

# Dimensiot ennen faktoja: karteesinen tulo

Marja Kärmeniemi,  
Oracle Finland Oy  
Business Intelligence and  
Warehouse Consulting

Turhien rivien lukeminen suuresta faktataulusta vie pitkän ajan, kuten Tapio Lahdenmäki artikkelissaan kertoo. Tähti-liitoksen tulisi siis ensin etsiä dimensiotaulukujen avulla kiinnostavien faktarivien avaimet ja vasta sen jälkeen lukea faktataulua. Jos optimoija ei osaa käsitellä ensin dimensioita ja sitten faktataulua, pitää kyseinen käsittely tehdä pahimmassa tapauksessa itse. Myynti-raporttiesimerkissämme algoritmi on seuraavan kaltainen (kuva: Dimensioiden karteesisen tulon muodostaminen sql-kielellä):

1. Lue tauluun niiden tuotteiden TUOTE\_ID, jotka kuuluvat kiinnostavaan tuoteryhmään.
2. Lue toiseen tauluun niiden päivämäärien AIKA\_ID, jotka osuvat raportoitaville kuukausille.
3. Yhdistä nämä taulut ilman liitoseh-

toa karteesiseksi tuloksi. Karteesisella tulolla tarkoitetaan taulua, jossa liitettävien taulujen kaikki rivit on yhdistetty toisten taulujen kaikkien rivien kanssa.

4. Liitä tämä tulotaulu faktatauluun hyödyntäen faktataulun indeksiiä (AIKA\_ID, TUOTE\_ID) ja tee samalla liitos liiketauluun liikkeen nimen hakemiseksi.

Vaihtoehtona olisi lukea läpi faktataulua ja käydä jokaisen rivin kohdalla katsomassa kohdissa 1 ja 2 muodostetuista välitauluista tai suoraan dimensiotauluista, täyttyvätkö hakuehdot.

Faktataulun kiinnostavat rivit etsitään siis muodostamalla ensin dimensioavaimista karteesinen tulo, jota verrataan fakta-aulun indeksiin, jonka kyseiset avaimet muodostavat. Karteesisen tulotaulun rivien lukumäärä on kunkin dimension kelpaavien avainten lukumäärät kerrottuna keskenään eli rivien lukumäärä voi muodostua todella suureksi. Tulotaulun rivinpituus on dimensioavainten pituuksien summa. Tulotaulussa

on rivi kutakin mahdollista avainyhdistelmää kohti ja tämä välitaulu yhdistetään faktatauluun nested loop liitoksella, jossa käytetään hyväksi avaimista muodostettua b-puuindeksiä.

## Milloin karteesinen tulo on hyvä saantipolku?

Karteesisen tulon muodostaminen on hyvä saantipolku silloin, kun dimensioiden karteesinen tulo on pieni eli välitaulusta tulee kooltaan pieni. Suuren välitaulun tallentaminen vaatii paljon muistia ja sen läpilukuun kuluu pitkä aika.

Tulotaulusta on hyötyä vain tilanteissa, joissa faktatauluun on olemassa sopiva b-puuindeksi, josta löytyvät tulotaulussa olevat avainsarakkeet.

## Myyntituottokyselyn karteesinen tulo

Esimerkkimme (kuvat: Myyntituottokysely, Myyntifaktat ja dimensiot, s.10) dimensioiden karteesinen tulo on verrattain pieni:

Aikadimensiosta löytyy kiinnostaville kuukausille 50 mahdollista päivämääräavainta. Kiinnostavaan tuoteryhmään kuuluu 50 tuotetta. Avainten pituudeksi oletetaan 4 B.

Tulotaulun koko on  $50 * 50 * (4 + 4)$  B = 2500 riviä \* 8B = 20 MB. Voimme olettaa sen mahtuvan puskurin käsittelyn ajaksi.

Myyntitaulun indeksiiä (AIKA\_ID, TUOTE\_ID, LIIKE\_ID) luetaan hajalukuna 2500 kertaa. Yhden päivän ja tuotteen avaimet ovat todennäköisesti yhdellä indeksin lehtisivulla. Koska aikaan ja tuotteen perustuvan hakuehdon läpäisykerroin on  $(50/1\ 000) * (50/10\ 000)$  eli 25/100 000 niin vastaava osuus indeksistä sisältää mahdollisesti kiinnostavia avaimia. 25 GB:n indeksistä tämä osuus on 6,25 MB. Avainten voidaan olet-

```
1. CREATE TABLE APU_TUOTEID
  (TUOTE_ID INT)
INSERT INTO APU_TUOTEID
  SELECT TUOTE_ID
  FROM TUOTE
  WHERE TUOTERYHMÄ = :TUOTERYHMÄ
```

| TUOTE_ID |
|----------|
| 3        |
| 8        |

```
2. CREATE TABLE APU_AIKAID
  (AIKA_ID INT)
INSERT INTO APU_AIKAID
  SELECT AIKA_ID
  FROM AIKA
  WHERE KUUKAUSI IN (:X,:Y)
```

| AIKA_ID |
|---------|
| 21      |
| 53      |

```
3. CREATE TABLE APU_AVAIN
  (TUOTE_ID INT, AIKA_ID INT)
INSERT INTO APU_AVAIN
  SELECT TUOTE_ID, AIKA_ID
  FROM APU_TUOTEID, APU_AIKAID
```

| TUOTE_ID | AIKA_ID |
|----------|---------|
| 3        | 21      |
| 8        | 53      |
| 3        | 53      |
| 8        | 21      |

```
4. SELECT ...
  FROM MYynti M, APU_AVAIN A, LIIKE L
  WHERE M.LIIKE_ID = L.LIIKE_ID
  AND M.TUOTE_ID = A.TUOTE_ID
  AND M.AIKA_ID = A.AIKA_ID
```

Kuva: Dimensioiden karteesisen tulon muodostaminen sql-kielellä

taa jakautuvan tasaisesti, jolloin yhden päivän ja tuotteen avaimet mahtuvat 2,5 KB (6,25 MB/2500) tilaan ja siis yhdelle sivulla. (kuva: Myyntitaulun indeksi kuvattuna tauluna)

Indeksin hajalukuun kuluu  $2500 * 20$  ms = 50 s

### Oracle Star Plan ja DB2 Cartesian product

Oracle Star Plan ja DB2 Cartesian Product saantipolut perustuvat karteeseen tuloon. Tähtiliitoksessa dimensiotaulut liitetään nested loop liitoksella faktatauluun hyödyntäen viiteavaimia. Faktatauluun luodaan indeksi, jossa on dimensiotaulujen avaimet. Saantipolku toimii hyvin tilanteissa, joissa dimensioavainten karteeseen tulo on selvästi pienempi kuin tietokannan sivuille muistista varattu puskurialue (Buffer Cache, Buffer Pool). Myyntiraportointiesimerkissämme puskurin tyypillinen koko olisi 1 GB luokkaa.

|   | (PVM-kkvv) | TUOTE_ID | LIIKE_ID | rowid/RID |
|---|------------|----------|----------|-----------|
| ↓ | 0298       | 3        | 111      |           |
|   | 0298       | 3        | 122      |           |
| ↓ | 0298       | 4        | 111      |           |
|   | 0298       | 8        | 133      |           |
| ↓ | 0298       | 9        | 111      |           |
|   | 0398       | 7        | 133      |           |
| ↓ | 0498       | 3        | 122      |           |
|   | ...        | ...      | ...      |           |
| ↓ | 0299       | 1        | 133      |           |
|   | 0299       | 8        | 111      |           |
| ↓ | 0299       | 8        | 133      |           |
|   |            |          |          |           |

AIKA\_ID on korvattu kuvassa kuukaudella ja vuodella, johon juoksevasti numeroitu AIKA\_ID viittaa.

Kuva: Myyntitaulun indeksi kuvattuna tauluna.

Marja Kärmeniemi,  
Oracle Finland Oy

Business Intelligence and Warehouse  
Consulting

# Data Warehousing

**Koulutus, konsultointi ja toteutustyö - vuosien kokemuksella.**

Kurssejamme mm.

|                                     |          |
|-------------------------------------|----------|
| Data Warehouse tietoisku            | 0,5 pvaa |
| Data Warehouse -hankkeen läpivienti | 2 pvaa   |
| Data Warehouse -kannan suunnittelu  | 2 pvaa   |
| Data Warehouse -projektipäällikkö   | 1 pva    |

**Lisätietoja: [www.vidac.fi](http://www.vidac.fi)**

- Tutustu myös muihin kursseihimme, mm.:  
Systeeminsuunnittelupäivät,  
Relaatiokannan suunnittelu, SQL, ylläpito
- Kirja  
Ari Hovi : "Data Warehousing - tietovarastotekniikka",  
Suomen atk-kustannus

**Vidac Oy**  
Visual Database Access

**ARI HOVI OY**