Tietokantakurssit / TKTL

Tietokantojen perusteet
- tietokannan käyttö: SQL, sovellukset

Tietokannan hallinta
- tietokannanhallintajärjestelmän ominaisuuksia:
  tallennusrakenteet
  kyselyjen toteutus
  tapahtumien (transaktioiden) hallinta

Laudatur:
Tietokannan mallinnus
- tietokannan suunnittelu, käsittelyä (kalkyylit)

Tietokantarakenteet ja –algoritmit
-syventää transaktioiden hallintaa ym.
Erikoiskursseja: tietovarastot, paikkatietojärjestelmät

1. Johdanto

Tietokanta (database) = loogisesti yhtenäinen kokoelma toisiinsa liittyviä tietoja

- fyysisellä tasolla: useita tiedostoja (tauluja)
  tietojen määrä voi olla suuri
- tietyssä mielessä pysyvä (ohjelmaan verrattuna):
  monen sovelluksen käytössä, jopa samanaikaisesti
- sovelluksista erillinen: suunnitellaan erikseen
  ohjelmien ja tietojen riippumattomuus
- tietojen väliset loogiset yhteydet, rajoitteet;
  rajoitteiden voimassaolon valvonta
- tieto tallessa vain kertaalleen (yleensä)

Tietokannan ominaisuuksia, (jatkoa)

- monipuolinen tiedonhaku (kyselyllä) normaalia
  ja helppoa (’korkean tason ohjelmointia’)
- tietojen käytön säätely:
  kontrolloitu samanaikainen käyttö, suojaus
- eri käyttäjiä varten erilaisia näkemyksiä (view)
- automaattinen varmistus ja elpyminen häiriöistä
- tietokanta(järjestelmä) sisältää datan lisäksi
  - kuvaustietoa (metadataa, tietohakemisto)
  - hallintatietoa: hakemistoja, lukitustietoja ym.

Esim. yrityksen henkilöstötietokanta

EMPLOYEE(ssn, name, bdate, address, salary, superssn, dno)
DEPARTMENT(dname, dnumber, mgrssn)
PROJECT(pname, pnumber, plocation, dnum)
WORKS_ON(essn, pno, hours)
DEPT_LOCATIONS(dnumber, dlocation)
DEPENDENT(essn, dep_name, relationship)

(vrt. E&N, s. 204-205)

Tietokannan määrittely on tietokannan **kaavio** (schema): ilmaisee, kuinka tietokannan kohteen (“miniworld”) tiedot voidaan jäsennellä.

relaatiot: attribuutit (ominaisuudet)
relaatioiden väliset suhteet (esim. viite-eheys)

- erilaisia kaavioita: tietokannan mallinnus
  loogisesti hyvä/huono tietokanta ...
  (toisen kurssin asiaa)

Kertaa Tietokantojen perusteet –kurssin alkuosasta:
- ***johdanto***
- ***relaatio (avain, viiteavain: 4.1)***
- ***SQL (luku 5; HETI vähintään perusteet)***

Henkilöstötietokannan määrittely SQL:llä

```
create table    employee
 (ssn           char(9)  not null,
  name          varchar(30)      not null,
  bdate         date,
  address       varchar(30),
  salary        decimal(10,2),
  superssn      char(9),
  dno           int              not null,
  primary key (ssn),
  foreign key (superssn) references employee(ssn),
  foreign key (dno) references department(dnumber) );
```


Henkilöstötietokannan määrittely SQL:llä - 2

```
create table    department
 (dname         varchar(15)      not null,
  dnumber       int              not null,
  mgrssn        char(9)  not null,
  primary key (dnumber),
  unique (dname),
  foreign key (mgrssn) references employee(ssn) );

create table    project
 (pname         varchar(15)      not null,
  pnumber       int              not null,
  plocation     varchar(15),
  dnum          int              not null,
  primary key (pnumber),   unique (pname),
  foreign key (dnum) references department(dnumber) );
```


Henkilöstötietokannan määrittely SQL:llä - 3

```
create table    works_on
 (essn          char(9)          not null,
  pno           int              not null,
  hours         decimal(3,1),
  primary key (essn, pno),
  foreign key (essn) references employee(ssn),
  foreign key (pno) references project(pnumber) );

create table    dept_locations
 (dnumber       int              not null,
  dlocation     varchar(15)      not null,
  primary key (dnumber, dlocation),
  foreign key (dnumber) references department(dnumber) );
```


Henkilöstötietokannan määrittely SQL:llä - 4

```
create table    dependent
 (essn          char(9)          not null,
  dep_name      varchar(15)      not null,
  relationship  varchar(8),
  primary key (essn, dep_name),
  foreign key (essn) references employee(ssn) );
```

Hieman laajempana: E&N, s. 204-205
- myös tietokannan sisältö (”tietokannan tila”)

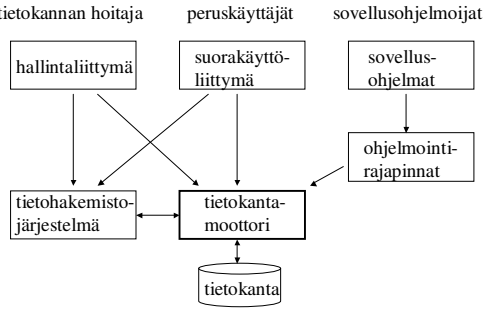
Tietokannanhallintajärjestelmä

Tietokannanhallintajärjestelmä (tkhj, dbms) =
ohjelmisto, jonka avulla luodaan, ylöäpäidetään ja käytetään tietokantoja.

Tietokannanhallintajärjestelmän osat:

- tietohakemistojärjestelmä
- tietokantamoottori
- kääntäjiä /liittymiä:
  - tiedonmäärittelykielelle
  - kyselyille
  - sovellusohjelmien käyttämille tietokantakielille (ohjelmointirajapinnoille)

Tietokannanhallintajärjestelmä - 2



Tietokannanhallintajärjestelmä - 3

Tietokantamoottori
- pääsyn valvonta (saantioikeudet)
- kyselyn optimoija
- tapahtuman hallinta (samanaikaiset käyttäjät, elvytys)
- eheyden valvonta (tietokannan osien yhteydet, oikeellisuus)
- puskurien hallinta ja saantimenetelmät
  (yhteys keskusmuistiin: siirto-operaatiot)
- ajonaikainen tk-prosessori toteuttaa kyselyt
  ja hoitaa toimintaan liittyvän kontrollin

Tietokantajärjestelmän arkkitehtuuri

Perinteinen kolmitasoarkkitehtuuri:

ulkoinen taso: käyttäjänäkymät, sovellusohjelmat

käsitetaso: tietokannan looginen kaavio
(yleinen rakenne, ”kaikille”)

sisäinen taso: tietokannan säilytyskaavio
(tiedostorakenteet, saantipolut)

Käyttöjärjestelmän kannalta eri vaihtoehtoja:
- henkilökohtainen järjestelmä
- keskuskonemalli
- tiedostopalvelinmalli
- asiakas-palvelin-malli
- monitasomalli (asiakas, välitaso, palvelin)
- hajautetut järjestelmät

Tietohakemistojärjestelmä

Tietohakemisto sisältää tietokannan kuvaustiedot (metatietoa)
= tietokantakaavio kaikkine yksityiskohtineen

- kuvaustiedot yleensä relaatioina, voidaan käyttää varsinaisten tietojen tavoin

Esimerkkejä Oraclen kaaviotauluista:

USER_CATALOG käyttäjän omat objektit (taulut, näkymät ym.)
USER_VIEWS käyttäjän näkymät
USER_TAB_COLUMNS käyttäjän taulujen ja näkymien sarakkeet
ALL_CATALOG kaikki objektit, joihin käyttäjällä on pääsy
ALL_TABLES, ALL_VIEWS jne
DBA_ALL_TABLES, DBA_EXTENTS jne tietokannanhoitajan tauluja

Tietohakemistojärjestelmä - 2

Esimerkkejä.

```
select * from user_catalog;
TABLE_NAME                    TABLE_TYPE
-------------------------------------------
DEPARTMENT                    TABLE
EMPLOYEE                      TABLE
  . . .
select column_name, data_type
from user_tab_columns
where table_name = 'EMPLOYEE;
```


Tietohakemistojärjestelmä - 3

```
COLUMN_NAME                DATA_TYPE
-------------------------------------
SSN                        CHAR
NAME                       VARCHAR2
BDATE                      DATE
ADDRESS                    VARCHAR2
SALARY                     NUMBER
SUPERSSN                   CHAR
DNO                        NUMBER
desc employee;
NAME          NULL?        TYPE
-------------------------------------
SSN           NOT NULL     CHAR(9)
NAME          NOT NULL     VARCHAR2(30)
BDATE                      DATE     jne
```


Tietohakemistojärjestelmä - 4

```
select * from user_views where view_name = 'ADM_EMPLOYEE_SAL';

VIEW_NAME              TEXT_LENGTH   TEXT
------------------------------------------------------------------
ADM_EMPLOYEE_SAL       102           select lname, salary
                                     from employee, department
                                     where dno = dnumber
                                     and dname = 'Administration'
SQL> select * from adm_employee_sal;
NAME                    SAL
--------------- ----------
Zelaya                25000
Wallace               43000
Jabbar                25000
```