

**Jyrki Takala**

**Tietovaraston käsittekaavion johtaminen operatiivisten  
järjestelmien käsittekaavioista**

Tietojärjestelmätieteen  
pro gradu -tutkielma  
26.6.2002

Jyväskylän yliopisto  
Tietojenkäsittelytieteiden laitos  
Jyväskylä

## TIIVISTELMÄ

Takala, Jyrki Kalervo

Tietovaraston käsitekaavion johtaminen operatiivisten järjestelmien käsitekaavioista/Jyrki Takala

Jyväskylä: Jyväskylän yliopisto, 2002.

83 s.

Pro gradu -tutkielma

Tietovarastointi yleistyi 1990-luvulla, koska operatiiviset järjestelmät eivät kyenneet tukemaan yritysjohton päätöksentekoa tarpeeksi tehokkaasti. Tietovarastoinnilla tarkoitetaan tavallisesti päätöksenteossa tarvittavien tietojen poimimista operatiivisista järjestelmistä ja tietojen integrointia kattavaksi tietovarastoksi, joka mahdollistaa integroidun ja ajantasaisen tiedon monipuolisen analysoinnin, kyselyn ja raportoinnin.

Tietovarasto voi olla rakenteeltaan hyvin laaja ja monimutkainen, joten tietovaraston suunnittelu on hankala tehtävä. Tietovaraston käsitteellisen suunnittelun tavoitteena on tuottaa tietovaraston käsitekaavio. Erään lähestymistavan mukaan tietovaraston käsitekaavion suunnittelun pohjana toimivat operatiivisten järjestelmien käsitekaaviot. Tämän tutkielman tavoitteena on kuvata, verrata ja arvioida menetelmiä ja malleja, joita on esitetty tietovaraston käsitekaavion johtamiseksi operatiivisten järjestelmien käsitekaavioista ja pyrkiä esittämään parannusehdotuksia näihin malleihin.

Tässä tutkielmassa esitellään ensin tietovarastoinnin taustaa, tavoitteita ja hyötyjä, jonka jälkeen kuvataan tietovaraston rakennetta ja toimintaa ja esitellään tietovaraston suunnittelun tyypillinen kulku. Sen jälkeen kuvaillaan tietovaraston käsitekaavion johtamiseksi esitettyjä menetelmiä, joita vertaillaan ja arvioidaan sekä pyritään esittämään niihin parannusehdotuksia.

AVAINSANAT: tietovarastot, tietovaraston käsitekaavio, käsitteellinen suunnittelu, tähtimalli

## SISÄLLYS

1. JOHDANTO .....	4
2. TIETOVARASTOT JA NIIDEN SUUNNITTELU.....	7
2.1 Tietovarastoinnin tausta, tavoitteet ja hyödyt.....	7
2.2 Tietovaraston rakenne ja toiminta.....	10
2.3 Tietovaraston suunnittelu.....	15
3. MALLEJA JA MENETELMIÄ TIETOVARASTON KÄSITEKAAVION JOHTAMISEKSI OPERATIIVISTEN JÄRJESTELMIEN KÄSITEKAAVIOISTA...	21
3.1 Päälähestymistavat .....	21
3.2 Käsitekaavioiden integrointi .....	24
3.3 Golfarellin, Maion ja Rizzin menetelmä.....	27
3.4 Cabibbon ja Torlonen menetelmä.....	33
3.5 Böhnleinin ja Ulbrich-vom Enden menetelmä .....	39
3.6 Hüsemannin, Lechtenböckerin ja Vossenin menetelmä .....	44
3.7 Moodyn ja Kortinkin menetelmä.....	50
4. MENETELMIEN VERTAILU JA ARVIOINTI .....	60
4.1 Menetelmien tietomallit.....	60
4.2 Menetelmien tehtävät.....	64
4.3 Menetelmien subjektiivista arviointia ja parannusehdotuksia .....	70
5. YHTEENVETO.....	77
LÄHTEET .....	80

Tietovaraston suunnittelu on usein varsin haastava tehtävä, koska tietovarastoon tarvittavat tiedot sijaitsevat hajallaan operatiivisissa järjestelmissä. Tieto operatiivisissa järjestelmissä on epäyhtenäistä ja ristiriitaista. Sama tieto saattaa esiintyä eri paikoissa eri nimellä tai samannimiset tiedot eri paikoissa ovatkin sisällöltään erilaisia. (Inmon, 1996.) Hovin (1997) mielestä operatiivisten järjestelmien tieto on usein heikkolaatuista ja jopa väärää. Tästä syystä tietojen poiminta, puhdistus ja lataus tulee tehdä hyvin suunnitellulla tavalla ja hyvin suunniteltuun kohteeseen.

Tietovaraston suunnittelulle ovat esittäneet kattavia menetelmiä muun muassa Anahory ja Murray (1997), Devlin (1997), Golfarelli ja Rizzi (1998), Hovi (1997), Inmon (1996) ja Kimball (1996). Golfarellin ja Rizzin (1998) menetelmässä tietovaraston suunnittelu alkaa operatiivisten järjestelmien analysoinnilla, jonka jälkeen suoritetaan tietovaraston vaatimusmäärittely. Seuraavissa vaiheissa suoritetaan tietovaraston käsitteellinen suunnittelu ja arvioidaan tietovaraston kuormitusta. Viimeisinä vaiheina ovat tietovaraston looginen ja fyysinen suunnittelu. Tietovaraston käsitteellisen suunnittelun tavoitteena on tuottaa tietovaraston käsitekaavio. Hasan ym. (2000) ovat esittäneet kolme lähestymistapaa tietovaraston käsitteelliseen suunnitteluun. Tärkeimmän lähestymistavan mukaan tietovaraston käsitekaavio johdetaan operatiivisten järjestelmien käsitekaavioiden pohjalta.

Kirjallisuudessa on esitetty joitakin malleja ja menetelmiä tietovaraston käsitekaavion johtamiseksi operatiivisten järjestelmien käsitekaavioista. Menetelmiä ovat esittäneet esimerkiksi Golfarelli ym. (1998), Cabibbo ja Torlone (1998), Böhnlein ja Ulbrich-vom Ende (1999), Hüsemann ym. (2000) sekä Kortink ja Moody (2000). Missään näistä tai muistakaan julkaisuista ei ole kuitenkaan kootusti ja perusteellisesti esitelty ja vertailtu malleja ja menetelmiä, joiden avulla tietovaraston käsitekaavio voidaan johtaa operatiivisten järjestelmien käsitekaavioista.

Tässä tutkielmassa on kaksi tutkimusongelmaa:

- 1) Millaisia menetelmiä ja malleja on esitetty tietovaraston käsitekaavion johtamiseksi operatiivisten järjestelmien käsitekaavioista?
- 2) Miten näitä menetelmiä ja malleja voitaisiin parantaa?

# 1. JOHDANTO

Tiedon määrän kasvaessa on yrityksissä tullut tarve käyttää tietoa mahdollisimman tehokkaasti johdon päätöksenteon tukena. Usein päätöksenteossa tarvittavat tiedot ovat hajallaan useassa operatiivisessa järjestelmässä, joten niiden tehokas käyttö päätöksenteossa on vaikeaa. *Operatiivinen järjestelmä* (legacy system) on yrityksen tiettyä toimintoa varten suunniteltu tietojärjestelmä. Se on usein suunniteltu tiheästi tapahtuvia päivityksiä ajatellen, jolloin se ei pysty vastaamaan monipuolisiin kyselyihin tarpeeksi tehokkaasti. (esim. Adriaans ja Zantinge, 1996, 26.) Päätöksentekoa tukevaan kyselyyn tai raporttiin tarvitaan tavallisesti tietoja monesta operatiivisesta järjestelmästä, mutta operatiivisten järjestelmien tietojen yhdistäminen on hankalaa niiden erilaisen muodon ja erilaisten käyttöympäristöjen takia. Tyypillisiä operatiivisia järjestelmiä ovat esimerkiksi kirjanpitojärjestelmä, kaupan tuotejärjestelmä ja paikanvarausjärjestelmä. (Smith, 1997.)

Devlinin (1997) mielestä menestyminen liiketoiminnassa riippuu saatavilla olevan informaation nopeasta ja päättäväisestä käytöstä. Informaation puuttuminen päätöksenteon hetkellä tietää usein päätöksen epäonnistumista. Koska operatiiviset järjestelmät eivät pysty palvelemaan yritysjohton tarpeita päätöksenteossa tarpeeksi nopeasti ja monipuolisesti, on yksi mahdollinen vaihtoehto tietovaraston rakentaminen.

*Tietovarasto* on kokoelma tietoja, jotka on muotoiltu erityisesti tehokkaita kyselyjä sekä monipuolista analysointia ja raportointia varten. Tietovarastoon siirretään tietoja operatiivisista järjestelmistä ja mahdollisesti ulkopuolisistakin lähteistä, jolloin eri lähteiden tietoja voidaan yhdistää samaan raporttiin tai kyselyyn. Tietovarasto sisältää perustietojen lisäksi historiatietoja ja perustiedoista johdettuja tietoja. (esim. Hovi, 2001.) Tietovarasto (data warehouse) esiintyi terminä ensimmäisen kerran tässä merkityksessä 1990-luvun alkupuolella. Amerikkalaista William Inmonia pidetään tietovarastointi-termin keksijänä. (Hovi, 1997.)

Tutkielman tavoitteena on kuvata, verrata ja arvioida menetelmiä ja malleja, joita on esitetty tietovaraston käsittekaavion johtamiseksi operatiivisten järjestelmien käsittekaavioista ja pyrkiä esittämään parannusehdotuksia näihin malleihin. Tutkimusmenetelmänä käytetään käsitteellisteoreettista tutkimusta. Menetelmien ja mallien arviointia varten luodaan yhtenäinen arviointikehys. Tutkimustuloksena saatavat arviot ja parannusehdotukset hyödyttävät tietovarastosuunnittelua tuntevia ammattilaisia. Tutkielman tuloksista selviävät kunkin menetelmän sisältämät vaiheet, vahvuudet ja heikkoudet sekä mahdolliset parannuskohteet. Näiden tulosten perusteella voidaan jatkotutkimusta tulevaisuudessa suunnata koskemaan menetelmien ja mallien sopivuutta erilaisten tietovarastojen yhteydessä.

Tutkielma on jäsennetty viiteen lukuun. Luvussa 2 tarkastellaan tietovarastoja yleisellä tasolla. Luvussa esitellään tietovarastoinnin taustaa, tavoitteita ja hyötyjä. Lisäksi esitellään tietovaraston rakenne ja toiminta sekä tietovaraston suunnittelun yleinen kulku. Luvussa 3 haetaan vastausta ensimmäiseen tutkimusongelmaan: ”Millaisia menetelmiä ja malleja on esitetty tietovaraston käsittekaavion johtamiseksi operatiivisten järjestelmien käsittekaavioista?” Luvussa esitellään ensin tietovaraston suunnittelun päälähestymistavat, jonka jälkeen käsitellään tietokannan käsittekaavioiden integrointia. Varsinaisia tietovaraston käsittekaavion johtamiseksi esitettyjä menetelmiä on käsiteltävänä viisi. Luvussa 4 on esitetty yhteenveto luvussa 3 esitetyistä malleista ja menetelmistä. Malleja ja menetelmiä myös vertaillaan toisiinsa ja esitetään vastauksia toiseen tutkimusongelmaan: ”Miten esitettyjä menetelmiä ja malleja voitaisiin parantaa?” Tutkielma päättyy luvun 5 yhteenvetoon.

## **2. TIETOVARASTOT JA NIIDEN SUUNNITTELU**

Tässä luvussa esitellään tietovarastoinnin taustaa, tavoitteita ja hyötyjä. Luvussa kuvataan myös tietovaraston rakenne sekä tietovaraston suunnittelun tyypillinen kulku. Esimerkkinä tietovaraston suunnittelusta käytetään Golfarellin ja Rizzin (1998) esittämää vaihejakoa.

### **2.1 Tietovarastoinnin tausta, tavoitteet ja hyödyt**

Tietovarastoinnin historia ei ulotu kovin kauas menneisyyteen, mutta tietovarastoinnin tarpeen ja merkityksen ymmärtämiseksi asioiden tarkastelu täytyy aloittaa hieman kauempaa. Tietovarastoinnin periaatteita oli nähtävissä 1980-luvulla johdon tietojärjestelmissä (management information systems) ja myöhemmin päätöstukijärjestelmissä (decision support systems). Jo näiden järjestelmien yhteydessä havaittiin ongelmaksi päätöksentekoon tarvittavien tietojen hajanaisuus operatiivisissa järjestelmissä. Aiemmin oli jo havaittu, että operatiiviset järjestelmät oli suunniteltu ensisijaisesti jatkuvia päivityksiä ja hakuja ajatellen, eivätkä ne sen takia pystyneet vastaamaan monipuolisiin kyselyihin, joita tehokas päätöksenteon tukeminen olisi vaatinut. Oli kehitettävä menetelmä, jolla tarpeellinen tieto saatiin poimittua operatiivisista järjestelmistä ja muutettua yhtenäiseen ja samalla käyttökelpoisempaan muotoon. Ensimmäisiä onnistuneita tietovarastoratkaisuja oli ABN AMRO:n pankkijärjestelmä, jonka kehitystyö tapahtui 1986-1989. (Devlin, 1997; Inmon, 1996.)

1990-luvulla yritysten liiketoimintavaatimukset muuttuivat. Menestyksellisiin liiketoimintaratkaisuihin tarvittavia koko yrityksen kattavia tietoja oli saatava yritysjohdon käyttöön. Myös tietohallinnolla oli tarve pystyä hallitsemaan yrityksen tietoja entistä paremmin. Tietovarastolle oli näin ollen selkeä tarve yhä useammassa

organisaatiossa. (Devlin, 1997) Simon ja Shaffer (2001) pitävät hajautettujen tietokannan hallintajärjestelmien epäonnistumista suurimpana syynä tietovarastoinnin yleistymiselle.

Tietovarasto (data warehouse) esiintyi terminä ensimmäisen kerran tässä merkityksessä 1990-luvun alkupuolella, jonka jälkeen aihe on ollut suosittu tutkimuskohde. Amerikkalaista William (Bill) Inmonia pidetään tietovarastointi-termin keksijänä. *Tietovarastolla* tarkoitetaan yritysjohdon päätöksenteon tueksi koottua tietokokonaisuutta, joka sisältää yleensä useasta operatiivisesta järjestelmästä integroitua perustietoa, perustietojen pohjalta johdettuja tietoja ja historiatietoja. Osa tiedoista voi olla peräisin myös ulkopuolisista lähteistä. Tietovarastoa käytetään analysointiin, raportointiin ja kyselyihin. (esim. Hovi, 2001.) Laine ja Rantanen (1997) määrittelevät tietovaraston olevan pelkästään hakuja varten laadittu. Suppeasti ajateltuna määritelmä pitää paikkansa.

Kirjallisuudessa on esitetty tietovarastosta toisenlaisiakin määritelmiä johtuen siitä, että termille ei ole olemassa vakiintunutta määritelmää (Simon ja Shaffer, 2001). Inmon (1996, 33) määrittelee tietovaraston kokoelmaksi kohdesuuntautunutta (subject oriented), yhtenäistettyä, pysyvää ja aikasidonnaista tietoa, jonka tarkoitus on tukea yritysjohtoa päätöksenteossa (ks. myös Aadrians ja Zantinge, 1996, 26). Simon ja Shaffer (2001, 9) ovat kuitenkin sitä mieltä, että Inmonin määritelmä ei päde enää nykyaikana. Heidän mielestään Inmonin perinteisessä määritelmässä tehdään liikaa oletuksia tietovaraston mahdollisuuksista. Nykyaikaisissa tietovarastoissa on muun muassa mahdollista päivittää tietovaraston tietoja reaaliaikaisesti, jolloin tietovaraston tiedot eivät enää olekaan pysyviä (vrt. Anahory ja Murray, 1997, 4). Lehtonen ja Leinonen (1987, 2) tarkoittavat tietovarastolla tietokantojen kokoelmaa.

Smith (1997) mainitsee tietovarastoinnin tavoitteeksi tarpeen laadukkaammalle tiedolle, joka on helposti saatavilla ja analysoitavissa. Toisena tavoitteena hän mainitsee tiedon keräämisen yhteen, millä säästetään tietohallinnon aikaa sekä lisätään tiedon käyttäjien ja järjestelmien tuottavuutta.



Tietovarastojen ensisijaisena tavoitteena on tukea yritysjohtoa päätöksenteossa. On kuitenkin olemassa myös muita tavoitteita. Kimball (1996, xxv-xxvii) esittää tietovarastoinnin tavoitteet kuutena vaatimuksena:

- Tietovaraston tulee tarjota kattavasti koko organisaatiosta kerättyjä tietoja yritysjohtoon ja muiden tietoja tarvitsevien päätöksenteon tueksi lyhyellä vasteajalla.
- Tietovaraston sisältämän tiedon tulee olla johdonmukaista.
- Tietovaraston sisältämiä tietoja tulee voida osittaa ja vertailla kaikin mahdollisin tavoin.
- Tietovaraston tulee tarjota tietosisältönsä lisäksi välineet myös tiedon kyselyyn, analysointiin ja esittämiseen.
- Tietovarastoon viedään vain käyttökelpoista ja tarpeelliseksi todettua tietoa.
- Tietovaraston sisältämän tiedon tulisi ohjata liiketoiminnan uudelleenjärjestelyä.

Kaikkia edellä mainittuja seikkoja ei voi pitää varsinaisina tavoitteina, vaan paremminkin toteamuksina siitä, millaisen hyvän tietovaraston tulisi olla. Tietovarastoinnin tavoitteiden yhteenvedona voitaisiin sanoa, että tietovaraston tärkein tehtävä on tarjota nopeasti oikeanlaista ja ajantasaista tietoa sitä tarvitsevan henkilön käyttöön aina, kun tarve vaatii.

Tietovaraston käytöstä on yleensä ollut hyötyä, jos edes osa edellä mainituista tavoitteista saavutetaan. Kirjallisuudessa on esitetty useita tietovaraston käytön hyötynäkökohtia.

Westermanin (2001) mielestä tietovaraston raporttien ja kyselyjen avulla on mahdollista ymmärtää paremmin mitä yrityksessä tapahtuu tai tapahtui. Parempi ymmärrys puolestaan tuottaa jatkossa parempia liiketoimintapäätöksiä. Tietovarastoinnin avulla on mahdollista löytää syitä tehdyille virheille ja ottaa niistä oppia. Adriaans ja Zantinge (1996) toteavat, että organisaatioiden on pystyttävä vastaamaan nopeasti markkinoilla tapahtuviin muutoksiin. Päätösten tueksi tarvitaan tietoa sekä nykyhetkestä että menneistä liiketoimintasuuntauksista. Tietovarastoon on mahdollista kerätä päätöksenteon kannalta tarpeellinen tieto ja saada se käyttöön tarvittavalla hetkellä. Smithin (1997) mukaan tietovarastoinnilla on ollut konkreettista vaikutusta yritysten

kilpailukykyyn kohonneiden tulojen muodossa sekä tiedonhallintaan vähentyneiden tiedonhallintakustannusten myötä.

Smith (1997) luettelee tietovarastoinnin hyödyiksi lisäksi seuraavia seikkoja:

- asiakasvaatimusten tarkempi ennustaminen tietovaraston historiatietojen perusteella tehtävän trendianalyysin avulla
- suoramarkkinointikampanjoiden tehostuminen talouksien taustatietojen ja asiakasanalyysien avulla
- kauppasuhteiden paraneminen ja hintojen aleneminen sen seurauksena, että etsitään ne toimittajat, jotka käyvät kauppaa yrityksen muidenkin osastojen kanssa
- huomattavat säästöt koko yrityksessä tiedon laadun paranemisen kautta.

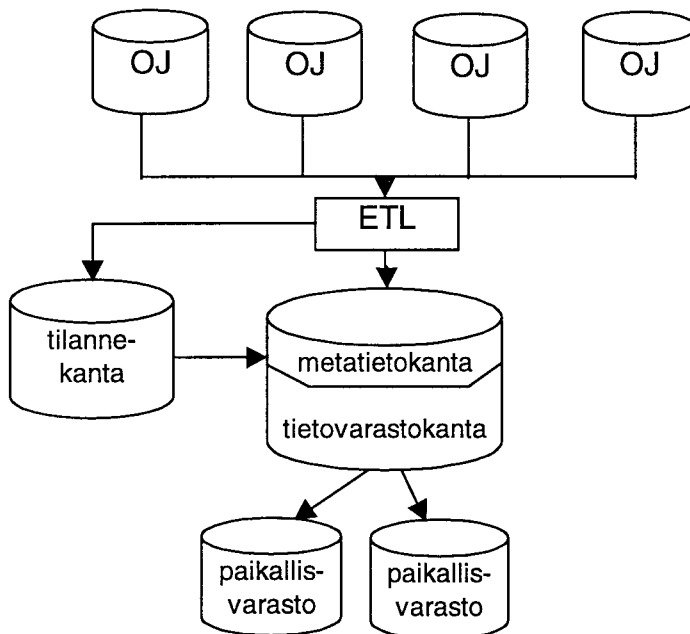
Hovin (2001) mielestä tietovaraston tietojen perusteella organisaation liiketoimintaprosesseja voidaan muokata uudelleen, jolloin tietovarasto palvelee myös yritysjohdon pitkän tähtäimen strategiaa ja tavoitteita. Tyypillisiksi tietovaraston käyttäjiksi organisaatioissa Hovi (1997) mainitsee tuotepäälliköt, markkinoinnin suunnittelijat, tuotevastaavat, asiakasvastaavat ja muut keskijohdon edustajat.

Tietovarastointi kehitettiin alun perin liiketoiminnan tarpeisiin. Nämä tarpeet ovat edelleen olemassa ja tulevaisuudessa tarpeet voivat jopa kasvaa liiketoiminnan teknologisen ympäristön muuttuessa entistä monimutkaisemmaksi. (Devlin, 1997.)

## 2.2 Tietovaraston rakenne ja toiminta

Kuviossa 1 on esitetty tietovaraston rakenne ja sitä ympäröivät järjestelmät. Tietovarasto koostuu varsinaisen tietovarastokannan lisäksi metatietokannasta, joka sisältää tietovaraston määrittävän metatiedon. Tietovarastossa olevien tietojen rakenne ja sisältö määritellään *metatiedon* avulla. Metatieto voi sisältää muun muassa lähdetietoja, tietomääritelmiä ja johdettujen tietojen kaavoja (esim. Berson ja Smith,

1997). Adriaansin ja Zantingen (1996) mielestä metatieto on oleellista loppukäyttäjien ja tietovaraston ylläpitäjän kannalta. Heidän mielestään tarkoituksenmukainen metatieto on välttämätöntä tietovaraston yhteydessä (ks. myös Inmon, 1996, 110).



KUVIO 1. Tietovaraston rakenne ja ympäröivät järjestelmät.

Tietovarastokanta sisältää perustauluja ja summatauluja. *Perustauluilla* tarkoitetaan yksityiskohtaista tietoa sisältäviä tauluja, jotka ovat yleensä isoja. Usein toistuvia kyselyjä varten tietovarastoon kannattaakin perustaa *summatauluja*. Ne on johdettu isoista perustauluista tietoja yhdistelemällä usein monenkin taulun tietoja. Summataulut voivat sisältää myös perustaulujen tiedoista kaavojen avulla johdettua tietoa. Johdettua tietoa voivat olla esimerkiksi erilaiset asiakastuotot. Summataulun koko on yleensä alle 10% perustaulun koosta. Summaamisen ansiosta tietovarastoon tehtävät kyselyt nopeutuvat huomattavasti. Tyypillisiä summatauluja ovat esimerkiksi liikkeen viikko- tai kuukausikohtaiset myyntimäärät. (Hovi, 1997.)

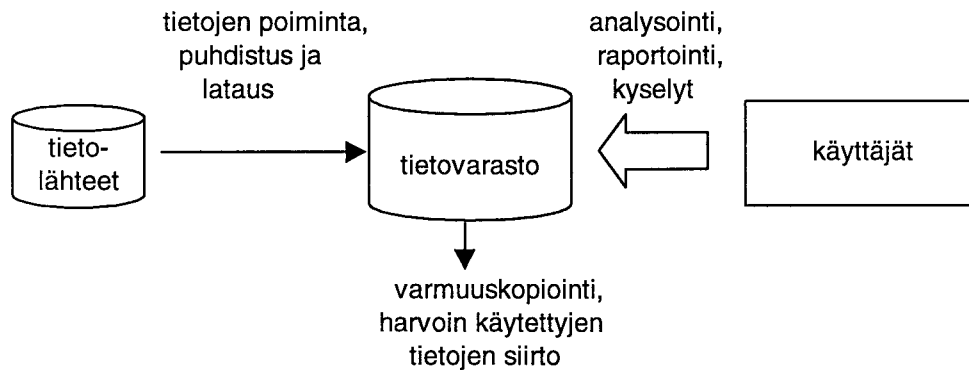
*Operatiivinen järjestelmä* (legacy system) on usein tapahtuvaa päivitystä ajatellen suunniteltu tietojärjestelmä. Tyypillisiä operatiivisia järjestelmiä ovat esimerkiksi pankin tilijärjestelmä ja kirjanpitojärjestelmä. (Hovi, 2001.) Tietojen poimimista operatiivisista järjestelmistä, tietojen puhdistamista ja latausta tietovarastoon kutsutaan ETL-prosessiksi (Extraction, Transformation, Loading) (Simon ja Shaffer, 2001, 11). Siirrettäessä tietoja operatiivisista järjestelmistä tietovarastoon koodeja yhdenmukaistetaan ja tiedoille tehdään oikeellisuustarkistuksia. Tätä kutsutaan tietojen puhdistamiseksi tai siivoamiseksi. Tavallisimmin tietovarasto sijoitetaan omalle koneelle, erilleen operatiivisista järjestelmistä, ja tietojen poimintaa ja siirtoa varten operatiivisista järjestelmistä on olemassa erityisiä ohjelmistoja. Tietojen lataus tietovarastoon vie aikaa. Lataamisessa tiedot ensin tarkistetaan, tarvittaessa muokataan, muunnetaan koodeja ja mahdollisesti denormalisoidaan. Lopulta tiedot ladataan fyysisesti tietovarastokantaan. (Hovi, 1997.)

*Tilannekanta* (Operational Data Store, ODS) sisältää lähes ajantasaista tietoa eri operatiivisista kannoista. Tilannekannan tehtävänä on toimia yhdistävänä tekijänä operatiivisten järjestelmien välillä siten, että niistä voidaan lukea ja kirjoittaa tietoa tilannekantaan. Tilannekannasta tietoja voidaan siirtää tietovarastoon tasaisin väliajoin. (Inmon, 1995.)

Tietovarastosta tietoja voidaan siirtää edelleen paikallisvarastoihin. Jarke ym. (2000, 183) määrittelevät *paikallisvaraston* (datamart) pieneksi tietovarastoksi, joka voi sisältää esimerkiksi yhteen osastoon liittyvät tiedot. Simonin ja Shafferin (2001) mukaan paikallisvarasto voi sisältää tietoja myös muualta kuin tietovarastosta.

Kuviossa 2 on esitetty tiedon kulku tietovarastossa ja sitä ympäröivissä järjestelmissä. Ensin tietovarastoon tarvittavat tiedot poimitaan tietolähteistä, joita ovat tavallisesti operatiiviset järjestelmät. Tiedot voivat olla peräisin myös ulkopuolisista lähteistä. Seuraavassa vaiheessa poimitut tiedot puhdistetaan ja ladataan fyysisesti tietovarastoon. Tämän jälkeen tiedot ovat tietovaraston käyttäjien käytettävissä analysointia, raportointia ja kyselyjä varten. Tasaisin väliajoin tietovaraston tiedoista on syytä ottaa varmuuskopio. Tietyin väliajoin hyvin harvoin käytettyjä tietoja siirretään

tallennettavaksi jonnekin muualle, varsinkin jos ylimääräinen tieto hidastaa tietovaraston toimintaa.

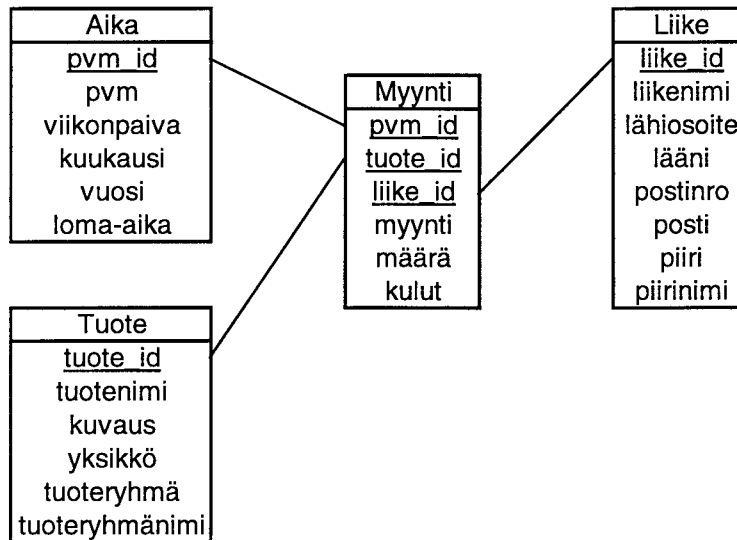


KUVIO 2. Tiedon kulku ja käyttö tietovarastossa.

Tietovarastossa on tietoa monella eri tarkkuustasolla. Esimerkiksi myynneistä voi olla saatavilla päivä-, viikko-, kuukausi- ja vuosikohtaisia lukemia. Pidemmän ajanjakson tiedot ovat lyhyemmän ajanjakson tiedoista johdettua tietoa. Kohteeseen liittyvillä tiedoilla on aina aikaelementti. Aika voi olla päivämäärä, viikko, kuukausi, vuosi tai kahden ajankohdan välinen ajanjakso. Näin ollen tietovarasto sisältää myös historiatietoja. (esim. Inmon, 1996.)

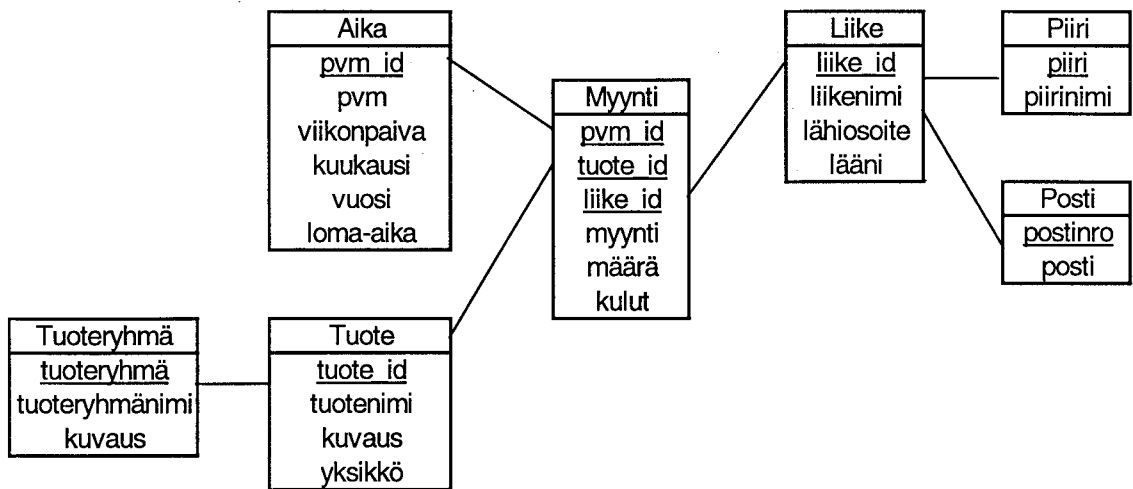
*Moniulotteisella mallilla* (multidimensional model) tarkoitetaan faktatauluista ja ulottuvuustauluista koostuvaa tietovaraston käsittekaaviota (esim. Cabibbo ja Torlone, 1998). Moniulotteisia malleja ovat esimerkiksi tähtimalli ja lumihiihtalemalli. Joskus käytetään myös nimitystä ulottuvuusmalli, vaikka tarkoitetaan samoja kaavioita (esim. Moody ja Kortink, 2000). *Tähtimalli* eli dimensiomalli on tietovarastokannan suunnittelumalli, joka soveltuu parhaiten numeerista ja moniulotteista tietoa sisältävän tietovaraston malliksi (esim. Hovi, 2001). Tähtimalli koostuu faktataulusta ja ulottuvuustauluista (KUVIO 3). *Faktataululla* tarkoitetaan tietovaraston tähtimallin keskuksena olevaa taulua, joka sisältää päätöksenteon kohteena olevat mittauskohteet ja avainattribuutteinaan ulottuvuustaulujen avainattribuutit. Faktataulun tiedot ovat

luonteeltaan tapahtumatyyppisiä. *Ulottuvuustauluissa* on yleensä tekstimuotoista tietoa, jotka kuvaavat faktatauluun liittyviä tietoja päätöksenteon kannalta kiinnostavasta näkökulmasta. Faktataulussa tiedot eivät toistu, mutta ulottuvuustaulut ovat normalisoimattomia eli niissä jotkut tiedot voivat toistua. Faktataulussa on perusavaimen lisäksi *mittauskohteet*, joilla voidaan laskea yhteenvedoja niiden ulottuvuuksien perusteella, joita tietovaraston käyttäjä haluaa. (esim. Kimball, 1997.) Esimerkiksi kauppaketjun tietovarastosta voidaan hakea tiedot yhden liikkeen myynneistä yhden viikon ajalta tietyn tuotteen osalta.



KUVIO 3. Myynnin seurannan tähtimalli, joka koostuu faktataulusta ja ulottuvuustauluista (vrt. Hovi, 2001, 96).

Jos ulottuvuustaulutkin normalisoidaan, tähtimalli muuttuu *lumihiihtomalliksi* (KUVIO 4), koska ulottuvuustauluja syntyy normalisoinnin seurauksena lisää. Normalisoinnilla tarkoitetaan sitä, että tähtimallin ulottuvuustauluissa mahdollisesti esiintyvistä sisäisistä hierarkioista hankkiudutaan eroon muodostamalla riippumattomille hierarkioille omat ulottuvuustaulut. (esim. Moody ja Kortink, 2000.) Esimerkiksi kuvion 3 tähtimallin liike-ulottuvuustaulussa posti ja postinumero sekä piiri ja piirinimi muodostavat kaksi erillistä hierarkiaa, joita varten kuviossa 4 on luotu omat ulottuvuustaulunsa.



KUVIO 4. Myynnin seurannan lumihiihtalemalli, joka koostuu faktataulusta ja normalisoiduista ulottuvuustauluista (vrt. Hovi, 2001, 101).

Sekä tähti- että lumihiihtalemalleista on olemassa monia muunnoksia (ks. Moody ja Kortink, 2000), joilla on omat hyvät ja huonot puolensa. Tavallisesti mallin lopullinen valinta tehdään tietojen tallennuskustannusten ja kyselyjen tehokkuuden perusteella. (Böhnlein ja Ulbrich-vom Ende, 1996.)

Tietovarasto kasvaa koko ajan, kun uusia tietoja lisätään. Vanhimpiä tietoja voidaan siirtää tietovarastosta pois tallennettavaksi muussa muodossa, jos tietoihin tehdään enää hyvin vähän kyselyjä. (Hovi, 1997.) Tietovarasto voi siis olla rakenteeltaan hyvinkin laaja ja monimutkainen. Siksi tietovaraston suunnittelu on haastava tehtävä.

### 2.3 Tietovaraston suunnittelu

Kirjallisuudessa on esitetty lukuisia menetelmiä tietovaraston suunnittelulle. Suunnittelumenetelmät poikkeavat toisistaan jonkin verran. Kattavia tietovaraston suunnittelumenetelmiä ovat esittäneet esimerkiksi Anahory ja Murray (1997), Devlin

(1997), Golfarelli ja Rizzi (1998), Hovi (1997), Inmon (1996) ja Kimball (1996). Kattavalla suunnittelumenetelmällä tarkoitan koko tietovaraston suunnitteluprosessia tukevaa ohjeistoa. Selkeän vaihejaon, mutta suppeamman esityksen tietovaraston suunnittelulle ovat esittäneet esimerkiksi Hüseman ym. (2000). Myös Moody ja Kortink (2000) käsittelevät tietovaraston suunnittelua.

Edellä mainituista suunnittelumenetelmistä olen valinnut tarkemman läpikäynnin kohteeksi Golfarellin ja Rizzin (1998) menetelmän, koska heidän esityksessään on selkeä ja helppotajuinen vaihejako. Lisäksi heidän menetelmänsä on käyttökelpoinen tämän tutkielman jatkoon kannalta, koska esitettävästä vaihejaosta löytyy selvästi paikka myös tietovaraston käsittekaavion johtamiselle operatiivisten järjestelmien käsittekaavioista. Seuraavaksi kuvataan tietovaraston suunnittelun kulku Golfarellin ja Rizzin (1998) mukaan ja sen jälkeen tarkastellaan lyhyesti muiden esittämiä suunnittelumenetelmiä.

### **Golfarellin ja Rizzin suunnittelumenetelmä**

Golfarellin ja Rizzin (1998) menetelmän mukaan tietovaraston suunnittelu jakaantuu seitsemään vaiheeseen (TAULUKKO 1). Operatiivisen järjestelmän analysointivaiheen tavoitteena on kerätä tietoja olemassa olevasta operatiivisesta järjestelmästä. Mukaan olisi mahdollisuuksien mukaan saatava operatiivisen järjestelmän suunnittelijoita ja ylläpitäjiä. Vaiheen tuotoksena on operatiivista järjestelmää tai sen osia kuvaavat käsitteelliset tai loogiset kaaviot. Tässä vaiheessa joudutaan usein tekemään näkemysten integrointia, koska operatiivinen järjestelmä voi sisältää eri tavoin toteutettuja tietokantoja.



TAULUKKO 1. Tietovaraston suunnittelun vaihejako Golfarellin ja Rizzin (1998) mukaan.

Vaihe	Syöte	Tuotos
Operatiivisen järjestelmän analysointi	operatiivisen järjestelmän dokumentaatio	tietokannan kaavio
Vaatimusten määrittely	tietokannan kaavio	faktat, alustava kuormitus
Käsitteellinen suunnittelu	tietokannan kaavio, faktat, alustava kuormitus	ulottuvuuskaavio
Kuormituksen tarkentaminen ja kaavion vahvistaminen	alustava kuormitus, ulottuvuuskaavio	kuormitus, vahvistettu ulottuvuuskaavio
Looginen suunnittelu	vahvistettu ulottuvuuskaavio, kuormitus	tietovaraston looginen kaavio
Fyysinen suunnittelu	looginen kaavio, kuormitus	tietovaraston fyysinen kaavio

Vaatimusten määrittelyn tavoitteena on kerätä tietovaraston suunnittelijoiden ja tulevien käyttäjien vaatimuksia tietovarastolle. Vaatimusmäärittelyn avulla kerätään tietoa tietovaraston faktoista ja tietovaraston kuormituksesta. Faktojen valinta tehdään pääasiallisesti edellisessä vaiheessa kerätyn dokumentaation tuella. ER-mallissa fakta vastaa joko kohdetyyppiä tai n-ääristä suhdetyyppiä. Tietovaraston kuormituksen arvioitu määrä kuvataan luonnollista kieltä muistuttavalla pseudokielellä, ja sen tarkoitus on auttaa tietovaraston suunnittelijaa tunnistamaan ulottuvuuksia ja mittauskohteita käsitteellisen suunnittelun aikana. Jokaista faktaa kohden tulisi määrittellä tärkeimmät mittauskohteet ja koostetiedot.

Käsitteellisessä suunnittelussa lähtökohtana toimivat operatiivisen järjestelmän käsittekaaviot. Apuna käytetään myös edellisissä vaiheissa tuotettuja dokumentteja, joiden pohjalta tuotetaan tietovaraston käsitteellistä rakennetta kuvaava ulottuvuuskaavio, joka koostuu useasta faktakaaviosta. Jokaisesta edellisvaiheesta löydetyistä faktoista tehdään faktakaavio, joka sisältää yhden faktan ja siihen liittyvät

attribuutit, ulottuvuudet, mittauskohteet ja ulottuvuuksien hierarkiat. Hierarkialla tarkoitetaan ulottuvuustaulun attribuutteja.

Seuraavassa vaiheessa tietovaraston arvioitua kuormitusta tarkennetaan ja lisäksi edellisessä vaiheessa tuotetun ulottuvuuskaavion kelpoisuutta tutkitaan. Kuormituksen tarkentaminen tapahtuu määrittelemällä tietovarastoon tehtäviä kyselyjä sekä arvioimalla käsiteltävän tiedon määrää. Kyselyjen mallintamista varten on käytettävissä yksinkertainen kieli, ja käsiteltävän tietomäärän laskemiselle on esitetty kaavoja.

Tietovaraston loogisen suunnittelun lähtökohtina toimivat aiemmin tuotettu tietovaraston ulottuvuuskaavio, kuormitustiedot sekä lisätiedot koskien muun muassa tietovaraston päivitysvälejä ja käytössä olevaa levytilaa. Näiden tietojen perusteella loogisen suunnittelun tavoitteena on tuottaa sellainen tietovaraston looginen malli, jonka avulla tietovarastoon tehtävien kyselyjen vastausaika olisi mahdollisimman pieni. Looginen suunnittelu on jaettu neljään osavaiheeseen:

- näkemysten toteuttaminen (view materialization)
- ulottuvuuskaavion muuntaminen loogiseksi kaavioksi
- faktataulujen pystysuora osittaminen
- faktataulujen vaakasuora osittaminen.

Näkemysten toteuttamisessa määritellään ne näkymät, joita käytetään kyselyjen tehostamiseen. Usein toistuvia kyselyjä varten kannattaa tehdä omat näkymänsä, vaikka samalla joudutaankin käyttämään enemmän levytilaa.

Golfarelli ja Rizzi (1998) esittävät ohjeet ulottuvuuskaavion muuntamiseksi tietovaraston loogiseksi kaavioksi. Muuntaminen tapahtuu varsin suoraviivaisesti selkeiden sääntöjen mukaan, mutta muuntamista ei ole tässä tutkielmassa tarpeellista käydä yksityiskohtaisesti läpi.

Faktataulujen pystysuoralla osittamisella tarkoitetaan faktataulujen jakamista osiin sellaisessa tapauksessa, että faktataulu sisältää mittauskohteita, joita ei yleensä tarvita samoissa kyselyissä. Tällöin on tietovaraston tehokkuuden kannalta parempi perustaa

kullekin mittauskohteelle omat faktataulunsä. Faktataulujen vaakasuoralla osittamisella pyritään hakutehokkuuden lisäämiseen ottamalla huomioon, miten usein kyselyt kohdistuvat tiettyihin faktataulun ulottuvuuksiin.

Tietovaraston fyysisessä suunnittelussa määritellään tietojen tallennusrakenteet ja indeksointi. Määrittely tehdään tietovaraston loogisen kaavion ja aiemmin määritellyn kuormitustiedon perusteella.

### **Muita suunnittelumenetelmiä**

Kirjallisuudessa on esitetty myös Golfarellin ja Rizzin (1998) menetelmästä poikkeavia tietovaraston suunnittelumenetelmiä. Hüsemann ym. (2000) esittävät tietovaraston suunnittelulle samanlaista vaihejakoa kuin tietokannan suunnittelussa on perinteisesti käytetty. Menetelmässä tietovaraston suunnittelu on jaettu neljään vaiheeseen:

- vaatimusten määrittely
- käsitteellinen suunnittelu
- looginen suunnittelu
- fyysinen suunnittelu.

Moody ja Kortink (2000) ovat myös sitä mieltä, että tietovaraston suunnitteluun tulisi käyttää samanlaista mallia kuin tietokannan suunnitteluun. Heidän mielestään vasta paikallisvarastojen suunnittelussa kannattaisi käyttää moniulotteisia suunnittelumalleja. Kimball (1996) ja Hovi (1997) taas lähestyvät tietovaraston suunnittelua käytännönläheisemmästä näkökulmasta kuin Golfarelli ja Rizzi (1998). Hüsemannin ym. (2000) esittämä tietovaraston suunnittelun vaihejako on melko paljon Golfarellin ja Rizzin (1998) mallin kaltainen.

Suunnitteluvaiheita määrittävien menetelmien lisäksi on kirjallisuudessa esitetty erityisiä tietomalleja. Esimerkiksi Tryfonan ym. (1999) starER-mallissa on yhdistetty perinteinen ER-malli tietovarastojen suunnittelussa yleisesti käytetyn tähtimallin kanssa. StarER-mallilla toteutettu tietovaraston käsitekaavio koostuu rakenteista, jotka

kuvaavat faktoja, kohdetyyppejä, suhdetyyppejä ja attribuutteja. Tietovaraston hierarkiat koostuvat yleensä kohdetyyppien ketjusta.

Joistakin eroista huolimatta useat tietovaraston suunnittelumenetelmät katsovat, että tietovaraston käsitekaavion suunnittelun perustana ovat olemassa olevien operatiivisten järjestelmien käsitekaaviot. Alan kirjallisuudessa on esitelty malleja ja menetelmiä, joiden avulla on mahdollista johtaa tietovaraston käsitekaavio olemassa olevien operatiivisten järjestelmien käsitekaavioista. Näitä malleja ja menetelmiä esitellään seuraavassa luvussa.

### **3. MALLEJA JA MENETELMIÄ TIETOVARASTON KÄSITEKAAVION JOHTAMISEKSI OPERATIIVISTEN JÄRJESTELMIEN KÄSITEKAAVIOISTA**

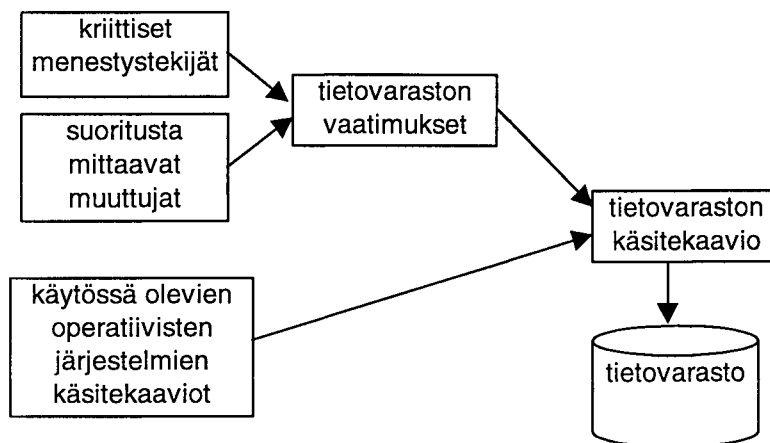
Kirjallisuudessa on esitetty useita malleja ja menetelmiä tietovaraston käsitekaavion johtamiseksi operatiivisten järjestelmien käsitekaavioista (esim. Golfarelli ym. (1998), Cabibbo ja Torlone (1998), Böhnlein ja Ulbrich-vom Ende (1999), Hüsemann ym. (2000) sekä Kortink ja Moody (2000)). Tässä luvussa esitellään näitä malleja ja menetelmiä. Kustakin menetelmästä kuvataan ensin käytetty malli, jonka jälkeen kuvataan yksityiskohtaisesti vaiheet tietovaraston käsitekaavion tuottamiseen. Luvun alussa esitellään tietovaraston suunnittelun päälähestymistavat, jonka jälkeen käsitellään tietokannan käsitekaavioiden integrointia, koska kaikki esiteltävät menetelmät edellyttävät integroinnin suorittamista ennen varsinaisen menetelmän käyttöä.

#### **3.1 Päälähestymistavat**

Hasan ym. (2000) esittävät kaksi erilaista päälähestymistapaa tietovaraston suunnitteluun: ylhäältä alas (top-down) ja alhaalta ylös (bottom-up). Lähestymistapoja on havainnollistettu kuviossa 5. Edellisten lähestymistapojen pohjalta Halen ym. (2000) ovat kehittäneet myös keskeltä ulos (middle-out) -lähestymistavan, joka sisältää piirteitä kahdesta muusta lähestymistavasta. Seuraavaksi esitellään lyhyesti kunkin lähestymistavan ominaispiirteet.

Ylhäältä alas -lähestymistavassa pyritään ensin määrittelemään yrityksen kriittiset menestystekijät ja yrityksen suoritusta mittaavat muuttujat. Niiden avulla selvitetään tietovaraston vaatimukset. Vaatimusten pohjalta määritetään tietovaraston moniulotteisen mallin faktat ja ulottuvuudet. Moniulotteisen mallin valmistuttua on

mahdollista rakentaa tietovaraston prototyyppi, jota voidaan testata. Testaamisen perusteella päätetään mahdollisista muutoksista ja lopulta aloitetaan lopullisen tietovaraston rakentaminen. Prototyypin toteutus- ja testausvaiheessa ei vielä päätetä, mistä lähteestä ja miten tietovarasto oikeasti saisi tarvitsemansa tiedot. Tämä onkin lähestymistavan heikkous, koska lopulliseen tietovarastoon tarvittavat tiedot on joka tapauksessa saatava olemassa olevista operatiivisista järjestelmistä. Operatiivisten järjestelmien tiedon siirtäminen tietovarastoon on lähestymistavasta riippumatta hankala tehtävä. Toisena heikkoutena voidaan pitää tietovaraston räätälöintiä pelkästään yhden käyttäjäryhmän tarpeisiin, jolloin tietovarastoa ei yleensä pystytä käyttämään missään muualla ilman tuntuvia muutoksia. Ylhäältä alas -lähestymistapa sopii parhaiten vahvasti liiketoimintapainotteisen tietovaraston suunnitteluun ja varsinkin sellaiseen tilanteeseen, jossa prototyypin tekeminen on mahdollisimman yksinkertaista.



KUVIO 5. Tietovaraston suunnittelun kaksi päälähestymistapaa.

Myös alhaalta ylös -lähestymistapaa voidaan soveltaa liiketoimintapainotteisen tietovaraston suunnittelussa, mutta soveltuu ylhäältä alas -mallia paremmin sellaiseen tilanteeseen, jossa käyttökelpoisen prototyypin rakentaminen on liian hidasta tai kallista. On myös mahdollista, että rakennettava tietovarasto on niin laaja, että prototyypin liiketoimintavaatimusten lähtökohtana pitäminen on syystä tai toisesta vaikeaa. Tässä lähestymistavassa lähtökohtana toimivat käytössä olevat operatiiviset järjestelmät,

joiden dokumentaatiota apuna käyttäen pyritään tunnistamaan tietovarastoon tulevat faktat ja ulottuvuudet. Tietovaraston moniulotteisen käsitekaavion tuottamiseksi on olemassa useita menetelmiä. Tavallisimmin menetelmien alkuvaiheessa tunnistetaan operatiivisten järjestelmien käsitekaavioista faktat, ulottuvuudet ja hierarkiat, jonka jälkeen tuotetaan mahdollisten muiden vaiheiden jälkeen tietovaraston moniulotteinen malli. Moniulotteinen malli on tyypillisimmin tähtimalli, mutta muitakin malleja menetelmissä käytetään. Alhaalta ylös –lähestymistavan ongelmana on operatiivisissa järjestelmissä olevan tiedon siirtäminen tietovarastoon. Hasan ym. (2000) pitävät varsinkin tiedon puhdistusta haasteellisena tehtävänä. Lähestymistapa ei sovellu sellaiseen tilanteeseen, jossa on epävarmuutta siitä, mistä lähteestä ja millaista tietoa tietovaraston pitäisi sisältää.

Keskeltä ulos –lähestymistavan lähtökohtana ovat yrityksen liiketoimintatarpeet. Yritysjohdon avustuksella pyritään löytämään tietovaraston käyttäjien kannalta oleelliset faktat ja ulottuvuudet ja niihin tarvittavien tietojen lähteet. Tietolähteistä valitaan myös kaikki sellainen tieto, joka voi olla tietovaraston kannalta tärkeää ja pyydetään yritysjohtoa arvioimaan, mikä tiedoista on oleellista ja mikä ei. Tarpeettomiksi katsotut ja vaikeasti tietovarastoon poimittavissa olevat tiedot jätetään tietovarastosta pois. Seuraavassa vaiheessa rakennetaan tietovaraston prototyyppi yritysjohdon testattavaksi ja toteutetaan tarpeelliset muutokset. Tarvittaessa tietovaraston lähteenä käytettäviä tietoja muokataan sellaisiksi, että niiden poimiminen helpottuisi. Keskeltä ulos –lähestymistapa sopii parhaiten sellaisiin tilanteisiin, johon edelliset tavat ovat heikosti soveltuvia. Tällainen tilanne on esimerkiksi silloin, jos tietovarastoon tulevien tietojen lähteestä ja muodosta ei vielä suunnitteluvaiheessa ole varmuutta, ja tietovarasto on tarkoitus suunnitella palvelemaan monen erilaisen yrityksen tarpeita. Tällöin ylhäältä alas ja alhaalta ylös –lähestymistavat eivät ole hyviä vaihtoehtoja. Keskeltä ulos –lähestymistavassa ei keskitien kulkijana ole varsinaisia heikkouksia, vaan sitä voi käsittääkseni soveltaa kaikissa sellaisissa tilanteissa, jossa joko ylhäältä alas tai alhaalta ylös –lähestymistapaa on mahdollista soveltaa. Heikkoutena voidaan pitää sitä, että luovuttaessa vaikeasti saavutettavista tiedoista jo tietovaraston suunnitteluvaiheessa, voidaan päätyä tilanteeseen, jossa myöhemmin huomataan puuttuva tieto tarpeelliseksi. Suunnitteluvaiheessa voidaankin liian helposti

käyttää tietojen saavuttamisen vaikeutta syynä jonkin tiedon jättämiselle pois tietovarastosta.

Alhaalta ylös –lähestymistapaa soveltavissa menetelmissä tavallisesti oletetaan, että operatiivisten järjestelmien käsitekaavioista on tuotettu yksi yhteinen, globaali, käsitekaavio. Seuraavaksi tarkastellaan sen tuottamista.

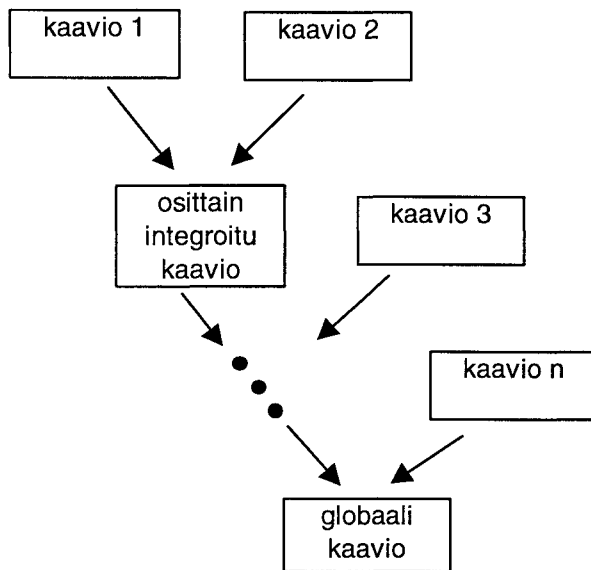
### **3.2 Käsitekaavioiden integrointi**

Otettaessa lähtökohdaksi operatiivisten järjestelmien tietokannat, törmätään monenlaisiin integrointiongelmiin. Näiden ongelmien ratkaisuksi voidaan käyttää menetelmiä, joita on kehitetty laajojen tietokantojen suunnitteluun. Kirjallisuudessa on esitetty menetelmiä (esim. Batini ym., 1992), joiden avulla useasta käsitekaaviosta on mahdollista tuottaa yksi yhteinen, globaali, käsitekaavio. Prosessia kutsutaan lähdeintegroinniksi (source integration) tai näkemysten integroinniksi (view integration). Tietovarastoa suunniteltaessa globaalista käsitekaaviosta on edelleen johdettava tietovarastoon sopiva käsitekaavio, jonka perustana voi olla esimerkiksi tähtimalli tai lumihiihtalemalli.

Tämän kohdan tarkastelu perustuu Batinin ym. (1992) esitykseen. He jakavat näkemysten integrointitapaukset laajoihin ja suppeisiin. Laajassa tarkastelussa integroitavia näkemyksiä on useita, kun taas suppeassa vain kaksi. Seuraavaksi esitettävän menetelmän avulla on mahdollista johtaa useasta käsitekaaviosta yksi yhteinen, globaali käsitekaavio. Menetelmässä ei kuitenkaan ole tarkoitus tuottaa tietovaraston moniulotteista käsitekaaviota, mutta globaalien käsitekaavien pohjalta esimerkiksi tähtimallin suunnittelu on mahdollista ja joka tapauksessa helpompaa kuin useiden erilaisten käsitekaavioiden pohjalta.



Ensin tarkastelun kohteena on tilanne, jossa integroitavia käsitekaavioita on useita, mahdollisesti jopa useita kymmeniä. Tällöin ei tavallisesti suoraan pyritäkään lopullisen globaalin käsitekaavion tuottamiseen, vaan tehdään niin sanottuja keskitason kaavioita integroimalla kaksi tai useampia käsitekaavioita kerrallaan. Lopuksi integroidaan keskitason kaaviot lopulliseksi globaaliksi kaavioksi. Tyypillisesti kaavioiden integraatio tapahtuu siten, että valitaan kerrallaan useampi toisiinsa jollain tavalla liittyvä kaavio integroitaviksi ja tuloksena saatavaan kaavioon integroidaan seuraavia kaavioita. Näin jatketaan, kunnes tuloksena on koko järjestelmän kattava globaali käsitekaavio. Batini ym. (1992) pitävät kuitenkin parempana tapana integroida kulloinkin vain kaksi kaaviota kerrallaan, koska ristiriitojen analysointi useammasta kaaviosta kerrallaan on hankalaa. Integroinnin kulku on esitetty kuviossa 6. Integrointi on syytä aloittaa keskeisimmistä käsitekaaviosta ja edetä vähemmän tärkeisiin käsitekaavioihin. Usean käsitekaavion integrointi siis palautuu yksityiskohtaisella tasolla tarkasteltuna kahden käsitekaavion integrointiin kerrallaan. Seuraavaksi esitelläänkin, miten kahden käsitekaavion integrointi tapahtuu.



KUVIO 6. Usean käsitekaavion integrointi globaaliksi käsitekaavioksi (Batini ym., 1992, 123).

Kahden käsitekaavion integrointi voidaan jakaa kolmeen tehtävään, jotka ovat nimiristiriitojen analyysi, rakenneristiriitojen analyysi ja näkymien yhdistäminen.

Nimiristiriitoja aiheuttavat synonyymit ja homonyymit. Synonyymiongelmassa saman tiedon kuvaamiseen on käytetty kaavioissa eri nimiä. Homonyymiongelmassa samaa nimeä on käytetty eri kaavioissa kuvaamaan eri tietoa. Nimiristiriidat voidaan ratkaista uudelleennimeämällä tai lisäämällä kaavioiden välisiä huomautuksia (interschema properties). Uudelleennimeämisessä jokin nimiristiriidan aiheuttavista elementeistä nimetään uudelleen. Esimerkiksi kohdetyypit asiakas ja tilaaja tarkoittavat eri kaavioissa samaa henkilöä eli on kyseessä synonyymiongelma. Ongelma ratkaistaan nimeämällä tilaaja asiakkaaksi tai asiakas tilaajaksi. Homonyymiongelmassa sama nimi voi tarkoittaa eri asioita. Esimerkiksi autojenvuokrausjärjestelmän kaavioissa rekisteröinti tarkoittaa toisessa kaaviossa auton vuokraajan rekisteröintiä järjestelmään, kun taas toisessa rekisteröinti tarkoittaa auton varaamista vuokrattavaksi tietyksi ajankohdaksi. Tämä homonyymiongelma ratkaistaan uudelleennimeämällä vuokravarauksia tarkoittava kohdetyyppi varaukseksi.

Kaavioiden välisten huomautusten lisääminen tulee kyseeseen, jos kaavioissa oleviin elementteihin kuuluu keskinäisiä rajoitteita. Esimerkiksi kohdetyyppi myyjä voi toisessa kaaviossa olla kohdetyypin työntekijä aliluokka. Tällöin asiasta on lisättävä huomautus, josta on hyötyä myöhemmässä vaiheessa.

Rakenneristiriitojen analyysissä oletetaan, että edellisvaiheessa kaavioissa on saavutettu nimiyhtenäisyys, jolloin samannimiset elementit kaavioissa tarkoittavat samaa reaali maailman asiaa. Rakenneristiriitojen analyysin avulla on tarkoitus selvittää, voidaanko kaavioissa olevia samannimisiä elementtejä integroida. Integroitavat elementit voidaan jakaa kolmeen eri kategoriaan: identtiset, yhteensopivat ja yhteensopimattomat elementit. Elementit ovat identtisiä, jos niiden rakenne ja ominaisuudet ovat täysin samanlaiset. Elementit ovat yhteensopivia, jos niiden rakenne tai ominaisuudet ovat hieman toisistaan poikkeavia, mutta eivät silti ristiriitaisia. Esimerkiksi sama elementti voidaan kuvata toisessa kaaviossa kohdetyyppinä ja toisessa attribuuttina. Tällöin ongelma ratkaistaan muuttamalla jompikumpi elementti toiseksi.

Yhteensopimattomilla elementeillä on ristiriitaisia ominaisuuksia. Yhteensopimattomuudet ER-kaavioissa on korjattava, ennen kuin kaavioita voidaan yhdistää. Ongelma voidaan ratkaista valitsemalla jompikumpi ristiriidassa olevista elementeistä ja käyttämällä jatkossa sitä tai luomalla uusi rakenne, joka ottaa huomioon molempien rakenteiden ominaisuudet.

Viimeisessä vaiheessa kaksi ER-kaaviota integroidaan siten, että syntyy yksi yhteinen ER-kaavio, globaali käsitekaavio. Tähän mennessä kaikki kaavioissa esiintyneet ristiriitaisuudet on ratkaistu ja kaavioiden integrointi on varsin suoraviivaista. Kohdetyypit, jotka vastaavat täysin toisiaan, siirretään sellaisenaan globaaliin kaavioon. Kohdetyypit, jotka vastaavat samaa reaali maailman käsitettä, mutta joilla on vain osittain samanlaiset attribuutit, siirretään globaaliin kaavioon siten, että mikään attribuuteista ei toistu. Tässä vaiheessa lisätään kaavioiden välisten huomautusten perusteella tarvittavat yleistysrakenteet. Kun kaikki kohdetyypit integroitavista kaavioista on siirretty globaaliin kaavioon, siirretään vielä suhdetyypit ja yleistysrakenteet.

Kaavioiden väliset huomautukset voivat johtaa myös globaalin kaavion muokkaamiseen elementtien lisäysten, poistojen tai uudelleenjärjestelyjen muodossa. Globaaliin kaavioon voidaan esimerkiksi lisätä käyttökelpoinen suhdetyyppi kahden toisiinsa liittyvän kohdetyypin välille.

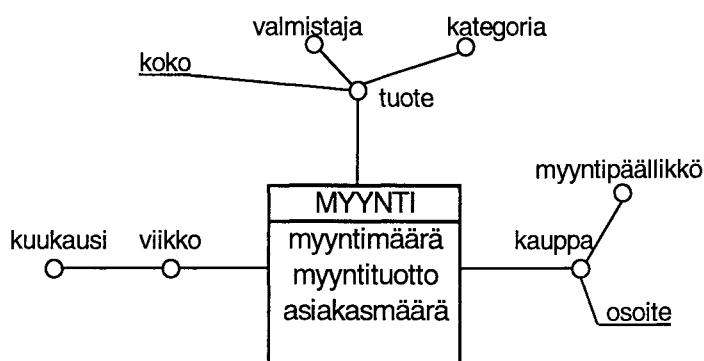
Kun integrointi on valmis, voidaan tuotettua globaalia kaaviota tarkastella ja pyrkiä etsimään siitä parannettavia kohtia. Kaaviota voidaan parannella tarvittaessa uudelleenjärjestelyn avulla.

### **3.3 Golfarellin, Maion ja Rizzin menetelmä**

Golfarellin ym. (1998) esittämä menetelmä on tietääkseni ensimmäinen, joka on esitetty tietovaraston käsitekaavion johtamiseksi operatiivisten järjestelmien käsitekaavioista.

## Malli

Golfarellin ym. (1998) mukaan tietovaraston käsitteellinen malli koostuu joukosta faktakaavioita, joiden osia ovat faktat, ulottuvuudet ja hierarkiat. Faktakaavio muistuttaa rakenteeltaan tietovaraston tähtimallia, mutta eroavaisuuksiakin on. Fakta on kaavion juuri, joka kuvataan laatikkona, jonka otsikkona on faktan nimi ja attribuutteina faktan attribuutit. Fakta on liitetty ulottuvuuksiin viivoilla. Ulottuvuudet kuvataan ympyröinä. Ulottuvuuksiin on edelleen viivoilla yhdistetty hierarkiat, jotka kuvataan myös ympyröinä. Ulottuvuuteen voi liittyä myös attribuutteja, joita ei voida käyttää yhteenvetotietojen tuottamisessa. Nämä attribuutit on kuvattu kaaviossa lyhyellä viivalla ympyrän sijasta. Esimerkkinä tällaisesta attribuutista voi olla kaupan osoite, joka on hyödyllistä tietää, mutta jota ei tarvita tietojen summaamisessa, koska kaupat erotellaan nimien tai tunnusten perusteella. Attribuutteja tai faktaa ja ulottuvuutta yhdistävän viivan leikkaavalla lyhyellä poikkiviivalla voidaan ilmaista kahden attribuutin tai faktan ja ulottuvuuden välistä valinnaista suhdetta. Valinnaisella suhteella tarkoitetaan sitä, että valinnainen attribuutti ei aina välttämättä esiinny sen faktan tai attribuutin yhteydessä, johon sillä on suhde. Jatkossa esimerkkitapauksena käytetään kauppaketjun tietovarastoa, jonka moniulotteinen faktakaavio on esitetty kuviossa 7.



KUVIO 7. Kauppaketjun tietovaraston moniulotteinen faktakaavio (Golfarelli ym., 1998).

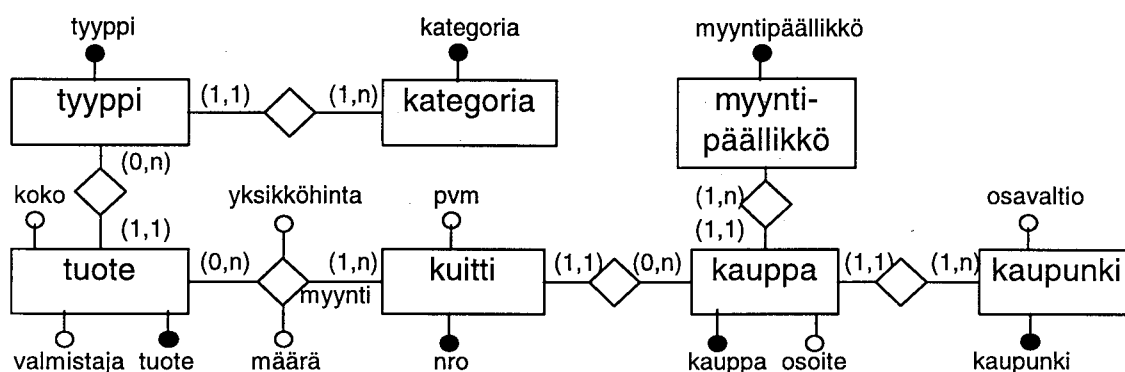
Esimerkiksi Hovin (2001, 96) esittämästä tähtimallista (ks. KUVIO 3) poiketen Golfarellin ym. mallin faktataulussa ei näy ulottuvuuksien perusavaimia. Lisäksi

ulottuvuuksia ei ole kuvattu laatikkoina eikä hierarkioita laatikon sisällä olevina riveinä kuten tyypillisessä tähtimallissa.

Faktan attribuutit ovat yleensä sellaisia, että niitä ja niihin liittyviä ulottuvuusattribuutteja voidaan summata, jolloin saadaan aikaan yhteenvetotietoja. Esimerkiksi, jos halutaan tietää tietylle myyntipäällikölle kuuluvat myynnit tietyltä aikaväliltä, on summaus välttämätöntä. Liian yksityiskohtaisesta tiedosta taas ei välttämättä ole etua päätöksenteossa. Siksi tietojen summaaminen onkin usein välttämätöntä. Faktan attribuuttia kutsutaan *puolisummattavaksi* (semi-additive), jos jotkut siihen liittyvät ulottuvuusattribuutit eivät ole summattavia. Usein jotain tasoa mittaavat attribuutit ovat puolisummattavia. Attribuuttia kutsutaan *ei-summattavaksi* (non-additive), jos mitään siihen liittyvistä ulottuvuusattribuuteista ei ole mahdollista summata. Puolisummattaville ja ei-summattaville attribuuteille on kuitenkin mahdollista tehdä muita kuin summaamisoperaatioita. Tyypillisiä muita operaatioita ovat esimerkiksi maksimi- ja minimiarvon etsiminen ja keskiarvon laskeminen. Esimerkkinä mainittakoon lämpötila, jolle voidaan laskea keskiarvo tai minimi- ja maksimiarvo, mutta pelkkä lämpötilojen yhteenlasku ei ole mielekäästä.

Faktassa ei välttämättä ole attribuutteja. Tällöin jokainen faktan ilmentymä tallentaa jonkin tapahtuman esiintymisen. Esimerkiksi fakta osanotto, jolla on ulottuvuudet päivämäärä, opiskelija, kurssi ja opettaja, voi tallentaa tiedon, josta selviää tiettyinä päivinä tietyn opiskelijan osallistuminen tietyn opettajan pitämälle kurssille.

Jatkossa esimerkkitapauksena käytetään kauppaketjun operatiivista järjestelmää, jonka käsitekaavio (KUVIO 8) muutetaan menetelmää käyttäen tietovaraston faktakaavioksi (KUVIO 7).



KUVIO 8. Kauppaketjun myyntitietoja tallentavan operatiivisen järjestelmän ER-kaavio (Golfarelli ym., 1998).

### Vaiheet

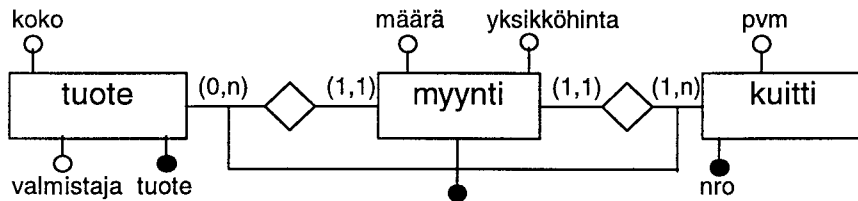
Golfarellin ym. (1998) menetelmä on jaettu kahteen päävaiheeseen. Ensin määritellään faktat ja toisessa vaiheessa jokaisen faktan kohdalla suoritetaan seuraavat tehtävät:

- attribuuttipuun rakentaminen
- attribuuttipuun karsiminen ja jalostaminen
- ulottuvuuksien määrittely
- fakta-attribuuttien määrittely
- hierarkioiden määrittely

Seuraavaksi käydään menetelmä läpi vaihe vaiheelta.

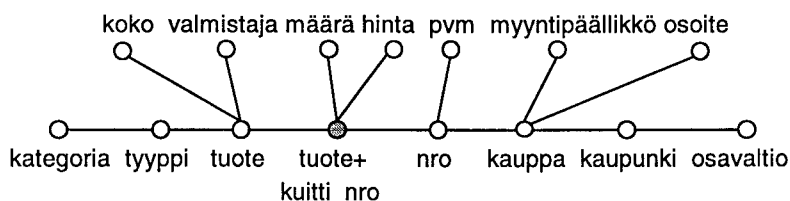
Ensiksi on määriteltävä faktat. Faktat ovat päätöksentekoprosessin kannalta tärkeitä kohteita. Operatiivisen järjestelmän ER-kaaviossa fakta on aina joko kohdetyyppi tai (n:m) -suhdetyyppi kahden tai useamman kohdetyypin välillä. Yksinkertaisuuden vuoksi on parasta muuntaa mahdolliset (n:m) -suhdetyypit kohdetyypeiksi. Tämä tapahtuu korvaamalla suhdetyyppi kohdetyypillä ja lisäämällä kohdetyypin jokaiselle puolelle uusi suhdetyyppi ja vaadittavat kardinaalisuudet. Esimerkkitapauksessa faktaksi sopii parhaiten suhdetyyppi myynti, joka on ensin muutettava kohdetyypiksi (KUVIO 9). Syntyvän myynti-kohdetyypin avainattribuutti koostuu tuotteen ja kuitin

avainattribuuteista. Kuviossa 8 tämä kuvataan viivana, joka kiertää myynti-kohdetyypin alla ja sivuilla. Usein päivitettäviä asioita kuvaavat kohde- tai suhdetyypit ovat tavallisesti mahdollisia faktoja. Rakennetta kuvaavat kohdetyypit taas eivät todennäköisesti ole faktoja. Jokainen fakta, joka ER-kaaviosta löydetään, asetetaan oman faktakaavionsa juureksi.



KUVIO 9. Suhdetyyppi myynti on muutettu kohdetyypiksi ja tarvittavat suhdetyypit ja kardinaalisuudet on lisätty kaavioon (Golfarelli ym., 1998).

Seuraavassa vaiheessa rakennetaan ensin attribuuttipuu. Lähtökohtana toimivat edellisvaiheessa löydettyjen faktojen avainattribuutit, joista tulee attribuuttipuiden juuria. Attribuuttipuuhun kuuluvat ER-kaaviosta kaikki sellaiset attribuutit, jotka avainattribuutti funktionaalisesti määrittelee. Attribuutit etsitään ER-kaaviosta ja attribuuttipuun lähtökohtana on kohdetyypin avainattribuutti. Attribuutit kuvataan ympyröillä ja niiden riippuvuudet viivoilla. Kohdetyypin avainattribuuttia kuvaava ympyrä on tummennettu. Esimerkkitapauksen ER-kaaviosta johdettu attribuuttipuu on esitetty kuviossa 10.



KUVIO 10. Kauppaketjun myyntitietoja kuvaavan faktakaavion pohjalta rakennettu attribuuttipuu (Golfarelli ym., 1998).

Seuraavana tehtävänä on attribuuttipuun karsiminen ja jalostaminen. Karsimisella tarkoitetaan sellaisten attribuuttien poistamista, jotka eivät ole tietovaraston kannalta oleellisia. Toinen syy karsimiselle voi olla tietovaraston liiallisen tarkkuuden vähentäminen. Karsiminen tapahtuu yksinkertaisesti poistamalla jokin tarpeeton osa attribuuttipuusta. Esimerkkitapauksessa kaupunkia ja osavaltiota ei tarvita, koska kaupat ryhmitellään myyntipäällikön mukaan. Näin ollen kaupunki ja osavaltio karsitaan pois attribuuttipuusta. Jalostaminen tulee kysymykseen silloin, kun attribuuttipuun poistettava osa on puun keskellä. Tällöin jalostamisen avulla osa puusta poistetaan ja poistetun osan takana olevat attribuutit yhdistellään uudelleen vastaamaan uutta tilannetta. Esimerkkitapauksessa tuotteet voidaan haluta ryhmitellä suoraan kategorian mukaan, eikä välissä olevan tyyppin mukaan. Tällaisessa tapauksessa attribuuttipuuta jalostetaan poistamalla attribuutti tyyppi ja yhdistämällä kategoria suoraan tuotteeseen. Myös nro-attribuutti, joka kuvaa kuitin numeroa, todetaan hyödyttömäksi ja se karsitaan pois. Kuitenkin nro-attribuuttiin liittyvä päivämäärä-attribuutti on kiinnostava ja se liitetään suoraan attribuuttipuun juuren attribuutiksi.

Seuraavana tehtävänä on ulottuvuuksien määrittely. Ulottuvuudet määrittelevät, millaisia koosteita tietovaraston tiedoista on mahdollista tehdä. Ulottuvuuksiksi kelpaavat sellaiset attribuutit, joilla on suora yhteys attribuuttipuun juureen. Tietovarastossa on tärkeä rooli ajalla ja se onkin lähes poikkeuksetta yksi ulottuvuus. Attribuuttipuun karsimisvaiheessa on tärkeää ottaa huomioon se, että aika-attribuutista tulee suora yhteys attribuuttipuun juureen. Jos käsitekaaviossa ei ole lainkaan aika-attribuutteja, lisätään aikaa varten oma ulottuvuutensa tässä vaiheessa. Tehtävän lopuksi valitut ulottuvuudet voidaan lisätä kuhunkin faktakaavioon juuren lisäksi. Esimerkkitapauksessa ulottuvuuksiksi valitaan tuote, kauppa ja aikaulottuvuudeksi päivämäärä viikoiksi muutettuna. Tulos on nähtävissä kuviossa 7.

Seuraavana tehtävä on fakta-attribuuttien eli mittauskohteiden määrittely. Ne ovat tyyppillisesti kohdetyypin ilmentymiä kuvaavia attribuutteja tai attribuutteja, jotka kohdistavat muita yleisimpiä operaatioita, kuten minimiarvon tai maksimiarvon etsimisen tai keskiarvon laskemisen, attribuuttipuussa oleviin attribuutteihin. Faktalla ei



ole attribuutteja, jos sen ainoa tarkoitus on tallentaa tiedot faktan esiintymisistä. Esimerkkitapauksessa fakta-attribuuteiksi valitaan myyntimäärä, myyntituotto ja asiakasmäärä. Fakta-attribuutit lisätään vastaaviin faktakaavioihin. Tulos on nähtävissä kuviossa 7. Tässä vaiheessa on myös järkevää määrittää metatietoa, jossa kerrotaan, miten kukin fakta-attribuutti on johdettu operatiivisen järjestelmän ER-kaaviosta. Esimerkkitapauksen fakta-attribuutti myyntimäärä on myynnin määrien summa ( $\text{SUM}(\text{myynti.määrä})$ ), myyntituotto on myynnin määrän ja yksikköhinnan tulojen summa ( $\text{SUM}(\text{myynti.määrä} * \text{myynti.yksikköhinta})$ ) ja asiakasmäärä on myyntien lukumäärä ( $\text{COUNT}(\text{myynti})$ ).

Viimeisenä tehtävänä on ulottuvuuksien hierarkioiden määrittely. Hierarkiat voidaan tunnistaa attribuuttipuusta. Tunnistaminen tapahtuu varsin automaattisesti. Lähtökohtana toimivat aiemmassa vaiheessa määritetyt ulottuvuudet. Ulottuvuuksiin yhdistyvät attribuutit ja niihin edelleen mahdollisesti yhdistyvät attribuutit kaavion loppuun asti ovat kuhunkin ulottuvuuteen kuuluvia hierarkioita. Aikaulottuvuuteen voidaan tarvittaessa lisätä uusia attribuutteja. Tavallisesti tietovarastosta halutaan tietoa päivä-, viikko-, kuukausi- ja vuositasolla. Jokaista ajanjaksoa varten on oltava oma attribuutti aikaulottuvuudessa. Esimerkkitapauksessa aikaulottuvuuden hierarkiaan lisätään kuukausi (ks. KUVIO 7).

Golfarellin ym. (1998) menetelmän käytön tuloksena saadaan tietovaraston käsitteellinen malli, joka koostuu joukosta faktakaavioita. Edellä on esitetty esimerkkitapauksesta yksi faktakaavio ja tuon kaavion tuottamisen vaiheet.

### **3.4 Cabibbon ja Torlonen menetelmä**

Cabibbon ja Torlonen (1998) menetelmän avulla operatiivisen järjestelmän käsitekaavio voidaan muuttaa tietovaraston moniulotteiseksi malliksi. Menetelmän käyttö edellyttää, että operatiivisten järjestelmien käsitekaavioista on tuotettu globaali käsitekaavio ER-

notaatiota käyttäen. ER-kaavion tulee sisältää liiketoiminnan kannalta kaiken oleellisen tiedon operatiivisista järjestelmistä. Kaavio ei saa sisältää yleistysrakenteita, ja kaavion attribuuttien on oltava atomisia. Kaavion täytyy olla myös normalisoitu, minimaalinen ja täydellinen. Minimaalisuudella tarkoitetaan sitä, että mikään tieto ei kaaviossa toistu. Täydellisyydellä tarkoitetaan sitä, että kaavio sisältää kaiken tiedon, joka on mahdollista saada operatiivisista järjestelmistä tietovaraston käyttöön.

Tässä kohdassa ei ole tarpeellista esitellä menetelmässä käytettyä mallia, koska lähtökohtana toimii perinteinen ER-kaavio ja lopputuloksena saadaan valinnan mukaan jokin tietovaraston moniulotteinen malli, esimerkiksi tähtimalli (ks. KUVIO 3), joka on esitelty jo aiemmin.

### **Vaiheet**

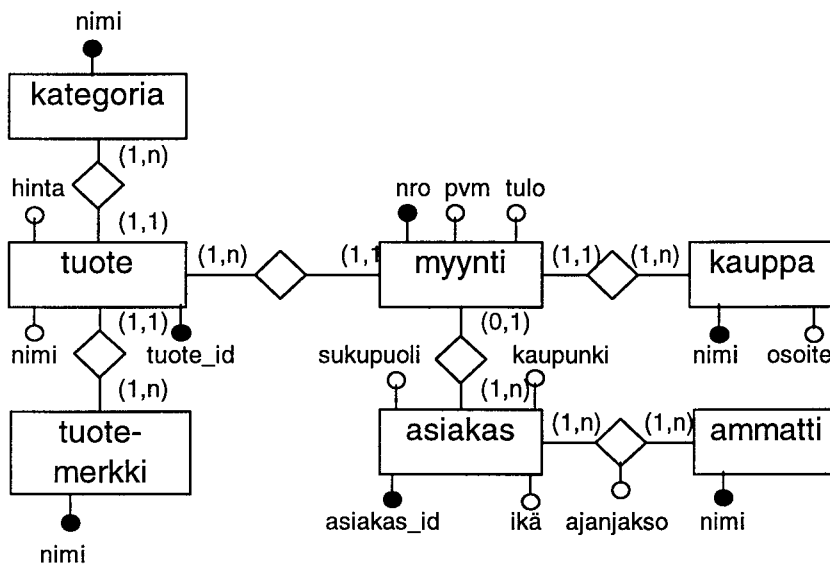
Tietovaraston moniulotteisen mallin johtaminen operatiivisten järjestelmien globaalista ER-kaaviosta tapahtuu neljässä vaiheessa:

- faktojen, mittauskohteiden ja ulottuvuuksien tunnistaminen
- käsitekaavion uudelleenjärjestely
- ulottuvuusgraafin (dimensional graph) johtaminen
- muuntaminen moniulotteiseksi malliksi

Usein kahden ensimmäisen vaiheen suorittaminen tapahtuu samanaikaisesti siten, että käsitekaaviota uudelleenjärjesteltäessä myös faktoja ja ulottuvuuksia muutellaan.

Ensimmäisessä vaiheessa pyritään tunnistamaan operatiivisista järjestelmistä tuotetusta globaalista ER-kaaviosta faktat, mittauskohteet ja ulottuvuudet. Faktat ovat ER-kaaviossa kohde- tai suhdetyyppejä tai attribuutteja, joihin päätöksentekoprosessi kohdistuu. Mittauskohde on faktan atominen attribuutti, jota halutaan analysoida. Yleensä mittauskohde on faktan numeerinen attribuutti tai attribuutti, joka ilmaisee faktojen ilmentymien lukumäärän. Ulottuvuus on ER-kaavion osa, joka kuvaa analysoitavan faktan tietoja jostakin päätöksenteon kannalta kiinnostavasta näkökulmasta.

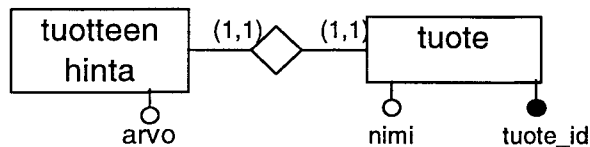
Esimerkkitapauksen kauppaketjua kuvaavasta ER-kaaviosta (KUVIO 11) faktoiksi kelpaavat kohdetyyppi myynti ja attribuutti tuotteen hinta. Myynnin mittauskohteina ovat myyntien lukumäärä ja myyntitulot. Tuotteen hinnan mittauskohteena on hinta itse. Ulottuvuudet löytyvät parhaiten pitämällä lähtökohtina valittuja faktoja ja etsimällä sellaisia faktoihin liittyviä kohdetyyppejä, jotka kuvaavat jotain tapaa tietojen ryhmittelemiseksi. Myyntejä voidaan ryhmitellä myytyjen tuotteiden, tuotenimien ja tuotekategorioiden mukaan. Lisäksi ryhmittelyä voidaan tehdä asiakkaiden iän, sukupuolen ja asuinpaikan mukaan sekä eri kaupoissa tapahtuneiden myyntien mukaan. Aikaulottuvuus on myös oleellinen ryhmittelytekijä tietovaraston yhteydessä.



KUVIO 11. Kauppaketjua kuvaavan operatiivisen järjestelmän ER-kaavio (Cabibbo ja Torlone, 1998).

Seuraavassa vaiheessa ER-kaaviota uudelleenjärjestellään siten, että se kuvaisi paremmin ja tarkemmin faktoja ja ulottuvuuksia. Vaiheen tuloksena syntyy uusi ER-kaavio, joka on helpompi muuttaa tietovaraston moniulotteiseksi malliksi.

Uudelleenjärjestely alkaa sillä, että kaikki edellisessä vaiheessa valitut faktat muutetaan ER-kaaviossa kohdetyypeiksi. Faktaksi valittu attribuutti korvataan kohdetyypillä ja lisätään kohdetyyppien väliin (1:1) -suhdetyyppi. Faktaksi valittu suhdetyyppi korvataan kohdetyypillä, jonka molemmille puolille lisätään suhdetyyppi ja tarvittavat kardinaalisuudet. Mikäli faktoiksi on valittu ainoastaan kohdetyyppejä, edellä mainittuja tehtäviä ei suoriteta. Esimerkkitapauksessa faktaksi on valittu tuotteen hinta, joka on ER-kaaviossa attribuuttina. Attribuutti muutetaan kohdetyypiksi kuten kuviossa 12 on esitetty. Tuotteen hinnasta tulee heikko kohdetyyppi, joka voidaan tunnistaa tuote-kohdetyypin avulla.



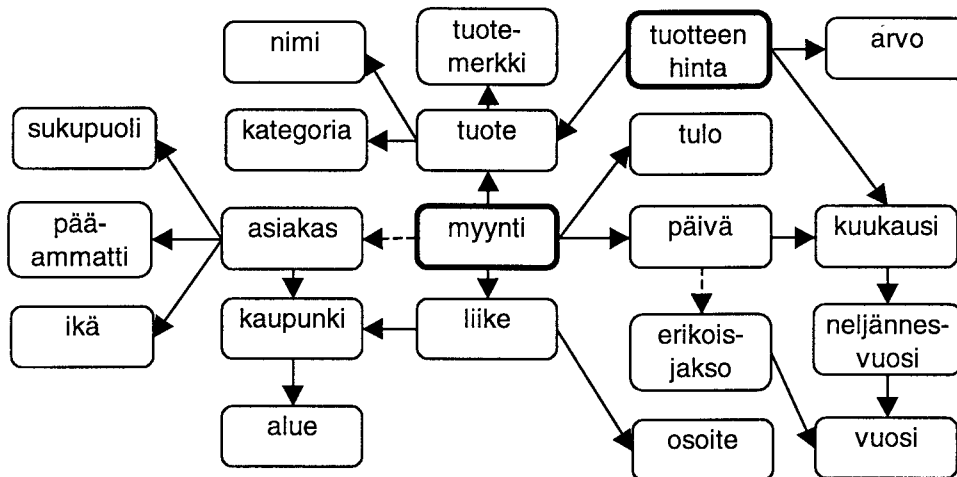
KUVIO 12. Attribuutti tuotteen hinta on muutettu ER-kaavion uudelleenjärjestelyssä kohdetyypiksi (Cabibbo ja Torlone, 1998).

Uudelleenjärjestelyn toisessa vaiheessa tarkastellaan, puuttuuko ER-kaaviosta joitakin ulottuvuuksissa tarvittavia kohdetyyppejä ja lisätään kaavioon tarvittaessa uusia kohdetyyppejä kaikkien tarvittavien ulottuvuuksien aikaansaamiseksi. Tarvittavat ulottuvuudet voivat koskea esimerkiksi historiatietoa tai jonkin tiedon maantieteellistä alkuperää. Lisättävien kohdetyyppien sisältämä tieto voidaan saada esimerkiksi ulkopuolisesta tietokannasta tai operatiivisiin järjestelmiin liittyvästä metatiedosta. Lisättävää ulottuvuutta varten ER-kaavioon lisätään tarvittavat kohdetyypit ja suhdetyypit tiedon esittämiseksi. Esimerkkitapauksessa halutaan mahdollistaa tuotteen hinnan muutos kerran kuukaudessa siten, että myös aiempien kuukausien hinnat ovat saatavilla analysointia varten. Tätä tarkoitusta varten kohdetyyppiin tuotteen hinta liitetään (1:n) -suhteella uusi kohdetyyppi kuukausi.

Uudelleenjärjestelyn viimeisessä vaiheessa tarkennetaan ER-kaaviosta löytyvien ulottuvuustasojen tietoja. Tässä vaiheessa on oltava selvillä, mistä eri näkökulmista halutaan tietoja faktoista, ja sen mukaan tehdään tarvittavia muutoksia ER-kaavioon. Muutos voi olla käytännössä (m:n) –suhdetyyppin korvaaminen kohdetyypillä, uusien kohdetyyppien tai attribuuttien lisääminen uusien ulottuvuuksien tai niiden osien esittämiseksi, tarpeettomien kohdetyyppien tai attribuuttien poistaminen ja yksilöivän tunnuksen valinta jokaiselle kaavioon lisätylle kohdetyypille. Esimerkkitapauksessa asiakkaita voidaan ryhmitellä iän, sukupuolen ja asuinpaikan mukaan, mutta lisäksi haluttaisiin mahdollisuus ryhmittelyyn myös heidän ammattiensa perusteella. Tällöin kohdetyyppien asiakas ja ammatti välillä oleva (m:n) -suhdetyyppi on muutettava (1:n) –suhdetyypiksi, jolloin kohdetyppi ammatti kertoo vain asiakkaan pääammatin. Samalla kohdetyypin nimi voidaan muuttaa pääammatiksi. Liikeulottuvuutta halutaan ryhmitellä kaupungin ja maantieteellisen alueen mukaan, joten ER-kaavioon lisätään kaksi kohdetyyppeä, kaupunki ja alue ja niille annetaan yksilöivät tunnuksset. Liikkeen ja kaupungin välille tulee (1:n) –suhdetyyppi, samoin kaupungin ja alueen välille. ER-kaaviosta puuttuu kokonaan aikaulottuvuus, jota tarvitaan tietovarastossa. Myyntejä halutaan ryhmitellä päivien, kuukausien, neljännesvuosien, vuosien ja erikoiskausien mukaan. Erikoiskausi voi olla esimerkiksi joulukuusi tai pääsiäinen. Jokaista ajanjaksoa varten lisätään ER-kaavioon kohdetyppi ja tarvittavat (1:n) –suhdetyypit kohdetyyppien välille.

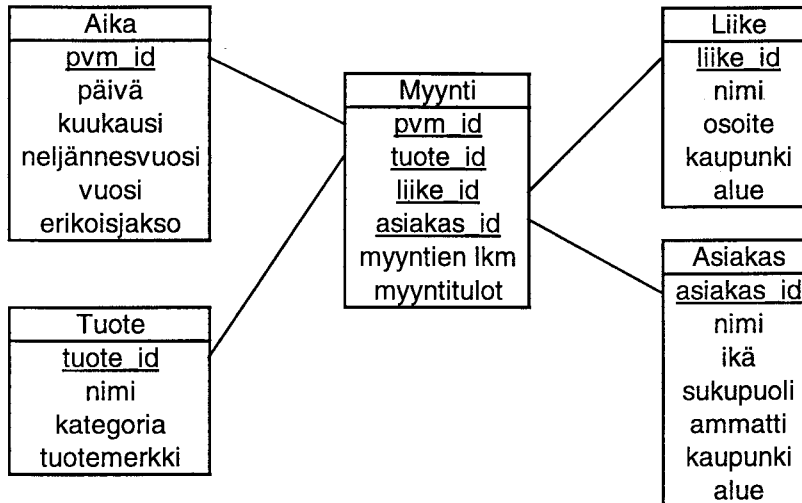
Seuraavassa vaiheessa uudelleenjärjestelystä ER-kaaviosta johdetaan ulottuvuusgraafi. Graafi esittää suppeasti kaikki uudelleenjärjestetyn ER-kaavion faktat ja ulottuvuudet. Graafin jokainen solmu vastaa ER-kaaviossa joko kohdetyyppeä tai attribuuttia. Solmut on kuvattu nurkista pyöristetyillä suorakaiteilla ja niiden väliset yhteydet on kuvattu nuolilla. Nuoli on piirretty katkoviivoin, jos solmujen välinen yhteys on osittainen. Esimerkkitapauksessa (KUVIO 13) päivän ja erikoisjakson välillä oleva yhteys on osittainen, koska suurin osa päivistä ei kuulu mihinkään erikoisjaksoon. Graafissa on neljänlaisia solmuja. Faktasolmut vastaavat ER-kaavion faktoja ja ne on ympäröity lihavoidulla viivalla. Tasosolmut vastaavat ER-kaavion kohdetyyppejä, jotka muodostavat ulottuvuuksia. Kuvaavat solmut yhdistyvät tasosolmuihin, joten ne vastaavat ER-kaaviossa kuvailevia attribuutteja. Mittauskohdesolmut yhdistyvät

faktoihin, joten ne vastaavat ER-kaaviosta valittuja mittauskohteita. Esimerkkitapauksessa syntyvä ulottuvuusgraafi on esitetty kuviossa 13.



KUVIO 13. Kauppaketjun operatiivista järjestelmää kuvaavasta ER-kaaviosta johdettu ulottuvuusgraafi (Cabibbo ja Torlone, 1998).

Menetelmän viimeisessä vaiheessa ulottuvuusgraafi muunnetaan tietovaraston moniulotteiseksi malliksi. Ulottuvuusgraafin faktasolmuista tulee faktatauluja. Esimerkkitapauksessa faktatauluiksi tulevat tuotteen hinta ja myynti. Tuotteen hinnan ainoa mittauskohde on tuotteen hinta itsessään. Myynnin mittauskohteita ovat myyntien lukumäärä ja myyntitulot. Ulottuvuusgraafin tasosolmuista rakennetaan tietovaraston moniulotteisen mallin ulottuvuustaulut. Esimerkkitapauksessa ulottuvuuksia on neljä: Asiakas, Tuote, Liike ja Aika. Asiakasulottuvuuden tasoja ovat nimi, ikä, sukupuoli, ammatti, kaupunki, alue. Tuoteulottuvuuden tasoja ovat nimi, kategoria ja tuotemerkki. Liikeulottuvuuden tasoja ovat nimi, osoite, kaupunki ja alue. Aikaulottuvuuden tasoja ovat päivä, kuukausi, neljännesvuosi, erikoisjakso ja vuosi. Myynnistä tuotettu moniulotteinen malli on esitetty kuviossa 14.



KUVIO 14. Ulottuvuusgraafi muunnettuna tietovaraston moniulotteiseksi malliksi, jossa faktatauluna on Myynti.

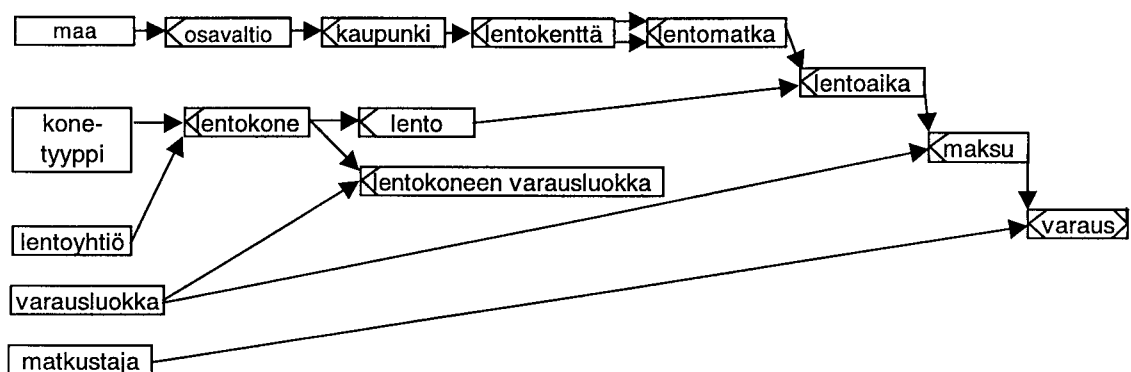
### 3.5 Böhnleinin ja Ulbrich-vom Enden menetelmä

Böhnleinin ja Ulbrich-vom Enden (1999) menetelmän lähtökohtana ovat operatiivisten järjestelmien käsitekaaviot. Menetelmän käyttö edellyttää, että operatiivisten järjestelmien käsitekaavioista on johdettu globaali käsitekaavio, joka on tehty SER-notaatiota (Structured Entity Relationship) käyttäen. Muilla notaatioilla tehdyt käsitekaaviot täytyy muuttaa SER-kaavioiksi, ennen kuin menetelmää voidaan käyttää. Gross (1992) on esittänyt ohjeet ER-kaavion muuttamiseksi SER-kaavioksi. Noihin ohjeisiin ei tässä tutkielmassa perehdytä. Aluksi kuvataan lyhyesti SER-mallia ja sen eroja ER-malliin verrattuna, koska SER-mallin ymmärtäminen on jatkoon kannalta tärkeää. Sen jälkeen esitellään menetelmän vaiheet.

#### Malli

SER-mallin kuvaus perustuu Böhnleinin ja Ulbrich-vom Enden (1999) artikkeliin. (ks. tarkemmin Sinz, 1988). Chenin (1976) ER-malliin perustuva SER-malli koostuu tieto-

objektityypeistä (data object types). Tieto-objektityyppi voi olla kohdetyyppi, suhdetyyppi tai kohde-suhdetyyppi. Kaikki riippumattomat tieto-objektityypit eli kohdetypit (entity type) sijoitetaan kaavion vasempaan laitaan, ja kaaviota voidaan siten lukea loogisesti vasemmalta oikealle. Tieto-objektityyppi on riippumaton, jos se ei ole riippuvainen mistään muusta tieto-objektityypistä. Esimerkkitapauksessa muun muassa matkustaja ja maa ovat riippumattomia tieto-objektityyppejä. Kohdetypit tunnistaa kaaviosta lisäksi siitä, että niitä kuvaavien laatikoiden vasemmassa laidassa ei ole vinoviivoja. Kaavion oikeaan reunaan sijoitettavien suhdetyyppien (relationship type) laatikoissa vinoviivat ovat sekä laatikoiden vasemmassa että oikeassa laidassa. Tieto-objektityyppi on suhdetyyppi, jos se on riippuvainen muista tieto-objektityypeistä, mutta mikään tieto-objektityyppi ei ole riippuvainen siitä. Suhdetyyppien lisäksi riippuvia tieto-objektityyppejä ovat kohde-suhdetypit (entity relationship type), jotka ovat kohdetyyppien ja suhdetyyppien välimuotoja. Ne ovat toisaalta riippuvia muista tieto-objektityypeistä, mutta myös muut tieto-objektityypit voivat olla riippuvia kohde-suhdetyypeistä. Riippuvuudet SER-kaaviossa olevien tieto-objektityyppien välillä on osoitettu viivoilla ja nuolilla. Yksinkertainen viiva tarkoittaa (0:1) -suhdetta ja kaksinkertainen viiva (1:1) -suhdetta. Tavallinen nuoli tarkoittaa (0:n) -suhdetta ja kaksinkertaisin viivoin piirretty nuoli (1:n) -suhdetta. Esimerkkitapauksen SER-kaaviossa (KUVIO 15) esiintyy ainoastaan (0:n) -suhteita kuvaavia tavallisia nuolia.



KUVIO 15. Lentojenvarausjärjestelmän SER-kaavio Böhnleinin ja Ulbrich-vom Enden (1999) esimerkin mukaan.



Böhnlein ja Ulbrich-vom Ende luettelevat seuraavia SER-mallin vahvuuksia ER-malliin nähden:

- Laajojen kaavioiden suunnittelu on helpompaa, koska SER-kaavio voidaan rakentaa aloittaen toisesta reunasta ja etenemällä järjestyksessä toista reunaa kohti.
- Tieto-objektityyppien välisten riippuvuuksien mallintaminen on selkeämpää, koska SER-malli perustuu riippuvuuksien esittämiseen.
- Epäjohdonmukaisuudet voidaan välttää. Esimerkiksi silmukan mallintaminen SER-kaaviossa ei ole hyväksyttävää.
- Tarpeettomien suhdetyyppien välttäminen helpottaa jatkossa esimerkiksi tietovaraston käsitekaavion johtamista, koska SER-kaavion ei yleensä ole tarvetta tehdä rakenteellisia muutoksia.

### **Vaiheet**

Tietovaraston käsitekaavion johtaminen operatiivisten järjestelmien käsitekaavioista on jaettu kolmeen vaiheeseen:

- mitattavien liiketoimintakohteiden tunnistaminen
- ulottuvuuksien ja ulottuvuushierarkioiden tunnistaminen
- ulottuvuushierarkioiden rajoitteiden tunnistaminen

Böhnlein ja Ulbrich-vom Ende (1999) käyttävät artikkelissaan esimerkkinä lentoyhtiön lentojenvarausjärjestelmää (KUVIO 15), jonka pohjalta on tarkoitus rakentaa tietovarasto tehostamaan lentojen varaamista ja sitä kautta tuottamaan yhtiölle enemmän voittoa. Seuraavaksi kuvataan menetelmä vaihe vaiheelta.

Ensimmäisessä vaiheessa liiketoimintakohteet pyritään tunnistamaan seuraavan päättelyketjun avulla: päämäärät – palvelut – mitattavat kohteet. Esimerkkitapauksessa lentoyhtiön päämääränä on kasvattaa liikevaihtoa tietovaraston käytön avulla. Päämäärään johtava palvelu on lentoyhtiön lennonvarausjärjestelmä, joka ilmaisee samalla mitattavan liiketoimintakohteen. Kohteen mittaamiseen voidaan käyttää varausten määrää, joka on ensimmäinen mittauskohde. Lisäämällä varausten määrään tieto lentokuljetusten hinnoista saadaan toinen mittauskohde, joka on varausten

liikevaihto. Lentojenvarausjärjestelmän SER-kaaviossa mittauskohteiden mittaamista varten tarvitaan kaksi tieto-objektityyppiä: varaus ja maksu. Mittauskohteet muodostavat tietovaraston moniulotteisen käsittekaavion perustan.

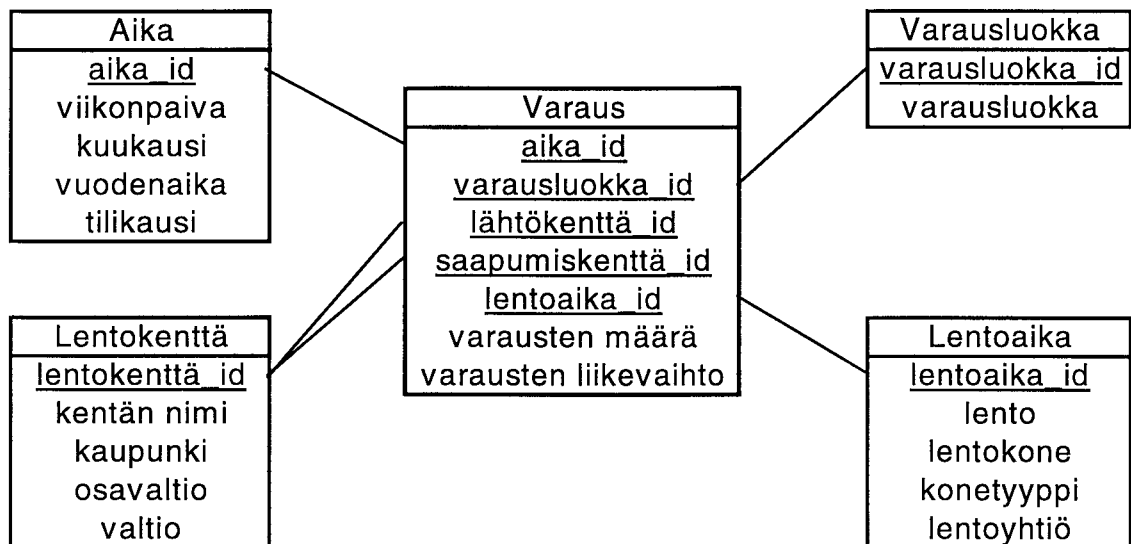
Toisessa vaiheessa pyritään tunnistamaan ulottuvuudet ja ulottuvuushierarkiat. Tunnistamista helpottavat SER-kaaviossa näkyvät riippuvuudet tieto-objektityyppien välillä. Tunnistamista ajatellen on ensin määriteltävä olemassaoloehdot (the closure of existency prerequisites) edellisessä vaiheessa määrätyle mittaaukohteille. Lähtökohtana toimii SER-mallin se tieto-objektityyppi, josta mittauskohde muodostettiin (esimerkkitaauksessa tieto-objektityypit varaus ja maksu). SER-kaaviota tarkastellaan tieto-objektityypeistä katsoen vasemmalle ja tutkitaan, mistä tieto-objektityypeistä lähtökohtana toimivat tieto-objektityypit riippuvat. Riippuvuusuhteet jatkuvat kaaviossa siihen asti, kunnes tulee vastaan riippumaton tieto-objektityyppi aivan kaavion vasemmassa reunassa. Esimerkkitaauksessa tieto-objektityypit varaus ja maksu riippuvat seuraavista tieto-objektityypeistä: matkustaja, lentoaika, varausluokka, lentomatka, lento, lentokenttä, lentokone, kaupunki, konetyyppi, lentoyhtiö, osavaltio ja valtio. Olemassaoloehtojen ulkopuolelle tieto-objektityypeistä jää lentokoneen varausluokka.

Ulottuvuuksien tunnistaminen on toisen vaiheen seuraava tehtävä, jossa tarvitaan luovuutta ja tietämystä rakennettavan järjestelmän kohdealueesta. Ulottuvuuksien tunnistaminen tapahtuu tutkimalla SER-kaaviota mittauskohteista katsoen vasemmalle pyrkien löytämään sellaiset tieto-objektityypit, jotka ovat tietovaraston tietojen analysoinnin kannalta merkittäviä. Esimerkkitaauksessa ensimmäisenä löytyvä ulottuvuus on varausluokka. Se sisältää oleellista tietoa varausten määrästä ja varausten liikevaihdosta. Toiseksi ulottuvuudeksi valitaan lentokenttä. Se vastaa tietovaraston kaaviossa sekä lennon lähtö- että saapumiskenttää, siksi tieto-objektityyppi lentomatka voidaan jättää ulottuvuuksista kokonaan pois. SER-kaaviossa ne kohdetyypit, joihin lentokenttä-kohdetyypillä on riippuvuusuhde, muodostavat ulottuvuushierarkian ja tulevat niin ollen lentokenttä-ulottuvuustaulun attribuuteiksi. Mahdollisen ulottuvuushierarkian tunnistaa  $(0,n)$  -suhteista, joita kuvaavat SER-kaaviossa yksinkertaiset nuolet. Kolmanneksi ulottuvuudeksi valitaan lentoaika, jonka

attribuuteiksi tulevat lento, lentokone, konetyyppi ja lentoyhtiö. Tietovarastossa tärkeä ulottuvuus on aika, ja sitä ei yleensä löydy SER-kaaviosta. Siispä aikaulottuvuus lisätään yhdeksi ulottuvuudeksi ja sen attribuuteiksi viikonpäivä, kuukausi, vuodenaika (neljännesvuosi) ja vuosi. Jokaisen ulottuvuuden perusavaimena on ID-tyyppinen attribuutti, josta kohde on yksiselitteisesti tunnistettavissa.

Viimeisessä vaiheessa pyritään tunnistamaan ulottuvuushierarkioiden rajoitteita. Rajoitteilla tarkoitetaan summaamissääntöjä, esimerkiksi mille attribuuteille sallitaan yhteenlaskeminen ja mille pelkästään suurimman tai pienimmän arvon etsiminen. Rajoitteita ei kuitenkaan pystytä esittämään tavallisissa tähti- tai lumihiihtalemalleissa, vaan niistä on tehtävä erillinen dokumentti.

Lopputuloksena syntyy tietovarastoa kuvaava tähtimalli (KUVIO 16), jonka faktataulun attribuutteina ovat ulottuvuustaulujen perusavaimet sekä mitattavat kohteet eli varausten määrä ja varausten liikevaihto. Tähtimallista voidaan tarvittaessa tuottaa lumihiihtalemallei denormalisoimalla ulottuvuustauluja.



KUVIO 16. Lentojenvarausjärjestelmän tietovaraston moniulotteinen tähtimalli (Böhnlein ja Ulbrich-vom Ende, 1999).

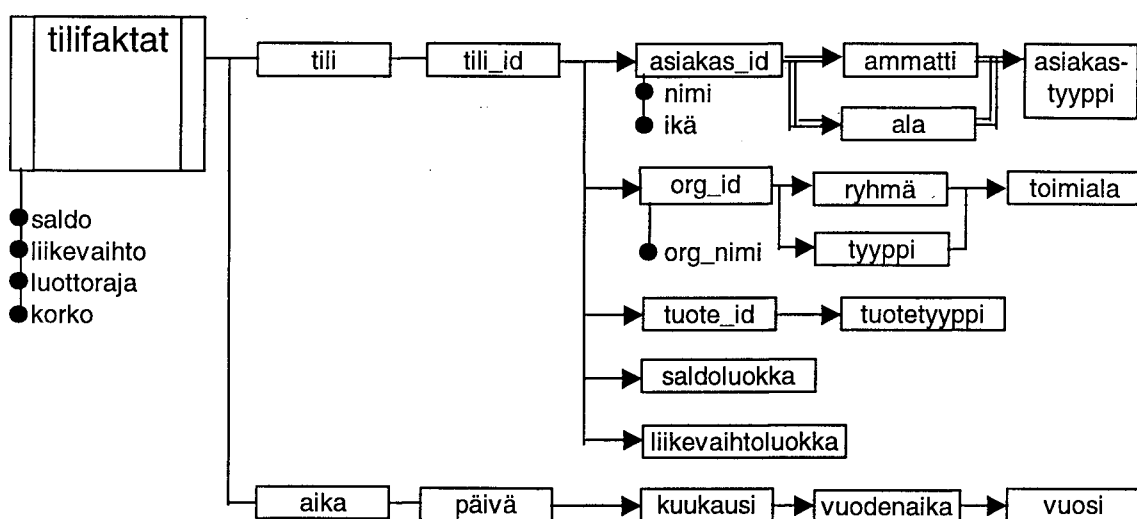
### 3.6 Hüsemanın, Lechtenbörgerin ja Vossenin menetelmä

Hüsemann ym. (2000) ovat esittäneet tietovaraston suunnittelulle vaihejaon, joka koostuu neljästä vaiheesta. Vaiheet ovat vaatimusmäärittely, käsitteellinen suunnittelu, looginen suunnittelu ja fyysinen suunnittelu. Tietovaraston käsittekaavion johtaminen operatiivisten järjestelmien globaalista käsittekaaviosta tapahtuu käsitteellisen suunnittelun vaiheessa, mutta sitä edeltävä vaatimusmäärittely on myös oleellinen osa kaavion tuottamista. Siksi vaatimusmäärittely kuvataan jatkossa menetelmän ensimmäiseksi vaiheeksi. Edellytyksenä käsitteelliselle suunnittelulle on, että operatiivisten järjestelmien käsittekaavioiden pohjalta on tuotettu globaali käsittekaavio ER-notaatiota käyttäen.

#### Malli

Menetelmän avulla voidaan tuottaa yleistetyssä moniulotteisessa normaalimuodossa oleva tietovaraston käsittekaavio. Jatkossa esimerkkitapauksena käytettävän tietovaraston käsittekaavio on esitetty kuviossa 17. Lehnerin ym. (1998) mukaan moniulotteinen käsittekaavio on yleistetyssä moniulotteisessa normaalimuodossa, kun seuraavat ehdot täyttyvät:

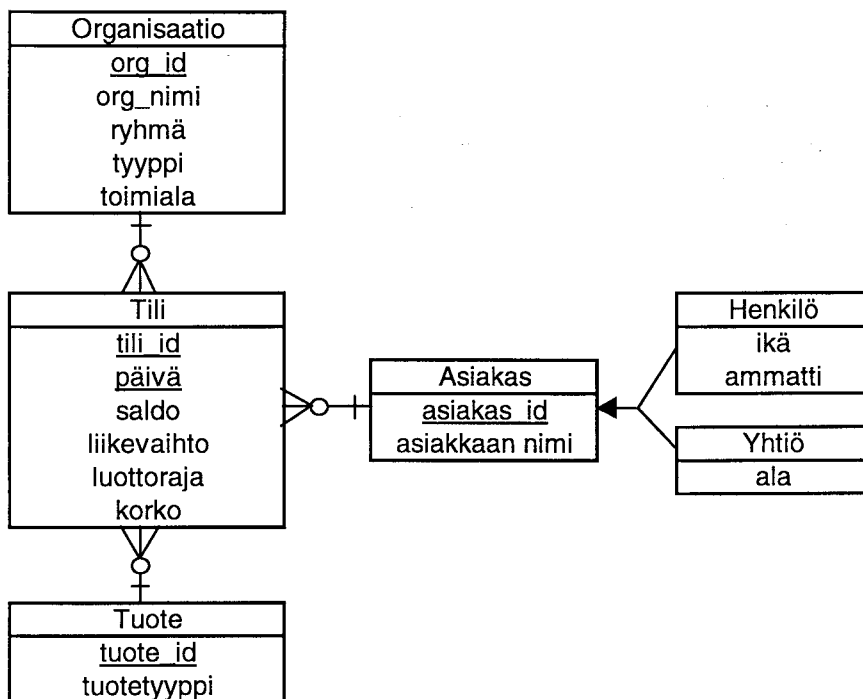
- jokainen ulottuvuus on ulottuvuusnormaalimuodossa
- ulottuvuuksien omien attribuuttien tai eri ulottuvuuksien attribuuttien välillä ei ole heikkoja funktionaalisia riippuvuuksia
- jokaista ominaisuusattribuuttia kohti on ulottuvuusattribuutti, josta ominaisuusattribuutti on riippuvainen
- ulottuvuuksien attribuutit määrittelevät funktionaalisesti kaikki faktan mittauskohteet
- joka ulottuvuudessa on vain yksi terminaalattribuutti eli sellainen attribuutti, joka ei ole riippuvainen mistään toisesta attribuutista
- attribuuttien arvot ovat täydellisiä.



KUVIO 17. Tietovaraston faktakaavio yleistetyssä moniulotteisessa normaalimuodossa (Hüsemann ym., 2000).

Tietovaraston käsitekaavio koostuu useasta faktakaaviosta, jotka koostuvat faktoista sekä niihin liittyvistä mittauskohteista ja ulottuvuuksista. Ulottuvuus koostuu tasoista, jotka vastaavat attribuutteja. Ulottuvuus voi sisältää myös ominaisuusattribuutteja, joita ei voida käyttää yhteenvetotietojen tuottamiseen, vaan niiden tehtävänä on antaa lisätietoa jostain ulottuvuustasosta.

Esimerkkitapauksena käytetään vähittäiskaupan tiliasiakkaiden tilejä kuvaavaa operatiivista järjestelmää, jonka käsitekaavio on esitetty kuviossa 18. Asiakkaalla voi olla useita tilejä, joista kullakin tilillä on luottoraja ja korko. Asiakas on joko yhtiö tai yksityishenkilö.



KUVIO 18. Vähittäiskaupan tiliasiakkaiden tilitapahtumia kuvaavan operatiivisen järjestelmän käsittekaavio (Hüsemann ym., 2000).

## Vaiheet

Menetelmä on jaettu neljään vaiheeseen, joista ensimmäinen on tietovaraston vaatimusmäärittely. Varsinainen tietovaraston käsittekaavion tuottaminen alkaa seuraavassa vaiheessa, käsitteellisen suunnittelun vaiheessa. Käsitteellinen suunnittelu on jaettu kolmeen osavaiheeseen. Niin ollen menetelmä käsittää yhteensä neljä vaihetta:

- tietovaraston vaatimusmäärittely
- mittauskohteiden kontekstien määrittely
- ulottuvuushierarkioiden suunnittelu
- summausrajoitteiden määrittely

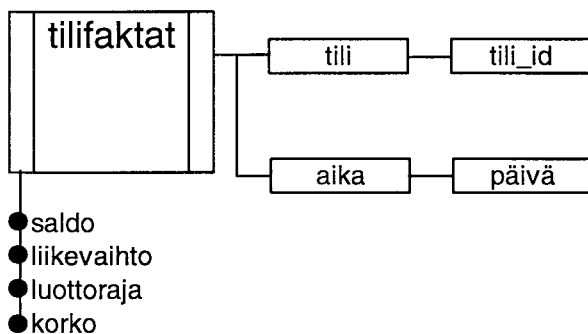
Menetelmän ensimmäisessä vaiheessa suoritetaan tietovaraston vaatimusmäärittely. Vaatimusmäärittelyssä liiketoiminnan asiantuntijat valitsevat operatiivisten järjestelmien attribuuteista liiketoimintastrategian kannalta tärkeimmät ja määrittelevät, soveltuvatko ne tietovaraston ulottuvuusattribuuteiksi vai mittauskohteiksi. Jokaisen

attribuutin kohdalla on oleellista päättää myös se, sisältääkö se valinnaista tietoa vai ei. Esimerkkitapauksessa valinnaista tietoa sisältävät Henkilön attribuutit ikä ja ammatti sekä yhtiön attribuutti ala. Vaatimusmäärittelyn tuloksena saadaan taulukko (TAULUKKO 2), jonka kullakin rivillä on attribuutin nimi, kuvaus sekä tieto siitä, onko attribuutti mittauskohde vai ulottuvuusattribuutti vai molempia ja sisältääkö attribuutti valinnaista tietoa vai ei. Esimerkkitapauksessa mittauskohteiksi valitaan saldo, liikevaihto, luottoraja ja korko. Taulukkoon voidaan tarvittaessa lisätä sellaisia attribuutteja, joita operatiivisen järjestelmän kaaviossa ei ole, mutta joita tietovaraston kannalta tarvitaan. Esimerkkitapauksessa taulukkoon lisätään attribuutit kuukausi, vuodenaika, vuosi, saldoluokka, liikevaihtoluokka ja asiakastyyppejä. Attribuuttia asiakastyyppejä käytetään kuvaamaan kahta erilaista asiakastyyppejä, henkilö ja yhtiö. Saldon ja liikevaihdon pidemmän aikavälin analysointia varten otetaan käyttöön kaksi ulottuvuusattribuuttia, saldoluokka ja liikevaihtoluokka. Attribuutit kuukausi, vuodenaika ja vuosi otetaan mukaan tietovaraston aikaulottuvuutta ajatellen. Taulukon lisäksi tuotetaan liite, joka sisältää muuta olennaista tietoa, kuten tietovarastoon tyypillisesti tehtäviä kyselyjä.

TAULUKKO 2. Vaatimusmäärittelyssä tuotetut tiedot sisältävä taulukko. Tässä taulukossa on vain osa tiedoista näkyvissä. (vrt. Husemann ym., 2000.)

attribuutin nimi	attribuutin kuvaus	mittaus- kohde	ulottuvuus- attribuutti	valinnainen attribuutti
saldo	saldo tietyssä päivänä	K	E	E
liikevaihto	liikevaihto tietyssä päivänä	K	E	E
luottoraja	tilin luottoraja	K	E	E
korko	tilin korko	K	E	E
asiakkaan nimi	asiakkaan nimi	E	K	E
henkilön ikä	yksityisasiakkaan ikä	E	K	K
kuukausi	aikaväli	E	K	E

Mittauskohteiden kontekstin määrittely alkaa funktionaalisten riippuvuuksien selvittämisellä ulottuvuustasojen ja mittauskohteiden välillä. Jokaiselle mittauskohteelle määritellään ensin avainattribuutit eli ne attribuutit, jotka vähintään tarvitaan mittauskohteen yksilöimiseksi. Seuraavaksi selvitetään, mistä attribuuteista mittauskohteet riippuvat funktionaalisesti. Jäljitystä jatketaan, kunnes löydetään ne attribuutit, jotka eivät ole funktionaalisesti riippuvia mistään attribuuteista. Nämä riippumattomat attribuutit toimivat ulottuvuushierarkioiden lähtökohtina ja vastaavat yleensä tähtimallin (ks. KUVIO 3) ulottuvuustaulujen otsikoita. Riippuvuudet esitetään taulukkomuodossa. Esimerkitapauksessa mittauskohteet saldo, liikevaihto, luottoraja ja korko ovat kaikki funktionaalisesti riippuvaisia attribuuteista tili\_id ja päivä. Ulottuvuushierarkioiden lähtökohdiksi valitaan niin ollen ulottuvuudet tili ja aika ja niiden attribuuteiksi tili\_id ja päivä. Vaiheen tuloksena syntyy myös tietovaraston käsitekaavion ensimmäinen versio, jossa näkyvät mittauskohteet ja ulottuvuushierarkioiden lähtökohdat. Esimerkitapauksessa syntyvän tietovaraston käsitekaavion ensimmäinen versio on esitetty kuviossa 19.



KUVIO 19. Vähittäiskaupan tiliasiakkaiden tilitapahtumia kuvaavan tietovaraston faktakaavion ensimmäinen versio (Hüsemann ym., 2000).

Seuraava vaihe on ulottuvuushierarkioiden suunnittelu. Edellisessä vaiheessa löydettyistä ulottuvuushierarkioiden lähtökohdista aloittamalla etsitään kaikki kuhunkin ulottuvuuteen funktionaalisten riippuvuuksien perusteella kuuluvat attribuutit. Ulottuvuustasot ja ominaisuusattribuutit tulee erotella toisistaan vaatimusmäärittelyn tietojen perusteella. Tuloksena syntyy suunnattu graafi, jonka solmuina ovat



ulottuvuustasot. Graafiin lisätään myös aiemmin erotellut ominaisuusattribuutit. Tässä vaiheessa on myös tarkistettava, sisältääkö ominaisuusattribuutti valinnaista tietoa vai ei. Lopuksi käydään ulottuvuushierarkiat kokonaisuudessaan läpi ja tutkitaan vaatimusmäärittelyssä tehdyn taulukon perusteella, löytyykö niistä valinnaisia polkuja. Valmiista graafista tulee sellaisenaan osa tietovaraston käsitekaaviota. Esimerkkitapauksessa attribuutit ammatti ja ala muodostavat valinnaisen ulottuvuustason. Ne erottuvat lopullisesta kaaviosta (KUVIO 17) tuplaviivoin piirrettyjen nuolien perusteella.

Seuraavana vaiheena on summausrajoitteiden määrittely. Tässä vaiheessa määritellään, mitä erilaisia operaatioita voidaan tehdä mittauskohteille. Esimerkiksi asiakkaiden ikäkeskiarvo voi olla hyödyllinen tieto, kun taas asiakkaiden yhteenlaskettu ikä ei. Tietovaraston käsitteellisen mallin tulisi tarjota mahdollisuudet erotella mittauskohteiden merkitykselliset ja merkityksettömät operaatiot toisistaan. Mittauskohteiden merkitykselliset operaatiot voitaisiin merkitä tietovaraston ulottuvuuskaavioon, mutta se vaikeuttaisi kaavion lukemista. Sen takia summausrajoitteista tehdään taulukkomuotoinen liite (TAULUKKO 2), josta selviää faktakaavion nimi, mittauskohteet ja kunkin mittauskohteen ulottuvuustasot sekä niiden rajoitustasot.

TAULUKKO 3. Summausrajoitteita kuvaava taulukko (Hüsemann ym., 2000)

faktakaavio	mittauskohde	ulottuvuustaso	rajoitustaso
tilifaktat	saldo	tili_id	1
		päivä	2
	liikevaihto	tili_id	1
		päivä	1
	luottoraja	tili_id	2
		päivä	2
	korko	tili_id	2
		päivä	2

Rajoitustasolla määritellään ne operaatiot, jotka kullakin ulottuvuustasolla ovat mahdollisia. Rajoitustasoja on neljä. Ensimmäinen taso sallii kaikki operaatiot, toinen taso kaikki muut operaatiot paitsi summaamisen. Kolmas taso sallii ainoastaan lukumäärän laskemisen (count), ja neljäs taso ei salli mitään operaatioita. Esimerkkitapauksessa suurin osa ulottuvuustasoista saa rajoitustasokseen tason kaksi. Saldolle tiliulottuvuudessa ja liikevaihdolle sekä tili- että aikaulottuvuudessa sallitaan kaikki operaatiot, joten niiden rajoitustaso on yksi.

Menetelmän käytön lopputuloksena syntyy tietovaraston käsitekaavio moniulotteisessa normaalimuodossa (KUVIO 17).

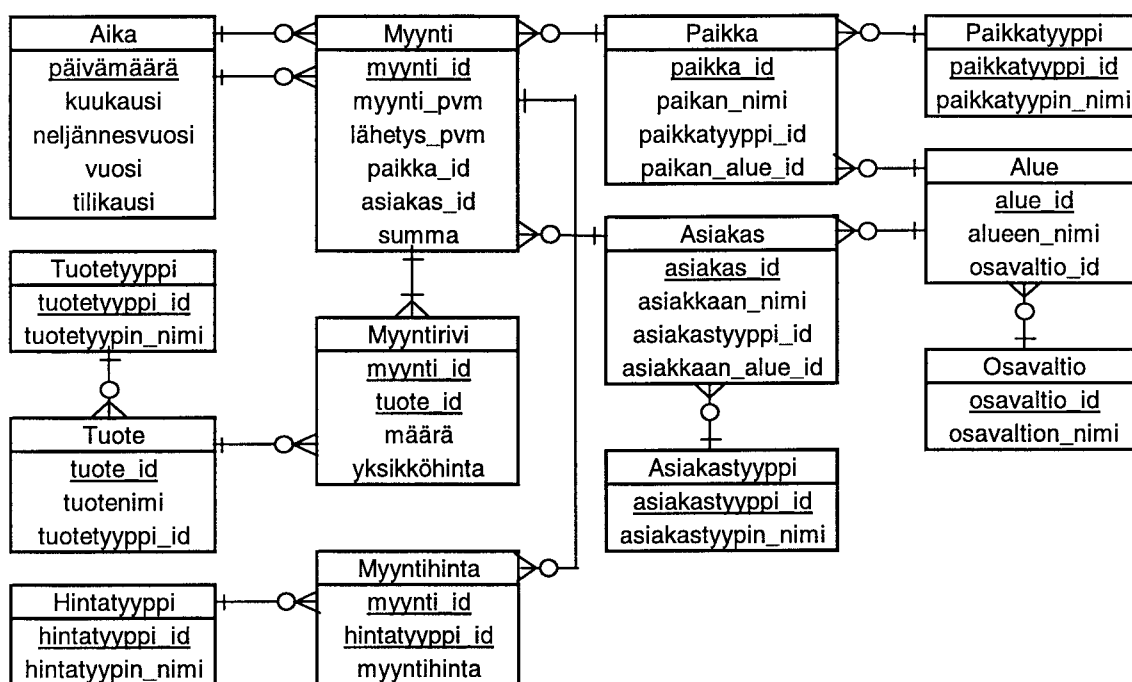
### **3.7 Moodyn ja Kortinkin menetelmä**

Moody ja Kortink (2000) esittelevät menetelmän, jonka avulla on mahdollista tuottaa moniulotteinen malli operatiivisten järjestelmien globaalien käsitekaavien pohjalta. Menetelmän käytön edellytyksenä on, että globaali käsitekaavio on tuotettu ER-notaatiota käyttäen. Moodyn ja Kortinkin (2000) mielestä tietovaraston suunnittelussa tulisi käyttää samanlaisia menetelmiä kuin perinteisesti on käytetty tietokantojen suunnittelussa, koska ne tukevat parhaiten tiedon johdonmukaisuutta ja oikeellisuutta ja antavat mahdollisuuden lisätä tietoa helposti tietovarastoon. Heidän mielestään taas paikallisvaraston suunnitteluun sopivat parhaiten moniulotteiset suunnittelumallit, koska niiden avulla käyttäjien on helpompi ymmärtää paikallisvaraston rakennetta ja sitä kautta helpompi käyttää paikallisvarastoa suunniteltuihin tehtäviin. Seuraavaksi esitettävä menetelmä onkin tarkoitettu paikallisvaraston moniulotteisen mallin johtamiseen operatiivisen järjestelmän käsitekaavien pohjalta. Tarvittaessa menetelmää voidaan käyttää myös varsinaisen tietovaraston moniulotteisen käsitekaavien tuottamiseen. Jatkossa oletetaan selvyuden vuoksi, että menetelmän avulla ollaan suunnittelemassa tietovarastoa.

## Malli

Tietovaraston moniulotteisen käsittekaavion malleiksi on annettu viisi vaihtoehtoa: litteä malli (flat schema), porrastettu malli (terraced schema), tähtimalli, lumihiutalemalli ja tähtisikermämalli (star cluster schema). Menetelmän käyttäjä voi valita tilanteeseen sopivimman mallin. Päätös käytettävästä mallista tarvitsee tehdä vasta menetelmän käytön loppuvaiheessa. Mallit kuvataan seuraavassa kohdassa tarkemmin.

Esimerkkitapauksena käytetään kauppaketjun myyntitapahtumia ja asiakastietoja tallentavaa järjestelmää, jonka käsittekaavio on esitetty kuviossa 20.



KUVIO 20. Kauppaketjun myyntitapahtumia ja asiakastietoja tallentavan järjestelmän ER-kaavio (Moody ja Kortink, 2000).

## Vaiheet

Menetelmä on jaettu neljään vaiheeseen:

- kohdetyyppien luokittelu
- hierarkioiden tunnistaminen
- ulottuvuusmallien tuottaminen
- arviointi ja jalostaminen

Ensimmäisessä vaiheessa kohdetyypit luokitellaan kolmeen kategoriaan niiden merkityksen perusteella ER-kaaviossa. *Tapahtumakohdetyyppi* kuvaa jotakin tapahtumaa, joka sattuu tietyllä hetkellä. Lisäksi kohdetyyppi sisältää määrällisiä mittauskohteita, joita voidaan laskea. Tapahtumakohdetyypit tallentavat tietoja liiketoimintatapahtumista, joita ovat esimerkiksi tilaukset ja palkanmaksut. Näitä tapahtumia koskevaa tietoa päätöksentekijät haluavat tietovaraston tietojen avulla analysoida. Tapahtumakohdetyypeistä tulee tietovaraston tähtimallin faktatauluja. Kuitenkaan kaikki tapahtumakohdetyypit eivät ole päätöksenteon kannalta oleellisia, joten käyttäjävaatimusten perusteella on tunnistettava tärkeimmät kohdetyypit. Esimerkkitapauksessa olennaisia tapahtumakohdetyyppejä ovat myynti, myyntirivi ja myyntihinta.

*Komponenttikohdetyypit* ovat niitä ER-kaavion kohdetyyppejä, jotka yhdistyvät tapahtumakohdetyyppeihin (1:n) -suhteella. Ne määrittelevät liiketoimintatapahtumien yksityiskohtia vastaten kysymyksiin kuka, missä ja milloin. Esimerkkeinä komponenttikohdetyypeistä mainittakoon asiakas ja tuote. Komponenttikohdetyypeistä tulee tietovaraston tähtimallin ulottuvuustauluja. Esimerkkitapauksessa komponenttikohdetyyppejä ovat aika, paikka, asiakas, tuote ja hintatyyppi.

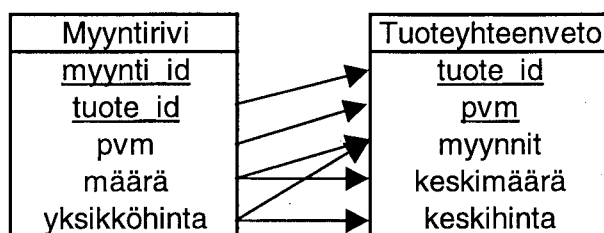
*Luokittelukohdetyypit* yhdistyvät puolestaan komponenttikohdetyyppeihin (1:n) -suhteiden ketjulla. Luokittelukohdetyypit riippuvat siis funktionaalisesti komponenttikohdetyypeistä. Luokittelukohdetyypit muodostavat ER-kaaviossa hierarkioita, ja tietovaraston tähtimallissa niistä ja niiden attribuuteista tulee ulottuvuustaulujen rivejä. Esimerkkitapauksessa luokittelukohdetyyppejä ovat tuotetyyppi, alue, osavaltio ja asiakastyypit.

On mahdollista, että jokin kohdetyyppi sopisi kahteen kategoriaan, joten kategoriat on asetettu arvojärjestykseen, jonka mukaan kohdetyypit luokitellaan. Tapahtumakohdetyyppi on arvokkain, luokittelukohdetyyppi on toiseksi arvokkain ja komponenttikohdetyyppi on vähiten arvokas. Kahteen kategoriaan sopiva kohdetyyppi luokitellaan aina arvojärjestyksessä korkeampaan kategoriaan.

Menetelmän toisena vaiheena on hierarkioiden tunnistaminen. ER-kaaviosta hierarkiat tunnistaa siitä, että ne ovat ketjuja kohdetyyppejä, joiden välissä on (n:1) –suhteet. Hierarkia on maksimaalinen, jos sitä ei voida laajentaa lisäämällä siihen uusia kohdetyyppejä. Kohdetyyppi on minimaalinen, jos se on maksimaalisen hierarkian pohjalla eli siihen ei liity yhtään (n:1) –suhdetyyppejä. Kohdetyyppi on maksimaalinen, jos se on maksimaalisen hierarkian huipulla eli siihen ei liity yhtään (1:n) -suhdetyyppejä. Esimerkkitapauksessa minimaalinen kohdetyyppi on esimerkiksi osavaltio ja maksimaalinen kohdetyyppi myyntirivi.

Kolmannessa vaiheessa tuotetaan ulottuvuusmallit. Näillä tarkoitetaan tämän tutkielman terminologian mukaan moniulotteisia malleja. Ulottuvuusmalleja voidaan muodostaa romahduttamalla (collapse) hierarkioita tai summaamalla tietoja. Hierarkioiden *romahduttaminen* tarkoittaa sitä, että hierarkian huipulla olevan kohdetyypin attribuutit sisällytetään seuraavana hierarkiassa olevaan kohdetyyppiin. Romahduttamista voidaan halutessa jatkaa, kunnes hierarkiasta on jäljellä vain yksi kohdetyyppi, joka sisältää alkuperäisen hierarkian kaikkien kohdetyyppeiden attribuutit. Hierarkioiden romahduttaminen on siis ER-kaavion denormalisointia.

Summaamista voidaan soveltaa vain tapahtumakohdetyyppeihin ja vain niiden numeerisiin attribuutteihin. Summaaminen tapahtuu luomalla uusi kohdetyyppi, johon attribuuteiksi tulevat summattavaksi haluttavat attribuutit sekä ryhmittelyattribuutit. Esimerkiksi yksittäistä myytyä tuotetta kuvaavasta kohdetyypistä myyntirivi voidaan muodostaa uusi kohdetyyppi tuoteyhteenveo, joka sisältää tuotteen myyntituoton (yksikköhinta\*myytyjen tuotteiden määrä), keskimäärin kerralla myydyin kappalemäärän ja tuotteen päivittäisen keskihinnan. Tapaus on esitetty kuviossa 21.



KUVIO 21. Yhteenvetokohdetyypin muodostaminen tapahtumakohdetyypistä (Moody ja Kortink, 2000).

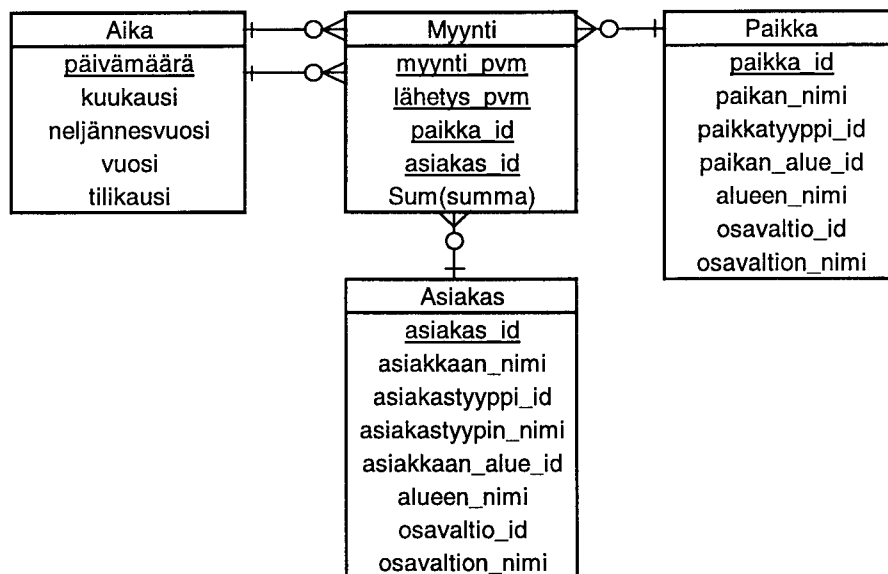
Varsinaisia ulottuvuusmalleja on valittavissa viisi erilaista: litteä malli (flat schema), porrastettu malli (terraced schema), tähtimalli, lumihuutalemalli ja tähtisikermämalli (star cluster schema). Seuraavaksi esitellään, miten kukin malli tuotetaan ER-kaavion pohjalta romahduttamista ja summaamista käyttäen.

Litteä malli muodostetaan romahduttamalla kaikki ER-kaavion kohdetyypit minimaaliseen kohdetyyppiin asti. Lopputuloksena saadaan yksi taulu kutakin minimaalista kohdetyyppiä kohden. Mitään tietoa alkuperäisestä ER-kaaviosta ei häviä romahduttamisen seurauksena. Litteä malli sisältää toistoa, mutta ei sisällä lainkaan summatauluja. Litteässä mallissa taulujen määrä voi olla vähäinen, mutta taulujen sisältämien attribuuttien määrä suuri. Toinen ongelma on se, että hierarkkiset suhteet tapahtumakohdetyyppien välillä saattavat johtaa tietovarastossa summaamisvirheisiin. Esimerkkitapauksessa litteä malli koostuisi kahdesta taulusta: myyntirivi ja myyntihinta. Myyntirivin attribuuteiksi tulisivat attribuutit ER-kaavion kohdetyypeistä myyntirivi, tuote, tuotetyyppi, myynti, aika, paikka, paikkatyyppi, asiakas, asiakastyyppi, alue ja osavaltio. Myyntihinnan attribuuteiksi tulisivat attribuutit ER-kaavion kohdetyypeistä hinta, hintatyyppi, myynti, aika, paikka, paikkatyyppi, asiakas, asiakastyyppi, alue ja osavaltio.

Porrastettu malli muodostetaan romahduttamalla kohdetyyppejä maksimaalisista kohdetyypeistä aloittaen, kunnes vastaan tulee tapahtumakohdetyyppi. Tuloksena saadaan taulu jokaista ER-kaavion tapahtumakohdetyyppiä kohden. Esimerkkitapauksessa porrastettu malli koostuisi kolmesta taulusta: myyntirivi, myynti

ja myyntihinta. Myyntirivin attribuuteiksi tulisivat ER-kaaviosta myyntirivin, tuotteen ja tuotetyypin attribuutit. Myynnin attribuuteiksi tulisivat attribuutit ER-kaavion kohdetyypeistä myynti, aika, paikka, paikkatyyppi, asiakas, asiakastyypin, alue ja osavaltio. Myyntihinnan attribuuteiksi tulisivat ER-kaaviosta myyntihinnan ja hintatyyppin attribuutit.

Tähtimalli tuotetaan jokaista ER-kaavion tapahtumakohdetyypistä kohden. ER-kaavion tapahtumakohdetyypistä tulee tähtimallin faktataulu ja faktataulun avaimiksi tulee tapahtumakohdetyypiin liittyvien komponenttikohdetyypien avainattribuutit. Tapahtumakohdetyypiin liittyvistä komponenttikohdetyypeistä tulee tähtimallin ulottuvuustauluja romahduttamalla kaikki siihen liittyvät luokittelukohdetyypit komponenttikohdetyypiin. Jos ER-kaavion tapahtumakohdetyypien välillä on hierarkkisia suhteita, perii lapsi kaikki isänsä ulottuvuudet ja avainattribuutit. Tapahtumakohdetyypien sisältämiä numeerisia attribuutteja pitäisi pystyä summaamaan avainattribuuttien mukaan. Esimerkkitapauksen ER-kaavion tapahtumakohdetyypistä myynti johdettu tähtimalli on esitetty kuviossa 22.



KUVIO 22. Myynnin tähtimalli (Moody ja Kortink, 2000).

Esimerkkitapauksessa kohdetyypin myynti lapsi on myyntirivi. Jos tähtimalli johdettaisiin kohdetyypistä myyntirivi, periytyisivät kohdetyypin myynti ulottuvuudet myyntiriville. Kuviossa 22 esitettyyn tähtimalliin lisättäisiin ulottuvuus Tuote ja sen attribuuteiksi kohdetyypin tuote attribuutit. Faktataulu olisi siinä tapauksessa luonnollisesti myyntirivi, jonka avainattribuutiksi tulisi kuviossa 22 esiintyvien attribuuttien lisäksi tuote\_id. Mittauskohteina olisivat myyntirivien määrä ( $\text{Sum}(\text{määrä})$ ) ja myyntirivien yksikköhintojen summa ( $\text{Sum}(\text{yksikköhinta})$ ).

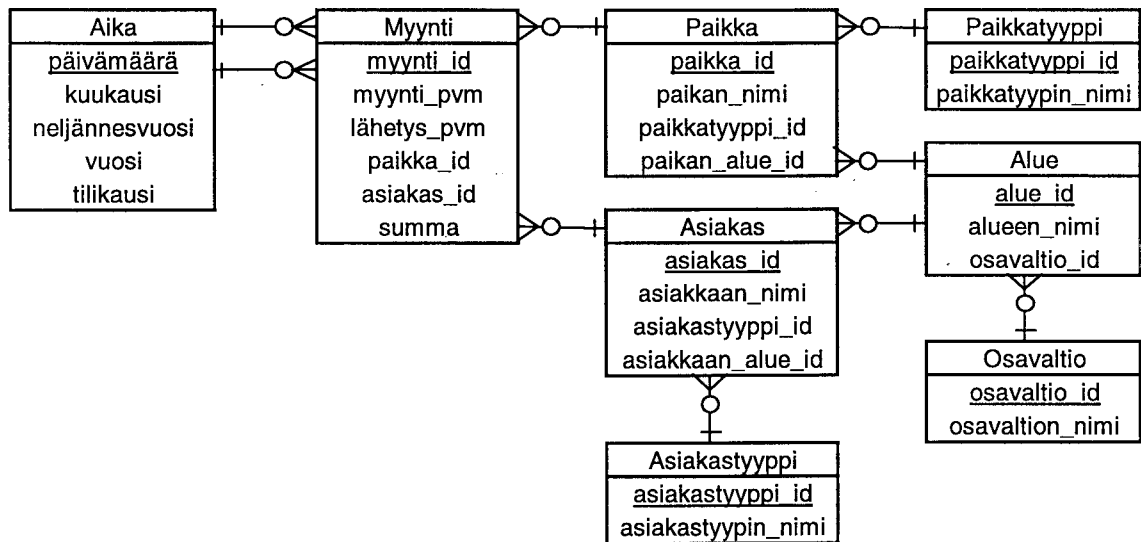
Jos ei haluta muodostaa useita erillisiä tähtimalleja, on myös mahdollista muodostaa tähtikuviomalli, joka muodostuu joukosta tähtimalleja, joiden faktataulut on hierarkkisesti liitetty toisiinsa. Faktataulujen väliset linkit antavat tietovarastossa mahdollisuuden porautua tiedoissa tarkemmalle tasolle. Tähtimalleista on mahdollista muodostaa myös galaksimalli, joka eroaa tähtikuviomallista siinä, että faktataulujen ei tarvitse liittyä toisiinsa. Lisäksi ulottuvuustaulut voivat olla jaettuja faktataulujen kesken. Tähtikuviomalli ja galaksimalli ovat kuitenkin sen verran harvinaisia, että niitä ei tässä tutkielmassa käsitellä enempää.

Lumihiutalemalli eroaa tähtimallista siinä, että lumihiutalemallissa hierarkioita ei romahduteta. Lumihiutalemalli voidaan johtaa tähtimallista normalisoimalla kunkin ulottuvuuden hierarkiatauluja. Toisaalta lumihiutalemalli voidaan tuottaa suoraan operatiivisen järjestelmän ER-kaaviosta seuraavia ohjeita noudattaen:

- Jokaisesta tapahtumakohdetyypistä tehdään lumihiutalemallin faktataulu, jonka perusavaimeksi tulee siihen liittyvien luokittelukohdetyypien avainattribuuttien yhdistelmä.
- Jokaisesta komponenttikohdetyypistä tehdään ulottuvuustaulu.
- Jos tapahtumakohdetyypien välillä on suhteita, niin lapsi perii kaikki isänsä suhteet komponenttikohdetyyppeihin.
- Tapahtumakohdetyypien numeerisia attribuutteja pitää pystyä summaamaan avainattribuuttien mukaan.

Esimerkkitapauksen kohdetyypistä myynti johdettu lumihiutalemalli on esitetty kuviossa 23.





KUVIO 23. Myynnin lumihiutalemalli (Moody ja Kortink, 2000).

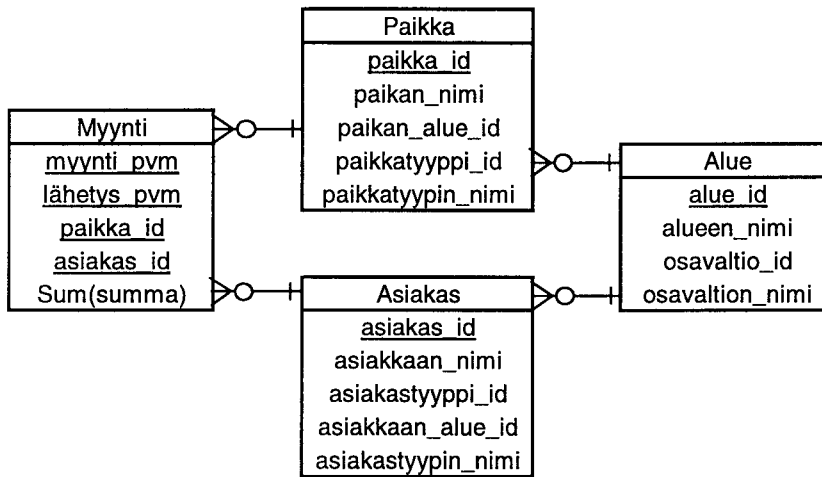
Tähtisikermämalli on tähtimallin ja lumihiutalemallin välimuoto. Siinä ulottuvuudet ovat ortogonaalisia eli niissä samat tiedot eivät toistu kuten lumihiutalemallissa, mutta toisaalta ulottuvuuksia ei ole romahdutettu liiaksi kuten tähtimallissa. Toisteista tietoa sisältävät ulottuvuudet voidaan tunnistaa ER-kaavion oksanhaaroista (fork). Oksanhaaralla tarkoitetaan rakennetta, jossa sama kohdetyyppi toimii kahden tai useamman eri ulottuvuushierarkian isänä. Esimerkkitapauksessa oksanhaararakenne on nähtävissä kohdassa, jossa kohdetyyppi alue on sekä paikan että asiakkaan isä.

Tähtisikermämalli voidaan tuottaa ER-kaaviosta seuraavien ohjeiden mukaan:

- Jokaisesta tapahtumakohdetyypistä tehdään lumihiutalemallin faktataulu, jonka perusavaimeksi tulee siihen liittyvien luokittelukohdetyyppien avainattribuuttien yhdistelmä.
- Luokittelukohdetyyppejä romahdutetaan, kunnes kohdataan oksanhaarakohdetyyppi tai luokittelukohdetyyppi. Jos ensimmäisenä saavutetaan oksanhaarakohdetyyppi, muodostetaan aliulottuvuustaulu, joka koostuu oksanhaarakohdetyypistä sekä sen isästä ja esi-isistä. Romahduttamista jatketaan oksanhaarakohdetyypin jälkeen, kunnes vastaan tulee komponenttikohdetyyppi, jolloin muodostetaan ulottuvuustaulu. Esimerkkitapauksessa aliulottuvuustaulu muodostetaan oksanhaarakohdetyyppi alueesta ja sen vanhemmasta osavaltiosta.

- Jos tapahtumakohdetyyppien välillä on suhteita, niin lapsi perii kaikki isänsä suhteet komponenttikohdetyyppeihin.
- Tapahtumakohdetyyppien numeerisia attribuutteja pitää pystyä summaamaan avainattribuuttien mukaan.

Esimerkitapauksen kohdetyypistä myynti johdettu tähtisikermämalli on esitetty kuviossa 24.



KUVIO 24. Myynnin tähtisikermämalli (Moody ja Kortink, 2000).

Neljännessä ja viimeisessä vaiheessa arvioidaan ja jalostetaan edellisessä vaiheessa tuotettua mallia. Tähtimallin faktatauluja voidaan yhdistää, jos niillä on samat avainattribuutit eli niihin liittyvät samat ulottuvuudet. Toisiinsa liittyvät ulottuvuustaulut voidaan yhdistää yhdeksi ulottuvuustauluksi tietovaraston rakenteen yksinkertaistamiseksi. Suurimpia ongelmia ER-kaavion muuntamisessa moniulotteiseksi kaavioksi tuottavat risteyskohdan kohdetyypit (intersection entity) tai (m:n) –suhdetyypit. Ongelmat voidaan ratkaista seuraavilla tavoilla:

- Jätetään risteyskohdan kohdetyyppi huomioimatta kokonaan eli ei oteta sitä mukaan tietovarastoon.

- Muunnetaan (m:n) –suhdetyyppi (1:n) –suhdetyypiksi määrittelemällä jokin suhteista tärkeimmäksi.
- Sisällytetään (m:n) –suhdetyyppi sellaisenaan tietovarastoon, jolloin siitä voi olla hyötyä kehittyneille käyttäjille, mutta estetään suhdetyypin käyttö tavanomaisten kyselyjen yhteydessä.

Isän ja lasten väliset suhdetyypit voidaan muuntaa hierarkkiseksi rakenteeksi luomalla luokittelukohdetyyppi lapsien erottamiseksi.

## 4. MENETELMIEN VERTAILU JA ARVIOINTI

Tässä luvussa vertaillaan ja arvioidaan edellisessä luvussa esitettyjä menetelmiä. Ensin tarkastelun kohteena ovat menetelmien tietomallit. Sen jälkeen tarkastellaan, miten tietovaraston käsittekaavion johtamiseen kuuluvat tehtävät menetelmien eri vaiheissa suoritetaan. Lopuksi esitetään subjektiivisia arvioita menetelmistä ja pyritään esittämään mahdollisiin heikkouksiin parannusehdotuksia. Arvioitavia menetelmiä ovat Golfarellin ym. (1998), Cabibbon ja Torlonen (1998), Böhnleinin ja Ulbrich-vom Enden (1999), Hüsemannin ym. (2000) sekä Kortinkin ja Moodyn (2000) menetelmät. Edellisessä luvussa edellä mainittujen menetelmien lisäksi esitelty käsittekaavioiden integrointi ei ole arvioitavana.

### 4.1 Menetelmien tietomallit

Taulukossa 3 on esitetty yhteenveto edellisessä luvussa esitetyissä menetelmissä käytetyistä tietomalleista. Kustakin menetelmästä on kerrottu lähde, menetelmän edellyttämä lähtömalli, mahdolliset välituloksena saatavat mallit sekä lopputuloksena saatava malli. Seuraavassa tarkastellaan lähemmin menetelmien lähtömallien, välitulostulosten ja lopputulostulosten ominaisuuksia, yhtenevyyksiä ja eroavuuksia sekä mallien vahvuuksia muihin malleihin nähden.

#### Lähtömallit

Kaikissa käsitellyissä menetelmissä lähtömallin on oltava ER-kaaviona paitsi Böhnleinin ja Ulbrich-vom Enden (1999) menetelmässä, jossa lähtömallin edellytetään olevan SER-kaaviona. ER-kaavio on kuitenkin melko helposti muutettavissa SER-kaavioksi annettujen sääntöjen mukaan. Jokaisessa menetelmässä edellytetään, että

lähtömallina toimiva käsitekaavio on globaali käsitekaavio, joka on tuotettu integroimalla operatiivisten järjestelmien käsitekaavioista, mikäli tietovarastoon tarvitaan tietoja useammasta järjestelmästä.

TAULUKKO 3. Menetelmien tietomallit.

Menetelmä	Lähtömalli	Välitulosmallit	Lopputulomalli
Golfarelli ym. (1998)	ER	attribuuttipuu	moniulotteinen faktakaavio
Cabibbo ja Torlone (1998)	ER	ulottuvuusgraafi	moniulotteinen malli, tyypillisesti tähtimalli
Böhnlein ja Ulbrich-vom Ende (1999)	SER		tähtimalli
Hüsemann ym. (2000)	ER	faktakaavion alkuversio	faktakaavio yleistetyssä moniulotteisessa normaalimuodossa, taulukko summausrajoitteista
Moody ja Kortink (2000)	ER		litteä malli, porrastettu malli, tähtimalli, tähtikuviomalli, galaksimalli, lumihiihtalemalli tai tähtisikermämalli

Golfarelli ym.(1998) sekä Cabibbo ja Torlone (1998) käyttävät lähtömallin notaationa Batinin ym. (1992) esittämää notaatiota, jossa kohdetyypit kuvataan laatikoina, suhdetyypit salmiakkeina ja attribuutit pieninä ympyröinä, jotka on viivalla yhdistetty kohdetyyppeihin. Hüsemann ym. (2000) sekä Moody ja Kortink (2000) käyttävät lähtömalleissaan Everestin (1986) kehittämää ”variksenjalka”-merkintätapaa kardinaalisuuksien ilmaisemiseen. Kohdetyypit ja attribuutit taas on esitetty Rumbaughin ym. (1991) oliokeskeisten tietojärjestelmien suunnittelua varten

kehittämässä OMT-menetelmässä käytetyn notaation mukaisesti, jossa laatikon otsikkona on kohdetyypin nimi ja sen alapuolisilla riveillä attribuuttien nimet.

Hüsemannin ym. (2000) sekä Moodyn ja Kortinkin (2000) käyttämät lähtömallit ovat mielestäni kaikista selkeimpiä, koska kohdetyyppien attribuutit on lueteltu allekkain saman laatikon sisälle. Golfarellin ym.(1998) sekä Cabibbon ja Torlonen (1998) käyttämässä mallissa kohdetyyppihin liittyvät attribuutit ovat hajallaan kohdetyypin ympärillä ja varsinkin tällä notaatiolla tuotettujen laajojen ER-kaavioiden tarkastelu olisi hankalaa. Böhnleinin ja Ulbrich-vom Enden (1999) SER-malli on luonteeltaan hyvin selkeä, koska sitä voidaan loogisesti tarkastella suunnasta toiseen. Sen heikkoutena voidaan pitää sitä, että se on monille varsin vieras malli.

### **Välitulospallit**

Golfarellin ym. (1998) menetelmässä syntyy välituloksena attribuuttipuu, josta eri vaiheiden jälkeen tuotetaan tietovaraston moniulotteinen faktakaavio. Cabibbon ja Torlonen (1998) menetelmässä välituloksena syntyy ulottuvuusgraafi, joka muunnetaan lopuksi tietovaraston moniulotteiseksi malliksi.

Muiden menetelmien tuloksena erillisiä välitulospalleja ei synny, jos ei sellaiseksi lasketa Hüsemannin ym. (2000) menetelmässä tuotettavaa tietovaraston moniulotteisen faktakaavion alustavaa versiota. Siihen lisätään jatkossa muita osia, jolloin lopulta muodostuu faktakaavio moniulotteisessa normaalimuodossa. Böhnleinin ja Ulbrich-vom Enden (1999) sekä Moodyn ja Kortinkin (2000) menetelmät ovat suoraviivaisempia, koska niissä ei erillisiä välitulospalleja tarvitse tehdä, vaan lopputulosmalli tuotetaan suoraan lähtömallin perusteella.

Golfarellin ym. (1998) välitulospallina käyttämä attribuuttipuu vaikuttaa hyvin toimivalta varsinkin siinä tapauksessa, että attribuuttipuun pohjana toimiva ER-kaavio on hyvin laaja ja sen tarkastelu sellaisenaan olisi liian hankalaa. Attribuuttipuu on rakenteeltaan hyvin yksinkertainen, koska siinä eivät näy suhdetyypit eivätkä kardinaalisuudet. Lisäksi attribuuttipuulla on tärkeä rooli, kun menetelmän

jatkovaiheissa määritellään ulottuvuuksia ja hierarkioita. Cabibbon ja Torlonen (1998) käyttämä ulottuvuusgraafi on periaatteeltaan samanlainen kuin attribuuttipuukin, vain hieman monimutkaisempi. Ulottuvuusgraafissakin esitetään ER-kaavion sisältämät osat yksinkertaisemmassa muodossa ja sitä hyödynnetään, kun lopulta tuotetaan tietovaraston moniulotteinen malli.

### **Lopputulospallit**

Golfarellin ym. (1998) menetelmässä lopputulospallina saadaan tietovaraston moniulotteinen faktakaavio, joka esitetään omaperäisellä notaatiolla. Muissa menetelmissä samanlaista notaatiota ei käytetä. Moniulotteinen faktakaavio olisi kuitenkin helposti muunnettavissa tavalliseksi tähtimalliksi. Moniulotteisen faktakaavion etuna on se, että siinä on ulottuvuuksissa selvästi eroteltu sellaiset attribuutit, joita ei voi käyttää yhteenvetotietojen tuottamiseen.

Cabibbon ja Torlonen (1998) sekä Böhnleinin ja Ulbrich-vom Enden (1999) menetelmien lopputuloksena syntyy perinteinen tähtimalli. Tosin Cabibbon ja Torlonen menetelmän avulla voidaan välitulospallina olevasta ulottuvuusgraafista johtaa myöskin joku muu tietovaraston moniulotteinen malli, esimerkiksi lumihiihtalemalli. Moodyn ja Kortinkin (2000) menetelmä antaa mahdollisuuden valita peräti seitsemän erilaisen tietovaraston käsitteellisen mallin väliltä, mukaan lukien tavallisimmat mallit eli tähtimalli ja lumihiihtalemalli.

Hüsemannin ym. (2000) menetelmän lopputulospallina on tietovaraston faktakaavio yleistetyssä moniulotteisessa normaalimuodossa. Malli poikkeaa selvästi kaikissa muissa menetelmissä käytetyistä malleista. Epäselvyyttä aiheuttaa monille merkitykseltään ja esitykseltään vieraan moniulotteisen normaalimuodon käyttö.

## 4.2 Menetelmien tehtävät

Menetelmien vaiheita tarkasteltaessa voidaan havaita, että varsin samanlaiset tehtävät toistuvat, vaikka niiden nimeäminen, toteuttamistapa ja toteuttamisjärjestys poikkeavatkin menetelmissä. Taulukkoon 4 on koottu menetelmien vaiheet. Siinä on myös esitetty, missä menetelmän vaiheissa suoritetaan seuraavat perustehtävät:

- valmistelutehtävät (taulukossa sarake numero 1)
- faktojen ja mittauskohteiden valinta (2)
- ulottuvuuksien määrittely (3)
- hierarkioiden määrittely (4)
- summaamisesta päättäminen (5).

Tällaiseen tehtäväjakoon on päädytty, koska tehtävissä näkyvät tietovaraston moniulotteisen mallin eri osien määrittelyvaiheet sekä tietovarastoinnin tärkeästä ominaisuudesta, summaamisesta, päättäminen. Valmistelutehtävät on otettu mukaan, koska niitä esiintyy muutamissa menetelmissä.

Seuraavaksi menetelmien vaiheita ja tehtäviä tarkastellaan taulukon tehtäväjäsennyksen mukaisessa järjestyksessä.

### Valmistelutehtävät

Valmistelutehtävillä tarkoitetaan niitä menetelmien tehtäviä, joiden tarkoituksena on avustaa seuraavia vaiheita suorittamalla käsittekaavion analysointia, uudelleenjärjestelyä tai muita tukitoimintoja, joista on jatkossa hyötyä varsinaisille tehtäville.

Golfarellin ym. (1998) menetelmässä valmistelutehtävät sisältyvät faktojen määrittelyyn. Siinä vaiheessa lähtömallina toimivassa ER-kaaviossa muunnetaan (n:m) –suhdetyyppit kohdetyypeiksi ja lisätään tarvittavat suhdetyypit ja kardinaalisuudet kohdetyyppien ympärille. Faktojen määrittelyn jälkeen rakennetaan kullekin faktalle attribuuttipuu, jonka juurena on faktan avainattribuutti. Seuraavassa vaiheessa



attribuuttipuita karsitaan ja jalostetaan poistamalla niistä tarpeettomia osia. Tästä on hyötyä jatkoa ajatellen, koska attribuuttipuun perusteella voidaan myöhemmissä vaiheissa varsin suoraviivaisesti tunnistaa niin ulottuvuudet kuin hierarkiakin. Kuitenkin attribuuttipuun karsintavaiheessa on jo mietittävä, mitä ulottuvuuksia tietovarastossa tarvitaan, ettei tule poistaneeksi attribuuttipuusta tarpeellisia osia.

TAULUKKO 4. Menetelmien vaiheet.

Menetelmien vaiheet	1	2	3	4	5
Golfarelli ym. (1998)					
- faktojen määrittely	X	X			
- attribuuttipuun rakentaminen	X				
- attribuuttipuun karsiminen ja jalostaminen	X				
- ulottuvuuksien määrittely			X		
- fakta-attribuuttien määrittely		X			
- hierarkioiden määrittely				X	
Cabibbo ja Torlone (1998)					
- faktojen, mittauskohteiden ja ulottuvuuksien tunnistaminen		X	X	X	
- käsittekaavion uudelleenjärjestely	X		X	X	
- ulottuvuusgraafin johtaminen		X	X	X	
- muuntaminen moniulotteiseksi malliksi		X	X	X	
Böhnlein ja Ulbrich-vom Ende (1999)					
- mitattavien liiketoiminta-kohteiden tunnistaminen		X			
- ulottuvuuksien ja ulottuvuushierarkioiden tunnistaminen			X	X	
- ulottuvuushierarkioiden rajoitteiden tunnistaminen					X
Hüsemann ym. (2000)					
- tietovaraston vaatimusmäärittely	X	X			
- mittauskohteiden kontekstin määrittely			X		
- ulottuvuushierarkioiden suunnittelu				X	
- summausrajoitteiden määrittely					X
Moody ja Kortink (2000)					
- kohdetyyppien luokittelu	X	X	X	X	
- hierarkioiden tunnistaminen				X	
- ulottuvuusmallien tuottaminen		X	X	X	
- arviointi ja jalostaminen			X	X	

Cabibbon ja Torlonen (1998) menetelmässä valmistelutehtävät ovat vasta toisessa vaiheessa, jossa käsittekaaviota uudelleenjärjestellään. Siinä on kuitenkin jo tiedettävä

menetelmän ensimmäisessä vaiheessa valitut faktat ja ulottuvuudet, koska uudelleenjärjestelyssä kaikki faktat muutetaan kohdetyypeiksi ja ulottuvuuksia varten lisätään ER-kaavioon tarvittaessa attribuutteja.

Hüsemannin ym. (2000) menetelmässä valmistelutehtäväksi voidaan lukea tietovaraston vaatimusmäärittelyyn liittyvä tehtävä, joka on varsinaisesti tietovaraston suunnittelussa käsitteellistä suunnittelua edeltävä vaihe. Tässä se on otettu mukaan tarkasteluun, koska sillä on olennaista merkitystä tietovaraston käsitekaavion tuottamisessa. Vaatimusmäärittelyssä operatiivisten järjestelmien attribuuteista valitaan liiketoimintastrategian kannalta tärkeimpiä attribuutteja. Vaatimusmäärittelyn aikana tuotetaan myös liite, johon on kerätty muun muassa tietovarastoon tyypillisesti tehtäviä kyselyjä.

Moodyn ja Kortinkin (2000) menetelmässä ensimmäisenä vaiheena on kohdetyyppien luokittelu. Luokittelu kolmeen kategoriaan tapahtuu kohdetyyppien merkityksen perusteella. Tässä menetelmässä valmistelutehtäviin on panostettu selvästi eniten ja jatkovaiheissa tästä onkin hyötyä, koska luokittelun perusteella voidaan helposti määrittellä tietovaraston faktat, ulottuvuudet ja hierarkiat. Hyvin suoritettu valmisteluvaihe takaa sen, että menetelmän tulevien vaiheiden suorittaminen on melkein automaattista eikä virheiden mahdollisuuksia juurikaan ole. Böhnleinin ja Ulbrich-vom Enden (1999) menetelmässä varsinaisia valmistelutehtäviä ei ole erotettavissa lainkaan.

### **Faktojen ja mittauskohteiden valinta**

Faktojen ja mittauskohteiden tunnistamisella tarkoitetaan niitä tehtäviä, jotka on suoritettava selvitetessä, mitkä käsitekaavion osat valitaan tietovaraston faktoiksi ja mittauskohteiksi.

Golfarellin ym. (1998) menetelmässä faktojen määrittely on ensimmäinen tehtävä. Faktojen etsiminen tapahtuu tarkastelemalla ER-kaaviota ja etsimällä usein päivitettäviä asioita kuvaavia kohde- ja suhdetyyppejä. Faktojen lopullinen valinta on kuitenkin suunnittelijan tehtävä. Fakta-attribuuttien eli mittauskohteiden tunnistaminen tapahtuu

etsimällä kohdetyyppien ilmentymiä kuvaavia attribuutteja tai attribuutteja, jotka kohdistavat muita yleisimpiä operaatioita attribuuttipuussa oleviin attribuutteihin.

Cabibbon ja Torlonen (1998) menetelmässä faktat ja mittauskohteet pyritään tunnistamaan ER-kaaviosta annetun ohjeistuksen avulla. Faktat ovat ER-kaavion osia, joihin päätöksentekoprosessi kohdistuu, ja mittauskohteet faktojen atomisia attribuutteja, joita halutaan analysoida.

Böhnleinin ja Ulbrich-vom Enden (1999) menetelmän ensimmäisessä vaiheessa pyritään tunnistamaan mitattavat liiketoimintakohteet ja niiden perusteella faktat ja niihin liittyvät mittauskohteet, joita käytetään liiketoimintakohteen mittaamiseen. Yleensä faktat löytyvät SER-kaavion oikeasta reunasta, jossa sijaitsevat riippumattomat tieto-objektityypit.

Hüsemannin ym. (2000) menetelmässä mittauskohteiden valinta tapahtuu ensimmäisessä vaiheessa, tietovaraston vaatimusmäärittelyssä. Siinä operatiivisten järjestelmien attribuuteista valitaan liiketoiminnan kannalta tärkeimmät ja määritellään, soveltuvatko ne tietovaraston ulottuvuusattribuuteiksi vai mittauskohteiksi. Menetelmässä ei tarvita faktojen määrittelyä lainkaan, vaan tietovaraston faktakaavion perustana toimivat mittauskohteet, joille keksitään yhteinen nimi. Esimerkkitapauksessa (KUVIO 17) mittauskohteet saldo, liikevaihto, luottoraja ja korko nimettiin tilifaktoiksi.

Moodyn ja Kortinkin (2000) menetelmässä valmisteluvaiheessa suoritettujen ER-kaavion kohdetyyppien luokittelun perusteella faktoiksi valitaan tärkeimmät tapahtumakohdetyypit. Mittauskohteeksi valitaan yleensä jokin summattava attribuutti, esimerkiksi myyntien summan tai lukumäärän laskeva attribuutti. Tässä menetelmässä faktojen valinta on hyvin pitkälle automatisoitu eikä suunnittelijan panos ole niin oleellinen kuin muissa menetelmissä. Toisaalta valmisteluvaiheen luokittelussa suunnittelijan asiantuntemusta on jo tarvittu.

## **Ulottuvuuksien valinta**

Ulottuvuuksien valinnalla tarkoitetaan niitä tehtäviä, jotka on suoritettava päätettäessä, mitkä globaalin käsitekaavion attribuuteista valitaan tietovaraston käsitekaavion ulottuvuuksiksi.

Golfarellin ym. (1998) menetelmässä ulottuvuuksien määrittely tapahtuu attribuuttipuun karsimisen ja jalostamisen jälkeen. Ulottuvuuksiksi kelpaavat sellaiset attribuutit, joilla on suora yhteys attribuuttipuun juureen. Yleensä kaikki attribuuttipuun juureen suorassa yhteydessä olevat attribuutit valitaan ulottuvuuksiksi ja tämä tekeekin vaiheen suorittamisen helpoksi.

Cabibbon ja Torlonen (1998) menetelmä ei anna kovin hyviä edellytyksiä ulottuvuuksien tunnistamiselle. Ulottuvuuksien tunnistaminen tapahtuu menetelmän ensimmäisessä vaiheessa, kun faktat ja mittauskohteet on ensin valittu. Ulottuvuudet on tunnistettava sen perusteella, että ne ovat sellaisia ER-kaavion osia, jotka kuvaavat analysoitavan faktan tietoja päätöksenteon kannalta kiinnostavista näkökulmista.

Böhnleinin ja Ulbrich-vom Enden (1999) menetelmässä ulottuvuuksien määrittely tapahtuu menetelmän toisessa vaiheessa. Ohjeeksi annetaan SER-kaavion tarkastelu mittauskohteista katsoen vasemmalle ja pyrkimys löytää sellaiset tieto-objektityypit, jotka ovat tietovaraston tietojen analysoinnin kannalta merkittäviä. Muita ohjeita ulottuvuuksien tunnistamiseen ei anneta, joten suunnittelijan asiantuntemus tietovaraston aihepiiristä on välttämätöntä.

Hüsemannin ym. (2000) menetelmässä ulottuvuuksien määrittely tapahtuu mittauskohteiden kontekstien määrittelyn yhteydessä. Ensin selvitetään, mistä attribuuteista mittauskohteet riippuvat funktionaalisesti. Jäljitystä jatketaan, kunnes löydetään ne attribuutit, jotka eivät ole funktionaalisesti riippuvia mistään attribuuteista. Nämä attribuutit valitaan ulottuvuuksiksi. Tässä menetelmässä ulottuvuuksien tunnistaminen tapahtuu siis selvien sääntöjen mukaan, joten tämän vaiheen suorittamiseen ei välttämättä tarvita tietovaraston aihepiirin asiantuntemusta. Apuna voi

käyttää myös valmisteluvaiheena olleen tietovaraston vaatimusmäärittelyn yhteydessä tuotettua taulukkoa, josta selviää, soveltuuko attribuutti ulottuvuusattribuutiksi vai mittauskohteeksi.

Moodyn ja Kortinkin (2000) menetelmässä ulottuvuuksien tunnistaminen on helppoa, koska valmisteluvaiheessa kohdetyypit on luokiteltu. Komponenttikohdetyypeiksi luokitelluista kohdetyypeistä tulee ulottuvuuksia.

### **Hierarkioiden määrittely**

Hierarkioiden määrittelyllä tarkoitetaan niitä tehtäviä, jotka on suoritettava valittaessa tietovaraston ulottuvuuksien hierarkioita eli ulottuvuusattribuuteista koostuvia kokonaisuuksia.

Golfarellin ym. (1998) menetelmässä hierarkioiden määrittely tapahtuu varsin suoraviivaisesti. Attribuuttipuussa välillisesti tai välittömästi ulottuvuuksiin yhdistyvät attribuutit puun lehtisolmuun asti muodostavat kunkin ulottuvuuden hierarkian.

Cabibbon ja Torlonen (1998) menetelmässä hierarkioihin kuuluvia osia poimitaan ER-kaaviosta jo ulottuvuuksien tunnistamisen yhteydessä. ER-kaavion uudelleenjärjestelyn aikana tarvittavia kohdetyyppejä voidaan vielä lisätä kaavioon haluttujen hierarkioiden aikaansaamiseksi.

Böhnleinin ja Ulbrich-vom Enden (1999) menetelmässä ulottuvuushierarkiat tunnistaa tarkastelemalla SER-kaaviota ulottuvuuksiksi valituista tieto-objektityypeistä vasemmalle ja valitsemalla kuhunkin hierarkiaan ulottuvuuteen yhteydessä olevat tieto-objektityypit.

Hüsemannin ym. (2000) menetelmässä valittuja ulottuvuuksia lähtökohtina pitäen etsitään kaikki kuhunkin ulottuvuuteen funktionaalisten riippuvuuksien perusteella kuuluvat attribuutit. Ominaisuusattribuutit tulee erotella ulottuvuusattribuuteista. Apuna

voi käyttää vaatimusmäärittelyn yhteydessä tuotettua taulukkoa, josta selviää, soveltuuko attribuutti ulottuvuusattribuutiksi vai mittauskohteeksi.

Moodyn ja Kortinkin (2000) menetelmässä valmisteluvaiheessa tehdyn kohdetyyppien luokittelun jälkeen hierarkioiden tunnistaminen on helppoa. Kaikki luokittelukohdetyypit kuuluvat hierarkioihin.

### **Summaamisesta päättäminen**

Summaamisesta päättämisellä tarkoitetaan niitä ohjeita, joita menetelmissä on esitetty, kun on tarkoitus päättää tietovaraston ulottuvuushierarkioiden summauskohteista ja -rajoitteista.

Böhleinin ja Ulbrich-vom Enden (1999) menetelmän viimeisessä vaiheessa pyritään määrittelemään ulottuvuushierarkioiden rajoitteita, jotka koskevat tietojen summaamista. Hüsemanın ym. (2000) menetelmässä summausrajoitteiden määrittely on oma vaiheensa. Siinä tuotetaan taulukkomuotoinen liite, josta selviävät kunkin mittauskohteen ulottuvuustasot ja ulottuvuustasojen rajoitustasot. Golfarelli ym. (1998), Cabibbo ja Torlone (1998) sekä Moody ja Kortink (2000) eivät ota lainkaan kantaa tietojen summaamiseen.

### **4.3 Menetelmien subjektiivista arviointia ja parannusehdotuksia**

Menetelmien subjektiivisessa arvioinnissa pyritään esittämään menetelmien vahvuuksia ja heikkouksia eri näkökulmista sekä esittämään mahdollisiin heikkoihin kohtiin parannusehdotuksia. Menetelmiä tarkastellaan niiden helppokäyttöisyyden, käytettävyyden ja annetun ohjeistuksen perusteella.

Golfarellin ym. (1998) menetelmässä vaiheiden suorittaminen on ohjeistettu melko hyvin ja attribuuttiin karsimisen jälkeiset vaiheet ovat helpohkoja suorittaa. Suurin ongelma on oikeiden faktojen ja fakta-attribuuttien valinta. Niiden määrittelyyn tarvittaisiin hankalimpien tapausten kohdalla lisäohjeita. Lisäksi lähtömallina olevan globaalin käsitekaavion erikoisrakenteiden käsittelyyn tarvittaisiin ohjeita. Esimerkiksi sitä, miten attribuuttiin rakentamisessa toimitaan yleistysrakenteen kohdalla, ei ole ohjeistettu. Attribuuttiin kasvavat hyvin helposti laajoiksi ja monimutkaisiksi, varsinkin jos attribuutteja on paljon ja kaikkien attribuuttien nimet tulee kirjoittaa näkyviin. Ehkä jo attribuuttiin rakentamisvaiheessa pitäisi suorittaa karsintaa ja jalostusta jättämällä selvästi tarpeettomia attribuutteja pois. Kuitenkin attribuuttiin vie vähemmän tilaa kuin vastaava Golfarellin ym. (1998) menetelmässä käytetyllä notaatiolla tehty ER-kaavio, joten attribuuttiin käyttö on hyvin perusteltua.

Cabibbon ja Torlonen (1998) menetelmän terminologiassa rajanveto hämärtyy tietovaraston käsitteellisen suunnittelun ja loogisen suunnittelun välillä. Menetelmällä tuotettavaa mallia kutsutaan tietovaraston loogiseksi malliksi, vaikka lopputuloksena saatava malli vastaa tietovaraston käsittemallia. (Hüsemann ym., 2000.) Faktojen ja ulottuvuuksien määrittelyyn ei anneta tarpeeksi tarkkoja ohjeita. Menetelmässä ei myöskään neuvota, miten pitäisi toimia lähtömallina toimivan käsitekaavion erikoisrakenteiden, esimerkiksi yleistyksen kohdalla, ulottuvuusgraafin johtamisen yhteydessä. Mahdollisesti yleistysrakenteista pitäisi pyrkiä pääsemään eroon jo käsitekaavion uudelleenjärjestelyvaiheessa.

Böhnleinin ja Ulbrich-vom Enden (1999) menetelmän lähtömallina edellyttämä SER-malli helpottaa tietovaraston käsitekaavion johtamista, koska menetelmän vaiheiden aikana usein neuvotaan tarkastelemaan SER-kaaviota tietystä tieto-objektityypistä vasemmalle ja näin löytämään tarvittavat tieto-objektityypit. Näin menetellään muun muassa ulottuvuuksien ja hierarkioiden tunnistamisen yhteydessä. SER-malli on varmaankin useimmille vieraampi kuin perinteinen ER-malli, mutta sen rakenne on kuitenkin melko helppo oppia. Attribuutit eivät kaaviossa näy ja joskus tästä voi olla haittaa, mutta laajojen kaavioiden yhteydessä siitä on ennemminkin hyötyä.

Menetelmässä mittauskohteiden tunnistaminen tapahtuu muista menetelmistä poikkeavalla tavalla, ja tapa vaikuttaakin toimivalta. Tietovaraston päämääriä lähtökohtina käyttäen tunnistetaan päämääriin johtavat palvelut ja palveluihin liittyvät mitattavat liiketoimintakohteet. Kun nämä taustatekijät on selvitetty, on vasta mittauskohteiden määrittelyn aika. Hankalimmaksi tehtäväksi saattaa koitua mittauskohteiden olemassaoloehtojen määrittely, vaikkakin tehtävän suorittamiseen on annettu ohjeet. Olemassaoloehtojen määrittelyn jälkeen ulottuvuuksien ja hierarkioiden määrittely tapahtuu melko helposti.

Hüsemannin ym. (2000) menetelmässä vaiheet on ohjeistettu varsin hyvin. Alussa tapahtuvaa mittauskohteiden ja ulottuvuusattribuuttien tunnistamista seuraavat vaiheet suoritetaan tiettyjen algoritmien mukaan. Lopputulosmallina saatava yleistetyssä moniulotteisessa normaalimuodossa oleva faktakaavio on laadittu annettujen kriteerien mukaan. Kriteerien avulla lopputulosmallin laatua voidaan myös arvioida. Hankaluuksia voi aiheuttaa se, että yleistetty moniulotteinen normaalimuoto on monille vieras käsite ja sen ymmärtäminen vaatii syvällistä tietämystä sekä normaalimuodoista että tietovarastojen suunnittelusta.

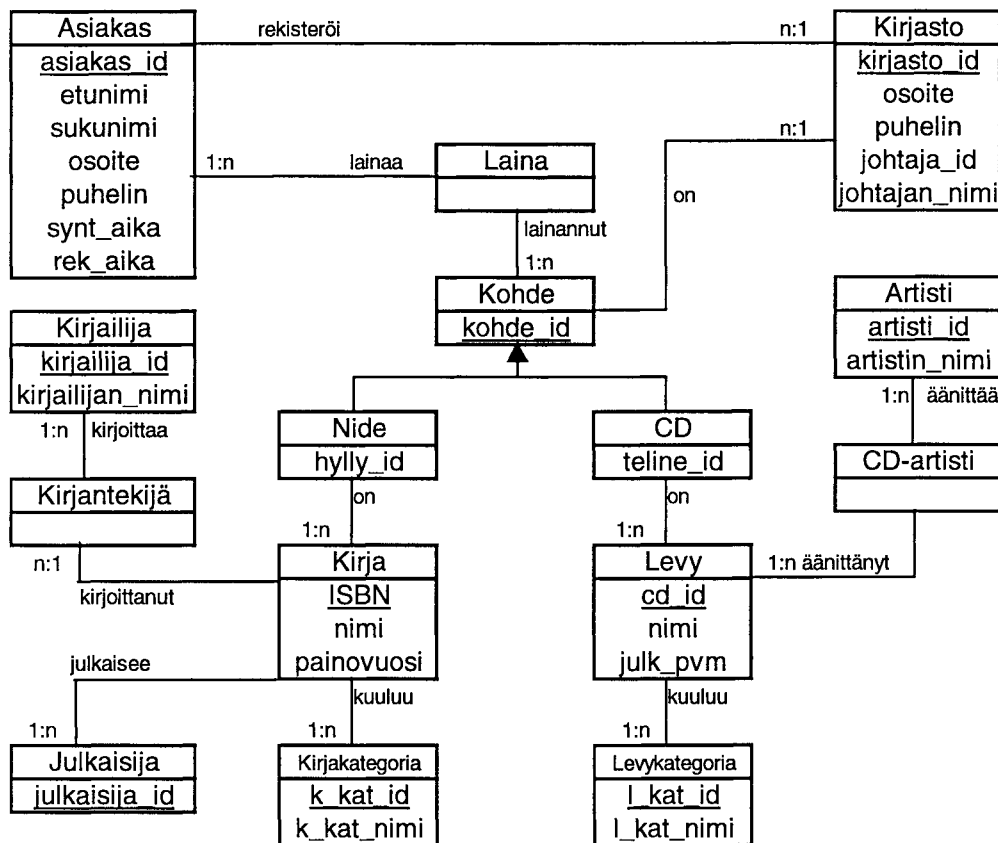
Menetelmän lopputuloksena saatava faktakaavio on tehty omaperäisellä notaatiolla, jota ei missään muussa menetelmässä käytetä. Menetelmän ytimenä olevaa yleistettyä moniulotteista normaalimuotoa olisi pitänyt soveltaa perinteiseen tähtimalliin. Nyt yleistetty moniulotteinen normaalimuoto yhdistettynä omaperäisellä notaatiolla toteutettuun tietovaraston faktakaavioon edellyttää menetelmän mahdolliselle käyttäjälle paljon opittavaa.

Moodyn ja Kortinkin (2000) menetelmässä ensimmäiseksi suoritettava kohdetyyppien luokittelu helpottaa jatkovaiheiden tehtäviä oleellisesti. Luokittelun jälkeen tietovaraston aihepiirin asiantuntemusta ei välttämättä tarvita, paitsi päätettäessä mikä kaavio otetaan tietovaraston käsitekaavion pohjaksi. Menetelmän vahvuus onkin useissa mahdollisissa lopputulosmalleissa, joista voidaan aina valita tilanteeseen sopiva vaihtoehto. Kuitenkaan Moody ja Kortink (2000) eivät anna ohjeita, millaiseen tilanteeseen kukin lopputulosmalli parhaiten soveltuisi. Epäselväksi jää myös se, miten



menetelmä toimisi, jos operatiivisia järjestelmiä kuvaava globaali käsitekaavio sisältäisi erikoisrakenteita, esimerkiksi yleistysrakenteita.

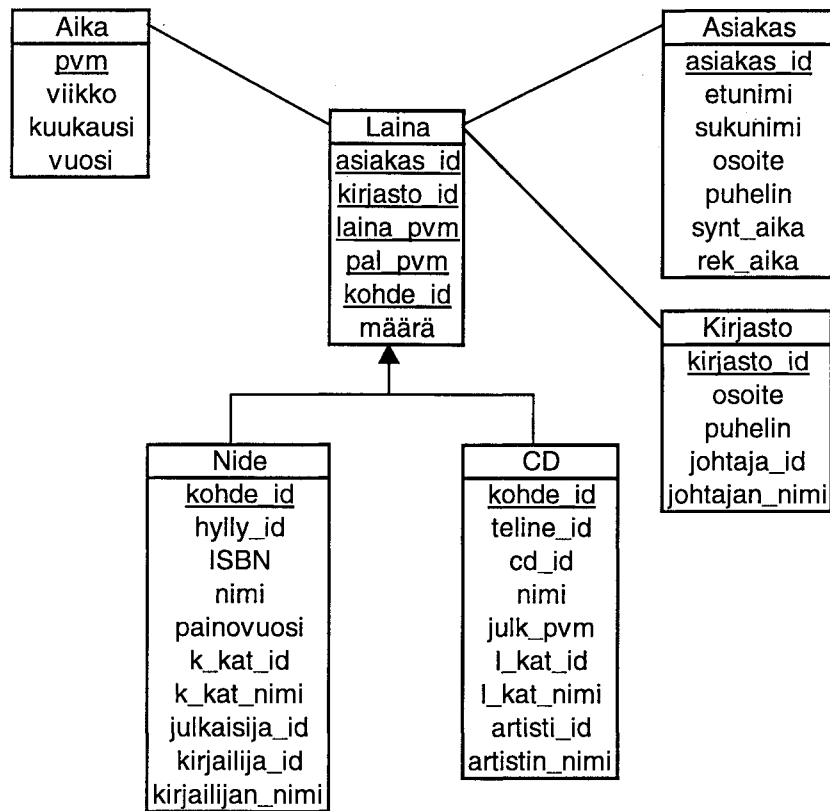
Golfarellin ym. (1998), Cabibbon ja Torlonen (1998) sekä Moodyn ja Kortinkin (2000) menetelmissä lähtömallina käytetään ER-mallia ja lopputulosmallina voidaan käyttää tähtimallia, mutta minkään menetelmän kohdalla ei ole ohjeistettu ER-kaaviossa mahdollisesti esiintyvän yleistysrakenteen näkymistä tietovaraston tähtimallissa. Niinpä seuraavassa on otettu esimerkiksi melko yksinkertaisen kirjastojärjestelmän käsitekaavio (KUVIO 25), joka sisältää yleistysrakenteen ja muunneltu tuo ER-kaavio tähtimalliksi (KUVIO 26) noudattaen soveltuvin osin Moodyn ja Kortinkin (2000) menetelmän ohjeistusta.



KUVIO 25. Kirjastojärjestelmän ER-kaavio (vrt. Connolly ja Begg, 2000).

Yleistysrakenteen kohdalla on menetelty siten, että se on siirretty sellaisenaan tähtimalliin ja sitä käsitellään yhtenä ulottuvuutena. Käytännössä ulottuvuuteen kuuluu kerrallaan jompikumpi yleistysrakenteen muodostavista tauluista, nide tai CD. Esimerkkitapauksen kirjastojärjestelmää tulee tarkastella vain esimerkkinä, koska se ei luultavasti todellisuudessa ole sellainen järjestelmä, josta tietovarastoa rakennettaisiin.

Kuvatuissa menetelmissä ei oltu myöskään otettu huomioon sitä, että kun ulottuvuuksien hierarkioita muodostetaan, niin tietyn hierarkian muodostavien kohdetyyppien ketju ei aina olekaan (n:1) –suhteiden ketju. Kuvion 25 ER-kaaviossa tällainen tilanne on kirjantekijä-kohdetyypin kohdalla. Kirjalla voi olla yksi tai useampi kirjoittaja. Tällöin tähtimallia ajatellen ensimmäinen vaihtoehto on hyväksyä kirjantekijäksi vain yksi kirjailija, joka ainoastaan näkyisi tietovaraston nide-ulottuvuudessa. On kuitenkin yleistä, että kirjalla on kaksi tai useampi kirjoittaja, ja voidaan haluta kaikkien kirjantekijöiden nimet mukaan tietovarastoon. Tällöin voidaan menetellä siten, että nide-ulottuvuudessa kaikkien kirjantekijöiden tunnukset on lueteltu peräkkäin kirjailija\_id-kohdassa ja kaikkien kirjantekijöiden nimet kirjailijan\_nimi-kohdassa.



KUVIO 26. Kirjastojärjestelmän ER-kaaviosta tuotettu tähtimalli.

Moodyn ja Kortinkin (2000) menetelmä vaikutti kuvatuista parhaimmalta, koska siinä vaiheet on ohjeistettu riittävän selkeästi ja muutenkin menetelmä vaikutti helppokäyttöisimmältä. Menetelmä on lisäksi monipuolinen, koska on tarjolla monta erilaista lopputulosmallia. Cabibbon ja Torlonen (1998) menetelmä vaikutti kuvatuista taas heikoimmalta. Siinä vaiheet on ohjeistettu liian epämääräisesti ja tietovaraston käsitteellisen ja loogisen suunnittelun sekoittuminen keskenään hankaloittaa entisestään menetelmän ymmärtämistä.

Lopuksi voidaan miettiä, onko mahdollista kehittää uusi menetelmä, johon on yhdistelty kuvattujen menetelmien parhaat puolet. Tällaisen menetelmän lähtömallina toimisi Hüsemannin ym. (2000) sekä Kortinkin ja Moodyn (2000) käyttämä malli, joka on mielestäni selkein. Lopputulosmalliksi voitaisiin taas valita mikä tahansa Kortinkin ja Moodyn (2000) esittämistä malleista tai vaihtoehtoisesti Hüsemannin ym. (2000)

käyttämä faktakaavio yleistetyssä moniulotteisessa normaalimuodossa. Välitulosmalleja ei mielestäni tarvittaisi, paitsi siinä tapauksessa, että ER-kaavio on niin laaja, että Golfarellin ym. (1998) käyttämän attribuuttipuun avulla sitä voitaisiin selkeyttää. Mielestäni Kortinkin ja Moodyn (2000) menetelmässä on kuitenkin niin paljon vahvuuksia, että sitä voidaan käyttää malliesimerkkinä hyvästä tietovaraston käsitekaavion johtamismenetelmästä.

## 5. YHTEENVETO

Tämän tutkielman tavoitteena on ollut tietovaraston käsittekaavion johtamiseen operatiivisten järjestelmien käsittekaavioista kehitettyjen menetelmien ja mallien kuvaaminen, vertailu ja arviointi sekä parannusehdotusten esittäminen.

Liiketoiminnassa menestyminen vaatii nopeita päätöksiä saatavilla olevan tiedon perusteella. Tietovarastojen käyttö yleistyi 1990-luvun lopulla, koska operatiiviset järjestelmät eivät kyenneet tarjoamaan yritysjohdolle tarpeellista tietoa käyttökelpoisessa muodossa. Tietovarasto tarjosi mahdollisuuden kerätä tietoa operatiivisista järjestelmistä ja mahdollisesti ulkopuolisista lähteistä ja integroida nämä tiedot yhtenäiseksi kokonaisuudeksi tietovarastoon.

Tietovaraston on kirjallisuudessa todettu mahdollistavan integroitujen tietojen tehokkaan analysoinnin, raportoinnin ja kyselyt, jotka ovat tarpeellisia yritysjohdolle liiketoimintapäätösten tekemisessä. Tietovaraston oikeanlaisen käytön nähdään parantavan yrityksen menestysmahdollisuuksia liikemaailmassa.

Tietovaraston suunnittelu on usein vaativa tehtävä ja se on suoritettava tarkkaan harkiten. Tietovaraston suunnittelumenetelmissä suunnittelu on jaettu useaan vaiheeseen. Golfarellin ja Rizzin (1998) mukaan vaiheet ovat operatiivisen järjestelmän analysointi, vaatimusten määrittely, käsitteellinen suunnittelu, kuormituksen tarkentaminen ja kaavion vahvistaminen, looginen suunnittelu ja fyysinen suunnittelu.

Tietovaraston käsitteellisen suunnittelun tavoitteena on tuottaa tietovaraston rakennetta kuvaava moniulotteinen käsittekaavio. Tämä voidaan suorittaa kolmen vaihtoehdoisen lähestymistavan mukaisesti: alhaalta ylös, ylhäältä alas tai keskeltä ulos. Alhaalta ylös -lähestymistavan mukaan tietovaraston käsittekaavion lähtökohtana toimivat olemassa olevien operatiivisten järjestelmien käsittekaaviot. Kirjallisuudessa on esitetty useita menetelmiä tietovaraston käsittekaavion johtamiseksi operatiivisten järjestelmien

käsitekaavioista. Tässä tutkielmassa on otettu tarkasteltaviksi Golfarellin ym. (1998), Cabibbon ja Torlonen (1998), Böhnleinin ja Ulbrich-vom Enden (1999), Hüseemannin ym. (2000) sekä Moodyn ja Kortinkin (2000) menetelmät. Tarkastelun tavoitteena on ollut kuvata kootusti olemassa olevia menetelmiä, jonka jälkeen menetelmiä on arvioitu ja vertailtu ja pyritty esittämään niihin parannusehdotuksia. Jokaisessa menetelmässä lähtömallina toimii operatiivisista järjestelmistä tuotettu globaali käsitekaavio, joka vaihe vaiheelta muunnetaan tietovaraston moniulotteiseksi malliksi.

Golfarellin ym. (1998) menetelmässä lähtömallina toimivasta käsitekaaviosta johdetaan ensin attribuuttipuun, jota jatkossa karsitaan ja jalostetaan. Attribuuttipuusta johdetaan lopulta faktakaavioista koostuva tietovaraston moniulotteinen malli. Koska lähtömallina käytetty notaatio saattaa varsinkin laajojen ER-kaavioiden yhteydessä muodostua hyvin monimutkaiseksi, on rakenteeltaan ER-kaaviota yksinkertaisemmasta attribuuttipuun käytöstä tämän menetelmän yhteydessä selvästi hyötyä. Vaiheiden suorittamiseen on annettu melko hyvät ohjeet.

Cabibbon ja Torlonen (1998) menetelmässä lähtömallista johdetaan ensin ulottuvuusgraafi, joka muunnetaan tietovaraston moniulotteiseksi malliksi. Menetelmän ymmärtämistä vaikeuttaa tietovaraston käsitteellisen ja loogisen suunnittelun rajanvedon hämärtyminen. Lisäksi vaiheiden suorittaminen on liian heikosti ohjeistettu.

Böhnleinin ja Ulbrich-vom Enden (1999) SER-notaatiolla tuotettu kaavio muunnetaan vaihe vaiheelta tietovaraston moniulotteiseksi tähtimalliksi. Lähtömallina käytettävä SER-malli on monille varsin tuntematon, mikä vaikeuttaa menetelmän ymmärtämistä. SER-malli on kuitenkin helppo omaksua ER-mallin pohjalta ja sen käyttö helpottaa joidenkin vaiheiden suorittamista menetelmän aikana.

Hüseemannin ym. (2000) menetelmän lopputulosmallina on yleistetyssä moniulotteisessa normaalimuodossa oleva tietovaraston käsitekaavio. Lopputulosmalli on monelle varsin tuntematon ja se vaikeuttaa menetelmän ymmärtämistä, vaikkakin vaiheiden suorittaminen on ohjeistettu varsin hyvin.

Moodyn ja Kortinkin (2000) menetelmässä lähtömallin kohdetyypit luokitellaan aluksi. Tästä on jatkossa hyötyä, kun romahduttamista ja summaamista apuna käyttäen tuotetaan tietovaraston moniulotteinen malli. Lopputulosmalleja on valittavissa viisi erilaista. Monipuoliset lopputulosmallit ovatkin menetelmän vahvuus. Lisäksi menetelmän ensimmäisenä vaiheena suoritettava kohdetyyppien luokittelu helpottaa suuresti jatkovaiheiden suorittamista. Muutenkin vaiheiden suorittaminen on ohjeistettu hyvin.

Tähtimallia lopputulosmallina käytävissä menetelmissä ei ole ohjeistettu, miten pitäisi toimia ER-kaavion erikoisrakenteiden, esimerkiksi yleistysrakenteen, kohdalla. Tästä syystä neljännessä luvussa on esitetty, miten yksinkertainen yleistysrakenteen sisältävä ER-kaavio voidaan muuntaa tähtimalliksi siten, että yleistysrakenne tulee näkyviin myös tähtimallissa.

Tutkielman tuloksena olevat arviot menetelmistä hyödyttävät tietovarastosuunnittelua tuntevia ammattilaisia. Jos tietovarastosuunnittelun lähestymistavaksi on valittu alhaalta ylös, niin tulokset auttavat valitsemaan parhaan tai käyttökelpoisimmalta vaikuttavan menetelmän tietovaraston käsitekaavion johtamiseksi operatiivisten järjestelmien globaalista käsitekaaviosta.

Jatkotutkimusaiheeksi sopii tietovaraston moniulotteisten mallien vertailu ja niiden soveltuvuuden tarkastelu erilaisten tietovarastotyyppien yhteydessä. Ylhäältä alas -lähestymistapaan kehitettyjen menetelmien kuvaus ja vertailu on myös yksi mahdollinen jatkotutkimusaihe.

## LÄHTEET

Adriaans P., Zantinge D. 1996. Data Mining. Harlow: Addison-Wesley.

Anahory S., Murray D. 1997. Data Warehousing in the Real World – A Practical Guide for Building Decision Support Systems. Harlow: Addison-Wesley.

Batini C., Ceri S., Navathe S. 1992. Conceptual Database Design: An Entity-Relationship Approach. Redwood City: Benjamin/Cummings.

Berson A., Smith S.J. 1997. Data Warehouse Components. Julkaisussa The Data Administration Newsletter [online], 1(3) [viitattu 26.5.2001]. Saatavilla [www-muodossa <http://www.tdan.com/i003fe11.htm>](http://www.muodossa.com/i003fe11.htm).

Böhnlein M., Ulbrich-vom Ende A. 1999. Deriving Initial Data Warehouse Structures from the Conceptual Data Models of the Underlying Operational Information Systems. Teoksessa I-Y. Song, T. Teorey (toim.) Proceedings of the ACM 2<sup>nd</sup> International Workshop on Data Warehousing and OLAP, Kansas City, Missouri, November 2-6. New York: ACM Press, 15-21.

Cabibbo L., Torlone R. 1998. A Logical Approach to Multidimensional Databases. Teoksessa H-J. Schek, F. Saltor, I. Ramos, G. Alonso (toim.) Proceedings of the 6<sup>th</sup> International Conference on Extending Database Technology [online], 6(1) [viitattu 26.5.2002]. Saatavilla [www-muodossa <http://citeseer.nj.nec.com/cabibbo98logical.html>](http://citeseer.nj.nec.com/cabibbo98logical.html).

Chen P. 1976. The Entity-Relationship Model – Towards a Unified View of Data. ACM Transactions of Database Systems 1(1), 9-36.



Connolly T., Begg C. 2000. Database solutions: a step-by-step guide to building databases. Harlow: Addison-Wesley.

Devlin B. 1997. Data Warehouse: from Architecture to Implementation. Harlow: Addison-Wesley.

Everest G. C. 1986. Database Management: Objectives, System Functions, and Administration. New York: McGraw-Hill.

Golfarelli M., Maio D., Rizzi S. 1998. Conceptual Design of Data Warehouses From E/R Schemes. Julkaisussa H. El-Rewini (toim.) Proceedings of the 31<sup>st</sup> Hawaii International Conference on System Sciences [online], 7(1) [viitattu 26.5.2002]. Saatavilla [www-muodossa <http://citeseer.nj.nec.com/golfarelli98conceptual.html>](http://citeseer.nj.nec.com/golfarelli98conceptual.html).

Golfarelli, M., Rizzi, S. 1998. A methodological framework for data warehouse design. Teoksessa I-Y. Song, T. Teorey (toim.) Proceedings of the ACM First International Workshop on Data Warehousing and OLAP, Washington, November 7. Bethesda: ACM, 3 - 9.

Gross H. 1992. Eine Semantiktreue Transformation vom Entity-Relationship-Modell in das Strukturierte Entity-Relationship-Modell. Bamberger Beiträge zur Wirtschaftsinformatik 1(9).

Hasan H., Hyland P., Dodds D., Veeraraghavan R. 2000. Approaches to the Development of Multi-Dimensional Databases: Lessons from Four Case Studies. The Data Base for Advances in Information Systems 31(3), 10-23.

Hovi A. 1997. Data Warehousing – Tietovarastotekniikka. Espoo: Suomen Atk-kustannus Oy.

Hovi A., Koistinen H., Ylinen J. 2001. Tietovarastot liiketoiminnan tukena. Helsinki: Satku.

Hüsemann B., Lechtenböcker J., Vossen G. 2000. Conceptual Data Warehouse Design. Teoksessa M.A. Jeusfeld, H. Shu, M. Staudt, G. Vossen (toim.) Proceedings of 2<sup>nd</sup> International Workshop on Design and Management of Data Warehouses [online], 28(1) [viitattu 26.5.2002]. Saatavilla [www-muodossa <http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-28/>](http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-28/).

Inmon W. 1995. The Operational Data Store. Julkaisussa InfoDB [online], 9(1) [viitattu 26.5.2002]. Saatavilla [www-muodossa <http://www.dbaint.com/InfoDBDW.html>](http://www.dbaint.com/InfoDBDW.html).

Inmon W. 1996. Building the Data Warehouse. New York: Wiley & Sons.

Jarke M., Lenzerini M., Vassiliadis P., Vassiliou Y. 2000. Fundamentals of Data Warehouses. Berliini: Springer.

Kimball R. 1996. The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses. New York: Wiley & Sons.

Kimball R. 1997. A Dimensional Modeling Manifesto. Julkaisussa DBMS Magazine [online], 10(9) [viitattu 26.5.2002]. Saatavilla [www-muodossa <http://www.dbmsmag.com/9708d15.html>](http://www.dbmsmag.com/9708d15.html).

Laine H., Rantanen J. 1997. Tietovarastosanasto. Systeemyö 3-4, 2-3.

Lehner W., Albrecht J., Wedekind H. 1998. Normal Forms for Multidimensional Databases. Teoksessa M. Rafanelli, M. Jarke (toim.) Proceedings of the 10th International Conference on Scientific and Statistical Database Management Capri, Italy, July 1-3. Los Alamitos: IEEE Computer Society, 63-72.

Lehtonen J., Leinonen T. 1991. Tietovarastotekniikan perusteet. 3. uusittu painos. Helsinki: VAPK-kustannus.

Moody D., Kortink M. 2000. From Enterprise Data Models to Dimensional Models: A Methodology for Data Warehouse and Data Mart Design. Teoksessa M.A. Jeusfeld, H. Shu, M. Staudt, G. Vossen (toim.) Proceedings of 2<sup>nd</sup> International Workshop on Design and Management of Data Warehouses [online], 28(1) [viitattu 26.5.2002]. Saatavilla [www-muodossa](http://www.muodossa) <<http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-28/>>.

Rumbaugh J., Blaha M., Premerlani W., Eddy F., Lorensen W. 1991. Object-Oriented Modeling and Design. Englewood Cliffs: Prentice-Hall.

Simon A., Shaffer S. 2001. Data Warehousing and Business Intelligence for E-Commerce. San Francisco: Kaufmann.

Sinz E. 1988. Das Strukturierte Entity-Relationship Modell (SER-Modell). Angewandte Informatik 30(5), 191-202.

Smith A.M. 1997. Data Warehousing: A Short Overview. Julkaisussa The Data Administration Newsletter [online], 1(3) [viitattu 26.5.2002]. Saatavilla [www-muodossa](http://www.muodossa) <<http://www.tdan.com/i003fe09.htm>>.

Tryfona N., Busborg F., Christiansen J. 1999. StarER: A Conceptual Model for Data Warehouse Design. Teoksessa Proceedings of the ACM 2<sup>nd</sup> International Workshop on Data Warehousing and OLAP, Kansas City, Missouri, November 2-6. New York: ACM Press, 3-8.

Vassiliadis P., Bouzeghoub M., Quix C. 2000. Towards Quality-Oriented Data Warehouse Usage and Evolution. Information Systems 25(2), 89-115.