시에 유지보수해야 하는 경우 분산 치수만 정의되도록 메타데이터를 변경해야 합니다.

7201: 권장 요약 테이블의 fullselect에 하나 이상의 널(NULL) 입력 가능 속성이 컬럼으로 있으므로 table_name 권장 요약 테이블에서 지연 새로 고침을 사용합니다.:

설명

요약 테이블에서 지금 새로 고침 옵션을 선택했습니다. 하지만 요약 테이블의 fullselect에 널(NULL) 입력 가능 컬럼으로 사용되는 속성이 권장 요약 테이블에 하나 이상 들어 있습니다. 요약 테이블의 fullselect에 있는 널(NULL) 입력 가능 컬럼을 사용하면 지금 새로 고침 성능 속도가 느려질 수 있습니다. 요약 테이블을 지연 새로 고침으로 설정하여 이러한 성능 문제점을 방지했습니다.

사용자 조치

속성의 널(NULL) 입력 기능을 변경하려면 해당 속성의 SQL 표현식 또는 DB2 테이블 컬럼의 널(NULL) 입력 가능(해당 속성 또는 둘 모두에서 사용됨)을 변경해야 합니다. 구현이 어려울 수 있으므로 이러한 변경은 보통 권장되지 않습니다.

7202: table_name 테이블에 통계가 없습니다.:

설명

최적화 어드바이저가 지정된 테이블에서 유효한 테이블 통계 값을 찾을 수 없습니다.

사용자 조치

RUNSTATS 명령을 사용하여 지정된 테이블에 통계를 작성하십시오. 그런 다음, 최적화 어드바이저 마법사를 다시 실행하십시오.

7203: cube_name 큐브의 하이브리드 추출 최적화 슬라이스 **slice_name**의 경우 큐브의 각 큐브 차원에 대해 큐브 레벨을 지정해야 합니다.:

설명

최적화 슬라이스가 제대로 정의되지 않았기 때문에 최적화 어드바이저가 지정된 하이브리드 추출 유형의 최적화 슬라이스에 대해 최적화될 수 없습니다. 하이브리드 추출 유형의 최적화 슬라이스는 각 큐브 차원에 대해 큐브 레벨을 정의해야 합니다. '지정하지 않음'을 선택할 수 없습니다.

사용자 조치

최적화 슬라이스에 큐브의 각 큐브 차원에 대해 지정된 큐브 레벨이 있는지 확인하십시오.

7203: cube_name 큐브의 하이브리드 추출 최적화 슬라이스 **slice_name**의 경우 큐브의 각 큐브 차원에 대해 큐브 레벨을 지정해야 합니다.:

설명

최적화 슬라이스가 제대로 정의되지 않았기 때문에 최적화 어드바이저가 지정된 하이브리드 추출 유형의 최적화 슬라이스에 대해 최적화될 수 없습니다. 하이브리드 추출 유형의 최적화 슬라이스는 각 큐브 차원에 대해 큐브 레벨을 정의해야 합니다. '지정하지 않음'을 선택할 수 없습니다.

사용자 조치

최적화 슬라이스에 큐브의 각 큐브 차원에 대해 지정된 큐브 레벨이 있는지 확인하십시오.

7205: 하나 이상의 큐브 모델 기본 테이블이 별칭이기 때문에 권장 요약 테이블이 지면 새로 고침 갱신 메소드를 사용해야 합니다.:

설명

요약 테이블에 대한 새로 고침 지금 갱신 옵션을 선택했지만 기본 테이블이 별칭인 경우 요약 테이블이 새로 고침 지금 갱신 메소드를 사용할 수 없습니다.

사용자 조치

조치가 필요하지 않습니다.

7206: 최적화 어드바이저가 권장 요약 테이블을 판별하는 데 데이터 샘플링을 사용할 수 없습니다.:

설명

최적화 어드바이저가 데이터 샘플링을 사용하여 최적 요약 테이블 권장사항을 판별하도록 지정했습니다. 최적화 어드바이저가 사실 오브젝트의 정의 방법 때문에 데이터 샘플링을 수행할 수 없습니다. 사실 오브젝트가 별칭으로 정의된 경우, 큐브 모델의 차원에 중간 카디널리티가 있고 사실-차원 조인이 단일 컬럼에 맵핑되는 단일 속성에 있는 경우를 제외하고 최적화 어드바이저가 데이터 샘플링을 수행하려고 시도하지만 이를 수행하지 못할 수 있습니다. 차원 키를 나타내는 사실 테이블 컬럼에 정의된 색인은 별칭으로 정의된 사실 오브젝트에 대해 데이터 샘플링을 수행하는 최적화 어드바이저의 기능에 영향을 줄 수도 있습니다. 사실 오브젝트가 뷰로 정의된 경우 데이터 샘플링이 지원되지 않습니다. 사실 오브젝트가 별칭으로 정의된 경우 별칭이 맵핑되는 대상에 따라 데이터 샘플링을 수행하지 못할 수 있습니다. 데이터 샘플링을 수행하지 않고 권장사항을 작성하면 최적화 어드바이저가 최적 요약 테이블을 권장할 수 없습니다. 사실 오브젝트가 테이블인 경우 최적화 어드바이저는 항상 데이터 샘플링을 수행할 수 있습니다.

사용자 조치

권장사항을 원래대로 사용하거나, 사실 오브젝트에 대해 뷰 또는 별명 대신 테이블을 지정하고 최적화 어드바이저를 다시 실행하여 권장사항을 개선할 수 있습니다.

최적화 정보 메시지

7400: 요약 테이블은 큐브 모델이 한 개이상의 비배분 치수를 포함하므로 ROLLUP 연산자를 사용하여 정의합니다.:

설명

치수는 분산이거나 비분산 유형입니다. 분산 치수는 임의의 중간 값에서 집계할 수 있는 SUM 및 COUNT와 같은 단순 집계 함수를 사용합니다. 비분산 치수는 STDDEV와 같은 보다 복합 집계 함수를 사용하며 이들 함수는 기본 테이블에서 집계됩니다. 기본 테이블에서 비분산 치수를 집계하는 데 비용이 들지 않게 하려면 비분산 치수를 사전에 집계하는 ROLLUP 연산자를 사용하여 요약 테이블을 정의합니다.

사용자 조치

조치는 필요하지 않습니다.

7401: table_name 요약 테이블이 권장됩니다. 행은 rows, MB 테이블 크기는 n, MB 색인 크기는 n으로 추정됩니다.:

설명

이 메시지는 추정된 행 수, 추정된 디스크 스페이스 및 인덱스에 사용되는 추정된 디스크 스페이스가 들어 있는 권장 요약 테이블의 설명입니다.

사용자 조치

조치는 필요하지 않습니다.

7402: 지정된 디스크 스페이스 한계에 맞지 않는 요약 테이블이 n개 있습니다. 이 테이블의 추정된 누적 크기는 n MB입니다.:

설명

이 메시지에서는 디스크 스페이스 한계에 맞지 않는 권장 요약 테이블에 관한 정보를 제공합니다.

사용자 조치

권장사항에서 이러한 요약 테이블을 보려면 디스크 스페이스 한계를 더 크게 지정하여 최적화 어드바이저를 다시 실행하십시오.

7403: 권장사항에 cube_name 큐브의 최적화가 들어 있습니다.:

설명

지정된 큐브에 요약 테이블을 권장합니다. 이 큐브의 일부 쿼리가 최적화됩니다.

사용자 조치

조치는 필요하지 않습니다.

7404: 권장사항에 cube_name 큐브의 최적화가 들어 있지 않습니다.:

설명

이 큐브에 요약 테이블을 특별히 권장하지 않습니다. 이 큐브에 특정한 쿼리는 성능 향상을 표시하지 않습니다.

사용자 조치

디스크 스페이스 한계로 인해 요약 테이블이 권장사항에 들어 있지 않는 경우 디스크 스페이스 한계를 더 크게 지정하여 최적화 어드바이저를 다시 실행하십시오. 권장사항에 하나 이상의 요약 테이블을 포함하여 이 큐브의 쿼리를 최적화할 수 있습니다.

7405: 최적화 어드바이저가 권장사항을 판별하는 동안 지정된 시간 제한이 지나갔습니다.:

설명

최적화 어드바이저가 권장사항을 작성했습니다. 시간이 더 허용되면 추가 분석을 수행할 수 있으므로 최적화 어드바이저가 보다 나은 권장사항을 작성할 수 있습니다. 최적화 어드바이저의 실행 시간이 길어도 더 나은 결과가 보장되지는 않습니다.

사용자 조치

더 긴 시간을 지정하여 최적화 어드바이저를 다시 실행하거나 권장 요약 테이블을 작성하여 성능의 승인 여부를 확인할 수 있습니다.

7406: dimension_name 차원에 최적화 어드바이저로 최적화할 수 있는 계층 구조가 없습니다.:

설명

최적화 어드바이저는 반복되는 계층 구조를 최적화할 수 없습니다. 지정된 차원에 최적화될 수 있는 계층 구조가 없으므로 최적화 어드바이저는 이 차원을 무시합니다. 이 차원에서 속성을 참조하는 쿼리는 최적화되지 않습니다.

사용자 조치

조치는 필요하지 않습니다. 이 차원에서 속성을 사용하는 쿼리는 성능이 향상되지 않습니다.

7407: 권장 요약 테이블이 큐브 모델에 있는 슬라이스의 n%를 최적화합니다. 최적화된 슬라이스에서 실행되는 쿼리의 성능이 향상되어야 합니다.:

설명

SQL 쿼리가 큐브 모델의 특정 슬라이스에 액세스합니다. 성능 향상을 분석하는 한 가지 방법으로 쿼리될 수 있는 슬라이스의 어느 부분이 향상되는지를 고려합니다. 큐브 모델에서 분산 치수를 사용하는 경우 논리적으로 요약 테이블 슬라이스 위에 있는 슬라이스에 액세스하는 쿼리의 성능이 향상됩니다.

예를 들어 계층 구조가 각각 [All-Year-Quarter-Month-Day]인 Time 차원 및 [All-Country-Region-State-City-Store]인 영역 차원을 갖는 큐브 모델의 가능한 슬라이스가 30개 있습니다. 차원 계층 구조의 레벨 수를 서로 곱하여 가능한 슬라이스 수를 계산할 수 있습니다. 권장 요약 테이블이 Month-City 슬라이스를 최적화하면 해당 슬라이스 이상의 모든 슬라이스가 최적화됩니다. 이 예에서 가능한 30개 슬라이스 중 20개 또는 슬라이스의 67%(20/30)가 최적화됩니다. 요약 테이블의 기본 테이블이 중복되어야 하므로 100%라는 범위는 가능하지 않습니다. 보통 가장 낮은 슬라이스는 기본 테이블과 많이 다르지 않으므로 이들은 최적화 혜택을 덜 받습니다.

사용자 조치

조치는 필요하지 않습니다. 백분율이 낮은 경우 디스크 스페이스 한계를 더 크게 지정하여 최적화 어드바이저 마법사를 다시 실행할 수 있습니다.

7408: 데이터베이스에서 큐브 모델 메타데이터를 읽고 있는 중입니다.:

설명

최적화 어드바이저가 큐브 모델을 설명하는 메타데이터를 읽습니다. 메타데이터에는 최적화 권장사항에 큰 영향을 미치는 정보가 들어 있습니다.

사용자 조치

조치는 필요하지 않습니다.

7409: 요약 테이블에 포함시킬 집계를 선택하는 중입니다.:

설명

최적화 어드바이저가 가능한 요약 테이블 구성을 테스트하여 지정된 기준에 최적인 구성을 판별합니다.

사용자 조치

조치는 필요하지 않습니다.

7410: 큐브 모델에서 데이터를 샘플링하는 중입니다.:

설명

요약 테이블의 크기를 추정할 수 있도록 최적화 어드바이저가 사실 테이블 및 차원 테이블의 데이터 서브세트를 읽습니다. 최적화 어드바이저가 가능한 요약 테이블을 고려하는 동안 샘플링이 여러 번 일어날 수 있습니다.

사용자 조치

조치는 필요하지 않습니다.

7411: 권장 요약 테이블에서 인덱스를 정의하는 중입니다.:

설명

최적화 어드바이저가 권장 요약 테이블을 판별했고 요약 테이블에 빌드할 인덱스를 선택합니다.

사용자 조치

조치는 필요하지 않습니다.

7413: cube_name 큐브가 최적화 슬라이스를 지정하지 않았으므로 최적화 어드바이저가 드릴 다운 유형의 쿼리에 대해 큐브를 최적화합니다.:

설명

큐브에 정의된 최적화 슬라이스가 없는 경우 최적화 어드바이저가 드릴 다운 유형의 쿼리에 대해 큐브를 최적화합니다.

7414: 최적화 중인 큐브 모델에 정의된 큐브가 없으므로 보고서 유형의 쿼리에 대한 큐브가 있는 것처럼 최적화 어드바이저가 이 큐브 모델을 최적화합니다.:

설명

큐브 모델에 정의된 큐브가 없는 경우 드릴 다운 유형의 쿼리에 사용된 큐브가 있는 것처럼 최적화 어드바이저가 큐브 모델을 최적화할 수 있습니다. 사용자가 최적화하는 큐브 모델에 대한 큐브를 정의하지 않았다면, 최적화 어드바이저는 사용자가 각 큐브 차원에 대한 모든 레벨에서 정의된 드릴 다운 최적화 슬라이스가 있는 큐브가 있는 것처럼 권장합니다.

7415: 최적화 어드바이저는 cube_name 큐브의 최적화 슬라이스 slice_name에 대한 요약 테이블을 권장하지 않았습니다.:

설명

지정된 최적화 슬라이스에 대해 요약 테이블이 권장되지 않습니다.

사용자 조치

조치가 필요하지 않습니다.

7416: 작업이 중지되었습니다. 권장사항이 적합하지 않을 수도 있습니다.:

설명

최적화 어드바이저가 권고 중지 조작에 의해 중지되었습니다. 권장사항이 최적이지 아닐 수 있습니다.

사용자 조치

최적 권장사항을 위해서는 시간을 제한하지 않고 최적화 어드바이저를 다시 실행하십시오.

7417: 설정된 시간 제한에 도달했기 때문에 최적화 어드바이저가 권장사항 작성을 중지했습니다. 추가 시간을 설정하면 최적화 어드바이저가 더 나은 요약 테이블을 권장할 수 있습니다.:

설명

최적화 어드바이저에 제공된 시간이 너무 짧을 수 있습니다. 권장: 시간 제한을 길게 하여 최적화 어드바이저를 실행하십시오.

사용자 조치

시간 제한을 길게 하여 최적화 어드바이저를 실행하십시오. 그러면 최적화 어드바이저가 더 나은 요약 테이블을 권장할 수 있습니다.

7418: 요약 테이블이 사실 테이블의 행 수에 비해 너무 많은 행을 가지고 있기 때문에 최적화 어드바이저가 cube_name 큐브에 대한 요약 테이블을 권장할 수 없습니

다.:

설명

최적화 어드바이저가 큐브에 대해 요약 테이블을 권장할 수 없습니다. 요약 테이블의 행 수가 사실 테이블 행 수의 비율에 비해 너무 많습니다.

사용자 조치

최적화 슬라이스를 지정한 경우 하나 이상의 최적화 슬라이스가 너무 많은 레벨을 지정했거나 사실 테이블의 일부에 지나치게 가까운 레벨을 지정했을 수 있습니다. 일부 큐브 차원에 임의의 레벨을 지정하거나 계층 구조 맨 위에 가까운 레벨을 지정할 수 있습니다. 최적화 어드바이저를 오랫동안 실행하여 권장사항을 개선하도록 할 수도 있습니다.

OLAP 센터 메시지

10000-10600

10000: OLAP 센터에서 데이터베이스 이름을 검색할 수 없습니다.

설명

DB2에서 데이터베이스 이름 목록을 검색하는 도중 오류가 발생했습니다.

사용자 조치

OLAP 센터가 올바르게 설치되었는지 확인하십시오. 문제가 지속되면, IBM 고객만족 센터에 문의하십시오.

10001: 사용자 이름을 입력하십시오.

설명

사용자 이름 필드가 비어 있습니다.

사용자 조치

사용자 이름 필드에 사용자 이름을 입력하십시오.

10002: 암호를 입력하십시오.

설명

암호 필드가 비어 있습니다.

사용자 조치

암호 필드에 암호를 입력하십시오.

10004: SQL 표현식에 입력된 속성을 구문 분석할 수 없습니다.

설명

입력된 SQL 표현식이 제공된 컨텍스트에서 유효하지 않거나 데이터베이스에 없는 속성을 참조합니다.

사용자 조치

SQL 표현식이 SQL 표현식 빌더의 데이터 섹션에 나타나는 속성만 참조하는지 확인하십시오.

10005: 메타데이터 오브젝트가 file_name 파일로 익스포트되었습니다.

설명

제대로 익스포트되었습니다.

사용자 조치

조치는 필요하지 않습니다.

10006: 메타데이터 오브젝트를 익스포트할 파일 이름을 입력하십시오.

설명

익스포트 유틸리티가 익스포트된 메타데이터 오브젝트를 사용자가 입력한 파일 이름에 작성합니다.

사용자 조치

파일 이름 필드에 파일 이름을 입력하십시오.

10007: 익스포트할 큐브 또는 큐브 모델을 선택하십시오.

설명

익스포트 창이 큐브 또는 큐브 모델을 익스포트할 수 있습니다.

사용자 조치

익스포트할 오브젝트를 선택하십시오.

10008: 작성 중인 오브젝트의 고유한 이름을 입력하십시오.

설명

SQL 표현식 빌더에서, 작성 중인 속성 또는 치수의 이름 필드가 비어 있습니다.

사용자 조치

이름 필드에 오브젝트 이름을 입력하십시오. 오브젝트 이름이 속성 및 치수의 이름 스페이스에서 고유해야 합니다.

10009: 오브젝트의 SQL 표현식을 입력하십시오.

설명

오브젝트의 SQL 표현식 필드가 비어 있습니다.

사용자 조치

오브젝트에 대한 SQL 표현식을 입력하십시오.

10010: column_name 컬럼은 테이블 이름에 대해 규정되지 않았습니다.

설명

SQL 표현식에 입력된 컬럼 이름이 테이블 이름으로 규정되지 않았습니다.

사용자 조치

SQL 표현식의 컬럼 이름이 분리문자로 '.'를 사용하여 테이블 이름으로 규정되어 있는지 확인하십시오.

10011: column_name 컬럼은 스키마 이름에 대해 규정되지 않았습니다.

설명

SQL 표현식의 컬럼 참조가 '.'로 분리된 스키마 이름 및 테이블 이름 모두로 규정되어야 합니다.

사용자 조치

SQL 표현식의 컬럼 이름이 '.'로 분리된 스키마 이름 및 테이블 이름 모두로 규정되어 있는지 확인하십시오.

10012: 집계 스크립트의 첫 번째 요소는 차원이 될 수 없습니다.

설명

첫 번째 요소로 차원을 사용하여 집계 스크립트를 작성했습니다.

사용자 조치

집계 스크립트의 첫 번째 요소로 집계 함수를 사용하십시오.

10013: 기존의 치수를 선택하거나 집계계의 다중 매개변수 함수 `function_name`의 두 번째 매개변수로 SQL 표현식을 입력하십시오.

설명

집계 스크립트의 다중 매개변수 함수를 사용하는 경우 첫 번째 매개변수를 집계와 연관된 치수로 정의합니다. 두 번째 매개변수의 경우 기존의 치수를 선택하거나 SQL 표현식을 입력하십시오.

사용자 조치

두 번째 매개변수로 치수 또는 SQL 표현식을 입력하십시오.

10014: `function_name` 집계 함수에 일치하는 차원이 없습니다.

설명

집계 스크립트의 각 집계 함수를 하나 이상의 차원에 적용해야 합니다.

사용자 조치

집계 스크립트의 각 집계 함수를 하나 이상의 차원에 적용했는지 확인하십시오.

10015: `object_name` 오브젝트의 닫는 괄호가 누락되었습니다.

설명

OLAP 센터의 경우 표현식에 지정된 속성, 치수 또는 컬럼은 각각 `@Attribute()`, `@Measure()` 또는 `@Column()`으로 닫아야 합니다.

사용자 조치

오브젝트에 닫기 괄호를 입력하십시오.

10016: SQL 표현식의 `object_name` 오브젝트를 참조하면 유효하지 않은 참조 루프가 작성됩니다.

설명

오브젝트가 해당 SQL 표현식에서 스스로를 참조합니다.

사용자 조치

SQL 표현식의 오브젝트가 참조 루프를 작성하지 않았는지 확인하십시오.

10017: 오류가 발견되지 않았습니다. SQL 표현식이 유효합니다.

설명

SQL 표현식이 유효하지 않습니다.

사용자 조치

조치는 필요하지 않습니다.

10018: 오류가 발견되지 않았습니다. 집계 스크립트가 유효합니다.

설명

집계 스크립트의 집계 세트가 유효합니다.

사용자 조치

조치는 필요하지 않습니다.

10020: 이름을 입력하십시오.

설명

오브젝트의 이름 필드가 비어 있습니다.

사용자 조치

이름 필드에 오브젝트 이름을 입력하십시오.

10021: 스키마 이름을 입력하십시오.

설명

스키마 필드가 비어 있습니다.

사용자 조치

스키마 이름 필드에 스키마 이름을 입력하십시오.

10022: 비즈니스 이름을 입력하십시오.

설명

비즈니스 이름 필드가 비어 있습니다.

사용자 조치

비즈니스 이름을 비즈니스 인텔리전스 응용프로그램에 표시하여 일반 사용자의 오브젝트를 식별할 수 있습니다. 비즈니스 이름 필드에 비즈니스 이름을 입력하십시오.

10023: 큐브 계층 구조에 포함시킬 레벨을 하나 이상 선택하십시오.

설명

레벨이 큐브 계층 구조에 지정되지 않았습니다.

사용자 조치

큐브 계층 구조에 포함시킬 레벨을 하나 이상 선택하십시오.

10024: 큐브 사실에 대해 최소한 하나의 치수를 선택하십시오.

설명

큐브 사실에 치수가 지정되지 않았습니다.

사용자 조치

큐브 사실에 대해 최소한 하나의 치수를 선택하십시오.

10025: 큐브 계층 구조에 포함시킬 레벨을 하나 이상 선택하십시오.

설명

속성이 큐브 계층 구조에 지정되지 않았습니다.

사용자 조치

큐브 계층 구조에 포함시킬 속성을 하나 이상 선택하십시오.

10026: 큐브에서 최소한 하나의 차원을 선택하십시오.

설명

차원이 큐브에 지정되지 않았습니다.

사용자 조치

차원을 하나 이상 선택한 후 [...] 단추를 눌러 큐브 차원의 세부사항을 지정하십시오.

10027: 현재 큐브에 있는 차원이 하나 이상 지워졌습니다. 큐브 차원을 삭제하려면 예를 누르십시오. 변경사항을 저장하지 않고 창을 닫으려면 아니오를 눌러 창으로 리턴한 후 취소를 누르십시오.

설명

하나 이상의 차원 선택이 지워졌습니다. 해당 큐브 차원이 큐브에서 삭제됩니다.

사용자 조치

큐브에서 큐브 차원을 제거하려면 창에서 예를 누르십시오. 저장하지 않고 창을 닫으려면 아니오를 눌러 큐브 차원을 보존한 후 취소를 누르십시오.

10028: 함수 종속성 선택란이 선택된 경우 다대다 카디널리티(cardinality)로 속성 관계를 정의할 수 없습니다.

설명

속성 관계를 선택된 함수 종속성과 다대다 카디널리티(cardinality)로 동시에 정의할 수 없습니다.

사용자 조치

속성 관계를 서로 다른 카디널리티(cardinality)로 선택하거나 함수 종속성 선택란을 지우십시오.

10029: 속성 관계에서 왼쪽 속성 및 오른쪽 속성을 선택하십시오.

설명

왼쪽 및 오른쪽 속성 모두를 선택하지 않으면 속성 관계를 정의할 수 없습니다.

사용자 조치

왼쪽 속성과 오른쪽 속성을 모두 선택하십시오.

10030: 속성 관계의 왼쪽 및 오른쪽 속성에 동일한 속성을 선택할 수 없습니다.

설명

왼쪽 및 오른쪽 속성이 동일하면 속성 관계를 정의할 수 없습니다.

사용자 조치

다른 왼쪽 속성 및 오른쪽 속성을 선택하십시오.

10031: 해당 스키마 및 이름이 지정된 오브젝트가 이미 데이터베이스에 있습니다. 다른 이름을 입력하십시오.

설명

작성 또는 수정된 유형의 오브젝트가 이미 동일한 이름 및 스키마가 지정된 데이터베이스에 있습니다.

사용자 조치

오브젝트의 고유 이름을 입력하십시오.

10032: 최소한 하나의 속성 쌍을 지정하십시오.

설명

조인은 최소한 하나의 속성 쌍을 갖고 있어야 합니다.

사용자 조치

최소한 하나의 속성 쌍을 지정하십시오.

10033: 중복 속성 쌍은 작성될 수 없습니다.

설명

새 선택사항이 서로 일치하는 속성 쌍이 이미 있습니다.

사용자 조치

다른 왼쪽 속성 및 오른쪽 속성을 선택하십시오.

10034: 최소한 하나의 테이블을 선택하십시오.

설명

테이블이 선택되지 않았습니다.

사용자 조치

계속 진행하려면 테이블을 하나 이상 선택하십시오.

10035: 새 조인을 선택하거나 작성하여 선택된 테이블 모두를 조인하십시오.

설명

조인이 선택되지 않았습니다.

사용자 조치

선택된 테이블 모두를 조인할 새 조인을 선택하거나 작성하십시오.

10036: 최소한 하나의 속성을 선택하십시오.

설명

속성이 선택되지 않았습니다.

사용자 조치

최소한 하나의 속성을 선택하십시오.

10037: 조인을 선택하여 사실 오브젝트가 있는 차원을 조인하십시오.

설명

조인이 선택되지 않았습니다.

사용자 조치

사실 오브젝트가 있는 차원을 조인할 조인을 하나 선택하십시오.

10038: 지정된 두 테이블에서 조인을 하나 지정하십시오. join_name1 및 join_name2 조인 모두는 동일한 두 테이블을 조인합니다.

설명

동일한 테이블 쌍에서 둘 이상의 조인이 선택되었습니다.

사용자 조치

각 테이블 쌍에 대해 하나의 조인만 선택하십시오.

10039: 선택된 테이블 모두를 조인해야 합니다. table_name 테이블에서 조인을 선택하십시오.

설명

선택된 테이블 모두를 조인해야 합니다.

사용자 조치

지정된 테이블에서 조인을 선택하십시오.

10040: 선택된 테이블 수가 선택된 조인 수와 일치하지 않습니다. 조인 루프가 없는지 및 모든 테이블이 조인되었는지 검증하십시오.

설명

선택된 테이블 모두를 조인해야 합니다.

사용자 조치

조인 루프가 없는지 및 모든 테이블이 조인되었는지를 확인하십시오.

10042: 최소한 하나의 치수를 선택하십시오.

설명

치수가 지정되지 않았습니다.

사용자 조치

최소한 하나의 치수를 선택하십시오.

10043: 테이블 컬럼을 선택하십시오.

설명

테이블 컬럼이 지정되지 않았습니다.

사용자 조치

컬럼을 선택하십시오.

10044: SQL 표현식을 선택하십시오.

설명

SQL 표현식이 지정되지 않았습니다.

사용자 조치

해당 표현식을 빌드하려면 표현식 빌드 단추를 누르십시오.

10045: 집계 스크립트가 지정되지 않았습니다.

설명

집계 스크립트가 지정되지 않았습니다.

사용자 조치

집계 스크립트를 빌드하려면 스크립트 빌드 단추를 누르십시오.

10046: 표현식 빌더를 열기 전에 치수를 선택하십시오.

설명

치수가 선택되지 않았습니다.

사용자 조치

테이블에서 치수를 선택하십시오.

10047: 집계 스크립트 빌더를 열기 전에 치수를 선택하십시오.

설명

치수가 선택되지 않았습니다.

사용자 조치

치수를 선택하십시오.

10048: model_name 큐브 모델에 하나 이상의 차원이 들어 있지 않으므로 집계 스크립트 빌더는 **measure_name** 치수로 시작할 수 없습니다.

설명

큐브 모델에 하나 이상의 차원이 들어 있지 않으면 집계 스크립트를 지정할 수 없습니다.

사용자 조치

집계 스크립트를 지정하기 전에 큐브 모델에 차원을 추가하십시오.

10049: 표현식을 편집하려면 속성을 지정하십시오.

설명

속성이 선택되지 않았습니다.

사용자 조치

속성을 선택하십시오.

10050: 데이터베이스에서 메타데이터를 새로 고칩니다. 오류가 발생할 때 변경된 사항이 유실됩니다.

설명

DB2 스토어드 프로시저를 호출하는 도중 오류가 발생합니다.

사용자 조치

확인을 눌러 OLAP 센터에서 표시하는 메타데이터를 새로 고칩니다. 오류가 발생할 때 변경된 사항이 유실됩니다. 사용자가 계속 작업하도록 허용하면서 OLAP 센터에 표시된 오브젝트를 데이터베이스의 대응하는 오브젝트로 새로 고칩니다.

10051: model_name 큐브 모델은 최적화 수행시 유효성을 확인하지 않습니다. DB2에서 **message** 메시지를 리턴합니다.

설명

선택된 큐브 모델이 스토어드 프로시저 API에서 수행하는 유효성 확인을 전송하지 않으므로 OLAP 센터는 선택된 큐브 모델에서 최적화 어드바이저 마법사를 시작하지 않습니다.

사용자 조치

큐브 모델 유효성 확인 규칙에서 스토어드 프로시저 API 문서를 점검하십시오. DB2의 리턴 메시지에서 해당 지시사항을 따르십시오.

10052: 로드된 속성 또는 치수 일부가 더 이상 데이터베이스에 없는 column_names 컬럼에 맵핑됩니다. 컬럼이 속한 곳에 테이블을 리스토어 하거나 유효하지 않은 속성이나 치수 또는 둘 다를 삭제하여 문제점을 해결하십시오.

설명

이 메시지는 OLAP 센터를 시작하는 경우 또는 보기 -> 새로 고침을 누른 후에 나타납니다. 로드된 속성 또는 치수가 맵핑된 테이블이 삭제되거나 이름이 바뀌었기 때문에 이 메시지가 나타납니다.

사용자 조치

다음 방법 중 하나로 문제점을 정정하십시오.

- 삭제되거나 이름이 바뀐 테이블을 리스토어하십시오.
- 데이터베이스에 있는 테이블에 속성 또는 치수를 맵핑하십시오.
- 존재하지 않는 컬럼에 맵핑된 속성 또는 치수를 삭제하십시오.

10053: model_name 큐브 모델 최적화 유효성 확인 시 경고가 리턴됩니다. DB2에서 message 메시지를 리턴합니다.

설명

최적화 어드바이저를 시작하기 전에 OLAP 센터는 큐브 모델의 유효성을 확인하려고 하지만 DB2에서 경고를 리턴합니다. 경고에서 최적화될 수 없는 큐브 모델이 있음을 표시할 수 있습니다. 예를 들어 큐브 모델에 테이블 사이에서 정의된 제한조건이 없는 테이블을 참조하는 보기가 들어 있을 수 있습니다.

사용자 조치

DB2에서 리턴한 메시지를 점검하고 최적화 어드바이저 마법사를 계속 실행할 것인지 결정하십시오.

10060: 큐브 모델이 완전하지 않습니다. 큐브를 작성하려면 큐브 모델에 하나의 사실 오브젝트, 하나 이상의 차원 및 각 차원당 하나 이상의 계층 구조가 들어 있어야 합니다.

설명

작성할 큐브에서 큐브 모델은 유효한 상태가 아닙니다.

사용자 조치

사실 오브젝트와 최소한 하나의 차원을 갖도록 큐브 모델을 수정하십시오. 각 차원에 계층 구조가 하나 이상 있는지 확인하십시오.

10061: 큐브 모델이 삭제되면 해당 차원이 제거되고 사실이 삭제됩니다. 제거된 차원은 모든 차원 폴더에서 계속 사용 가능합니다. **model_name** 큐브 모델을 삭제하시겠습니까?

설명

확인 메시지를 삭제합니다.

사용자 조치

선택된 오브젝트가 삭제하려는 오브젝트인지 확인하고 예를 누르십시오. 선택된 오브젝트를 삭제하지 않으려면 아니오를 누르십시오.

10062: 차원이 삭제되면 해당 계층 구조 및 이에 대응하는 큐브 차원도 삭제됩니다. **dimension_name** 차원을 삭제하시겠습니까?

설명

확인 메시지를 삭제합니다.

사용자 조치

선택된 오브젝트가 삭제하려는 오브젝트인지 확인하고 예를 누르십시오. 선택된 오브젝트를 삭제하지 않으려면 아니오를 누르십시오.

10063: 큐브가 삭제되면 해당 큐브 자원, 큐브 계층 구조 및 큐브 사실도 삭제됩니다. **cube_name** 큐브를 삭제하시겠습니까?

설명

확인 메시지를 삭제합니다.

사용자 조치

선택된 오브젝트가 삭제하려는 오브젝트인지 확인하고 예를 누르십시오. 선택된 오브젝트를 삭제하지 않으려면 아니오를 누르십시오.

10064: 큐브 차원이 삭제되면 해당 큐브 계층 구조도 삭제됩니다. **cube_dimension_name** 큐브 차원을 삭제하시겠습니까?

설명

확인 메시지를 삭제합니다.

사용자 조치

선택된 오브젝트가 삭제하려는 오브젝트인지 확인하고 예를 누르십시오. 선택된 오브젝트를 삭제하지 않으려면 아니오를 누르십시오.

10065: object_name을 삭제하시겠습니까?

설명

확인 메시지를 삭제합니다.

사용자 조치

선택된 오브젝트가 삭제하려는 오브젝트인지 확인하고 예를 누르십시오. 선택된 오브젝트를 삭제하지 않으려면 아니오를 누르십시오.

10066: 차원이 제거되면 이에 대응하는 모든 큐브 차원이 해당 큐브에서 제거됩니다. object_name에서 dimension_name 차원을 제거하시겠습니까?

설명

차원 확인 메시지를 제거합니다.

사용자 조치

선택된 오브젝트가 제거할 오브젝트인지 확인하고 예를 누르십시오. 선택된 오브젝트를 제거하지 않으려면 아니오를 누르십시오.

10067: 이름이 file_name인 파일이 이미 존재합니다. 해당 콘텐츠를 겹쳐 쓰시겠습니까?

설명

파일 확인 메시지를 겹쳐 씁니다.

사용자 조치

입력한 파일 이름의 콘텐츠를 겹쳐 쓸지 확인하십시오.

10068: 이름이 object_name인 오브젝트 및 schema_name 스키마의 데이터 유형을 판별할 수 없습니다. 데이터베이스에서 message 정보를 리턴합니다.

설명

OLAP 센터가 지정된 오브젝트의 소스 데이터 유형 또는 집계 데이터 유형을 판별할 수 없습니다.

사용자 조치

지정된 오브젝트의 SQL 표현식이 올바른지 확인하십시오. 문제를 해결할 수 없으면, IBM 고객만족센터에 문의하십시오.

10069: 이름이 measure_name인 치수 및 schema_name 스키마의 소스 데이터 유형을 판별할 수 없습니다.

설명

지정된 치수에 유효하지 않은 소스 표현식이 있으므로 OLAP 센터가 지정된 치수의 소스 데이터 유형을 판별할 수 없습니다. 해당 치수의 유효성은 독립형 표현식이 아닌 참조되는 치수의 집계로 확인되므로 없음(None) 집계 설정이 적용될 때 치수에 유효하지 않은 소스 표현식이 들어 있을 수 있습니다.

사용자 조치

다음 조치 중 하나를 수행할 수 있습니다.

- 없음(None) 집계 설정으로 유효성 확인을 제대로 수행하도록 지정된 치수의 소스 표현식을 변경하십시오.
- 표현식에서 지정된 치수를 사용하지 마십시오.

10070: 사실 오브젝트가 삭제되면 치수도 삭제됩니다. facts_name 사실을 삭제하시겠습니까?

설명

확인 메시지를 삭제합니다.

사용자 조치

선택된 오브젝트가 삭제하려는 오브젝트인지 확인하고 예를 누르십시오. 선택된 오브젝트를 삭제하지 않으려면 아니오를 누르십시오.

10071: 선택된 오브젝트 모두가 데이터베이스에서 삭제됩니다. 이러한 오브젝트를 삭제하시겠습니까?

설명

두 개 이상의 오브젝트를 선택한 후 삭제 옵션을 선택했습니다.

사용자 조치

선택된 오브젝트가 삭제하려는 오브젝트인지 확인하고 예를 누르십시오. 선택된 오브젝트를 삭제하지 않으려면 아니오를 누르십시오.

10072: 선택된 오브젝트 중 일부를 삭제할 수 없습니다. 이러한 오브젝트가 데이터베이스에 남아 있습니다.

설명

OLAP 센터가 선택된 오브젝트 모두를 삭제할 수 없습니다. 이는 선택된 오브젝트 중 일부가 데이터베이스의 기타 오브젝트에서 참조되므로 선택된 오브젝트를 삭제하면 참조 중인 오브젝트가 유효하지 않게 되기 때문일 수 있습니다.

사용자 조치

조치는 필요하지 않습니다.

10073: 선택된 오브젝트를 삭제할 수 없습니다.

설명

OLAP 센터가 선택된 오브젝트를 삭제할 수 없습니다. 이것은 선택된 오브젝트가 데이터베이스의 다른 오브젝트에서 참조되기 때문일 수 있으며 선택된 오브젝트를 삭제하면 참조 오브젝트가 유효하지 않게 됩니다.

사용자 조치

조치가 필요하지 않습니다.

10074: directory_name에서 object_name(를) 제거하시겠습니까?

설명

확인 메시지를 제거합니다.

사용자 조치

선택된 오브젝트가 제거하려는 오브젝트인지 확인한 후 예를 누르십시오. 선택된 오브젝트를 제거하지 않으려면 아니오를 누르십시오.

10075: 선택된 모든 오브젝트가 directory_name에서 제거됩니다. 이 오브젝트를 제거하시겠습니까?

설명

둘 이상의 오브젝트가 선택되었으며 제거 옵션이 선택되었습니다.

사용자 조치

선택된 오브젝트가 제거하려는 오브젝트인지 확인한 후 예를 누르십시오. 선택된 오브젝트를 제거하지 않으려면 아니오를 누르십시오.

10076: 선택된 오브젝트 중 일부를 제거할 수 없습니다. 이러한 오브젝트가 데이터베이스에 남아 있습니다.

설명

OLAP 센터가 선택된 모든 오브젝트를 제거할 수 없습니다. 이것은 선택된 오브젝트 중 일부가 데이터베이스의 다른 오브젝트에서 참조되기 때문일 수 있으며 선택된 오브젝트를 삭제하면 참조 오브젝트가 유효하지 않게 됩니다. 예를 들어 큐브 계층 구조의 경우 상위 오브젝트에서 최소한 하나의 하위 오브젝트가 필요하기 때문일 수도 있습니다.

사용자 조치

조치가 필요하지 않습니다.

10077: 선택된 오브젝트를 제거할 수 없습니다.

설명

OLAP 센터가 선택된 오브젝트를 제거할 수 없습니다. 이것은 선택된 오브젝트가 데이터베이스의 다른 오브젝트에서 참조되기 때문일 수 있으며 선택된 오브젝트를 삭제하면 참조 오브젝트가 유효하지 않게 됩니다.

사용자 조치

조치가 필요하지 않습니다.

10078: 조작을 완료했습니다. 데이터베이스에서 정보 메시지 database_name을 리턴했습니다.

설명

데이터베이스 조작에 성공했지만 일부 정보 메시지가 리턴되었습니다.

사용자 조치

정보 메시지를 확인한 후 후속 조치를 취해야 하는지 여부를 결정하십시오.

10080: XML의 두 번째 패스 중 type 유형의 오브젝트를 찾을 수 없습니다.

설명

읽고 있는 XML에서 참조된 오브젝트를 찾을 수 없었습니다.

사용자 조치

임포트 중인 XML 파일이 올바르게 형성되었는지 확인하십시오. OLAP 센터를 시작하는 중에 이 오류가 발생하면, IBM 고객만족센터에 문의하십시오.

10081: 시스템이 파일 `file_name`에서 XML 구문 분석에 실패했습니다. 행 `line_number` 문자 `character_number`에서 오류가 발생했습니다. 구문 분석기가 다음 정보를 리턴했습니다. 정보.

설명

XML 파일을 импорт하려는 중 구문 분석 오류가 발생했습니다.

사용자 조치

임포트하려는 XML 파일이 올바르게 구성되어 있는지, 유효한 Cube Views XML 메타데이터 파일인지 각각 확인하십시오. 행 및 문자 번호를 확인하여 오류를 찾으십시오.

10082: 파일 `file_name`에서 예기치 않은 구문 분석기 예외가 발생했습니다. 다음 정보가 리턴되었습니다. 정보.

설명

XML 파일을 импорт하려는 중 예기치 않은 구문 분석 오류가 발생했습니다.

사용자 조치

임포트하려는 XML 파일이 올바르게 구성되어 있는지, 유효한 Cube Views XML 메타데이터 파일인지 각각 확인하십시오. 행 및 문자 번호를 확인하여 오류를 찾으십시오.

10084: `schema_name` 스키마에 이름이 `object_name`인 오브젝트가 이미 존재합니다. 해당 오브젝트를 작성할 수 없습니다. 새 오브젝트의 고유 이름, 스키마 또는 둘다를 입력하십시오.

설명

OLAP 센터에서 새 오브젝트를 작성하려고 하지만 이름 및 스키마가 동일한 유형의 오브젝트가 이미 존재합니다.

사용자 조치

작성 중인 오브젝트의 다른 이름, 스키마 또는 둘다를 입력하십시오.

10085: `schema_name` 스키마에 이름이 `object_name`인 오브젝트 이미 존재합니다. 오브젝트 이름을 바꿀 수 없습니다. 이름을 바꾸는 중인 오브젝트의 고유 이름, 스키마 또는 둘다를 입력하십시오.

설명

OLAP 센터에서 오브젝트의 이름을 바꾸려고 하지만 이름 및 스키마가 동일한 유형의 오브젝트가 이미 존재합니다.

사용자 조치

이름을 바꾸는 중인 오브젝트의 다른 이름, 스키마 또는 둘다를 입력하십시오.

10086: 데이터베이스에 연결할 수 없습니다. DB2가 message 메시지를 리턴합니다.

설명

OLAP 센터에서 데이터베이스에 연결할 수 없었습니다. DB2에서 제공하는 일부 오류 정보가 메시지에 들어 있습니다.

사용자 조치

DB2가 리턴한 텍스트를 읽고 문제점을 수정하십시오.

10087: object_name1 메타데이터 오브젝트는 type 유형의 object_name2 오브젝트가 참조하므로 삭제될 수 없습니다.

설명

하나 이상의 기타 메타데이터 오브젝트에서 사용 중이므로 선택된 메타데이터 오브젝트를 삭제할 수 없습니다.

사용자 조치

기타 메타데이터 오브젝트에서 부분에 속하는 오브젝트를 제거한 후 다시 오브젝트 삭제를 시도하십시오.

10088: JDBC 드라이버 관리 프로그램으로 DB2 드라이버를 등록하는 도중 오류가 발생했습니다. 데이터베이스에 연결할 수 없습니다. message 정보가 리턴됩니다.

설명

DB2 데이터베이스에 연결하기 전에 OLAP 센터는 드라이버 관리 프로그램으로 사용할 JDBC 드라이버를 등록해야 합니다. JDBC 드라이버를 등록하는 도중 오류가 발생했습니다.

사용자 조치

DB2 설치를 점검하여 db2java.zip 및 db2jcc.jar 파일이 설치되었는지 확인하십시오. Java 및 JDBC 구성요소가 제대로 설치되었는지 확인하십시오. 문제점을 해결하는데 도움이 되도록 메시지에 리턴된 정보를 읽으십시오.

10089: 데이터베이스에 액세스하는 도중 오류가 발생했습니다. 데이터베이스가 \n SQL State: message\n SQL Error Code: code\n SQL Message: SQL_message 정보를 리턴합니다.

설명

OLAP 센터 응용프로그램이 API 스토어드 프로시저를 사용하여 DB2를 호출했습니다. 실행 명령이 OLAP 센터에서 처리할 수 없는 SQLException을 전달했습니다.

사용자 조치

메시지에 제공된 추가 오류 정보를 사용하여 문제점을 해결하십시오. 문제를 해결할 수 없으면, IBM 고객만족센터에 문의하십시오.

10090: DB2 스토어드 프로시저의 실행으로 거짓 리턴 코드가 발생했습니다. 리턴된 XML 문서에서 오류 정보를 찾지 못했습니다. IBM 고객만족센터에 문의하십시오.

설명

OLAP 센터 응용프로그램이 API 스토어드 프로시저를 사용하여 DB2를 호출했습니다. 실행 명령이 거짓을 리턴하지만 스토어드 프로시저에서 리턴한 XML 문서에 오류 정보가 없었습니다.

사용자 조치

조작을 제대로 완료할 수 있지만 이 문제점을 IBM 고객만족센터에 보고해야 합니다.

10091: 데이터베이스 API 호출을 처리하는 도중 오류가 발생했습니다. \n SQL State:message\n SQL Error Code: code\n Operation:operation\n Status ID: ID\n Status Text: text 정보가 리턴됩니다.

설명

일부 OLAP 센터 변경사항을 실행하는 도중 OLAP 센터 스토어드 프로시저 API 호출 시 오류가 발생했습니다.

사용자 조치

메시지에 포함된 정보를 참조하십시오. 문제를 해결할 수 없으면, IBM 고객만족센터에 문의하십시오.

10092: 데이터베이스 API 호출에서 리턴한 XML을 구문 분석하는 도중 오류가 발생했습니다. message 정보가 리턴됩니다.

설명

OLAP 센터 스토어드 프로시저 API 호출 시 불완전하거나 잘못 형성된 XML을 리턴합니다. OLAP 센터가 리턴한 XML을 읽을 수 없습니다.

사용자 조치

메시지에 들어 있는 정보를 사용하여 문제점을 해결하십시오. 문제를 해결할 수 없으면, IBM 고객만족센터에 문의하십시오.

10093: file_name 파일이 존재하지 않습니다.

설명

지정된 파일이 존재하지 않습니다.

사용자 조치

존재하는 파일을 지정하십시오.

10094: file_name 파일을 읽는 도중 I/O 오류가 발생했습니다. message 시스템 정보가 리턴됩니다.

설명

파일에서 읽는 도중 I/O 오류가 발생했습니다.

사용자 조치

시스템 정보를 점검하여 문제점 해결을 시도하거나 다른 파일을 지정하십시오.

10095: file_name 파일에 쓰는 도중 I/O 오류가 발생했습니다. message 시스템 정보가 리턴됩니다.

설명

파일에 쓰는 도중 I/O 오류가 발생했습니다.

사용자 조치

시스템 정보를 점검하여 문제점 해결을 시도하거나 다른 파일을 지정하십시오.

10096: 데이터베이스 스키마를 검색하는 쿼리에 실패했습니다. 데이터베이스에서 다음 정보를 리턴했습니다. message.

설명

데이터베이스 스키마를 검색하는 쿼리에 실패했습니다.

사용자 조치

데이터베이스 정보를 확인하여 문제를 해결하십시오.

10097: 스키마 테이블을 검색하는 쿼리에 실패했습니다. 데이터베이스에서 다음 정보를 리턴했습니다. message.

설명

스키마 테이블을 검색하는 쿼리에 실패했습니다.

사용자 조치

데이터베이스 정보를 확인하여 문제를 해결하십시오.

10098: 테이블 컬럼을 검색하는 쿼리에 실패했습니다. 데이터베이스에서 다음 정보를 리턴했습니다. message.

설명

테이블 컬럼을 검색하는 쿼리에 실패했습니다.

사용자 조치

데이터베이스 정보를 확인하여 문제를 해결하십시오.

10099: DB2 연결 커미트에 실패했습니다. 데이터베이스에서 다음 정보를 리턴했습니다. message.

설명

DB2 연결 커미트에 실패했습니다.

사용자 조치

데이터베이스 정보를 확인하여 문제를 해결하십시오.

10100: DB2 연결 롤백에 실패했습니다. 데이터베이스에서 다음 정보를 리턴했습니다. message.

설명

DB2 연결 롤백에 실패했습니다.

사용자 조치

데이터베이스 정보를 확인하여 문제를 해결하십시오.

10101: Object_name 오브젝트는 **cube_name** 큐브의 마지막 큐브 차원
이므로 삭제할 수 없습니다. 큐브에 유효한 큐브 차원이 하나 이상 있어야
합니다.

설명

OLAP 센터가 큐브의 마지막 큐브 차원을 삭제하려고 했습니다.

사용자 조치

큐브에 유효한 큐브 차원이 하나 이상 있어야 합니다. 큐브에서 마지막 큐브 차원 삭
제를 시도하지 마십시오.

10102: type1 유형의 **object_name1** 오브젝트가 찾을 수 없는
object_name2 오브젝트 또는 **type2** 유형을 참조합니다.

설명

읽고 있는 XML 파일의 오브젝트가 찾을 수 없는 오브젝트를 참조합니다. импорт 중
에 오류가 발생하면 참조될 오브젝트가 импорт 중인 파일에 없을 수도 있습니다.

사용자 조치

импорт를 수행하는 경우 импорт를 제대로 수행하는 데 필요한 모든 오브젝트가 파일에
들어 있는지 확인하십시오. OLAP 센터를 시작하는 중에 오류가 발생하면, IBM 고객
만족센터에 문의하십시오.

10103: 사실 오브젝트에 하나 이상의 치수가 들어 있어야 하므로 삭제할 수
없습니다.

설명

사실 오브젝트에 하나 이상의 치수가 들어 있어야 하므로 치수를 삭제할 수 없습니다.

사용자 조치

조치가 필요하지 않습니다.

10104: 큐브 사실에 하나 이상의 치수가 들어 있어야 하므로 제거할 수 없
습니다.

설명

큐브 사실에 하나 이상의 치수가 들어 있어야 하므로 치수를 제거할 수 없습니다.

사용자 조치

조치가 필요하지 않습니다.

10105: 큐브 계층 구조에 하나 이상의 큐브 레벨이 들어 있어야 하므로 큐브를 제거할 수 없습니다.

설명

큐브 계층 구조에 하나 이상의 큐브 레벨이 들어 있어야 하므로 큐브 레벨을 제거할 수 없습니다.

사용자 조치

조치가 필요하지 않습니다.

10106: hierarchy_name 계층 구조와 연관된 hierarchy_name 큐브 계층 구조의 큐브 레벨이 참조하므로 제거할 수 없습니다.

설명

속성을 제거 중인 계층 구조와 연관된 큐브 계층 구조의 큐브 레벨이 참조하므로 레벨을 제거할 수 없습니다.

사용자 조치

조치가 필요하지 않습니다.

10107: level_name 레벨과 연관된 level_name 큐브 레벨이 참조하므로 제거할 수 없습니다.

설명

속성을 제거 중인 레벨과 연관된 큐브 레벨이 참조하므로 이 속성을 제거할 수 없습니다.

사용자 조치

조치가 필요하지 않습니다.

10108: dimension_name 차원과 연관된 hierarchy_name 계층 구조가 참조하므로 제거할 수 없습니다.

설명

속성을 제거 중인 차원과 연관된 계층 구조가 참조하므로 이 속성을 제거할 수 없습니다.

사용자 조치

조치가 필요하지 않습니다.

10109: dimension_name 차원과 연관된 **join_name** 조인이 참조하므로 제거할 수 없습니다.

설명

속성을 제거 중인 차원과 연관되어 있는 조인이 참조하므로 이 속성을 제거할 수 없습니다.

사용자 조치

조치가 필요하지 않습니다.

10110: hierarchy_name 계층 구조와 연관된 **hierarchy_name** 큐브 계층 구조의 큐브 레벨이 참조하므로 제거할 수 없습니다.

설명

편집 중인 계층 구조와 연관된 큐브 계층 구조의 큐브 레벨이 참조하므로 레벨을 위 또는 아래로 이동시킬 수 없습니다.

사용자 조치

조치가 필요하지 않습니다.

10111: level_name 레벨의 유일한 레벨 키 속성이므로 제거할 수 없습니다.

설명

제거 중인 레벨의 유일한 레벨 키 속성이므로 이 속성을 제거할 수 없으며, 레벨에는 하나 이상의 레벨 키 속성이 있어야 합니다.

사용자 조치

조치가 필요하지 않습니다.

10112: level_name 레벨의 디폴트 속성이므로 제거할 수 없습니다.

설명

속성을 제거 중인 레벨의 디폴트 속성이므로 이 속성을 제거할 수 없습니다.

사용자 조치

조치가 필요하지 않습니다.

10113: dimension_name 차원과 연관된 level_name 레벨이 참조하므로 제거할 수 없습니다.

설명

속성을 제거 중인 차원과 연관된 레벨이 참조하므로 이 속성을 제거할 수 없습니다.

사용자 조치

조치가 필요하지 않습니다.

10114: hierarchy_name 계층 구조의 유일한 레벨이므로 제거할 수 없습니다.

설명

계층 구조에서 유일한 레벨이므로 이 레벨을 제거할 수 없으며, 계층 구조에는 하나 이상의 레벨이 들어 있어야 합니다.

사용자 조치

조치가 필요하지 않습니다.

10200: 임포트되고 있는 파일은 UTF-8 인코딩이 없습니다. UTF-8 인코딩이 있는 파일을 선택하십시오.

설명

OLAP 센터는 단지 UTF-8 인코딩에 있는 파일만 임포트할 수 있습니다.

사용자 조치

지원되는 인코딩이 있는 파일을 임포트하십시오.

10201: 요약 테이블을 새로 고치는 데 사용하는 SQL 스크립트의 파일 이름을 입력하십시오.

설명

지연 갱신 옵션을 선택하면 최적화 어드바이저 마법사가 SQL 스크립트를 작성하여 요약 테이블을 새로 고칩니다. 이 스크립트를 파일에 저장한 후 실행하여 요약 테이블을 새로 고쳐야 합니다.

사용자 조치

파일 이름을 입력하여 SQL 스크립트를 저장하십시오.

10202: 요약 테이블을 작성하는 데 사용하는 SQL 스크립트의 파일 이름을 입력하십시오.

설명

최적화 어드바이저 마법사가 SQL 스크립트를 생성하여 요약 테이블을 작성합니다. 이 스크립트를 파일에 저장한 후 실행하여 요약 테이블을 작성해야 합니다.

사용자 조치

파일 이름을 입력하여 SQL 스크립트를 저장하십시오.

10203: 선택된 치수의 집계 설정은 없음(None)일 수 없습니다. 표현식의 다른 치수를 독점적으로 참조하는 계산된 치수만 없음(None) 집계 설정을 지정할 수 있습니다.

설명

다른 치수를 독점적으로 참조하는 표현식만 사용하는 치수에서만 없음(None) 집계 설정을 선택할 수 있습니다.

사용자 조치

다른 집계를 선택하십시오.

10204: 차원이 없습니다. 큐브 모델에 추가하는 새 차원을 작성하십시오.

설명

차원이 존재하지 않습니다. 새 차원을 작성하여 큐브 모델에 추가하십시오.

사용자 조치

차원을 추가하는 대신 새 차원을 작성하십시오.

10205: 모든 기존 차원이 큐브 모델에 이미 들어 있으므로 추가할 차원이 없습니다.

설명

모든 기존 차원이 큐브 모델에 추가되었습니다.

사용자 조치

조치는 필요하지 않습니다.

10206: 선택한 옵션을 변경했습니다. 최적화 어드바이저 마법사에서 새 권장 사항을 보려면, 최적화 어드바이저 마법사를 다시 실행해야 합니다. 최적화 어드바이저 마법사 프로세스를 다시 실행하지 않으면 이전 옵션에 작성된 권장사항이 표시됩니다. 최적화 어드바이저 마법사 프로세스를 다시 실행하시겠습니까?

설명

최적화 어드바이저 마법사 프로세스를 실행한 후 선택한 옵션을 변경했습니다. 요약 테이블의 갱신된 권장사항을 보려면 최적화 어드바이저 마법사 프로세스를 다시 실행하십시오. 최적화 어드바이저 마법사 프로세스를 다시 실행하지 않으면 이전 옵션에 작성된 권장사항이 표시됩니다.

사용자 조치

최적화 어드바이저 마법사 프로세스를 실행하려면 예를 누르십시오. 최적화 어드바이저 마법사 프로세스를 다시 실행하지 않으려면 아니오를 누르십시오.

10207: 차원 테이블이 발견되지 않았습니다.

설명

차원 테이블이 발견되지 않았습니다.

사용자 조치

참조 무결성 제한조건이 올바르게 설정되었는지 확인하십시오.

10208: OLAP 센터가 직접 표시할 수 없는 Object_names 오브젝트가 데이터베이스에 존재합니다. 나중에 이 오브젝트 때문에 OLAP 센터에 문제점이 발생할 수 있습니다. 예를 눌러 오브젝트를 삭제하거나 아니오를 눌러 데이터베이스의 오브젝트를 보존하십시오.

설명

OLAP 센터가 직접 표시할 수 없는 데이터베이스의 여러 오브젝트(계층 구조 또는 사실)를 발견했습니다. 이러한 오브젝트는 이미 존재하거나 메타데이터 импорт 후에 작성되었을 수 있습니다. 나중에 이러한 오브젝트 때문에 OLAP 센터에서 이름 충돌 및 참조 문제점이 발생할 수 있습니다. 이러한 오브젝트를 보존할 이유가 없으면 삭제하는 것이 좋습니다.

사용자 조치

예를 눌러 오브젝트를 삭제하거나 아니오를 눌러 데이터베이스의 오브젝트를 보존하십시오.

10209: 임포트 작업 중 예기치 않은 오류가 발생했습니다. 입력 XML 파일에 오류가 있는지 점검하십시오.

설명

임포트 도중 스토어드 프로시저 API가 출력 XML에 아무 내용이 없는 경고를 리턴했습니다.

사용자 조치

입력 XML 메타데이터가 OLAP 메타데이터 스키마에 정의된 형식과 일치하고 XML 파일이 해당 파일에서 참조되는 모든 메타데이터 오브젝트를 정의하는지 확인하십시오.

10210: 임포트 작업에 실패했습니다. 스토어드 프로시저 API가 message 메시지를 리턴합니다.

설명

임포트 프로세스 도중 스토어드 프로시저 API가 오류 메시지를 리턴합니다.

사용자 조치

메시지에 제공된 정보를 사용하여 문제점을 해결하십시오. 문제를 해결할 수 없으면, IBM 고객만족센터에 문의하십시오.

10211: function_name 집계 함수가 숫자 인수를 예상하므로 숫자가 아닌 measure_name 치수는 해당 집계 함수를 사용할 수 없습니다.

설명

숫자가 아닌 데이터 유형의 치수는 숫자 집계 함수를 사용할 수 없습니다. 숫자가 아닌 데이터의 집계 함수로 MIN, MAX 또는 COUNT만을 선택할 수 있습니다.

사용자 조치

다른 집계 함수를 선택하십시오.

10212: 입력 XML 파일에서 오브젝트를 읽을 수 없습니다. 입력 XML 파일에 오류가 있는지 점검하십시오.

설명

OLAP 센터가 입력 XML 파일에서 오브젝트를 읽는 데 실패했습니다.

사용자 조치

입력 XML 메타데이터가 OLAP 메타데이터 스키마에 정의된 형식과 일치하고 XML 파일이 해당 파일에서 참조되는 모든 메타데이터 오브젝트를 정의하는지 확인하십시오.

10213: file_name 입력 XML 파일이 지정된 디렉토리에 존재하지 않습니다.

설명

입력 XML 파일이 지정된 디렉토리에 존재하지 않습니다.

사용자 조치

입력 XML 파일이 지정된 디렉토리에 존재하는지 확인하십시오.

10214: 임포트할 파일에 포함된 object_name 오브젝트가 데이터베이스에 존재하지 않는 **column_name** 컬럼을 참조합니다. 파일을 임포트하기 전에 임포트할 파일의 메타데이터 오브젝트에서 참조하는 테이블 및 컬럼이 확인하십시오.

설명

데이터베이스에 존재하지 않는 테이블 및 컬럼을 참조하는 오브젝트가 임포트할 XML 파일에 들어 있습니다.

사용자 조치

임포트할 XML 파일의 오브젝트가 참조하는 테이블이 파일을 임포트하기 전에 데이터베이스에 존재하는지 확인하십시오.

10215: OLAP 센터가 최적화 어드바이저 마법사에서 권장하는 **SQL** 스크립트를 실행할 수 없습니다. 데이터베이스에서 **message** 정보를 리턴합니다.

설명

OLAP 센터가 최적화 어드바이저 마법사에서 권장하는 SQL 스크립트를 실행할 수 없습니다. SQL 스크립트를 실행하는 데 필요한 특권이 충분하지 않을 수 있습니다.

사용자 조치

최적화 어드바이저 권장사항을 실행하는 데 필요한 올바른 권한이 있는지 확인하십시오. 필요한 권한은 OLAP 센터 온라인 도움말의 "권한 및 특권" 주제에 설명되어 있습니다. 큐브 모델의 최적화에 관한 정보는 *DB2 Cube Views Setup and User's Guide* 를 참조하십시오.

10216: 최적화 어드바이저의 권장사항이 지정된 파일에 저장되었습니다.

설명

권장되는 요약 테이블 작성 SQL 스크립트 및 가능한 경우 요약 테이블 새로 고침 SQL 스크립트가 지정된 파일에 저장되었습니다.

사용자 조치

조치는 필요하지 않습니다.

10217: 요약 테이블 및 해당 인덱스가 작성되었습니다.

설명

최적화 어드바이저에서 권장한 요약 테이블 및 인덱스가 데이터베이스에 제대로 작성되었습니다.

사용자 조치

조치는 필요하지 않습니다.

10218: 뷰를 선택했습니다. 최적화 어드바이저가 해당 뷰에서 참조하는 테이블의 참조 제한조건이 존재하는지 확인할 수 없습니다.

설명

제한조건이 없는 테이블을 참조하는 뷰를 사용하여 큐브 모델의 요약 테이블을 작성하는 경우 최적화는 효과가 없을 수 있습니다. 뷰에서 참조하는 테이블에 제한조건이 있는지 최적화 어드바이저가 발견할 수 없습니다.

사용자 조치

뷰에서 참조하는 테이블에 제한조건이 없으며 최적화 어드바이저를 실행하려고 경우 1. 큐브 모델에서 뷰를 사용할 수 없습니다. 2. 또는 최적화 어드바이저를 실행하기 전에 테이블의 제한조건을 작성할 수 있습니다.

10219: 최적화 어드바이저 마법사를 취소하면 권장 SQL 스크립트가 저장되지 않고 버려집니다. 권장 SQL 스크립트를 저장하지 않고 최적화 어드바이저 마법사를 닫으려면 예를 누르십시오. 권장 SQL 스크립트를 저장할 수 있도록 최적화 어드바이저 마법사로 리턴하려면 아니오를 누르십시오.

설명

최적화 어드바이저 마법사에서 취소 단추를 누르면 권장 SQL 스크립트가 저장되지 않고 버려집니다.

사용자 조치

권장 SQL 스크립트를 저장하지 않고 최적화 어드바이저 마법사를 닫으려면 예를 누르고, 최적화 어드바이저 마법사를 계속 사용하여 SQL 스크립트를 저장할 수 있도록 최적화 어드바이저 마법사로 리턴하려면 아니오를 누르십시오.

10220: 지정된 파일이 이전 DB2 Cube Views 버전의 형식입니다. импорт 마법사가 지정된 파일에서 설명한 메타데이터 오브젝트를 DB2 Cube Views의 현재 버전으로 변환할 수 있습니다. 지정된 파일에서 설명한 메타데이터 오브젝트를 변환하여 계속 импорт하려면 예를 누르십시오. 다른 파일을 지정하거나 импорт 마법사를 닫으려면 아니오를 눌러서 Imports를 중지하십시오.

설명

지정된 메타데이터 소스 파일에 이전 DB2 Cube Views 버전의 형식으로 된 XML이 들어 있습니다. импорт 마법사가 파일을 읽은 후 DB2 Cube Views의 현재 버전에서 인식할 수 있는 형식으로 콘텐츠를 변환할 수 있습니다.

사용자 조치

지정된 소스 파일에서 설명한 메타데이터 오브젝트를 DB2 Cube Views의 현재 형식으로 변환하려면 예를 누르십시오. импорт 옵션 페이지는 새 DB2 Cube Views 버전으로 변환된 지정된 파일의 오브젝트를 표시합니다. 지정된 파일에서 설명한 메타데이터 오브젝트를 변환하지 않도록 импорт 마법사를 중지하려면 아니오를 누르십시오. 다른 메타데이터 소스 파일을 지정하거나 импорт 마법사를 닫을 수 있습니다.

10221: 권장 요약 테이블이 disk_space_sizeMB의 디스크 공간을 사용할 것으로 예상됩니다.

설명

최적화 어드바이저에서 권장하는 요약 테이블이 지정된 디스크 공간을 사용할 것으로 예상됩니다.

사용자 조치

권장 스크립트를 실행하기 전에 사용 가능한 최소 지정 디스크 공간이 있는지 확인하십시오.

10222: 최적화 어드바이저를 중지하시겠습니까? 최적화 어드바이저를 중지하여 권장사항을 작성하지 않도록 하고 지금까지 확보한 권장사항을 리턴하려면 예를 누르십시오. 최적화 어드바이저가 권장사항을 계속해서 작성하도록 허용하려면 아니오를 누르십시오.

설명

최적화 어드바이저가 최적화 권장사항을 작성하는 중에 중지를 눌렀습니다.

사용자 조치

권장사항을 작성하지 못하도록 최적화 어드바이저를 중지하고 지금까지 확보한 권장사항을 리턴하려면 **예**를 누르십시오. 최적화 어드바이저가 권장사항을 계속해서 작성하도록 허용하려면 **아니오**를 누르십시오. **예**를 누르면 어드바이저 마법사는 지금까지 확보한 권장사항을 표시합니다.

10300: SQL 표현식에 입력된 measure_name 치수를 구문 분석하는 데 실패했습니다.

설명

지정된 SQL 표현식이 제공된 컨텍스트에서 유효하지 않거나 데이터베이스에 없는 치수를 참조합니다.

사용자 조치

SQL 표현식이 SQL 표현식 빌더의 데이터 목록에 나타나는 치수만 참조하는지 확인하십시오.

10301: SQL 표현식에 입력된 column_name 컬럼을 구문 분석하는 데 실패했습니다.

설명

지정된 SQL 표현식이 제공된 컨텍스트에서 유효하지 않거나 데이터베이스에 없는 컬럼을 참조합니다.

사용자 조치

SQL 표현식이 SQL 표현식 빌더의 데이터 목록에 나타나는 컬럼만 참조하는지 확인하십시오.

10302: attribute_name 속성이 스키마 이름으로 규정되지 않았습니다.

설명

SQL 표현식의 속성 참조가 '.'로 분리된 스키마 이름으로 규정되어야 합니다.

사용자 조치

SQL 표현식의 모든 속성 참조가 '.'로 분리된 스키마 이름으로 규정되어 있는지 확인하십시오.

10303: measure_name 치수가 스키마 이름으로 규정되지 않았습니다.

설명

SQL 표현식의 치수 참조가 '.'로 분리된 스키마 이름으로 규정되어야 합니다.

사용자 조치

SQL 표현식의 모든 치수 참조가 '.'로 분리된 스키마 이름으로 규정되어 있는지 확인하십시오.

10304: object_name 오브젝트 태그에서 오브젝트 이름이 누락되었습니다.

설명

지정된 SQL 표현식에 빈 컬럼 태그 @Column, 빈 속성 태그 @Attribute 또는 빈 치수 태그 @Measure가 들어 있습니다.

사용자 조치

오브젝트 유형 태그 @Column, @Measure 및 @Attribute에 닫힌 오브젝트 이름이 있는지 확인하십시오.

10305: 지정된 표현식이 유효하지 않습니다. 데이터베이스에서 다음 정보를 리턴했습니다. message.

설명

SQL 표현식에 구문 오류가 있습니다. 이 오류는 SQL 표현식에서 태그가 닫히지 않은 컬럼, 속성 또는 치수를 참조할 때도 표시됩니다. 컬럼, 속성 또는 치수 참조는 각각 @Column(), @Attribute() 또는 @Measure() 태그 안에 닫혀 있어야 합니다.

사용자 조치

구문 오류를 수정하십시오. 각 컬럼, 속성 및 치수가 해당 태그 안에 닫혀 있는지 확인하십시오.

10306: 입력한 표현식의 데이터 유형이 숫자가 아닙니다. 두 번째 매개변수로 숫자 표현식을 입력하십시오.

설명

두 번째 매개변수의 데이터 유형이 숫자여야 합니다.

사용자 조치

입력한 표현식의 데이터 유형이 숫자 데이터 유형으로 나타나는지 확인하십시오.

10307: measure_name 치수의 표현식이 숫자가 아닌 데이터 유형으로 나타납니다. 표현식이 숫자 데이터 유형으로 나타나는 치수를 선택하십시오.

설명

두 번째 매개변수의 데이터 유형이 숫자여야 합니다.

사용자 조치

선택된 치수 표현식의 데이터 유형이 숫자인지 확인하십시오.

10308: OLAP 센터가 지정된 데이터베이스와 통신할 수 없습니다. 이는 데이터베이스가 **DB2 Cube Views**에 제대로 구성되지 않았기 때문일 수 있습니다. 데이터베이스 구성에 다소 시간이 걸릴 수 있습니다. 지정된 데이터베이스를 구성하려면 예를 누르십시오. 지정된 데이터베이스를 지금 구성하지 않으려면 아니오를 누르십시오.

설명

OLAP 센터는 제공된 사용자 이름 및 암호를 사용하여 데이터베이스에 연결할 수 있지만 스토어드 프로시저 API와는 통신할 수 없습니다.

가능한 원인은 다음과 같습니다.

- DB2 Cube Views 스토어드 프로시저 API가 지정된 데이터베이스에 등록되지 않습니다.
- DB2 Cube Views 카탈로그 테이블이 지정된 데이터베이스에 없습니다.

사용자 조치

DB2 Cube Views에 데이터베이스를 구성하려면 예를 누르고 그렇지 않으면 아니오를 누르십시오.

10309: 데이터베이스가 이전 DB2 Cube Views 버전에서 구성되어 있으므로 OLAP 센터가 지정된 데이터베이스에 연결할 수 없습니다. 데이터베이스를 **DB2 Cube Views**의 현재 버전으로 이주해야 합니다. OLAP 센터가 지정된 데이터베이스를 이주하도록 하려면 예를 누르십시오. OLAP 센터가 지정된 데이터베이스를 이주하지 않도록 하려면 아니오를 누르십시오.

설명

OLAP 센터가 제공된 사용자 이름 및 암호를 사용하여 데이터베이스에 연결할 수 있지만 카탈로그에서 메타데이터를 검색할 수 없습니다. 이것은 DB2 Cube Views 카탈로그 테이블이 이전 DB2 Cube Views 버전에서 구성되었기 때문일 수 있습니다.

사용자 조치

적절한 버전으로 DB2 Cube Views 카탈로그를 이주하려면 예를 누르고 그렇지 않으면 아니오를 누르십시오.

10310: database_name 데이터베이스가 제대로 구성되었습니다.

설명

OLAP 센터가 DB2 Cube Views 카탈로그 테이블을 제대로 작성했고 지정된 데이터베이스에 스토어드 프로시저 API를 등록했습니다.

사용자 조치

조치는 필요하지 않습니다.

10311: OLAP 센터가 DB2 Cube Views에 데이터베이스를 구성할 수 없습니다. 데이터베이스에서 다음 정보를 리턴했습니다. information.

설명

OLAP 센터가 DB2 Cube Views에 지정된 데이터베이스를 구성할 수 없습니다.

가능한 원인은 다음과 같습니다.

- OLAP 센터가 DB2 Cube Views 스토어드 프로시저 API를 등록할 수 없습니다.
- OLAP 센터가 하나 이상의 DB2 Cube Views 카탈로그 테이블을 작성할 수 없습니다.

사용자 조치

OLAP 센터 온라인 도움말의 "권한 및 특권" 주제에 설명된 올바른 설정 및 설치 권한이 있는지 확인하십시오. 데이터베이스 구성 정보는 DB2 Cube Views Setup and User's Guide를 참조하십시오.

10312: 집계 유효성 확인에 실패했습니다. 하나 이상의 지정된 집계 함수가 소스 SQL 표현식과 호환되지 않습니다.

설명

하나 이상의 지정된 집계 함수가 소스 SQL 표현식과 호환되지 않습니다. 지정된 집계 함수가 데이터 유형이 소스 SQL 표현식 데이터 유형과 다른 매개변수를 예상한 것이 원인일 수 있습니다.

사용자 조치

집계 함수가 지정된 치수의 소스 데이터 유형에 유효한지 확인하십시오.

10313: 치수의 소스 표현식은 구문이 없음(None) 집계 설정인 경우에만 올바릅니다. 치수가 없음(None) 집계 설정을 사용해야 합니다.

설명

다음 경우 치수는 없음(None) 집계 설정을 예상합니다.

- SQL 표현식의 구문은 집계 함수의 참조된 치수에 해당 함수를 적용하지 않을 경우 올바르지 않지만 해당 함수를 적용할 경우에는 올바릅니다. 예를 들어 char + int 구문은 올바르지 않지만 COUNT(char) + SUM(int) 구문은 올바릅니다.
- SQL 표현식은 RANK(), DENSE_RANK() 및 ROW_NUMBER()와 같은 OLAP 함수를 사용합니다.

사용자 조치

치수에 없음(None) 집계 설정을 적용했는지 확인하십시오.

10401: 표현식에 컬럼 함수, 스칼라 fullselect 또는 서브쿼리가 포함될 수 없습니다.

설명

SQL 표현식에 컬럼 함수, 스칼라 fullselect 또는 서브쿼리가 포함될 수 없습니다.

사용자 조치

컬럼 함수의 사용을 정정하여 유효하지 않은 표현식을 제거하십시오.

10501: 스키마 이름은 prefix 접두부로 시작할 수 없습니다.

설명

스키마 이름이 'SYS' 및 'SESSION'으로 시작될 수 없습니다.

사용자 조치

다른 스키마 이름을 입력하십시오.

10502: 조인 등록 정보가 큐브 모델의 성능 최적화에 유효하지 않습니다. 이 문제점을 해결한 후 최적화 어드바이저 마법사를 다시 여십시오. 데이터베이스에서 message 정보를 리턴합니다.

설명

조인 등록 정보가 큐브 모델의 성능 최적화에 유효하지 않습니다.

사용자 조치

최적화 유효성 확인 규칙을 적용하여 조인 시 올바른 설정을 지정하십시오.

10503: 계층 구조에는 연관된 큐브 계층 구조가 있으므로 해당 계층 구조를 수정할 수 없습니다.

설명

계층 구조에 큐브 계층 구조가 있는 경우 계층 구조를 수정할 수 없습니다.

사용자 조치

계층 구조를 변경하기 전에 큐브 계층 구조가 수정할 계층 구조를 참조하지 않는지 확인하십시오. 수정해야 하는 다른 계층 구조를 작성할 수도 있습니다.

10504: 이 치수는 다중 매개변수 집계 함수를 사용하는 치수를 참조하므로 없음(None) 집계 설정을 사용해야 합니다.

설명

없음(None) 집계 설정을 사용하는 치수만 다중 매개변수 함수를 사용하는 치수를 참조할 수 있습니다. 집계 설정을 없음(None)에서 다른 함수로 변경할 수 없습니다.

사용자 조치

다음 조치 중 하나를 수행할 수 있습니다.

- 치수의 집계 설정을 변경하지 마십시오.
- 다중 매개변수 함수를 사용하지 않도록 지정된 치수를 변경하십시오.

10505: 없음(None) 이외의 집계 설정을 사용하는 measure_name 치수가 이 치수를 참조하므로 이 치수는 다중 매개변수 함수를 사용할 수 없습니다.

설명

없음(None) 집계 설정을 사용하는 치수만 다중 매개변수 함수를 사용하는 치수를 참조할 수 있습니다. 없음(None) 집계 설정을 사용하지 않는 치수에서 편집할 치수를 참조하므로 다중 매개변수 함수를 포함시키기 위해 편집할 치수의 집계 스크립트를 변경할 수 없습니다.

사용자 조치

다음 조치 중 하나를 수행할 수 있습니다.

- 치수의 집계 스크립트를 변경하지 마십시오.
- 편집할 치수를 참조하지 않도록 지정된 치수를 변경하십시오.

10506: 기존의 집계 설정이 지정된 SQL 표현식에 유효하지 않습니다. OLAP 센터가 집계 설정을 setting 값으로 재설정합니다.

설명

기존의 집계 설정이 새 SQL 표현식에 유효하지 않으므로 기본 집계 설정으로 재설정되었습니다. 가능한 원인은 다음과 같습니다.

- 소스 SQL 표현식의 데이터 유형이 변경되었습니다.
- 현재 집계 설정이 없음(None)으로 예상됩니다. 다음 경우 없음(None)이어야 합니다.

- SQL 표현식의 구문은 집계 함수의 참조된 치수에 해당 함수를 적용하지 않을 경우 올바르지 않지만 해당 함수를 적용할 경우에는 올바릅니다. 예를 들어 char + int 구문은 올바르지 않지만 COUNT(char) + SUM(int) 구문은 올바릅니다.
- SQL 표현식은 RANK(), DENSE_RANK() 및 ROW_NUMBER()와 같은 OLAP 함수를 사용합니다.

사용자 조치

조치는 필요하지 않습니다.

10507: 큐브 모델의 차원 하나 이상에 계층 구조가 없습니다. 이들 차원은 큐브에 포함시킬 때 사용할 수 없습니다.

설명

작성된 큐브 차원은 하나 이상의 계층 구조가 있는 차원에 기반해야 합니다. 계층 구조가 없는 하나 이상의 차원이 있는 큐브를 작성하거나 수정하려고 합니다. 이러한 차원은 큐브 차원을 정의하는 데 사용하는 선택사항 목록에서 생략됩니다.

사용자 조치

생략된 차원을 참조하지 않고 큐브를 작성하거나 수정하십시오. 또는 큐브 모델의 각 차원에 계층 구조가 있는지 확인하십시오.

10508: 계층 구조에 2개의 레벨이 선택되는 경우에만 반복 전개 옵션이 유효합니다.

설명

정확히 두 개의 레벨이 반복 전개에 존재해야 합니다.

사용자 조치

반복 전개에서 두 개의 레벨만 선택하거나 다른 전개 옵션을 선택하십시오.

10509: 최적화 슬라이스에 대해 모든 필수 등록 정보를 지정하지 않았습니다. 각 최적화 슬라이스에 대해 큐브 차원마다 쿼리 유형 및 옵션을 지정하십시오.

설명

최적화 슬라이스에 대해 모든 필수 등록 정보를 지정하지 않았습니다.

사용자 조치

각 최적화 슬라이스에 대해 큐브 차원마다 쿼리 유형 및 옵션을 지정하십시오.

10510: 레벨에 하나 이상의 레벨 키 속성을 선택하십시오.

설명

레벨에 레벨 키 속성이 지정되어 있지 않습니다.

사용자 조치

레벨에 하나 이상의 레벨 키 속성을 선택하십시오.

10511: 동일한 큐브에 MOLAP 추출 유형 슬라이스와 하이브리드 추출 유형 슬라이스를 함께 지정할 수 없습니다. 큐브에서 슬라이스 중 하나를 제거해야 합니다.

설명

동일한 큐브에 MOLAP 추출 유형 최적화 슬라이스와 하이브리드 추출 유형 최적화 슬라이스를 함께 지정할 수 없습니다. 큐브 당 하나의 추출 유형 최적화 슬라이스만 지정할 수 있습니다.

사용자 조치

큐브에서 MOLAP 추출 슬라이스 또는 하이브리드 추출 슬라이스를 제거하십시오.

10512: 큐브 당 최대 하나의 MOLAP 추출 유형 최적화 슬라이스만 허용됩니다.

설명

큐브 당 최대 하나의 MOLAP 추출 유형 최적화 슬라이스만 허용됩니다.

사용자 조치

둘 이상의 MOLAP 추출 유형 최적화 슬라이스가 있는지 확인하십시오.

10513: 큐브 당 최대 하나의 하이브리드 추출 최적화 슬라이스만 허용됩니다.

설명

큐브 당 최대 하나의 하이브리드 추출 최적화 슬라이스만 허용됩니다.

사용자 조치

둘 이상의 하이브리드 추출 유형 최적화 슬라이스가 있는지 확인하십시오.

10514: 하이브리드 추출 유형 최적화 슬라이스가 이미 큐브에 정의되어 있을 경우에만 드릴 쓰루 유형의 최적화 슬라이스를 작성할 수 있습니다.

설명

하이브리드 추출 최적화 슬라이스는 드릴 쓰루 유형의 최적화 슬라이스를 지정하는 데 필요합니다.

사용자 조치

하이브리드 추출 유형의 최적화 슬라이스를 작성하십시오. 그러면 드릴 쓰루 유형의 최적화 슬라이스를 작성할 수 있습니다.

10515: DB2 Cube Views가 기존 최적화 슬라이스 각각의 새 큐브 차원에 대한 최적화 레벨로 '임의'를 지정했습니다.

설명

큐브의 최적화 슬라이스를 정의한 후 큐브 차원을 큐브에 추가할 경우 DB2 Cube Views는 새 큐브 차원의 최적화 레벨로 '임의'를 지정하여 기존 최적화 슬라이스를 확장합니다.

사용자 조치

큐브 등록 정보 창에 있는 쿼리 유형 페이지의 '지정'을 눌러 슬라이스의 최적화 레벨을 수정할 수 있습니다.

10516: 큐브의 쿼리 유형에 '고급' 설정을 지정했으므로 최적화 슬라이스 창에서 하나 이상의 최적화 슬라이스를 지정해야 합니다.

설명

큐브의 쿼리 유형에 '고급' 설정을 선택했기 때문에 최적화 슬라이스 창에서 하나 이상의 최적화 슬라이스를 지정해야 합니다.

사용자 조치

지정을 눌러서 최적화 슬라이스 창에 하나 이상의 최적화 슬라이스를 작성하십시오.

10517: 차원 페이지를 종료할 때 큐브 및 큐브 차원에 대한 현재 변경사항이 저장됩니다. 데이터베이스의 변경사항을 변경하려면 예를 누르십시오. 추가 변경사항을 작성하거나 취소하기 위해 차원 페이지에 남아 있으려면 아니오를 누르십시오.

설명

큐브 등록 정보 창에서 큐브 차원을 수정하고 차원 페이지에서 나오면 큐브 차원에 대한 모든 변경사항이 데이터베이스에 저장되므로 취소를 눌러 수행을 취소할 수 없습니다.

사용자 조치

현재 변경사항을 저장하려면 예를 누르십시오. 트랜잭션을 취소하고 차원 페이지에 남아 있으려면 아니오를 누르십시오.

10518: `cube_name` 큐브에 '고급' 최적화 슬라이스를 이미 정의했으므로 큐브 계층 구조를 수정할 수 없습니다. 모든 '고급' 최적화 슬라이스를 삭제해야 큐브 계층 구조를 수정할 수 있습니다.

설명

큐브에 '고급' 최적화 슬라이스를 이미 정의했으므로 큐브 계층 구조를 수정할 수 없습니다.

사용자 조치

모든 '고급' 최적화 슬라이스를 삭제해야 큐브 계층 구조를 수정할 수 있습니다.

10519: `cube_name` 큐브에 '고급' 최적화 슬라이스를 이미 정의했으므로 큐브 차원 세트를 수정할 수 없습니다. 모든 '고급' 최적화 슬라이스를 삭제해야 큐브에 큐브 차원을 추가하거나 삭제할 수 있습니다.

설명

큐브에 '고급' 최적화 슬라이스를 이미 정의한 경우 큐브 차원 세트를 수정할 수 없습니다.

사용자 조치

모든 '고급' 최적화 슬라이스를 삭제해야 큐브에 큐브 차원을 추가하거나 삭제할 수 있습니다.

10520: 등록 정보를 변경하면 큐브의 기존 최적화 슬라이스도 수정됩니다. 최적화 슬라이스를 검토하여 변경사항을 보고 수정하거나 등록 정보 창에서 취소 버튼을 눌러 창을 닫고 변경사항을 취소할 수 있습니다.

설명

큐브 차원 세트 또는 큐브 계층 구조를 수정할 경우 해당 큐브의 기존 고급 최적화 슬라이스가 다음 방법 중 하나로 수정됩니다.

- 새 큐브 차원이 추가되면 기존의 모든 최적화 슬라이스가 임의 레벨에서 새 큐브 차원을 포함하도록 확장됩니다.
- 큐브 차원이 제거되면 해당 레벨이 기존의 모든 최적화 슬라이스에서 제거됩니다.
- 최적화 슬라이스에 정의되어 있는 레벨이 존재하지 않도록 큐브 계층 구조가 변경되면 누락된 계층 구조 레벨이 적용 가능한 모든 최적화 슬라이스에 있는 임의 레벨로 바뀝니다.
- 기존 큐브 계층 구조가 새 큐브 계층 구조로 바뀌면 해당 큐브 차원의 최적화 슬라이스에 있는 레벨이 임의로 설정됩니다.

사용자 조치

최적화 슬라이스 창을 열고 OLAP 센터의 수정 사항을 확인하십시오. 필요한 모든 항목을 변경하십시오.

DB2 및 DB2 Cube Views의 상태 메시지

스토어드 프로시저의 실행 여부에 상관없이 DB2 Cube Views 스토어드 프로시저를 호출하면 DB2가 SQLCODE 및 SQLSTATE를 호출 응용프로그램에 리턴합니다. DB2 Cube Views 스토어드 프로시저를 실행할 수 있는 경우 스토어드 프로시저가 호출 응용프로그램에 전송된 XML 데이터의 일부로 상태 메시지를 리턴합니다.

다음 테이블에서는 메타데이터 조작에서 리턴한 상태 메시지와 호출 시 DB2에서 스토어드 프로시저에 리턴한 SQLSTATE 사이의 관계를 표시합니다.

표 63. 메타데이터 조작 ID 대 SQLSTATE 코드

SQL 코드	SQL 상태	메타데이터 조작 상태 메시지 ID	메타데이터 조작 상태 메시지 유형	리턴된 메타데이터 조작 상태 메시지
0	0	0 2	정보용	아니오
0	0	1	정보용	예

표 63. 메타데이터 조작 ID 대 SQLSTATE 코드 (계속)

SQL 코드	SQL 상태	메타데이터 조작 상태 메시지 ID	메타데이터 조작 상태 메시지 유형	리턴된 메타데이터 조작 상태 메시지
0	0	599 6006 6299 7200 7201 7202	경고	아니오
462	01HQ1	0 - 7999 (기타 행에 나열된 ID 제외)	오류	아니오
443	38Q00	적용할 수 없음	적용할 수 없음	적용할 수 없음
443	38Q01	적용할 수 없음	적용할 수 없음	적용할 수 없음
443	38Q02	적용할 수 없음	적용할 수 없음	적용할 수 없음
443	38Q03	적용할 수 없음	적용할 수 없음	적용할 수 없음

주의사항

이 정보는 미국에서 제공되는 제품 및 서비스용으로 작성된 것입니다.

IBM은 다른 국가에서는 이 자료에 기술된 제품, 서비스 또는 기능을 제공하지 않을 수도 있습니다. 현재 사용할 수 있는 제품 및 서비스에 대한 정보는 한국 IBM 담당자에게 문의하십시오. 이 책에서 IBM 제품, 프로그램 또는 서비스를 언급하는 것이 해당 IBM 제품, 프로그램 또는 서비스만을 사용할 수 있다는 것을 의미하지는 않습니다. IBM의 지적 재산을 침해하지 않는 한, 기능상으로 동등한 제품, 프로그램 또는 서비스를 대신 사용할 수 있습니다. 그러나 비IBM 제품, 프로그램 또는 서비스의 운용에 대한 평가 및 검증은 사용자의 책임입니다.

IBM은 이 책에서 다루고 있는 특정 내용에 대해 특허를 보유하고 있거나 현재 특허 출원 중일 수 있습니다. 이 책을 제공한다고 해서 특허에 대한 라이선스까지 부여하는 것은 아닙니다. 라이선스에 대한 의문사항은 다음으로 문의하십시오.

135-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩

한국 아이.비.엠 주식회사

고객만족센터

전화번호: 080-023-8080

2바이트(DBCS) 정보에 관한 라이선스 문의는 한국 IBM 고객만족센터에 문의하거나 다음 주소로 서면 문의하시기 바랍니다

IBM World Trade Asia Corporation

Licensing

2-31 Roppongi 3-chome, Minato-ku

Tokyo 106, Japan

다음 단락은 현지법과 상충하는 영국이나 기타 국가에서는 적용되지 않습니다. IBM은 타인의 권리 비침해, 상품성 및 특정 목적에의 적합성에 대한 묵시적 보증을 포함하여 (단, 이에 한하지 않음) 묵시적이든 명시적이든 어떠한 종류의 보증없이 이 책을 『현상 태대로』 제공합니다. 일부 국가에서는 특정 거래에서 명시적 또는 묵시적 보증의 면책 사항을 허용하지 않으므로, 이 사항이 적용되지 않을 수도 있습니다.

이 정보에는 기술적으로 부정확한 내용이나 인쇄상의 오류가 있을 수 있습니다. 이 정보는 주기적으로 변경되며, 변경된 사항은 최신판에 통합됩니다. IBM은 이 책에서 설명한 제품 및(또는) 프로그램을 사전 통지없이 언제든지 개선 및(또는) 변경할 수 있습니다.

IBM은 귀하의 권리를 침해하지 않는 범위 내에서 적절하다고 생각하는 방식으로 귀하가 제공한 정보를 사용하거나 배포할 수 있습니다.

- (i) 독립적으로 작성된 프로그램과 기타 프로그램(본 프로그램 포함) 간의 정보 교환 및
- (ii) 교환된 정보의 상호 이용을 목적으로 정보를 원하는 프로그램 라이선스 사용자는 다음 주소로 문의하십시오.

135-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩

한국 아이.비.엠 주식회사

이러한 정보는 해당 조건(예를 들어, 사용료 지불 등)에 따라 사용할 수 있습니다.

이 정보에 기술된 라이선스가 부여된 프로그램 및 사용 가능한 모든 라이선스가 있는 자료는 IBM이 IBM 기본 계약, IBM 프로그램 라이선스 계약(IPLA) 또는 이와 동등한 계약에 따라 제공된 것입니다.

비IBM 제품에 관한 정보는 해당 제품의 공급업체, 공개 자료 또는 다른 기타 범용 소스로부터 얻은 것입니다. IBM에서는 이러한 비IBM 제품을 테스트하지 않았으므로, 이들 제품과 관련된 성능의 정확성, 호환성 또는 기타 주장에 대해서는 확신할 수 없습니다. 비IBM 제품의 성능에 대한 의문사항은 해당 제품의 공급업체에 문의하십시오.

이 정보에는 일상의 비즈니스 운영에서 사용되는 자료 및 보고서에 대한 예제가 들어 있습니다. 이 예제에는 가능한 완벽하게 개념을 설명하기 위해 개인, 회사, 상표 및 제품의 이름이 사용될 수 있습니다. 이들 이름은 모두 가공의 것이며 실제 기업의 이름 및 주소와 유사하더라도 이는 전적으로 우연입니다.

본 문서에 포함된 모든 성능 데이터는 제한된 환경에서 산출된 것입니다. 따라서 다른 운영 환경에서 얻어진 결과는 상당히 다를 수 있습니다. 일부 성능은 개발 레벨 상태의 시스템에서 측정되었을 수 있으므로 이러한 측정치가 일반적으로 사용되고 있는 시스템에서도 동일하게 나타날 것이라고 보증할 수 없습니다. 또한, 일부 성능은 추정을 통해 추측되었을 수도 있으므로 실제 결과는 다를 수 있습니다. 이 책의 사용자는 해당 데이터를 사용자의 특정 환경에서 검증해야 합니다.

비IBM 제품에 관한 정보는 해당 제품의 공급업체, 공개 자료 또는 다른 기타 범용 소스로부터 얻은 것입니다. IBM에서는 이러한 비IBM 제품을 테스트하지 않았으므로, 이들 제품과 관련된 성능의 정확성, 호환성 또는 기타 주장에 대해서는 확신할 수 없습니다. 비IBM 제품의 성능에 대한 의문사항은 해당 제품의 공급업체에 문의하십시오.

이 정보를 소프트웨어로 보는 경우에는 사진과 컬러 삽화가 표시되지 않을 수도 있습니다.

상표

다음 용어는 미국 또는 기타 국가에서 IBM Corporation의 상표입니다.

AIX
DB2
DB2 Connect
DB2 Universal Database
IBM
Office Connect
Redbooks

다음은 타사의 상표 또는 등록상표입니다.

Microsoft, Windows, Windows NT, Windows 2000, Windows XP 및 Microsoft Excel은 Microsoft Corporation의 등록 상표입니다.

Java 또는 모든 Java 기반 상표와 로고 및 Solaris는 미국 또는 기타 국가에서 사용되는 Sun Microsystems, Inc.의 상표입니다.

UNIX은 미국 또는 기타 국가에서 사용되는 등록 상표이며 X/Open Company Limited의 독점으로 라이선스를 가지고 있습니다.

Linux는 Linus Torvalds의 등록상표입니다. Red Hat 및 모든 Red Hat 기반 상표 및 로고는 미국 및 기타 국가에서 사용되는 Red Hat, Inc.의 상표 또는 등록 상표입니다.

기타 회사, 제품 및 서비스 이름은 타사의 상표 또는 서비스표입니다.

용어

이 용어집에서는 본 책에 사용되는 용어를 정의합니다.

가

계산된 치수(calculated measure). SQL 또는 OLAP 센터의 표현식 빌더를 사용하여 작성한 내장 계산이 들어 있습니다. 계산된 치수에는 계산을 수행하지만 단일 컬럼 또는 속성으로 맵핑되지 않는 SQL 표현식이 들어 있습니다.

계층 구조(hierarchy). 큐브 모델의 차원 내에서 하나 이상의 속성 세트 간의 관계를 정의하는 DB2 오브젝트. DB2 Cube Views는 균형, 불균형, 불완전 및 네트워크와 같은 네 가지 유형의 계층 구조를 지원합니다. 계층 구조를 표준 또는 반복으로 전개할 수 있습니다.

구체화된 쿼리 테이블(materialized query table). 정의는 쿼리 결과에 기반하고 데이터는 구체화된 쿼리 테이블 정의가 기반한 하나 이상의 테이블에서 취한 미리 계산된 결과 양식인 테이블입니다.

균형 계층 구조(balanced hierarchy). 깊이가 일관적인 의미있는 레벨과 분기를 가진 계층 구조입니다. 각 속성의 논리적 상위는 해당 속성 바로 위의 레벨에 있습니다. 네트워크 계층 구조, 불완전 계층 구조 및 불균형 계층 구조를 참조하십시오.

기본 차원 테이블(primary dimension table). 눈송이 스키마에서 사실 테이블에 조인되는 차원 테이블입니다.

나

네트워크 계층 구조(network hierarchy). 레벨의 순서가 지정되어 있지는 않지만 레벨이 의미론적 의미를 갖는 계층 구조입니다. 속성 레벨에 고유한 상위-하위 관계가 없으므로 레벨의 순서는 중요하지 않습니다. 균형 계층 구조, 불완전 계층 구조 및 불균형 계층 구조를 참조하십시오.

눈송이 스키마(snowflake schema). 여러 테이블로 정의된 하나 이상의 차원과 같은 스타 스키마의 확장. 눈송이 스키마에서 기본 차원 테이블만 사실 테이블에 조인됩니다. 추가 차원 테이블은 기본 차원 테이블에 조인됩니다.

라

레벨(level). 계층 구조의 정렬에 하나의 논리적 단계로 함께 작동하며 서로 관련된 하나 이상의 속성으로 구성된 DB2 오브젝트입니다.

마

메타데이터(metadata). 컬럼의 데이터 유형(숫자, 텍스트 등) 또는 컬럼 길이와 같은 데이터 등록 정보에 관한 정보입니다. 또한 큐브 또는 차원과 같이 오브젝트의 설계를 지정하는 정보나 데이터 구조에 관한 정보일 수 있습니다.

바

반복 전개(recursive deployment). 계층 구조의 속성들 간에 고유한 상위-하위 관계를 사용합니다. 반복 전개를 사용하는 불균형 계층 구조는 상위-하위 속성 쌍으로 표현됩니다.

불균형 계층 구조(unbalanced hierarchy). 상위 및 하위 관계는 일관되지만 특정 레벨의 모든 구성원에서 의미가 일관되지 않는 레벨의 계층 구조입니다. 또한 계층 구조 분기의 깊이도 일관적이지 않습니다. 균형 계층 구조, 네트워크 계층 구조 및 불완전 계층 구조를 참조하십시오.

불완전 계층 구조(ragged hierarchy). 각 레벨은 일관적인 의미를 갖지만 분기 레벨의 최소 한 구성원 속성이 채워져 있지 않아서 분기의 깊이가 일관적이지 않은 계층 구조입니다. 균형 계층 구조, 네트워크 계층 구조 및 불균형 계층 구조를 참조하십시오.

사

사실 오브젝트(facts object). 특정 응용프로그램이 관심을 두고 있는 관련 치수를 묶는 DB2 오브젝트. 사실 오브젝트는 사실 차원 조인에서 사용되는 속성에 대한 정보 및 다중 데이터베이스 테이블에서 추가적인 치수들의 맵핑에 사용되는 속성 및 조인에 대한 정보를 저장합니다. 따라서 치수 세트 외에도 사실 오브젝트는 속성 세트 및 조인 세트를 저장합니다. 사실 오브젝트는 큐브 모델에서 스타 스키마의 중심으로 사용됩니다.

사실 테이블(fact table). 사실이 차원 테이블과 관련된 숫자 치수 및 키가 들어 있는 데이터 웨어하우스 스키마의 중심 테이블입니다. 사실 테이블에는 비즈니스의 특정 이벤트(예: 은행 거래 또는 제품 판매)를 설명하는 데이터가 들어 있습니다.

속성 관계(attribute relationship). 일반적인 속성 오브젝트의 관계를 설명합니다. 왼쪽 및 오른쪽 속성, 유형, 카디널리티(cardinality), 함수 종속성의 판별 여부에 따라 관계를 설명합니다. 유형은 왼쪽 속성과 관련하여 오른쪽 속성의 역할이 무엇인지를 설명합니다. 설명 및 연관의 두 가지 가능한 유형이 있습니다. 설명 유형은 오른쪽 속성이 왼쪽 속성의 디스크립터임을 지정합니다.

속성(attribute). 테이블의 단일 컬럼 또는 컬럼 세트나 기타 속성의 조합인 표현식에 맵핑되는 DB2 오브젝트입니다. 하나의 속성은 여러 역할을 수행할 수 있습니다. 예를 들어 큐브에 있는 데이터에 대한 참조이거나 조인 또는 기타 속성 관계에 사용되는 컬럼에 대한 참조일 수 있습니다.

스키마(schema). SQL-92 표준에서 단일 사용자가 소유하며 단일 이름 스페이스를 형성하는 데이터베이스 오브젝트 콜렉션입니다. 이름 스페이스는 중복 이름을 가질 수 없는 오브젝트 세트입니다. 예를 들어 두 개의 테이블이 개별 스키마에 있는 경우에만 이들 테이블의 이름이 동일할 수 있지만 동일한 스키마에 있는 경우 이들의 이름은 동일할 수 없습니다.

스타 스키마(star schema). 추가 차원 데이터가 차원 테이블에 저장된 스키마 중심에서 단일 사실 테이블에서 데이터가 유지보수되는 관계형 데이터베이스 구조입니다. 각 차원 테이블은 키 컬럼을 통해 사실 테이블에 직접 관련되고 보통 이 테이블에 조인되어 있습니다. 스타 스키마는 데이터 웨어하우스에서 사용됩니다.

스타 조인(star join). 사실 테이블(보통 대형 사실 테이블)과 둘 이상의 차원 테이블 간 조인입니다. 사실 테이블은 차원 키의 각 차원 테이블과 조인됩니다.

슬라이스(slice). 다차원 데이터베이스 또는 큐브의 영역입니다.

아

아웃트리거 테이블(outtrigger table). 차원의 기본 차원 테이블이 아닌 눈송이 스키마의 차원 테이블입니다.

요약 테이블(summary table). 큐브 모델에서 사용하는 기본 테이블의 집계 데이터가 들어 있습니다. DB2 Cube Views는 DB2 요약 테이블을 사용하여 큐브 모델에 발행된 쿼리 성능을 향상시킵니다. 요약 테이블은 특히 요약 데이터가 들어 있는 구체화

된 쿼리 테이블(MQT)의 특수 유형입니다. DB2 Cube Views는 데이터를 요약하는 MQT를 항상 권장하므로 DB2 Cube Views 문서에서 용어 요약 테이블을 사용하여 권장된 MQT를 설명합니다. 구체화된 쿼리 테이블을 참조하십시오.

이중 큐브(hybrid cube). 기본 테이블의 하위 레벨 데이터를 쿼리할 수 있도록 다중 차원 데이터가 들어 있으며 관련 데이터를 참조합니다.

자

제한조건(constraint). 데이터베이스 관리 프로그램이 강제 실행하는 규칙입니다. 고유, 참조, 테이블 점검 및 정보용과 같은 네 가지 유형의 제한조건이 있습니다.

조인(join). 두 개의 관계 테이블을 조인합니다. 그런 다음, 조인은 조인되는 테이블의 컬럼을 참조하는 속성을 참조합니다. 가장 간단한 조인 형식은 두 개의 속성(즉, 첫 번째 테이블의 컬럼에 맵핑되는 속성과 두 번째 테이블의 컬럼에 맵핑되는 속성)을 참조합니다. 조인에는 컬럼의 비교 방식을 나타내는 연산자도 포함됩니다. 조인 오브젝트는 첫 번째 테이블에서 둘 이상의 컬럼을 두 번째 테이블에 있는 동일한 수의 컬럼에 조인하는 복합 조인을 모델링하는 데에도 사용할 수 있습니다. 복합 조인은 속성 쌍을 사용하여 해당 컬럼을 맵핑합니다. 각 속성 쌍에는 컬럼 쌍을 비교하는 방법을 나타내는 연산자가 있습니다. 조인에는 유형과 카디널리티도 있습니다. 조인을 차원에 사용하면 차원 테이블을 함께 조인할 수 있으며 큐브 모델에 사용하면 큐브 모델의 차원을 사실 오브젝트로 조인할 수 있으며 사실 오브젝트 내에서 사용하면 여러 사실 테이블을 조인할 수 있습니다.

집계 함수(aggregation function). SUM, AVG, MIN 및 MAX와 같은 DB2 SQL 집계 함수 중의 하나입니다. 집계 함수는 치수에서 롤업을 수행하는 방법을 제어하는 데 사용됩니다.

차

차원 테이블(dimension table). 항목이 사실 테이블의 데이터를 설명하는 데이터 웨어하우스의 테이블입니다. 차원 테이블에는 차원이 작성되는 데이터가 들어 있습니다.

차원(dimension). 치수 세트의 일부 측면을 설명하는 관련 속성의 콜렉션을 참조하는 DB2 오브젝트입니다. 차원은 하나 이상의 차원 테이블에서 속성을 참조할 수 있습니다. 하지만 다중 차원 테이블의 속성을 사용하는 경우 테이블에 속성 간 조인이 있어야 하며 차원에서 해당 조인을 참조해야 합니다. 또한 한 차원이 계층 구조를 하나 이상 참조하고 해당 속성 간 관계를 참조할 수 있습니다.

치수(measure). 치수 엔티티를 정의하며 사실 오브젝트에서 사용되는 DB2 오브젝트. 치수는 차원의 컨텍스트 내에서 의미를 갖습니다. 치수의 일반적인 예는 세입, 비용 및 이익입니다.

카

큐브 계층 구조(cube hierarchy). 큐브 차원의 일부이고 큐브 차원에 대응하는 차원의 계층 구조에서 파생된 DB2 오브젝트입니다. 큐브 계층 구조는 파생된 계층 구조의 속성 서브세트를 참조하며, 속성의 순서가 계층 구조와 동일한 순서여야 합니다.

큐브 레벨(cube level). 레벨의 서브세트인 DB2 오브젝트이며 큐브에서 사용됩니다. 큐브 레벨은 자신이 파생된 레벨(상위 레벨)을 참조하고 상위 레벨에 정의된 레벨 키 속성 및 디폴트 속성을 상속합니다.

큐브 모델(cube model). 치수 컬렉션과 관련된 모든 데이터를 설명하는 DB2 오브젝트입니다. 보통 큐브 모델은 데이터베이스의 스타 스키마 또는 눈송이 스키마와 관련됩니다. 큐브 모델은 단일 사실 오브젝트 및 하나 이상의 차원을 참조합니다. 큐브 모델을 최적화하여 큐브 모델의 스타 스키마 또는 눈송이 스키마에 발행된 SQL 쿼리의 성능을 향상시킬 수 있습니다.

큐브 사실(cube facts). 큐브의 일부이고 큐브에 대응하는 큐브 모델의 차원에서 파생된 DB2 오브젝트입니다. 큐브 사실은 파생된 사실 오브젝트에서 치수의 서브세트를 참조합니다.

큐브 차원(cube dimension). 큐브의 일부이고 큐브에 대응하는 큐브 모델의 차원에서 파생된 DB2 오브젝트입니다. 큐브 은 파생된 차원의 속성 서브세트를 참조합니다. 단일 큐브 계층 구조를 참조할 수도 있습니다.

큐브(cube). 큐브 모델에서 파생된 DB2 오브젝트입니다. 큐브 사실 및 큐브 차원은 큐브 모델의 해당 내용에 대한 서브세트입니다. 모델. 큐브 차원은 큐브 차원 당 하나의 큐브 계층 구조만을 허용하기 때문에 큐브는 여러 계층 구조를 사용하지 않는 도구와 응용프로그램에 적합합니다.

과

표준 전개(standard deployment). 계층 구조의 레벨 정의를 사용합니다. 여기서 계층 구조의 각 속성은 하나의 레벨을 정의합니다. 예를 들어, 시간 차원에 대한 균형 계층 구조는 대개 년, 분기 및 월과 같은 각각의 정의된 레벨에 의해 구성될 수 있습니다. 표준 전개는 네 개의 모든 계층 구조 유형과 함께 사용 가능합니다.

하

함수 종속성(functional dependency). 레벨 오브젝트의 디폴트 속성 및 관련 속성을 해당 레벨의 키 속성이 기능적으로 판별함을 표시하는 DB2 오브젝트입니다. 모든 컬럼이 동일한 테이블에 있는 경우 함수 종속성에 따라 기능적으로 하나 이상의 컬럼이 하나 이상의 다른 컬럼에 종속되도록 지정할 수 있습니다.

M

MQT. 구체화된 쿼리 테이블을 참조하십시오.

색인

[가]

간단한 치수 58
개요
 메타데이터 오브젝트 13
검색 결과 151
검색 조작 166
계산
 값 68
 이익 79
 이익 여백 79
 플로우 68
계산된 치수 58
계층 구조
 관계형 맵핑 14, 20
 기본 규칙 43
 등록 정보 29
 레벨 작성 60
 설명 26
 유형 26
 작성 59
 전개 28
 최적화 모델링 105
계층 구조 마법사 59
관계형 테이블 14
관련 속성
 설명 30
 큐브 계층 구조에 추가 63
관리 조작 167
구성 파일 175, 179
구체화된 쿼리 테이블 92
권한 및 특권 50
규칙
 기본 118
 최적화 118
 큐브 모델 완전성 118
균형 계층 구조 26
기본 규칙 118
 계층 구조 43
 레벨 43
 사실 오브젝트 43
 속성 44
 속성 관계 45
 조인 45

기본 규칙 (계속)
 차원 43
 치수 44
 큐브 45
 큐브 계층 구조 47
 큐브 레벨 46
 큐브 모델 42
 큐브 사실 오브젝트 46
 큐브 차원 46

[나]

네트워크 계층 구조 28

[다]

데이터 소스
 리모트 138, 141
데이터베이스
 리모트 138
데이터베이스 연결 창 53
드릴 다운 쿼리 106
등록 정보
 계층 구조 29
 레벨 33
 사실 오브젝트 25
 속성 36
 속성 관계 38
 일반 22
 조인 38
 차원 26
 치수 35
 큐브 39
 큐브 계층 구조 41
 큐브 레벨 41
 큐브 모델 24
 큐브 사실 40
 등록 정보 40
 큐브 차원 40
디폴트 속성
 설명 30

[라]

런타임 추적 174
레벨
 관계형 맵핑 14, 17
 관련 속성 30
 기본 규칙 43
 등록 정보 33
 디폴트 속성 30
 레벨 키 속성 30
 설명 30
 일반적인 모델링 32
 작성 60
 최적의 모델링 31
 최적화 모델링 105
 큐브 계층 구조에 추가 63
레벨 마법사 60
레벨 키 속성
 설명 30
로그
 추적 174
리모트
 데이터 소스 138
리모트 데이터 소스
 정의 141

[마]

메모리 관리 146
메타데이터 브릿지 4
메타데이터 오브젝트 4
 개요 13
 계층 구조 26
 계층 구조, 작성 59
 교환 53
 기본 규칙 42
 레벨 30
 레벨, 작성 60
 사실 오브젝트 25
 사실 오브젝트, 작성 57
 속성 35
 속성 관계 36
 이름 지정 규칙 23
 익스포트 54

메타데이터 오브젝트 (계속)

- 일반 등록 정보 22
- 임포트 54
- 조인 38
- 조인, 작성 62
- 차원 25
- 차원, 작성 58
- 차원, 추가 61
- 최적화 모델링 104
- 치수 33
- 큐브 39
- 큐브 계층 구조 41
- 큐브 레벨 41
- 큐브 모델 24
- 큐브 모델, 빠른 시작 마법사로 작성 55
- 큐브 모델, 작성 57
- 큐브 사실 오브젝트 40
- 큐브 차원 40
- 큐브, 작성 62
- 형식 180

메타데이터 오브젝트 삭제 65

메타데이터 조작

- 변경 159
- 변환 164
- 삭제 160
- 설명 151
- 유효성 확인 163
- 이름 바꾸기 160
- 임포트 161
- 작성 158

모델링

- 일반적인 32
- 최적의 31

[바]

- 반복 전개 29
- 변경 조작 159
- 변환 조작 164
- 별칭 142
- 보고서 쿼리 106
- 불균형 계층 구조 26
- 불완전 계층 구조 27
- 브릿지 4
- 빠른 시작 마법사 55

[사]

- 사실 대 차원 조인 118
- 사실 마법사 57
- 사실 오브젝트
 - 관계형 맵핑 14
 - 기본 규칙 43
 - 등록 정보 25
 - 설명 25
 - 작성 57
- 사실-차원 조인 61
 - 작성 61
- 사실-차원 조인, 작성 58
- 삭제 조작 160
- 상관
 - 광고 74
 - 판매액 74
- 샘플
 - 데이터베이스 파일 185
 - 응용프로그램 파일 186
- API 매개변수
 - 검색 조작 166
 - 관리 조작 167
 - 수정 조작 166
- 서버
 - 페더레이티드 138
- 설명 조작 151
- 설치
 - AIX 7
 - Linux 7
 - Solaris 운영 체제 7
 - Windows 7
- 설치 요구사항 5
- 속성
 - 관계형 맵핑 14, 17
 - 관련 30
 - 기본 규칙 44
 - 등록 정보 36
 - 디폴트 30
 - 레벨 키 30
 - 설명 35
 - 요약 테이블 97
- 속성 관계
 - 기본 규칙 45
 - 등록 정보 38
 - 설명 36
 - 요약 테이블 97

속성 관계 (계속)

- 유형
 - 설명 36
 - 연관 36
- 수정 조작 166
- 스키마
 - 스타 14
 - snowflake 14, 121
 - 스타 스키마 14
- 스토어드 프로시저
 - md_message 148
- 슬라이스 94
 - 최적화 107
- 시스템 구성 146
- 시스템 요구사항 5

[아]

- 아우트리거 테이블 121
- 오류 로그 175
- 오류 조절 147
- 완전성 규칙
 - 큐브 모델 42
- 요구사항
 - 소프트웨어 5
 - 하드웨어 5
- 요약 테이블 92, 101, 126
 - 삭제 104, 134
 - 유지보수 103, 133
 - 작성 103
- 제한조건 98
- 즉시 새로 고침 133
- 지연 새로 고침 132, 134
- 함수 종속성 98
 - SQL 스크립트 127
- 유효성 확인 모드 171
- 유효성 확인 조작 163
- 이름 바꾸기 조작 160
- 익스포트
 - 메타데이터 오브젝트 54
- 익스포트 창 54
- 일반적인 모델링 32
- 임포트
 - 메타데이터 오브젝트 54
- 임포트 마법사 54
- 임포트 조작 161
- 임포트용 모드 169

[자]

작성 조작 158
 전제조건 6
 정보용 제한조건 92, 118, 126
 제한조건 92, 126, 142
 외부 키 118
 요약 테이블 98
 정보용 118
 조인
 관계형 맵핑 15, 17
 기본 규칙 45
 등록 정보 38
 사실 대 차원 118
 설명 38
 유형 119
 작성 62
 최적화 규칙 45
 카다널리티 119
 조인 마법사 62
 조작 매개변수 168
 유효성 확인 모드 171
 임포트용 모드 169
 objectType 168
 recurse 168
 restriction 169
 조작 피연산자 171
 즉시 새로 고침 요약 테이블 133
 지연 새로 고침 요약 테이블 134
 만기 132

[차]

차원
 계층 구조, 작성 59
 관계형 맵핑 14, 17, 20
 기본 규칙 43
 등록 정보 26
 설명 25
 작성 58
 최적화 규칙 43
 추가 61
 큐브 모델에서 제거 65
 차원 마법사 58
 차원 추가 마법사 61
 최적의 모델링 31
 최적화 98
 규칙 118

최적화 (계속)

데이터 샘플링 126
 디스크 스페이스 제한사항 124
 벤치마크 130
 성능 벤치마킹 102
 슬라이스 124
 시간 제한사항 125
 요약 테이블 사용 92
 쿼리 유형 124
 큐브 모델 126
 프로세스 101
 SQL 스크립트 127
 최적화 규칙
 조인 45
 차원 43
 큐브 모델 42
 최적화 슬라이스 105, 107, 116
 드릴 다운 108
 드릴 쓰루 114
 레벨 107
 모두 107
 보고서 109
 임의 107
 지정 63
 하이브리드 추출 113
 MOLAP 추출 111
 최적화 슬라이스, 지정 63
 최적화 어드바이저 마법사 92, 103, 104,
 124, 126, 133
 추적 174
 로그 174

치수

간단한 58
 계산된 58
 관계형 맵핑 14
 기본 규칙 44
 등록 정보 35
 설명 33
 요약 테이블 97
 최적화 모델링 105

[카]

카다널리티 37
 카운팅 82
 코드 페이지 176
 쿼리
 다시 라우팅 92, 95, 98

쿼리 (계속)

드릴 다운 106, 108
 드릴 쓰루 114
 벤치마크 130
 보고서 106, 109
 유형 105
 캡처 131
 하이브리드 추출 113
 DB2EXPLAIN 131
 MOLAP 추출 106, 111
 쿼리 다시 라우팅 92, 98
 큐브 63
 관계형 맵핑 21
 기본 규칙 45
 등록 정보 39
 설명 39
 작성 62
 최적화 모델링 105
 큐브 계층 구조
 관계형 맵핑 21
 기본 규칙 47
 등록 정보 41
 설명 41
 지정 63
 큐브 계층 구조
 지정 63
 큐브 차원에 추가 63
 큐브 레벨
 기본 규칙 46
 등록 정보 41
 설명 41
 큐브 마법사 62
 큐브 모델
 계층 구조, 작성 59
 관계형 맵핑 14
 기본 규칙 42
 등록 정보 24
 빠른 시작 마법사로 작성 55
 사실 오브젝트, 작성 57
 설명 24
 완전성 규칙 42, 118
 작성 56, 57
 차원 제거 65
 차원, 작성 58
 차원, 추가 61
 최적화 126
 최적화 규칙 42
 큐브 모델 마법사 57

- 큐브 사실 오브젝트
 - 관계형 맵핑 21
 - 기본 규칙 46
 - 설명 40
- 큐브 차원
 - 관계형 맵핑 21
 - 기본 규칙 46
 - 등록 정보 40
 - 설명 40
 - 큐브 계층 구조 추가 63

[타]

트랜잭션 146

[과]

- 판매액 수치 순위
 - DENSERANK 84
 - RANK 84
 - ROWNUMBER 84
- 페더레이티드
 - 데이터베이스 138
 - 서버 138
- 페더레이티드 시스템 139
 - 개요 137
 - 데이터 소스 137
 - 리모트 데이터 소스 141, 142
 - 별칭 142
 - 제한조건 142
 - 클라이언트 137
 - 페더레이티드 데이터베이스 137
 - 페더레이티드 서버 137
 - DB2 Cube Views에서 사용 가능 140
- 표준 전개 28
- 피연산자 171

[하]

- 함수 종속성
 - 요약 테이블 98
 - 지정 60

A

- API(application programming interface)
 - 개요 145
 - 데이터 교환 145

- API(application programming interface) (계속)
 - 매개변수 150
 - 스토어드 프로시저 148
 - 추적 174
- API의 메시지 구조 172

C

- currentRef 피연산자 171
- CVSAMPLE 데이터베이스 185, 186

D

- DB2 Cube Views를 사용 139
- DB2 EXPLAIN 가능 95
- DB2 SQL 스냅샷 모니터 131
- DB2 데이터베이스
 - 메타데이터 오브젝트 삭제 65
 - 연결 53
 - 연결 끊기 53
- DB2 옵티마이저 92, 98
- db2batch 벤치마크 도구 130

M

- md_message
 - 스토어드 프로시저 148
- MOLAP 추출 쿼리 106

N

- newRef피연산자 171

O

- object 피연산자 171
- objectType 매개변수 168

R

- recurse 매개변수 168
- restriction 매개변수 169

S

- snowflake 스키마 14, 17, 121
- SQL 스크립트 126, 127

T

- Time 차원 88

X

- XML 구문 분석 147

IBM에 문의

기술 문제점이 발생하면 DB2 Cube Views 고객만족센터에 문의하기 전에 제품 문서에서 제공하는 조치를 검토하고 수행하십시오. 이 안내서에서는 DB2 Cube Views 고객만족센터가 더 나은 서비스를 제공하는 데 도움이 되도록 사용자가 수집할 수 있는 정보를 제공합니다.

정보를 얻거나 DB2 Cube Views 제품을 주문하려면 해당 지방 사무소의 IBM 담당자에게 문의하거나 권한이 부여된 IBM 소프트웨어 대리점에 문의하십시오.

미국에 거주하는 사용자는 다음 번호 중 하나를 호출할 수 있습니다.

- 1-800-237-5511(고객 지원용)
- 1-888-426-4343(사용 가능한 서비스 옵션에 대한 문의)

제품 정보

미국에 거주하는 사용자는 다음 번호 중 하나를 호출할 수 있습니다.

- 1-800-IBM-CALL(1-800-426-2255) 또는 1-800-3IBM-OS2(1-800-342-6672)(제품 주문 및 일반 정보 문의).
- 1-800-879-2755(책 주문).

<http://www.ibm.com/software/data/db2/db2md/>

DB2 Cube Views에 관한 정보 링크를 제공합니다.

<http://www.ibm.com/software/data/db2/udb>

DB2 Universal Database 웹 페이지는 뉴스, 제품 설명, 교육 스케줄 등에 관한 현재 정보를 제공합니다.

<http://www.elink.ibm.com/>

책 주문 방법에 대한 정보를 제공하는 국제 책 주문 웹 사이트를 열려면 Publications를 누르십시오.

<http://www.ibm.com/education/certify/>

IBM 웹 사이트의 Professional Certification Program에서는 다양한 IBM 제품에 관한 인증 테스트 정보를 제공합니다.

주: 일부 국가에서 IBM이 권한을 부여한 판매자는 IBM 고객만족센터 대신 해당 판매자지원센터에 문의해야 합니다.

문서에 관한 주석

사용자 피드백은 IBM이 양질의 정보를 제공하는 데 도움이 됩니다. 이 책 또는 기타 DB2 Cube Views 문서에 관한 주석을 보내주십시오. 다음 방법 중 하나로 주석을 제공할 수 있습니다.

- www.ibm.com/software/data/rcf의 온라인 독자 주석 양식을 사용하여 주석을 보내 주십시오.
- 전자 우편을 통해 ibmkspoe@kr.ibm.com으로 주석을 보내주십시오. 보내실 때에는 제품 이름, 제품의 버전 번호 및 책의 이름과 부품 번호(가능한 경우)를 써 주십시오. 특정 텍스트에 주석을 달 경우 텍스트의 위치(예: 제목, 테이블 번호 또는 페이지 번호)를 써 주십시오.



프로그램 번호: 5724-E15

SA30-1910-01

