

IBM DB2 Cube Views



Veiledning og referanse

Versjon 8.2

IBM DB2 Cube Views



Veiledning og referanse

Versjon 8.2

Merk:

Før du bruker opplysningene i denne boken og det produktet det blir henvist til, må du lese "Merknader" på side 289.

Andre utgave (september 2004)

Dette dokumentet inneholder informasjon som eies av IBM. Det leveres i henhold til lisensbetingelser og er opphavsrettslig beskyttet. Informasjonen i denne håndboken omfatter ingen produktgarantier, og eventuelle merknader i denne håndboken må ikke tolkes som garantier.

Du kan bestille IBM-publikasjoner elektronisk eller via IBM-representanten.

- Hvis du vil bestille publikasjoner elektronisk, går du til IBM Publications Center på www.ibm.com/shop/publications/order
- IBM-representanten finner du ved å gå til IBM Directory of Worldwide Contacts på www.ibm.com/planetwide

Hvis du vil bestille DB2-publikasjoner fra DB2 Marketing and Sales i USA eller Canada, må du ringe 1800-IBM-4YOU (425-4968).

Når du sender informasjon til IBM, gir du IBM en ikke-eksklusiv rett til å bruke eller distribuere informasjonen på den måten IBM mener er best, uten forpliktelser i noen retning.

© Copyright International Business Machines Corporation 2004. All rights reserved.

Innhold

Om denne boken	v
Hvem boken er beregnet på	v
Syntaks i denne boken	v
Informasjon på nettet	v

Kapittel 1. Installere, migrere og konfigurere DB2 Cube Views. 1

DB2 Cube Views-forbedringer	1
Introduksjon til DB2 Cube Views	3
Installeringskrav	4
Installere DB2 Cube Views.	6
Migrere XML-filer fra DB2 Cube Views fra versjon 8.1 til versjon 8.2	6
Konfigurere en database for DB2 Cube Views	7
Konfigurere CVSAMPLE-eksempeldatabasen	8
Problemsøking for en databasetilkobling	8

Kapittel 2. Om DB2 Cube Views-metadatoobjekter 11

Metadatoobjekter som er tilordnet relasjonstabeller Eksempel på en kubemodell som er tilordnet relasjonstabeller	11
Generelle egenskaper for metadatoobjekter	14
Kubemodeller	19
Faktoobjekter	21
Dimensjoner	21
Hierarkier	22
Nivåer	23
Mål	27
Attributter	30
Attributtforhold	32
Kombineringer	33
Kuber	34
Kubefaktoobjekt	35
Kubedimensjoner	36
Kubehierarkier	36
Kubenivåer	37
Regler for metadatoobjekter	38

Kapittel 3. Utforme DB2 Cube Views-metadatoobjektmodeller 43

Starte og oppdatere OLAP-senteret	43
Kompatibilitet mellom ulike OLAP-senter- og API-versjoner.	43
Autorisasjoner og rettigheter for bruk av DB2 Cube Views	44
Opprette DB2 Cube Views-metadatoobjekter	46
Utvexle metadato mellom DB2 Cube Views og OLAP-verktøy	46
Opprette en kubemodell ved hjelp av hurtigstartveiviseren	48
Opprette en fullstendig kubemodell	49
Tilføye en eksisterende dimensjon til en kubemodell	52

Opprette en kombinerings	53
Opprette en kube	54
Fjerne en dimensjon fra en kubemodell	56
Slette et metadatoobjekt fra en database	56

Kapittel 4. DB2 Cube Views, scenarier for forretningsmodellering 59

Beregne flyt og verdi av lagerbeholdning i et varelager over tid	59
Korrelere annonseringskostnader og salg	64
Beregne fortjenesten og fortjenestemarginen for en butikk	69
Telle antallet Internett-bestillinger	72
Rangere salgstall.	74
Bruke tidsdata lagret i faktatabellen for å opprette en tidsdimensjon	77

Kapittel 5. DB2 Cube Views, optimalisering av kubemodeller 81

Sammendragstabeller	82
Sammendragstabeller med funksjonelle avhengigheter og begrensninger	88
Oversikt over optimaliseringsprosessen	90
Utforming av metadato og optimalisering	93
Optimaliseringssektorer for kuber	95
Analysere spørringer for mulige optimaliseringssektorer	104
Begrensningsdefinisjoner for optimalisering	107
Parametere for optimaliseringsrådgiveren	112
Optimalisere en kubemodell	114
Eksempel på et SQL-skript for oppretting av sammendragstabellene	115
Teste resultater fra spørring.	118
Problemløsning for sammendragstabeller	119
Vedlikehold av sammendragstabeller	121
Slette en sammendragstabell	122

Kapittel 6. DB2 Cube Views og forente datakilder 123

Oversikt over forente systemer	123
Oversikt over optimalisering av fjerndatakilder med DB2 Cube Views	124
Aktivere et forent system for DB2 Cube Views	126
Definere fjerntliggende datakilder	126
Definere kallenavn for fjerntliggende tabeller for DB2 Cube Views	127
Definere informasjonsbegrensninger for kallenavn for DB2 Cube Views	127
Problemløsning for spørringsytelse ved fjerntliggende datakilder.	128

Kapittel 7. DB2 Cube Views-API 129

DB2 Cube Views, API-oversikt	129
--	-----

DB2 Cube Views API: Lagrede DB2-prosedyrer og XML-analyse	130
Lagret prosedyre for DB2 Cube Views	131
Konfigurasjonsfil for DB2 Cube Views	133
Konfigurasjonsfiler	134
DB2 Cube Views-metadatatabeller og XML-skjemafiler	134
Metadataobjektformat	135
Sekvens for operasjonstrinnene	136
DB2 Cube Views, API-parametere	136
Parametere for DB2 Cube Views API-metadataoperasjoner	136
Inndata- og utdataparametere	136
Eksempler på inndata- og utdataparametere i metadataoperasjoner	137
Metadataoperasjoner	139
Parametere for operasjoner	139
Operander i operasjoner	142
Henteoperasjon: Describe	143
Advise-operasjonen: Advise	144
Endringsoperasjoner: Alter, Create, Drop, Import og Rename	149
Administrasjonsoperasjoner: Validate og Translate	154
Meldinger, logging og sporing	156
Meldingers struktur	156
Kjøretidssporing for APIen for DB2 Cube Views	157
Loggfiler for DB2 Cube Views-APIen	158
Feillogging	158
Scenarier for logging og sporing	159
Støtte for kodesett	160

Kapittel 8. Eksempelfiler.	163
Oversikt over db2mdapiclient-funksjonen	163
Kommandoen db2mdapiclient: manipulere metadataobjekter	164
Eksempeldatabasefiler	165
API-eksempelfiler	166

Tillegg. Meldinger	171
SQLSTATE, API og andre tjenermeldinger	171
API SQL-statuser	171
Common	173
Eksterne API-feil	196
Optimalisering	242
OLAP-senter, meldinger	252
10000-10600	252
Statusmeldinger fra DB2 og DB2 Cube Views.	288

Merknader.	289
Varemerker	290

Ordliste	293
---------------------------	------------

Stikkordregister	297
-----------------------------------	------------

Kontakte IBM	301
Produktinformasjon	301
Kommentarer til dokumentasjonen	301

Om denne boken

Denne boken inneholder informasjon om disse emnene for DB2 Cube Views:

- Komme i gang med DB2 Cube Views
- OLAP-senteret (grafisk brukergrensesnitt), som du kan bruke når du skal importere og eksportere metadata og opprette kubemodeller og kuber
- Optimalisering, som gir deg hjelp til å forbedre ytelsen til OLAP-spørringer
- Metadataobjekter, som kan lagres i kataloger i DB2 Universal Database (DB2)
- Programmeringsgrensesnittet (API), som du kan bruke til å opprette applikasjoner som har tilgang til data ved hjelp av SQL
- Eksempler på hvordan du bygger dimensjoner og komplekse mål fra metadataobjekter, som du igjen kan bruke til å modellere typiske forretningsscenarier

Hvem boken er beregnet på

Med DB2 Cube Views kan du avlese flerdimensjonale metadata fra OLAP- og databaseverktøy og lagre disse metadataene i DB2-katalogene. Du kan deretter ved hjelp av disse metadataene opprette OLAP-kubemodeller og -kuber (OLAP=Online Analytical Processing). (Kuber er delsett av kubemodeller.)

DB2 Cube Views har også en Optimaliseringsrådgiver som hjelper deg med å forbedre ytelsen til spørringer du utfører på kubemodeller, ved hjelp av SQL-skript for å bygge sammendragstabeller.

Les denne boken hvis du er en databaseadministrator som arbeider med OLAP-metadata og DB2 Universal Database (DB2 UDB). Du bør kjenne til dette:

- DB2-kataloger og sammendragstabeller
- OLAP-begreper, som kuber, dimensjoner, hierarkier og mål
- API-begreper og CLI, ODBC, JDBC, XML, og lagrede DB2-prosedyrer

Syntaks i denne boken

Kommandosyntaks vises i løpende tekstformat med disse konvensjonene:

- Kommandoer vises med små bokstaver, fet skrift.
- Variabler vises i kursiv, og er forklart like etter kommandoen i en liste.
- Hvis du oppgir ett av to alternative parametere, blir de tilgjengelige parametere vist, skilt med loddrette streker, og standardparameteren er understreket.
- Valgfrie variabler og parametere står i hakeparenteser.

Syntaksen for kommandoen **copy** kan for eksempel vises slik:

copy *filnavn* [*filtype*]

filnavn er det nødvendige filnavnet.

[*filtype*] er den valgfri filtypen.

Informasjon på nettet

Dette avsnittet inneholder web-adresser som er knyttet til dette produktet.

www.ibm.com/redbooks

IBMs nettsted for Redbooks

Søk etter, se på og last ned, eller bestill trykte versjoner eller CD-versjoner av følgende "redbooks" fra Redbooks-nettstedet.

- *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

Nettstedet IBM DB2 Cube Views

www.ibm.com/software/data/

IBMs nettsted Data Management

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

Nettstedet Online Support for DB2 Universal Database og DB2 Connect

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

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

Nettsted for DB2-vedlikehold - opprettingspakker for DB2 UDB

www.ibm.com/software/data/developer

Nettstedet DB2 Developer Domain

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

Nettstedet DB2 Product and Service Technical Library

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

[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)

Nettstedet med DB2-publikasjoner

Kapittel 1. Installere, migrere og konfigurere DB2 Cube Views

Denne delen beskriver disse emnene:

DB2 Cube Views-forbedringer

Denne utgaven har flere endringer til DB2 Cube Views-metadata siden DB2 Cube Views versjon 8.1.

Introduksjon til DB2 Cube Views

DB2 Cube Views er en tilleggsfunksjon til DB2 Universal Database som gjør det enklere for DB2 UDB å utføre OLAP-behandling. Ved hjelp av DB2 Cube Views kan du strømlinjeforme distribuering og administrasjon av OLAP-løsninger og forbedre ytelsen til OLAP-verktøyer og -applikasjoner.

Installeringskrav

Før du installerer DB2 Cube Views, må alle kravene være oppfylt.

Installere DB2 Cube Views

Installer DB2 Cube Views på plattformene Windows, AIX, Linux og Solaris ved hjelp av startpanelet for installering.

Migrere DB2 Cube Views

Du kan migrere den DB2 Cube Views-aktiverte databasen fra versjon 8.1 til versjon 8.2 ved å kjøre et skript manuelt, eller ved hjelp av OLAP-senteret.

Konfigurere en database for DB2 Cube Views

Du kan konfigurere en ny database til bruk for DB2 Cube Views.

Konfigurere CVSAMPLE-eksempeldatabasen

DB2 Cube Views inneholder eksempeldata som du kan bruke til å opprette eksempeldatabasen CVSAMPLE.

Problemsøking for en databasetilkobling

Hvis du ikke kan koble deg til en database ved hjelp av OLAP-senteret, kontrollerer du at DB2 Cube Views er av samme versjon som metadatatabellene i DB2-katalogene.

DB2 Cube Views-forbedringer

Denne utgaven har flere endringer til DB2 Cube Views-metadata siden DB2 Cube Views versjon 8.1.

Informasjon om begrensninger og funksjonelle avhengigheter for å forbedre sammendragstabeller

DB2 Cube Views bruker informasjon om begrensninger og funksjonelle avhengigheter for å opprette sammendragstabeller som er mindre og gir bedre spørringsdekning.

Begrensningsinformasjon

I DB2 Cube Views er kubemodellbegrensninger definert mellom fakta- og dimensjonstabellene (eller mellom tabellene i en snøfnuggdimensjon). Hvis en sammendragstabell inneholder primærnøkkelen til en dimensjonstabell, kan DB2 UDB omdirigere spørringen til sammendragstabellen, selv om spørringen refererer til andre kolonner fra dimensjonen.

Funksjonell avhengighet

DB2 UDB har en intelligent dirigeringsfunksjon som er basert på informasjon om funksjonell avhengighet for å besvare SQL-spørringer.

Med funksjonelle avhengigheter kan du oppgi at en eller flere kolonner er funksjonelt avhengige av en eller flere andre kolonner, såfremt alle kolonnene finnes i den samme tabellen.

Hvis en SQL-spørring inneholder en kolonne fra en sammendragstabell og en kolonne fra en tabell der sammendragstabellen ble konstruert, og det finnes en funksjonell avhengighet mellom de to kolonnene, kan DB2-optimalisatoren løse spørringen ved å kombinere de to tabellene for å danne det endelige resultatsettet.

DB2 Cube Views kan anbefale sammendragstabeller som bare inkluderer nøkkelkolonnene fra valgte dimensjoner og nivåer, slik at sammendragstabellen blir snevrere.

Brukerspesifiserte optimaliseringssektorer for å forbedre sammendragstabeller

Denne utgaven av DB2 Cube Views inneholder en mekanisme der du kan angi hvilke typer spørringer som vil bli brukt, og hvilket område av kubene de vil referere til. Optimaliseringsrådgiveren kan bruke disse tilleggsmetadataene til å fokusere optimaliseringen på bestemte områder av en kube. Dette vil øke ytelsen til alle typer spørringer, men den største økningen vil du sannsynligvis oppleve for rapportspørringer.

Flere XML-skjemaversjoner støttes av metadata-APIen

Metadata-APIen for DB2 Cube Views har bare fullstendig støtte for XML-skjema versjon 8.2.0.1.0, inkludert de nye DESCRIBE- og TRANSLATE-operasjonene.

Alle XML-dokumentene som blir overført til og fra metadata-APIen, må ha et versjonsnummer. Ved hjelp av versjonsnummeret kan den lagrede prosedyren identifisere hvilket XML-skjema klienten bruker.

Nullverdiegenskaper for attributter og mål

Attributtene og målene for metadata-APIen er utvidet med egenskapen nullverdiegenskaper, som har et verdisett som består av *ja*, *nei* og *ukjent*.

Nullverdiegenskapen *ja*

Attributtet eller målet kan inneholde nullverdier.

Nullverdiegenskapen *nei*

Attributtet eller målet kan inneholde nullverdier.

Nullverdiegenskapen *ukjent*

Nullverdiegenskapen kunne ikke fastsettes av APIen, eller den kunne ikke fastsettes av migreringsfunksjonen da attributtet eller målet ble migrert fra versjon 8.1 til versjon 8.2.

Modellering av hierarkier ved hjelp av nivåer

I tidligere utgaver av DB2 Cube Views ble hierarkier modellert fra en sortert liste med attributter, som bestod av attributter og objekter kalt attributtforhold. I denne utgaven blir hierarkier modellert fra en sortert liste med nivåobjekter der hvert nivå refererer til ett eller flere attributter.

Denne utgaven inkluderer en ny nivåveiviser og vinduet Nivåegenskaper, der du kan opprette og endre de nye nivåobjektene.

Funksjonelle avhengigheter for nivåobjekter

Når du oppretter et nivå, kan du be om at DB2 forsøker å opprette et funksjonelt avhengighetsobjekt.

En funksjonell avhengighet angir at nivåobjektets standardattributt og beslektede attributter er funksjonelt bestemt av nivåets nøkkelattributter. Optimaliseringsrådgiveren kan deretter bruke den funksjonelle avhengigheten til å redusere størrelsen på sammendragstabellene den anbefaler.

Hvis en funksjonell avhengighet ikke kan opprettes, blir det returnert en varselmelding. Funksjonelle avhengigheter kan for eksempel ikke peke til kolonner som går over mer enn tabell.

Begrensning: Resultatene fra spørringen kan være ukorrekte hvis de underliggende kolonnene ikke er i samsvar med forholdet som er definert i den funksjonelle avhengigheten for nivåobjektet. Fordi DB2 UDB ikke kontrollerer gyldigheten til dataene med hensyn til funksjonelle avhengigheter, må du forsikre deg om at dataene i tabellkolonnene er funksjonelt avhengige på den måten du oppgav.

Du finner flere opplysninger om funksjonelle avhengigheter under "Nivåer" på side 27.

Automatisk drift av optimaliseringsrådgiveren

Denne utgaven inkluderer den nye API ADVISE-operasjonen. Denne operasjonen anbefaler hvilke sammendragstabeller som skal bygges for å forbedre spørringsytelsen for en kubemodell. Denne operasjonen har argumenter som begrenser hvor lenge optimaliseringsrådgiveren kan kjøre, og hvor mye lagerplass den kan bruke for sammendragstabellene.

Oppdatert eksempeldatabase med navnet CVSAMPLE

En ny og forbedret eksempeldatabase følger med, CVSAMPLE. CVSAMPLE-databasen er et mer robust snøfnuggskjema. DB2 Cube Views-metadata følger også med, som reflekterer metadataendringene i denne utgaven. Du kan lese om hvordan du konfigurerer CVSAMPLE-databasen under "Konfigurere CVSAMPLE-eksempeldatabasen" på side 8.

Introduksjon til DB2 Cube Views

DB2 Cube Views er en tilleggsfunksjon til DB2 Universal Database som gjør det enklere for DB2 UDB å utføre OLAP-behandling. Ved hjelp av DB2 Cube Views kan du strømlinjeforme distribuering og administrasjon av OLAP-løsninger og forbedre ytelsen til OLAP-verktøyer og -applikasjoner.

Med DB2 Cube Views kan du beskrive dimensjonsstrukturen til relasjonstabeller og opprette OLAP-konstruksjoner. Du kan lagre den strukturelle informasjonen og OLAP-konstruksjonene som flerdimensjonale metadata i DB2-databasen.

Ved å bruke de nye flerdimensjonale metadataene i DB2 UDB oppnår du to hovedfordeler:

Forbedrer flyten av flerdimensjonale metadata mellom Business Intelligence-verktøy og applikasjoner

Ved å bruke OLAP-senteret, et grafisk grensesnitt som leveres, kan brukere av datavarehus- og Business Intelligence-verktøy lagre de flerdimensjonale metadataene som en del av DB2-databasen og gjøre dem tilgjengelige for alle verktøy og applikasjoner.

Forbedrer ytelsen til spørringer i OLAP-stil

Basert på de flerdimensjonale metadataene kan du opprette DB2-sammendragstabeller ved hjelp av anbefalingene fra optimaliseringsrådgiveren i OLAP-senteret. Sammendragstabellene inneholder forhåndsbergnede data som er tilordnet OLAP-strukturene. Spørringer som blir generert fra datavarehus- eller Business Intelligence-applikasjoner med samme OLAP-struktur, vil få forbedret ytelse.

DB2 Cube Views utnytter DB2-funksjoner som sammendragstabeller, ulike indeksskjemaer, operatører i OLAP-stil og samlingsfunksjoner. Disse komponentene finnes:

Flerdimensjonale metadataobjekter

Du kan opprette et sett med metadataobjekter for dimensjonell modellering av relasjonsdata og OLAP-strukturer. DB2 Cube Views lagrer disse metadataobjektene som du oppretter, i DB2-katalogen.

OLAP-senter

Med OLAP-senteret kan du opprette, manipulere, importere eller eksportere kubemodeller, kuber og andre metadataobjekter som du skal bruke i OLAP-verktøy. OLAP-senteret inneholder veivisere som er enkle å bruke, og vinduer der du kan arbeide med metadataobjektene. Optimaliseringsrådgiveren analyserer for eksempel metadataobjektene og anbefaler hvordan du skal bygge sammendragstabeller som lagrer og indekserer samlede data for SQL-spørringer i OLAP-stil. Du finner opplysninger om hvordan du starter OLAP-senteret under Starte OLAP-senteret. Når du har startet OLAP-senteret, kan du lese i Optimalisere en kubemodell om hvordan du bruker veiviseren for optimaliseringsrådgiveren.

Multidimensional Services

DB2 Cube Views inneholder et SQL-basert og XML-basert programmeringsgrensesnitt (API) for utviklere av OLAP-verktøy og OLAP-applikasjoner. Ved hjelp av CLI-, ODBC- eller JDBC-tilkoblinger eller ved hjelp av innfelt SQL til DB2 UDB, kan applikasjoner og verktøy bruke en enkelt lagret prosedyre til å opprette, endre og hente metadataobjekter.

Eksempeldata

Det finnes en eksempelapplikasjon og eksempeldatabase som hjelper deg å lære hvordan du bruker produktet.

Du kan også utveksle metadataobjekter mellom DB2-katalogen og OLAP-verktøyene. For import og eksport av metadataobjekter til eller fra DB2-katalogen finnes det funksjoner kalt metadatabroer for bestemte OLAP- og databaseverktøy. Se i dokumentasjonen til ditt OLAP- eller databaseverktøy hvis du vil finne ut om det finnes en metadatabro du kan bruke.

Installeringskrav

Før du installerer DB2 Cube Views, må alle kravene være oppfylt.

Systemkrav

Du må installere de nødvendige komponentene på alle DB2 UDB-tjenere og -klienter du vil tilkobles. Multidimensional Services må være installert på tjeneren og klienten. Du kan også installere OLAP-senter på klienten.

Begrensning: Du må installere den samme versjonen av DB2 Cube Views for alle klient- og tjenerkomponenter. OLAP-senter versjon 8.1-klienten kan tilkobles en DB2 UDB, versjon 8.2-tjener, men kan ikke utføre operasjonene Create, Alter eller Drop.

Du trenger følgende tjener-, klient- og maskinvarekomponenter:

- Tjenerkomponenten:

Microsoft Windows

Windows NT 4, Windows 2000 32-biters, Windows XP Professional 32-biters, Windows Server 2003 32-biters eller Windows Server 2003 64-biters

AIX AIX versjon 4.3.3 32-biters, AIX 5L 32-biters eller AIX 5L 64-biters

Linux: Linux Red Hat 8 (kernel 2.4.18, glibc 2.2.93-5) 32-biters, Linux SuSE 8.0 (kernel 2.4.18, glibc 2.2.5) 32-biters, Linux SLES 8 SP3 (kernel 2.4.21, glibc 2.2.5) 32-biters eller Linux RHEL 3 oppdatering 2 (kernel 2.4.21-9, glibc 2.3.2.) 32-biters.

Du finner oppdatert informasjon om distribusjons- og kjernnivåer som støttes av DB2 UDB, på: www.ibm.com/db2/linux/validate

På Sun Solaris

Solaris 8 32-biters, Solaris 8 64-biters, Solaris 9 32-biters eller Solaris 9 64-biters

På HP-UX

HP-UX 11i v2 64-biters for Intel Itanium

- Klientkomponent: Windows NT 4, Windows 2000 32-biters, Windows XP 32-biters, Windows XP 64-biters, Windows Server 2003 32-biters eller Windows Server 2003 64-biters
- Maskinvarekomponenter:
 - 500 MB lagerplass
 - 256 MB RAM

Forutsetninger for å installere DB2 Cube Views

Før du installerer DB2 Cube Views, må du installere disse komponentene:

DB2 Informasjonssenter

Hvis du vil ha tilgang til hjelpesystemet for OLAP-senteret og den elektroniske versjonen av *Veiledning og referanse for DB2 Cube Views*, må du installere DB2 Informasjonssenter. Installer DB2 Informasjonssenter fra CDen DB2 Informasjonssenter.

DB2 Universal Database versjon 8.2

Du må installere DB2 UDB versjon 8.2 før du kan installere DB2 Cube Views.

Merk: Fjern alle tidligere versjoner av DB2 UDB før du installerer DB2 UDB versjon 8.2.

Installere DB2 Cube Views

Installer DB2 Cube Views på plattformene Windows, AIX, Linux og Solaris ved hjelp av startpanelet for installering.

Installere DB2 Cube Views i Windows

Før du installerer DB2 Cube Views, kontrollerer du følgende:

- Du har installert DB2 Informasjonssenter.
- Du har fullført en vellykket installering av DB2 Universal Database versjon 8.2.
- Du oppfyller alle installeringskravene.

Slik installerer du DB2 Cube Views i Windows:

1. Sett inn CDen med DB2 Cube Views. Installeringsprogrammet starter automatisk.
2. På startpanelet klikker du på **Versjonsmerknader** for å få oppdatert informasjon om krav til lagerplass og minne. Se også om filen readme.txt i rotkatalogen på DB2 Cube Views-CDen inneholder ytterligere instruksjoner.
3. Klikk på **Installer produkter** for å starte installeringen, og følg instruksjonene.

Installere DB2 Cube Views på AIX, Linux eller Solaris:

Før du installerer DB2 Cube Views, kontrollerer du følgende:

- Du har fullført en vellykket installering av DB2 Universal Database versjon 8.2.
- Du oppfyller alle installeringskravene.

Slik installerer du DB2 Cube Views på AIX, Linux eller Solaris:

1. Sett inn CDen med DB2 Cube Views.
2. Bytt til katalogen for ditt UNIX-operativsystem og start filen db2setup.
3. På startpanelet klikker du på **Versjonsmerknader** for å få oppdatert informasjon om krav til lagerplass og minne. Se også om filen readme.txt i rotkatalogen på DB2 Cube Views-CDen inneholder ytterligere instruksjoner.
4. Klikk på **Installer produkter** for å starte installeringen, og følg instruksjonene.

Migrere XML-filer fra DB2 Cube Views fra versjon 8.1 til versjon 8.2

Du kan migrere den DB2 Cube Views-aktiverte databasen fra versjon 8.1 til versjon 8.2 ved å kjøre et skript manuelt, eller ved hjelp av OLAP-senteret.

Migrere med et DB2-skript

Anbefaling: Reservekopier og eksporter XML-metadatabasene fra versjon 8.1 før du utfører dette migreringsskriptet.

Bruk filen db2mdmigrate.sql i katalogen sqllib\misc for å opprette de nye metadatatabellene og tilleggs-SQL som er nødvendig for å migrere metadata fra versjon 8.1 til versjon 8.2. Slik bruker du denne filen:

1. Koble deg til databasen du vil migrere ved å oppgi denne kommandoen:
`db2 connect to databasenavn`
2. Oppgi denne kommandoen fra DB2-kommandovinduet for å kjøre filen db2mdmigrate.sql.
`db2 -tvf db2mdmigrate.sql`

Dette skriptet har ingen feilhåndtering. Skriptet går gjennom et sett med DDL- og SQL-setninger. Hvis noen av setningene mislykkes, eller hvis du avbryter skriptet før det er ferdig, blir migreringen bare delvis utført, og det er ikke sikkert at DB2 Cube Views fungerer slik det skal.

3. Hvis det oppstår feil, utfører du en av disse oppgavene:
 - a. Slett alle DB2INFO.*-tabeller.
 - b. Slett den lagrede prosedyren DB2INFO.MD_MESSAGE.
 - c. Gjenopprett metadatatabellene ved hjelp av filen db2mdapi.sql i katalogen sqllib/misc.

Migrere med OLAP-senteret

Når OLAP-senteret kobles til en DB2 UDB-database, registrerer OLAP-senteret automatisk gjeldende versjon av DB2 UDB, og avgjør om det er nødvendig å migrere metadatatabellene. Hvis OLAP-senteret finner ut at du må migrere, får du en feilmelding som forteller dette.

Hvis du godtar OLAP-senterets anbefaling om migrering, kobler OLAP-senteret deg til databasen og migrerer metadatatabellene til versjon 8.2. Hvis du ikke vil migrere, blir ikke OLAP-senteret koblet til databasen.

Hvis det oppstår feil under migreringen, vil OLAP-senteret tilbake stille transaksjonen, og databasen blir ikke migrert.

Konfigurere en database for DB2 Cube Views

Du kan konfigurere en ny database til bruk for DB2 Cube Views.

Konfigurering av en database omfatter disse trinnene:

- Registrere den lagrede prosedyren for DB2 Cube Views sammen med databasen
- Opprette metadatakatalogtabellene for DB2 Cube Views

Første gang du logger deg på en database som ikke er konfigurert for DB2 Cube Views ved å bruke OLAP-senteret, konfigurerer OLAP-senteret databasen for deg. Alternativt kan du konfigurere databasen ved hjelp av filen db2mdapi.sql.

Viktig: Du må ikke endre filen db2mdapi.sql, for da kan du få uventede resultater.

Slik konfigurerer du en database med filen db2mdapi.sql:

1. Åpne vinduet DB2-kommando og koble deg til databasen.
2. Bytt til katalogen SQLLIB/misc og oppgi denne kommandoen i vinduet DB2-kommando:

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

Kjør skriptet db2mdapi.sql bare en gang for en database. Hvis det oppstår feil, løser du problemet, og utfører deretter disse oppgavene:

- a. Slett alle DB2INFO.*-tabeller.
- b. Slett den lagrede prosedyren DB2INFO.MD_MESSAGE.
- c. Gjenopprett metadatatabellene ved hjelp av filen db2mdapi.sql i katalogen sqllib/misc.

Konfigurere CVSAMPLE-eksempeldatabasen

DB2 Cube Views inneholder eksempeldata som du kan bruke til å opprette eksempeldatabasen CVSAMPLE.

Eksempeldataene omfatter et sett med tabeller som inneholder data om et tenkt selskap som produserer og selger drikkevarer. Et sett med metadataobjekter som beskriver eksempeldataene, er også med. De oppgitte CVSAMPLE-dataene er en forbedret og utvidet versjon av MDSAMPLE-databasen som ble levert med versjon 8.1. De fleste av eksemplene i *Veiledning og referanse for DB2 Cube Views* er basert på CVSAMPLE-databasen og tilhørende kubemodell.

Opprett og legg inn data i CVSAMPLE-databasen ved å åpne vinduet DB2-kommando og oppgi disse kommandoene:

1. Opprett en eksempeldatabase med navnet CVSAMPLE:
`db2 create db cvsample`
2. Koble til databasen:
`db2 connect to cvsample`
3. Kjør skriptet db2mdapi.sql for å konfigurere databasen for DB2 Cube Views. Bytt til katalogen QLLIB\misc og oppgi denne DB2-kommandoen:
`db2 -tvf db2mdapi.sql`
4. Bytt til katalogen QLLIB\samples\olap\cvsample og oppgi denne DB2-kommandoen for å opprette CVSAMPLE-tabellene:
`db2 -tvf CVSampleTables.sql`

Når du har opprettet CVSAMPLE-databasen, kan du opprette DB2 Cube Views-metadataobjektene ved å importere definisjonene fra en XML-fil som er eksportert fra en Business Intelligence-applikasjon.

Fremgangsmåten nedenfor legger for eksempel inn data i DB2 Cube Views-katalogtabellene med en fullstendig beskrivelse av CVSAMPLE-databasen.

Slik importerer du CVSAMPLE-metadataene:

1. Start OLAP-senteret og koble deg til CVSAMPLE-databasen.
2. Klikk på **OLAP-senter** → **Importer**.
3. Bla deg frem til filen CVSampleMetadata.xml som ligger i katalogen QLLIB/samples/olap/cvsample. Klikk på **Neste**.

Du kan bla gjennom metadataobjektene i OLAP-senteret. Du finner opplysninger om hvordan du bruker OLAP-senteret under "Opprette DB2 Cube Views-metadataobjekter" på side 46.

Problemsøking for en databasetilkobling

Hvis du ikke kan koble deg til en database ved hjelp av OLAP-senteret, kontrollerer du at DB2 Cube Views er av samme versjon som metadata-tabellene i DB2-katalogene.

Tabellen nedenfor viser hvordan OLAP-senteret fungerer når versjonen av DB2 Cube Views ikke er identisk med versjonen av metadata-tabellene i DB2-katalogene.

Versjon av DB2 Cube Views	Versjon av metadatatabellene i DB2-katalogen	Virkemåte til OLAP-senteret
Ikke installert	Gjelder ikke	Tilkoblingen mislykkes, og OLAP-senteret viser en feilmelding
Versjon 8.1	Ingen	Tilkoblingen mislykkes, og OLAP-senteret viser en feilmelding
Versjon 8.1	Versjon 8.1	Tilkoblingen mislykkes, og OLAP-senteret viser en feilmelding
Versjon 8.2	Ingen	OLAP-senteret kan konfigurere databasen for bruk med DB2 Cube Views versjon 8.2
Versjon 8.2	Versjon 8.1	OLAP-senteret kan migrere databasen for bruk med DB2 Cube Views versjon 8.2
Versjon 8.2	Versjon 8.2	Tilkobling er vellykket.

Kapittel 2. Om DB2 Cube Views-metadatoobjekter

Metadatoobjektene i DB2 Cube Views beskriver relasjonstabeller som OLAP-strukturer, men disse metadatoobjektene er forskjellige fra tradisjonelle OLAP-objekter. Metadatoobjekter lagrer metadato om dataene i basistabellene, de beskriver hvor viktige data er plassert, og de beskriver forhold innenfor basistabellene.

DB2 Cube Views lagrer informasjon om relasjonsdataene i metadatoobjekter som lar deg se dataene i et nytt perspektiv. DB2 Cube Views utvider DB2-katalogen slik at den i tillegg til å lagre informasjon om tabeller og kolonner, inneholder informasjon om hvordan tabellene og kolonnene er beslektet med OLAP-objekter, og relasjonsforholdene mellom disse metadatoobjektene.

Noen metadatoobjekter fungerer som et grunnlag for direkte tilgang til relasjonsdata ved at de samler data eller samsvarer direkte med bestemte kolonner i relasjonstabeller. Andre metadatoobjekter beskriver forholdene mellom grunnleggende metadatoobjekter, og kobler sammen disse basismetadatoobjektene. Alle metadatoobjektene kan grupperes etter sine forhold til hverandre, i et metadatoobjekt som kalles en kubemodell. En kubemodell representerer hovedsakelig en bestemt gruppering og konfigurasjon av relasjonstabeller.

DB2 Cube Views administrerer følgende metadatoobjekter, og lagrer dem i DB2-katalogen:

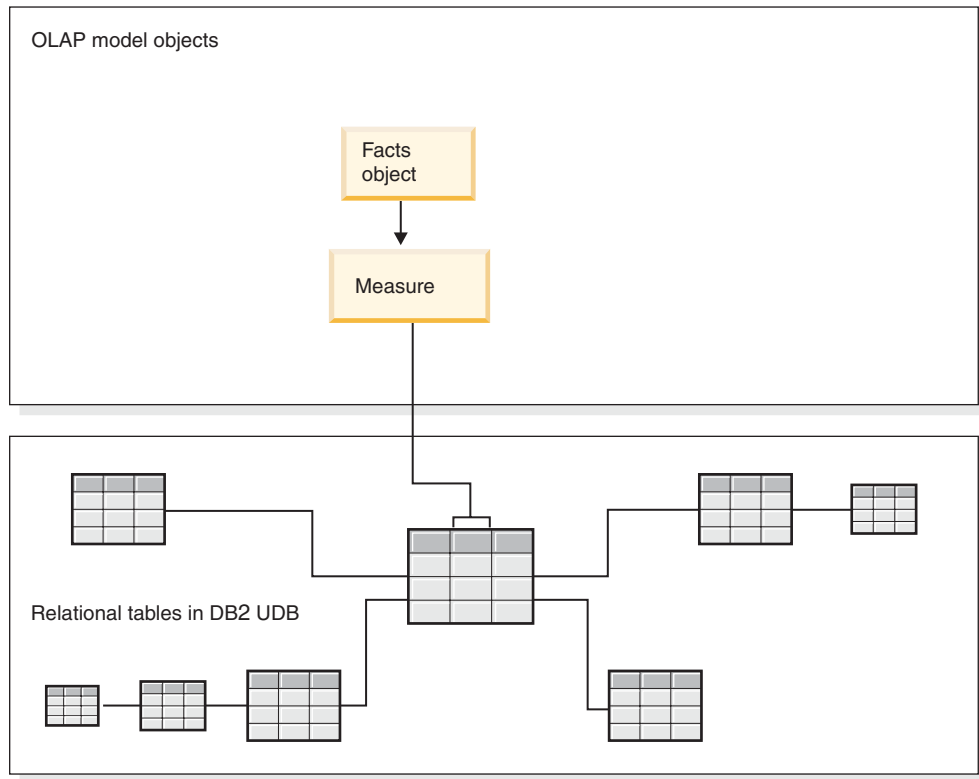
- Kubemodeller
- Faktaobjekter
- Dimensjoner
- Hierarkier
- Nivåer
- Mål
- Attributter
- Attributtforhold
- Kombineringer
- Kuber
- Kubefaktaobjekter
- Kubedimensjoner
- Kubehierarkier
- Kubenivåer

Metadatoobjekter som er tilordnet relasjonstabeller

En kubemodell kan konstrueres på mange måter, men den blir ofte bygd for å representere et relasjonsstjerneskjema eller -snøfnuggskjema. En kubemodell inneholder metadatoobjekter som beskriver relasjoner i relasjonsdataene.

Et stjerneskjema har en faktatabell i sentrum og en eller flere dimensjonstabeller koblet til faktatabellen, og et snøfnuggskjema er en utvidelse av et stjerneskjema, slik at en eller flere dimensjoner er definert av flere tabeller. En kubemodell som er basert på et enkelt stjerneskjema, er bygd rundt et sentralt faktaobjekt.

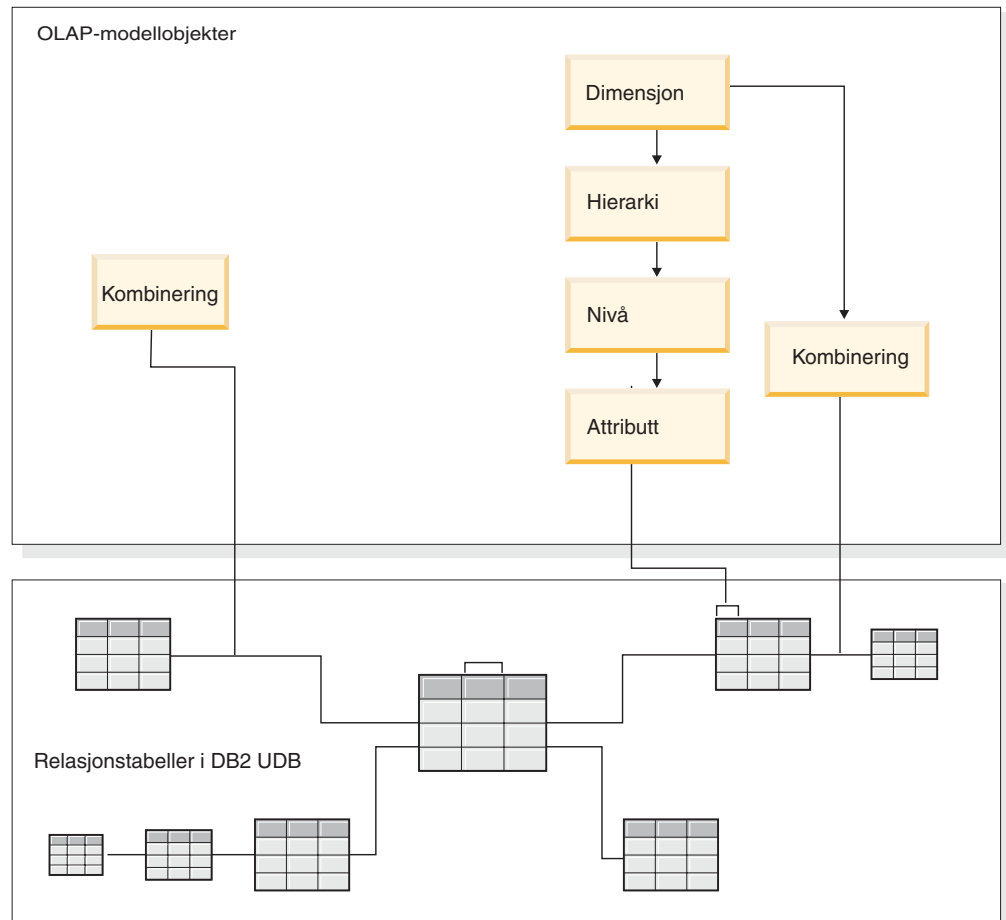
Faktaobjektet inneholder et sett med mål som beskriver hvordan data fra faktatabellen skal samles på tvers av dimensjoner. Mål beskriver databeregninger fra kolonner i en relasjonstabell, og de er kombinert for å opprette faktaobjektet. I figur 1 ser du hvordan mål og et faktaobjekt er relatert til relasjonsdata.



Figur 1. Faktaobjekt. Slik er et faktaobjekt og mål relatert til relasjonsdata

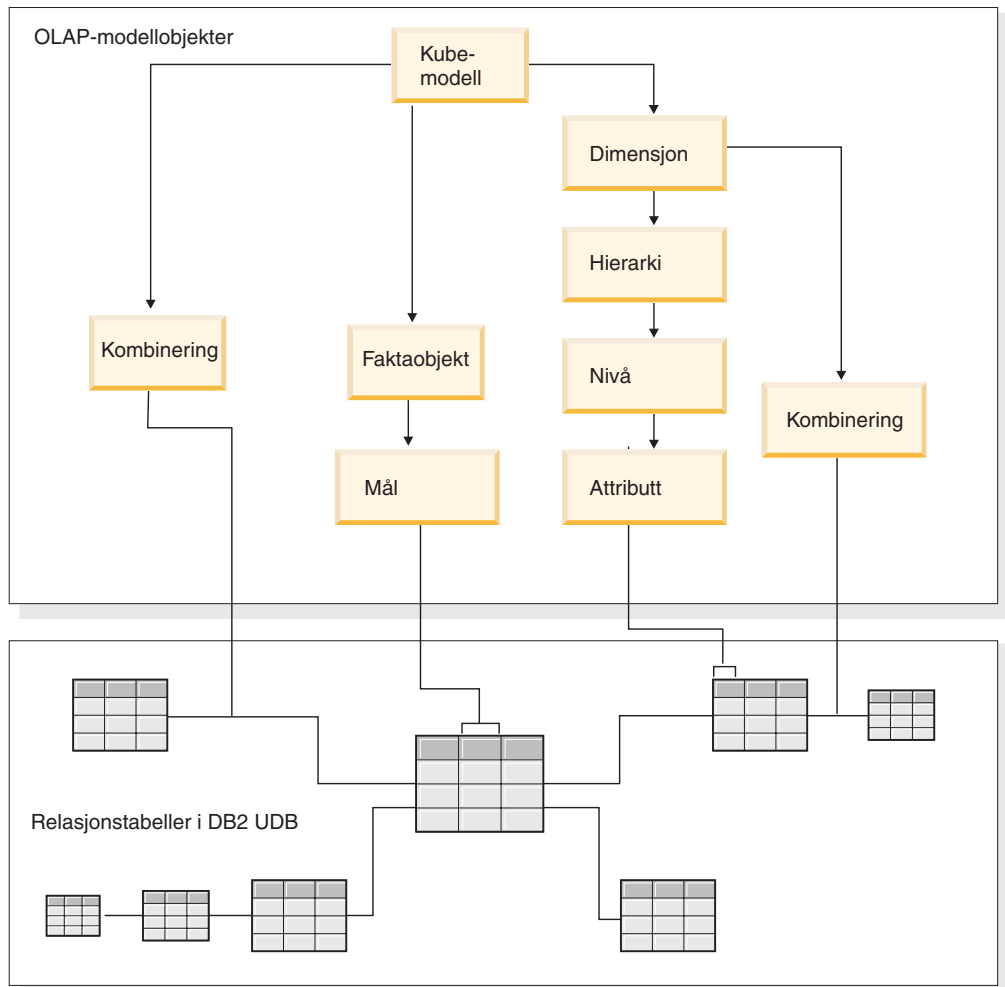
Dimensjoner er knyttet til faktaobjektet i en kubemodell på samme måte som dimensjonstabellene er knyttet til faktatabellen i et stjerneskjema. Kolonner med data fra relasjonstabeller representeres av attributter som er organisert for å utgjøre en dimensjon.

I figur 2 på side 13 ser du hvordan dimensjoner bygges fra relasjonstabeller. Hierarkier lagrer informasjon om hvordan nivåene i en dimensjon er relatert til hverandre og hvordan de er strukturert. Et hierarki gir en metode for å beregne og navigere på tvers av dimensjonen. Hver dimensjon har et tilsvarende hierarki som inneholder nivåer med sett med beslektede attributter. I en kubemodell kan hver dimensjon ha flere hierarkier.



Figur 2. Dimensjon. Slik bygges dimensjoner fra relasjonstabeller

Alle dimensjonene er knyttet til et faktaobjekt i en kubemodell basert på et stjerneskjema eller snøfuggskjema. Kombineringer kan kombinere tabeller for å opprette et faktaobjekt eller en dimensjon. I en kubemodell kan kombineringer koble faktaobjekter til dimensjoner. Dimensjonene refererer til deres tilhørende hierarkier, nivåer, attributter og beslektede kombineringer. Faktaobjekter refererer til mål, attributter og beslektede kombineringer. I figur 3 på side 14 ser du hvordan metadataobjektene passer sammen i en kubemodell, og hvordan de tilordnes til et relasjonssnøfuggskjema.



Figur 3. Kubemodell. Slik passer metadataobjekter sammen og slik tilordnes de til et relasjonssnøfnuggskjema

Du kan bruke komponentene i en kubemodell om igjen for å opprette mer presise kuber for bestemte applikasjoner. En kube er det mest presise metadataobjektet, og det objektet som er nærmest en konseptuell OLAP-kube. En kube er en bestemt forekomst eller et delsett av en kubemodell. En kube har et bestemt sett av liknende, men mer restriktive metadataobjekter som er utledet fra den overordnede kubemodellen, inkludert kubedimensjoner, kubehierarkier, kubenivåer og et kubefaktaobjekt. En kube kan bare ha ett definert kubehierarki for hver kubedimensjon, mens en dimensjon kan ha mange definerte hierarkier for kubemodellen. På grunn av denne strukturelle forskjellen mellom en kube og en kubemodell kan du hente de fleste kuber med en enkelt SQL-setning.

Eksempel på en kubemodell som er tilordnet relasjonstabeller

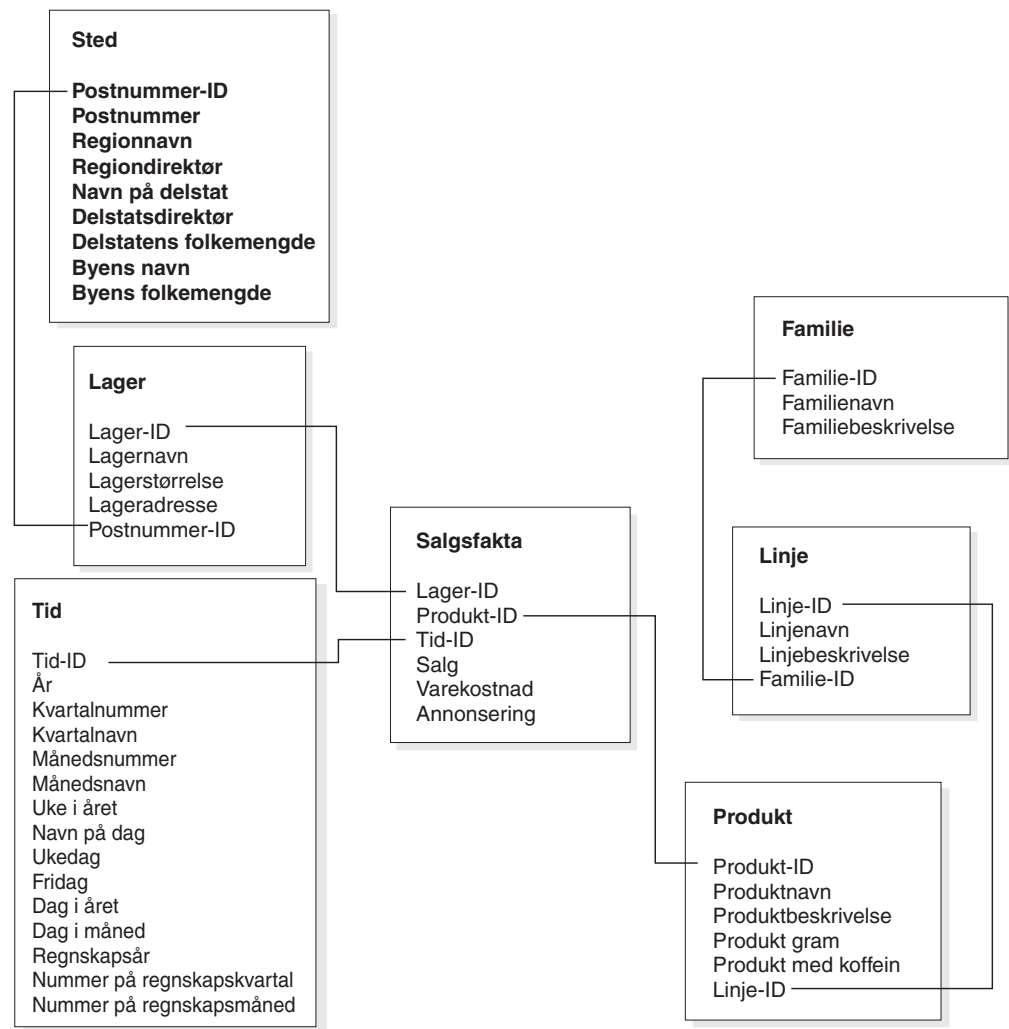
En kubemodell og et tilhørende sett med metadata er bygd for CVSAMPLE-databasen, basert på et snøfnuggskjema.

I figur 4 på side 15 er det vist et snøfnuggskjema med faktatabellen Salg og dimensjonstabellene Lager, Plassering, Tid, Produkt, Linje og Familie. Dimensjonen Marked har to dimensjonstabeller med Lager som den primære dimensjonstabellen og Plassering som uttriggerdimensjonstabellen. Dimensjonen Produksjon har tre

dimensjonstabeller, der Produkt er den primære dimensjonstabellen og tabellen Linje og Familie er uttriggerdimensjonstabellene.

Primærnøkkelen i hver primær dimensjonstabell (Lager, Tid og Produkt) kombineres med den tilsvarende fremmednøkkelen i faktatabellen Salg. Eksempel: Lager.LagerID = Salg.LagerID, Tid.TidID = Salg.TidID, og Produkt.ProduktID = Salg.Produkt.ID.

I snøfnuggdimensjonene er primærnøkkelen i hver dimensjonstabell kombinert med den tilhørende fremmednøkkelen i en annen dimensjonstabell. Eksempel: Plassering.PostnummerID = Lager.PostnummerID, Familie.FamilieID = Linje.FamilieID og Linje.LinjeID = Produkt.Linje.ID.



Figur 4. Snøfnuggskjema. Eksempel basert på CVSAMPLE-snøfnuggskjemaet

Kubemodellen som er basert på CVSAMPLE-snøfnuggskjemaet er bygd rundt faktaobjektet Salg, som beskriver relasjonsdata som er samlet fra faktatabellen Salg. Mål beskriver hvordan data skal beregnes fra kolonnene i tabellen Salg. Faktaobjektet omfatter også attributter som svarer til fremmednøkklene i faktatabellen som brukes til å kombinere dimensjonene med faktaobjektet. I dette eksempelet har faktaobjektet Salg sju mål: Salg, Varekostnad, Annonsering,

Totalutgifter, Annonse/salg-korrelasjon, Fortjeneste og Fortjenestemargin. Objektet Salgsfakta har tre attributter : Lager-ID (Salgsfakta), TidID (Salgsfakta) og Produkt-ID (Salgsfakta).

Dimensjoner er knyttet til faktaobjektet i en kubemodell på samme måten som dimensjonstabellene er knyttet til faktatabellen i et stjerneskjema. Kolonner med data fra relasjonstabeller representeres av attributtobjekter som dimensjonen refererer til.

Produkt-dimensjonen refererer til disse attributtene:

- Familie-ID
- Familienavn
- Familiebeskrivelse
- Linje-ID
- Linjenavn
- Linjebeskrivelse
- Produkt-ID
- Produktnavn
- Produktbeskrivelse
- Produktvekt
- Produkt med koffein

Dimensjonen Tid refererer til disse attributtene:

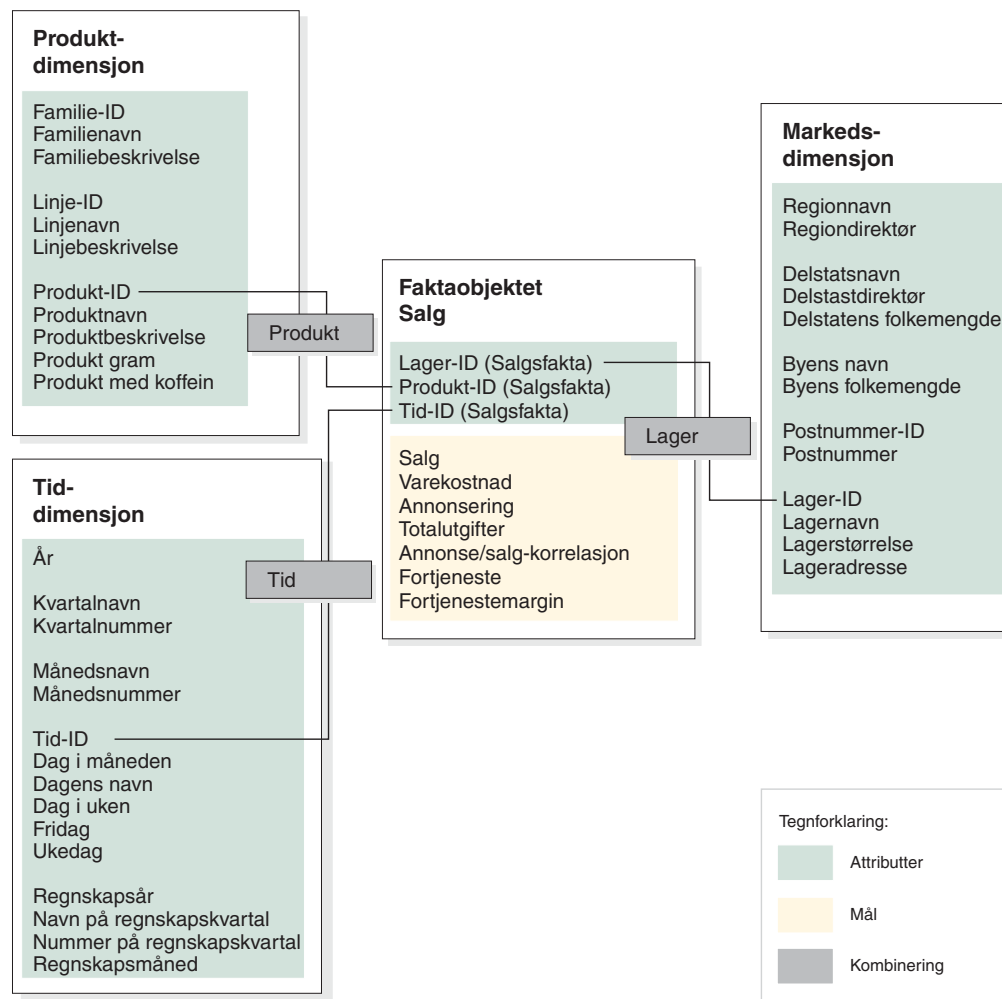
- År
- Kvartalnavn
- Kvartalnummer
- Månednavn
- Månednummer
- Tid-ID
- Dag i måneden
- Dagens navn
- Dag i uken
- Fridag
- Ukedag
- Regnskapsår
- Navn på regnskapskvartal
- Regnskapskvartalnummer
- Regnskapsmåned

Produkt-dimensjonen refererer til disse attributtene:

- Regionnavn
- Regionsjef
- Delstatsnavn
- Delstatsdirektør
- Delstatens folkemengde
- Byens navn
- Byens folkemengde

- Postnummer-ID
- Postnummer
- Lager-ID
- Lagernavn
- Lagerstørrelse
- Lageradresse

Det blir opprettet en kombinerings for å kombinere hver dimensjon med faktaobjektet. De tre kombineringsene i dette eksempelet er Produkt, Tid og Lager. I figur 5 vises kubemodellen som er beskrevet i dette eksempelet.

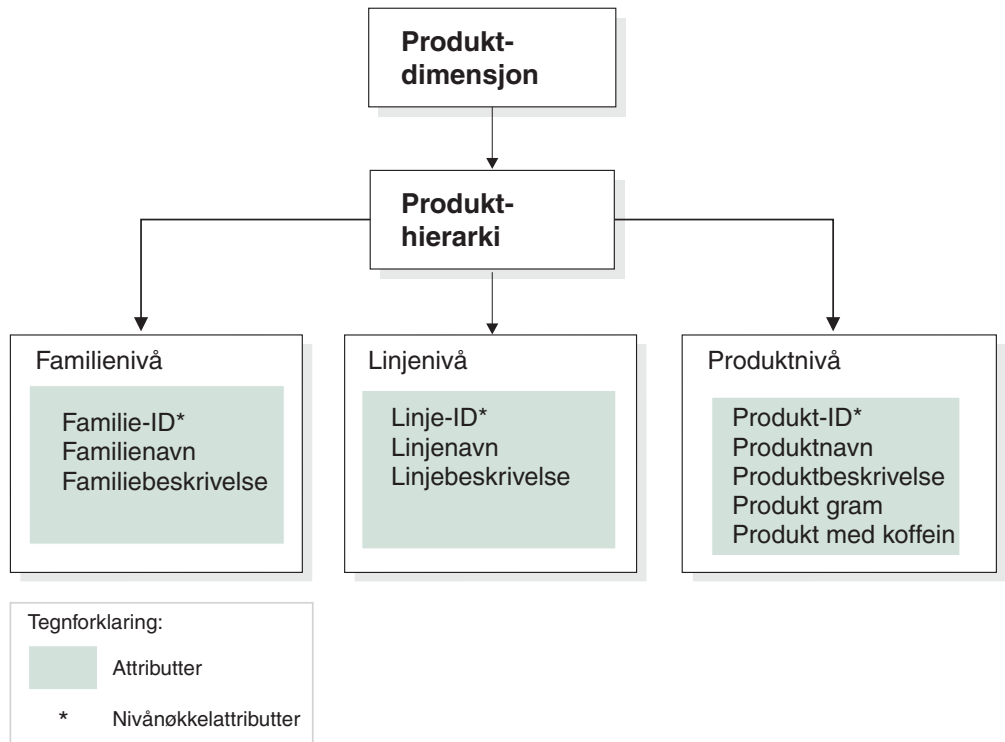


Figur 5. Kubemodell. Eksempel på en kubemodell basert på CVSAMPLE-stjerneskjema

Hierarkier lagrer informasjon om hvordan attributtene som er gruppert i nivåer i dimensjon er relatert til hverandre, og hvordan de er strukturert. Som et metadataobjekt gir et hierarki en metode for å beregne og navigere på tvers av dimensjonen. Hver dimensjon har et tilsvarende hierarki med nivåer som grupperer beslektede attributter. I en kubemodell kan hver dimensjon ha flere hierarkier.

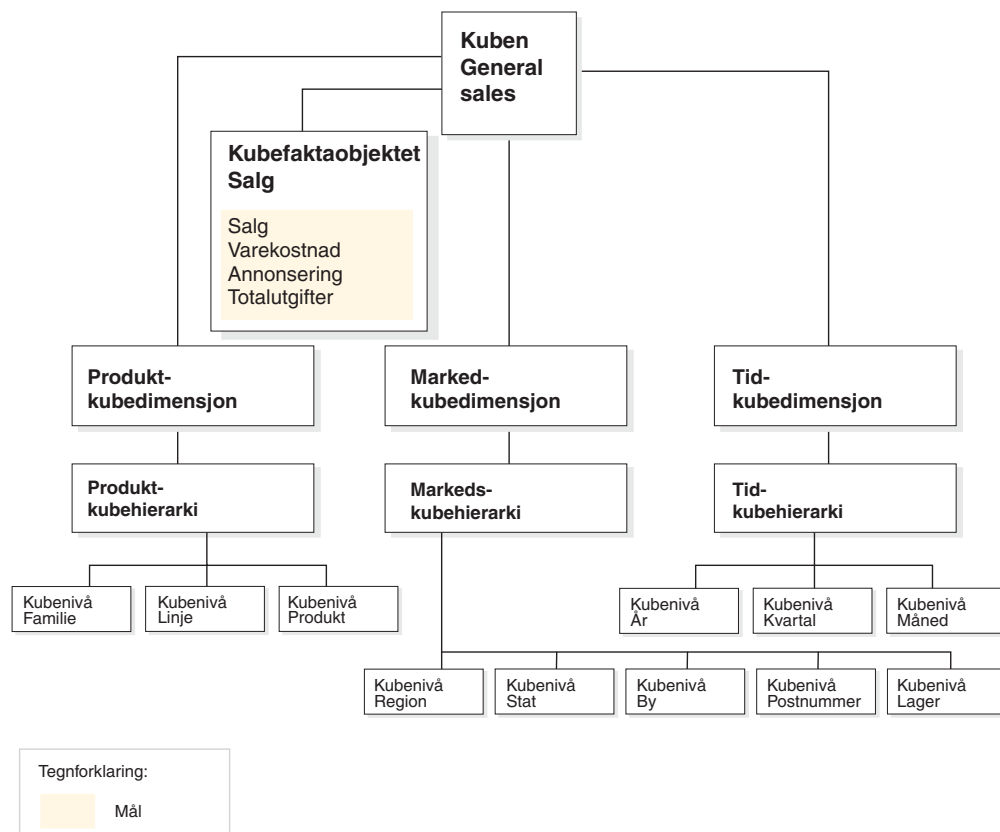
Hierarkiet Produkt inkluderer alle attributtene i dimensjonen Produkt, slik det er vist i figur 6 på side 18. Attributtene i dimensjonen Produkt er gruppert i tre

nivåer. Nivået Familie er det øverste nivået i hierarkiet Produkt. Nivået Familie har Familie-ID som nivånøkkelattributt, Familienavn som standardattributt og Familiebeskrivelse som beslektet attributt. Det andre nivået, nivået Linje, har Linje-ID som nivånøkkelattributt, Linjenavn som standardattributt og Linjebeskrivelse som beslektet attributt. Det nederste nivået, Produkt, har Produkt-ID som nivånøkkelattributt, Produktnavn som standardattributt og Produktbeskrivelse, Produktvekt og Produkt med koffein som beslektede attributter.



Figur 6. Dimensjon. Eksempel på dimensjon, basert på dimensjonen Produkt i CVSAMPLE-databasen

Du kan også bygge en eller flere kuber for kubemodellen. CVSAMPLE-databasen har to kuber, men her er bare kuben General sales beskrevet. Kuben General sales er vist i figur 7 på side 19. Kubefaktaobjektet refererer til et delsett med målene (Salg, Varekostnad, Annonsering og Totalutgifter) fra kubemodellens faktaobjekt. Kuben har tre kubedimensjoner, og hver kubedimensjon refererer til en av de tre dimensjonene i kubemodellen. Produkt-kubedimensjonen har et Produkt-kubehierarki med kubnivåer som refererer til nivåene Familie, Linje og Produkt. Marked-kubedimensjonen har et Marked-kubehierarki med kubnivåer som refererer til Region, Delstat, By, Postnummer og Lager. Tid-kubedimensjonen har et Tid-kubehierarki med kubnivåer som refererer til et delsett med nivåer som inkluderer År, Kvartal og Måned. Kuben har bare ett definert kubehierarki per kubedimensjon. (En kube kan bare ha ett kubehierarki per kubedimensjon.)



Figur 7. Kube. Kubeeksempel basert på kuben General sales i CVSAMPLE-databasen

Generelle egenskaper for metadataobjekter

Hvert metadataobjekt har et sett med felles egenskaper og metadataobjektspesifikke egenskaper. De felles egenskapene brukes til å identifisere metadataobjektforekomstene, til å beskrive bruken av eller rollen for metadataobjektforekomstene, og til å spore endringer i metadataobjektforekomster. Metadataobjektene navngis ved hjelp av et skjema på samme måte som andre Db2-objekter. Hvis du ikke vil bruke et standard bruker-ID-skjema for et metadataobjekt, må du kvalifisere metadataobjektet fullstendig med ønsket skjemanavn.

Tabellen nedenfor beskriver de felles egenskapene som finnes for alle metadataobjekter.

Tabell 1. Generelle egenskaper for metadataobjekter

Egenskap	Beskrivelse
Navn	Navn på metadataobjektet.
Skjema	Skjemaet som eier metadataobjektet.
Forretningsnavn	Et navn som vises til brukeren. Dette navnet kan brukes i et grafisk brukergrensesnitt som et navn som har mening for brukeren.
Kommentarer	En tekstbeskrivelse eller kommentar for metadataobjektet.

Tabell 1. Generelle egenskaper for metadataobjekter (fortsettelse)

Egenskap	Beskrivelse
Opprettingstidspunkt	Tidspunktet da metadataobjektet ble opprettet.
Oppretter	Brukeren (skjemaet) som definerte metadataobjektet.
Endringstidspunkt	Tidspunktet da metadataobjektet sist ble endret.
Modifikator	Brukeren (skjemaet) som utførte endringen.

I tillegg til et felles sett med egenskaper, har hvert metadataobjekt et sett av spesifikke egenskaper. Disse spesifikke egenskapene beskriver komponentene og kvalitetene som definerer metadataobjektet. Hvis du ønsker informasjon om egenskaper som er spesifikke for hvert metadataobjekt, kan du lese emnet som omhandler dette metadataobjektet.

Navngivningsregler for metadataobjekter

DB2 UDB har to forskjellige navngivningsregler for navn på objekter, regler for vanlige navn og for navn med skilletegn. For metadataobjekter brukes reglene for skilletegn ved navngivning av objekter og referanser til DB2-tabeller og kolonner. Reglene for skilletegn tillater at navn inneholder både store og små bokstaver, mellomrom og spesialtegn, for eksempel nasjonale tegn. Det fullstendige settet med tegn er bestemt av kodesettet til databasen der metadataobjektene er lagret.

Disse reglene gjelder for metadataobjekter:

Tabell 2. Navngivningsregler for metadataobjekter

Objekt	Regler
Skjema	<ul style="list-style-type: none"> Lengde: 1-30 byte Beskyttede navn: Skjemanavn kan ikke være <i>SESSION</i> eller begynne med <i>SYS</i>. Bare navn med store bokstaver er beskyttet.
Navn på metadataobjekt	<ul style="list-style-type: none"> Lengde: 1-128 byte Ingen andre begrensninger
Forretningsnavn på metadataobjekt	<ul style="list-style-type: none"> Lengde: 1-128 byte Ingen andre begrensninger
Kommentarer for metadataobjekter	<ul style="list-style-type: none"> Lengde: 0-254 byte Ingen andre begrensninger
Tabellskjema som brukes i referanser til kolonner	<ul style="list-style-type: none"> Lengde: 1-128 byte Ingen andre begrensninger
Tabellnavn som brukes i referanser til kolonner	<ul style="list-style-type: none"> Lengde: 1-128 byte Ingen andre begrensninger
Kolonnenavn som brukes i referanser til kolonner	<ul style="list-style-type: none"> Lengde: 1-128 byte Ingen andre begrensninger

Kubemodeller

DB2 Cube Views-kubemodellen er en representasjon av et logisk stjerneskjema eller snøfnuggskjema, og grupperer relevante dimensjonsobjekter omkring et sentralt faktaobjekt.

Hver dimensjon kan ha flere hierarkier. Den strukturelle informasjonen om hvordan tabellen skal kombineres, som brukes av faktaobjektet og dimensjonsobjektene, er referert til i kubemodellen. I kubemodellen er det også lagret nok informasjon til å bygge SQL-spørringer og hente OLAP-data. Andre rapporterings- og OLAP-verktøy som forstår kubemodellen og som kan vise flere utsnitt av en bestemt dimensjon, kan ha fordeler av å bruke kubemodellen.

Kubemodeller definerer et sammensatt sett av forhold, og kan brukes til selektiv avdekking av relevante faktaobjekter og dimensjoner for en dimensjon. Hvert enkelt kombineringsobjekt som kobler en dimensjon til det sentrale faktaobjektet, lagres med den tilsvarende dimensjonen som et sett. Delsett av kubemodellkomponenter kan brukes av mange kuber til ulike analyseformål.

Du kan opprette en tom kubemodell i OLAP-senteret ved hjelp av kubemodellveiviseren. En tom kubemodell har ikke et faktaobjekt eller noen dimensjoner. Med veiviserne i OLAP-senteret kan du fullføre kubemodellen ved å opprette faktaobjektet og en eller flere dimensjoner. Du kan også opprette en fullstendig kubemodell ved hjelp av hurtigstartveiviseren. DB2 Cube Views validerer kubemodellen når optimaliseringsrådgiveren åpnes. Hvis en kubemodell skal optimaliseres, må kubemodellen inneholde disse obligatoriske komponentene:

- Et faktaobjekt
- Minst en dimensjon
- Et hierarki definert for minst en dimensjon
- Kombineringer mellom de eksisterende faktaobjektene og dimensjonene
- Attributter som refererer til eksisterende tabellkolonner

Egenskapene som er spesifikke for kubemodeller, beskrives i tabellen nedenfor.

Tabell 3. Egenskaper for kubemodeller

Egenskap	Beskrivelse
Faktaobjekt	Faktaobjekter som brukes i kubemodellen
Sett av (dimensjon, kombinerings)	Dimensjoner som brukes i kubemodellen, og deres tilsvarende kombinerings

Faktaobjekter

Et faktaobjekt brukes i en kubemodell som sentrum i et stjerneskjema og grupperer beslektede mål som er interessante for en bestemt applikasjon.

Faktaobjektet refererer til attributtene som er brukt i kombineringsene av faktaobjekt og dimensjon, og attributtene og kombineringsene som brukes til å tilordne de ekstra målene på tvers av flere databasetabeller. Flere relasjonsfaktatabeller kan kombineres på bestemte attributter for å tilordne ekstra beslektede mål. I tillegg til et sett av mål, lagrer et faktaobjekt derfor et sett av attributter og et sett av kombinerings.

Du kan bruke faktaveiviseren i OLAP-senteret til å opprette et faktaobjekt. I faktaveiviseren oppgir du en eller flere faktatabeller og eventuelle nødvendige kombinerings, mål og samlinger for målene.

De spesifikke egenskapene for et faktaobjekt beskrives i tabellen nedenfor.

Tabell 4. Egenskaper for faktaobjekter

Egenskap	Beskrivelse
Sett av mål	Et sett av alle beslektede mål i faktaobjektet
Sett av attributter	Et sett av alle attributtene som brukes i faktaobjektet
Sett av kombinerings	Et sett av alle kombineringsene som trengs for å kombinere alle de oppgitte målene og attributtene

Dimensjoner

Dimensjoner gir en metode for å kategorisere et sett av beslektede attributter som sammen beskriver ett aspekt av et mål. Dimensjoner brukes i kubemodeller til å organisere dataene i faktaobjektet i henhold til logiske kategorier som Region, Produkt eller Tid.

Dimensjoner refererer til null eller flere hierarkier. Hierarkier beskriver forholdet og strukturen for de refererte attributtene som er gruppert i nivåer, og kan brukes i navigeringen og beregningen av dimensjonen.

Beslektede attributter og kombineringsene som er nødvendige for å gruppere disse attributtene, defineres i dimensjonens egenskaper.

Dimensjoner har også en type som beskriver om dimensjonen er tidsorientert. En dimensjon med navnet Tid inneholder for eksempel attributter som År, Kvartal og Måned, og er av typen Tid. En annen dimensjon med navnet Marked som inneholder nivåer som Region, Delstat og By, er av typen Vanlig. Typeinformasjonen kan brukes av applikasjoner for å utføre tidsrelaterte funksjoner på en intelligent og passende måte.

Du kan bruke dimensjonsveiviseren i OLAP-senteret til å opprette en ny dimensjon i forbindelse med en kubemodell eller uten referanse til en kubemodell. Du kan dele dimensjoner på tvers av kubemodeller ved å tilføye en eksisterende dimensjon til en kubemodell med veiviseren Tilføy dimensjon.

De spesifikke egenskapene for dimensjoner beskrives i tabellen nedenfor.

Tabell 5. Dimensjonsegenskaper

Egenskap	Beskrivelse
Sett av attributter	Et sett av alle attributtene som brukes i dimensjonen.
Sett av kombinerings	Et sett av alle kombineringsene som trengs for å kombinere alle de oppgitte attributtene. Bare kombineringsene som er nødvendig for å kombinere dimensjonstabellene, oppgis her.
Sett av hierarkier	Et sett av hierarkier som gjelder for dimensjonen.
Sett av nivåer	Et sett av nivåer som det blir referert til av dimensjonen.
Type	Dimensjonstype som kan være Vanlig eller Tid

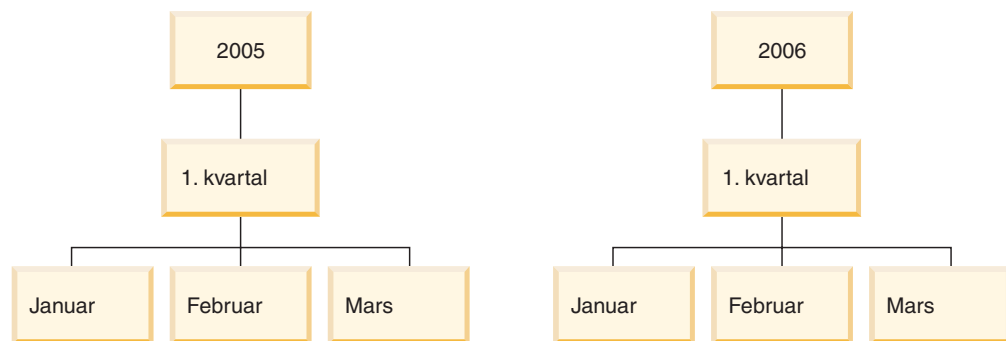
Hierarkier

Et hierarki definerer forhold mellom et sett av attributter som er gruppert på nivåer i dimensjonen av en kubemodell. Definerings- og behandlingsmetode for å krysse dimensjoner. Det kan defineres flere hierarkier for en dimensjon av en kubemodell.

Hierarkitypen beskriver forholdet mellom nivåene i hierarkiet. De følgende fire hierarkitypene støttes:

Balansert

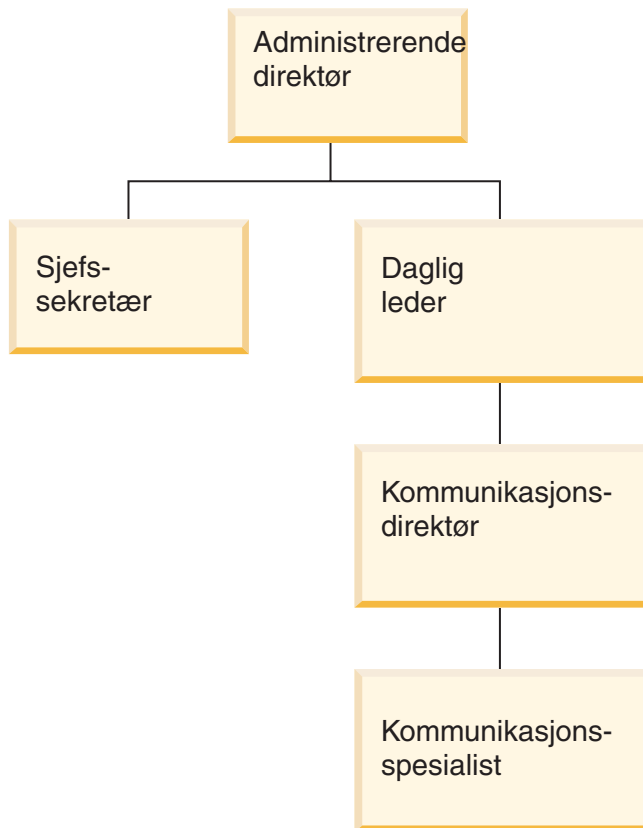
Et hierarki med meningsfulle nivåer og forgreninger som har en konsistent dybde. Det overordnede nivået for hvert nivå er nivået direkte over det. Et balansert hierarki kan representere tid der betydningen og dybden for hvert nivå, for eksempel År, Kvartal og Måned, er konsistent. De er konsistente fordi hvert nivå representerer samme type informasjon, og hvert nivå er logisk ekvivalent. figur 8 viser et eksempel på et balansert tidshierarki.



Figur 8. Balansert hierarki. Eksempel på et balansert hierarki

Ubalansert

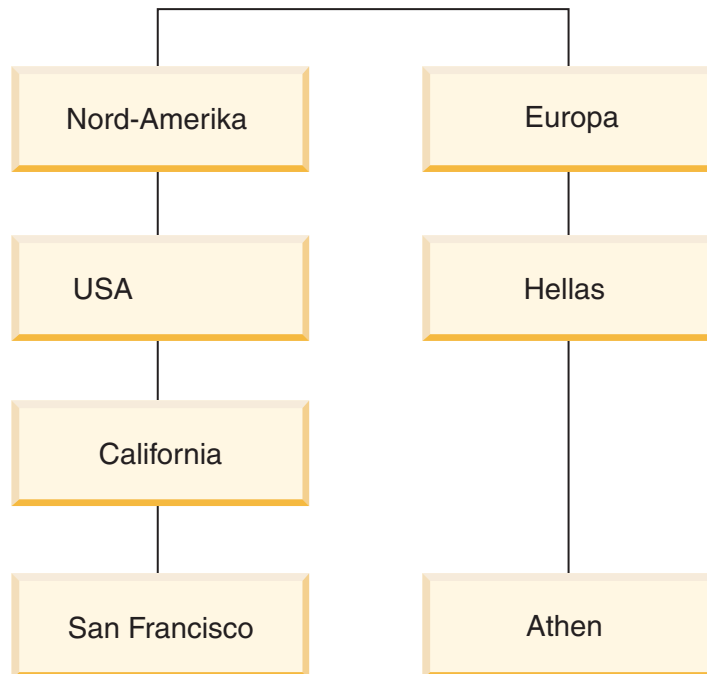
Et hierarki med nivåer som har et konsistent forhold mellom over- og underordnede nivåer, men som har logisk inkonsistente nivåer. Hierarkigrenene kan også ha inkonsistente dybder. Et ubalansert hierarki kan representere et organisasjonskart. I figur 9 på side 24 vises for eksempel en direktør på toppnivået av hierarkiet og minst to av personene i forgreningene under, inkludert administrerende direktør og sjefssekretæren. Den administrerende direktøren har flere personer under seg, men sjefssekretæren har ikke det. Forholdet mellom over- og underordnede nivåer i begge grenene av hierarkiet er konsistent. Men nivåene i begge grenene er ikke logiske ekvivalenter. En sjefssekretær er for eksempel ikke en logisk ekvivalent til en administrerende direktør.



Figur 9. Ubalansert hierarki. Eksempel på et ubalansert hierarki

Ujevnt

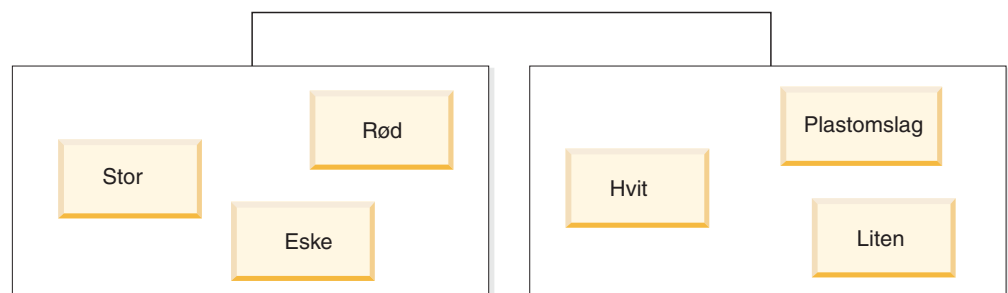
Et hierarki der hvert nivå har en konsistent betydning, men grenene har inkonsistente dybder fordi minst ett medlemsattributt på et grennivå ikke er utfyllt. Et ujevnt hierarki kan representere et geografisk hierarki der betydningen av hvert nivå, for eksempel by eller land, brukes konsistent, mens dybden i hierarkiet varierer. I figur 10 på side 25 vises et geografisk hierarki der nivåene kontinent, land, delstat og by er definert. En gren har Nord-Amerika som kontinent, USA som land, California som delstat og San Francisco som by. Hierarkiet blir imidlertid ujevnt når ett av medlemmene ikke har en post på alle nivåene. En annen gren har for eksempel Europa som kontinent, Hellas som land og Athen som by, men de har ingen post for delstatnivået, fordi dette nivået ikke gjelder for Hellas i dette eksempelet. I dette eksempelet går grenene Hellas og USA til ulike dybder, noe som oppretter et ujevnt hierarki.



Figur 10. Ujevnt hierarki. Eksempel på et ujevnt hierarki

Nettverk

Et hierarki der rekkefølgen for nivåene ikke er oppgitt, men der nivåene har semantisk betydning. I figur 11 er det for eksempel vist et nettverkshierarki som beskriver produktattributter som Farge, Størrelse og Pakketype. Siden nivåene ikke har et over-/underordnet forhold, er rekkefølgen for nivåene ikke viktig. Et produksjonselskap kan ha medlemsposter som hvit for Farge, liten for Størrelse og plastomslag for Pakketype. En andre medlemspost kan ha rød for Farge, stor for Størrelse og eske for Pakketype.



Figur 11. Nettverkshierarki. Eksempel på et nettverkshierarki

Et hierarki oppgir også distribueringsmekanismene for hierarkiet. En distribueringsmekanisme definerer hvordan dataene i hierarkinivåer skal tolkes. De følgende to distribueringsmekanismene støttes:

Standard

Bruker forholdet til nivådefinisjonene i hierarkiet, der hvert nivå i hierarkiet brukes som ett element i distribueringen. Et balansert hierarki for en tidsdimensjon vil for eksempel være organisert av hvert definerte nivå, blant annet År, Kvartal og Måned. Distribueringsmekanismen Standard kan brukes med alle fire hierarkitypene. I tabell 6 på side 26 vises

hvordan noen av de balanserte hierarkiattributtene for en tidsdimensjon organiseres ved hjelp av en standarddistribuering

Tabell 6. Standarddistribuering. Standarddistribuering av et balansert hierarki for en tidsdimensjon

År	Kvartal	Måned
2003	1. kv.	Jan
2003	1. kv.	Feb
2003	1. kv.	Mar
2004	1. kv.	Jan
2004	1. kv.	Feb
2004	1. kv.	Mar

Rekursiv

Bruker de over-/underordnede forholdene mellom nivåene i hierarkiet. Et ubalansert hierarki som bruker rekursiv distribuering, representeres av paret av over- og underordnede nivåer. I tabell 7 vises for eksempel nivåparene for det ubalanserte hierarkiet som beskriver et organisasjonskart som er vist i figur 9 på side 24. Parene av under- og overordnede attributter omfatter: direktør og sjefssekretær, direktør og administrerende direktør, administrerende direktør og kommunikasjonsdirektør, kommunikasjonsdirektør og kommunikasjonsspesialist. Rekursiv distribuering kan bare brukes med et ubalansert hierarki.

Tabell 7. Rekursiv distribuering. Rekursiv distribuering av et ubalansert hierarki for en organisasjonsdimensjon

Overordnet attributt	Underordnet attributt
Direktør	Sjefssekretær
Direktør	Administrerende direktør
Administrerende direktør	Kommunikasjonsdirektør
Kommunikasjonsdirektør	Kommunikasjonsspesialist

Du kan opprette et hierarki i OLAP-senteret ved hjelp av hierarkiveiviseren. Du kan definere et hierarki for en dimensjon når du har opprettet dimensjonen.

Egenskapene for et hierarkiobjekt beskrives i tabellen nedenfor.

Tabell 8. Hierarkiegenskaper

Egenskap	Beskrivelse
Sortert sett av nivåer	En sortert liste over nivåer fra toppen til bunnen i et hierarki.
Type	Hierarkitypen kan være Balansert, Ubalansert, Ujevn eller Nettverk
Distribuering	Hierarkidistribuering kan være Standard, Rekursiv

Nivåer

Et nivå består av et sett med attributter som fungerer sammen som et logisk trinn i en hierarkirekkefølge. Et nivå inneholder ett eller flere attributter som er relatert og som kan fungere i en eller flere roller på nivået. Forholdet mellom attributtene i et nivå er vanligvis definert ved hjelp av en funksjonell avhengighet.

Et nivå kan bruke et attributt på en av tre måter:

Nivånøkkelattributter

Ett eller flere attributter med verdier som entydig identifiserer hver av nivåets forekomster. For nivået By er den enkleste måten å definere et nivånøkkelattributt på, å bruke en ID-kolonne, for eksempel By-ID, som garantert er entydig. Attributtet Bynavn kan ikke være et nivånøkkelattributt, fordi bynavn ikke nødvendigvis er unike i en delstat eller et land. Du kan imidlertid inkludere settet Landnavn, Delstatsnavn og Bynavn som nivånøkkelattributter, fordi disse tre attributtene samlet kan utgjøre en entydig definisjon av en by. Du må kontrollere at nivånøkkelattributtene definerer nivået entydig, fordi DB2 Cube Views oppretter funksjonelle avhengigheter for å forbedre optimaliseringen, basert på nivåene du definerer.

Standardattributt

Et attributt med en verdi som kan vises ved hjelp av en rapporteringsapplikasjon for å gi meningsfylte navn for hver forekomst av et nivå i dataene. Standardattributtet er obligatorisk, og må bestemmes funksjonelt av nivånøkkelattributtet. Hvis for eksempel nivånøkkelattributtet er definert som By-ID, vil ikke verdiene for kolonnen By-ID ha stor nytte i rapporter. Du kan definere Bynavn som standardattributtet som blir vist i rapporter, slik at dataene som vises, er nyttige. Du kan bruke et attributt fra settet med nivånøkkelattributter som standardattributtet. Hvis du for eksempel definerer nivånøkkelattributtet for nivået By som settet med Landnavn, Delstatsnavn og Bynavn, kan du definere standardattributtet som Bynavn.

Beslektede attributter

En samling med null eller flere attributter som inneholder mer informasjon om forekomstene av nivåene som er definert som nivånøkkelattributter. Alle beslektede attributter må være funksjonelt bestemt av nivånøkkelattributtene. Nivået By kan for eksempel ha beslektede attributter som By_Ordfører, By_Befolkning, By_Beliggenhet og By_Beskrivelse.

Du kan bruke nivåveiviseren i OLAP-senteret til å opprette et nytt nivå, uavhengig av eller i sammenheng med et hierarki. Du kan dele nivåer på tvers av kubemodeller i dimensjoner. Når du oppretter et nivå, oppretter DB2 Cube Views en funksjonell avhengighet som standard for attributtene i nivået, slik at standardattributtene og de beslektede attributtene er funksjonelt avhengige av nivånøkkelattributtene i nivået. Optimaliseringsrådgiveren bruker de funksjonelle avhengighetene til å anbefale de beste sammendragstabellene, og DB2-optimalisatoren bruker de funksjonelle avhengighetene til å rute SQL-spørringene på riktig måte.

Det er mange metoder for å modellere et hierarki ved hjelp av nivåer. Enten du følger ideelle eller ikke-ideelle modelleringsteknikker, er det viktig at du definerer hvert nivå slik at nivånøkkelattributtene funksjonelt bestemmer standardattributtet

og beslektede attributter. Nivånøkkelattributtene i et nivå må entydig identifisere verdiene i dette nivået. Du må definere at en funksjonell avhengighet skal opprettes mellom attributtene i hvert nivå. Funksjonelle avhengigheter brukes av optimaliseringsrådgiveren for å anbefale sammendragstabeller, og av DB2-optimalisatoren for å rute SQL-spørringene på riktig måte. Ved hjelp av de funksjonelle avhengighetene kan optimaliseringsrådgiveren anbefale mindre sammendragstabeller, som kan returnere spørringsresultatene raskere.

Ideell modellering

Det ideelle er at hver dimensjons relasjonsdata er lagret i en enkelt dimensjonstabell som inneholder ID-kolonner for hvert av nivåene i dimensjonen, og at hver ID-kolonne entydig identifiserer verdiene i nivået. Du kan for eksempel ha en enkelt dimensjonstabell for dimensjonen Region, som inneholder følgende kolonner:

Tabell 9. Ideell modellering av en dimensjonstabell

Kolonner i en ideell dimensjonstabell for dimensjonen Region
By-ID (primærnøkkel)
Byens navn
Ordfører
Delstats-ID
Delstatsnavn
Delstatsguvernør
Land-ID
Landets navn

Du kan opprette et hierarki som inneholder nivåene Land, Delstat og By. I hvert av nivåene kan du definere en funksjonell avhengighet mellom ID-kolonnen som er definert som nivånøkkelattributtet og eventuelle beslektede attributter, slik det er vist i tabell 10.

Tabell 10. Ideell modellering av et hierarki

Nivå	Nivånøkkelattributt	Nivåbeslektede attributter
Land	Land-ID	Landets navn
Delstat	Delstats-ID	Delstatsnavn Delstatsguvernør
By	By-ID	Bynavn Ordfører

Det blir opprettet funksjonelle avhengigheter mellom disse attributtparene:

- Land-ID → Landnavn
- Delstats-ID → Delstatsnavn, Delstatsguvernør
- By-ID → Bynavn, Ordfører

DB2 Cube Views oppretter ikke en funksjonell avhengighet for By-ID og de beslektede attributtene, fordi By-ID er primærnøkkel, og skal allerede ha en definert begrensning.

Ikke-ideell modellering

Hvis du ikke har entydige ID-datakolonner for hvert nivå i hierarkiet, må du være mer forsiktig når du definerer nivånøkkelattributtene for hvert nivå, slik at du oppretter en gyldig funksjonell avhengighet. Du kan for eksempel ha en enkelt dimensjonstabell for dimensjonen Region, som inneholder følgende kolonner:

Tabell 11. Ikke-ideell modellering av en dimensjonstabell

Kolonner i en ikke-ideell dimensjonstabell for dimensjonen Region
By-ID (primærnøkkel)
Bynavn
Ordfører
Delstatsnavn
Delstatsguvernør
Landnavn

Du kan opprette et hierarki som inneholder nivåene Land, Delstat og By, som i eksempelet på ideell modellering. Nivånøkkelattributtene må imidlertid være mer nøyaktig definert for å sikre at hver rad i nivået kan defineres entydig. Bynavn definerer for eksempel ikke nivået By entydig, fordi det finnes byer med navnet Leicester både i USA og i England. Den eneste måten nivået By kan defineres entydig på, er ved hjelp av kombinasjonen av attributtene Landnavn, Delstatsnavn og Bynavn, slik det er vist i tabell 12.

Tabell 12. Ikke-ideell modellering av et hierarki

Nivå	Nivånøkkelattributter	Nivåbeslektede attributter
Land	Landets navn	
Delstat	Landnavn Delstatsnavn	Delstatsguvernør
By	Landnavn Delstatsnavn Bynavn	Ordfører

Det blir opprettet funksjonelle avhengighet mellom disse to attributtkombinasjonene:

- Landnavn, Delstatsnavn og Bynavn → Byens ordfører
- Landnavn og Delstatsnavn → Delstatsguvernør

Egenskapene for et nivåobjekt beskrives i tabellen nedenfor.

Tabell 13. Nivåegenskaper

Egenskap	Beskrivelse
Sett med nivånøkkelattributter	Sett med ett eller flere attributter som samlet definerer nivået entydig.
Ett standardattributt	Et obligatorisk standardattributt som er funksjonelt bestemt av nivånøkkelattributtene, og som kan brukes av rapporteringsapplikasjoner for å vise meningsfylte data.

Tabell 13. Nivåegenskaper (fortsettelse)

Egenskap	Beskrivelse
Sett med beslektede attributter	Sett med null eller flere attributter som inneholder mer informasjon om nivået, og som er funksjonelt bestemt av nivånøkkelattributtene.
Funksjonell avhengighet	Boolske verdier (Ja eller Nei) som angir om nivået har en tilsvarende DB2 UDB-funksjonell avhengighet.
Navnet på den funksjonelle avhengigheten	Hvis Funksjonell avhengighet er satt til Ja, inneholder denne egenskapen navnet (opptil 18 byte) på den funksjonelle avhengigheten. Hvis Funksjonell avhengighet er satt til Nei, blir denne egenskapen oversett.

Mål

Mål definerer en målentitet og brukes i faktaobjekter.

Mål får mening i forbindelse med et sett med dimensjoner i en kubemodell. En inntekt på 300 har ikke mening i seg selv. Når du setter et inntektsmål i forbindelse med dimensjoner, for eksempel Region og Tid, får målet mening. Inntekten for New York i januar er 300. Vanlige eksempler på mål er Inntekt, Kostnad og Fortjeneste.

Et mål defineres av en kombinasjon av to egenskaper: en liste over SQL-uttrykk og en samlingsliste. Tabellkolonner, attributter og mål tilordnes til en mal for å bygge SQL-uttrykk. De resulterende SQL-uttrykkene brukes deretter som inndata for den første samlingsfunksjonen for målet. Hvis et mål har mer enn en samling, utføres samlingsfunksjonene i den rekkefølgen de er oppført, der hver påfølgende samling bruker den foregående samlingens resultater som inndata. Hvis målets SQL-uttrykk bare refererer til andre mål, blir samlingsfunksjonen valgfri. Samlingsfunksjonen blir valgfri fordi målet det refererer til, gir samlingen.

Et måls SQL-uttrykk blir opprettet av kombinasjonen av to egenskaper: en mal og en liste over kolonner, attributter og mål. Malen bruker en symbolnotasjon der $\{\$n\}$ er symbolet og n refererer til en bestemt kolonne, et bestemt attributt eller et bestemt mål fra listen. Listen over kolonner, attributter og mål er sortert, og posisjonen for en kolonne, et attributt eller et mål i listen tilsvarer verdien av symbolet n.

SQL-uttrykk brukes som inndata til den første samlingen. Hver samling oppgir en funksjon som blir brukt til en tilsvarende liste over dimensjoner.

Samlingsfunksjonen kan være en hvilken som helst funksjon som støttes av den underliggende databasen. De følgende samlingsfunksjonene støttes i DB2 Cube Views:

- AVG
- CORRELATION
- COUNT
- COUNT_BIG
- COVARIANCE
- MAX
- MIN

- REGRESSION-funksjoner (alle 9 typer)
- STTDEV
- SUM
- VARIANCE

Hver dimensjon kan bare samles én gang av målobjektet. Et mål må ha en samling med en tom liste over dimensjoner, og hver enkelt av de andre samlingene må ha en eksplisitt liste over dimensjoner. Samlingen for en tom liste over dimensjoner blir brukt til alle dimensjoner i kubemodellen som ikke spesifikt blir brukt av en annen samling.

Et eksempel på et enkelt mål som tilordnes direkte til en kolonne, er målet for inntekt, Revenue. Målet Revenue kan opprettes for en kubemodell med tre dimensjoner: Product, Market og Time. Revenue har en SQL-uttrykksmal template = "\$\$1" som representerer en enkel tilordning til kolonnen som er oppgitt i listen over kolonner, attributter og mål med bare en enkelt post, list = "Column Fact.Rev". Samlingslisten er (SUM, <NULL>), der SUM er samlingsfunksjonen, og <NULL> er en tom liste med dimensjoner. SQL-uttrykket brukes som inndata for samlingsfunksjonen SUM, som resulterer i dette SQL-uttrykket: SUM(Fact.Rev).

Et mer komplisert mål for fortjeneste, Profit, kan ha en SQL-uttrykksmal template = "\$\$1 - \$\$2", der listen over attributter, kolonner og mål er list = "Measure Revenue, Column Fact.Cost". Når symbolene blir erstattet de riktige referansene, blir SQL-uttrykket slik: "Revenue - Fact.Cost". Hvis inntektsmålreferansen utvides til kolonnereferansen, blir SQL-uttrykket slik: "Fact.Rev - Fact.Cost". Profit-målets samlingsliste er: (SUM, <NULL>). SQL-uttrykket for fortjeneste brukes som inndata for samlingsfunksjonen SUM, og dermed blir SQL-uttrykket for fortjenestemålet slik: SUM(Fact.Rev - Fact.Cost).

Hvis målet har en samlingsfunksjon, for eksempel CORRELATION, som krever to eller flere parametere, vil målet ha to eller flere SQL-uttrykk.

Mål har også en datatype som er basert på SQL-datatyper. DB2 Cube Views fastsetter automatisk datatypen for et mål. Hvert navn må, når det er fullstendig kvalifisert av et skjema, være entydig blant mål og attributter.

OLAP-senteret skjuler mye av kompleksiteten til metadataobjektdefinisjonen. I OLAP-senteret trenger du ikke å definere liste over SQL-uttrykk eller samlingsliste eksplisitt. Hvis du vil opprette et mål som tilordnes direkte til en kolonne, et attributt eller et annet mål, velger du kilden når du oppretter målet i faktaveiviseren eller vinduet Faktaegenskaper. Hvis du vil opprette et beregnet mål, kan du bruke vinduet Bygging av SQL-uttrykk til å opprette kildeuttrykket. Bygging av SQL-uttrykk inneholder lister over tilgjengelige kolonner, attributter og mål, operatorer, og funksjoner og konstanter. I vinduet Målegenskaper kan du vise datatypen for målets kildedata, og datatypen for målet etter at kildedataene er samlet.

Tabellen nedenfor beskriver de spesifikke egenskapene som definerer et mål. OLAP-senteret definerer hvert enkelt av disse når du oppretter et mål.

Tabell 14. Målegenskaper

Egenskap	Beskrivelse
Liste over SQL-uttrykk (mal, [(liste over kolonner, attributter og mål)])	En liste over SQL-uttrykk som brukes som inndata til den første samlingsfunksjonene for målet. Hvert SQL-uttrykk har en mal og en sortert liste over kolonner, attributter og mål.
Lister over samlinger (funksjoner, liste over dimensjoner)	Lister over samlinger som oppgir hvordan målet skal beregnes. Hver samling har en SQL-samlingsfunksjon og en valgfri liste over dimensjoner som funksjonen skal brukes til.
Datatype (skjema, navn, lengde, skala, kan ha nullverdier)	Fastsetter målets datatype. Datatypen er basert på SQL-datatyper, og er sammensatt av datatypens skjema, navn, lengde, skala, og om den kan inneholde nullverdier. OLAP-senteret viser bare skjemaet hvis det er et annet skjema enn SYSIBM.

Attributter

Et attributt representerer den grunnleggende abstraksjonen av en databasetabellkolonne. Et attributt inneholder et SQL-uttrykk som kan være enten en enkel tilordning til en tabellkolonne eller et mer komplekst uttrykk. Disse mer komplekse uttrykkene kan kombinere flere kolonner og attributter, og de kan bruke alle SQL-funksjonene, også brukerdefinerte funksjoner, ved behov.

OLAP-senteret for DB2 Cube Views skjuler mye av kompleksiteten i attributtobjektdefinisjonen. I OLAP-senteret trenger du ikke å eksplisitt definere attributtets uttrykksmal eller parameterliste. Hvis du vil opprette et attributt som tilordnes direkte til en kolonne, velger du kildekolonnen når du oppretter attributtet i dimensjonsveiviseren eller vinduet Dimensjonsegenskaper. Hvis du vil opprette et beregnet attributt, kan du bruke vinduet Bygging av SQL-uttrykk til å opprette kildeuttrykket. Bygging av SQL-uttrykk inneholder lister over tilgjengelige attributter, kolonner, operatører, funksjoner og konstanter.

Hvis du vil opprette et attributt uten å bruke OLAP-senteret, må du opprette attributtets SQL-uttrykksdefinisjon som en kombinasjon av to egenskaper, med en mal og en liste over kolonner og attributter. Malen bruker en symbolnotasjon der `$$$n` er symbolet og `n` refererer til en bestemt kolonne eller et bestemt attributt fra listen. Listen over kolonner og attributter er sortert, og en kolonnes eller et attributts plassering i listen tilsvarer symbolets `n`-verdi. Malen `template = "{{$1}} | ' ' | | {{$2}}"` kan for eksempel brukes med en tilsvarende liste som for eksempel `list = "Column CUSTOMER.FIRSTNAME, Attribute LastName"` for å sammenkjede kundenes fornavn og etternavn med et mellomrom mellom dem. Hvis malsymbolene erstattes med de riktige listereferansene, blir SQL-uttrykket `"Customer.FirstName | ' ' | | LastName"`. Attributtreferansen blir videre utvidet til en kolonnereferanse for å danne SQL-uttrykket: `"Customer.FirstName | ' ' | | Customer.LastName"`.

Når andre attributter brukes i SQL-uttrykket som definerer et attributt, kan ikke de andre attributtene danne attributtreferansesløyfer. Hvis Attributt A for eksempel refererer til Attributt B, kan Attributt B ikke referere til Attributt A.

Hvert navn må, når det er fullstendig kvalifisert av et skjema, være entydig blant alle andre attributter og mål i databasen.

Tabellen nedenfor beskriver de spesifikke egenskapene som definerer et attributt. OLAP-senteret definerer hvert av disse for deg når du oppretter et attributtobjekt.

Tabell 15. Attributtegenskaper

Egenskap	Beskrivelse
SQL-uttrykksmal	SQL-uttrykk som definerer attributtet. Malen refererer til kolonner og attributter ved å bruke en {\$n}-notasjon, der n er et ordenstall som samsvarer med listen over kolonner og attributter.
Liste over kolonner og attributter for SQL-uttrykk	Sortert liste over alle kolonner og attributter som attributtet består av. Disse kolonnene og attributtene brukes slik det er oppgitt i SQL-uttrykksmalen.
Datatype (skjema, navn, lengde, skala, kan ha nullverdier)	Bestemmer attributtets datatype. Datatypen er basert på SQL-datatyper, og er sammensatt av datatypens skjema, navn, lengde, skala, og om den kan inneholde nullverdier. OLAP-senteret viser bare skjemaet hvis det er et annet skjema enn SYSIBM.

Attributtforhold

Et attributtforhold beskriver generelle relasjonsforhold for attributter, men attributtforhold er ikke en del av kubemodellen.

Forholdene består av disse egenskapene:

- Et venstre- og et høyreattributt
- En type
- En kardinalitet
- En mulig funksjonell avhengighet

Typen beskriver hvilken rolle høyreattributtet spiller i forhold til venstreattributtet. Det finnes to mulige typer: Beskrivende og Tilknyttet.

Beskrivende

Oppgir at høyreattributtet er en beskriver av venstreattributtet. Høyreattributtet Produktnavn beskriver for eksempel venstreattributtet Produktkode.

Tilknyttet

Oppgir at høyreattributtet er knyttet til venstreattributtet, men det er ikke en beskriver av venstreattributtet. Høyreattributtet Befolkning er for eksempel knyttet til, men ikke en beskriver av By-ID.

Kardinaliteten beskriver hvordan forekomstene av venstre- og høyreattributtene er beslektet. Du kan bruke disse kardinalitetene for attributtforhold:

1:1 Det finnes maksimalt en venstreattributtforekomst for hver høyreattributtforekomst, og maksimalt en høyreattributtforekomst for hver venstreattributtforekomst.

1:mange

Det finnes maksimalt en venstreattributtforekomst for hver høyreattributtforekomst, og et hvilket som helst antall høyreattributtforekomster for hver venstreattributtforekomst.

Mange:1

Det finnes et hvilket som helst antall venstreattributtforekomster for hver høyreattributtforekomst, og maksimalt en høyreattributtforekomst for hver venstreattributtforekomst.

Mange:mange

Det finnes et hvilket som helst antall venstreattributtforekomster for hver høyreattributtforekomst, og et hvilket som helst antall høyreattributtforekomster for hver venstreattributtforekomst.

Egenskapen for funksjonell avhengighet forteller om attributtforholdet definerer et funksjonelt forhold mellom to attributter. Hvis du oppgir at et attributtforhold er en funksjonell avhengighet, betyr det at du garanterer at hver forekomst av venstreattributtet vil bestemme forekomsten av høyreattributtet. DB2 Cube Views oppretter ikke en funksjonell avhengighet mellom attributtene som er beskrevet av attributtforholdet, uansett hvordan du definerer den funksjonelle avhengighetsegenskapen.

Du kan definere flere attributtforhold som angir at det kan finnes en funksjonell avhengighet mellom Land-ID og Land, Delstats-ID og Delstat, og By-ID og By_Befolkning.

Du kan eksplisitt opprette et attributtforholdsobjekt i OLAP-senteret. Du kan eksplisitt opprette et attributtforhold som bruker veiviseren for attributtforhold. Åpne veiviseren for attributtforhold fra relasjonsobjektoversikten, og oppgi alle egenskapene for objektdefinisjonene.

De spesifikke egenskapene som definerer et attributtforholdsobjekt, beskrives i tabellen nedenfor.

Tabell 16. Egenskaper for attributtforhold

Egenskap	Beskrivelse
Venstreattributt	Venstreattributtet som er brukt i forholdet.
Høyreattributt	Høyreattributtet som er brukt i forholdet.
Type	Typen forhold som er beskrevet av attributtforholdet. Typen brukes til å fastsette hvilken rolle et attributt har, deskriptivt eller tilknyttet.
Kardinalitet	Forventet kardinalitet i kombineringen: 1:1, 1:mange, mange:1, mange:mange
Funksjonell avhengighet	Bestemmer om attributtforholdet også er en funksjonell avhengighet, ja eller nei. Merk: DB2 Cube Views oppretter ikke en funksjonell avhengighet for attributtforhold, uansett om du oppgir ja eller nei for denne egenskapen.

Kombineringer

En kombinerer er et metadataobjekt som beskriver en kombinerer mellom to relasjonstabeller. En kombinerer refererer til attributter som refererer til kolonner i tabellene som kombineres.

Den enkleste formen for kombinerings refererer til to attributter, ett som tilordnes til en kolonne i den første tabellen, og ett som tilordnes til en kolonne i den andre tabellen. Du oppgir også en operator som oppgir hvordan kolonnene skal sammenliknes.

En kombinerings kan også brukes til å modellere sammensatte kombinerings, der to eller flere kolonner fra den første tabellen kombineres med samme antall kolonner i den andre tabellen. En sammensatt kombinerings bruker par av attributter til å tilordne tilsvarende kolonner. Hvert par av attributter har en operator som oppgir hvordan dette paret av kolonner sammenliknes.

En kombinerings har også en type og en kardinalitet. Kombineringsstypene er tilordnet relasjonskombineringsstyper. Koblinger kan brukes i dimensjoner for å kombinere dimensjonstabeller, eller i en kubemodell for å kombinere kubemodellens dimensjoner til faktaobjektet, eller i et faktaobjekt for å kombinere flere faktatabeller. Du kan bruke kombineringsveiviseren i OLAP-senteret til å opprette en kombinerings.

De spesifikke egenskapene som definerer en kombinerings, beskrives i tabellen nedenfor.

Tabell 17. Kombineringsegenskaper

Egenskap	Beskrivelse
Liste over (venstreattributt, høyreattributt, operator)	Venstreattributt: Attributtet på venstre side av kombinerings. Høyreattributt: Attributtet på høyre side av kombinerings. Operator: Operatoren som er forventet i kombinerings =, <, >, <>, >=, <=.
Type	Forventet kombineringsstype: Inner, Full outer, Left outer, Right outer
Kardinalitet	Forventet kardinalitet i kombinerings: 1:1, 1:mange, mange:1, mange:mange

Kuber

En kube er en presis definisjon av en OLAP-kube som noen ganger kan leveres ved hjelp av en enkelt SQL-setning. En kube, som er utledet fra en kubemodell, inneholder et delsett av metadataobjekter som er basert på metadataobjektene i kubemodellen.

Kubefaktaobjektet og listen over kubedimensjoner er delsett av de som finnes i kubemodellen det refereres til. Kubene passer til verktøy og applikasjoner som ikke bruker flere hierarkier, fordi kubedimensjonene bare tillater ett kubehierarki per kubedimensjon.

Kuber kan brukes ved optimalisering av en kubemodell for å oppgi hvilke områder av kubemodellen som er mest aktive og viktigst. Du kan oppgi optimaliseringssektorer som definerer hvilke områder av kuben som blir hyppigst brukt i spørringer.

Du kan bruke kubeveiviseren i OLAP-senteret til å opprette en kube. Du må ha en fullstendig kubemodell for å kunne opprette en tilknyttet kube. Egenskapene for en kube beskrives i tabellen nedenfor.

Tabell 18. Kubeegenskaper

Egenskap	Beskrivelse
Kubemodell	Kubemodellen som kuben er utledet fra.
Kubefaktaobjekt	Kubefaktaobjekt som brukes i kuben. Kubefaktaobjektet er utledet fra faktaobjektet i kubemodellen.
Liste over kubedimensjoner	En sortert liste over kubedimensjonene som brukes i kuben. Hver kubedimensjon er utledet fra en tilhørende dimensjon i kubemodellen. Ett kubehierarki er knyttet til hver kubedimensjon.
Liste over optimaliseringssektorer	Sett med null eller flere optimaliseringssektorer. Hver optimaliseringssektor inkluderer: <ul style="list-style-type: none"> • Type kan være neddrilling, rapport, MOLAP-uttrekking, hybriduttrekking eller gjennomdrilling. • Sett med optimaliseringsnivåer der det finnes ett optimaliseringsnivå for hver kubedimensjon i kuben. Et optimaliseringsnivå refererer til: <ul style="list-style-type: none"> – En kubedimensjon og tilhørende kubehierarki – Et kubenivå, Vilkårlig eller Alle

Kubefaktaobjekt

Et kubefaktaobjekt har et delsett av mål i en sortert liste fra et bestemt faktaobjekt. Et kubefaktaobjekt gir en kube fleksibilitet til å finne omfanget av en kubemodells faktaobjekt. Den strukturelle informasjonen, som for eksempel kombineringsene og attributtene, er referert til fra det overordnede faktaobjektet.

I OLAP-senteret oppretter du en kube i forbindelse med en kubemodell, ved hjelp av kubeveiviseren. Du trenger ikke å definere kubefaktaobjektet eksplisitt, fordi OLAP-senteret vet at kubefaktaobjektet er utledet fra faktaobjektet i den tilknyttede kubemodellen. Du velger hvilke av målene fra kubemodellens faktaobjekt du vil bruke i kuben.

De spesifikke egenskapene som definerer et kubefaktaobjekt, beskrives i tabellen nedenfor.

Tabell 19. Egenskaper for kubefaktaobjekter

Egenskap	Beskrivelse
Faktaobjekt	Faktaobjekt som kubefaktaobjektet er utledet fra.
Liste over mål	En sortert liste over mål som brukes i en kube. Alle mål må være en del av faktaobjektet som kubefaktaobjektet er utledet fra.

Kubedimensjoner

En kubedimensjon brukes til å måle omfanget av en dimensjon for bruk i en kube. Kubedimensjonen refererer til dimensjonen den er utledet fra, og til det relevante kubehierarkiet for den gitte kuben.

Bare ett kubehierarki kan brukes til en kubedimensjon. Kombineringene og attributtene som gjelder for kubedimensjonen, er referert til fra dimensjonsdefinisjonen.

I OLAP-senteret oppretter du en kube i forbindelse med en kubemodell, ved hjelp av kubeveiviseren. Du velger hvilke av kubemodelldimensjonene du vil ha i kuben. For hver dimensjon du tar med som en kubedimensjon, kan du velge hvilke nivåer du vil ta med i kubehierarkiet.

De spesifikke egenskapene som definerer et kubedimensjonsobjekt, beskrives i tabellen nedenfor.

Tabell 20. Egenskaper for kubedimensjoner

Egenskap	Beskrivelse
Dimensjon	Dimensjonen som kubedimensjonen er utledet fra.
Kubehierarki	Kubehierarkiet som gjelder for kubedimensjonen.

Kubehierarkier

Et kubehierarki er et delsett av et hierarki, og det brukes i en kube. Et kubehierarki refererer til hierarkiet som det er utledet fra (overordnet hierarki), og det kan ha et sett med kubenivåer som er et delsett av de overordnede nivåene fra det overordnede hierarkiet.

En kubedimensjon kan bare ha ett kubehierarki. Generelt har et kubehierarki de samme hierarkitypene og distribueringsmekanismene som hierarkiet det er utledet fra. Hvis hierarkiet er av nettverkstypen, er kubehierarkiet balansert hvis det ikke mangler noen medlemmer, og ujevnt hvis det mangler medlemmer.

I OLAP-senteret oppretter du en kube i forbindelse med en kubemodell, ved hjelp av kubeveiviseren. Du velger hvilke av kubemodelldimensjonene du vil ha i kuben. For hver dimensjon du tar med som en kubedimensjon, kan du velge hvilke kubenivåer du vil ta med i kubehierarkiet.

De spesifikke egenskapene som definerer et kubehierarki, beskrives i tabellen nedenfor.

Tabell 21. Egenskaper for kubehierarkier

Egenskap	Beskrivelse
Hierarki	Hierarkiet som kubehierarkiet er utledet fra.
Sortert sett med kubenivåer	Sortert sett med ett eller flere kubenivåer, som er et delsett av nivåene som er tatt med i det overordnede hierarkiet. Rekkefølgen for kubenivåene må være den samme som i det overordnede hierarkiet.

Kubenivåer

Et kubenivå er et delsett av et nivå, og det brukes i en kube. En kubenivåreferanse refererer til nivået den er utledet fra (det overordnede nivået), og arver nivå-nøkkelattributtene og standardattributtet som er definert for det overordnede nivået.

Et kubenivå kan ha et sett med attributter som er et delsett av de beslektede attributtene fra det overordnede nivået.

Egenskapene for et kubenivå beskrives i tabellen nedenfor.

Tabell 22. Kubenivåegenskaper

Egenskap	Beskrivelse
Sett av attributter	Et sett med null eller flere beslektede attributter som er et delsett av det overordnede nivåets beslektede attributter.

Regler for metadataobjekter

Det finnes tre typer regler som gjelder for metadataobjekter: grunnleggende regler, fullstendighetsregler for kubemodeller og optimaliseringsregler. Disse reglene sikrer at hvert enkelt objekt er gyldig både innenfor og utenfor forbindelsen med en kubemodell, og at effektive SQL-spørringer kan skrives og optimaliseres.

Grunnleggende regler

De grunnleggende reglene definerer et objekts gyldighet når det ikke er i bruk. Hvert enkelt metadataobjekt har et eget sett med regler. Et metadataobjekt er gyldig hvis det følger alle de grunnleggende reglene.

Fullstendighetsregler

Fullstendighetsreglene gjelder bare for kubemodeller, og utvider de grunnleggende reglene for å sikre at en kubemodell kan kobles til andre metadataobjekter på riktig måte, og at det kan skrives effektive SQL-spørringer.

Optimaliseringsregler

Optimaliseringsreglene er en utvidelse av de grunnleggende reglene og fullstendighetsreglene for kubemodeller. Disse reglene sikrer at SQL-spørringene som er opprettet for metadataobjektene, kan optimaliseres.

Kubemodellregler

Grunnleggende regler:

- Parene av dimensjon og kombinerings må ha både en dimensjon og en kombinerings.
- Alle attributtene på en side av en kombinerings av dimensjoner og fakta må finnes i dimensjonens attributtliste, og alle attributtene på den andre siden av kombineringsen må finnes i faktaobjektets attributtliste.
- En kubemodell må referere til alle de eksplisitte dimensjonene det er referert til av samlingene av mål fra kubemodellens faktaobjekt. Hvis en kubemodell har dimensjoner, må en samling med en tom liste over dimensjoner samsvare med minst en dimensjon fra kubemodellen. Kontroller at det ikke er referert til

dimensjoner i andre samlinger av det samme målet. Hvis en kubemodell ikke har noen dimensjoner, må alle målene bare ha samlinger med en tom liste over dimensjoner.

Fullstendighetsregler:

- En kubemodell må referere til ett faktaobjekt.
- En kubemodell må referere til minst en dimensjon.

Optimaliseringsregler:

- Kombineringsregler som er brukt til å kombinere faktaobjektet og dimensjonen, må ha kardinaliteten 1:1 eller mange:1 og må kombinere en faktatabell med en dimensjons primærtabel.
- Minst en dimensjon i kubemodellen må ha minst ett hierarki.

Regler for faktaobjekter

Grunnleggende regler:

- Et faktaobjekt må referere til minst ett mål.
- Alle attributter og mål som det refereres til av et faktaobjekt, må kunne kombineres i faktaobjektet. Bare kombineringsreglene av faktaobjektet blir vurdert.
- Nøyaktig en kombineringsregel kan defineres mellom hvilke som helst to tabeller i faktaobjektet.
- Kombineringsregler er ikke tillatt i et faktaobjekt.
- Kombineringsregler som refererer til i et faktaobjekt, må referere til attributtene til faktaobjektet.

Regler for dimensjoner

Grunnleggende regler:

- En dimensjon må referere til minst ett attributt.
- Alle attributter og mål som det refereres til av en dimensjon, må kunne kombineres i dimensjonen. Bare kombineringsreglene av dimensjonen blir vurdert.
- Kombineringsregler er ikke tillatt i en dimensjon.
- Nøyaktig en kombineringsregel kan være definert mellom hvilke som helst to tabeller i dimensjonen.
- Kombineringsregler som refereres til av en dimensjon, må referere til attributtene til dimensjonen.
- Når det refereres til av en dimensjon, må referere til attributtene til dimensjonen.
- Hierarkier som refereres til av en dimensjon, må referere til nivåene til dimensjonen.

Optimaliseringsregel:

- En dimensjon må ha en primærtabel som det knyttes kombineringsregler til med kardinaliteten 1:1 eller mange:1.

Regler for nivåer

Grunnleggende regler:

- Et nivå må referere til minst ett nivånøkkelattributt.
- Et nivå må referere til minst ett standardattributt.

Regler for hierarkier

Grunnleggende regler:

- Et hierarki må referere til minst ett nivå.
- Nøyaktig to nivåer må finnes for en rekursiv distribuering.
- Standard distribuering kan brukes til alle typer hierarkier, og rekursiv distribuering kan bare brukes til ubalanserte hierarkier.

Regler for mål

Grunnleggende regler:

- Hver SQL-uttrykksmal kan ha null eller flere av de følgende parameterne: attributter, kolonner og mål.
- Attributter og mål som brukes som parametere for en SQL-uttrykksmal, kan ikke danne en avhengighetssløyfe.
- SQL-malen for et mål kan ikke være en tom streng.
- En SQL-mal kan ikke bruke samlingsfunksjoner.
- Hvis minst ett mål er referert til, og det bare er referert til mål, er det valgfritt å definere en samling.
- Antall SQL-maler må samsvare med antall parametere for den første samlingsfunksjonen, hvis det finnes en samling.
- Et mål med flere SQL-maler må definere minst ett trinn i samlingskriptet.
- Hvis et mål refererer til et andre mål som definerer flere SQL-maler, kan det refererende målet ikke ha et samlingskript.
- En samlingsfunksjon med flere parametere kan bare brukes i den første samlingen.
- Hvis et mål definerer en eller flere samlinger, må en samling oppgi en tom liste over dimensjoner.
- Et mål kan referere til hver dimensjon bare en gang enten i en samling eller i flere samlinger.
- I en SQL-mal må symbolindikatorer være nummerert fra og med 1, og nummereringen må være fortløpende.
- I et SQL-uttrykk må hver kolonne, hvert attributt og hvert mål være referert til minst en gang.

Regler for attributter

Grunnleggende regler:

- Hver SQL-mal kan ha null eller flere av de følgende parameterne: attributter og kolonner.
- Attributter som brukes som parametere for en SQL-uttrykksmal, kan ikke danne en avhengighetssløyfe.
- SQL-malen for et attributt kan ikke være en tom streng.
- SQL-malen kan ikke ha samlingsfunksjoner.
- I en SQL-mal må symbolindikatorer være nummerert fra og med 1, og nummereringen må være fortløpende.
- I et SQL-uttrykk må hver kolonne og hvert attributt være referert til minst en gang.

Regler for attributtforhold

Grunnleggende regler:

- Et attributtforhold må referere til to attributter.
- Et attributtforhold kan ikke defineres som en funksjonell avhengighet med kardinaliteten mange:mange.

Regler for kombineringsregler

Grunnleggende regler:

- En kombineringsregel må referere til minst en tripplett (venstreattributt, høyreattributt, operator).
- En gyldig operasjon må være definert for hver kombineringsregeltipp. Datatypene for venstre- og høyreattributtene må være kompatible med hverandre og med den oppgitte operasjonen.
- Alle venstreattributtene må tolkes til en eller flere kolonner i en enkelt tabell.
- Alle høyreattributtene må tolkes til en eller flere kolonner i en enkelt tabell.

Optimaliseringsregler:

- Det må defineres en begrensning for kolonnene som deltar i en kombineringsregel. Hvis det samme settet av kolonner brukes på begge sider av likhetstegnet, må det defineres en primærnøkkel som samsvarer med settet av kolonner. Hvis det brukes forskjellige sett av kolonner på hver side av likhetstegnet, må en side ha en samsvarende primærnøkkel, og den andre siden må ha en samsvarende fremmednøkkel og må referere til primærnøkkelens.
- Kombineringsreglets kardinalitet må være 1:1, mange:1 eller 1:mange. I en kombineringsregel med det samme settet av kolonner på begge sider av likhetstegnet, må kardinaliteten være 1:1. For alle andre kombineringsregler, må kardinaliteten ha 1 på siden med den definerte primærnøkkelens, og N på siden med den definerte fremmednøkkelens. Hvis fremmednøkkelens side også har en definert primærnøkkel, må kardinaliteten 1 brukes for denne siden.
- Alle attributter som brukes i kombineringsreglet, må tolkes til SQL-uttrykk som ikke kan ha nullverdier.
- Kombineringsreglens type må være en intern kombineringsregel.

Regler for kuber

Grunnleggende regler:

- En kube må referere til en kubemodell.
- En kube må referere til ett kubefaktaobjekt.
- En kube må referere til minst en kubedimensjon.
- Et kubefaktaobjekt må være utledet fra faktaobjektet som brukes i kubemodellen det er referert til.
- Alle kubemodeller må være utledet fra dimensjonene som brukes i kubemodellen det er referert til.
- En kube må ha null eller flere optimaliseringssektorer.
- En optimaliseringssektor må ha ett optimaliseringsnivå for hver kubedimensjon i kuben.
- En optimaliseringssektor må ha ett eller flere optimaliseringsnivåer.
- Et optimaliseringsnivå må ha en kubedimensjonsreferanse og en hierarkireferanse.

- Et optimaliseringsnivå må en referanse på nivået **Alle**, **Vilkårlig** eller på kubenivå.
- I et optimaliseringsnivåobjekt må kubedimensjonen og kubehierarkiet det refereres til, være utledet fra objekter i kubeobjektet. Kubehierarkiet må tilhøre kubedimensjonen.
- Hvis kubereferansen i et optimaliseringsnivåobjekt ikke er **Alle** eller **Vilkårlig**, må kubenivået tilhøre kubehierarkiet.
- En referanse til en kubedimensjon eller et kubehierarki i et optimaliseringsnivå skal ikke gjentas i en optimaliseringssektor.
- En optimaliseringssektor av typen MOLAP-uttrekking kan ikke eksistere hvis det finnes en annen optimaliseringssektor av typen hybriduttrekking, og omvendt.
- Hver kube har maksimalt en optimaliseringssektor av typen MOLAP-uttrekking.
- Hver kube har maksimalt en optimaliseringssektor av typen hybriduttrekking.
- En optimaliseringssektor av typen gjennomdrilling kan eksistere hvis det finnes en annen optimaliseringssektor av typen hybriduttrekking.

Regler for kubefaktaobjekter

Grunnleggende regler:

- Et kubefaktaobjekt må referere til ett faktaobjekt.
- Et kubefaktaobjekt må referere til minst ett mål.
- Alle mål det refereres til av et kubefaktaobjekt, må også refereres til i det tilsvarende faktaobjektet.

Regler for kubedimensjoner

Grunnleggende regler:

- En kubedimensjon må referere til en dimensjon.
- En kubedimensjon må referere til ett kubehierarki.
- Kubehierarkiet det refereres til, må være utledet fra et hierarki som brukes i dimensjonen det refereres til.

Regler for kubenivåer

Grunnleggende regler:

- Et kubenivå må referere til ett nivå.
- Et kubenivå må referere til minst ett beslektet attributt.
- Alle attributter det refereres til, må være beslektede attributter på nivået.

Regler for kubehierarkier

Grunnleggende regler:

- Et kubehierarki må referere til ett hierarki.
- Et kubehierarki må referere til minst ett kubenivå.
- Alle kubenivåer det refereres til av et kubehierarki, må være utledet fra nivåer som er en del av det tilhørende hierarkiet.
- Kubehierarkiets kubenivåer må være oppført i samme rekkefølge som i de tilsvarende nivåene i hierarkiet det er referert til. Kubenivårekkefølgen gjelder ikke for nettverkshierarkier.

Kapittel 3. Utforme DB2 Cube Views-metadatoobjektmodeller

Denne delen beskriver disse emnene:

Starte OLAP-senteret

Start OLAP-senteret slik at du kan administrere metadatoobjektene.

OLAP-senteret og API-kompatibilitet

DB2 Cube Views versjon har begrenset kompatibilitet med tidligere versjoner av OLAP-senteret og metadata-APIen.

Autorisasjoner og rettigheter for DB2 Cube Views

Når du skal utføre oppgaver med OLAP-senteret, må du ta kontakt med din DB2-databaseansvarlige (DBA) for å få nødvendige autorisasjoner og rettigheter, avhengig av operativsystemet.

Opprette DB2 Cube Views-metadatoobjekter

Du kan opprette DB2 Cube Views-metadatoobjektene med OLAP-senteret.

Fjerne en dimensjon fra en kubemodell

Du kan fjerne en dimensjon fra en kubemodell hvis du ikke lenger har behov for dimensjonen. Du kan fjerne en dimensjon uten å slette dimensjonen, hvis dimensjonen blir brukt av en annen kubemodell.

Slette et metadatoobjekt fra en database

Du kan slette et metadatoobjekt hvis du ikke lenger bruker metadatoobjektet i en kubemodell i denne databasen.

Starte og oppdatere OLAP-senteret

Start OLAP-senteret slik at du kan administrere metadatoobjektene.

Slik starter du OLAP-senteret:

1. Klikk på **Start** -> **Programmer** -> **IBM DB2** -> **Verktøy for Business Intelligence** -> **OLAP-senter**. Et databasetilkoblingsvindu blir åpnet.
2. I databasetilkoblingsvinduet logger du deg på den databasen du vil administrere metadata for.

OLAP-senteret viser et snapshot av metadatoobjektene i databasen. Selv om DB2 Cube Views alltid sørger for integriteten til metadatoobjektene programmet administrerer, blir ikke innholdet i OLAP-senter-vinduet oppdatert automatisk når en annen bruker av OLAP-senteret eller en API-applikasjon oppretter metadatoobjekter i databasen. Hvis en annen bruker eller en API-applikasjon endrer metadatoobjektene, kan du velge **Utsnitt** -> **Forny** for å se den nye statusen til databasen. Hvis flere brukere arbeider med samme metadatoobjekter samtidig, kan det oppstå feil fordi brukerne ikke ser de oppdaterte dataene i databasen. Flere brukere bør ikke arbeide med samme sett med metadatoobjekter samtidig.

Kompatibilitet mellom ulike OLAP-senter- og API-versjoner

DB2 Cube Views versjon 8.2 har begrenset kompatibilitet med tidligere versjoner av OLAP-senteret og metadata-APIen.

DB2 Cube Views API støtter følgende:

- Versjon 8.2 API støtter Describe-forespørsler fra en versjon 8.1-klient. Describe er den eneste versjon 8.1-operasjonen som støttes fra foregående utgave.

DB2 Cube Views OLAP-senter støtter følgende:

- Versjon 8.2 av OLAP-senteret har ikke lenger støtte for API eller metadatatabeller fra versjon 8.1.
- Versjon 8.1 av OLAP-senteret har ikke støtte for API eller metadatatabeller fra versjon 8.2.
- Versjon 8.1 av OLAP-senteret kan ikke kobles til en DB2-database fra versjon 8.2.
- OLAP-senteret støtter import av versjon 8.1 av XML. Når du importerer en XML-fil av versjon 8.1, migrerer OLAP-senteret XML-koden til versjon 8.2 ved hjelp av Translate-operasjonen før metadataene blir importert.
- OLAP-senteret kan eksportere XML i både versjon 8.2 og versjon 8.1.

DB2 Cube Views db2mdapiclient støtter følgende:

- Versjon 8.1 av db2mdapiclient støtter versjon 8.2 av APIen hvis du bruker versjon 8.2-XML.
- Versjon 8.2 av db2mdapiclient støtter versjon 8.1 av APIen hvis du bruker versjon 8.1-XML.

Autorisasjoner og rettigheter for bruk av DB2 Cube Views

Når du skal utføre oppgaver med OLAP-senteret, må du ta kontakt med din DB2-databaseansvarlige (DBA) for å få nødvendige autorisasjoner og rettigheter, avhengig av operativsystemet.

Med mindre annet er oppgitt, gjelder alle henvisninger til rettighetene SELECT, INSERT, UPDATE og DELETE i tabellene nedenfor, for tabeller i DB2INFO-skjemaet for databasen der DB2 Cube Views er installert.

Autorisasjoner og rettigheter for Windows

I Windows-operativsystemer gjelder autorisasjoner for oppgaver i OLAP-senteret for bruker-IDen som brukes til å koble til databasen.

Tabell 23. Generelle autorisasjoner og rettigheter for Windows

Oppgave	Autorisasjoner og rettigheter
Alle OLAP-senteroppgaver	<ul style="list-style-type: none"> • SYSADM eller DBADM for databasen der DB2 Cube Views er installert • EXECUTE for den lagrede prosedyren DB2INFO.MD_MESSAGE i databasen der DB2 Cube Views er installert (gjelder bare DB2 UDB versjon 8) • CONNECT for måldatabaser
Opprette metadataobjekter	SELECT, INSERT
Endre metadataobjekter	SELECT, INSERT, UPDATE, DELETE
Slette metadataobjekter	SELECT, DELETE
Eksportere metadataobjekter til en XML-fil	SELECT
Importere metadataobjekter til DB2 UDB fra en XML-fil	SELECT, INSERT, UPDATE
Kjøre optimaliseringsrådgiveren	SELECT for systemtabeller og basistabeller

Tabell 23. Generelle autorisasjoner og rettigheter for Windows (fortsettelse)

Oppgave	Autorisasjoner og rettigheter
Kjør DB2-kommandoene som er generert av optimaliseringsrådgiveren	<ul style="list-style-type: none"> • CREATEIN, DROPIN for skjemaet DB2INFO • SELECT og ALTER (eller CONTROL) for basistabeller

Noen konfigurerings- og installeringsoppgaver kan utføres med OLAP-senteret.

Tabell 24. Autorisasjoner og rettigheter for Windows-konfigurering og installering

Oppgave	Autorisasjoner og rettigheter
Alle konfigurerings- og installeringsoppgaver	<ul style="list-style-type: none"> • SYSADM for databasen der DB2 Cube Views er installert, for å opprette skjemaene for metadataobjekter • CONNECT, CREATETAB, IMPLICIT_SCHEMA for databasen der DB2 Cube Views skal installeres • CREATEIN, DROPIN for skjemaet DB2INFO i databasen der DB2 Cube Views er installert • SELECT, INSERT, UPDATE, DELETE, CONTROL for alle tabeller i DB2INFO-skjemaet for databasen der DB2 Cube Views er installert

AIX, autorisasjoner og rettigheter

I AIX er det to ulike typer bruker-IDer med to distinkte sett med autorisasjoner. En type bruker-ID må være konfigurert for å koble til en database og utføre den lagrede prosedyren DB2INFO.MD_MESSAGE. Nedenfor blir det henvist til denne typen bruker-ID som bruker-IDen for tilkobling.

Den andre typen bruker-ID må konfigureres for å kjøre alle andre oppgaver i OLAP-senteret. Denne typen bruker-ID blir nedenfor referert til som bruker-IDen for utføring. Denne bruker-IDen er en spesiell bruker-ID som er konfigurert i AIX for å kjøre de lagrede FENCED-prosedyrene. Lagrede FENCED-prosedyrer kjører under denne bruker-IDen, som er definert til eieren av .fenced-filen i sql11b/adm. Denne bruker-IDen kontrollerer systemressursene som er tilgjengelig for den lagrede prosedyren.

Tabell 25. Generelle autorisasjoner og rettigheter for bruker-IDen for tilkobling i AIX

Oppgave	Autorisasjoner og rettigheter
Alle OLAP-senteroppgaver	<ul style="list-style-type: none"> • SYSADM eller DBADM for databasen der DB2 Cube Views er installert • EXECUTE for den lagrede prosedyren DB2INFO.MD_MESSAGE i databasen der DB2 Cube Views er installert (gjelder bare DB2 UDB versjon 8) • CONNECT for måldatabaser

Tabell 26. Autorisasjoner og rettigheter for bruker-IDen for utføring i AIX

Oppgave	Autorisasjoner og rettigheter
Opprette metadataobjekter	SELECT, INSERT
Endre metadataobjekter	SELECT, INSERT, UPDATE, DELETE
Slette metadataobjekter	SELECT, DELETE
Eksportere metadataobjekter til en XML-fil	SELECT
Importere metadataobjekter til DB2 UDB fra en XML-fil	SELECT, INSERT, UPDATE, DELETE
Kjøre optimaliseringsrådgiveren	SELECT for systemtabeller og basistabeller
Kjør DB2-kommandoene som er generert av optimaliseringsrådgiveren	<ul style="list-style-type: none"> • CREATEIN, DROPIN for skjemaet DB2INFO • SELECT og ALTER (eller CONTROL) for basistabeller

Noen konfigurerings- og installeringsoppgaver kan utføres med OLAP-senteret. Disse oppgavene krever ekstra autorisasjoner for bruker-IDen for tilkobling.

Tabell 27. Autorisasjoner og rettigheter for konfigurering og installering i AIX for bruker-IDen for tilkobling

Oppgave	Autorisasjoner og rettigheter
Alle konfigurerings- og installeringsoppgaver	<ul style="list-style-type: none"> • SYSADM for databasen der DB2 Cube Views er installert, for å opprette skjemaene for metadataobjekter • CONNECT, CREATETAB, IMPLICIT_SCHEMA for databasen der DB2 Cube Views skal installeres • CREATEIN, DROPIN for skjemaet DB2INFO i databasen der DB2 Cube Views er installert • SELECT, INSERT, UPDATE, DELETE, CONTROL for alle tabeller i DB2INFO-skjemaet for databasen der DB2 Cube Views er installert

Opprette DB2 Cube Views-metadataobjekter

Du kan opprette DB2 Cube Views-metadataobjektene med OLAP-senteret.

Utveksle metadata mellom DB2 Cube Views og OLAP-verktøy

Bruk OLAP-senteret til å koble til en DB2 UDB-database og importere eller eksportere metadataobjekter.

Koble til en DB2-database

Du kan koble til eller fra en DB2-database.

Før du kobler til en DB2-database ved hjelp av OLAP-senteret, må du konfigurere databasen for bruk med metadata fra DB2 Cube Views. Hvis databasen du prøver å koble deg til, ikke er konfigurert for bruk med metadata for DB2 Cube Views, får du en melding når du prøver å koble deg til. OLAP-senteret kan utføre konfigureringsoppgaver hvis bruker-IDen du bruker for å koble til, har de riktige autorisasjonene og rettighetene.

Merk: Hvis du allerede er koblet til en database, og du deretter kobler deg til en annen database, blir du koblet fra den opprinnelige databasen. Ingen av metadataobjektene i den opprinnelige databasen blir lenger vist i objektoversikten i OLAP-senteret.

Slik kobler du til en DB2-database:

1. Åpne vinduet for databasetilkobling ved å klikke på **OLAP-senter** → **Tilkoble**, og oppgi nødvendig informasjon.
 - I feltet **Databasenavn** velger du databasen du vil koble deg til.
 - I feltet **Bruker-ID** oppgir du bruker-IDen for den databasen du har oppgitt.
 - I feltet **Passord** oppgir du passordet for den bruker-IDen du har oppgitt.
2. Klikk på **OK**. Metadataobjektene i DB2-databasen du koblet til, blir vist i objektoversikten i OLAP-senteret.

Slik kobler du fra en DB2-database:

Klikk på **OLAP-senter** → **Frakoble** fra hovedvinduet i OLAP-senteret. Alle metadataobjekter blir fjernet fra objektoversikten for OLAP-senteret.

Importere metadata

Du kan importere metadataobjektene til DB2 UDB slik at du kan administrere metadataobjektene ved hjelp av DB2 Cube Views.

Før du importerer metadataobjekter til DB2 UDB, må du forsikre deg om at du har en XML-fil for DB2 Cube Views. Du kan opprette en XML-fil for DB2 Cube Views ved å bruke en metadatabro til applikasjon, ved å eksportere metadata fra en eksisterende DB2 UDB-database eller ved å bruke en XML-redigeringsprogram.

Du kan importere XML-filer for DB2 Cube Views versjon 8.1 og versjon 8.2. Importveiviseren kan konvertere metadataobjekter i versjon 8.1 til versjon 8.2 når importveiviseren oppretter objektene.

Slik importerer du metadata:

1. Åpne importveiviseren ved å klikke på **OLAP-senter** → **Importer**. Importveiviseren blir åpnet.
2. På siden Kilde oppgir du XML-filen med metadatainformasjonen du vil importere.
3. På siden Importalternativer ser du gjennom listen over metadataobjekter som skal importeres, og oppgir importalternativer. Du kan se om et objekt finnes eller om det vil bli opprettet etter at du har importert objektene.
4. På siden Sammendrag ser du gjennom valgene du har foretatt for metadataimporten.
5. Klikk på **Fullfør** for å importere metadataobjektene. Objektene du importerte, blir vist i objektoversikten i OLAP-senteret.

Eksportere metadata

Du kan eksportere metadataobjekter i DB2 Cube Views slik at du kan bruke metadataobjektene i applikasjoner fra forretningspartnere.

Før du eksporterer metadataobjekter til en XML-fil for DB2 Cube Views XML, må du opprette metadataobjektene du vil eksportere.

Slik eksporterer du metadata:

1. Åpne eksportvinduet ved å klikke på **OLAP-senter -> Eksporter**.
2. Velg hvilken versjon av metadata du vil eksportere. Hvis du bruker en bro til et verktøy fra andre leverandører, må du kontrollere hvilken versjon av metadataene dette verktøyet støtter.
3. Velg **Alle metadataobjekter** for å eksportere alle metadataobjekter i databasen, eller **Valgte metadataobjekter** for å oppgi et bestemt sett med metadataobjekter som skal eksporteres. Du kan velge en kubemodell eller en eller flere kuber med samme overordnede kubemodell. Hvis du velger en eller flere kuber, blir den overordnede kubemodellen eksportert automatisk.
4. Oppgi navnet på XML-filen som det skal eksporteres til. Hvis filen finnes allerede, blir den overskrevet.
5. Klikk på **OK**. Det blir opprettet en XML-fil for DB2 Cube Views som inneholder informasjon om metadataobjektene du oppgav. Du kan bruke XML-filen for DB2 Cube Views XML med applikasjoner fra forretningspartnere.

Opprette en kubemodell ved hjelp av hurtigstartveiviseren

Du kan opprette en kubemodell og tilhørende faktaobjekt, mål, dimensjoner, attributter og kombineringer samtidig på grunnlag av relasjonsskjemaet.

Før du kan opprette en kubemodell og de tilhørende metadataene, må du definere referanseintegritetsbegrensningene for tabellene i databasen.

Hurtigstartveiviseren vil opprette de metadataobjektene den kan utlede logisk fra skjemaet. Du oppgir faktatabellen og målkolonnene, og veiviseren vil finne de tilsvarende dimensjonene, kombineringsene og attributtene. Når du har fullført hurtigstartveiviseren, kan du tilføye, slette og endre metadataobjektene slik du ønsker.

Slik oppretter du en kubemodell og de tilhørende metadataene:

1. Åpne hurtigstartveiviseren ved å høyreklikke på databasen eller på mappen **Kubemodeller** i OLAP-senterets objektoversikt, og deretter klikke på **Opprett kubemodell - Hurtigstart**. Hurtigstartveiviseren blir åpnet. Hvis objektoversikten ikke inneholder mappen **Kubemodeller**, må du kontrollere at du er i OLAP-objektoversikten ved å klikke på **Vis -> Vis OLAP-objekter**. Innledningssiden beskriver hva hurtigstartveiviseren kan gjøre.
2. På siden Faktatabell velger du skjemaet og faktatabellen for kubemodellen. Faktatabellen du velger, vil tilordne direkte til faktaobjektet i kubemodellen. Kolonnene som brukes som fremmednøkler i den oppgitte faktatabellen, vil tilordne direkte til attributtobjektene i kubemodellen. Veiviseren kan ikke oppdage implisitte dimensjoner (dimensjoner med kolonner som finnes i faktatabellen). Du kan tilføye implisitte dimensjoner til kubemodellen etter at du har fullført hurtigstartveiviseren.
3. På siden Mål velger du kolonner fra den oppgitte faktatabellen som du vil tilordne direkte til mål. Primærnøkler og fremmednøkler kan ikke tilordnes til mål, og er ikke tatt med på listen. Standard samlingsfunksjon er SUM for mål som er basert på numeriske kolonner, og COUNT for mål som er basert på tegnkolonner. Du kan ikke opprette beregnede mål i denne veiviseren. Etter at du har fullført hurtigstartveiviseren, kan du opprette beregnede mål.
4. På siden Sammendrag kan du vise metadataobjektene som vil bli opprettet. Klikk på **Fullfør** for å opprette kubemodellen og de tilhørende metadataobjektene. Når du klikker på **Fullfør**, oppretter hurtigstartveiviseren disse metadataobjektene i DB2-katalogen:
 - En kubemodell som inneholder alle de andre metadataobjektene.

- Et faktaobjekt som tilsvare faktatabellen du har oppgitt.
- Mål som tilsvare faktatabellkolonnene du har oppgitt.
- Dimensjoner som tilsvare hver dimensjonstabell som er kombinert med faktatabellen. Utriggertabeller er tabeller som er kombinert med en dimensjonstabell, og som er tatt med i det aktuelle dimensjonsobjektet.
- Attributter som samsvarer med hver kolonne i dimensjons- og uttriggertabellen, og med eventuelle fremmednøkler i faktatabellen.
- Kombineringsobjekter som tjener som kombineringer mellom faktaobjekter og dimensjoner, og kombineringer i et dimensjonsobjekt som kombinerer dimensjonstabellen og eventuelle tilsvarende uttriggertabeller.

Du kan tilføye hierarkier, beregnede mål og kuber til kubemodellen.

Opprette en fullstendig kubemodell

Når du skal opprette en fullstendig kubemodell, oppretter du en tom kubemodell og tilføyer deretter et faktaobjekt, dimensjoner, samt hierarkier og nivåer for hver dimensjon i kubemodellen. Kubemodeller definerer forholdene mellom relasjonsdata i stjerneskjemaet eller snøfnuggskjemaet, slik at du kan optimalisere spørringer som sendes til relasjonsdataene.

Du kan opprette en tom kubemodell, eller du kan følge trinnene nedenfor for å opprette alle objektene som er nødvendig for å fullføre kubemodellen. Når du har fullført kubemodellen, kan du opprette kubeobjekter som kan inneholde alle eller et delsett av kubemodellobjektene egenskaper.

Kubemodellen i DB2 Cube Views er en representasjon av et logisk stjerneskjema eller snøfnuggskjema. Kubemodellen er en gruppering av relevante dimensjonsobjekter rundt et sentralt faktaobjekt. Hver dimensjon kan ha flere hierarkier, noe som øker kubemodellens fleksibilitet. Den strukturelle informasjonen om kombineringen av tabellene som brukes av fakta- og dimensjonsobjektene, er lagret i kubemodellen. I kubemodellen er det også lagret nok informasjon til å hente OLAP-data. Andre rapporterings- og OLAP-verktøy som forstår kubemodellen og kan vise flere utsnitt av en bestemt dimensjon, kan ha fordeler av at det er definert flere hierarkier for en dimensjon.

Slik oppretter du en fullstendig kubemodell:

1. Opprett en kubemodell.
2. Opprett et faktaobjekt.
3. Opprett en dimensjon.
4. Opprett et hierarki for en dimensjon.
5. Opprett nivåer for hvert hierarki.

Opprette en kubemodell

Du oppretter en kubemodell for å gruppere relevante dimensjonsobjekter rundt et sentralt faktaobjekt.

Den strukturelle informasjonen om kombineringen av tabellene som brukes av fakta- og dimensjonsobjektene, er lagret i kubemodellen. Du kan opprette kuber med delsett av objekter fra en kubemodell slik at du kan avsløre relevante faktaobjekter og dimensjoner du har valgt, i en applikasjon.

Kubemodellveiviseren leder deg gjennom trinnene for å opprette en tom kubemodell. Når du har opprettet en kubemodell ved hjelp av veiviseren, tilføyer

du et faktaobjekt og dimensjoner med tilhørende kombineringsobjekter. Dimensjoner kan deles mellom forskjellige kubemodeller.

Slik oppretter du en kubemodell:

1. Åpne kubemodellveiviseren ved å høyreklikke på databasen eller mappen **Kubemodeller** i OLAP-senterets objektoversikt, og klikke på **Opprett kubemodell**. Hvis objektoversikten ikke inneholder mappen **Kubemodeller**, må du kontrollere at du er i OLAP-objektoversikten ved å klikke på **Vis -> Vis OLAP-objekter**.
2. På siden **Navn** oppgir du et navn, forretningsnavn, skjema og kommentarer for kubemodellen.
3. Klikk på **Fullfør** for å opprette kubemodellen.

Opprette et faktaobjekt

Du oppretter et faktaobjekt som en del av en eksisterende kubemodell. Et faktaobjekt grupperer beslektede mål som er interessante for en gitt applikasjon. Et faktaobjekt brukes i en kubemodell som sentrum i et stjerneskjema. Du kan opprette enkle og beregnede mål.

Før du kan opprette et faktaobjekt, må du opprette en kubemodell.

Slik oppretter du et faktaobjekt:

1. Åpne faktaveiviseren ved å høyreklikke på en kubemodell og klikke på **Opprett faktaobjekt**. Hvis alternativet **Opprett faktaobjekt** ikke er tilgjengelig, er det allerede definert et faktaobjekt for den valgte kubemodellen.
2. På siden **Navn** oppgir du et navn, forretningsnavn og skjema for faktaobjektet. Du kan også skrive kommentarer til faktaobjektet. Alle beregnede og ikke-beregnede mål du oppretter i faktaveiviseren, vil ha samme skjemanavn som du oppretter for faktaobjektet.
3. På siden **Tabeller** velger du en eller flere faktatabeller fra listen over tilgjengelige tabeller.
4. På siden **Kombineringer** oppgir du kombineringer mellom tabellene du har valgt. Du kan velge en eksisterende kombinerings, eller du kan klikke på **Opprett kombinerings** for å åpne en veiviser der du kan opprette en ny kombinerings. Alle de valgte tabellene må være kombinert uten sløyfer. Du kan bare oppgi en kombinerings mellom hvert par av tabeller. Hvis du bare valgte en tabell på siden **Tabeller**, trenger du ikke å oppgi kombinerings.
5. På siden **Mål** oppretter du enkle mål, beregnede mål eller begge deler.

Enkle mål

Mål som er tilordnet direkte til en kolonne. Hvis du flytter en kolonne til listen over valgte mål, blir det opprettet et enkelt mål som tilordnes direkte til kolonnen.

Beregnede mål

Mål som refererer til et uttrykk og som blir opprettet fra en eller flere kolonner, attributter eller andre mål. Du oppretter et beregnet mål ved å klikke på **Opprett beregnet mål**.

6. På siden **Samlinger** oppgir du en enkelt samlingsfunksjon for hvert mål. Standard samlingsfunksjon er SUM for mål med en numerisk datatype. For mål med en tegndatatype, er standard samlingsfunksjon COUNT. Hvis du vil endre samlingen, klikker du på samlingen i tabellen og velger en annen funksjon eller ingen.
7. Klikk på **Fullfør** for å opprette faktaobjektet.

Opprette en dimensjon

Du kan opprette en dimensjon som en del av en kubemodell, eller du kan opprette en dimensjon du vil bruke senere.

Før du kan opprette en dimensjon for en kubemodell, må du opprette et faktaobjekt.

Et dimensjonsobjekt definerer et sett av beslektede attributter og kombineringer av attributtene som sammen beskriver et aspekt av et mål. Når det er nødvendig med kombineringer for å gruppere attributter, for eksempel med et snøfnuggskjema, er de nødvendige kombineringsene og attributtene oppført i definisjonen av dimensjonen. En dimensjon med navnet Region kan inneholde attributter som Delstat, Land, By og Befolkning. Dimensjoner refererer til hierarkier som kan brukes til å drive navigeringen og beregningen av dimensjonen. Dimensjoner har en type som beskriver beskaffenheten av dimensjonen.

Slik oppretter du en dimensjon:

1. Åpne dimensjonsveiviseren ved å høyreklikke på en kubemodell og klikke på **Opprett dimensjon**.
2. På siden Navn oppgir du navn, forretningsnavn, skjema og kommentarer for dimensjonen. Alle attributter du oppretter i dimensjonsveiviseren, vil ha samme skjemanavn som du oppgir for dimensjonen.
3. På siden Tabeller velger du kildetabeller som du vil opprette dimensjonen fra. Du må velge minst en tabell.
4. På siden Kombineringer oppgir du kombineringer mellom tabellene du har valgt. Du kan velge en eksisterende kombinerings, eller du kan klikke på **Opprett kombinerings** for å åpne en veiviser slik at du kan opprette en kombinerings. Alle de valgte tabellene må være kombinert uten sløyfer. Du kan bare oppgi en kombinerings mellom hvert sett med to tabeller. Hvis du bare har valgt en tabell på siden **Tabeller**, trenger du ikke å oppgi kombinerings.
5. På siden Attributter oppgir du attributtene som skal tas med i dimensjonen. Du må oppgi minst ett attributt. Du kan velge et attributt fra listen, eller du kan klikke på **Opprett beregnet attributt** for å åpne et vindu der du kan opprette et uttrykk. Objektoversikten viser kildetabeller med de tilgjengelige kolonnene og attributtene. Hvis det allerede finnes et attributt i databasen som er tilordnet til en av kolonnene i tabellen, blir attributtet vist. Hvis en kolonne ikke er tilordnet til et attributt, blir selve kolonnen vist. Hvis en kolonne blir valgt og flyttet til listen **Valgte attributter**, blir det opprettet et attributt som er tilordnet til kolonnen. Velg alle attributtene du vil bruke i hierarkiet eller hierarkiene i dimensjonen. OLAP-senteret vil automatisk tilføye eventuelle ID-attributter som brukes av de oppgitte kombineringsene, i dimensjonen.
6. På siden Type oppgir du typen dimensjon du vil opprette. Velg **Tid** hvis du vil oppgi at dimensjonen er en tidsdimensjon, eller velg **Vanlig** hvis du vil oppgi at dimensjonen ikke er en tidsdimensjon.
7. På siden Velg kombinerings av faktaobjekt og dimensjon oppgir du et kombineringsobjekt for å kombinere dimensjonen med faktaobjektet. Du kan også opprette kombinerings med kombineringsveiviseren. Denne siden blir bare vist hvis du oppretter en dimensjon som en del av en kubemodell.
8. Klikk på **Fullfør** for å opprette dimensjonen.

Opprette et hierarki for en dimensjon

Et hierarki definerer forhold mellom to eller flere nivåer i en gitt dimensjon av en kubemodell. Du kan også definere et hierarki som bare bruker ett nivå. Definerings

av disse forholdene gir en navigerings- og behandlingsmetode for å krysse den oppgitte dimensjonen. Hierarkiet Kalenderår kan for eksempel inneholde nivåene År, Kvartal og Måned.

Før du oppretter et hierarki for en dimensjon, må du opprette en dimensjon.

Du kan definere flere hierarkier for en dimensjon i en kubemodell. Forholdet mellom nivåene bestemmes av hierarkitypen.

Slik oppretter du et hierarki for en dimensjon:

1. Åpne hierarkiveviseren ved å utvide mappen **Dimensjoner** slik at du ser de eksisterende dimensjonene. Høyreklikk på en dimensjon, og klikk deretter på **Opprett hierarki**.
2. På siden Navn oppgir du navn, forretningsnavn, skjema og eventuelle kommentarer for hierarkiet.
3. På siden Nivåer oppgir du nivåer for hierarkiet og hierarkitypen:
 - a. Hvis det ikke finnes noen nivåer på listen **Tilgjengelige nivåer**, klikker du på **Opprett nivå** og bruker veiviseren Opprett nivå til å opprette nivåer.
 - b. Velg de nivåene du ønsker. Du må velge minst ett nivå.
 - c. Oppgi typen og distribueringen for hierarkiet. Hvis du oppgir en **rekursiv** hierarkitype, kan du bare velge to nivåer.

Når du har valgt minst ett nivå, kan du klikke på **Vis eksempel** for å vise eksempeldata i hierarkiet.

4. Klikk på **Fullfør** for å tilføye det nye hierarkiet i kubemodellen.

Opprette nivåer for hvert hierarki

Nivåer definerer forhold mellom et sett med beslektede attributter. Når det er mulig, oppretter DB2 Cube Views funksjonelle avhengigheter som definerer forholdet mellom nivåattributtene.

1. Åpne nivåveiviseren ved å utvide en dimensjon under mappen **Dimensjoner** for en kubemodell i objektoversikten i OLAP-senteret. Høyreklikk på mappen **Nivåer** og klikk på **Opprett nivå**.
2. På siden Navn oppgir du navn, forretningsnavn, skjema og kommentarer for kubemodellen.
3. På siden Nivånøkkelattributter velger du ett eller flere attributter som samlet definerer nivået entydig.
4. På siden Standardattributt velger du et attributt som beskriver nivåets data.
5. På siden Beslektede attributter velger du null eller flere attributter som gir mer informasjon om nivået.
6. På siden Optimaliser med funksjonell avhengighet oppgir du om du ønsker at det skal opprettes funksjonelle avhengigheter mellom nivånøkkelattributtene og standard- og beslektede attributter. Du bør bare velge dette alternativet hvis du er sikker på at nivånøkkelattributtene funksjonelt bestemmer både standardattributtet og eventuelle beslektede attributter.
7. Klikk på **Fullfør** for å opprette nivået.

Tilføye en eksisterende dimensjon til en kubemodell

Du kan tilføye en eksisterende dimensjon til en kubemodell. Det kan allerede finnes en dimensjon hvis du bruker dimensjonen i en annen kubemodell, eller hvis du har brukt dimensjonen for en annen kubemodell, men senere har fjernet dimensjonen.

Før du tilføyer en dimensjon, må du opprette en kubemodell og et faktaobjekt. Du må også ha en eksisterende dimensjon som ikke er en del av kubemodellen.

Slik tilføyer du en eksisterende dimensjon:

1. Åpne veiviseren Tilføy dimensjon ved å høyreklikke på en kubemodell og klikke på **Tilføy dimensjoner**.
2. På siden Dimensjoner velger du en eller flere eksisterende dimensjoner som du vil tilføye i kubemodellen. Du må velge minst en dimensjon.
3. På siden Kombineringer av faktaobjekt og dimensjon oppgir du et kombineringsobjekt for å kombinere den tilføyde dimensjonen med faktaobjektet i kubemodellen.

Hvis det finnes noen hensiktsmessig eksisterende kombineringsobjekt for en dimensjon, vises denne kombineringsobjektet for den tilhørende dimensjonen. Kontroller at standardkombineringen virker fornuftig, at attributtene på den ene siden av kombineringsobjektet refererer til kolonner i den tilhørende dimensjonstabellen, og at attributtene på den andre siden av kombineringsobjektet refererer til kolonner i faktatabellen.

Hvis det ikke finnes noen hensiktsmessige eksisterende kombineringsobjekt, eller hvis det finnes flere enn en, må du oppgi en kombineringsobjekt. Hvis du skal oppgi en annen kombineringsobjekt eller opprette en kombineringsobjekt for en bestemt dimensjon, velger du dimensjonen og klikker på **Oppgi kombineringsobjekt**.

4. Klikk på **Fullfør** for å tilføye dimensjonen til kubemodellen.

Oppgi en kombineringsobjekt mellom faktaobjekt og dimensjon for en eksisterende dimensjon

Du kan oppgi kombineringsobjekt mellom faktaobjekt og dimensjon for en eksisterende dimensjon til en kubemodell.

- Slik oppgir du en eksisterende kombineringsobjekt:
 1. Velg en kombineringsobjekt fra listen over kombineringsobjekt.
 2. Klikk på **OK**.
- Slik oppretter du en kombineringsobjekt:
 1. Klikk på **Opprett kombineringsobjekt**. Kombineringsveiviseren blir åpnet.
 2. Opprett den kombineringsobjekt du ønsker. Den nye kombineringsobjekt blir vist i listen over eksisterende kombineringsobjekt.
 3. Velg kombineringsobjekt du opprettet fra listen over kombineringsobjekt.
 4. Klikk på **OK**.

Opprette en kombineringsobjekt

Du kan opprette en kombineringsobjekt for å kombinere en dimensjon til et faktaobjekt, som en del av et faktaobjekt, eller som en del av en dimensjon.

Et kombineringsobjekt kombinerer to relasjonstabeller. En kombineringsobjekt refererer til attributter som igjen refererer til kolonner i tabellene som kombineres. En kombineringsobjekt har også en type og en kardinalitet.

Den enkleste formen for kombineringsobjekt refererer til to attributter, ett som tilordner til en kolonne i den første tabellen og ett som tilordner til en kolonne i den andre tabellen, sammen med en operator for å oppgi hvordan kolonnene skal sammenliknes.

Et kombineringsobjekt kan også brukes til å modellere sammensatte kombineringsobjekt, der to eller flere kolonner fra den første tabellen kombineres med

det samme antall kolonner i den andre tabellen. En sammensatt kombinerings bruker par av attributter til å tilordne tilsvarende kolonner. Hvert par av attributter har en operator som oppgir hvordan dette paret av kolonner skal sammenliknes.

Kombineringsobjekter brukes først og fremst i en kubemodell for å kombinere kubemodellens dimensjoner til kubemodellens faktaobjekt. Kombineringer kan også brukes til å kombinere dimensjonstabeller i et snøfnuggskjema, eller brukes noen ganger i et faktaobjekt for å kombinere flere faktatabeller.

Slik oppretter du en kombineringsveiviseren:

1. Åpne kombineringsveiviseren.
2. På siden Navn oppgir du navn, forretningsnavn, skjema og eventuelle kommentarer for kombineringsveiviseren.

Tips: Bruk et navn på kombineringsveiviseren som inneholder navnene på begge tabellene som kombineres. Hvis du for eksempel kombinerer tabellene Salgsfakta og Produkt, kan du kalle kombineringsveiviseren Salgsfakta-Produkt.

3. På siden Kombineringer tilføyer du ett eller flere attributtpar og velger kombineringsoperator og kardinalitet for kombineringsveiviseren. Opprett et attributtpar ved å velge et venstreattributt og et høyreattributt, og deretter klikke på **Tilføy**. Attributtparet blir vist i tabellen over attributtpar. Standard kombineringsoperator er =. Du kan endre operatoren ved å klikke på den gjeldende operatoren i tabellen, og deretter velge en ny operator. Standard kombineringsoperator er Intern, og standard kardinalitet er 1:1.
4. Klikk på **Fullfør** for å opprette kombineringsveiviseren.

Opprette en kube

Du kan opprette en kube for å oppgi områder i kubemodellen som er viktige. Du kan også bruke en kube for å definere et delsett av dataene for en partnerapplikasjon.

Før du kan opprette en kube, må du opprette eller importere en kubemodell. Kubemodellen må ha et faktaobjekt og minst en dimensjon, og hver dimensjon må ha ett hierarki.

En kube er en nøyaktig definisjon av en OLAP-kube som det kan utføres spørringer mot ved hjelp av en enkelt SQL-setning. En kube er utledet fra en eksisterende kubemodell. Kubefaktaene og listen over kubedimensjoner er delsett av faktaene og dimensjonene som finnes i kubemodellen det refereres til. Kubene passer til verktøy og applikasjoner som ikke bruker flere hierarkier, fordi kubedimensjonene bare tillater ett kubehierarki per kubedimensjon.

Slik oppretter du en kube:

1. Åpne kubeveiviseren ved å høyreklikke på en kubemodell og klikke på **Opprett kube**.
2. På siden Navn oppgir du navn, forretningsnavn, skjema og kommentarer for kuben.
3. På siden Mål velger du hvilke mål som skal tas med i kuben. Du må velge minst ett mål.
4. På siden Dimensjoner velger du hvilke kubedimensjoner som skal tas med i kuben. Du må velge minst en kubedimensjon. Velg en kubedimensjon og klikk på skjermknappen ved siden av den valgte kubedimensjonen for å åpne et vindu slik at du kan oppgi hierarkiinformasjon for kubedimensjonen.

5. På siden Oppgi spørringstype oppgir du hvordan du forventer å bruke denne kuben. Hvis du velger **Avanserte innstillinger**, klikker du på **Spesifiser** for å oppgi de optimaliseringssektorene du forventer flest spørringer for i denne kuben.
6. Klikk på **Fullfør** for å opprette kuben.

Oppgi et kubehierarki og kubenivåer for en kubedimensjon

Du kan opprette tilpassede kuber for ulike applikasjoner ved å oppgi kubenivåene og beslektede attributter som skal tas med i et kubehierarki for hver kubedimensjon. Du kan oppgi et delsett med kubenivåer og beslektede attributter som skal referere til det i kubehierarkiet.

Slik oppgir du et kubehierarki for en kubedimensjon fra vinduet Velg attributter for kubehierarkiet, som du åpner fra kubeveiviseren:

1. Velg et kubehierarki fra listen **Mulige hierarkier**.
2. På listen **Nivåer og attributter** kontrollerer du at valgruten ved siden av kubenivået eller det beslektede attributtet du vil bruke, er merket av. Nivånøkkelattributter og standardattributter blir tatt med når du velger et nivå.
3. Klikk på **OK** for å tilføye kubenivåene og de beslektede attributtene til kubehierarkiet og gå tilbake til kubeveiviseren.

Angi kubeoptimaliseringssektorer

Du kan oppgi optimaliseringssektorer for en kube som et valgfritt, men kraftig for optimaliseringsrådgiveren til å opprette sammendragstabeller som er fokusert på de viktigste områdene i kubemodellen.

Optimaliseringssektorer angir hvilke områder av kuben du regner med å utføre flest spørringer i. Typen optimaliseringssektor angir hvordan du regner med å utføre spørringer i dette området. Hvis for eksempel 50 % av spørringene inkluderer Måned, kan du definere en optimaliseringssektor på sektoren Vilkårlig-Måned-Vilkårlig. Optimaliseringsrådgiveren kan bruke optimaliseringssektorene som du oppgir, til å anbefale sammendragstabeller som samsvarer godt dine behov.

Slik oppgir du optimaliseringssektorer for en kube:

1. Fra kubeveiviseren eller vinduet Kubeegenskaper åpner du siden Spørringstype, velger **Avanserte innstillinger**, og klikker deretter på **Spesifiser**.
2. Definer en eller flere optimaliseringssektorer for kuben.

Tilføy en sektor

Klikk på **Ny** for å tilføye en sektor. Den nye sektoren vises i den interaktive grafikken, og som en rad på tabellen **Liste over optimaliseringssektorer**. Etter at du har tilføyd en sektor, endrer du sektoren ved å oppgi typen spørring og kubenivået i hver kubedimensjon som du regner med å utføre flest spørringer i.

Endre en sektor

Du kan endre en sektor i den interaktive grafikken eller i tabellen **Liste over optimaliseringssektorer** nedenfor grafikken.

Hvis du vil endre en sektor i tabellen, velger du den raden i tabellen som representerer sektoren. Klikk på typen eller kubedimensjonen som du vil endre, og velg et alternativ.

Hvis du vil endre en sektor i den interaktive datagrafikken, endrer du nivå ved å dra noden i kubedimensjonen til et nivå, og endrer type ved å høyreklikke på sektoren og velge spørringstype.

Du kan velge et av disse alternativene for hver kubedimensjon:

- Velg **Vilkårlig** hvis du ikke har noen preferanser om hvilket kub nivå sektoren skal være på. Optimaliseringsrådgiveren bestemmer hvilket kub nivå det skal optimaliseres for.
- Velg **Alle** hvis du ofte utfører spørringer i dataene på det høyeste samlingsnivået, for eksempel for alle produkter eller alle regioner.
- Velg et bestemt kub nivå som er definert for kubedimensjonen hvis du vet hvilket kub nivå mange av spørringene bruker. Du kan for eksempel definere en optimaliseringssektor på kub nivåene **Vilkårlig-Måned-Vilkårlig**, der du velger **Vilkårlig** i kubedimensjonen **Marked**, **Måned** i kubedimensjonen **Tid**, og **Vilkårlig** i kubedimensjonen **Produkt**.

Anbefaling: For å oppnå best resultater bør du bare definere et lite antall (tre eller færre) fokuserte sektorer per kube.

Fjern en sektor

Du kan velge en sektor i tabellen eller i den interaktive grafikken og klikke på **Fjern** for å slette sektoren.

Fjerne en dimensjon fra en kubemodell

Du kan fjerne en dimensjon fra en kubemodell hvis du ikke lenger har behov for dimensjonen. Du kan fjerne en dimensjon uten å slette dimensjonen, hvis dimensjonen blir brukt av en annen kubemodell.

Når en dimensjon er fjernet fra en kubemodell, gjelder følgende:

- Dimensjonen blir bare fjernet fra den valgte kubemodellen, men den beholdes i eventuelle andre kubemodeller som refererer til den.
- Eventuelle kubedimensjoner som er basert på denne dimensjonen i kubemodellen, blir fjernet fra tilsvarende kuber.
- Dimensjonen blir ikke slettet fra databasen.
- Dimensjonen vil være tilgjengelig i mappen **Alle dimensjoner**.

Slik fjerner du en dimensjon fra en kubemodell:

1. I objektoversikten for OLAP-senteret utvider du mappen **Kubemodeller** slik at du ser de eksisterende kubemodellene. Hvis objektoversikten ikke inneholder mappen **Kubemodeller**, må du kontrollere at du er i OLAP-objektoversikten ved å klikke på **Vis** → **Vis OLAP-objekter**.
2. Utvid en kubemodellnode slik at du ser objektkategoriene i kubemodellen.
3. Utvid mappen **Dimensjoner** slik at du ser de eksisterende dimensjonene.
4. Høyreklikk på en dimensjon, og klikk deretter på **Fjern**.

Slette et metadataobjekt fra en database

Du kan slette et metadataobjekt hvis du ikke lenger bruker metadataobjektet i en kubemodell i denne databasen.

Du kan slette de fleste objekter fra objektoversikten i OLAP eller relasjonsobjektoversikten. Du kan bare slette objekter hvis de ikke blir referert til av andre objekter.

Slik sletter du et metadataobjekt fra en database:

Velg ett eller flere objekter i objektoversikten i OLAP-senteret, og klikk på **Slett**. Hvis menyen ikke inneholder valget **Slett**, kan du ikke slette det valgte objektet. Når du sletter et overordnet objekt, blir alle de underordnede objektene slettet. Hvis du for eksempel sletter en dimensjon, vil det også slette alle de tilsvarende hierarkiene.

Kapittel 4. DB2 Cube Views, scenarier for forretningsmodellering

Denne delen beskriver disse emnene:

Beregne flyt og verdi av lagerbeholdning i et varelager over tid

En detaljist, XYZ Detalj, lagrer varene i en lagerbygning før varene blir sendt til en av butikkene for å selges. XYZ Detalj har data for tilstanden til lagerbeholdningen i varelageret over tid, og ønsker å analysere disse dataene.

Korrelere annonseringskostnader og salg

En bilforhandler vurderer å bruke mer penger på annonsering. For å kunne treffe en velbegrunnet beslutning ønsker forhandleren først å analysere forholdet mellom tidligere annonseringskostnader og salg. Forhandleren er interessert i å se om forskjellige nivåer av annonsering har påvirket salget, og spesielt om økt annonsering er nært knyttet til økt salg.

Beregne fortjenesten og fortjenestemarginen for en butikk

Butikksjefen i en leketøysbutikk ønsker å analysere hvordan forskjellige faktorer som tid på året og type produkt påvirker fortjenesten og fortjenestemarginen.

Telle antallet Internett-bestillinger

En detaljist utvidet for noen få år siden forretningsdriften ved å starte salg på Internett. Nå ønsker firmaet å analysere virkningen av sitt Internett-salg. En av de første beregningene firmaet trenger, er hvor mange bestillinger som er levert over Internett.

Rangere salgshall

En butikkjede som selger kontorrekvisita, har ekspandert raskt de siste årene. Ledelsen vurderer å legge ned enkelte av butikkene med dårligst resultat for å redusere kostnadene og øke fortjenesten.

Bruke tidsdata lagret i faktatabellen for å opprette en tidsdimensjon

En detaljist, XYZ Detalj, modellerer salgstransaksjonsdataene i DB2 Cube Views slik at dataene kan analyseres mer effektivt. Siden dette er transaksjonsdata, er den eneste tilgjengelige tidsinformasjonen en dato som er knyttet til hver enkelt transaksjon.

Beregne flyt og verdi av lagerbeholdning i et varelager over tid

En detaljist, XYZ Detalj, lagrer varene i en lagerbygning før varene blir sendt til en av butikkene for å selges. XYZ Detalj har data for tilstanden til lagerbeholdningen i varelageret over tid, og ønsker å analysere disse dataene.

Firmaet ønsker spesielt å undersøke to aspekter av varelageret:

- Flyten av varer inn og ut av lageret
- Verdien av varene som finnes i varelageret på et gitt tidspunkt

Det første aspektet, flyten av varer, ser på data over tid. Det andre aspektet, verdien av varene, tar et snapshot av varelageret på et bestemt tidspunkt.

Detaljer for scenariet

XYZ Detalj har en faktatabell med følgende kolonner som gjelder varelageret: QUANTITY_IN, QUANTITY_OUT, CURRENT_QUANTITY, PRODUCT_VALUE, PRODUCT_ID og TIME_ID. Disse dataene registreres i tabellen ukentlig. Databasen har også tabellene Product og Time. Et eksempel på et sett med tabelldata er vist i tabell 28.

Tabell 28. Eksempel på faktatabelldata

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

PRODUCT_ID-verdien for hver av de tre eksempeldatapostene er den samme fordi en produkttype kan gå inn og ut av lageret flere ganger.

Databaseadministratoren for XYZ Detalj må opprette tre forskjellige mål:

Flow In

Modell for flyten av varer inn til lageret.

Flow Out

Modell for flyten ut av lageret.

Current Value

Modell for verdien av varene på et gitt tidspunkt.

Når de to første målene skal opprettes, Flow In og Flow Out, oppretter databaseadministratoren mål som er tilordnet kolonnene QUANTITY_IN og QUANTITY_OUT, og summerer dataene på tvers av alle dimensjoner. Dette kalles et fullstendig additivt mål fordi dataene samles kun ved hjelp av funksjonen SUM på tvers av alle dimensjoner. I tabell 29 ser du et eksempel på data for kolonnene QUANTITY_IN og QUANTITY_OUT i tre måneder for produktet med PRODUCT_ID 1234. Målene Flow In og Flow Out summerer disse månedlige verdiene og beregner totalt antall som kom inn og gikk ut av lageret i første kvartal.

Tabell 29. Beregning av eksempeldata for de fullstendig additive målene Flow In og Flow Out for PRODUCT_ID 1234

	Januar	Februar	Mars	1. kvartal
QUANTITY_IN	5	20	10	35
QUANTITY_OUT	0	10	20	30

Fullstendig additive mål er de enkleste og vanligste målene du oppretter, og de brukes ofte som byggestener i mer komplekse mål. For mål som er basert på numeriske kildedata, oppretter OLAP-senteret et fullstendig additivt mål.

Når det tredje målet skal opprettes, Current Value, oppretter databaseadministratoren et beregnet mål som beregner verdien ved å multiplisere PRODUCT_VALUE med CURRENT_QUANTITY. Hvis for eksempel verdien til produktet med PRODUCT_ID=1234 er 5, blir målet Current Value for eksempeldataene slik det er vist i tabell 30 på side 61.

Tabell 30. Beregning av eksempeldata for målet Current Value for PRODUCT_ID 1234

	Januar	Februar	Mars
CURRENT_QUANTITY	5	10	20
Current Value	25	50	100

Disse dataene må deretter samles på tvers av dimensjonene. Siden dette målet beregner verdien på et bestemt tidspunkt, er det imidlertid uten mening å summere på tvers av tidsdimensjonen. Samlingen summerer derimot dataene på tvers av produktdimensjonen og finner gjennomsnittet for dataene over tid. Dette kalles et semi-additivt mål fordi bare deler av samlingen omfatter funksjonen SUM.

Mål som beregner snapshotdata, data som representerer et bestemt tidspunkt (som en måneds lagerdata), er ofte semi-additive mål fordi det er uten mening å summere månedene i kvartaler. Hvis et produkt ligger på lager i hele kvartalet, blir dette produktet tatt med i CURRENT_QUANTITY-snapshotdataene for varebeholdningen i hver av de tre månedene i kvartalet. Hvis CURRENT_QUANTITY-dataene summeres over tid, blir produktet som lå på lager i tre måneder, telt tre ganger. Som vist i tabell 31 er verdien 25 for første kvartal uten verdi for aktiviteten på lageret. Tabellen viser at det aldri var 25 produkter på lager, og beregningen av verdien 25 er uten mening.

Tabell 31. Beregning av eksempeldata for kolonnen CURRENT_QUANTITY med funksjonen SUM for tidsdimensjonen for PRODUCT_ID 1234

	Januar	Februar	Mars	1. kvartal
SUM(CURRENT_QUANTITY)	5	15	5	25

I stedet for å bruke funksjonen SUM på tvers av alle dimensjoner, kan du utføre andre samlingsfunksjoner som AVG, MIN og MAX for tidsdimensjonen. Med samme sett med eksempeldata for januar, februar og mars kan du bruke andre samlingsfunksjoner for tidsdimensjonen slik det er vist i tabell 32, for å opprette meningsfylte verdier for kvartalet. Målet Current Value kan representere gjennomsnittlig total verdi for varene som er lagret på lageret i løpet av kvartalet, eller maksimums- eller minimumsverdien på et hvilket som helst tidspunkt i løpet av kvartalet.

Tabell 32. Beregning av eksempeldata for kolonnen CURRENT_QUANTITY med funksjonene AVG, MAX og MIN for tidsdimensjonen for PRODUCT_ID 1234

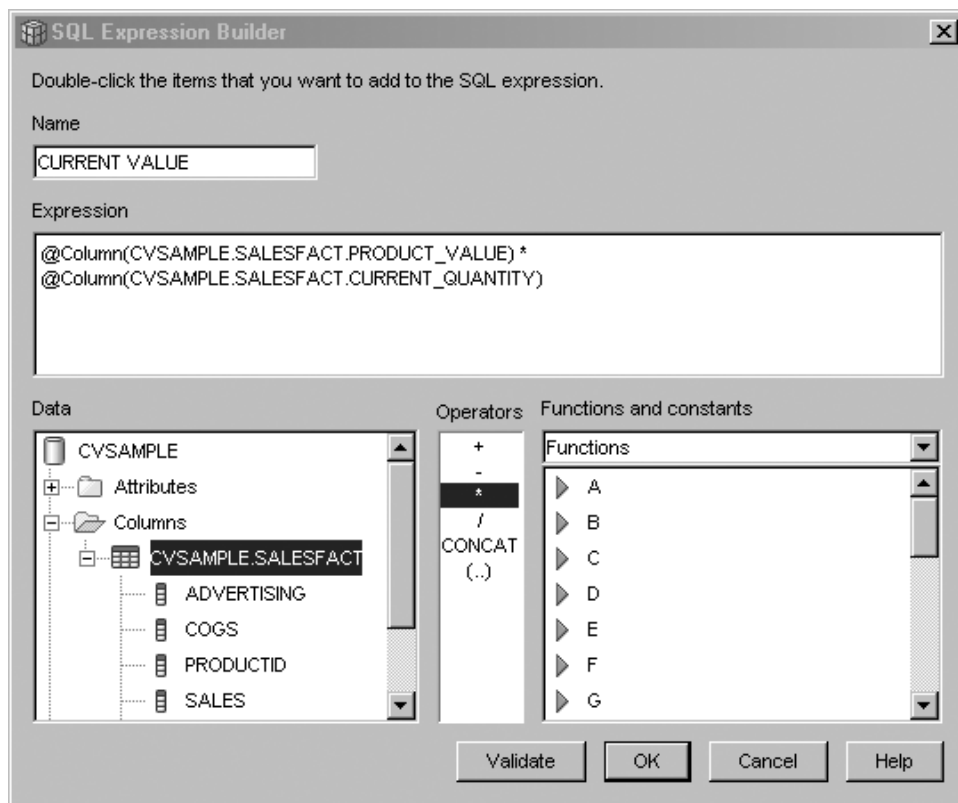
	Januar	Februar	Mars	1. kvartal
AVG(CURRENT_QUANTITY)	5	15	5	8,3
MAX(CURRENT_QUANTITY)	5	15	5	15
MIN(CURRENT_QUANTITY)	5	15	5	5

Opprette målene


Fremgangsmåten nedenfor beskriver hvordan du kan opprette målene Flow In, Flow Out og Current Value i et eksisterende faktaobjekt i vinduet Faktaegenskaper i OLAP-senteret:

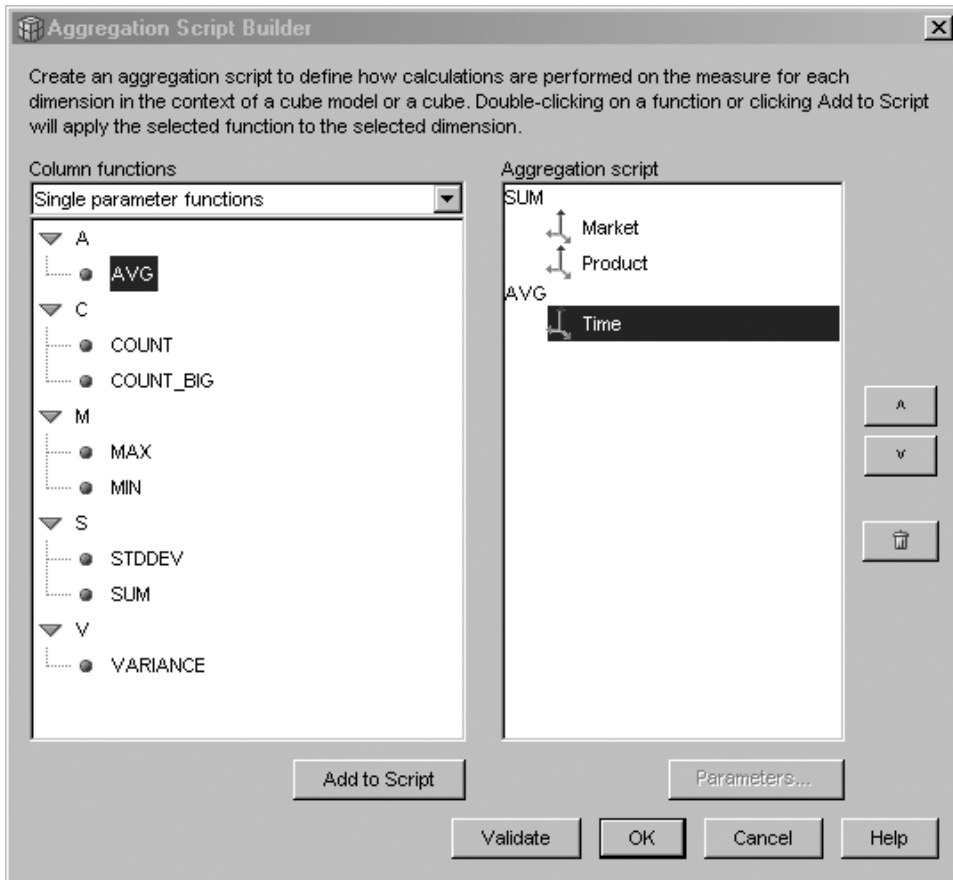
1. Åpne vinduet Faktaegenskaper ved å høyreklikke på faktaobjektet i objektoversikten i OLAP-senteret, og klikk på **Rediger mål**. Vinduet Faktaegenskaper blir åpnet.
2. Opprett målet Flow In:
 - a. Klikk på **Opprett beregnet mål** på siden Mål. Vinduet Bygging av SQL-uttrykk blir åpnet.
 - b. Skriv FLOW IN i feltet **Navn** i vinduet Bygging av SQL-uttrykk.
 - c. Slik oppretter du Flow In-uttrykket:
 - Utvid mappen **Kolonner** og faktatabellen i listen Data.
 - Dobbeltklikk på kolonnen **QUANTITY_IN** for å tilføye den i uttrykket.
 - Klikk på **OK** for å lukke vinduet Bygging av SQL-uttrykk. Du trenger ikke å endre standard samlingsfunksjon, SUM, på siden Samlinger. Funksjonen SUM er standard for målet Flow In fordi datakilden er numerisk og målet refererer til en kolonne, ikke til bare eksisterende mål.
3. Opprett målet Flow Out:
 - a. Klikk på **Opprett beregnet mål** på siden Mål. Vinduet Bygging av SQL-uttrykk blir åpnet.
 - b. Skriv FLOW OUT i feltet **Navn** i vinduet Bygging av SQL-uttrykk.
 - c. Slik oppretter du Flow Out-uttrykket:
 - Utvid mappen **Kolonner** og faktatabellen i listen **Data**.
 - Dobbeltklikk på kolonnen **QUANTITY_OUT**.
 - d. Klikk på **OK** for å lukke vinduet Bygging av SQL-uttrykk. Du trenger ikke å endre standard samlingsfunksjon, SUM, på siden Samlinger. Funksjonen SUM er standard for målet Flow Out fordi datakilden er numerisk og målet refererer til en kolonne, ikke til bare eksisterende mål.
4. Opprett målet Current Value:
 - a. Klikk på **Opprett beregnet mål** på siden Mål. Vinduet Bygging av SQL-uttrykk blir åpnet.
 - b. Skriv CURRENT VALUE i feltet **Navn** i vinduet Bygging av SQL-uttrykk.
 - c. Slik oppretter du Current Value-uttrykket:
 - Utvid mappen **Kolonner** og faktatabellen i listen **Data**.
 - Dobbeltklikk på kolonnen **PRODUCT_VALUE** på listen **Data**.
 - Dobbeltklikk på operatoren * på listen **Operatorer**.
 - Dobbeltklikk på kolonnen **CURRENT_QUANTITY** på listen **Data**.

I figur 12 på side 63 ser du Current Value-uttrykket du kan opprette i vinduet Bygging av SQL-uttrykk.



Figur 12. Fullfør Current Value-uttrykket i vinduet Bygging av SQL-uttrykk

- d. Klikk på **OK** for å lukke vinduet Bygging av SQL-uttrykk.
- e. Klikk på samlingen for målet Current Value på siden Samlinger og velg **Samlingskript** fra listen. Vinduet Bygging av samlingskript blir åpnet. Standard samlingskript har funksjonen SUM for alle dimensjoner.
- f. Hvis det er nødvendig, flytter du tidsdimensjonen ned ved å velge **Tid** og klikke på  knappen for å flytte ned, slik at dette blir den siste dimensjonen i skriptet. Mens tidsdimensjonen fortsatt er valgt, dobbeltklikker du på funksjonen **AVG** på listen **Kolonnefunksjoner**. Samlingskriptet som er vist i figur 13 på side 64, summerer dataene på tvers av alle dimensjoner unntatt tid, mens det beregner gjennomsnitt for dataene over tid.



Figur 13. Samlingskript for målet Current Value

- g. Klikk på **Valider** i vinduet Bygging av samlingskript for å kontrollere at samlingskriptet er gyldig. Klikk på **OK** for å lagre samlingskriptet og lukke vinduet.
5. Klikk på **OK** for å lagre endringen i faktaobjektet og lukke vinduet Faktaegenskaper.

Du har nå tre beregnede mål for lagerbeholdningen i varelageret. Ved hjelp av disse tre målene kan du analysere mønsteret til produktflyten inn og ut av varelageret.

Korrelere annonseringskostnader og salg

En bilforhandler vurderer å bruke mer penger på annonsering. For å kunne treffe en velbegrunnet beslutning ønsker forhandleren først å analysere forholdet mellom tidligere annonseringskostnader og salg. Forhandleren er interessert i å se om forskjellige nivåer av annonsering har påvirket salget, og spesielt om økt annonsering er nært knyttet til økt salg.

Detaljer for scenariet

Forhandlerens database har en faktatabell med kolonnene Sales og Ad Costs. Databasen har dessuten flere andre dimensjonstabeller. Databaseadministratoren kan opprette et mål som bruker DB2-funksjonen CORRELATION, for å utføre korrelasjonsberegninger mellom kostnadene og salget. Funksjonen CORRELATION

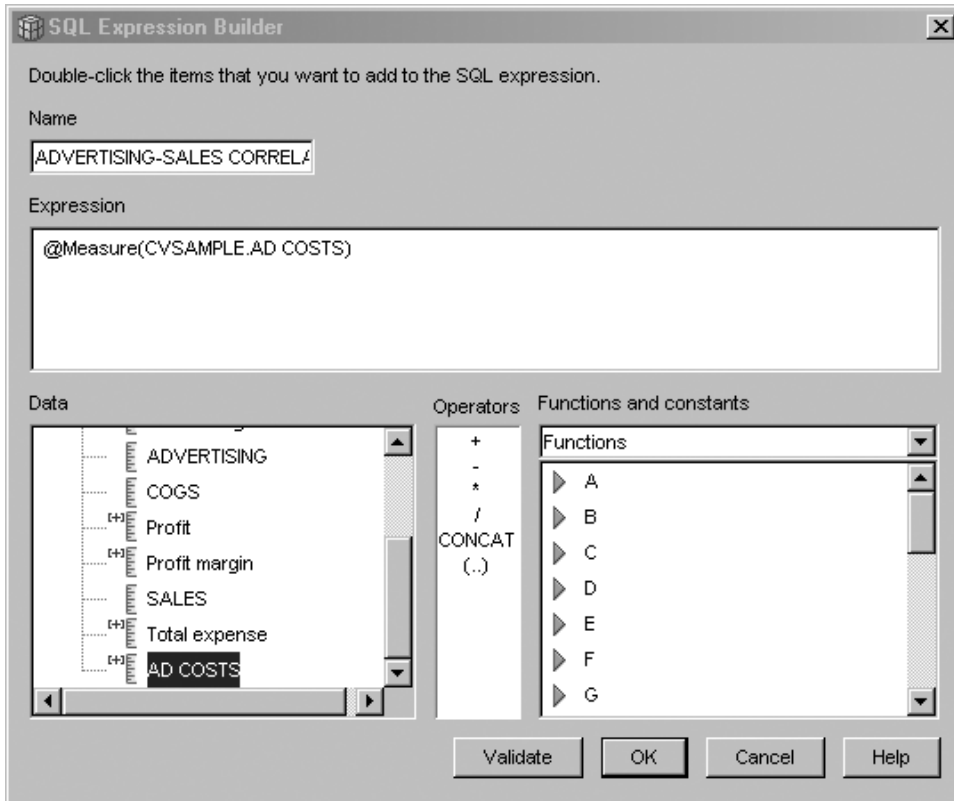
er en funksjon med flere parametere, som krever to inndataparametere. I dette tilfellet bruker databaseadministratoren kolonnene Sales og Ad Costs som de to inndataparameterne.

Databaseadministratoren må bruke samlingsfunksjonen med flere parametere først i samlingskriptet. Funksjonen med flere parametere kan brukes på tvers av alle dimensjonene, eller den kan brukes først for alle dimensjonene unntatt tidsdimensjonen, og en annen funksjon, for eksempel funksjonen MAX, kan brukes for tidsdimensjonen. Databaseadministratoren definerer SQL-uttrykket for målet slik at det er direkte tilordnet kolonnen Ad Costs. SQL-uttrykket er den første av de to parameterne som brukes i funksjonen med flere parametere. Databaseadministratoren definerer den andre parameteren som et SQL-uttrykk som er direkte tilordnet kolonnen Sales. Funksjonen CORRELATION er definert som den eneste samlingsfunksjonen slik at målet kan beregne den statistiske korrelasjonen mellom annonseringskostnader og salgsresultater på tvers av alle dimensjoner.

Opprette mål

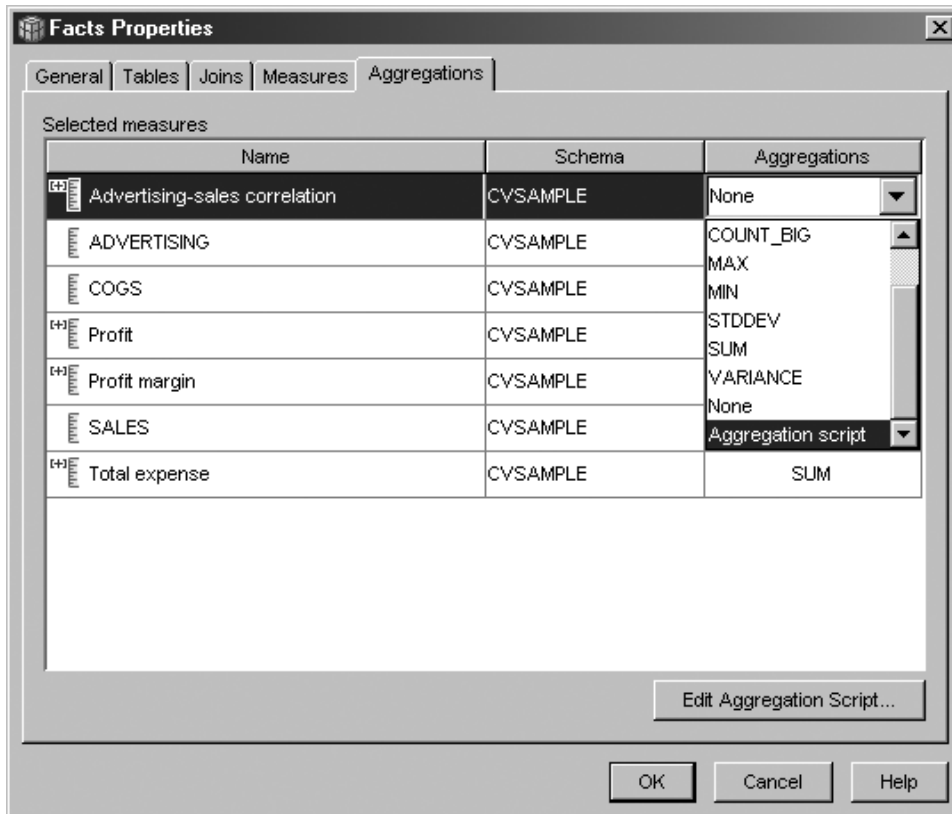
Fremgangsmåten nedenfor beskriver hvordan du kan opprette målet Advertising-Sales Correlation i et eksisterende faktaobjekt i vinduet Faktaegenskaper i OLAP-senteret:

1. Åpne vinduet Faktaegenskaper ved å høyreklikke på faktaobjektet i OLAP-senterets objektoversikt og klikke på **Rediger mål**.
2. Klikk på knappen **Opprett beregnet mål**. Vinduet Bygging av SQL-uttrykk blir åpnet.
3. Skriv ADVERTISING-SALES CORRELATION i feltet **Navn** i vinduet Bygging av SQL-uttrykk.
4. Definer målets uttrykk, som også skal brukes som første parameter for funksjonen CORRELATION i samlingskriptet. Du definerer uttrykket ved å utvide mappen **Mål** i listen **Data** og dobbeltklikke på målet **AD COSTS** for å tilføye det til listen **Expression**. figur 14 på side 66 viser uttrykket du oppretter i vinduet Bygging av SQL-uttrykk.



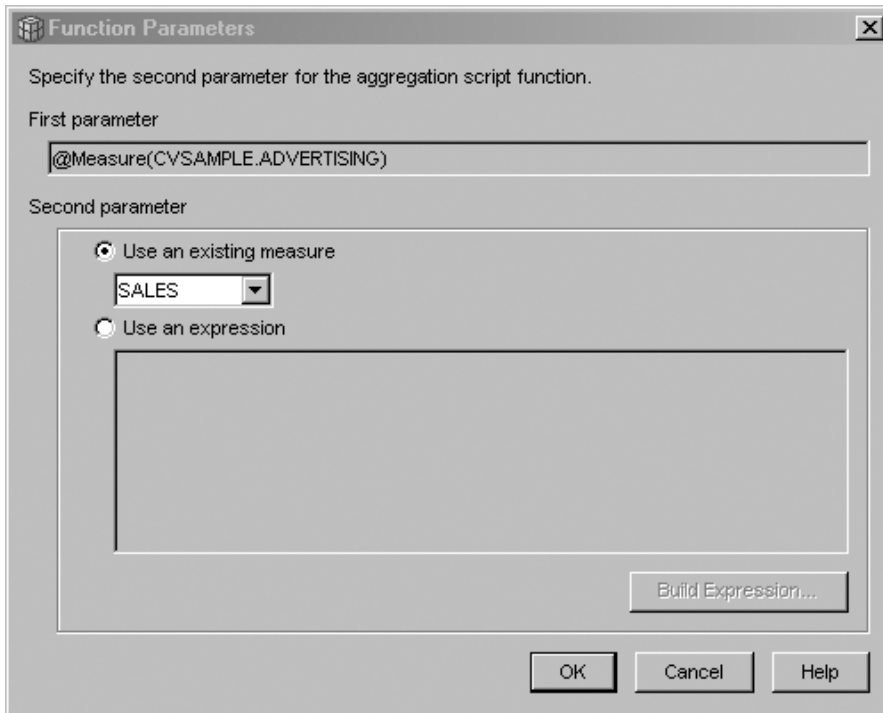
Figur 14. Fullført AdvertisingSales Correlation-uttrykk i vinduet Bygging av SQL-uttrykk

5. Klikk på samlingsfunksjonen for målet **ADVERTISING-SALES CORRELATION** på siden Samlinger og velg **Samlingsskript**, slik det er vist i figur 15 på side 67. Vinduet Bygging av samlingskript blir åpnet.



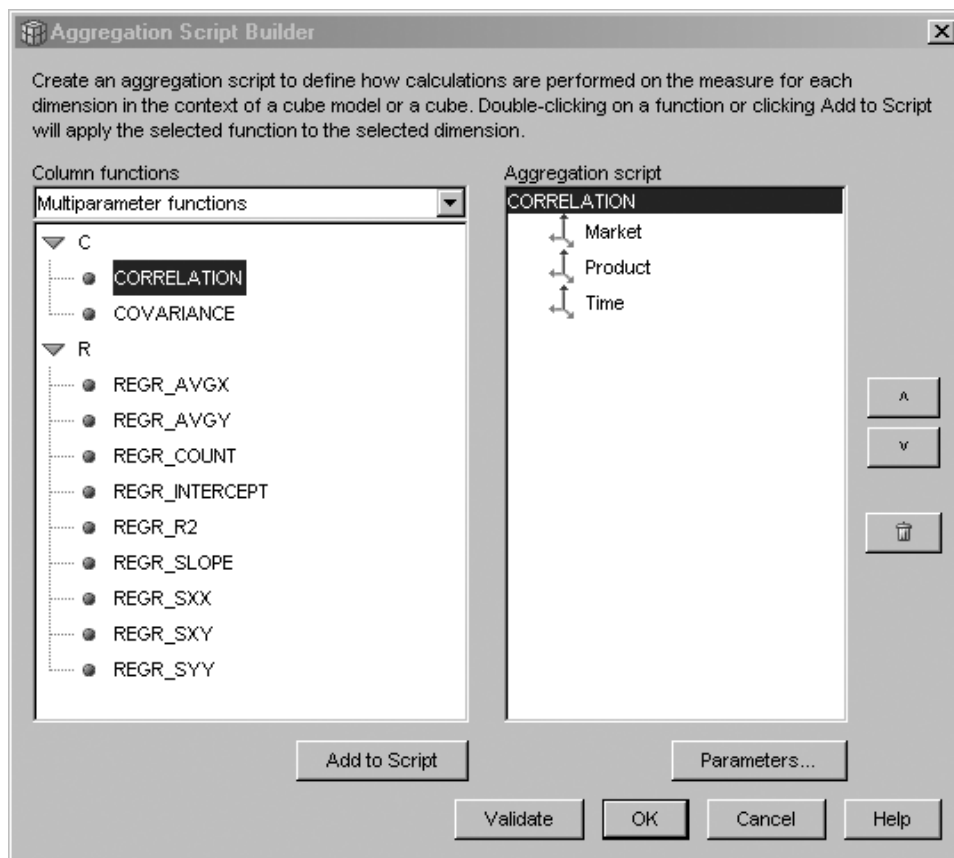
Figur 15. Siden Samlinger i vinduet Faktaegenskaper

6. Velg **Funksjoner med flere parametere** i feltet **Kolonnefunksjoner**. Velg funksjonen **CORRELATION** fra listen over funksjoner med flere parametere og klikk på **Tilføy i skript**. Vinduet Funksjonsparametere blir åpnet.
7. Velg **Bruk et eksisterende mål** og velg **SALES** fra listen, slik det er vist i figur 16 på side 68.



Figur 16. Målet Sales oppgitt som andre parameter i vinduet Funksjonsparametere

8. Klikk på **OK** for å lagre det du har valgt og lukke vinduet Funksjonsparametere.
9. I figur 17 på side 69 ser du at funksjonen CORRELATION ligger øverst på listen over dimensjoner i skriptet.



Figur 17. Samlingskript for målet Advertising-Sales Correlation

10. Klikk på **Valider** i vinduet Bygging av samlingskript for å kontrollere at samlingskriptet er gyldig.
11. Klikk på **OK** for å lagre samlingskriptet og lukke vinduet.
12. Klikk på **OK** for å lagre endringen i faktaobjektet og lukke vinduet Faktaegenskaper.

Du har nå et mål som korrelerer to typer av data i databasen. Ved hjelp av dette målet kan du treffe beslutninger om fremtidig bruk av annonsering på grunnlag av historiske resultater.

Beregne fortjenesten og fortjenestemarginen for en butikk

Butikksjefen i en leketøysbutikk ønsker å analysere hvordan forskjellige faktorer som tid på året og type produkt påvirker fortjenesten og fortjenestemarginen.

Før en mer avansert analyse kan utføres, må databaseadministratoren for leketøysbutikken opprette målet Profit for fortjeneste og Profit Margin for fortjenestemargin. Deretter kan databaseadministratoren opprette flere mål som korrelerer og sammenlikner forskjellige faktorer med målene Profit og Profit Margin.

Detaljer for scenariet

Leketøysbutikkens database har en faktatabell med kolonnene Sales, Costs of Goods Sold (COGS) og Expense i tillegg til tilsvarende fremmednøkkelkolonner for

hver av de mange dimensjonstabellene. Databaseadministratoren har allerede opprettet målene Sales, COGS og Expense, som er tilordnet kolonnene Sales, COGS og Expense. Målene Profit og Profit Margin kan opprettes i sin helhet fra disse eksisterende målene.

Når databaseadministratoren skal opprette målet Profit, oppretter han et mål som beregner SALES-(COGS+EXPENSE) i SQL-uttrykket og summerer de beregnede dataene på tvers av alle dimensjoner. Målet Profit kan opprettes ved at det blir referert til eksisterende mål eller kolonner, eller til begge deler.

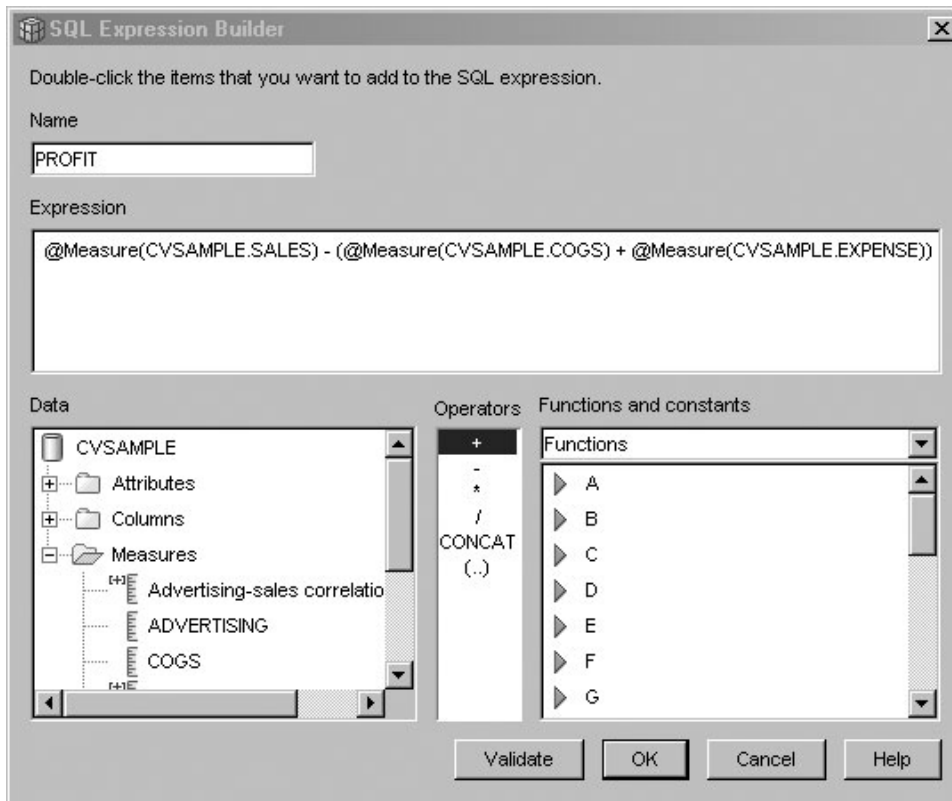
Når målet Profit er opprettet, kan databaseadministratoren opprette målet Profit Margin. Målet Profit Margin er et forhold mellom to eksisterende mål uttrykt som en prosentdel $(\text{Profit} / \text{Sales}) * 100$, og det krever ikke en egen samlingsfunksjon. Det er ikke nødvendig med en samlingsfunksjon fordi målet bare refererer til to andre mål der dataene allerede er samlet. Hvis databaseadministratoren bruker et sammensatt mål, et mål som bare refererer til andre mål, til beregningen av et forhold, trenger ikke databaseadministratoren å definere en ekstra samling. De fleste samlingsfunksjonene, for eksempel SUM, kan ikke brukes sammen med forhold. Hvis for eksempel leketøysbutikken har fortjenestemarginer på 40 %, 32 %, 28 % og 37 % i fire påfølgende kvartaler, ville en summering av forholdene gi en fortjenestemargin på 137 % for året, noe som er uten mening.

Opprette målene

Fremgangsmåten nedenfor forklarer hvordan du kan opprette målene Profit og Profit Margin i et eksisterende faktaobjekt i vinduet Faktaegenskaper i OLAP-senteret.

1. Åpne vinduet Faktaegenskaper ved å høyreklikke på faktaobjektet i objektoversikten i OLAP-senteret, og klikk på **Rediger mål**. Vinduet Faktaegenskaper blir åpnet.
2. Opprett målet Profit:
 - a. Klikk på knappen **Opprett beregnet mål** på siden Mål. Vinduet Bygging av SQL-uttrykk blir åpnet.
 - b. Skriv PROFIT i feltet **Navn** i vinduet Bygging av SQL-uttrykk.
 - c. Utvid mappen **Mål** i listen **Data** og utfør disse trinnene for å opprette Profit-uttrykket:
 - Dobbeltklikk på målet **SALES** på listen **Data** for å tilføye det til uttrykket.
 - Dobbeltklikk på operatoren - på listen **Operatorer**.
 - Dobbeltklikk på målet **COGS** på listen **Data**.
 - Dobbeltklikk på operatoren + på listen **Operatorer**.
 - Dobbeltklikk på målet **EXPENSE** på listen **Data**.
 - I feltet **Uttrykk** merker du `@Measure(CVSAMPLE.COGS)+@Measure(CVSAMPLE.EXPENSE)` i uttrykket og dobbeltklikker på operatoren `(.)` på listen **Operatorer** for å sette den merkede delen av uttrykket i parentes.

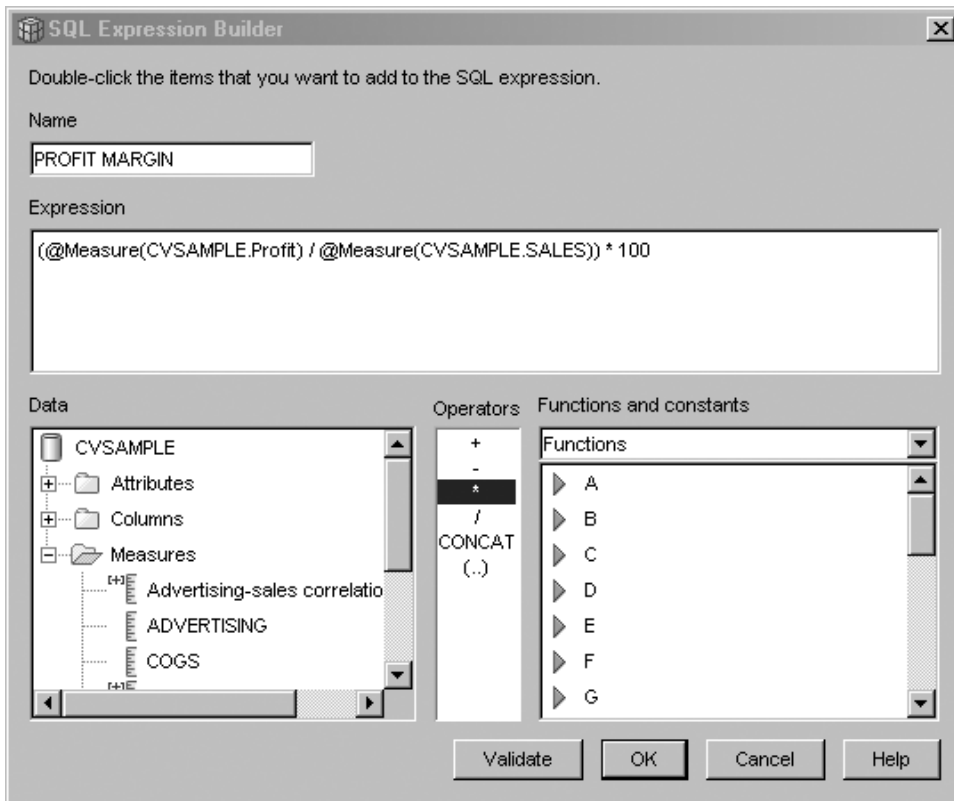
I figur 18 på side 71 ser du Profit-uttrykket du kan opprette i vinduet Bygging av SQL-uttrykk.



Figur 18. Fullført Profit-uttrykk i vinduet Bygging av SQL-uttrykk

- d. Klikk på **OK** for å opprette målet Profit og lukke vinduet Bygging av SQL-uttrykk.
 - e. Klikk på samlingen for målet Profit på siden Samlinger og velg funksjonen SUM. Målet Profit er opprettet.
3. Opprett målet Profit Margin:
- a. Klikk på **Opprett beregnet mål** på siden Mål. Vinduet Bygging av SQL-uttrykk blir åpnet.
 - b. Skriv PROFIT MARGIN i feltet **Navn** i vinduet Bygging av SQL-uttrykk.
 - c. Utvid mappen **Mål** i listen **Data** og utfør disse trinnene for å opprette Profit Margin-uttrykket:
 - Dobbeltklikk på målet **PROFIT** på listen **Data** for å tilføye det til uttrykket.
 - Dobbeltklikk på operatoren **/** fra listen **Operatorer**.
 - Dobbeltklikk på målet **SALES** på listen **Data**.
 - Sett hele uttrykket i parentes ved å skrive inn parenteser i feltet **Uttrykk**.
 - Plasser markøren på slutten av uttrykket og dobbeltklikk på operatoren ***** på listen **Operatorer**.
 - Skriv **100** på slutten av uttrykket i feltet **Uttrykk**.

I figur 19 på side 72 ser du Profit Margin-uttrykket du kan opprette i vinduet Bygging av SQL-uttrykk.



Figur 19. Fullført Profit Margin-uttrykk i vinduet Bygging av SQL-uttrykk

- d. Klikk på **OK** for å opprette målet Profit Margin og lukke vinduet Bygging av SQL-uttrykk.

På siden Samlinger setter OLAP-senteret som standard samlingsfunksjonen til NONE for sammensatte mål, så du trenger ikke å endre samlingsfunksjonen.

4. Klikk på **OK** for å lukke vinduet Faktaegenskaper og lagre de to nye målene du har tilføyd til faktaobjektet.

Når databaseadministratoren har opprettet disse to målene, kan det utføres nyttige analyser ved hjelp av målene.

Telle antallet Internett-bestillinger

En detaljist utvidet for noen få år siden forretningsdriften ved å starte salg på Internett. Nå ønsker firmaet å analysere virkningen av sitt Internett-salg. En av de første beregningene firmaet trenger, er hvor mange bestillinger som er levert over Internett.

Detaljer for scenariet

Firmaets database har en faktatabell for Internett-bestillinger med kolonnene ORDER_ID, PRODUCT_ID, QUANTITY og TIME_ID. Kolonnen PRODUCT_ID inneholder hvert enkelt produkt som er solgt i en bestilling, og kolonnen QUANTITY inneholder produktantallet som er solgt i bestillingen. Bestillinger med mer enn ett produkt har like mange rader som antallet av produkter som er solgt i bestillingen. I tabell 33 på side 73 ser du et eksempel der bestilling 1 omfatter antallet 3 av produkt A, 1 av produkt O og 1 av produkt G.

Tabell 33. Innhold i faktatabell

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

Databaseadministratoren kan opprette målet Order Count som teller hver unike post i kolonnen ORDER_ID. Målet Order Count blir definert med nøkkelordet DISTINCT i SQL-uttrykket og funksjonen COUNT for samling på tvers av alle dimensjonene. Målets SQL-uttrykk oppretter en liste over distinkte bestillinger som telles under samlingen. Fordi målet ikke omfatter noen summering, blir det kalt et ikke-additivt mål.

Ikke-additive mål er også nyttige når du har tegndata eller andre data som du vil telle. Du kan for eksempel bruke ikke-additive mål for å telle antall postnumre du har sendt produkter til.

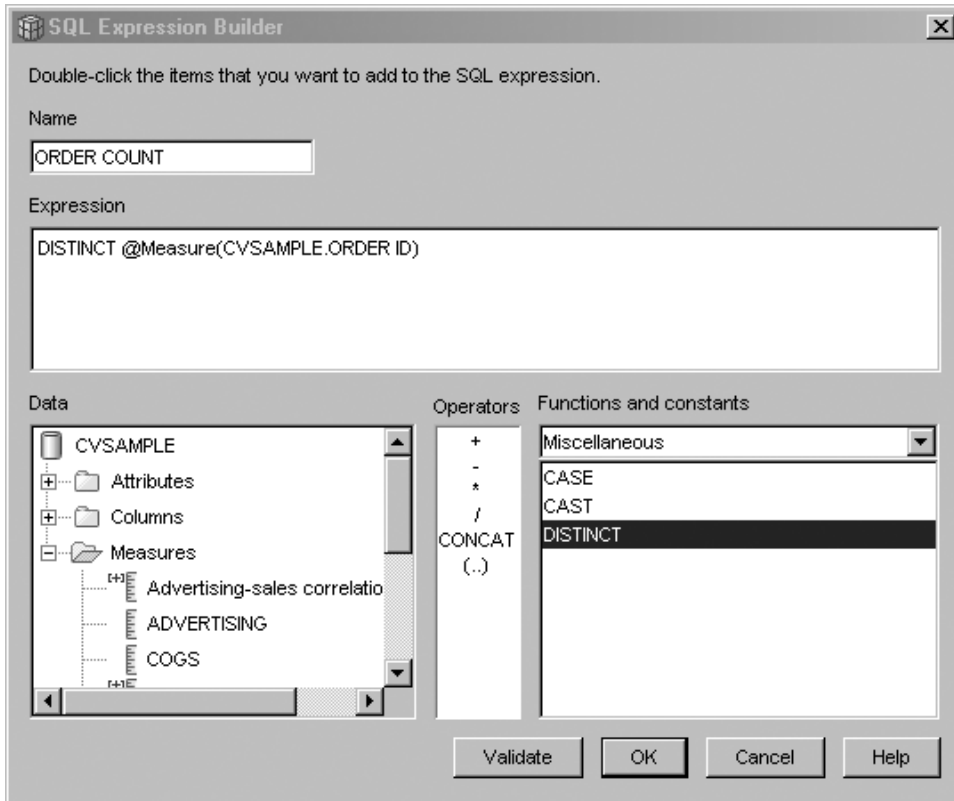
I dette eksempelet valgte databaseadministratoren å definere målet Order ID som er direkte tilordnet kolonnen ORDER_ID. Du kan imidlertid velge å bruke kolonnen ORDER_ID på samme måte. Standardsamlingen er forskjellig basert på om du bruker en kolonne eller et mål i SQL-uttrykket, men i begge tilfeller må du endre standardsamlingen til funksjonen COUNT slik det er beskrevet i "Opprette målene".

Opprette målene

Fremgangsmåten nedenfor beskriver hvordan du kan opprette målet Order Count i et eksisterende faktaobjekt i vinduet Faktaegenskaper i OLAP-senteret:

1. Åpne vinduet Faktaegenskaper ved å høyreklikke på faktaobjektet i objektoversikten i OLAP-senteret, og klikk på **Rediger mål**. Vinduet Faktaegenskaper blir åpnet.
2. Klikk på knappen **Opprett beregnet mål** på siden Mål. Vinduet Bygging av SQL-uttrykk blir åpnet.
3. Skriv ORDER COUNT i feltet **Navn** i vinduet Bygging av SQL-uttrykk.
4. Utvid mappen **Mål** i listen **Data** og utfør disse trinnene for å opprette Order Count-uttrykket:
 - Velg **Diverse** i feltet **Funksjoner og konstanter**. Dobbeltklikk på nøkkelordet **DISTINCT** i listen over diverse funksjoner og konstanter.
 - Dobbeltklikk på målet **ORDER ID** på listen **Data**.

I figur 20 på side 74 ser du Order Count-uttrykket du kan opprette i vinduet Bygging av SQL-uttrykk.



Figur 20. Fullført Order Count-uttrykk i vinduet Bygging av SQL-uttrykk

5. Klikk på **OK** for å lukke vinduet Bygging av SQL-uttrykk.
6. Klikk på samlingen for målet Order Count på siden Samlinger og velg funksjonen **COUNT**.
7. Klikk på **OK** for å lagre endringen i faktaobjektet og lukke vinduet Faktaegenskaper.

Du har nå målet som teller antallet distinkte Order ID-radverdier. Du kan bruke dette målet sammen med andre mål for å analysere dataene dine.

Rangere salgstall

En butikkjede som selger kontorrekvisita, har ekspandert raskt de siste årene. Ledelsen vurderer å legge ned enkelte av butikkene med dårligst resultat for å redusere kostnadene og øke fortjenesten.

Salgshistorikken til en butikk er en viktig faktor ved vurderingen av om en butikk skal legges ned. Under analysen må det være mulig å rangere salgstallene og drille ned på tvers av dimensjonene.

Detaljer for scenariet

Butikkjedens database har en faktatabell med kolonnen Sales sammen med andre kolonner. Databasen har dessuten flere dimensjonstabeller.

Databaseadministratoren kan opprette målet Sales Rank som bruker funksjonen RANK, en OLAP-funksjon som finnes i DB2 Universal Database (DB2 UDB).

DB2 Cube Views støtter følgende OLAP-funksjoner i DB2 UDB:

RANK

Ordner radene og tildeler en rangfølge til hver rad. Rangfølgen er definert som 1 pluss antall forutgående rader som er distinkte med hensyn til rekkefølge. Hvis den relative rekkefølgen til to eller flere rader ikke kan fastslås fordi de har like radverdier, får radene tildelt samme rangfølge. Rangeringsresultatene kan ha hull i rangfølgen hvis det finnes like radverdier. tabell 34 på side 76 viser et eksempel på rangeringsresultatene fra RANK-funksjonen for et sett med radverdier.

Syntaksen for funksjonen RANK er

```
RANK ( ) OVER (ORDER BY  
sorteringsnøkkeluttrykk  
uttrykksrekkefølge)
```

der *sorteringsnøkkeluttrykk* er settet med data som skal rangeres, og *uttrykksrekkefølge* er et nøkkelord, **ASC** eller **DESC**, som ordner verdiene i sorteringsnøkkeluttrykket i stigende eller synkende rekkefølge. DB2 Cube Views krever at *sorteringsnøkkeluttrykk* er et eksisterende mål, ikke en kolonne eller et attributt. DB2 Cube Views støtter ikke PARTITION BY-leddet som leveres sammen med denne funksjonen i DB2 UDB. Du finner mer informasjon om funksjonen RANK i DB2 UDB Information Center.

DENSERANK

Ordner radene og tildeler en rangfølge til hver rad. Rangfølgen til en rad er definert som 1 pluss antall strengt forutgående rader. Derfor er rangeringsresultatene sekvensielle og uten hull i rangordningen. tabell 34 på side 76 viser et eksempel på rangeringsresultatene fra funksjonen DENSERANK for et sett med radverdier.

Syntaksen for funksjonen DENSERANK er

```
DENSERANK ( ) OVER (ORDER BY sorteringsnøkkeluttrykk uttrykksrekkefølge)
```

der *sorteringsnøkkeluttrykk* er settet med data som skal rangeres, og *uttrykksrekkefølge* er et nøkkelord, **ASC** eller **DESC**, som ordner verdiene i sorteringsnøkkeluttrykket i stigende eller synkende rekkefølge. DB2 Cube Views krever at *sorteringsnøkkeluttrykk* er et eksisterende mål, ikke en kolonne eller et attributt. DB2 Cube Views støtter ikke PARTITION BY-leddet som leveres sammen med denne funksjonen i DB2 UDB. Du finner mer informasjon om funksjonen DENSERANK i DB2 UDB Informasjonssenter.

ROWNUMBER

Beregner sekvensielt radnummer for raden på grunnlag av rekkefølgen, og starter med 1 for den første raden. Hvis ORDER BY-leddet ikke er oppgitt, blir radnumrene tildelt til radene i tilfeldig rekkefølge.

Syntaksen for funksjonen ROWNUMBER er

```
ROWNUMBER ( ) OVER ([ORDER BY  
sorteringsnøkkeluttrykk  
uttrykksrekkefølge])
```

der *sorteringsnøkkeluttrykk* er settet med data som skal rangeres, og *uttrykksrekkefølge* er et nøkkelord, **ASC** eller **DESC**, som ordner verdiene i sorteringsnøkkeluttrykket i stigende eller synkende rekkefølge. DB2 Cube Views krever at et eksisterende mål, ikke en kolonne eller et attributt, blir brukt som datakilde for denne funksjonen. DB2 Cube Views støtter ikke

PARTITION BY-leddet som leveres sammen med denne funksjonen i DB2 UDB. Du finner mer informasjon om funksjonen ROWNUMBER i DB2 UDB Information Center.

Disse OLAP-funksjonene er ikke med på listen Funksjoner og konstanter i vinduet Bygging av SQL-uttrykk.

Tabell 34. Rangeringsresultater for et sett med verdier når du bruker funksjonene RANK og DENSERANK

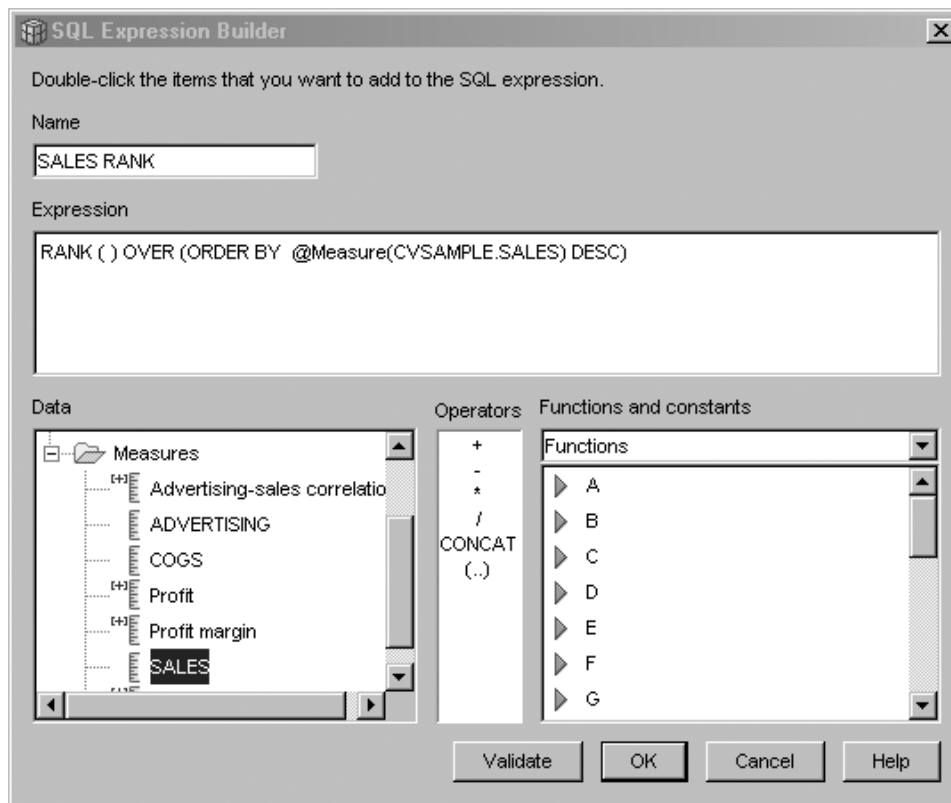
Radverdier	Rekkefølge	Rangeringsresultater fra funksjonen RANK	Rangeringsresultater fra funksjonen DENSERANK
100	1	1	1
35	2	2	2
23	3	3	3
8	4	4	4
8	4	4	5
6	5	6	6

Opprette målene

Fremgangsmåten nedenfor forklarer hvordan du kan opprette målet Sales Rank i et eksisterende faktaobjekt i vinduet Faktaegenskaper for OLAP-senteret:

1. Åpne vinduet Faktaegenskaper ved å høyreklikke på faktaobjektet i objektoversikten i OLAP-senteret, og klikk på **Rediger mål**. Vinduet Faktaegenskaper blir åpnet.
2. Klikk på **Opprett beregnet mål** på siden Mål for å opprette målet Sales Rank. Vinduet Bygging av SQL-uttrykk blir åpnet.
3. Skriv SALES RANK i feltet **Navn** i vinduet Bygging av SQL-uttrykk.
4. Slik oppretter du Sales Rank-uttrykket:
 - Skriv denne funksjonssyntaksen i feltet **Uttrykk**: RANK () OVER (ORDER BY measure DESC).
 - Utvid mappen **Mål** på listen **Data**.
 - Merk ordet measure i funksjonssyntaksen i feltet **Uttrykk** og dobbeltklikk på målet **SALES** for å tilføye målet SALES til uttrykket.

Det endelige uttrykket er vist i figur 21 på side 77.



Figur 21. Fullført Sales Rank-uttrykk i vinduet Bygging av SQL-uttrykk

- Klikk på **Valider** for å kontrollere at uttrykket er gyldig. Klikk på **OK** for å lukke vinduet Bygging av SQL-uttrykk.
Du trenger ikke å endre standard samlingsfunksjon, Ingen, på siden Samlinger. Alternativet Ingen er standard for målet Sales Rank fordi datakilden er numerisk og bare refererer til eksisterende mål.

Ved å bruke funksjonen RANK i målets uttrykk for å ordne kolonnen Sales i synkende rekkefølge, kan den som utfører analysen for eksempel drille ned på tvers av andre dimensjoner for å finne ut hvilken butikk som har de dårligste salgsresultatene det siste året for en bestemt produktlinje eller med hensyn til andre dimensjonsdata som er lagret i databasen.

Bruke tidsdata lagret i faktatabellen for å opprette en tidsdimensjon

En detaljist, XYZ Detalj, modellerer salgstransaksjonsdataene i DB2 Cube Views slik at dataene kan analyseres mer effektivt. Siden dette er transaksjonsdata, er den eneste tilgjengelige tidsinformasjonen en dato som er knyttet til hver enkelt transaksjon.

Det er imidlertid nødvendig med tidsinformasjon som er modellert i en tidsdimensjon, hvis det skal være mulig å analysere salgstrender per kvartal eller beregne gjennomsnittlig lagerbeholdningsverdi for hver enkelt uke.

Mange databaseadministratorer prøver å unngå å lagre tidsdata som en dato eller som systemtid, for hvis det ikke finnes noen transaksjoner en dag, blir det hull i dataene, noe som kan gjøre det vanskelig å samle og vise dataene på riktig måte. Vanligvis er det bedre å modellere tidsdataene i en tidstabell.

Databaseadministratoren hos XYZ Detalj er imidlertid sikker på at det blir minst en transaksjon hver dag, og velger å beholde den gjeldende strukturen til dataene.

Detaljer for scenariet

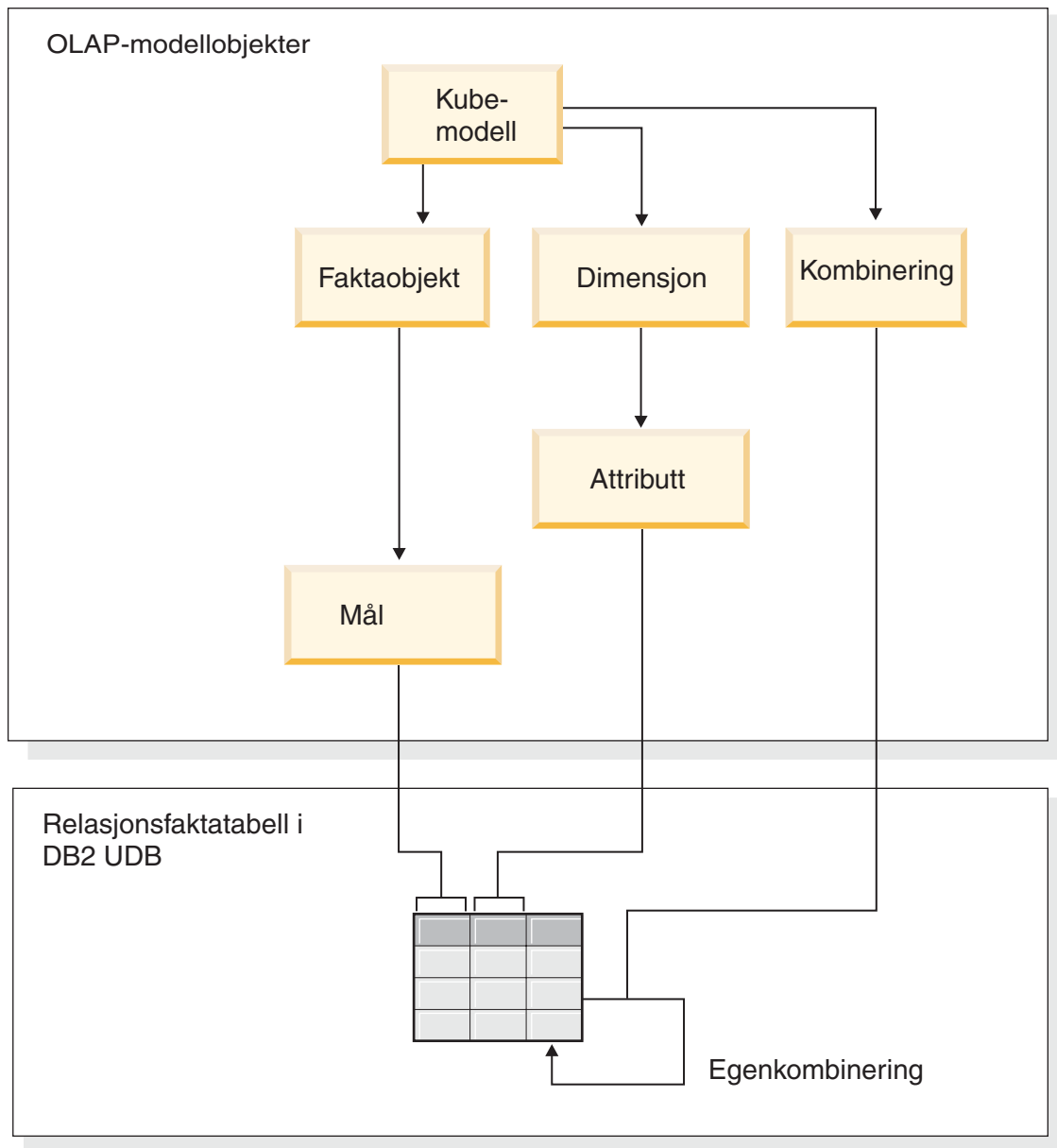
XYZ Detalj har en faktatabell med målbare data for hver transaksjon, som omfatter salg, kostnader, mengde solgt og dato. Databaseen inneholder dessuten en dimensjonstabell for region og en for produkt. Problemet er at tidsdataene er tatt med i faktatabellen og ikke lagret i en egen dimensjonstabell. Databaseadministratoren må opprette et dimensjonsobjekt basert på datodataene i faktaobjektet.

Når administratoren oppretter en tidsdimensjon på grunnlag av en enkelt kolonne med datodata i faktatabellen, stilles det to krav:

- Fordi alle dimensjonsobjektene i en gyldig kubemodell må være kombinert med faktaobjektet og tidsdimensjonsobjektet og faktaobjektet er basert på den samme faktatabellen, må tidsdimensjonsobjektet kombineres med faktaobjektet ved hjelp av en egenkombinering som kombinerer faktatabellen med seg selv.
- Databaseadministratoren må bygge beregnede attributter som samler datodataene i meningsfylte nivåer som uke, måned, kvartal og år.

En egenkombinering er en type kombinering som kombinerer en tabell med seg selv, og i dette tilfellet gjelder det faktatabellen. Egenkombineringen må kombinere en eller flere kolonner som sammen kan identifisere entydig en hvilken som helst rad i faktatabellen. Primærnøkkelen er det beste valget. Hvis det ikke er definert noen primærnøkkel, er settet med kolonner som blir brukt til å kombinere faktatabellen med dimensjonstabellene, en god kandidat for primærnøkkelen. Når du skal optimalisere kubemodellen, må det være definert en primærnøkkel. Kombineringskardinalitet må være 1:1, og kombineringsstypen må være intern.

I figur 22 på side 79 ser du hvordan et faktaobjekt, en dimensjon basert på faktatabellen og en kombinering av fakta og dimensjon kan være tilordnet samme faktatabell.



Figur 22. Hvordan en egenkombinering kombinerer tabellen med seg selv

Opprette attributtene og dimensjonen

Fremgangsmåten nedenfor beskriver hvordan du kan bruke dimensjonsveiviseren i OLAP-senteret til å opprette dimensjonen Tid og beregnede attributter basert på faktatabellen:

1. Du åpner dimensjonsveiviseren ved å høyreklikke på kubemodellen i OLAP-senterets objektoversikt og klikke på **Opprett dimensjon**. Dimensjonsveiviseren blir åpnet.
2. Skriv Time i feltet **Navn** på siden **Navn**. Du kan eventuelt endre forretningsnavnet og skrive inn en kommentar. Klikk på **Neste**.
3. Velg kubemodellens faktatabell. Klikk på **Neste**. Du trenger ikke å oppgi dimensjonskombineringer fordi du bare har en tabell i dimensjonen. Klikk på **Neste** på siden **Dimensjonskombineringer**.
4. Velg kolonnen **Systemtid** på siden **Dimensjonsattributter**.

5. Valgfritt: Opprett de beregnede attributtene som samler systemtiddataene i større enheter som måned, kvartal og år. Når du skal opprette de beregnede attributtene, klikker du på knappen **Opprett beregnet attributt** for å åpne Bygging av SQL-uttrykk og definere uttrykket for hvert attributt som beregner måneder, kvartal og år på grunnlag av systemtidskolonnen. Når du har definert hvert av de beregnede attributtene, klikker du på **Valider** for å kontrollere at uttrykket er gyldig, og deretter på **OK** for å lukke Bygging av SQL-uttrykk og gå tilbake til dimensjonsveiviseren. Klikk på **Neste** når du har opprettet alle attributtene du ønsker.
6. Velg **Tid** på siden Dimensjonstype. Klikk på **Neste**.
7. Klikk på **Opprett kombinerings** på siden Kombineringsveiviseren av faktaobjekt og dimensjon. Kombineringsveiviseren blir åpnet, der du oppretter egenkombineringen. Oppgi et navn og klikk på **Neste**. Velg kolonnen eller settet med kolonner som entydig definerer en rad i faktatabellen, for eksempel primærnøkkelen, for både venstre- og høyreattributtet. Velg ett par om gangen og klikk på **Tilføy** for å tilføye attributtparet til kombineringsveiviseren. Velg kombineringsstypen indre kombineringsveiviser og kardinaliteten 1:1. Når du har tilføyd de nødvendige attributtparene, klikker du på **Fullfør**. Kombineringsveiviseren blir lukket.
8. Klikk på **Fullfør** på siden Kombineringsveiviseren av faktaobjekt og dimensjon.

Når tidsdimensjonen er definert i kubemodellen, har XYZ Detalj fått nye muligheter i sin dataanalyse. De kan nå utføre tidsrelaterte analyser som omfatter varelageret.

Kapittel 5. DB2 Cube Views, optimalisering av kubemodeller

Denne delen beskriver disse emnene:

Sammendragstabeller

DB2 Cube Views bruker DB2-sammendragstabeller for å forbedre ytelsen til spørringer som utføres på kubemodeller. En sammendragstabell er en spesiell type materialisert spørretabell (MQT) som spesifikt tar med sammendragsdata.

Sammendragstabeller med funksjonelle avhengigheter og begrensninger

Optimaliseringsrådgiveren bruker informasjon om forholdene mellom data, for eksempel funksjonelle avhengigheter og begrensninger, for å anbefale sammendragstabeller som inneholder samlede mål, og nivåattributtene som er nødvendige for at DB2-optimalisatoren skal kunne besvare spørringer på en effektiv måte.

Oversikt over optimaliseringsprosessen

Ved å optimalisere stjerneskjemaet eller snøfnuggskjemaet med DB2 Cube Views kan du øke ytelsen til SQL-spørringer i OLAP-stil. Optimaliseringsprosessen består av å opprette, implementere og vedlikeholde sammendragstabellene som blir anbefalt av optimaliseringsrådgiveren.

Utforming av metadata og optimalisering

Måten du utformer metadataobjektene på, inkludert nivåer og hierarkier, mål, kuber og optimaliseringssektorer, påvirker sammendragstabellene som veiviseren for optimaliseringsrådgiveren anbefaler.

Optimaliseringssektorer for kuber

En optimaliseringssektor er et valgfritt, men kraftig hjelpemiddel for optimaliseringsrådgiveren til å lage sammendragstabeller som er fokusert på de viktigste områdene i kubemodellen.

Analysere spørringer for optimaliseringssektorer

Optimaliseringssektorer er et kraftig verktøy for å øke spørringsytelsen, men de er bare effektive hvis de presist gjenspeiler brukernes spørringer.

Begrensningsdefinisjoner for optimalisering

Begrensninger gir verdifull informasjon om optimaliseringsrådgiveren og DB2-optimalisatoren. Du må definere informasjonsbegrensninger eller tvungne begrensninger for fremmednøkler og primærnøkler i stjerneskjemaet eller snøfnuggskjemaet.

Parametere for optimaliseringsrådgiveren

Informasjonen du oppgir for hver parametere i veiviseren for optimaliseringsrådgiveren, påvirker hvilke sammendragstabeller veiviseren anbefaler, og hvilke forbedringer du oppnår. Pass på å oppgi nøyaktig informasjon og vurder grundig kravene til kostnader og ytelse.

Optimalisere en kubemodell

Ved å optimalisere for spørringer som utføres på en kubemodell, kan du forbedre ytelsen for produkter som utsteder SQL-spørringer i OLAP-stil.

Eksempel på et SQL-skript for oppretting av sammendragstabeller

Veiviseren Optimaliseringsrådgiver har et SQL-skript for å opprette de anbefalte sammendragstabellene. SQL-skriptet inneholder de nødvendige SQL-kommandoene for å bygge en eller flere sammendragstabeller.

Teste resultater fra spørring

Ved hjelp av verktøyet db2batch for ytelsestesting (benchmark) i DB2 Universal Database kan du teste ytelsen til spørreresultater før og etter at du har opprettet sammendragstabellene med optimaliseringsrådgiveren.

Problemløsning for sammendragstabeller

Hvis spørringsytelsen ikke blir bedre etter at du har opprettet sammendragstabellen, kan du bruke funksjonen DB2EXPLAIN til å søke etter feil i rutingen av spørringene.

Vedlikehold av sammendragstabeller

Når dataene i basistabellene blir endret, må du oppdatere sammendragstabellene. Du kan oppdatere sammendragstabellene på to måter, med umiddelbar eller utsatt oppdatering.

Slette en sammendragstabell

DB2 Cube Views sletter ikke de tilknyttede sammendragstabellene når du sletter en kubemodell. Hvis du ikke bruker sammendragstabellene til noe annet, kan du slette tabellene for å frigjøre lagerplass.

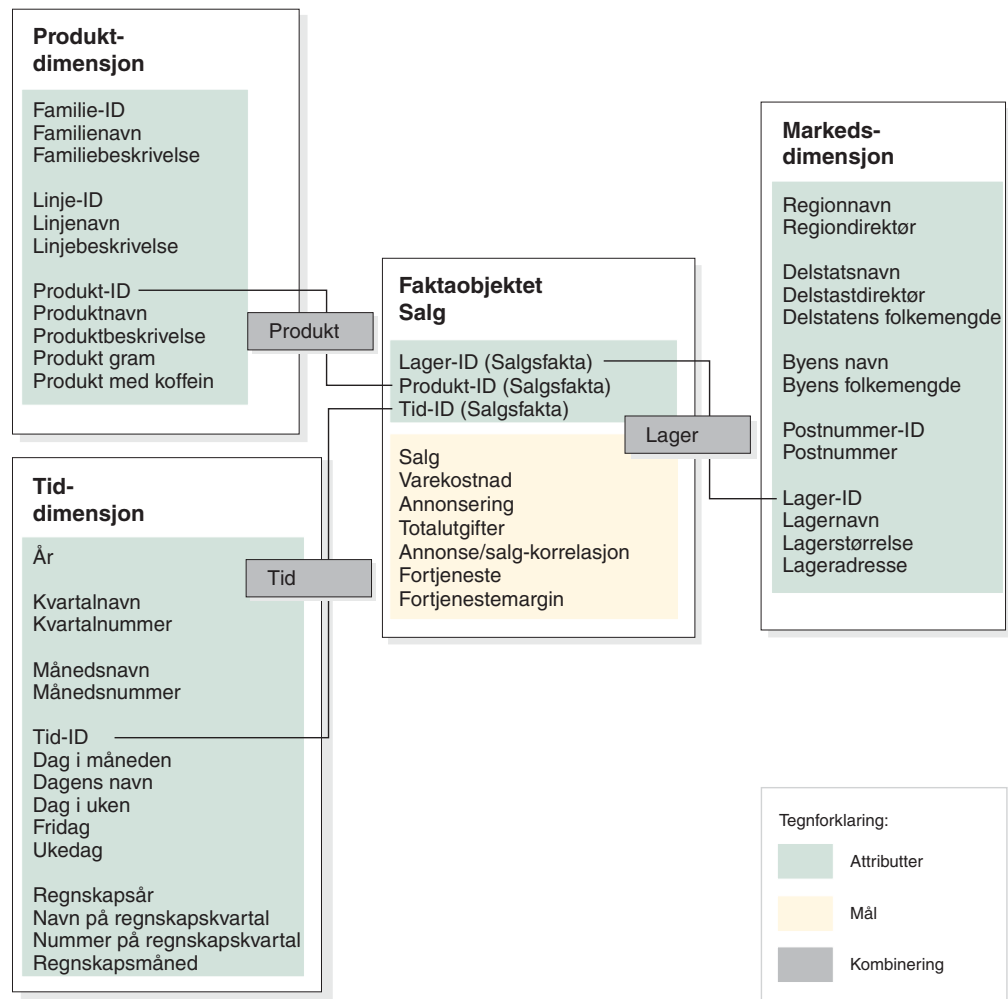
Sammendragstabeller

DB2 Cube Views bruker DB2-sammendragstabeller for å forbedre ytelsen til spørringer som utføres på kubemodeller og kuber. En sammendragstabell er en spesiell type materialisert spørretabell (MQT) som spesifikt tar med sammendragsdata.

Fordi optimaliseringsrådgiveren alltid anbefaler materialiserte spørretabeller som har sammendragsdata, blir termen sammendragstabell (summary table) brukt i dokumentasjonen til DB2 Cube Views som en beskrivelse av anbefalte materialiserte spørretabeller.

Du kan utføre kostbare beregninger og kombineringer for spørringer på forhånd og lagre disse dataene i en sammendragstabell. Når du kjører spørringer som kan bruke de forhåndsbergnede dataene, omdirigerer DB2 UDB spørringene til sammendragstabellen. En spørring trenger ikke å svare nøyaktig til forhåndsbergnene. Hvis du bruker enkle analysefunksjoner som SUM og COUNT, kan DB2 UDB samle dynamisk fra de forhåndsbergnede dataene. Mange forskjellige spørringer kan bruke den samme sammendragstabellen. Ved å bruke sammendragstabeller kan du få store ytelsesforbedringer for spørringer som trenger tilgang til data som brukes mye, eller for spørringer som omfatter sammendragsdata fra en eller flere dimensjoner eller tabeller.

I figur 23 på side 83 er det vist en kubemodell som er basert på et snøfnuggskjema med faktaobjektet Salg og dimensjonene Tid, Marked og Produkt. Faktaobjektet har mål og attributter, og hvert sett med dimensjoner er kombinert med faktaobjektet med en fakta-dimensjon-kombinering.



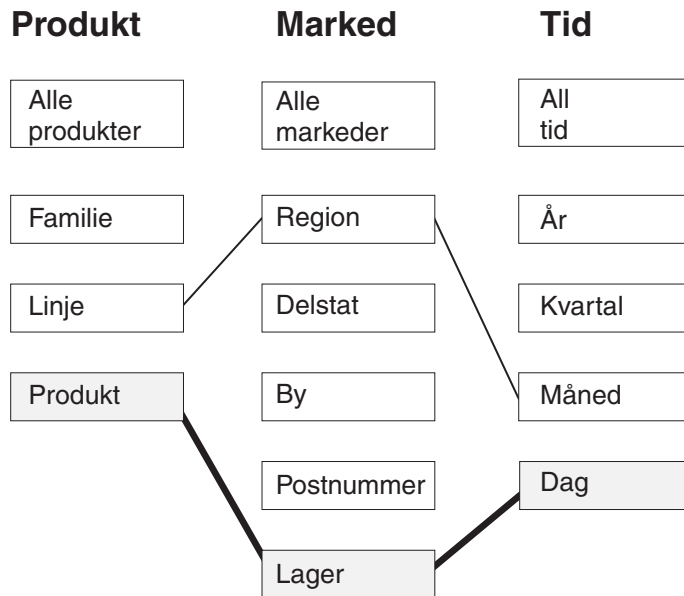
Figur 23. Kubemodell. Kubemodell med faktaobjektet Salg og dimensjonene Tid, Produkt og Marked

Hierarkiet for hver dimensjon i kubemodellen er vist i figur 24 på side 84. De uthevede boksene som er knyttet sammen med den tykke mørke streken bunn langs bunn av hierarkiene, representerer dataene som faktisk finnes i basistabellene. Salgsdataene er lagret på Dag-nivået, Postnummer-nivået og Produkt-nivået. Data over basisnivået i hierarkiet må samles. Hvis du utfører en spørring i basistabellen etter salgsdata fra en bestemt måned, må DB2 UDB summere de daglige salgsdataene dynamisk for å kunne returnere det månedlige salgstallet. Du kan for eksempel bruke følgende spørring hvis du ønsker å se salgsdataene for hver produktlinje for hver region for hver måned i 2004:

```
SELECT LINJE_ID, REGION_NAVN, MND_NUMMER, SUM(SALG)
FROM TID, LAGER, PLASSERING, PRODUKT, LINJE, SALGSFAKTA
WHERE SALGSFAKTA.LAGERID = LAGER.LAGERID
  AND LAGER.POSTNUMMERID = PLASSERING.POSTNUMMERID
  AND SALGSFAKTA.PRODUKTID = PRODUKT.PRODUKTID
  AND PRODUKT.LINJEID = LINJE.LINJEID
  AND SALGSFAKTA.TIDID = TID.TIDID
  AND ÅR = '2004'
GROUP BY LINJE_ID, MND_NUMMER;
```

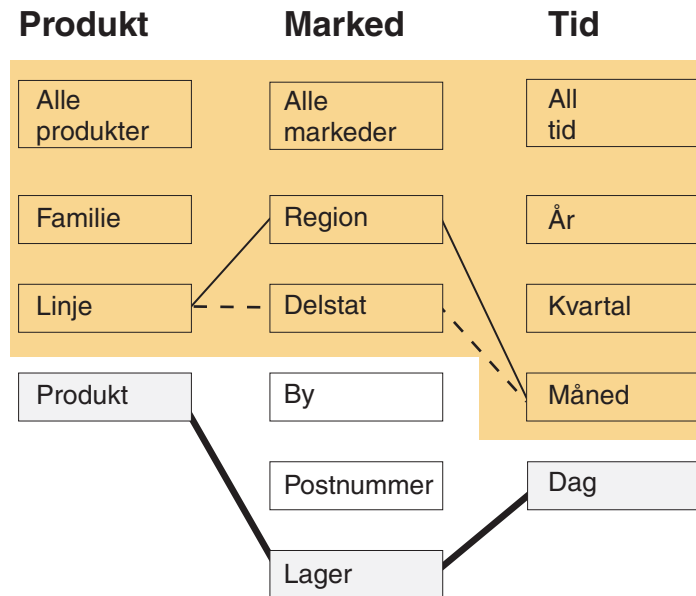
Den tynne streken som knytter sammensektoren Linje-Region-Måned i figur 24 på side 84, representerer sektoren som spørringen får tilgang til. Linje-Region-Måned er en sektor av kubemodellen som omfatter ett nivå fra hvert hierarki. Du kan

definere sammendragstabeller for å tilfredsstille spørringer på eller over en bestemt sektor. Du kan bygge en sammendragstabell for sektoren Linje-Region-Måned som spørringen trenger tilgang til. Alle andre spørringer som trenger tilgang til data på eller over denne sektoren, inklusive All tid, År, Kvartal, Alle markeder, Alle produkter og Familie, kan bruke denne sammendragstabellen med noe samling i tillegg. Hvis du spør etter mer detaljerte data under sektoren, for eksempel Dag eller By, kan ikke DB2 UDB bruke sammendragstabellen til denne spørringen.



Figur 24. Spørringssektor. Hierarkiene Produkt, Marked og Tid. Viser sektoren Linje-Region-Måned og at basisdataene finnes i sektoren Produkt-Lager-Dag.

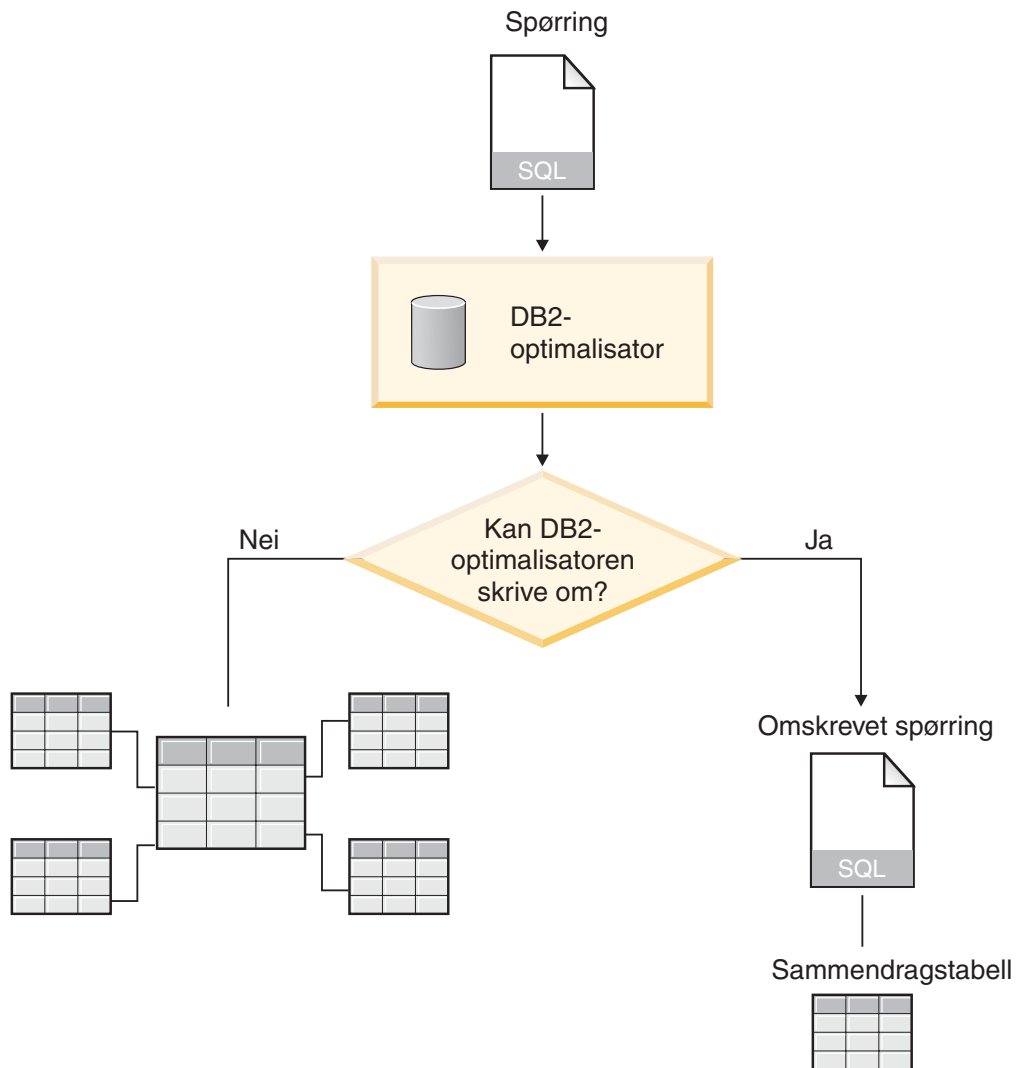
I figur 25 på side 85 definerer den stiplede linjen sektoren Linje-Delstat-Måned. En sammendragstabell som er bygd for sektoren Linje-Delstat-Måned, kan tilfredsstille alle spørringer som trenger tilgang til data på eller over denne sektoren. Alle dataene som en sammendragstabell som er bygd for sektoren Linje-Delstat-Måned kan benyttes for, er med i de øverste boksene som er uthevet.



Figur 25. Sammendragstabellsektor. Hierarkiene Produkt, Marked og Tid. En sammendragstabell som er bygd for sektoren Linje-Delstat-Måned, kan benyttes for de uthevede dataene.

Omskrivingsfunksjonen i SQL-kompilatoren i DB2 kjenner til de eksisterende sammendragstabellene, og kan skrive om spørringer automatisk slik at de leser fra sammendragstabellen i stedet for basistabellene. Omskrevne spørringer er vanligvis mye raskere fordi sammendragstabellene vanligvis er mye mindre enn basistabellene og inneholder data som allerede er samlet. En bruker kan fortsette å skrive spørringer mot basistabellene. DB2 UDB vil bestemme når en sammendragstabell skal brukes til en bestemt spørring, og vil skrive om brukerens spørring slik at den går mot sammendragstabellene i stedet, slik det er vist i figur 26 på side 86. Den omskrevne spørringen gjør bruk av en sammendragstabell som inneholder forhåndsregnedte data. En sammendragstabell er ofte betydelig mindre, og dermed betydelig raskere, enn basistabellene, og returnerer de samme resultatene som basistabellene.

Du kan bruke funksjonen DB2 EXPLAIN til å se om spørringen ble omdirigert, og i så fall hvilken tabell den ble omdirigert til.



Figur 26. Omskriving av spørring. DB2 UDB-prosessen for omskriving av en spørring

Spørringen for å se salgsdataene for hver produktlinje i hver region, for hver måned i 2004, kan skrives om til å bruke sammendragstabellen som er bygd for sektoren Linje-Region-Måned. Den opprinnelige spørringen:

```
SELECT LINJE_ID, REGION_NAVN, MND_NUMMER, SUM(SALG)
FROM TID, LÅGER, PLASSERING, PRODUKT, LINJE, SALGSFAKTA
WHERE SALGSFAKTA.LÅGERID = LÅGER.LÅGERID
  AND LÅGER.POSTNUMMERID = PLASSERING.POSTNUMMERID
  AND SALGSFAKTA.PRODUKTID = PRODUKT.PRODUKTID
  AND PRODUKT.LINJEID = LINJE.LINJEID
  AND SALGSFAKTA.TIDID = TID.TIDID
  AND ÅR = '2004'
GROUP BY LINJE_ID, MND_NUMMER;
```

Den omskrevne spørringen:

```
SELECT LINJE_ID, REGION_NAVN, MND_NUMMER, SUM(SALG)
FROM SAMMENDRAGSTABELL1
WHERE ÅR = '2004'
GROUP BY LINJE_ID, OMRÅDENAVN, MND_NUMMER;
```

Den omskrevne spørringen er mye enklere og raskere å fullføre fordi dataene er aggregert på forhånd og mange av tabellene er beregnet på forhånd, slik at DB2

UDB bare trenger tilgang til en liten tabell i stedet for seks tabeller, inkludert en stor faktatabell. Besparelsene med sammendragstabeller kan være enorme, spesielt for skjemaer med store faktatabeller. En faktatabell med 1 milliard rader kan for eksempel være forhåndssamlet til en sammendragstabell på bare 1 million rader, og beregningene som denne samlingen medfører, skjer bare en gang, i stedet for hver gang en spørring blir sendt. Det går langt raskere å bruke en sammendragstabell som er 1000 ganger mindre, enn å bruke de store basistabellene.

I dette eksempelet viser figur 27 sammendragstabellen for sektoren Linje-Delstat-Måned. DB2 UDB må beregne data for Region fra nivået Delstat i stedet for fra det lavere nivået Lager, så sammendragstabellen har færre rader enn basistabellene, siden det er færre delstater enn det er lagere. DB2 UDB trenger ikke å utføre noen tilleggsberegninger for å returnere salgsdata per Måned og Linje fordi dataene allerede er samlet på disse nivåene. Denne spørringen er fullstendig tilfredsstillende av dataene i sammendragstabellen som kombinerer tabellene som brukes i spørringen, og kombineringsene trenger ikke å utføres når spørringen blir utstedt. For mer komplekse spørringer kan ytelsesforbedringen være stor.

Navn på region	Navn på delstat	Linje-ID	År	Kvartalsnummer	Kvartalnavn	Månednummer	Salg	Varekostnad	Annonsering	Totalutgifter	Fortjeneste
West	Idaho	054	2004	1	2. kv.	2	9700	2500	700	3200	6500
East	Maine	102	2004	2	2. kv.	5	3000	500	200	700	2300
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Figur 27. *Sammendragstabell.* Eksempel på sammendragstabellen som blir opprettet for sektoren Linje-Region-Måned

I noen tilfeller kan en spørring gjelde et attributt som er beslektet med et attributt som er tatt med i sammendragstabellen. DB2-optimalisatoren kan bruke funksjonelle avhengigheter og begrensninger for å kombinere sammendragstabellen dynamisk med den tilhørende dimensjonstabellen.

Når optimaliseringsrådgiveren anbefaler en sammendragstabell, tar den med alle målene i kubemodellen. I dette eksempelet har objektet Salgsfakta bare fem mål, inkludert Salg, Varekostnad, Annonsering, Totalutgifter og Fortjeneste, som alle er tatt med i sammendragstabellen. Hvis du definerer femti mål for kubemodellen, blir alle de femti målene tatt med i sammendragstabellen.

Optimaliseringsrådgiveren trenger ikke å ta med alle de beslektede attributtene som er definert for et nivå i sammendragstabellen, fordi DB2 Cube Views definerer funksjonelle avhengigheter mellom attributtene i et nivå.

Sammendragstabeller med funksjonelle avhengigheter og begrensninger

Optimaliseringsrådgiveren bruker informasjon om forholdene mellom data, for eksempel funksjonelle avhengigheter og begrensninger, for å anbefale sammendragstabeller som inneholder samlede mål, og nivåattributtene som er nødvendige for at DB2-optimalisatoren skal kunne besvare spørringer på en effektiv måte.

DB2 Cube Views definerer funksjonelle avhengigheter, der det er mulig, mellom attributter i et nivå. Når du definerer et nivå, definerer du et forhold mellom nivånøkkelattributtene og de andre attributtene i nivået (standardattributt og beslektede attributter). Forholdet angir at nivånøkkelattributter kan brukes sammen for å bestemme de andre attributtene i nivået. DB2 Cube Views dokumenterer forholdet mellom nivåets attributtet ved å definere funksjonelle avhengigheter mellom attributtene. DB2 UDB og DB2 Cube Views kan bruke forholdene, definert ved hjelp av funksjonelle avhengigheter, til å utføre intelligent optimalisering av dataene.

Du må kontrollere at de underliggende dataene for nivåattributtene er funksjonelt avhengige slik det er beskrevet av den funksjonelle avhengigheten. DB2 UDB verifiserer ikke gyldigheten av de funksjonelle avhengighetene.

Hvis det finnes en funksjonell avhengighet mellom et nivånøkkelattributt og nivåets beslektede attributter, kan optimaliseringsrådgiveren inkludere nivånøkkelattributtet uten de beslektede attributtene i sammendragstabellen. Spørringer som er interessert i nivåets beslektede attributter, kan fremdeles rutes til sammendragstabellen, fordi DB2-optimalisatoren kombinerer sammendragstabellen med dimensjonstabellen når spørringen utstedes, for å opprette det endelige resultatsettet.

Du kan for eksempel bruke en spørring som er svært lik spørringen som er beskrevet i "Sammendragstabeller" på side 82, hvis du ønsker å se salgsdataene for hver produktlinje, i alle regioner, for hver måned i 2004. Den følgende spørringen er forskjellig i og med at den grupperer resultatene etter Listenavn i stedet for Linje-ID, og Månedsnavn i stedet for Månedsnummer. Resultatsettet er enklere å bruke, men krever funksjonelle avhengigheter og begrensninger for å få tilgang til dataene.

```
SELECT LINJENAVN, REGIONNAVN, MÅNEDSNAVN, SUM(SALG)
FROM TID, LAGER, PLASSERING, PRODUKT, LINJE, SALGSFAKTA
WHERE SALGSFAKTA.LAGERID = LAGER.LAGERID
  AND LAGER.POSTNUMMERID = PLASSERING.POSTNUMMERID
  AND SALGSFAKTA.PRODUKTID = PRODUKT.PRODUKTID
  AND PRODUKT.LINJEID = LINJE.LINJEID
  AND SALGSFAKTA.TIDID = TID.TIDID
  AND ÅR = '2004'
GROUP BY LINJENAVN, REGIONNAVN, MÅNEDSNAVN;
```

DB2-optimalisatoren omskriver spørringen slik at den kombinerer sammendragstabellen med tilhørende dimensjonstabeller, slik det er vist i figur 28 på side 89.

Nivået Line inneholder disse attributtene:

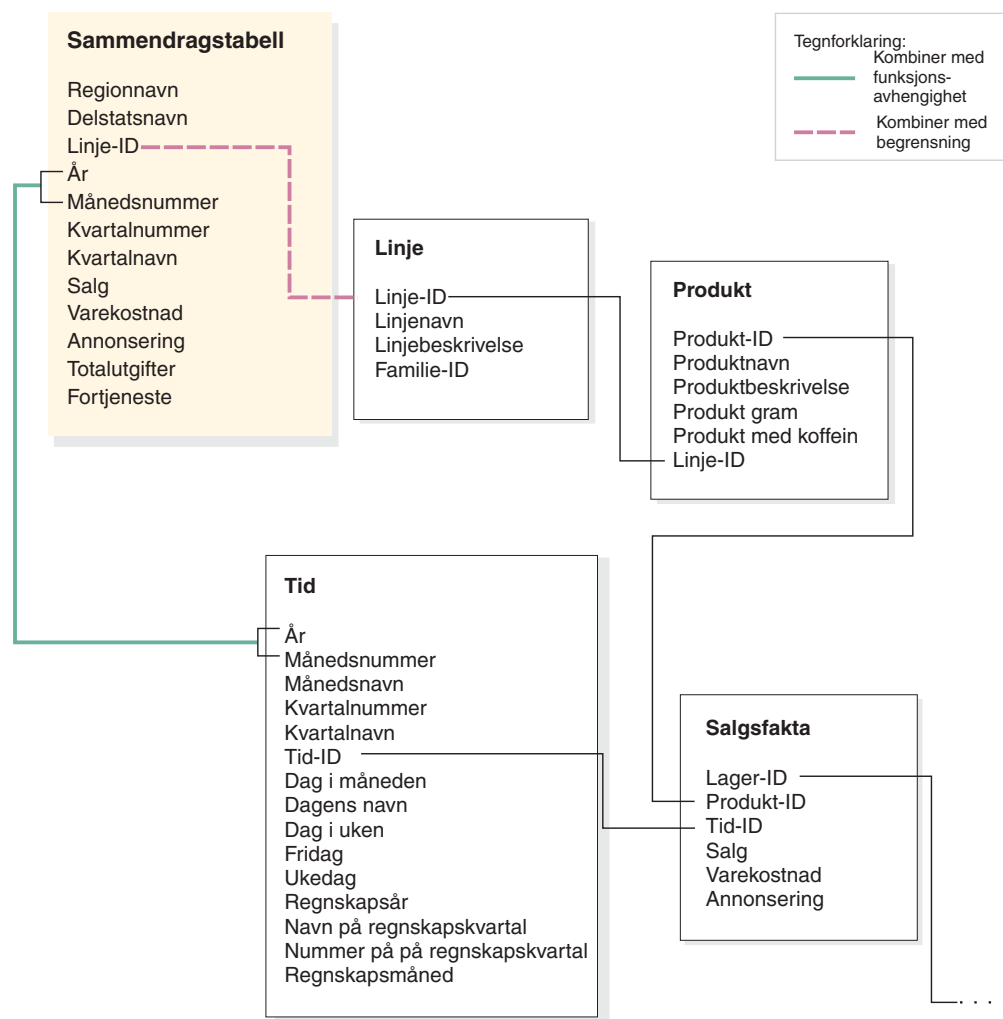
- Linje-ID som nivånøkkelattributtet
- Linjenavn som standardattributtet
- Linjebeskrivelse som det beslektede attributtet

Linje-ID er primærnøkkelen til tabellen Linje, så det finnes en begrensning på kolonnen Linje-ID. Siden det finnes en begrensning for kolonnen Linje-ID, oppretter ikke DB2 Cube Views en funksjonell avhengighet for dette nivået. DB2-optimalisatoren bruker begrensningen til å kombinere sammendragstabellen med tabellen Line, og til å få tilgang til linjensnavndataene for spørringens resultatsett.

Nivået Måned inneholder disse attributtene:

- År og Månednummer som nivånøkkelattributtene
- Månednavn som standardattributtet

Det finnes en funksjonell avhengighet for nivået Måned, som angir at Månednavn er funksjonelt avhengig av kombinasjonen av År og Månednummer. Kolonnene År og Månednummer er ikke en del av primærnøkkelen eller fremmednøkkelen for tabellen Tid, så det er ingen begrensning på disse kolonnene. DB2-optimalisatoren bruker den funksjonelle avhengigheten mellom månednavn og kombinasjonen av år og månednummer, til å kombinere sammendragstabellen med tidtabellen, og til å få tilgang til månednavndataene for spørringens resultatsett.



Figur 28. Kombineringer. DB2-optimalisatoren kombinerer sammen dragstabelen dynamisk med tilhørende dimensjonstabeller når en spørring blir utstedt.

DB2-optimalisatoren skriver om spørringen slik:

```
SELECT KV4.LINJENAVN, KV4.REGIONNAVN, KV4.MÅNEDSNAVN, SUM(KV.SALG)
FROM (
  SELECT DISTINCT KV1.LINJENAVN, KV3.REGIONNAVN, KV2.MÅNEDSNAVN,
    KV3.SALG, KV2.ÅR, KV2.MÅNEDSNUMMER
  FROM LINJE AS KV1, TID AS KV2, SUMMARYTABLE1 AS KV3
  WHERE (KV3.ÅR=2004)
    AND KV.LINJEID=KV1.LINJEID)
    AND (KV.MÅNEDSNUMMER=KV2.MÅNEDSNUMMER)
    AND (2004=KV2.ÅR)
) AS KV4
GROUP BY KV4.LINJENAVN, KV4.REGIONNAVN, KV4.MÅNEDSNAVN
```

Den omskrevne spørringen kombinerer bare tre tabeller i stedet for de seks tabellene i den opprinnelige spørringen, og den omskrevne spørringen trenger ikke tilgang til den store faktatabellen som vanligvis er temmelig stor og langsom. Den omskrevne spørringen går langt raskere, fordi sammendragstabellen allerede inneholder de samlede måldataene.

Oversikt over optimaliseringsprosessen

Ved å optimalisere stjerneskjemaet eller snøfnuggskjemaet med DB2 Cube Views kan du øke ytelsen til SQL-spørringer i OLAP-stil. Optimaliseringsprosessen består av å opprette, implementere og vedlikeholde sammendragstabellene som blir anbefalt av optimaliseringsrådgiveren.

Optimaliseringsrådgiveren kan hjelpe deg med å optimalisere kubemodellene ved å anbefale sammendragstabeller. DB2 UDB-sammendragstabeller kan forbedre ytelsen i en spørring fordi de inneholder forhåndsberregnede resultater fra en eller flere tabeller som kan brukes i en spørring. Kostbare tabellkombineringer og kompliserte beregninger kan forhåndsbehandles og lagres i en sammendragstabell slik at fremtidige spørringer som bruker disse samlingene, kan kjøres mye raskere. Du finner opplysninger om sammendragstabeller under "Sammendragstabeller" på side 82.

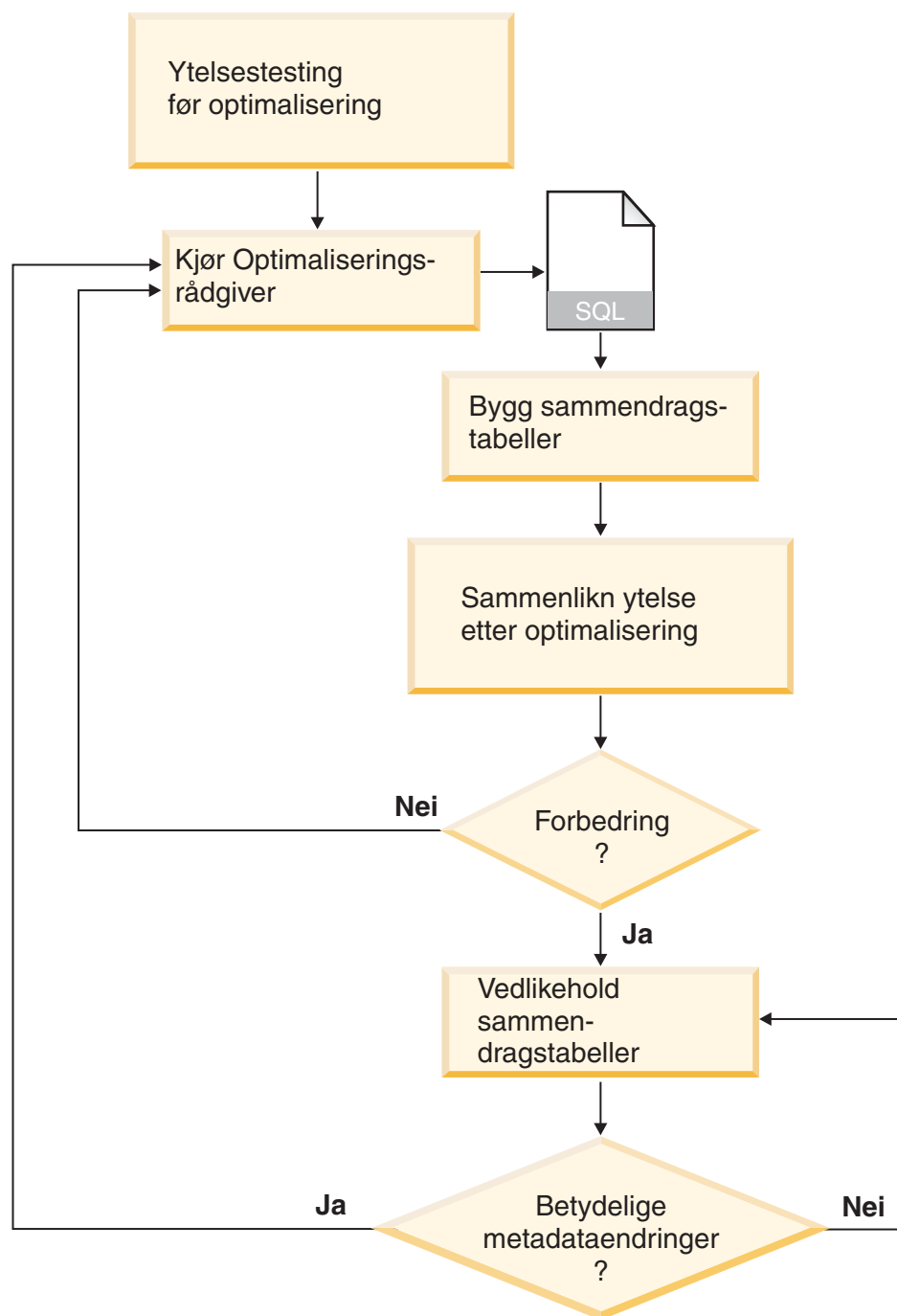
Optimaliseringsrådgiveren vil analysere metadataene og informasjonen du oppgir til veiviseren, og anbefale de riktige sammendragstabellene. Når du har kjørt optimaliseringsrådgiveren, får du en SQL-fil som kan bygge settet av anbefalte sammendragstabeller. Du har muligheten til å endre SQL-filen før du kjører den for å opprette sammendragstabellen.

Kjøring av optimaliseringsrådgiveren er bare ett trinn i optimaliseringsprosessen. Før du begynner optimaliseringen, må du ta hensyn til flere saker. Blant annet disse:

- Hvordan du effektivt kan bruke DB2-begrensninger på basistabellene
- Hvordan du kan definere kubemodellen slik at den følger valideringsreglene for optimalisering
- Hvilken type spørringer du vil optimalisere for
- Hvor mye plass du vil bruke
- Hvordan du vil vedlikeholde sammendragstabellene slik at dataen de inneholder, er oppdatert

Du må definere begrensninger for basistabellene før du kan optimalisere. Du finner flere opplysninger om hvilke typer av begrensninger du trenger, i "Begrensningsdefinisjoner for optimalisering" på side 107.

Flere deler av optimaliseringsprosessen er interaktive, og må kanskje gjentas for å finjustere og vedlikeholde ytelsesforbedringene. figur 29 viser en oversikt over hovedtrinnene i optimaliseringsprosessen.



Figur 29. Optimaliseringsprosessen. Oversikt over hovedtrinnene i optimaliseringsprosessen

Optimaliseringsprosessen omfatter disse generelle oppgavene:

- **Måle ytelsen**

Før du kjører optimaliseringsråd-giveren i DB2 Cube Views, bør du måle den gjeldende ytelsen for et bestemt sett av typiske spørringer. Måling av ytelsen er et valgfritt trinn som gir et resultat som du kan bruke til å måle om optimaliseringen er vellykket. Ved hjelp av verktøyet db2batch for ytelsestesting

(benchmark) i DB2 UDB kan du opprette en ytelsestest. Du finner opplysninger om hvordan du bruker db2batch under "Teste resultater fra spørring" på side 118. Du kjører eksempelspørringer for å fullføre ytelsestesten, men optimaliseringsrådgiveren krever ikke eksempelspørringer fordi den er metadatabasert og kan gi anbefalinger uten å kjenne de bestemte spørringene som vil bli sendt.

- **Kjøre veviseren for optimaliseringsrådgiveren**

Du oppgir mange viktige parametere til veviseren, blant annet hvilke typer av spørringer du vil optimalisere for, lagerplass- og tidsbegrensninger, oppdateringsmetode og tabellplassplasseringer. Du finner flere opplysninger om disse parametervalgene under "Parametere for optimaliseringsrådgiveren" på side 112. Optimaliseringsrådgiveren gir anbefalinger basert på informasjonen du oppgir, metadata, DB2-statistikk og eventuelt dataavlesing hvis du tillater det. Optimaliseringsrådgiveren vurderer parameterne du oppgir og genererer to SQL-filer. En SQL-fil inneholder SQL-kommandoene for å bygge et sett av anbefalte sammendragstabeller. Den andre SQL-filen inneholder SQL-kommandoene for å oppdatere de anbefalte sammendragstabellene.

- **Opprette sammendragstabellene**

Du kan opprette sammendragstabellene umiddelbart etter at du har kjørt veviseren, eller du kan tilføye operasjonen i din normale plan for vedlikeholdet av databasen. Oppretting av sammendragstabellene kan ta lang tid og kreve store behandlingsressurser. Når sammendragstabellene er bygd, må du kontrollere at ytelsen er forbedret for spørringene mot den optimaliserte kubemodellen. Kjør det samme settet av eksempelspørringer som du kjørte før optimaliseringen, og sammenlikn ytelsesresultatene. Hvis du ikke ser en betydelig forbedret ytelse, bør du kjøre optimaliseringsrådgiveren på nytt og tildele mer lagerplass eller tid eller begge deler, eller endre andre innstillinger. Du finner opplysninger om hvordan du verifiserer og analyserer ytelsesresultatene, under "Teste resultater fra spørring" på side 118.

- **Vedlikeholde sammendragstabellene**

Når sammendragstabellene er opprettet, må du utføre jevnlig vedlikehold av tabellene for å sikre at de holder seg synkronisert med dataene. Når du kjører optimaliseringsrådgiveren, velger du umiddelbar oppdatering eller utsatt oppdatering.

Alternativ for umiddelbar oppdatering

Hvis du velger alternativet for umiddelbar oppdatering, holder DB2 UDB basistabellene og sammendragstabellene synkronisert og oppdaterer sammendragstabellene inkrementelt når de underliggende tabellene endres. DB2 UDB støtter inkrementelt vedlikehold for enkle samlinger som SUM og COUNT. For andre samlinger anbefaler optimaliseringsrådgiveren sammendragstabeller som bruker alternativet for utsatt oppdatering, uansett hvilket oppdateringsalternativ du har valgt.

Alternativ for utsatt oppdatering

Hvis du velger utsatt oppdatering, bygger du sammendragstabellene på nytt for å oppdatere dem. Du kan selv velge når du vil oppdatere sammendragstabellene. Hvis du gjør større endringer i basistabellene, kan det være mer effektivt å utsette oppdateringen enn å utføre den inkrementelt.

Når du velger mellom disse alternativene, må du gjøre en avveining mellom hvilke ressurser du kan tildele til vedlikeholdet, og hvor nøyaktig dataene må synkroniseres. Du finner mer informasjon om fornyingsalternativene umiddelbart og utsatt, under "Vedlikehold av sammendragstabeller" på side 121.

- **Periodisk reevaluering**

Du må med jevne mellomrom reevaluere sammendragstabellene for å sikre at de fortsatt imøtekommer dine behov. Hvis metadataene blir sterkt endret ved at du tilføyer eller oppdaterer en kubemodell, er det mulig at du må kjøre optimaliseringsrådgiveren på nytt og bygge et nytt sett av sammendragstabeller:

- Hvis du tilføyer et metadataobjekt som en ny dimensjon eller et nytt mål, kan ikke spørringer som trenger tilgang til data fra det nye objektet, bruke de eksisterende sammendragstabellene. Spørringer som ikke bruker det nye objektet, kan imidlertid fortsette å bruke sammendragstabellene.
- Hvis du oppdaterer et metadataobjekt til å omfatte data som det tidligere ikke er optimalisert for, kan ikke spørringer som trenger tilgang til det oppdaterte objektet, bruke sammendragstabellene.
- Hvis du sletter ett eller flere objekter, endres ikke effektiviteten til sammendragstabellene, men du bruker lagerplass på samlinger som ikke lenger brukes.

I tillegg til å kjøre veiviseren på nytt ved større endringer av metadataene er det mulig du må gjøre det samme hvis de typene av spørringer som du utfører jevnlig, blir endret og ikke lenger er den typen du optimaliserte for.

Hver gang du kjører veiviseren og bygger nye tabeller, må du fullføre hele optimaliseringsprosessen på nytt, også opprette en ytelsestest og analysere ytelsen til sammendragstabellene.

Hvis du sletter en kubemodell, kan du også slette de tilknyttede sammendragstabellene hvis de ikke brukes til andre formål. DB2 Cube Views sletter ikke sammendragstabeller når den tilknyttede kubemodellen blir slettet. Du finner opplysninger om hvordan du sletter en sammendragstabell, i “Slette en sammendragstabell” på side 122.

Utforming av metadata og optimalisering

Måten du utformer metadataobjektene på, inkludert nivåer og hierarkier, mål, kuber og optimaliseringssektorer, påvirker sammendragstabellene som veiviseren for optimaliseringsrådgiveren anbefaler.

Generelt vil du definere faktaobjekter, dimensjoner og kombineringer på en måte som svarer til strukturen dataene har. Du har få valg i den grunnleggende strukturen til disse høynivåobjektene i kubemodellen, så du kan sjelden forbedre de anbefalte sammendragstabellene ved å endre disse objektene. Du har større fleksibilitet når det gjelder hvordan du definerer nivåer og hierarkier, mål, kuber og optimaliseringssektorer.

Nivåer og hierarkier

Hvis det er mulig, definerer du nivåene ved hjelp av den ideelle modelleringsmetoden som er beskrevet under “Nivåer” på side 27. Hvis du bruker den ideelle modelleringsmetoden, kan du redusere lagerplassen de anbefalte sammendragstabellene bruker, og den midlertidige plassen som brukes til å oppdatere de anbefalte sammendragstabellene.

Mål

Hvis du har begrenset med lagerplass, kan du velge å bare ta med kritiske mål i kubene og slette mål du ikke forventer å bruke jevnlig, eller som ikke har noen forretningsverdi, fordi jo flere mål du definerer i kubene, jo større blir sammendragstabellene.

Hvilke typer mål, enten distributive eller ikke-distributive, du tar med i kubene, kan også påvirke optimaliseringen:

- Distributive mål kan alltid samles fra ett nivå til det neste. SUM(Salg) for Kvartal kan for eksempel beregnes ved å summere de månedlige salgsdataene.
- Ikke-distributive mål, for eksempel standardavvik, må alltid beregnes direkte fra basisdataene, og kan ikke aggregeres fra ett nivå til det neste.

Generelt har optimaliseringsrådgiveren og DB2-optimalisatoren større fleksibilitet og flere muligheter når kubemodellen bare inneholder distributive mål. Du kan optimalisere en kubemodeller som inneholder ikke-distributive mål, men du kan få bedre optimaliseringsresultater hvis du utelater de ikke-distributive målene som ikke er nødvendige, fra kubene.

Kuber

Opprett kuber som samsvarer med behovene. Kuber fungerer mest effektivt i optimaliseringen når du bygger en kube som fokuserer på et viktig område av kubemodellens data.

Hvis du kjenner til en eller flere egenskaper ved spørringene som brukerne ofte utsteder, kan du oppgi denne informasjonen i optimaliseringssektorer for optimaliseringsrådgiveren. Optimaliseringsrådgiveren anbefaler sammendragstabeller som forbedrer spørringer som sendes til de bestemte områdene av kubene. En optimaliseringssektor er definert som et sett med spesifikke nivåer eller **Vilkårlig** nivåangivelse for hver kubedimensjon, og spørringstypen som er forventet i sektoren, for eksempel neddrilling, rapport, MOLAP-uttrekking, hybriduttrekking eller gjennomdrilling.

I OLAP-senteret kan du oppgi at en kube vanligvis brukes for en type spørring, for eksempel neddrilling. Når du oppgir en spørringstype for en kube, definerer OLAP-senteret en optimaliseringssektor med den typen du oppgav, og parameteren **Vilkårlig** for hver kubedimensjon. Du finner flere opplysninger om optimaliseringssektorer under "Optimaliseringssektorer for kuber" på side 95.

Listen nedenfor beskriver hvilke typer spørringer du kan oppgi for en kube i OLAP-senteret:

Neddrillingsspørringer

Neddrillingsspørringer får vanligvis tilgang til data som er fokusert øverst i en kubemodell. Spørringer kan gå til et hvilket som helst nivå i kubemodellen. Når brukere driller dypt ned i en dimensjon, holder de seg vanligvis mye høyere i de andre dimensjonene. Optimalisering for neddrillingsspørringer vil få best innvirkning på spørringer som holder seg i de øverste nivåene av kubemodellen. ROLAP-regnearkapplikasjoner (Relational OLAP) blir vanligvis brukt til å utføre neddrillingsspørringer. En bruker av en regnearkapplikasjon kan for eksempel starte med å få tilgang til inntekten for alle regioner og alle produkter for året 2004. Deretter kan brukeren bevege seg lengre inn i dataene ved å spørre etter inntekt per kvartal i alle regioner og for hvert land.

Ytelse er vanligvis svært viktig for disse typene av spørringer fordi de utføres i sanntid av en bruker som må vente på behandlingen av resultatene.

Rapportspørringer

Rapportspørringer har like stor mulighet til å få tilgang til alle deler i kubemodellen. Rapportspørringer utføres ofte satsvist i

grupper. Spøringsytelsen er vanligvis ikke like kritisk for rapportspøringer som for neddrillingsspøringer fordi det er mindre sannsynlig at brukeren venter på å få et svar på hver enkelt spørring.

MOLAP-uttrekkingsspøringer

MOLAP-uttrekkingsspøringer bruker enten basisnivået til en kube eller optimaliseringssektoren som er definert for kuben. Kuben brukes til å laste inn data i en MOLAP-datastruktur. Kubeoptimaliseringssektoren er logisk tilordnet til uttrekkingssektoren som du laster dataene inn i MOLAP-applikasjonen med for videre behandling.

Hvis du oppgir MOLAP-uttrekkingstype for en kube i OLAP-senteret, eller hvis du oppgir en optimaliseringssektor med alternativet **Vilkårlig** på tvers av alle kubedimensjonene, oppretter optimaliseringsrådgiveren sammendragstabeller som optimaliserer dataene for uttrekking på kubens basisnivå. Kontroller at kubens basisnivåer er tilordnet til sektoren du trekker ut ved.

Avanserte innstillinger

Hvis du oppgir Avanserte innstillinger for en kube i OLAP-senteret, kan du oppgi optimaliseringssektorer for bestemte områder i kuben som spøringer ofte bruker.

Hvis du kjenner til en eller flere egenskaper ved spørringene som ofte blir brukt, kan du oppgi denne informasjonen i optimaliseringssektorer for optimaliseringsrådgiveren. Optimaliseringsrådgiveren vurderer disse sektorene ved anbefaling av sammendragstabeller. En optimaliseringssektor er definert som et sett med spesifikke nivåer eller en vilkårlig nivåangivelse for hver kubedimensjon, og spørringstypen som er forventet i sektoren, for eksempel neddrilling, rapport, MOLAP-uttrekking, hybriduttrekking eller gjennomdrilling. Du finner opplysninger om optimaliseringssektorer under "Optimaliseringssektorer for kuber"

Optimaliseringssektorer for kuber

En optimaliseringssektor er et valgfritt, men kraftig hjelpemiddel for optimaliseringsrådgiveren til å lage sammendragstabeller som er fokusert på de viktigste områdene i kubemodellen.

Kuber kan ofte ha mange kubedimensjoner. Ved å oppgi en eller flere optimaliseringssektorer kan du oppgi hvilket område av kuben har størst spøringsaktivitet. En optimaliseringssektor er definert som et sett med ett eller flere nivåer, og spørringstypen som er forventet i sektoren, for eksempel neddrilling, rapport, MOLAP-uttrekking, hybriduttrekking eller gjennomdrilling. For alle typer spøringer gir det en positiv effekt å definere en optimaliseringssektor, men det har sannsynligvis størst effekt på rapportspøringer.

Du må oppgi ett alternativ per kubedimensjon når du oppretter optimaliseringssektoren. Bruk de følgende retningslinjene når du oppgir et alternativ for hver kubedimensjon:

- Oppgi et spesifikt nivå i en kubedimensjon, for eksempel Måned i kubedimensjonen Tidsrom, hvis du vet at det oppgitte nivået er viktig eller at det ofte blir tatt med i spøringer.

- Oppgi **Alle** i en kubedimensjon hvis den høyeste samlingen av kubedimensjonen er viktig eller mye brukt i spørringer.
- Oppgi **Vilkårlig** i en kubedimensjon hvis ingen nivåer er betydelig viktigere enn andre nivåer i kubedimensjonen, hvis det blir spurt på mange nivåer i kubedimensjonen, eller hvis du ikke vet hvor ofte det blir spurt i de ulike nivåene i kubedimensjonen.

I en kube med 10 kubedimensjoner kan du for eksempel ha viktige nivåer i bare to kubedimensjoner, så du kan oppgi spesifikke nivåer i disse to kubedimensjonene og **Vilkårlig** i alle andre kubedimensjoner.

De følgende avsnittene beskriver eksempler på optimaliseringssektorer for hver spørringstype, og sammendragstabellene som optimaliseringsrådgiveren kan anbefale.

Optimaliseringssektorer for neddrilling

En optimaliseringssektor av typen neddrilling tyder på at brukerne ofte driller ned til nivåene som er definert i kubedimensjonene. Derfor bør optimaliseringsrådgiveren inkludere disse nivåene i en eller flere av de anbefalte sammendragstabellene. Siden typen er neddrilling, er det mulig at optimaliseringsrådgiveren optimaliserer for både grunne og dype spørringer i noen av kubedimensjonene.

Recommended: Du bør bare oppgi et bestemt nivå for en kubedimensjon hvis du vet at dette nivået er spesielt viktig. Generelt bruker du alternativet **Vilkårlig** for de fleste kubedimensjonene, og velger bare et bestemt nivå når dette nivået er involvert i intens spørringsaktivitet.

I figur 30 på side 97 ser du et eksempel på en optimaliseringssektor for neddrilling, og sektorene som optimaliseringsrådgiveren kan anbefale sammendragstabeller for. Optimaliseringssektoren er definert på nivået **Vilkårlig** i kubedimensjonen Produkt og kubedimensjonen Marked, og på nivået **Måned** i kubedimensjonen Tid. Denne optimaliseringssektoren tyder på at brukerne ofte driller ned til nivået **Måned**, men at de ikke har et bestemt neddrillingsmønster i kubedimensjonene Produkt eller Marked.

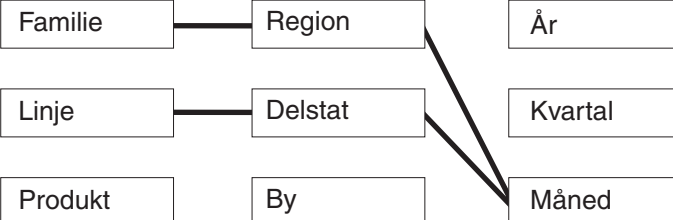
De mulige sammendragstabellanbefalingene omfatter to samlingsnivåer. Ett samlingsnivå er definert på sektoren Linje-Delstat-Måned, og det andre samlingsnivået er definert på sektoren Familie-Region-Måned. Begge samlingsnivåene inkluderer samlinger på nivået **Måned** i dimensjonen Tid, men gir ulike samlingsnivåer i de andre to dimensjonene. Den øverste sektoren gir størst ytelsesforbedring for grunne neddrillingsspørringer, fordi spørringene kan tilfredsstilles umiddelbart, og DB2 UDB trenger ikke å samle opp. Det laveste samlingsnivået gir ytelsesforbedring for spørringer med neddrilling til et dypere nivå.

Sektor for neddrilling

Produkt	Marked	Tid
Alle produkter	Alle markeder	All tid
Familie	Region	År
Linje	Delstat	Kvartal
Produkt	By	Måned

Mulig anbefaling

Produkt	Marked	Tid
Alle produkter	Alle markeder	All tid
Familie	Region	År
Linje	Delstat	Kvartal
Produkt	By	Måned



Figur 30. Neddrilling. Optimaliseringssektor for neddrilling og mulige sammendragstabellanbefalinger

Optimaliseringssektorer for rapporter

En optimaliseringssektor av typen rapport tyder på at brukerne ofte oppretter rapporter på de nivåene som er definert i kubedimensjonene. Derfor bør optimaliseringsrådgiveren inkludere disse nivåene i en eller flere av de anbefalte sammendragstabellene.

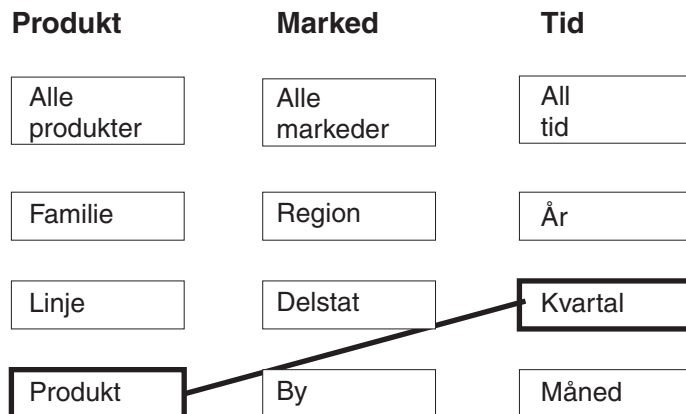
Recommended: Du bør bare oppgi et bestemt nivå for en kubedimensjon hvis du vet at dette nivået er spesielt viktig. Generelt bruker du alternativet **Vilkårlig** for de fleste kubedimensjonene, og velger bare et bestemt nivå når dette nivået er involvert i intens spørringsaktivitet.

I figur 31 på side 99 ser du et eksempel på en rapportoptimaliseringssektor, og sektorene som optimaliseringsrådgiveren kan anbefale sammendragstabeller for.

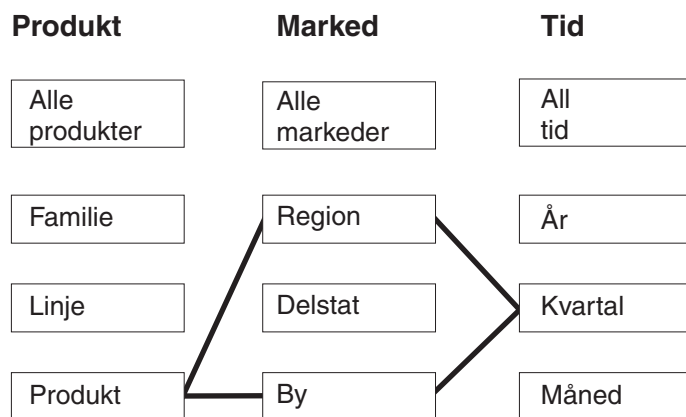
Optimaliseringssektoren er definert på nivået Produkt i kubedimensjonen Produkt, nivået Vilkårlig i kubedimensjonen Marked, og nivået Kvartal i kubedimensjonen Tid. Denne optimaliseringssektoren tyder på at brukere oppretter rapporter som omfatter nivåene Produkt og Kvartal, og enten at det ikke omfatter kubedimensjonen Marked, at det kan inneholde flere nivåer i kubedimensjonen Marked, eller at du ikke vet hvilket nivå i kubedimensjonen Marked brukerne tar med. Det kan være slik at brukere ofte oppretter rapporter som viser salgsdata for hver produktfamilie for de fire siste kvartalene, men rapportene veksler mellom å vise salgsdataene per region, delstat eller by.

De mulige sammendragstabellbefalingene omfatter to samlingsnivåer. Ett samlingsnivået er definert på sektoren Produkt-By-Kvartal, og det andre samlingsnivået er på sektoren Produkt-Region-Kvartal. Begge samlingsnivåene omfatter nivåene Produkt og Kvartal, som er oppgitt i sektoren. På grunnlag av dataavlesing og andre metadata, besluttet optimaliseringsrådgiveren å opprette sammendragstabeller som dekker to samlingsnivåer, ett som inkluderer nivået By i kubedimensjonen Marked, og et annet som inkluderer nivået Region i kubedimensjonen Marked.

Rapportsektor



Mulig anbefaling



Figur 31. Rapport. Optimaliseringssektor for rapport og mulige sammendragstabellanbefalinger

Optimaliseringssektorer for MOLAP-uttrekking

En optimaliseringssektor for MOLAP-uttrekking angir at du ofte trekker ut data på den oppgitte sektoren til en MOLAP-kube i et leverandørprodukt. Derfor bør optimaliseringsrådgiveren anbefale sammendragstabeller som sikrer at spørringer til den oppgitte sektoren, går raskt.

Recommended: Oppgi et spesifikt nivå for hver kubedimensjon, slik at optimaliseringssektoren samsvarer med datanivået du trekker ut i MOLAP-kuben.

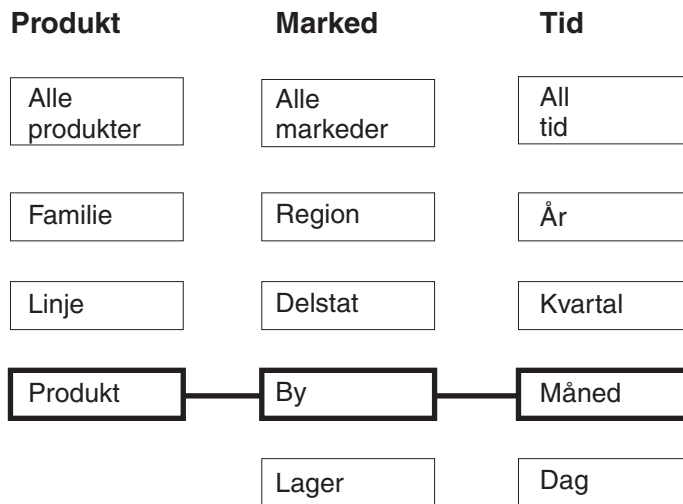
Du kan bare definere en optimaliseringssektor for MOLAP-uttrekking per kube. Du kan ikke definere en optimaliseringssektor for hybriduttrekking i en kube som inneholder en optimaliseringssektor for MOLAP-uttrekking.

I figur 32 på side 100 ser du et eksempel på en optimaliseringssektor for MOLAP-uttrekking, og sektoren som optimaliseringsrådgiveren kan

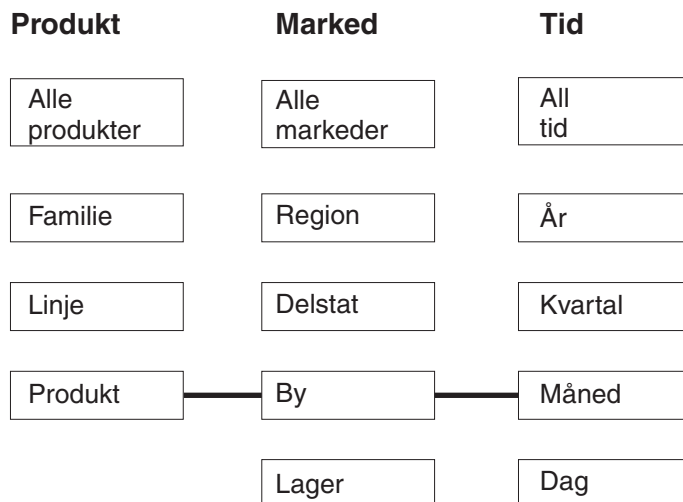
anbefale en sammendragstabell for. Optimaliseringssektoren er definert på nivået Produkt i kubedimensjonen Produkt, nivået By i kubedimensjonen Marked, og nivået Måned i kubedimensjonen Tid. Denne optimaliseringssektoren tyder på at du vil trekke ut dataene på Produkt-By-Måned-nivåene i en MOLAP-kube.

Den mulige anbefalingen omfatter en sammendragstabell som direkte oppfyller MOLAP-uttrekkings spørringen som er oppgitt av optimaliseringssektoren Produkt-By-Måned.

MOLAP-uttrekkingssektor



Mulig anbefaling



Figur 32. MOLAP-uttrekking. Optimaliseringssektor for MOLAP-uttrekking og mulige sammendragstabellanbefalinger

Optimaliseringssektorer for hybriduttrekking

En optimaliseringssektor for hybriduttrekking angir at du ofte trekker ut data på den oppgitte sektoren til en hybrid OLAP-kube (HOLAP-kube) i et

leverandørprodukt. Derfor bør optimaliseringssektoren inkludere den oppgitte sektoren i en anbefalt sammendragstabell.

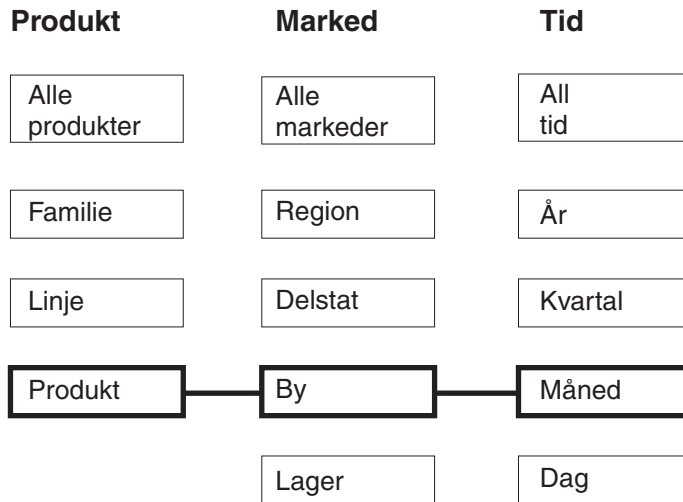
Recommended: Oppgi et spesifikt nivå for hver kubedimensjon, slik at optimaliseringssektoren samsvarer med datanivået du trekker ut i HOLAP-kuben.

Du kan bare definere en optimaliseringssektor for hybriduttrekking per kube. Du kan ikke definere en optimaliseringssektor for MOLAP-uttrekking i en kube som inneholder en optimaliseringssektor for hybriduttrekking. Du kan definere null eller flere optimaliseringssektorer for gjennomdrilling i den samme kuben som inneholder en optimaliseringssektor for hybriduttrekking. Optimaliseringsrådgiveren regner med at du kan ha gjennomdrillingsspøringer nedenfor den oppgitte optimaliseringssektoren for hybriduttrekking, og forsøker å optimalisere for gjennomdrillingsspøringer nedenfor den oppgitte sektoren, samt for hybriduttrekkingsspøringer på den oppgitte sektoren.

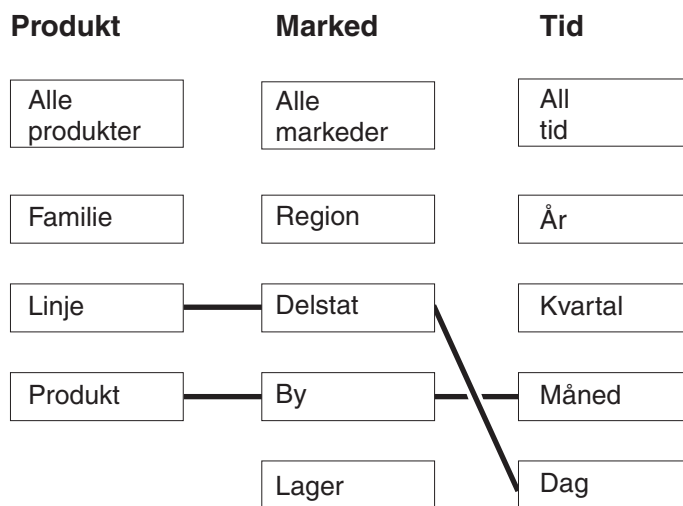
I figur 33 på side 102 ser du et eksempel på en optimaliseringssektor for hybriduttrekking, og sektoren som optimaliseringsrådgiveren kan anbefale en sammendragstabell for. Optimaliseringssektoren er definert på nivået Produkt i kubedimensjonen Produkt, nivået By i kubedimensjonen Marked, og nivået Måned i kubedimensjonen Tid. Denne optimaliseringssektoren tyder på at du vil trekke ut dataene på Produkt-By-Måned-nivåene i en HOLAP-kube.

De mulige sammendragstabellanbefalingene omfatter to samlingsnivåer. Samlingsnivået på sektoren Produkt-By-Måned oppfyller HOLAP-uttrekkingsspøringen som er angitt av optimaliseringssektoren. Samlingsnivået på sektoren Linje-Delstat-Dag inkluderer nivået Dag, som er nedenfor hybriduttrekkingssektoren, for tilfredsstillende mulige gjennomdrillingsspøringer i kubedimensjonen Tid. Optimaliseringsrådgiveren analyserte de andre metadataene og utførte dataavlesing for å utvikle denne anbefalte sammendragstabellen.

Hybriduttrekingssektor



Mulig anbefaling



Figur 33. Hybriduttrekking. Optimaliseringssektor for hybriduttrekking og mulige sammendragstabellanbefalinger

Optimaliseringssektorer for gjennomdrilling

For en optimaliseringssektor for gjennomdrilling må det være definert en optimaliseringssektor for hybriduttrekking i kuben. En optimaliseringssektor for gjennomdrilling angir at du ofte driller ned til den oppgitte sektoren fra en hybrid OLAP-kube (HOLAP-kube) i et leverandørprodukt. Derfor bør optimaliseringsrådgiveren inkludere en sektor på eller under de oppgitte nivåene i en anbefalt sammendragstabell.

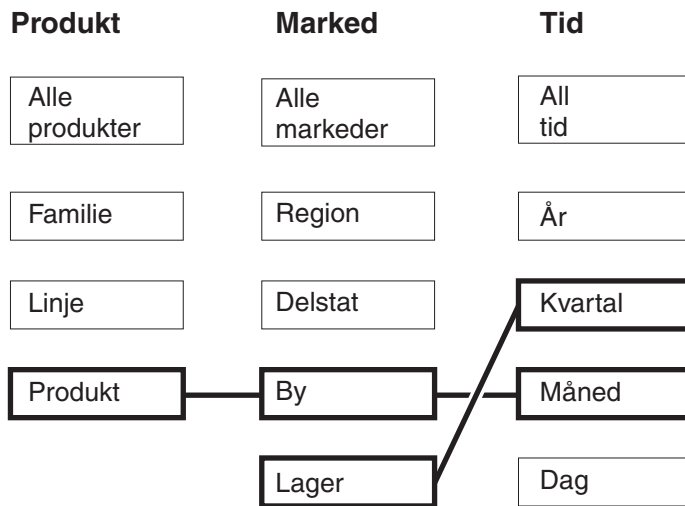
Recommended: Du bør bare oppgi et bestemt nivå i en kubedimensjon for en optimaliseringssektor av typen gjennomdrilling hvis du vet at dette nivået er spesielt viktig. Generelt for optimaliseringssektorer av typen gjennomdrilling bruker du alternativet **Vilkårlig** for de fleste kubedimensjonene,

og velger bare et bestemt nivå når dette nivået er involvert i intens spørringsaktivitet.

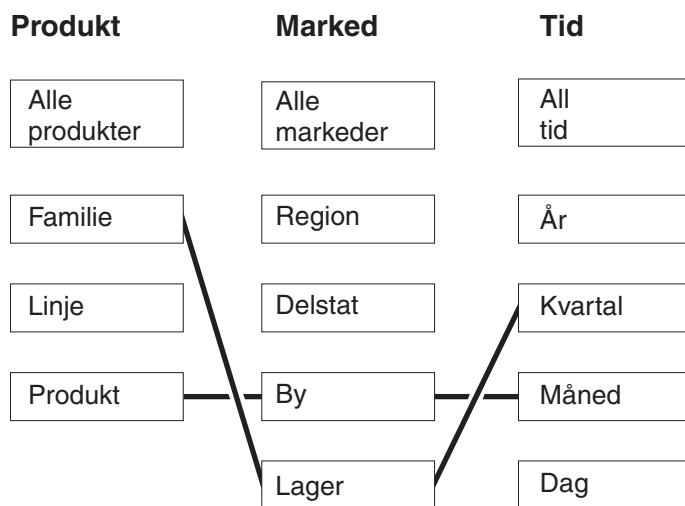
I figur 34 på side 104 ser du et eksempel på en optimaliseringssektor for gjennomdrilling og den tilhørende optimaliseringssektoren for hybriduttrekking, og sektorene som optimaliseringsrådgiveren kan anbefale sammendragstabeller for. Optimaliseringssektoren for hybriduttrekking er definert på nivået Produkt i kubedimensjonen Produkt, nivået By i kubedimensjonen Marked, og nivået Måned i kubedimensjonen Tid. Denne optimaliseringssektoren tyder på at du trekker ut dataene på Produkt-By-Måned-nivåene i en HOLAP-kube. Optimaliseringssektoren for gjennomdrillingen er definert på nivået Vilkårilig i kubedimensjonen Produkt, nivået Lager i kubedimensjonen Marked og nivået Kvartal i kubedimensjonen Tid. Denne optimaliseringssektoren tyder på at gjennomdrillingsspørringer fra HOLAP-kuben vanligvis omfatter nivåene Lager og Kvartal, og at gjennomdrillingsspørringer refererer til, eller ikke refererer til, bestemte nivåer fra kubedimensjonen Produkt.

De mulige sammendragstabellanbefalingene omfatter to samlingsnivåer. Sammendragstabellen på sektoren Produkt-By-Måned oppfyller HOLAP-uttrekkingsspørringen som er angitt av optimaliseringssektoren. Samlingsnivået i sektoren Familie-Lager-Kvartal inkluderer nivået Lager og nivået Kvartal, som er oppgitt av optimaliseringssektoren for gjennomdrilling. Optimaliseringsrådgiveren analyserte de andre metadataene og utførte dataavlesing for å anbefale at denne sammendragstabellen også inkluderer nivået Familie i kubedimensjonen Produkt.

Hybriduttrekking med sektor for gjennomdrilling



Mulig anbefaling



Figur 34. Gjennomdrilling. Optimaliseringssektor for hybriduttrekking, optimaliseringssektor for gjennomdrilling og mulige sammendragstabellanbefalinger

Analysere spørringer for mulige optimaliseringssektorer

Optimaliseringssektorer er et kraftig verktøy for å øke spørringsytelsen, men de er bare effektive hvis de presist gjenspeiler SQL-spørringene som blir sendt.

Når du analyserer brukerens spørremønstre, bør du se på følgende:

- Hvilken type spørringer brukerne sannsynligvis vil sende
- Hvilke hierarknivåer som ofte blir brukt av spørringer
- Hvilket produkt brukerne sender spørringene fra

For å bestemme hvilke områder av kubene du bør opprette optimaliseringssektorer for, kan du se gjennom historikken for brukernes spørringer. Det kan også være en

god ide å spørre brukerne om de forventer at spørringsbehovene vil bli endret i fremtiden. Du ser etter de områdene det er viktigst å optimalisere for.

Anbefaling: Ikke angi flere enn tre optimaliseringssektorer for en kube.

Eksempler på optimaliseringssektorer du kan definere for bestemte situasjoner

Hvert av scenariene er basert på kubene Daily sales i CVSAMPLE-databasen, som har tre kubedimensjoner, Produkt, Marked og Tid, hver med disse kubehierarkiene:

Tabell 35. CVSAMPLE. Kubedimensjoner og tilhørende kubehierarkier for kubene Daily Sales i CVSAMPLE-databasen.

Produkt-kubedimensjon	Marked-kubedimensjon	Tid-kubedimensjon
Alle produkter	Alle markeder	All tid
Familie	Region	År
Linje	Delstat	Kvartal
Produkt	By	Måned
	Postnummer	Dag
	Lager	

Situasjonene som er beskrevet i tabellen nedenfor, er eksempler på hvilke optimaliseringssektorer du kan definere for bestemte spørringsbelastninger, basert på CVSAMPLE-databasen.

Tabell 36. Eksempel på optimaliseringssektorer

Situasjon	Scenario	Optimaliseringssektorer som skal defineres
Brukerne bruker et spørringsprodukt som utsteder en bestemt spørringstype	Du vet at brukerne først og fremst bruker et produkt som utsteder neddrillingsspørringer. Du har ingen informasjon om hvilke deler av kubene som oftest blir brukt i spørringer.	Siden du ikke har noen informasjon om hvilke områder av kubene som er viktigst for brukerne, vil ikke en spesifikk optimaliseringssektor gi deg store fordeler. I denne generelle situasjonen kan du definere hele kubene for neddrillingsspørringer når du oppretter kubene i kubeveiviseren, eller ved å endre kubeegenskapene etter at du har opprettet kubene.

Tabell 36. Eksempel på optimaliseringssektorer (fortsettelse)

Situasjon	Scenario	Optimaliseringssektorer som skal defineres
De fleste spørringer er av en type og fokusert på en nivågruppering.	Du vet at brukerne først og fremst utsteder rapportspørringer, og at nesten alle spørringene refererer til nivået Status til dimensjonen Marked.	Du har spesifikk informasjon om et spesielt viktig område av kuben, så en optimaliseringssektor gir store fordeler. Du kan definere optimaliseringssektoren Vilkårlik-Delstat-Vilkårlik, som er en rapporttype. Denne sektoren refererer til nivået Vilkårlik i kubedimensjonen Produkt, nivået Delstat i kubedimensjonen Marked, og nivået Vilkårlik i kubedimensjonen Tid.
De fleste spørringer er av en type og fokusert på noen få nivågrupperinger.	Du vet at brukerne først og fremst utsteder spørringer. Omtrent halvparten av spørringene refererer til nivået Delstat i kubedimensjonen Marked, og den andre halvparten av spørringene er tilfeldig fordelt på de andre nivåene i kubedimensjonen.	Du kan definere de to følgende optimaliseringssektorene for å gjenspeile denne spørringsfordelingen: <ul style="list-style-type: none"> • Definer en rapportoptimaliseringssektor med nivåene Vilkårlik-Delstat-Vilkårlik for å gjenspeile spørringene som refererer til nivået Delstat. Denne sektoren refererer til nivået Vilkårlik i kubedimensjonen Produkt, nivået Status i kubedimensjonen Marked og nivået Vilkårlik i kubedimensjonen Tid. • Definer en rapportoptimaliseringssektor med nivåene Vilkårlik-Vilkårlik-Vilkårlik for de andre spørringene som kan referere til hvilke som helst nivåer i hvilke som helst av kubedimensjonene. Ved å oppgi denne andre sektoren angir du at det er et betydelig antall spørringer som går inn på andre nivåer i dimensjonen Marked.

Tabell 36. Eksempel på optimaliseringssektorer (fortsettelse)

Situasjon	Scenario	Optimaliseringssektorer som skal defineres
De fleste spørringer er av en type og fokusert på mange nivågrupperinger	Du vet at brukerne først og fremst utsteder spørringer. Spørringsaktiviteten er ikke tilfeldig, den er fordelt på omtrent 15 områder av kubene.	En optimaliseringssektor skal oppgi områder med høy spørringsaktivitet, og i denne situasjonen er spørringene for spredt til at man kan oppgi et bestemt sett med optimaliseringssektorer. I stedet for å oppgi optimaliseringssektorer kan du definere hele kubene for rapportspørringer når du oppretter kubene i kubeveiviseren, eller ved å endre kubeegenskapene etter at du har opprettet kubene.

Begrensningsdefinisjoner for optimalisering

Begrensninger gir verdifull informasjon om optimaliseringsrådgiveren og DB2-optimalisatoren. Du må definere informasjonsbegrensninger eller tvungne begrensninger for fremmednøkler og primærnøkler i stjerneskjemaet eller snøfnuggskjemaet.

Du må definere begrensninger for basistabellene før du kan bruke optimaliseringsrådgiveren. Begrensningene må støtte de grunnleggende reglene, fullstendighetsreglene for kubemodeller og optimaliseringsreglene som er beskrevet i "Regler for metadataobjekter" på side 38, slik at kubemodellen er gyldig for optimalisering. Reglene definerer først og fremst hvordan du kan kombinere metadataobjektene i en kubemodell.

Du kan bruke informasjonsbegrensninger for fremmednøkkelbegrensningene du trenger å definere. Informasjonsbegrensninger er en ny type begrensninger i DB2 Universal Database, versjon 8. Informasjonsbegrensninger er en ny metode for å forbedre ytelsen i spørringer uten å øke vedlikeholdskostnadene. Disse begrensningene kan brukes av SQL-kompilatoren i DB2, men de blir ikke påtvunget av databasesystemet. Denne typen begrensning gjør at DB2 UDB kjenner til forholdene i dataene uten at forholdet blir påtvunget. For primærnøkkelbegrensninger må du bruke de databasetvungne begrensningene i DB2 UDB.

Hver kombinerings må ha definert en tilhørende begrensning. Kolonner som deltar i kombinerings av fakta og dimensjon, og eventuelt i kombinerings av dimensjon og dimensjon, som brukes i et snøfnuggskjema, trenger for eksempel begrensninger.

Hvis du skal optimalisere en kubemodell basert på snøfnuggskjemaet som er vist i figur 35 på side 108, må du definere begrensninger for hver kombinerings av fakta og dimensjon. Dette er de tre kombineringsene av fakta og dimensjon:

- mellom Lager.LagerID og Salg.LagerID
- mellom Tid.TidID og Salg.TidID
- mellom Produkt.ProduktID og Salg.ProduktID

Flere regler gjelder for hver av disse kombineringsene. Du kan bruke informasjonsbegrensninger bare for fremmednøkkelbegrensninger.



Figur 35. Snøfnuggskjema. Relasjonstabeller bruker snøfnuggskjemaet fra CVSAMPLE-databasen.

For kombineringsen mellom Lager- og Salg-tabellene, må du definere disse begrensningene:

- LagerID er primærnøkkelen i tabellen Lager.
- Lager.LagerID og Salg.LagerID er kolonner som ikke kan inneholde nullverdier.
- Salg.LagerID er en fremmednøkkel som refererer til Lager.LagerID. Fremmednøkkelbegrensninger kan defineres som informasjonsbegrensninger.
- Hvis Salg.LagerID ikke er primærnøkkelen for tabellen Salg, er kombineringskardinaliteten 1:mange (Lager.LagerID : Salg.LagerID). Hvis Salg.LagerID er primærnøkkelen for tabellen Salg, er kombineringskardinaliteten 1:1.
- Kombineringstypen er INNER JOIN.

For kombineringsen mellom Tid- og Salg-tabellen, må du definere disse begrensningene:

- TidID er primærnøkkelen i tabellen Tid.
- Tid.TidID og Salg.TidID er kolonner som ikke kan inneholde nullverdier.
- Salg.TidID er en fremmednøkkel som refererer til Tid.TidID. Fremmednøkkelbegrensninger kan defineres som informasjonsbegrensninger.
- Hvis Salg.TidID ikke er primærnøkkelen for tabellen Salg, er kombineringskardinaliteten 1:mange (Tid.TidID : Salg.TidID). Hvis Salg.TidID er primærnøkkelen for tabellen Salg, er kombineringskardinaliteten 1:1.
- Kombinererstypen er INNER JOIN.

For kombineringen mellom Produkt- og Salg-tabellen må du definere disse begrensningene:

- ProduktID er primærnøkkelen i tabellen Produkt.
- Produkt.ProduktID og Salg.ProduktID er kolonner som ikke kan inneholde nullverdier.
- Salg.ProduktID er en fremmednøkkel som refererer til Produkt.ProduktID. Fremmednøkkelbegrensninger kan defineres som informasjonsbegrensninger.
- Hvis Salg.ProduktID ikke er primærnøkkelen for tabellen Salg, er kombineringskardinaliteten 1:mange (Produkt.ProduktID : Salg.ProduktID). Hvis Salg.ProduktID er primærnøkkelen for tabellen Salg, er kombineringskardinaliteten 1:1.
- Kombinererstypen er INNER JOIN.

I et snøfnuggskjema har hver dimensjon en primær dimensjonstabell som en eller flere andre dimensjoner kan kombineres med. Den primære dimensjonstabellen er den eneste tabellen som kan kombineres med faktatabellen. Hver av utriggertabellene som kombineres direkte med den primære tabellen, må ha kombineringskardinaliteten mange:1 (der mange er på den primære tabellens side) eller 1:1. Den primære dimensjonstabellen har vanligvis det mest detaljerte informasjonsnivået av dimensjonstabellene på grunn av disse reglene for kombineringskardinalitet. Hvis et sett med dimensjonstabeller bare bruker kombineringskardinaliteten 1:1, har alle tabellene samme detaljnivå.

Denne kubemodellen er basert på et snøfnuggskjema, så du må definere ytterligere begrensninger for kombineringsdimensjonstabellene. De tre kombineringsdimensjonstabellene mellom dimensjoner er:

- Mellom tabellen Lager og tabellen Plassering
- Mellom tabellen Produkt og tabellen Linje
- Mellom tabellen Linje og tabellen Familie

Flere regler gjelder for hver av disse kombineringsdimensjonstabellene. Du kan bruke informasjonsbegrensninger bare for fremmednøkkelbegrensninger.

For kombineringen mellom Lager- og Plassering-tabellene, må du definere disse begrensningene:

- PostnummerID er primærnøkkelen i tabellen Plassering.
- Plassering.PostnummerID og Lager.PostnummerID er kolonner som ikke kan inneholde nullverdier.
- Lager.PostnummerID er en fremmednøkkel som refererer til Plassering.PostnummerID. Fremmednøkkelbegrensninger kan defineres som informasjonsbegrensninger.

- Kombineringskardinaliteten er 1:mange (Plassering.PostnummerID : Lager.PostnummerID) fordi Lager.PostnummerID ikke er primærnøkkelen eller den entydige nøkkelen for tabellen Lager.
- Kombinererstypen er INNER JOIN.

For kombineringen mellom Produkt- og Linje-tabellene må du definere disse begrensningene:

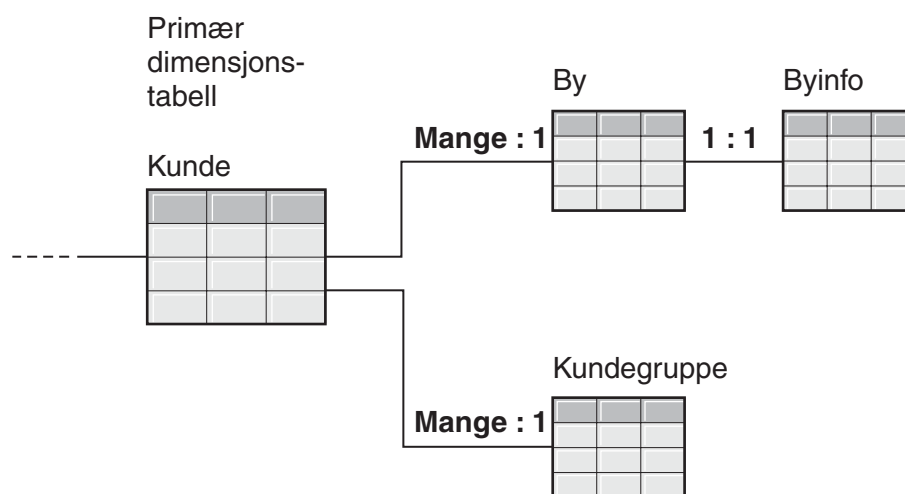
- LinjeID er primærnøkkelen i tabellen Linje.
- Linje.LinjeID og Produkt.LinjeID er kolonner som ikke kan inneholde nullverdier.
- Produkt.LinjeID er en fremmednøkkel som refererer til Linje.LinjeID. Fremmednøkkelbegrensninger kan defineres som informasjonsbegrensninger.
- Koblingskardinaliteten er 1:mange (Linje.LinjeID : Produkt.LinjeID), fordi Produkt.LinjeID ikke er primærnøkkelen eller den entydige nøkkelen for tabellen Produkt.
- Kombinererstypen er INNER JOIN.

For kombineringen mellom Linje- og Familie-tabellene, må du definere disse begrensningene:

- FamilieID er primærnøkkelen i tabellen Familie.
- Familie.FamilieID og Linje.FamilieID er kolonner som kan inneholde nullverdier.
- Linje.FamilieID er en fremmednøkkel som refererer til Familie.FamilieID. Fremmednøkkelbegrensninger kan defineres som informasjonsbegrensninger.
- Kombineringskardinaliteten er 1:mange (Familie.FamilieID : Linje.FamilieID), fordi Linje.FamilieID ikke er primærnøkkelen eller den entydige nøkkelen for tabellen Linje.
- Kombinererstypen er INNER JOIN.

I figur 36 på side 111 er det vist et gyldig sett med dimensjonstabeller i en snøfnuggskjemadimensjon. Den primære dimensjonstabellen er tabellen Kunde med tre ekstra utriggertabeller, By og KundeGruppe kombinert direkte med Kunde og ByInfo kombinert med By. Kombineringskardinalitetene er semantisk gyldige fordi det kan være mange kunder i en by eller kundegruppe og ett sett med byinformasjon finnes per by. Dette er en gyldig dimensjon for optimalisering fordi den følger reglene for optimaliseringsvalidering. Dimensjonen har bare en primær tabell, og tabellene By og KundeGruppe som er kombinert direkte med den primære tabellen, er kombinert med kardinaliteten mange:1. Tabellen ByInfo er kombinert med kardinaliteten 1:1, som også er gyldig. Tabellen Kunde har det mest detaljerte informasjonsnivået av de fire dimensjonstabellene.

Gyldig dimensjon for optimalisering

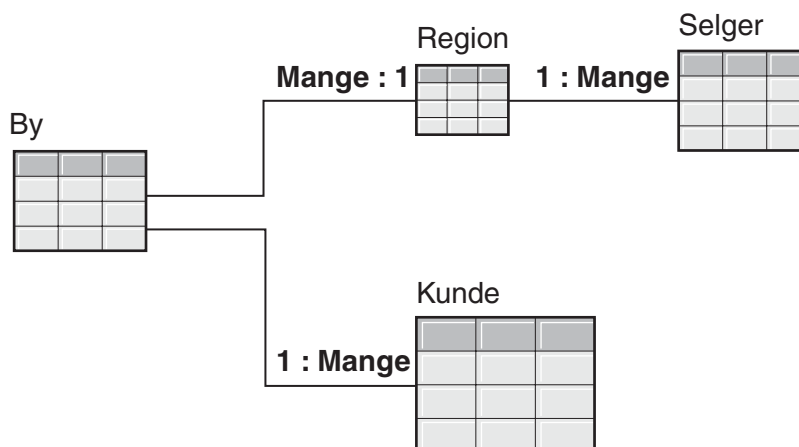


Figur 36. Gyldig dimensjon. Et sett med dimensjonstabeller brukt i en dimensjon som kan optimaliseres

I figur 37 på side 112 er det vist et ugyldig sett med dimensjonstabeller i en snøfnuggskjemadimensjon. På grunn av kardinalitetsforholdene som er definert, kan ingen av disse tabellene være den primære dimensjonstabellen i en kubemodell som skal optimaliseres. Selv om kardinalitetene er semantisk gyldige, er situasjonen den at hvis noen av disse tabellene ble kombinert med faktatabellen som den primære dimensjonstabellen, ville dataene i faktatabellen forøkes, noe som fører til en såkalt "fan trap".

Hvis for eksempel Kunde er den primære dimensjonstabellen, blir dimensjonen ugyldig for optimalisering på grunn av kombineringskardinaliteten 1:mange mellom Region og Selger. Hvis hver region har fem selgere, blir det fem poster for hver region når tabellene Selger og Region blir kombinert. Når disse tabellene blir kombinert med tabellene By og Kunde, og endelig med faktatabellen, blir det tilføyd fem ekstra rader for hver eksisterende rad i By, Kunde og faktatabellen. Fordi samme faktarad er gjentatt fem ganger, blir målet beregnet feil. De andre tabellene i dimensjonen har samme problem. Tabellen By kan ikke være den primære dimensjonstabellen på grunn av kombineringskardinaliteten 1:mange mellom By og Kunde og mellom Region og Selger. Tabellen Region kan ikke kombineres med faktatabellen fordi hver kombineringskardinalitet i dimensjonen er 1:mange-kombinering med tabellen Region. Tabellen Selger kan ikke være den primære dimensjonstabellen på grunn av 1:mange-kombineringen mellom tabellene Region og By og tabellene By og Kunde.

Ugyldig dimensjon for optimalisering



Figur 37. Ugyldig dimensjon. Et sett med dimensjonstabeller brukt i en dimensjon som ikke kan optimaliseres

Parametere for optimaliseringsrådgiveren

Informasjonen du oppgir for hver parameter i veiviseren for optimaliseringsrådgiveren, påvirker hvilke sammendragstabeller veiviseren anbefaler, og hvilke forbedringer du oppnår. Pass på å oppgi nøyaktig informasjon og vurder grundig kravene til kostnader og ytelse.

Forventet spørringsaktivitet

Du oppgir spørringstypene du vil optimalisere hver kube i modellen for. Spørringstypen beskriver når og hvordan det normalt blir gitt tilgang til DB2-relasjonsdata. Denne informasjonen hjelper optimaliseringsrådgiveren å forstå hvilke deler av kubemodellen det oftest utføres spørringer for. Du kan også oppgi optimaliseringssektorer for en kube hvis du vet at det oftest blir utført spørringer for noen få, bestemte sektorer.

Du definerer spørringstyper for hver kube når du oppretter kuben. Du kan bruke optimaliseringsrådgiveren til å vurdere det du oppgav for hver kube, og gjøre eventuelle nødvendige endringer.

Lagerplassbegrensninger

Du oppgir en mengde lagerplass som kan brukes til sammendragstabeller. Optimaliseringsrådgiveren kjenner ikke den nøyaktige størrelsen på sammendragstabellene før de er bygd, men anbefaler sammendragstabeller med en størrelse som ligger så nær som mulig den mengden lagerplass du har oppgitt. Sammendragstabellene som blir bygd, kan bruke mer eller mindre lagerplass enn det du har oppgitt.

Mengden lagerplass som du oppgir, er direkte knyttet til optimaliseringsresultatene. Hvis du øker lagerplassen, kan både antall spørringer med forbedret ytelse og graden av forbedring øke. Du bør vurdere disse faktorene før du velger størrelsen på lagerplassen:

- Hvilke nivåer av ytelse du ønsker for spørringene
- Antall kubemodeller du utfører optimaliseringen for
- Hvor kritisk hver kubemodell er
- Hvor ofte hver kubemodell blir brukt
- Tilgjengeligheten og kostnadene for lagerplassen

Vanligvis vil du se signifikant forbedring ved å tillate en middels mengde lagerplass, for eksempel 1 % til 10 % av plassen som brukes av relasjonstabellene kubemodellen refererer til. tabell 37 viser forholdet mellom mengden lagerplass som brukes til sammendragstabellene og den forventede ytelsesforbedringen for spørringer. Når du har avgjort hvor mye plass du vil bruke, må du vurdere hver kubemodell i konteksten av alle metadataene og basistabellene.

Tabell 37. Lagerplass. Prosentdel av lagerplass brukt og tilsvarende forventede ytelsesforbedringer

Prosentdel av lagerplass for basistabeller brukt til sammendragstabeller	Forventet forbedring for relevante spørringer
Mindre enn 1 %	Lav
5 %	Middels
50 %	Høy
Ubegrenset	Høyeste

Tidsbegrensning

Tiden du oppgir, er den maksimale tiden optimaliseringsrådgiveren kan bruke til å opprette anbefalingene. Jo mer tid du tillater at optimaliseringsrådgiveren bruker på kjøringen, desto bedre blir resultatet. Tabellen nedenfor gir enkelte retningslinjer for hvor mye tid du bør gi optimaliseringsrådgiveren. Ytelsesresultatene vil variere, og det er mulig du må tillate mer tid enn det som er oppgitt i tabell 38.

Tabell 38. Tidsbegrensning. Retningslinjer for hvor mye tid du bør gi optimaliseringsrådgiveren

Databaseoptimaliseringsscenario	Omtrentlig tidsgrense
Utfører ikke dataavlesing	5 til 30 minutter
Utfører dataavlesing for en liten database på under 10 gigabyte	1 time eller mindre
Utfører dataavlesing for en stor database på over 10 gigabyte	Flere timer

Dataavlesing

Dataavlesing er en metode optimaliseringsrådgiveren bruker for å undersøke dataene i kubemodellen. Det gir optimaliseringsrådgiveren mer informasjon slik at den kan opprette best mulig anbefalinger. Anbefalinger som opprettes med dataavlesing, vil svare mer nøyaktig til lagerplassen som er oppgitt. Uten dataavlesing vil optimaliseringsrådgiveren bare analysere metadataene og DB2-statistikken når den oppretter anbefalinger.

Optimalisere en kubemodell

Ved å optimalisere for spørringer som utføres på en kubemodell, kan du forbedre ytelsen for produkter som utsteder SQL-spørringer i OLAP-stil.

Krav: Du må ha oppgitt DB2-begrensninger for basistabellene som brukes i kubemodellen. Det må være oppgitt begrensninger mellom hver faktatabell og dimensjonstabell og mellom hver dimensjonstabell i et snøfnuggskjema. Det må være oppgitt begrensninger for kolonner som ikke kan ha nullverdier. Du finner flere opplysninger om hvordan du definerer begrensninger, i "Begrensningsdefinisjoner for optimalisering" på side 107.

Når du optimaliserer en kubemodell, oppretter veiviseren for optimaliseringsrådgiveren SQL-setninger som kan bygge et sett med anbefalte sammendragstabeller for en kubemodell. Sammendragstabeller samler data som det ofte blir gitt tilgang til, for å øke spørreytelsen.

Slik optimaliserer du en kubemodell:

1. Åpne optimaliseringsrådgiveren ved å høyreklikke på en kubemodell i objektoversikten i OLAP-senteret og klikke på **Optimaliseringsrådgiver**.
2. På siden Spørringstyper oppgir du de spørringstypene du vil optimalisere hver kube for. Du kan endre spørringstypen eller oppgi optimaliseringssektorer for kubene. Kubens spørringstype brukes til å forbedre optimaliseringsresultatene. Du finner flere opplysninger om optimaliseringssektorer under "Optimaliseringssektorer for kuber" på side 95.
3. På siden Sammendragstabeller oppgir du om du ønsker umiddelbar eller utsatt oppdatering av sammendragstabellene. Du finner flere opplysninger om oppdateringsalternativene under "Vedlikehold av sammendragstabeller" på side 121. Oppgi hvilken tabellplass du vil lagre sammendragstabellene og sammendragstabellindeksene i.
4. På siden Begrensninger oppgir du hvor mye lagerplass du vil tillate for sammendragstabellene og indeksene som skal bygges. Oppgi om du vil tillate dataavlesing. Oppgi også den maksimale tiden du vil tillate at optimaliseringsrådgiveren bruker til å bestemme anbefalingene. Jo mer plass, informasjon og tid du oppgir, desto mer vil ytelsesresultatene bli forbedret. Du finner informasjon om hvordan du oppgir parameterne for veiviseren for optimaliseringsrådgiveren, under "Parametere for optimaliseringsrådgiveren" på side 112.

Dette er den siste siden med inndataparametere i veiviseren for optimaliseringsrådgiveren. Klikk på **Neste** for å åpne et statusvindu der du kan overvåke (og om nødvendig stoppe) optimaliseringsrådgiverens arbeid med å bestemme anbefalinger for å opprette og oppdatere sammendragstabellene. Du kan endre tidsbegrensningen for optimaliseringsrådgiveren. Hvis du klikker på **Stopp**, returnerer optimaliseringsovervåkeren sammendragstabellanbefalingene den har bestemt hittil.

5. På siden SQL-skript skriver du et entydig filnavn i feltet **SQL-skript for oppretting av sammendragstabellene**, og et entydig filnavn i feltet **SQL-skript for fornying av sammendragstabellene**.
6. Klikk på **Fullfør** for å lagre de anbefalte SQL-skriptene med filnavnene du har oppgitt.
7. Utfør SQL-skriptene. Hvis du oppretter store sammendragstabeller, kan det ta lang tid å fullføre byggingen av sammendragstabellene. Du kan bruke DB2 Kommandosenter eller kommandovinduet når du skal kjøre SQL-skriptene. Slik kjører du SQL-skriptene fra vinduet DB2-kommando:

- a. Bytt til katalogen der du lagret SQL-skriptene.
- b. Koble til databasen for kubemodellen du optimaliserte. Oppgi for eksempel `db2 connect to CVSAMPLE`.
- c. Oppgi denne kommandoen:
`db2 -tvf filnavn`

der *filnavn* er navnet på SQL-skriptet for å opprette sammendragstabell.

Eksempel på et SQL-skript for oppretting av sammendragstabellene

Veiviseren Optimaliseringsrådgiver har et SQL-skript for å opprette de anbefalte sammendragstabellene. SQL-skriptet inneholder de nødvendige SQL-kommandoene for å bygge en eller flere sammendragstabeller.

I figur 38 på side 116 er det vist en del av et SQL-eksempelskript som oppretter en sammendragstabell, og hvordan metadataobjekter tilordnes til SQL-koden. I SQL-eksempelskriptet er sammendragstabellen kalt `DB2INFO.MQT0000000021T01`, der 21 er kubemodell-IDen og T01 er sammendragstabell-IDen. Kubemodell-IDen kan inneholde opptil 10 sifre. Sammendragstabell-IDen identifiserer sammendragstabellen i kubemodellen. Sammendragstabell-IDen tillater opptil 99 sammendragstabeller i en kubemodell. Du må ikke endre navnet på sammendragstabellen som veiviseren for optimaliseringsrådgiveren definerer. Hvis du endrer tabellnavnet, kan ikke DB2 Cube Views identifisere sammendragstabellene som blir opprettet for kubemodellen.

```

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;

```

Tilordner til mål

Tilordner til attributter

Fakta- og dimensjonstabeller

Kombinerer fakta- og dimensjonstabeller.
Kombinerer dimensjon til dimensjonstabeller i et snøfnuggskjema der det er aktuelt

Tilordner til en kubemodellsektor

Oppretter indekser

Figur 38. Eksempel på SQL-skript

Hvis det blir anbefalt mer enn en sammendragstabell for kubemodellen, vil SQL-skriptet for oppretting av sammendragstabeller inneholde et sett med disse setningene for hver sammendragstabell.

Avsnittene nedenfor beskriver setningene i eksempelet på SQL-skript for oppretting av sammendragstabeller:

DROP TABLE-setning

For hver sammendragstabell som blir opprettet, blir denne tabellen først slettet slik at det ikke vil finnes noen tabell med dette navnet. I figur 38, blir tabellen DB2INFO.MQT0000000021T01 slettet med denne setningen: DROP TABLE DB2INFO.MQT0000000021T01;

CREATE TABLE-setning

Skriptet oppretter sammendragstabellen med en CREATE TABLE-setning. Denne setningen utgjør den største delen av skriptet og omfatter SELECT-setningen med leddene SELECT, FROM, WHERE og GROUP BY, og definisjonen av oppdateringsmetoden. Sammendragstabellen blir opprettet med riktige kolonner,

men uten data. SQL-skriptet som fornyer sammendragstabellene, samler dataene fra fakta- og dimensjonstabellene, og fyller sammendragstabellen med data.

Tabellnavnet er oppgitt på første linje i CREATE TABLE-setningen: CREATE SUMMARY TABLE DB2INFO.MQT0000000021T01.

SELECT-leddet som er vist i figur 38 på side 116 har fem linjer som begynner med SUM. Hver av disse linjene er tilordnet en av kubemodellens mål. SUM(T1."SALES"-(T1."COGS"+T1." ADVERTISING')) AS "Profit" er for eksempel tilordnet til det beregnede Profit med samlingsfunksjonen SUM. Kubemodellen som sammendragstabellen blir opprettet for, har disse målene: Sales, COGS, Advertising, Total expense, Profit. De neste sju linjene som velger en kolonne uten å utføre noen beregninger, er tilordnet attributter. T5."REGION_NAME" AS "REGION_NAME" tilorder for eksempel til attributtet Region name. Sammendragstabellen har disse attributtene fra kubemodellen: regionnavn, delstatsnavn, linje-ID, år, kvartalnummer, kvartalnavn og månedsnavn.

Tabellene i FROM-leddet er fakta- og dimensjonstabellene som brukes i kubemodellen. Dette eksempelet bruker tabellene Salgsfakta, Lager, Produkt, Tid, Plassering og Linje.

WHERE-leddet definerer kombineringsene mellom fakta- og dimensjonstabellene, og hver kombineringsobjekt er tilordnet et kombineringsobjekt i kubemodellen. Kubemodellen som optimaliseres er basert på et snøfnuggskjema, så kobmineringene av dimensjon og dimensjon er også tatt med i WHERE-leddet.

GROUP BY-leddet er tilordnet sektorer som er definert for kubemodellen. figur 38 på side 116 viser et grupperingssett som er tilordnet en bestemt sektor. Grupperingene kan omfatte følgende typer av metadata som definerer sektoren:

- Nivånøkkelattributter fra hierarkiet på sektornivå
- Nivånøkkelattributter som er over sektornivået
- Beslektede attributter som ikke er funksjonelt avhengige av et nivå-nøkkelattributt

Denne delen av SQL-skriptet kan inneholde GROUPING SETS, slik at sammendragstabellen kan inneholde flere nivåer. Hvis en kubemodell inneholder ikke-distributive mål, kan denne delen av SQL-skriptet inneholde en ROLLUP.

I det viste eksempelet har kubemodellen som blir optimalisert, disse hierarkiene: Market [Region, State, City, Postal code, Store], Product [Family, Line, Product], Time [Year, Quarter, Month, Day] og Fiscal Time [Fiscal year, Fiscal quarter, Fiscal month]. Hvis et nivå i hierarkiet ikke er tatt med i grupperingssettet, finnes sektoren på det høyeste nivået, for eksempel All Time, All Regions eller All Products. Sektoren i GROUP BY-leddet er sektoren Delstat-Linje-Måned, og inkluderer attributtene regionnavn, år, kvartalnummer og kvartalnavn. Regionnavn er over nivået delstat, og år, kvartalnummer og kvartalnavn er attributter som er over månedsnivået. Hvert av nivåattributtene i sektoren er attributter som SELECT-attributtet er tilordnet.

Den siste delen av CREATE TABLE-setningen er definisjonen av oppdateringsmetoden. I figur 38 på side 116 definerer de siste tre linjene i CREATE TABLE-setningen sammendragstabellen med utsatt oppdatering:

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

Hvis du definerer en sammendragstabell med umiddelbar oppdatering, blir setningene slik:

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

CREATE INDEX-setninger

Veiviseren for optimaliseringsrådgiveren anbefaler en eller flere indekser for sammendragstabellen, og de blir opprettet etter at sammendragstabellen er opprettet. I figur 38 på side 116 blir det opprettet både grupperingsindekser og ikke-grupperingsindekser. Når indeksene er opprettet, omorganiserer REORG-setningen tabellene i samsvar med grupperingsindeksen. I noen tilfeller kan dette forbedre leseytelsen for tabellen.

RUNSTATS-setning

Når alle de anbefalte aspektene av sammendragstabellen er opprettet, oppdaterer RUNSTATS-setningen DB2-optimalisatorstatistikken som DB2-optimalisatoren bruker ved vurdering av sammendragstabellene og indeksene for omdirigering av spørringer.

Teste resultater fra spørring

Ved hjelp av verktøyet db2batch for ytelsestesting (benchmark) i DB2 Universal Database kan du teste ytelsen til spørreresultater før og etter at du har opprettet sammendragstabellene med optimaliseringsrådgiveren.

Slik tester du ytelsen til spørringer:

1. Opprett en inndatafil med spørringene du vil teste, atskilt med semikolon.
2. Oppgi denne kommandoen på en kommandolinje:

```
db2batch -d dbnavn -f
filnavn -cli
```

der *dbnavn* er databasen du vil kjøre spørringene mot, *filnavn* er inndatafilen med SQL-spørringene og *-cli* oppgir at spørringene skal kjøres i CLI-modus. Verktøyet db2batch oppsummerer ytelsesresultatene og viser både aritmetisk og geometrisk gjennomsnitt. Hvis du vil se syntaks og parametere, oppgi du db2batch -h på en kommandolinje. DB2 Informasjonssenter inneholder flere opplysninger om ytelsesverktøyet db2batch og hvordan du oppretter ytelsestester.

Hvis du er fornøyd med ytelsesresultatene etter at du har opprettet de anbefalte sammendragstabellene, trenger du ikke å utføre flere ytelsesanalyser.

Hvis spørringene ikke forbedres slik du ventet det, kan du kjøre veiviseren for optimaliseringsrådgiveren på nytt og tillate mer lagerplass og tid. Du kan også aktivere dataavlesing hvis du ikke har aktivert det før. Å tillate mer lagerplass vil sannsynligvis påvirke ytelsen mest. Jo mer plass du gir til sammendragstabellene,

desto større forbedring vil du få. Hvis du tillater at veiviseren utfører dataavlesing, kan veiviseren gi bedre anbefalinger. Og jo mer tid du tillater at veiviseren bruker på å opprette anbefalinger, desto bedre vil sannsynligvis anbefalingene bli.

Hvis du ikke er fornøyd med ytelsesresultatene fordi spørringene ikke forbedres i det hele tatt eller svært lite, eller hvis spørringene fungerer tilfredsstillende en tid og deretter får dårligere ytelse, kan du lese "Problemløsning for sammendragstabeller".

Problemløsning for sammendragstabeller

Hvis spørringsytelsen ikke blir bedre etter at du har opprettet sammendragstabellen, kan du bruke funksjonen DB2EXPLAIN til å søke etter feil i rutingen av spørringene.

Før du bruker DB2EXPLAIN for å bekrefte at DB2 UDB bruker sammendragstabellene, bør du:

- Kontroller at statistikken er oppdatert for basistabellene og sammendragstabellene.
- Identifiser hvilke spørringer som ikke utføres på akseptabel måte, hvis du ikke allerede vet det. Du kan registrere langsomme spørringer med DB2 SQL Snapshot Monitor.

Slik finner du ut hvorfor spørringene ikke gir den ytelsen du forventer:

1. Opprett forklaringstabellene. Du oppretter forklaringstabeller for databasen ved å tilkoble databasen og kjøre denne kommandoen fra katalogen \SQLLIB\misc:

```
db2 -tvf explain.ddl
```

2. Utfør forklaringsfunksjonen. Når forklaringsmodus er aktivert, blir ikke SQL-spørringene utført, og bare informasjonsforespørsler for explain-kommandoen blir utført. Aktiver forklaringsmodus med følgende serie av SQL-kommandoer, definer fornyingsalderen slik at DB2 UDB vurderer sammendragstabeller om de har utsatt fornying, kjør spørringen, deaktivert forklaringsmodus og kjør en spørring mot forklaringstabellen for å se om spørringen ble omdirigert

```
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. Vis forklaringsinformasjonen og kontroller at den omskrevne spørringen er omdirigert til en sammendragstabell. Du kan for eksempel få en rapport lik dette eksempelet:

```
2002-06-30-23.22.12.325002 0 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
```

Det finnes to linjer for en utføring av spørringen. Linjen merket med 0 er den opprinnelige spørringene som er sendt til DB2 UDB. Linjen merket med P er

spørringen som den er omskrevet av DB2-optimalisatoren. du kan se i den omskrevne spørringen i dette eksempelet at DB2-optimalisatoren valgte data fra sammendragstabellen DB2INFO.MQT0000000021T01.

Hvis spørringen er omdirigert til sammendragstabellen, men ikke gir den ytelsen du forventer, er det mulig at du må kjøre veiviseren for optimaliseringsrådgiveren på nytt med andre parametere.

Hvis spørringen ikke er omdirigert til en sammendragstabell, må du finne årsaken og gjøre nødvendige endringer. Mulige årsaker til at en spørring ikke blir omdirigert til en sammendragstabell:

Sammendragstabellen finnes ikke.

Kontroller først om sammendragstabellen finnes. Hvis den ikke finnes, kjører du veiviseren for optimaliseringsrådgiveren for å generere SQL-skriptet for oppretting av sammendragstabeller. Kjør deretter skriptet for å opprette sammendragstabellene.

Sammendragstabell med utsatt oppdatering er utløpt

Hvis sammendragstabellen finnes og den er definert med utsatt oppdatering, er det mulig du må oppdatere fornyingsalderen. Du kan definere tabellens fornyingsalder så høy som mulig og sesjonsuavhengig ved å oppgi (DFT_REFRESH_AGE) = 99999999999999.

Spørringen trenger tilgang til data som ikke er med i sammendragstabellen.

hvis spørringen trenger tilgang til data som ikke er med i sammendragstabellen, vil ikke DB2-optimalisatoren omdirigere spørringen. Hvis du har tilføyd et nytt mål etter at du opprettet sammendragstabellene, finnes ikke dette nye målet i sammendragstabellene. Hvis du prøver å spørre på det nye målet, kan ikke DB2-optimalisatoren omdirigere spørringen til sammendragstabellen fordi sammendragstabellen ikke inneholder alle dataene som er nødvendige for spørringen.

Hvis du prøver å spørre på data som ligger under den sektoren i kubemodellen som sammendragstabellen er bygd for, kan du heller ikke bruke sammendragstabellen. Hvis spørringen gjelder data som er samlet på nivået By, men sammendragstabellen inneholder data som er samlet på nivået Delstat (som er over nivået By), kan ikke spørringen bruke sammendragstabellen.

spørringen inneholder konstruksjoner som ikke kan omdirigeres

DB2-optimalisatoren kan ikke omdirigere spørringer som bruker enkelte komplekse spørringskonstruksjoner. Enkelte komplekse konstruksjoner som hindrer at DB2-optimalisatoren omdirigerer spørringene, er rekursive funksjoner og funksjoner for fysiske egenskaper, blant annet disse:

- NODENUMBER
- Ekstern kombinerings
- Union
- XMLAGG
- Vindussamlingsfunksjoner som er samlingsfunksjoner som er oppgitt med OVER-leddet

Vedlikehold av sammendragstabeller

Når dataene i basistabellene blir endret, må du oppdatere sammendragstabellene. Du kan oppdatere sammendragstabellene på to måter, med umiddelbar eller utsatt oppdatering.

Du velger å opprette sammendragstabeller med umiddelbar eller utsatt fornying når du kjører veiviseren for optimaliseringsrådgiveren. Valget du gjør, påvirker oppdateringsinnstillingen for tabellene og SQL-skriptet for oppdatering av sammendragstabeller. For begge alternativene må du kjøre skriptet for oppdatering av sammendragstabeller som en del av den normale planen for vedlikeholdet av databasen. Kjøring av oppdateringsskriptet kan ta lang tid og kreve store behandlingsressurser. Pass på å tildele nok tid for satsvist vedlikehold slik at du får fullført oppdateringene.

Forny umiddelbart

Sammendragstabeller med umiddelbar oppdatering holdes tett synkronisert med basistabellene. DB2 UDB sporer endringene i basistabellene og utfører inkrementelle oppdateringer av sammendragstabellene ved å endre bare den delen av sammendragstabellene som samsvarer med den endrede delen av basistabellene. Hvis det er viktig at dataene i sammendragstabellene svarer til dataene i basistabellene, bør du bruke alternativet Oppdater umiddelbart. Oppdater umiddelbart kan for eksempel være et godt valg hvis basistabellene blir oppdatert med ukentlige salgsdata og brukerne lager ukentlige rapporter som skal gjenspeile de oppdaterte salgsdataene.

Hvis du vanligvis har mange endringer spredt rundt i basistabellene, er ikke Oppdater umiddelbart det beste valget, fordi det kan kreve mye behandling i DB2 UDB å spore endringene og utføre oppdateringssetningene enkeltvis for å samle endringene igjen.

Hvis du oppdaterer basistabellene ved hjelp av vanlige SQL-setninger som INSERT, UPDATE og DELETE, synkroniserer DB2 UDB automatisk de påvirkede sammendragstabellene når du har endret basistabellene. Men hvis du oppdaterer basistabellene ved hjelp av DB2-kommandoen LOAD eller IMPORT, må du starte synkroniseringen manuelt ved å kjøre oppdateringsskriptet når du har fullført oppdateringen.

Umiddelbar oppdatering kan ikke brukes i alle situasjoner, og veiviseren for optimaliseringsrådgiveren kan anbefale utsatt oppdatering hvis det er nødvendig.

Fornyings utsatt

Tabeller med utsatt fornying oppdateres vanligvis sjeldnere enn sammendragstabeller med umiddelbar oppdatering, fordi du må synkronisere sammendragstabellene manuelt med basistabellene. Sammendragstabellene er basert på et snapshot av dataene på det tidspunktet de blir opprettet. Hver oppdatering oppretter sammendragstabellen på nytt på grunnlag av de gjeldende dataene, men uten kjennskap til hvordan dataene er endret siden sammendragstabellen sist ble opprettet.

Oppdatering utsatt er et godt valg når du gjør større endringer i de tilsvarende basistabellene, eller hvis du oppdaterer data oftere enn du trenger tilgang til dem. Hvis for eksempel salgsdataene dine blir oppdatert ukentlig, men du bare trenger å opprette rapporter hvert kvartal, kan du

bruke alternativet Oppdatering utsatt og bygge sammendragstabellene på nytt hvert kvartal før du lager rapporten.

Slette en sammendragstabell

DB2 Cube Views sletter ikke de tilknyttede sammendragstabellene når du sletter en kubemodell. Hvis du ikke bruker sammendragstabellene til noe annet, kan du slette tabellene for å frigjøre lagerplass.

Sammendragstabeller er en type tabeller, og de kan slettes på vanlig måte i DB2 fra kontrollsenteret eller kommandolinjen. Eventuelle tilhørende indekser blir slettet sammen med sammendragstabellen.

Sammendragstabellene defineres i DB2INFO-skjemaet. Navnet på sammendragstabellen inneholder kubemodell-IDen. En sammendragstabell kan for eksempel få navnet DB2INFO.MQT000000021T01, der 21 er kubemodell-IDen og T01 identifiserer sammendragstabellen entydig innenfor kubemodellen. Kubemodell-IDen kan inneholde opptil 10 sifre. Når du skal slette en sammendragstabell fra en kommandolinje, oppgir du `DROP TABLE tabellnavn`.

Kapittel 6. DB2 Cube Views og forente datakilder

Denne delen beskriver disse emnene:

Oversikt over forente systemer

Du kan bruke IBM DB2 Information Integrator som bedriftens løsning for informasjonsintegrering. DB2 Information Integrator er en samling med teknologier som kombinerer datastyringssystemer og forente systemer, samt flere andre teknologier i en felles plattform.

Oversikt over optimalisering av fjerndatakilder med DB2 Cube Views

Med DB2 Cube Views kan du optimalisere et forent stjerneskjema eller snøfnuggskjema, og oppnå betydelige forbedringer i spørringsytelse.

Aktivere et forent system for DB2 Cube Views

Når du skal aktivere det forente systemet for DB2 Cube Views, må du definere den fjerntliggende datakilden, definere kallenavn på den forente tjeneren og definere informasjonsbegrensninger for kallenavnene.

Problemløsning for omdirigering av spørringer for forente datakilder

Hvis spørringene til en fjerntliggende datakilde ikke gir den forventede effekten etter optimalisering, kontrollerer du at det forente systemet er riktig konfigurert for DB2 Cube Views.

Oversikt over forente systemer

Du kan bruke IBM DB2 Information Integrator som bedriftens løsning for informasjonsintegrering. DB2 Information Integrator er en samling med teknologier som kombinerer datastyringssystemer og forente systemer, samt flere andre teknologier i en felles plattform.

Et forent DB2-system er en spesiell type distribuert databasesystem (DBMS). Teknologien for integrasjon av informasjon i forente systemer kan brukes til å få tilgang til ulike typer data som er spredt på tvers av ulike datakilder. Et forent system består av disse komponentene:

- En DB2-forekomst som opererer som en forent tjener
- En database som fungerer som den forente databasen
- En eller flere datakilder
- Klienter (bruker og applikasjoner) som bruker databasen og datakildene

I et forent system kan du sende distribuerte forespørsler til flere datakilder innenfor en enkelt SQL-setning. Du kan for eksempel kombinere data som er plassert i en DB2 Universal Database-tabell, og en XML-kodet fil i en enkelt SQL-setning.

Forente tjenere

DB2-tjeneren i et forent system blir kalt forent tjener. Et hvilket som helst antall DB2-forekomster kan konfigureres til å fungere som forente tjenere. Du kan bruke eksisterende DB2-forekomster som forente tjenere, eller du kan opprette nye, spesifikt for det forente systemet.

DB2-forekomsten som styrer det forente systemet, kalles en tjener, fordi den svarer på forespørsler fra sluttbrukere og klientapplikasjoner. Den forente tjeneren sender ofte deler av forespørslene til datakildene for

behandling. En 'pushdown'-operasjon er en operasjon som utføres på en fjernliggende maskin. DB2-forekomsten som administrerer det forente systemet, er referert til som den forente tjeneren, selv om den fungerer som en klient når den skyver forespørsler nedover til datakildene.

Den forente tjeneren virker sammen med datakilder via innpakker. Den forente tjeneren bruker rutiner som er lagret i et bibliotek som kalles en innpakkermodul når en innpakker skal implementeres. Disse rutinene gjør det mulig for den forente tjeneren å utføre operasjoner som å koble til en datakilde og hente data fra den gjentatte ganger. Vanligvis bruker eieren av den forente DB2-forekomsten setningen CREATE WRAPPER til å registrere en innpakker i den forente databasen.

Forente databaser

For sluttbrukere og klientapplikasjoner ser datakilder ut som en enkelt samledatabase i DB2 UDB. Brukere og applikasjoner er koblet mot den forente databasen som styres av den forente tjeneren. Den forente databasen inneholder en systemkatalog. Systemkatalogen til den forente databasen inneholder poster som identifiserer datakilder og deres egenskaper. Den forente tjeneren undersøker informasjonen som er lagret i den forente databasesystemkatalogen og datakildeinnpakkeren for å finne den beste planen for å behandle SQL-setninger.

Datakilder

I et forent system kan en datakilde være en relasjons-DBMS-forekomst (f.eks. DB2, Informix, Oracle eller Sybase) eller en ikke-relasjonsdatakilde (f.eks. BLAST-søkealgoritme eller en XML-kodet fil).

Hvilken metode eller protokoll som brukes for å få tilgang til en datakilde, avhenger av typen datakilde. DRDA brukes for eksempel for å få tilgang til datakilder i DB2-familien som DB2 for z/OS og OS/390.

Klienter

Klienter kan omfatte brukere og applikasjoner som bruker den forente databasen og datakildene du konfigurerer. DB2 Cube Views er et eksempel på en applikasjon som kan fungere som en klient som bruker en forent database og datakilde.

Oversikt over optimalisering av fjerndatakilder med DB2 Cube Views

Med DB2 Cube Views kan du optimalisere et forent stjerneskjema eller snøfnuggskjema, og oppnå betydelige forbedringer i spørringsytelse.

Fordelene omfatter:

- Du har en integrert DB2-plattform der du kan få tilgang til flere IBM- og tredjepartsprodukter.
- Utvidet funksjonalitet for DB2 UDB for Linux, UNIX, og Windows til DB2 UDB for z/OS og DB2 UDB for iSeries
- Økt ytelse for spørringer til forente databaser og datakilder

I det forente systemet kan fakta- og dimensjonstabellene være på en fjernliggende tjener eller på tvers av flere fjernliggende tjenere. Stjerneskjemaet eller snøfnuggskjemaet for DB2 Cube Views må være representert lokalt. Du kan representere stjerneskjemaet eller snøfnuggskjemaet lokalt med kallenavn som refererer til de fjernliggende tabellene, replikerte kopier av de fjernliggende tabellene, eller en kombinasjon av kallenavn og replikerte tabeller. Alle

faktatabellene og dimensjonstabellen i stjerneskjemaet eller snøfnuggskjemaet må være representert lokalt på den lokale tjeneren.

Når stjerneskjemaet eller snøfnuggskjemaet finnes på den lokale tjeneren, kan du bruke DB2 Cube Views til å bygge en kubemodell basert på dette stjerneskjemaet eller snøfnuggskjemaet, og optimalisere denne kubemodellen ved hjelp av optimaliseringsrådgiveren. Hvis du lokalt representerer noen av tabellene med kallenavn, slik at veiviseren for optimaliseringsrådgiveren kan bruke dataavlesing, får du langt bedre sammendragstabellanbefalinger, men det tar lengre tid å opprette anbefalingene. Det er ikke sikkert at optimaliseringsrådgiveren kan bruke dataavlesing i alle situasjoner, men det anbefales at du tillater dataavlesing der det er mulig.

Du kan opprette de anbefalte sammendragstabellene på den lokale tjeneren, og en spørring som er dirigert til tabeller i det fjerntliggende stjerne- eller snøfnuggskjemaet, kan besvares på en av disse tre måtene:

Spørringen rutes til en lokal sammendragstabell

Hvis spørringen kan besvares av dataene i sammendragstabellen, vil DB2-optimalisatoren rute spørringen direkte til den lokale sammendragstabellen, trenger ingen interaksjon med de fjerntliggende tabellene. Dette scenariet gir den største ytelsesforbedringen.

Ofte er det slik at en spørring ikke kan besvares av sammendragstabellen alene, men den kan besvares hvis du kombinerer sammendragstabellen med en eller flere dimensjonstabeller. I dette tilfellet kan du vurdere å sette sammen noen av eller alle dimensjonstabellene (men ikke faktatabellen) på den forente tjeneren, slik at DB2-optimalisatoren kan fullføre kombineringsene på lokale tabeller.

Spørringen flyttes ned til den fjerntliggende datakilden

Hvis spørringen ikke kan besvares av dataene i sammendragstabellen og eventuelle sammensatte dimensjonstabeller, forsøker DB2-optimalisatoren å flytte spørringen ned til den fjerntliggende tjeneren. Den fjerntliggende tjeneren fullfører spørringen og returnerer resultatsettet til den lokale tjeneren.

Selv om resultatsettet er lite, kan du oppnå betydelige ytelsesforbedringer. Denne metoden kan brukes hvis alle tabellene i det fjerntliggende stjerneskjemaet eller snøfnuggskjemaet finnes på bare en fjerntliggende tjener.

Spørringen kjøres lokalt på data hentet fra fjerntliggende datakilder

Hvis spørringen ikke kan besvares av dataene i sammendragstabellen og eventuelle sammensatte dimensjonstabeller, og ikke kan skyves ned til den fjerntliggende datakilden, blir tabellene som kreves for å tilfredsstille spørringen, hentet opp og kopiert fra den fjerntliggende tjeneren til den lokale tjeneren, og spørringen fullføres lokalt. Hvis for eksempel stjerneskjemaet er på en fjerntliggende zSeries-tjener og spørringen bruker en funksjon som er tilgjengelig i DB2 UDB, men ikke i DB2 for z/OS, kan ikke spørringen utføres på den fjerntliggende tjeneren.

Det er ikke sikkert at dette scenariet forbedrer ytelsen, hvis fakta- og dimensjonstabellene som trengs for å besvare spørringen, er store.

Aktivere et forent system for DB2 Cube Views

Når du skal aktivere det forente systemet for DB2 Cube Views, må du definere den fjerntliggende datakilden, definere kallenavn på den forente tjeneren og definere informasjonsbegrensninger for kallenavnene.

Prerequisites: Kontroller at det er aktivert støtte for DB2 Information Integrator. Hvis du vil aktivere støtte for forente systemer fra en kommandolinje, skriver du: `db2 update dbm cfg using federated yes`. Stopp og start DB2 UDB etter at du har aktivert støtte for forente systemer.

Slik aktiverer du en fjerntliggende datakilde for DB2 Cube Views:

1. Definer fjerntliggende datakilder.
2. Definer kallenavn for fjerntliggende tabeller.
3. Definer informasjonsbegrensninger for kallenavn.

Definere fjerntliggende datakilder

Hvis du skal definere og konfigurere en forent tjener for DB2 Cube Views, må du gi den forente tjeneren informasjon om de fjerntliggende datakildene og objektene du vil bruke.

Slik definerer du en fjerntliggende datakilde:

1. Katalogiser den fjerntliggende databasen og den tilhørende noden i den forente tjenerens databasekatalog, slik at den forente tjeneren vet hvilken fjerntliggende datakilde den skal kobles til. Du kan bruke veiviseren Tilføy database i konfigureringsassistenten for å katalogisere den fjerntliggende databasen og den tilhørende noden.
2. Koble deg til den lokale databasen på den forente tjeneren. Hvis du ikke allerede har en eksisterende lokal database, oppretter du en lokal database, og kobler deg deretter til databasen.
3. Registrer innpakkeren ved å utstede kommandoen `CREATE WRAPPER`. Hvis den fjerntliggende datakilden er fra DB2-familien, for eksempel z/OS, bruker du denne kommandoen:

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

Du må vite hvilket bibliotek som kreves av datakilden.

4. Registrer tjenerdefinisjonene for hver tjener som brukes av de fjerntliggende datakildene du vil ha tilgang til. Opprett for eksempel en DRDA-innpakke for tjenerer i DB2-familien. Bruk denne kommandoen for å registrere en fjerntliggende z/OS-tjener:

```
CREATE SERVER tjenernavn
  TYPE DB2/ZOS
  VERSION 8.1
  WRAPPER DRDA
  AUTHORIZATION "bruker-ID" PASSWORD "passord"
  OPTIONS (DBNAME 'databasenavn')
```

Tips: Du kan kjøre denne kommandoen direkte fra en SQL-skriptfil. Du kan oppgi kommandoen direkte fra et DB2-kommandovindu ved å skrive kommandoen i doble anførselstegn, og bruke doble anførselstegn etter omvendt skråstrek (`\`) rundt bruker-IDen og passordet.

```
DB2 "CREATE SERVER tjenernavn
TYPE DB2/ZOS
VERSION 8.1
WRAPPER drda
AUTHORIZATION \"bruker-ID\" PASSWORD \"passord\"
OPTIONS (DBNAME 'databasenavn')"
```

5. For hver tjener du definerte, angir du at spørringsbelastningen skal flyttes ned fra den forente tjeneren til den fjerntliggende datakilden. Datamengdene på den fjerntliggende datakilden er sannsynligvis store, og du ønsker ikke at DB2 UDB på den forente tjeneren skal forsøke å kopiere dataene fra den fjerntliggende datakilden til den forente tjeneren. Ved å angi maksimal 'pushdown' for tjeneren, vil DB2 UDB alltid først forsøke å flytte spørringen ned til den fjerntliggende datakilden, slik at bare resultatsetter fra spørringen blir kopiert til den forente tjeneren. Denne innstillingen er nødvendig hvis du vil oppnå ytelsesforbedringer fra DB2 Cube Views-optimalisering. For eksempel:

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

6. Opprett brukertilordningen slik at innpakkeren automatisk kan koble til tjeneren. For eksempel:

```
CREATE USER MAPPING FOR USER
SERVER tjenernavn
OPTIONS(REMOTE_AUTHID 'bruker-ID', REMOTE_PASSWORD 'passord')
```

Definere kallenavn for fjerntliggende tabeller for DB2 Cube Views

Definer kallenavn for hver fjerntliggende tabell som er en del av stjerneskjemaet eller snøfnuggskjemaet du oppretter på den forente tjeneren. DB2 Cube Views krever at kallenavnene arbeider med de fjerntliggende tabellene.

Slik definerer du et kallenavn for en tabell på en fjerntliggende datakilde: Med setningen CREATE NICKNAME definerer du et kallenavn for hver fjerntliggende tabell du trenger tilgang til. For eksempel:

```
CREATE NICKNAME lokalt_skjemanavn.lokalt_kallenavn
FOR fjerntliggende_tjenernavn.fjerntliggende.skjemanavn.fjerntliggende_tabellnavn
```

Tips: Bruk det samme skjemanavnet for lokale kallenavn som er definert på den fjerntliggende tjeneren. For eksempel:

```
CREATE NICKNAME CVSAMPLE.FAMILY FOR fjerntliggende_tjenernavn.CVSAMPLE.FAMILY
```

Definere informasjonsbegrensinger for kallenavn for DB2 Cube Views

Definer informasjonsbegrensinger for å dokumentere forhold mellom data som kan forbedre ytelsen. Du bør definere informasjonsbegrensinger for fremmednøkler mellom tabeller i stjerneskjemaet eller snøfnuggskjemaet på den forente tjeneren.

Veiviseren for optimaliseringsrådgiveren i DB2 Cube Views trenger informasjonsbegrensninger for å kunne anbefale sammendragstabeller som forbedrer ytelsen. DB2-optimalisatoren bruker også begrensningene til å behandle spørringene mer effektivt, og for å rute spørringer til eksisterende sammendragstabeller.

DB2 Information Integrator definerer automatisk primærnøkkelbegrensninger på den forente tjeneren som samsvarer med eksisterende primærnøkkelbegrensninger

på den fjerntliggende datakilden. Du må opprette informasjonsbegrensninger for fremmednøkler du bruker til å bygge stjerneskjemaet eller snøfnuggskjemaet på den forente tjeneren.

Slik definerer du en fremmednøkkel:

Bruk setningen ALTER NICKNAME for å tilføye informasjonsbegrensninger til kallenavnene. For eksempel:

```
ALTER NICKNAME lokalt_skjemanavn
  ADD FOREIGN KEY(kolonnenavn)
  REFERENCES lokalt_skjemanavn.lokalt_kallenavn (kallenavn)
  ON DELETE RESTRICT
  NOT ENFORCED
  ENABLE QUERY OPTIMIZATION
```

Du har fullført aktiveringen av en fjerntliggende datakilde for DB2 Cube Views.

Deretter bruker du DB2 Cube Views til å bygge en fullstendig kubemodell for stjerneskjemaet eller snøfnuggskjemaet på den forente tjeneren. Når du har bygd en fullstendig kubemodell, kan du bruke veiviseren for optimaliseringsrådgiveren til å optimalisere kubemodellen.

Problemløsning for spørringsytelse ved fjerntliggende datakilder

Hvis spørringene til en fjerntliggende datakilde ikke gir den forventede effekten etter optimalisering, kontrollerer du at det forente systemet er riktig konfigurert for DB2 Cube Views.

Du må fullføre trinnene nedenfor før du kan forvente at ytelsen ved spørringer til fjerntliggende datakilder blir forbedret:

1. Aktivere ditt forente system for DB2 Cube Views.
2. Opprette en fullstendig kubemodell som oppfyller de grunnleggende reglene, reglene for kubemodellers fullstendighet og reglene for optimalisering som er beskrevet i "Regler for metadataobjekter" på side 38.
3. Optimaliser en kubemodell.
4. Hvis spørringsytelsen ikke blir bedre, kontrollerer du trinnene som er beskrevet i "Problemløsning for sammendragstabeller" på side 119.

Hvis spørringsytelsen ikke blir forbedret etter at du har fullført trinnene over, bør du vurdere følgende:

- Kontroller at alle aktuelle begrensninger er definert.
- Kontroller at innstillingen DB2_MAXIMAL_PUSHDOWN er satt til Ja, slik det er beskrevet i "Definere fjerntliggende datakilder" på side 126.
- Vurder å slå sammen dimensjonstabellene som er involvert i spørringene på den forente tjeneren. Ved å slå sammen dimensjonene, slik at en replikert kopi av dimensjonstabellen finnes på den forente tjeneren, kan gi bedre ytelse.

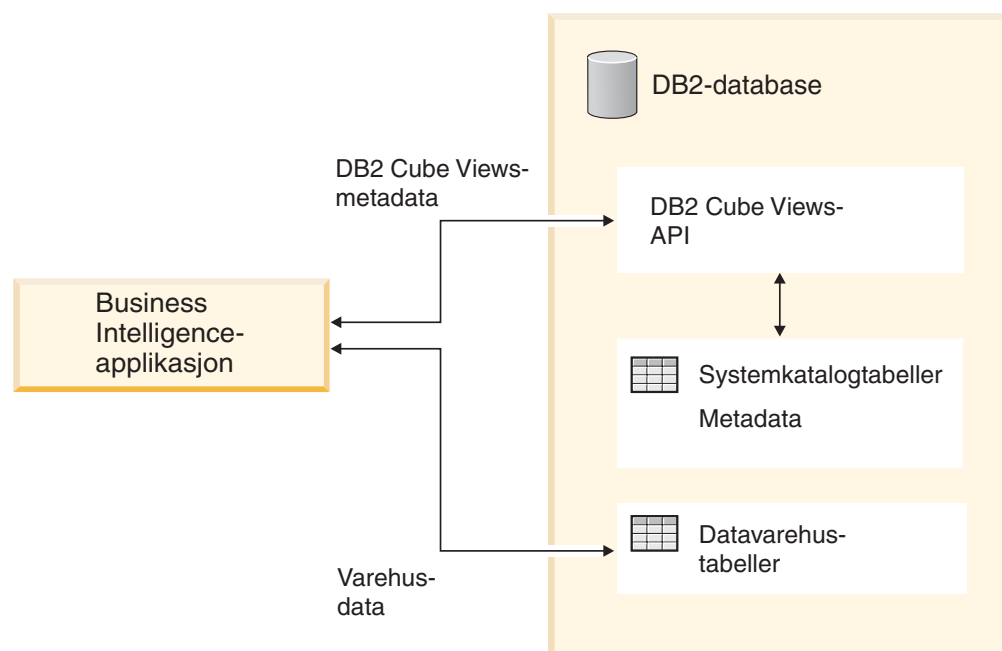
Kapittel 7. DB2 Cube Views-API

DB2 Cube Views, API-oversikt

DB2 Cube Views inneholder et programmeringsgrensesnitt (API) som gir programmeringstilgang til metadata som er lagret i DB2 Cube Views. Ved hjelp av APIen kan applikasjoner virke sammen med metadata ved hjelp av DB2 Cube Views-metadataobjekter uten å måtte virke sammen med relasjonstabeller og kombineringer.

APIen for DB2 Cube Views gir tilgang til metadata som er lagret i systemkatalogtabellene i en DB2-database. Applikasjoner som bruker APIen kan opprette og endre metadataobjekter som modellerer flerdimensjonale strukturer og OLAP-strukturer i et datavarehus.

I figur 39 er det vist hvordan data og metadata utveksles gjennom APIen.



Figur 39. Datautveksling gjennom APIen for DB2 Cube Views

APIen er en enkelt lagret prosedyre som er registrert i en DB2-database. Denne lagrede prosedyren godtar inn- og utdataparametere der du kan uttrykke komplekse metadata og metadataoperasjoner. API-parameterformatet er definert av et XML-skjema.

APIen bruker følgende teknologier for å utveksle metadata mellom og Business Intelligence-applikasjoner:

- SQL med ODBC
- DB2 CLI og JDBC
- XML

DB2 Cube Views API: Lagrede DB2-prosedyrer og XML-analyse

DB2 Cube Views-APIen utveksler metadata mellom en Business Intelligence-applikasjon og en DB2-database via den lagrede DB2 Cube Views-prosedyren og XML-analysering.

Du finner opplysninger om programmering med lagrede DB2-prosedyrer i *DB2 Application Development Guide*. Før du programmer med DB2 Cube Views-APIen, bør du forstå disse begrepene:

Transaksjon

DB2 UDB støtter transaksjonsbasert flerbrukertilgang til metadata. (Se *DB2 Application Development Guide* hvis du ønsker mer informasjon). Alle databasehandlinger som blir utført med DB2 Cube Views-APIen, tilhører databasetransaksjonen til applikasjonen som sender kallet. Derfor kan en applikasjon utføre COMMIT eller ROLLBACK etter at APIen kaller den lagrede prosedyren `md_message` for å administrere databasens arbeidsenheter.

Minnestyring

Parametere blir utvekslet mellom applikasjoner og den lagrede prosedyren `md_message` i form av CLOB-strukturer. Applikasjoner som sender kall til den lagrede prosedyren `md_message`, må forhåndstildele CLOB-parameterstrukturer som har samme størrelse som de som ble brukt til å kategorisere den lagrede prosedyren. APIen støtter den maksimale størrelsen på en CLOB i DB2 UDB, som er 2 GB. Standard CLOB-størrelse er 1 MB.

Systemkonfigurasjon

For å støtte utvekslingen av store parametere er det mulig du må endre disse DB2 UDB-innstillingene:

- Databaseklientapplikasjonen som sender kall til den lagrede prosedyren `md_message`, må kanskje tilkobles med større minneområde- og stakkstørrelse.
- DB2-minneområdet for spørring for databasen må kanskje økes ved hjelp av `query_heap_sz`-innstillingen.

XML-analyse

Applikasjoner som bruker APIen må analysere utdataparameteren som returneres av den lagrede prosedyren `md_message`. En rekke XML-analysatorer er tilgjengelige for utviklere som ønsker å bruke APIen.

Feilbehandling

Feilinformasjon genereres i tre formater av APIen:

- SQLCODE- og SQLSTATE-informasjon returnert fra den lagrede prosedyren til applikasjonen som sendte kallet
- XML-strukturer som leveres til applikasjonene som sendte kallet, ved hjelp av API-parameteren **response**
- Feil- og kjøretidsloggfiler som er lagret på databasetjeneren som kjører APIen

Hvis det oppstår en feil på grunnlag av XML-validering, -analyse eller -koding, blir parameteren **response** returnert til applikasjonen som sendte kallet, med koden

<error> i stedet for en operasjonskode. Denne <error> XML-koden inneholder en <status>-kode med en returkode og en melding som beskriver problemet APIen oppdaget.

Hvis det oppstår en feil i APIen, som ikke er knyttet til XML-behandling, men som likevel er knyttet til utføringen av en metadataoperasjon, blir innholdet i **response**-parameteren returnert.

Eksempelet nedenfor viser typen av informasjon i en <error>-kode. I dette eksempelet bruker beskrivelser av parameterstrukturene bare et begrenset antall XML-koder. De fleste parametere har flere XML-koder enn det som er vist her, og parameterinnholdet blir validert ved hjelp av XML-skjemaet.

```
<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="Systemet kunne ikke analysere XML for
  &quot;INPUT PARAMETER&quot; (linje:&quot;3&quot;,
  tegn:&quot;26&quot;, melding:&quot;Ukjent element
  'dropa'&quot;)." type="error"/>
</error>
</olap:response>
```

Når det blir sendt et kall til den lagrede prosedyren for DB2 Cube Views, uavhengig av om den lagrede prosedyren faktisk ble utført eller ikke, returnerer DB2 UDB en SQLCODE og en SQLSTATE til applikasjonen som sendte kallet. Hvis den lagrede prosedyren for DB2 Cube Views kan utføres, returnerer den lagrede prosedyren en statusmelding som en del av XML-dataene som blir sendt til applikasjonen som sendte kallet.

Lagret prosedyre for DB2 Cube Views

Den lagrede prosedyren kalles `md_message`, og den behandler parametere som er uttrykt i parameterformatet til DB2 Cube Views.

Prosedyren trekker ut operasjons- og metadatainformasjon fra inndataparameterne og utfører deretter de forespurte metadataoperasjonene. Prosedyren genererer utdataparametere som inneholder utføringsstatus (vellykket eller mislykket) for operasjonene og metadatainformasjon, avhengig av operasjonen.

Den lagrede prosedyren kjører som brukeren som er oppgitt i `.fenced`-filen på UNIX-systemer. Beskyttede brukere må ha skrivetilgang til loggfilene som er oppgitt i `filendb2md_config.xml`. Ellers kan ikke den lagrede prosedyren logge noe som helst.

Den lagrede DB2 Cube Views-prosedyren er implementert som en lagret DB2 UDB-prosedyre. Den kan brukes av alle applikasjoner som bruker noen av programmeringsgrensesnittene for DB2 UDB. Det skiller mellom store og små bokstaver i navnet på prosedyren, men ikke i navnet på og innholdet i parameterne for den lagrede prosedyren. Syntaksen til `md_message` og en prototype er:

```
Syntaks:  call md_message (request, metadata, response)
Prototyp: md_message (request IN  CLOB(1M),
                      metadata INOUT CLOB(1M),
                      response OUT  CLOB(1M))
```

Parameterne **request**, **metadata** og **response** er av typen CLOB, som er en DB2 UDB-datatype. En applikasjon legger inn data i parameteren **request** med en

beskrivelse av operasjonen som skal utføres, og den kan eventuelt legge inn metadataene som operasjonen skal utføres på, i parameteren **metadata**. Etter å ha implementert inndataparameterne, returnerer md_message status for operasjonen i parameteren **response** og returnerer de forespurte metadataene i parameteren **metadata**. Parameteren **metadata** blir brukt til både inn- og utmetadata. DB2 UDB overfører parameterstrukturene mellom Business Intelligence-applikasjoner og den lagrede prosedyren md_message på databasetjeneren.

Størrelsen på CLOB-argumentene kan variere. Standardstørrelsen er 1 MB. CLOB-argumentstørrelsene er oppgitt i skriptet sqllib/misc/db2mdapi.sql. Du kan katalogisere den lagrede prosedyren på nytt med en hvilken som helst størrelse på CLOB-parameterne opptil 2 GB. Når du øker størrelsen på parameteren, bruker den lagrede prosedyren mer minne når den kjøres, fordi utdataparameterbuffere blir tildelt på forhånd til den katalogiserte størrelsen når den lagrede prosedyren startes. Hvis størrelsen er for liten, blir dataene i inn- og utdataparameterne avkuttet.

Hvis du vil endre standard CLOB-størrelse, registrerer du den lagrede prosedyrens API på nytt med økte størrelsesgrenser.

1. Rediger filen sqllib/misc/db2mdapi.sql og øk størrelsesgrensene for CREATE PROCEDURE-setningen.

```
CREATE PROCEDURE
  DB2INFO.MD_MESSAGE(IN  request CLOB(1M),
                    INOUT metadata CLOB(10M),
                    OUT  response CLOB(10M))
```

2. Kjør filen db2mdapi.sql på nytt for å registrere den lagrede prosedyrens API på nytt. Ignorer eventuelle feil mens skriptet forsøker å opprette tabeller som allerede finnes.

Hvis du ønsker informasjon om hvordan du sender et kall til den lagrede prosedyrens API fra C++, kan du se eksempelet på C++-kildekode i filen sqllib/samples/olap/client/db2mdapiclient.cpp.

Eksempelet nedenfor viser hvordan du sender et kall til den lagrede prosedyren for DB2 Cube Views fra en ennfelt SQL-applikasjon:

```
// 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
// ...

```

Konfigurasjonsfil for DB2 Cube Views

APIen kan konfigureres på DB2-forekomstnivå. Du kan endre parameterne i en konfigurasjonsfil med navnet `db2md_config.xml`.

Hver installasjon av DB2 Cube Views har en standard konfigurasjonsfil i katalogen `db2-installeringsbane/cfg`. På Windows kan for eksempel konfigurasjonsfilen ligge i katalogen `c:\sqlib\cfg`, og på AIX kan konfigurasjonsfilen ligge i katalogen `/usr/opt/db2_08_01/cfg`.

Hver DB2-forekomst som kjører DB2 Cube Views, har en fysisk kopi av filen `db2md_config.xml` i katalogen `db2_forekomstbane`. På Windows kan for eksempel den fysiske kopien ligge i katalogen `c:\sqlib\min_forek`, og på AIX kan den fysiske kopien ligge i katalogen `~min_forek/sqlib`.

Funksjonen `db2icrt` kopierer standard konfigurasjonsfil til katalogen `db2_forekomstbane`, og oppretter en ny forekomst. For DB2-forekomster som ble opprettet før DB2 Cube Views ble installert, kan du kopiere konfigurasjonsfilen manuelt til forekomstkatalogen hvis installeringsprogrammet ikke kopierte filen. Hvis APIen ikke finner konfigurasjonsfilen i forekomstkatalogen, prøver APIen å kopiere standard konfigurasjonsfil til forekomstkatalogen.

En konfigurasjonsfil med navnet `db2md_config.xml` blir brukt til å definere feillogging og kjøretidssporing. Ved å endre innholdet i konfigurasjonsfilen kan en administrator oppgi sporingsnivået, alvorlighetsgraden for feil som skal logges, og bufferstørrelsen (i byte) som skal brukes til logging.

Innholdsstrukturen til konfigurasjonsfilen `db2md_config.xml` er definert av XML-skjemafilen `db2md_config.xsd`. Eksempelet nedenfor viser innholdet i konfigurasjonsfilen.

```
<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>
```

Konfigurasjonsfiler

DB2 Cube Views-metadatatabeller og XML-skjemafilene

Noen av funksjonene i APIen støttes ikke hvis versjonsnummeret ikke er riktig.

Versjoner av metadatatabeller

APIen fungerer bare hvis den er tilkoblet en DB2-database med et gjeldende sett med DB2 Cube Views-metadatatabeller. Den gjeldende versjonen for DB2 Cube Views versjon 8.2 er 8.2.0.1.0. Versjonsnummeret for DB2 Cube Views versjon 8.1 er 8.1.2.0. Versjonsnummeret for metadatakatalogtabellen for DB2 Cube Views er lagret i tabellen SYSINFOVERSION.

Du finner flere opplysninger om metadatatabellversjoner under "Oversikt over db2mdapiclient-funksjonen" på side 163.

DB2 Cube Views XML-skjemafilene, eller XSD-filene, brukes for DB2 Cube Views-APIen. XML-skjemafilene brukes av den lagrede prosedyren API DB2INFO.MD_MESSAGE.

Versjoner av XML-skjemafilene

Alle XML-dokumentene som blir overført til og fra metadata-APIen, må ha et versjonsnummer. Ved hjelp av dette versjonsnummeret vet den lagrede prosedyren hvilket XML-skjema klienten bruker. XSD-skjemafilene angir hvilke versjonsnumre som er akseptable for bestemte operasjoner.

Den gjeldende versjonen for DB2 Cube Views versjon 8.2 er 8.2.0.1.0.

De følgende skjemafilene inneholder informasjon om den lagrede prosedyren md_message().

Filen db2md_parameter.xsd

Denne filen inneholder informasjon om request- og response-operasjoner. Denne filen fungerer med versjon 8.1.2.1.0 og versjon 8.2.0.1.0.

Filen db2md_metadata.xsd

Den filen inkluderer filen db2md_types.xsd, og angir bare metadataelementer. Den filen inneholder ikke versjonsinformasjon.

Filen db2md_types.xsd

Denne filen inneholder informasjon om alle metadataobjekter. Denne filen fungerer med versjon 8.1.2.1.0 og versjon 8.2.0.1.0.

APIen kan støtte den tidligere versjonen av skjemafiler, versjon 8.1.2.1.0, men støtter bare Describe- og Translate-operasjonene. APIen returnerer en feil hvis den ønskede versjon 8.1.2.1.0-operasjonen er en annen enn Describe eller Translate.

Metadataobjektformat

XML-skjemaet i DB2 Cube Views definerer XML-basiselementene som er direkte tilordnet til objektene i metadataobjektmodellen i DB2 Cube Views. Komplekse metadatastrukturer representeres deretter som sekvenser av disse basiselementene. Tilknytninger mellom objekter i komplekse metadatastrukturer blir avlest gjennom navnerreferanser mellom basiselementer.

Et eksempel på en navnerreferanse er hvordan et kubeelement kan inneholde en referanse til et dimensjonselement. Det følgende eksempelet viser typen av data som oppgis for et kubeobjekt slik det er definert av XML-skjemaet. I dette eksempelet vises bare tekstbeskrivelser. I applikasjonen brukes XML-representasjoner av informasjonen.

```
kube
->kubemodellreferanse
->kubedimensjonsreferanser
->kubefaktareferanse
->utsnitt
```

I tilfellet av et kubeobjekt finnes alle referansene til andre typer av objekter i basiselementet som representerer kubene. Med ikke-rekursive henteoperasjoner blir sekvenser av kubeobjekter (og kun kubeobjekter) presentert. Med rekursive henteoperasjoner presenteres informasjon om kubeobjekter i tillegg til informasjon om alle andre objekter (av forskjellige typer) som de identifiserte kubene refererer til.

Rekkefølgen til objektene er definert av XML-skjemaet i DB2 Cube Views. Innenfor omfanget av en enkelt operasjon blir objekter av samme type (f.eks. kubeobjekter) samlet i en gruppe. Innenfor disse gruppene påvirkes rekkefølgen til elementene av referansene mellom objekter av samme type. Objekter det er referert til, må forekomme før objektene som refererer til dem. Rekkefølgen mellom disse gruppene:

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

Rekkefølgen til objekttypegruppene er uavhengig av tilknytningene mellom objektene. Det faktum at attributter og kombineringsroller spiller forskjellige roller når de er knyttet til forskjellige objekttyper, påvirker ikke rekkefølgen deres innenfor tilknytningsformatet.

For et eksempel på en XML-fil som viser typen og rekkefølgen for metadatainformasjonen, kan du se XML-metadadatafilen som er inkludert med CVSAMPLE.

Sekvens for operasjonstrinnene

Bare tre av operasjonsargumentene bestemmer omfanget til en operasjon.

Disse tre argumentene er oppført her i den rekkefølgen de brukes i en operasjon:

1. objectType
2. restriction
3. recurse

Eksempelet nedenfor viser hvordan du kan få returnert objekter som i utgangspunktet ikke samsvarer med begrensningen du oppgir, men blir returnert som en del av den rekursive fasen av operasjonen.

Eksempel: Rekursiv describe-operasjon for kubene som tilhører skjemaet myschema:

Operasjonsargumenter:

```
objectType = "cube"  
restriction = <restriction>  
    <predicate property="schema" operator="=" value="myschema" />  
    </restriction>  
recurse = "yes"
```

Describe-operasjonen begynner med å begrense omfanget til kubeobjekter. Av disse kubeobjektene velges bare de som tilhører skjemaet myschema. For hvert av de valgte kubeobjektene velges objekter som de refererer til, og disse objektene er av forskjellig type og kan tilhøre forskjellige skjemaer. Alle de valgte objektene returneres deretter som en del av en response-parameter til kubeforespørselen.

DB2 Cube Views, API-parametere

Parametere for DB2 Cube Views API-metadatooperasjoner

APIen for DB2 Cube Views inneholder tre typer av metadatooperasjoner: henting, endring og administrasjon. Hver type omfatter en eller flere operasjoner, og hver operasjon har ett sett med parametere.

Parameterformatet definerer standarden for hvordan metadatooperasjonene og -objektene fremstilles og utveksles mellom Business Intelligence-applikasjoner og DB2 Cube Views. Parameterformatet bruker XML til fremstillingen av metadatooperasjoner og -objekter i DB2 Cube Views. XML-skjemaet definerer parameterformatet.

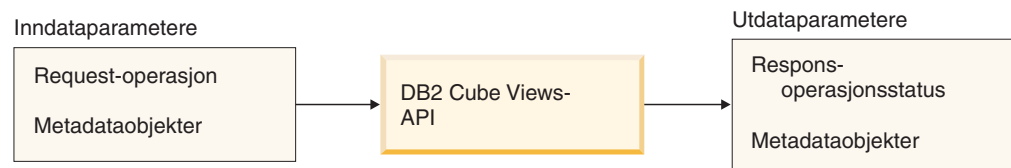
Inndata- og utdataparametere

Hver operasjon i den lagrede prosedyren md_message har to inndata- og to utdataparametere.

Inndataparameterne er **request**-parameteren og **metadata**-parameteren. Utdataparameterne er **response**-parameteren og **metadata**-parameteren.

Inndata	Parameterinnhold
Request	Inneholder operasjonen som er forespurt av den lagrede prosedyren. Forespørselen kan inkludere alternativer som påvirker forespørselens virkemåte og omfang.
Metadata	Inneholder metadataobjekter som skal brukes sammen med operasjonen som er beskrevet i parameteren request . Noen request-operasjoner, for eksempel Describe, krever ikke inndatametadata.
Utdata	
Response	Inneholder alle resultatene til operasjonen som ble utført av den lagrede prosedyren, unntatt metadataobjektene.
Metadata	Inneholder metadataobjekter som ble forespurt av operasjonen som er beskrevet i inndataparameteren request . Noen operasjoner, som Create, returnerer ikke utdatametadata.

I figur 40 vises det hvordan inndataparameterne fra applikasjonen som sender forespørselen, går via DB2 Cube Views-APIen til utdataparameterne. APIen oppretter en representasjon av metadataobjektene som kan leses av DB2 UDB.



Figur 40. API-parametere

Eksempler på inndata- og utdataparametere i metadataoperasjoner

Eksemplene nedenfor viser hvordan du kan strukturere parametere i tre typer av metadataoperasjoner. I disse eksemplene er deler av XML-strukturene utelatt, men de er representert med en ellipse (...).

Henteoperasjon

Eksemplene nedenfor viser hvordan en henteoperasjon kalt describe er strukturert. Se "Henteoperasjon: Describe" på side 143 hvis du ønsker flere opplysninger om operasjonen describe. I det følgende eksempelet er parameteren **metadata** tom i inndataene, men inneholder data i utdataene.

Request og metadata

```

<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" ... />

```

Response og metadata

```

<olap:response xmlns:olap="http://www.ibm.com/olap" ... >
  <describe>
    <status id="0" text="Operasjonen er
      fullført." 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>

```

Endringsoperasjoner

Eksempelene nedenfor viser hvordan en endringsoperasjon kalt create er strukturert. Se “Endringsoperasjoner: Alter, Create, Drop, Import og Rename” på side 149 hvis du ønsker flere opplysninger om create-operasjonen og andre endringsoperasjoner. Parameteren **metadata** inneholder data i inndataene, men er tom i utdataene.

Request og metadata

```

<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>

```

Response og metadata

```

<olap:response xmlns:olap="http://www.ibm.com/olap" ... >
  <create>
    <status id="0" text="Operation completed
      successfully." type="informational"/>
  </create>
</olap:response>

<olap:metadata xmlns:olap="http://www.ibm.com/olap" ... >

```

Administrasjonsoperasjon

Eksempelene nedenfor viser hvordan en administrasjonsoperasjon kalt validate er strukturert. Se “Administrasjonsoperasjoner: Validate og Translate” på side 154 hvis du ønsker flere opplysninger om operasjonen validate.

Request og metadata

```
<olap:request xmlns:olap="http://www.ibm.com/olap" ... >
  <validate objectType="cube" mode="base">
    <restriction>
      <predicate property="schema" operator="=" value="myschema"/>
    </restriction>
  </validate>
</olap:request>

<olap:metadata xmlns:olap="http://www.ibm.com/olap" ... />
```

Response og metadata

```
<olap:response xmlns:olap="http://www.ibm.com/olap" ... >
  <validate>
    <status id="1" text="...Ytterligere informasjon
      ble returnert." type="informational"/>
    <info><message id="6299" text="Minst ett
      databaseutsnitt ble funnet under valideringen."
      type="warning"/></info>
  </validate>
</olap:response>

<olap:metadata xmlns:olap="http://www.ibm.com/olap" ... >
```

Du finner flere XML-eksempelfiler som kan brukes med funksjonen db2mdapiclient i katalogen SQLLIB\samples\olap\xml\input.

Metadataoperasjoner

Parametere for operasjoner

Forskjellige parametere er tilgjengelige for hver metadataoperasjon. Disse parametrene tilpasser virkemåten til en operasjon til et bestemt bruksområde.

DB2 Cube Views har fem parametere for metadataoperasjonene:

- objectType
- recurse
- restriction
- mode (for Import-operasjonen)
- mode (for Validate-operasjonen)

objectType-parameter

Denne parameteren angir typen av metadataobjekter som er med i den forespurte operasjonen. Disse objekttypene svarer direkte til metadataobjektmodellen i DB2 Cube Views.

- all
- cubeModel
- cube
- cubeDimension
- cubeFacts
- cubeHierarchy
- cubeLevel
- dimension
- facts

- hierarchy
- attributeRelationship
- level
- join
- measure
- attribute

recurse-parameter

Denne parameteren styrer om en operasjon skal utføres rekursivt. I ikke-rekursiv modus utfører en operasjon handlingene bare på metadataobjektene som svarer direkte til spesifikasjonene for parameterne **objectType** og **restriction**. En operasjon starter med settet med metadataobjekter som tilsvarer ikke-rekursiv modus, og utfører i tillegg handlingene på alle andre andre metadataobjekter som påtreffes langs alle objekttilknytningsbanene som begynner fra settet med objekter fra ikke-rekursiv modus. Parameteren **recurse** inkluderer alternativet *yes* og alternativet *no*.

En ikke-rekursiv operasjon kan for eksempel returnere en liste med dimensjoner, mens en rekursiv operasjon kan returnere en liste med dimensjoner i tillegg til alle andre objekter (av forskjellig type) som det er referert til av disse dimensjonene, og objekter det er referert til av disse dimensjonsobjektene.

restriction-parameteren

Denne parameteren oppgir at en metadataoperasjon skal være begrenset i omfang. Denne parameteren tilsvarer predikater i en SQL-spørring. Begrensninger blir uttrykt i XML ved hjelp av kodene <restriction> og <predicate>, som er definert av DB2 Cube Views-XML-skjemaet.

Begrensninger kan baseres på de objekttegenskapene som er felles for alle metadataobjektene, og på forholdene mellom metadataobjekter.

Egenskapsbaserte predikater inneholder disse attributtene:

egenskap

Tilknyttet en predikatkode, og må oppgi et *navn*-attributt eller et *skjema*-attributt.

operator

Tilknyttet en predikatkode, og må oppgi likehetstegnet (=).

verdi

Knyttet til predicate-koden, og er strengrepresentasjonen av verdien som skal sammenliknes med egenskapen som er oppgitt av attributtet *property*.

Se "Sekvens for operasjonstrinnene" på side 136 hvis du ønsker en beskrivelse av hvordan parameteren **restriction** er knyttet til den generelle sekvensen av operasjonstrinn.

Dette eksempelet begrenser omfanget av en operasjon til objektene i skjemaet ABC:

```
<restriction>
  <predicate property="schema" operator="=" value="ABC">
</restriction>
```

mode-parameter (for Import)

Denne parameteren definerer modusen for operasjonen Import. Den følgende tabellen beskriver de tilgjengelige modusene.

Det oppstår en kollisjon hvis et objekt som blir overført til importoperasjonen som inndata, allerede finnes i metadatakatalogen.

Tabell 39. Import-modi

Modus	Beskrivelse	Returnerte referanselister
Create new: ignore collisions	<ul style="list-style-type: none">Inndataobjekter som ikke kolliderer, blir opprettet.Inndataobjekter som kolliderer, blir ikke opprettet.Objekter som finnes fra før, blir ikke endret.Det blir ikke generert feil av kollisjoner.	newList Inneholder paret av skjema og navn for objektene som ble opprettet. collisionList Inneholder parene av navn og skjema for objektene i en kollisjon som ble oversett og ikke ble opprettet.
Create new: replace collisions	<ul style="list-style-type: none">Inndataobjekter som ikke kolliderer, blir opprettet.Inndataobjekter som kolliderer, erstatter objekter som finnes fra før.Objekter som finnes fra før, blir erstattet av inndataobjektene.Det blir ikke generert feil av kollisjoner.	newList Inneholder paret av skjema og navn for objektene som ble opprettet. collisionList Inneholder parene av navn og skjema for objektene i en kollisjon som ble erstattet.
Create new: abort on collision	<ul style="list-style-type: none">Inndataobjekter blir bare opprettet hvis det ikke finnes noen kollisjon for hele operasjonen.Hvis det finnes en kollisjon, blir ingen objekter opprettet i operasjonen.Objekter som finnes fra før, blir ikke endret.Det blir generert feil ved kollisjoner.	newList Inneholder parene av navn og skjema for objektene som ble opprettet, eller de ikke-kolliderende objektene som ikke ble opprettet. collisionList Inneholder parene av navn og skjema for objektene i en kollisjon som ikke ble opprettet.
Report new: report collisions	<ul style="list-style-type: none">Ingen objekter blir opprettet.Rapporterer kollisjonsstatus for inndataobjektene.Objekter som finnes fra før, blir ikke endret.Det blir ikke generert feil av kollisjoner.	newList Inneholder parene av navn og skjema for objektene som ikke var i en kollisjon og som ikke ble opprettet. collisionList Inneholder parene av navn og skjema for objektene i en kollisjon som ikke ble opprettet.

Koder for newList- og collisionList-referanselister må overholde en forhåndsdefinert rekkefølge. Den følgende listen viser rekkefølgen til referansetyper for IMPORT-operasjonen:

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-parameter (for Validate)

Denne parameteren definerer modusen for operasjonen Validate. Den følgende tabellen beskriver de tilgjengelige modusene for Validate-operasjonen. Reglene i den følgende tabellen refererer til kategorier i DB2 Cube Views-objektreglene.

Tabell 40. Validate-modi

Modus	Beskrivelse
base	Kontrollerer overholdelse av de grunnleggende reglene
cubeModel completeness	<ul style="list-style-type: none"> • Kontrollerer overholdelse av fullstendighetsreglene for kubemodeller • Kontrollerer overholdelse av de grunnleggende reglene
optimalisering	<ul style="list-style-type: none"> • Kontrollerer overholdelse av optimaliseringsreglene • Kontrollerer overholdelse av fullstendighetsreglene for kubemodeller • Kontrollerer overholdelse av de grunnleggende reglene

Operander i operasjoner

Når en operasjon krever at metadataobjekter eller deres referanser finnes i forespørselen, kalles disse objektene eller referansene operander i operasjonen.

Operandene som overføres til metadataoperasjoner av enten **request-** eller **metadata-**parameterne, er:

objekt Denne operanden inneholder metadataobjektene som det blir utført handlinger på. Formatet som brukes for metadataobjekter, er beskrevet i "Metadataobjektformat" på side 135.

currentRef

Denne operanden er for Rename-operasjonen, og den inneholder metadataobjektets skjema og navn.

newRef

Som for operanden *currentRef* brukes denne operanden for Rename-operasjonen, og inneholder metadataobjektets skjema og navn.

Henteoperasjon: Describe

DB2 Cube Views har en henteoperasjon: Describe.

Describe-operasjon støtter både XML-versjon 8.1.2.1.0 og 8.2.0.1.0. Alle andre operasjoner krever XML-versjon 8.2.0.1.0. Du finner informasjon om versjoner av XML-skjemafilene under "DB2 Cube Views-metadatatabeller og XML-skjemafilene" på side 134.

Describe

Denne operasjonen henter informasjon om metadataobjekter. Operasjonen returnerer informasjon for ett eller flere metadataobjekter med oppgitt *objectType* (f.eks. ett enkelt dimensjonsobjekt, et sett med dimensjonsobjekter, et sett med objekter som omfatter alle objekttyper). Metadataobjekter blir returnert i **metadata**-parameteren. Formatet som brukes for de hentede metadataobjektene, er beskrevet i "Metadataobjektformat" på side 135. I rekursiv modus returnerer denne operasjonen også informasjon for alle andre metadataobjekter som er påtruffet langs alle objekttilknytningsbaner som begynner fra settet med objekter som oppfyller *objectType*-spesifikasjonen.

I tabell 41 vises komponentene i **request**-parameteren i en Describe-operasjon. I tabell 42 vises komponentene i Describe-operasjonens **response**-parameter.

Tabell 41. Request-parametere i Describe

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
objectType	argument	XML-streng	all cubeModel cube cubeFacts cubeDimension cubeHierarchy cubeLevel dimension facts hierarchy attributeRelationship level join measure attribute	Typer DB2 Cube Views-metadataobjekter som kan hentes. Du kan oppgi en av de gyldige verdiene for denne parameteren.
restriction [valgfritt]	argument	XML-streng	Se "Parametere for operasjoner" på side 139.	Begrenser omfanget av henting av metadata. Dette tilsvarer predikater i SQL.
recurse	argument	XML-streng	yes no	Aktiverer eller deaktiverer rekursiv henting av metadataobjekter.

Tabell 42. Response-parametere i Describe

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
status	status-melding	DB2 Cube Views, meldingsstruktur	Se "Meldingers struktur" på side 156.	Melding som angir statusen for Describe-operasjonen.
objekt	hente-resultater	XML	Se "Operander i operasjoner" på side 142.	De forespurte metadataobjektene. Denne verdien er en tom streng hvis det oppstod en feil under operasjonen.

Advise-operasjonen: Advise

Denne operasjonen anbefaler hvilke sammendragstabeller som skal bygges for å forbedre spørringsytelsen for den oppgitte kubemodellen. Denne operasjonen har argumenter som begrenser hvor lenge rådgiveren kan kjøre, og hvor mye lagerplass den kan bruke for sammendragstabellene.

Tabell 43. Request-parametere i Advise

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
cubeModelRef	argument	XML-element		Kubemodell som skal optimaliseres.
tablespaceName [valgfritt]	argument	XML-streng		Tabellplass for sammendragstabellene. Hvis det ikke finnes noen tabellplass, vil ikke den genererte SQL-spørringen angi noen tabellplass. I dette tilfellet legger DB2 UDB tabellene i en standard tabellplass.
indexspaceName [valgfritt]	argument	XML-streng		Tabellplass for sammendragstabellindeksene. Hvis det ikke finnes noen tabellplass, vil ikke den genererte SQL-spørringen angi noen tabellplass. I dette tilfellet legger DB2 UDB indeksene i en standard tabellplass.
diskspaceLimit [valgfritt]	argument	XML ikke-negativt heltall		Lagerplass (i byte) som er tilgjengelig for sammendragstabeller og indekser. Hvis du oppgir, bestemmer rådgiveren hvor mye lagerplass som skal brukes. Rådgiveren gir bedre anbefalinger hvis du oppgir større lagerplass. 0 betyr ubegrenset. Standardverdien er 0.
timeLimit [valgfritt]	argument	XML ikke-negativt heltall		Tiden (i sekunder) som skal brukes til å anbefale spørringer. Rådgiveren gir bedre anbefalinger hvis du gir den mer tid. 0 betyr ubegrenset, og rådgiveren stopper når den finner ut at det ikke vil gi bedre anbefalinger hvis den bruker mer tid. Standardverdien er 0.

Tabell 43. Request-parametere i Advise (fortsettelse)

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
dataavlesing [valgfritt]	argument	XML-streng	ja eller nei	Angir om dataavlesing av kubemodellens basistabeller skal utføres. Dataavlesing gir bedre anbefalinger, men gjør at rådgiveren bruker lengre tid. Hvis dataavlesing ikke tillattes, er rådgiverens anbefalinger basert på databasestatistikk. Med dataavlesing utnytter rådgiveren både databasestatistikk og avlesing. Hvis du har et lite replikat av originaldataene, kan DB2 UDB-statistikken endres slik at tabellene ser like store ut som de opprinnelige tabellene. Hvis dataavlesing ikke er angitt, gir rådgiveren de samme anbefalingene som den gav for de opprinnelige tabellene uten dataavlesing.
forny [valgfritt]	argument	XML-streng	utsatt eller umiddelbar	Oppgir om DB2 UDB skal forsøke å fornye sammendragstabellene umiddelbart når basistabellene endres, for å sikre at de er synkronisert. Fornying utsatt betyr at du må oppgi når sammendragstabellene blir fornyet. For umiddelbar fornying vil DB2 oppdatere sammendragstabellene samtidig med at basistabellene blir oppdatert. Hvis basistabellene blir oppdatert med datainnlastinger, må du oppgi fornyingen manuelt. Sammendragstabellene med umiddelbar fornying blir inkrementelt oppdatert, mens sammendragstabeller med utsatt fornying blir helt gjenoppbygd. Det er mange begrensninger ved bruk av Forny umiddelbart, og rådgiveren kan velge å bruke utsatt, selv om du har oppgitt umiddelbar fornying.

Følgende eksempel viser en Advise-operasjon:

```
<olap:request xmlns:olap="http://www.ibm.com/olap" ... >
<advise
  tablespaceName="TS_MQTTABLE"
  indexspaceName="TS_MQTINDEX"
  diskpaceLimit="75000"
  timeLimit="300"
  sampling="yes"
  refresh="immediate">
<cubeModelRef name="SalesModel" schema="MDSAMPLE"/>
</advise>
</olap:request>
```

Tabell 44. Response-parametere i Advise

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
status	statusmelding	DB2 Cube Views, meldingsstruktur		Melding som angir statusen for Advise-operasjonen.
info	meldingsliste	Liste over meldingsstrukturer		Varselmeldinger som angir mindre alvorlige feil som oppstod, og mulige endringer i resultatene av Advise-operasjonen fra det du oppgave i Advise-forespørselen. En Advise-operasjon kan for eksempel overstyre spesifikasjonen av REFRESH IMMEDIATE for materialiserte spørretabeller, fordi det finnes ikke-distributive mål i den anbefalte materialiserte spørretabellen. Informasjonsmeldinger kan vise betingelser som forårsaket at deler av modellen ikke ble optimalisert, eller vise informasjon om hvilke optimaliseringer som ble utført, inkludert hvorfor samlingene og indeksene ble valgt.
anbefaling		XML-element		En beregning av den samlede lagerplasen som kreves av alle anbefalte sammendragstabeller og indekser.
sql		XML-element		SQL for å opprette en sammendragstabell, fyll den med data og opprette indekser for den. Du er ansvarlig for å kjøre SQL-koden.
refreshSql		XML-element		SQL for å fornye sammendragstabellene for å synkronisere dem med basistabeller som ble oppdatert. Du er ansvarlig for å kjøre SQL-koden. For sammendragstabeller av typen forny umiddelbart oppdaterer DB2 UDB noen ganger sammendragstabellene automatisk når basistabellene blir oppdatert. Denne handlingen avhenger av hvordan basistabellene er endret.

Det følgende eksempelet viser en Advise response-operasjon:

```
<olap:response xmlns:olap="http://www.ibm.com/olap" ... >
<advise>
<status id="0" text="Operasjonen er fullført.
Ingen feil ble funnet." type="informational"/>
<info>
<message id="7401" text="Sammendragstabellen DB2INFO.MQT0000000041T01
anbefales.
Den forventes å ha 100 rader, tabellstørrelse 55 kB og indeksstørrelse 5 kB."/>
<message id="7401" text="Sammendragstabellen DB2INFO.MQT0000000041T02
anbefales.
```

```

Den forventes å ha 8 rader, tabellstørrelse 4 kB og indeksstørrelse 1 kB./>
<message id="7406" text="Dimensjonen PART har ingen hierarkier
som kan optimaliseres av optimaliseringsrådgiveren.
Anbefalingene vil ikke optimalisere for noen attributter
fra denne dimensjonen."/>
</info>
<recommendation diskpace="65108"/>
<sql>
<![CDATA[
-- *****
-- * Skript for å opprette/fornye sammendragstabeller.
-- *
-- * Kubemodellskjema:  MDSAMPLE
-- * Kubemodellnavn:    SalesModel
-- * Lagerplassgrense: 75000
-- * Tidsgrense:       300
-- * Avlesing:         Ja
-- * Fornyingsstype:   Forny umiddelbart
-- * Tabellplassnavn:  TS_MQTTABLE
-- * Indeksplassnavn:  TS_MQTINDEX
-- *****

DROP TABLE DB2INFO.MQT000000041T01;

DROP TABLE DB2INFO.MQT000000041T02;

UPDATE COMMAND OPTIONS USING c OFF;

CREATE SUMMARY TABLE DB2INFO.MQT000000041T01 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 model 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[
-- *****
-- * Skript for å opprette/fornye sammendragstabeller.
-- *
-- * Kubemodellskjema:  MDSAMPLE
-- * Kubemodellnavn:    SalesModel
-- * Lagerplassgrense: 75000
-- * Tidsgrense:       300
-- * Avlesing:         Ja
-- * Fornyingsstype:   Forny umiddelbart
-- * Tabellplassnavn:  TS_MQTTABLE
-- * Indekspllassnavn: TS_MQTINDEX
-- *****

UPDATE COMMAND OPTIONS USING c OFF;

REFRESH TABLE DB2INFO.MQT0000000041T01;
REFRESH TABLE DB2INFO.MQT0000000041T02;
COMMIT;

]]>
</refreshSql>
</advise>
</olap:response>

```

Endringsoperasjoner: Alter, Create, Drop, Import og Rename

DB2 Cube Views-APIen inkluderer fem endringsoperasjoner:

- Alter
- Create
- Drop
- Importer
- Rename

Når den lagrede prosedyren implementerer endringsoperasjonen, sikrer den også at objektene er fullstendige og at de inneholder gyldige referanser.

Create

Create-operasjonen oppretter metadataobjekter. Den godtar en eller flere metadataobjektoperander og oppretter disse objektene i DB2 Cube Views i samme rekkefølge som de sendes til operasjonen. Sekvenser med objekter som sendes til denne operasjonen, kan inkludere objekter av forskjellige typer.

Objekter som sendes til denne operasjonen, kan referere til andre objekter. Hvis det finnes referanser mellom objekter, må de gjenspeiles av rekkefølgen til objektene. Hvis for eksempel objektet `MittObjekt` refererer til objektet `DittObjekt`, må `DittObjekt` sendes til operasjonen før `MittObjekt`. Du finner flere opplysninger om hvordan metadataobjekter kan referere til hverandre, i “Metadataobjektformat” på side 135.

Create-operasjonen validerer hvert objekt. Det blir returnert feil hvis objektet du oppretter, allerede finnes, eller hvis et objekt det blir referert til av objektet du oppretter, ikke finnes fra før. Hvis et inndataobjekt oppgir et skjema som ikke finnes, oppretter operasjonen skjemaet hvis du har tilstrekkelig autorisasjon i databasen.

Tabell 45. Request-parametere i Create

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
objekt	operand	XML-element	Se "Operander i operasjoner" på side 142.	Objekter som kan opprettes.

Tabell 46. Response-parametere i Create

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
status	status-melding	DB2 Cube Views, meldingsstruktur	Se "Meldingers struktur" på side 156.	Melding som angir statusen for Create-operasjonen.

Alter

Alter-operasjonen oppdaterer informasjon om metadataobjekter. Den godtar en eller flere metadataobjektoperander og oppdaterer de tilsvarende objektene i metadataobjekttabellene. Objekter blir oppdatert i den rekkefølgen de sendes til operasjonen. Sekvenser med objekter som sendes til denne operasjonen, kan inkludere objekter av ulike typer.

Objekter som sendes til denne operasjonen, kan referere til andre objekter. Hvis det finnes referanser mellom objekter, må de gjenspeiles av rekkefølgen til objektene. Du finner flere opplysninger om hvordan metadataobjekter kan referere til hverandre, i "Metadataobjektformat" på side 135.

Denne operasjonen kan ikke oppdatere skjemaet eller navnet for et objekt. Du kan endre objektnavn med operasjonen Rename.

Alter-operasjonen validerer hvert objekt. Det blir returnert feil hvis objektet som oppdateres, ikke finnes.

Tabell 47. Request-parametere i Alter

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
objekt	operand	XML-element	Se "Operander i operasjoner" på side 142.	Objekter som kan oppdateres.

Tabell 48. Response-parametere i Alter

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
status	status-melding	DB2 Cube Views, meldingsstruktur	Se "Meldingers struktur" på side 156.	Melding som angir statusen for Alter-operasjonen.

Rename

Rename-operasjonen endrer navn på et enkelt DB2 Cube Views-metadataobjekt som er identifisert av gjeldende skjema og navn. Du kan bare endre navnet på et objekt, ikke skjemaet. Rename-operasjonen kan endre navn på objekter selv om det er referert til dem av andre metadataobjekter.

Tabell 49. Request-parametere i Rename

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
objectType	argument	XML-streng	cubeModel cube cubeFacts cubeDimension cubeHierachy cubeLevel dimension facts hierarchy attributeRelationship level join measure attribute	Typen DB2 Cube Views-metadadataobjekter som blir endret navn. Du kan oppgi en av de gyldige verdiene for denne parameteren.
currentRef	operand	DB2 Cube Views-metadadataobjekt-referanse	Se "Operander i operasjoner" på side 142.	Gjeldende skjema og navn for metadadataobjektet du endrer navn på.
newRef	operand	DB2 Cube Views-metadadataobjekt-referanse	Se "Operander i operasjoner" på side 142.	Nytt skjema og navn for metadadataobjektet du endrer navn på.

Tabell 50. Response-parametere i Rename

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
status	status-melding	DB2 Cube Views, meldingsstruktur	Se "Meldingers struktur" på side 156.	Melding som angir statusen for Rename-operasjonen.

Drop

Drop-operasjonen sletter metadadataobjekter fra DB2 Cube Views. Denne operasjonen sletter ett eller flere metadadataobjekter, avhengig av hva som oppgis for *objectType* og *restriction*. Hvis objektet du sletter, er referert til av et annet metadadataobjekt, blir det returnert en feilmelding.

Tabell 51. Request-parametere i Drop

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
objectType	argument	XML-streng	Se "Parametere for operasjoner" på side 139.	Typen av metadadataobjekter som skal slettes. Du kan oppgi en av de gyldige verdiene for denne parameteren.
restriction [valgfritt]	argument	XML-streng	Se "Parametere for operasjoner" på side 139.	Begrenser omfanget av slettingen av metadadata. Dette tilsvarer predikater i SQL.

Tabell 52. Response-parametere i Drop

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
status	status-melding	DB2 Cube Views, meldingsstruktur	Se "Meldingers struktur" på side 156.	Melding som angir statusen for Drop-operasjonen.

Importer

Import-operasjonen oppretter metadataobjekter eller rapporterer eksistensen av metadataobjekter i metadatakatalogen. Denne operasjonen minner om Create-operasjonen, bortsett fra hvordan den håndterer metadataobjekter som finnes fra før.

Du kan definere valgfrie modi for Import-operasjonen. Disse ulike modiene bestemmer hvilken handling som skal utføres når du forsøker å importere objekter med samme navn som objekter som allerede finnes i katalogen.

Avhengig av hvilken modus du bruker, blir det returnert feil hvis objektet du oppretter allerede finnes, eller hvis et objekt som det blir referert til av objektet du oppretter, ikke finnes allerede. Hvis et inndataobjekt oppgir et skjema som ikke finnes, oppretter Import-operasjonen skjemaet hvis du har tilstrekkelig autorisasjon i databasen.

Import-operasjonen validerer hvert objekt.

Se "Parametere for operasjoner" på side 139 hvis du ønsker en detaljert beskrivelse av de forskjellige operasjonsmodusene.

Tabell 53. Request-parametere i Import

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
mode	argument	XML-streng	create new - ignore collisions create new - replace collisions create new - abort on collision report new - report collisions	Definerer handlingene for nye og eksisterende objekter som blir importert. Se beskrivelsen for <i>mode</i> i "Parametere for operasjoner" på side 139.
objekt	operand	XML-element	Se "Operander i operasjoner" på side 142.	Objekter som importeres.

Tabell 54. Response-parametere i Import

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
status	status-melding	DB2 Cube Views, meldingsstruktur	Se "Meldingers struktur" på side 156.	Melding som angir statusen for Import-operasjonen.

Tabell 54. Response-parametere i Import (fortsettelse)

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
newList	referanse-liste	XML-element	Se beskrivelsen for <i>mode</i> i "Parametere for operasjoner" på side 139.	Liste over navn/skjema-par som refererer til de nye objektene.
collisionList	referanse-liste	XML-element	Se beskrivelsen for <i>mode</i> i "Parametere for operasjoner" på side 139.	Lister over navn/skjema-par som refererer til objekter som kolliderer med andre objekter.

Hvordan APIen håndterer funksjonelle avhengigheter for endringsoperasjoner

Metadata-APIen for DB2 Cube Views administrerer de funksjonelle avhengighetene for et nivåobjekt i henhold til bestemte regler. I tabell 55 vises handlingen som APIen utfører for hver nivåendringsforespørsel.

Tabell 55. API-handling for funksjonelle avhengigheter

Forespørsel	Handling
Opprette et nivå.	APIen oppretter en tilsvarende funksjonell DB2-avhengighet hvis det er mulig. Hvis en funksjonell avhengighet ikke kan opprettes, endrer APIen nivået uten en funksjonell avhengighet, og returnerer en varselmelding.
Slette et nivå	APIen sletter nivået og den tilknyttede funksjonelle avhengigheten. Hvis det oppstår en feil når APIen forsøker å slette nivået eller den funksjonelle avhengigheten, returnerer APIen en feilmelding, og nivåobjektet blir ikke slettet.
Endre et nivå (uten en funksjonell avhengighet)	Hvis du ikke oppretter en funksjonell avhengighet når du endrer et nivå, blir det ikke utført noen handling. Hvis du oppretter en funksjonell avhengighet når du endrer et nivå, oppretter APIen en tilhørende DB2-funksjonell avhengighet hvis det er mulig. Hvis en funksjonell avhengighet ikke kan opprettes, endrer APIen nivået uten en funksjonell avhengighet, og returnerer en varselmelding.
Endre et nivå (med en funksjonell avhengighet)	Hvis nivået har en funksjonell avhengighet og du vil slette den funksjonelle avhengigheten når du endrer nivået, endrer APIen nivået og sletter den tilknyttede DB2-funksjonelle avhengigheten. Hvis det oppstår en feil når APIen forsøker å endre nivået, returnerer APIen en feilmelding, og nivåobjektet blir ikke endret. Hvis du vil endre nivået og bevare den funksjonelle avhengigheten, sletter og deretter gjenoppretter APIen den funksjonelle avhengigheten. Hvis en funksjonell avhengighet ikke kan slettes, returnerer APIen en feilmelding, men fortsetter med å endre nivået og gjenopprette den nye funksjonelle avhengigheten med et nytt navn.
Importere et nivå (Create new: ignore collisions mode)	APIen oppretter en tilsvarende funksjonell DB2-avhengighet hvis det er mulig. Hvis en funksjonell avhengighet ikke kan opprettes, endrer APIen nivået uten en funksjonell avhengighet, og returnerer en varselmelding.

Tabell 55. API-handling for funksjonelle avhengigheter (fortsettelse)

Forespørsel	Handling
Importere et nivå (Create new: replace collisions mode)	Samme som Alter.

Administrasjonsoperasjoner: Validate og Translate

DB2 Cube Views har to administrasjonsoperasjoner: Validate og Translate. Translate-operasjonen bruker du bare når du migrerer fra DB2 Cube Views versjon 8.1 til DB2 Cube Views versjon 8.2. Translate-operasjonen konverterer XML-metadatas for versjon 8.1 til XML-metadatas for versjon 8.2.

Validate

Validate-operasjonen kontrollerer gyldigheten til ett eller flere metadataobjekter. Gyldigheten defineres av hvorvidt et objekt oppfyller objektreglene for DB2 Cube Views. For objektene som er validert av denne operasjonen, oppgir du *objectType*-argumentet og *restriction*-parameterne. Bruk *mode*-parameteren til å oppgi valideringsens omfang.

Validate-operasjonen kontrollerer følgende:

- Fullstendig metadataobjektinformasjon
- Referanseintegritet mellom metadataobjekter
- Eksistensen av refererte relasjonstabeller, utsnitt, tilnavn- og kallenavnkolonner.
- Korrekte SQL-uttrykk lagret i metadataobjekter, for eksempel attributter og mål

Validate-operasjonen stopper når den finner et ugyldig metadataobjekt. En melding som beskriver valideringsfeilen, returneres av operasjonen når den oppdager et brudd på reglene. De andre operasjonene (Create, Alter og Import) validerer også metadataobjekter implisitt. Translate-operasjonen validerer ikke metadataobjektene.

Tabell 56. Request-parametere i Validate

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
objectType	argument	XML-streng	Se "Parametere for operasjoner" på side 139.	Typer av metadataobjekter for DB2 Cube Views som skal valideres. Du kan oppgi en av de gyldige verdiene for denne parameteren.
restriction (valgfritt)	argument [valgfritt]	XML-streng	Se "Parametere for operasjoner" på side 139.	Begrenser omfanget av valideringen av metadata. Dette tilsvarer predikater i SQL.
mode	argument	XML-streng	base cubeModel completeness optimalisering	Definerer omfanget av valideringshandlingene som utføres.

Tabell 57. Response-parametere i Validate

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
status	status-melding	DB2 Cube Views, meldingsstruktur	Se "Meldingers struktur" på side 156.	Meldinger som angir statusen for valideringsoperasjonen.

Tabell 57. Response-parametere i Validate (fortsettelse)

Navn	Type	Datatype	Gyldige verdier	Beskrivelse
info	meldings-liste	Liste over meldingsstrukturer	Se "Meldingers struktur" på side 156.	Liste over meldinger som beskriver advarslene og feilene som er generert av validate-operasjonen.

Translate

Translate-operasjonen kontrollerer at de innkommende XML-metadatatene har riktig syntaks ved å validere XML-skjemaet, men den validerer ikke kolonnereferanser eller andre referanser. Du kan bruke Translate-operasjonen til å konvertere tilfeldige DB2 Cube Views-metadatatene. Translate-operasjonen krever et fullstendig XML-dokument. Referanser til objekter som ikke er i XML-dokumentet, kan skape feil.

Translate-operasjonen tilordner objekter slik det er vist i den følgende tabellen. Hvert objekt i versjon 8.1 er tilordnet til et identisk objekt i versjon 8.2, med mindre annet er oppgitt.

Tabell 58. Hvordan versjon 8.1-objekter er tilordnet til versjon 8.2-objekter

Versjon 8.1-objekter	Versjon 8.2-objekter
Attributt	<ul style="list-style-type: none"> For attributter kan man oppgi som ny egenskap om attributtene <i>kan ha nullverdier</i>, For eksisterende attributter er egenskapen for <i>kan ha nullverdier</i> ukjent.
Kombinering	Ingen endringer.
Attributtforhold	<ul style="list-style-type: none"> Ikke lenger referert til av hierarkier og kubehierarkier. Alle attributtforhold blir løse. (Ingen andre objekter refererer til disse attributtforholdene.) <p>Når du eksporterer en kubemodell eller kube fra OLAP-senteret, ser du ikke lenger attributtforhold i den eksporterte XML-filen. Fra OLAP-senteret må du eksportere alle metadatatene for å eksportere attributtforhold.</p>
Hierarkier + attributter + attributtforhold	<ul style="list-style-type: none"> Det blir opprettet nøyaktig ett nivåobjekt for hvert attributt som det blir referert til i et versjon 8.1-hierarki. Nivåobjektet bruker samme navn (skjema, navn osv.) som versjon 8.1-kildeattributtet. Nivåets standardattributt vil være kildeattributtet. Nivåets nøkkel består av kildeattributtet pluss overordnede attributter over kildeattributtet, basert på ett av versjon 8.1-hierarkiene. Hvis det er referert til kildeattributtet i mer enn ett versjon 8.1-hierarki, velges det hierarkiet med færrest overordnede (hvis det er like mange, velges et tilfeldig), og de overordnede attributtene blir tilføyd ovenfra og ned til kilden som nøkkelattributtene i versjon 8.2-nivået. alle versjon 8.1-attributtforhold som det er referert til av hierarkier der venstreattributtet er lik høyreattributtet, blir brukt til å opprette de beslektede attributtene i nivået. For hvert samsvarende (på venstre side) attributtforhold blir høyreattributtet tilføyd til nivået som et beslektet attributt. Når listen over beslektede attributter blir opprettet, fjernes duplikater.

Tabell 58. Hvordan versjon 8.1-objekter er tilordnet til versjon 8.2-objekter (fortsettelse)

Versjon 8.1-objekter	Versjon 8.2-objekter
Kubehierarkier + attributter + attributtforhold	<ul style="list-style-type: none"> • Det blir opprettet ett entydig kubnivåobjekt for hvert attributt det blir referert til i hvert versjon 8.1-kubehierarki. • Dette kubnivået får navn etter attributtets navn og kubens navn. Hvis kubens navn er Salg, får kubnivået navnet Region (Salg). • Hver kubnivå refererer til sitt tilsvarende nivåobjekt. Region (Salg) peker for eksempel til nivået Region. • Kubnivåets beslektede attributter er basert på attributtforholdsobjektene i det gitte kubehierarkiet. For hvert attributtforhold der venstreattributtet er likt kildeattributtet (i det overordnede nivået), blir høyreattributtet tilføyd til kubnivået som et beslektet attributt.
Hierarki	Hver versjon 8.1-hierarki tilordnes til et versjon 8.2-hierarki, med et sortert sett med nivåer (ingen attributter og ingen attributtforhold).
Kubehierarki	Hvert versjon 8.1-kubehierarki tilordnes til et versjon 8.2-kubehierarki med et sortert sett med kubnivåer (ingen attributter og ingen attributtforhold).
Dimensjon	Tilhørende nivåer blir tilføyd.
Kubedimensjon	Ingen endringer.
Mål	På samme måte som for attributter kan man oppgi egenskapen om mål <i>kan ha nullverdier</i> . For eksisterende mål er egenskapen for <i>kan ha nullverdier</i> ukjent.
Fakta	Ingen endringer.
Kubefakta	Ingen endringer.
Kubemodell	Ingen endringer.
Kube	Ingen endringer under konvertering, men i versjon 8.2 kan en kube inkludere bruksinformasjon.

Meldinger, logging og sporing

Meldingers struktur

DB2 Cube Views-APIen returnerer informasjonsmeldinger, advarsler og feilmeldinger som har en bestemt struktur.

Tabellen nedenfor beskriver komponentene i en DB2 Cube Views-melding.

Tabell 59. Meldingskomponent

Komponent	Beskrivelse
id	En entydig numerisk ID for meldingen.
type	En melding kan være en av tre typer: <ul style="list-style-type: none"> • informasjon • warning • error
text	Tegnstrengen som inneholder teksten i meldingen.

Tabell 59. Meldingskomponent (fortsettelse)

Komponent	Beskrivelse
symboler	<p>Verdiene som erstattes i tekststrengen for meldingen. En melding kan ha et hvilket som helst antall av tokens-symboler. Disse XML-elementene kan brukes som symboler i en melding:</p> <ul style="list-style-type: none"> • attributeRef • joinRef • attributeRelationshipRef • levelRef • cubeLevelRef • hierarchyRef • cubeHierarchyRef • dimensionRef • cubeDimensionRef • measureRef • factsRef • cubeFactsRef • cubeModelRef • cubeRef • column • text

Dette er et eksempel på en melding uten symboler:

```
<status id="0" text="0operasjonen er fullført." type="informational"/>
```

Dette er et eksempel på en melding med symboler:

```
<status id="6331" text="Venstreattributtet for
'&MDOBJ_ID_ATTRIBUTERELATIONSHIP.MDSAMPLE.State_PopGroup'
er ikke en del av hierarkiet
'&MDOBJ_HIERARCHY.MDSAMPLE.RegionState';."
type="error">
<tokens>
<attributeRelationshipRef name="State_PopGroup" schema="MDSAMPLE"/>
<text value="MDOBJ_HIERARCHY.MDSAMPLE.RegionState"/>
</tokens>
</status><
```

Kjøretidssporing for APIen for DB2 Cube Views

APIen støtter tre prioriteringer for sporing. Ved hjelp av konfigurasjonsfilen kan en administrator definere sporingsnivået som skal logges. Som standard er kjøretidssporing slått av. Standard navn på sporingsfilen er db2mdtrace.log.

Vanligvis er det ikke nødvendig å kjøre sporing. Sporing kan være nødvendig hvis det oppstår en feil i APIen og IBM Kundeservice ber deg om en sporingsfil.

Tabellen nedenfor beskriver de ulike sporingsnivåene.

Tabell 60. Sporingsnivåer

Nivå	Beskrivelse	Eksempler
Ingen	<ul style="list-style-type: none"> • Sporing er deaktivert 	Gjelder ikke

Tabell 60. Sporingsnivåer (fortsettelse)

Nivå	Beskrivelse	Eksempler
Høy	<ul style="list-style-type: none"> • Sporer bare eksterne og interne inngangs- og utgangspunkter for APIen • Sporer flyten mellom komponenter • Kan omfatte funksjonsargumenter 	<ul style="list-style-type: none"> • Begynner og avslutter analysering • Begynner og avslutter Create, Describe, Drop osv.
Middels	<ul style="list-style-type: none"> • Sporer flyten av kontroll mellom komplekse funksjoner i den eksterne og interne APIen • Sporer flyten mellom komponenter • Omfatter sporingspunkter på høyt nivå 	<ul style="list-style-type: none"> • Viser kall til komplekse funksjoner fra create-operasjonen
Lav	<ul style="list-style-type: none"> • Sporer enkle eller sammensatt funksjonen innenfor intern API • Omfatter sporingspunkter på høyt og middels nivå • Bruk dette nivået for de fleste sporingspunktene 	<ul style="list-style-type: none"> • Viser kall til get- eller set-metodene for metadataobjektene

Når sporing er aktivert med et annet nivå enn Ingen, kan feil som oppstår i APIen, registreres både i feilloggen og i sporingsloggen, avhengig av loggenes innstillinger for nivå og alvorsgrad.

Loggfiler for DB2 Cube Views-APIen

API-loggfilene genereres på DB2-forekomstnivå. Navnet på feilloggfilen er `db2mderror.log`, og navnet på sporingsloggfilen er `db2mdtrace.log`.

For en gitt DB2-forekomst som kjører DB2 Cube Views-APIen, blir loggfilene for APIen generert i DB2-katalogen for feilsøkningsdata, DB2DIAG. Denne DB2-feilsøkningskatalogen finner du vanligvis i følgende kataloger:

På Windows

Banen for DB2-forekomsten, for eksempel `c:\sql11ib\`

På AIX

DB2-forekomstbane/db2dump, for eksempel `~minforekomst/sql11ib/db2dump`

Du kan endre standard DB2DIAG-bane med innstillingen `DB2DIAG db2 dbm cfg`.

Funksjonen `db2idrop` rydder opp i loggfilene som er knyttet til en DB2-forekomst. Hvis du ikke bruker standardverdien for DB2DIAG, kan ikke `db2idrop` rydde opp i loggfilene for DB2 Cube Views-APIen. Loggfilene som `db2idrop` ikke kan rydde opp i, må du rydde opp i manuelt. Feil som oppstår når DB2 Cube Views-APIen laster inn konfigurasjonsfilen, logges i filen `db2mdapi.log`. Filen `db2mdapi.log` ligger i DB2DIAG-banen som de andre API-loggene.

Feillogging

APIen skiller mellom tre alvorsgrader for feil. Standardinnstillingen for alvorsgrad er medium, og standard feilloggfil er `mderror.log`. Når det oppstår en feil under lesing av konfigurasjonsfilen, blir denne feilen logget i filen `db2mdapi.log`.

Tabellen nedenfor beskriver alvorsgradene for feil.

Tabell 61. Alvorsgrader for feil

Alvorsgrad	Beskrivelse	Eksempler
Ingen	<ul style="list-style-type: none"> • Overse alle feil og advarsler 	Gjelder ikke
Høy (De fleste feil har denne alvorsgraden)	<ul style="list-style-type: none"> • Registrerer bare kritiske, uopprettelige feil • Dumper en anropskø til loggen 	<ul style="list-style-type: none"> • Intern kodefeil
Middels	<ul style="list-style-type: none"> • Registrerer feil som kan gjenoprettes av brukeren • Logger også feil med høy alvorsgrad • Dumper en anropskø til loggen 	<ul style="list-style-type: none"> • Sluttbrukerfeil, for eksempel forsøk på å opprette et duplikat objekt • Metadatavalideringsfeil • For lite minne. Du kan øke minnet eller redusere bruken
Lav	<ul style="list-style-type: none"> • Registrerer advarselssituasjoner • Logger også feil med høy og middels alvorsgrad • Feil med lav alvorsgrad dumper ikke en anropskø 	<ul style="list-style-type: none"> • Advarsel om intern feil • Informasjonsmeldinger

Når APIen er konfigurert for high eller medium feillogging og det oppstår en feil med high eller medium alvorsgrad, genererer APIen en anropskø som begynner fra punktet der feilen oppstår i APIen. Denne anropskøen er lik en sporing med middels nivå, men dataene blir sendt til feilloggen i stedet for til sporingsloggen.

Scenarier for logging og sporing

De følgende scenariene for logging og sporing viser hvordan feil kan bli registrert.

Scenario 1 (sporingsnivå: medium; alvorsgrad for feil: high): Når det oppstår en feil med høy alvorsgrad, registreres den både i feilloggen og sporingsloggen.

```
<log>
  <trace level="medium" logFile="db2mdtrace.log" bufferSize="0" />
  <error level="high" logFile="db2mderror.log" bufferSize="0" />
</log>
```

Scenario 2 (sporingsnivå: medium; alvorsgrad for feil: low): Når det oppstår en feil med lav alvorsgrad, registreres den bare i feilloggen fordi sporingsloggen bare tillater poster med middels eller høyt nivå.

```
<log>
  <trace level="medium" logFile="db2mdtrace.log" bufferSize="0" />
  <error level="low" logFile="db2mderror.log" bufferSize="0" />
</log>
```

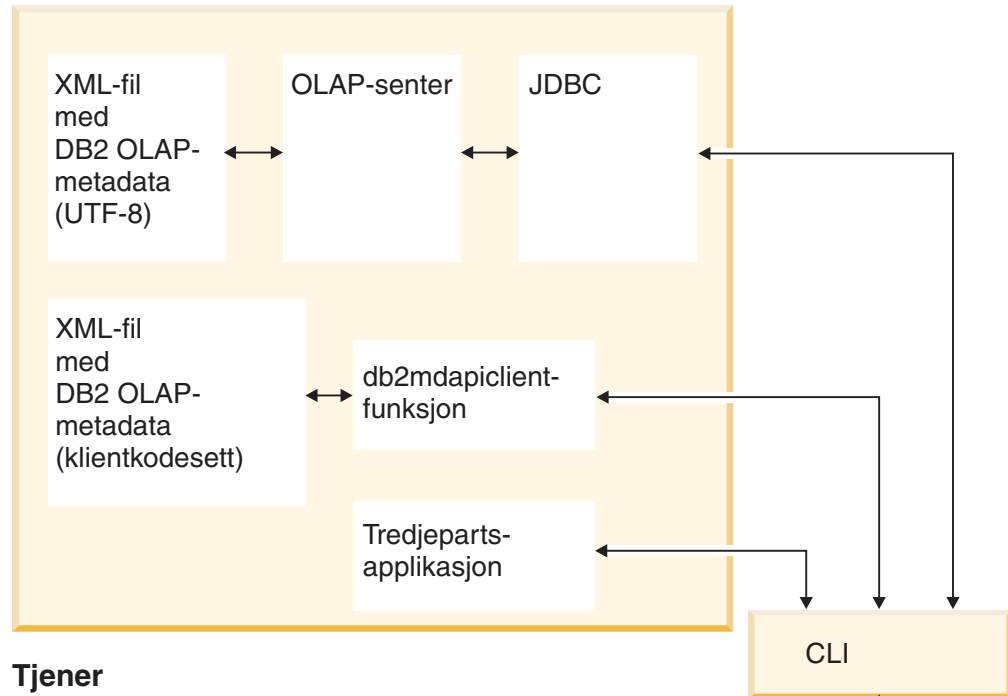
Feil som gjelder manglende systemvariabler eller feil ved tilgang til loggfiler, returneres via SQLSTATE i den lagrede prosedyrens kall til databaseklientapplikasjonen. Når det oppstår en feil under behandling av konfigurasjonsfilen, blir denne feilen logget i filen db2mdapi.log. Hvis det oppstår en feil under åpning av en av de brukerspesifiserte loggfilerne, blir det ikke registrert noen feil.

Støtte for kodesett

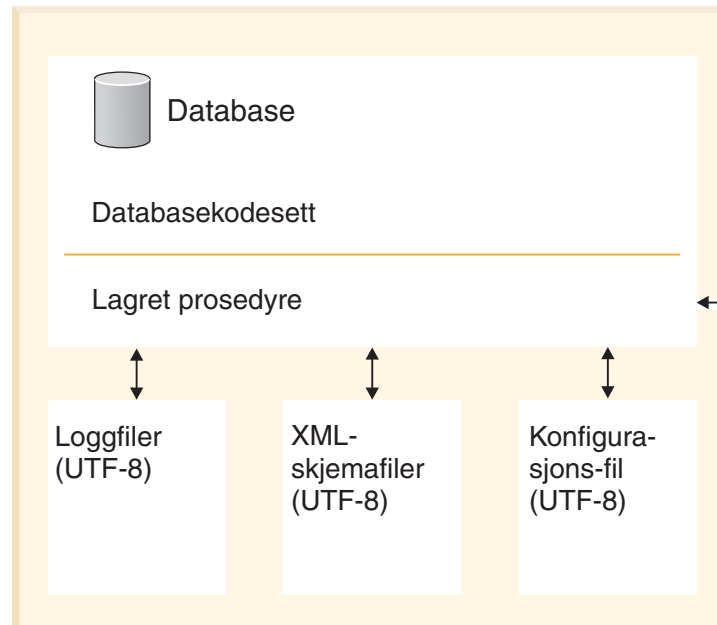
DB2 Cube Views bruker to kodesett, DB2-klientkodesettet (applikasjonskodesettet) og DB2-databasekodesettet. Boken *DB2 Administration Guide: Planning, "Supported territory codes and code pages"*, inneholder informasjon om hvordan du bestemmer DB2-klientkodesettet. Den lagrede prosedyren for DB2 Cube Views-APIen kjører i DB2-databasekodesettet. DB2-databasekodesettet fastsettes når databasen opprettes. DB2-klientkodesettet og DB2-databasekodesettet kan være forskjellige. CLI-grensesnittet konverterer den lagrede prosedyrens CLOB-parametere (CLOB=Character Large Object) fra klientkodesett til databasekodesett for den lagrede prosedyren.

Illustrasjonen nedenfor viser hvordan klienten kommuniserer med tjeneren gjennom et CLI-grensesnitt (CLI=Call-level Interface). CLI-grensesnittet konverterer klientkodesett til databasekodesettet.

Klient



Tjener



Figur 41. Hvordan data flyter fra forskjellige klienter som bruker kodesett eller UTF-8-format, gjennom DB2 CLI til databasetjeneren

CLI styrer konverteringen mellom applikasjonskodesett og databasekodesett. Data som er sendt fra DB2 Cube Views-klienten til APIen, regnes som inndata. Data som er sendt fra APIen til DB2 Cube Views-klienten, regnes som utdata. Dataene i inn- og utdataene er kodet i DB2-klientkodesettet.

Komponentene i DB2 Cube Views har følgende kodesettspesifikasjoner.

OLAP-senteret:

- godtar og genererer bare XML-filer for DB2 Cube Views som er kodet i UTF-8

- returnerer en feil hvis en XML-inndatafil for DB2 Cube Views har en annen kodingsspesifikasjon enn UTF-8
- tolker mangelen på en kodingsspesifikasjon i en fil som at filen er kodet i UTF-8
- oppretter, med eksportfunksjonen, XML-filer for DB2 Cube Views med eksplisitt kodingsspesifikasjon UTF-8

db2mdapiclient

- tolker XML-inndatafiler for DB2 Cube Views som kodet i UTF-8, med mindre alternativet -l er oppgitt, og overser derfor eksplisitt kodingsspesifikasjon i filene
- genererer XML-filer for DB2 Cube Views kodet i UTF-8 med mindre alternativet -l er oppgitt, og inkluderer ikke eksplisitt kodingsspesifikasjon i disse filene.

APIen for den lagrede prosedyren

- tolker CLOB-parametere som kodet i DB2-klientkodesettet
- overser eksplisitt kodingsspesifikasjon i XML-filer for DB2 Cube Views
- genererer XML-filer for DB2 Cube Views uten eksplisitt kodingsspesifikasjon
- behandler XML-inndatafiler og -utdatafiler ved hjelp av DB2-databasekodesettet
- generere API-loggfiler som er kodet med UTF-8, og dette gjelder også eventuelle innfelte DB2-meldinger. API-loggfiler er ikke i XML.
- oppretter ikke loggfiler som eksplisitt oppgir UTF-8-koding
- koder XML-skjemafilene som brukes av APIen, i UTF-8
- koder XML-konfigurasjonsfilen for APIen i UTF-8

For tredjepartsapplikasjoner gjelder at andre applikasjoner som sender direkte kall til APIen for DB2 Cube Views, må sende og godta som parametere XML-filer som er kodet i DB2-klientkodesettet.

Kapittel 8. Eksempelfiler

Oversikt over db2mdapiclient-funksjonen

Funksjonen db2mdapiclient leveres som eksempel kildekode for å vise hvordan du kan kode en applikasjon for DB2 Cube Views Multidimensional Services. Med denne funksjonen kan du importere, eksportere og optimalisere metadataobjekter.

Merk: Funksjonen som leveres med DB2 Cube Views versjon 8.2 er forbedret, og tilsvarer ikke nøyaktig kildekodeeksempelet i filen `sqllib\samples\olap\client\db2mdapiclient.cpp`.

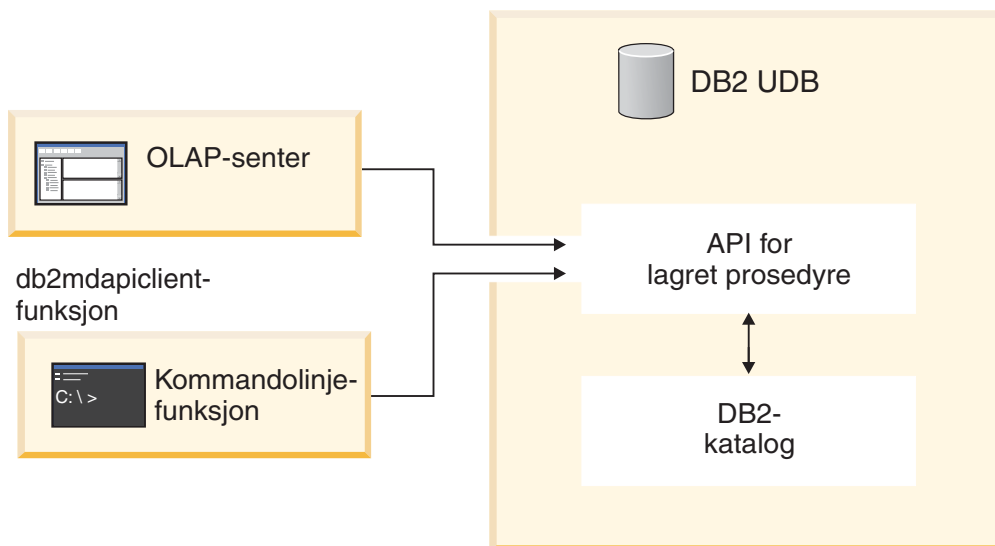
Med funksjonen db2mdapiclient kan du utføre alle operasjonene som støttes av den lagrede prosedyren for DB2 Cube Views, MD_MESSAGE(), som er beskrevet i tabellen nedenfor:

Tabell 62. Operasjoner. Multidimensional Services-operasjoner som funksjonen db2mdapiclient kan utføre

Oppgave	Operasjon
Eksportere metadataobjekter til en fil	DESCRIBE
Importere metadataobjekter fra en fil	CREATE eller IMPORT
Endre eksisterende metadataobjekter	ALTER eller RENAME
Slette eksisterende metadataobjekter	DROP
Kontrollere at de eksisterende metadataobjektene er gyldige	VALIDATE
Optimalisere en kubemodell	ADVISE
Migrere XML-metadata fra versjon 8.1 til versjon 8.2	TRANSLATE

Funksjonen db2mdapiclient bruker filer som inneholder XML-dataene som sendes til og mottas fra den lagrede prosedyren MD_MESSAGE().

Ved import bruker db2mdapiclient vanligvis en XML-fil som er laget av en brofunksjon i DB2 Cube Views, eller som er eksportert fra OLAP-senteret. Ved eksport lager db2mdapiclient en XML-fil som en brofunksjon i DB2 Cube Views kan bruke når den tilføyer metadata i en database eller et OLAP-verktøy. Tegnkodingen som brukes i XML-inndatafiler og -utdatafiler, er viktig. Du finner flere opplysninger om tegnkode i "Støtte for kodesett" på side 160.



Figur 42. Metadataoverføring. Funksjonen db2mdapiclient og OLAP-senteret overfører metadata gjennom Multidimensional Services

Kommandoen db2mdapiclient: manipulere metadataobjekter

Du kan bruke funksjonen db2mdapiclient fra kommandolinjen.

Hvis du vil se en liste overparameterne til db2mdapiclient-kommandoen, skriver du db2mdapiclient på en kommandolinje. Dette er syntaksen for db2mdapiclient-kommandoen:

```
db2mdapiclient -d db-navn [-u bruker] [-p passord] -i request.xml -o respons.xml
  [-m inputmetadata.xml] [-n outputmetadata.xml] [-a parameter_buffer_size]
  [-b metadatabufferstørrelse] [-v] [-l] [-h]
```

-d db-navn angir navnet på databasen.

-u bruker angir bruker-IDen som skal tilkobles den angitte databasen.

-p passord angir passordet for tilkobling til den angitte databasen.

-i request.xml angir den obligatoriske inndatafilen som inneholder operasjonen som skal utføres.

-o response.xml angir den obligatoriske utdatafilen som inneholder XML-respons fra den lagrede prosedyren MD_MESSAGE(). Det tredje argumentet i den lagrede prosedyren MD_MESSAGE() returnerer denne XML-responsfilen.

-m inndatametadata.xml er inndatafilen som inneholder DB2 Cube Views metadataobjekt-XML. Dette alternativet er obligatorisk for opprettings- eller importoperasjoner.

-n utdatametadata.xml er den valgfrie utdatafilen som inneholder XML-responsmetadata, hvis det er aktuelt, fra det andre argumentet av den lagrede prosedyren MD_MESSAGE().

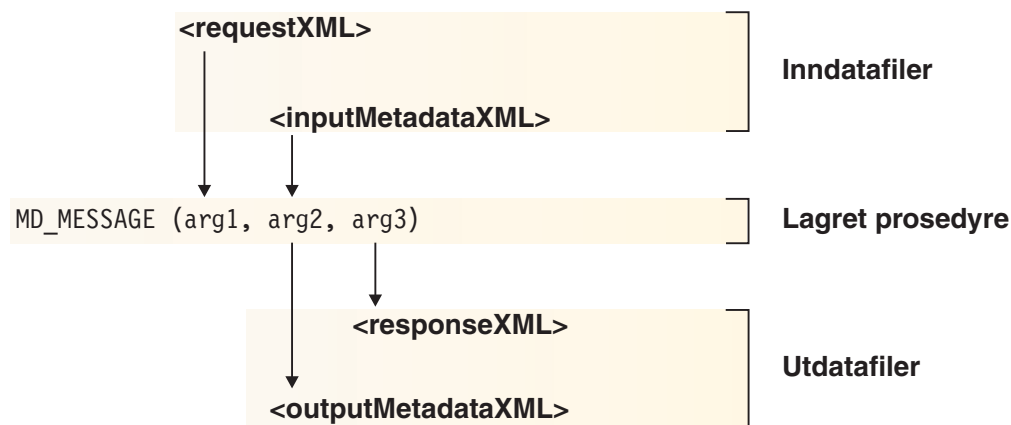
-a parameterbufferstørrelse er bufferstørrelsen for parameterne. Standardverdien er 1048576 byte.

-b metadatabufferstørrelse er bufferstørrelsen for metadataobjektinformasjonen. Standardverdien er 1048576 byte.

-v angir at du vil skrive ut ekstra informasjon mens kommandoen blir behandlet.

- l deaktiverer UTF-8-støtte, og angir at inndata- og utdatafilene er i det lokale kodesettet.
- h viser informasjon om bruken.

Diagrammet nedenfor viser hvordan den lagrede prosedyren MD_MESSAGE() er knyttet til de to inndatafilene og de to utdatafilene:



Figur 43. Hvordan den lagrede prosedyrene behandler de to inndata- og utdatafilene fra db2mdapiclient-funksjonen

Hvis du for eksempel skal importere metadataobjekter fra DB2 Cube Views for CVSAMPLE-databasen for Windows, bytter du til katalogen ..\SQLLIB\samples\olap\xml\input og oppgir denne kommandoen:

```
db2mdapiclient -d CVSAMPLE -u db2admin -p mittpassord -i create.xml
-o myresponse.xml -m ..\..\CVSAMPLE\CVSampleMetadata.xml.xml
```

Du finner en beskrivelse av eksempelfilene i "API-eksempelfiler" på side 166. Du finner flere opplysninger om Multidimensional Services-operasjoner i "DB2 Cube Views, API-oversikt" på side 129.

Eksempeldatabasefiler

Alle filene nedenfor, som er knyttet til databasen CVSAMPLE, finnes i katalogen \SQLLIB\samples\olap\cvsample\.

CVSampleMetadata.xml

En XML-fil som inneholder CVSAMPLE-metadataene. Med denne filen kan du importere CVSAMPLE-metadataene med OLAP-senteret og db2mdapiclient-funksjonen.

CVSampleTables.sql

Et SQL-skript du bruker når du skal legge inn data i CVSAMPLE-tabellene.

FAMILY.txt, LINE.txt, LOCATION.txt, PRODUCT.txt, SALESFACT.txt, STORE.txt, TIME.txt

Et sett med tekstfiler som inneholder CVSAMPLE-tabelldataene.

CVSampleExplain.sql

Et SQL-skript som du kan bruke når du skal avgjøre om DB2 omdirigerer en spørring til en sammendragstabell.

Katalogen \SQLLIB\samples\olap\xml\input inneholder også filer som er relatert til CVSAMPLE-databasen.

Create.xml

En XML-fil med CREATE-operasjonen. Med denne filen laster du inn eksempelet med db2mdapiclient-funksjonen.

API-eksempelfiler

API-eksempelfiler for CVSAMPLE-databasen leveres med DB2 Cube Views. Ved hjelp av eksempefilene kan du utføre eksempelscenarier med db2mdapiclient-funksjonen. Funksjonen db2mdapiclient er en tynn innpakker til grensesnittet for den lagrede prosedyren for Multidimensional Services. Funksjonen leveres som eksempelkode for å vise hvordan du kan kode en applikasjon for APIen. Kildekoden finnes i \SQLLIB\samples\olap\client\db2mdapiclient.cpp.

Du sender innholdet i eksempefilene som er oppgitt for hvert scenario, som parametere til den lagrede prosedyren MD_MESSAGE(). Noen ganger blir metadataparameteren til den lagrede prosedyren oversett som inndata, eller ikke returnert som utdata, noe som i de følgende scenariene er angitt som <tom>. Når du ikke trenger inndatametadadata, oppgir du ikke parameteren -m i db2mdapiclient-funksjonen. Når du ikke trenger metadatautdata, oppgir du ikke parameteren -n i db2mdapiclient-funksjonen.

Du finner flere opplysninger om hvordan du bruker den lagrede prosedyren MD_MESSAGE() sammen med db2mdapiclient-funksjonen, i "Oversikt over db2mdapiclient-funksjonen" på side 163. Du finner flere opplysninger om hvordan du bruker den lagrede prosedyren MD_MESSAGE() alene, i "Lagret prosedyre for DB2 Cube Views" på side 131.

Alle eksempefilene ligger i katalogen \SQLLIB\samples\olap\xml\. Du kan bruke eksempefilene til å utføre disse eksempelscenariene:

DROP

Med disse filene kan du slette alle metadataobjektene i metadatakatalogen. Dette eksempelet forutsetter at metadatakatalogen ikke er tom. Hvis metadatakatalogen er tom, får du en varselmelding om at det ikke ble funnet noen objekter for operasjonen.

Filer som inneholder inndataparametere

- **Request:** input\Drop.xml
- **Metadata:** <tom>
Ikke oppgi parameteren -m i db2mdapiclient-funksjonen.

Filer som inneholder utdataparametere

- **Response:** output\Drop_response.xml
- **Metadata:** <tom>
Ikke oppgi parameteren -n i db2mdapiclient-funksjonen.

CREATE

Med disse filene kan du opprette metadataobjekter i metadatakatalogen. Dette eksempelet forutsetter at metadatakatalogen er tom.

Filer som inneholder inndataparametere

- **Request:** input\Create.xml
- **Metadata:** input\CVSampleTestMetadata.xml

Filer som inneholder utdataparametere

- **Response:** output\Create_response.xml

- **Metadata:** <tom>
Ikke oppgi parameteren -n i db2mdapiclient-funksjonen.

DESCRIBE

Bruk disse filene til å beskrive alle metadataobjektene i metadatakatalogen.

Filer som inneholder inndataparametere

- **Request:** input\Describe.xml
- **Metadata:** <tom>
Ikke oppgi parameteren -m i db2mdapiclient-funksjonen.

Filer som inneholder utdataparametere

- **Response:** output\Describe_response.xml
- **Metadata:** <metadata-XML-fil>

DESCRIBE (begrenset)

Med disse filene kan du rekursivt beskrive kuben CVSample.Daily Sales. Dette eksempelet forutsetter at du tidligere har importert metadataene CVSampleMetadata.xml for CVSAMPLE.

Filer som inneholder inndataparametere

- **Request:** input\CVSampleDescribe_restricted.xml
- **Metadata:** <tom>
Ikke oppgi parameteren -m i db2mdapiclient-funksjonen.

Filer som inneholder utdataparametere

- **Response:** output\DescribeRestricted_response.xml
- **Metadata:** <metadata-XML-fil>

ALTER

Med disse filene kan du endre metadataobjekter i metadatakatalogen. Dette eksempelet forutsetter at du tidligere har importert metadataene CVSampleMetadata.xml for CVSAMPLE.

Filer som inneholder inndataparametere

- **Request:** input\Alter.xml
- **Metadata:** input\CVSampleAlter.xml

Filer som inneholder utdataparametere

- **Response:** output\Alter_response.xml
- **Metadata:** <tom>
Ikke oppgi parameteren -n i db2mdapiclient-funksjonen.

RENAME

Med disse filene kan du endre navnet på kubemodellen CVSAMPLE.Sales Model. Navnet på kubemodellen blir endret til CVSAMPLE.Sales Model (2004). Dette eksempelet forutsetter at du tidligere har importert metadataene CVSampleMetadata.xml for CVSAMPLE.

Filer som inneholder inndataparametere

- **Request:** input\CVSampleRename.xml
- **Metadata:** <tom>
Ikke oppgi parameteren -m i db2mdapiclient-funksjonen.

Filer som inneholder utdataparametere

- **Response:** output\Rename_response.xml
- **Metadata:** <tom>
Ikke oppgi parameteren -n i db2mdapiclient-funksjonen.

VALIDATE

Med disse filene kan du validere alle metadataobjektene i metadatakatalogen ved hjelp av en optimaliseringsvalideringsmodus.

Filer som inneholder inndataparametere

- **Request:** input\Validate.xml
- **Metadata:** <tom>
Ikke oppgi parameteren -m i db2mdapiclient-funksjonen.

Filer som inneholder utdataparametere

- **Response:** output\Validate_response.xml
- **Metadata:** <tom>
Ikke oppgi parameteren -n i db2mdapiclient-funksjonen.

VALIDATE (begrenset)

Med disse filene kan du validere kubens CVSAMPLE.Daily Sales ved hjelp av en optimaliseringsvalideringsmodus. Dette eksempelet forutsetter at du tidligere har importert metadataene CVSampleMetadata.xml for CVSAMPLE.

Filer som inneholder inndataparametere

- **Request:** input\CVSampleValidate_restricted.xml
- **Metadata:** <tom>
Ikke oppgi parameteren -m i db2mdapiclient-funksjonen.

Filer som inneholder utdataparametere

- **Response:** output\ValidateRestricted_response.xml
- **Metadata:** <tom>
Ikke oppgi parameteren -n i db2mdapiclient-funksjonen.

IMPORT med modusen *create new - ignore collisions*

Med disse filene kan du importere metadataobjekter i metadatakatalogen med importmodusen *create new - ignore collisions*. Dette eksempelet forutsetter at metadatakatalogen er tom.

Filer som inneholder inndataparametere

- **Request:** input\Import_model.xml
- **Metadata:** CVSampleTestMetadata.xml

Filer som inneholder utdataparametere

- **Response:** output\Import_model_response.xml
- **Metadata:** <tom>
Ikke oppgi parameteren -n i db2mdapiclient-funksjonen.

IMPORT med modusen *create new - replace collisions*

Med disse filene kan du importere metadataobjekter i metadatakatalogen med importmodusen *create new - replace collisions*. Dette eksempelet forutsetter at du har fullført IMPORT-scenariet med importmodusen *create new - ignore collisions*.

Filer som inneholder inndataparametere

- **Request:** input\Import_mode2.xml
- **Metadata:** CVSsampleTestMetadata.xml

Filer som inneholder utdataparametere

- **Response:** output\Import_mode2_response.xml
- **Metadata:** <tom>

Ikke oppgi parameteren -n i db2mdapiclient-funksjonen.

IMPORT med modusen *create new - abort on collision*

Med disse filene kan du importere metadataobjekter i metadata katalogen med importmodusen *create new - abort on collision*. Dette eksempelet forutsetter at du har fullført IMPORT-scenariet med importmodusen *create new - replace collisions*.

Filer som inneholder inndataparametere

- **Request:** input\Import_mode3.xml
- **Metadata:** CVSsampleTestMetadata.xml

Filer som inneholder utdataparametere

- **Response:** output\Import_mode3_response.xml
- **Metadata:** <tom>

Ikke oppgi parameteren -n i db2mdapiclient-funksjonen.

IMPORT med modusen *create new - report collisions*

Med disse filene kan du importere metadataobjekter i metadata katalogen med importmodusen *create new - report collisions*. Dette eksempelet forutsetter at du har fullført IMPORT-scenariet med importmodusen *create new - abort on collisions*.

Filer som inneholder inndataparametere

- **Request:** input\Import_mode4.xml
- **Metadata:** CVSsampleTestMetadata.xml

Filer som inneholder utdataparametere

- **Response:** output\Import_mode4_response.xml
- **Metadata:** <tom>

Ikke oppgi parameteren -n i db2mdapiclient-funksjonen.

TRANSLATE

Med disse filene kan du konvertere Cube Views versjon 8.1-metadata til Cube Views versjon 8.2-metadata.

Filer som inneholder inndataparametere

- **Request:** input\Translate.xml
- **Metadata:** CVSsampleTranslateMetadata.xml

Filer som inneholder utdataparametere

- **Response:** output\Translate_response.xml
- **Metadata:** <metadata XML file>

ADVISE

Med disse filene kan du kjøre optimaliseringsrådgiveren for kubemodellen CVSAMPLE.Sales Model. Dette eksempelet forutsetter at du tidligere har importert metadataene CVSsampleMetadata.xml for CVSAMPLE, og at du ikke har kjørt det foregående Rename-scenariet.

Filer som inneholder inndataparametere

- **Request:** input\CVSampleAdvise.xml
- **Metadata:** <tom>

Ikke oppgi parameteren -m i db2mdapiclient-funksjonen.

Filer som inneholder utdataparametere

- **Response:** output\Advise_response.xml
- **Metadata:** <tom>

Ikke oppgi parameteren -n i db2mdapiclient-funksjonen.

Tillegg. Meldinger

Disse meldinger er fra tjeneren, APIen og OLAP-senteret i DB2 Cube Views.

Socket error: Hvis du åpner eller lukker en databasetilkobling flere ganger, kan det oppstå en socket-feil. I sjeldne tilfeller kan det oppstå en socket-feil når du kjører DB2 Cube Views med DB2 Universal Database Enterprise Server Edition, versjon 8.1.2, i et partisjonert miljø på Windows 2000 Advanced Server. Feilen kan oppstå hvis du gjentar disse handlingene mer enn 10 000 ganger raskt innenfor samme Windows-sesjon:

1. Åpner en tilkobling til en DB2-database.
2. Send et kall til den lagrede DB2 Cube Views-prosedyren for å utføre en metadataoperasjon.
3. Lukker databasetilkoblingen.

Løsningen er å starte Windows-arbeidsstasjonen på nytt for å aktivere socket.

SQLSTATE, API og andre tjenermeldinger

API SQL-statuser

01HQ1: Se utdataloggen for XML og tjenerloggene.

Forklaring

Kallet til den lagrede prosedyren ble fullført, men det ble oppdaget feil under utføringen av en av de forespurte metadataoperasjonene.

Gjør slik:

Kontroller om innholdet i utdataparameterne fra den lagrede prosedyren inneholder flere opplysninger. Postene i tjenerloggene kan også inneholde flere opplysninger.

38Q00: Tjenerloggene inneholder flere opplysninger.

Forklaring

Kallet til den lagrede prosedyren mislyktes. Den eller de forespurte metadataoperasjonene ble ikke utført. Det ble ikke returnert noen informasjon fra den lagrede prosedyren gjennom utdataparameterne.

Gjør slik:

Se etter flere opplysninger i tjenerloggene.

38Q01: Installeringsbanen er ukjent.

Forklaring

Kallet til den lagrede prosedyren mislyktes fordi DB2-installeringskatalogen ikke kan fastslås av den lagrede prosedyrens prosess. Den eller de forespurte

metadataoperasjonene ble ikke utført. Det ble ikke returnert noen informasjon fra den lagrede prosedyren gjennom utdataparameterne.

Gjør slik:

Hvis du bruker et Windows-operativsystem, må du kontrollere at systemvariabelen DB2PATH er satt til riktig verdi som standard eller ved en brukerhandling. Start databasesystemet på nytt og send kallet til den lagrede prosedyren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

38Q02: Kan ikke åpne tjenerloggfilen.

Forklaring

Kallet til den lagrede prosedyren mislyktes fordi minst en av loggfilene som brukes av den lagrede prosedyren, ikke kunne åpnes for skriving av den lagrede prosedyrens prosess. Den eller de forespurte metadataoperasjonene ble ikke utført. Det ble ikke returnert noen informasjon fra den lagrede prosedyren gjennom utdataparameterne.

Gjør slik:

Kontroller at loggfilene som er oppgitt i konfigurasjonsfilen for den lagrede prosedyren (for eksempel olap_config.xml) kan opprettes eller åpnes for lesing og skriving i det aktuelle filsystemet. Hvis loggfilene ikke finnes allerede, prøver den lagrede prosedyren å opprette disse filene. På AIX må du kontrollere at den beskyttede databasebruker-IDen kan lese og skrive til loggfilene.

38Q03: Inndataparameteren for metadata mangler.

Forklaring

Kallet til den lagrede prosedyren mislyktes fordi den forespurte metadataoperasjonen krever at metadata sendes som inndata til den lagrede prosedyren, men det ble ikke levert metadata gjennom den inndataparameteren for metadata. Det ble ikke returnert noen informasjon fra den lagrede prosedyren gjennom utdataparameterne.

Gjør slik:

Lever metadataene som er nødvendige for den forespurte metadataoperasjonen, ved hjelp av inndataparameteren for metadata til den lagrede prosedyren neste gang du sender et kall til den lagrede prosedyren.

38Q04: [error_type] FEIL: Responsutdatabufferen er for liten.

Forklaring

Kallet til den lagrede prosedyren mislyktes fordi utdataparameterbufferen for operasjonsrespons er for liten til CLOB-strukturen som returneres. Det ble ikke returnert noen informasjon fra den lagrede prosedyren gjennom utdataparameterne.

Gjør slik:

Katalogiser den lagrede prosedyren på nytt med en større størrelse for response-parameteren.

Common

Koder for vellykket utføring

0: Operasjonen er fullført. Ingen feil ble funnet.:

Forklaring

Den forespurte metadataoperasjonen ble fullført på vellykket måte. Det oppstod ingen feil under utføringen av operasjonen.

Gjør slik:

Denne meldingen er bare til orientering. Du trenger ikke å gjøre noe.

1: Operasjonen er fullført. Det ble returnert tilleggsopplysninger.:

Forklaring

Den forespurte metadataoperasjonen ble fullført. Operasjonen har returnert tilleggsinformasjon som kan beskrive advarsels- eller feilsituasjoner.

Gjør slik:

Se etter den returnerte tilleggsinformasjonen i INFO-elementet.

2: Operasjonen er fullført. Det ble ikke gjort noen endringer i metadataene.:

Forklaring

Den forespurte metadataoperasjonen ble fullført. Operasjonen førte ikke til endringer i metadataene i databasekatalogen.

Gjør slik:

Send forespørselen om metadataoperasjonen på nytt med en annen modus hvis du vil endre metadataene i databasekatalogen.

Vanlige eksterne feil/advarselkoder

100: Kunne ikke tildele minne for *operasjon*. Kontroller at det finnes tilgjengelig minne.:

Forklaring

Den lagrede prosedyren kunne ikke tildele nødvendige minnesegmenter under utføringen av den forespurte metadataoperasjonen.

Gjør slik:

Øk tilgjengelig minne for den lagrede prosedyrens beskyttede prosess.

101: Det oppstod en intern feil under behandlingen av objektet *objektnavn*.:

Forklaring

Det oppstod en uventet intern feil under utføringen av den forespurte metadataoperasjonen.

Gjør slik:

Kontakt IBM hvis du ønsker mer hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

102: Utdatabufferen med størrelsen *bufferstørrelse* er for liten. Endre bufferstørrelsen til minst *størrelse*..**Forklaring**

Utdataparameterbufferen som er tilgjengelig for den lagrede prosedyren, er for liten til CLOB-strukturen som genereres av den lagrede prosedyren.

Gjør slik:

Hvis det er mulig, bør du katalogisere den lagrede prosedyren på nytt og bruke større størrelser for OUT- og INOUT-parameteren. Ellers må du begrense spørringen slik at den lagrede prosedyren returnerer mindre informasjon.

103: Det finnes ikke en gyldig lisens for dette produktet.:**Forklaring**

Du kan ikke utføre noen metadataoperasjoner fordi det ikke finnes noen gyldig lisens for denne installasjonen av produktet.

Gjør slik:

Installer en gyldig produktlisens på systemet, eller kontakt IBM Kundeservice for å kjøpe en ny produktlisens.

104: Det oppstod en intern feil. Disse symbolene ble returnert: *symbol0*, *symbol1*, *symbol2*, *symbol3*..**Forklaring**

Det oppstod en uventet intern feil under utføringen av den forespurte metadataoperasjonen.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

599: Operasjonen ble ikke utført.:**Forklaring**

Det ble funnet en feil før utføringen av operasjonen. Resultatet var at operasjonen ikke ble utført.

Gjør slik:

Kontroller resultatene fra tidligere metadataoperasjoner som ble utført under det samme kallet til den lagrede prosedyren. Postene i tjenerloggene kan også inneholde flere opplysninger. Når du har rettet problemet som førte til at den tidligere operasjonen mislyktes, sender du på nytt et kall til den lagrede prosedyren og ber om de samme metadataoperasjonene.

Vanlige interne feil/advarelskoder

**600: Inndataparameteren *parameternavn* er ugyldig med denne meldingen: *melding*.
Kontroller parameteren og prøv på nytt.:
Forklaring**

En av parameterne som ble sendt som inndata til en metode internt i den lagrede prosedyren, er ugyldig.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

**601: Inndataparameteren *parameternavn* er NULL.:
Forklaring**

En av parameterne som ble sendt som inndata til en metode internt i den lagrede prosedyren, har den ugyldige verdien NULL.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

**602: Parameteren *parameternavn* med verdien *verdi* er ikke i det gyldige verdiområdet *områdeverdi1*, *områdeverdi2*.:
Forklaring**

En av parameterne som ble sendt som inndata til en metode internt i den lagrede prosedyren, har en verdi som ligger utenfor det gyldige verdiområdet.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

**603: Unicode-strengen *streng* er enten feil eller ugyldig. Det kan finnes et minneproblem.:
Forklaring**

En Unicode-streng i den lagrede prosedyren er feil eller ugyldig. Dette kan bety at det finnes et minneproblem på systemet eller i den lagrede prosedyren. Det kan også skyldes at feil versjon av ICU-bibliotekene blir lastet inn av den lagrede prosedyren.

Gjør slik:

Sørg for at det er tilstrekkelig minne tilgjengelig for datavolumet som blir behandlet av den lagrede prosedyren. Kontroller at riktig versjon av ICU-bibliotekene du bruker sammen med den gjeldende versjonen av den lagrede

prosedyren, blir lastet inn. Det er mulig du må kontrollere søkebanen for kjøretidsbiblioteker som er definert i systemvariablene, for å bestemme riktig konfigurasjon.

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

604: Kunne ikke konvertere innholdet i strengen *streng* fra Unicode til det aktive kodesettet *kodesett*..

Forklaring

Konverteringen av et Unicode-strengobjekt til en streng som bruker en annen koding, mislyktes. Dette kan bety at det finnes et minneprobblem på systemet eller i den lagrede prosedyren. Det kan også skyldes et problem med kodekonverteringen på systemet, eller at feil versjon av ICU-bibliotekene ble lastet inn av den lagrede prosedyren.

Gjør slik:

Kontroller at de nødvendige ICU-filene for kodesettkonvertering er installert på databasetjeneren. Sørg for at det er tilstrekkelig minne tilgjengelig for datavolumet som blir behandlet av den lagrede prosedyren. Kontroller at riktig versjon av ICU-bibliotekene du bruker sammen med den gjeldende versjonen av den lagrede prosedyren, blir lastet inn. Det er mulig du må kontrollere søkebanen for kjøretidsbiblioteker som er definert i systemvariablene, for å bestemme riktig konfigurasjon.

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

605: Det tildelte minnet på *minnebuffer* må frigis..

Forklaring

En metode internt i den lagrede prosedyren har returnert en minnebuffer som trenger å frigis av en annen intern metode.

Gjør slik:

En metode internt i den lagrede prosedyren skal friggi den returnerte minnebufferen. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

606: Konvertering av XMLCh til UChar for *UChar* mislyktes..

Forklaring

Konvertering mellom et XMLCh-tegn og et UChar-tegn mislyktes.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

607: Inndatabufferen med størrelsen *størrelse* er for liten. Endre bufferstørrelsen til minst *ny_størrelse*..

Forklaring

En minnebuffer internt i den lagrede prosedyren er for liten til teksten i en nødvendig melding.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

608: typen *lagret_prosedyrenavn* er ikke gyldig i denne konteksten.:

Forklaring

Det ble oppdaget en uventet type under behandlingen av den lagrede prosedyren.

Gjør slik:

Ta kontakt med IBM Kundeservice, og oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

609: Datatypen *datatype* er ikke gyldig i denne konteksten.:

Forklaring

Det ble oppdaget en uventet datatype under behandlingen av den lagrede prosedyren.

Gjør slik:

Ta kontakt med IBM Kundeservice, og oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

MDOBJECT.LIB-feil

1000: Kunne ikke klonе objektet *objektnavn*..

Forklaring

Det oppstod en feil under kloningen av et klasseobjekt internt i den lagrede prosedyren.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

1001: Distribueringen for hierarkiet *hierarkinaavn* kan ikke settes til rekursiv fordi hierarkiet har flere enn to nivåer.:

Forklaring

Hierarkiet har for mange nivåer for rekursiv distribuering.

Gjør slik:

Endre hierarkiet slik at det har to eller færre nivåer før du endrer distribueringen til rekursiv.

1002: Den anropte funksjonen *funksjonsnavn* støttes ikke.:**Forklaring**

En virtuell metode internt i den lagrede prosedyren ble ikke implementert for en av den lagrede prosedyrens klasser.

Gjør slik:

Kontakt IBM hvis du ønsker mer hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

1003: Containeren er uventet tom.:**Forklaring**

En containerstruktur internt i den lagrede prosedyren er uventet tom.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

1004: Objektet *objektnavn* ble ikke funnet i containeren.:**Forklaring**

Et objekt det ble søkt etter i en av den lagrede prosedyrens interne containerstrukturer, mangler uventet.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

1005: Et duplikat av elementet *elementnavn* finnes allerede i containeren.:**Forklaring**

Et objekt som ikke skulle finnes i en av den lagrede prosedyrens interne containerstrukturer, finnes allerede.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

1006: Det oppstod et unntak under en listeoperasjon.:

Forklaring

Det oppstod et uventet unntak under utføringen av en operasjon på en av den lagrede prosedyrens interne listestrukturer.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

1007: Det oppstod en intern feil i containeren med feilkoden *feil og nummer* og meldingen *melding*.

Forklaring

Det oppstod en feil under utføringen av en operasjon på en av den lagrede prosedyrens interne containerstrukturer.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

1008: Kopieringsoperasjonen kopierte ikke alle egenskapene fullstendig. Kopieringsoperasjonen mislyktes for egenskapen *egenskapsnavn* med verdien *verdi*.

Forklaring

Det oppstod en feil under utføringen av en kopieringsoperasjon på en av den lagrede prosedyrens interne objekter. En av egenskapene til det interne objektet ble ikke kopiert.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

1009: Objekttypen *type1* er ikke gyldig. *type2* var forventet.

Forklaring

Det ble oppdaget en uventet objekttype under behandlingen av den lagrede prosedyren.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

1010: Parameteren *parameternavn* har ikke en fullstendig ID.

Forklaring

En av parameterne som ble sendt som inndata til en metode internt i den lagrede prosedyren, er en metadataobjekt-ID som er ufullstendig.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

1011: Objektet *objektnavn* har ikke en fullstendig ID.:**Forklaring**

En metadataobjekt-ID er uventet ufullstendig i den lagrede prosedyren.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

1012: Parameteren *parameternavn* er den samme som objektet.:**Forklaring**

En av parameterne som ble sendt som inndata til en objektmetode internt i den lagrede prosedyren, er et objekt som uventet er likt objektet som eier metoden.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

1013: Det ble funnet en uventet NULL-peker.:**Forklaring**

Det ble oppdaget en uventet NULL-peker under behandlingen av den lagrede prosedyren.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

1014: Containerpekeren nådde slutten av containeren.:**Forklaring**

En peker på en av containerstrukturene internt i den lagrede prosedyren har uventet nådd slutten av containeren.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

1015: Objektet *objektnavn* er ugyldig. Årsak: ID=ID, melding=melding.:

Forklaring

Et metadataobjekt internt i den lagrede prosedyren er ugyldig.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

1016: Objektet *objektnavn* har en liste av typen *listetype* som er uventet tom.:

Forklaring

En containerstruktur internt i den lagrede prosedyren er uventet tom.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

MDOPERATION.LIB-feil

2001: Den genererte spørringen *spørring* inneholder ikke den nødvendige kolonneobjekttypen.:

Forklaring

En SQL-spørring som er generert av den lagrede prosedyren, mangler en nødvendig kolonne.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

MDPARSER.LIB-feil

3001: Analysatoren fant et XML-unntak under operasjon med meldingen *melding*.:

Forklaring

Det oppstod et uventet unntak i den lagrede prosedyren under analysen av XML-filen som ble sendt til den lagrede prosedyren.

Gjør slik:

Kontroller at XML-filen som sendes til den lagrede prosedyren, er riktig utformet og er gyldig for XML-skjemaet som er publisert for dette produktet. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

3002: Det ble funnet et uventet analysatorunntak i operasjon.:

Forklaring

Det oppstod et uventet unntak i den lagrede prosedyren under analysen av XML-filen som ble sendt til den lagrede prosedyren.

Gjør slik:

Kontroller at XML-filen som sendes til den lagrede prosedyren, er riktig utformet og er gyldig for XML-skjemaet som er publisert for dette produktet. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

3003: Analysatoren fant en SAXParseException under operasjon med meldingen melding.:

Forklaring

Det oppstod et uventet unntak i den lagrede prosedyren under analysen av XML-filen som ble sendt til den lagrede prosedyren.

Gjør slik:

Kontroller at XML-filen som sendes til den lagrede prosedyren, er riktig utformet og er gyldig for XML-skjemaet som er publisert for dette produktet. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

3004: Systemet kunne ikke hente analysatorfeilmeldingen for operasjon.:

Forklaring

Det oppstod en uventet feil i den lagrede prosedyren under analysen av XML-filen som ble sendt til den lagrede prosedyren. Det kunne ikke hentes en feilmelding fra XML-analysatoren.

Gjør slik:

Kontroller at XML-filen som sendes til den lagrede prosedyren, er riktig utformet og er gyldig for XML-skjemaet som er publisert for dette produktet. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

3100: Systemet kunne ikke analysere XML for parametertype (linje: linje, tegn:tegn, melding: melding).:

Forklaring

Den lagrede prosedyren kunne ikke analysere XML-inndatafilen. Det er mulig at XML-inndatafilen ikke er riktig utformet, eller den kan være ugyldig for XML-skjemaet som ble publisert for dette produktet.

Gjør slik:

Kontroller at XML-filen som sendes til den lagrede prosedyren, er riktig utformet og er gyldig for XML-skjemaet som er publisert for dette produktet.

3101: Det ble funnet et ukjent metadataobjekt. analysatormelding.:

Forklaring

Det finnes en ukjent type metadataobjekt i XML-filen som ble sendt til den lagrede prosedyren. Disse XML-inndataene kan ikke behandles av den lagrede prosedyren.

Gjør slik:

Kontroller at XML-filen som sendes til den lagrede prosedyren, er riktig utformet og er gyldig for XML-skjemaet som er publisert for dette produktet.

3102: Det ble funnet et ukjent XML-attributt. attributtnavn, attributtverdi.:**Forklaring**

Det finnes en ukjent type XML-attributt i XML-filen som ble sendt til den lagrede prosedyren. Disse XML-inndataene kan ikke behandles av den lagrede prosedyren.

Gjør slik:

Kontroller at XML-filen som sendes til den lagrede prosedyren, er riktig utformet og er gyldig for XML-skjemaet som er publisert for dette produktet.

3103: Styreeren fant en ugyldig spesifiseringsverdi for attributtet med navnet *navn* og verdien *verdi*.:**Forklaring**

Det finnes en ugyldig spesifiseringsverdi i XML-filen som ble sendt til den lagrede prosedyren. Disse XML-inndataene kan ikke behandles av den lagrede prosedyren.

Gjør slik:

Kontroller at XML-filen som sendes til den lagrede prosedyren, er riktig utformet og er gyldig for XML-skjemaet som er publisert for dette produktet.

3500: Data er nødvendig for attributtet eller elementet med navnet *navn*.:**Forklaring**

Den lagrede prosedyren kunne ikke definere verdien for det oppgitte XML-attributtet eller elementet i XML-filen som skal returneres av den lagrede prosedyren.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

3501: Data er nødvendig for attributtene *attributtnavn1* og *attributtnavn2*.:**Forklaring**

Den lagrede prosedyren kunne ikke definere verdien for det oppgitte XML-attributtet eller elementet i XML-filen som skal returneres av den lagrede prosedyren.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

3502: Formateringsfunksjonen fant en ugyldig spesifiseringsverdi for attributtet med navnet *navn* og verdien *verdi*.:

Forklaring

En ugyldig spesifiseringsverdi ble funnet i den lagrede prosedyren under formateringen av XML-filen som skulle returneres av den lagrede prosedyren.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

MDDATABASE.LIB-feil

4000: Databasetilkoblingen mislyktes. Databasenavn *databasenavn*, bruker-ID *bruker-ID*.

Forklaring

Den lagrede prosedyren kunne ikke opprette sin egen tilkobling til databasen.

Gjør slik:

Kontroller at bruker-IDen som den lagrede prosedyren bruker, har nødvendige rettigheter for tilkobling til databasen.

4001: Databasetilkoblingen ble ikke opprettet fordi det allerede finnes en tilkobling.:

Forklaring

Den lagrede prosedyren oppdaget uventet en duplikat intern tilkobling til databasen. Den forespurte metadataoperasjonen kunne ikke utføres.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4002: Databaseoperasjonen mislyktes.:

Forklaring

Det oppstod en feil under utføringen av en SQL-setning som ble sendt fra den lagrede prosedyren til databasen.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4003: Utføringen av CLI-kallet *kallnavn* mislyktes.:

Forklaring

Det oppstod en feil under utføringen av det oppgitte CLI-kallet.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4004: De returnerte dataene er kuttet.:**Forklaring**

Feilsøkingsinformasjon som ble returnert under den mislykkede databaseoperasjonen, var avkuttet.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

**4005: Det ble mottatt en advarsel fra databasen. SQLSTATE=*kode*,
melding=*melding*.:****Forklaring**

Advarselsinformasjon ble returnert av et CLI-kall som ble sendt fra den lagrede prosedyren.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren.

4008: Det ble funnet en ukjent DB2-datatype.:**Forklaring**

Den lagrede prosedyren oppdaget en ukjent datatype under utføringen av en databaseforespørsel.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4009: Det ble ikke generert noe gyldig lagringspunktnavn.:**Forklaring**

Den lagrede prosedyren kunne ikke generere et gyldig lagringspunktnavn for databasetransaksjonen. Den lagrede prosedyren danner lagringspunktnavnet ved hjelp av sin databaseapplikasjons-ID.

Gjør slik:

Send kallet til den lagrede prosedyren på nytt. Hvis du sender kallet på nytt, kan det bli generert en ny databaseapplikasjons-ID for den lagrede prosedyren, og den kan gjøre det mulig å generere et gyldig lagringspunktnavn. Ta kontakt med IBM

Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4010: Forsøket på å definere et DB2-lagringspunkt mislyktes.:

Forklaring

Den lagrede prosedyren kunne ikke definere et lagringspunkt for databasetransaksjonen. Den gjeldende transaksjonen kan allerede inneholde et lagringspunkt med det samme navnet som denne forekomsten av den lagrede prosedyren bruker.

Gjør slik:

Hvis det er mulig, bør du frigi lagringspunktene for den gjeldende transaksjonen før du sender kallet til den lagrede prosedyren på nytt. Du kan også sende kallet til den lagrede prosedyren på nytt med en ny transaksjon.

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4011: Et lagringspunkt var ikke definert før dette utføringspunktet.:

Forklaring

Et transaksjonslagringspunkt mangler uventet på et punkt i den lagrede prosedyren. Det er mulig at det manglende lagringspunktet ikke ble definert av den lagrede prosedyren, eller at lagringspunktet ble frigitt gjennom databasehandlinger som ble utført utenfor den lagrede prosedyren.

Gjør slik:

Send kallet til den lagrede prosedyren på nytt. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4012: Det fantes en ugyldig lagringspunktstreng.:

Forklaring

Lagringspunktnavnet for databasetransaksjonen ble ikke lagret på riktig måte i datastrukturen internt i den lagrede prosedyren. Det er mulig det ikke er nok tilgjengelig minne for den lagrede prosedyrens prosess.

Gjør slik:

Send kallet til den lagrede prosedyren på nytt. Hvis problemet gjentar seg, må du øke minnet som er tilgjengelig for den lagrede prosedyrens beskyttede prosess. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4013: Lagringspunktet ble ikke tømt.:

Forklaring

Den lagrede prosedyren kunne ikke tømme et lagringspunkt for databasetransaksjonen. Det er mulig at det manglende lagringspunktet ikke ble definert av den lagrede prosedyren, eller at lagringspunktet ble frigitt gjennom databasehandlinger som ble utført utenfor den lagrede prosedyren.

Gjør slik:

Send kallet til den lagrede prosedyren på nytt. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4014: Forsøket på å finne DB2 AUTOCOMMIT-innstillingen, mislyktes.:

Forklaring

Den lagrede prosedyrens forsøk på å bestemme om DB2 AUTOCOMMIT-funksjonen er aktivert eller deaktivert, mislyktes.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4015: Forsøket på å definere DB2 AUTOCOMMIT OFF mislyktes.:

Forklaring

Den lagrede prosedyrens forsøk på å deaktivere DB2 AUTOCOMMIT-funksjonen mislyktes.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4016: Ingen data ble returnert fra CLI-kallet SQLFetch().:

Forklaring

Det ble ikke returnert noen data til den lagrede prosedyren fra CLI-funksjonen SQLFetch(). Dette kan være akseptabelt, men den lagrede prosedyren skulle ikke ha tillatt at feilen ble propagert uendret gjennom den lagrede prosedyren.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4017: Objektet *objektnavn* var ikke riktig konstruert.:

Forklaring

Et databaseobjekt internt i den lagrede prosedyren ble ikke klargjort på riktig måte.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4018: Databasefrakoblingen mislyktes.:

Forklaring

Den lagrede prosedyren ble ikke koblet fra databasen.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4019: DB2 SQL-feil - SQLCODE *sqlkode*, SQLSTATE *sqlstatus*, SQLMESG *sqlmelding*.:

Forklaring

Det oppstod en feil under utføringen av en SQL-setning som ble sendt fra den lagrede prosedyren til databasen.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4020: DB2 SQL-feil - Ingen opplysninger er tilgjengelig.:

Forklaring

Feilsøkinginformasjon er ikke tilgjengelig for en feil som oppstod under utføringen av en SQL-setning som ble sendt fra den lagrede prosedyren til databasen.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4021: DB2 SQL-feil - Ingen opplysninger er tilgjengelig.:

Forklaring

Det oppstod en feil under forsøk på å samle inn feilsøkinginformasjon for en annen feil som oppstod under utføringen av en SQL-setning som ble sendt fra den lagrede prosedyren til databasen.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

**4022: Tildelingen av DB2-referansen *referansenavn* mislyktes.:
Forklaring**

Det oppstod en feil under forsøket på å tildele en DB2-referanse i den lagrede prosedyren.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

**4023: Frigjøringen av DB2-referansen *referansenavn* mislyktes.:
Forklaring**

Det oppstod en feil under forsøket på å frigi en DB2-referanse i den lagrede prosedyren.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

**4028: Transaksjonen ble ikke stoppet.:
Forklaring**

Det oppstod en feil under forsøket på å avslutte den lagrede prosedyrens transaksjon.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

**4029: Det ble funnet like rader som deler samme navn og skjema i en hovedobjekttabell.:
Forklaring**

Liker rader som deler samme navn og skjema ble uventet funnet i en av metadatakatalogtabellene. Denne delingen angir en intern feil i den lagrede prosedyren.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

**4030: DBINFO-strukturen er ikke klargjort. Kontroller at den lagrede prosedyren er opprettet i databasen ved hjelp av parameteren DBINFO.:
Forklaring**

Den lagrede prosedyren mottok ikke en DBINFO-struktur fra databaseklienten.

Gjør slik:

Kontroller at den lagrede prosedyren er katalogisert i den riktige databasen ved hjelp av parameteren DBINFO.

**4031: Definerings av skjemaet som DB2INFO mislyktes.:
Forklaring**

Den lagrede prosedyren definerte ikke DB2INFO som gjeldende skjema.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Send kallet til den lagrede prosedyren på nytt.

**4032: Operasjonen mislyktes på grunn av en kollisjon mellom et objekt i hovedobjekttabellen og objektet som ble satt inn.:
Forklaring**

En SQL INSERT-setning mislyktes i den lagrede prosedyren fordi den ville gi en duplikat metadataobjektpost i en av metadatakatalogtabellene.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

**4033: Operanden for kolonnefunksjonen omfatter en kolonnefunksjon.:
Forklaring**

En kolonnefunksjon som er nestet i en annen kolonnefunksjon, ble oppdaget i en av SQL-setningene som ble sendt av den lagrede prosedyren. Kolonnefunksjoner kan ikke være nestet i SQL-setninger.

Gjør slik:

Endre SQL-uttrykksmalen for inndataattributtet eller målobjektet slik at det ikke lenger finnes nestede kolonnefunksjoner i SQL-setningene som genereres av den lagrede prosedyren.

4034: Innstillingen for DB2 ISOLATION LEVEL er ikke fastsatt.:

Forklaring

Den lagrede prosedyrens forsøk på å fastsette isolasjonsnivået for databasetransaksjonen mislyktes. Isolasjonsnivået kunne ikke fastsettes.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4035: Definerings av DB2 ISOLATION LEVEL som READ STABILITY mislyktes.:

Forklaring

Den lagrede prosedyrens forsøk på å definere isolasjonsnivået for databasetransaksjonen mislyktes. Den lagrede prosedyren krever isolasjonsnivået READ STABILITY.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4036: Det var ikke mulig å fastsette hvilken versjon av DB2 Universal Database som er installert.:

Forklaring

Den lagrede prosedyrens forsøk på å bestemme versjonsnivået til databasesystemet mislyktes.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4037: DB2 Universal Database, versjon *versjonsnummer*, opprettingspakke *nummer* er installert, men er ikke kompatibel med denne versjonen av DB2 Cube Views. Oppgrader din versjon av DB2 Universal Database.:

Forklaring

Den installerte versjonen av DB2 Universal Database er ikke kompatibel med denne versjonen av DB2 Cube Views. Oppgrader versjonen av DB2 Universal Database slik at den har samme versjonsnivå eller nyere versjonsnivå enn DB2 Cube Views.

Gjør slik:

Kontroller at det er installert kompatible versjoner av DB2 Universal Database og DB2 Cube Views på samme tjener. Du finner flere opplysninger i installerings- og konfigureringsdokumentasjonen.

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøver å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

**4038: En SQL-setning kunne ikke behandles fordi den er for lang eller for kompleks.:
Forklaring**

Den lagrede prosedyren utførte en setning som ikke kunne behandles fordi den overskrider en systemgrense enten for lengde eller kompleksitet, eller fordi det er tatt med for mange begrensninger eller utløsere.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

**4039: Den nødvendige SYSINFOVERSION-metadatatabellen finnes ikke i DB2-katalogtabellene. Migrer metadataene.:
Forklaring**

Den nødvendige SYSINFOVERSION-metadatatabellen finnes ikke i DB2-katalogtabellene. Migrer metadataene.

Gjør slik:

Oppdater DB2-katalogtabellene ved å migrere metadataene. Du finner informasjon om migrering i *Veiledning og referanse for DB2 Cube Views*. Ta kontakt med IBM Kundeservice, og oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

**4040: Versjonen av metadatatabellen samsvarer ikke med den gjeldende versjonen av DB2 Cube Views-APIen. Versjonen av metadatatabellene er versjonsnummer1, og den gjeldende API-versjonen er versjonsnummer2. Du må migrere metadatatabellene til den gjeldende API-versjonen.:
Forklaring**

VERSION-verdien i tabellen SYSINFOVERSION samsvarer ikke med den versjonen av DB2 Cube Views-APIen som du forsøker å utføre en operasjon med. Du må migrere metadatatabellene for kontrollere at SYSINFOVERSION-tabellen inneholder riktig VERSION-verdi.

Gjør slik:

Migrer metadatatabellene i DB2-katalogen. Du finner informasjon om migrering i *Veiledning og referanse for DB2 Cube Views*. Hvis du trenger mer hjelp, ta kontakt med IBM Kundeservice og oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4041: Versjonen av metadatatabellene er ikke oppgitt. Du må migrere metadatatabellene til den gjeldende API-versjonen.:

Forklaring

VERSION-verdien i SYSINFOVERSION-tabellen finnes ikke. Du må kjøre skriptet db2mdmigrate.sql script for å migrere metadatatabellene i DB2-katalogen.

Gjør slik:

Migrer de eksisterende metadatatabellene. Du finner informasjon om migrering i *Veiledning og referanse for DB2 Cube Views*. Hvis du trenger mer hjelp, ta kontakt med IBM Kundeservice og oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

4042: Det er oppgitt flere eller duplikate versjoner av metadatatabellene.

Kontroller metadatatabellene slik at bare en, riktig versjon er oppgitt.:

Forklaring

Det er oppgitt flere eller duplikate versjoner av metadatatabellene i SYSINFOVERSION-katalogtabellen. Du skal bare ha oppgitt en korrekt versjon i tabellen SYSINFOVERSION.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

MDUTILITY.LIB-feil

5000: Funksjonen kunne ikke analysere strengen *streng*.:

Forklaring

En metode internt i den lagrede prosedyren oppdaget en feil under analyse av en intern strengverdi.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

5001: Funksjonens nummerformat ble ikke riktig klargjort. Kontroller at det finnes nok tilgjengelig minne.:

Forklaring

Formateringsfunksjonen for ICU-nummer ble ikke klargjort på riktig måte i den lagrede prosedyren. Dette kan skyldes at det ikke er nok minneressurser tilgjengelig for den lagrede prosedyrens prosess. Det kan også skyldes at feil versjon av ICU-bibliotekene blir lastet inn av den lagrede prosedyren.

Gjør slik:

Øk tilgjengelig minne for den lagrede prosedyrens beskyttede prosess og send kallet til den lagrede prosedyren på nytt. Kontroller at riktig versjon av ICU-bibliotekene du bruker sammen med den gjeldende versjonen av den lagrede

prosedyren, blir lastet inn. Det er mulig du må kontrollere søkebanen for kjøretidsbiblioteker som er definert i systemvariablene, for å bestemme riktig konfigurasjon.

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

5002: Funksjonens ressurspakke ble ikke riktig klargjort. Feilkode=*kode*.

Kontroller at pakken finnes og at den ligger i banen *banenavn*.

Forklaring

En ICU-ressurspakke ble ikke klargjort på riktig måte i den lagrede prosedyren. Feil klargjøring kan skyldes disse problemene: det er ikke nok minneressurser tilgjengelig for den lagrede prosedyrens prosess, feil versjon av ICU-bibliotekene ble lastet inn av den lagrede prosedyren, eller feil ressurspakke ble lastet inn for den lagrede prosedyren.

Gjør slik:

Øk tilgjengelig minne for den lagrede prosedyrens beskyttede prosess og send kallet til den lagrede prosedyren på nytt. Kontroller at riktig versjon av ICU-bibliotekene du bruker sammen med den gjeldende versjonen av den lagrede prosedyren, blir lastet inn. Det er mulig du må kontrollere søkebanen for kjøretidsbiblioteker som er definert i systemvariablene, for å bestemme riktig konfigurasjon. Kontroller at riktig versjon av den lagrede prosedyrens ressurspakke er installert på databasetjeneren.

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

5003: Databanen fra systemvariabelen *variabelnavn* ble ikke funnet. Kontroller at systemvariabelen er riktig definert.

Forklaring

En DB2-systemvariabel som den lagrede prosedyren bruker, er ikke definert.

Gjør slik:

Kontroller at DB2 er installert riktig på systemet. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

5004: Måldatastrømmen er lukket.

Forklaring

En datastrøm som brukes internt av den lagrede prosedyren, er uventet lukket. Det er mulig at det ikke finnes nok filreferanser tilgjengelig på databasesystemet.

Gjør slik:

Kontroller at nok filreferanser er tilgjengelig fra operativsystemet. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for

metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

5005: Målet skriver tegnene ved hjelp av standardkoding.:

Forklaring

Standardkodingen som er dokumentert for den lagrede prosedyren, blir brukt av den lagrede prosedyren når den skriver data til filer i filsystemet på databasetjeneren.

Gjør slik:

Applikasjoner som leser filene som den lagrede prosedyren har skrevet til, må kunne tolke data som er kodet med standardkodingen til den lagrede prosedyren.

5006: Inndataloggstrengen *streng* ble ikke skrevet. Strengnivået er *strengnivå* og loggnivået er *loggnivå*.:

Forklaring

Det gjeldende loggnivået tillater ikke at den angitte meldingen skrives til en av loggfilene som er definert for den lagrede prosedyren.

Gjør slik:

Endre loggnivået hvis du trenger å skrive den angitte meldingen til en av loggfilene for den lagrede prosedyren.

5007: Meldingsteksten for feilkoden *kode* ble ikke funnet.:

Forklaring

Teksten for den viste feilkoden ble ikke funnet i ressurspakkefilen for den lagrede prosedyren. Det er mulig du bruker feil versjon av ressurspakkefilen.

Gjør slik:

Kontroller at riktig versjon av den lagrede prosedyrens ressurspakkefil er installert på databasetjeneren. Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

5008: Det oppstod en feil ved tilgang til *operasjon* for det globale statiske MsgBase-objektet.:

Forklaring

Det oppstod en feil i den lagrede prosedyren under forsøk på å få tilgang til et internt meldingsobjekt.

Gjør slik:

Ta kontakt med IBM Kundeservice for å få hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

Eksterne API-feil

6000-6199

6000: OLAPMSG() mislyktes med feilkoden *kode*:

Forklaring

Den lagrede prosedyren mislyktes under utføringen.

Gjør slik:

Avhengig av returkoden kan du enten rette problemet og sende kallet til den lagrede prosedyren på nytt, eller kontakte IBM Kundeservice og be om hjelp. Hvis du kontakter IBM Kundeservice, må du oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

6001: De klargjorte SQL-malene for objektet *objektnavn* er ugyldige med verdien *verdi*. Årsaks-ID=*ID*, melding *melding*:

Forklaring

Den klargjorte SQL-malen er SQL-setningsfragmentet du kan forme ved å slå sammen SQL-uttrykksmalene for alle attributtene og målene som er med i et sammensatt attributt eller sammensatt mål. Det ble funnet et problem med den klargjorte SQL-malen for det oppgitte objektet.

Gjør slik:

Se årsaks-IDen og meldingen som er oppgitt. Send kallet til den lagrede prosedyren på nytt når du har gjort endringene som er forslått i årsaksmeldingen.

6002: Objektet *objekt1* refererer til objektet *objekt2*, men objektet *objekt2* finnes ikke i databasen:

Forklaring

Databasobjekter kan referere til andre objekter bare hvis disse objektene finnes i databasen.

Gjør slik:

Opprett objektet det skal refereres til, i databasen og send deretter forespørselen om metadataoperasjonen på nytt. Du kan eventuelt fjerne referansen til det manglende objektet og deretter sende forespørselen om metadataoperasjonen på nytt.

6003: Loggen *loggnavn* i den oppgitte banen kunne ikke åpnes. Kontroller at den oppgitte banen finnes og at det er skrivetilgang til filen:

Forklaring

Minst en av loggfilene som brukes av den lagrede prosedyren, kunne ikke åpnes.

Gjør slik:

Kontroller at banen som er oppgitt i konfigurasjonsfilen for den lagrede prosedyren, finnes. Kontroller at bruker_IDen som kjører den lagrede prosedyren på databasetjeneren, har autorisasjon til å opprette, lese og skrive til de nødvendige loggfilene.

**6005: Inndataparameteren for metadata er uventet tom for denne operasjonen. Den manglende metadataparameteren er nødvendig for denne operasjonen.:
Forklaring**

Den forespurte metadataoperasjonen krever at metadata gis som inndata. Parameteren for utveksling av metadata for den lagrede prosedyren er uventet tom.

Gjør slik:

Send forespørselen om metadataoperasjonen på nytt med de nødvendige metadataene.

**6006: Det ble ikke funnet noen objekter som samsvarer med søkekriteriene: søkekriterier.:
Forklaring**

Metadataoperasjonen fant ingen metadataobjekter som samsvarer med de oppgitte søkekriteriene. Det ble ikke gjort noen endringer i innholdet i metadatakatalogen.

Gjør slik:

Send forespørselen om metadataoperasjonen på nytt med nye søkekriterier hvis du vil endre innholdet i metadatakatalogen.

**6007: Det ble funnet kollisjoner mellom objekter i katalogen og objekter som blir importert. Det ble ikke gjort noen endringer i metadataene.:
Forklaring**

Det ble oppdaget kollisjoner mellom objektene som blir importert, og objektene som allerede finnes i metadatakatalogen. På grunn av importmodusen du oppgav, ble det ikke gjort noen endringer i objektene i metadatakatalogen.

Gjør slik:

Send forespørselen om metadataoperasjonen på nytt med ny importmodus hvis du vil endre innholdet i metadatakatalogen.

**6008: Det finnes et duplikat av objekt i metadatainndata med identiteten ID.:
Forklaring**

Duplikate metadataobjekter ble oppdaget i inndataene for metadata for denne metadataoperasjonen. Duplikate objekter er ikke tillatt som inndata for metadataoperasjoner.

Gjør slik:

Fjern det duplikate metadataobjektet fra inndataene for metadata og send forespørselen om metadataoperasjonen på nytt.

6009: Det finnes allerede et objekt som deler den samme identiteten som inndataobjektet *objektnavn* i metadatakatalogen.:

Forklaring

Metadataoperasjonen kunne ikke utføres fordi et metadataobjekt med samme identitet allerede finnes i metadatakatalogen.

Gjør slik:

Slett objektet som deler samme identitet som objektet du oppretter, fra metadatakatalogen før du sender forespørslene om metadataoperasjonen på nytt. Du kan også endre det eksisterende objektet slik at det får egenskaper som svarer til objektet du oppretter. Ellers må du utelukke det nye objektet som er årsaken til feilen, fra metadataoperasjonen du utfører.

6010: Referansen til objektet *objektnavn* finnes allerede for inndataobjektet *objektnavn*.:

Forklaring

Det er allerede definert en referanse mellom de oppgitte objektene i metadatakatalogen. Du kan ikke ha to like referanser.

Gjør slik:

Fjern en av de like referansene fra forespørselen om metadataoperasjon.

6011: Skjemaet til objektet *objektnavn* kan ikke endres ved hjelp av operasjonen for å endre navn.:

Forklaring

Operasjonen for endring av navn kan ikke brukes til å endre skjemaet for et metadataobjekt.

Gjør slik:

Sørg for at skjemaet som er oppgitt for objektet du endrer navnet på, blir beholdt, eller bruk ALTER-operasjonen.

6013: Versjon *versjon1* av XML-skjemaet som brukes av klienten, støttes ikke av APIen på tjeneren. APIen på tjeneren støtter versjon *versjon2* av XML-skjemaet.:

Forklaring

Versjonen av XML-skjemaet som brukes av klienten og er innebygd i inndataparameterstrengene, støttes ikke av versjonen av den lagrede prosedyren på tjeneren.

Gjør slik:

Kontroller at klientapplikasjonen og den lagrede prosedyren bruker samme versjon av XML-skjemaet som er publisert med dette produktet.

6014: SQL-malene for objektet *objektnavn* kan ikke formuleres. Årsaks-ID *ID*, melding *melding*.:

Forklaring

Den lagrede prosedyren formulerer SQL-malene for attributter og mål ved å slå sammen SQL-uttrykkmalene for alle attributtene og målene som er med i et sammensatt attributt eller sammensatt mål. Det ble oppdaget et problem i den lagrede prosedyren under formuleringen av en SQL-mal.

Gjør slik:

Se årsaks-IDen og meldingen som er oppgitt. Send kallet til den lagrede prosedyren på nytt når du har gjort endringene som er forslått i årsaksmeldingen.

6015: Databasebruker-IDen har ikke autorisasjon til å opprette et databaseskjema i den aktive databasen.:

Forklaring

Bruker-IDen som eier den lagrede prosedyrens prosess på databasetjeneren, har ikke autorisasjon til å opprette et databaseskjema i den aktive databasen. Det blir opprettet et databaseskjema for hvert entydige metadataobjektskjema.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Gi bruker-IDen som eier den lagrede prosedyrens prosess, autorisasjon til å opprette et skjema i den aktive databasen. Send kallet til den lagrede prosedyren på nytt.

6016: Databasebruker-IDen har ikke autorisasjon til å utføre en nødvendig handling i den aktive databasen. Følgende feilmelding ble returnert fra databasetjeneren: *melding*.:

Forklaring

Bruker-IDen som eier den lagrede prosedyrens prosess på databasetjeneren, har ikke autorisasjon til å utføre en nødvendig handling i den aktive databasen.

Gjør slik:

Kontroller databasesystemets loggfiler på klienten og tjeneren. Gi bruker-IDen som eier den lagrede prosedyrens prosess, den nødvendige autorisasjonen. Send kallet til den lagrede prosedyren på nytt.

6017: Objektet *objektnavn* finnes ikke i metadatakatalogen.:

Forklaring

Den forespurte operasjonen krever at det oppgitte objektet finnes i metadatakatalogen.

Gjør slik:

Opprett objektet i metadatakatalogen før du sender forespørselen om metadataoperasjonen på nytt.

6018: En nødvendig tabell finnes ikke i databasen. Følgende feilmelding ble returnert fra databasetjeneren: *melding*.:

Forklaring

En tabell som er nødvendig for den forespurte operasjonen, finnes ikke i databasen.

Gjør slik:

Hvis tabellen som mangler er en brukertabell, oppretter du tabellen og sender forespørselen om metadataoperasjonen på nytt. Hvis tabellen som mangler er en metadatakatalogtabell eller en databasesystemtabell, må du kontakte IBM Kundeservice og be om hjelp. Oppgi status-IDen og teksten for metadataoperasjonen du prøvde å utføre. Hvis det er mulig, bør du også ha loggfilene for den lagrede prosedyren fra databasetjeneren tilgjengelig.

6019: API-operasjonen bruker versjon *versjonsnummer* av XML, og metadataene bruker versjon *versjonsnummer* av XML. API-operasjonen og metadataene må bruke samme XML-versjon.:

Forklaring

API-operasjonen og metadataene må bruke samme versjon av XML.

Gjør slik:

Kontroller at API-operasjonen og metadataene bruker samme versjon av XML.

6020: API-operasjonen *operasjonsnavn* støtter ikke versjon *versjonsnummer* av metadata-XML.:

Forklaring

XML-versjonen for API-operasjonen må være 8.2.0.1.0, og XML-versjonen for metadataene må være 8.1.2.1.0 for TRANSLATE-operasjonen.

Gjør slik:

Du finner flere opplysninger om metadataregler, metadatavalidering og spørringsoptimalisering i *Veiledning og referanse for DB2 Cube Views*.

6021: Du kan bruke versjon *versjonsnummer* av XML bare for DESCRIBE-operasjonen.:

Forklaring

Du kan bruke den oppgitte XML-versjonen bare for DESCRIBE-operasjonen.

Gjør slik:

Du finner flere opplysninger om metadataregler, metadatavalidering og spørringsoptimalisering i *Veiledning og referanse for DB2 Cube Views*.

Valideringsfeil

6200: Objektet *objektnavn* er ikke fullstendig. Kontroller at de nødvendige egenskapene er definert.:

Forklaring

Enkelte av de nødvendige egenskapene er ikke definert for det oppgitte inndataobjektet.

Gjør slik:

Definer de nødvendige egenskapene for objektet og send forespørselen om metadataoperasjon på nytt.

6201: Hierarkiet *hierarkinaavn* er ugyldig fordi det er av typen rekursiv, men det har ikke nøyaktig to attributter.:

Forklaring

Et rekursivt hierarki må referere til nøyaktig to attributter. Det identifiserte hierarkiet bryter denne regelen.

Gjør slik:

Endre hierarkiet slik at det refererer til nøyaktig to attributter. *Installerings- og brukerhåndboken* inneholder flere opplysninger om metadataregler.

6202: Objektet *objektnavn* må ha minst en SQL-mal.:

Forklaring

På grunnlag av metadataobjektdefinisjonene som er oppgitt i produktdokumentasjonen, må det være definert minst en SQL-mal for det identifiserte målet.

Gjør slik:

Endre det identifiserte målet slik at det får minst en SQL-mal definert. *Installerings- og brukerhåndboken* inneholder flere opplysninger om metadataregler.

6206: Attributtet *attributtnavn* kan bare ha en SQL-mal.:

Forklaring

På grunnlag av metadataobjektdefinisjonene som er oppgitt i produktdokumentasjonen, må det bare være definert en enkelt SQL-mal for det identifiserte målet.

Gjør slik:

Endre det identifiserte målet slik at det får bare en SQL-mal definert. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6207: Attributtet *attributtnavn* er en del av en kombinerings, men har ingen kolonnereferanse.:

Forklaring

Det identifiserte attributtobjektet må referere til en databasekolonne hvis det skal være gyldig at et metadatakombineringsobjekt refererer til det.

Gjør slik:

Endre det identifiserte attributtobjektet slik at det refererer til en databasekolonne, eller endre det tilhørende kombineringsobjektet slik at det refererer til et annet attributtobjekt som refererer til en databasekolonne. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6208: Attributtet *attributtnavn* er en del av en kombinerings, og det må peke på den samme tabellen som *tabellnavn*..

Forklaring

Det første identifiserte attributtobjektet refererer til den samme databasetabellen som det andre identifiserte objektet.

Gjør slik:

Endre det første identifiserte attributtet slik at det refererer til den samme databasetabellen som det andre identifiserte objektet, eller endre det tilhørende kombineringsobjektet slik at det refererer til et annet attributtobjekt som refererer til den samme databasetabellen som det andre identifiserte objektet. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6209: Skjemaet for objektet *objektnavn* overskrider maksimal lengde.:

Forklaring

Skjemaet for det identifiserte objektet overskrider den maksimale lengden.

Gjør slik:

Reduser lengden på skjemaet for det identifiserte objektet. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6210: Navnet på objektet *objektnavn* overskrider maksimal lengde.:

Forklaring

Navnet på det identifiserte objektet overskrider den maksimale lengden.

Gjør slik:

Reduser lengden på navnet på det identifiserte objektet. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6211: Tabellnavnet for objektet *objektnavn* overskrider maksimal lengde.:

Forklaring

Tabellnavnet for det identifiserte objektet overskrider den maksimale lengden.

Gjør slik:

Reduser lengden på navnet på tabellen. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6212: Forretningsnavnet på objektet *objektnavn* overskrider maksimal lengde.:

Forklaring

Forretningsnavnet for det identifiserte objektet overskrider den maksimale lengden.

Gjør slik:

Reduser lengden på forretningsnavnet. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6213: Kommentarene for objektet *objektnavn* overskrider maksimal lengde.:

Forklaring

Kommentarene for det identifiserte objektet overskrider den maksimale lengden.

Gjør slik:

Reduser lengden på kommentarene for det identifiserte objektet. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6214: Skjemaet for objektet *objektnavn* kan ikke begynne med SYS.:

Forklaring

Skjemaet for metadataobjekter kan ikke begynne med strengen SYS.

Gjør slik:

Bruk et skjema som ikke begynner med SYS, for metadataobjekter. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6215: Skjemaet for objektet *objektnavn* kan ikke være SESSION.:

Forklaring

Skjemaet for metadataobjekter kan ikke være strengen SESSION.

Gjør slik:

Bruk et skjema som ikke er strengen SESSION, for metadataobjekter. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6216: Navnet og skjemaet for objektet *objektnavn* er ikke fullstendige.

Årsaks-ID=*ID*, melding=*melding*..

Forklaring

Navnet og/eller skjemaet for det identifiserte objektet mangler eller er ugyldig.

Gjør slik:

Oppgi gyldige strenger for både navn og skjema for det identifiserte objektet. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6217: Kubehierarkiet *hierarkinaavn* er ikke gyldig fordi det refererer til nivåer som hierarkiet *hierarkinaavn* ikke refererer til.:

Forklaring

Kubehierarkiet refererer til nivåer som det overordnede hierarkiet ikke refererer til.

Gjør slik:

Endre kubehierarkiet slik at det bare refererer til nivåer som også blir referert til av det overordnede hierarkiet. Du kan også endre hierarkiet slik at det refererer til de samme nivåene som kubehierarkiet refererer til. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

Advarselskoder

6250: APIen kan ikke opprette en funksjonell avhengighet for nivåobjektet *nivånavn* fordi nivånøkkelattributtet *nivånavn* ikke er tilordnet til en tabellkolonne.:

Forklaring

APIen kan ikke opprette den funksjonelle avhengigheten for nivåobjektet, fordi nivånøkkelattributtet ikke er tilordnet til en enkelt tabellkolonne.

Gjør slik:

Du finner flere opplysninger om funksjonelle avhengigheter i *Veiledning og referanse for DB2 Cube Views*.

6251: APIen oppretter ikke en funksjonell avhengighet for nivåobjektet *nivånavn*, fordi nivånøkkelattributtene samsvarer med en eksisterende, entydig begrensning på dimensjonstabellen.:

Forklaring

APIen oppretter ikke den funksjonelle avhengigheten for nivåobjektet, fordi nivånøkkelattributtet samsvarer med en eksisterende, entydig begrensning på dimensjonstabellen. Den funksjonelle avhengigheten er overflødig.

Gjør slik:

Du finner flere opplysninger om funksjonelle avhengigheter i *Veiledning og referanse for DB2 Cube Views*.

6252: APIen kan ikke opprette en funksjonell avhengighet for nivåobjektet *nivånavn*, fordi nivånøkkelattributtet *nivånavn* kan ha nullverdier.:

Forklaring

APIen kan ikke opprette en funksjonell avhengighet for et nivåobjekt hvis noen av nivånøkkelattributtene kan ha nullverdier.

Gjør slik:

Du finner flere opplysninger om funksjonelle avhengigheter i *Veiledning og referanse for DB2 Cube Views*.

6253: APIen kan ikke opprette en funksjonell avhengighet for nivåobjektet *nivånavn*, fordi attributtene gjelder for mer enn en tabell.:

Forklaring

APIen kan ikke opprette en funksjonell avhengighet for et nivåobjekt hvis nivånøkkelattributtene, standardattributtene og relaterte attributter gjelder for mer enn en tabell.

Gjør slik:

Du finner flere opplysninger om funksjonelle avhengigheter i *Veiledning og referanse for DB2 Cube Views*.

6254: Den funksjonelle avhengigheten for nivåobjektet *nivånavn* utelater standardattributtet eller det beslektede attributtet *attributtnavn*, fordi dette

attributtet allerede er tatt med som et nivåøkkelattributt.:

Forklaring

Den funksjonelle avhengigheten for et nivåobjekt utelater beslektede attributter som allerede er tatt med som nivåøkkelattributter.

Gjør slik:

Du finner flere opplysninger om funksjonelle avhengigheter i *Veiledning og referanse for DB2 Cube Views*.

6255: Den funksjonelle avhengigheten for nivåobjektet *nivånavn* utelater standardattributtet eller det beslektede attributtet *attributtnavn*, fordi dette attributtet er ikke tilordnet til en tabellkolonne.:

Forklaring

Den funksjonelle avhengigheten for nivåobjektet utelater standardattributtet eller det beslektede attributtet fordi attributtet ikke er tilordnet til en tabellkolonne.

Gjør slik:

Du finner flere opplysninger om funksjonelle avhengigheter i *Veiledning og referanse for DB2 Cube Views*.

6256: APIen kan ikke opprette en funksjonell avhengighet for nivåobjektet *nivånavn*, fordi alle standardattributtene og de beslektede attributtene var utelatt.:

Forklaring

Det kreves minst ett standardattributt eller beslektet attributt for å opprette en funksjonell avhengighet som tilsvarer et nivåobjekt.

Gjør slik:

Du finner flere opplysninger om funksjonelle avhengigheter i *Veiledning og referanse for DB2 Cube Views*.

6257: APIen kan ikke opprette en funksjonell avhengighet for nivåobjektet *nivånavn*, fordi det oppstod en feil under forsøk på å opprette den funksjonelle avhengigheten.:

Forklaring

Det oppstod en feil mens APIen forsøkte å utføre en SQL-setning som oppretter den funksjonelle avhengigheten.

Gjør slik:

Du finner flere opplysninger om funksjonelle avhengigheter i *Veiledning og referanse for DB2 Cube Views*. Postene i tjenerloggene kan også inneholde flere opplysninger.

6258: APIen kan ikke endre eller slette nivåobjektet *nivånavn*, fordi det oppstod en feil under forsøk på slette den tilhørende funksjonelle avhengigheten *avhengighetsnavn*.:

Forklaring

DB2 Cube Views kan ikke endre eller slette nivåobjektet fordi det oppstod en feil mens APIen forsøkte å utføre en SQL-setning som sletter den funksjonelle avhengigheten. Kontroller at du har autorisasjon til å opprette eller slette den funksjonelle avhengigheten.

Gjør slik:

Du finner flere opplysninger om funksjonelle avhengigheter i *Veiledning og referanse for DB2 Cube Views*. Postene i tjenerloggene kan også inneholde flere opplysninger.

6299: Minst ett databaseutsnitt ble funnet under valideringen. Det ble ikke utført begrensingsrelaterte valideringer for kombineringsene som omfatter utsnittskolonner. All annen validering ble utført.:

Forklaring

Det ble ikke utført begrensingsrelaterte valideringer for de kombineringsene som omfatter utsnittskolonner. Begrensingsrelaterte valideringer ble utført for alle andre kombineringsene, og alle andre valideringer ble utført for alle objektene.

Gjør slik:

Du finner flere opplysninger om metadatarregler, metadatavalidering og spørringsoptimalisering i *Veiledning og referanse for DB2 Cube Views*.

Regelrelaterte valideringsfeil

6300: Kubemodellen *modellnavn* refererer ikke til ett eller flere faktaobjekter.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte kubemodellobjektet. En kubemodell må referere til ett eller flere faktaobjekter.

Gjør slik:

Endre den identifiserte kubemodellen slik at den refererer til ett eller flere faktaobjekter. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6301: Kubemodellen *modellnavn* refererer ikke til null eller flere dimensjoner.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte kubemodellobjektet. En kubemodell må referere til null eller flere dimensjoner.

Gjør slik:

Endre den identifiserte kubemodellen slik at den refererer til null eller flere dimensjoner. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6302: Kubemodellen *modellnavn* mangler en dimensjon og/eller kombinerings for ett av parene av dimensjon og kombinerings.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte kubemodellobjektet. Et par av dimensjon og kombineringsreferanse for en kubemodell må referere til både en dimensjon og en kombineringsreferanse.

Gjør slik:

Endre den identifiserte kubemodellen slik at alle parene av dimensjon og kombineringsreferanse refererer til både en dimensjon og en kombineringsreferanse. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6303: Kombineringsreferansen kombineringsnavn som det er referert til av kubemodellen modellnavn, er ikke gyldig. Alle attributtene på en av sidene må være referert til av faktaobjektene faktaobjektnavn, og alle attributtene på den andre siden må være referert til av en av kubemodellens dimensjoner.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte kubemodellobjektet. Hver av kombineringsreferansene for en kubemodell må referere til attributtene til kubemodellens faktaobjekter på den ene siden, og til attributtene til en av kubemodellens dimensjoner på den andre siden.

Gjør slik:

Endre den ugyldige kombineringsreferansen for den identifiserte kubemodellen slik at alle attributtene på den ene siden av kombineringsreferansen kommer fra kubemodellens faktaobjekter, og alle attributtene på den andre siden av kombineringsreferansen kommer fra en av kubemodellens dimensjoner. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6304: En av samlingene i målet målnavn refererer direkte til dimensjonen dimensjonsnavn, som ikke er direkte referert til av kubemodellen modellnavn.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte kubemodellobjektet. Samlingene i et mål som brukes av en kubemodell, må bare referere til de dimensjonene som brukes av den samme kubemodellen.

Gjør slik:

Endre samlingen for det identifiserte målet slik at den bare refererer til de dimensjonene som brukes av den identifiserte kubemodellen. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6305: Samlingen av tomme dimensjonslister i målet målnavn samsvarer ikke med minst en tidligere dimensjon som ikke samsvarer, fra kubemodellen modellnavn.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte kubemodellobjektet. Samlinger av tomme dimensjonslister i målene som brukes av kubemodeller, må samsvare med minst en dimensjon som ellers ikke samsvarer i hver kubemodell.

Gjør slik:

Endre samlingen for det identifiserte målet slik at samlingen av tomme dimensjonslister samsvarer med minst en tidligere dimensjon som ikke samsvarer, i den identifiserte kubemodellen. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6306: Målet *målnavn* kan bare inneholde samlingen av tomme dimensjonslister, fordi kubemodellen *modellnavn* ikke refererer til noen dimensjonsobjekter.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte kubemodellobjektet. Når en kubemodell ikke refererer til noen dimensjoner, må kubemodellens mål bare inneholde samlingen av tomme dimensjonslister.

Gjør slik:

Endre det identifiserte målet slik at det bare inneholder samlingen av tomme dimensjonslister. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6307: Kuben *kubenaavn* refererer ikke til ett kubefaktaobjekt.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte kubeobjektet. En kube må referere til ett kubefaktaobjekt.

Gjør slik:

Endre den identifiserte kubenaavn slik at den refererer til ett kubefaktaobjekt. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6308: Kuben *kubenaavn* refererer ikke til minst ett kubedimensjonsobjekt.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte kubeobjektet. En kube må referere til minst ett kubedimensjonsobjekt.

Gjør slik:

Endre den identifiserte kubenaavn slik at den refererer til minst ett kubedimensjonsobjekt. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6309: Kubefaktaene *kubefaktanaavn* som det refereres til av kubenaavn, er ikke utledet fra faktaobjektet som det refereres til av kubemodellen *modellnavn*.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte kubeobjektet. Kubefaktaobjektet som brukes av den identifiserte kubenaavn, må være utledet fra faktaobjektet som brukes av den identifiserte kubemodellen.

Gjør slik:

Endre ett eller flere av de identifiserte objektene slik at ikke lenger er brudd på den oppgitte regelen. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6310: Kubedimensjonen *kubedimensjonsnavn* **som det refereres til av kubens** *kubnavn*, **er ikke utledet fra ett av dimensjonsobjektene som det refereres til av kubemodellen** *modellnavn*..

Forklaring

En metadataobjektregel ble brutt av det identifiserte kubeobjektet. En kubedimensjon som brukes av den identifiserte kubens, må være utledet fra en av dimensjonene som brukes av den identifiserte kubemodellen.

Gjør slik:

Endre ett eller flere av de identifiserte objektene slik at ikke lenger er brudd på den oppgitte regelen. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6311: Faktaobjektet *faktanavn* **refererer ikke til noen mål**..

Forklaring

En metadataobjektregel ble brutt av det identifiserte faktaobjektet. Et faktaobjekt må referere til minst ett mål.

Gjør slik:

Endre det identifiserte faktaobjektet slik at det refererer til minst ett mål. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6312: Noen av attributtene og målene som det refereres til av faktaobjektet *faktanavn*, **kan ikke kombineres ved hjelp av faktaobjektkombineringsregler**..

Forklaring

En metadataobjektregel ble brutt av det identifiserte faktaobjektet. Attributtene og målene til et faktaobjekt må alle kunne kombineres ved hjelp av kombineringsobjektene til faktaobjektet.

Gjør slik:

Gjør alle attributtene og målene som det er referert til av det identifiserte faktaobjektet, kombinerbare ved å referere til flere kombineringsobjekter fra faktaobjektet. Du kan også fjerne de attributtene som ikke er kombinerbare med faktaobjektets gjeldende kombineringsregler, fra faktaobjektet. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6313: Faktaobjektet *faktanavn* **har flere kombineringsregler mellom to tabeller**..

Forklaring

En metadataobjektregel er blitt brutt av det identifiserte faktaobjektet. Et faktaobjekt kan ikke ha flere kombineringsregler mellom de samme to tabellene.

Gjør slik:

Endre det identifiserte faktaobjektet slik at det bare har en kombinerings mellom to gitte tabeller. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6314: Faktaobjektet *faktanavn* inneholder en kombineringsløyfe.:**Forklaring**

En metadataobjektregel ble brutt av det identifiserte faktaobjektet. Kombineringsene for det identifiserte faktaobjektet danner en banesløyfe. Dette er ikke tillatt.

Gjør slik:

Fjern en av kombineringsene som er årsaken til sløyfen, fra det identifiserte faktaobjektet, eller endre en av kombineringsene som er årsak til sløyfen, slik at det ikke lenger finnes en sløyfe. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6315: Kombineringsnavn refererer ikke bare til disse attributtene i faktaobjektet *faktanavn*.:**Forklaring**

En metadataobjektregel ble brutt av det identifiserte faktaobjektet. Kombineringsene til et faktaobjekt må referere bare til attributtene til dette faktaobjektet.

Gjør slik:

Endre den identifiserte koblingen slik at den bare refererer til attributtene til det identifiserte faktaobjektet, eller tilføy i faktaobjektet de manglende attributtene som det identifiserte kombineringsobjektet refererer til. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6316: Kubefaktaobjektet *kubefaktanavn* refererer ikke til et faktaobjekt eller det refererer til mer enn ett faktaobjekt.:**Forklaring**

En metadataobjektregel ble brutt av det identifiserte kubefaktaobjektet. Et kubefaktaobjekt må referere til ett faktaobjekt.

Gjør slik:

Endre det identifiserte kubefaktaobjektet slik at det refererer til ett faktaobjekt. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6317: Kubefaktaobjektet *kubefaktanavn* refererer ikke til noen mål.:**Forklaring**

En metadataobjektregel ble brutt av det identifiserte kubefaktaobjektet. Et kubefaktaobjekt må referere til minst ett mål.

Gjør slik:

Endre det identifiserte kubefaktaobjektet slik at det refererer til minst ett mål. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6318: Målet *målnavn* som det refereres til av kubefaktaobjektet *kubefaktanavn*, er ikke en del av faktaobjektet *faktanavn*.

Forklaring

En metadataobjektregel ble brutt av det identifiserte kubefaktaobjektet. Et kubefaktaobjekt må referere til mål som det er referert til av faktaobjektet som kubefaktaobjektet er utledet fra.

Gjør slik:

Tilføy det identifiserte målet til det identifiserte faktaobjektet, eller fjern det identifiserte målet fra det identifiserte kubefaktaobjektet. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6319: Dimensjonen *dimensjonsnavn* refererer ikke til noen attributter. En dimensjon må referere til minst ett attributt.

Forklaring

En metadataobjektregel ble brutt av det identifiserte dimensjonsobjektet. Et dimensjonsobjekt må referere til minst ett attributt.

Gjør slik:

Endre det identifiserte dimensjonsobjektet slik at det refererer til minst ett attributt. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6320: Noen av attributtene det er referert til av dimensjonen *dimensjonsnavn*, kan ikke kombineres ved hjelp av dimensjonskombineringer.

Forklaring

En metadataobjektregel ble brutt av det identifiserte dimensjonsobjektet. Attributtene til et dimensjonsobjekt må alle kunne kombineres ved hjelp av kombineringsobjektene til dimensjonen.

Gjør slik:

Gjør alle attributtene som det er referert til av det identifiserte dimensjonsobjektet, kombinerbare ved å referere til flere kombineringsobjekter fra dimensjonsobjektet. Du kan også fjerne de attributtene som ikke er kombinerbare med dimensjonens gjeldende kombineringer, fra dimensjonsobjektet. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6321: Dimensjonen *dimensjonsnavn* inneholder en kombineringsløyfe.

Forklaring

En metadataobjektregel ble brutt av det identifiserte dimensjonsobjektet. Kombineringene for det identifiserte dimensjonsobjektet danner en banesløyfe. Dette er ikke tillatt.

Gjør slik:

Fjern en av kombineringsene som er årsaken til sløyfen, fra det identifiserte dimensjonsobjektet, eller endre en av kombineringsene som er årsak til sløyfen, slik at det ikke lenger finnes en sløyfe. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6322: Dimensjonen *dimensjonsnavn* har flere kombinerings mellom to tabeller.: Forklaring

En metadataobjektregel er blitt brutt av det identifiserte dimensjonsobjektet. Et dimensjonsobjekt kan ikke ha flere kombinerings mellom de samme to tabellene.

Gjør slik:

Endre det identifiserte dimensjonsobjektet slik at det bare har en kombinerings mellom to gitte tabeller. Produktdokumentasjonen inneholder flere opplysninger om metadatarregler.

6323: Hierarkiet *hierarkinaavn* refererer til nivåer som det ikke refereres til av hierarkiets overordnede dimensjon *hierarkinaavn*.: Forklaring

Hierarkiene til en dimensjon må bare referere til nivåene til dette dimensjonsobjektet. Dette hierarkiet refererer til nivåer som den overordnede dimensjonen ikke refererer til.

Gjør slik:

Endre hierarkiet slik at det bare refererer til nivået til det overordnede dimensjonsobjektet, eller tilføy nivåene som hierarkiet refererer, til dimensjonsobjektet. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6325: Kombinerings til en dimensjon må referere til attributtene til kun denne dimensjonen. Kombinerings *kombineringsnavn* refererer til attributter som ikke er i dimensjonen *dimensjonsnavn*.: Forklaring

Dimensjonen og kombineringsen bryter metadataobjektregelen om at en kombinerings av en dimensjon bare kan referere til attributter for denne dimensjonen.

Gjør slik:

Endre kombineringsen slik at den bare refererer til attributtene til den overordnede dimensjonen, eller tilføy attributtene som kombineringsen refererer, til dimensjonen. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6326: Kubedimensjonen *kubedimensjonsnavn* refererer ikke til en dimensjon.: Forklaring

En metadataobjektregel ble brutt av det identifiserte kubedimensjonsobjektet. Et kubedimensjonsobjekt må referere til en dimensjon.

Gjør slik:

Endre det identifiserte kubedimensjonsobjektet slik at det refererer til en dimensjon. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6327: Kubedimensjonen *kubedimensjonsnavn* refererer ikke til et kubehierarki.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte kubedimensjonsobjektet. Et kubedimensjonsobjekt må referere til et kubehierarki.

Gjør slik:

Endre det identifiserte kubedimensjonsobjektet slik at det refererer til et kubehierarki. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6328: Kubehierarkiet *kubehierarkinaavn* som det refereres til av kubedimensjonen *kubedimensjonsnavn*, er ikke utledet fra noen av hierarkiene som det refereres til av dimensjonen *dimensjonsnavn*.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte kubedimensjonsobjektet. Kubehierarkiet som brukes av den identifiserte kubedimensjonen, må være utledet fra et av hierarkiene som brukes av den identifiserte dimensjonen.

Gjør slik:

Endre ett eller flere av de identifiserte objektene slik at ikke lenger er brudd på den oppgitte regelen. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6329: Hierarkiet *hierarkinaavn* refererer ikke til noen nivåer.:

Forklaring

Et hierarkiobjekt må referere til minst ett nivå.

Gjør slik:

Endre det identifiserte hierarkiobjektet slik at det refererer til minst ett nivå. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6330: Hierarkiet *hierarkinaavn*, som bruker en rekursiv distribuering, refererer ikke til nøyaktig to nivåer.:

Forklaring

En metadataobjektregel er blitt brutt av det identifiserte hierarkiobjektet. Et hierarkiobjekt som bruker rekursiv distribuering, må referere til to nivåer.

Gjør slik:

Endre det identifiserte hierarkiobjektet slik at det refererer til to nivåer. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

**6332: Typen for hierarkiet *hierarkinaavn* er ikke kompatibel med distribueringen.:
Forklaring**

En metadataobjektregel er blitt brutt av det identifiserte hierarkiobjektet. Kompatibilitet for hierarkityper og distribuering er beskrevet i installerings- og brukerhåndboken.

Gjør slik:

Endre det identifiserte hierarkiet slik at typen er kompatibel med distribueringen. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

**6334: Kubehierarkiet *hierarkinaavn* må referere til nøyaktig ett hierarki.:
Forklaring**

Kubehierarkiet bryter en metadataobjektregel om at et kubehierarki må referere til nøyaktig ett hierarki.

Gjør slik:

Endre det identifiserte kubehierarkiet slik at det refererer til ett hierarki. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

**6335: Kubehierarkiet *hierarkinaavn* må referere til minst ett kubnivå.:
Forklaring**

En metadataobjektregel er blitt brutt av det identifiserte kubehierarkiobjektet. Et kubehierarkiobjekt må referere til minst ett kubnivå.

Gjør slik:

Endre det identifiserte kubehierarkiobjektet slik at det refererer til minst et kubnivå. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

**6336: Kubehierarkiet *hierarkinaavn1* refererer til et kubnivå, men det tilsvarende hierarkiet *hierarkinaavn2* refererer ikke til det tilsvarende nivået *nivånavn*. Du må tilføye nivået *nivånavn* til hierarkiet *hierarkinaavn2*, eller fjerne det tilsvarende kubnivået fra kubehierarkiet *hierarkinaavn1*.:
Forklaring**

Kubehierarkiet bryter metadataobjektregelen om at et kubehierarki må referere til kubnivåer som er utledet fra nivåene det refereres til av det tilsvarende hierarkiet.

Gjør slik:

Tilføy nivået til hierarkiet eller fjern nivået fra kubehierarkiet. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6337: Rekkefølgen for kubnivåene i kubehierarkiet *hierarkinaavn* samsvarer ikke med rekkefølgen for de samme nivåene i hierarkiet *hierarkinaavn*.:

Forklaring

En metadataobjektregel er blitt brutt av det identifiserte kubehierarkiobjektet. Den relative rekkefølgen til kubnivåene i et kubehierarki må være den samme som den relative rekkefølgen til de samme nivåene i hierarkiet som kubehierarkiet er utledet fra.

Gjør slik:

Endre et av de identifiserte objektene slik at den relative rekkefølgen til attributtene er konsistent i begge de identifiserte objektene. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6340: En av SQL-uttrykksmalene for målet *målnavn* bruker en parameter som ikke er et attributt, et mål eller en kolonne.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte målobjektet. SQL-uttrykksmalene for målobjekter må bruke parametere som er attributter, mål eller kolonner.

Gjør slik:

Endre det identifiserte målet slik at SQL-uttrykksmalene bruker attributter, mål eller kolonner som parametere. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6341: Det finnes en avhengighetssløyfe mellom attributtene eller målene som brukes som parametere i SQL-uttrykksmalen for målet *målnavn*.:

Forklaring

En metadataobjektregel er blitt brutt av det identifiserte målobjektet. Attributtene og målene som brukes som parametere for SQL-uttrykksmalen for et mål, kan ikke danne en avhengighetssløyfe.

Gjør slik:

Endre det identifiserte målet slik at SQL-uttrykksmalene ikke inneholder avhengighetssløyfer som omfatter deres parametere. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6342: Målet *målnavn* har en tom streng definert for en av SQL-uttrykksmalene.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte målobjektet. SQL-uttrykksmalen for et mål kan ikke være en tom streng.

Gjør slik:

Endre det identifiserte målet slik at SQL-uttrykksmalen ikke lenger er en tom streng. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6343: SQL-uttrykksmalen for målet *målnavn* inneholder en samlingsfunksjon.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte målobjektet. SQL-uttrykksmalen for et mål kan ikke inneholde en samlingsfunksjon.

Gjør slik:

Endre det identifiserte målet slik at SQL-uttrykksmalen ikke lenger inneholder en samlingsfunksjon. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6344: Målet *målnavn* mangler en samling eller det refererer feilaktig til andre objekter enn mål.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte målobjektet. En samling er ikke nødvendig for et mål hvis dette målet refererer til minst ett annet mål og kun refererer til mål.

Gjør slik:

Endre det identifiserte målet ved å tilføye en samling eller sørge for at det identifiserte målet refererer til minst ett annet mål og kun refererer til mål. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6345: Antall SQL-uttrykksmaler i målet *målnavn* samsvarer ikke med antall parametere som er brukt med den første samlingsfunksjonen.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte målobjektet. Antall SQL-maler i et mål må samsvare med antall parametere for den første samlingsfunksjonen for dette målet, hvis det finnes en samling.

Gjør slik:

Endre det identifiserte målet slik at antall parametere for den første samlingsfunksjonen samsvarer med antall SQL-uttrykksmaler i målet. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6346: Målet *målnavn*, som har flere SQL-uttrykksmaler, definerer ikke minst ett trinn i samlingskriptet.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte målobjektet. Et mål med flere SQL-uttrykksmaler må definere minst ett trinn i samlingskriptet.

Gjør slik:

Endre det identifiserte målet slik at samlingskriptet får minst ett trinn. Du kan også fjerne en av målets SQL-uttrykksmaler forutsatt at den gjenværende SQL-uttrykksmalen bare refererer til andre mål. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6347: Målet *målnavn1* har et definert samlingskript. Det skal imidlertid ikke ha definerte samlingskript fordi målet *målnavn2*, som det refereres til, definerer

flere maler for SQL-uttrykk.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte målobjektet. Hvis målet A refererer til målet B, som definerer flere SQL-maler, kan ikke målet A ha et samlingskript. Denne regelen gjelder for alle nivåene i en målreferansetreoversikt.

Gjør slik:

Fjern samlingskriptet fra målet som gir dette problemet, eller fjern en av SQL-uttrykksmalene fra målet det refereres til. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6348: Målet *målnavn* inneholder en samlingsfunksjon med flere parametere som ikke brukes som den første samlingen.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte målobjektet. En samlingsfunksjon med flere parametere kan bare brukes som den første samlingen for et mål.

Gjør slik:

Gjør samlingsfunksjonen med flere parametere til den første samlingen som brukes av det identifiserte målet, eller fjern samlingsfunksjonen med flere parametere fra det identifiserte målet. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6349: Målet *målnavn* har ikke nøyaktig en samling av tomme dimensjonslister.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte målobjektet. Når et mål definerer en eller flere samlinger, må en av samlingene ha en tom liste over dimensjoner.

Gjør slik:

Endre det identifiserte målet slik at det får en enkelt tom liste over dimensjoner, eller endre det identifiserte målet slik at det ikke definerer noen samlinger. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6350: Dimensjonen *dimensjonsnavn* refereres til flere ganger i målet *målnavn* *measure*.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte målobjektet. I et mål kan det ikke refereres til en dimensjon mer enn en gang i en samling eller på tvers av flere samlinger.

Gjør slik:

Endre det identifiserte målet slik at det refererer bare en gang til den identifiserte dimensjonen. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6351: SQL-uttrykksmalen for objektet *objektnavn* mangler en symbolindikator med nummeret *nummer*. Symbolindikatorer må være nummerert fra og med nummer 1, og nummereringen må være fortløpende.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte målobjektet. I SQL-uttrykksmalen for et mål må symbolindikatorer være nummerert fra og med 1, og nummereringen må være fortløpende.

Gjør slik:

Endre det identifiserte målet slik at symbolindikatorene for SQL-uttrykksmalene er nummerert fortløpende fra og med 1. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6352: Målet *målnavn* inneholder en SQL-uttrykksmal som ikke bruker den oppgitte referansen, *referanse*.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte målobjektet. SQL-uttrykksmalen for et mål må bruke hver kolonne-, attributt- og målreferanse som er oppgitt. Hver referanse kan brukes mer enn en gang.

Gjør slik:

Endre SQL-uttrykksmalen for det identifiserte målet så slik at den bruker hver kolonne-, attributt- og målreferanse som er oppgitt. Du kan også fjerne kolonne-, attributt- og målreferansene som ikke brukes av det identifiserte målets SQL-uttrykksmal. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6353: En av SQL-uttrykksmalene for attributtet *attributtnavn* bruker en parameter som ikke er et attributt eller en kolonne.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte attributtobjektet. SQL-uttrykksmalene for attributtobjekter må bruke parametere som er attributter eller kolonner.

Gjør slik:

Endre det identifiserte attributtet slik at SQL-uttrykksmalene bruker attributter eller kolonner som parametere. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6354: Det finnes en avhengighetssløyfe mellom attributtene som brukes som parametere i SQL-uttrykksmalen for attributtet *attributtnavn*.:

Forklaring

En metadataobjektregel er blitt brutt av det identifiserte attributtobjektet. Attributtene som brukes som parametere for SQL-uttrykksmalen for et attributt, kan ikke danne en avhengighetssløyfe.

Gjør slik:

Endre det identifiserte målet slik at SQL-uttrykksmalene ikke inneholder avhengighetssløyfer som omfatter deres parametere. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6355: Attributtet *attributtnavn* har en tom streng definert i en av SQL-uttrykksmalene.:**Forklaring**

En metadataobjektregel ble brutt av det identifiserte attributtobjektet. SQL-uttrykksmalen for et attributt kan ikke være en tom streng.

Gjør slik:

Endre det identifiserte attributtet slik at SQL-uttrykksmalen ikke lenger er en tom streng. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6356: SQL-uttrykksmalen for attributtet *attributtnavn* inneholder en samlingsfunksjon.:**Forklaring**

En metadataobjektregel ble brutt av det identifiserte attributtobjektet. SQL-uttrykksmalen for et attributt kan ikke inneholde en samlingsfunksjon.

Gjør slik:

Endre det identifiserte attributtet slik at SQL-uttrykksmalen ikke lenger inneholder en samlingsfunksjon. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6358: Attributtet *attributtnavn* inneholder en SQL-uttrykksmal som ikke bruker den oppgitte referansen, *referanse*.:**Forklaring**

En metadataobjektregel ble brutt av det identifiserte attributtobjektet. SQL-uttrykksmalen for et attributt må bruke hver kolonne- og attributtreferanse som er oppgitt. Hver referanse kan brukes mer enn en gang.

Gjør slik:

Endre SQL-uttrykksmalen for det identifiserte attributtet må slik at den bruker hver kolonne- og attributtreferanse som er oppgitt. Du kan også fjerne kolonne- og attributtreferansene som ikke brukes av det identifiserte attributtets SQL-uttrykksmal. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6359: Attributtforholdet *forholdsnavn* refererer ikke til to distinkte attributter.:**Forklaring**

En metadataobjektregel ble brutt av det identifiserte attributtforholdsobjektet. Et attributtforholdsobjekt må referere til to distinkte attributter.

Gjør slik:

Endre det identifiserte attributtforholdsobjektet slik at det refererer til to distinkte attributter. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6360: Attributtforholdet *forholdsnavn* er definert feil. Egenskapen kardinalitet er definert til N:N, men egenskapen funksjonell avhengighet er definert til YES.: Forklaring

En metadataobjektregel ble brutt av det identifiserte attributtforholdsobjektet. Når funksjonsavhengighetsegenskapen til et attributtforhold er satt til YES, kan ikke kardinalitetsegenskapen til attributtforholdet være N:N.

Gjør slik:

Endre det identifiserte attributtforholdet slik at kardinaliteten ikke blir NN, eller funksjonsavhengighetsegenskapen blir NO. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6361: Kombineringsnavn refererer ikke til minst en triplett. En triplett inneholder et venstreattributt, et høyreattributt og en operator.: Forklaring

En metadataobjektregel ble brutt av det identifiserte kombineringsobjektet. En kombineringsnavn må referere til minst en triplett som inneholder et venstreattributt, et høyreattributt og en operator.

Gjør slik:

Endre det identifiserte kombineringsobjektet slik at det refererer til minst en triplett. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6362: Alle venstreattributtene i kombineringsnavn blir ikke tolket til en eller flere kolonner i en enkelt tabell.: Forklaring

En metadataobjektregel ble brutt av det identifiserte kombineringsobjektet. Venstreattributtene for en kombineringsnavn må alle tolkes til en kolonne eller kolonnene i en enkelt databasetabell.

Gjør slik:

Endre det identifiserte kombineringsobjektet slik at alle venstreattributtene tolkes til en kolonne eller kolonnene i en enkelt tabell. Du kan også endre venstreattributtene for det identifiserte kombineringsobjektet slik at de alle oppfyller metadataregelen som er oppgitt. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6363: Alle høyreattributtene i kombineringsnavn blir ikke tolket til en eller flere kolonner i en enkelt tabell.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte kombineringsobjektet. Høyreattributtene for en kombineringsobjekt må alle tolkes til en kolonne eller kolonnene i en enkelt databasetabell.

Gjør slik:

Endre det identifiserte kombineringsobjektet slik at alle høyreattributtene tolkes til en kolonne eller kolonnene i en enkelt tabell. Du kan også endre høyreattributtene for det identifiserte kombineringsobjektet slik at de alle oppfyller metadataregelen som er oppgitt. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6364: Minst en av triplettene for kombineringsobjektet *kombineringsnavn* definerer ikke en gyldig operasjon. Datatypene for venstre- og høyreattributtene er kanskje ikke kompatible med hverandre, eller de er kanskje ikke kompatible med operatoren.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte kombineringsobjektet. Hver triplett i et kombineringsobjekt må definere en gyldig operasjon. Datatypene for høyre- og venstreattributtene må være kompatible med hverandre når en tar hensyn til operasjonen som er oppgitt.

Gjør slik:

Endre det identifiserte kombineringsobjektet slik at hver triplett definerer en gyldig operasjon. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6365: Kubemodellen *modellnavn* refererer ikke til ett og bare ett faktaobjekt.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte kubemodellobjektet. En fullstendig kubemodell må referere til ett faktaobjekt.

Gjør slik:

Endre det identifiserte kubemodellobjektet slik at det refererer til ett faktaobjekt. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6366: Kubemodellen *modellnavn* refererer ikke til en eller flere dimensjoner.:

Forklaring

En metadataobjektregel ble brutt av det identifiserte kubemodellobjektet. En fullstendig kubemodell må referere til minst ett dimensjonsobjekt.

Gjør slik:

Endre det identifiserte kubemodellobjektet slik at det refererer til minst ett dimensjonsobjekt. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6367: Kardinaliteten for kombineringsobjektene og dimensjoner, kombineringsnavn, er ikke definert til enten 1:1 eller N:1.

Forklaring

Kubemodellen kan ikke dra nytte av anbefalingene fra optimaliseringsrådgiveren, fordi kardinaliteten for kombineringsobjektene og dimensjoner ikke er 1:1 eller N:1. Optimalisering blir ikke utført.

Gjør slik:

Hvis kubemodellen skal dra nytte av anbefalingene fra optimaliseringsrådgiveren, må kardinaliteten for hver av kombineringsobjektene som går fra faktaobjektet til et dimensjonsobjekt, være enten 1:1 eller N:1. Kardinaliteten for kombineringsobjektene på faktaattributtene må være 1 eller N, og kardinaliteten for dimensjonens attributter må være 1. Installerings- og brukerhåndboken inneholder flere opplysninger om optimaliseringsregler. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6368: Kombineringsobjektene og dimensjoner, kombineringsnavn kombinerer ikke tabellen for faktaobjektet *faktabavn* med en primærtabell for dimensjonen *dimensjonsnavn*.

Forklaring

En optimaliseringsregel ble brutt av det identifiserte kombineringsobjektet. Når du tar hensyn til kombineringsnettverket som dannes av dimensjonens kombineringsobjektene, må du ha minst en tabell (den primære tabellen) der alle kombineringsobjektene som stråler ut fra denne tabellen, har en kardinalitet på N:1 eller 1:1. I kubemodellen må kombineringsobjektene fra fakta- til dimensjonsobjektene omfatte denne primære tabellen for en dimensjon.

Gjør slik:

I kubemodellobjektet må du sørge for at alle kombineringsobjektene fra fakta- til dimensjonsobjektene er fra faktaobjektet til den primære tabellen for hver dimensjon. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6369: Dimensjonen *dimensjonsnavn* har ikke en primærtabell, slik det er oppgitt av kombineringsnettverket som er dannet av kombineringsobjektene for dimensjonen.

Forklaring

En optimaliseringsregel ble brutt av det identifiserte dimensjonsobjektet. Når du tar hensyn til kombineringsnettverket som dannes av dimensjonens kombineringsobjektene, må du ha minst en tabell der alle kombineringsobjektene som stråler ut fra denne tabellen, har en kardinalitet på N:1 eller 1:1. Optimalisering blir ikke utført hvis det ikke finnes en slik primær tabell for en dimensjon.

Gjør slik:

Kontroller kardinaliteten for kombineringsobjektene som brukes i dimensjonen. Hvis optimalisering skal utføres, må dimensjonen ha en primær tabell slik det er beskrevet i optimaliseringsreglene. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6370: Kombineringsobjektene *kombineringsnavn* omfatter kolonner som det ikke er definert en referansebegrensning for.

Forklaring

En optimaliseringsregel er blitt brutt av det identifiserte kombineringsobjektet. Det må være definert en begrensning på kolonnene som deltar i kombineringsregelen. Hvis kombineringsregelen er en egenkombinering, dvs. at det samme settet av kolonner brukes på begge sider av likhetstegnet, må det defineres en primærnøkkel som samsvarer med settet av kolonner. I alle andre tilfeller der settet med kolonner på den ene siden er forskjellig fra den andre siden av kombineringsregelen, må en primærnøkkel samsvare med kolonnene på den ene siden av kombineringsregelen, og en fremmednøkkel må samsvare med det andre settet med kolonner, og referere til primærnøkkelens. Optimalisering blir ikke utført på grunn av den manglende begrensningen.

Gjør slik:

Opprett en begrensning for kolonnene som deltar i kombineringsregelen. Hvis du ikke vil bruke standardbegrensningen av hensyn til ytelsen, kan du opprette informasjonsbegrensninger med spørreoptimalisering aktivert. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6371: Det er ikke definert en primærnøkkel ved hjelp av kolonnene som er omfattet av egenkombineringen *kombineringsnavn.:*

Forklaring

En optimaliseringsregel ble brutt av det identifiserte kombineringsobjektet. Du må definere en begrensning for kolonnene som deltar i kombineringsregelen. Hvis kombineringsregelen er en egenkombinering, dvs. at det samme settet av kolonner brukes på begge sider av likhetstegnet, må det defineres en primærnøkkel som samsvarer med settet av kolonner. Optimalisering blir ikke utført på grunn av den manglende begrensningen.

Gjør slik:

Hvis tabellen har en primærnøkkel definert, kan du sette attributtene for egenkombineringen til attributtene som representerer primærnøkkelkolonnene i tabellen. Ellers må du opprette en primærnøkkel for kolonnene som deltar i egenkombineringen. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6372: Det er ikke definert en primærnøkkel ved hjelp av kolonnene fra en side av kombineringsregelen *kombineringsnavn.:*

Forklaring

En optimaliseringsregel ble brutt av det identifiserte kombineringsobjektet. Du må definere en begrensning for kolonnene som deltar i kombineringsregelen. Når settet med kolonner på den ene siden er forskjellig fra den andre siden av kombineringsregelen, må en primærnøkkel samsvare med kolonnene på den ene siden av kombineringsregelen, og en fremmednøkkel må samsvare med det andre settet med kolonner og referere til primærnøkkelens. Optimalisering blir ikke utført på grunn av den manglende begrensningen.

Gjør slik:

Opprett en primærnøkkel for kolonnene på den ene siden av kombineringsregelen. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6373: Det er ikke definert en fremmednøkkel ved hjelp av kolonnene fra en side av kombineringsnavn.

Forklaring

En optimaliseringsregel ble brutt av det identifiserte kombineringsobjektet. Du må definere en begrensning for kolonnene som deltar i kombineringsregelen. Når settet med kolonner på den ene siden er forskjellig fra den andre siden av kombineringsregelen, må en primærnøkkel samsvare med kolonnene på den ene siden av kombineringsregelen, og en fremmednøkkel må samsvare med det andre settet med kolonner og referere til primærnøkkelkolonnen. Optimalisering blir ikke utført på grunn av den manglende begrensningen.

Gjør slik:

Opprett en fremmednøkkelbegrensning mellom primærnøkkelkolonnene for kombineringsregelen og kolonnene på den andre siden av kombineringsregelen. Hvis du ikke vil bruke standardbegrensningen av hensyn til ytelsen, kan du opprette informasjonsbegrensninger med spørreoptimalisering aktivert. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6374: Fremmednøkkelkolonnen som bruker kolonnene fra en side av kombineringsnavn refererer ikke til primærnøkkelkolonnen ved hjelp av kolonnene fra den andre siden av kombineringsregelen.

Forklaring

En optimaliseringsregel ble brutt av det identifiserte kombineringsobjektet. Du må definere en begrensning for kolonnene som deltar i kombineringsregelen. Når settet med kolonner på den ene siden er forskjellig fra den andre siden av kombineringsregelen, må en primærnøkkel samsvare med kolonnene på den ene siden av kombineringsregelen, og en fremmednøkkel må samsvare med det andre settet med kolonner og referere til primærnøkkelkolonnen. Optimalisering blir ikke utført på grunn av den manglende begrensningen.

Gjør slik:

Opprett en fremmednøkkelbegrensning mellom primærnøkkelkolonnene for kombineringsregelen og kolonnene på den andre siden av kombineringsregelen. Hvis du ikke vil bruke standardbegrensningen av hensyn til ytelsen, kan du opprette informasjonsbegrensninger med spørreoptimalisering aktivert. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6375: Kardinaliteten for kombineringsnavn er ikke definert til 1:1, N:1 eller 1:N.

Forklaring

En optimaliseringsregel ble brutt av det identifiserte kombineringsobjektet. Optimalisering kan ikke utføres når kardinaliteten for kombineringsregelen er M:N.

Gjør slik:

Sett kardinaliteten for kombineringsregelen til 1:1, 1:N eller N:1, avhengig av hvilke begrensninger kombineringsregelen er basert på. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6376: Kardinaliteten for egenkombineringsnavn er ikke definert til 1:1.

Forklaring

En optimaliseringsregel ble brutt av det identifiserte kombineringsobjektet. Optimalisering kan ikke utføres når kombineringskardinaliteten for en egenkombinering ikke er definert til 1:1.

Gjør slik:

Definer kardinaliteten for egenkombineringen til 1:1. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6377: Kardinaliteten for kombineringen *kombineringsnavn* er ikke definert til 1 for den som den som primærnøkkelen er definert for.:

Forklaring

En optimaliseringsregel ble brutt av det identifiserte kombineringsobjektet. Kardinaliteten for kombineringen må være 1 på den siden der det er definert en primærnøkkel, og N på den siden der det er definert en fremmednøkkel. Hvis fremmednøkkelsiden også har en definert primærnøkkel, må kardinaliteten være 1. Optimalisering kan ikke utføres hvis dette ikke er tilfelle.

Gjør slik:

Kardinaliteten for kombineringen må settes til 1 for den siden som primærnøkkelen er definert for. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6378: Kardinaliteten for kombineringen *kombineringsnavn* er ikke satt til N for den siden som fremmednøkkelen er definert for.:

Forklaring

En optimaliseringsregel ble brutt av det identifiserte kombineringsobjektet. Kardinaliteten for kombineringen må være 1 på den siden der det er definert en primærnøkkel, og N på den siden der det er definert en fremmednøkkel. Hvis fremmednøkkelsiden også har en definert primærnøkkel, må kardinaliteten være 1. Optimalisering kan ikke utføres hvis dette ikke er tilfelle.

Gjør slik:

Kardinaliteten for kombineringen må settes til N for den siden som fremmednøkkelen er definert for. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6379: Kardinaliteten for kombineringen *kombineringsnavn* er ikke definert til 1 for den siden som det er definert både en primærnøkkel og en fremmednøkkel for.:

Forklaring

En optimaliseringsregel ble brutt av det identifiserte kombineringsobjektet. Kardinaliteten for kombineringen må være 1 på den siden der det er definert en primærnøkkel, og 1 på den siden der det er definert både en primærnøkkel og en fremmednøkkel. Optimalisering kan ikke utføres hvis dette ikke er tilfelle.

Gjør slik:

Kardinaliteten for kombineringsregler må settes til 1:1. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6380: Attributtet *attributtnavn*, som det refereres til av kombineringsregler, tolkes ikke til et SQL-uttrykk som ikke kan inneholde nullverdier.:

Forklaring

En optimaliseringsregel ble brutt av det identifiserte kombineringsobjektet. Alle attributter som brukes i kombineringsregler, må tolkes til SQL-uttrykk som ikke kan ha nullverdier. Optimalisering kan ikke utføres hvis en kombineringsregel refererer til et attributt som tolkes til et SQL-uttrykk som kan inneholde nullverdier.

Gjør slik:

Fjern referansen til attributtet som kan inneholde nullverdier, fra kombineringsregler. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6381: Kombineringsregler *kombineringsnavn* har ikke en type av intern kombineringsregel.:

Forklaring

En optimaliseringsregel ble brutt av det identifiserte kombineringsobjektet. Kombineringsreglertypen må defineres som en intern kombineringsregel (INNER JOIN). Optimalisering kan ikke utføres.

Gjør slik:

Endre kombineringsregler slik at den bare refererer til attributter som tolkes til en enkelt kolonne. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6382: Attributtreferansen *attributtnavn* for kombineringsregler *kombineringsnavn* tolkes ikke til ett enkelt kolonneuttrykk, som er obligatorisk for deltakelsen i en begrensning.:

Forklaring

En optimaliseringsregel ble brutt av det identifiserte kombineringsobjektet. DB2-begrensninger må brukes på attributtene det refereres til av en kombineringsregel. Begrensninger kan bare brukes på kolonner, derfor må attributtene det refereres til av en kombineringsregel, tolkes til en enkelt kolonne i en tabell. Optimalisering kan ikke utføres hvis dette ikke er tilfelle.

Gjør slik:

Endre kombineringsregler slik at den bare refererer til attributter som tolkes til en enkelt kolonne. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6385: Kubemodellen *modellnavn* må referere til minst en dimensjon som har et hierarki.:

Forklaring

Den identifiserte kubemodellen bryter optimaliseringsregelen som krever at en kubemodell må referere til minst en dimensjon som har et hierarki.

Gjør slik:

Endre den identifiserte kubemodellens dimensjon slik at dimensjonen refererer til minst ett hierarki. Du finner flere opplysninger om optimaliseringsregler i *Veiledning og referanse for DB2 Cube Views*.

6386: Hver optimaliseringssektor må ha nøyaktig ett optimaliseringsnivå definert per kubedimensjon i kuben *kubenavn*.

Forklaring

Kuben oppfyller ikke regelen om at en optimaliseringssektor må ha nøyaktig ett optimaliseringsnivå per kubedimensjon i kuben.

Gjør slik:

Endre optimaliseringssektoren slik at den refererer til ett optimaliseringsnivå per kubedimensjon i kubeobjektet. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6387: Optimaliseringsnivået må referere til en kubedimensjon i kuben *kubenavn*.

Forklaring

Kuben oppfyller ikke metadataobjektregelen om at en optimaliseringssektor må referere til nøyaktig en kubedimensjon som tilhører kuben.

Gjør slik:

Endre optimaliseringsnivået slik at det refererer til en kubedimensjon i kubeobjektet. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6388: Optimaliseringsnivået må referere til ett kubehierarki i kuben *kubenavn*.

Forklaring

Kuben oppfyller ikke metadataobjektregelen om at et optimaliseringsnivå må referere til nøyaktig ett kubehierarki som tilhører kuben.

Gjør slik:

Endre optimaliseringsnivået slik at det refererer til en kubedimensjon og ett kubehierarki i kubeobjektet.

6389: Du må sette optimaliseringsnivået til *allLevel*, *anyLevel* eller til en kubnivåreferanse i kuben *kubenavn*.

Forklaring

Kuben oppfyller ikke metadataobjektregelen om at et optimaliseringsnivå må ha en *allLevel*-, en *anyLevel*- eller en kubnivåreferanse.

Gjør slik:

Endre det identifiserte optimaliseringsnivået slik at det refererer til en allLevel-, en anyLevel- eller en kubenivåreferanse i kubeobjektet. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6390: I optimaliseringsnivået må kubedimensjonen og kubehierarkiet det refereres til, utledes fra objekter i kuben *kubenaavn*. Kubehierarkiet må tilhøre kubedimensjonen.:

Forklaring

Kuben oppfyller ikke metadataobjektregelen om at kubedimensjonen og kubehierarkiet det refereres til, må utledes fra objektene i kuben for optimaliseringsnivået. Kubehierarkiet må tilhøre kubedimensjonen.

Gjør slik:

Endre optimaliseringsnivået slik at kubedimensjonen og kubehierarkiet det refereres til, utledes fra objekter i kuben. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6391: Hvis kubenivåreferansen i optimaliseringsnivået ikke er anyLevel eller allLevel, må kubenivået *nivånaavn* tilhøre kubehierarkiet *hierarkinaavn* i kuben *kubenaavn*.:

Forklaring

Kuben oppfyller ikke metadataobjektregelen i optimaliseringsnivået, at hvis kubenivåreferansen ikke er anyLevel eller allLevel, må kubenivået tilhøre kubehierarkiet.

Gjør slik:

Endre optimaliseringsnivået slik at det refererte kubenivået tilhører kubehierarkiet. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6392: Du kan ikke ha både en optimaliseringssektor for MOLAP-uttrekking og en optimaliseringssektor for hybriduttrekking i kuben *kubenaavn*.:

Forklaring

Kuben oppfyller ikke metadataobjektregelen om at en kube ikke kan ha både en optimaliseringssektor for MOLAP-uttrekking og en optimaliseringssektor for hybriduttrekking.

Gjør slik:

Endre en av optimaliseringssektorene slik at du ikke har både en optimaliseringssektor for MOLAP-uttrekking og en optimaliseringssektor for hybriduttrekking i en kube. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6393: Du kan ikke ha mer enn en optimaliseringssektor for MOLAP-uttrekking i kuben *kubenaavn*.:

Forklaring

Kuben oppfyller ikke metadataobjektregelen om at en kube ikke kan ha flere enn en optimaliseringssektor av typen MOLAP-uttrekking.

Gjør slik:

Endre kuben slik at den har null eller en optimaliseringssektor av typen MOLAP-uttrekking. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6394: Du kan ikke ha mer enn en optimaliseringssektor for hybriduttrekking i kuben *kubenavn*.

Forklaring

Kuben oppfyller ikke metadataobjektregelen om at en kube ikke kan ha flere enn en optimaliseringssektor av typen hybriduttrekking.

Gjør slik:

Endre kuben slik at den har null eller en optimaliseringssektorer av typen hybriduttrekking. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6395: Optimaliseringssektoren for gjennomdrilling kan bare eksistere hvis det finnes en optimaliseringssektor for hybriduttrekking i kuben *kubenavn*.

Forklaring

Kuben oppfyller ikke metadataobjektregelen om at en optimaliseringssektor av typen gjennomdrilling bare kan eksistere hvis det finnes en optimaliseringssektor for hybriduttrekking i kuben.

Gjør slik:

Endre typen optimaliseringssektor fra gjennomdrilling til en annen type hvis du ikke har en optimaliseringssektor av typen hybriduttrekking i kuben. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6396: Attributtet *attributtnavn* som det refereres til av nivået *nivånavn* i dimensjonen *dimensjonsnavn*, må være med på dimensjonens attributtliste.

Forklaring

Dimensjonen oppfyller ikke metadataobjektregelen om at alle attributter det blir referert til i nivåene i en dimensjon, må være inkludert i dimensjonens attributtliste.

Gjør slik:

Endre nivået slik at det ikke refererer til det identifiserte attributtet, eller tilføy det identifiserte attributtet til dimensjonen. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6397: Nivået *nivånavn* må referere til minst ett nivånøkkelattributt.

Forklaring

Nivået oppfyller ikke metadataobjektregelen om at et nivå må referere til minst ett nivånøkkelattributt.

Gjør slik:

Endre nivået slik at det refererer til minst ett nivånøkkelattributt. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6398: Nivået *nivånavn* kan ikke inneholde attributtet *attributtnavn* mer enn en gang i settet av nivånøkkelattributter.:

Forklaring

Nivået oppfyller ikke metadataobjektregelen om at settet med nivånøkkelattributter ikke kan inneholde duplikate attributter.

Gjør slik:

Endre nivået slik at det ikke inneholder duplikate nivånøkkelattributter. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6399: Nivået *nivånavn* må referere til nøyaktig ett standardattributt.:

Forklaring

Nivået oppfyller ikke metadataobjektregelen om at et nivå må ha nøyaktig ett standardattributt.

Gjør slik:

Endre nivået slik at det inneholder nøyaktig ett standardattributt. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6400-6499

6401: Du kan ikke bruke ett attributt som både standardattributt og et beslektet attributt i nivået *nivånavn*.:

Forklaring

Nivået oppfyller ikke metadataregelen om at ett attributt ikke kan brukes som både standardattributt og et beslektet attributt.

Gjør slik:

Endre nivået slik at standardattributtet ikke blir brukt som et beslektet attributt. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6402: Du kan ikke bruke ett attributt som både standardattributt og et beslektet attributt i nivået *nivånavn*.:

Forklaring

Nivået oppfyller ikke metadataobjektregelen om at settet med beslektede attributter ikke kan inneholde duplikate attributter.

Gjør slik:

Endre nivået slik at det ikke inneholder duplikate beslektede attributter. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6403: Nivået *nivånavn* kan ikke innholdet attributtet *attributtnavn* mer enn en gang i settet med beslektede attributter.:

Forklaring

Kubenivået oppfyller ikke metadataobjektregelen om at et kubenivå må referere til nøyaktig ett nivå.

Gjør slik:

Endre kubenivået slik at det refererer til nøyaktig ett nivå. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6404: Kubenivået *nivånavn* må referere til nøyaktig ett nivå.:

Forklaring

Kubenivået oppfyller ikke metadataobjektregelen om at alle de relaterte attributtene i et kubenivå også må være beslektede attributter for det tilhørende nivået.

Gjør slik:

Endre kubenivået slik at det refererer til attributter som også blir referert til av det overordnede nivået. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6405: Kubenivået *nivånavn* kan ikke inneholde attributtet *attributtnavn* mer enn en gang i settet med beslektede attributter.:

Forklaring

Kubenivået oppfyller ikke metadataobjektregelen om at settet med beslektede attributtet ikke kan inneholde duplikate attributter.

Gjør slik:

Endre kubenivået slik at det ikke inneholder duplikate beslektede attributter. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

Referansebegrensningsrelaterte feil

6500: Denne operasjonen kan ikke utføres fordi SQL-malen for attributtet *attributtnavn* eller målet fremdeles omfatter referanser til andre attributter, mål eller kolonner. Disse referansene må slettes før denne operasjonen kan utføres.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. SQL-uttrykksmalen for det identifiserte objektet omfatter referanser til andre attributter, mål eller kolonner som må fjernes fra det identifiserte objektet før utføringen av denne operasjonen.

Gjør slik:

Før du sletter det identifiserte objektet, må du endre det identifiserte objektet slik at SQL-uttrykksmalen ikke lenger refererer til attributter, mål eller kolonner. Du finner flere opplysninger om metadatar regler i *Veiledning og referanse for DB2 Cube Views*.

6501: Operasjonen kan ikke utføres fordi attributtet *attributtnavn* eller målet er referert til av et annet attributt eller mål.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det identifiserte attributtet eller målet er referert til av et annet attributt eller mål, og derfor kan det identifiserte attributtet eller målet ikke slettes.

Gjør slik:

Før du sletter det identifiserte attributtet eller målet, må du endre objektene som refererer til det, slik at de ikke lenger refererer til det identifiserte attributtet eller målet. Du finner flere opplysninger om metadatar regler i *Veiledning og referanse for DB2 Cube Views*.

6502: Operasjonen kan ikke utføres fordi dimensjonen *dimensjonsnavn* er referert til av en samling som er definert i et mål.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til den identifiserte dimensjonen av en samling av et mål, og derfor kan den identifiserte dimensjonen ikke slettes.

Gjør slik:

Før du sletter den identifiserte dimensjonen, må du endre objektene som refererer til den, slik at de ikke lenger refererer til den identifiserte dimensjonen. Du finner flere opplysninger om metadatar regler i *Veiledning og referanse for DB2 Cube Views*.

6503: Operasjonen kan ikke utføres for objektet *objektnavn*. Et kubehierarki må referere til attributter som det allerede refereres til av hierarkiet som ble brukt til å utlede kubehierarkiet.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Problemet skyldes en av disse situasjonene:

- Det ble gjort forsøk på å fjerne et attributt fra et hierarki når attributtet som skulle fjernes, fortsatt brukes av et beslektet kubehierarki.
- Det ble gjort forsøk på å tilføye et attributt til et kubehierarki når attributtet som skulle tilføyes, ikke allerede brukes av et beslektet hierarki.

Gjør slik:

Utfør en av disse handlingene:

- Fjern attributter fra kubehierarkier før du fjerner de samme attributtene fra beslektede hierarkier.
- Tilføy attributter til hierarkier før du tilføyer de samme attributtene til beslektede kubehierarkier.

Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6504: Operasjonen kan ikke utføres for objektet *objektnavn*. Et kubehierarki må referere til attributtforhold som det allerede refereres til av hierarkiet som ble brukt til å utlede kubehierarkiet.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Problemet skyldes en av disse situasjonene:

- Det ble gjort forsøk på å fjerne et attributtforhold fra et hierarki når attributtforholdet som skulle fjernes, fortsatt brukes av et beslektet kubehierarki.
- Det ble gjort forsøk på å tilføye et attributtforhold til et kubehierarki når attributtforholdet som skulle tilføyes, ikke allerede brukes av et beslektet hierarki.

Gjør slik:

Utfør en av disse handlingene:

- Fjern attributtforhold fra kubehierarkier før du fjerner de samme attributtforholdene fra beslektede hierarkier.
- Tilføy attributtforhold til hierarkier før du tilføyer de samme attributtforholdene til beslektede kubehierarkier.

Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6505: Operasjonen kan ikke utføres fordi hierarkiet *hierarkinavn* er referert til av et kubehierarki.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til det identifiserte hierarkiet av et kubehierarki, og derfor kan det identifiserte hierarkiet ikke slettes.

Gjør slik:

Før du sletter det identifiserte hierarkiet, må du endre objektene som refererer til det, slik at de ikke lenger refererer til det identifiserte hierarkiet. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6506: Operasjonen kan ikke utføres for objektet *objektnavn*. Et kubefaktaobjekt må referere til mål som det allerede refereres til av faktaobjektene som ble brukt til å utlede kubefaktaobjektene.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Problemet skyldes en av disse situasjonene:

- Det ble gjort forsøk på å fjerne et mål fra et faktaobjekt når målet som skulle fjernes, fortsatt brukes av et beslektet kubefaktaobjekt.
- Det ble gjort forsøk på å tilføye et mål til et kubefaktaobjekt når målet som skulle tilføyes, ikke allerede brukes av et beslektet faktaobjekt.

Gjør slik:

Utfør en av disse handlingene:

- Fjern mål fra kubefaktaobjekter før du fjerner det samme målet fra beslektede faktaobjekter.
- Tilføy mål til faktaobjekter før du tilføyer det samme målet til beslektede kubefaktaobjekter.

Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6507: Operasjonen kan ikke utføres fordi kubnivået *faktanavn* er referert til av et kubefaktaobjekt.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til det identifiserte faktaobjektet av et kubefaktaobjekt, og derfor kan det identifiserte faktaobjektet ikke slettes.

Gjør slik:

Før du sletter det identifiserte faktaobjektet, må du endre objektene som refererer til det, slik at de ikke lenger refererer til det identifiserte faktaobjektet. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6508: Operasjonen kan ikke utføres fordi hierarkiet *hierarkinavn* er referert til av en dimensjon.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til det identifiserte hierarkiet av en dimensjon, og derfor kan det identifiserte hierarkiet ikke slettes.

Gjør slik:

Før du sletter det identifiserte hierarkiet, må du endre objektene som refererer til det, slik at de ikke lenger refererer til det identifiserte hierarkiet. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6509: Operasjonen kan ikke utføres fordi kombineringsnavnet *kombineringsnavn* er referert til av et faktaobjekt.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til den identifiserte kombineringsen av et faktaobjekt, og derfor kan den identifiserte kombineringsen ikke slettes.

Gjør slik:

Før du sletter den identifiserte kombineringsen, må du endre objektene som refererer til den, slik at de ikke lenger refererer til den identifiserte kombineringsen. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6510: Operasjonen kan ikke utføres fordi kubedimensjonen *kubedimensjonsnavn* er referert til av en kube.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til den identifiserte kubedimensjonen av en kube, og derfor kan den identifiserte kubedimensjonen ikke slettes.

Gjør slik:

Før du sletter den identifiserte kubedimensjonen, må du endre objektene som refererer til den, slik at de ikke lenger refererer til den identifiserte kubedimensjonen. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6511: Operasjonen kan ikke utføres for objektet *objektnavn*. Kubedimensjonene for en kube må være utledet fra dimensjonen som det refereres til av kubemodellen som kuben var utledet fra.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Problemet skyldes en av disse situasjonene:

- Det ble gjort forsøk på å fjerne en dimensjon fra en kubemodell når dimensjonen som skulle fjernes, fortsatt brukes av en beslektet kubens kubedimensjon.
- Det ble gjort forsøk på å tilføye en kubedimensjon til en kube når dimensjonen for kubedimensjonen som skulle tilføyes, ikke allerede brukes av en beslektet kubemodell.

Gjør slik:

Utfør en av disse handlingene:

- Fjern kubedimensjoner fra kuber før du fjerner beslektede dimensjoner fra beslektede kubemodeller.
- Tilføy dimensjoner til kubemodeller før du tilføyer beslektede kubedimensjoner til beslektede kuber.

Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6512: Operasjonen kan ikke utføres fordi dimensjonen *dimensjonsnavn* er referert til av en kubedimensjon.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til den identifiserte dimensjonen av en kubedimensjon, og derfor kan den identifiserte dimensjonen ikke slettes.

Gjør slik:

Før du sletter den identifiserte dimensjonen, må du endre objektene som refererer til den, slik at de ikke lenger refererer til den identifiserte dimensjonen. Du finner flere opplysninger om metadatareregler i *Veiledning og referanse for DB2 Cube Views*.

6513: Operasjonen kan ikke utføres for objektet *objektnavn*. En kubedimensjons kubehierarki må være utledet fra hierarkiet som det refereres til av den samme dimensjonen som den som ble brukt til å utlede kubedimensjonen.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Problemet skyldes en av disse situasjonene:

- Det ble gjort forsøk på å fjerne et hierarki fra en dimensjon når hierarkiet som skulle fjernes, fortsatt brukes av en beslektet kubedimensjons kubehierarki.
- Det ble gjort forsøk på å tilføye et kubehierarki til en kubedimensjon når hierarkiet for kubehierarkiet som skulle tilføyes, ikke allerede brukes av en beslektet dimensjon.

Gjør slik:

Utfør en av disse handlingene:

- Fjern kubehierarkier fra kubedimensjoner før du fjerner beslektede hierarkier fra beslektede dimensjoner.
- Tilføy hierarkier til dimensjoner før du tilføyer beslektede kubehierarkier til beslektede kubedimensjoner.

Du finner flere opplysninger om metadatareregler i *Veiledning og referanse for DB2 Cube Views*.

6514: Operasjonen kan ikke utføres fordi kubehierarkiet *kubehierarkinavn* er referert til av en kubedimensjon.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til det identifiserte kubehierarkiet av en kubedimensjon, og derfor kan det identifiserte kubehierarkiet ikke slettes.

Gjør slik:

Før du sletter det identifiserte kubehierarkiet, må du endre objektene som refererer til det, slik at de ikke lenger refererer til det identifiserte kubehierarkiet. Du finner flere opplysninger om metadatareregler i *Veiledning og referanse for DB2 Cube Views*.

6515: Operasjonen kan ikke utføres for objektet *objektnavn*. En kubedimensjons kubehierarki må være utledet fra hierarkiet som det refereres til av den samme dimensjonen som den som ble brukt til å utlede kubedimensjonen.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Problemet skyldes en av disse situasjonene:

- Det ble gjort forsøk på å fjerne et hierarki fra en dimensjon når hierarkiet som skulle fjernes, fortsatt brukes av en beslektet kubedimensjons kubehierarki.
- Det ble gjort forsøk på å tilføye et kubehierarki til en kubedimensjon når hierarkiet for kubehierarkiet som skulle tilføyes, ikke allerede brukes av en beslektet dimensjon.

Gjør slik:

Utfør en av disse handlingene:

- Fjern kubehierarkier fra kubedimensjoner før du fjerner beslektede hierarkier fra beslektede dimensjoner.
- Tilføy hierarkier til dimensjoner før du tilføyer beslektede kubehierarkier til beslektede kubedimensjoner.

Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6516: Operasjonen kan ikke utføres fordi kombineringsnavn er referert til av en dimensjon.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til den identifiserte kombineringsnavn av en dimensjon, og derfor kan den identifiserte kombineringsnavn ikke slettes.

Gjør slik:

Før du sletter den identifiserte kombineringsnavn, må du endre objektene som refererer til den, slik at de ikke lenger refererer til den identifiserte kombineringsnavn. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6517: Operasjonen kan ikke utføres fordi attributtet *attributtnavn* er referert til av en dimensjon.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til det identifiserte attributtet av en dimensjon, og derfor kan det identifiserte attributtet ikke slettes.

Gjør slik:

Før du sletter det identifiserte attributtet, må du endre objektene som refererer til det, slik at de ikke lenger refererer til det identifiserte attributtet. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6518: Operasjonen kan ikke utføres fordi attributtet *attributtnavn* er referert til av et hierarki.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til det identifiserte attributtet av et hierarki, og derfor kan det identifiserte attributtet ikke slettes.

Gjør slik:

Før du sletter det identifiserte attributtet, må du endre objektene som refererer til det, slik at de ikke lenger refererer til det identifiserte attributtet. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6519: Operasjonen kan ikke utføres fordi attributtforholdet *forholdsnavn* er referert til av et hierarki.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til det identifiserte attributtforholdet av et hierarki, og derfor kan det identifiserte attributtforholdet ikke slettes.

Gjør slik:

Før du sletter det identifiserte attributtforholdet, må du endre objektene som refererer til det, slik at de ikke lenger refererer til det identifiserte attributtforholdet. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6520: Operasjonen kan ikke utføres fordi dimensjonen *dimensjonsnavn* er referert til av en kubemodell.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til den identifiserte dimensjonen av en kubemodell, og derfor kan det identifiserte dimensjonsforholdet ikke slettes.

Gjør slik:

Før du sletter den identifiserte dimensjonen, må du endre objektene som refererer til den, slik at de ikke lenger refererer til den identifiserte dimensjonen. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6521: Operasjonen kan ikke utføres fordi kombineringsnavnet *kombineringsnavn* er referert til av en kubemodell.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til den identifiserte kombineringsnavnet av en kubemodell, og derfor kan den identifiserte kombineringsnavnet ikke slettes.

Gjør slik:

Før du sletter den identifiserte kombineringsen, må du endre objektene som refererer til den, slik at de ikke lenger refererer til den identifiserte kombineringsen. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6522: Operasjonen kan ikke utføres fordi objektet *objektnavn* er referert til av et faktaobjekt.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til det identifiserte objektet av et faktaobjekt, og derfor kan det identifiserte objektet ikke slettes.

Gjør slik:

Før du sletter det identifiserte objektet, må du endre objektene som refererer til den, slik at de ikke lenger refererer til det identifiserte objektet. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6523: Operasjonen kan ikke utføres fordi venstreattributtet *attributtnavn* er referert til av et attributtforhold.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til det identifiserte attributtet av et attributtforhold, og derfor kan det identifiserte attributtet ikke slettes.

Gjør slik:

Før du sletter det identifiserte attributtet, må du endre objektene som refererer til det, slik at de ikke lenger refererer til det identifiserte attributtet. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6524: Operasjonen kan ikke utføres fordi høyreattributtet *attributtnavn* er referert til av et attributtforhold.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til det identifiserte attributtet av et attributtforhold, og derfor kan det identifiserte attributtet ikke slettes.

Gjør slik:

Før du sletter det identifiserte attributtet, må du endre objektene som refererer til det, slik at de ikke lenger refererer til det identifiserte attributtet. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6525: Operasjonen kan ikke utføres fordi høyreattributtet *attributtnavn* er referert til av en kombineringsen.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til det identifiserte attributtet av en kombinerings, og derfor kan det identifiserte attributtet ikke slettes.

Gjør slik:

Før du sletter det identifiserte attributtet, må du endre objektene som refererer til det, slik at de ikke lenger refererer til det identifiserte attributtet. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6526: Operasjonen kan ikke utføres fordi venstreattributtet *attributtnavn* er referert til av en kombinerings.

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til det identifiserte attributtet av en kombinerings, og derfor kan det identifiserte attributtet ikke slettes.

Gjør slik:

Før du sletter det identifiserte attributtet, må du endre objektene som refererer til det, slik at de ikke lenger refererer til det identifiserte attributtet. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6527: Operasjonen kan ikke utføres fordi kubemodellen *modellnavn* er referert til av en kube.

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til den identifiserte kubemodellen av en kube, og derfor kan den identifiserte kubemodellen ikke slettes.

Gjør slik:

Før du sletter den identifiserte kubemodellen, må du endre objektene som refererer til den, slik at de ikke lenger refererer til den identifiserte kubemodellen. Du finner flere opplysninger om metadataregler i *Veiledning og referanse for DB2 Cube Views*.

6528: Operasjonen kan ikke utføres fordi kubefaktaobjektet *kubefaktanavn* er referert til av en kube.

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til det identifiserte kubefaktaobjektet av en kube, og derfor kan det identifiserte kubefaktaobjektet ikke slettes.

Gjør slik:

Før du sletter det identifiserte kubefaktaobjektet, må du endre objektene som refererer til det, slik at de ikke lenger refererer til det identifiserte kubefaktaobjektet. Du finner flere opplysninger om metadatarregler i *Veiledning og referanse for DB2 Cube Views*.

6529: Operasjonen kan ikke utføres fordi faktaobjektet *faktanavn* er referert til av en kubemodell.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til det identifiserte faktaobjektet av en kubemodell, og derfor kan det identifiserte faktaobjektet ikke slettes.

Gjør slik:

Før du sletter det identifiserte faktaobjektet, må du endre objektene som refererer til det, slik at de ikke lenger refererer til det identifiserte faktaobjektet. Du finner flere opplysninger om metadatarregler og referansebegrensninger mellom metadataobjekter, i *Veiledning og referanse for DB2 Cube Views*.

6530: Operasjonen kan ikke utføres fordi nivået *nivånavn* er referert til av en dimensjon.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til nivået av en dimensjon, og derfor kan nivået ikke slettes.

Gjør slik:

Før du sletter nivået, må du endre objektene som refererer til det, slik at de ikke lenger refererer til nivået. Du finner flere opplysninger om metadatarregler og referansebegrensninger mellom metadataobjekter, i *Veiledning og referanse for DB2 Cube Views*.

6531: Operasjonen kan ikke utføres fordi nivået *nivånavn* er referert til av et hierarki.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til nivået av et hierarki, og derfor kan nivået ikke slettes.

Gjør slik:

Før du sletter nivået, må du endre objektene som refererer til det, slik at de ikke lenger refererer til nivået. Du finner flere opplysninger om metadatarregler og referansebegrensninger mellom metadataobjekter, i *Veiledning og referanse for DB2 Cube Views*.

6532: Operasjonen kan ikke utføres fordi kubnivået *nivånavn* er referert til av et kubehierarki.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til kubnivået av et kubehierarki, og derfor kan kubnivået ikke slettes.

Gjør slik:

Før du sletter nivået, må du endre objektene som refererer til det, slik at de ikke lenger refererer til kubnivået. Du finner flere opplysninger om metadataregler og referansebegrensninger mellom metadataobjekter, i *Veiledning og referanse for DB2 Cube Views*.

6533: Operasjonen kan ikke utføres for kub *kubnavn*. Kubedimensjonen og kubehierarkiet må være utledet fra kub. Kubehierarkiet må tilhøre kubedimensjonen.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Det er referert til kubens kubedimensjon i kubedimensjonen og kubehierarkiet til en optimaliseringssektor, og derfor kan kubens optimaliseringssektor ikke slettes.

Gjør slik:

Før du sletter kubens optimaliseringssektor, må du endre objektene som refererer til den, slik at de ikke lenger refererer til kubens optimaliseringssektor. Du finner flere opplysninger om metadataregler og referansebegrensninger mellom metadataobjekter, i *Veiledning og referanse for DB2 Cube Views*.

6534: Operasjonen kan ikke utføres på nivået *nivånavn*. Det beslektede attributtet kan ikke fjernes fra nivået fordi et tilsvarende kubnivå refererer til det beslektede attributtet.:

Forklaring

Den forespurte operasjonen kan ikke utføres fordi den er i strid med en referansebegrensning som finnes mellom metadataobjektene i metadatakatalogen. Du kan ikke slette det beslektede attributtet fra nivået fordi det beslektede attributtet blir referert til av det tilhørende kubnivået.

Gjør slik:

Hvis du vil slette det beslektede attributtet, endrer du kubnivåobjektet som refererer til det, slik at kubnivået ikke refererer til nivåets beslektede attributt som du ønsker å slette. Du finner flere opplysninger om metadataregler og referansebegrensninger mellom metadataobjekter, i *Veiledning og referanse for DB2 Cube Views*.

Optimalisering

Optimaliseringsfeilkoder

7001: Ingen kuber er definert for kubemodellen *modellnavn*.:

Forklaring

Det er ikke definert noen kuber for kubemodellen.

Gjør slik:

Hvis du vil optimalisere for uttrekkings spørringer som leser data fra kubemodellen og legger dem inn i en MOLAP-kube, må du definere kuber som representerer MOLAP-kubene. Du kan ikke optimalisere for uttrekkings spørringer uten å definere en eller flere kuber.

7002: Kubemodellen *modellnavn* finnes ikke.:

Forklaring

Det er ikke definert noen kubemodell med navnet du oppgav.

Gjør slik:

Kontroller at du har oppgitt riktig navn på kubemodell og skjema. Det skilles mellom store og små bokstaver i navn og skjemaer. Vis listen over eksisterende kubemodeller i OLAP-senter.

7003: Tabellplassen *tabellplassnavn* ble ikke funnet.:

Forklaring

Det er ikke definert noen tabellplass med dette navnet.

Gjør slik:

Kontroller at du har oppgitt riktig navn på tabellplassen.

7004: Optimaliseringsrådgiveren kan ikke bestemme anbefalinger.:

Forklaring

Du oppgav en bestemt grense for hvor mye lagerplass som kan brukes, for optimaliseringen av denne kubemodellen. Rådgiveren kunne ikke lage anbefalinger som bruker denne mengden eller mindre lagerplass.

Gjør slik:

Oppgi en høyere grense for lagerplass og kjør veiviseren Optimaliseringsrådgiver på nytt.

7005: Tabellplassen *tabellplassnavn* kan ikke brukes til å lagre sammendragstabellene.:

Forklaring

Tabellplassen har ikke den typen datalager som kreves for å lagre tabelldata. Tabellplassen må være av typen REGULAR. Tabellplasser av typen LONG, USER TEMPORARY og SYSTEM TEMPORARY kan ikke brukes til lagring av sammendragstabeller.

Gjør slik:

Oppgi en tabellplass av typen REGULAR til lagring av sammendragstabellene.

7006: Tabellplassen *tabellplassnavn* kan ikke brukes til å lagre indeksene.:

Forklaring

Tabellplassen har ikke den typen datalager som kreves for å lagre indeksdata. Tabellplassen må være av typen REGULAR eller LONG. Tabellplasser av typen USER TEMPORARY og SYSTEM TEMPORARY kan ikke brukes til lagring av indekser.

Gjør slik:

Oppgi en tabellplass av typen REGULAR eller LONG til lagringen av indeksene.

7007: Optimaliseringsvalideringen av kubemodellen *modellnavn* mislyktes.:

Forklaring

Kubemodellen og de tilknyttede metadataobjektene bryter en eller flere av metadataobjektreglene som kreves oppfylt ved optimalisering. Optimalisering blir ikke utført.

Gjør slik:

Hvis optimalisering skal utføres, må kubemodellen og de tilknyttede metadataobjektene oppfylle metadataobjektreglene for optimalisering. *Installerings- og brukerhåndboken* inneholder flere opplysninger om optimaliseringsregler.

7008: Kubemodellen har ingen dimensjoner med hierarkier som kan optimaliseres.:

Forklaring

Optimalisering blir ikke utført fordi optimaliseringsrådgiveren ikke finner dimensjoner med hierarkier som kan optimaliseres.

Gjør slik:

Kontroller at kubemodellen har minst en dimensjon som har et ikke-rekursivt hierarki.

7009: Du kan ikke oppgi mer enn en optimaliseringssektor av typen MOLAP-uttrekking eller hybriduttrekking, eller både en optimaliseringssektor av typen MOLAP-uttrekking og av typen hybriduttrekking for en kube. Det er oppgitt optimaliseringssektorer av mer enn en uttrekkingstype for kuben *kubenaavn*.:

Forklaring

Den valgte kuben er ikke optimalisert, fordi det er oppgitt optimaliseringssektorer av mer enn en type for kuben. Du kan bare oppgi en optimaliseringssektor av typen MOLAP-uttrekking eller hybriduttrekking per kube.

Gjør slik:

Kontroller at det bare finnes en optimaliseringssektor av typen MOLAP-uttrekking eller hybriduttrekking per kube.

**7010: En optimaliseringssektor av typen neddrilling må være definert på et kubenivå nedenfor hybriduttrekkingssektoren i minst en kubedimensjon.:
Forklaring**

Optimaliseringssektoren av typen neddrilling må være definert på et kubenivå nedenfor hybriduttrekkingssektoren i minst en kubedimensjon.

Gjør slik:

Kontroller at optimaliseringssektoren av typen neddrilling er definert på et kubenivå nedenfor hybriduttrekkingssektoren i minst en kubedimensjon.

**7011: Optimaliseringsrådgiveren ble stoppet og kunne ikke bestemme noen anbefalinger innenfor den tildelte tiden.:
Forklaring**

Optimaliseringsrådgiveren ble stoppet av operasjonen 'stop advise' før den kunne gi noen anbefalinger.

Gjør slik:

La optimaliseringsrådgiveren kjøre lenger, slik at den kan bestemme anbefalinger.

**7012: Ugyldig ID-verdi.:
Forklaring**

Den oppgitte ID-verdien var ugyldig.

**7013: Det ble gjort forsøk på utvelging av data fra DB2-tabellen på et faktaobjekt som er basert på et utsnitt, et kallenavn (alias) eller et annet databaseobjekt som ikke støtter datautvelging (sampling).:
Forklaring**

DB2 Cube Views forsøkte utvelging av data fra et faktaobjekt som er basert på et utsnitt, tilnavn eller kallenavn. DB2-tabellutvelging støttes ikke for utsnitt, tilnavn og kallenavn.

Gjør slik:

DB2 Cube Views kan utføre tabellutvelging på faktaobjekter bare hvis de er definert på tabeller, og ikke definert på utsnitt, tilnavn eller kallenavn. Slå av utvelgingsalternativet for optimaliseringsrådgiveren.

**7014: Optimaliseringsrådgiveren kan ikke bestemme noen anbefalinger.:
Forklaring**

Optimaliseringsrådgiveren kan ikke bestemme noen anbefalinger for kubemodellen med parameterne du oppgav.

Gjør slik:

Kontroller om informasjons- og varselmeldingene inneholder flere opplysninger om hvorfor optimaliseringsrådgiveren ikke kan anbefale noen sammendragstabeller.

Optimaliseringsadvarsler

7200: De anbefalte sammendragstabellene vil bruke utsatt oppdatering fordi kubemodellen inneholder en eller flere ikke-distributive mål:

Forklaring

Alternativet for umiddelbar oppdatering ble valgt for sammendragstabellene. Sammendragstabeller kan imidlertid ikke oppdateres umiddelbart hvis det er definert ikke-distributive mål i kubemodellen. Distributive mål bruker enkle samlingsfunksjoner som SUM og COUNT, og de kan samles fra alle midlertidige verdier. Ikke-distributive mål bruker mer komplekse samlingsfunksjoner som STDDEV, og de må samles fra basistabellene.

Gjør slik:

Hvis det ikke er nødvendig å vedlikeholde sammendragstabellene synkront med basistabellene, trenger du ikke å gjøre noe. Hvis sammendragstabellene må vedlikeholdes synkront med basistabellene, må du endre metadataene slik at bare distributive mål er definert.

7201: Den anbefalte sammendragstabellen *tabellnavn* vil bruke utsatt oppdatering fordi ett eller flere attributter som kan inneholde nullverdier, ble funnet som kolonner i den fulle SELECT-setningen for denne anbefalte sammendragstabellen.:

Forklaring

Alternativet for umiddelbar oppdatering ble valgt for sammendragstabellene. Den anbefalte sammendragstabellen inneholder imidlertid ett eller flere attributter som brukes som kolonner som kan inneholde nullverdier, i sammendragstabellens fulle SELECT-setning. Å bruke kolonner som kan inneholde nullverdier, i en sammendragstabells fulle SELECT-setning kan føre til dårlig ytelse for umiddelbar oppdatering. Sammendragstabellen er definert for *utsatt oppdatering* for å unngå dette ytelsesproblemet.

Gjør slik:

Hvis du skal endre at et attributt kan inneholde nullverdier, må du endre attributtets SQL-uttrykk og/eller endre DB2-tabellkolonnene som brukes av attributtet, slik at de ikke kan inneholde nullverdier. Disse endringene anbefales vanligvis ikke, fordi det kan være vanskelig å implementere dem.

7202: Tabellen *tabellnavn* har ikke statistikk.:

Forklaring

Optimaliseringsrådgiveren kan ikke finne gyldig tabellstatistikk for tabellen som er oppgitt.

Gjør slik:

Opprett statistikk for tabellen som er oppgitt, ved hjelp av RUNSTATS-kommandoen. Kjør deretter veiviseren for optimaliseringsrådgiveren på nytt.

7203: For optimaliseringssektoren av typen hybriduttrekking *sektornavn* i kubene *kubenaavn*, må du oppgi et kubenivå for hver kubedimensjon i kubene.:

Forklaring

Optimaliseringsrådgiveren kan ikke optimalisere for optimaliseringssektoren av typen hybriduttrekking, fordi optimaliseringssektoren ikke er riktig definert. Optimaliseringssektorer av typen hybriduttrekking må ha definert et kubenivå for hver kubedimensjon. Du kan ikke velge Ikke spesifisert.

Gjør slik:

Kontroller at optimaliseringssektoren har et kubenivå oppgitt for hver kubedimensjon i kuben.

7203: For optimaliseringssektoren av typen hybriduttrekking *sektornavn* i kuben *kubenavn*, må du oppgi et kubenivå for hver kubedimensjon i kuben:

Forklaring

Optimaliseringsrådgiveren kan ikke optimalisere for optimaliseringssektoren av typen hybriduttrekking, fordi optimaliseringssektoren ikke er riktig definert. Optimaliseringssektorer av typen hybriduttrekking må ha definert et kubenivå for hver kubedimensjon. Du kan ikke velge Ikke spesifisert.

Gjør slik:

Kontroller at optimaliseringssektoren har et kubenivå oppgitt for hver kubedimensjon i kuben.

7205: De anbefalte sammendragstabellene må bruke oppdateringsmetoden Utsatt fornying fordi en eller flere av de underliggende basistabellene i kubemodellen er et kallenavn.:

Forklaring

Du valgte oppdateringsalternativet Forny umiddelbart for sammendragstabellene, men sammendragstabellene kan ikke bruke oppdateringsmetoden Forny umiddelbart hvis noen av de underliggende tabellene er kallenavn.

Gjør slik:

Du trenger ikke å gjøre noe.

7206: Optimaliseringsrådgiveren kunne ikke bruke dataavlesing til å bestemme de anbefalte sammendragstabellene.:

Forklaring

Du oppgav at optimaliseringsrådgiveren skulle bruke dataavlesing (sampling) for å bestemme optimale anbefalinger av sammendragstabeller.

Optimaliseringsrådgiveren kan ikke utføre dataavlesing på grunn av måten faktaobjektet er definert på. Hvis faktaobjektet er definert som et kallenavn, forsøker optimaliseringsrådgiveren å utføre dataavlesing, men er kanskje ikke i stand til det, med mindre kubemodellens dimensjoner har en moderat kardinalitet, og kombineringsene av faktaobjekter og dimensjoner er på et enkelt attributt som er tilordnet til en enkelt kolonne. Indekser som er definert på faktatabellkolonnene som representerer dimensjonsnøkler, kan også påvirke optimaliseringsrådgiverens evne til å utføre dataavlesing for et faktaobjekt som er definert som et kallenavn. Hvis faktaobjektet er definert som et utsnitt, støttes ikke dataavlesing. Hvis faktaobjektet er definert som et tilnavn, er utvelging av data kanskje ikke mulig, avhengig av hva tilnavnet er tilordnet. Effekten av å opprette anbefalinger uten å

utføre dataavlesing, er at optimaliseringsrådgiveren ikke kan anbefale de optimale sammendragstabellene. Hvis faktaobjektet er en tabell, kan optimaliseringsrådgiveren alltid utføre dataavlesing.

Gjør slik:

Du kan bruke anbefalingene slik de er, eller du kan forsøke å forbedre anbefalingene ved å angi en tabell i stedet for et utsnitt eller et kallenavn for faktaobjektet, og kjøre optimaliseringsrådgiveren på nytt.

Optimalisering, informasjonsmeldinger

7400: Sammendragstabellene er definert ved hjelp av operatoren ROLLUP fordi kubemodellen inneholder ett eller flere ikke-distributive mål.:

Forklaring

Mål kan være distributive eller ikke-distributive. Distributive mål bruker enkle samlingsfunksjoner som SUM og COUNT, og de kan samles fra alle midlertidige verdier. Ikke-distributive mål bruker mer komplekse samlingsfunksjoner som STDDEV, og de må samles fra basistabellene. For å unngå kostnader ved å samle ikke-distributive mål fra basistabellene, er sammendragstabellene definert med ROLLUP-operatoren, som forhåndssamler de ikke-distributive målene.

Gjør slik:

Du trenger ikke å gjøre noe.

7401: Sammendragstabellen *tabellnavn* anbefales. Den forventes å ha *rader rader*, en tabellstørrelse på *n* MB og en indeksstørrelse på *n* MB.:

Forklaring

Dette målet er en beskrivelse av den anbefalte sammendragstabellen, inkludert beregnet radantall, beregnet lagerplass og beregnet lagerplass for indekser.

Gjør slik:

Du trenger ikke å gjøre noe.

7402: Det finnes *n* sammendragstabeller som ikke passer i den oppgitte grensen for lagerstørrelse. De har en beregnet akkumulert størrelse på *n* MB.:

Forklaring

Dette målet gir informasjon om de anbefalte sammendragstabellene som ikke passer i den oppgitte grensen for lagerstørrelse.

Gjør slik:

Hvis du vil vise disse sammendragstabellene i anbefalingene, må du kjøre optimaliseringsrådgiveren på nytt med en høyere grense for lagerplass.

7403: Anbefalingene omfatter optimaliseringer for kubene *kubenaon*.:

Forklaring

Sammendragstabeller er anbefalt for den oppgitte kubene. Enkelte spørringer vil bli optimalisert for denne kubene.

Gjør slik:

Du trenger ikke å gjøre noe.

7404: Anbefalingene omfatter ikke optimaliseringer for kubene *kubenaavn*.

Forklaring

Sammendragstabeller er ikke anbefalt spesifikt for denne kubene. Spøringer som er spesifikke for denne kubene, vil sannsynligvis ikke vise forbedret ytelse.

Gjør slik:

Hvis sammendragstabeller ikke er med i anbefalingene på grunn av en lagerplassbegrensning, kan du kjøre optimaliseringsrådgiveren på nytt med en høyere grense for lagerplass. Anbefalingene kan omfatte en eller flere sammendragstabeller for å optimalisere spøringer for denne kubene.

7405: Den oppgitte tidsgrensen ble oppbrukt mens optimaliseringsrådgiveren bestemte anbefalingene.

Forklaring

Optimaliseringsrådgiveren gjorde en anbefaling. Hvis mer tid er tillatt, er det mulig at optimaliseringsrådgiveren kan gi en bedre anbefaling fordi den kan utføre mer analyse. Å kjøre optimaliseringsrådgiveren en lengre tid garanterer ikke et bedre resultat.

Gjør slik:

Du kan kjøre optimaliseringsrådgiveren igjen med mer tid oppgitt, eller du kan opprette de anbefalte sammendragstabellene og se om ytelsen er akseptabel.

7406: Dimensjonen *dimensjonsnavn* har ingen hierarkier som kan optimaliseres av optimaliseringsrådgiveren.

Forklaring

Optimaliseringsrådgiveren kan ikke optimalisere for rekursive hierarkier. Den oppgitte dimensjonen inneholder ingen hierarkier som kan optimaliseres, og derfor overser optimaliseringsrådgiveren denne dimensjonen. Spøringer som refererer til attributter fra denne dimensjonen, blir ikke optimalisert.

Gjør slik:

Du behøver ikke å gjøre noe. Spøringer som bruker attributter fra denne dimensjonen, vil ikke få forbedret ytelse.

7407: De anbefalte sammendragstabellene optimaliserer for n prosent av sektorene i kubemodellen. Spøringer som kjører mot de optimaliserte sektorene, får sannsynligvis forbedret ytelse.

Forklaring

SQL-spøringer får tilgang til bestemte sektorer i kubemodellen. En måte å analysere ytelsesforbedringen på er å vurdere hvilken del av sektorene som det kan utføres spøringer på, som vil bli forbedret. Hvis kubemodellen bruker distributive mål, vil spøringer som får tilgang til sektorer som logisk ligger over sammendragstabellsektoren, få forbedret ytelse.

Det er for eksempel 30 mulige sektorer i en kubemodell som har en tidsdimensjon med hierarkiet [Alle-År-Kvartal-Måned-Dag] og en regiondimensjon med hierarkiet [Alle-Land-Region-Delstat-By-Butikk]. Du kan beregne antall mulige sektorer ved å multiplisere antallet nivåer i dimensjonshierarkiet. Hvis den anbefalte sammendragstabellen optimaliserer for sektoren Måned-By, blir alle sektorene i eller iver denne sektoren optimalisert. I dette eksempelet blir 20 av 30 mulige sektorer, eller 67 % (20/30) av sektorene, optimalisert. Det vil aldri være 100 % dekning fordi det vil kreve duplisering av basistabellene i sammendragstabellene. Vanligvis er det mindre nyttig å optimalisere de laveste sektorene fordi de ikke er så forskjellige fra basistabellene.

Gjør slik:

Du behøver ikke å gjøre noe. Hvis prosentdelen er lav, kan du kjøre optimaliseringsrådgiveren på nytt med en høyere grense for lagerplass.

7408: Leser kubemodellens metadata fra databasen.:

Forklaring

Optimaliseringsrådgiveren leser metadataene som beskriver kubemodellen. Metadataene inneholder informasjon som signifikant påvirker optimaliseringsanbefalingene.

Gjør slik:

Du trenger ikke å gjøre noe.

7409: Velger hvilke samlinger som skal tas med i sammendragstabellene.:

Forklaring

Optimaliseringsrådgiveren tester potensielle sammendragstabellkonfigurasjoner for å bestemme hvilken konfigurasjon som er optimal for de oppgitte kriteriene.

Gjør slik:

Du trenger ikke å gjøre noe.

7410: Henter eksempeldata fra kubemodellen.:

Forklaring

Optimaliseringsrådgiveren leser et delsett av dataene fra fakta- og dimensjonstabellene slik at den kan beregne størrelsen på sammendragstabellen. Avlesing kan skje flere ganger mens optimaliseringsrådgiveren vurderer potensielle sammendragstabeller.

Gjør slik:

Du trenger ikke å gjøre noe.

7411: Definerer indekser for anbefalte sammendragstabeller.:

Forklaring

Optimaliseringsrådgiveren har bestemt den anbefalte sammendragstabellen og velger indekser som skal bygges for sammendragstabellen.

Gjør slik:

Du trenger ikke å gjøre noe.

7413: Det er ikke oppgitt noen optimaliseringssektorer for kubene *kubenaavn*, så optimaliseringsrådgiveren optimaliserer kubene for spørringer av neddrillingstypen.:

Forklaring

Hvis det ikke er definert noen optimaliseringssektorer for en kube, optimaliserer optimaliseringsrådgiveren kubene for spørringer av neddrillingstypen.

7414: Det er ikke definert noen kube for kubemodellen som blir optimalisert, så optimaliseringsrådgiveren optimaliserer kubemodellen som om den hadde en kube for spørringer av rapporttypen.:

Forklaring

Hvis det ikke er definert noen kuber for en kubemodell, kan optimaliseringsrådgiveren optimalisere kubemodellen som om den hadde en kube som brukes for spørringer av neddrillingstypen. Hvis du ikke har definert noen kuber for kubemodellen du optimaliserer, gir optimaliseringsrådgiveren anbefalinger som om du har en kube med en optimaliseringssektor av typen neddrilling på nivået Alle for hver kubedimensjon.

7415: Optimaliseringsrådgiveren anbefalte ikke en sammendragstabell for optimaliseringssektor *sektornavn* i kubene *kubenaavn*.:

Forklaring

En sammendragstabell er ikke anbefalt for den oppgitte optimaliseringssektoren.

Gjør slik:

Du trenger ikke å gjøre noe.

7416: Operasjonen stoppet. Anbefalingene er kanskje ikke optimale.:

Forklaring

Optimaliseringsrådgiveren ble stoppet av operasjonen 'stop advise'. Anbefalingene er kanskje ikke optimale.

Gjør slik:

For optimale anbefalinger bør du kjøre optimaliseringsrådgiveren på nytt uten tidsbegrensning.

7417: Optimaliseringsrådgiveren stoppet å opprette anbefalinger, fordi den nådde tidsgrensen som var definert. Optimaliseringsrådgiveren kan anbefale bedre sammendragstabeller hvis du gir den mer tid.:

Forklaring

Tidsrommet du har oppgitt for optimaliseringsrådgiveren, kan være for kort. Anbefaling: Kjør optimaliseringsrådgiveren med lengre tidsgrense.

Gjør slik:

Kjør optimaliseringsrådgiveren med lengre tidsgrense. Optimaliseringsrådgiveren vil da kunne anbefale bedre sammendragstabeller.

7418: Optimaliseringsrådgiveren kan ikke anbefale en sammendragstabell for kuben *kubenavn*, fordi sammendragstabellen har for mange rader sammenliknet med antallet rader i faktatabellen.:

Forklaring

Optimaliseringsrådgiveren kan ikke anbefale en sammendragstabell for kuben. Antallet rader i sammendragstabellen er for høyt, i prosent av antall rader i faktatabellen.

Gjør slik:

Hvis du har angitt optimaliseringssektorer, kan en eller flere av optimaliseringssektorene ha angitt for mange nivåer, eller nivåer som ligger for nær faktatabellens antall. Du kan oppgi nivået *Vilkårlig* for noen av kubedimensjonene, eller oppgi nivåer som er nærmere toppen av hierarkiet. Du kan også la optimaliseringsrådgiveren kjøre over en lengre periode for å oppnå bedre anbefalinger.

OLAP-senter, meldinger

10000-10600

10000: OLAP-senteret kan ikke hente noen databasenavn.

Forklaring

Det oppstod en feil under henting av listen over databasenavn fra DB2.

Gjør slik:

Kontroller at OLAP-senteret er riktig installert. Hvis problemet vedvarer, kontakter du IBM Kundeservice.

10001: Skriv et brukernavn.

Forklaring

Feltet **Bruker-ID** er tomt.

Gjør slik:

Oppgi en bruker-ID i feltet **Bruker-ID**.

10002: Oppgi et passord.

Forklaring

Feltet **Passord** er tomt.

Gjør slik:

Oppgi et passord i feltet **Passord**.

10004: Kan ikke analysere attributtet som ble oppgitt i SQL-uttrykket.

Forklaring

SQL-uttrykk som ble oppgitt, refererer til at attributt som verken er gyldig i denne konteksten eller finnes i databasen.

Gjør slik:

Kontroller at SQL-uttrykket bare refererer til de attributtene som vises i feltet **Data** i SQL Expression Builder.

10005: Metadataobjektene ble eksportert til filen *filnavn*.

Forklaring

Eksporten var vellykket.

Gjør slik:

Du trenger ikke å gjøre noe.

10006: Oppgi navnet på filen som metadataobjektene skal eksporteres til.

Forklaring

Eksportfunksjonen skriver de eksporterte metadataobjektene til filen som er oppgitt av brukeren.

Gjør slik:

Oppgi et filnavn i feltet **Filnavn**.

10007: Velg en kube eller en kubemodell som skal eksporteres.

Forklaring

Du kan eksportere en kube eller en kubemodell i eksportvinduet.

Gjør slik:

Velg et objekt du vil eksportere.

10008: Oppgi et entydig navn på objektet du oppretter.

Forklaring

Feltet **Navn** i SQL Expression Builder er tomt for attributtet eller målet du oppretter.

Gjør slik:

Oppgi et objektnavn i feltet **Navn**. Objektnavnet må være entydig i navneområdet for attributtene eller målene.

10009: Oppgi et SQL-uttrykk for objektet.

Forklaring

Feltet SQL-uttrykk er tomt for objektet.

Gjør slik:

Oppgi et SQL-uttrykk for objektet.

10010: Kolonnen *kolonnenavn* er ikke kvalifisert med et tabellnavn.

Forklaring

Kolonnenavnet som er oppgitt i SQL-uttrykket, er ikke kvalifisert med et tabellnavn.

Gjør slik:

Sørg for at kolonnenavnet i SQL-uttrykket er kvalifisert med et tabellnavn med '.' som skillenavn.

10011: Kolonnen *kolonnenavn* er ikke kvalifisert med et skjemanavn.

Forklaring

Kolonnereferanser i SQL-uttrykket må være kvalifisert med både et tabellnavn og et skjemanavn, atskilt med '..'.

Gjør slik:

Sørg for at kolonnenavnet i SQL-uttrykket er kvalifisert med både et tabellnavn og et skjemanavn med '..' som skilletegn.

10012: Det første elementet i et samlingsskript kan ikke være en dimensjon.

Forklaring

Et samlingsskript ble opprettet med en dimensjon som første element.

Gjør slik:

Bruk en samlingsfunksjon som første element i et samlingsskript.

10013: Velg et eksisterende mål eller oppgi et SQL-uttrykk som den andre parameteren for funksjonen med flere parametere, *funksjonsnavn*, i samlingen.

Forklaring

Når du bruker en funksjon med flere parametere i samlingsskriptet, er den første parameteren definert som målet som samlingen er knyttet til. Velg et eksisterende mål eller oppgi et SQL-uttrykk for den andre parameteren.

Gjør slik:

Oppgi et mål eller et SQL-uttrykk som andre parameter.

10014: Samlingsfunksjonen *funksjonsnavn* har ingen samsvarende dimensjoner.

Forklaring

Hver samlingsfunksjon i samlingsskriptet må brukes på minst en dimensjon.

Gjør slik:

Sørg for at hver samlingsfunksjon i samlingskriptet brukes på minst en dimensjon.

10015: Høyreparentesen mangler for objektet *objektnavn*.

Forklaring

I OLAP-senter må attributter, mål eller kolonner som er oppgitt i et uttrykk, være omsluttet av henholdsvis @Attribute(), @Measure() eller @Column().

Gjør slik:

Legg inn en høyreparentes for objektet.

10016: Hvis du refererer til objektet *objektnavn* i SQL-uttrykket, får du en ugyldig referansesløyfe.

Forklaring

Objektet refererer til seg selv i SQL-uttrykket.

Gjør slik:

Sørg for at objektene i SQL-uttrykket ikke danner referansesløyfer.

10017: Ingen feil ble funnet. SQL-uttrykket er gyldig.

Forklaring

SQL-uttrykket er gyldig.

Gjør slik:

Du trenger ikke å gjøre noe.

10018: Ingen feil ble funnet. Samlingskriptet er gyldig.

Forklaring

Settet med samlinger i samlingskriptet er gyldig.

Gjør slik:

Du trenger ikke å gjøre noe.

10020: Skriv et navn.

Forklaring

Feltet **Navn** er tomt for objektet.

Gjør slik:

Oppgi et objektnavn i feltet **Navn**.

10021: Skriv et skjemanavn.

Forklaring

Feltet **Skjema** er tomt.

Gjør slik:

Oppgi et skjemanavn i feltet **Skjema**.

10022: Skriv et forretningsnavn.

Forklaring

Feltet **Forretningsnavn** er tomt.

Gjør slik:

Forretningsnavnet kan vises i Business Intelligence-applikasjoner for å identifisere objektet for en sluttbruker. Oppgi et forretningsnavn i feltet **Forretningsnavn**.

10023: Velg minst et nivå som skal tas med i kubehierarkiet.

Forklaring

Det er ikke oppgitt noen nivåer i kubehierarkiet.

Gjør slik:

Velg minst ett nivå som skal tas med i kubehierarkiet.

10024: Velg minst ett mål for kubefaktaene.

Forklaring

Det er ikke oppgitt noen mål for kubefaktaene.

Gjør slik:

Velg minst ett mål for kubefaktaene.

10025: Velg minst ett nivå som skal tas med i kubehierarkiet.

Forklaring

Det er ikke oppgitt noen attributter i kubehierarkiet.

Gjør slik:

Velg minst ett attributt som skal tas med i kubehierarkiet.

10026: Velg minst en dimensjon i kubene.

Forklaring

Det er ikke oppgitt noen dimensjoner i kubene.

Gjør slik:

Velg minst en dimensjon og klikk på knappen [...] for å oppgi detaljer for kubedimensjonen.

10027: En eller flere dimensjoner som finnes i kuben, er fjernet. Klikk på Ja hvis du vil slette kubedimensjonene. Klikk på Nei hvis du vil gå tilbake til vinduet, og klikk deretter på Avbryt hvis du vil lukke vinduet uten å lagre endringene.

Forklaring

Ett eller flere dimensjonsvalg er fjernet. De tilsvarende kubedimensjonene vil bli slettet fra kuben.

Gjør slik:

Klikk på **Ja** i vinduet hvis du vil fjerne kubedimensjonene fra kuben. Klikk på **Nei** hvis du vil beholde kubedimensjonene, og klikk deretter på **Avbryt** for å lukke vinduet uten å lagre endringene.

10028: Et attributtforhold kan ikke defineres med kardinaliteten mange:mange hvis det er merket av for funksjonell avhengighet.

Forklaring

Et attributtforhold kan ikke defineres med kardinaliteten *mange:mange* hvis funksjonell avhengighet er valgt.

Gjør slik:

Velg en annen kardinalitet for attributtforholdet eller fjern merket i valgruten for funksjonell avhengighet.

10029: Velg et venstreattributt og et høyreattributt for attributtforholdet.

Forklaring

Det kan ikke defineres et attributtforhold hvis verken høyre- og venstreattributtet er valgt.

Gjør slik:

Velg både venstre- og høyreattributt.

10030: Det samme attributtet kan ikke velges for både venstre- og høyreattributtet i et attributtforhold.

Forklaring

Det kan ikke defineres et attributtforhold hvis høyre- og venstreattributtet er like.

Gjør slik:

Velg forskjellig venstre- og høyreattributt.

10031: Det finnes allerede et objekt med dette navnet og skjemaet i databasen. Oppgi et annet navn.

Forklaring

Et objekt av typen som blir opprettet eller endret, finnes allerede i databasen med samme navn og skjema oppgitt.

Gjør slik:

Oppgi et entydig navn for objektet.

10032: Oppgi minst ett attributtpar.

Forklaring

En kombinerings må ha minst ett attributtpar.

Gjør slik:

Oppgi minst ett attributtpar.

10033: Like attributtpar kan ikke opprettes.

Forklaring

Det finnes allerede et attributtpar som samsvarer med de nye valgene.

Gjør slik:

Velg forskjellig venstre- og høyreattributt.

10034: Velg minst en tabell.

Forklaring

Det er ikke valgt noen tabeller.

Gjør slik:

Du må velg minst en tabell for å fortsette.

10035: 10035: Velg eller opprett nye kombinerings for å kombinere alle de valgte tabellene.

Forklaring

Det er ikke valgt noen kombinerings.

Gjør slik:

Velg eller opprett nye kombinerings som skal kombinere alle de valgte tabellene.

10036: Velg minst ett attributt.

Forklaring

Det er ikke valgt noen attributter.

Gjør slik:

Velg minst ett attributt.

10037: Velg en kombinerings for å kombinere dimensjonen med faktaobjektet.

Forklaring

Det er ikke valgt noen kombinerings.

Gjør slik:

Velg en kombineringsnavn som skal kombinere dimensjonen med faktaobjektet.

10038: Oppgi bare en kombineringsnavn mellom to oppgitte tabeller. Kombineringsnavn1 og kombineringsnavn2 kombinerer begge de samme to tabellene.

Forklaring

Mer enn en kombineringsnavn ble valgt for samme par med tabeller.

Gjør slik:

Velg bare en kombineringsnavn for hvert par av tabeller.

10039: Alle valgte tabeller må kombineres. Velg en kombineringsnavn for tabellen tabellnavn.

Forklaring

Alle valgte tabeller må kombineres.

Gjør slik:

Velg en kombineringsnavn for den oppgitte tabellen.

10040: Antall valgte tabeller samsvarer ikke med antallet valgte kombineringsnavn. Kontroller at det ikke finnes kombineringsnavn, og at alle tabellene er kombinert.

Forklaring

Alle valgte tabeller må kombineres.

Gjør slik:

Kontroller at det ikke finnes kombineringsnavn, og at alle tabellene er kombinert.

10042: Velg minst ett mål.

Forklaring

Det er ikke oppgitt noen mål.

Gjør slik:

Velg minst ett mål.

10043: Velg en tabellkolonne.

Forklaring

Det var ikke oppgitt en tabellkolonne.

Gjør slik:

Velg en kolonne.

10044: Velg et SQL-uttrykk.

Forklaring

Det var ikke oppgitt et SQL-uttrykk.

Gjør slik:

Klikk på **Bygg uttrykk** for å bygge uttrykket.

10045: Det var ikke oppgitt et samlingskript.

Forklaring

Det var ikke oppgitt et samlingskript.

Gjør slik:

Klikk på **Bygg skript** for å bygge samlingskriptet.

10046: Velg et mål før du åpner Expression Builder.

Forklaring

Det er ikke valgt et mål.

Gjør slik:

Velg et mål fra tabellen.

10047: Velg et mål før du åpner Bygging av samlingskript.

Forklaring

Det er ikke valgt et mål.

Gjør slik:

Velg et mål.

10048: Bygging av samlingskript kan ikke startes for målet *målnavn* fordi kubemodellen *modellnavn* ikke har minst en dimensjon.

Forklaring

Du kan ikke oppgi et samlingskript hvis ikke kubemodellen har minst en dimensjon.

Gjør slik:

Tilføy dimensjoner til kubemodellen før du oppgir et samlingskript.

10049: Oppgi et attributt hvis du vil redigere uttrykket.

Forklaring

Det er ikke valgt et attributt.

Gjør slik:

Velg et attributt.

10050: Metadataene vil bli oppdatert fra databasen. Eventuelle endringer som var gjort da feilen oppstod, vil gå tapt.

Forklaring

Det oppstod en feil under kallet til den lagrede DB2-prosedyren.

Gjør slik:

Klikk på **OK** for å oppdatere metadataene som vises av OLAP-senteret. Eventuelle endringer som var gjort da feilen oppstod, vil gå tapt. Objektene som vises i OLAP-senteret, blir oppdatert med de tilsvarende objektene i databasen slik at du kan fortsette å arbeide med dem.

10051: Kubemodellen *modellnavn* kan ikke valideres for optimalisering. DB2 returnerte den følgende meldingen: *melding*.

Forklaring

OLAP-senteret kan ikke starte veiviseren Optimaliseringsrådgiver for den valgte kubemodellen fordi den valgte kubemodellen ikke bestod valideringen som ble utført av APIen for den valgte prosedyren.

Gjør slik:

Les om regelen for kubemodellvalidering i dokumentasjonen til APIen for den lagrede prosedyren. Følg instruksjonene i meldingen fra DB2.

10052: Noen av de innlastede attributtene eller målene tilordner til kolonnene *kolonnenavn* som ikke lenger finnes i databasen. Løs problemet enten ved å gjenopprette tabellene som kolonnene tilhører, eller ved å slette de ugyldige attributtene og/eller målene.

Forklaring

Denne meldingen vises når du starter OLAP-senteret, eller når du klikker på **Vis** —> **Forny**. Den vises fordi en tabell som de lastede attributtene eller målene er tilordnet til, ble slettet eller fikk endret navn.

Gjør slik:

Rett problemet på en av disse måtene:

- Gjenopprett tabellen som ble slettet eller fikk endret navn.
- Tilordne attributtene/målene til en tabell som finnes i databasen.
- Slett attributtet/målene som er tilordnet til kolonnene som ikke finnes.

10053: Optimaliseringsvalideringen av kubemodellen *modellnavn* returnerte en advarsel. DB2 returnerte den følgende meldingen: *melding*.

Forklaring

OLAP-senteret prøvde å validere kubemodellen før optimaliseringsrådgiveren ble startet, og DB2 returnerte en advarsel. Advarselen kan bety at du har en kubemodell som ikke kan optimaliseres. Kubemodellen kan for eksempel inneholde utsnitt som refererer til tabeller som det ikke er definert begrensninger mellom.

Gjør slik:

Kontroller meldingen som er returnert av DB2, og avgjør om du vil fortsette kjøringen av optimaliseringsrådgiveren.

10060: Kubemodellen er ikke fullstendig. Før en kube kan opprettes, må kubemodellen inneholde et faktaobjekt, minst en dimensjon og minst ett hierarki for hver dimensjon.

Forklaring

Kubemodellen har ikke riktig status for oppretting av en kube.

Gjør slik:

Endre kubemodellen slik at den inneholder et faktaobjekt og minst en dimensjon. Kontroller at hver dimensjon har minst ett hierarki.

10061: Når en kubemodell slettes, fjernes tilhørende dimensjoner og faktaobjekter. De fjernede dimensjonene vil fortsatt være tilgjengelige fra mappen Alle dimensjoner. Er du sikker på at du vil slette kubemodellen *modellnavn*?

Forklaring

Dette er en bekreftelsesmelding for sletting.

Gjør slik:

Kontroller at det valgte objektet er det objektet du vil slette, og klikk på **Ja**. Klikk på **Nei** hvis du ikke vil slette det valgte objektet.

10062: Når en dimensjon slettes, slettes også de tilhørende hierarkiene og kubedimensjonene. Er du sikker på at du vil slette dimensjonen *dimensjonsnavn*?

Forklaring

Dette er en bekreftelsesmelding for sletting.

Gjør slik:

Kontroller at det valgte objektet er det objektet du vil slette, og klikk på **Ja**. Klikk på **Nei** hvis du ikke vil slette det valgte objektet.

10063: Når en kube slettes, slettes også de tilhørende kubedimensjonene, kubehierarkiene og kubefaktaene. Er du sikker på at du vil slette kuben *kubenavn*?

Forklaring

Dette er en bekreftelsesmelding for sletting.

Gjør slik:

Kontroller at det valgte objektet er det objektet du vil slette, og klikk på **Ja**. Klikk på **Nei** hvis du ikke vil slette det valgte objektet.

10064: Når en kubedimensjon slettes, slettes også de tilhørende kubehierarkiene. Er du sikker på at du vil slette kubedimensjonen *kubedimensjonsnavn*?

Forklaring

Dette er en bekreftelsesmelding for sletting.

Gjør slik:

Kontroller at det valgte objektet er det objektet du vil slette, og klikk på **Ja**. Klikk på **Nei** hvis du ikke vil slette det valgte objektet.

10065: Er du sikker på at du vil slette *objektnavn*?

Forklaring

Dette er en bekreftelsesmelding for sletting.

Gjør slik:

Kontroller at det valgte objektet er det objektet du vil slette, og klikk på **Ja**. Klikk på **Nei** hvis du ikke vil slette det valgte objektet.

10066: Når en dimensjons fjernes, fjernes alle de tilhørende kubedimensjonene fra kubene. Er du sikker på at du vil fjerne dimensjonen *dimensjonsnavn* fra *objektnavn*?

Forklaring

Dette er en bekreftelsesmelding for fjerning.

Gjør slik:

Kontroller at det valgte objektet er det objektet du vil fjerne, og klikk på **Ja**. Klikk på **Nei** hvis du ikke vil fjerne det valgte objektet.

10067: Filen med navnet *filnavn* finnes allerede. Vil du overskrive innholdet?

Forklaring

Dette er en bekreftelsesmelding for overskriving.

Gjør slik:

Kontroller at du vil skrive over innholdet i filen du har oppgitt.

10068: Kan ikke bestemme datatypen for objektet med navnet *objektnavn* og skjemaet *skjemanavn*. Databasen returnerte den følgende informasjonen: *melding*.

Forklaring

OLAP-senteret kan ikke bestemme kildedatatypen eller samlingsdatatypen for objektet som er oppgitt.

Gjør slik:

Kontroller at SQL-uttrykket for det oppgitte objektet er riktig. Hvis du ikke kan løse problemet, kontakter du IBM Kundeservice.

10069: Kan ikke bestemme kildedatatypen for målet med navnet *målnavn* og skjemaet *skjemanavn*.

Forklaring

OLAP-senteret kan ikke bestemme kildedatatypen for målet som er oppgitt, fordi målet har et ugyldig kildeuttrykk. Et mål kan ha et ugyldig kildeuttrykk når det har samlingsinnstillingen Ingen, fordi målet blir validert med samlinger av refererte mål i stedet for som et innebygd uttrykk.

Gjør slik:

Utfør en av disse handlingene:

- Endre kildeuttrykket for målet som er oppgitt, slik at det validerer riktig med samlingsinnstillingen Ingen.
- Ikke bruk målet som er oppgitt, i uttrykket.

10070: Når et faktaobjekt slettes, slettes også de tilhørende målene. Er du sikker på at du vil slette faktaobjektet *faktanavn*?

Forklaring

Dette er en bekreftelsesmelding for sletting.

Gjør slik:

Kontroller at det valgte objektet er det objektet du vil slette, og klikk på **Ja**. Klikk på **Nei** hvis du ikke vil slette det valgte objektet.

10071: Alle de valgte objektene blir slettet fra databasen. Vil du slette disse objektene?

Forklaring

Du har valgt mer enn ett objekt og alternativet for sletting.

Gjør slik:

Kontroller at de valgte objektene er de objektene du vil slette, og klikk på **Ja**. Klikk på **Nei** hvis du ikke vil slette de valgte objektene.

10072: Enkelte av de valgte objektene kan ikke slettes. Disse objektene blir igjen i databasen.

Forklaring

OLAP-senteret kan slette alle de valgte objektene. Dette er sannsynligvis fordi det er referert til enkelte av de valgte objektene av andre objekter i databasen, og hvis du sletter det valgte objektet, blir de refererende objektene ugyldige.

Gjør slik:

Du trenger ikke å gjøre noe.

10073: Ingen av de valgte objektene kan slettes.

Forklaring

OLAP-senteret kan ikke slette noen av de valgte objektene. Dette er sannsynligvis fordi det er referert til enkelte av de valgte objektene av andre objekter i databasen, og hvis du sletter de valgte objektene, blir de refererende objektene ugyldige.

Gjør slik:

Du trenger ikke å gjøre noe.

10074: Er du sikker på at du vil fjerne objektnavn fra katalognavn?

Forklaring

Bekreftelsesmelding for fjerning.

Gjør slik:

Kontroller at det valgte objektet er det du vil fjerne, og klikk på **Ja**. Klikk på **Nei** hvis du ikke vil fjerne det valgte objektet.

10075: Alle de valgte objektene blir fjernet fra katalognavn. Vil du fjerne disse objektene?

Forklaring

Du har valgt mer enn ett objekt og alternativet for fjerning.

Gjør slik:

Kontroller at de valgte objektene er de objektene du vil slette, og klikk på **Ja**. Klikk på **Nei** hvis du ikke vil fjerne de valgte objektene.

10076: Enkelte av de valgte objektene kan ikke fjernes. Disse objektene blir igjen i databasen.

Forklaring

OLAP-senteret kan ikke slette alle de valgte objektene. Dette er sannsynligvis fordi det er referert til enkelte av de valgte objektene av andre objekter i databasen, og hvis du fjerner det valgte objektet, blir de refererende objektene ugyldige. Det kan også være fordi det overordnede objektet trenger minst ett underordnet objekt, for eksempel i et kubehierarki.

Gjør slik:

Du trenger ikke å gjøre noe.

10077: Ingen av de valgte objektene kan fjernes.

Forklaring

OLAP-senteret kan ikke fjerne noen av de valgte objektene. Dette er sannsynligvis fordi det er referert til enkelte av de valgte objektene av andre objekter i databasen, og hvis du fjerner de valgte objektene, blir det refererende objektet ugyldig.

Gjør slik:

Du trenger ikke å gjøre noe.

10078: Operasjonen var vellykket. Følgende informasjonsmeldinger ble returnert av databasen: *databasenavn*.

Forklaring

Databaseoperasjonen ble fullført, men det ble returnert noen informasjonsmeldinger.

Gjør slik:

Undersøk informasjonsmeldingene og avgjør om ytterligere tiltak må iverksettes.

10080: Det ble ikke funnet et objekt av typen *type* i andre runde av XML.

Forklaring

Et objekt det er referert til i XML-filen som leses, ble ikke funnet.

Gjør slik:

Kontroller at XML-filen du importerer, er riktig utformet. Hvis denne feilen oppstår når du starter OLAP-senteret, kontakter du IBM Kundeservice.

10081: Systemet kunne ikke analysere XML fra filen *filnavn*. Feilen oppstod på linjen *linjenummer*, tegnet *tegnummer*. Analysatoren returnerte følgende informasjon: informasjon.

Forklaring

Det oppstod en analysefeil under forsøk på å importere en XML-fil

Gjør slik:

Kontroller at XML-filen du importerer, er riktig utformet, og at den er en gyldig Cube Views XML-metadatafil. Kontroller linje- og tegnummeret for å finne feilen.

10082: Det ble funnet et uventet analysatorunntak i filen *filnavn*. Den følgende informasjonen ble returnert: informasjon.

Forklaring

Det oppstod en uventet analysefeil under forsøk på å importere en XML-fil.

Gjør slik:

Kontroller at XML-filen du importerer, er riktig utformet, og at den er en gyldig Cube Views XML-metadatafil. Kontroller linje- og tegnummeret for å finne feilen.

10084: Det finnes allerede et objekt med navnet *objektnavn* i skjemaet *skjemanavn*. Objektet kan ikke opprettes. Oppgi et entydig navn, skjema eller begge deler for det nye objektet.

Forklaring

OLAP-senteret forsøkte å opprette et nytt objekt, men det finnes allerede et objekt av denne typen med samme navn og skjema.

Gjør slik:

Oppgi et annet navn, skjema eller begge deler for objektet som blir opprettet.

10085: Det finnes allerede et objekt med navnet *objektnavn* i skjemaet *skjemanavn*. Det er ikke mulig å endre navn på objektet. Oppgi et entydig navn, skjema eller begge deler for objektet som det blir endret navn for.

Forklaring

OLAP-senteret forsøkte å endre navnet på et objekt, men det finnes allerede et objekt av denne typen med samme navn og skjema.

Gjør slik:

Oppgi et annet navn, skjema eller begge deler for objektet som det blir endret navn for.

10086: Det var ikke mulig å opprette en databasetilkobling. DB2 returnerte: *melding*.

Forklaring

OLAP-senteret kunne ikke tilkobles databasen. Feilinformasjon fra DB2 er tatt med i meldingen.

Gjør slik:

Les teksten som er returnert av DB2, og rett problemet.

10087: Metadataobjektet *objektnavn1* kan ikke slettes fordi det er referert til av objektet *objektnavn2* av typen *type*.

Forklaring

Det valgte metadataobjektet kan ikke slettes fordi det brukes av minst ett annet metadataobjekt.

Gjør slik:

Fjern objektet fra andre metadataobjekter som det er en del av, og prøv å slette objektet på nytt.

10088: Det oppstod en feil under registrering av DB2-styreprogrammet med JDBC Driver Manager. Det var ikke mulig å opprette en databasetilkobling. Den følgende informasjonen ble returnert: *melding*.

Forklaring

Før tilkobling til en DB2-database må OLAP-senteret registrere JDBC-styreprogrammet det skal bruke, med Driver Manager. Det oppstod en feil under registreringen av JDBC-styreprogrammet.

Gjør slik:

Kontroller DB2-installeringen og se om filene *db2java.zip* og *db2jcc.jar* er installert. Kontroller at Java og eventuelle JDBC-komponenter er riktig installert. Les informasjonen i meldingen for å få hjelp til å løse problemet.

10089: Det oppstod en feil ved tilgang til databasen. Databasen returnerte den følgende informasjonen: \n SQL-status: *melding*\n SQL-feilkode: *kode*\n SQL-melding: *SQL-melding*.

Forklaring

OLAP-senterapplikasjonen sendte kall til DB2 med APIen for den lagrede prosedyren. Utføringskommandoen førte til en `SQLException` som ikke kunne håndteres av OLAP-senteret.

Gjør slik:

Les informasjonen i meldingen for å få hjelp til å løse problemet. Hvis du ikke kan løse problemet, kontakter du IBM Kundeservice.

10090: Utføringen av den lagrede DB2-prosedyren forårsaket en usann returkode. Ingen feilinformasjon ble funnet i det returnerte XML-dokumentet. Ta kontakt med IBM Kundeservice.

Forklaring

OLAP-senterapplikasjonen sendte kall til DB2 med APIen for den lagrede prosedyren. Utføringskommandoen returnerte *false*, men det var ingen feilinformasjon i XML-dokumentet som ble returnert av den lagrede prosedyren.

Gjør slik:

Det er mulig at operasjonen ble fullført på vellykket måte, men du bør rapportere problemet til IBM Kundeservice.

10091: Det oppstod en feil under behandling av et API-kall til databasen. Den følgende informasjonen ble returnert: \n SQL-status:*melding*\n SQL-feilkode: *kode*\n Operasjon:*operasjon*\n Status-ID: *ID*\n Statustekst: *tekst*.

Forklaring

Det oppstod en feil i API-kallet fra OLAP-senteret under utføringen av enkelt endringer i OLAP-senteret.

Gjør slik:

Se informasjonen i meldingen. Hvis du ikke kan løse problemet, kontakter du IBM Kundeservice.

10092: Det oppstod en feil under analyse av XML-koden som ble returnert av API-kallet til databasen. Den følgende informasjonen ble returnert: *melding*.

Forklaring

API-kallet fra OLAP-senteret returnerte en XML-fil som var ufullstendig eller feil utformet. OLAP-senteret kunne ikke lese den returnerte XML-filen.

Gjør slik:

Løs problemet ved hjelp av informasjonen i meldingen. Hvis du ikke kan løse problemet, kontakter du IBM Kundeservice.

10093: Filen *filnavn* finnes ikke.

Forklaring

Den oppgitte filen finnes ikke.

Gjør slik:

Oppgi en fil som finnes.

10094: Det oppstod en I/U-feil under lesing av filen *filnavn*. Den følgende systeminformasjonen ble returnert: *melding*.

Forklaring

Det oppstod en I/U-feil under lesing fra en fil.

Gjør slik:

Kontroller systeminformasjonen og prøv å løse problemet, eller oppgi en annen fil.

10095: Det oppstod en I/U-feil under skriving til filen *filnavn*. Den følgende systeminformasjonen ble returnert: *melding*.

Forklaring

Det oppstod en I/U-feil under skriving til en fil.

Gjør slik:

Kontroller systeminformasjonen og prøv å løse problemet, eller oppgi en annen fil.

10096: En spørring for å hente databaseskjemaet mislyktes. Databasen returnerte den følgende informasjonen: *melding*.

Forklaring

En spørring for å hente databaseskjemaet mislyktes.

Gjør slik:

Kontroller databaseinformasjonen og prøv å løse problemet.

10097: En spørring for å hente skjemaets tabeller mislyktes. Databasen returnerte den følgende informasjonen: *melding*.

Forklaring

En spørring for å hente skjemaets tabeller mislyktes.

Gjør slik:

Kontroller databaseinformasjonen og prøv å løse problemet.

10098: En spørring for å hente tabellens kolonner mislyktes. Databasen returnerte den følgende informasjonen: *melding*.

Forklaring

En spørring for å hente tabellens kolonner mislyktes.

Gjør slik:

Kontroller databaseinformasjonen og prøv å løse problemet.

10099: En iverksetting av en DB2-tilkobling mislyktes. Databasen returnerte den følgende informasjonen: *melding*.

Forklaring

En iverksetting av en DB2-tilkobling mislyktes.

Gjør slik:

Kontroller databaseinformasjonen og prøv å løse problemet.

10100: En tilbakestilling av en DB2-tilkobling mislyktes. Databasen returnerte den følgende informasjonen: *melding*.

Forklaring

En tilbakestilling av en DB2-tilkobling mislyktes.

Gjør slik:

Kontroller databaseinformasjonen og prøv å løse problemet.

10101: *Objektnavn* kan ikke slettes fordi dette er den siste kubedimensjonen i kuben *kubenavn*. En kube må ha minst en kubedimensjon for å være gyldig.

Forklaring

OLAP-senteret prøvde å slette den siste kubedimensjonen i en kube.

Gjør slik:

En kube må ha minst en kubedimensjon for å være gyldig. Ikke prøv å slette den siste kubedimensjonen i en kube.

10102: Objektet *objektnavn1* av typen *type1* refererer til objektet *objektnavn2* eller typen *type2*, som ikke ble funnet.

Forklaring

Et objekt i XML-filen som blir lest, refererer til et objekt som ikke ble funnet. Hvis feilen oppstår under import, er det mulig at objektet det blir referert til, ikke ble funnet i filen som blir importert.

Gjør slik:

Hvis en import blir utført, må du kontrollere at filen inneholder alle objektene som er nødvendig for en vellykket import. Hvis denne feilen oppstår når du starter OLAP-senteret, kontakter du IBM Kundeservice.

10103: Målet kan ikke slettes fordi faktaobjektet må inneholde minst ett mål.

Forklaring

Målet kan ikke slettes fordi faktaobjektet må inneholde minst ett mål.

Gjør slik:

Du trenger ikke å gjøre noe.

10104: Målet kan ikke fjernes fordi kubefaktaene må inneholde minst ett mål.

Forklaring

Målet kan ikke fjernes fordi kubefaktaene må inneholde minst ett mål.

Gjør slik:

Du trenger ikke å gjøre noe.

10105: Kubenivået kan ikke fjernes fordi kubehierarkiet må inneholde minst ett kub nivå.

Forklaring

Kubenivået kan ikke fjernes fordi kubehierarkiet må inneholde minst ett kub nivå.

Gjør slik:

Du trenger ikke å gjøre noe.

10106: Nivået kan ikke fjernes fordi det er referert til det i et kub nivå i kubehierarkiet *hierarkinavn*, som er knyttet til hierarkiet *hierarkinavn*.

Forklaring

Nivået kan ikke fjernes fordi det er referert til det i et kub nivå i et kubehierarki som er knyttet til hierarkiet som attributtet blir fjernet fra.

Gjør slik:

Du trenger ikke å gjøre noe.

10107: Attributtet kan ikke fjernes fordi det er referert til det på kub nivået *nivånavn*, som er knyttet til nivået *nivånavn*.

Forklaring

Attributtet kan ikke fjernes fordi det er referert til det i et kub nivå som er knyttet til nivået som attributtet blir fjernet fra.

Gjør slik:

Du trenger ikke å gjøre noe.

10108: Attributtet kan ikke fjernes fordi det er referert til det i hierarkiet *hierarkinavn*, som er knyttet til dimensjonen *dimensjonsnavn*.

Forklaring

Attributtet kan ikke fjernes fordi det er referert til i et hierarki som er knyttet til dimensjonen som attributtet blir fjernet fra.

Gjør slik:

Du trenger ikke å gjøre noe.

10109: Attributtet kan ikke fjernes fordi det er referert til det i kombinasjonen *kombineringsnavn*, som er knyttet til dimensjonen *dimensjonsnavn*.

Forklaring

Attributtet kan ikke fjernes fordi det er referert til i en kombinasjon som er knyttet til dimensjonen som attributtet blir fjernet fra.

Gjør slik:

Du trenger ikke å gjøre noe.

10110: Nivået kan ikke flyttes fordi det er referert til det i kubehierarkiet *hierarkinavn*, som er knyttet til hierarkiet *hierarkinavn*.

Forklaring

Nivået kan ikke flyttes opp eller ned fordi det er referert til det i et kubnivå i et kubehierarki som er knyttet til hierarkiet som blir redigert.

Gjør slik:

Du trenger ikke å gjøre noe.

10111: Attributtet kan ikke fjernes fordi det er det eneste nivånøkkelattributtet på nivået *nivånavn*.

Forklaring

Attributtet kan ikke fjernes fordi det er det eneste nivånøkkelattributtet på nivået det blir fjernet fra, og nivået må ha minst ett nivånøkkelattributt.

Gjør slik:

Du trenger ikke å gjøre noe.

10112: Attributtet kan ikke fjernes fordi det er standardattributtet på nivået *nivånavn*.

Forklaring

Attributtet kan ikke fjernes fordi det er standardattributtet på nivået som attributtet fjernes fra.

Gjør slik:

Du trenger ikke å gjøre noe.

10113: Attributtet kan ikke fjernes fordi det er referert til det på nivået *nivånavn*, som er knyttet til dimensjonen *dimensjonsnavn*.

Forklaring

Attributtet kan ikke fjernes fordi det er referert til i et nivå som er knyttet til dimensjonen som attributtet blir fjernet fra.

Gjør slik:

Du trenger ikke å gjøre noe.

10114: Nivået kan ikke fjernes fordi det er det eneste nivået i hierarkiet *hierarkinavn*.

Forklaring

Nivået kan ikke fjernes fordi det er det eneste nivået i hierarkiet, og hierarkiet må inneholde minst ett nivå.

Gjør slik:

Du trenger ikke å gjøre noe.

10200: Filen som blir importert, har ikke UTF-8-koding. Velg en fil med UTF-8-koding.

Forklaring

OLAP-senteret kan bare importere filer med UTF-8-koding.

Gjør slik:

Importer en fil med støttet koding.

10201: Oppgi et filnavn for SQL-skriptet som skal brukes til å oppdatere sammendragstabeller.

Forklaring

Veiviseren Optimaliseringsrådgiver oppretter et SQL-skript for å oppdatere sammendragstabeller når alternativet for utsatt oppdatering er valgt. Dette skriptet kan du lagre i en fil og kjøre for å oppdatere sammendragstabellene.

Gjør slik:

Oppgi navnet på en fil som du vil lagre SQL-skriptet i.

10202: Oppgi et filnavn for SQL-skriptet som skal brukes til å opprette sammendragstabeller.

Forklaring

Veiviseren Optimaliseringsrådgiver genererer et SQL-skript for å opprette sammendragstabeller. Dette skriptet kan du lagre i en fil og kjøre for å opprette sammendragstabellene.

Gjør slik:

Oppgi navnet på en fil som du vil lagre SQL-skriptet i.

10203: Det valgte målet kan ikke ha Ingen som samlingsinnstilling. Bare beregnede mål som bare refererer til andre mål i uttrykkene, kan oppgi samlingsinnstillingen Ingen.

Forklaring

Samlingsinnstillingen Ingen kan bare velges for mål som bare bruker uttrykk som refererer eksklusivt til andre mål.

Gjør slik:

Oppgi en annen samlingsinnstilling.

10204: Det finnes ikke noen dimensjoner. Opprett en ny dimensjon for å tilføye den til kubemodellen.

Forklaring

Det finnes ikke noen dimensjoner. Opprett en ny dimensjon for å tilføye den til kubemodellen.

Gjør slik:

Opprett en ny dimensjon i stedet for å tilføye en dimensjon.

10205: Det finnes ingen dimensjoner å tilføye fordi alle de eksisterende dimensjonene allerede er tatt med i kubemodellen.

Forklaring

Alle de eksisterende dimensjonene er tilføyd i kubemodellen.

Gjør slik:

Du trenger ikke å gjøre noe.

10206: Du har endret de valgte alternativene. Hvis du vil se nye anbefalinger fra veiviseren for optimaliseringsrådgiveren, må du kjøre veiviseren for optimaliseringsrådgiveren på nytt. Hvis du ikke kjører veiviseren for optimaliseringsrådgiveren på nytt, får du se anbefalingene som er opprettet for tidligere alternativer. Vil du kjøre veiviseren for optimaliseringsrådgiveren på nytt?

Forklaring

Du har endret de valgte alternativene etter at du har kjørt veiviseren for optimaliseringsrådgiveren. Hvis du vil se oppdaterte anbefalinger for sammendragstabellene, må du kjøre veiviseren for optimaliseringsrådgiveren på nytt. Hvis du ikke kjører veiviseren for optimaliseringsrådgiveren på nytt, får du se anbefalingene som er opprettet for tidligere alternativer.

Gjør slik:

Klikk på **Ja** hvis du vil kjøre veiviseren for optimaliseringsrådgiveren. Klikk på **Nei** hvis du ikke vil kjøre veiviseren for optimaliseringsrådgiveren på nytt.

10207: Ingen dimensjonstabeller ble oppdaget.

Forklaring

Ingen dimensjonstabeller ble oppdaget.

Gjør slik:

Kontroller at referanseintegritetsbegrensningene er riktig definert.

10208: Det finnes objekter *objektnavn* som OLAP-senteret ikke kan vise direkte. Disse objektene kan forårsake fremtidige problemer med OLAP-senter. Klikk på Ja for å slette objektene, eller klikk på Nei hvis du vil beholde objektene i databasen.

Forklaring

OLAP-senteret oppdaget et antall objekter (som hierarkier eller faktaobjekter) i databasen, som det ikke kan vise direkte. Disse objektene kan finnes fra før, eller de kan være opprettet etter import av metadata. Objektene kan bli årsak til navnekonflikter og referanseproblemer i OLAP-senteret i fremtiden. Med mindre du har en god grunn til å beholde disse objektene, anbefales du å slette dem.

Gjør slik:

Klikk på **Ja** hvis du vil slette objektene, eller klikk på **Nei** hvis du vil beholde objektene i databasen.

10209: Det oppstod en uventet feil under importoperasjonen. Kontroller om det er feil i XML-inndatafilen.

Forklaring

Den lagrede prosedyrens API returnerte en advarsel under import uten noe i XML-utdatafilen.

Gjør slik:

Kontroller at metadataene i XML-inndatafilen er i overensstemmelse med formatet som er definert i OLAP-metadataskjemaet, og at XML-filen definerer alle metadataobjektene det refereres til i den.

10210: Importoperasjonen mislyktes. Den lagrede prosedyrens API returnerte den følgende meldingen: *melding*.

Forklaring

Den lagrede prosedyrens API returnerte en feilmelding under import.

Gjør slik:

Løs problemet ved hjelp av informasjonen i meldingen. Hvis du ikke kan løse problemet, kontakter du IBM Kundeservice.

10211: Det ikke-numeriske målet *målnavn* kan ikke bruke samlingsfunksjonen *funksjonsnavn* fordi denne funksjonen forventer et numerisk argument.

Forklaring

Mål med ikke-numeriske datatyper kan ikke ha numeriske samlingsfunksjoner. Du kan bare velge **MIN**, **MAX**, eller **COUNT** som samlingsfunksjoner for ikke-numeriske data.

Gjør slik:

Velg en annen samlingsfunksjon.

10212: Kan ikke lese objektene fra XML-inndatafilen. Kontroller om det er feil i XML-inndatafilen.

Forklaring

OLAP-senteret kan ikke lese objektene fra XML-inndatafilen.

Gjør slik:

Kontroller at metadataene i XML-inndatafilen er i overensstemmelse med formatet som er definert i OLAP-metadataschemaet, og at XML-filen definerer alle metadataobjektene det refereres til i den.

10213: XML-inndatafilen *filnavn* finnes ikke i den oppgitte katalogen.

Forklaring

XML-inndatafilen finnes ikke i den oppgitte katalogen.

Gjør slik:

Sørg for at XML-inndatafilen finnes i den oppgitte katalogen.

10214: Objektet *objektnavn* i importfilen refererer til kolonnen *kolonnenavn* som ikke finnes i databasen. Kontroller at tabellene og kolonnene det refereres til av metadataobjektene i importfilen, finnes før du importerer filen.

Forklaring

XML-importfilen inneholder objekter som refererer til tabeller og kolonner som ikke finnes i databasen.

Gjør slik:

Kontroller at tabellene det refereres til av objektene i XML-importfilen, finnes i databasen før du importerer filen.

10215: OLAP-senteret kan ikke kjøre SQL-skriptet som anbefales av veiviseren for optimaliseringsrådgiveren. Databasen returnerte den følgende informasjonen: *melding*.

Forklaring

OLAP-senteret kan ikke kjøre SQL-skriptet som anbefales av veiviseren for optimaliseringsrådgiveren. Det er mulig du ikke har nødvendige rettigheter for å utføre SQL-skriptet.

Gjør slik:

Kontroller at du har riktig autorisasjon til å utføre optimaliseringsrådgiverens anbefalinger. De nødvendige autorisasjonene er beskrevet i emnet Autorisasjoner og rettigheter i hjelpen til OLAP-senteret. I *DB2 Cube Views Setup and User's Guide* finner du opplysninger om optimalisering av kubemodeller.

10216: Anbefalingene fra optimaliseringsrådgiveren ble lagret i den eller de oppgitte filene.

Forklaring

De anbefalte SQL-skriptene for oppretting av sammendragstabeller, og eventuelt SQL-skriptene for oppdatering av sammendragstabeller, ble lagret i de oppgitte filene.

Gjør slik:

Du trenger ikke å gjøre noe.

10217: Sammendragstabellene og de tilhørende indeksene ble opprettet.

Forklaring

Sammendragstabellene og indeksene som er anbefalt av optimaliseringsrådgiveren, ble opprettet i databasen.

Gjør slik:

Du trenger ikke å gjøre noe.

10218: Du har valgt et utsnitt. Optimaliseringsrådgiveren kan ikke kontrollere at det finnes referansebegrensninger for tabeller det refereres til av utsnittet.

Forklaring

Optimalisering er kanskje ikke effektiv når du oppretter sammendragstabeller for kubemodeller som bruker utsnitt som refererer til tabeller uten begrensninger. Optimaliseringsrådgiveren kan ikke avgjøre om det finnes begrensninger for tabellene det er referert til av utsnittet.

Gjør slik:

Hvis tabellene utsnittet refererer til ikke har begrensninger og du vil kjøre optimaliseringsrådgiveren, kan du velge ett av følgende: 1. Ikke bruk utsnittet i kubemodellen. 2. Opprett begrensninger for tabellene før du kjører optimaliseringsrådgiveren.

10219: Hvis du avbryter veiviseren for optimaliseringsrådgiveren, blir de anbefalte SQL-skriptene slettet uten at de blir lagret. Klikk på Ja hvis du vil lukke optimaliseringsrådgiveren uten å lagre de anbefalte SQL-skriptene. Klikk på Nei hvis du vil gå tilbake til optimaliseringsrådgiveren slik at du kan lagre de anbefalte SQL-skriptene.

Forklaring

Hvis du klikker på knappen **Avbryt** i veiviseren for optimaliseringsrådgiveren, blir de anbefalte SQL-skriptene slettet uten at de blir lagret.

Gjør slik:

Klikk på **Ja** hvis du vil lukke optimaliseringsrådgiveren uten å lagre de anbefalte SQL-skriptene, eller klikk på **Nei** hvis du vil gå tilbake til

optimaliseringsrådgiveren slik at du kan fortsette å bruke veiviseren for optimaliseringsrådgiveren og lagre SQL-skriptene.

10220: Den oppgitte filen har et format som er for en gammel versjon av DB2 Cube Views. Importveiviseren kan konvertere metadataobjektene som er beskrevet av den oppgitte filen, til gjeldende versjon av DB2 Cube Views. Klikk på Ja hvis du vil konvertere metadataobjektene som er beskrevet i den oppgitte filen, og fortsette importen. Klikk på Nei hvis du vil stoppe importen slik at du kan oppgi en annen fil eller lukke importveiviseren.

Forklaring

Den oppgitte metadatakildefilen inneholder XML i et format som er for en eldre versjon av DB2 Cube Views. Importveiviseren kan lese filen og deretter konvertere innholdet til et format som kan leses av den gjeldende versjonen av DB2 Cube Views.

Gjør slik:

Klikk på **Ja** hvis du vil konvertere metadataobjektene som er beskrevet i den oppgitte kildefilen, til gjeldende versjon av DB2 Cube Views. På siden Importalternativer vises objektene fra den oppgitte filen, konvertert til den nye versjonen av DB2 Cube Views. Klikk på **Nei** hvis importveiviseren ikke skal konvertere metadataobjektene som er beskrevet i den oppgitte filen. Du kan oppgi en annen metadatakildefil eller lukke importveiviseren.

10221: Det forventes at de anbefalte sammendragstabellene bruker *lagerplass MB* med *lagerplass*.

Forklaring

Sammendragstabellene som anbefales av optimaliseringsrådgiveren, forventes å bruke den oppgitte lagerplassen.

Gjør slik:

Kontroller at du har minst den oppgitte lagerplassen tilgjengelig før du kjører de anbefalte skriptene.

10222: Vil du stoppe optimaliseringsrådgiveren? Klikk på Ja for å stoppe optimaliseringsrådgiveren og returnere anbefalingene den har innhentet så langt. Klikk på Nei hvis du vil at optimaliseringsrådgiveren skal fortsette å opprette anbefalinger.

Forklaring

Du klikket på **Stopp** mens optimaliseringsrådgiveren var i ferd med å opprette optimaliseringsanbefalinger.

Gjør slik:

Klikk på **Ja** for å stoppe optimaliseringsrådgiveren og returnere anbefalingene den har innhentet så langt. Klikk på **Nei** hvis du vil at optimaliseringsrådgiveren skal fortsette å opprette anbefalinger. Hvis du klikker på **Ja**, viser veiviseren hvilke anbefalinger som er opprettet så langt.

10300: Kunne ikke analysere målet *målnavn* som var oppgitt i SQL-uttrykket.

Forklaring

SQL-uttrykket som er oppgitt, refererer til et mål som enten er ugyldig i denne konteksten eller ikke finnes i databasen.

Gjør slik:

Kontroller at SQL-uttrykket bare refererer til de målene som vises i datalistene for SQL Expression Builder.

10301: Kunne ikke analysere kolonnen *kolonnenavn* som var oppgitt i SQL-uttrykket.

Forklaring

SQL-uttrykket som er oppgitt, refererer til en kolonne som enten er ugyldig i denne konteksten eller ikke finnes i databasen.

Gjør slik:

Kontroller at SQL-uttrykket bare refererer til de kolonnene som vises i datalistene for SQL Expression Builder.

10302: Attributtet *attributtnavn* er ikke kvalifisert med et skjemanavn.

Forklaring

Referanser til attributter i SQL-uttrykket må være kvalifisert med et skjemanavn, atskilt med '.'.

Gjør slik:

Kontroller at alle referansene til attributter i SQL-uttrykket er kvalifisert med et skjemanavn, atskilt med '.'.

10303: Målet *målnavn* er ikke kvalifisert med et skjemanavn.

Forklaring

Referanser til mål i SQL-uttrykket må være kvalifisert med et skjemanavn, atskilt med '.'.

Gjør slik:

Kontroller at alle referansene til mål i SQL-uttrykket er kvalifisert med et skjemanavn, atskilt med '.'.

10304: Manglende objektnavn i objektkoden *objektnavn*.

Forklaring

SQL-uttrykket som er oppgitt, har en tom kolonnekode @Column, en tom attributtkode @Attribute eller en tom målkode @Measure.

Gjør slik:

Kontroller at objekttypekodene @Column, @Measure og @Attribute har et omsluttet objektnavn.

10305: Det oppgitte uttrykket er ugyldig. Databasen returnerte den følgende informasjonen: *melding*.

Forklaring

Det er en syntaksfeil i SQL-uttrykket. Denne feilen vises også når SQL-uttrykket refererer til kolonner, attributter eller mål uten omsluttende koder. En referanse til en kolonne, et attributt eller et mål må være omsluttet av henholdsvis koden @Column(), @Attribute() eller @Measure().

Gjør slik:

Rett opp syntaksfeilen. Kontroller at alle kolonner, attributter og mål er omsluttet av den riktige koden.

10306: Datatypen for uttrykket du har oppgitt, er ikke-numerisk. Oppgi et numerisk uttrykk som en andre parameter.

Forklaring

Datatypen til den andre parameteren må være numerisk.

Gjør slik:

Kontroller at datatypen for uttrykket du har oppgitt, gir en numerisk datatype.

10307: Uttrykket for målet *målnavn* resulterer i en ikke-numerisk datatype. Velg et mål med et uttrykk som resulterer i en numerisk datatype.

Forklaring

Datatypen til den andre parameteren må være numerisk.

Gjør slik:

Kontroller at datatypen for det valgte målets uttrykk er numerisk.

10308: OLAP-senteret kan ikke kommunisere med den oppgitte databasen. Dette kan komme av at databasen ikke er riktig konfigurert for DB2 Cube Views. Det kan ta litt tid å konfigurere databasen. Klikk på Ja hvis du vil konfigurere den oppgitte databasen. Klikk på Nei hvis du ikke vil konfigurere den oppgitte databasen nå.

Forklaring

OLAP-senteret kan kobles til databasen med bruker-IDen og passordet som er oppgitt, men det kan ikke kommunisere med APIen for den lagrede prosedyren.

Mulige årsaker:

- APIen for den lagrede DB2 Cube Views-prosedyren er ikke registrert for den oppgitte databasen.
- DB2 Cube Views-katalogtabellene finnes ikke for den oppgitte databasen.

Gjør slik:

Klikk på **Ja** hvis du vil konfigurere databasen for DB2 Cube Views. Hvis ikke, klikker du på **Nei**.

10309: OLAP-senteret kan ikke tilkoble til den oppgitte databasen, fordi databasen er konfigurert for en eldre versjon av DB2 Cube Views. Databasen må migreres til den gjeldende versjonen av DB2 Cube Views. Klikk på Ja hvis du vil at OLAP-senteret skal migrere den oppgitte databasen. Klikk på Nei hvis du ikke vil at OLAP-senteret skal migrere den oppgitte databasen.

Forklaring

OLAP-senteret kan kobles til databasen med bruker-IDen og passordet som er oppgitt, men det kan ikke hente metadata fra katalogen. Dette kan skyldes at DB2 Cube Views-katalogtabellene er konfigurert for en eldre versjon av DB2 Cube Views.

Gjør slik:

Klikk på **Ja** for å migrere DB2 Cube Views-katalogen til riktig versjon, eller klikk på **Nei**.

10310: Databasen *databasenavn* ble konfigurert.

Forklaring

OLAP-senteret opprettet DB2 Cube Views-katalogtabellene og registrerte APIen for den lagrede prosedyren for databasen som er oppgitt.

Gjør slik:

Du trenger ikke å gjøre noe.

10311: OLAP-senteret kan ikke konfigurere databasen for DB2 Cube Views. Databasen returnerte følgende informasjon:
informasjon.

Forklaring

OLAP-senteret kan ikke konfigurere den oppgitte databasen for DB2 Cube Views.

Mulige årsaker:

- OLAP-senteret kan ikke registrere APIen for den lagrede DB2 Cube Views-prosedyren.
- OLAP-senteret kan ikke opprette en eller flere DB2 Cube Views-katalogtabeller.

Gjør slik:

Kontroller at du har de nødvendige autorisasjonene for installering og konfigurering, som er beskrevet i emnet Autorisasjoner og rettigheter i hjelpen til OLAP-senteret. Installerings- og brukerhåndboken for DB2 Cube Views inneholder opplysninger om hvordan du konfigurerer en database.

10312: Samlingsvalideringen mislyktes. En eller flere oppgitte samlingsfunksjoner er ikke kompatible med kilde-SQL-uttrykket.

Forklaring

En eller flere oppgitte samlingsfunksjoner er ikke kompatible med kilde-SQL-uttrykket. Dette kan være fordi samlingsfunksjonen som er oppgitt, forventer en parameter med en datatype som er forskjellig fra kilde-SQL-uttrykkets datatype.

Gjør slik:

Kontroller at samlingsfunksjonen er gyldig for det oppgitte målets kildedatatype.

10313: Målets kildeuttrykk har riktig syntaks bare med samlingsinnstillingen Ingen. Målet må bruke samlingsinnstillingen Ingen.

Forklaring

Målet forventer samlingsinnstillingen Ingen i disse situasjonene:

- SQL-uttrykket er syntaktisk feil når samlingsfunksjonene ikke brukes på målene det refereres til, men den er syntaktisk riktig når disse samlingsfunksjonene brukes. For eksempel er char + int syntaktisk feil, men COUNT(char) + SUM(int) er syntaktisk riktig.
- SQL-uttrykket bruker OLAP-funksjoner som RANK(), DENSE_RANK() og ROW_NUMBER().

Gjør slik:

Kontroller at målet har samlingsinnstilling Ingen.

10401: Uttrykket kan ikke inneholde en kolonnefunksjon, en skalar, full SELECT-setning eller en delspørring.

Forklaring

SQL-uttrykket kan ikke inneholde en kolonnefunksjon, en skalar, full SELECT-setning eller en delspørring.

Gjør slik:

Rett bruken av kolonnefunksjonen for å fjerne det ugyldige uttrykket.

10501: Skjemanavnet kan ikke starte med *prefiks*.

Forklaring

Skjemanavnet kan ikke starte med 'SYS' og 'SESSION'.

Gjør slik:

Oppgi et annet skjemanavn.

10502: Kombineringssegenskapene er ikke gyldige for optimalisering av kubemodellens ytelse. Løs dette problemet og kjør deretter veiviseren for optimaliseringsrådgiveren på nytt. Databasen returnerte den følgende informasjonen: *melding*.

Forklaring

Kombineringssegenskapene er ugyldige for optimalisering av kubemodellens ytelse.

Gjør slik:

Oppgi riktige innstillinger for kombineringsreglene ved å bruke reglene for optimaliseringsvalidering.

10503: Hierarkiet kan ikke endres fordi det har et tilknyttet kubehierarki.

Forklaring

Hvis det finnes et kubehierarki for hierarkiet, kan ikke hierarkiet endres.

Gjør slik:

Kontroller at ingen kubehierarkier refererer til hierarkiet du skal endre, før du endrer hierarkiet. Du kan også opprette et annet hierarki med de nødvendige endringene.

10504: Dette målet må bruke samlingsinnstillingen Ingen fordi det refererer til målet som bruker en samlingsfunksjon med flere parametere.

Forklaring

Bare mål som bruker samlingsinnstillingen Ingen, kan referere til mål som bruker en funksjon med flere parametere. Du kan ikke endre samlingsinnstillingen fra Ingen til en annen funksjon.

Gjør slik:

Utfør en av disse handlingene:

- Ikke endre målets samlingsinnstilling.
- Endre det oppgitte målet slik at det ikke bruker en funksjon med flere parametere.

10505: Dette målet kan ikke bruke en funksjon med flere parametere fordi målet *målnavn* som bruker en annen samlingsinnstilling enn Ingen, refererer til dette målet.

Forklaring

Bare mål som bruker samlingsinnstillingen Ingen, kan referere til mål som bruker en funksjon med flere parametere. Du kan ikke endre samlingsinnstillingen for målet du redigerer, for å ta med en funksjon med flere parametere, fordi det er referert til målet av et mål som ikke bruker samlingsinnstillingen Ingen.

Gjør slik:

Utfør en av disse handlingene:

- Ikke endre målets samlingsinnstilling.

- Endre det oppgitte målet slik at det ikke refererer til målet du redigerer.

10506: Den eksisterende samlingsinnstillingen er ugyldig med det oppgitte SQL-uttrykket. OLAP-senteret tilbakestill samlingsinnstillingen til *innstilling*.

Forklaring

Den eksisterende samlingsinnstillingen er ugyldig sammen med det nye SQL-uttrykket, og den ble tilbakestillt til standard samlingsinnstilling. Mulige årsaker:

- Datatypen til kilde-SQL-uttrykket ble endret.
- Den gjeldende samlingsinnstillingen er forventet å være Ingen. Den må være Ingen i disse tilfellene:
 - SQL-uttrykket er syntaktisk feil når samlingsfunksjonene ikke brukes på målene det refereres til, men den er syntaktisk riktig når disse samlingsfunksjonene brukes. For eksempel er char + int syntaktisk feil, men COUNT(char) + SUM(int) er syntaktisk riktig.
 - SQL-uttrykket bruker OLAP-funksjoner som RANK(), DENSE_RANK() og ROW_NUMBER().

Gjør slik:

Du trenger ikke å gjøre noe.

10507: En eller flere dimensjoner i kubemodellen har ikke noe hierarki. De er ikke tilgjengelige for inkludering i kubene.

Forklaring

Når en kubedimensjon skal opprettes, må den være basert på en dimensjon som har minst ett hierarki. Du prøver å opprette eller endre en kube som har en eller flere dimensjoner som ikke har et hierarki. Disse dimensjonene blir utelatt fra valglisten som brukes til definering av kubedimensjoner.

Gjør slik:

Opprett eller endre kubene uten referanser til de utelatte dimensjonene, eller sørg for at hver dimensjon i kubemodellen har et hierarki.

10508: Alternativet for rekursiv distribuering er bare gyldig hvis to nivåer er valgt for hierarkiet.

Forklaring

Nøyaktig to nivåer må finnes for en rekursiv distribuering.

Gjør slik:

Velg enten nøyaktig to nivåer for rekursiv distribuering, eller velg et annet distribueringalternativ.

10509: Du oppgav ikke alle de nødvendige egenskapene for optimaliseringssektorene. Oppgi en spørringstype og et alternativ for hver kubedimensjon for hver optimaliseringssektor.

Forklaring

Du oppgav ikke alle de nødvendige egenskapene for optimaliseringssektorene.

Gjør slik:

Oppgi en spørningstype og et alternativ for hver kubedimensjon for hver optimaliseringssektor.

10510: Velg minst ett nivånøkkelattributt for nivået.

Forklaring

Det er ikke oppgitt noen nivånøkkelattributter for nivået.

Gjør slik:

Velg minst ett nivånøkkelattributt for nivået.

10511: Du kan ikke oppgi både en sektor av typen MOLAP-uttrekking og en sektor av typen hybriduttrekking for den samme kuben. Du må fjerne en av sektorene fra kuben.

Forklaring

En kube kan ikke ha både en sektor av typen MOLAP-uttrekking og en sektor av typen hybriduttrekking. Du kan bare oppgi en optimaliseringstype uttrekking per kube.

Gjør slik:

Fjern sektoren av typen MOLAP-uttrekking eller sektoren av typen hybriduttrekking.

10512: Maksimalt en optimaliseringssektor av typen MOLAP-uttrekking er tillatt per kube.

Forklaring

Maksimalt en optimaliseringssektor av typen MOLAP-uttrekking er tillatt per kube.

Gjør slik:

Kontroller at du ikke har mer enn en optimaliseringssektor av typen MOLAP-uttrekking.

10513: Maksimalt en optimaliseringssektor av typen hybriduttrekking er tillatt per kube.

Forklaring

Maksimalt en optimaliseringssektor av typen hybriduttrekking er tillatt per kube.

Gjør slik:

Kontroller at du ikke har mer enn en optimaliseringssektor av typen hybriduttrekking.

10514: Du kan bare opprette en optimaliseringssektor av typen neddrilling hvis en optimaliseringssektor av typen hybriduttrekking allerede er definert for kubene.

Forklaring

En optimaliseringssektor for hybriduttrekking er nødvendig for å angi en optimaliseringssektor av typen gjennomdrilling.

Gjør slik:

Opprett en optimaliseringssektor av typen hybriduttrekking. Deretter kan du opprette en optimaliseringssektor av typen neddrilling.

10515: DB2 Cube Views oppgav Vilkårlig som optimaliseringsnivå for de nye kubedimensjonene for hver av de eksisterende optimaliseringssektorene.

Forklaring

Hvis du tilføyer kubedimensjoner til en kube etter at du har definert optimaliseringssektorer for kubene, utvider DB2 Cube Views de eksisterende optimaliseringssektorene ved å angi Vilkårlig som optimaliseringsnivå for de nye kubedimensjonene.

Gjør slik:

Du kan endre optimaliseringsnivåene for sektorene ved å klikke på Spesifiser på siden Spørringstyper i vinduet Kubeegenskaper.

10516: Du må oppgi minst en optimaliseringssektor i vinduet Optimaliseringssektorer, fordi du oppgav avanserte innstillinger for spørringstypen til kubene.

Forklaring

Du må oppgi minst en optimaliseringssektor i vinduet Optimaliseringssektorer, fordi du valgte Avanserte innstillinger for spørringstypen til kubene.

Gjør slik:

Klikk på Spesifiser og opprett minst en optimaliseringssektor i vinduet Optimaliseringssektorer.

10517: Eventuelle endringer for kubene og kubedimensjonene blir lagret når du lukker siden Dimensjoner. Klikk på Ja hvis du vil lagre endringene i databasen. Klikk på Nei hvis du fortsatt vil være på siden Dimensjoner for å gjøre flere endringer eller avslutte.

Forklaring

Når du har endret kubedimensjonene i vinduet Kubeegenskaper og forlater siden Dimensjoner, blir alle endringer av kubedimensjonene lagret i databasen, og endringene kan ikke oppheves ved å klikke på Avbryt.

Gjør slik:

Klikk på Ja hvis du vil lagre endringene. Klikk på Nei hvis du vil avbryte transaksjonen og bli værende på siden Dimensjoner.

10518: Du kan ikke endre kubehierarkiet fordi du allerede har definert avanserte optimaliseringssektorer for kuben *kubenavn*. Du må slette alle de avanserte optimaliseringssektorene, og deretter kan du endre kubehierarkiet.

Forklaring

Du kan ikke endre kubehierarkiet hvis du allerede har definert avanserte optimaliseringssektorer for kuben.

Gjør slik:

Du må slette alle de avanserte optimaliseringssektorene, og deretter kan du endre kubehierarkiet.

10519: Du kan ikke endre settet med kubedimensjoner fordi du allerede har definert avanserte optimaliseringssektorer for kuben *kubenavn*. Du må slette alle de avanserte optimaliseringssektorene før du kan tilføye eller slette kubedimensjoner fra kuben.

Forklaring

Du kan ikke endre settet med kubedimensjoner hvis du allerede har definert avanserte optimaliseringssektorer for kuben.

Gjør slik:

Du må slette alle de avanserte optimaliseringssektorene før du kan tilføye eller slette kubedimensjoner fra kuben.

10520: Endringene i egenskapene kan føre til at også de eksisterende optimaliseringssektorene for kuben blir endret. Du kan se gjennom optimaliseringssektorene for å vise og endre endringene, eller du kan klikke på Avbryt i egenskapsvinduet for å lukke vinduet uten å lagre endringene.

Forklaring

Hvis du endrer settet med kubedimensjoner eller kubehierarkiet, blir de eksisterende avanserte optimaliseringssektorene for kuben endret på en av disse måtene:

- Hvis det blir tilføyd en ny kubedimensjon, blir alle eksisterende optimaliseringssektorer utvidet, slik at de inkluderer den nye kubedimensjonen på nivået Vilkårlig.
- Hvis en kubedimensjon blir fjernet, blir det tilhørende nivået fjernet fra alle eksisterende optimaliseringssektorer.
- Hvis et kubehierarki endres slik at nivået som er definert i optimaliseringssektorene, ikke lenger finnes, blir det manglende hierarkinivået erstattet med nivået Vilkårlig i alle aktuelle optimaliseringssektorer.
- Hvis et eksisterende kubehierarki blir erstattet med et nytt kubehierarki, blir nivåene i optimaliseringssektorene for den tilsvarende kubedimensjonen, satt til Vilkårlig.

Gjør slik:

Åpne vinduet Optimaliseringssektorer og bekreft endringene til OLAP-senteret. Gjør eventuelle endringer.

Statusmeldinger fra DB2 og DB2 Cube Views

Når det blir sendt et kall til den lagrede prosedyren for DB2 Cube Views, uavhengig av om den lagrede prosedyren ble utført eller ikke, returnerer DB2 en SQLCODE og en SQLSTATE til applikasjonen som sendte kallet. Hvis den lagrede prosedyren for DB2 Cube Views kan utføres, returnerer den lagrede prosedyren en statusmelding som en del av XML-dataene som blir sendt til applikasjonen som sendte kallet.

Tabellen nedenfor viser forholdet mellom statusmeldingene som returneres av metadataoperasjoner, og SQLSTATE-kodene som returneres av DB2 for kallet til den lagrede prosedyren.

Tabell 63. Metadataoperasjons-IDer og SQLSTATE-koder

SQL CODE	SQL STATE	Statusmeldings-IDer for metadataoperasjoner	Statusmeldings-typer for metadataoperasjoner	Statusmeldinger returnert for metadataoperasjoner
0	0	0 2	Informasjon	No
0	0	1	Informasjon	Yes
0	0	599 6006 6299 7200 7201 7202	Advarsel	No
462	01HQ1	0 – 7999 (unntatt IDer oppført i andre rader)	Feil	No
443	38Q00	Gjelder ikke	Gjelder ikke	Gjelder ikke
443	38Q01	Gjelder ikke	Gjelder ikke	Gjelder ikke
443	38Q02	Gjelder ikke	Gjelder ikke	Gjelder ikke
443	38Q03	Gjelder ikke	Gjelder ikke	Gjelder ikke

Merknader

Denne boken er utarbeidet for produkter og tjenester som er tilgjengelige i USA.

Henvisninger i boken til IBMs produkter, programmer eller tjenester betyr ikke at IBM har til hensikt å gjøre dem tilgjengelige i alle land der IBM driver virksomhet. Kontakt din lokale IBM-representant for å få informasjon om hvilke produkter og tjenester som er tilgjengelige i Norge. Henvisninger til IBMs produkter, programmer eller tjenester betyr heller ikke at det bare er de som kan benyttes. Andre produkter, programmer eller tjenester som har tilsvarende funksjoner, kan brukes i stedet, forutsatt at de ikke gjør inngrep i noen av IBMs patent- eller opphavsrettigheter eller andre lovbeskyttede rettigheter. Vurdering og verifisering ved bruk sammen med andre produkter, programmer eller tjenester enn de som uttrykkelig er angitt av IBM, er brukerens ansvar.

IBM kan ha patent på eller patentsøknader til behandling for de produktene som er omtalt i denne publikasjonen. At du har mottatt denne publikasjonen, innebærer ikke at du får lisensrettighet til disse produktene. Du kan sende spørsmål angående lisenser til

Director of Commercial Relations - Europe
IBM Deutschland GmbH
Schönaicher Str. 220
D - 7030 Böblingen
Tyskland

Lisensforespørsler om dobbeltbyteinformasjon (DBCS) kan rettes til IBMs advokat eller til:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

INTERNATIONAL BUSINESS MACHINES CORPORATION LEVERER DENNE BOKEN SOM DEN ER ("AS IS") UTEN FORPLIKTELSER AV NOE SLAG.

Denne boken kan inneholde tekniske unøyaktigheter eller typografiske feil. Opplysninger i denne boken kan bli endret. Slike endringer blir tatt med i nye utgaver av boken. IBM kan uten varsel endre produktene og/eller programmene som er beskrevet i denne boken.

IBM kan bruke eller distribuere informasjonen du gir til IBM på den måten IBM mener er best, uten forpliktelser i noen retning.

Hvis du som lisensinnehaver av dette programmet ønsker informasjon om programmet for å kunne (i) utveksle informasjon mellom selvstendig utviklede programmer og andre programmer (inkludert dette) og (ii) dra gjensidig nytte av informasjonen som er utvekslet, kan du kontakte:

IBM Norge
Software Marketing
Postboks 500
1411 KOLBOTN
Tyskland

Slik informasjon kan være tilgjengelig under gjeldende betingelser, eventuelt mot betaling.

Det lisensierte programmet som er beskrevet i denne boken, og alt lisensiert materiale som er tilgjengelig for programmet, leveres av IBM i henhold til IBMs generelle betingelser, IBMs internasjonale bruksbetingelser eller en tilsvarende avtale mellom partene.

Informasjon om andre produkter enn IBMs egne er hentet fra leverandørene av produktene, fra deres annonseringer eller fra andre tilgjengelige kilder. IBM har ikke testet disse produktene, og kan ikke bekrefte påstander om ytelse, kompatibilitet eller andre egenskaper ved dem. Spørsmål om funksjonene i ikke-IBM-produkter må rettes til leverandøren av produktet.

Denne boken kan inneholde eksempler på data og rapporter som brukes i den daglige driften av et firma. For å illustrere eksemplene så godt som mulig, blir det brukt navn på personer, firmaer og produkter. Alle disse navnene er fiktive, og enhver likhet med virkelige navn er tilfeldig.

Alle ytelsesdataene du finner i dette dokumentet, ble hentet i et kontrollert miljø. Resultatene du kan oppnå i andre operativmiljøer, kan variere betraktelig. Noen av målingene er foretatt på systemer som er under utvikling, og det er ikke sikkert at du oppnår samme resultat på alminnelige tilgjengelige systemer. Noen av målingene kan dessuten ha blitt beregnet ved hjelp av ekstrapolasjon. De faktiske resultatene kan variere. Brukerne av dette dokumentet bør bekrefte dataene som brukes i sitt bestemte miljø.

Informasjon om andre produkter enn IBMs egne er hentet fra leverandørene av produktene, fra deres annonseringer eller fra andre tilgjengelige kilder. IBM har ikke testet disse produktene, og kan ikke bekrefte påstander om ytelse, kompatibilitet eller andre egenskaper ved dem. Spørsmål om funksjonene i ikke-IBM-produkter må rettes til leverandøren av produktet.

Hvis du ser på den maskinleselige versjonen av denne boken, kan det hende at fotografier og fargeillustrasjoner ikke blir vist.

Varemerker

Navnene nedenfor er varemerker for International Business Machines.

AIX
DB2
DB2 Connect
DB2 Universal Database
IBM
Office Connect
Redbooks

Navnene nedenfor er varemerker for andre selskaper.

Microsoft, Windows, Windows NT, Windows 2000, Windows XP og Microsoft Excel er varemerker eller registrerte varemerker for Microsoft Corporation.

Java og alle Java-baserte varemerker og logoer, samt Solaris er varemerker for Sun Microsystems, Inc.

UNIX er et registrert varemerke i USA og/eller andre land, og lisensieres kun gjennom X/Open Company Limited.

Linux er et registrert varemerke for Linus Torvalds. Red Hat og alle Red Hat-baserte varemerker og logoer er varemerker eller registrerte varemerker for Red Hat, Inc. i USA og/eller andre land.

Andre navn på selskaper, produkter eller tjenester kan være varemerker for andre selskaper.

Ordlister

Denne ordlisten definerer termer som blir brukt i denne boken.

samlingsfunksjon. En av DB2 SQL-samlingsfunksjonene som SUM, AVG, MIN og MAX. Samlingsfunksjoner blir brukt til å styre hvordan opprullinger utføres på mål.

attributt. Et DB2-objekt som er tilordnet enten en enkelt kolonne i en tabell eller et uttrykk som er en kombinasjon av et sett med kolonner eller andre attributter, eller begge deler. Et attributt kan utføre flere roller. Det kan for eksempel referere til data som finnes i kubene, eller det kan være en referanse til en kolonne som brukes av en kombinerings-, eller andre attributtforhold.

attributtforhold. Beskriver generelle relasjonsforhold for attributtobjekter. Forholdene beskrives med et venstre- og et høyreattributt, en type, en kardinalitet, og om de bestemmer en funksjonell avhengighet. Typen beskriver hvilken rolle høyreattributtet spiller i forhold til venstreattributtet. Det finnes to mulige typer: Beskrivende og Tilknyttet. Den beskrivende typen oppgir at høyreattributtet er en beskriver av venstreattributtet.

balansert hierarki. Et hierarki med meningsfulle nivåer og forgreninger som har en konsistent dybde. Det overordnede nivået for hvert attributt er nivået direkte over det. Se *nettverkshierarki*, *ujevnt hierarki* og *ubalansert hierarki*.

beregnet mål. Inneholder innebygde beregninger som du oppretter ved hjelp av Expression Builder i OLAP-senteret eller med SQL. Et beregnet mål har et SQL-uttrykk som utfører beregninger og ikke tilordner til en enkelt kolonne eller ett enkelt attributt.

begrensning. En regel som databasesystemet iverksetter. Det finnes fire typer begrensninger: entydige begrensninger, referansebegrensninger, tabellkontrollbegrensninger og informasjonsbegrensninger.

kube. Et DB2-objekt som er utledet fra en kubemodell. Kubefaktaene og kubedimensjonene er delsett av dem det refereres til i kubemodellen. Kubene passer til verktøy og applikasjoner som ikke bruker flere hierarkier, fordi kubedimensjonene bare tillater ett kubehierarki per kubedimensjon.

kubedimensjon. Et DB2-objekt som er en del av en kube, og som er utledet fra en dimensjon i kubemodellen som svarer til kubene. En kubedimensjon

refererer til et delsett av attributtene til dimensjonen den er utledet fra. Den refererer også til ett enkelt kubehierarki.

kubefakta. Et DB2-objekt som er en del av en kube, og som er utledet fra en dimensjon i kubemodellen som svarer til kubene. Et kubefaktaobjekt refererer til et delsett av målene fra faktaobjektet det er utledet fra.

kubehierarki. Et DB2-objekt som er en del av en kubedimensjon, og som er utledet fra et hierarki i dimensjonen som svarer til kubedimensjonen. Et kubehierarki refererer til et delsett av attributtene til hierarkiet det er utledet fra, der rekkefølgen til attributtene må være den samme som i hierarkiet.

kubenivå. Et DB2-objekt er et delsett av et nivå, og det brukes i en kube. En kubnivåreferanse refererer til nivået den er utledet fra (det overordnede nivået), og arver nivånøkkelattributtene og standardattributtet som er definert for det overordnede nivået.

kubemodell. Et DB2-objekt som beskriver alle dataene som er knyttet til en samling av mål. Vanligvis er kubemodellen knyttet til et stjerneskjema eller snøfnuggskjema i databasen. Kubemodellen refererer til ett enkelt faktaobjekt og en eller flere dimensjoner. Kubemodeller kan optimaliseres for å forbedre ytelsen til SQL-spørringer som sendes til stjerneskjemaet eller snøfnuggskjemaet til kubemodellen.

dimensjon. Et DB2-objekt som refererer til en samling av beslektede attributter som beskriver enkelte aspekter av et sett med mål. En dimensjon kan referere til attributter fra en eller flere dimensjonstabeller. Hvis attributter fra flere dimensjonstabeller blir brukt, må det imidlertid finnes kombinerings mellom tabellene, og dimensjonen må referere til disse kombineringsene. En dimensjon refererer også til ett eller flere hierarkier og kan referere til relasjoner mellom dimensjonens attributter.

dimensjonstabell. En tabell i et datavarehus, med tabellposter som beskriver dataene i en faktatabell. Dimensjonstabeller inneholder data som dimensjoner opprettes fra.

faktaobjekt. Et DB2-objekt som grupperer beslektede mål som er interessante for en bestemt applikasjon. Faktaobjektet lagrer informasjon om attributtene som er brukt i kombineringsene av faktaobjekt og dimensjon, og attributtene og kombineringsene som brukes til å tilordne de ekstra målene på tvers av flere databasetabeller. I tillegg til et sett av mål, lagrer et faktaobjekt derfor et sett av attributter og et sett av kombineringsene. Et faktaobjekt brukes i en kubemodell som sentrum i et stjerneskjema.

faktatabell. En sentral tabell i et datavarehuskjema, som inneholder numeriske mål og nøkler som knytter fakta til dimensjonstabeller. Faktatabeller inneholder data som beskriver bestemte hendelser innefor et firma, for eksempel banktransaksjoner eller produktsalg.

funksjonell avhengighet. Et DB2-objekt som angir at nivåobjektets standardattributt og beslektede attributter er funksjonelt bestemt av nivåets nøkkelattributter. Med funksjonelle avhengigheter kan du oppgi at en eller flere kolonner er funksjonelt avhengige av en eller flere andre kolonner, såfremt alle kolonnene finnes i den samme tabellen.

hierarki. Et DB2-objekt som definerer forhold mellom et sett av ett eller flere attributter i en bestemt dimensjon av en kubemodell. DB2 Cube Views støtter fire typer av hierarkier, balanserte, ubalanserte og ujevne hierarkier samt nettverkshierarkier. Distribuering av hierarkier kan være standard eller rekursiv.

hybridkube. Inneholder flerdimensjonale data og refererer til relasjonsdata slik at du kan utføre spørring på data på lavere nivå i basistabellene.

kombinering. Kombinerer to relasjonstabeller. En kombinering refererer til attributter som deretter refererer til kolonner i tabellene som blir kombinert. Den enkleste formen for kombinering refererer til to attributter, ett som tilordnes til en kolonne i den første tabellen, og ett som tilordnes til en kolonne i den andre tabellen. Kombineringen inneholder også en operator som oppgir hvordan kolonnene sammenliknes. Et kombineringsobjekt kan også brukes til å modellere sammensatte kombineringer, der to eller flere kolonner fra den første tabellen kombineres med samme antall kolonner i den andre tabellen. En sammensatt kombinering bruker par av attributter til å tilordne tilsvarende kolonner. Hvert par av attributter har en operator som oppgir hvordan dette paret av kolonner sammenliknes. En kombinering har også en type og en kardinalitet. Kombineringer kan brukes i dimensjoner for å kombinere dimensjonstabeller, eller i en kubemodell for å kombinere kubemodellens dimensjoner til faktaobjektet, eller i et faktaobjekt for å kombinere flere faktatabeller.

nivå. Et DB2-objekt som består av ett eller flere attributter som er beslektede, og som virker sammen som ett logisk trinn i en hierarkirekkefølge.

materialisert spørretabell. En tabell der definisjonen er basert på resultatet av en spørring, og der dataene er forhåndsbergnede resultater som er hentet fra en eller flere tabeller som den materialiserte spørretabelldefinisjonen er basert på.

mål. Et DB2-objekt som definerer en målentitet og brukes i faktaobjekter. Mål får mening i forbindelse med en dimensjon. Vanlige eksempler på målobjekter er Inntekt, Kostnad og Fortjeneste.

metadata. Informasjon om egenskapene til data, for eksempel datatypen i en kolonne (numerisk, tekst, osv.) eller lengden til en kolonne. Det kan også være informasjon om strukturen til data eller informasjon som spesifiserer utformingen av objekter som kuber eller dimensjoner.

MQT. Se *materialisert spørretabell*.

nettverkshierarki. Et hierarki der rekkefølgen for nivåene ikke er oppgitt, men der nivåene har semantisk betydning. Siden attributtnivåene ikke har et over-/underordnet forhold, er rekkefølgen for nivåene ikke viktig. Se *balansert hierarki*, *ujevnt hierarki* og *ubalansert hierarki*.

utriggertabell. En dimensjonstabell i et snøfnuggskjema, som ikke er den primære dimensjonstabellen i dimensjonen.

primær dimensjonstabell. Den dimensjonstabellen i et snøfnuggskjema, som kombineres med faktatabellen.

ujevnt hierarki. Et hierarki der hvert nivå har en konsistent betydning, men grenene har inkonsistente dybder fordi minst ett medlemsattributt på et grennivå ikke er utfylt. Se *balansert hierarki*, *nettverkshierarki* og *ubalansert hierarki*.

rekursiv distribuering. Bruker de over-/underordnede forholdene mellom hierarkiets attributter. Et ubalansert hierarki som bruker en rekursiv distribuering, representeres av paret av over- og underordnede attributter.

skjema. Ifølge standarden SQL-92 en samling med databaseobjekter som eies av en enkelt bruker og utgjør ett enkelt navneområde. Et navneområde er et sett med objekter som ikke kan ha like navn. To tabeller kan for eksempel bare ha like navn dersom de er i separate skjemaer. To tabeller i det samme skjemaet kan ikke ha samme navn.

snøfnuggskjema. En utvidelse av et stjerneskjema, slik at en eller flere dimensjoner er definert av flere tabeller. I et snøfnuggskjema kombineres bare primære dimensjonstabeller med faktatabellen. Andre dimensjonstabeller kombineres med de primære dimensjonstabellene.

standarddistribuering. Bruker nivådefinisjonene for hierarkiet, der hvert attributt i hierarkiet definerer ett nivå. Et balansert hierarki for en tidsdimensjon er for eksempel vanligvis organisert av hvert definerte nivå, blant annet År, Kvartal og Måned. Distribueringsmekanismen Standard kan brukes med alle fire hierarkitypene.

stjernekombinering. En kombinering mellom en faktatabell (vanligvis en stor faktatabell) og minst to dimensjonstabeller. Faktatabellen blir kombinert med hver dimensjonstabell i en dimensjonsnøkkel.

stjerneskjema. En relasjonsdatabasestruktur der data blir vedlikeholdt i en enkelt faktatabell i sentrum av skjemaet, med dimensjonsdata lagret i dimensjonstabeller. Hver dimensjonstabell er direkte relatert til og vanligvis kombinert med faktatabellen ved hjelp av en nøkkelkolonne. Stjerneskjemaer brukes i datavarehus.

sammendragstabell. Inneholder samlede data fra basistabellene, som brukes av kubemodellen. DB2 Cube Views bruker DB2-sammendragstabeller for å forbedre ytelsen til spørringer som utføres på kubemodeller. En sammendragstabell er en spesiell type materialisert spørretabell (MQT) som spesifikt tar med sammendragsdata. Fordi DB2 Cube Views alltid anbefaler materialiserte spørretabeller som har sammendragsdata, blir termen sammendragstabell (summary table) brukt i dokumentasjonen til DB2 Cube Views som en beskrivelse av anbefalte materialiserte spørretabeller. Se *materialisert spørretabell*.

sektor. Et område av en flerdimensjonal database eller kube.

ubalansert hierarki. Et hierarki med nivåer som har et konsistent forhold mellom over- og underordnede nivåer, men som har en inkonsistent semantisk betydning for alle medlemmene på et bestemt nivå. Hierarkigrenene har også inkonsistente dybder. Se *balansert hierarki*, *nettverkshierarki* og *ujevnt hierarki*.

Stikkordregister

A

- administrasjonsoperasjoner 138
- Alter-operasjonen 150
- API (programmeringsgrensesnitt)
 - datautveksling 129
 - lagret prosedyre 131
 - oversikt 129
 - parametere 136
 - sporing 157
- attributter
 - beskrivelse 32
 - beslektede 27
 - egenskaper 33
 - grunnleggende regler 40
 - i sammendragstabeller 86
 - nivånøkkel 27
 - relasjonstilordning 12, 15
 - standard 27
- attributtforhold
 - beskrivelse 33
 - egenskaper 34
 - grunnleggende regler 41
 - i sammendragstabeller 86
 - typer
 - beskrivende 33
 - tilknyttede 33
- autorisasjoner og rettigheter 44

B

- balanserte hierarkier 23
- begrensninger 82, 114, 127
 - fremmednøkkel 107
 - informasjon 107
 - sammendragstabeller 88
- Beregne
 - flyt 59
 - fortjeneste 69
 - fortjenestemargin 69
 - verdi 59
- beregnete mål 50
- beslektede attributter
 - beskrivelse 27
 - tilføye et kubehierarki 55
- broer 3

C

- Create-operasjonen 149
- currentRef, operand 142
- CVSAMPLE, database 165, 166

D

- databaser
 - fjern 124
- datakilder
 - fjern 124, 126

- DB2-database
 - koble fra 46
 - koble til 46
 - slette metadataobjekter 57
- DB2 EXPLAIN, funksjon 85
- DB2-optimalisator 82, 88
- DB2 SQL Snapshot Monitor 119
- db2batch, ytelsesverktøy (benchmark) 118
- Describe-operasjonen 143
- dimensjoner
 - beskrivelse 22
 - egenskaper 22
 - fjerne fra en kubemodell 56
 - grunnleggende regler 39
 - hierarkier, opprette 52
 - opprette 51
 - optimaliseringsregler 39
 - relasjonstilordning 12, 15, 17
 - tilføye 53
- dimensjonsveiviseren 51
- Drop-operasjonen 151

E

- egenskaper
 - attributter 33
 - attributtforhold 34
 - dimensjoner 22
 - faktaobjekter 22
 - felles 19
 - hierarkier 26
 - kombineringer 35
 - kubedimensjoner 37
 - kubefakta 36
 - egenskaper 36
 - kubehierarkier 37
 - kubemodeller 21
 - kubenivåer 38
 - kuber 36
 - mål 32
 - nivåer 29
- eksempel
 - applikasjonsfiler 166
 - databasefiler 165
- eksempler
 - API-parametere
 - administrasjonsoperasjoner 138
 - endringsoperasjoner 138
 - henteoperasjoner 137
- eksportere
 - metadataobjekter 47
- endringsoperasjoner 138
- enkle mål 50

F

- fakta og dimensjon, kombineringer 53, 107
 - opprette 53

- faktaobjekter
 - beskrivelse 21
 - egenskaper 22
 - grunnleggende regler 39
 - opprette 50
 - relasjonstilordning 11
- Faktaveiviser 50
- feilbehandling 130
- feillogging 158
- fjern
 - datakilder 124
- fjertliggende datakilder
 - definere 126
- forent
 - tjenere 123
- forente
 - databaser 124
 - forente systemer 124
 - aktivere for DB2 Cube Views 126
 - begrensninger 127
 - datakilder 123
 - fjertliggende datakilder 126, 127
 - forent tjener 123
 - forente databaser 123
 - kallenavn 127
 - klienter 123
 - oversikt 123
 - forutsetninger 5
 - fullstendighetsregler
 - kubemodeller 39
 - funksjonelle avhengigheter
 - oppgi 52
 - sammendragstabeller 88

G

- grunnleggende regler 107
 - attributter 40
 - attributtforhold 41
 - dimensjoner 39
 - faktaobjekter 39
 - hierarkier 40
 - kombineringer 41
 - kubedimensjoner 42
 - kubefaktaobjekter 42
 - kubehierarkier 42
 - kubemodeller 38
 - kubenivåer 42
 - kuber 41
 - mål 40
 - nivåer 39

H

- hente- resultater 143
- henteoperasjoner 137
- hierarkier
 - beskrivelse 23
 - distribuering 25
 - egenskaper 26

hierarkier (*fortsettelse*)
 grunnleggende regler 40
 modellering for optimalisering 93
 opprette 52
 opprette nivåer 52
 relasjonstilordning 12, 17
 typer 23
hierarkiveviseren 52
hurtigstartveviseren 48

I
ideell modellering 28
ikke-ideell modellering 29
Import-operasjonen 152
importere
 metadataobjekter 47
Importveiser 47
informasjonsbegrensning 82, 107, 114
installere
 AIX 6
 Linux 6
 Solaris Operating System 6
 Windows 6
installeringskrav 5

K
kallenavn 127
kardinaliteter 33
kjøretidssporing 157
kodesett 160
kombineringer
 beskrivelse 35
 egenskaper 35
 fakta og dimensjon 107
 grunnleggende regler 41
 kardinaliteter 108
 opprette 53
 optimaliseringsregler 41
 relasjonstilordning 13, 15
 typer 108
kombineringer av faktaobjekt og
 dimensjon, opprette 51
kombineringsveviseren 53
konfigurasjonsfil 133, 159
Korrelasjon
 Annonsering 64
 Salg 64
krav
 maskinvare 5
 programvare 5
kubedimensjoner
 beskrivelse 37
 egenskaper 37
 grunnleggende regler 42
 relasjonstilordning 18
 tilføy et kubehierarki 55
kubefaktaobjekter
 beskrivelse 36
 grunnleggende regler 42
 relasjonstilordning 18
kubehierarkier
 beskrivelse 37
 egenskaper 37
 grunnleggende regler 42

kubehierarkier (*fortsettelse*)
 kubehierarkier
 oppgi 55
 oppgi 55
 relasjonstilordning 18
 tilføy i en kubedimensjon 55
kubemodeller
 beskrivelse 21
 dimensjoner, opprette 51
 dimensjoner, tilføy 53
 egenskaper 21
 faktaobjekter, opprette 50
 fjerne dimensjoner 56
 fullstendighetsregler 39, 107
 grunnleggende regler 38
 hierarkier, opprette 52
 opprette 49
 opprette med
 hurtigstartveviseren 48
 optimalisere 114
 optimaliseringsregler 39
 relasjonstilordning 11
Kubemodellveiser 49
kubenivåer
 beskrivelse 38
 egenskaper 38
 grunnleggende regler 42
kuber 55
 beskrivelse 35
 egenskaper 36
 grunnleggende regler 41
 modellering for optimalisering 94
 opprette 54
 relasjonstilordning 18
Kubeveiser 54

L
lagrede prosedyrer
 md_message 131
logge feil 158
logging
 sporing 157

M
materialiserte spørretabeller 82
md_message
 lagrede prosedyrer 131
med DB2 Cube Views 124
meldingsstruktur i API 156
metadatabroer 3
metadataobjekter 3
 attributter 32
 attributtforhold 33
 dimensjoner 22
 dimensjoner, opprette 51
 dimensjoner, tilføy 53
 eksportere 47
 faktaobjekter 21
 faktaobjekter, opprette 50
 format 135
 generelle egenskaper 19
 grunnleggende regler 38
 hierarkier 23
 hierarkier, opprette 52

metadataobjekter (*fortsettelse*)
 importere 47
 kombineringer 35
 kombineringer, opprette 53
 kubedimensjoner 37
 kubefaktaobjekter 36
 kubehierarkier 37
 kubemodeller 21
 kubemodeller, opprette 49
 kubemodeller, opprette med
 hurtigstartveviseren 48
 kubenivåer 38
 kuber 35
 kuber, opprette 54
 modellering for optimalisering 93
 mål 30
 navngivningsregler 20
 nivåer 27
 nivåer, opprette 52
 oversikt 11
 utveksle 46
metadataoperasjoner
 Alter 150
 Create 149
 Describe 143
 Drop 151
 Importer 152
 Rename 150
 Translate 155
 Validate 154
minnestyring 130
modellering
 ideell 28
 ikke-ideell 29
modus for import 141
modus for Validate 142
MOLAP-uttrekkingsspøringer 95
mål
 beregnete 50
 beskrivelse 30
 egenskaper 32
 enkle 50
 grunnleggende regler 40
 i sammendragstabeller 86
 modellering for optimalisering 93
 relasjonstilordning 11

N
neddrillingsspøringer 94
nettverkshierarkier 25
newRef, operand 142
nivåer
 beskrivelse 27
 beslektede attributter 27
 egenskaper 29
 grunnleggende regler 39
 ideell modellering 28
 ikke-ideell modellering 29
 modellering for optimalisering 93
 nivånøkkelattributter 27
 opprette 52
 relasjonstilordning 12, 15
 standardattributter 27
 tilføy et kubehierarki 55
nivånøkkelattributter
 beskrivelse 27

Nivåveiviser 52

O

- object, operand 142
- objectType-parameter 139
- omdirigerte spørringer 82, 88
- operander 142
- operasjonsoperander 142
- operasjonsparametere 139
 - modus for import 141
 - modus for Validate 142
- objectType 139
- recurse 140
- restriction 140
- oppdatere umiddelbart, sammendragstabeller 121
- oppdatering utsatt, sammendragstabeller 121
 - utløpe 120
- optimalisere
 - kubemodeller 114
- optimalisering 88
 - dataavlesing 113
 - lagerplassbegrensninger 112
 - med sammendragstabeller 82
 - prosess 90
 - regler 107
 - sektorer 112
 - spørringstyper 112
 - SQL-skript 115
 - tidsbegrensning 113
 - ytelsestesting 91, 118
- optimaliseringsregler
 - dimensjoner 39
 - kombineringer 41
 - kubemodeller 39
- optimaliseringssektorer 94, 95, 104
 - Alle 95
 - gjennomdrilling 102
 - hybriduttrekking 100
 - MOLAP-uttrekking 99
 - neddrilling 96
 - nivåer 95
 - oppgi 55
 - rapport 97
 - Vilkårlig 95
- optimaliseringssektorer, oppgi 55
- oversikt
 - metadataobjekter 11

R

- Rangerte salgstell
 - DENSERANK 74
 - RANK 74
 - ROWNUMBER 74
- rapportspørringer 94
- recurse-parameter 140
- regler
 - base 107
 - kubemodellens fullstendighet 107
 - optimalisering 107
- rekursiv distribuering 26
- relasjonstabeller 11
- Rename-operasjonen 150

restriction-parameteren 140

S

- sammendragstabeller 82, 90, 114
 - begrensninger 88
 - funksjonelle avhengigheter 88
 - oppdatere umiddelbart 121
 - oppdatering utsatt 120, 121
 - opprette 92
 - slette 93, 122
 - SQL-skript 115
 - vedlikeholde 92, 121
- sektor 83
- sektorer
 - optimalisering 95
- skjemaer
 - snøfnugg 11, 109
 - stjerne 11
- slette metadataobjekter 57
- snøfnuggskjema 14
- snøfnuggskjemaer 11, 109
- sporing 157
 - logging 157
- spørringer
 - DB2EXPLAIN 119
 - gjennomdrilling 102
 - hybriduttrekking 100
 - MOLAP-uttrekking 95, 99
 - neddrilling 94, 96
 - omdirigerte 82, 85, 88
 - rapport 94, 97
 - registrere 119
 - typer 94
 - ytelsestesting 118
- SQL-skript 114, 115
- standardattributter
 - beskrivelse 27
- standarddistribuering 25
- stjerneskjemaer 11
- systemkonfigurasjon 130
- systemkrav 5

T

- Telle 72
- Tids- dimensjon 77
- tjenere
 - forent 123
- transaksjon 130
- Translate-operasjonen 155

U

- ubalanserte hierarkier 23
- ujevne hierarkier 24
- utriggertabeller 109

V

- Validate-operasjonen 154
- Veiviser for optimaliseringsrådgiver 82, 92, 93, 112, 114, 121
- veiviseren for tilføyning av dimensjoner 53

Vinduet Databasetilkobling 46
vinduet Eksporter 47

X

XML-analyse 130

Kontakte IBM

Hvis du har et teknisk problem, bør du se gjennom og utføre handlingene som er foreslått i produktdokumentasjonen før du kontakter kundestøtten for DB2 Cube Views. Denne boken inneholder tips til informasjonsinnsamling som kan gjøre det enklere for kundestøtten for DB2 Cube Views å hjelpe deg.

Hvis du vil ha informasjon eller bestille noen av DB2 Cube Views-produktene, kontakter du en IBM-representant på et lokalt avdelingskontor eller en autorisert IBM-programvareforhandler.

Hvis du er i USA, kan du ringe et av disse numrene:

- 1-800-237-5511 for kundestøtte
- 1-888-426-4343 hvis du vil vite mer om tilleggstjenester

Produktinformasjon

Hvis du er i USA, kan du ringe et av disse numrene:

- 1-800-IBM-CALL (1-800-426-2255) eller 1-800-3IBM-OS2 (1-800-342-6672) for å bestille produkter eller få generell informasjon.
- 1-800-879-2755 for å bestille publikasjoner.

<http://www.ibm.com/software/data/db2/db2md/>

Inneholder linker til informasjon om DB2 Cube Views.

<http://www.ibm.com/software/data/db2/udb>

Nettsidene til DB2 Universal Database inneholder siste nytt, produktbeskrivelser, opplæringsplaner og så videre.

<http://www.elink.ibm.com/>

Klikk på Publications for å åpne nettstedet for bestilling av publikasjoner internasjonalt, som har informasjon om hvordan du bestiller bøker.

<http://www.ibm.com/education/certify/>

Professional Certification Program fra IBM-nettstedet har opplysninger om sertifiseringstesting for en rekke IBM-produkter.

Merk: I noen land bør autoriserte IBM-forhandlere kontakte sin forhandlerkontakt i stedet for IBM Kundeservice.

Kommentarer til dokumentasjonen

Din tilbakemelding er viktig for at vi skal kunne gi informasjon av høy kvalitet. Send oss eventuelle kommentarer du har til denne boken eller til annen DB2 Cube Views-dokumentasjon. Du kan sende kommentarer på disse måtene:

- Send kommentarer med det elektroniske kommentarskjemaet på www.ibm.com/software/data/rcf.
- Send kommentarer via e-post til comments@us.ibm.com. Pass på å ta med navnet på produktet, versjonsnummeret til produktet, tittelen på boken og bestillingsnummeret til boken (der det er aktuelt). Hvis du kommenterer en bestemt tekst, bør du ta med en henvisning til teksten (for eksempel en tittel, et tabellnummer eller sidetallet).



Programmnummer: 5724-E15

SA15-4837-01

