.

Ville Niemijärvi

Metatieto tietovarastoympäristössä

Tietojärjestelmätieteen
pro gradu –tutkielma
20.12.2002

Jyväskylän yliopisto
Tietojenkäsittelytieteiden laitos
Jyväskylä

TIIVISTELMÄ

Niemijärvi, Ville-Markus Olavi
Metatieto tietovarastoympäristössä/Ville Niemijärvi
Jyväskylä: Jyväskylän yliopisto, 2002.
96 s.
Pro gradu –tutkielma

Tietovarastoinnilla tarkoitetaan käsitteiden, teknologioiden ja sovellusten joukkoa, joiden tarkoituksena on tarjota yhtenäinen näkymä liiketoimintatietoon päätöksenteon tukemista varten. Tutkimuksen tarkoituksena on selvittää mitä metatieto on tai mitä sen pitäisi olla tietovarastoympäristössä.

Metatieto nähdään tietovarastoinnissa välttämättömyytenä, jopa elinehtona. Silti alalla vallitsee epätietoisuus käsitteen laajuudesta ja merkityksestä. Lisäksi metatiedon käyttöön, hallintaan sekä vaatimusmäärittelyyn liittyy epätietoisuutta. Aiheen käsittely tässä tutkielmassa perustuu kirjallisuudesta tehtyjen havaintojen kokoamiseen ja analysointiin, joten tutkimus on käsitteellis-teoreettinen.

Tämän tutkimuksen keskeisenä tuloksena on perusteellinen kuvaus metatiedosta tietovarastoissa. Tutkielmassa analysoidaan mitä eri tyyppistä tietoa tietovarastoissa esiintyy ja tarkastellaan tiedon ja metatiedon eroa tietovarastokontekstissa. Tutkimuksessa konstruoidaan luokittelu, josta ilmenee mitä metatietotyyppejä tietovarastoista voidaan mallintaa ja tallentaa. Tämän jälkeen tutkielmassa näytetään mikä merkitys metatiedolla on tietovaraston menestyksekkäässä käytössä. Lisäksi tutkimuksessa esitellään RDF-malli metatiedon esittämiseen ja arvioidaan sen soveltuvuutta tietovarastojen metatiedon kuvaamiseen.

AVAINSANAT: tietovarastointi, tietovarastoarkkitehtuuri, metatieto,
metatiedon tyypit, RDF

ABSTRACT

Niemijärvi, Ville-Markus Olavi
Metadata in Data Warehouse Environment/Ville Niemijärvi
Jyväskylä: University of Jyväskylä, 2002.
96 s.
Master's Thesis

Data warehousing is a collection of concepts, technologies and software aimed at enabling a consistent view on the business information to support decision making. The objective of this study is to find out what metadata is or what it should be in data warehouse environment.

Metadata is seen as a necessity in data warehousing. However there is uncertainty about the extent and meaning of the concept. Furthermore, there is uncertainty concerning the use, management and requirements spesification of metadata in data warehouse environment. This study approaches the topic by gathering and analysing material from previous studies and literature hence the study is theoretical in it's nature.

The main finding of this research is a thorough description of metadata in data warehouse environment. This research analyses the different types of information that can be found in data warehouses and examines the difference between data and metadata. First, a classification concerning the types of metadata that can be modeled and stored in data warehouses is constructed. Second, this study illustrates the meaning of metadata in successful use of data warehouse. Furthermore, this study introduces RDF model for presenting metadata and evaluates it's applicability in data warehouse context.

KEYWORDS: data warehousing, data warehouse architecture, metadata, metadata types, RDF

SISÄLLYSLUETTELO

1 JOHDANTO

Organisaatiot tarvitsevat oikeaa tietoa oikeaan aikaan oikeassa paikassa ja mielellään sopivilla kustannuksilla (Geiger, Inmon & Zachman 1997; Jarke, Lenzerini, Vassiliadis & Vassiliou 2000). Useissa organisaatioissa on valtavat määrät tietoa, mutta sen tehokas hyödyntäminen liiketoiminnan ja etenkin päätöksenteon tarpeisiin on usein ongelmallista. Tietovarastointi on suuntaus tarjota organisaatioille oikea-aikaista relevanttia tietoa päätöksenteon tueksi.

Tietovarastointi on liiketoimintana kasvanut merkittävästi viime aikoina (Gray & Watson 1998; Haley & Watson 1998) ja sen suosio on kasvanut lähes räjähdysmäisesti (Dittrich & Vaduva 2001). Goodhue, Watson ja Wixom (2002) näkevätkin tietovarastoinnin olevan yksi tärkeimmistä kehityssuunnista tietojärjestelmien alalla. Silti useat tutkijat näkevät, että tietovarastointia ei ole huomioitu riittävästi akateemisessa tutkimuksessa (Gray & Watson 1998; Haley & Watson 1998; Widom 1995) ja esimerkiksi Kim, Kim ja Lee (2001) painottavat akateemisen lähestymistavan tärkeyttä ohjelmisto-/laitteistosuuntautuneen sijaan. Goodhue ym. (2002) lisäävät, että lähes kaikki kirjallisuus koskien tietovarastointia on kirjoitettu yritysten puolesta eli lähes aina kyseessä on käytännöllinen näkökulma, ei akateeminen. Nämä huomiot motivoivat osittain tätä tutkimusta.

Pitkällä tähtäimellä organisaation elinvoimaisuus on suoraan suhteessa sen kykyyn järjestää tiedon jakaminen organisaatiossa (Hackathorn 1999). Haley ja Watson (1998) argumentoivatkin, että tietovarasto itsessään ei tuo arvoa organisaatiolle; todellinen arvo tulee tietovaraston sisältämän tiedon käytöstä. Pelkkä uusien teknologioiden haaliminen ei myöskään Braynerin ja Carneiron (2002) mukaan riitä tuomaan strategista kilpailuetua. Täytyy osata myös käyttää noita teknologioita tehokkaasti ja lisäksi käyttäjien tulee pystyä etsimään, löytämään ja tulkitsemaan informaatiota paremmin. Gardner (1998)

sanookin, että tietovarasto on hyödyllinen vain jos se tuo kilpailullista etua ts. tietovaraston tiedon tulee pystyä vastaamaan käyttäjien vaatimuksiin. Metatieto tietovarastoissa on tarkoitettu osittain juuri tätä varten: helpottamaan tietovaraston hallintaa ja hyödyntämistä (Kietz, Vaduva & Zücker 2001; Staudt, Vaduva & Vetterli 2000).

Metatiedon (metadata) määritellään yleisesti olevan tietoa tiedosta (esim. Gardner 1998, s. 59; Hackathorn 1999, s. 165; Jacob & Sen 1998, s. 31). Tietovarastoympäristössä metatiedon merkitys on kuitenkin laajempi. Täten hieman parempi ja kattavampi määritys metatiedolle on muun muassa Gardnerin (1998) seuraava havainto: metatieto voidaan käsittää kaikeksi tiedoksi, joka määrittää tietovarastossa esiintyvää kohdetta, kuten taulukkoa, saraketta, kyselyä, raporttia, liiketoimintasääntöä tai transformointialgoritmia. Dittrich ja Vaduva (2001) määrittävät metatiedon olevan mitä tahansa tietoa, jota voidaan käyttää tukemaan tietovaraston hallintaa ja tehokasta hyödyntämistä.

Metatieto on erittäin tärkeässä asemassa tietovarastoissa. Oikeastaan se nähdään nykyään tietovaraston elinehtona (Geiger ym. 1997) sekä tärkeimpänä menestystekijänä tietovarastointiprojekteissa (Staudt ym. 2000). Silti useat tutkijat ovat huomioineet, että käsite metatieto on ymmärretty hyvin erilailla. Metatiedon hallintaan, käyttötarkoitukseen ja vaatimuksiin liittyykin epäselvyyksiä (Dittrich & Vaduva 2001). Vaduvan ja Vetterlin (2001) mukaan vain harvat tietovarastointiin liittyvät tutkimusprojektit ovat keskittyneet metatietoon. Lisäksi useimmat kaupalliset sekä tieteelliset lähestymistavat metatiedon hallintaan ja kuvaamiseen tarjoavat vain rajoitettuja ratkaisuja eivätkä ota huomioon kaikkia olennaisia metatiedon tyyppejä (Müller, Rahm & Stöhr 1999).

Viime aikoina ovat useat tutkijat kuitenkin kiinnittäneet huomiota metatietoon ja etenkin sen hallintaan. Metatiedon merkitys on kyllä tiedostettu (mm. Geiger

ym. 1997; Widom 1995). Metatietoa tietovarastoympäristössä ovat käsitelleet tarkemmin mm. Brayner ja Carneiro (2002), Müller ym. (1999) sekä Jarke ym. (2000). Yhteistä aikaisemmille pyrkimyksille ymmärtää ja määrittää metatietoa on kuitenkin se, että mikään niistä ei ole kattavasti määrittänyt, mitä kaikkea metatietoa tietovarastoympäristöissä on. Tämä tutkimus keskittyy korjaamaan tämän puutteen ja näyttää konkreettisella tasolla ja kattavasti, mitä metatietoja tietovarastoista voi ja tulisi kuvata.

Tutkimusongelmat tässä tutkielmassa ovat seuraavat:

1. Mikä merkitys metatiedolla on tietovarastoympäristössä?

2. Soveltuuko RDF kuvaamaan tietovaraston metatietoja?

Ensimmäisestä ongelmasta voidaan johtaa osaongelmat, jotka ovat:

1.1 Mitä jako tieto ja metatieto merkitsevät tietovarastoissa?

1.2 Mitä metatieto-tyyppejä tietovarastoista löytyy?

Metatietotyyppien luokittelu ja viitekehyksen muodostaminen tässä tutkimuksessa perustuu aikaisempien tutkijoiden havaintojen synteesiin sekä omaan analyysiin tietovaraston arkkitehtuurista, komponenteista sekä koko ympäristön sisältämistä tiedoista. Täten tutkimus on käsitteellis-teoreettinen.

Käsitteellis-teoreettiseen tutkimukseen kuuluu usein kaksi vaihetta: analyysivaihe sekä synteesivaihe (Järvinen & Järvinen 2000). Tässä tutkimuksessa edellisiin vaiheisiin on lisätty niin sanottu evaluoiva eli arvioiva vaihe.

Analyysivaihe käsittää perustan tälle työlle eli tässä vaiheessa tuodaan esille tietovarastoinnin perusteet, joiden ymmärtäminen on välttämätöntä tutkimuksen jatkolle. Lisäksi analyysivaihe kattaa käsitteiden tieto ja metatieto analysoinnin ja erottelun tietovarastokontekstissa.

Synteesivaihe pohjautuu kirjallisuudesta tehtyjen havaintojen kokoamiseen. Siinä tuodaan esille metatiedon merkitys tietovarastoinnissa sekä kerätään viitekehyksen muotoon mahdollisimman kattava kategorisointi eri metatietotyypeistä esimerkkeineen.

Evaluoivassa vaiheessa arvioidaan miten RDF:n avulla voidaan kuvata tietovarastoinnin sisältämiä metatietotyyppejä.

Luvussa 2 esitellään osa tutkimuksessa tarpeellisista käsitteistä ja määritellään se konteksti, jossa liikutaan eli tietovarastoympäristö ja sen oleellisimmat komponentit ja siihen liittyvät prosessit. Luvun 2 taustalla on aiemmissa tutkimuksissa (Jarke ym. 2000; Vaduva & Vetterli 2001) tehty huomio, että kaikki tietovarastoympäristön komponentit käyttävät ja/tai tuottavat metatietoa. Tätä varten täytyy siis ensiksi identifioida mitä komponentteja, prosesseja, käyttäjiä jne. löytyy tietovarastoympäristöstä. Luvussa 3 keskitytään tunnistamaan tietovaraston sisältämän tiedon ominaisuuksia ja mitä eri tietotyyppejä tietovarastossa esiintyy. Tässä tehdään myös erottelu tiedon ja metatiedon välillä. Tämän jälkeen luvussa 4 näytetään, mitä metatietoa tietovarastoympäristössä esiintyy ja esitetään kirjallisuudessa ja aikaisemmissa tutkimuksissa tehtyjä luokitteluja metatiedolle. Lisäksi luvussa 4 analysoidaan ja tuodaan esimerkkien avulla esille, mikä merkitys metatiedolla on tietovaraston eri osa-alueilla. Tavoitteena on erityisesti tuoda esille, miksi metatietoa tarvitaan tietovaraston menestyksekkäässä käytössä, ylläpidossa ja kehittämisessä. Luvussa 5 esitellään RDF–malli, jonka sopivuutta tietovaraston metatietojen kuvaamisessa arvioidaan. Luvussa 6 esitellään tutkimuksessa esille tulleita keskeisiä huomioita ja johtopäätöksiä.

2 TIETOVARASTOYMPÄRISTÖ

Oikea-aikainen tieto on organisaatioille välttämättömyys. Hyvin hallittu ja hyödynnetty tieto tuo yritykselle todellisen kilpailuedun (Hovi 1997). Nonakan (1991) mukaan menestyvät organisaatiot ovat niitä, jotka pystyvät jatkuvasti tuottamaan uutta tietämystä, jakamaan sitä laajalti organisaation sisällä ja pystyvät nopeasti sisällyttämään tämän tietämyksen uusiin teknologioihin ja tuotteisiin.

Tietovarastointi on yksi suuntaus tarjota organisaatioille paremmat mahdollisuudet hallita tietojaan ja saavuttaa edellä mainittua kilpailuetua. Tämä kilpailuetu voidaan saavuttaa tarjoamalla organisaatioille yksi ja yhtenäinen näkymä heidän liiketoimintaansa (Gardner 1998), joka Hackathornin (1995) mielestä on tietovarastoinnin olennaisin tekijä.

Tässä luvussa määritellään tarkemmin tietovarastoympäristö. Ensiksi määritellään käsite tietovarastointi ja sen keskeisimmät ominaisuudet. Tämän jälkeen syvennytään tarkastelemaan millainen on tietovarastoarkkitehtuuri ja minkälaisia komponentteja sekä prosesseja siihen kuuluu. Lisäksi luvussa tehdään erottelu operatiivisten järjestelmien sekä tietovarastojen välillä. Tämä luku luo välttämättömän perustan seuraaville luvuille ja auttaa myöhemmissä luvuissa ymmärtämään paremmin miksi tietovarastossa on tietyn tyyppistä tietoa ja miksi metatieto on niin olennaista tässä kokonaisuudessa.

2.1 Tietovarastointi – tiedon jalostamista päätöksenteon tueksi

Tietovarastoinnille ei ole olemassa yksiselitteistä määritystä (Elmasri & Navathe 2000) ja useat tutkijat näkevät käsitteen tietovarastointi eri tavalla. Eri määritelmistä löytyy kuitenkin paljon yhteneväisyyksiä, joita tässä kohdassa pyritään tuomaan esille. Lisäksi tässä kohdassa tuodaan esille tietovarastoinnin asema suhteessa erilaisiin päätöksenteon tukijärjestelmiin.

Tietovarastointi (Data warehousing) voidaan Jarken ym. (2000) mukaan nähdä joukoksi teknologioita, joiden tarkoituksena on antaa tietotyöläiselle (ylempi johto, analyytikot ym.) paremmat mahdollisuudet tehdä parempia ja nopeampia liiketoimintapäätöksiä. Gardner (1998) sen sijaan määrittää tietovarastoinnin olevan prosessi (ei siis tuote), joka kokoaa ja hallitsee eri lähteistä kerättyä tietoa, tarkoituksena saada yksi yhtenäinen näkökulma liiketoimintaan.

Määriteltäessä tietovarastointia tulee huomioida kaksi näkökulmaa. Ensiksi voidaan erottaa varsinainen *tietovarasto (data warehouse)*, joka tarkoittaa enemmänkin paikkaa, jonne tietomassat tallennetaan. Tällöin se muistuttaa läheisesti tietokantajärjestelmää (mm. Devlin 1997; Gray & Watson 1998). Hackathorn (1995) sen sijaan painottaa järjestelmän dynaamisuutta ja kehottaakin tällöin käyttämään termiä tietovarastointi (data warehousing), jolloin tarkoitetaan enemmänkin niitä aktiviteetteja ja prosesseja, joissa tieto tuodaan edellä mainittuun tietovarastoon, jalostetaan merkityksellisempään muotoon sekä jaetaan myös loppukäyttäjille.

Jones (1998) ei myöskään näe tietovarastoinnin olevan tuote vaan enemmänkin tietojärjestelmäympäristö (IS environment). Hän toteaa, että tietovarastosta ei voida erottaa niitä lukuisia ohjelmia ja työkaluja, jotka ovat yhteydessä siihen. Tässä tutkimuksessa pääpaino on käsitteellä *tietovarastoympäristö*. Toisaalta se sisältää sekä tietokantapohjaisen tiedon säilytyspaikan (tietovarasto-näkökulma) että aktiviteetit, prosessit ja ohjelmat, jotka liikuttavat, muokkaavat sekä jakavat tietoa (tietovarastointi-näkökulma). Lisäksi tietovarastoympäristö-käsitteeseen voidaan tässä tutkimuksessa lukea kuuluvan myös järjestelmän kehittämiseen ja suunnitteluun liittyvät näkökulmat. Täten voidaan sanoa, että tietovarastoympäristö sisältää erityisiä menetelmiä, käsitteitä ja teknologioita, joiden avulla voidaan rakentaa ja ylläpitää varsinaista tietovarastoa. Tietovarastossa on kerättynä yhtenäiseen muotoon organisaation tietoja ja se

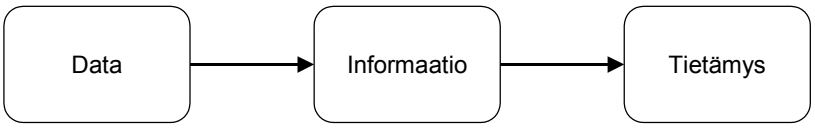
tarjoaa eri ohjelmistojen avulla yhtenäisen näkymän liiketoimintaan tukien täten päätöksentekoa.

Perimmäisenä tavoitteena tietovarastoinnilla on siis tietojen hallinta, yhtenäisen näkökulman tarjoaminen tuohon tietoon ja täten päätöksenteon tukeminen. Sinänsä idea organisaation tietoresurssien paremmasta hallinnasta, jakamisesta, yhtenäistämisestä ja näin ollen paremman kilpailuedun saavuttamisesta ei ole uusi. Voidaankin puhua myös niin sanotusta organisaation informaatio arkkitehtuurista (enterprise information architecture, EIA) (mm. Watson 2000) tai yleisemmin tietojärjestelmä arkkitehtuureista (IS architecture) (mm. Everest & Kim 1994), joiden molempien tavoitteena voidaan nähdä olevan organisaation suurten tietomassojen ja –resurssien hallinta sekä esimerkiksi tiedon parempi jakaminen. Tietovarastointi poikkeaa edellisistä siinä, että sen pääfunktio on erityisesti päätöksenteon tukeminen mutta myös se, että tietovarastoinnissa ei itsessään mallinneta organisaation tietoresursseja (tiedot, prosessit, ohjelmistot) kokonaisvaltaisesti vaan poimitaan päätöksenteon kannalta relevantimmat tiedot analysointia ja raportointia varten edellä määriteltyyn tietovarastoon.

Olennaista tietovarastoissa on siis niiden suhde päätöksenteon tukeen. Tietovarastot on suunniteltu pääosin päätöksenteon tukijärjestelmiä varten ja ne on optimoitu tiedon hakemista (read-only) (Elmasri & Navathe 2000) ja monimutkaisten kyselyiden tekemistä varten (Gray & Watson 1998). *Päätöksenteon tukijärjestelmillä* (Decision support systems, DSS) tarkoitetaan teknologisia ratkaisuja, joita voidaan hyödyntää monimutkaisen päätöksenteon sekä ongelmanratkaisun apuna (Carlsson ym. 2002, 111). Laudon ja Laudon (1998, 50) jakavat johdon informaatiojärjestelmät kolmeen luokkaan: MIS- (Management information systems), DSS- sekä ESS- (Executive support systems) järjestelmiin, jotka ovat kehittyneet em. järjestyksessä 60-luvulta lähtien. MIS–järjestelmät on kehitetty enimmäkseen johdon tarpeisiin tarjoten johdolle raportteja sekä yhteenvetoja organisaation tilasta. DSS–järjestelmät on

suunniteltu tyydyttämään johdon tietotarpeita, tarjoten hyvät puitteet analysoida tehokkaasti tietoa ja näin tukea päätöksentekoa. ESS–järjestelmät on kehitetty palvelemaan ylemmän johdon eli strategisen tason päätöksentekoa, tarjoten helppokäyttöiset graafiset käyttöliittymät tiedon käsittelyyn ja tarkasteluun. On huomioitava, että tietovarastointi ei ole itsessään päätöksenteon tukijärjestelmä vaan toimii näiden järjestelmien tukena ja tekee olemassa ololaan mahdolliseksi niiden rakentamisen (Grossman & McCabe 1996). Carlsson ym. (2002) katsovatkin, että tietovarastointi voidaan käsittää uudeksi teknologiaksi, joka luo paremman pohjan rakentaa päätöksenteon tukijärjestelmiä. Eri järjestelmiä ei siis välttämättä tarvitse nähdä irrallisina ja erillisinä toisistaan, vaan ne voidaan nähdä evolutiivisen kehityksen tuloksina, jotka tukevat toisiaan. Herschel, Iyer, Nemati ja Steiger (2002) tiivistävätkin tietovarastojen roolin suhteessa päätöksenteon tukijärjestelmiin: tietovarastot ovat olennainen osa modernia päätöksenteontukiympäristöä.

Toisaalta tietovarastoinnin tavoitteena voidaan pohjimmiltaan ajatella olevan tiedon jalostaminen, kuten kuvasta 1 nähdään. Organisaatioiden perusjärjestelmissä on usein massoittain dataa eli niin sanottua ”raakatietoa”.



Kuva 1: Tiedon jalostaminen tietämykseksi (mm. Hackathorn 1999, s. 17)

Tämä tieto on kuitenkin usein hyvin yksityiskohtaista ja atomista. Näin ollen se ei ole niin hyödyllistä organisaatioiden päätöksenteon tai liiketoimintaprosessien muuttamisen ja kehittämisen tarpeisiin. Kuten mm. Hackathorn (1999) määritteleekin: informaatio on jalostetumpaa tietoa, joka muuttaa yksilöiden päätöksiä. Tietämys (knowledge) taasen on edelleen jalostettua informaatiota, joka muuttaa liiketoimintaprosesseja. Datalla tässä yhteydessä tarkoitetaan esimerkiksi tietokannan taulussa sijaitsevaa

merkkijonoa tai lukuarvoa. Informaatio sen sijaan on tietoa jossain kontekstissa eli sillä on jokin merkitys sitä tulkitsevalle ihmiselle. Esimerkiksi tietty lukuarvo voidaan käsittää siis raakatiedoksi. Mutta kun yksilö tietää, että tämä arvo on esimerkiksi organisaation liikevaihto tietyltä ajalta, on se informaatiota. Herschel ym. (2002) katsovatkin, että tietovarastoinnin perustavoitteena on tarjota tietotyöläiselle informaatiota, joka mahdollistaa päätöksenteon perustuen varmoihin faktoihin. Tietovarastoon koottavat yksityiskohtaiset tiedot organisaatiosta sekä sen ulkopuolisista lähteistä saatetaan saman käsitteellisen ja teknologisen katon alle (Jarke ym. 2000). Lisäksi tietoa muokataan, summataan ja yhdistellään niin, että tiedolla on enemmän merkitystä ja se on informatiivisempaa. Toisaalta tietovarastoympäristö tarjoaa kaikkine teknologioineen sekä etenkin pitkältä aikaväliltä kerätyn suuren tietomassan avulla uusia mahdollisuuksia tuottaa uutta tietämystä uusilla tavoilla. Tietovarastointi mahdollistaa esimerkiksi trendien ennustamisen, porautumisen tiedon yksityiskohtiin tai tiedon tiivistämisen yleisempään muotoon. Tietovaraston voima onkin siis sen kyvyssä jalostaa operatiivisesta tiedosta strategisessa päätöksenteossa käytettävää informaatiota (Kambayashi ym. 1998).

Tietovarastointi voidaan nähdä ns. innokkaana (eager) tai aktiivisena lähestymistapana tiedon integrointiin, poiketen ns. laiskasta (lazy) eli pyynnöstä suoritettavasta (on-demand) integroinnista (Garcia-Molina ym. 1995; Houben & Vdovjak 2001; Mohania ym. 1998; Widom 1995). Aktiivisessa tiedon integroinnissa relevantti tieto poimitaan, suodatetaan, transformoidaan ja integroidaan yhtenevään muotoon ennen siihen kohdistuvia kyselyitä eli etukäteen. Tietovarastoinnin etuna onkin Koellerin, Rundensteinerin ja Zhangin (2000) mukaan juuri mahdollisuus hakea tietoa yhdestä kohteesta ja täten se soveltuu tilanteisiin, joissa tarvitaan tehokasta kyselyiden prosessointia sekä tilanteisiin, joissa tehokas tiedon analysointi on tärkeää.

Tietovarastoinnin isäksi mainitun William Inmonin tietovarasto-määritelmään (Inmon 1996) on viitattu kirjallisuudessa hyvin paljon ja se tuokin tietovarastoinnin ominaisuudet sangen hyvin esille ja kokoaa hieman edellä annettuja määritelmiä. Inmonin mukaan tietovarasto on aihe-aluepainotteinen (subject-oriented), integroitu (integrated), aikasidonnainen (time-variant) sekä vakaa (nonvolatile) kokoelma tietoa päätöksenteon tukea varten. Taulukossa 1 on tiivistetty tietovarastoinnin ominaisuudet pääosin Inmonin (1996) havaintoihin perustuen.

TAULUKKO 1: Tietovarastoinnin ominaisuudet (Inmon 1996, 33-36, mukaan)

Ominaisuus	Kuvaus
aihe-alue painotteinen	Kiinnostuksen kohteena eivät ole sovellukset vaan tieto on organisoitu sen mukaan, miten käyttäjät viittaavat tai käyttävät tietoa.
integroitu	Operatiivisista ja ulkoisista lähteistä tuleva tieto transformoidaan yhtenevään muotoon.
vakaa	Tietovaraston tiedot ladataan yleensä yhtenä isona joukkona ja tämän jälkeen tietoja käytetään vain lukemiseen (read-only). Tietojen päivitystä perinteisessä mielessä ei tapahdu.
aika-sidonnainen	Tieto on relevanttia hyvin pitkään, jopa useita vuosia. Tiedolla on myös aina ajallinen ulottuvuus.

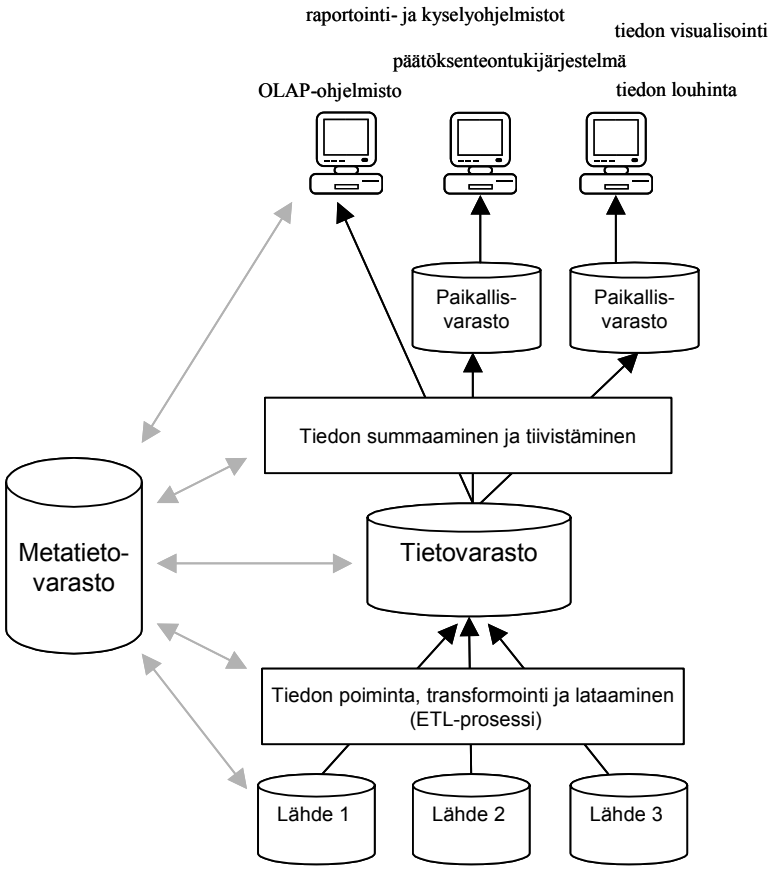
Yhteenvetona voidaan sanoa, että tietovarastoympäristö ei ole yksittäinen tuote vaan enemmänkin joukko teknologioita yhdistettynä suunnittelumenetelmiin ja ajatusmalleihin. Lisäksi sen päätavoitteena on tuoda tietoa yhteen paikkaan (ainakin loogisesti yhteen keskitettyyn varastoon), jotta organisaatio voisi hyödyntää tätä tietoa analysoinnissa, raportoinnissa ja lopulta päätöksenteon tukena. Edelleen voidaan sanoa, että tietovarastoinnissa tavoitteena on toisaalta

tukea päätöksentekoa sekä toisaalta mahdollistaa organisaatioiden suurten tietomassojen hallinta, jakaminen ja jalostaminen.

2.2 Tietovarastoarkkitehtuuri ja -prosessit

Tietojärjestelmien koon ja monimutkaisuuden kasvaessa on välttämätöntä käyttää joitakin loogisia käsitteitä tai rakenteita, ts. arkkitehtuuria, määrittämään ja kontrolloimaan järjestelmän rajapintoja sekä komponentteja (Zachman 1987). Myös Geigerin ym. (1997) mielestä tarvitaan erityinen arkkitehtuuri täyttämään uudet vaatimukset tietojärjestelmille. Seuraavaksi esitellään tietovarastoarkkitehtuuri ja siihen kuuluvat komponentit.

Hyvin yleisesti esitetty tietovarastoarkkitehtuuri (mm. Dittrich & Vaduva 2001; Inmon 1996; Jarke ym. 2000; Koeller ym. 2000) muodostuu useammasta tasosta kuten kuvasta 2 nähdään. Alimmalla tasolla ovat tiedon lähteet, jotka tarjoavat sisältöä tietovarastoon. Nämä ovat usein operatiivisia tietokantoja ja -järjestelmiä tai ulkoisia lähteitä kuten esimerkiksi www-sivuja. Keskellä on varsinainen tietovarasto, joka sisältää Inmonin (1996) mukaan suuret määrät atomista eli varsin primitiivistä ja yksityiskohtaista tietoa mutta myös hieman summattua ja yhdisteltyä tietoa (eli tietoa, joka on johdettu toisista tiedoista tai joka on erinäisten matemaattisten laskutoimitusten tulos). Tietovarasto voi toimia syötteenä niin sanotuille paikallisvarastoille (data mart). *Paikallisvarasto* on tarkoitettu tiettyä osastoa ja sen tietotarpeita varten ja se sisältää Inmonin (1996) mukaan lähes yksinomaan summattua ja johdettua tietoa. Ylimmällä tasolla ovat loppukäyttäjät eli yksilöt tai sovellukset, jotka käyttävät, analysoivat ja hyödyntävät tietovaraston tietoja. Tällä tasolla voi olla muun muassa raportointi- ja analysointityökaluja, päätöksenteon tukijärjestelmiä (DSS), taulukkolaskentaohjelmistoja tai tiedon moniulotteista käsittelyä (OLAP -työkaluja). Tietovarastoarkkitehtuuriin kuuluu myös metatietovarasto. Se voidaan nähdä tietovarastoympäristön integroivana komponenttina (Vaduva & Vetterli 2001) ja kaikkien tietovaraston osa-alueiden hallinta tulisi tapahtua



Kuva 2: Tietovarastoarkkitehtuuri

metatietovarastosta käsin (Jarke ym. 2000). Metatietovarasto sisältää metatietoa, jota tarkastellaan yksityiskohtaisemmin myöhemmissä luvuissa.

Tiedon jalostaminen loppukäyttäjälle mielekkääseen muotoon tapahtuu prosesseissa, joista esimerkkinä näkyy muutama kuvassa 2. Kun tietoa siirretään operatiivisista tai ulkoisista lähteistä tietovarastoon, täytyy ensiksi poimia halutut ja oleelliset tietueet, jotka halutaan siirtää. Tämän jälkeen tieto tulee muokata ja integroida yhtenäiseen muotoon eli sille tulee tehdä transformointeja, tietojen tarkistamista, tiedon puhdistamista ja tarpeen mukaan summausta. Tässä prosessissa tieto saatetaan yleisen tietomallin mukaiseksi ja näin tieto on yhteneväistä suhteessa muihin tietovaraston tietoihin. Tämä prosessi tunnetaan yleisesti ETL–prosessina (extract, transform,

load). Kun tieto on tietovarastossa ja se halutaan siirtää paikallisvarastoihin, tulee suorittaa lisää tiedon summausta ja yhdistämistä. Tämä sen vuoksi, että paikallisvarastoissa ei olla niinkään kiinnostuttu kovin yksityiskohtaisesta tiedosta vaan enemmänkin yleisestä, johdetusta tiedosta, joka on informatiivisempaa. Toisaalta tiedon summaamisella sekä tiivistämisellä on myös merkitystä vasteaikoihin; mitä tiiviimmässä muodossa tieto on, sitä nopeammin käyttäjät saavat vastaukset kyselyihinsä (Geiger ym. 1997; Hackathorn 1995). Tietoa siis jalostetaan yhä informatiivisempaan muotoon mitä ylemmälle tasolle mennään.

Tietovarastoympäristö voidaan nähdä myös eri medioista muodostuvana kokonaisuutena. Medialla tarkoitetaan tässä tapauksessa tiedon tallennusalustaa. Toisin sanoen tietovarastoympäristössä kaikkea tietoa ei säilytetä samalla tallennusalustalla vaan kriittiset ja hyvin kysytyt tiedot ovat medialla, joka mahdollistaa hyvän vasteajan. Tällainen media on toisaalta kalliimpi vaihtoehto. Tieto, jota ei käytetä enää niin aktiivisesti tai tieto, joka on saavuttanut tietyn ennalta määrätyn iän sijoitetaan ns. massatallennusmedialle kuten magneettinauhalle. Tällöin tieto on hitaammin saatavilla mutta kustannukset sen säilyttämiseen ovat pienemmät. Inmon (1996) mainitseekin, että koska tietoa esiintyy eri medioilla, voidaan mahdollisesti tarvita enemmän kuin yksi tietokannan hallintajärjestelmä. Tai sitten joitakin tietoja ei hallita laisinkaan tietokannan hallintajärjestelmän avulla. Tällaiset tiedot voivat olla siis esimerkiksi CD-levyillä tai jopa paperilla. Inmon (1996) painottaa kuitenkin, että vaikkakin tieto siirretään halvemmalle massatallennusalustalle ei se tarkoita että tieto siirtyisi pois tietovarastoympäristöstä.

Edellä on kuvattu yleisellä tasolla tietovarastoympäristön hyvin yleinen arkkitehtuuri ja prosesseja, jotka liittyvät tietovarastointiin ja tiedon kulkuun tässä kokonaisuudessa. Tarkoituksena on näyttää, että tietovarastoympäristölle on ominaista eri komponenttien lukuisa määrä sekä heterogeenisyys näiden komponenttien, ohjelmistojen ja tietolähteiden välillä. Lisäksi erilaista tietoa

esiintyy arkkitehtuurin eri tasoilla. Seuraavassa kohdassa käsitellään hieman tarkemmin kuvassa 2 esille tulleita osa-alueita, pohjautuen Grayn ja Watsonin (1998) jaotteluun, jonka mukaan tietovarastoarkkitehtuuri voidaan jakaa kolmeen osaan: taustaprosessointiin (back-end processing), varsinaiseen tietovarastoon sekä edustakäyttöön (front-end processing) eli asiakaspäähän.

2.2.1 Tietolähteet – taustaprosessit

Tietovaraston tietolähteet ovat useimmiten heterogeenisiä (Jarke ym. 2000) sekä autonomisia eli toimivat myös itsenäisesti suhteessa tietovarastoihin (Kambayashi ym. 1998; Koeller ym. 2000). Täten sama tieto voi olla esitettynä eri tavalla eri lähteissä. Singhin (1998) mukaan heterogeenisissä järjestelmissä onkin yleistä, että organisaation informaatio ei ole yhteensopivaa: informaatio voi olla organisoitu erilailla (erilaiset avaimet, erilaiset tietokannan kaavat) tai esimerkiksi samoilla sanoilla voi olla eri merkitys eli semantiikka.

Jarke ym. (2000, 6) sekä Hovi (1997, 48) ovat listanneet tietovaraston eri tietolähteitä, joita ovat muun muassa: organisaation tietokannat (esim. relaatio- tai oliotietokanta), standardiohjelmistojen tai ohjelmistopakettien tiedostot (esim. Excel tai COBOL ohjelmistot) sekä ulkoiset lähteet (esim. www-sivut, tilastokeskus, pörssikurssit yms.). Etenkin muiden kuin operatiivisten lähteiden merkitys kasvaa jatkuvasti. Tietovarastoon täytyy pystyä keräämään ja yhdistämään tietoa, joka voi olla esimerkiksi sähköpostia, kuvia tai multimediaa (Hackathorn 1995).

Tietovarasto täytetään tiedoilla yhdellä isolla latauksella, jonka jälkeen pienempiä eriä tietoja ladataan tietyin aikavälein. Tätä prosessia kutsutaan myös tietovaraston virkistysprosessiksi (refresh cycle) (ks. Bontempo & Zagelow 1998, 41). Kimballin (1996) mukaan nämä yksittäiset lataukset voivat käsittää tuhansia tai jopa miljoonia tietueita. Tietojen lisäys tietovarastoon

tapahtuu yleensä yöllä niin, että se ei häiritse organisaation päivittäistä työtä (Hovi 1997).

Jotta tieto saataisiin integroitua yhtenevään muotoon tulee sitä muokata ja transformoida monella tapaa. Ainakin seuraavat toiminnot voivat olla tarpeellisia siirrettäessä tietoa tietovarastoon (ks. Jarke ym. 2000, 6):

- poiminta (poimitaan halutut tiedostot eri lähteistä)
- transformointi (muunnokset tietotyyppien, kielien yms. välillä)
- analysointi (väärien/odottamattomien arvojen tunnistaminen)
- puhdistus (korjataan löydetyt virheet ja epäyhteneväisyydet)
- lataaminen (ladataan tieto tietovarastoon).

Varsinkin ulkoiset lähteet ovat saaneet suurta huomiota viime aikoina. Etenkin internetin vaikutus organisaation tiedon lähteenä on kasvanut huomattavaksi (Pokorný 2001). WWW:n yhdistämistä tietovarastointiin ovat tutkineet etenkin Hackathorn (1999) sekä Bhowmick, Lim, Madria ja Ng (1999). Molemmissa lähestymistavoissa on kyse systemaattisesta www-resurssien jalostamisesta liiketoimintatietämyksen tarpeisiin. Tavoitteena voidaan ajatella olevan liiketoiminnallisesti tärkeän ja relevantin informaation tarjoaminen organisaatiolle ja näin ollen tuottavuuden nostaminen (Hackathorn 1999) sekä toisaalta hyödyllisen www-informaation löytäminen sekä analysointi päätöksenteon tukea varten (Bhowmick ym. 1999). Tähän asti www:tä on kyllä hyödynnetty tietovarastoinnin yhteydessä, mutta tällöin suunta on ollut tietovarastosta www:hen. Toisin sanoen www:tä hyödynnetään tietovaraston tietojen julkaisemisessa, mutta Hackathornin (1999) mukaan aiemmin ei ole hyödynnetty www:tä sisällön tarjoajana tietovarastolle. Ulkoista tietoa tietolähteenä tarkastellaan tarkemmin kohdassa 3.4.

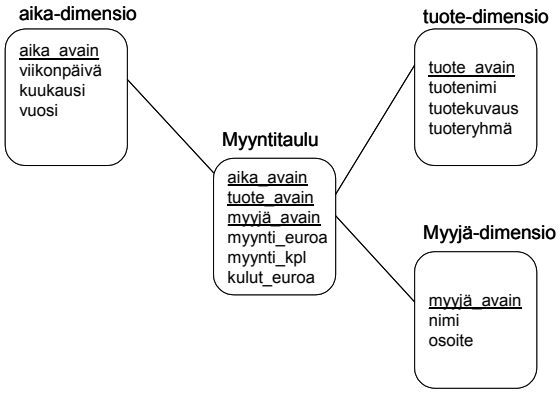
2.2.2 Tietovarasto

Tietovarastoympäristön keskuksena toimii varsinainen tietovarasto, joka toteutetaan suurimmaksi osaksi relaatiotietokantatekniikalla (Hovi 1997, 19;

Dittrich & Vaduva 2001; Grossman & McCabe 1996, 144). Tietovarasto sisältää huomattavat määrät eri muotoista tietoa. Tätä keskeistä osaa tietovarastoympäristössä kutsutaan usein globaaliksi tai yrityksenlaajuiseksi tietovarastoksi, joka sisältää tietoa hyvin pitkältä aikaväliltä (Jarke ym. 2000). Tietovaraston sisältöä voidaan pitää otoksina niistä organisaation relevanteista tiedoista, joista ollaan kiinnostuttu. Nämä otokset edustavat aina jotakin tiettyä aikaa, siis sitä hetkeä kun ne on otettu ja ladattu tietovarastoon. Vanhoja tietoja ei poisteta uusien tilalta lataamisten yhteydessä, vaan uudet tiedot asetetaan ns. jonoon vanhojen kanssa. Täten tietovarastoon kertyy eri tiedon sukupolvia (Hovi 1997). Näin ollen tietovarasto mahdollistaa pitkän aikavälin tietojen analysoinnin.

Yleinen näkemys Hackathornin (1999) mukaan on, että organisaatioiden tulisi rakentaa koko yrityksen laajuinen tietomalli, jonka pohjalta tietovarasto rakennettaisiin. Tällöin voidaan puhua myös organisaation tietoarkkitehtuurista (information architecture), jonka tarkoituksena on mallintaa organisaation tietoresursseja (Everest & Kim 1994). Tässä yhteydessä pääpaino on kuitenkin varsinaisten tietojen ja niiden välisten suhteiden kuvaamisesta (eli tietomallinnuksesta). Tietoresursseiksi voidaan nimittäin käsittää myös esimerkiksi ohjelmistot ja laitteistot. Myös Inmon (1996) sekä Calvanese ym. (2001) katsovat, että organisaation laajuinen tietomalli kuvastaa yrityksen tietotarpeita ja täten sen tulisi toimia tietovaraston suunnittelun pohjana. Ilman organisaation laajuista tietomallia tietovaraston kehittäminen saattaa keskittyä liikaa siihen, mitä tietoa esiintyy operatiivisissa lähteissä, eikä siihen mitä tietoa oikeastaan tarvitaan (Devlin 1997).

Moniulotteinen mallinnus on Kimballin (1996) mukaan tekniikka, jonka avulla tietokannoista voidaan tehdä yksinkertaisempia sekä ymmärrettävimpiä. Moniulotteista mallia voidaan myös kutsua tähtimalliksi. Tähtimalli muodostuu keskellä olevasta fakta-taulusta sekä sitä ympäröivistä dimensio- eli ulottuvuustauluista (KUVA 3). Fakta-taulu sisältää johonkin liiketoiminta-



Kuva 3: Tähtimalli (vrt. Hovi 1997, 73; Kimball 1996, 11)

tapahtumaan liittyvää tietoa kuten myynti- tai laskutustietoja. Fakta-taulu sisältää myös viiteavaimet kaikkiin siihen liittyviin ulottuvuustauluihin. Lisäksi fakta-taulu on yleensä normalisoitu ja voi sisältää jopa miljoonia rivejä tietoa (Hovi 1997). Ulottuvuustaulut sen sijaan ovat usein normalisoimattomia sekä huomattavasti pienempiä kuin fakta-taulu. Ulottuvuustaulujen kenttiä kutsutaan attribuuteiksi. Fakta-taulu sisältää siis johonkin liiketoimintatapahtumaan liittyviä tietoja ja ulottuvuustaulujen kentät eli attribuutit kuvaavat tarkemmin kulloistakin tapahtumaa. Näin ollen voidaan esimerkiksi tarkastella kustakin myyntitapahtumasta kuka oli myyjä, mikä tuote myytiin ja milloin tämä tapahtui. Voitaisiin myös tarkastella esimerkiksi kuinka paljon tietty myyjä on myynyt tiettyä tuotetta tietyllä aikavälillä. Tiedot tallennetaan tietovarastoon yleensä tähtimallin tai lumihiutalemallin mukaisesti (Jacob & Sen 1998; Miller, Nilakanta & Wu 2001).

OLAP (Online analytic processing) on sen sijaan termi tai konsepti, jota käytetään kuvaamaan tietovaraston sisältämän monimutkaisen tiedon analysointia (Elmasri & Navathe 2000, 844) ja OLAP–sovellukset ovat ainakin Dittrichin & Vaduvan (2001) näkemyksen mukaan kaikista yleisin keino analysoida tietovaraston tietoja. Jarken ym. (2000) mukaan OLAP–työkalut mahdollistavat ns. moniulotteisen näkymän tietoon. Hovin (1997, 65) mukaan

moniulotteinen analyysi sopii parhaiten paikallisvarastotasoiselle tietovarastoinnille.

2.2.3 Asiakaspää

Loppukäyttäjä etsii informaatiota tietovarastosta käyttäen erityyppisiä ohjelmistoja: OLAP–työkaluja, DSS–järjestelmiä, taulukkolaskentaohjelmia, tiedon louhinta –työkaluja. Tietovaraston tietoja voidaan hyödyntää myöskin tiedon visualisoinnin, tilastollisen analyysin tai esimerkiksi raportointityökalujen avulla (ks. Jarke ym. 2000, 7).

Olennaista tietovarastoinnissa on tarjota loppukäyttäjälle merkityksellistä ja hyödyllistä tietoa. Tämä tapahtuu osaltansa jo edellä esille tulleiden summaamis- ja tiivistämisprosessien avulla. Toisaalta lisäarvoa luodaan tarjoamalla tieto käyttäjälle hyödyllisessä ja käytännöllisessä formaatissa: tekstidokumenttina, taulukkolaskentaohjelmassa, animaationa tai muulla tavalla visuaalisena ja graafisessa muodossa. Hackathorn (1995) uskoo, että investoimalla loppukäyttäjälle tarjottavan tiedon esitysmuodon ja teknologioiden laatuun, voidaan saavuttaa mahdollisesti suuria hyötyjä.

Edellä mainittu moniulotteinen mallinnus sekä OLAP–työkalut mahdollistavat monipuolisemman tiedon analysoinnin ja käsittelyn. Muun muassa seuraavat toiminnot ovat mahdollisia (mm. Elmasri & Navathe 2000, 846; Hovi 1997, 60; Jarke ym. 2000, 6):

- karkeistaminen (roll-up): tietoa summataan/tiivistetään yleisemmäksi
- porautuminen (drill-down): mahdollisuus edetä yleisestä tiedosta yksityiseen (karkeistamisen vastakohta)
- viipalointi (slice and dice)
- parhaat arvot -kyselyt (Top ten)
- ristiintaulukointi.

Edellä mainittujen tekniikoiden avulla tietotyöläinen voi esimerkiksi tarkastella myyntilukuja hyvin yleisellä tasolla eli karkeistettuna. Esimerkkinä tästä toimii myyntilukujen tarkastelu maittain vuositasolla. Vaihtoehtoisesti hän voi porautua yksityiskohtaisempaan tietoon ja tarkastella myyntilukuja liikekohtaisesti kuukausitasolla.

Esimerkkinä tietovaraston käytöstä voidaan ajatella seuraavaa. Loppukäyttäjä (analyytikko, johtaja yms.) ottaa yhteyden OLAP–palvelimeen. Tämän jälkeen OLAP–palvelin tulkitsee asiakkaan kyselyn ja kääntää sen monimutkaiseksi SQL–kyselyksi, jotta päästään käsiksi itse tietovaraston tietoon ja palauttaa vastauksen käyttäjälle. (Kambayashi ym. 1998)

Usein tietovaraston loppukäyttäjät suorittavat ns. ad hoc –kyselyitä (mm. Elmasri & Navathe 2000; Miller ym. 2001). He siis hakevat tietoa kyselyillä, jotka ovat usein tilapäisiä, kenties vain yhden kerran suoritettavia. Operatiivisissa tietojärjestelmissä käyttäjillä on sen sijaan totutut tavat käyttää järjestelmiä: samat rutiinit toistuvat tiedon päivittämisessä, poistamisessa ja lukemisessa.

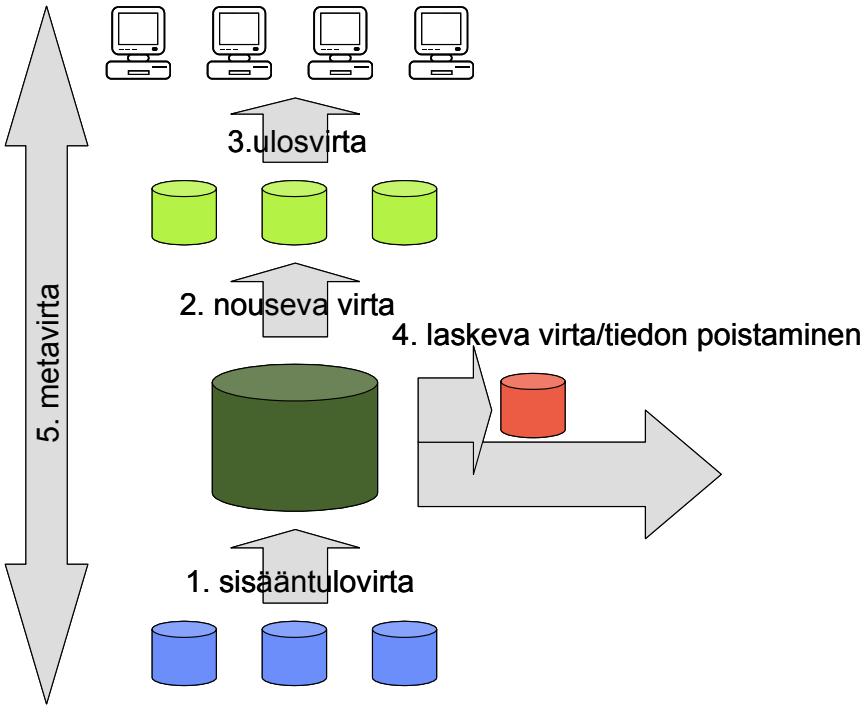
Erityisen huomioitavaa on, että tietovaraston loppukäyttäjät eivät ole useinkaan ATK-ammattilaisia vaan itse liiketoiminnan asiantuntijoita (business users) (Grossman & McCabe 1996). Tämä tarkoittaa sitä, että loppukäyttäjillä ei välttämättä ole tietotaitoa käyttää monimutkaisia ohjelmistoja, tehdä vaikeita kyselyitä tai yleisesti ymmärtää tietovaraston toimintaa.

2.2.4 Tietovaraston viisi tietovirtaa

Edellä on lueteltu eri komponentteja ja prosesseja, joista tietovarastoympäristö voi koostua. Lisäksi esille tuotiin komponenttien merkitys erilaisen tiedon tallentamisessa. Hackthornin (1995) mukaan tietovaraston todellinen arvo on kuitenkin tietovirtojen hallinta eikä niinkään pelkkä tietojoukkojen kerääminen. Hän esittääkin viisi tietovirtaa (Hackathorn 1995; Hackathorn 1999), joista

tietovarastoympäristö koostuu. Hackathornin näkemys ei ole ristiriidassa aiemmin esitellyille havainnoille. Se enemmänkin kokoaa yhteen aiemmat huomiot ja esittää erilaisen näkökulman samalle asialle. Lisäksi on huomioitavaa, että Hackathorn korostaa sitä, että tietovarasto ei ole tuote tai staattinen kokonaisuus, vaan enemmänkin jatkuvasti elävä prosessi tai aktiviteetti (tällöin voidaankin puhua enemmänkin tietovarastoinnista). Seuraavaksi esitellään tietovarastoinnin viisi tietovirtaa perustuen Hackathornin (1995; 1999) näkemyksiin.

Kuvassa 4 nähdään tietovarastoinnin viisi tietovirtaa. Sisääntulovirta (inflow) tarkoittaa tiedon kulkua operatiivisista lähteistä tietovarastoon. Tässä tietovirrassa tieto puhdistetaan, integroidaan ja transformoidaan sopivaksi tietovarastoon. Tämä vastaa kuvan 2 ETL–prosessia.



Kuva 4: Tietovarastoinnin viisi tietovirtaa (Hackathornin 1999, mukaan)

Nouseva virta (upflow) tarkoittaa prosessia, jossa hyvin yksityiskohtaista tietoa tiivistetään, summataan yhteen tai muuten prosessoidaan niin, että käyttäjien on helpompi ja nopeampi tehdä kyselyitä. Tämä virta on Hackathornin (1995, 39) mielestä se prosessi, jossa luodaan lisäarvoa tietovaraston tiedoille.

Kolmas tietovirta on ulosvirta (outflow), joka ei nimestään huolimatta poista tietoja. Tämä virta kuljettaa tiedot loppukäyttäjille eli virta edustaa käyttäjien tekemiä kyselyitä ja tiedon hakuja eri sovelluksilla.

Laskevavirta (downflow) on se prosessi, jossa vähän käytetty, hyödytön tieto poistetaan varsinaisesta tietovarastosta johonkin halvempaan säilytyspaikkaan kuten CD–levylle tai magneettinauhalle, kuten jo kohdassa 2.2 todettiin. Tämän tiedon liiketoiminta-arvo ei ole enää niin suuri, että sitä kannattaisi säilyttää kalliilla alustalla hidastamassa tietovaraston käyttöä. Tämä ei siis kuitenkaan tarkoita, että tieto välttämättä poistuisi tietovarastoympäristöstä. Tieto pidetään saatavilla, se on vain eri medialla. On tietenkin myös mahdollista, että tieto poistetaan kokonaan tietovarastoympäristöstä.

Metavirta (metaflow) on Hackathornin (1995) mukaan tietovaraston tärkein, mutta myös vähiten ymmärretty ja hyödynnetty tietovirta. Metatietovirtaa käytetään periaatteessa hallitsemaan itse tietovarastoa ja sen tietoja. Metatietovirta on prosessi, joka liikuttaa tietovaraston metatietoja eli toisin sanoen tietoa toisista tietovirroista. Metatieto voidaankin nähdä sääntöinä tai ohjeina, jotka hallitsevat sekä suuntaavat tietovaraston toimintaa ja käyttöä: tiedon poimintaa, transformointia, summausta sekä käyttäjien toimintaa tiedon hakemisessa (vrt. Hackathorn 1999, 81). Tässä kohdassa tyydytään suppeaan määritelmään metatiedosta ja metavirrasta. Luvussa 4 syvennytään tarkemmin metatietoon.

Edellä on esitetty tiivistetysti tiedon elinkaari tietovarastossa. Inmonin (1996) mukaan tiedon elinkaaren huomioonottaminen kokonaisuudessaan, mukaan lukien myös tiedon lopullisen poistamisen tietovarastosta tai siirtämisen

halvemmalle alustalle, tulisi olla aktiivinen osa tietovaraston suunnitteluprosessia.

2.3 Tietovarasto ja operatiiviset järjestelmät

On olennaista tehdä jako tietovaraston ja operatiivisten järjestelmien välillä, jotta voidaan paremmin ymmärtää tietovarastoinnin luonnetta sekä miksi tietovarasto sisältää juuri sille ominaista tietoa. Tässä kohdassa tehdään toisaalta erottelu tietovarastojen ja operatiivisten tietokantojen/-järjestelmien välillä sekä toisaalta informatiivisen ja operatiivisen tiedon välillä.

Tietovarastojen tavoitteena on, kuten edellä on mainittu, toimia päätöksenteon tukena. Tavalliset tietokannat sen sijaan tukevat online transaction processing (OLTP) –järjestelmiä, jotka toimivat päivittäisten operaatioiden tukena (mm. Laudon & Laudon 1998, 40). Operatiiviset järjestelmät eivät sisällä tietoa pitkältä aikaväliltä, tiedot eivät ole välttämättä yhteneväisiä ja monimutkaisten kyselyiden tekeminen niihin saattaa viedä erityisen kauan ja näin ollen myös vaikeuttaa päivittäisten operaatioiden hoitamista (Gray & Watson 1998). Tästä johtuen hyvin yleinen havainto on, että OLTP–järjestelmät eivät sovellu niin hyvin päätöksenteon tueksi (esim. Elmasri & Navathe 2000; Jarke ym. 2000; Miller ym. 2001). Lisäksi operatiiviset tietojärjestelmät eroavat huomattavasti tietovarastoista (Kimball 1996, 1). Seuraavaksi tuodaan esille selkeimmät poikkeavuudet.

Tietovarastojen ja operatiivisten järjestelmien eroja ovat listanneet ainakin Hovi (1997), Inmon (1996), Jones (1998) sekä Kimball (1996). Olennaisimpina eroina voidaan pitää seuraavia:

- järjestelmien taustalla olevat teknologiat poikkeavat huomattavasti toisistaan (laitteistot, ohjelmistot)
- käyttäjäryhmät ovat erilaisia
- tiedon prosessoinnin ominaisuudet poikkeavat toisistaan
- järjestelmien sisältämä tieto on fyysisesti erilaista
- ylläpito on erilaista

- kehityselinkaari on erilainen.

Operatiivisen ympäristön teknologian täytyy huomioida sellaisia asioita kuten tiedon ja transaktioiden lukitus, joista ei tarvitse huolehtia tietovarastoympäristössä, koska se on vain lukemista varten. Operatiiviset järjestelmät on viritetty hoitamaan tehokkaasti tiedon päivittämistä kun taas tietovarastoympäristön keskeisenä vaatimuksena on vastata tehokkaasti monimutkaisiin kysymyksiin. Lisäksi tietovarastojen tulee pystyä säilyttämään huomattavasti suurempia määriä tietoja kuin operatiivisten tietokantojen.

Operatiivinen ympäristö perustuu OLTP-tekniikalle kun taas tietovarastoinnin yhteydessä puhutaan OLAP-tekniikasta. Toisin sanoen operatiiviset järjestelmät ovat transaktio-orientoituneita ja ovat keskittyneet sovellusten ympärille. Tietovarasto ovat analyysi- ja päätöksentekoon suuntautuneita ja ovat keskittyneet kiinnostuksen kohteena olevien tietojen ympärille.

Operatiivisia järjestelmiä käyttävät toimihenkilötason ihmiset, jotka käsittelevät yksityiskohtaista tietoja kuten asiakas-, tuote- tai henkilötietoja. Tietovaraston käyttäjät ovat usein keskijohdon edustajia kuten markkinoinnin suunnittelijoita, analyytikkoja, tuotepäällikköjä jne. Kimballin (1996, 3) vertauskuva tiivistää hyvin käyttäjien erot: operatiivisten järjestelmien käyttäjät pyörittävät organisaatiota, kun taas tietovaraston käyttäjät katsovat organisaation pyörien pyörimistä, keräävät näin tietoa organisaatiosta ja tekevät tästä johtopäätöksiä. Inmonin (1996) mukaan analyyttisen ympäristön käyttäjiä on huomattavasti pienempi määrä verrattuna operatiivisen ympäristön käyttäjiin. Hackathorn (1999) kuitenkin mainitsee, että tietovarastojen käyttäjäkunta on laajentunut käsittämään nykyään myös muita organisaatiotasoja johtotason lisäksi.

Jacob ja Sen (1998) näkevät, että OLTP–järjestelmien kehityksessä tyypilliset tekniikat, kuten ER–mallit, niiden transformointi relaatioiksi sekä näihin liittyvät normalisointitekniikat eivät ole hyödyllisiä tietovarastoinnissa.

Myöskään Kimball (1996) ei usko ER-mallien soveltuvuuteen organisaatioiden laajuisten tietovarastojen perustana.

Operatiivisten järjestelmien ja tietovarastojen välinen erottelu voidaan tehdä myös sen perusteella, minkälaista tietoa ne sisältävät. Taulukkoon 2 on tiivistetty operatiivisen ja informatiivisen tiedon ominaisuudet. Devlin (1997, 44-45) tiivistää operatiivisen sekä informatiivisen tiedon erot: ensiksi mainittua käytetään liiketoiminnan päivittäiseen ylläpitoon ja sitä voidaan hyödyntää lyhyen aikavälin toiminnoissa ja päätöksissä. Jälkimmäinen sen sijaan hyödyntää ja auttaa ylläpitämään liiketoimintaa pidemmällä aikavälillä.

TAULUKKO 2: Operatiivisen ja informatiivisen tiedon erot (ks. Bontempo & Zagelow 1998, ; Devlin 1997, 14-15; Inmon 1996, 18)

Operatiivinen/primitiivinen tieto	Informatiivinen/johdettu tieto
• tukee päivittäisiä operaatioita	• tukee johdon päätöksentekoa pidemmällä aikavälillä
• sovelluspainotteinen	• aihe-aluepainotteinen
• yksityiskohtaista	• summattua tai muuten jalostettua tietoa
• täsmällistä tietoa juuri sillä hetkellä kun sitä haetaan	• historiatietoa
• palvelee laajaa henkilökuntaa (operatiivista tasoa)	• palvelee johtotasoa
• voidaan päivittää (read-write)	• ei päivitetä, vain lukukäytössä
• ei toisteisia arvoja (nonredundancy), normalisoitu	• denormalisoitu
• staattinen rakenne, muuttuva sisältö	• joustava rakenne
• korkea saatavuusaste	• saatavuusaste ei niin kriittinen
• ennustettavissa olevat kyselyt	• ad hoc –kyselyt
• päivitysten tehokkuus tärkeää	• hakutehokkuus tärkeää

2.4 Yhteenveto

Tässä luvussa määriteltiin käsite tietovarastointi ja joitakin sen keskeisempiä piirteitä tämän tutkimuksen kannalta. Luvussa esiteltiin tyypillinen tietovarastoarkkitehtuuri, tietovarastointiin liittyvät prosessit ja ohjelmistot/komponentit sekä tuotiin esille miten tietovarastot eroavat perinteisimmistä operatiivisista tietojärjestelmistä.

Tietovarastoinnin perusideana on relevantin tiedon poimiminen heterogeenisistä tietolähteistä, tiedon puhdistaminen, transformointi ja lataaminen tietovarastoon. Tämän jälkeen tietoon päästään käsiksi erilaisilla loppukäyttäjien työkaluilla kuten DSS–järjestelmillä, OLAP–työkaluilla, raportointivälineillä jne. Keskeistä tietovarastoinnissa on, että se tarjoaa paremmat mahdollisuudet analysoida organisaation tietoja sekä tehdä parempia ja nopeampia liiketoimintapäätöksiä.

Jarke ym. (2000) korostavat, että tietovarastot eivät voi olla ns. pakettiratkaisuja (off-the-shelf) vaan ne tulee suunnitella ja optimoida aina kulloisenkin asiakkaan mukaan. Tietovarasto ei ole myöskään staattinen kokonaisuus. On tärkeää huomioida, että kaikki tietoa sisältävät komponentit kuten operatiiviset lähteet, tietovarasto, paikallisvarastot muuttuvat ajan myötä. Liiketoimintasääntöjen muutokset vaikuttavat myös operatiivisten järjestelmien tietomalleihin sekä toisaalta loppukäyttäjien vaatimuksiin eli myös tietovaraston (tai paikallisvaraston) tietomalleihin (Jarke ym. 2000). Täten tietovaraston suunnittelu on jatkuva prosessi (Hovi 1997).

Tietovarastoympäristö on hyvin monimutkainen kokonaisuus. Siihen liittyy taustakomponentteja (tietolähteet, ETL-työkalut), varsinainen tietovarasto ja mahdollisia paikallisvarastoja sekä asiakaspään työkaluja eli esimerkiksi kysely-, analysointi-, tiedon louhinta ja tiedon visualisointiin liittyviä työkaluja. Kimball (1996) esittää hyvän huomion sanoessaan, että tietovarastoympäristö ei

pidä sisällään pelkästään tietoa ja sitä varastoivaa tietovarastoa. Hänen mukaansa 40% tietovarastoympäristöstä muodostuu asiakaspään työkaluista.

Huomioitavaa on myös tiedon kulku tietovarastoarkkitehtuurin eri tasoilla. Tietoa jalostetaan informatiivisempaan muotoon, jotta se olisi hyödyllisempää ja merkityksellisempää pidemmän tähtäimen päätöksenteossa ja esimerkiksi organisaation tilan analysoinnissa. Tieto kulkee läpi useiden prosessien ja sitä muokataan, tiivistetään, integroidaan, summataan ja siirretään eri medioille läpi sen elinkaaren.

Kohdan 2.3 lopussa tehtiin myös erottelu operatiivisen ja informatiivisen tiedon välillä. Operatiivinen tieto on yksityiskohtaista tietoa, jota käytetään organisaation päivittäisten operaatioiden hoitamisessa. Informatiivinen tieto on sen sijaan johdettua tietoa eli se on esimerkiksi summattua tietoa ja sopii näin paremmin päätöksenteon tueksi. Seuraavassa luvussa syvennytään tarkastelemaan lähemmin tietovarastoinnin eri tietotyyppejä.

3 TIETOTYYPIT TIETOVARASTOSSA

Tämän luvun tavoitteena on tuoda esille minkälaista tietoa tietovarasto sisältää, etenkin sen eri tasoilla. Luvussa näytetään myös, mitkä tiedon tyypit ovat tämän tutkimuksen kannalta oleellisia. Päätavoitteena tässä luvussa on tehdä erottelu tiedon ja metatiedon välillä tietovarastokontekstissa. Lisäksi tavoitteena on selvittää mitä ominaisuuksia tietovaraston sisältämillä tiedoilla on ja mikä merkitys eri tietotyypeillä on tietovarastoympäristössä. Lähtökohtana on, että organisaatioissa esiintyvän tiedon voidaan nähdä edustavan reaalimaailmaa eli liiketoimintaa (ks. Devlin 1997, 10-11; Hackathorn 1999, 33) ja metatieto, kuten aiemmin jo määriteltiin, sen sijaan voidaan nähdä kuvaavan tietoa, ainakin suppean määritelmän mukaan. Myöhemmissä luvuissa metatiedon määritelmää laajennetaan.

On huomioitava, että eri käyttäjillä on erilaisia tietotarpeita ja täten tietoon käsiksi pääseminen sekä tiedon esittäminen täytyy järjestää eri tavalla eri tilanteissa (Gardner 1998). Näin ollen on olennaista määrittää millaisissa eri muodoissa tietoa esiintyy, ts. mitä eri tietotyyppejä tietovarastoissa voi olla.

Organisaatioissa voidaan ajatella olevan pääasiallisesti kahdenlaista tietoa: liiketoimintatietoa sekä metatietoa. Kuten jo johdannossa määriteltiin, metatieto on mitä tahansa tietoa, jonka tarkoituksena on helpottaa ja tukea tietovarastoinnin hallintaa sekä sen hyödyntämistä. Liiketoimintatieto (business data) sen sijaan voidaan Devlinin (1997, 44) mukaan nähdä tietona, jota tarvitaan organisaation varsinaisen liiketoiminnan eteenpäin viemiseen ja johtamiseen. Se edustaa sekä organisaation suorittamia aktiviteetteja että tosimaailman kohteita, joiden kanssa se on tekemisissä, kuten tuotteita tai asiakkaita. Esimerkkinä liiketoimintatiedosta voidaan ajatella tietokannan tauluun tallennettua asiakkaan nimeä, osoitetta jne. Metatieto sen sijaan tässä tapauksessa olisi taulun kaava eli sarakkeiden nimet. Lisäksi metatieto voisi olla myös tietoa siitä, mistä ko. tieto on peräisin tai mitä prosessointia sille on

tehty. Liiketoimintatieto pitää sisällään jo kohdassa 2.3 erotellut operatiivisen sekä informatiivisen tiedon. Ne ovat liiketoimintatiedon eri lajeja. Devlin (1997, 43) esittää vielä yhden tiedon tyypin, joka organisaatioista voi löytyä: tieto tuotteena. Tällä tarkoitetaan kohdetta, joka periaatteessa on tietoa (esim. elektroniset kirjat, elokuvat) mutta jota kohdellaan kuten fyysistä tuotetta. Toisin sanoen tällainen tieto ei edusta tai kuvaa reaalimaailmaa tai liiketoimintaa.

Seuraavaksi esitellään pääasiallisesti Devlinin (1997) esittämiä luokitteluperusteita tietovaraston sisältämälle tiedolle. Tämän jälkeen niitä käsitellään tarkemmin aliluvuissa. Luvun lopuksi saadaan eroteltua tietovaraston eri tietotyypit ja eritoten liiketoimintatieto sekä metatieto.

3.1 Tietovarastotiedon luokitteluperusteet

Kuten aiemmin mainittiin, organisaatioiden sisältämä tieto voidaan Devlinin (1997, 42) mukaan jakaa kolmeen osaan pääasiallisesti tiedon merkityksen mukaan. Nämä ovat siis tieto tuotteena, liiketoimintatieto sekä metatieto. Tieto tuotteena ei kuulu tietovarastoon, koska se ei edusta reaalimaailmaa tai liiketoimintaa vaan se on liiketoiminnan tuote (Devlin 1997).

Devlin (1997) on näin erottanut toisistaan liiketoimintatiedon sekä metatiedon. Hän jatkaa liiketoimintatiedon pilkkomista ja asettaa luokitteluperusteita, joiden mukaan voidaan tyypitellä liiketoimintatietoa. Luokitteluperusteet voidaan myös johtaa helposti aikaisemmissa luvuissa esitellyistä tietovarastoympäristön ominaisuuksista. Organisaation tietoja voidaan kuvata ainakin seuraavasti:

- tiedolla on aina ajallinen ulottuvuus
- tiedolla on jokin granulariteetti
- tietoa käytetään jokapäiväisessä toiminnassa tai pidemmän aikavälin päätöksenteossa (operatiivinen vs. informatiivinen)
- tietoa voidaan joko sekä lukea että kirjoittaa, tai vain lukea (read-write vs. read-only)

- tieto tulee joko organisaation sisäisistä tai ulkoisista lähteistä
- tieto on tarkoitettu organisaatiossa joko yleiseen käyttöön tai vain yhden ihmisen käyttöön.

Edellä listatuista tiedon ominaisuuksista voidaan heti todeta, että tietovarastoympäristöön kuuluva tieto edustaa pitemmän aikavälin päätöksenteossa käytettävää informatiivista tietoa. Tämä asia käsiteltiin jo kohdassa 2.3. Samassa kohdassa tuotiin myös esille, että tietovarastoon kuuluva tieto on vain lukemista varten. Myös tiedon jako yksityiseen ja yleiseen käyttöön on selkeä: tietovarastoympäristössä ei ole Devlinin (1997) mukaan juurikaan yksityiseen käyttöön ja hallintaan tarkoitettua tietoa. Yksi perustelu yksityisten tietojen poislukemiseen Devlinin (1997) mukaan on se, että tällainen tieto on vain yhden yksilön käytössä: hän luo, käyttää ja tuhoaa tietoa mielensä mukaan. Jos tarkoituksena on siis rakentaa organisaation laajuinen (tai osastokohtainen) tietovarasto, niin tällöin pyrkimyksenä on saattaa kaikki tieto yhtenäiseen muotoon. Jos yksilöllä on valta päättää tiedon käytöstä, niin yksi tietovaraston perimmäisistä tavoitteista ei toteudu ja tiedon laatua ja eheyttä ei voida valvoa.

Muut osa-alueet kaipaavat tarkempaa analysointia ja niitä tarkastellaan seuraavassa tarkemmin.

3.2 Tiedon ajallisuus

Tieto edustaa aina jotain tiettyä aikaa. Tietovarastoympäristössä ajalla on vielä suurempi merkitys kuin tavallisissa operatiivisissa järjestelmissä. Kuten aiemmin määriteltiin, tietovarastotiedolla on aina ajallinen ulottuvuus.

Devlin (1997) jakaa tiedon ajallisuuden sen mukaan miten tieto sijoittuu liiketoiminnan aikajanalle: tieto voi edustaa nykyhetkeä eli olla reaaliaikaista, se voi edustaa jotain tiettyä hetkeä historiassa tai sitten tieto voi olla joltain tietyltä ajanjaksolta eli edustaa jotain tiettyä ajanjaksoa historiassa. Reaaliaikainen tieto

voi olla esimerkiksi tiettyjen tuotteiden tämän hetkinen määrä varastossa. Tieto, joka edustaa tiettyä hetkeä historiassa on otos yrityksen silloisista tiedoista ja tilasta. Tällainen tieto voi olla esimerkiksi edellisen päivän myyntitiedot. Tieto, joka edustaa jotain tiettyä ajanjaksoa kertoo miten liiketoiminta muuttuu ajan kuluessa. Tällainen tieto voi olla esimerkiksi koko viime vuoden myyntiluvut tai tietyn asiakkaan tiedot tietyltä ajanjaksolta. Tämän kaltainen jaksottainen tieto on välttämättä jonkinlaisen yhdistelyn ja summaamisen tulos. Pitkältä aikaväliltä kerätty tieto on informatiivista ja sen avulla voidaan tunnistaa trendejä, liiketoiminnan suuntaviivoja tai kehitystä. On ilmiselvää, että koska tietovarastoa ei päivitetä kuten operatiivisia järjestelmiä vaan tietoa ladataan tietyin väliajoin kuten kerran vuorokaudessa, ei se näin ollen voi sisältää reaaliaikaista tietoa. Se voi kylläkin sisältää varsin uutta tietoa mutta ei samalla tavalla reaaliaikaista kuten operatiiviset järjestelmät. Näin ollen reaaliaikainen tieto voidaan laskea pois varsinaisesta tietovarastosta. Täytyy kuitenkin muistaa, että reaaliaikainen tieto toimii syötteenä tietovarastolle ja näin ollen se tulee huomioida tietovarastoympäristössä.

On myös tärkeää päättää kuinka pitkään tietoja säilytetään tietovarastossa. Hovi (1997, 69) näkee säilytysajan riippuvan käyttäjien tietotarpeista. Toisin sanoen, mitä pidemmän aikavälin analysointia ja ajallisia vertailuja käyttäjät haluavat tehdä, sitä pidempään tietoja säilytetään tietovarastossa. Loppukäyttäjän tulisikin tietää mikä on tietovaraston tiedon aikaskaala niin, että hän tietää mitä kyselyitä on mahdollista tehdä (Inmon 1996). Kriittiset ja nykyhetken kannalta olennaiset tiedot edustavat nykyistä yksityiskohtaista tasoa (current detail information) (Gray & Watson 1998) ja ne tallennetaan aktiivisiin perustauluihin (Hovi 1997, 56). Vähemmän kriittinen tieto, johon ei enää kohdistu juurikaan kyselyitä, edustaa vanhempaa yksityiskohtaista tietoa (older detail) (Gray & Watson 1998) eli arkistoitua tietoa (Geiger ym. 1997; Hovi 1997). Geiger ym. (1997, 35) katsovat, että aktiivisissa perustauluissa oleva tieto eli nykyisen yksityiskohtaisuuden tason tiedot tulisi pitää on-line tilassa

vuodesta kahteen. Tämän jälkeen se tulisi siirtää toiselle medialle, jossa siihen voitaisiin myöhemmin halutessa päästä käsiksi.

3.3 Tiedon granulariteetti

Tiedon granulariteetti eli karkeusaste tarkoittaa tiedon yksityiskohtaisuuden tai summaamisen tasoa. Toisin sanoen karkeusaste lisääntyy kun tietoa summataan tai koostetaan.

Summaamisen ideana on, että tuotetaan informatiivisempaa tietoa. Summaamalla yksityiskohtaisempaa tietoa tiiviimpään muotoon helpotetaan ja nopeutetaan loppukäyttäjän pääsyä käsiksi hänelle tietyssä tilanteessa tarpeellisiin tietoihin (Geiger ym. 1997, 35). Gardner (1998) kuitenkin huomauttaa, että vaikkakin summattu tieto paljastaa ja tuo esille joitakin mahdollisesti olennaisia asioita, saattaa se piilottaa joitakin toisia. Toisin sanoen summattu tieto on relevanttia vain niin kauan kuin liiketoimintavaatimukset pysyvät ennallaan. Kun käyttäjien kyselytarpeet muuttuvat saatetaan kaivata yksityiskohtaisempaa tietoa, jota ei enää välttämättä ole saatavilla. Tällöin organisaatio ei pääse hyödyntämään sen informaatiovaroja.

Tiedon granulariteetti voidaan jakaa organisaatiossa kahteen pääluokkaan: yksityiskohtaiseen eli atomiseen sekä summattuun. Näistä yksityiskohtainen tieto edustaa organisaation tietojen alkeistasoa kuten asiakkaan tai tuotteen nimi. Tällainen tieto on yleistä juuri operatiivisissa järjestelmissä (Devlin 1997). Summattu tieto voidaan ainakin Grayn ja Watsonin (1998), Hovin (1997) sekä Inmonin (1996) mukaan jakaa vielä kevyesti summattuun sekä tiiviisti summattuun tietoon. Vaikkakin tietovaraston käyttäjät ovatkin kiinnostuneet informatiivisesta ja jalostetusta tiedosta, niin on huomioitava, että tietovarastoissa on kaikkia näitä rakeisuuden tasoja, myös atomista tietoa. Eri rakeisuusasteen omaavat tiedot esiintyvät usein kuitenkin arkkitehtuurin eri tasoilla.

Aktiivisissa perustauluissa eli varsinaisessa tietovarastossa on usein tietoa, joka on atomista ja yksityiskohtaista. Nämä tiedot ovat tärkeitä, koska joskus ilmenee tarvetta tehdä kyselyitä hyvin yksityiskohtaiseen tietoon, jopa päätöksentekotilanteissa. Tällöin ei ole kuitenkaan toivottavaa, että jumitetaan operatiiviset järjestelmät vaan siirretään tieto tietovarastoon ja kohdistetaan kyselyt sinne (Gray & Watson 1998). Aktiivisissa perustauluissa on myös mahdollista olla kevyesti summattua tietoa (Inmon 1996), vaikkakin yleensä summattu tieto esiintyy paikallisvarastotasolla. Suurin osa kyselyistä kohdistuu summattuihin tietoihin (Geiger 1997; Hackathorn 1995).

Inmon (1996) korostaa, että tiedon granulariteetti on tärkein yksittäinen osa-alue tietovaraston suunnittelussa. Organisaation tulisikin päättää millä tasolla se aikoo tietoja summata. Ei ole tarkoituksenmukaista pitää esimerkiksi jonkin tuotteen myyntitietoja päivätasolla jos suurin osa kyselyistä, eli käyttäjien tarpeista kohdistuu kuukausitasolle.

3.4 Sisäinen ja ulkoinen tieto

Aikaisemmin suurin osa liiketoiminnan kannalta relevantista tiedosta oli peräisin organisaation sisältä (Devlin 1997). Nykyään kuitenkin organisaatioiden on yhä tarpeellisempaa saada tietoa sen asiakkaista, kilpailijoista, lainsäädännöstä, hallinnollisista määräyksistä tai jopa säätilasta (Hackathorn 1999). Etenkin internetin ja www:n suosion takia on elektronisen tiedon määrä organisaatioissa kasvanut eksponentiaalisesti (Devlin 1997).

Bhowmick ym. (1999) uskovat, että www:n sisältämä informaatio on tärkeää liiketoiminnassa etenkin kun kyseessä ovat kriittiset päätöksentekotilanteet. Hackathorn (1999) menee pidemmälle ja nimeää www:n kaikkien tietovarastojen äidiksi.

Ballou ja Tayi (1999) katsovat kuitenkin, että ulkoinen relevantti tieto jätetään usein huomioimatta organisaatioissa, vaikka tuo tieto saattaisi olla erittäin

tärkeää monissa päätöksentekotilanteissa. Myös Blackwood (2000) näkee ulkoisen tiedon potentiaalin sillä ulkoinen tieto voi hänen mukaansa lisätä sisäisen tiedon arvoa. Yksi esimerkki tästä on organisaation myynnin kasvun vertaaminen markkinoiden kasvuun. Hackathorn (1999) jatkaa samaa teemaa ja toteaa, että sisäisen ja ulkoisen tiedon kombinaatiosta saatava synergia luo suurimman liiketoiminnallisen edun yritykselle. Organisaation tulisikin hänen mukaansa seurata aktiivisesti ulkomaailmaa. Hovin (1997) mukaan organisaation tulisi ainakin alustavasti kerätä ulkoisista lähteistä sellaisia kriittisiä tietoja, joita halutaan yhdistellä organisaation sisäisiin tietoihin tai jotka vaikuttavat tietovaraston tietojen johtamissääntöihin, kuten valuutta- tai pörssikurssit.

Inmonin (1996) mukaan tietovarasto on ideaalinen paikka tallentaa ulkoista tietoa. Näin ollen tieto tallennetaan keskitettyyn paikkaan ja sen koordinointi ja hallinta on helpompaa. Lisäksi etuna on se, että eri käyttäjät eivät tallenna samaa tai poikkeavaa tietoa järjestelmiin, vaan tieto syötetään keskitetysti yhdestä paikasta ja se käy läpi tarvittavat eheys- ja laatutarkistukset.

Edellä mainituista huomioista päätellen voidaan todeta, että tietovarasto sisältää sekä sisäistä että ulkoista tietoa. Pääpaino vielä nykyään on sisäisen tiedon keräämisessä tietovarastoon, mutta etenkin www:n yleistyttyä yhä enemmän kerätään (tai tulisi kerätä) relevanttia tietoa internetistä. Erittäin suuri merkitys on ulkoisen tiedon ja sisäisen tiedon yhdistelemisessä.

3.5 Tietoa vai metatietoa?

Edellä on käsitelty liiketoimintatietoa ja sen ominaisia piirteitä. Tietyn tyyppiset liiketoimintatiedot kuuluvat tietovarastoon ja tietyt tyypit jäävät sen ulkopuolelle operatiivisiin lähteisiin. Tämä tutkimus kuitenkin käsittelee tietovarastoympäristöä ja tällöin täytyy ottaa huomioon muitakin tiedon lajeja kuin varsinainen liiketoimintatieto. Tietovarastoympäristöön kuuluu muitakin tietotyyppejä, jotka toimivat tietovaraston tukena tai ovat olennainen osa sen

toimintaa. Tällaista tietoa ovat esimerkiksi tietovaraston monitorointitieto ja ilmoitustiedostot. Seuraavaksi käsitellään näitä tarkemmin, määritellään niiden funktio tietovarastossa ja määritellään mihin tiedon tyyppeihin ne kuuluvat.

Jotta voidaan hallita ja ylläpitää tietovarastoympäristöä, tulee monitoroida sen sisältämää tietoa sekä aktiviteetteja (Inmon 1996; 1997). Tietovarastoon tulee jatkuvasti lisää tietoa, joten jossain vaiheessa tulee poistaa osa epärelevanteista tiedoista joista ei enää olla kiinnostuneita. Tällöin tulee tietää mitkä tiedot ovat käyttäjille olennaisia ja mistä tiedoista he ovat kiinnostuneita. Lisäksi on tärkeää tietää kuinka nopeasti tietomäärät kasvavat ja eritoten mitä nimenomaisia tietoja kasvu koskee. Myös vasteajat, tietovaraston käyttöajankohdat ja käyttäjät ovat kiinnostuksen kohteina. Monitorointitieto kuuluu metatietoon, ainakin tämän tutkielman esittämän määritelmän mukaan, vaikkakaan Inmon (1996; 1997) ei sitä varsinaisesti näin määrittelekään. Mutta kuten aikaisemmin tässä tutkimuksessa todettiin, metatieto kuvaa sekä tietoa että tietovarastoympäristöä: käyttäjiä, komponentteja, prosesseja jne. Edellä mainitut monitorointitiedot kuuluvat täten varsin selvästi metatiedon piiriin.

Käyttäjien tiedonhakua voidaan helpottaa tekemällä niin sanottu ilmoitustiedosto (notification file), jonka Inmon (1996) katsoo liittyvän metatietoon. Ilmoitustiedosto sisältää listan kutakin käyttäjää kiinnostavista aihealueista tai tiedoista. Näin ollen, kun uutta tietoa tulee tietovarastoon, tarkastetaan ketä ko. tieto kiinnostaa ja kyseiselle henkilölle voidaan ilmoittaa tiedon saapumisesta tietovarastoon. Myös Hackathorn (1999) nostaa edellä mainitun kaltaisen ilmoitustiedon olennaiseksi. Hänen mukaansa muutosten havainnoiminen, etenkin ulkoisten lähteiden ollessa kyseessä, voi olla arvokkaampaa kuin itse tieto, joka muuttuu. Kuten sanottu, Inmon (1996) ei suoranaisesti lue ilmoitustiedon kuuluvan metatietoon, mutta tämän tutkimuksen määritelmän mukaan se voidaan nähdä olevan osa metatietoa. On ilmiselvää, että ilmoitustieto ei ole liiketoimintatietoa eli se ei edusta mitään

liiketoimintakohdetta tai aktiviteettia, mutta sen sijaan se helpottaa tietovaraston käyttöä, joten se sisältyy metatiedon määritelmään.

Jos siis oletamme, että metatietoa on kaikki se tieto, jota voidaan käyttää helpottamaan tietovaraston ylläpitoa, käyttöä ja kehittämistä, niin täten ylläpitoon tarkoitettu monitorointitieto sekä ilmoitustieto voidaan lukea metatiedoksi.

3.6 Yhteenveto

Tässä luvussa esiteltiin tietovarastoympäristön eri tietotyypit ja niiden ominaisuuksia. Luvun keskeisempänä tuloksena on erottelu liiketoimintatiedon ja metatiedon välillä. Lisäksi luvun kontribuutiona voidaan nähdä tietovaraston sisältämien tietojen ominaisuuksien määrittely ja eri ominaisuuksien merkitys tietovarastossa.

Tietovarastoympäristössä on hyvin monimuotoista tietoa sen eri arkkitehtuurin tasoilla. Tieto liikkuu ikääntyessään paikasta toiseen ja tietoa tulee myös ulkopuolisista lähteistä kuten www:stä.

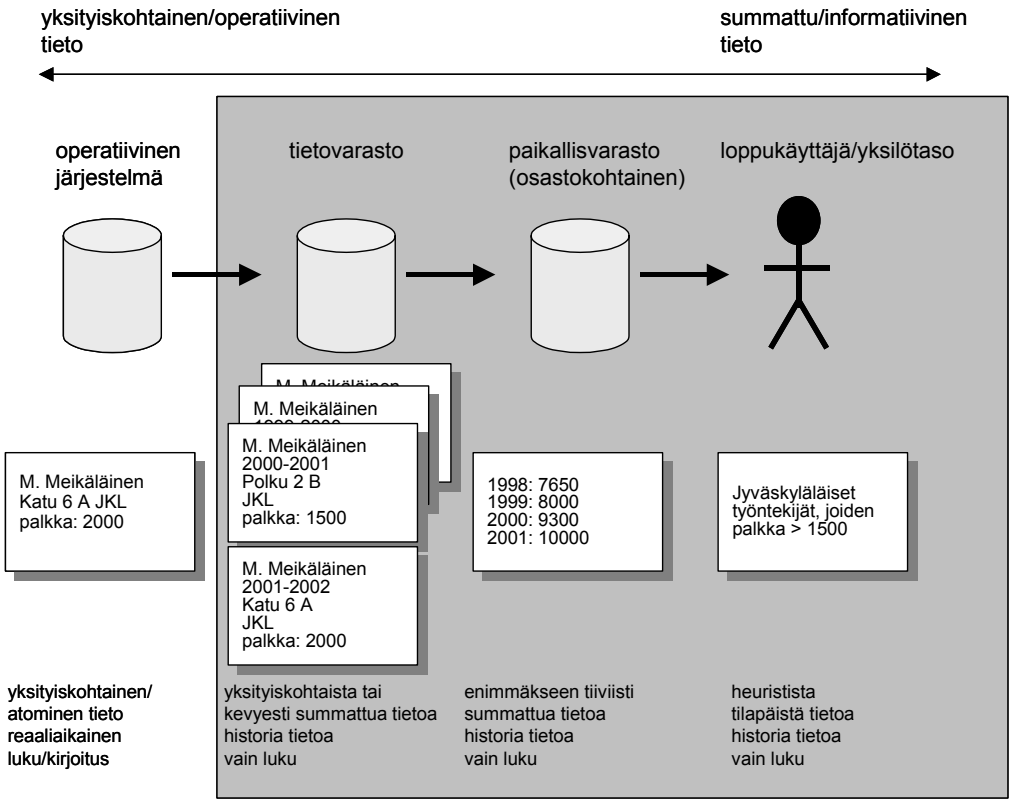
Organisaation sisältämä tieto voidaan jakaa liiketoimintatietoon sekä metatietoon. Liiketoimintatieto edustaa reaalimaailmaa eli liiketoiminnan toimintoja sekä prosesseja. Metatieto sen sijaan kuvaa liiketoimintatietoa, mutta myös itse tietovarastoympäristöä ja sen toimintaa. Liiketoimintatiedolla on siis edustava merkitys kun taas metatiedolla on kuvaava merkitys (Devlin 1997, 42).

Liiketoimintatieto voidaan jakaa eri kategorioihin eri kriteerien mukaan, joita ovat: tiedon ajallinen ulottuvuus, granulariteetti, käyttäjäkunta, käyttötapa sekä tiedon lähde.

Tietovarastoympäristössä ollaan kuitenkin kiinnostuttu vain osasta liiketoimintatiedon tyypeistä. Erityisesti tietovarastotieto on vain lukemista varten, informatiivista, yleiseen käyttöön tarkoitettua, historiallista sekä tieto

voi olla sekä atomista eli hyvin yksityiskohtaista että kevyesti tai tiiviisti summattua. Tieto voi olla myös peräisin sekä sisäisistä että ulkoisista lähteistä. Tietovarastoon ei tallennetta tietoa, joka on luku/kirjoitusta varten, yksityistä tai reaaliaikaista.

Kuvasta 5 nähdään tiivistetysti tietovarastoympäristössä sijaitsevan tiedon eri ominaisuuksia ja eri tietotyyppien sijainti tietovarastoarkkitehtuurin eri tasoilla. Lisäksi kuva 5 näyttää esimerkin summaamisesta.



Kuva 5: Eri tietotyyppejä arkkitehtuurin eri tasoilla (vrt. Inmon 1996, 19-21)

Esimerkistä nähdään kuinka operatiivisissa järjestelmissä esiintyy yksityiskohtaista tietoa, joka on relevanttia ja ajankohtaista juuri sillä hetkellä kun sitä tarkastellaan. Tässä tapauksessa tarkastellaan organisaation työntekijää. Tietovarastossa sen sijaan on tietyin väliajoin otettuja otoksia

tietolähteistä. Nämä otokset edustavat jotakin tiettyä hetkeä menneisyydessä eli ovat historiatietoa. Tässä tapauksessa kohdehenkilö M. Meikäläisestä on tallennettuna tietovarastoon mm. hänen osoitteensa sekä palkkansa. Osastokohtaisella tasolla ollaan kiinnostuttu yleisemmästä tiedosta ja sinne onkin summattu tässä tapauksessa eri vuosilta osaston palkkakustannukset eli laskettu yhteen kaikkien osaston työntekijöiden palkat. Tässä tapauksessa yksilötasolla käyttäjä haluaa hakea kaikki Jyväskylässä asuvat työntekijät, joiden palkka on suurempi kuin 1500 euroa.

On huomioitava, että toisen sovelluksen tai henkilön metatieto voi olla toiselle tietoa (Gilliland-Swetland 1998; Hackathorn 1999). Täten jaottelu tiedon ja metatiedon välillä ei ole absoluuttinen. Silti edellä tehty erottelu antaa paremman käsitteellisen pohjan ymmärtää tietovaraston sisältämää ja sitä ympäröivää tietoa sekä toisaalta se auttaa paremmin ymmärtämään käsitettä metatieto, johon syvennytään paremmin seuraavassa luvussa.

4 METATIETO

Kaikki tietovaraston komponentit, käyttäjät ja ohjelmistot hyödyntävät ja/tai tuottavat metatietoa (Jarke ym. 2000; Vaduva & Vetterli 2001). Metatiedon rooli tietovarastoympäristössä poikkeaakin huomattavasti sen roolista operatiivisissa ympäristöissä (Inmon 1996; Tannenbaum 2002). Tietovarastoympäristössä metatietoa pidetään jopa elinehtona (Geiger ym. 1997). Tämän luvun päätavoitteena on näyttää, mitä kaikkea metatietoa tietovarastoympäristöön kuuluu ja selvittää, miksi metatieto on elinehto tietovarastoympäristössä.

Ilman metatietoa tietovarasto ja siihen liittyvät komponentit ovat erillisiä palikoita työskennellen itsenäisesti ja erillisin tavoittein (Inmon 1997). Onkin olennaista ymmärtää eri metatietotyyppien kriittiset roolit tehokkaiden, yhteistoiminnallisten ja laajennettavien tietojärjestelmien kehityksessä (Gilliland-Svetland 1998).

Metatiedon määritellään yleisesti olevan tietoa tiedosta (mm. Gardner 1998, 59; Hackathorn 1999, s. 165; Jacob & Sen 1998, s. 31; Lassila & Swick 1999). Metatiedon voidaan nähdä antavan niin sanotulle raakatiedolle merkityksen eli luovan siitä informaatiota (Kim ym. 2001). Toisaalta metatieto auttaa yleisemmin tiedon identifioimisessa, hakemisessa, hallitsemisessa sekä hyödyntämisessä. Johtuen muun muassa tietovarastoympäristön monimutkaisuudesta, teknologioiden ja ohjelmistojen suuresta määrästä, loppukäyttäjistä sekä varsinaisen tiedon monimuotoisuudesta, metatiedolla on hyvin laaja sekä tärkeä merkitys tietovarastoympäristössä. Metatieto ei kuvaa siis vain tietoa vaan myös itse tietovarastoympäristöä ja sen kehittymistä, komponentteja, prosesseja ja laitteistoja.

Tämän luvun tavoitteena on antaa kattava kuva metatiedosta tietovarastoympäristössä. Ensinnäkin kohdassa 4.1 näytetään miten aikaisemmissa tutkimuksissa ja kirjallisuudessa metatietoa on luokiteltu.

Tarkoituksena on syntetisoida eri tutkijoiden havaintoja metatietotyypeistä ja tuoda esille mitä metatietoa tietovarastoympäristössä tyypillisesti esiintyy. Kohdassa 4.2 ja sen alakohdissa tuodaan esille metatiedon merkitys ja rooli tietovarastoympäristön eri osa-alueilla. Tässä tapauksessa tarkastelun kohdetta laajennetaan käsittämään myös tietovaraston kehitykseen liittyvät näkökulmat.

4.1 Metatiedon tyypit

Tietovarastoympäristössä esiintyy hyvin erilaista ja monen tyyppistä metatietoa. Tässä kohdassa käydään ensiksi läpi aikaisemmissa tutkimuksissa tehtyjä luokitteluja metatiedolle ja esitellään lyhyesti mitä metatietoa nämä luokat eli metatietotyypit sisältävät. Eri metatietotyypit sisältävät päällekkäisyyksiä ja luokittelu ei pyri olemaan ehdoton. Luokittelujen esittely paljastaa kuitenkin eri näkökulmia metatietoon. Tämän jälkeen muodostetaan viitekehys, josta käy ilmi tarkemmin, mitä kaikkea metatietoa on mahdollista kuvata tietovarastoympäristössä.

Metatietoa tietovarastoissa voidaan luokitella seuraavien kriteerien perusteella:

- käyttö: passiivinen, aktiivinen ja puoliaktiivinen
- kohde: tieto ja prosessit
- abstraktiotaso: käsitteellinen, looginen ja fyysinen
- käyttäjäryhmä: tekninen ja liiketoiminnallinen.

Käytön mukaan voidaan metatieto jakaa kolmeen osaan: aktiivisesti, passiivisesti ja puoliaktiivisesti käytettävään metatietoon (Devlin 1997; Huynh ym. 2000; Vaduva & Vetterli 2001). *Passiivisesti* käytettävä metatieto tarjoaa yhtenäisen dokumentaation tietovarastoympäristöstä: sen rakenteesta, kehittämisprosessista ja käytöstä. Passiivinen metatieto tukee kaikkia tietovaraston hyödyntäjiä (ts. loppukäyttäjiä, ylläpitoa ja järjestelmäkehittäjiä). *Aktiivisesti* käytettävä metatieto toimii niin sanottuna kontrolli-informaationa. Se sisältää transformointisääntöjä tai metodeja, joita ohjelmat tulkitsevat ja toteuttavat ohjelmien ajonaikana. Tällöin voidaankin puhua niin sanotusta

metatieto-johtoisesta ohjelmistosta (Kietz ym. 2001). Tämä tarkoittaa, että ohjelman toiminta on jossain määrin riippuvainen sen ulkopuolella määritellystä metatiedosta. *Puoliaktiivinen* metatieto sisältää staattista dokumentaatiota, kuten tietorakenne- tai konfiguraatiomäärityksiä, joita muut ohjelmat voivat lukea ajonaikana. Poiketen aktiivisesta metatiedosta, puoliaktiivista metatietoa vain luetaan, ei ajeta itsessään.

Metatiedolla on jokin kohde, jota se kuvaa. Kohde voi olla mikä tahansa ajateltavissa oleva entiteetti tietovarastoympäristössä: tietokanta, tiedosto, taulu, sarake, ihminen, prosessi tai vaikka ohjelma. Pääasiallisesti metatieto luokitellaan kuvaamaan joko tietoa tai prosesseja. Kolmas luokka voisi olla myös metatietoa ympäristöstä (Mohania ym. 1998), mutta tämä voidaan myös lukea kuuluvaksi tieto-kategoriaan (Vaduva & Vetterli 2001). *Metatieto tiedosta* kuvaa kaikkea sitä tietoa, joka sijaitsee tietolähteissä, tietovarastossa tai paikallisvarastoissa. Lisäksi se voidaan nähdä kuvaavan myös muita staattisia kohteita (poiketen siis prosesseista), kuten laitteistoja, tietokantoja tai ohjelmia. Metatieto tiedosta sisältää paljolti samoja metatiedon ilmentymiä kuin edellä mainittu passiivinen metatieto, kuten tietovaraston taulujen kaavat, tilastollista tietoa tietovaraston koosta tai esimerkiksi kuvaus mistä mikin tieto löytyy. Lisäksi otettaessa huomioon organisaatio- tai ulkoinen ympäristö, metatieto voi kuvata organisaatiohierarkiaa, käyttöoikeuksia, käyttäjiä, laillisia rajoitteita jne. *Metatieto prosesseista* kuvaa tiedon prosessointia tietovarastossa. Tämä sisältää informaatiota koskien tiedon poimimista, transformointia ja lataamista. Mukaan luetaan myös kuvaus virkistämis-, summaus- sekä analyysiprosessista. Metatieto prosesseista menee paljolti päällekkäin aktiivisen ja puoliaktiivisen metatiedon kanssa. On kuitenkin huomioitava, että metatieto prosesseista voi olla myös passiivisia kuvauksia prosesseista.

Metatietoa voidaan kuvata kolmella eri abstraktiotasolla: käsitteellisellä (conceptual), loogisella (logical) sekä fyysisellä (physical) (Elmasri & Navathe 2000, 25; Vaduva & Vetterli 2001). *Käsitteellinen taso* sisältää luonnollisella

kielellä ilmaistun kuvauksen liiketoiminnasta eli liiketoimintamallin. Tämä sisältää kuvaukset oleellisimmista liiketoimintakohteista (asiakas, tilaus). Myös suhteet eri kohteiden välillä on kuvattu. Lisäksi tälle tasolle kuuluu selkokielellä ilmaistut kuvaukset tiedon prosessoinnista sekä ennalta määritellyistä kyselyistä tai olemassa olevista analysointiohjelmista. Käsitteellisen tason tietoja voidaan kuvata esimerkiksi ER-kaavioiden avulla. *Looginen taso* muuntaa käsitteellisen näkymän alemmalle tasolle. Tämä taso sisältää esimerkiksi tietolähteiden relaatiokaavat tai transformointiprosessien kuvaukset pseudokoodilla. *Fyysinen taso* kuvaa toteutustasoa. Tämä taso sisältää ylemmillä tasoilla kuvattujen prosessien vastaavat ohjelmakoodit ja kuvaukset niistä, indeksitiedostot tai esimerkiksi analysointiohjelmien ohjelmakoodit. On huomioitava, että metatiedolla tulisi kuvata myös käsitteellisen tason ja vastaavan fyysisen tason väliset yhteydet (Müller, Rahm & Stöhr 1999). Toisin sanoen jos käsitteellisellä tasolla on kuvattu jokin liiketoimintasääntö, joka ilmenee fyysisellä tasolla SQL-lauseena, tulee tämä yhteys ilmetä metatiedosta. Vaduva ja Vetterli (2001) näkevät, että fyysinen ja looginen taso voidaan yhdistää, jos erottelu ei ole perusteltua käyttäjien kannalta.

Kenties yleisin tietovaraston metatiedon jaottelu koskee metatiedon käyttäjäkuntaa ja käyttäjien eri tarpeita. Käyttäjien mukaan metatieto voidaan jakaa tekniseen ja liiketoiminnalliseen/semanttiseen (mm. Hess & West 2002; Kim ym. 2001; Mohania ym. 1998; Müller, Rahm & Stöhr 1999; Staudt ym. 2000; Vaduva & Vetterli 2001). *Tekninen metatieto* on tarkoitettu pääasiallisesti sellaisille tietovaraston käyttäjille, jotka ovat kiinnostuneita metatiedosta teknisen toteutuksen kannalta eivätkä niinkään vastaavista liiketoimintamääritelmistä. Tällaisia käyttäjiä ovat esimerkiksi suunnittelijat, järjestelmän kehittäjät, ylläpito tai ohjelmoijat. Lisäksi teknisen metatiedon käyttäjiin kuuluu myös komponentit ja ohjelmistot, jotka hyödyntävät metatietoa toiminnassaan. Tekninen metatieto sisältää esimerkiksi tietokannan

kaavojen kuvaukset, tietorakenteet, tietoa fyysisestä tallennustavasta ja ajonaikaista tietoa kuten loki-tiedostot. *Liiketoiminnallinen metatieto* on tarkoitettu pääasiallisesti loppukäyttäjille, jotka eivät ole usein tietotekniikan ammattilaisia, vaan varsinaisen liiketoiminnan tuntijoita. He tarvitsevat metatietoa, joka on selkokielistä ja helposti ymmärrettävää. Liiketoiminnallinen metatieto kuvaa tietovarastoa, sen prosesseja ja tietoja liiketoiminnan kannalta ja luonnollisella kielellä. Tämä metatieto on erityisen tärkeää loppukäyttäjän toiminnan kannalta ja auttaa häntä navigoimaan tietovaraston tietomassojen seassa ja ymmärtämään paremmin haettuja tietoja. Liiketoiminnallista metatietoa ovat esimerkiksi sanalliset kuvaukset ennalta määrätyistä raporteista, selkokieliset kuvaukset poiminto-, transformointi- ja summausprosesseista sekä kuvaus tietojen lähteistä. Müller ym. (1999) näkevät, että liiketoiminnallisen metatiedon tehtävänä on toisaalta tarjota liiketoiminnallinen näkymä tekniseen metatietoon ja toisaalta kuvata sellaiset tiedot liiketoimintatermein, joita tekninen metatieto ei ota huomioon.

Kuvassa 6 on yhdistelty aikaisemmissa tutkimuksissa esiintyviä metatietotyyppejä esimerkkeineen. Kuvasta on selvyyden vuoksi jätetty pois eri

	tekninen	liiketoiminnallinen
tieto	• tietomallit (operatiivisista lähteistä, tietovarastosta yms.) • taulujen kaavat ja nimet • kuvaus attribuuttien rajoitteista • tiedostorakenne • konfiguraatio spesifikaatiot • kuvaus tietorakenteiden/tietojen muutoksista	• käsitteellinen organisaation laajuinen tietomalli • liiketoimintakäsitteet • kuvaus tietovaraston sisällöstä • summaustaso • tiedon ajantasaisuus/aikaleima
prosessit	• tekniset kuvaukset poiminto-, transformointi- sekä summausprosesseista • kuvaus herättimistä (triggers) • prosessimallit • pseudokoodi • monitorointitieto (aktiviteeteista)	• liiketoiminnalliset kuvaukset poiminto-, transformointi- sekä summausprosesseista • tietolähteiden päivitystiheys • kuvaus ennaltamääritellyistä SQL-kyselyistä • prosessikaaviot

Kuva 6: Metatiedon tyyppejä

käsitteelliset tasot. Pystysuunnassa metatieto on luokiteltu kohteen mukaan, jota metatieto kuvaa. Tässä tapauksessa siis metatiedon kohteet on jaettu tietoon ja prosesseihin. Luokittelu voitaisiin pilkkoa pienempiin osiin niin, että tulisi ilmi kaikki mahdolliset kohteet, joita metatiedolla on mahdollista kuvata. Vaakasuunnassa metatieto on luokiteltu käyttäjäryhmien ja –tarpeiden mukaan. Metatiedon luokittelu käytön mukaan (passiivinen, aktiivinen ja puoliaktiivinen) on myös jätetty kuvasta pois sen vuoksi, että passiivinen metatieto voi kuulua mihin tahansa muuhun luokkaan. Täten sitä ei tarvitse erottaa omaksi luokaksi. Lisäksi aktiivinen ja puoliaktiivinen metatieto on selkeästi teknistä metatietoa. Nämä tyypit sisältävät selvästi teknisiä kuvauksia, joita esimerkiksi ohjelmat hyödyntävät käytössään.

Metatiedon luokittelu eri kategorioihin ei ole kuitenkaan täysin ehdoton. Toisin sanoen rajat eri luokkien välillä ovat hämärät sillä samaa metatietoa voidaan käyttää useaan tarkoitukseen. Esimerkiksi tietovaraston taulun mallin määritystä (i.e. metatietoa taulusta) voidaan hyödyntää ylläpitoon, analysointiin tai taulun tietojen lataamiseen tauluun (Vaduva & Vetterli 2001).

Edellä esitetystä luokittelusta sekä kuvasta 6 voi huomata, että samaa metatietoa on sekä teknisellä että liiketoiminnallisella puolella. On kuitenkin huomioitava, että liiketoiminnallinen osa käsittää siis yleensä luonnollisella kielellä ilmaistut kuvaukset. Ihanteellisinta olisi, jos esimerkiksi liiketoimintasäännöt kuvattaisiin luonnollisella kielellä loppukäyttäjää varten, mutta samalla tarjottaisiin tekniselle henkilöstölle formaalimmat tekniset määritykset. Lisäksi nämä määritykset tulisi linkittää vastaavien transformointiprosessien kuvauksiin (Staudt ym. 2000). Erityisen tärkeää onkin pystyä kuvaamaan eri metatietotyyppien väliset suhteet (Mohania ym. 1998; Müller ym. 1999). Toisin sanoen metatiedolla tulisi kuvata miten tietty liiketoiminnallinen määritys vastaa mitäkin teknistä määritystä.

Grossman ja McCabe (1996) sekä Marco (2001) tähdentävät, että tietovaraston metatietojen ylläpito tulisi pyrkiä hoitamaan automaattisesti. Näin varmistetaan ajantasainen sekä täsmällinen metatieto. Jos metatiedon ylläpito hoidetaan liiaksi manuaalisesti saattaa se jopa estää tai myöhästyttää koko metatietoratkaisun käyttöönottoa. Mullen (2002) on vieläkin kriittisempi todetessaan, että mikä tahansa metatietolähestymistapa, joka pohjautuu metatiedon manuaaliseen keräämiseen on tuomittu epäonnistumaan.

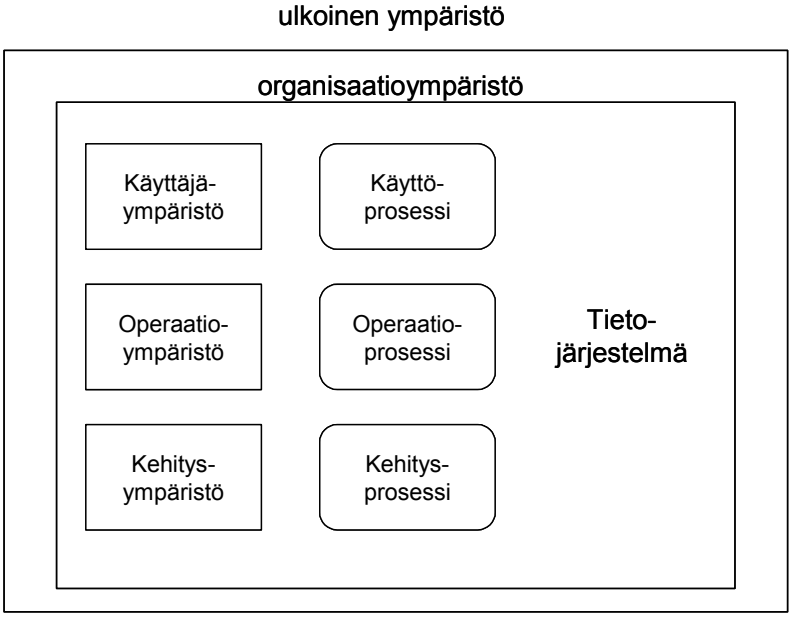
Edellä konstruoitu viitekehys ei pyri ehdottamaan tarkasti rajattua luokittelua metatiedolle. Sen sijaan tarkoituksena on tuoda esille eri näkökulmia sekä käsitteitä, joilla tietovaraston metatietoja voidaan hahmotella ja niistä voidaan keskustella mielekkäästi. Kuten edellä on näytetty, erilaista metatietoa on todella paljon ja sitä voidaan tarkastella hyvin useasta näkökulmasta. Myös sellaisia metatiedon tyyppejä ja esimerkkejä on otettu mukaan, joita kaikki eivät välttämättä ole ennen luokitelleet metatiedoksi. Kaikki tyypit ovat kuitenkin tärkeitä tietovaraston hallinnassa ja tehokkaassa hyödyntämisessä. Seuraavaksi tarkastellaan, miten metatieto tukee ja helpottaa tietovaraston kehittämistä, ylläpitoa ja käyttöä. Toisin sanoen seuraavaksi näytetään miksi metatieto on tietovarastojen elinehto.

4.2 Metatiedon merkitys tietovarastoinnissa

Monimutkaisten tietojärjestelmien, kuten tietovarastoympäristöjen ollessa kyseessä, metatiedon helppo saatavuus sekä jakaminen ovat erittäin tärkeässä asemassa järjestelmien käytössä, ylläpidossa sekä rakentamisessa (Kietz ym. 2001). Metatietoa tarvitaan helpottamaan muun muassa tietojärjestelmien analysointia, suunnittelua, käyttöönottoa sekä varsinaista käyttöä (Vaduva & Vetterli 2001). Toisin sanoen metatiedolla on erityinen merkitys koko tietovaraston elinkaaressa.

Edellisissä luvuissa on määritelty mitä eri komponentteja ja osa-alueita tietovarastoympäristöön kuuluu. Lisäksi on määritelty minkälaista tietoa

tietovarastoissa esiintyy. Tässä kohdassa tarkastellaan mikä merkitys metatiedolla on näille eri tietovarastoinnin osa-alueille. Mutta koska tietovarastointiin liittyy muutakin kuin edellä esitelty arkkitehtuuri ja sen sisältämät tiedot, täytyy tietovarastointia tarkastella hieman laajemmalta kannalta. Tätä varten tässä luvussa metatiedon merkitystä tietovarastoympäristössä käsitellään Davisin, Hamiltonin ja Ivesin (1980) esittämän tietojärjestelmien tutkimisen mallin mukaan (KUVA 7). Lyytisen (1987) mukaan tämä malli on hyvin tunnettu ja se ottaa huomioon myös tietojärjestelmien kehittämisen. Malli auttaa ymmärtämään paremmin tietojärjestelmien sekä niiden ympäristöjen moninaisia piirteitä.



Kuva 7: Tietojärjestelmien tutkimuksen malli (Davis, Hamilton & Ives 1980, 917, mukaan)

Malli erottaa kuvan 7 mukaisesti tietojärjestelmän sekä siihen liittyvät ympäristöt että prosessit. Tietojärjestelmään liittyy mallin mukaisesti ulkoinen, organisaatio-, käyttäjä-, kehitys- sekä operaatioympäristö. Tietojärjestelmään liittyviä prosesseja ovat käyttöprosessi, operaatioprosessi sekä kehitysprosessi.

Prosessien kautta tietojärjestelmä on vuorovaikutuksessa sen eri ympäristöjen kanssa.

Seuraavissa kohdissa esitellään mallin eri osa-alueet ja liitetään ne tietovarastokontekstiin. Tavoitteena on asettaa metatieto laajempaan yhteyteen, jossa tulee ilmi myös tietovaraston kehittäminen sekä eri ympäristöt, jotka tulee huomioida. Tarkastelu tapahtuu seuraavaksi kahdesta näkökulmasta. Mallin avulla voidaan esittää ja tarkastella:

a) Mitä seikkoja metatiedolla tulisi kuvata kustakin ympäristöstä ja prosessista.

b) Miksi metatieto on tärkeää kullakin tietovarastoinnin osa-alueella?

Ensimmäisessä näkökulmassa esille tulevat seikat ovat verrattavissa kohdassa 4.1 esiteltyyn metatiedon luokitteluun kohteen mukaan. Toisin sanoen tässä tapauksessa tuodaan kohtaa 4.1 laajemmin ja eri näkökulmasta esille niitä relevantteja kohteita, joita tietovarastoympäristöstä tulisi kuvata. Toisessa näkökulmassa, eli käsiteltäessä metatiedon merkitystä tietovarastoinnissa, tarkastellaan mikä merkitys metatiedolla on eri prosesseille (käyttö-, operaatio- ja kehitysprosessi), koska mallin mukaisesti juuri prosessien kautta eri ympäristöt käyttävät ja ovat yhteydessä varsinaiseen tietojärjestelmään. Kohdissa tuodaan esille mikä merkitys metatiedolla on kullekin osa-alueelle eri käyttöskenaarioiden muodossa.

Alakohdat etenevät seuraavasti: ensiksi esitellään tietojärjestelmien tutkimuksen mallin eri osa-alueiden ominaisuudet Davisin ym. (1980) mukaan. Tämän jälkeen näytetään, mitä metatietoa kustakin osa-alueesta (eli ympäristöstä ja prosesseista) tulisi kuvata. Lopuksi prosessien osalta esitetään, miksi metatieto on niissä tärkeää.

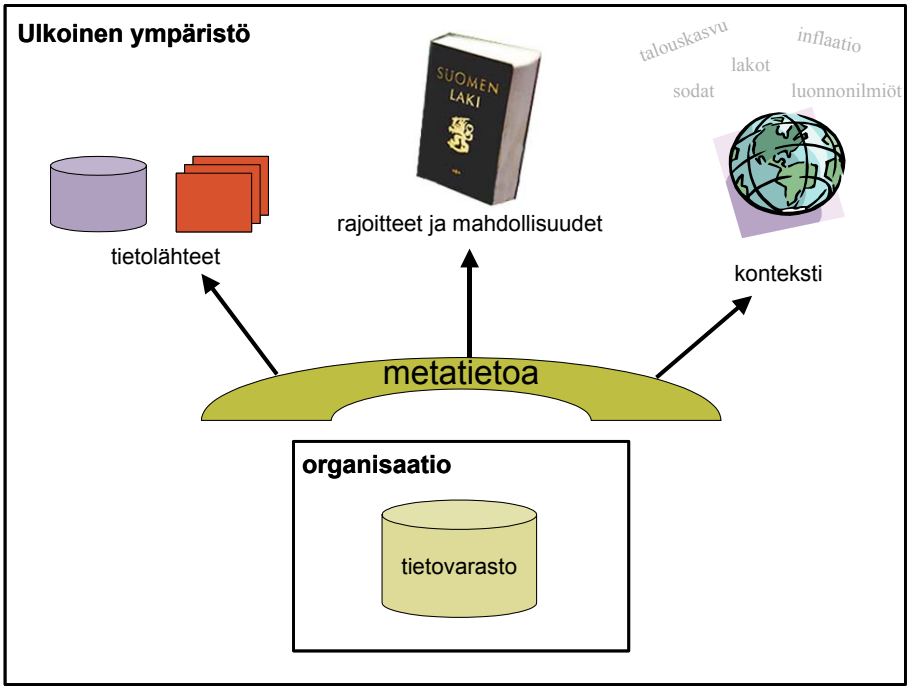
4.2.1 Metatieto ja ulkoinen ympäristö

Ulkoisella ympäristöllä tarkoitetaan sitä ympäristöä ja tekijöitä, jotka vaikuttavat organisaatioon. Ulkoisesta ympäristöstä tulee Davisin ym. (1980) mukaan huomioida esimerkiksi lailliset, sosiaaliset, kulttuurilliset, opetukselliset, taloudelliset tai teolliset näkökulmat. Ulkoinen ympäristö voi asettaa rajoitteita tietojärjestelmälle, tässä tapauksessa siis tietovarastolle. Esimerkiksi yksityisyyslaki tai tietyillä aloilla raportointivelvollisuudet asettavat rajoitteita tietovaraston kehittämiselle sekä käytölle. Mutta toisaalta ulkoisesta ympäristöstä tulee ottaa huomioon esimerkiksi uudet kaupalliset ohjelmistopaketit, jotka voivat toimia resurssina tietovaraston kehittämiselle.

Mahdollisten rajoitteiden sekä mahdollisuuksien lisäksi tietovarastoon vaikuttavat ulkoisen ympäristön sisältämät tietolähteet, joita käsiteltiin jo hieman kohdassa 3.4. Lisäksi huomiota täytyy kiinnittää niin sanottuun ulkoiseen kontekstiin. Seuraavaksi käsitellään tarkemmin metatiedon roolia kunkin edellä mainitun kolmen osa-alueen kohdalta eli rajoitteiden ja mahdollisuuksien, ulkoisen kontekstin sekä ulkoisten tietolähteiden (ks. KUVA 8).

Riippuu paljolti toimialasta, kuinka paljon ulkoinen ympäristö asettaa rajoitteita tietovaraston hyväksikäytölle, mutta on selvää, että ulkoisen ympäristön asettamat rajoitteet kuten lainsäädännölliset määräykset tulisi kirjata ylös ja pitää ajan tasalla. Tämän kaltainen tieto ei välttämättä asetu kaikkien mielessä metatiedon kategoriaan. Mutta jos ajatellaan tämän tiedon auttavan ja tietyssä mielessä säätävän ja rajoittavan tietovaraston kehittämistä ja hyödyntämistä, niin silloin se johdannossa asetetun määrityksen mukaan kuuluu myös metatietoon.

Kuten jo aikaisemmin on mainittu, tietovarastoon kerätään tietoa hyvin pitkältä aikaväliltä. Inmonin (1996) mukaan on hyvin yleistä, että tietoa säilytetään 5-10 vuotta tai jopa pidempään tietovarastossa. Koska tiedolla on näin pitkä historia,



Kuva 8: Metatieto ulkoisen ympäristön kuvaamisessa

saattaa se vaikeuttaa tietovaraston tietojen ymmärtämistä. Tällöin tiedon analysoijan tulisi pystyä liittämään tieto siihen asiayhteyteen eli kontekstiin, joka vallitsi silloin kun tieto ladattiin tietovarastoon. Inmon (1996) nimittää tätä ulkoiseksi kontekstuaaliseksi tiedoksi. Seuraava esimerkki valaisee käsitettä.

Oletetaan, että loppukäyttäjä haluaa analysoida organisaation liikevaihtoa viimeisten kymmenen vuoden ajalta. Koska tietovarasto sisältää historiatietoa on helppo tehdä halutut raportit. Mutta loppukäyttäjä saattaa yllättyä jos esimerkiksi vuoden 1995 liikevaihto näyttää 200000 ja vuoden 1997 vastaava luku 2000000. Toisin sanoen organisaation liikevaihto näyttäisi kymmenkertaistuneen, mikä on sangen poikkeuksellinen muutos. Loppukäyttäjä saattaisi helposti luulla, että kyseessä on virhe. Mutta saattaa ollakin, että vuoden 1995 ja 1997 välillä organisaatio osti kaksi muuta yritystä, kasvattaen näin liikevaihtoaan huomattavasti. Lisäksi saattaa olla, että organisaation pahin kilpailija on tehnyt konkurssin ja sen kaikki asiakkaat ovat

siirtyneet ko. yrityksen asiakkaiksi. Toisena yksinkertaisena esimerkkinä voidaan ajatella organisaatiota, joka valmistaa jäätelöitä. Saattaisi hyvinkin olla, että yrityksen liikevaihto jäätelöiden osalta olisi viime kesänä ollut poikkeuksellisen suuri. Loppukäyttäjä, joka analysoi viime kesän myyntilukuja ja vertaa niitä aikaisempien vuosien lukuihin saattaisi epäillä jälleen poikkeuksellisia tuloksia. Mutta jos hänelle kerrottaisiin, että viime kesä oli lämpimin sataan vuoteen, ymmärtäisi hän paremmin lukujen taustalla vaikuttavat seikat. Toisin sanoen organisaation tulisi olla tietoinen ulkoisesta ympäristöstään ja mahdollisesti kuvata oleellisia tietoja, jotta organisaation omat tiedot voidaan liittää oikeaan asiayhteyteen. Niin kasvusuhdanteet, inflaatio, teknologiset läpimurrot, kuin kilpailijoiden toimenpiteet voivat olla seikkoja, jotka tulee huomioida analysoidessa historiatietoa (Inmon 1996, 190). Ulkoisen kontekstuaalisen tiedon eli metatiedon kerääminen näistä seikoista voi olla tietyissä organisaatioissa erittäin tarpeellista.

Metatiedon merkitys ulkoisten tietolähteiden osalta on Inmonin (1996) mukaan erityisen suuri. Metatiedon avulla ulkoinen tieto rekisteröidään, siihen päästään käsiksi ja sitä kontrolloidaan. Kaikkea ulkoista tietoa ei välttämättä kannata eikä tarvitse tallentaa tietovarastoon. Metatiedon avulla voidaan kuvata mistä varsinainen tieto löytyy (Inmon 1996). Näin ollen tieto voidaan tarvittaessa hakea analysointia varten mutta sitä ei tarvitse säilyttää varsinaisessa tietovarastossa. Käyttäjät voivat myös analysoida pelkästään metatietoa ja tarkastella löytyykö haluttua tietoa mahdollisesti ulkoisista lähteistä. Näin ollen käyttäjien ei tarvitse välttämättä edes katsoa varsinaista lähdedokumenttia.

4.2.2 Metatieto ja organisaatioympäristö

Organisaatioympäristö käsittää Davisin ym. (1980) mukaan organisaation tavoitteet, tehtävät, rakenteen sekä johtamistyylin. Toisin sanoen nämä seikat vaikuttavat tietojärjestelmän kehittämiseen. Esimerkiksi kehittämistyö keskitetyssä organisaatiossa eroaa kehittämisestä hajautetussa organisaatiossa.

Organisaatioympäristöstä kumpuavat myös liiketoimintatarpeet, jotka ovat tietotarpeiden taustalla. Nämä taasen vaikuttavat siihen, mitä tietoa esimerkiksi tietovarastoon sisällytetään.

Metatiedolla on tärkeä rooli organisaatioympäristön kuvaamisessa. Seuraavassa muutamia havaintoja missä metatiedon tärkeä merkitys tulee jälleen esille.

Tietovarastoja kehitetään liiketoimintatarpeita varten ja niiden tavoitteena on vastata käyttäjien kyselyihin. Tällöin on itsestään selvää, että organisaation tavoitteet, motiivit ja tehtävät on julkilausuttu koska ne vaikuttavat tietovaraston kehittämiseen. Kuten jo kohdassa 2.2.2 mainittiin, organisaatioiden tulisi rakentaa koko yrityksen laajuinen tietomalli (tai tietoarkkitehtuuri), jonka pohjalta tietovarasto rakennettaisiin. Tämä tietomalli kuvastaa yrityksen tietotarpeita ja toimii tietovaraston rakentamisen lähtökohtana. Tietovaraston hyödyntämisen ja kehittämisen kannalta tämän kaltainen tieto on metatietoa (Müller ym. 1999). Edelleen yleisimmät ja johtavimmat motiivit ja tavoitteet voi myös kirjata ylös, kuten mm. Geiger ym. (1997) suosittelevat. Organisaation laajuinen tietomalli tai kuten White (1999) sen nimeää, yleinen liiketoimintamalli (common business model) auttaa myös suunnitteluhenkilöstöä, kun he keskustelevat loppukäyttäjien kanssa heidän tietotarpeistaan. White (1999) näkee, että tietomalli auttaa selvittämään paremmin käyttäjien tietotarpeet.

Organisaatiot kuitenkin muuttuvat jatkuvasti ja näin ollen niiden liiketoimintatarpeet ja edelleen tietotarpeet vaihtuvat. Tämän vuoksi tietovarastoympäristön hallinnoinnissa on tarpeen seurata muuttuvaa liiketoimintaympäristöä (Hackathorn 1995) ja täten myös organisaation tavoitteita, motiiveja sekä näistä kumpuavia tietotarpeita. On huomioitava siis, että metatieto kuvaa esimerkiksi tietovarastoa sekä sen sisältämiä tietoja mutta myös itse organisaation liiketoimintaa (Mullen 2002).

4.2.3 Metatieto ja käyttäjäympäristö sekä –prosessit

Käyttäjäympäristö ympäröi ja pitää sisällään tietojärjestelmän pääasialliset käyttäjät. Eri käyttäjiä voidaan luokitella muun muassa heidän tehtäviensä mukaan (Davis ym. 1980). Tietovarastoympäristössä pääasialliset käyttäjät ovat siis liiketoiminnan ammattilaisia, jotka analysoivat tietovaraston tietoja.

Käyttöprosessi sen sijaan tarkoittaa loppukäyttäjän suorittamaa tietojärjestelmän käyttöä. Tätä osa-aluetta ja sen toimivuutta voidaan mitata sen mukaan kuinka hyvin tehtävät tulevat suoritetuksi eli mikä on käyttöprosessin vaikutus tuottavuuteen ja päätösten laatuun. Käyttöprosessia voidaan myös mitata käyttäjätyytyväisyydellä (Davis ym. 1980). Tietovarastoissa käyttöprosessi tarkoittaa tietovaraston varsinaista hyötykäyttöä eli esimerkiksi kyselyiden tekemistä ja raportointi- tai analysointiohjelmien käyttämistä tiedon hakemiseen ja käsittelyyn. Tämä voi olla verrattavissa kohdassa 2.2.4 esiteltyyn ulosvirtaan (outflow).

Käyttäjäympäristöstä on oleellista määrittää ja kuvata, ketkä ovat tietovaraston pääasiallisia käyttäjiä ja mitkä ovat heidän roolinsa tietovaraston käytössä (Geiger ym. 1997). Lisäksi on kuvattava kunkin käyttäjän käyttöoikeudet eli käyttäjäprofiili/turvallisuusluokka. Käyttöprosessista voidaan kuvata esimerkiksi käyttäjien tekemiä kyselyitä ja toimintoja, jotka vaaditaan kyselyiden suorittamiseksi (Wiener 2000).

Metatiedolla on ehkä suurin merkitys käyttöprosessille kaikista tietovarastoinnin osa-alueista (Inmon 1997). Ennen kuin tietovarastoa voidaan ylipäätään laisinkaan hyödyntää tulee käyttäjien tietää mitä tietoa tietovarastossa on ja mistä tuo tieto löytyy. Metatieto kuvaa nämä seikat (Inmon 1996; Jarke ym. 2000). Metatieto vastaa muun muassa seuraaviin käyttäjien kysymyksiin:

- Mitä tietoa tietovarastossa on?
- Missä tieto sijaitsee?

- Miten tietoon pääsee käsiksi?
- Mitä haettu tieto merkitsee?
- Mikä on tiedon alkuperä?
- Miten tietoa on summattua/koostettu?
- Milloin tieto on ladattu? (mm. Gardner 1998; Mohania ym. 1998)

Voidaan sanoa, että metatieto toimii karttana tai pohjapiirustuksena tietovarastolle (Gardner 1998; Geiger ym. 1997). Inmonin (1996) mukaan metatieto on ensimmäinen asia, jota loppukäyttäjä katsoo alkaessaan hyödyntämään tietovarastoa analysointiin tai raportointiin. Seuraavaksi esitellään, miksi käyttöprosessissa tarvitaan edellä listattuja metatietoja.

Operatiivisten järjestelmien käyttäjillä on usein totutut rutiinit käyttää järjestelmää (Geiger ym. 1997; Inmon 1997). Ei olekaan kovin vaikea ennustaa esimerkiksi mitä tuotetietojärjestelmä pitää sisällään. Tietovaraston käyttäjät sen sijaan haluavat porautua organisaation suurten tietomassojen ytimeen ja yhdistellä tietoa eri alueilta. Tällöin tarvitaan metatietoa kuvaamaan, mitä tietoa itse asiassa tietovarasto sisältää: mistä fyysisistä tietokannoista se muodostuu, mitä fakta-tauluja ja ulottuvuustauluja tietovarastossa on ja mitä aihealueita (esim. asiakas, tuote, tilaus) siellä esiintyy.

Tietovarastoon kohdistuu paljon ns. ad hoc –kyselyitä, kuten luvussa 2 tuotiin esille. Nämä kyselyt ovat usein tilapäisiä, ehkä vain kerran suoritettavia. Juuri ad hoc –kyselyt edellyttävät tietovaraston rakenteen sekä tietovaraston sisältämien tietojen tarkkaa tietämystä (Hovi 1997). Tämän lisäksi, kuten kohdassa 2.2.3 mainittiin, loppukäyttäjät eivät useinkaan osaa käyttää tietotekniikkaa kovin hyvin ja ovatkin yleensä johtotasoa ja liiketoiminta-orientoituneita. Metatieto ohjaa käyttäjää tiedon pariin ja opastaa mistä mikin tieto löytyy. Lisäksi se tarjoaa loppukäyttäjälle mahdollisuuden käyttää liiketoimintatermejä tiedon etsimisessä tai ainakin ilmaisee mikä on kunkin teknisen termin liiketoiminnallinen vastaavuus. Kimball (1998a) nimeää tietovaraston helppokäyttöisyyden kaikkein tärkeimmäksi vaatimukseksi

modernille tietovarastolle. Hän jatkaa ja toteaa, että loppukäyttäjät eivät yksinkertaisesti käytä mitään minkä käyttö on vaikeaa. Metatieto on tietovarastoissa helppokäyttöisyyden edellytys.

Yksi metatiedon tärkeimmistä rooleista on antaa tiedolla merkitys, kuten jo aiemmin esitellyssä määritelmässäkin sanottiin. Tällä tarkoitetaan sitä, että metatiedolla voidaan kertoa, mitä esimerkiksi tietty lukuarvo, tekninen nimi tai koodi tarkoittavat. Tämä auttaa käyttäjiä ymmärtämään heidän etsimiään tietoja. Etenkin tietojärjestelmien yhteydessä käytetään usein hyvin teknistä kieltä, jota liiketoiminnan ammattilaisten voi olla vaikea ymmärtää. Tämän vuoksi täytyy metatiedon avulla kertoa selkokielellä, mitä kukin termi tai luku tarkoittaa. Tämän kaltainen metatieto, joka on tarkoitettu etenkin loppukäyttäjille, on verrattavissa kohdan 4.1 liiketoiminnalliseen metatietoon.

Olennaista loppukäyttäjän kannalta on tietää tiedon alkuperä (data lineage) eli mistä tieto on kotoisin ja mitä prosesseja siihen liittyy (Hovi 1997; Variar 2002). Toisin sanoen kun käyttäjät tarkastelevat raporttia, tulee heidän tietää minkä tietolähteen mistäkin kentästä kukin raportin tieto on tullut (Kietz ym. 2001). Tämä on olennaisesta kahdesta syystä. Ensinnäkin tietovaraston tieto saattaa olla virheellistä. Tällöin tulee pystyä selvittämään missä virhe on sattunut. Onko virheen aiheuttanut summausprosessi vai transformointiprosessi siirrettäessä tietoa tietovarastoon? Vai oliko tieto virheellistä jo operatiivisissa lähteissä? Jäljittämällä tiedon alkuperä metatiedon avulla voidaan selvittää mistä virhe johtuu ja korjata se. Toiseksi käyttäjät joissain tapauksissa voivat haluta myös tutkia tietoa hyvin yksityiskohtaisella tasolla, mahdollisesti sellaisella tasolla, jota tietovarasto ei tarjoa. Tällöin olisi tarpeellista, että loppukäyttäjä tietäisi mistä operatiivisista lähteistä vastaava atominen ja reaaliaikainen tieto löytyy. Toisin sanoen niin ylläpitäjän (ks. kohta 4.1.4) kuin loppukäyttäjänkin tulee pystyä jäljittämään tiedon alkuperä. Metatiedolla on mahdollista (sekä tulisi) kuvata koko tiedon historia ja elinkaari tietovarastossa.

Loppukäyttäjän (ja mahdollisesti muidenkin) on tarpeen myös tietää, miten tietoa on summattu eli mitä summausalgoritmia on käytetty (Gray & Watson 1998). Näin tiedon analysoija pystyy paremmin ymmärtämään mistä tietyt luvut ovat tulleet. Inmonin (1996) mukaan summattu tai muuten koostettu tieto esiintyy aina sen prosessin kanssa, joka summauksen on luonut. Näin ollen metatiedolla tulee kuvata summauslogiikka. Loppukäyttäjälle suunnattu kuvaus summausprosessista, tai mistä tahansa muusta prosessista, ei tulisi olla tekninen vaan helposti ymmärrettävä. Lisäksi käyttöprosessi ja tiedon hyödyntäminen helpottuu, kun käyttäjä tietää millä tasolla tietoa on summattu. Toisin sanoen metatiedolla voidaan kuvata luvussa 3 esiteltyjä tietotyyppejä. Käyttäjän tulee tietää esimerkiksi onko jokin tieto atomista vai kevyesti tai tiiviisti summattua.

Loppukäyttäjän tulee myöskin tietää tietojen ajantasaisuuden aste. Toisin sanoen kuinka tuoretta tietoa tietovarastossa on ja mikä on tietojen lataustiheys (päivittäin, viikoittain, kerran kuukaudessa etc.) (Gardner 1998; Watson & Haley 1998). Näin loppukäyttäjä tietää, mitä kyselyitä on ylipäätään järkevä tehdä (Inmon 1996). Hovi (1997) puhuukin niin sanotusta aikaleimasta. Tämä tarkoittaa, että jokaisella tiedolla/tietueella tulisi olla aikaleima, joka kertoo minkä kuukauden, viikon tai päivän tiedosta on kyse. Ainakin Kimball (1998) luokittelee aikaleiman metatiedoksi.

Haley & Watson (1998) ovat huomioineet, että ilman kunnollista metatietoa käyttäjät pidättäytyvät käyttämästä tietovarastoa, tuhlaavat kohtuuttomasti aikaa sopivien kyselyiden kehittämiseen ja testaamiseen tai pyytävät jotakuta taitavampaa henkilöä kirjoittamaan heille haluamansa kyselyn. Huomio ei ole laisinkaan uusi. Jo 70-luvulla Lucas (1975, Lyytinen 1987, mukaan) toi esille puutteita tietojärjestelmien käyttöön liittyvissä prosesseissa. Käyttäjät eivät hänen mukaansa ymmärrä paljoakaan tiedosta, jota he saavat ulos tietojärjestelmästä. Lisäksi tietoon liittyvien epätarkkuuksien johdosta käyttäjät usein väheksyvät saatua tietoa. Vaikkakin tietovarastointi onkin suunniteltu

tarjoamaan käyttäjille yhtenäisempää ja laadukkaampaa tietoa ja korjaamaan edellä mainittuja ongelmia, ilman kattavaa metatietoa eivät nuo ongelmat katoa minnekään.

Yritysten tulisikin Hovin (1997) mukaan oppia luottamaan tietoihinsa. Tällöin lähestytään tiedon laatuun liittyviä käsitteitä, joihin juuri metatieto voi tuoda parannusta. Eikä pelkästään tiedon laatuun vaan myös päätösten laatuun. Sillä kuten Ballou ja Tayi (1999) mainitsevat, tiedon laatu vaikuttaa päätösten laatuun, joita tehdään tietovarastosta saadun informaation pohjalta. Metatiedon avulla loppukäyttäjä voi paremmin uskoa saamaansa kyselyyn, sillä saatu tieto on tällöin paremmin yhteydessä kontekstiinsa. Lisäksi jos käyttäjät voivat yhdistää tietovarastossa sijaitsevan tiedon niiden taustalla oleviin operatiivisiin tietoihin, heidän luottamus tietovarastoa kohti kasvaa (Variar 2002).

Metatiedon avulla loppukäyttäjät voidaan suojata teknologiselta infrastruktuurilta (Haley & Watson 1998). Toisin sanoen he hyödyntävät tietovarastoa käyttämällä liiketoimintatermejä ilman, että heidän on tarpeen tuntea tietovaraston teknologisia ratkaisuja.

4.2.4 Metatieto ja operaatioympäristö sekä –prosessit

Operaatioympäristö pitää Davis ym. (1980) mukaan sisällään tietojärjestelmän toiminnalle välttämättömät resurssit. Näitä resursseja ovat tarpeelliset ohjelmistot, laitteistot sekä tietokannat. Lisäksi operaatioympäristöön kuuluu tietojärjestelmän operaatioiden hallinta ja organisointi.

Operaatioprosessi tarkoittaa Davis ym. (1980) mukaan tietojärjestelmän fyysistä toimintaa. Operaatioprosessia voidaan mitata resurssien käytön tehokkuudella, toimivuudella tai laadulla. Lisäksi operaatioprosessia voidaan tarkastella sekä toissijaisten käyttäjien (i.e. ylläpitäjien) tyytyväisyydellä että sen mukaan kuinka se palvelee loppukäyttäjää (esim. vasteaikojen pituus, virheiden määrä,

saatavuus). Tämä tutkimus luokittelee tietovaraston ylläpidon vaatimat toimet tähän kategoriaan. Toisin sanoen ylläpitoprosessit ja ylläpitohenkilöstö pitävät yllä tietovaraston operaatioita sekä itse varsinaista ympäristöä (laitteistot ja ohjelmistot). Tutkimus myös luokittelee ETL-prosessin sekä summaus- ja koosteprosessit tähän luokkaan.

Operaatioympäristöstä sekä –prosessista tulee kuvata metatiedon avulla useita seikkoja. Nämä ovatkin verrattavissa kohdassa 4.1 esille tuotuun metatiedon jaotteluun kohteen mukaan. Operaatioympäristöstä tulee kuvata esimerkiksi mitä fyysisiä laitteita ja ohjelmistoja siihen kuuluu ja mistä tietokannoista tietovarasto muodostuu. Operaatioprosessista tulee kuvata kaikki ne prosessit, jotka toimivat tietovarastoympäristössä ja muokkaavat ja liikuttavat tietoa. Erityisen tärkeää on kuvata operatiivisten lähteiden ja kohteena toimivan tietovaraston välinen transformaatioprosessi.

Seuraavaksi tarkastellaan metatiedon merkitystä operaatioprosessille. Ensiksi tuodaan esille miksi tietovarastoympäristön komponentit, laitteistot ja ohjelmistot ja täten myös eri prosessit tarvitsevat metatietoa toiminnoissaan. Tämän jälkeen tarkastellaan metatietoa ylläpidon toiminnan näkökulmasta.

On väärin ajatella, että vain ihmiset käyttävät ja tuottavat metatietoa (Tannenbaum 2002). Kuten tämän tutkimuksen lähtökohtana onkin: kaikki tietovaraston komponentit ja tekijät käyttävät ja tuottavat metatietoa. Toisin sanoen pohdittaessa esimerkiksi vaatimuksia metatiedolle tulee huomioida myös useat ohjelmistot ja työkalut tietovarastoympäristössä.

Metatiedon merkitys useille tietovarastoympäristön ohjelmille on tärkeä. Joidenkin työkalujen ja ohjelmien tulee päästä käsiksi useisiin tietovarastoympäristön komponentteihin. Tämän vuoksi näiden ohjelmien tulee tietää kyseessä olevien komponenttien rakenne ja sisältö, jotka voidaan kuvata metatiedolla. (Müller ym. 1999) Kimballin (1996) mukaan myös esimerkiksi kyselytyökalut ja tiedon poimintavaiheen työkalut hyödyntävät yhä

suuremmissa määrin metatietoa. Työkalu voi esimerkiksi metatiedon avulla tehdä laatutarkistuksia ladatulle tiedolle tai havaita muuttuneita tietoja lähteissä. Metatieto voi toimia myös niin sanottuna kontrolli-informaationa joillekin työkaluille (Kietz ym. 2001). Tällöin metatieto tallennetaan erilleen ohjelmista. Ajonaikana ohjelmat lukevat tätä metatietoa, tulkitsevat sitä ja sitovat sen dynaamisesti toimintaansa. Tämän kaltainen metatieto on verrattavissa kohdan 4.1 tekniseen metatietoon. Hyötynä on etenkin se, että ohjelmien käyttäytymistä voidaan muuttaa päivittämällä vain kontrolli-informaatiota (eli metatietoa). Tämä tukee sekä uudelleenkäyttöä että joustavuutta. Näin ollen metatiedolla on olennainen merkitys tietovaraston komponenttien ja ohjelmistojen välisessä yhteistoiminnassa. Oikeastaan metatieto on ensisijainen keino saavuttaa yhteistoimintaa missä tahansa heterogeenisessä ympäristössä (Poole 2001).

Kuten kohdassa 2.2 tuotiin esille, tietovarastoympäristössä tietoa esiintyy eri medioilla. Inmon (1996) näkee, että eri tieto pitäisi sijoittaa eri medialle, riippuen tiedon määrästä sekä sen käytön todennäköisyydestä. Tämä tarkoittaa sitä, että jos tiettyä tietoa/tietokokonaisuutta ei käytetä pitkään aikaan, tulisi se siirtää halvemmalle medialle, josta sitä on kuitenkin vaikeampi hakea. Sen sijaan tieto, joka on kriittistä loppukäyttäjille ja jota hyödynnetään aktiivisesti tulisi sijoittaa medialle, johon on helppo päästä käsiksi. Näin ollen on erinomaisen tärkeää seurata, mitä tietoa käytetään ja kuinka paljon. Tämä huomio tuotiin esille jo luvussa 3, jossa luokiteltiin tämän kaltaisen monitorointitiedon olevan metatietoa. Ylläpitäjän tuleekin tietää esimerkiksi mitä tietoja kysellään paljon, onko jokin tieto käyttämätön, kasvavatko jotkin fakta-taulut liikaa tai milloin tietovarastoa käytetään eniten. Tehostamalla tiedon indeksointia, poistamalla turhat tiedot tai muuten organisoimalla tietovarastoa ylläpitäjä helpottaa käyttäjän työtä (Inmon 1997).

Tämän lisäksi on tärkeää, että tiedetään millä medialla on mitäkin tietoa. Kuten kohdassa 4.1.3 tuotiin loppukäyttäjän osalta esille; on tärkeää, että

loppukäyttäjä tietää mistä tieto on peräisin. Tällä tarkoitettiin sitä, mistä operatiivisista lähteistä tieto on tullut ja mistä mahdollisista pienemmistä tietokokonaisuuksista se koostuu. Tähän voidaan siis lisätä, että on tärkeää tietää myös mille medialle tieto on tallennettuna. Näin ollen loppukäyttäjä tai tietovaraston ylläpitäjä tietävät onko tieto helposti saatavissa ja mistä sitä tulisi hakea. On nopeampaa hakea tietoa relaatiotauluista lyhyellä SQL–kyselyllä kuin hakea sitä magneettinauhalta.

Mohania ym. (1998) tiivistävät, että tietovaraston ylläpitäjien tulee olla tietoisia tietorakenteiden muutoksista, tiedon liikkumisesta tietovaraston sisällä sekä tiedon transformointiprosessien muutoksista. Täten ylläpitäjille tulee tarjota helppo pääsy tämän kaltaiseen tekniseen metatietoon.

Kohdassa 2.2.1 esiteltiin prosesseja, joita tarvitaan jotta tieto saadaan lähteistä tietovarastoon. Metatiedolla on myös suuri merkitys näille prosesseille. Jarke ym. (2000) toteavatkin, että esimerkiksi tiedon poiminta ja yhtenäistäminen tietovarastoon suoritetaan yhä enimmäkseen intuitiivisesti; esimerkiksi integrointiprosessia ei ole dokumentoitu ja kuvattu tarpeeksi yksityiskohtaisesti. Näin ollen tiedon integrointia ja yhtenäistämistä on vaikea ymmärtää ja arvioida. Jarke ym. (2000) tähdentävätkin, että integrointiprosessiin liittyvän informaation määrittäminen ja tallentaminen on äärimmäisen tärkeää tukien korkealaatuista inkrementaalista integrointia. Ilman metatietoa operatiivisten järjestelmien sekä tietovaraston välillä olevan rajapinnan hallinta on hyvin vaikeaa (Inmon 1996). Jarke ym. (2000) näkevät metatiedon keskeisimmäksi ylläpitäjän työkaluksi tietovaraston hallinnoimisessa sekä kehittämisessä.

4.2.5 Metatieto ja kehitysympäristö sekä –prosessit

Kehitysympäristö koostuu tietojärjestelmän kehittämismenetelmistä ja -tekniikoista sekä suunnitteluhenkilöstöstä. Lisäksi kehitysympäristöön kuuluu

tietojärjestelmän kehittämisen sekä ylläpidon hallinnoiminen ja organisointi. (Davis ym. 1980)

Kehitysprosessilla tarkoitetaan organisaation resurssien hyödyntämistä tavoitteena valmis tietojärjestelmä. Kehitysprosessi voidaan siis käsittää perinteiseksi järjestelmänkehitykseksi (system development). Kehitysprosessia voidaan Davis ym. (1980) mukaan mitata tai tarkastella sen mukaan mikä vaikutus sillä on organisaatioympäristöön sekä toisaalta kuinka paljon organisaatioympäristö sitoutuu tietojärjestelmän kehittämiseen.

Kehitysympäristön tai –prosessin kuvaamiseen ei kirjallisuudessa tai aikaisemmissa tutkimuksissa juurikaan ole otettu kantaa ainakaan metatiedon näkökulmasta. Niiden kuvaaminen metatiedolla ei suoranaisesti vaikuta tietovaraston käyttöön. Toki on olennaista määrittää esimerkiksi vastuuhenkilöt tai dokumentoida kehittämismenetelmä. Ainakaan loppukäyttäjälle tällä ei kuitenkaan ole merkitystä.

Tietovaraston suunnittelussa ja kehittämisessä eli kehitysprosessissa metatieto on kuitenkin elintärkeää, vaikkakaan mitään yleisesti hyväksyttyä ratkaisua sen toteuttamiseen ei ole (Jarke ym. 2000). Tämäkään tutkimus ei pyri määrittämään ratkaisua tietovaraston kehittämiseen ja suunnitteluun metatiedon avulla, vaan keskittyy tässä kohdassa tuomaan esille miksi metatieto on olennaista suunnittelussa ja kehittämisessä. On olennaista huomata, että puhuttaessa tietovaraston suunnittelusta ja kehittämisestä, tarkoitetaan pääasiallisesti tietovarastoympäristön jatkuvaa kehittämistä, jotta se vastaisi paremmin liiketoiminnan tarpeita ja toimisi paremmin. Toisin sanoen varsinainen tietovarastoympäristö on jo rakennettu, mutta se kehittyy ja sitä muokataan jatkuvasti uusien vaatimusten noustessa esille. Tietovaraston suunnittelulla tarkoitetaan tässä kohdassa siten sekä tietovaraston alkuperäistä suunnittelua että sen koko elinkaarten kattavaa kehitystyötä.

Tietovaraston suunnittelijoiden ja kehittäjien tulee olla tietoisia tietolähteiden fyysisistä rakenteista, alkuperäisestä organisaation tietomallista sekä tietovaraston tietomallista, jotta tietovaraston tieto olisi tarkkaa sekä eheää (Mohania 1998). Inmonin (1996) mukaan esimerkiksi tietorakenteiden muutosten kuvaaminen ja seuranta on yksi metatiedon luonnollisista tehtävistä.

Inmon (1996) painottaa, että tietovaraston suunnittelu ja sen kehittämiselinkaari poikkeaa normaaleista tietojärjestelmistä. Etenkin palaute-iteraatio on Inmonin mukaan erityisen suuressa merkityksessä tietovaraston suunnittelussa ja sen koko elinkaaressa. Samaa teemaa painottavat mm. Jarke ym. (2000) sekä Hovin (1997) jo aiemmin esitetty huomio tietovaraston prosessi–luonteesta. Tietovarastoympäristö muuttuu jatkuvasti, sen lähteet muuttuvat ja käyttäjien tietotarpeet muuttuvat. Kimball (1998a) täsmentääkin, että tietovaraston suunnittelussa tulee valmistautua jatkuvaan muutokseen. Täten metatieto on ensiarvoisen tärkeää tämän muutosprosessin hallitsemisessa sekä dokumentoinnissa. Metatieto kuvaakin sekä tietovarastoa että sen kehittymistä (Jarke ym. 2000). Näin ollen metatieto voi toimia niin sanottuna organisaatiomuistina koskien tietovarastoa. Metatietoon voidaan sisällyttää tietoa tietovaraston muutoksista, versioinnista, tietomallien muutoksista jne.

Miller ym. (2001) ovat tutkineet konkreettisesti metatiedon käyttöä tietovaraston suunnitteluvälineiden tukena. He esittävät työkalun, joka käyttää metatietoa apunaan tietovaraston taulujen suunnittelussa ja muodostaa tarpeelliset SQL-lauseet metatiedon avulla. Käyttämällä hyvin määriteltyä ja rikasta metatietoa, suunnittelijoiden ei välttämättä tarvitse tuntea niin tarkkaan tietolähteiden rakenteita. Myös Campos, Freitas ja Laender (2002) esittävät konkreettisen työkalun tietovarastosovellusten rakentamiseen. Työkalu käyttää hyväkseen metatietovarastoa, joka auttaa keräämään tarpeellista metatietoa moniulotteisten mallien rakentamisessa sekä raporttien generoinnissa. White (1999) uskoo, että integroitu ja yhtenäinen metatieto luo tehokkaamman kehitysympäristön etenkin teknisestä kehityksestä vastaavalle henkilöstölle.

Kehitysympäristölle sekä –prosessille tarkoitettu metatieto on pääasiallisesti verrattavissa kohdan 4.1 tekniseen metatietoon. Vaikkakin kehitysprosessi on enemmänkin tekninen luonteeltaan, tarvitaan siinä kuitenkin myös liiketoiminnallista metatietoa. Myös teknisillä käyttäjillä voi olla tarvetta tarkastella metatietoa korkeammalla abstraktiotasolla ja liiketoiminnalliselta kannalta. Esimerkiksi tietovaraston kehittäjillä tulee olla käytössä organisaation käsitteellinen tietomalli suunnitellessaan tietovarastoa, pystyäkseen johtamaan oikeat tietovaatimukset.

4.3 Yhteenveto

Tässä luvussa esiteltiin eri tapoja luokitella tietovarastoympäristön metatietoja. Lisäksi näytettiin esimerkkejä tyypillisistä metatiedon ilmentymistä. Tämän jälkeen tuotiin esille, mikä merkitys metatiedolla on tietovaraston eri osa-alueilla. Luvussa tarkasteltiin tietovarastoa laajemman tietojärjestelmien tutkimisen mallin mukaan ja liitettiin metatieto täten laajempaan kontekstiin. Metatietoa tarkasteltiin kahdelta kannalta. Yhtäältä tarkasteltiin, mitä asioita metatiedolla tulisi kuvata eri ympäristöistä ja prosesseista. Toisaalta tuotiin esille, mikä merkitys metatiedolla on eri käyttöskenaarioissa. Toisin sanoen näytettiin, miksi metatieto on olennaista niin tietovaraston kehittämisessä (kehittämisprosessi), ylläpidossa ja päivittäisessä toiminnassa (operaatioprosessi) kuin varsinaisessa käytössäkin (käyttöprosessi).

Metatiedon määrittely ’tiedoksi tiedosta’ ei ole riittävä (Wiener 2000) kuten tässä luvussa on selkeästi tuotu esille. Metatieto on erittäin tärkeässä asemassa tietovarastossa. Osittain tästä johtuen metatiedoksi voidaan tietovarastokontekstissa käsittää hyvin erilaista tietoa: tietoa tiedosta, prosesseista, ohjelmista, laitteistosta, ihmisistä ja organisaatiosta. Metatiedoksi voidaan katsoa kuuluvan jopa SQL-lauseet tai tietyt ohjelmakoodit. Näin ollen metatieto käsitteenä laajenee pelkästä tiedon kuvaajasta.

Käsitteen metatieto laajuudesta johtuen on luonnollista, että metatietoa sijaitsee lähes kaikkialla tietovarastoympäristössä. Sitä voi olla niin metatietovarastossa, tiettyjen työkalujen omissa varastoissa, kuin piilotettuna koodiin, ohjelmiin, käyttäjämanuaaleihin tai paperidokumentteihin (Kietz ym. 2001).

Metatietoinfrastruktuuri, eli kokonaisvaltainen rakenne, arkkitehtuuri ja laitteistot metatiedon hallinnoimiseen, on monimutkainen ja vaatii useiden eri teknologioiden hyödyntämistä. Näistä syistä johtuen metatietoinfrastruktuurin hallinta ja rakentaminen on myös kallista. (Inmon 1997). Wiener (2000) toteaakin, että koska tietovarastointiin liittyy useita laitealustoja niin vaadittavan metatiedon määrä kasvaa eksponentiaalisesti eri laitealustojen ja järjestelmien lukumäärän kasvaessa.

Metatiedon hallintaan liittyykin useita ongelmia. Dittrichin & Vaduvan (2001) mukaan nykyään metatieto käsittää hyvin laajan skaalan erilaista informaatiota, kuten tämäkin tutkimus on osoittanut. Jopa ohjelmakoodin osia voidaan pitää metatietona. Metatietoa tuottavat useat eri tuotteet tietovarastoympäristössä ja täten metatieto sijaitsee eri paikoissa. Tietovarastojen suosion kasvu on johtanut tietovarastotuotteiden suureen määrään markkinoilla. Tästä johtuen tiedon ja prosessien hallinta on monimutkaistunut. Jotta tietovaraston käyttö, seuranta, analyysi sekä hallinta olisi mahdollista ja tehokasta, tulisi tietovarastoympäristön metatietojen hallinta olla yhtenäistä. Tällöin tietovarastoympäristössä sijaitseva metatieto tulee integroida eli sitä tulee pystyä jakamaan eri tuotteiden ja ohjelmien välillä.

Dittrich ja Vaduva (2001) huomauttavat, että metatiedon osittainen hallinta jonkin osa-alueen kohdalta tarjoaa vain pieniä hyötyjä. Sen sijaan metatietojen mahdollisimman kokonaisvaltainen määrittely, niiden integrointi keskenään ja yhtenäinen hallinta tuovat suurempia etuja: yhtenäisyys lisääntyy ja tuo esimerkiksi paremman mahdollisuuden jäljittää tiedon alkuperä sekä tehdä vaikutusseurantaa. Blackwood (2000) korostaa myös, että metatiedon

kerääminen tulisi aloittaa mahdollisimman ajoissa. Brayner ja Carneiro (2002) laajentavat suosituksia ja toteavat, että jo tietovarastoprojektin alussa tulisi määrittää strategia metatiedon keräämiselle, ylläpidolle sekä välittämiselle.

Metatietojen integroimiseen liittyy kuitenkin useita ongelmia. Nämä johtuvat muun muassa juuri laitteistojen ja ohjelmistojen paljoudesta ja heterogeenisyydestä. Tietovarastotyökaluilla on usein kullakin omat metatietostandardinsa ja määrityksensä. Näin ollen ne eivät toimi yhdessä (Bontempo & Zagelow 1998; Haley & Watson 1998; Marco 2002). Chang (2000) tuokin esille, että ei ole olemassa standardia tapaa jakaa metatietoa. Hän tuo myös esille joitakin kustannusongelmia; metatiedon luominen, jakaminen sekä hallinnoiminen on aikaa vievää, virhealtista sekä metatiedon integrointikulut ovat merkittäviä. Haley ja Watson (1998) ottavat erittäin skeptisen kannan metatiedon integrointiin ja toteavatkin, että metatiedon integrointi on lähestulkoon mahdotonta. Hyvin suuri ongelmanaihe tähän saakka on ollut yhteisen standardin puuttuminen metatiedon kuvaamiselle ja välittämiselle.

Perusvaatimus metatietotyökaluille ja -ratkaisuille onkin niiden yhteistoiminnallisuus metatietostandardien kanssa (Campos ym. 2002). Toisin sanoen, jotta metatietotyökalu olisi yhteistoiminnallinen tietovarastokomponenttien kanssa, tulisi sen tukea esimerkiksi OMG:n Common Warehouse Metamodel (CWM) –standardia (OMG 2001). Fletcher (2002) huomauttaa, että standardeihin tukeutuminen tuo lisäkustannuksia lyhyellä aikavälillä, mutta myöhemmin se maksaa itsensä takaisin helpomman metatiedon jakamisen, yhteistoiminnallisuuden ja laajennettavuuden muodossa.

Pelkkä metatietotyyppien ja –ilmentymien identifioiminen ei kuitenkaan riitä. Metatieto tulee pystyä myös kuvaamaan. Periaatteessa metatietokuvaukset voisivat olla paperille kirjoitettuna ja sijaita organisaation työntekijöiden mapeissa. Tällöin metatieto ei olisi kuitenkaan ohjelmien ja koneiden

ymmärtämässä muodossa ja esimerkiksi metatiedon jakaminen, uudelleenkäyttö ja tehokas hyödyntäminen ei olisi mahdollista. Seuraavassa luvussa käsitelläänkin metatiedon kuvaamista RDF:n avulla, joka mahdollistaa metatiedon kuvaamisen niin, että metatiedolla on sekä semantiikka, syntaksi että rakenne (Miller 1998).

5 RDF METATIEDON KUVAAMISESSA

Tässä luvussa käsitellään Resource Description Framework (RDF) –standardia. Luvussa esitellään mikä RDF on, RDF-tietomalli sekä RDF-syntaksi. Tämän jälkeen luvussa tarkastellaan, miten RDF:ää on sovellettu eri kohdealueilla. Lopuksi arvioidaan RDF:n soveltuvuutta tietovarastoympäristön metatietojen kuvaamisessa.

Resource Description Framework (RDF) on W3C:n suosittelema standardi metatiedon liittämiseen resursseihin. RDF voidaan nähdä infrastruktuurina, joka mahdollistaa rakenteisen metatiedon kuvaamisen, välittämisen sekä uudelleenkäytön (Miller 1998).

Lassila ja Swick (1999) näkevät, että RDF:ää voidaan hyödyntää useilla eri sovellusalueilla. Tämän vuoksi tässä tutkimuksessa ollaan kiinnostuttu sen soveltuvuudesta tietovarastoympäristöön.

RDF–mallin soveltuvuutta tietovarastoympäristöön arvioidaan ongelmalähtöisesti. Lähtökohtana ja oletuksena on, että RDF:n avulla tulisi pystyä kuvaamaan lähes mitä tahansa informaatioresurssia. Tämän tutkimuksen huomion kohteena ovat ne ongelmat, jotka saattavat nousta esille jos RDF:ää käytettäisiin tietovarastometatiedon kuvaamisessa. Toisin sanoen tutkimuksessa tarkastellaan ja analysoidaan aikaisempia havaintoja ja yrityksiä soveltaa RDF:ää eri sovellusalueille. Tämän jälkeen analysoidaan ovatko aiemmin esille tulleet ongelmat relevantteja tietovarastoympäristössä ja pyritään löytämään muita hankaluuksia, joita RDF:n soveltamisessa saatettaisiin kohdata. Tavoitteena on tutkia kuinka helppoa ja järkevää sen käyttöönotto voisi olla tietovarastoympäristössä.

Kohdassa 5.1 esitellään lyhyesti RDF ja tarkastellaan sen tietomallia ja syntaksia eli mistä osasista RDF muodostuu. Tämän jälkeen kohdassa 5.2 esitellään aikaisempia tutkimuksia, joissa RDF:ää on hyödynnetty. Kohdassa 5.3

arvioidaan tarkemmin sen sopivuutta tietovarastoympäristön metatiedon kuvaamisessa.

5.1 RDF:n yleiskuva ja tavoitteet

RDF:n pääasiallisena tarkoituksena on kuvata www-resursseja (Lassila & Swick 1999). RDF on malli, joka tarjoaa perustan liittää metatietoa esimerkiksi www-sivuun. Bebee ym. (1999) näkevät RDF:n XML-sovelluksena, joka sisältää vaadittavat rakenteelliset rajoitteet mahdollistaakseen yksiselitteiset menetelmät kuvata semantiikkaa. Haroldin (2000) mukaan RDF:n tavoitteena on määritellä yleinen käytäntö, miten metatiedon semantiikka, syntaksi ja rakenne muotoillaan eri toimialueilla.

RDF on W3C:n suositus metatiedon standardille esittämiselle. Beckett (2001) näkee RDF:n yleisenä kuvausteknologiana, jota voidaan soveltaa useilla alueilla. Taustavaikuttajina RDF:n kehittämisessä ovat olleet www-standardointiyhteisö, kirjastoyhteisö ja tietämyksen esittämisyhteisö (knowledge representation). Myöskin olio-ohjelmointi sekä tietokantateknologia ovat vaikuttaneet RDF:n kehitykseen (Lassila & Swick 1999).

RDF tarjoaa välineet julkaista metatietosanastoja, joita ihmiset voivat tulkita (human-readable), mutta jotka ovat myös niin formaaleja, että koneet pystyvät prosessoimaan niitä (machine-processable). Täten se tukee metatiedon uudelleenkäyttöä ja laajennettavuutta eri käyttäjäkuntien parissa. (Bebee ym. 1999).

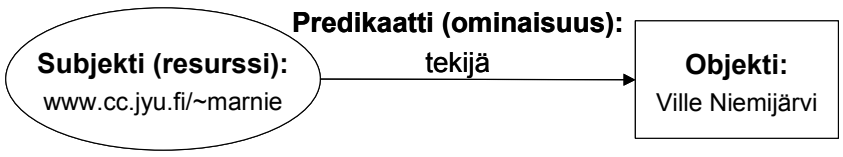
RDF muodostuu kahdesta spesifikaatiosta: RDF-spesifikaatiosta (Lassila & Swick 1999) sekä RDF Schema -spesifikaatiosta (Brickley & Guha 2002). Tässä tutkimuksessa ei käydä yksityiskohtaisesti läpi kumpaakaan spesifikaatiota, vaan esitellään RDF:n perusteet lyhyesti.

5.1.1 RDF-tietomalli

RDF:n perustana on RDF-tietomalli. Tietomalli ei ole sidonnainen mihinkään nimenomaiseen syntaksiin ja sen tarkoituksena on esittää RDF-ilmauksia (expressions). RDF-tietomalli koostuu kolmesta osasta: resursseista, ominaisuuksista sekä lauseista (Lassila & Swick 1999).

Resurssiksi (resource) voidaan käsittää mikä tahansa kohde, joka on identifioitavissa URI:lla. Kaikki asiat, joita RDF-ilmaukset kuvaavat, ovat resursseja. Periaatteessa kaikki kuviteltavissa olevat entiteetit voivat olla resursseja. *Ominaisuudet (property)* kuvaavat resursseja. Ominaisuus on resurssin tietty piirre tai attribuutti. *Lauseet (statements)* tarkoittavat tietyn resurssin, sitä kuvaavan ominaisuuden sekä tuon ominaisuuden arvon yhdistelmää. Näitä kolmea lauseen osaa nimitetään subjektiksi (subject), predikaatiksi (predicate) ja objektiksi (object). RDF:n ideana on siis kuvata resursseja. Tämän se tekee määrittämällä joukon lauseita (statement), jotka toteavat ominaisuuksia resurssista. Seuraava esimerkki valaisee käsitteitä.

Oletetaan, että halutaan kuvata tietyn www-sivun ominaisuuksia. Olkoon www-sivu tässä tapauksessa *www.cc.jyu.fi/~marnie*. Jos haluttaisiin kertoa, että kyseisen sivun tekijä on henkilö nimeltä Ville Niemijärvi, niin voisimme ilmaista sen seuraavalla lauseella: *"Ville Niemijärvi on resurssin www.cc.jyu.fi/~marnie tekijä"*. Tässä esimerkissä www-sivun osoite viittaa resurssiin, jota halutaan kuvata. Predikaattina eli ominaisuutena toimii termi *tekijä* ja nimi *Ville Niemijärvi* on objekti eli tuon ominaisuuden arvo. Lauseet voidaan esittää myös graafisesti. Kuvassa 9 on esitetty edellä mainittu



Kuva 9: Yksinkertainen RDF lause graafisesti esitettynä

esimerkki graafisesti.

Kaikki lauseen osat (lukuun ottamatta merkkijonoja) voidaan tunnistaa URI:n (Uniform Resource Identifier) avulla. Ehkä yleisin esimerkki URI:sta on URL. Täten lauseita voidaan kirjoittaa mistä tahansa resurssista, jolla on URI. Tämä taas voidaan periaatteessa määrittää mille vain. Täten resurssiksi voidaan käsittää esimerkiksi www-sivut, sivujen osat, tietokannat, tiedostot tai ihmiset. Toisin sanoen resurssin ei tarvitse olla yhteydessä internetiin, se täytyy vain identifioida.

5.1.2 RDF-syntaksi

Jotta RDF:ää voidaan prosessoida ohjelmilla ja jakaa ohjelmien välillä, tarvitsee se syntaksin (Miller 1998). RDF-tietomalli tarjoaakin Lassilan & Swickin (1999) mukaan abstraktin ja käsitteellisen viitekehyksen määrittää ja käyttää metatietoa. Konkreetti syntaksi sen sijaan tarvitaan metatiedon luomiseen ja jakamiseen. XML on syntaksi, jolla voidaan esittää RDF-kuvauksia. Se mahdollistaa RDF:llä määritellyn metatiedon prosessoinnin sekä jakamisen sovellusten välillä. On huomioitava, että XML on vain yksi mahdollinen syntaksi esittää RDF:ää (Lassila & Swick 1999). Myös muita syntakseja voidaan käyttää.

Kuva 9 voidaan esittää RDF/XML-syntaksilla seuraavasti.

```
<rdf:RDF>
  <rdf:Description about=”www.cc.jyu.fi/~marnie”>
    <s:Creator>Ville Niemijärvi</s:Creator>
  </rdf:Description>
</rdf:RDF>
```

Käsitteiden ja lauseiden merkitys kuvataan RDF-kaavalla (RDF schema). Kaava voidaan ajatella olevan eräänlainen sanakirja, joka ilmaisee mitä mikin termi RDF-lauseissa merkitsee ja antaa spesifin merkityksen termeille (Brickley & Guha 2000). Lisäksi kaava ilmaisee määrityksiä ja rajoitteita ominaisuuksien

käytölle. RDF-kaava antaa kehittäjien määritellä haluamansa sanaston RDF-tiedolle.

5.2 RDF metatiedon kuvaamisessa – käytännön esimerkkejä

RDF:n soveltaminen ja tutkiminen on keskittynyt pääasiallisesti sen syntysijoille eli www-ympäristöihin. RDF:n soveltamista ovat tutkineet ainakin Bebee, Mack ja Shafi (1999), Houben ja Vdovjak (2001) sekä Fillies, Weichardt ja Wood-Albrecht (2002). Seuraavaksi lyhyt katsaus näihin tutkimuksiin ja muutamia huomioita niistä.

Bebee, Mack ja Shafi (1999) tutkivat RDF:n käyttöä metatiedon kuvaamisessa hajautetussa komponenttipohjaisessa järjestelmässä. He tutkivat RDF:ää kahdelta kannalta: a) helppokäyttöisyyttä ja hallittavuutta sekä b) RDF-metatiedon käytettävyyttä hajautettujen ohjelmistokomponenttien yhteydessä.

Tutkijat kuvasivat aikaisemmin metatiedot C++ -olioina, joihin pääsi käsiksi CORBA-rajapinnan kautta. Tämä osoittautui hankalaksi, koska päästäkseen metatietoon käsiksi, vastapuolen tuli implementoida CORBA-rajapinta. Näin ollen uusien sovellusten integrointi järjestelmään osoittautui vaikeaksi. (Bebee ym. 1999) Päästäkseen tästä rajoitteesta tutkijat testasivat RDF:n käyttöä metatiedon kuvaamiselle. Tutkijat kehittivät selain-pohjaisen järjestelmän, joka mahdollisti RDF-metatiedon näyttämisen selaimessa, mutta myös metatiedon luomisen ja muokkaamisen selaimen kautta.

Bebee ym. (1999) toivat tutkimuksissaan esille, että RDF suoriutui ko. kohdealueella hyvin. RDF sisälsi tarvittavat ominaisuudet kuvata metatietoa, joka aikaisemmin oli kuvattu C++ -olioilla. Lisäksi RDF mahdollisti eri metatietokohteiden linkittämisen viittaamalla. Suurimpana ongelmana RDF:n hyödyntämisessä oli Netscape 4.X -selaimen tuen puute XML/RDF-prosessoinnille.

Houben ja Vdovjak (2001) kuvaavat niin sanottua integroivaa arkkitehtuuria, joka pohjautuu semanttiseen integrointiin sekä pyynnöstä suoritettavaan tiedon hakemiseen (ks. kohta 2.1). Arkkitehtuurin perustana on RDF-malli. Arkkitehtuuria on tarkoitus hyödyntää tiedon hakuun heterogeenisistä lähteistä.

Tutkijat kehittivät käsitteellisen mallin, jonka he kuvasivat RDF:llä. Tämä malli toimii arkkitehtuurin taustalla ja kuvaa kohdealueen hierarkkiset käsitteet. Malli on rinnastettavissa ontologiaan. Ontologia voidaan määritellä tietyn kohdealueen formaaliksi käsitteellistämiseksi. Toisin sanoen ontologiat tarjoavat yhteisen ymmärryksen kohdealueesta ja sen käsitteistä, joista voidaan viestiä sekä ihmisten, että sovellusten välillä (Broekstra ym. 2000). Ontologia sisältää yleensä käsitteiden hierarkian tietyltä kohdealueelta. Lisäksi se kuvaa kunkin käsitteen ominaisuudet attribuutti-arvo –mekanismin avulla. RDF:n avulla voidaan kuvata ontologioita tai sitten RDF:ää joudutaan laajentamaan korkeamman tason ontologiakielellä (Houben & Vdovjak 2001).

Houbenin ja Vdovjakin (2001) esittämää mallia ei voi kuitenkaan sellaisenaan soveltaa tietovarastoympäristöön. Tietovarastot soveltavat ns. aktiivista (eager) tiedon hakemista ja integrointia eikä pyynnöstä suoritettavaa kuten tässä tapauksessa.

Fillies, Weichardt ja Wood-Albrecht (2002) esittävät SemTalkin, jonka ideana on tarjota loppukäyttäjille mahdollisuus tuottaa helppokäyttöisen MS Visio –ohjelman avulla RDF-kaavoja (schemas). Näin ollen käyttäjät voivat helposti luoda ontologioita sekä prosesseja, jotka pohjautuvat RDF:ään. Tutkijat tuovat esille, että SemTalkin taustalla olevaa standardia RDF:ää piti laajentaa ottamalla mukaan sellaisia olio-maailman ominaisuuksia kuten metodit sekä tilat. Chang (1998) toteaakin, että RDF-kaava ei eksplisiittisesti tue käsitystä metodeista tai operaatioista.

Fillies, Weichardt ja Wood-Albrecht (2002) toteavat myös, että SemTalkin avulla RDF-mallit voidaan tallentaa yksittäisinä html-sivuina ja julkaista selaimen kautta. Lisäksi SemTalkin etuna nähdään sen mahdollisuus kuvata myös liiketoimintaprosesseja. Suurin ongelma tutkijoiden esittämissä projekteissa oli saada käyttäjät omaksumaan oliosuuntautunut ajattelu prosessimallinnuksessa. Edelleen tutkijat näkevät, että käyttämällä yhtenäisiä ja yhdistettyjä XML-pohjaisia sanastoja organisaatioilla on uudenlaiset mahdollisuudet jakaa metatietoa sovellusten välillä. Tämä mahdollistaa muu muassa sisällön hallinnan, dokumenttien hallinnan sekä tietovarastoratkaisujen merkityksekkään integroinnin.

5.3 RDF:n arviointi

Tässä kohdassa arvioidaan lyhyesti RDF:n sopivuutta metatiedon kuvaamisessa. Ensiksi käsitellään RDF:n vahvuuksia ja etuja. Tämän jälkeen tuodaan esille mahdollisia rajoitteita.

Vaduva ja Vetterli (2001) ovat huomioineet, että metatiedolla on yleisesti heterogeenisempi rakenne kuin ns. tavallisella tiedolla: metatieto kuvaa tietoa sekä järjestelmiä eri abstraktio-tasoilla ja eri käyttäjille. Tämä asettaa erityisiä joustavuusvaatimuksia sekä metatiedon esittämiselle ja välittämiskielelle.

Candanin ym. (2001) mukaan RDF tarjoaa tietomallin, jossa voidaan kuvata kohteita (objects) ja kohteiden välisiä suhteita. Mikä tekee RDF:stä erikoisen verrattuna esimerkiksi olio-pohjaisiin tietomalleihin tai XML:ään on se, että suhteita voidaan pitää eräänlaisina kohteina. Toisin sanoen suhteita kahden kohteen välillä voidaan kuvata mielivaltaisesti ja tallentaa erillään näistä kahdesta kohteesta. Tämä sopii Candanin ym. (2001) mielestä hyvin dynaamisesti muuttuviin, hajautettuihin www-ympäristöihin.

Kuten aiemmissa luvuissa on määritelty, tietovarastoympäristö sisältää lukuisia ohjelmistoja sekä laitealustoja. Metatietoa tuleekin pystyä jakamaan sekä eri

laitealustojen että eri työkalujen välillä (Inmon 1997). RDF:n käyttämä alustariippumaton XML-syntaksi soveltuu tähän tarkoitukseen mainiosti.

XML-syntaksin hyödyntäminen onkin yksi RDF:n vahvuuksista. Vaikkakin XML on saanut kohdalleen suuren määrän epärealistisia odotuksia alentaa se ainakin Rosenthalin ja Seligmanin (2001) mukaan kynnystä jakaa tietoa erilaisten ohjelmien ja tietokantojen välillä. XML helpottaa tiedon jakamista tarjoamalla yleisen formaatin ilmaista tietorakenteita sekä sisältöä.

XML-työkalut edistävät organisaatioiden kykyä hallita ja jakaa myös puolirakenteistettua (semistructured) tietoa, sisältäen erityisesti www:n sisältämää tietoa, jota on vaikea kuvata ennalta määrätyn mallin mukaan (Rosenthal & Seligman 2001). Täten voidaan olettaa, että myös RDF, joka hyödyntää XML:n syntaksia, mahdollistaa puolirakenteistetun tiedon kuvaamisen paremmin.

Poolen (2001) mukaan XML:n etuna on se, että XML-merkatussa tiedossa yhdistyy metatieto sekä XML-entiteettien kuvaama tietosisältö. Toisin sanoen tietosisällön viestintä eri laitteistojen sekä ohjelmistojen välillä on itseään kuvaavaa (Harold 2000; Poole 2001).

Jarken ym. (2000) mukaan metatietovaraston metamallin tulee olla ilmaisuvoimainen. Broekstra ym. (2000) argumentoivat RDF:n ilmaisuvoiman puolesta. He toteavat, että koska RDF sisältää objekti-attribuutti –rakenteen, se on tarpeeksi ilmaisuvoimainen kuvaamaan mitä tahansa tietoa.

Bebee ym. (1999) toivat tutkimuksessaan esille muutamia RDF:n etuja. Käyttämällä tekstipohjaista XML-syntaksia, RDF-metatietoa voidaan jakaa useiden protokollien kautta (kuten HTTP, FTP, SMTP, IIOP, JAVA RMI) ja sitä voidaan prosessoida yleisesti saatavilla XML-työkaluilla (Bebee ym. 1999). Täten mikä tahansa komponentti, jolla on pääsy näihin protokolliin, voi hakea RDF:ää. XML-ohjelmien kasvava määrä helpottaa ohjelmia sekä komponentteja

operoimaan RDF-metatietoa. RDF:ssä on myös sisäänrakennettu tuki hajautettuihin resursseihin viittaamiselle. RDF-spesifikaatio määrittää siis resurssin miksikä tahansa, johon voidaan viitata URI:lla. Lisäksi metatietoa voidaan linkittää URI:en avulla toisiinsa.

RDF:n heikkoutena on sen uutuus. Toisin sanoen se on standardina uusi ja täten vähän testattu käytännössä. Lisäksi kaupallisia sovelluksia ja konkreettisia työkaluja ei ole kovinkaan paljon saatavilla (Fillies ym. 2002). Voidaan myös olettaa, että sovelluskehittäjillä ei välttämättä ole kokemusta RDF:stä. Täten jo perinteisiin teknologioihin tottuneet käyttäjät voivat vierastaa sitä.

Kimball (1998b) tuo esille, että useiden metatietojen tulee olla lähellä niitä ohjelmia, joiden toimintaan ne vaikuttavat. Toisin sanoen esimerkiksi ohjelmien, asetusten ja spesifikaatioiden, jotka vaikuttavat tai ohjaavat prosesseja tulee olla tietyissä paikoissa tallennettuna ja hyvin spesifissä formaateissa. Kimballin mukaan tämä tilanne ei tule muuttumaan lähiaikoina. Kaikkea metatietoa ei siis voida kuvata yhdellä ainoalla kielellä tai yhdellä formaatilla.

Candanin ym. (2001) mukaan RDF:n laaja-alainen hyväksyntä riippuu sekä RDF:n tarjoamasta ilmaisuvoimasta kuvata metatietoa sekä saatavilla olevista työkaluista, jotka tekisivät RDF:stä helppokäyttöisen viitekehyksen kuvata metatietoa. Tällaisia työkaluja ovat sellaiset, jotka luovat ja tallentavat metatietoa RDF-muodossa sekä sellaiset, jotka tarjoavat graafisen käyttöliittymän RDF-metamallien muokkaamiseen. Kohdan 5.2 esimerkeissä tuotiin esille tutkimuksia, joissa näytettiin, että RDF Schema -kielen ilmaisuvoima riittää kuvaamaan ontologioita. Lisäksi esiteltiin lyhyesti SemTalk, joka mahdollistaa helppokäyttöisen RDF-kaavojen ja mallien luomisen. Candan ym. (2001) kuitenkin huomauttavat, että esimerkiksi automaattisia metatiedon poimintatyökaluja on vain kourallinen. Tämän lisäksi

useimmat näistä ovat rajoittuneet hyödyntämään hyvin tarkkaan määriteltyä kaavaa. Tämä hankaloittaa resurssien kuvaamisprosessin automatisointia.

Tietovarastoympäristössä metatietoa esiintyy kaikkialla. Metatieto on lisäksi hyvin monimuotoista ja sitä esiintyy kaikkialla tietovarastoympäristössä. Metatieto on myös erittäin tärkeää kaikkialla tietovarastoympäristössä ja kriittisin menestystekijä tietovarastoprojekteissa, kuten tämä tutkimus on tuonut esille. Metatietoa tuottavat myös hyvin paljon mm. CASE-työkalut, ETL-komponentit ja monet muut ohjelmistot tietovarastoympäristössä. Tästä seuraa vääjäämättömästi, että kaikkea metatietoa ei voida kuvata millään yhdellä syntaksilla. Voidaan myös olettaa, että esimerkiksi ohjelmien ja komponenttien tuottaman metatiedon muokkaaminen RDF-metamallin mukaiseksi XML-syntaksiin toisi lisäkustannuksia sekä aiheuttaisi kenties liikaa vaivaa.

RDF:llä on kiistatta ansionsa www-resurssien kuvaamisessa sekä semanttisen webin kehittämisessä (Broekstra ym. 2000). Harold (2000) uskoo, että vaikka RDF:ää voidaan hyödyntää useilla sovellusalueilla, niin ainakin aluksi sen pääpaino on metatiedon liittämisessä www-sivuille. Tulevaisuus näyttää kuinka laajalle ja mille sovellusalueille RDF leviää. RDF:n kehityksen painopisteen ollessa www-yhteisön parissa, voidaan olettaa, että kestää jonkin aikaa ennen kuin RDF-sovellukset kehittyvät ja laajentuvat muille sovellusalueilla.

Kerschberg (2001) uskoo, että perinteiset tietovarastoympäristöt kehittyvät tietämyksen hallintaympäristöiksi, joissa tulee ottaa paremmin huomioon rakenteistamaton tai puolirakenteistettu heterogeeninen tieto. Tällaisissa ympäristöissä RDF:n kaltainen metamalli ja infrastruktuuri saattaa olla erittäin käytännöllinen.

Ulkoisen tiedon yleistyessä tietovarastojen tietolähteenä tulee RDF:n rooli mahdollisesti suuremmaksi. Kun www-resurssien kuvaaminen RDF:n avulla yleistyy ja automatisoituu, on organisaatioidenkin helpompi hyödyntää

www:tä todellisena resurssina. Hackathorn (1999, 165) näkeekin selvän tarpeen standardille tavalle kuvata ja välittää metatietoa etenkin www:stä kerättävän informaation osalta. RDF:n puolesta puhujana voidaan nähdä selainkäyttöliittymien yleistyminen tiedon analysoinnissa. Kimball (1998a) nostaa esille tarpeen julkaista tietovaraston tietoja usealla eri medialla, mieluiten internetin välityksellä. Tällöin voidaan olettaa, että myös tietovaraston metatietoon tulisi päästä käsiksi helppokäyttöisen selainkäyttöliittymän kautta tai mahdollisesti muiden medioiden avulla. Kimball (2001) uskookin, että XML-teknologioiden laaja-alainen hyödyntäminen niin sanotusti irrottaa tietovaraston loppukäyttäjät työasemiltaan. Ainakin Bebeen ym. (1999) tutkimusten tulokset tukevat RDF:n käyttöä (yhdisteltynä XML-tyylisivuihin sekä mahdollisesti javascriptiin) metatiedon esittämisessä selainkäyttöliittymässä. RDF voisikin oletettavasti soveltua liiketoiminnallisen metatiedon kuvaamiseen. Tämä metatieto on loppukäyttäjälle tarkoitettu ja sen tulee olla semanttisesti rikasta, toisin sanoen helposti ymmärrettävää loppukäyttäjälle. Lisäksi käyttämällä sopivia XSL-tyylisivuja XML-koodatut metatietokuvaukset voitaisiin esittää loppukäyttäjälle esimerkiksi selaimella, mutta myös muilla alustoilla.

Candan ym. (2001) uskovat, että RDF:llä on valoisa tulevaisuus etenkin www:n puolella, mutta myös esimerkiksi tiedon louhinta (data mining) voisi hyötyä suuresti RDF:stä.

Vaikka RDF-sovelluksia kehitettäisiinkin tietovarastointia varten, niin silti metatiedon hallinnoiminen ei välttämättä helpotu. Vaikkakin tuote pystyisi jakamaan ja esittämään metatietoa XML-muodossa, niin usein tällä tuotteella on oma spesifi kaavansa (schema). Toisin sanoen ei riitä, että tuote hyödyntää XML-syntaksia. Pystyäkseen jakamaan metatietoa, tuotteiden tulee jakaa myös metatietokaavansa. Jotta tämä olisi mahdollista, tuotteiden tulee mukautua yhteiseen malliin metatiedosta (Moriarty 2000). Mikäli tuotteiden- ja

ohjelmistojen toimittajat eivät omaksu yhtä standardia metatiedon esittämiseen metamallin muodossa, eivät ongelmat häviä minnekään.

5.4 Yhteenveto

Tässä luvussa esiteltiin RDF-malli ja arvioitiin sen soveltuvuutta tietovarastometatiedon kuvaamisessa. RDF:n avulla voidaan liittää metatietoa resursseihin. Resurssi voi olla mikä tahansa asia, jolla on URI. RDF:n katsotaan myös soveltuvan useille sovellusalueille. Toisin sanoen sillä voidaan kuvata mitä tahansa resurssia: www-sivuja, tiedostoja, ihmisiä, palvelimia, ohjelmia jne.

Luvussa tuotiin esille, että ainakin periaatteessa RDF:llä on mahdollista kuvata mitä tahansa kohdealuetta. RDF Scheman ilmaisuvoiman metatiedon ja metamallien (tai ontologioiden) kuvaamisessa pitäisi olla riittävä.

RDF:n soveltamista on tutkittu hieman, mutta enimmäkseen www-ympäristössä. Huomio ei oletettavasti ole kiinnittynyt tässä vaiheessa juurikaan laajemmille sovellusalueille, kuten tietovarastoympäristöön. On huomioitava, että RDF on teknologiana uusi ja se kehittyy jatkuvasti (Candan ym. 2001).

Kimball (2001) on skeptinen etenkin W3C:n kehittämien lukuisien standardien suhteen, ainakin mitä tulee tietovarastoympäristöön. Hän kehottaa odottamaan, että organisaatiolle tutut ohjelmistotuottajat ottavat ensiksi esimerkiksi XML:n tarjoamat hyödyt käyttöön. Tämän jälkeen, mahdollisesti hyväksi todettuna, voisi kääntyä katsomaan, miten uudet standardit voisivat tukea organisaation tietovarastoympäristöä tai sen kehittämistä.

6 JOHTOPÄÄTÖKSET

Metatieto on tietovarastojen elinehto. Syyt tähän johtuvat sekä itse tietovarastoympäristöstä että sen käyttäjistä.

Ensinnäkin tietovarastoympäristö on hyvin monimutkainen kokonaisuus ja sen sisältämät teknologiat, komponentit, ohjelmistot sekä tiedot ovat hyvin heterogeenisiä, kuten luvuissa 2 ja 3 on tuotu esille. Jotta yhteistoiminta, hallinta, ylläpito ja kehitystyö olisi mahdollista, tulee tämä monimutkainen ympäristö kuvata metatiedon avulla. Lisäksi tietovarastoympäristö on dynaaminen ja muuttuva kokonaisuus. Se sisältää useita prosesseja, joissa tieto liikkuu ja muokkaantuu ajan kuluessa. Täten on tärkeää kuvata tietovaraston prosesseja mutta myös itse tietovarastoa, koska se on itsessäänkin eräänlainen prosessi.

Tietovarastoa käyttävät erilaiset käyttäjäryhmät. Suurin syy, miksi metatietoa tarvitaan käyttöprosessissa on se, että varsinaiset loppukäyttäjät eivät useimmiten ole tietotekniikan ammattilaisia. He tuntevat liiketoiminnan ja käyttävät tietovarastoa etenkin strategisen päätöksenteon tukena. He eivät sen sijaan tunne, eikä heidän tarvitsekaan tuntea, teknologisia ratkaisuja tai teknisiä termejä, jotka ovat tietovarastoympäristön taustalla. Metatieto auttaa heitä hyödyntämään ja käyttämään tehokkaasti tietovarastoa ja lisäksi ymmärtämään mitä haettu tieto tarkoittaa.

Toisaalta tietovarastoa käytetään hyvin erilailla verrattuna perinteisempiin tietojärjestelmiin. Tietovarasto yhdistää useita tietolähteitä, tietoa summataan ja koostetaan hyvin eri tavalla ja käyttäjillä ei usein ole totuttuja rutiineita käyttää järjestelmää. Usein tietovarastoon kohdistuvat kyselyt ovatkin kertaluontoisia, joiden tarkoitus on porautua ja yhdistellä tietoa hyvin eri alueilta. Ilman metatietoa tietovaraston tehokas hyödyntäminen ja oikean tiedon löytäminen olisi lähes mahdotonta. Toisaalta tärkeä seikka on tiedon luotettavuus.

Loppukäyttäjien tulee tietää mistä operatiivisista tai ulkoisista lähteistä tieto on peräisin, miten sitä on prosessoitu ja mikä on sen summaustaso. Metatiedolla tuleekin kuvata tiedon koko elinkaari. Loppukäyttäjille tuleekin tarjota helppo pääsy niin sanottuun liiketoiminnalliseen metatietoon.

Metatieto kuvaa tietovarastossa sekä tietoa, prosesseja, käyttäjiä, ohjelmistoja, laitteistoja kuin myös organisaation liiketoiminnallisia näkökohtia. Lisäksi metatieto kuvaa itse tietovaraston kehittymistä ja toimintaa: tietojen ja taulujen kasvunopeuksia tai tietoihin kohdistuvia kyselyitä ja niiden määriä. Metatieto sisältää myös kuvaukset käyttöoikeuksista kullekin käyttäjälle, mutta ottaa huomioon myös ulkoisen ympäristön ja asettaa näin organisaation tiedot oikeaan historialliseen kontekstiin.

Tietovarastointi on jatkuva prosessi (mm. Hovi 1997, s. 106; Jarke ym. 2000); niin organisaation ulkoinen ympäristö kuin sisäiset tietotarpeet muuttuvat ajan myötä ja näin ollen myös tietolähteet ja itse tietovarasto muuttuu jatkuvasti. Onkin huomioitava, että myös metatietovarasto ja metatiedon kerääminen, integrointi sekä hallinta on myös jatkuva prosessi (Marco 2001). Metatiedon tulee siis muuttua tietovaraston muuttuessa. Lisäksi muutokset metatietoon tulee pitää ajan tasalla ja dokumentoituna.

Olennainen tästä tutkimuksesta esiin tuleva johtopäätös on se, että tietovarastoinnista puhuttaessa tulee huomioida niin staattinen tiedon säilytyspaikka (tietovarasto), prosessit, aktiviteetit ja ohjelmistot, jotka käsittelevät tietoa (tietovarastointi) kuin suunnittelumenetelmät ja ylläpidon vaatimat tehtävät. Täten voidaankin puhua enemmänkin tietovarastoympäristöstä. Yhdistämällä edellä mainitut osa-alueet ympäristö-käsitteen alle, voidaan tarkastella esimerkiksi metatietoa laajemmassa mittakaavassa, kuten tehtiin kohdassa 4.2.

Metatiedon kuvaaminen tietovarastossa ei ole triviaali tehtävä; organisaatiossa tulee pohtia tarkkaan, millä tasolla sille metatiedon kerääminen ja

tallentaminen on hyödyllistä (Gilliland-Swetland 1998). Kaiken tiedon kuvaaminen metatiedon avulla on erittäin raskasta sekä kallista (Geiger ym. 1997). Fletcher (2002) neuvoo kuvaamaan sekä teknistä että semanttista metatietoa. Lisäksi kuvaaminen tulisi tapahtua yhtenäisellä tasolla. Huomioitavaa on myös mm. Sigalin (1998) havainto, että monet tietovarastoinnin osa-alueista, kuten esimerkiksi metatieto, ovat vielä kypsymässä, joten teknologian kypsyys tulee ottaa huomioon eri ratkaisuja käyttöönotettaessa.

Tietovarastoprojektit sisältävät suuria riskejä (Goodhue ym. 2002) ja ovat kalliita sekä erittäin vaativia projekteja (Jacob & Sen 1998). Lisäksi tietovarastointi teknologiana on hyvin monimutkainen (Campos ym. 2002). Pystyäkseen saavuttamaan etenkin käyttäjien hyväksynnän sekä toisaalta tyydyttämään modernit informaatiotarpeet täytyy metatieto, sen hallinta, kuvaaminen ja jakaminen ottaa yhdeksi tärkeimmäksi osaksi tietovarastointiprojektia. Gardner (1998) sekä Jarke ym. (2000) painottavat, että kaikki tietovaraston komponentit, prosessit sekä tiedot tulisi hallita metatietovarastosta käsin. Brayner ja Carneiro (2002) suosittelevat, että jo tietovarastoprojektin alussa tulisi määrittää strategia metatiedon keräämiselle, ylläpidolle sekä välittämiselle.

Tannenbaumin (2002) mukaan monet metatietoratkaisut ovat epäonnistuneet sen vuoksi, että organisaatiot ovat ensiksi hankkineet metatietotuotteen ja vasta tämän jälkeen koettaneet sovittaa omia vaatimuksiaan tuotteen tarjoamiin ominaisuuksiin. Marco (2001) sekä Fletcher (2002) korostavatkin, että metatietovaraston vaatimusten kerääminen ja niiden määrittäminen on ensiarvoisen tärkeää ja vaatimusten tulisi ohjata metatietotyökalujen valintaa eikä päinvastoin. Myös Geiger ym. (1997) ovat huomioineet, että markkinoilla ei ole tarjolla mitään yhtä mekanismia tai tuotetta, joka pystyisi tallentamaan ja hallitsemaan kaikkia mahdollisia metatiedon ilmentymiä. Tämän vuoksi organisaatioissa tulee analysoida metatietotarpeet ennen ratkaisujen tekemistä.

Lisäksi tulee arvioida organisaation resurssit ja mahdollisuudet kerätä ja ylläpitää vaadittuja metatietoja. Vasta tämän jälkeen tulisi verrata eri tuotteita keskenään ja analysoida, mikä tuote voisi palvella organisaation tarpeita parhaiten. Tämän tutkimuksen tulokset osaltansa auttavat tunnistamaan mitä kaikkea metatietoa on mahdollista kerätä sekä toisaalta arvioimaan mitkä metatiedot ovat relevantteja organisaation kannalta.

Tietovarastot kehitetään yleensä inkrementaalisesti eli niitä rakennetaan asteittain kasvattaen sen sijaan, että rakennettaisiin kerralla koko ympäristö. Myös metatietoratkaisut tulisi kehittää pienissä osissa ja iteratiivisesti (Marco 2001). Tämä tutkimus ehdottaakin, että metatietovaraston suunnittelussa ja rakentamisessa otetaan huomioon ensiksi vain tärkeimmät metatietovaatimukset. Vaatimukset, joiden identifioimiseen tämän tutkimuksen toivotaan tuovan helpotusta. Tämän jälkeen iteratiivisesti sekä inkrementaalisesti lisätään metatieto-ominaisuuksia, jatkuvasti hakien palautetta loppukäyttäjältä. Kuten kohdassa 4.1.9 todettiin, palaute-iteraatio on hyvin keskeisessä osassa koko tietovaraston elinkaaren ajan (Inmon 1996). Näin ollen voidaan olettaa, että samankaltainen lähestymistapa otettaisiin käyttöön myös metatietovaraston ollessa kyseessä. Marco (2001) sanookin, että metatietovaraston rakentamiseen tarvitaan vastaavat resurssit kuin varsinaisen tietovaraston rakentamiseen. Näin ollen metatietoratkaisujen suunnitteluun ja rakentamiseen kohdistuvia resursseja ei tule vähätellä.

Metatietoratkaisun käyttöönotto vaatii ainakin Fletcherin (2002) mukaan enemmän vetoapua liiketoiminnan ja johdon suunnalta kuin pelkkiä teknologisia ponnisteluja. Marco (2002) näkee, että organisaatioissa aletaan hiljalleen ymmärtämään, että metatietoratkaisuihin täytyy tehdä suuriakin investointeja, jotta järjestelmät tuottaisivat todellista arvoa. Olennaista metatietoratkaisujen menestyksessä pidemmällä aikavälillä on niiden tuki standardeille. Standardit luovat yhtenäiset pelisäännöt, joiden puitteissa

tuotteet ja komponentit voivat jakaa metatietoa. Näin luodaan yhtenäinen ympäristö metatiedon hallitsemiselle.

Sekä Marco (2002) että Kerschberg (2001) uskovat, että kun organisaatioiden tietojärjestelmät kehittyvät, ne siirtyvät tiedon keräämisestä ja hallinnoimisesta tietämyksen keräämiseen ja hallinnointiin. Myös Dittrich ja Vaduva (2001) näkevät, että yksi tulevaisuuden haasteista on metatiedon hallinnan sisällyttäminen organisaation laajuisiin tietämyksen hallintajärjestelmiin.

7 YHTEENVETO

Tässä tutkimuksessa keskityttiin tarkastelemaan metatietoa tietovarastoympäristössä. Metatieto on elintärkeää tietovaraston kehittämisessä, ylläpidossa, hallinnassa ja etenkin hyödyntämisessä. Metatieto nähdään myös kriittisimmäksi menestystekijäksi tietovarastoprojekteissa. Tästä huolimatta metatiedon hallintaan, käyttötarkoitukseen ja vaatimusmäärittelyyn liittyy epäselvyyksiä.

Tutkimuksen päätavoitteena oli antaa kattava kuva metatiedosta tietovarastoympäristössä. Tutkimus oli käsitteellis-teoreettinen.

Ensisijaisena tutkimusongelmana oli selvittää, mikä merkitys metatiedolla on tietovarastoympäristössä. Tähän vastattiin luvussa neljä tarkastelemalla metatietoa tietovarastoympäristön eri osa-alueilla ja tuomalla esille miten metatieto tukee tietovaraston ylläpitoa, kehittämistä, komponenttien välistä yhteistoimintaa ja etenkin loppukäyttäjän suorittamaa työtä. Tutkimus toi esille, että metatieto koskettaa jokaista osa-aluetta tietovarastoinnissa. Etenkin loppukäyttäjät tarvitsevat tukea tietovaraston käyttämisessä, tiedon hakemisessa ja haetun tiedon ymmärtämisessä.

Ensimmäinen osaongelma oli: ”Mitä jako tieto ja metatieto merkitsevät tietovarastoissa?” Tähän vastattiin luvussa kolme. Luvussa esiteltiin ensiksi mitä tietotyyppejä tietovarastossa esiintyy ja tehtiin erottelu etenkin liiketoimintatiedon sekä metatiedon välillä. Tutkimus toi myös esille, että selkeää rajausta tiedon ja metatiedon avulla ei ole mahdollista tehdä, sillä yhden sovelluksen tieto voi olla toisen ohjelman metatietoa.

Toinen osa-ongelma oli selvittää: ”Mitä metatieto-tyyppejä tietovarastoista löytyy?” Tähän annettiin vastaus kohdassa 4.1 esittelemällä erilaisia luokitteluja metatiedolle. Lisäksi luvussa kerättiin kirjallisuudesta ja

aikaisemmista tutkimuksista eri metatiedon ilmentymiä ja esimerkkejä viitekehyksen muotoon.

Toissijaisena tutkimusongelmana oli selvittää miten RDF soveltuu tietovarastoympäristössä metatiedon kuvaamiseen. RDF esiteltiin luvussa 5 ja todettiin, että ainakin periaatteessa sen pitäisi soveltua mille tahansa kohdealueelle. Lisäksi se on riittävän ilmaisuvoimainen kuvaamaan korkean tason malleja. Tämä tutkimus ei lähde suosittelemaan mutta ei myöskään hillitsemään RDF:n tai minkään muunkaan metatietomallin tai kielen käyttöä tietovarastojen metatietojen kuvaamisessa. Sen sijaan tärkeämpää on huomioida mikä merkitys metatiedolla on tietovarastoympäristössä. Vasta sen jälkeen voidaan tarttua teknologisten ratkaisujen pohtimiseen.

Tutkimuksen keskeisenä kontribuutiona voidaan nähdä tietovarastoihin liittyvien metatietotyyppien kokoaminen ja luokittelu. Tutkimuksen tuloksia voidaan hyödyntää esimerkiksi metatietojen vaatimusmäärittelyssä tietovarastohankkeissa. Tulosten pohjalta voidaan myös arvioida eri metatietoratkaisuja, lisäksi tulokset vahvistavat ymmärrystä tietovarastoympäristön luonteesta.

Tutkimuksen rajoitteena on etenkin sen käsitteellis-teoreettinen ote, joka on luonnollisesti altis liialliselle subjektiivisuudelle. Jatkotutkimuksessa analyysiä on mahdollista syventää esimerkiksi suunnittelemalla ja toteuttamalla metatietoratkaisuja jossakin tapausympäristössä.

LÄHDELUETTELO

Ballou D., Tayi G. 1999. Enhancing data quality in data warehouse environments. Communication of the ACM 42(1), 73-78.

Bebee B.R., Mack G., Shafi I. 1999. Distributed meta data objects using RDF. Proceedings of the IEEE 8th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, California, USA, June 16-18, 325-329.

Beckett D. 2001. The desing and implementation of the Redland RDF application framework. Proceedings of the Tenth International Conference on World Wide Web (WWW10), Hong Kong, May 1-5, 449-456.

Bhowmick S., Lim E., Madria S., Ng W. 1999. Research issues in web data mining. Proceedings of the First International Conference on Data Warehousing and Knowledge Discovery, Florence, Italy, August 30 – September 1, 303-312.

Blackwood P. 2000. 11 steps to success in data warehousing. Transportation & Distribution 41(2), 60-62.

Bontempo C., Zagelow G. 1998. The IBM data warehouse architecture. Communications of the ACM 41(9), 38-48.

Brayner A., Carneiro L. 2002. X-META: A Methodology for data warehouse desing with metadata management. Proceedings of the 4th International Workshop on Design and Management of Data Warehouses, Toronto, Canada, May 27, 13-22.

Brickley D., Guha R.V. (toim.) 2002. RDF Vocabulary Description Language 1.0: RDF Schema. W3C Working Draft 30 April 2002 [online], [viitattu 16.8.2002]. Saatavilla www-muodossa <http://www.w3c.org/TR/rdf-schema>.

Broekstra J., Decker S., Erdmann M., Fensel D., van Harmelen F., Horrocks I., Klein M., Melnik S. 2000. The semantic web: the roles of XML and RDF. IEEE Internet Computing 4(5), 63-74.

Calvanese D., Giacomo G., Lenzerini M., Nardi D., Rosati R. 2001. Data integration in data warehousing. International Journal of Cooperative Information Systems 10(3), 237-271.

Campos M., Freitas G., Laender A. 2002. MD2 – Getting users involved in the development of data warehouse applications. Proceedings of the 4th International Workshop on Desing and Management of Data Warehouses, Toronto, Canada, May 27, 3-12.

Candan K.S., Liu H., Suvarna R. 2001. Resource description framework: metadata and its applications. ACM SIGKDD Explorations Newsletter 3(1), 6-19.

Carlsson C., Courtney J., Power D., Sharda R., Shim J.P., Warkentin M. 2002. Past, present and future of decision support technology. Decision Support Systems 33(2), 111-126.

Chang D. 2000. Common Warehouse Metamodel (CWM), UML and XML. Metadata Conference, Arlington, VA, March 22 [online], [viitattu 31.3.2002]. Saatavilla www-muodossa <http://www.cwmforum.org/paperpresent.htm>.

Chang W. 1998. A discussion of the relationship between RDF-schema and UML [online], [viitattu 8.8.2002]. Saatavilla www-muodossa <www.w3c.org/TR/1998/NOTE-rdf-uml-19980804>.

Davis G., Hamilton S., Ives B. 1980. A framework for research in computer-based management information systems. Management Science 26(9), 910-934.

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

Dittrich K., Vaduva A. 2001. Metadata management for data warehousing: between vision and reality. Proceedings of the International Database Engineering and Applications Symposium, IDEAS'01, Grenoble, France, July 2001.

Elmasri R., Navathe S. 2000. Fundamentals of Database Systems, 3. Edition. Addison-Wesley.

Everest G., Kim Y-G. 1994. Building an IS architecture: collective wisdom from the field. Information and Management 26(1), 1-11.

Fillies C., Weichardt F., Wood-Albrecht G. 2002. A pragmatic application of semantic web using SemTalk. Proceedings of the 11th International World Wide Web Conference, Honolulu, Hawaii, Usa, May 7-11, 686-692.

Fletcher T. 2002. The "taxing" job of implementing metadata [online], [viitattu 4.6.2002]. Saatavilla www-muodossa <http://www.datawarehouse.com/iknowledge/articles>.

Garcia-Molina H., Hammer J., Labio W.J., Widom J., Zhuge Y. 1995. The Stanford data warehousing project. IEEE Data Engineering Bulletin, Special Issue on Materialized Views and Data Warehousing, 18(2), 41-48.

Gardner S. 1998. Building the data warehouse. Communications of the ACM 41(9), 52-60.

Geiger J., Inmon W.H., Zachman J. 1997. Data Warehousing and the Zachman Framework: Managing Enterprise Knowledge. McGraw-Hill.

Gilliland-Swetland A. 1998. Defining metadata. Teoksessa M. Baca (toim.) Introduction to Metadata: Pathways to Digital Information, Los Angeles: Getty Information Institute, 1-8 [online], [viitattu 1.8.2002]. Saatavilla www-muodossa <http://www.getty.edu/research/institute/standards/intrometadata/2_articles/index.html>.

Goodhue D., Watson H., Wixom B. 2002. The benefits of data warehousing: why some organizations realize exceptional payoffs. Information & Management 39(6), 491-502.

Gray P., Watson H. 1998. Present and future directions in data warehousing. Data Base 29(3), 83-90.

Grossman D., McCabe M.C. 1996. The role of tools in development of a data warehouse. Proceedings of the Fourth International Symposium on Assessment of Software Tools, Toronto, Canada, May 22-24, 139-145.

Hackathorn R. 1995. Data warehousing energizes your enterprise. Datamation 1. February, 38-41.

Hackathorn R. 1999. Web Farming for the Data Warehouse. Morgan Kaufmann.

Haley B., Watson H. 1998. Managerial considerations. Communications of the ACM 41(9), 32-37.

Harold E. 2000. XML –tehokäyttäjän opas. Helsinki: Suomen atk-kustannus.

Herschel R., Iyer L., Nemati H., Steiger D. 2002. Knowledge warehouse: an architectural integration of knowledge management, decision support, artificial intelligence and data warehousing. Decision Support Systems 33(2), 143-161.

Hess T., West L. 2002. Metadata as a knowledge management tool: supporting intelligent agent and end user access to spatial data. Decision Support Systems 32(3), 247-264.

Houben Geert-Jan., Vdovjak R. 2001. RDF based architecture for semantic integration of heterogeneous information sources. International Workshop on Information Integration on the Web (WIIW), Rio de Janeiro, Brazil, April 9-11, 51-57.

Hovi A. 1997. Data warehousing: tietovarastotekniikka. Espoo: Suomen atk-kustannus.

Huynh T., Mangisengi O., Tjoa A. 2000. Metadata for object-relational data warehouse. Proceedings of the International Workshop on Design and Management of Data Warehouses (DMDW'00), Stockholm, Sweden, June 5-6, 3/1-3/9.

Inmon W. H. 1996. Building the Data Warehouse, 2. Edition. John Wiley

Inmon W.H. 1997. Managing the Data Warehouse. John Wiley.

Jacob V., Sen A. 1998. Industrial-strenght data warehousing. Communications of the ACM 41(9), 29-31.

Jarke M., Lenzerini M., Vassiliadis P., Vassiliou Y. 2000. Fundamentals of Data Warehouses. Springer-Verlag.

Jones K. 1998. An introduction to data warehousing: What are the implications for the network? International Journal of Network Management 8(1), 42-56.

Järvinen A., Järvinen P. 2000. Tutkimustyön metodeista. Tampere: Opinpaja Oy.

Kambayashi Y., Kumar V., Mohania M., Samtani S. 1998. Recent advances and research problems in data warehousing. Proceedings of the International Workshop on Data Warehousing and Data mining, Mobile Data Access and New Database Technologies for Collaborative Work Support and Spatio – Temporal Data Management, Singapore, November 19-20, 81-92.

Kerschberg L. 2001. Knowledge management in heterogeneous data warehouse environments. Lecture Notes in Computer Science 2114, 1-10. Springer-Verlag.

Kietz J-U., Vaduva A., Zücker R. 2001. M^4 : a metamodel for data preprocessing. Proceedings of the fourth ACM international workshop on Data warehousing and OLAP, Atlanta, Georgia, USA, November 9, 85-92.

Kim J., Kim T., Lee H. 2001. A Metadata oriented architecture for building data warehouse. Journal of Database Management 12(4), 15-25.

Kimball R. 1996. The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses. John Wiley.

Kimball R. 1998a. Brave new requirements for data warehousing. Intelligent Enterprise Magazine [online] 1(1), [viitattu 24.5.2002]. Saatavilla www-muodossa <http://www.intelligententerprise.com/db_area/archives/1998/9810/warehouse.shtml>.

Kimball R. 1998b. Meta meta data data. DBMS [online] 11(3), [viitattu 20.6.2002]. Saatavilla www-muodossa <http://www.dbmsmag.com/9803d05.html>.

Kimball R. 2001. XML will make it easier. Intelligent Enterprise Magazine [online] 4(6), [viitattu 24.5.2002]. Saatavilla www-muodossa <http://www.intelligententerprise.com/010416/webhouse1_1.shtml>.

Koeller A., Rundensteiner E., Zhang X. 2000. Maintaining data warehouses over changing information sources. Communications of the ACM 43(6), 57-62.

Lassila O., Swick R. (toim.) 1999. Resource Description Framework (RDF) model and syntax specification. W3C Recommendation 22 February 1999 [online], [viitattu 15.8.2002]. Saatavilla www-muodossa <http://www.w3.org/TR/1999/REC-rdf-syntax-19990222>.

Laudon J., Laudon K. 1998. Management Information Systems: New Approaches to Organization and Technology, 5. Edition. Prentice Hall.

Lucas H.C. 1975. Why Information Systems Fail. Columbia University Press.

Lyytinen K. 1987. Different perspectives on information systems: problems and solutions. ACM Computing Surveys 19(1), 5-46.

Marco D. 2001. Top 10 mistakes to avoid when implementing a meta data repository [online], [viitattu 4.6.2002]. Saatavilla www-muodossa <http://www.datawarehouse.com/iknowledge/articles>.

Marco D. 2002. Meta data repositories: Where we've been and where we're going [online], [viitattu 26.4. 2002]. Saatavilla www-muodossa <http://www.dmreview.com/portal_ros.cfm?NavID=91&EdID=4612&PortalID=18>.

Miller E. 1998. An introduction to the Resource Description Framework. D-Lib Magazine [online] 4(5), [viitattu 12.8.2002]. Saatavilla www-muodossa <http://www.dlib.org/dlib/may98/miller/05miller.html>.

Miller L., Nilakanta S., Wu L. 2001. Design of data warehouses using metadata. Information and Software Technology 43(2), 109-119.

Mohania M., Vincent M., van Zyl J. 1998. Representation of metadata in a data warehouse. Proceedings of the IEEE Region 10th International Conference on Global Connectivity in Energy, Computer, Communication and Control, New Delhi, India, December 17-19, 103-106.

Moriarty T. 2000. The naked truth. Intelligent Enterprise Magazine [online] 3(17), [viitattu 24.5.2002]. Saatavilla www-muodossa <http://www.intelligententerprise.com/001110/metaprise.shtml>.

Mullen N. 2002. Information for innovation: positioning metadata [online], [viitattu 3.7. 2002]. Saatavilla www-muodossa <http://www.dmreview.com/master.cfm?NavID=198&EdID=5349>.

Müller R., Rahm E., Stöhr T. 1999. An integrative and uniform model for metadata management in data warehousing environments. Proceedings of the International Workshop on Desing and Management of Data Warehouses, Heidelberg, Germany, June 14-15, 12/1-12/16.

Nonaka I. 1991. The Knowledge-creating company**.** Harvard Business Review 69, November-December, 96-104.

Object Management Group (OMG). 2001. The Common warehouse metamodel spesification 1.0 [online], [viitattu 14.6.2002]. Saatavilla www-muodossa <http://www.cwmforum.org/spec.htm>.

Pokorný J. 2001. Modelling stars using XML. Proceedings of the Fourth ACM International Workshop on Data Warehousing and OLAP, Atlanta, USA, November 9, 24-31.

Poole J. 2001. Model-Driven architecture: vision, standards and emerging technologies. ECOOP 2001 Workshop on Metamodeling and Adaptive Object Models [online], [viitattu 25.4. 2002]. Saatavilla www-muodossa <http://www.cwmforum.org/Model-Driven%20Architecture.pdf>.

Rosenthal A., Seligman L. 2001. XML's impact on databases and data sharing. Computer 34(6), 59-67.

Sigal M. 1998. A common sense development strategy. Communications of the ACM 41(9), 42-43.

Singh N. 1998. Unifying heterogenous information models. Communications of the ACM 41(5), 37-44.

Staudt M., Vaduva A., Vetterli T. 2000. Metadata standards for data warehousing: Open Information Model vs. Common Warehouse Metamodel. ACM Sigmod Record 29(3), 68 – 75.

Tannenbaum A. 2002. Identifying Meta data requirements. Journal of Data Warehousing [online], 7(2) [viitattu 23.5.2002]. Saatavilla pdf-muodossa <http://www.dw-institute.com/research/display.asp?id=6380>.

Vaduva A., Vetterli T. 2001. Metadata management for data warehousing: an overview. International Journal of Cooperative Information Systems 10(3), 273-298.

Variar G. The origin of data. Intelligent Enterprise Magazine [online] 5(3), [viitattu 24.5.2002]. Saatavilla www-muodossa <http://www.intelligententerprise.com/020201/503feat3_1.shtml>.

Watson R. 2000. An enterprise information architecture: a case study for decentralized organizations. Proceedings of the 33rd Hawaii International Conference on System Sciences, Maui, Hawaii, January 4-7.

White C. 1999. Managing distributed data warehouse metadata [online], [viitattu 26.4. 2002]. Saatavilla www-muodossa <http://www.dmreview.com/master.cfm?NavID=55&EdID=159>.

Widom J. 1995. Research problems in data warehousing. Proceedings of the 1995 Conference on International Conference on Information and Knowledge Management, Baltimore, Maryland, USA, 25-30.

Wiener J. 2000. Metadata in context [online], [viitattu 4.6.2002]. Saatavilla www-muodossa <http://www.datawarehouse.com/iknowledge/articles> .

Zachman J. 1987. A framework for information systems architecture. IBM systems Journal 26(3).