

Tietokannan hallinta - kurssin tavoite

- Kurssilla opitaan periaatteet
 - fyysisen tietokannan tallennuksesta ja käsittelystä
 - tietokantakyselyiden muuntamisesta fyysisen tietokannan käsittelyoperaatioiksi
 - kyselyn kustannusten arvioinnista
 - tietokantatransaktioista ja tietokannan samanaikaisesta käytöstä
 - varmistuksesta ja elvytyksestä
- Edellytyksenä osallistumiselle on Tietokantojen perusteiden hallinta

1

Abstraktiotasot tietoon

KÄSITETASO: Sisältö ja merkitys, ER-malli, oliomallit

RAKENNETASO: Looginen rakenne, relaatiomalli, oliomallit

FYYSINEN TASO: Tekninen rakenne

2

Abstraktiotasot tietoon

KÄSITETASO

RAKENNETASO

FYYSINEN TASO

Tietokannanhallintajärjestelmän hallinnassa

3

Abstraktiotasot tietoon

Tietokannan fyysinen taso voidaan edelleen jakaa useisiin abstraktiotasoihin

KÄSITETASO

RAKENNETASO

FYYSINEN TASO

Tietokannanhallintajärjestelmän hallinnassa

tiedostorakennetaso

puskuritaso

levynhallintataso

4

Abstraktiotasot tietoon

- Tietokantojen perusteet kurssilla käsiteltiin relaatiotietokantoja rakennetasolla
 - tietokanta muodostuu **tauluista** (table), joita käsitellään SQL:n kysely- (select) ja muokkausoperaatioilla (insert, delete, update)
- Siirryttäessä fyysiselle tasolle täytyy rakennetason operaatiot muuttaa fyysisen tason rakenteita hyväksikäyttäviksi operaatioiksi.

5

Tietokannan hallintajärjestelmän arkkitehtuurikomponentteja

jäsentäjä

kustannuslaskenta

kyselyn käsitteijä

optimoija

suorittaja

pääsynvalvonta

transaktioiden hallinta

tiedostorakenteet


puskurien hallinta

levyjen hallinta

lukkojen hallinta

rinnakkaisuuden hallinta


toimimishallinta



Tietokannan hallintajärjestelmän arkkitehtuurikomponentteja

- jäsentäjä (parser) purkaa SQL kyselyn rakennetason käsittelyoperaatioiksi – (vrt. relaatioalgebra)
- kustannuslaskenta (operation evaluation) – tuottaa arvioita operaatioiden kustannuksista
- optimoija (optimizer) laatii kyselylle toteutus suunnitelman (plan)
- suorittaja (plan executor) toteuttaa kyselyn suunnitelman mukaisesti
- pääsynvalvonta (authorization) varmistaa käyttäjän oikeudet operaatioihin


7



Tietokannan hallintajärjestelmän arkkitehtuurikomponentteja

- transaktioiden hallinta (transaction manager) ja lukkojen hallinta (lock manager) huolehtivat tietokannan rinnakkaiseen käyttöön ja peruutuksiin liittyvistä ongelmista
- toipumisen hallinta (recovery manager) huolehtii häiriötilanteisiin varautumisesta, varmistuksista ja elvytyksestä


8



Data - metadata

- Tietokantaan tallennettu tieto jakautuu
 - varsinaiseen dataan
 - metadataan = varsinaista dataa ja sen käsittelyä kuvaavaa dataa
 - kaikki tietokantaan liittyvä kuvaustieto tallennetaan erityiseen metatietokantaan – tietohakemistoon (data dictionary, catalog)
 - relaatiotietokantojen tapauksessa metatietokanta on myös rakenteeltaan relaatiotietokanta ja siihen voi kohdistaa kyselyjä kuten tavalliseen tietokantaan.


9



Metadata

- Relaatiotietokannan tietohakemiston rakenne on määritelty tietohakemistonäkyminä SQL-92 standardissa. Kuitenkin esimerkiksi Oracle käyttää yhä omia taulurakenteitaan.
- Oracle-tietohakemistossa
 - 'user_' -alkuiset taulut kuvaavat lähinnä käyttäjän omistamia rakenteita, esimerkiksi
 - user_tables kuvaa käyttäjän taulut (lyhenne tabs)
 - user_tab_columns kuvaa käyttäjän taulujen sarakkeet
 - user_views kuvaa käyttäjän määrittelemät näkymät
 - 'all_' -alkuiset taulut kuvaavat kohteita, joihin käyttäjällä on pääsy
 - 'dba_' -alkuiset taulut sisältävät tiedot kaikista tietokannan kohteista. Näihin tauluihin on pääsy vain tietokannan hoitajilla (DBA, database administrator)


10



Metadatan käytöstä

- Käyttäjille tarkoitettujen taulujen lisäksi on tietokannan dynaamista käyttäytymistä kuvaavia tauluja (Oraclessa V\$-alkuisia) – esim v\$_open_cursors, joka kertoo käynnissä olevat kyselyt, käytön seurantaan liittyviä lokitauluja, jne. Useimpiin näistä on pääsy vain tietokannan hoitajalla ja TKHJ:n sisäisillä rutiineilla.

11




Esimerkki metadatatista

```
SQL> desc ALL_TABLES
```

Name	Null?	Type
OWNER	NOT NULL	VARCHAR2(30)
TABLE_NAME	NOT NULL	VARCHAR2(30)
// tilan hallintaan liittyvää tietoa		
TABLESPACE_NAME		VARCHAR2(30)
CLUSTER_NAME		VARCHAR2(30)
IOT_NAME		VARCHAR2(30)
PCT_FREE		NUMBER
PCT_USED		NUMBER
// tapahtumien hallintatietoa		
INI_TRANS		NUMBER
MAX_TRANS		NUMBER


12


Esimerkki metadatatista

```

// jälleen tilanhallintaa
INITIAL_EXTENT          NUMBER
NEXT_EXTENT            NUMBER
MIN_EXTENTS            NUMBER
MAX_EXTENTS            NUMBER
PCT_INCREASE           NUMBER
FREELISTS              NUMBER
FREELIST_GROUPS        NUMBER
// elvytystietoa
LOGGING                VARCHAR2(3)
BACKED_UP              VARCHAR2(1)
    
```


13


Esimerkki metadatatista

```

//kustannuslaskentatietoa
NUM_ROWS                NUMBER
BLOCKS                  NUMBER
EMPTY_BLOCKS            NUMBER
AVG_SPACE               NUMBER
CHAIN_CNT               NUMBER
AVG_ROW_LEN             NUMBER
AVG_SPACE_FREELIST_BLOCKS NUMBER
NUM_FREELIST_BLOCKS    NUMBER
DEGREE                  VARCHAR2(10)
INSTANCES               VARCHAR2(10)
CACHE                   VARCHAR2(5)
TABLE_LOCK              VARCHAR2(8)
SAMPLE_SIZE             NUMBER
LAST_ANALYZED           DATE
PARTITIONED             VARCHAR2(3)
IOT_TYPE                VARCHAR2(12)
TEMPORARY               VARCHAR2(1)
NESTED                  VARCHAR2(3)
BUFFER_POOL             VARCHAR2(7)
    
```


14


Esimerkki metadatatista

```

SQL> desc all_tab_columns
Name          Null?    Type
-----
OWNER         NOT NULL VARCHAR2(30)
TABLE_NAME   NOT NULL VARCHAR2(30)
COLUMN_NAME  NOT NULL VARCHAR2(30)
DATA_TYPE    VARCHAR2(30)
DATA_TYPE_MOD          VARCHAR2(3)
DATA_TYPE_OWNER        VARCHAR2(30)
DATA_LENGTH   NOT NULL NUMBER
DATA_PRECISION          NUMBER
DATA_SCALE    NUMBER
NULLABLE      VARCHAR2(1)
COLUMN_ID     NOT NULL NUMBER
    
```


15


Esimerkki metadatatista

```

DEFAULT_LENGTH          NUMBER
DATA_DEFAULT            LONG
// kustannuslaskentatietoa
NUM_DISTINCT           NUMBER
LOW_VALUE               RAW(32)
HIGH_VALUE              RAW(32)
DENSITY                 NUMBER
NUM_NULLS               NUMBER
NUM_BUCKETS             NUMBER
LAST_ANALYZED           DATE
SAMPLE_SIZE             NUMBER
CHARACTER_SET_NAME      VARCHAR2(44)
CHAR_COL_DECL_LENGTH    NUMBER
    
```

16


Metadatan käytöstä

- Metadattaa tarvitaan monessa eri yhteydessä
 - jäsentäjä tarvitsee tietoja taulujen ja sarakkeiden nimistä ja tietotyypeistä
 - kustannuslaskenta tarvitsee tietoja esim. taulun rivien lukumäärästä, sarakkeiden arvojen jakautumisesta, taulun sijoittelusta tiedostoihin, hakemistojen olemassaolosta, jne.
 - pääsynvalvonta tarvitsee tietoja käyttäjistä ja heidän oikeuksistaan
 - transaktion hallinta tarvitsee dynaamista tietoa istunnoista ja operaatioista
 - lukkojen hallinta tarvitsee tietoja tapahtumien suorittamista operaatioista, samoin toipumisen hallinta

17