Ville Porkka

ASIAKKAAN DATAN MALLINNUS JA SEN ANALYSOINTI BUSINESS INTELLIGENCE -JÄRJESTELMÄSSÄ

Tietojenkäsittelyn koulutusohjelma

2017

samk
Satakunnan ammattikorkeakoulu
Satakunta University of Applied Sciences

ASIAKKAAN DATAN MALLINNUS JA SEN ANALYSOINTI BUSINESS INTELLIGENCE -JÄRJESTELMÄSSÄ

Porkka, Ville
Satakunnan ammattikorkeakoulu
Tietojenkäsittelyn koulutusohjelma
Toukokuu 2017
Ohjaaja: Nieminen, Hans
Sivumäärä: 34
Liitteitä: 1

Asiasanat: business intelligence, tietokannat, tietovarastot, SQL, analyysi

---

Opinnäytetyön tavoitteena oli toteuttaa asiakkaalle Business Intelligence -järjestelmää käyttäen ympäristö kyselytutkimuksella kerätyn datan analysointia varten. Asiakkaan analyyseille asettamien vaatimusten vuoksi aiemmin kerätty data oli muutettava Business Intelligence -järjestelmän, ja erityisesti tietovarastoinnin kannalta optimaaliseen, moniulotteiseen muotoon. Asiakkaalle luotiin uusi relaatiotietokanta moniulotteisessa mallinnuksessa yleisesti käytetyn tähtimallin pohjalta. Asiakkaan alkuperäinen data siirrettiin luotuun relaatiotietokantaan. Toteutukseen kuului myös valitun Business Intelligence -järjestelmän asentaminen, ja analyysien luominen siihen liitetyn uuden tietokannan pohjalta.

Toteutuksessa käytettäviksi ohjelmistoiksi valikoituivat Pentaho Community Edition Business Intelligence -järjestelmäksi ja MySQL tietokannan hallintajärjestelmäksi.

Opinnäytetyön teoreettisessa osuudessa käsiteltiin Business Intelligence -käsitteen sisältöä erityisesti toteutuksessa käytettävien Business Intelligencen osa-alueiden ja komponenttien osalta.

Opinnäytetyön valmistuttua asiakkaalla on käytössään tietokanta ja Business Intelligence -järjestelmä, sekä sen sisältämät työkalut aiemman ja myös mahdollisesti uuden datan analysointiin ja niistä raportointiin.

MODELING OF CLIENT'S DATA AND ANALYZING IT IN A BUSINESS INTELLIGENCE SYSTEM

Porkka, Ville
Satakunnan ammattikorkeakoulu, Satakunta University of Applied Sciences
Degree Programme in Business Information Technology
May 2017
Supervisor: Nieminen, Hans
Number of pages: 34
Appendices: 1

Keywords: Business Intelligence, Database, Data Warehouse, SQL, Data Analysis

---

The purpose of this thesis was to implement a Business Intelligence system for the client to enable analysis of the survey data. Due to the client's requirements for the analyzes, the previously collected data had to be re-modeled using dimensional model optimized for Data Warehouse / Business Intelligence environment. A new database was created for a client using most widely used dimensional model, star schema. The client's old data was migrated to the created database. The implementation also included the installation of the selected Business Intelligence system and data analysis from the connected database.

The programs used in this thesis were Pentaho Community Edition as a Business Intelligence software and MySQL as a database management system.

The theory section of the thesis focused on the concept of Business Intelligence, particularly on the parts and components used in the implementation.

After the implementation is completed, the client has a database and a functional Business Intelligence system including the tools for data analysis and reporting.

SISÄLLYS

1 JOHDANTO

Opinnäytetyön asiakkaana on TTY:ssä turvallisuuskulttuuriin liittyvässä tutkimuksessa päätutkijana toiminut Pasi Porkka. Tutkimuksessa tehtiin asiakasyrityksiin turvallisuuskulttuuriin liittyvä kysely, jonka avulla saatu data on tallennettu relaatiotietokantaan. Kyselydata on mallinnettu tapahtumakeskeisellä tietomallilla. Tämän opinnäytetyön tarkoituksena on asiakkaan kyselytutkimuksella saaman datan jatkojalostaminen ja monipuolisempien analyysien mahdollistaminen.

Asiakas on asettanut analyyseille vaatimuksia, jotka liittyvät erilaisten ryhmien välisiin vertailuihin. Kyselydatan tietomalli ei sellaisenaan sovellu vertailuihin, sillä tapahtumakeskeisessä tietomallissa vertailuissa käytettävät attribuutit ovat jakautuneet useisiin tietokannan tauluihin, jolloin ryhmittely vaatii aina useita taulujen välisiä liitoksia. Moniulotteisten analyysien toteuttamiseksi käytetään liiketoimintatiedon hallintaan (BI, Business Intelligence) liittyvää tosiaikaista tiedonjalostusta, OLAP:ia (Online analytical processing). Tosiaikainen tiedonjalostus perustuu datan moniulotteisuuteen, joten asiakkaan data on uudelleenmallinnettava. Moniulotteisessa mallinnuksessa varsinaiset tietoalkiot ovat faktatauluissa (fact tables) ja ryhmittelyissä käytettävät attribuutit ulottuvuustauluissa (dimension tables). Näin tarvittavien taulujen välisten liitosten määrä pienenee ja analyysi nopeutuu merkittävästi.

Asiakkaan datan ja analyysien vaatimusten pohjalta luotiin uusi tietokanta moniulotteisessa mallinnuksessa käytettävän tähtimallin mukaisesti ja asiakkaan data siirrettiin uuteen tietokantaan. Työssä käytettävänä tietokannan hallintajärjestelmänä toimii MySQL. Toteutuksessa käytettäväksi BI-järjestelmäksi valittiin Pentaho Community Edition, joka tarjoaa kaikki tarvittavat työkalut datan kokonaisvaltaiseen käsittelyyn. Toteutukseen kuuluvat myös Pentahon asennus ja käyttöönotto, sekä datan analysoinnin mahdollistaminen.

Työn teoreettisen osan tärkein yksittäinen lähde on pitkän uran tietovarastojen ja Business Intelligencen parissa tehneen Ralph Kimballin kirja The Data Warehouse Toolkit. Myös tietovarastoinnin (paikallisvaraston) toteutus vastaa pääpiirteittäin Kimballin lähestymistapaa tietovarastointiin ja Business Intelligenceen.

Toisessa kappaleessa kerrotaan tarkemmin asiakkaan viitekehyksestä ja työn tavoitteista ja toiveista. Kolmannessa kappaleessa kerrotaan Business Intelligencen ja erityisesti tämän työn kannalta tärkeiden käsitteiden teoriasta. Neljäs kappale keskittyy datan moniulotteisen mallinnuksen toteuttamiseen, sekä tietokannan fyysiseen toteutukseen. Viidennessä kappaleessa kerrotaan tarkemmin Pentahosta ja sen tärkeimmistä komponenteista. Kuudes kappale sisältää itse työn toteutuksen Pentahossa. Viimeisessä kappaleessa arvioidaan tavoitteiden toteutumista ja mahdollisia parannus- ja lisäysehdotuksia.

2 ASIAKKAAN VIITEKEHYS

Tässä kappaleessa esitellään lyhyesti asiakkaan tavoitteet ja toiveet työn kannalta. Erityistä huomiota kiinnitetään asiakkaalta saatuun dataan, jota on tarkoitus analysoida tässä työssä kehitetyillä menetelmillä. Data on kerätty Työsuojelurahaston rahoittamassa PRO-Turva -projektissa. Projektissa tehtiin turvallisuuteen liittyvä kysely, johon vastasi 10 yritystä Harjavallan teollisuuspuistosta ja 4 energia-alan yritystä. Kyselyyn vastasi kaikista yrityksistä yhteensä 565 vastaajaa. (Porkka 2012.)

PRO-Turva -projektissa määriteltiin organisaation turvallisuuskulttuuriin liittyviä piirteitä. Tutkittavia piirteitä valikoitui 17 kappaletta ja ne luokiteltiin hierarkkisesti kahdeksaan alaryhmään ja kahteen yläryhmään. Piirteiden tutkimista varten tehtiin kysely, jossa oli 51 väittämää. Piirteiden ja niihin liittyvien väittämien muodostamaa kokonaisuutta kutsuttiin ontologiaksi. Kukin vastaaja arvioi jokaisen väittämän nykytilan, eli miten koki väittämän esittämän asian olevan yrityksessä vastaushetkellä. Vastaaja arvioi myös tavoitetilan, eli mihin toivoi väittämän tulevaisuudessa muuttuvan. Vastaukset tallennettiin tietokantaan numeerisessa muodossa. Tavoitteen ja nykytilan välinen erotus kertoo muutostoiveen suuruuden, eli luovan jännitteen (Senge 1990). Luova jännite laskettiin tavoite- ja nykytilan numeeristen arvojen erotuksena. Jokaista väittä-

mää kohti on yhden vastaajan datassa siis kolme numeerista arvoa. (Porkka henkilökohtainen tiedonanto 14.2.2017). Koska vastaajia oli 565, kertyi vastauksista data-alkioita 565 * 51 * 3 = 86 445.

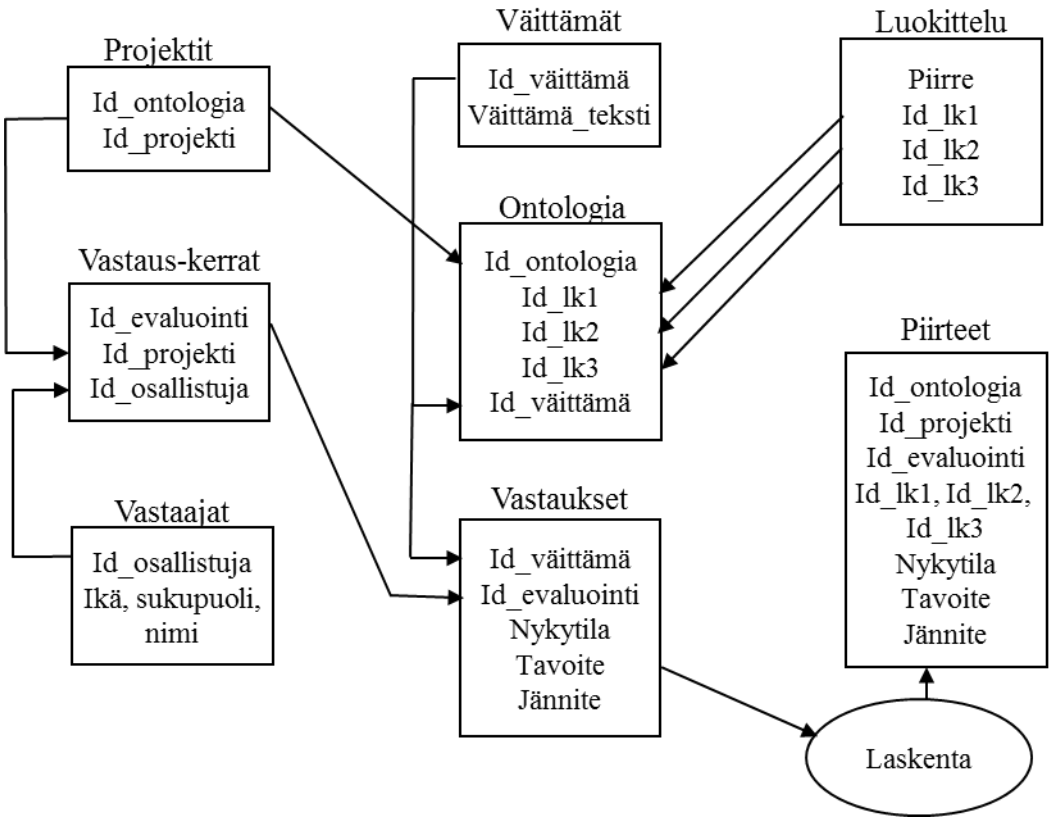
Tutkimuskohteena oleviin 17 piirteiseen liittyi 2-5 väittämää, joiden avulla laskettiin myös piirteille nykytilan, tavoitetilan ja luovan jännitteen numeeriset arvot. Piirteisiin liittyviä data-alkioita kertyi 565 * 3 * 17 = 28 815. Lisäksi piirteiden luokittelusta saatiin 565 * 3 * (8+2) = 16 950 data-alkiota. Kokonaisuudessaan kyselyn avulla tuotettiin 132 210 data-alkiota.

Varsinaisen kyselydatan lisäksi vastaajilta kerättiin demografista dataa, kuten sukupuoli, ikä, työkokemus, asema yrityksessä, osasto, työntekijä / esimies, hallinto / tuotanto, jne. Tämän työn tavoitteena on toteuttaa asiakkaan toivomat uudet analyysit, jotka pohjautuvat kyselyllä kerättyyn dataan. Analyysien tavoitteena on mahdollistaa demografiseen dataan perustuvat ryhmien vertailut. Analyysit edellyttävät datan uudelleen mallinnusta moniulotteiseksi ja sen hyödyntämistä BI-järjestelmässä.

2.1 Asiakkaalta saatu data

Kyselyillä saatu data on talletettu MySQL-relaatiotietokantaan. Tietokanta koostuu 21 taulusta, jotka liittyivät toisiinsa viiteavaimilla. Tiedon mallinnus on toteutettu siten, että kunkin vastaajan demografinen data, arviot väittämistä, sekä piirteiden lasketut arvot muodostavat kokonaisuuden. Mallinnuksen tarkoituksena on yhden vastaajan kyselyyn liittyvien arvojen tehokas tallennus. Mallinnus on täten tapahtumakeskeinen (transactional).

Kuvassa 1 on esitetty yksinkertaistettu malli tietokannan rakenteesta. Esiteltäviin tauluihin on valittu vain tärkeimmät viiteattribuutit sekä muutamia kuvaavia attribuutteja.



Kuva 1. Yksinkertaistettu malli asiakkaan kyselydataa sisältävästä tietokannasta

Tässä työssä käsiteltävä turvallisuusdata oli asiakkaan tietokannassa vain yksi useista ontologioista, tästä syystä myös ontologialla on oma tunnisteensa (Id_ontologia). Seuraavassa lyhyet kuvaukset tietokannan oleellisista tauluista.

Projektit-taulu sisältää projektin yleistä tietoa. Vastaajan organisaatioon liittyvä tieto oli omassa taulussaan, jota ei ole otettu tähän mukaan. Asiakkaan projekteihin saattoi liittyä useaan eri ontologiaan vastaaminen, joten myös ontologia-taulun Id_ontologia on viiteavain.

Vastaus-kerrat-taulu mahdollistaa saman henkilön usean vastaamisen samaan ontologiaan. Yhdestä vastauskerrasta käytetään termiä evaluointi, jonka tunnus on Id_evaluointi. Usean vastaamiskerran tarjoaminen mahdollistaa pitkittäistutkimuksessa saman henkilön eri vastauskertojen vertailun.

Vastaajat-taulu sisältää vastaajan perustiedot, eli henkilökohtaisen demografisen datan. Tauluun viitataan Id_osallistuja avaimella.

Väittämät-taulussa esitetään väittämät.

Luokittelu-taulussa on piirteet sekä ryhmä ja pääryhmä, johon piirre sisältyy.

Ontologia-taulu kokoaa yhteen ontologian. Siinä yhdistetään ontologian väittämät piirteisiin. Sama väittämä voi esiintyä useassa eri ontologiassa.

Vastaukset-tauluun on talletettu vastaajan väittämiin antamat arvot. Koska vastaajalla on mahdollisuus vastata samaan kyselyyn useammin kuin kerran ja koska sama väittämä voi esiintyä toisessa ontologiassa, johon myös on vastattu, sidotaan vastaus myös evaluointikertaan viiteattribuutilla Id_evaluointi. Saman viiteattribuutin kautta päästään myös vastaajan tietoihin.

Piirteet-taulu sisältää piirteille lasketut vastaajakohtaiset arvot. Ulkoisella laskennalla on saatu väittämien arvoista piirrekohtaiset nykytila, tavoitetila ja luovan jännitteen arvot.

2.2 Analyysien kriteerit

PRO-Turva –projektissa analyysien pääpaino oli yrityskohtaisissa kokonaistuloksissa. Yrityskohtaisissa raporteissa esitettiin muutamia ryhmäkohtaisia vertailuja. Esimerkiksi tuotannon työntekijöiden ja esimiesten vertailu toi esille selkeitä näkemyseroja, jotka olivat merkittäviä turvallisuuden kannalta. (Porkka henkilökohtainen tiedonanto 14.2.2017.) Yhteen vastaajaan keskittyvä tiedonmallinnus ei tue ryhmien välisiä monipuolisia vertailuja helposti, sillä vertailujen toteuttaminen vaatii useita taulujen välisiä liitoksia. Tässä työssä toteutettavat analyysit perustuvat erilaisten ryhmien välisiin vertailuihin yli yritysten rajojen.

Asiakkaan toiveena on useiden samanaikaisten ryhmittelyjen vertailu, esimerkiksi kahden eri yrityksen naispuolisten toimihenkilöiden käsitysten vertailu. Koska mahdollisten ryhmien kombinaatioiden määrä on suuri, esitetään ainoastaan asiakkaan ryh-

mittelykriteerit. Ryhmittelykriteerit määrittelevät moniulotteisen tietomallin dimensiotaulut. Toteutusalustaksi valittu Pentaho mahdollistaa ryhmittelyjen monipuolisen valinnan dimensiotaulujen avulla.

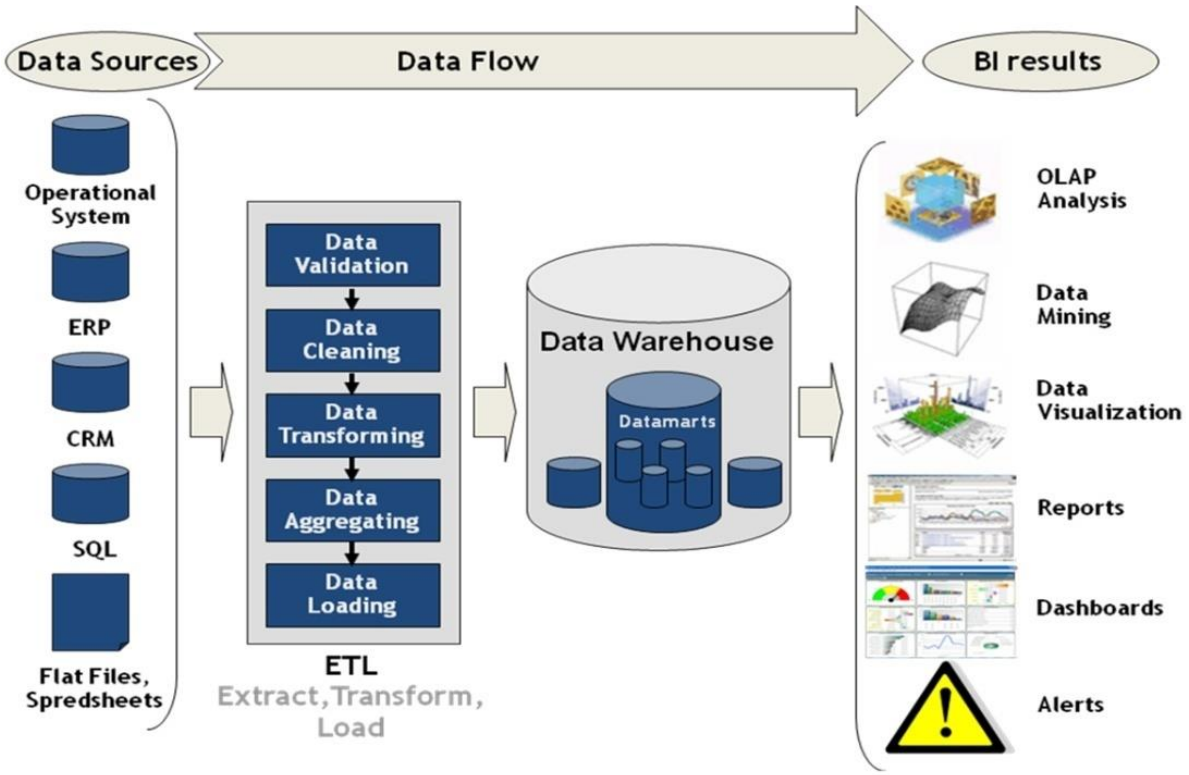
Osalla ryhmittelyyn käytettävillä attribuuteilla on hierarkia, jonka tulee esiintyä mallinnuksessa. Ryhmittelyn tulee olla mahdollista seuraavien attribuuttien kohdalla:

1. Henkilö: sukupuoli, koulutus, kokemus
2. Piirre: piirre → luokka → pääluokka
3. Evaluointi: projekti → organisaatio → maa

Analyysit tehdään sekä piirteiden että väittämien osalta. Molempien kohdalla tulee kaikkien ryhmittelyattribuuttien olla käytettävissä. Analyysit on voitava tehdä nykytilan, tavoitetilan ja luovan jännitteen osalta. Kunkin vastaajan kohdalla on laskettava sekä piirteiden että väittämien sijaluvut (ranks) lajittelemalla vastaajan antamat arvot nousevaan suuruusjärjestykseen. Sijaluvut on tallennettava tietokantaan ja ryhmittelyn on oltava mahdollista myös sijalukujen kohdalla.

3 BUSINESS INTELLIGENCE

Business Intelligence (liiketoimintatiedon hallinta) on käsitteenä laaja kokonaisuus, joka sisältää prosessit, teknologian ja muut tarvittavat työkalut yrityksen keräämän datan hyödyntämiseksi liiketoiminnassa (Loshin 2003, 6). Aiemmin kerrottujen asiakkaan analyyseille asettamien kriteerien vuoksi työn toteutuksessa syntyi tarve moniulotteisen ja historiallisen datan analysointiin sekä raportointiin kykenevälle ohjelmistolle. BI / Data Warehouse -pohjaiset järjestelmät vastasivat tähän tarpeeseen täydellisesti. Kuvassa 2 on esitetty teoreettisen BI-järjestelmän sisältöä ja toiminnallisuutta.



Kuva 2. BI-järjestelmän toiminnallisuus (Krmac 2011)

Tietoa kerätään ETL-prosessin (Extract, transform, load) avulla eri lähteistä (Data Sources) kuten SQL-tietokannoista tai yksittäisistä tiedostoista. ETL-prosessissa tieto puhdistetaan muun muassa korjaamalla mahdolliset kirjoitusvirheet ja päällekkäisyydet, sekä yhtenäistämällä nimeämiskäytäntöjä ja tietotyyppejä. ETL-prosessin lopussa tieto ladataan puhdistettuna tietovarastoon (Data Warehouse) (Kimball & Ross 2013, 19). Tiedon hyödyntäminen (analysointi, visualisointi, tiedonlouhinta, raportointi yms.) tapahtuu prosessin lopussa tietovaraston dataa käyttäen.

3.1 Tietovarasto

Yksi Business Intelligencen keskeisimmistä osista on tietovarasto (Data Warehouse). Tietovarastojen kehitys ja käyttöönotto tapahtuivat 1980-luvun loppupuolella ja laajemmin 1990-luvulla yritysten huomatessa strategisen tiedon merkityksen liiketoiminnassa. Perinteiset tosiaikaiseen tapahtumakäsittelyyn (Online transaction processing,

OLTP) perustuvat järjestelmät tukevat yritysten perustoimintaa tallentamalla tapahtumakeskeistä ja toiminnallista tietoa, kuten tilauksen tai myyntitapahtuman tiedot, yrityksen operatiivisiin tietokantoihin. Tietovaraston tehtävänä on jalostaa ja koota operatiivisten tietokantojen suuria datamääriä kysely-ystävällisempään ja analyyttisempaan muotoon. Taulukossa 1 esitetään operatiivisen järjestelmän ja tietovarastojärjestelmän oleellisimmat erot. (Ponniah 2010, 13-14.)

Taulukko 1. Operatiivisen järjestelmän ja tietovarastojärjestelmän vertailu (Ponniah 2010, 13)

	OPERATIIVINEN	TIETOVARASTO
Tiedon sisältö	Nykyistä tietoa	Historiallista, koottua tietoa
Tiedon rakenne	Tapahtumapohjaista	Optimoitua, moniulotteista
Hakutiheys	Jatkuvaa tai hyvin suurta	Keskivertoa tai harvaa
Funktionaalisuus	Lukeminen, päivitys, poistaminen	Lukeminen
Käyttö	Ennakoitavaa, toistuvaa	Asiakohtaista, satunnaista, heuristista
Vasteaika	Erittäin nopea	Kyselyistä riippuen hitaampaa
Käyttäjät	Paljon käyttäjiä	Suhteellisen vähän käyttäjiä

Tietovarastojen isänäkin pidetty Bill Inmon määrittelee tietovarastot aihekeskeisiksi (subject oriented), koostetuiksi (integrated), pysyviksi (nonvolatile) ja aikariippuvaisiksi (time-variant) kokoelmiksi dataa, jota käytetään liikkeenjohdon päätösten tukena. Aihekeskeisyydellä tarkoitetaan tiedon keräämistä ja analysointia liiketoiminnasta riippuen tietyltä osa-alueelta, kuten myynnistä, tuotteista tai asiakkaista. Koosteisuus tarkoittaa tiedon keräämistä eri lähteistä ja tiedon yhtenäistämistä tietovarastoon. Pysyvyys tarkoittaa historiallisen tiedon säilyvyyttä ja vakautta. Tietovarastossa oleva data on siis vain luettavissa ja sitä ei voi muuttaa. Aikariippuvuus tarkoittaa kaiken tietovaraston sisältämän tiedon olevan kiinnitettynä tiettyyn ajanhetkeen. Aikariippuvuus mahdollistaa tietovaraston historiallisuuden, joka on aikahorisontista riippuen yleensä viidestä kymmeneen vuotta. (Inmon 2002, 31-35.)

Tietovarastot koostuvat usein pienemmistä paikallisvarastoista (data mart). Paikallisvarastot on suunnattu pienemmälle käyttäjäryhmälle, kuten yrityksen osastolle. Ne ku-

vaavat yleensä yksittäistä liiketoimintaprosessia ja tarjoavat siten tietovarastoon verrattuna yksityiskohtaisempaa ja tarkempaa tietoa. Tietovarastoa pienemmän kokonsa vuoksi paikallisvarastot ovat halvempia ja helpompia toteuttaa, ja ne mahdollistavat yksinkertaisemmat kyselyt ja nopeammat vastausajat. Paikallisvarastot rakennetaan moniulotteisiksi. (Ponniah 2010, 30.)

3.2 OLAP

OLAP (Online analytical processing), suomennettuna ”tosiaikainen tiedonjalostus”, on muiden Business Intelligenceen liittyvien käsitteiden tavoin määritelty usealla eri tavalla kirjoittajasta ja ajankohdasta riippuen.

OLAP-termin vuonna 1993 lanseerannut, myös relaatiomallin luojana tunnettu Edgar F. Codd määrittelee OLAP:in dynaamiseksi analyysiksi, jota tarvitaan luomaan, käsittelemään, elävöittämään ja yhdistämään tietoa yrityksen tietovarastoista. Siihen sisältyy mahdollisuus huomata uudet tai odottamattomat yhteydet muuttujien välillä, mahdollisuus tunnistaa tarvittavat parametrit suurien tietomäärien hallitsemiseen, mahdollisuus luoda rajoittamaton määrä dimensioita, sekä mahdollisuus määritellä dimensioiden väliset ehdot ja lausekkeet. (Codd, Codd & Salley 1993, 8-9.) Vuosituhannen lopussa useiden suurten ohjelmistoalan yritysten perustama OLAP Council taas määrittelee OLAP:in ohjelmistoiksi, jotka mahdollistavat analyytikoiden ja johtajien saavan nopean, yhtenäisen, interaktiivisen ja laajan näkymän tietoon, joka on muutettu raa’asta datasta käyttäjän ymmärtämään, yrityksen oikeaa toimintaa kuvaavaan, moniulotteiseen muotoon. (Olap Council, 1997.) Yksinkertaistettuna OLAP tarkoittaa lähes kaikista BI-järjestelmistä löytyviä esityskerroksen työkaluja, joilla esitetään tietovarastosta saatua dataa moniulotteisena.

3.3 Moniulotteinen mallinnus

Moniulotteinen mallinnus (dimensional modeling) on tietovarastoissa yleisesti käytettävä tiedon mallinnustapa. Sen avulla tietokannoista saadaan yksinkertaisempia, suorituskyvyltään tehokkaampia ja helpommin muokattavia (Kimball & Ross 2013, 7).

Moniulotteinen mallinnus liittyy läheisesti useisiin Business Intelligencen osiin, erityisesti tietovarastoihin ja sen seurauksena OLAPiin.

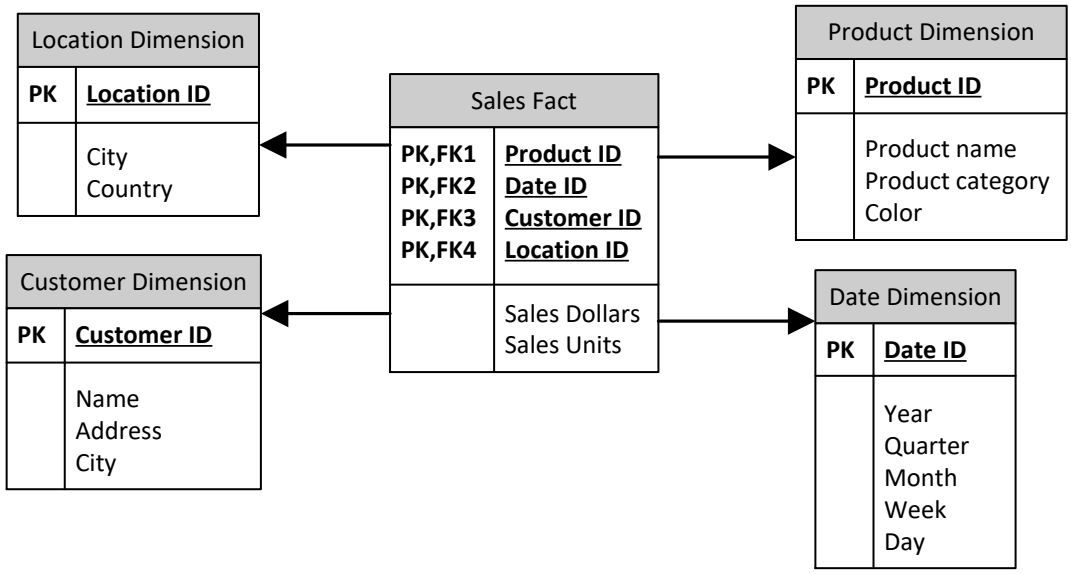
Yleisimmät datan moniulotteisessa mallinnuksessa käytettävät tietomallit ovat tähtimalli (star schema) ja tähtimallin normalisoidumpi versio, lumihiutalemalli (snowflake schema). Moniulotteiset tietomallit koostuvat faktoista (fact) ja dimensioista (dimension). Faktat ovat lähes aina laskettavassa, numeerisessa muodossa olevia arvoja, ja dimensiot ovat usein hierarkkisia ryhmiä, jotka määrittelevät millä tavoin faktoja voidaan analysoida.

Tähtimalli (Star schema) on yleisin ja yksinkertaisin moniulotteisessa mallinnuksessa käytetty tietomalli. Tähtimalli koostuu yhdestä tai useammasta faktataulusta ja se viittaa määrittelemättömään määrään dimensiotauluja. Tähtimalli saa nimensä tähteä muistuttavasta rakenteestaan, jossa dimensiotaulut ovat tähden sakaroita. Kuvassa 3 on yksinkertainen esimerkki tähtimallista, jossa faktataulu Sales Fact on keskellä ja dimensiotaulut reunoilla.

Faktataulujen tiedot ovat tapahtumatyyppisiä, ja jokainen rivi vastaa yhtä tapahtumaa, kuten myyntitapahtumaa. Jokaisen faktataulun rivin on oltava yhtä yksityiskohtainen (grain), esimerkiksi yksi rivi jokaista myyntitapahtumaa kohden. Faktataulujen arvot ovat lähes aina numeerisia, ja niitä voidaan laskea yhteen erilaisilla dimensioista saaduilla tasoilla. Myös keskiarvoja ja lukumääriä voidaan laskea valmiiksi. Faktataulujen tulisi olla mahdollisimman niukkoja (sparse). Niukkuudesta huolimatta faktataulut vievät yleensä jopa 90% kaikesta tähtimallin käyttämästä tilasta. Kaikilla faktatauluilla on kaksi tai useampi viiteavain, joilla viitataan dimensiotaulujen perusavaimiin. Esimerkiksi kuvassa 3 Sales-faktataulun viiteavain Product ID on Product-dimensiotaulun perusavain. Viiteavainten avulla turvataan tietokannan viite-eheys. Faktataulun perusavain on yleensä yhdistelmäavain (composite key), joka muodostuu viiteavainten joukosta. (Kimball & Ross 2013, 10-12.)

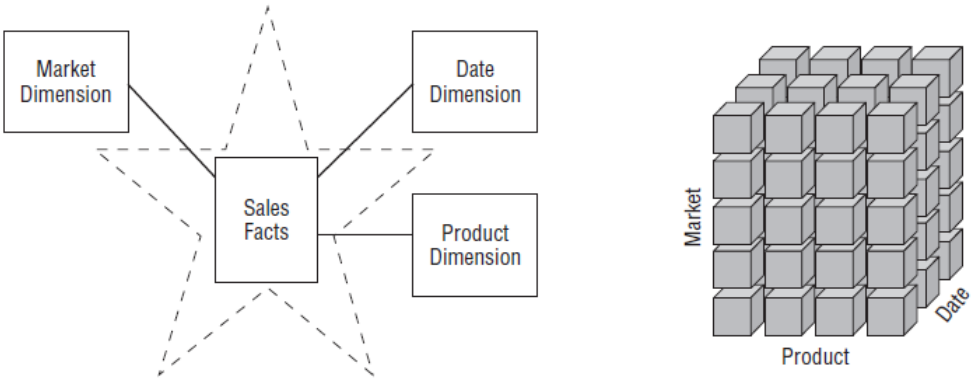
Dimensiotaulut ovat faktataulujen pohja, joissa säilytetään kaikki faktataulujen tapahtumat määrittelevät attribuutit. Dimensiotaulujen attribuutit määrittelevät millä tavoin faktataulujen tapahtumia voidaan ryhmitellä ja analysoida. Dimensiotaulut siis käytännössä määrittelevät koko BI-järjestelmän tehokkuuden. Dimensiotaulut ovat usein

järjestetty hierarkkisesti, jolloin tiedon analysointi on helpompaa ja suorituskyky parempi. (Kimball & Ross 2013, 13-15.)



Kuva 3. Esimerkki yksinkertaisesta tähtimallista

Moniulotteisuuden kuvaamiseen käytetään myös OLAPiin liittyvää käsitettä tietokuutio (data cube, OLAP cube), joka esittää tietomallin kuutiomaisessa muodossa. Tietokuutiot ovat usein relaatiopohjaisen tähtimallin pohjalta rakennettuja (ROLAP, relational online analytical processing), mutta ne voivat myös kuvata itsenäisiä, moniulotteisia tietokantoja (MOLAP, multidimensional online analytical processing). Kuvassa 4 on vierekkäin tähtimalli ja tietokuutio.



Kuva 4. Tähtimalli verrattuna tietokuutioon (Kimball & Ross 2013, 9)

Kun data on ladattu OLAP-kuutioon, se säilötään ja indeksoidaan käyttäen moniulotteisen datan käsittelyyn optimoituja formaatteja ja tekniikoita. Kuution perusdata saadaan faktatauluista, jonka lisäksi OLAP-järjestelmät voivat laskea ja säilöä koosteita (aggregate) valmiiksi nopeuttaen kyselyaikoja. (Kimball & Ross 2013, 8-9.) OLAP-järjestelmä sisältää kuution käsittelyyn ja sen sisältämän datan esittämiseen käytettäviä operaatioita, joista yleisimmät ovat siirtyminen ylemmälle tasolle (roll-up), porautuminen yksityiskohtaisempaan tietoon (drill down), kuution määritellyn osan, viipaleen, tarkastelu (slicing), ja osakuution tekeminen määriteltyjen dimensioiden arvojen mukaan (dicing). (Ponniah 2010, 390-392.)

4 MONIULOTTEISEN MALLINNUKSEN TOTEUTUS

Asiakkaan kappaleessa 2.2 analyyseille asettamien kriteereiden vuoksi kyselyillä kerätty data on muutettava moniulotteiseen muotoon. Moniulotteisen mallinnuksen toteuttamiseksi luodaan uusi relaatiotietokanta käyttäen moniulotteista tietomallia, tähtimallia. Tietokannan hallintajärjestelmänä käytetään Oraclen MySQL Community -ohjelmistoa. Tietokannan luomisen jälkeen asiakkaan data siirretään uuteen tietokantaan.

4.1 Mallin suunnittelu

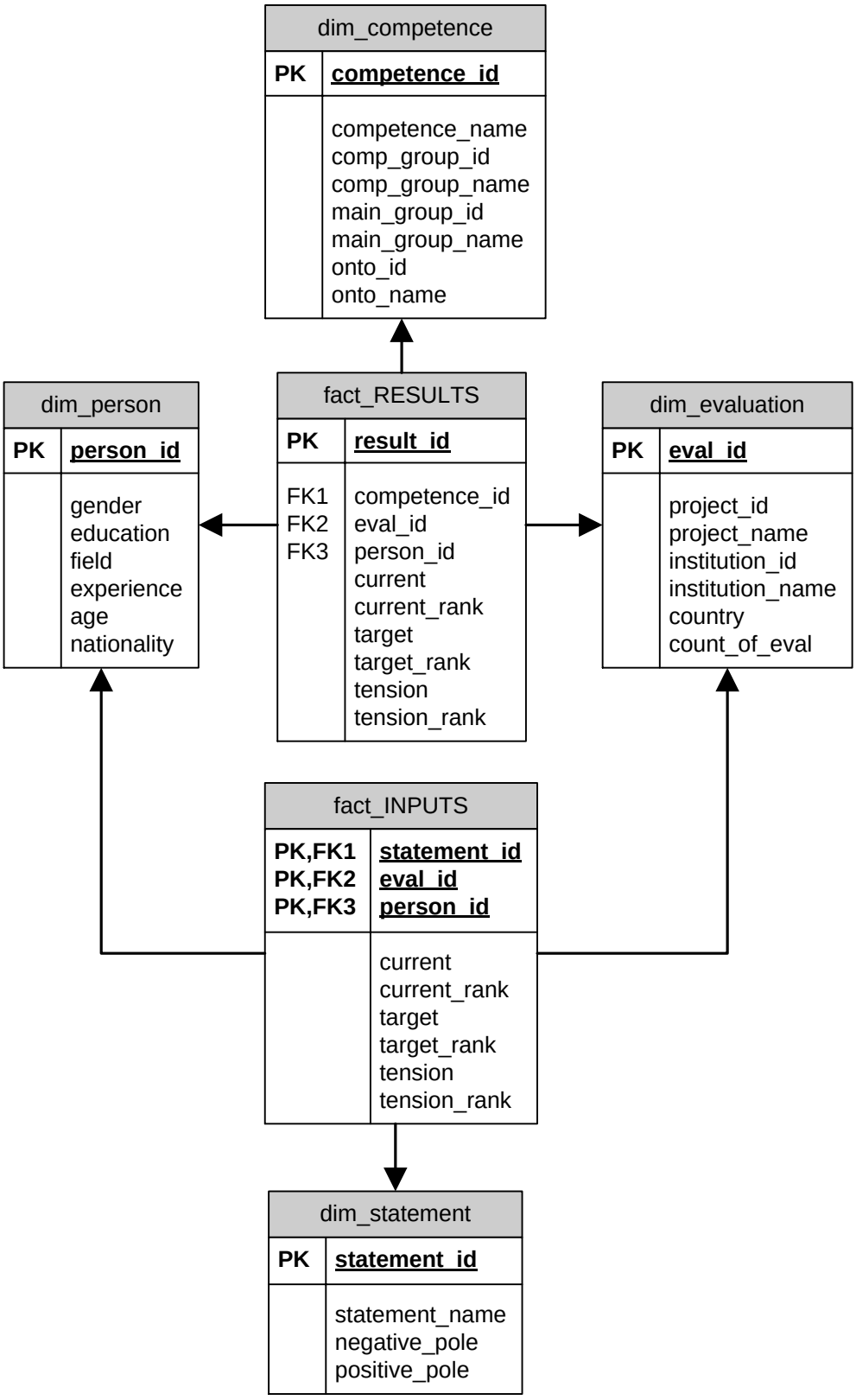
Moniulotteisen mallinnuksen toteuttamisessa käytetään Ralph Kimballin (Kimball & Ross 2013, 38) suunnittelemaa neliosaista prosessia:

1. Valitaan liiketoiminnallinen prosessi, jonka päälle malli rakentuu
2. Määritetään tietokannan yksityiskohtaisuuden taso (grain). Se ilmaisee faktataulun yksityiskohtaisuuden.
3. Määritetään dimensiot karkeuden pohjalta.
4. Määritetään faktat. Faktat vastaavat kysymykseen ”Mitä prosessissa mitataan?”. Faktojen tulee vastata aiemmin määriteltyä karkeutta.

Liiketoiminnallinen prosessi on kappaleen 2 mukaisesti asiakkaan yrityksille suuntaama kyselytutkimus. Tietokannan yksityiskohtaisuuden tasot ovat vastaukset yksittäiseen väittämään, sekä ryhmät ennalta määriteltyjen vastausten joukkojen, kompetenssien, mukaisesti. Luodaan faktataulut fact_INPUTS, joka sisältää väittämille annetut arvot sekä fact_RESULTS, johon on laskettu kompetenssien arvot. Dimensiot määritellään asiakkaalta saadun datan, sekä luvussa 2.2 määriteltyjen ryhmittelyiden ja attribuuttien hierarkioiden mukaisesti. Luodaan dimensiotaulut dim_competence, dim_evaluation, dim_statement ja dim_person, ja lisätään niihin attribuutit. Lopuksi määritellään faktat, joita ovat nykytila (current), tavoite (target) ja jännite (tension). Faktatauluihin lasketaan myös valmiiksi faktojen sijaluvut (ranks) yksittäisten vastaajien mukaiseen järjestykseen.

Asiakkaan datassa on kaksi analysoitavaa tasoa, väittämä ja kompetenssi. Koska yksi väittämä voi liittyä useampaan kompetenssiin, lasketaan kompetenssien arvot omaan tauluunsa. Tämän takia tarvitaan kaksi faktataulua. Moniulotteisen mallinnuksen toteuttamiseksi on useita eri vaihtoehtoja ja lähestymistapoja, eikä toteutukseen ole olemassa yhtä oikeaa toteutustapaa. Toteutettu tähtimalli eroaa esimerkiksi useimmista kirjallisuudessa mainituista esimerkeistä kahden faktataulunsa vuoksi.

Kuvassa 5 on tähtimallin ja edellisten kappaleiden määrittelyiden mukainen tietokantakaavio.



Kuva 5. Tietokantakaavio työssä toteutettavasta tietokannasta

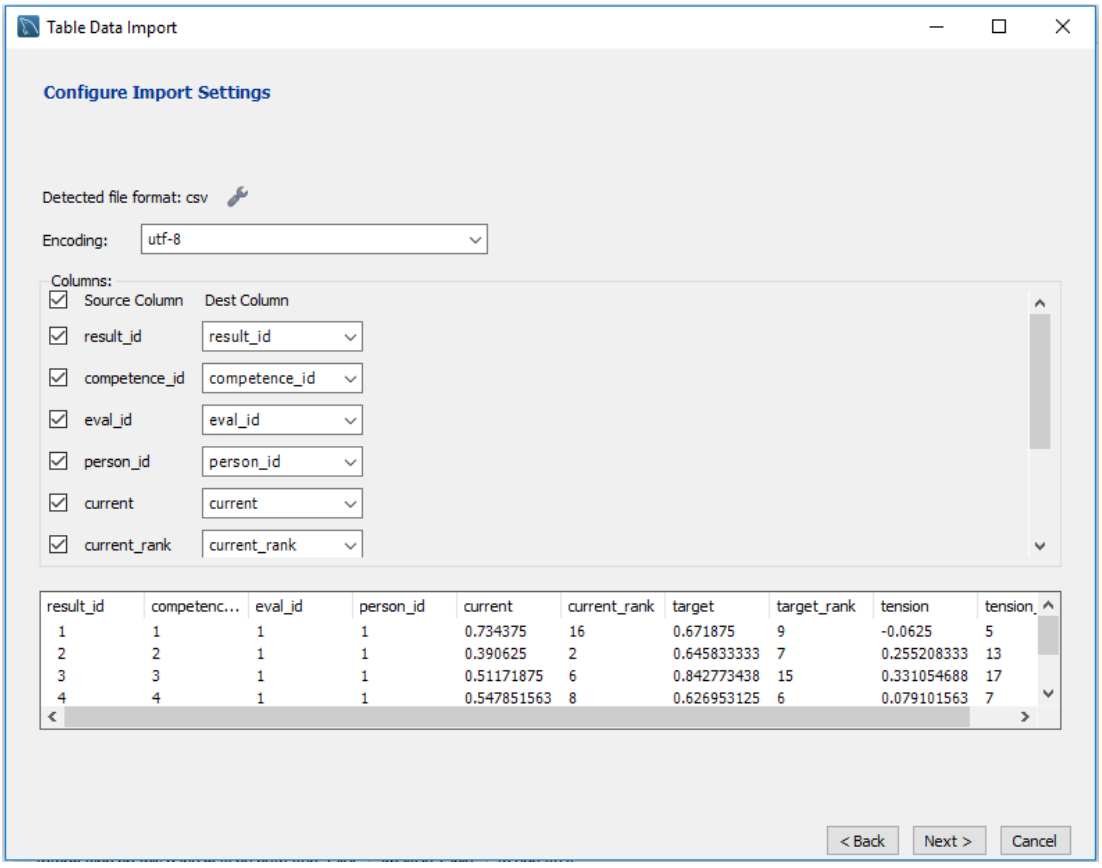
4.2 Mallin toteutus tietokannassa

Tietokannan luomiseen ja sen ylläpitoon käytetään MySQL:n käyttöohjeesta löytyvää syntaksia (MySQL, 2017a). Kuvassa 6 on esimerkkejä työssä toteutettavan tietokannan määrittelevistä SQL-lauseista. Aluksi luodaan uusi tietokanta ”tahtimalli” lauseella CREATE DATABASE. Toisella rivillä luodaan uusi käyttäjä ”user”, jonka sallitaan muodostaa yhteys mistä tahansa, ja annetaan sille salasanaksi ”pass”. Kolmannella rivillä annetaan käyttäjälle “user” kaikki oikeudet tietokannan tahtimalli tauluihin. Neljännellä rivillä valitaan äsken luotu tietokanta tahtimalli käytettäväksi USE lauseella. Riveillä 6-20 määritellään tietokantaan taulu. Taulu fact_INPUTS luodaan lauseella CREATE TABLE. Määritellään attribuuttien nimet ja tietotyypit. Lisätään perusavaimelle ja mahdollisille viiteavaimille rajoite NOT NULL, joka pakottaa ne sisältämään arvon. Määritellään perusavain lauseella PRIMARY KEY ja mahdolliset viiteavaimet lauseella FOREIGN KEY (id) REFERENCES table(id). Loput SQL-lauseet kuvan 5 mukaisen tietokannan luomiseen ovat Liittessä 1.

```
1  CREATE DATABASE tahtimalli;
2  CREATE USER 'user'@'%' IDENTIFIED BY 'pass';
3  GRANT ALL PRIVILEGES ON tahtimalli.* TO 'user'@'%';
4  USE tahtimalli;
5
6  CREATE TABLE fact_INPUTS (
7  statement_id INT NOT NULL,
8  eval_id INT NOT NULL,
9  person_id INT NOT NULL,
10 current FLOAT,
11 current_rank FLOAT,
12 target FLOAT,
13 target_rank FLOAT,
14 tension FLOAT,
15 tension_rank FLOAT,
16 PRIMARY KEY (statement_id, eval_id, person_id),
17 FOREIGN KEY (statement_id) REFERENCES dim_statement(statement_id),
18 FOREIGN KEY (eval_id) REFERENCES dim_evaluation(eval_id),
19 FOREIGN KEY (person_id) REFERENCES dim_person(person_id)
20 );
```

Kuva 6. Esimerkkejä työssä toteutettavan tietokannan SQL-lauseista

Datan siirtämiseen vanhasta tietokannasta uuteen tietokantaan käytettiin MySQL Workbenchin työkalua (kuva 7). Vanhasta tietokannasta tehtiin liitosten avulla tilapäinen taulu, jonka data kopioitiin Excelin tukemaan CSV (comma-separated values) -tiedostoon. Excelillä faktoille laskettiin sijaluvut, ja niille luotiin omat sarakkeet (current_rank, target_rank, tension_rank). CSV-tiedoston data tuotiin MySQL Workbenchin työkalulla uuteen tietokantaan (kuva 7). Työkalulla voidaan määritellä lähdetiedoston sarakkeiden vastaavuus tietokannan sarakkeisiin.



Kuva 7. Kuvankaappaus MySQL Workbenchin Table Data Import -työkalusta (MySQL Workbench 2017)

5 PENTAHO

BI ja sen ympärillä oleva markkina ovat tällä vuosituhannella kehittyneet räjähdysmäisesti, ja saatavilla on satoja eri tarkoituksiin soveltuvia ohjelmistoja. Tämän työn

tekemiseen valikoitui kuitenkin ilmainen, avoimeen lähdekoodiin perustuva Pentaho Community Edition, joka tarjosi kaikki toteutukseen tarvittavat työkalut.

Pentaho perustettiin vuonna 2004, ja sen toiminta perustuu open core -liiketoimintamalliin. Se tarjoaa ilmaisen version lisäksi myös maksullista Pentaho Enterprise Edition -versiota. Enterprise Edition sisältää joitakin lisäominaisuuksia ja tukipalvelun, joka puuttuu kokonaan ilmaisesta Community Editionista. Vuodesta 2015 lähtien Pentaho on ollut japanilaisen Hitachi Data Systemsin omistuksessa. (Pentaho 2017a.)

Pentaho on Java-pohjainen ja sen toiminnallisuus koostuu useista eri sovelluksista ja komponenteista, joista tämän työn kannalta tärkeimmistä kerrotaan enemmän seuraavassa luvussa. Pentahoa käytetään web-pohjaisen käyttöliittymän, Pentaho User Consolen kautta.

5.1 Pentahon komponentit

Pentahon rakenne ja komponenttien sisältö vaihtelee versioista riippuen, mutta ne vastaavat suurin piirtein kappaleessa 3 esitettyä BI-järjestelmän toiminnallisuutta (Kuva 2). Uusimmissa versioissa rakennetta on yksinkertaistettu, ja koko toiminnallisuus on jaettu kahteen peruskomponenttiin: Pentaho Business Analyticsiin (BA) ja Pentaho Data Integrationiin (DI). Nämä kaksi komponenttia pitävät sisällään useimmat aiemmissa versioissa erillään olleet komponentit ja ohjelmat. Jotkin ohjelmat ovat silti vielä erillisiä, kuten yksityiskohtaisten raporttien suunnitteluun käytettävä Report Designer. Pentahon komponentit ovat yhteydessä toisiinsa, joten esimerkiksi Business Analytics Platformissa määritelty tietokantayhteys toimii myös muissa komponenteissa.

Avoimen lähdekoodin ansiosta Pentahoon on tarjolla myös paljon käyttäjien tekemiä lisäosia. Lisäosia voidaan ladata BA:n Marketplace-toiminnon tai Pentahon verkkosivujen kautta. Esimerkkinä ladattavasta lisäosasta on tässäkin työssä käytettävä Saiku Analytics -ohjelmisto datan analysointiin.

Tässä työssä käyttämättömiä, mutta tärkeitä Pentaho Community Editioniin saatavia komponentteja ovat mm. Data Mining (Weka) tiedonlouhintaan ja Raport Designer raporttien suunnitteluun.

5.1.1 Business Analytics Platform

Koko Pentahon ytimenä toimii Business Analytics Platform -ohjelmisto, josta käytetään versioista riippuen myös nimiä BA-server ja BI-server. Business Analytics Platform rakentuu Java-pohjaisen Apache Tomcat -palvelimen ympärille. Business Analytics Platform sisältää muun muassa Pentaho user console -käyttöliittymän, käyttäjienvalvonnan, marketplacen, valmiit työkalut tietolähteiden määrittelyyn, sekä ohjelmat analysointiin ja raportointiin.

5.1.2 Pentaho Analysis Services (Mondrian)

Mondrian, kuten muutkin Pentahon komponentit, on avoimeen lähdekoodiin perustuva ja Java-pohjainen OLAP-palvelin. Se käyttää jaettuja OLAP-kuutioita (partitioned cubes), jotka mahdollistavat useiden faktataulujen mallinnuksen samaan kuutioon. Mondrianin toiminnallisuuden kuvaaminen voidaan esittää neljän kerroksen avulla. Päällimmäinen kerros on lähimpänä käyttäjää ja alin kerros lähimpänä dataa. Ensimmäisenä kerroksena toimii esityskerros (presentation layer). Se esittää moniulotteisen datan visualisoituna käyttäjän määrittelemällä tavalla. Toisena kerroksena on ulotteinen kerros (dimensional layer), joka jäsentää, validoi ja suorittaa MDX (multidimensional expression) -kyselyitä. MDX on OLAPin kyselykieli. Kolmas kerros on tähtikerros (star layer), joka vastaa tietokoosteiden (aggregates) säilömisestä. Tietokoosteet ovat ennalta laskettua tietoa, jonka säilyttäminen nopeuttaa kyselyiden suorittamista. Neljäs ja alin kerros on varastokerros (storage layer), joka tarkoittaa relaatiotietokannan hallintajärjestelmää. Varastokerros, eli tietokannan hallintajärjestelmä voi myös olla toisessa tietokoneessa Javan rajapinnan, JDBC:n (Java database connectivity) muodostaman yhteyden avulla. Pentahon uusimmissa versioissa Mondrian on valmiiksi liitetty Business Analytics Platformiin, eikä sitä tarvitse ladata erikseen. (Pentaho 2017b.)

5.1.3 Pentaho Data Integration (Kettle)

Pentaho Data Integration (PDI) huolehtii Penathon ETL-prosessista ja tarjoaa graafisen käyttöliittymän datan eheyttämisen ja muuttamisen helpottamiseksi. PDI:n yleisiä käyttökohteita ovat muun muassa datan siirtäminen eri tietokantojen ja sovellusten välillä, suurien tietomäärien lataaminen tietokantoihin ja tiedon puhdistaminen. (Pentaho 2017c.) Tämän työn toteutuksessa ei tarvita Data Integrationia, koska tietokannassa oleva data on jo valmiiksi oikeassa muodossa ja toteutuksessa käytetään vain yhden tietokannan sisältämää dataa.

6 TOTEUTUS PENTAHOSSA

6.1 Pentahon asennus

Pentaho asennetaan tässä vaiheessa asiakkaan paikalliselle Windows-käyttöjärjestelmää käyttävälle tietokoneelle. Pentaho voidaan asentaa myös Linux ja MacOS -ympäristöihin. Pentaho on myöhemmin mahdollista siirtää ulkoiselle palvelimelle asiakkaan niin halutessa. Asennettavat ohjelmat haettiin Pentahon verkkosivujen Community-osion kautta (Pentaho 2017d). Ladattava versio on viimeisin, Business Analytics Platform 7.0. Ladatut ZIP-pakatut tiedostot puretaan C-levyaseman juureen tehtyyn Pentaho-kansioon.

Pentaho tarvitsee toimiakseen 64-bittisen Java Runtime Environment (JRE) -ajoympäristön, jonka saa ladattua Oraclen verkkosivuilta (Oracle 2017). Tässä työssä käytetään Java SE (Standard Edition) Runtime Environment 8 -versiota, joka on vaatimus uusimmille Pentahon versioille. Pentahon toimimiseksi on Windows-ympäristössä määriteltävä ympäristömuuttujan PENTAHO_JAVA_HOME arvoksi JRE:n hakemistopolku. Windows mahdollistaa ympäristömuuttujien määrittelyn tietokoneen asetuksista graafisen käyttöliittymän avulla.

Pentahon yhteensopivuuden varmistamiseksi MySQL -ohjelmiston kanssa ladattiin MySQL:n sivuilta viimeisin JDBC-ajuri 5.1.41 (MySQL 2017b), joka mahdollistaa tietokannan toimimisen Java-pohjaisen Pentahon kanssa. JDBC-ajuri siirrettiin Pentahon pentaho-server\tomcat\lib -hakemistoon.

Asiakkaalle asennettiin myös muut Pentahon komponentit, vaikka niitä ei varsinaisesti tässä työssä käytetä. Useat komponentit, kuten Pentaho Data Integration ja Raport designer ovat kuitenkin tärkeitä työn jatkokehityksen kannalta.

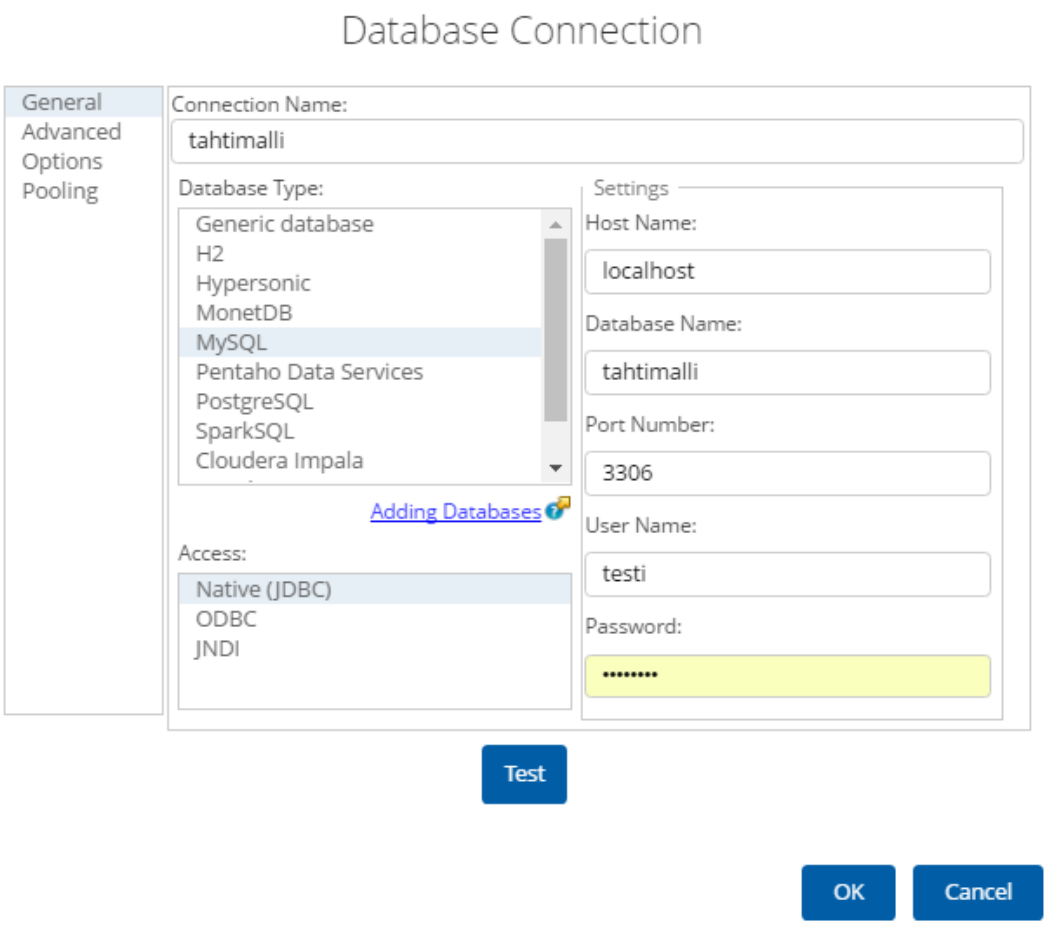
6.2 Käyttöliittymä

Pentaho käynnistetään pentaho-server -kansiossa sijaitsevalla start-pentaho.bat -skriptillä. Skripti käynnistää Tomcat-palvelimen portissa 8080. Samalla käynnistyy Tomcatin konsoli, johon lokimerkinnät kirjautuvat.

Pentahon käyttöliittymänä toimii Pentaho User Console (PUC). PUCn osoite on oletusarvoisesti localhost:8080, jossa localhost on paikallisen tietokoneen nimi ja 8080 portin numero. PUC huolehtii myös käytönvalvonnasta. Pentahon järjestelmänvalvoja voi luoda käyttäjiä ja antaa niille erilaisia käyttöoikeuksia rajoittavia rooleja.

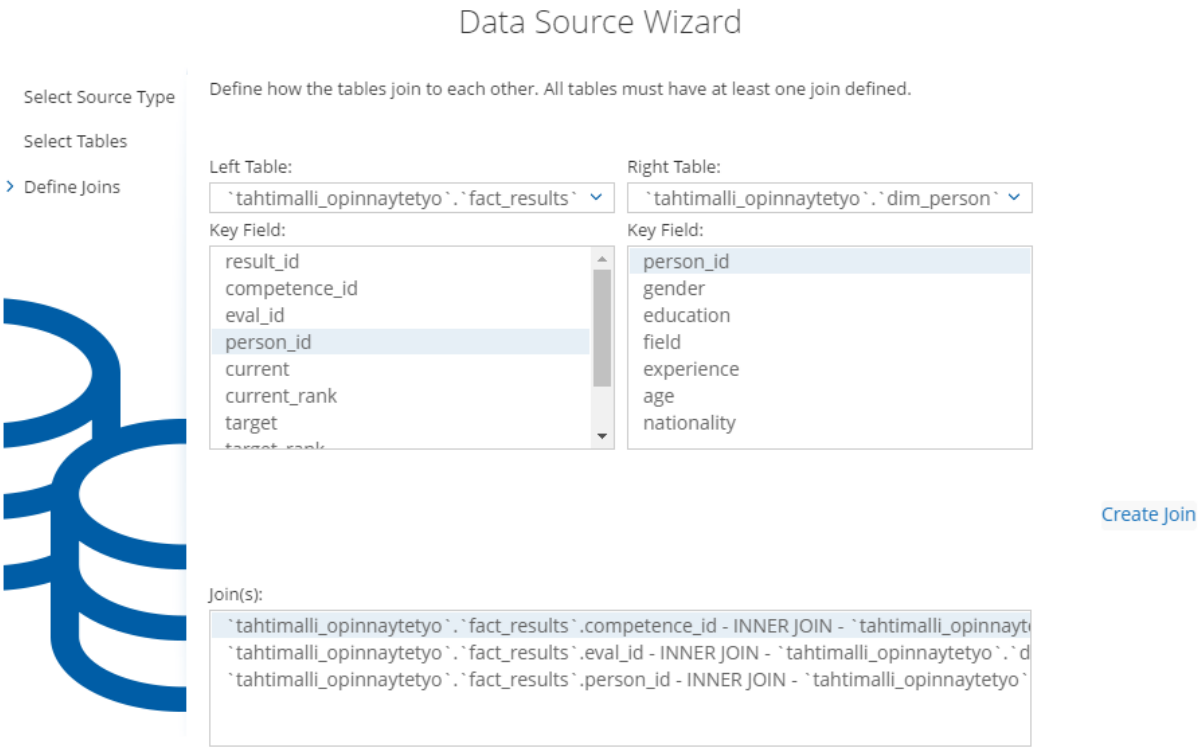
6.3 Datan tuonti

Pentaho sisältää Data Source Wizard -ohjelman (kuva 8), jonka avulla tietokannan yhdistäminen Pentahoon on helppoa. Ohjelmassa valitaan tietokannan tyyppi, käytettävä rajapinta, sekä tietokannan yhteysasetukset. Kuvassa 8 muodostetaan yhteys paikallisessa koneessa (localhost), portissa 3306 sijaitsevaan tahtimalli-nimiseen MySQL tietokantaan JDBC-rajapinnan avulla.



Kuva 8. Tietokantayhteyden määrittely Pentaho-ohjelmiston Data Source Wizardin avulla (2017e Pentaho)

Tietokantayhteyden muodostamisen jälkeen valitaan tietokannasta tuotavat taulut. Datan tuonti vain raportointia varten onnistuu ilman moniulotteisesti mallinnettua dataa, mutta datan tuonti analyysia ja raportointia varten tarvitsee tähtimallin mukaisesti mallinnetun tietokannan ja faktataulun määrittelyn toimiakseen. Faktataulun valinnan jälkeen määritellään faktataulun ja dimensiotaulujen liitokset (Kuva 9). Kuvassa yhdistetään tietokannan tahtimalli_opinnaytetyo taulun fact_results attribuutti person_id taulun dim_person vastaavaan attribuuttiin person_id.



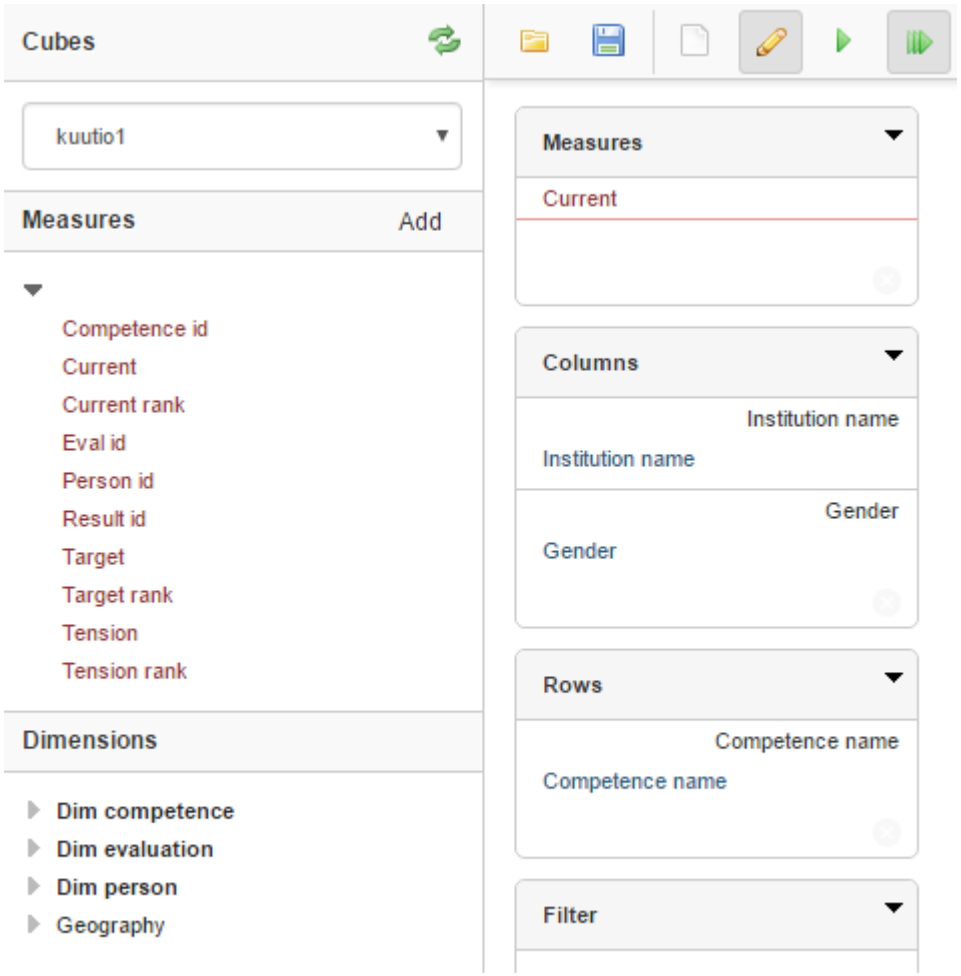
Kuva 9. Tietokannan liitosten määrittely Pentaho-ohjelmiston Data Source Wizardin avulla (2017e Pentaho)

Pentaho luo Mondrianin (OLAP-moottori) avulla tuodusta tähtimallin datasta OLAP-kuution, jolle se hakee dimensiot, dimensioiden attribuutit ja mitattavat faktat (measures) valmiiksi. Edellä mainittuja voidaan jälkikäteen muuttaa Data Source Model Editorilla. Data Source Model Ediotrilla voidaan myös määritellä mitattavien faktojen oletuskooste (default aggregate), joka on oletuksena yhteenlasku (SUM), sekä luoda uusia mitattavia faktoja määritellyistä faktataulun sarakkeesta, esimerkiksi tietyn faktan keskiarvo.

Tuodusta tietokannasta voidaan määritellä vain yksi faktataulu, ja sen mukaiset dimensiotaulut kerrallaan. Toteutuksessa käytettävästä tietokannasta täytyy siis luoda kaksi kuutiota kahden faktataulun takia.

6.4 Toteutettu toiminnallisuus

Tietokannan yhdistämisen jälkeen Pentahossa voidaan käyttää useita eri ohjelmia datan analysointiin. Tämän työn esimerkkianalyyseissä käytetään Pentahon marketplacesta saatavaa Saiku Analytics -ohjelmaa. Saiku Analytics mahdollistaa MDX-kyselyiden tekemisen Mondrianin luomaan OLAP-kuutioon ja visualisoi saadut vastaukset. Kuvassa 10 on esitetty Saiku Analyticsin käyttöliittymän toiminnallisuutta.



Kuva 10. Saiku Analytics -lisäosan toiminnallisuus Pentaho Business Analytics -ohjelmistossa (2017e Pentaho)

Kuution (Cube) valinnan jälkeen Saiku Analytics tarjoaa valmiiksi Mondrianilta haetut faktataulujen attribuutit ja mahdollisesti aiemmin luodut koosteiset faktat (measu-

res), sekä dimensiot ja dimensioiden attribuutit. Kuvan oikealla puolella näkyy käyttäjän valinnat suoritettavaan kyselyyn. Mitattavana faktana on current (oletuskoosteena yhteenlasku). Sarakkeet järjestetään ensin yrityksen nimen (institution name) ja sitten vastaajien sukupuolen (gender) mukaisesti. Riveillä esitetään kompetenssien nimet. Yrityksien nimistä valitaan vain 3 esimerkkiyrityksen tulokset (Yritys 1, Yritys 2, Yritys 3). Saiku Anlaytics luo valintojen mukaisen MDX-kyselyn (Kuva 11) ja suorittaa sen automaattisesti. Kyselyitä pystyy myös halutessaan kirjoittamaan ja muokkaamaan itse.

```
WITH
SET [~COLUMNS_Dim evaluation_Dim evaluation.Institution name] AS
    {[Dim evaluation.Institution name].[Yritys 1],
    [Dim evaluation.Institution name].[Yritys 2],
    [Dim evaluation.Institution name].[Yritys 3]}
SET [~COLUMNS_Dim person_Dim person.Gender] AS
    {[Dim person.Gender].[Gender].Members}
SET [~ROWS] AS
    {[Dim competence.Competence name].[Competence name].Members}
SELECT
NON EMPTY CrossJoin(NonEmptyCrossJoin([~COLUMNS_Dim evaluation_Dim
evaluation.Institution name],
[~COLUMNS_Dim person_Dim person.Gender]),
{[Measures].[Current]}) ON COLUMNS,
NON EMPTY [~ROWS] ON ROWS
FROM [kuutio1]
```

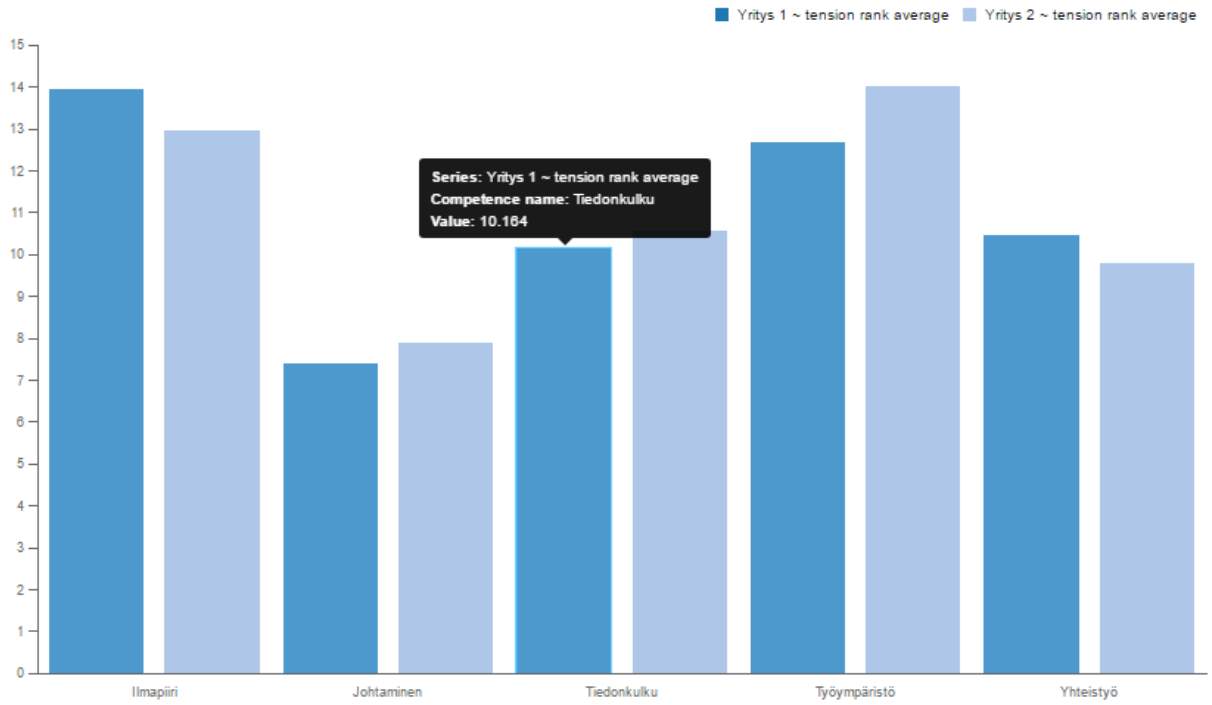
Kuva 11. Saiku Analytics -ohjelman käyttäjän valintojen pohjalta luoma MDX-kysely

Saiku Analytics palauttaa kyselyn vastaukset taulukossa (Kuva 12). Näytettävät current-faktan arvot ovat aiemmin kerrotun oletuskoosteen mukaisesti summattuja arvoja. Saiku Analytics mahdollistaa myös vastausten visualisoinnin kuvioiksi ja vastausten siirtämisen eri tiedostomuodoissa, kuten kuvina, excel-tiedostoina ja pdf-tiedostoina.

	Yritys 1		Yritys 2		Yritys 3	
	Mies	Nainen	Mies	Nainen	Mies	Nainen
Competence name	Current	Current	Current	Current	Current	Current
Ilmapiiri	56.811	6.898	39.537	9.324	22.535	4.538
Johtaminen	85.813	9.787	52.519	12.305	28.057	4.633
Organisaation avoimuus uusille ideoille	76.523	8.656	47.424	11.08	26.414	4.99
Riskienhallinta	89.279	9.498	52.961	12.887	27.802	5.114
Tekemällä oppiminen	75.697	7.727	46.84	10.513	25.224	4.657
Tiedonkulku	79.633	8.549	47.162	11.752	26.77	4.728
Tukeminen ja kannustaminen	79.516	9.266	50.909	12.014	23.889	4.119
Turvallisuusasenteet	88.957	9.285	55.128	12.891	28.52	5.43
Turvallisuuskoulutus	86.775	9.201	51.289	12.973	27.738	4.835
Turvallisuusohjeistus ja -määräykset	79.5	8.84	48.741	11.493	25.44	4.539
Turvallisuuspolitiikka	88.35	9.659	54.648	12.317	27.113	5.08
Turvallisuusresursointi	81.281	8.842	48.216	12.317	25.909	4.705
Turvallisuustietoisuus ja -vastuullisuus	92.498	9.819	56.912	13.579	30.435	5.463
Turvallisuustoimien tehokkuus	79.417	8.928	48.242	11.319	27.396	5.059
Työympäristö	70.123	8.399	38.882	8.792	21.553	3.52
Uuden tiedon luominen	72.646	7.698	43.691	11.009	23.224	4.194
Yhteistyö	73.694	8.155	47.54	10.652	26.829	4.558

Kuva 12. Kuvankaappaus Saiku Analytics -ohjelman luomasta taulukosta, jossa on esitettynä MDX-kyselyn vastaukset (2017e Pentaho)

Saiku Analytics -ohjelmassa on valmiina useita kaavioita, joilla kyselyillä saatua dataa voidaan mallintaa. Kaaviot ovat interaktiivisia mahdollistaen muun muassa hierarkialta toiseen etenemisen kuviota klikkaamalla. Kuvassa 13 on kuvankaappaus interaktiivisesta kaaviosta. Käyttäjän viedessä kursorin pylvään päälle, ilmestyy musta ikkuna, josta käyttäjä näkee lisäinformaatiota kuten pylvään tarkan arvon. Kuvan pylväsdiagrammit esittävät kahden yrityksen (Yritys 1 ja Yritys 2) viittä kompetenssia (ilmapiiri, johtaminen, tiedonkulku, työympäristö ja yhteistyö). Vertailtavana ovat luovan jännitteen sijalukujen keskiarvot.



Kuva 13. Kuvankaappaus Saiku Analytics -ohjelman luomasta, interaktiivisesta pylväsdiagrammista (2017e Pentaho)

7 LOPPUSANAT

Opinnäytetyö on ollut sisällöltään erittäin monipuolinen ja siinä on käytetty useita erilaisia malleja, alustoja ja tekniikoita. Opinnäytetyön tekeminen oli laaja, mutta palkitseva oppimisprosessi. Työn aihe oli tietokantoja lukuun ottamatta koulussa opitun kokonaisuuden ulkopuolella. Esimerkiksi kaikki Business Intelligenceen liittyvät teoriat ja tekniikat oli opeteltava kokonaan itsenäisesti. Opinnäytetyön aiheista oli onneksi olemassa runsaasti kirjallisuutta. Monipuolinen lähdekirjallisuus helpotti huomattavasti työn toteutusta. Kaikesta huolimatta aikaa kului paljon niin toteutuksen useisiin yksityiskohtiin kuin myös teorian opiskeluunkin.

Kokonaisuutena opinnäytetyö on laajentanut tietämystäni ja ammatillista osaamistani merkittävästi varsinkin Business Intelligencen ja tietovarastojen osalta.

7.1 Tavoitteiden toteutuminen

Työn tavoitteena oli mahdollistaa kyselytutkimuksella kerätyn datan monipuolinen analysointi BI-järjestelmän avulla. Tavoitteeseen nähden työ onnistui hyvin ja myös asiakas oli tyytyväinen työn lopputulokseen. Asiakkaan datan mallinnus moniulotteiseksi tähtimallin avulla mahdollisti kappaleessa 2.2 esitettyjen asiakkaan vaatimusten toteutumisen ryhmittelyjen, ja niiden välisten vertailujen osalta.

Toteutuksen valmistumisen jälkeen asiakkaalla on käytössään toimiva BI-järjestelmä, Pentaho, ja siihen liitetty tietokanta. Pentaho mahdollistaa datan analysoinnin usealla eri tavalla, kuten helppokäyttöistä graafista käyttöliittymää käyttävällä Saiku Analytics -ohjelmalla. Asiakas voi analysoida dataansa kaikkien tietokannassa olevien dimensiotaulujen attribuuttien mukaisesti.

7.2 Parannus- ja kehitysehdotuksia

Työn tarkoituksena oli toteuttaa BI-järjestelmän avulla ympäristö, jossa asiakas voi analysoida kyselytutkimuksella keräämäänsä dataa. Työn valmistuttua kyseisenlainen ympäristö toteutettiin onnistuneesti. BI-järjestelmä mahdollistaa työn jatkokehityksen usealla eri alueella. Työssä esitetyn Pentaho Business Analytics Platformin lisäksi asiakkaalle asennettiin myös muita Pentahon komponentteja, kuten Pentaho Data Integration ja Raport designer, jotka ovat tärkeitä työn jatkokehityksen kannalta.

Pentaho Data Integration (PDI) mahdollistaa ETL-prosessit, joiden avulla asiakas voi jatkossa tuoda datan automaattisesti suoraan kyselyistä. PDIn avulla on myös mahdollista laskea faktataulujen attribuuttien sijaluvut automaattisesti.

Asiakkaan tavoitteena on tehdä datasta syvempiä analyysejä. Pentaho mahdollistaa asiakkaan oman koodin ja funktioiden käyttämisen, joiden avulla on mahdollista toteuttaa analyysejä, joita ei ole saatavilla valmiina analyysiohjelmistoissa. Myös Pentahoon saatava Weka-lisäosa tiedonlouhintaan tarjoaa mahdollisuuden tarkempaan analyysiin.

Raport designer mahdollistaa lopullisten analyysien julkaisemisen raporttien muodossa. Raportteihin voidaan myös upottaa asiakkaan toivomusten mukaan kustomoituja graafeja.

LÄHTEET

Codd, E. F. Codd, S. B. Salley, C. T. 1993. Providing OLAP to User-Analysts: An IT Mandate. E. F. Codd & Associates

Inmon, W. H. 2002. Building the Data Warehouse. Third Edition. Hoboken: John Wiley & Sons.

Kimball, R. Ross, M. 2013. The Data Warehouse Toolkit. Third Edition. Hoboken: John Wiley & Sons.

Krmac, E. V. 2011. Intelligent Value Chain Networks: Business Intelligence and Other ICT Tools and Technologies in Supply/Demand Chains. Viitattu 10.3.2017. http://www.intechopen.com/books/supply-chain-management-new-perspectives/intelligent-value-chain-networks-business-intelligence-and-other-ict-tools-and-technologies-in-suppl

Loshin, D. 2003. Business Intelligence: The Savvy Manager's Guide. Burlington: Morgan Kaufmnann Publishers. Viitattu 9.3.2017. http://site.ebrary.com.lillukka.samk.fi/lib/SAMK/detail.action?docID=10206776

MySQL Workbench CE (Version 6.3). 2017. Redwood City: Oracle Corporation.

MySQL. 2017a. MySQL 5.7 Reference Manual. Viitattu 3.5.2017. https://dev.mysql.com/doc/refman/5.7/en/

MySQL. 2017b. Download Connector/J. Viitattu 22.4.2017. https://dev.mysql.com/downloads/connector/j/5.1.html

Olap Council. 1997. OLAP AND OLAP Server Definitions. Viitattu 27.3.2017. http://olapcouncil.org/research/resrchly.htm

Oracle. 2017. Java SE Runtime Environment 8 Downloads. Viitattu 22.4.2017. http://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html

Pentaho. 2017a. About Us. Viitattu 1.4.2017. http://www.pentaho.com/about

Pentaho. 2017b. Mondrian. Viitattu 5.4.2017. http://community.pentaho.com/projects/mondrian/

Pentaho. 2017c. Data Integration – Kettle. Viitattu 4.4.2017. http://community.pentaho.com/projects/data-integration/

Pentaho. 2017d. Pentaho Community Edition 7.0. Viitattu 12.5.2017. http://community.pentaho.com/

Pentaho. 2017e. Pentaho Business Analytics Platform 7.0. Viitattu 15.5.2017. Orlando: Pentaho Corporation.

Ponniah, P. 2010. Data Warehousing Fundamentals for IT Professionals. Second Edition. Hoboken: John Wiley & Sons.

Porkka, P. 2012. Harjavallan teollisuuspuiston Turvallisuuskulttuurin kartoitus Pro-Turva projektin loppuraportti. https://www.tsr.fi/documents/20181/40645/110071-loppuraportti-Loppuraportti+110071.pdf

Porkka, P. 2017. Henkilökohtainen tiedonanto. Pori. 14.2.2017.

Senge, P. 1990. The Fifth Discipline. New York: Doubleday/Currency.

```
CREATE database tahtimalli;
USE tahtimalli;

CREATE USER 'testikayttaja'@'%' IDENTIFIED BY 'salasana';
GRANT ALL PRIVILEGES ON tahtimalli.* TO 'testikayttaja'@'%';


CREATE TABLE dim_competence (
competence_id INT NOT NULL,
competence_name VARCHAR(255),
comp_group_id INT,
comp_group_name VARCHAR(255),
main_group_id INT,
main_group_name VARCHAR(255),
onto_id INT,
onto_name VARCHAR(255),
PRIMARY KEY (competence_id)
);


CREATE TABLE dim_person (
person_id INT NOT NULL,
gender VARCHAR(255),
education VARCHAR(255),
field VARCHAR(255),
experience VARCHAR(255),
age INT,
nationality VARCHAR(255),
PRIMARY KEY (person_id)
);


CREATE TABLE dim_evaluation (
eval_id INT NOT NULL,
project_id INT,
project_name VARCHAR(255),
institution_id INT,
institution_name VARCHAR(255),
country VARCHAR(255),
count_of_eval INT,
PRIMARY KEY (eval_id)
);


CREATE TABLE dim_statement (
statement_id INT NOT NULL,
statement_name VARCHAR(255),
negative_pole VARCHAR(255),
positive_pole VARCHAR(255),
PRIMARY KEY (statement_id)
);


CREATE TABLE fact_RESULTS (
result_id INT NOT NULL,
competence_id INT NOT NULL,
eval_id INT NOT NULL,
person_id INT NOT NULL,
current FLOAT,
```

```
current_rank FLOAT,
target FLOAT,
target_rank FLOAT,
tension FLOAT,
tension_rank FLOAT,
PRIMARY KEY (result_id),
FOREIGN KEY (competence_id) REFERENCES dim_competence(competence_id),
FOREIGN KEY (eval_id) REFERENCES dim_evaluation(eval_id),
FOREIGN KEY (person_id) REFERENCES dim_person(person_id)
);


CREATE TABLE fact_INPUTS (
statement_id INT NOT NULL,
eval_id INT NOT NULL,
person_id INT NOT NULL,
current FLOAT,
current_rank FLOAT,
target FLOAT,
target_rank FLOAT,
tension FLOAT,
tension_rank FLOAT,
PRIMARY KEY (statement_id, eval_id, person_id),
FOREIGN KEY (statement_id) REFERENCES dim_statement(statement_id),
FOREIGN KEY (eval_id) REFERENCES dim_evaluation(eval_id),
FOREIGN KEY (person_id) REFERENCES dim_person(person_id)
);
```