

# Tähti ja lumihiutale

Jaakko Rantanen,  
Hämeen ammattikorkeakoulu,  
Tietojenkäsittelyn koulutusohjelma

Mikä on moniulotteisen tietokannan tähtikaavio (star schema) ja millä menettelyllä sellainen suunnitellaan. Mikä on lumihiutalekaavio (snowflake schema) ero tähtikaavioon. Tässä esitetään tiivistelmä tähtikaavio suunnittelusta. Oppimestarina on Ralph Kimball.

## Keittokirja

Ralph Kimball on kirjoittanut tietovarastokannan suunnittelusta alan bestselleriksi muodostuneen kirjan Data Warehouse Toolkit (John Wiley & Sons 1996). Kyseessä on oikeastaan keittokirjan tapainen reseptikokoelma ja niksipaketti. Kimball esittelee useita tyyppi-esimerkkejä, esimerkiksi päivittäistavara-kaupan, varastonvalvonnan, tavarantoimitusten, pankkipalveluiden, puhelintilausten käsittelyn, vakuutusmyynnin, lentolaivamatkojen myynnin ja muutamien muun aihepiirien mukaisen tietovarastokannan suunnittelua.

Esimerkit ovat pelkistettyjä siten,

että suunnittelupäätösten kohteista on karsittu turha detajiiikka. Esitystapa etenee siten, että ensin kerrotaan esimerkkiaiheen tunnuspiirteistä, sitten tehdään yhteenve- to suunnittelun päävaiheista. Kun- kin vaiheen keskeiset kohdat esitel- lään tarkemmin ja näiden yhteydes- sä tarkastellaan jotakin tyypillistä yksityiskohtaa. Kunkin esimerkin lopuksi tehdään volyymilaskelmat, jotka osoittavat tietomäärän suuruusluokan arvointiperiaatteet.

Kimball suosittelee lukijaa pe- rehtymään kirjansa kaikkiin esi- merkkeihin, myös itselle vieraiden toimialojen tapauksiin. Suositusta voi pitää aiheellisena, koska kukin esimerkki ovat pelkistetty, tilanne yleistajuisesti kerrottu ja samantapai- sia tilanteita esiintyy muuallakin.

## Alkuresepti

Tähtikaavio mukaisen tietokannan suunnittelu alkaa neljällä päävaiheella, joista seuraavassa esi- tetään suppea esimerkki. Esimerkin jälkeen kerrotaan yhteenve- to viides- tä loppuvaiheesta. Vaiheet on syytä suorittaa annetussa järjestyksessä.

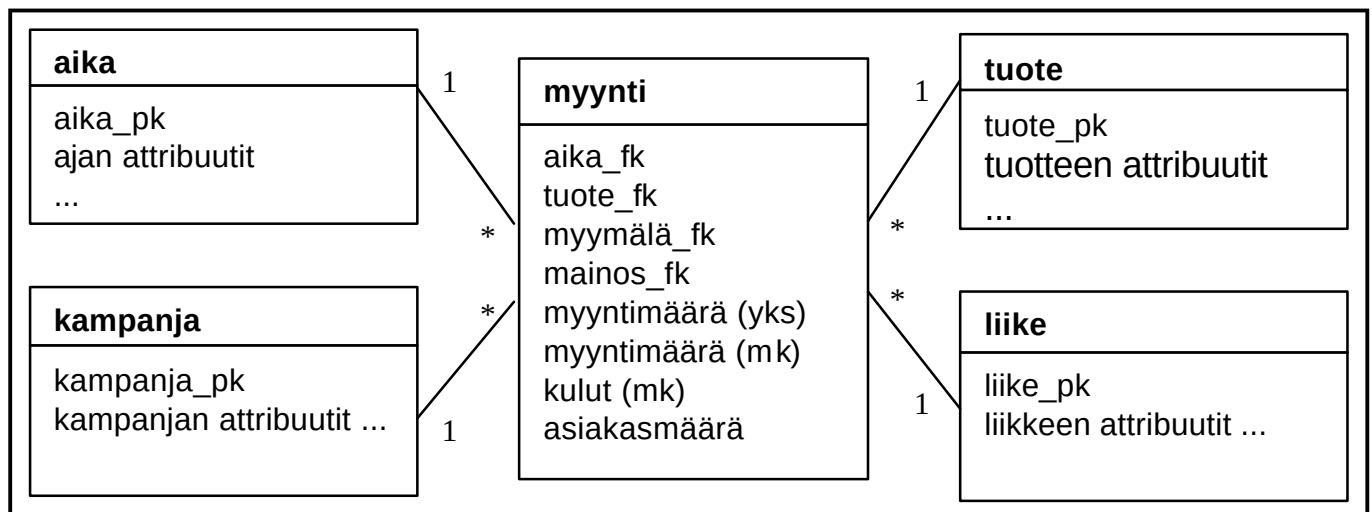
1. Määritetään mallin kohteena oleva liiketoimintaprosessi ja

perusjärjestelmä tai järjestel- mät, joista tietovaraston data saadaan kerättyä. Esimerkkejä prosesseista ovat mm. tilausten käsittely, varaston hoito tai kir- janpito.

2. Valitaan liiketoimintaprosessin tietojen esittämisen hienojakoi- suutta tai karkeisuutta kuvaava perusyksikkö, jyvä (*grain*). Pe- rusyksiköllä tarkoitetaan tiedon pienimmän elementin esitys- tarkkuutta, joka faktataulussa esitetään. Perusyksikkö voi olla yksittäinen tapahtuma, yhden päivän ote tai yhden kuukau- den ote. Seuraavaan vaiheeseen voi edetä vasta kun perusyksik- kö on määrätty.

3. Valitaan dimensiot (tekijät eli ulottuvuudet), jotka ovat liike- toiminnan käsitteitä (esim. aika tai asiakas), joihin perustuen tiedoista voidaan haluta yhteen- vetoja. Jokainen dimensio esi- tetään omana taulunaan (*di- mension table*). Määritetään dimensiotaulukujen sarakkeet.

4. Valitaan raportoitavat suureet (*measured facts*) eli faktojen mittasuureet, jotka sijoitetaan faktatauluun. Tavallisia mitta- suureita ovat esimerkiksi raha- tai kappale- ja muut yksikkö-



Kuva 1. Moniulotteisen tietokannan tähtikaavio periaate.

määrät. Mittasuureiden lisäksi faktatauluun tulee vain jokaiseen dimensiotauluun viittaavat avaimet.

Tietokannan kaavio muistuttaa tähtä: keskellä on faktataulu ja tähden sakaroina ovat dimensiotaulut. Kuvasa 1 on esimerkki moniulotteisen tietokannan periaatteesta.

### Esimerkki

Tässä käsiteltävän esimerkin aihepiiri lienee useimmille ainakin asiakkaan näkökulmasta tuttu, nimittäin vähittäiskauppaan myyntikampanja. Tarkasteltavana liiketoimintaprosessina on kampanjamyynä. Tietojen esittämisen perusyksiköksi eli jyväksi valitaan yhden kampanjatuotteen vähittäis-pakkausten myynnin summa yhden päivän aikana yhdessä liikkeessä. Dimensiot eli tekijät ovat Aika, Tuote, Liike ja Kampanja. Faktataulun nimi on Myynti.

Tietokannan on tarkoitus palvella raportointia. Faktataulujen mittasuureista lasketaan yhteenvetoja niiden dimensioiden perusteella joita raportin tarvitsija haluaa. Kuva 2 esittää osaa yhdestä yksinkertaisesta raportista, johon liitetään kahden eri dimensiotaulun tietoja.

Tarkastellaan lähemmin kahta dimensiotaulua: Aika ja Tuote.

Aika-taulu esiintyy tähtikaaviossa tavallisesti yhtenä dimensiona, mitä voi ihmetellä. Sisältäähän esimerkiksi SQL tietotyyppit, joilla päiväys ja

kellonaika ovat melko monipuolisesti käsiteltävissä. Mihin aika-taulua siis tarvitaan?

Tietojen analysoinnissa voidaan olla kiinnostuneita sellaisista asioista kuten esimerkiksi onko kyseessä vaikkapa kuukauden viimeinen päivä, juhlapäivä tai juhlapäivän aatto. Jos tällaisia tärkeitä erityispiirteitä esitetään, voi samalla esittää vaikka viikonpäivän tai päivän numeron vuoden alusta. Nämä olisivat toki myös laskettavissa, mutta miksi tehdä asia niin hankalasti.

Esimerkkimme aika-taulussa on yksi rivi kutakin päivää kohti. Jos kannassa on kolmen vuoden myyntitiedot, on aika-taulussa noin tuhat ri-

| aika                  |
|-----------------------|
| aika_pk               |
| viikonpv              |
| kkpv_no               |
| kuukausi              |
| kk_nro                |
| vuosipv_nro           |
| vuosivko_nro          |
| pyhä_lippu            |
| tyopv_lippu           |
| kk_viimeinen_pv_lippu |
| sesonki               |
| ...                   |

Kuva 3. Ajan attribuutteja.

| tuote               |
|---------------------|
| tuote_pk            |
| tuoteyksikön_kuvaus |
| tuote_no            |
| tuoteryhmä          |
| pakkaustyyppi       |
| pakkauskoko         |
| painoyksikkö        |
| paino               |
| hyllytilan_leveys   |
| ...                 |

Kuva 4. Tuotteen attribuutteja.

viä.

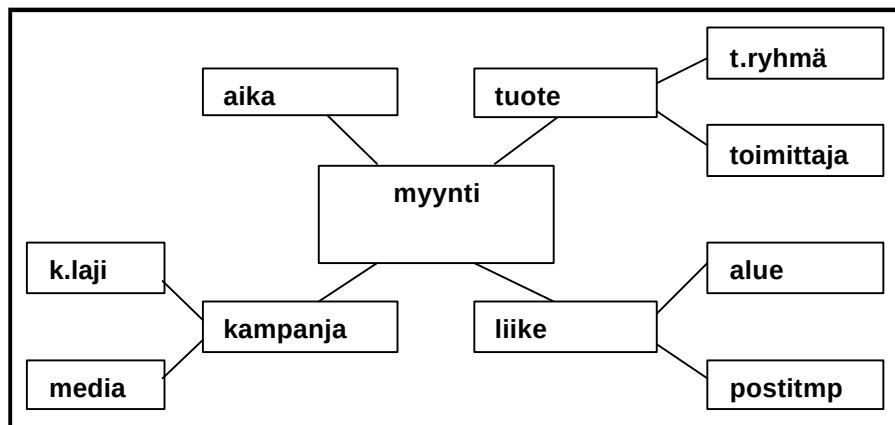
Yksi dimensiotauluista on Tuote.

Tuotekohtaisten tietojen lisäksi siinä esitetään attribuutteja, jotka selvästi ovat esimerkiksi tuoteryhmän eivätkä yksittäisen tuotteen tietoja. Nämä toistuvat, jos saman ryhmän tuotteita esiintyy useita. Tällaisessa tilanteessa on tietokannoissa totuttu normalisoimaan toisteisuuden haitat pois. Haittoinahan koituu hankalia päivitystilanteita: tuoteryhmän viimeisen tuotteen poistossa katoaa tieto tuoteryhmän olemassaolosta, jokaisen tuotteen lisäyksessä on syötettävä ryhmän tiedot jne. Sama normalisoinnin peruste ei päde tietovarastokannassa, koska tiedot tarkistetaan ja siirretään muualta eikä päivityksestä edes koitu niitä ongelmia, joita operatiivisessa kannassa on tärkeitä välttää.

Jos dimensiotauluja normalisoidaan, muuttuu tietokannan

| viikonpv | tuoteryhmä | sum(myynti)  | sum(kulut) | sum(asiakasmäärä) |
|----------|------------|--------------|------------|-------------------|
| ma       | juomat     | 1 198 498,00 | 539 387,00 | 41 309,00         |
| ti       | juomat     | 2 202 874,00 | 410 434,00 | 43 431,00         |
| ke       | juomat     | 1 780 989,00 | 370 213,00 | 39 879,00         |

Kuva 2. Ote yhdestä raportista.



Kuva 5. Lumihiutalekaavion rakenteen periaate.

kaavion muoto tähdestä lumihiutaleeksi. Kuva 5 esittää lumihiutalekaavion periaatetta. Kimball suosittelee vahvasti yksinkertaisemman tähtimuodon säilyttämistä. Kimball varoittaa lumihiutalekaavion aiheuttamasta käsittelyn monipuolisuuden kärsimisestä dynaamisten raporttien mutkistumisesta. Tässä tapauksessa denormalisointi tuottaa yleispätevämmän ratkaisun kuin normalisointi - siis täsmälleen päinvastoin kuin operatiivisessa tietokannassa.

Faktatauluna toimii Myynti. Mitallisina suureina esiintyy raha-, tuoteyksikkö- ja asiakasmääriä. Rahamäärät kuvaavat sekä tuottoja että erilaisia kuluja.

Myynti-taulun mittaluvuista kulut ja raha- sekä yksikkömäärät ovat täydellisesti summautuvia, mutta asiakasmäärä vain osittain. Esimerkiksi kahden tuotteen, vaikka X-maidon ja Y-keksin tapauksessa voi olla niin, että 20 asiakasta on ostanut X-maitoa ja 5 asiakasta Y-keksiä. Asiakasmäärä ei ole täydellisesti summautuva. Jos nimittäin halutaan tietää kuinka moni osti vain jompaa kumpaa muttei molempia, voidaan päätellä ainoastaan, että näitä asiakkaita oli vähintään 15 ja enintään 25.

Mitalliset suureet olisi syytä suunnitella täydellisesti summautuviksi. Muutoin malliin sisältyy virhepäätelmien riskejä. Jos osittain summautuvia suureita (*semiadditive measures*) esiintyy, on käyttäjien oltava niiden käyttäytymisestä selvillä.

Arvioidaan esimerkin taulujen tilantarve. Oletetaan tarkastelujaksoksi kahden vuoden aika, siis noin 600 päivää. Liikkeitä oletetaan olevan 500, joiden tiedot ladataan kantaa päivittäin. Tuotteita oletetaan olevan 12000, joista päivittäin myytyjä on keskimäärin 1500. Kampanjoita on tuotetta, liikettä ja päivää kohti enintään yksi. Faktataulussa on siis  $600 \times 500 \times 1500 \times 1 = 450.000.000$  riviä.

Faktataulussa on 4 viiteavainta á 4B mutta mittalukuja on käytännössä useampia kuin oheisessa kaaviossa on esitetty. Jos faktataulussa on 10 mittalukua (á 4B), rivinpituus on  $14 \times 4B = 56B$  ja tilantarve  $450M \times 56B = 25,2 \text{ GB}$

Dimensiotaulujen tilantarve on faktataulun rinnalla hyvin pieni, joten kokonaistilan tarpeen suuruusluokka määräytyy pelkästään faktataulusta.

### Reseptin loppu

Kimball esittää tietovarastokannan suunnittelun kaikkiaan yhdeksän kohdan ohjelmana. Edellä esitettyjen neljän alkukohdan lisäksi päätetään seuraavista asioista tässä järjestyksessä:

5. Esitetään dimensio- eli tekijätaulujen attribuutit ja laaditaan niille kunnollista terminologiaa käyttäen tyhjentävät kuvaukset.
6. Valitaan tekniikka hitaasti muuttuvien dimensioiden (*slowly changing dimensions*) hallintaan. Hitaasti muuttuvia dimensioita

ovat esimerkiksi asiakkaiden tai tuotteiden perustiedot. Näiden hallintaan on olemassa useita tapoja.

7. Talletusrakenteeseen vaikuttavat yksityiskohdat kuten koostettujen, summatasointien tietojen ja heterogeenisten dimensioiden esittämistavat. Esimerkiksi tuotteet voivat olla nk. heterogeenisiä dimensioita, jos eri tuotteiden ominaisuuksien joukot poikkeavat toisistaan suuresti.
8. Miten pitkä aika, montako vuotta tietokantaan aiotaan dataa kerätä.
9. Millä viiveellä ajantasainen data viedään tietokantaan. Vaikka esitystarkkuudeksi onkin valittu vaikkapa yksi päivä, ei datan lataamisen tarvitse välttämättä tapahtua joka päivä.

Keittokirjassa on tavallisesti reseptien lisäksi ohje työn etenemisestä. Kimballkin esittää prosessin kuvauksen. Kirjassa on jopa luettelot kysymyksistä, jollaisia käyttäjille kannattaa tehdä, jotta halutut tiedot saadaan ei-teknisellä esitystavalla epäsuorasti selville. Toinen ohje on niiden perusjärjestelmien tietokannahoitajien haastatteluun, joiden kannoista tarvittavaa dataa joudutaan lataamaan.

Kimball on julkaissut kolmen muun tekijän kanssa toisenkin, kaksi kertaa paksumman menetelmäkirjan *The Data Warehouse Lifecycle Toolkit* (John Wiley & Sons 1998). Tämä osin toistaa mutta lähinnä täydentää edellä esiteltyä verrattomasti keittokirjaa ja keskittyy esittelemään hyvinkin yksityiskohtaisesti kehittämisprosessia ja projektin työ-, dokumentaatio- jne menettelyitä.

Jaakko Rantanen,  
Hämeen ammattikorkeakoulu,  
Tietojenkäsittelyn koulutusohjelma  
jaakko.rantanen@iki.fi