TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

MATILDA HIETALA

RAPORTOINTITYÖKALUJEN HYÖDYNTÄMINEN NOSQL-TIETOKANNAN TESTAUKSESSA

Diplomityö

Tarkastajat: professori Samuli Pekkola ja professori Kari Systä
Tarkastajat ja aihe hyväksytty
Talouden ja rakentamisen tiedekuntaneuvoston kokouksessa
4. marraskuuta 2015

TIIVISTELMÄ

MATILDA HIETALA: Raportointityökalujen hyödyntäminen NoSQL-tietokannan testauksessa
Tampereen teknillinen yliopisto
Diplomityö, 86 sivua, 2 liitesivua
Maaliskuu 2016
Tietojohtamisen diplomi-insinöörin tutkinto-ohjelma
Pääaine: Tiedonhallinta
Tarkastajat: professori Samuli Pekkola, professori Kari Systä

Avainsanat: raportointityökalu, NoSQL-tietokanta, testaus, testaussuunnitelma

Liiketoimintatiedon hallinnan prosessissa dataa tallennetaan tietokantaan, josta sitä voidaan hyödyntää liiketoimintatiedon hallinnan työkalujen, kuten raportointityökalujen, avulla. Raportointityökaluilla tietokannan datasta voidaan muodostaa raporttimuotoisia analyyseja, joita voidaan käyttää päätöksenteon tukena. Jotta päätöksenteko perustuu luotettavaan tietoon, on tärkeää että tietokannassa sijaitseva data on oikeellista. Tietokannan datan oikeellisuudesta voidaan varmistua testaamalla sitä. Loppukäyttäjä on hyvä testaaja datan oikeellisuudelle, sillä hän tietää mitä datalta odotetaan ja lisäksi hän on tietokannan kehityksen kannalta ulkopuolinen tarkastelija.

Tietokannat ovat perinteisesti perustuneet relaatiomalliin, mutta datan määrän ja moninaisuuden jatkuva lisääntyminen ovat myös luoneet tarpeen hyvin skaalautuville ja joustaville NoSQL-tietokannoille. NoSQL-tietokantojen rakenne on vaihteleva, eikä niille ole yhtä yksikäsitteistä kyselykieltä, joten datan tarkastelu voi olla haasteellista. Liiketoimintaorientoitunut loppukäyttäjä voi kuitenkin yhdistää NoSQL-tietokannan raportointityökaluun ja tarkastella dataa työkalun avulla. Tämä mahdollistaa datan oikeellisuuden tarkastelun ilman ymmärrystä tietokantarakenteesta.

Tämän tutkimuksen tavoitteena oli muodostaa käsitys raportointityökalujen hyödyntämismahdollisuuksista NoSQL-tietokannan datan testaamisessa. Kirjallisuustarkastelun avulla keskityttiin ensin tarkastelemaan tutkimuksen kontekstia ja kasvatettiin ymmärrystä kontekstin osa-alueista. Teoreettisen tarkastelun pohjalta muodostettiin testaussuunnitelma, jonka pohjalta suoritettiin empiirinen tutkimus. Empirian tarkoituksena oli selvittää, kuinka hyvin virheet datassa näkyvät raportoinnissa ja mikä raportointityökalu sopisi parhaiten tämän tyyppiseen testaukseen. Testaussuunnitelma toteutettiin kolmella eri raportointityökalulla ja tuloksia vertailtiin helppokäyttöisyyden, loogisuuden ja graafisen esitysvoiman perusteella.

Empiirinen tutkimus osoitti, että NoSQL-tietokannan datan oikeellisuuden testaaminen on mahdollista raportointityökalujen avulla. Kaikki testausaineistoon luodut tahalliset virheet ilmenivät raportointityökaluilla suoritetussa korkean tason tarkastelussa ja graafiset esitystavat toivat lisäarvoa dataan. Raportointityökalut olivat myös helppokäyttöisiä testaustyökaluja, ja tarkastelluista kolmesta raportointityökalusta Tableau ja Power BI olivat selvästi helppokäyttöisempiä kuin Qlik Sense.

ABSTRACT

MATILDA HIETALA: Use of reporting tools in testing NoSQL database
Tampere University of Technology
Master of Science Thesis, 86 pages, 2 Appendix pages
March 2016
Master's Degree Programme in Information and Knowledge Management
Major: Business Information Management
Examiners: Professor Samuli Pekkola, Professor Kari Systä

Keywords: reporting tool, NoSQL database, testing, test plan

In business intelligence process data is being saved into a database, from which it can be utilized using business intelligence tools such as reporting tools. With the help of reporting tools database data can be analysed and presented as reports, which can be then used in supporting decision making. For the decision making to be based on trusted information, it is important that the data in database is correct. The correctness of the data can be verified by testing it. End user is a good tester for data correctness, since he knows what is expected of the data. Also, in the sense of developing the database, the end user is an external observer.

Databases are traditionally based on a relational model, but the increase in data volume and variety have created a need for well-scalable and flexible NoSQL databases. The structure of NoSQL databases is varying and they lack mutual query language, so data examination might be challenging. It is possible for a business-oriented end user to connect NoSQL database to a reporting tool, and examine the data with the help of the reporting tool. This enables end user to examine correctness of the database data, without understanding of the database structure.

The aim of the study was to form an assessment about the usefulness of reporting tools in testing NoSQL database data. In the literature review, the focus was to examine the context of the study and increase knowledge in different areas of the context. Based on theoretical examination a test plan was constructed, based on which the empirical study was executed. The aim of the empirical study was to clarify how well data errors were visible in reporting, and which reporting tool was most suitable for this kind of use. The test plan was executed with three different reporting tools and results were compared based on graphical output, how easy it was to use the tool and how logical the tool was.

The empirical study showed that it is possible to test NoSQL database data correctness with reporting tools. All data errors created in test material were visible in high level examination and graphical outputs brought added value to the data. Reporting tools were easy to use as testing tools. From the examined three tools Tableau and Power BI outperformed Qlik Sense.

ALKUSANAT

Diplomityö ja tutkinto tulivat vihdoin valmiiksi, enkä voisi olla iloisempi. Näiden asioiden lisäksi elämässä on kaikki muutenkin kohdillaan, joten voin varauksetta hymyillä joka päivä. Opiskeluaika on ollut erittäin antoisaa ja opettavaa kaikin puolin, mutta tyytyväisenä siirryn nyt myös ”aikuiseksi”.

Diplomityöprosessi ei ollut kaikkein helpoin, mutta olen ylpeä, että sain sen maaliin. Diplomityön ohjaamisesta ja arvokkaista kommenteista haluan kiittää professori Samuli Pekkolaa ja professori Kari Systää.

Haluan kiittää ystäviäni tuesta ja ymmärryksestä ja erityisesti Lauraa, jolla oli myös todellista kontribuutiota työhön. Erityisesti haluan kiittää perhettäni korvaamattomasta tuesta ja rakkaudesta – ilman teitä koko opiskeluaika olisi ollut tukahduttavaa. Ja viimeisenä muttei vähäisimpänä: kiitos rakas Aki, että olet elämässäni ja näät kanssani tulevaisuuteen.

Tampereella, 11.3.2016

Matilda Hietala

SISÄLLYSLUETTELO

LYHENTEET JA MERKINNÄT

Big data	Valtavaa datan määrää ja moninaisuutta, sekä niiden nopeaa kasvua kuvaava termi.
BSON	*Binary JSON*. Binäärinen JSON-tiedostomuoto.
Drill	Avoimen lähdekoodin kyselymoottori Hadoop viitekehyksessä.
ETL	*Extract - Transfer - Load*. Prosessi, jossa data haetaan, muokataan ja ladataan lähdejärjestelmistä eteenpäin tietovarastointiratkaisuihin
Hadoop	Avoimen lähdekoodin viitekehys, joka tarjoaa puitteet suurien datamäärien varastointiin ja käsittelyyn.
HBase	Hadoop viitekehykseen kuuluva NoSQL-tietokanta.
HDFS	*Hadoop Distributed File System*. Hajautettu Hadoop tiedostojärjestelmä.
JSON	*Javascript Option Notation*. Avoimen standardin tiedostomuoto tiedonvälitykseen.
NoSQL	*Not Only SQL*. Käsite, jolla kuvataan relaatiomallista poikkeavia tietokantoja.
ODBC	*Open Database Connectivity*. Standardoitu avoin rajapinta tietokannoille, jonka avulla sovellukset voivat kommunikoida tietokantapalvelimen kanssa.
Power BI	Microsoftin visualisointi- ja raportointityökalu.
SQL	*Structured Query Language*. Tietokantojen standardoitu kyselykieli.
Tableau	Tableau Softwaren visualisointi- ja raportointityökalu.
Qlik Sense	QlikTechin visualisointi- ja raportointityökalu.

1. JOHDANTO

Tutkimuksen taustalla on kirjoittajan mielenkiinto aihetta ja siihen liittyviä ratkaisuja kohtaan. NoSQL-tietokantojen testaus on verrattain uusi aihealue ja lisäksi raportointityökalujen yhdistäminen testaukseen on uusi ja kiinnostava lähetymistapa. Ensimmäisessä luvussa on esitetty tutkimuksen taustaa ja tavoitteet sekä tutkimusmenetelmät.

1.1 Tutkimuksen tausta

Jatkuvasti muuttuva ja kehittyvä toimintaympäristö ohjaa toimintaamme kohti digitalisaatiota. Digitaalinen maailma ja sen kompleksisuus kasvavat jatkuvasti kiihtyvällä nopeudella. Kompleksisuus kasvaa datan jatkuvasti moninkertaistuvan määrän, moninaisuuden ja lisääntymisnopeuden vuoksi (Moniruzzaman & Hossain 2013; Salo 2013, s. 24). Tätä ilmiötä nimitetään yleisesti big dataksi, jolla siis tarkoitetaan suurten, järjestelemättömien ja jatkuvasti lisääntyvien datamassojen hallintaa (Hurwitz et al. 2013, s. 14). Big data haastaa perinteisiä tietokantaratkaisuja, sillä se vaatii tietokannalta tehokasta skaalautuvuutta, jatkuvaa saatavuutta, joustavia datamalleja sekä edullista datan varastointitilaa. Suorituskyvyltään tehokkaat NoSQL-tietokannat ovat syntyneet vastaamaan tähän big data -haasteeseen. (Altrafi et al. 2014)

NoSQL-tietokantoja hyödynnetään pääasiassa suurten datamassojen analysointi- ja varastointitarpeiden täyttämiseen (Altrafi et al. 2014). Tietokannan käyttöönoton takana on kehitys- ja käyttöönottoprojekti, jonka tavoitteena on tuottaa analysointia ja varastointia tukeva tietokantaratkaisu. Kuten mikä tahansa kehitysprojekti, vaatii NoSQL-tietokannan käyttöönotto ja kehitys testausta. Testauksen myötä voidaan varmistua ratkaisun oikeellisuudesta ja siitä, että toteutus täyttää kaikki määritellyt tavoitteet (Andreou & Sofokleous 2008). Testauksen avulla pyritään kasvattamaan ratkaisun luotettavuutta löytämällä ja poistamalla virheitä (Myers et al. 2011). On tärkeää, että testausta tekee kehittäjän lisäksi myös loppukäyttäjää edustava osapuoli, sillä kehittäjän on vaikea havaita omia virheitään. Ulkopuolisen tarkastelijan on helpompi huomata virheitä, epäjohdonmukaisuuksia tai epäselvyyksiä ratkaisuissa, ja näin testauksen tehokkuus ja luotettavuus kasvavat. (Hambling et al. 2010, s. 135)

Tietokannan dataa hyödynnetään liiketoimintatiedon hallinnan järjestelmissä, joiden avulla data voidaan nostaa mielekkäässä muodossa päätöksentekijöiden hyödynnettäväksi (Hovi et al. 2009, s. 74). Kun tietokannan dataa käytetään päätöksentekoprosessissa ja liiketoiminnan ohjaamisessa, on loppukäyttäjälle tärkeintä, että tietokannan datan oikeellisuus on tarkistettu, eli että tietokanta on eheä ja sen dataan

voidaan luottaa (Hovi et al. 2009, ss. 167-168). Datan oikeellisuus voidaan tarkistaa testaamalla, ja tässä loppukäyttäjä on hyvä testaaja sillä hän tietää, miltä datan kuuluisi näyttää ja voi verrata tietokannan dataa alkuperäiseen datajoukkoon (Collins 2008). Loppukäyttäjä on yleensä liiketoimintaorientoitunut käyttäjä, joka tarkastelee lukuja raporteilta, mutta ei ymmärrä tietokantoja tai niiden rakenteita ja kyselykieliä (Hovi et al. 2009, s. 166). Kyselykieli mahdollistaa tietokannan datan käsittelyn ja kyselykielen hallitsevat voivat tehdä tietokannan datan testausta kyselykielen avulla. NoSQL-tietokantojen rakenteet ovat kuitenkin relaatiotietokantoja haastavampia ymmärtää ja niille ei ole yhtä yhteistä kyselykieltä (Pokorny 2013), joten erityisesti NoSQL-tietokantojen tapauksessa tulisi miettiä vaihtoehtoista tapaa, jolla liiketoimintaorientoitunut loppukäyttäjä voisi tarkastella tietokannan dataa ja sen oikeellisuutta.

Tietokannat ovat paikkoja tallentaa dataa, jota voidaan hyödyntää jatkossa liiketoimintatiedon hallinnan prosessin avulla. Liiketoimintatiedon hallinnan järjestelmien avulla tietokannan dataa voidaan analysoida ja jalostaa informaatioksi. Analysoitu data esitetään usein raportteina, joiden perusteella liiketoimintaorientoituneet päättäjät voivat tehdä järkeviä, tietoon perustuvia päätöksiä. (Hovi et al. 2009, s. 74) Liiketoimintaorientoitunut loppukäyttäjä tulee siis tarkastelemaan valmiin tietokannan dataa raporteilta, jotka on rakennettu raportointityökalujen avulla tietokannan datan päälle. Raportit ovat yleensä lopputulos, mutta miksei loppukäyttäjä voisi tarkastella raportointityökalujen avulla tietokannan dataa jo kehitys- ja testausvaiheessa. Jos käyttäjän saama raportti on oikein, niin todennäköisesti järjestelmä sen taustalla toimii oikein (Hovi et al. 2009, s. 172). Tämän ajatuksen pohjalta loppukäyttäjä voi hyödyntää raportointityökaluja NoSQL-tietokannan datan testaamisessa.

Raportointityökalut auttavat käyttäjää pääsemään käsiksi dataan ja luomaan, hallitsemaan ja ajamaan päätöksenteko-objekteja, kuten kyselyitä, analyyseja ja visualisaatioita (Thierauf 2001, s. 69). Raportointityökalut mahdollistavat esimerkiksi datan summauksen, ryhmittelyn ja järjestämisen, joten datan oikeellisuutta voidaan tarkastella tätä kautta ja muodostaa esimerkiksi summia, joita voidaan vertailla odotettuihin tuloksiin. Datan käsittelyn lisäksi raportointityökalut tarjoavat mahdollisuuden visualisoida dataa, joten ne tuovat välitöntä lisäarvoa dataan ja helpottavat tulosten tarkastelua.

Tässä tutkimuksessa tausta-ajatuksena on, että liiketoimintaorientoituneen loppukäyttäjän tulisi testata NoSQL-tietokannan datan oikeellisuutta mahdollisimman tehokkaasti ja helpoilla menetelmillä. Tätä varten raportointityökaluja ymmärtävä loppukäyttäjä voi yhdistää raportointityökalun tietokannan dataan ja tarkastella raportointityökalujen tarjoamien ominaisuuksien avulla, onko tietokannan data oikeellista eli vastaako se odotettuja tuloksia. Tutkimus ottaa kantaa raportointityökalujen yleiseen soveltuvuuteen NoSQL-tietokannan datan testauksessa, minkä lisäksi tarkastellaan ja vertaillaan kolmea eri raportointityökalua tässä tarkoituksessa.

1.2 Tutkimuskysymykset ja tavoitteet

Tutkimuksen tavoite ja tutkimusongelma on esitetty tutkimuskysymyksinä:

- Miten raportointityökaluja voidaan hyödyntää NoSQL-tietokannan datan testauksessa?
- Miten eri raportointityökalut toimivat testausnäkökulmasta?

Tutkimuksen tavoitteena on muodostaa käsitys raportointityökalujen hyödyntämismahdollisuuksista NSQL-tietokannan datan testauksessa. Tutkimuksen tarkoituksena on selvittää, kuinka hyvin virheet tietokannan datassa näkyvät raportointityökaluilla ja mikä raportointityökalu sopii parhaiten tämän tyyppiseen testaukseen.

Tutkimus ottaa kantaa raportointityökalujen yleiseen soveltuvuuteen NoSQL-tietokannan datan testauksessa, minkä lisäksi tarkastellaan ja vertaillaan kolmea eri raportointityökalua tässä tarkoituksessa. Tarkasteltavat raportointityökalut ovat Tableau, Power BI ja Qlik Sense. Työkalujen vertailun avulla on tarkoitus nostaa esiin, mikä/mitkä työkaluista soveltuvat parhaiten liiketoimintaorientoituneen loppukäyttäjän testaustyökaluksi.

Tutkimuksen lähtökohtana ja odotettuna tuloksena on, että raportointityökalujen hyödyntäminen on kannattavaa NoSQL-tietokannan datan oikeellisuuden testauksessa. Menetelmän avulla liiketoimintaorientoitunut loppukäyttäjä voi varmistaa datan oikeellisuuden ja testit ovat aina helposti toistettavissa ja dokumentoitavissa, ilman että testaajan tarvitsee ymmärtää tietokannan rakennetta. Raportointityökaluissa ei odoteta olevan suuria eroja.

1.3 Tieteenkäsitys ja tutkimusote

Tutkimusotteella tarkoitetaan tutkimuksen menetelmällisten ratkaisujen kokonaisuutta, josta voidaan erottaa suppeampana käsitteenä tutkimusmenetelmä (Kasanen et al. 1991, s. 313; Hirsjärvi et al. 2007, s. 128). Tutkimusote ja tieteenkäsitykset voidaan nähdä tapana hankkia ja käyttää tietoa, kun tutkimusmenetelmä kuvaa tutkimusotteen toteuttamista käytännössä (Olkkonen 1993, ss. 64-65; Hirsjärvi et al. 2007, s. 128).

Tieteenkäsityksellä tarkoitetaan eri aikoina vallinneiden käsitysten, tiedettä tutkineiden filosofien ja eri tieteenalojen perinteiden ja tavoitteiden muodostamaa käsitystä tieteenalasta (Olkkonen 1993, s. 26). Merkittävimpiä ja yleisimpiä tutkimuksia ohjaavia tieteenkäsityksiä ovat positivismi ja hermeneutiikka (Olkkonen 1993, s. 26; Gummesson 2000, ss. 176-177). Positivismi nojautuu todettuihin tosiasioihin ja perusajatuksena on, että tutkimuksen tulee olla tutkijasta riippumaton ja toistettavissa. Positivismissa ongelma on strukturoitavissa ja lähdeaineistona pidetään aiempaa teoreettista tietoa.

Hermeneuttisessa tieteenkäsityksessä puolestaan teoriataustaa ei välttämättä ole ja tutkijan subjektiivisuus on lähes välttämätöntä, ja siksi voi olla vaikea erottaa, perustuuko päättely tieteelliseen tietoon vai tutkijan arvoihin ja näin ollen tulos on aina ehdollinen selitys. (Olkkonen 1993, ss. 35-37; Gummesson 2000, ss. 176-178)

Tähän tutkimukseen liittyy hermeneuttisia piirteitä, sillä testattava aineisto on suppea ja sen käsittely vaatii tutkijan innovatiivista päättelyä. Ehdollisuutta lisää myös ainoastaan yhden tapauksen tarkastelu tutkimuksen empiirisessä osassa. Lisäksi on huomioitava, että tutkija toimii työkseen NoSQL-tietokannan testaajana, joten aihepiiri on tutkijalle tuttu ja käytännönkokemus ohjaa tutkimusta ilman, että tutkija välttämättä huomaa subjektiivisuuttaan. Subjektiivisuutta vähentää kuitenkin ulkopuolisten koehenkilöiden suorittama testaus empiriaosuudessa. Tutkimuksen pohjana on teoriatausta, joten sitä kautta tutkimuksen toistettavuus on hyvä.

Tutkimuksia voidaan lähestyä myös luokittelemalla ne kvantitatiiviseen ja kvalitatiiviseen, eli määrälliseen ja laadulliseen tutkimukseen (Creswell 2003, s. 13). Kvantitatiiviselle tutkimukselle on keskeistä, että johtopäätöksiä voidaan tehdä aiemmista tutkimuksesta ja teoriasta ja, että aineisto on saatettavissa tilastollisesti käsiteltävään muotoon (Hirsjärvi et al. 2007, s. 136). Kvalitatiivisessa tutkimuksessa lähtökohtana on todellisen elämän kuvaaminen ja kohdetta pyritään tutkimaan mahdollisimman kokonaisvaltaisesti. Kvalitatiivisessa tutkimuksessa täydellistä objektiivisuutta on mahdotonta saavuttaa, sillä tutkija ja tutkittava kohde sitoutuvat toisiinsa. (Hirsjärvi et al. 2007, s. 157). Kvantitatiivinen tutkimus yhdistetään usein positivismiin ja kvalitatiivinen tutkimus hermeneuttiseen tieteenkäsitykseen (Olkkonen 1993, s. 39). Tässä tutkimuksessa tarkastellaan todellisen elämän tilannetta, jota ei ole mahdollista asettaa tilastolliseen muotoon, joten kyseessä on kvalitatiivinen tutkimus ja hermeneuttinen tieteenkäsitys.

Creswell (2003) esittelee kvalitatiiviselle lähestymistavalle viisi yleisimmin käytettyä tutkimusstrategiaa: etnografia, grounded theory, fenomenologia, narratiivi ja tapaustutkimus. Etnografiassa tutkitaan ihmisten toimintaa ja työskennellään ei-rakenteellisen aineiston kanssa, jolloin tavoitteena on omaksua tutkimuksen osanottajien näkökulma ja kuvata se. Grounded theoryn avulla kehitetään teoriaa aineistoista löytyvien havaintojen, riippuvuussuhteiden ja järjestämisen kautta. Fenomenologiassa puolestaan tutkitaan, miten ihminen havaitsee todellisuuden omassa kokemusmaailmassaan ja narratiivi kuvaa tutkimuksen kertomuksenomaisesti etenevänä juonena. Tapaustutkimuksessa on puolestaan yksityiskohtaisin ja siinä suoritetaan syvällinen tutkimus valitusta yksittäisesti tapauksesta keräämällä siitä yksityiskohtaista tietoa. (Creswell 2003, s. 13) Tässä tutkimuksessa käsitellään yksittäistä tapausta ja tutkimus lähestyy siis tapaustutkimusta.

Tapaustutkimus on empiirinen tutkimus, joka tutkii nykyajan ilmiötä perusteellisesti sen tosielämän tilanteessa. Tapaustutkimukselle luonteenomaista on, että tutkittavan ilmiön

ja sen asiayhteyden rajat eivät ole selvästi havaittavissa. Tutkimusstrategiana tapaustutkimus kattaa sekä suunnittelun, datankeruun että lähestymistavan datan analysoinnille. Näin ollen tapaustutkimus on kaiken kattava metodi ja tutkimusstrategia. (Yin 2003, ss. 13-14) Tapaustutkimus metodina mahdollistaa tositapahtumien kokonaisvaltaisen ja tarkoituksenmukaisen tarkastelun, jonka tarkoituksena on keskittyä nykyaikaiseen ilmiöön. Tapaustutkimukselle tyypillisiä tutkimuskysymyksiä ovat *miten* ja *miksi*. (Yin 2003, ss. 2-5) Tapaustutkimus käsittelee yksityiskohtaista ja intensiivistä tietoa yksittäisestä tapauksesta ja aineistoa kerätään useita metodeja, kuten havainnointia, haastatteluja käyttäen (Hirsjärvi et al. 2007, ss. 130-131). Tässä tutkimuksessa käsitellään yksittäistä tapausta ja kvalitatiivisin metodein ja tutkimuskysymykset ovat *miten*-kysymyksiä, joten tutkimusotteena ja -strategiana on tapaustutkimus.

Tapaustutkimus keskittyy yhden yksityiskohtaisen ja yleensä tosielämää kuvaavan tapauksen tutkintaan. Tässä tutkimuksessa tutkittavana tapauksena on liiketoimintaorientoituneen loppukäyttäjän suorittama NoSQL-tietokannan datan testaus raportointityökalujen avulla. Metodologia tätä tapaustutkimusta varten on esitetty luvussa 1.4.

1.4 Tutkimusmetodologia ja työn rakenne

Kuten on aiemmin mainittu, diplomityö on tapaustutkimus ja sen tutkimusosuudessa on sekä kirjallisuuteen perustuva teoriapohja että empiriaan perustuva tutkimusosuus. Tutkimuksen ensimmäisenä osana on kirjallisuustutkimus. Kirjallisuustutkimuksen avulla tarkastellaan diplomityön keskeisimpiä käsitteitä ja niiden välisiä suhteita, ja tavoitteena on kuvata ja selventää aihealuetta siihen liittyvien käsitteiden avulla, sekä luoda teoriapohja empiiriselle tutkimukselle. Lähdemateriaalina kirjallisuustutkimukselle toimivat tieteelliset artikkelit ja aihealueen kirjallisuus. Lähdemateriaalin saatavuus on pääosin Tampereen teknillisen yliopiston tarjoamien artikkeli- ja kirjastotietokantojen varassa. Lähdemateriaalin luotettavuutta pyritään arvioimaan sen ajankohtaisuuden ja siihen viittaavien artikkeleiden perusteella.

Tämä tutkimus on tapaustutkimus ja empiirisesti tutkittavaksi tapaukseksi voidaan määritellä tilanne, jossa liiketoimintaorientoitunut loppukäyttäjä saa testattavakseen NoSQL-tietokannassa olevaa dataa. Loppukäyttäjä ei tunne NoSQL-tietokantojen rakennetta tai kyselykieliä, joten hän tarvitsee datan tarkasteluun erillisen työkalun. Työkaluvaihtoehtona nähdään itseohjaavat raportointityökalut, jotka mahdollistavat helpon datan tarkastelun ja käsittelyn. Tavoitteena on huomata, sopivatko raportointityökalut testauskäyttöön ja miten vertailtavissa olevat työkalut soveltuvat testaukseen. Teorian ja empirian osa-alueet voidaan tunnistaa kuvasta 1.2.

Kuva 1.2. *Diplomityön rakenne*

Tutkimuksessa luodaan luvuissa 2-4 ensin teorian pohjalta ymmärrystä aiheesta ja haetaan tieteellistä viitekehystä. Teoriaosuuden pohjalta muodostetaan testaussuunnitelma, josta ilmenee NoSQL-tietokannan datan testauksessa huomioon otettavat asiat, jotka voidaan testata raportointityökalujen avulla loppukäyttäjän toimesta. Testaussuunnitelmassa otetaan huomioon testattava datajoukko ja käytettävät testaustyökalut. Testaussuunnitelma toteutetaan ja saadaan tuloksena näkemys siitä, pystyytäänkö raportointityökalujen avulla huomaamaan tietokannan datan oikeellisuuden, ja miten eri työkalut sopivat testaussuunnitelman toteuttajiksi. Empiirisen tarkastelun avulla pyritään siis vastaamaan tutkimuskysymyksiin.

Tarkan empiirisen testauksen suorittaa tutkijan lisäksi kaksi koehenkilöä. Kaikki etsivät testaussuunnitelman mukaisia asioita testattavasta aineistosta ja lisäksi vertailevat kolmea eri työkalua testaustarkoituksessa. Tutkija dokumentoi omat löydöksensä sanallisesti ja kuvakaappauksien avulla. Kun tutkija on suorittanut testauksen itse, tarkkailee hän koehenkilöiden suorittamaa testausta ja dokumentoi koenhenkilöiden testausprosessin sanallisesti ja verraten sitä omaan prosessiinsa. Testaus toteutetaan ennalta määritetyillä raportointityökaluilla. Raportointityökaluina tutkimuksessa käytetään liiketoimintatiedon hallinnan työkaluja Tableauta, Power BI:ta ja Qlik Sensea, joista kaikki ovat suurten toimijoiden toimittamia työkaluja. Työkalujen valinnan pohjalla on, että työkaluista on mahdollista saada maksuton kokeiluversio käyttöön ja käytön aloittaminen on helppoa. Kun testaussuunnitelma on toteutettu ja tulokset eri työkaluilla dokumentoitu, voidaan

tuloksia vertailla ja niiden pohjalta muodostaa johtopäätöksiä. Johtopäätösten pohjalta on tarkoitus muodostaa käsitys raportointityökalujen soveltuvuudesta NoSQL-tietokannan datan testaukseen ja mahdollisesti antaa ehdotus parhaiten soveltuvasta työkalusta

Testauksen toteuttamista varten luodaan datajoukko, johon tehdään testaussuunnitelman mukaisia oletuksia ja poikkeamia. Datajoukko muodostuu Energiateollisuus ry:n tarjoamasta avoimesta sähköntuotannon ja -kulutuksen tuntidatasta Suomessa vuonna 2014 (Energiateollisuus 2015). Jatkossa tähän datajoukkoon viitataan termillä *sähködata*. Empiirisen tutkimuksen toteuttamiseksi sähködata tallennetaan NoSQL-tietokantaan, johon muodostetaan yhteys raportointityökaluilla. Yhteyden muodostus dokumentoidaan, jotta raportointityökaluja voidaan vertailla. Tutkimuksessa pohjana on oletus, että NoSQL-tietokanta on jo olemassa ja siellä on testattavaa dataa, joten tietokannan pystytykseen ja datan sinne tallentamisen vaiheisiin ei oteta kantaa. Tutkimuksessa käytettävän tutkimustilanteen lähtökohdat on kuvattu tarkemmin luvussa 5.1.

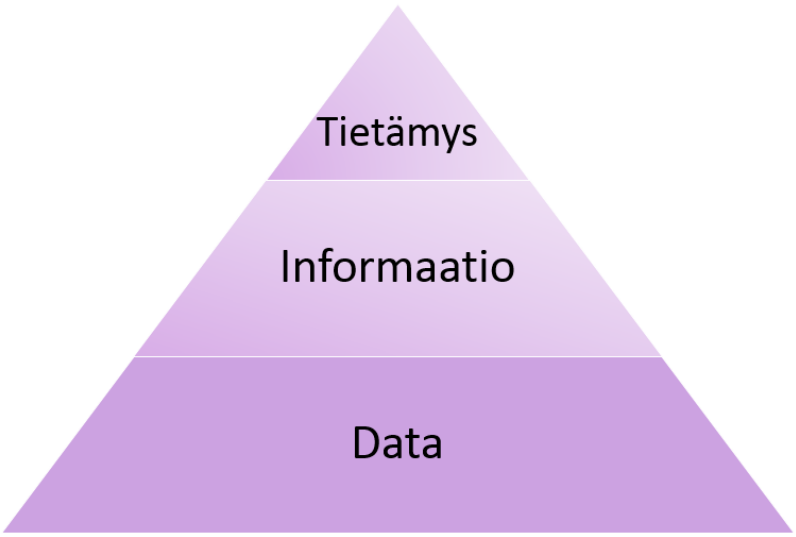
2. LIIKETOIMINTATIEDON HALLINTA JA RAPORTOINTITYÖKALUT

Tämän tutkimuksen kannalta on tärkeää ymmärtää, mitä raportointityökaluilla tarkoitetaan ja mihin tarkoitukseen niitä voidaan käyttää. On myös oleellista ymmärtää, että raportointityökalut ovat liiketoimintatiedon hallinnan työkaluja, joiden avulla voidaan käsitellä tietoa. Tässä luvussa tarkastellaan ensin liiketoimintatiedon hallinnan käsitettä ja teknistä näkökulmaa, ja sen jälkeen esitellään raportointityökalujen rooli tässä kentässä.

2.1 Liiketoimintatiedon hallinnan käsite

Organisaatioiden käytettävissä olevan tiedon määrä on lisääntynyt merkittävästi tiedonvälitysteknologioiden kehittyessä. Suuren tietomäärän vuoksi organisaation on haastavaa löytää oleellista tietoa päätöksenteon tueksi ja näin saavuttaa liiketoimintaympäristönsä syvällistä tuntemusta ja ymmärrystä. (Hannula et al. 2002, s. 75) Tietoa organisaatioon kertyy monesta eri lähteestä ja toiminnosta, ja tieto tallentuu datana muun muassa operatiivisiin tietojärjestelmiin. Datasta ja sen määrästä ei itsessään ole kuitenkaan hyötyä, ellei sitä osata hyödyntää. (Hovi et al. 2009, ss. 4-6) Hyödyntämisellä tarkoitetaan datan saattamista informaatioksi ja tietämykseksi, joiden avulla liiketoimintaa voidaan strategisesti johtaa. Tässä apuna ovat liiketoimintatiedon hallinnan ratkaisut. (Hovi et al. 2009, s. 73)

Datan saattaminen informaatioksi ja tietämykseksi voidaan osittain selittää tiedon hierarkkisten tasojen avulla. Kirjallisuudessa tiedolle tunnistetaan kolmesta kuuteen tasoa. Kaikki mallit jakavat kolme alinta tasoa, jotka ovat data (engl. data), informaatio (engl. information) ja tietämys tai tieto (engl. knowledge). (Esimerkiksi Thierauf 2001; Liebowitz 2006; Sydänmaanlakka 2007) Tämän tutkimuksen kannalta olennaisia ovat nämä tiedon kolme alinta tasoa: data, informaatio ja tietämys. Nämä tasot voidaan tunnistaa kuvasta 2.1. Näitä ylemmät tasot liittyvät vahvasti ihmisen asiantuntijuuteen ja kokemukseen, joten niiden tarkastelu ei ole olennaista tietokantoja ja raportointityökaluja ajatellen.



Kuva 2.1. *Tiedon tasot*

Tiedon hierarkian alin taso on **data**. Data on merkkijono tai merkkien kokoelma, jolla ei ole laajempaa merkitystä ilman tulkintaa (Ståhle & Grönroos 1999, s. 207; Kaario & Peltola 2008, s. 6). Data on numeroita, tekstiä ja kuvia sekä niiden yhdistelmiä ja sitä voidaan tallentaa tietokantoihin (Sydänmaanlakka 2007, s. 187). Datan voidaan ajatella olevan raaka-ainetta, josta informaatio syntyy, sillä erilaisten muokkaus ja -muunnosprosessien avulla data voidaan saattaa informaatioksi (Hakala 2006, s. 66; Sydänmaanlakka 2007, s. 187). Data muuttuu informaatioksi, kun se kytketään kokonaisuuteen, analysoidaan, korjataan tai tiivistetään. Tällöin merkkijono muuttuu viestiksi, jolla on merkitystä vastaanottajalle. (Liebowitz, 2006, s. 7; Sydänmaanlakka 2007, s. 188)

Tiedon toiseksi alin taso on **informaatio**. Se on dataa kontekstissa, jonka vastaanottaja voi ymmärtää, jos sillä on hänelle informaatioarvoa eli jos vastaanottaja pystyy antamaan merkkijonolle merkityksen (Ståhle & Grönroos 1999, s. 49; Liebowitz 2006, s. 7; Kaario & Peltola 2008, s. 6). Esimerkiksi nuottivihon merkinnät ovat dataa ja muuttuvat informaatioksi sille, joka osaa lukea nuotteja; muille merkit ovat merkityksettömiä (Ståhle & Grönroos 1999, s. 49). Tietojärjestelmien kannalta voidaan sanoa, että informaatio on dataa, joka on muunnettu merkitykselliseksi kokonaisuudeksi (Sydänmaanlakka 2007, s. 187). Erilaisten tietoteknisten työkalujen avulla datasta voidaan muodostaa informaatiota, jolla on merkitystä sen hyödyntäjälle (Thierauf 2001, s. 8).

Tiedon kolmas taso **tietämys** on puolestaan aktiivinen käsite, joka sisältää informaation lisäksi vaikutuksen eli informaatio on muuttunut inhimilliseksi tiedoksi, jota voidaan soveltaa jonkun tehtävän suorittamiseksi tai ongelman ratkaisemiseksi (Ståhle & Grönroos 1999, s. 49). Tietämys on siis informaatiota, johon liittyy myös henkilön kokemuksia ja näkemyksiä (Liebowitz 2006, s. 7). Tulkitusta ja sisäistetystä informaatiosta syntyy tietämystä (Kaario & Peltola 2008, s. 6). Nämä tiedon kolme tasoa

voidaan löytää liiketoiminatiedon hallinnan järjestelmistä. Liiketoimintatiedon hallinnan työkalut hyödyntävät tietokantaan tai muihin tietolähteisiin tallennettua dataa. Työkalujen muodostamien valmiiden analyysien tai jäsentelyjen avulla data voidaan kommunikoida jäsenneltynä informaationa loppukäyttäjälle, joka suhteuttaa informaation oikeaan kontekstiin, jolloin syntyy tietämystä. (Koskinen et al. 2005, s. 5)

Liiketoimintatiedon hallinta (engl. business intelligence, BI) ja sen järjestelmät ovat termeinä melko laveita (Hannula et al. 2002, ss. 75-76). Laajan näkemyksen mukaan liiketoimintatiedon hallinta kattaa terminä kaikenlaisen sisäisestä ja ulkoisesta ympäristöstä muodostuvan tiedon, jota voidaan hyödyntää organisaation ohjaamisessa (Hovi et al. 2009, ss. 78-79). Pirttimäen (2007, s. 64) mukaan liiketoimintatiedon hallinta on prosessi, johon kuuluu sarja systemaattisia toimintoja, joiden taustalla on spesifi informaatiotarve päätöksentekijöille, jotta voidaan saavuttaa kilpailuetu.

Liiketoimintatiedon hallinta on yksinkertaistettuna datan jalostamista ja saattamista päätöksentekijöiden hyödynnettäväksi tukemaan strategisia päätöksiä. Liiketoimintatiedon hallinta on lähestymistapa, jonka avulla voidaan löytää ja selittää dataa suuremmassa kontekstissa, ja käyttää näitä löydöksiä ja selityksiä päätöksenteon tukena (Liebowitz 2006, ss. 19-20). Lähestymistapa viittaa moninaisiin prosesseihin, tuotteisiin, tekniikkoihin ja työkaluihin, joiden avulla voidaan tehdä nopeampia ja parempia päätöksiä (Pirttimäki 2007, s. 2). Jotta päätöksenteko on aidosti nopeampaa ja parempaa, tulee prosessissa kiinnittää huomiota tiedon tarkkuuteen, ajantasaisuuteen ja käytön yleisyyteen (Thierauf 2001, s. 67).

Nykypäivän organisaatiot tarvitsevat liiketoimintatiedon hallintaa, sillä datan määrä on kasvanut eksponentiaalisesti ja sen hyödyntäminen on entistä haasteellisempaa (Pirttimäki 2001, s. 5). Liiketoimintatiedon hallinnan prosessin avulla data voidaan saattaa paremmin hyödynnettävään muotoon. Kun liiketoimintatiedon hallintaa ajatellaan prosessina, on siinä useita vaiheita. Giladin ja Giladin (1986, s. 53) mukaan liiketoimintatiedon hallinnalla on viisi tärkeää vaihetta:

1) raakadatan kerääminen
2) datan ja informaation luotettavuuden ja oikeellisuuden varmistaminen
3) datan ja informaation analysointi
4) informaation varastointi
5) analysoidun informaation jakaminen päätöksentekijöille.

Liiketoimintatiedon hallinnan ratkaisujen loppukäyttäjänä toimiva päätöksentekijä on yleensä osallisena ainoastaan prosessin viidennessä vaiheessa, jossa hänelle jaetaan valmiiksi analysoitu informaatio. Tätä ennen data on etsitty tietojärjestelmistä, sen luotettavuus ja oikeellisuus on varmistettu, ja tämä varmistettu data on analysoitu ja varastoitu selkeään tietokantarakenteeseen (Gilad & Gilad 1986, s. 53). Prosessissa dataa tallennetaan tietokantaan, josta sitä voidaan hyödyntää liiketoimintatiedon hallinnan

työkalujen avulla. Tietokannan pohjalta organisaatio voi organisoida, analysoida ja kommunikoida omaa dataansa, jolloin siitä muodostuu informaatiota. (Thierauf 2001, s. 3) Kun tämä informaatio jaetaan päätöksentekijöille, voivat he yhdistää uuden informaation jo olemassa olevaan tietämykseensä ja saavuttaa näin lisää tietämystä. Tämän lisääntyneen tietämyksen pohjalta voidaan tehdä parempia ja valistuneempia päätöksiä, jotka tuottavat todellisen kilpailuedun. (Pirttimäki 2007, s. 3) Liiketoimintatiedon hallinnan prosessissa liikutaan siis tiedon hierarkiassa alhaalta ylös: datasta informaatioksi ja edelleen tietämykseksi. Tekninen näkökulma prosessille on selitetty luvussa 2.3.

Liiketoimintatiedon hallinta käsittelee ja yhdistelee monen tyyppisiä ja muotoisia tietoja organisaation toimintojen ohjaamisen tueksi. Yhdisteltävää tietoa voidaan saada organisaation työntekijöiden ja asiakkaiden toimista, tarpeista ja vaatimuksista, sekä organisaation omista toiminnoista. Tietolähteet voivat olla myös yhteydessä ulkoisiin lähteisiin. (Thierauf 2001, ss. 9-10; Kaario & Peltola 2008, s. 61) Tätä organisaation ja sen järjestelmien käytössä olevaa tietoa kutsutaan tietopääomaksi. Useimmat yritykset eivät välttämättä osaa mitata, arvostaa, eivätkä viestiä tietopääomansa ominaisuuksia ja suhdetta strategiaansa. Tämä tuottaa haasteita organisaation johtamiselle ja myös sen sidosryhmille, kuten sijoittajille. (Ståhle & Grönroos 1999, ss. 58-59) Suurin ongelma on, jos tietoa ei ymmärretä ja osata hyödyntää osana prosesseja. Oleellista tiedon hyödyntämisen ja liiketoimintatiedon hallinnan kannalta on, että organisaation tietopääoma on saatettu sähköiseen muotoon, jotta sen varastointi on tehokasta ja tietoa voidaan hyödyntää ja jakaa tietojärjestelmäpohjaisten työkalujen avulla (Sydänmaanlakka 2007, s. 184). Jotta tietopääomaa voidaan hallita ja hyödyntää, tulee se varastoida järkevästi. Organisaation data voidaan esimerkiksi tallentaa tietokantaan, jonka päälle voidaan rakentaa informaation ja tietämyksen syntymistä tukevia liiketoimintatiedon hallinnan järjestelemiä (Hovi et al. 2009, s. 73).

Datan ja informaation määrän jatkuva lisääntyminen haastaa liiketoimintaa ja sen ohjaamista. Voimakas informaation lisääntyminen organisaation kaikissa prosesseissa on samalla haaste sekä mahdollisuus. (Ståhle & Grönroos 1999, s. 34; Xiang et al. 2010) Organisaatiossa on oltava saatavilla informaatiota ja sen on pystyttävä suodattamaan siitä juuri se osa, joka voi tyydyttää mahdollisimman taloudellisesti ja tehokkaasti vaihtuvan tietotarpeen eikä yhtään liikaa (Hakala 2006, s. 129). Moni organisaatio kerää tietoa, mutta ei osaa, pysty, halua tai ehdi hyödyntää kaikkea sitä. Ongelmana ei ole siis tiedon talteenotto vaan jatkojalostaminen, analysointi ja hyödyntäminen. Liiketoimintatiedon hallinnan ratkaisut pyrkivät vastaamaan nöihin haasteisiin. (Kubina et al. 2015)

Tiedon perusteella tehdään päätöksiä ja strategisia liikkeitä. Liiketoimintatiedon hallinnan ratkaisujen käyttöönoton avulla organisaatio voi muun muassa oppia ennakoimaan asiakkaidensa ja kilpailijoidensa toimintaa ja markkina-alueensa trendejä ja markkinoilla tapahtuvia muutoksia (Pirttimäki 2007, s. 2; Kubina et al. 2015).

Informaation luonnille ja hyödyntämiselle voidaan Choon (1998) mukaan tunnistaa kolme pääasiallista strategista aluetta:

1) Organisaatio käyttää informaatiota ymmärtääkseen sen liiketoimintaympäristössä tapahtuvia muutoksia ja kehitystä.
2) Organisaatio luo, jäsentää ja prosessoi informaatiota luodakseen uutta tietämystä.
3) Organisaatio etsii ja arvioi informaatiota päätöksenteon tueksi.

Näillä tietoa tulvivilla markkinoilla kilpailuedun saavuttaa se yritys, joka tehokkaimmin ennakoi muutokset ja muokkaa tiedosta tietämystä sekä hyödyntää syntynyttä osaamista. Liiketoimintatiedon hallinnan avulla voidaan tuottaa ja hallita tätä prosessia ja tietoa. (Ståhle & Grönroos 1999, ss. 48-51; Hannula et al. 2002, s. 75) Liiketoimintatiedon hallinnan ratkaisujen avulla tietojärjestelmissä oleva data voidaan nostaa mielekkäässä muodossa päätöksentekijöiden hyödynnettäväksi. (Hovi et al. 2009, s. 74) Liiketoimintatiedon hallinnan työkalut auttavat käyttäjää pääsemään käsiksi dataan sekä luomaan, hallitsemaan ja ajamaan päätöksenteko-objekteja, kuten kyselyitä, raportteja ja analyyseja. Liiketoimintatiedon hallinnan prosessin ymmärtämisessä on kuitenkin tärkeää huomata, että usein loppukäyttäjänä toimiva päätöksentekijä ei tarvitse tai hän ei osaa itse luoda ja hallita tietoa. Tämän takia datan käsittely ja analysointi on yleensä automatisoitu määrittelyjen avulla ja päätöksentekijä näkee vain omalle toiminnalleen olennaiset analyysit ja raportit. (Thierauf 2001, s. 69; Hovi et al. 2009, s. 74)

2.2 Liiketoimintatiedon hallinnan tekninen näkökulma

Liiketoimintatiedon hallinnassa pääosassa ovat liiketoimintatiedon hallinnan työkalut ja niiden avulla toteutetut ratkaisut. Työkalujen tehtävänä on tukea tiedon luomista, varastointia ja siirtoa (Alavi & Leidner 2001, s. 107). Liiketoimintatiedon hallinnan työkaluilla rakennettujen ratkaisujen avulla organisaatioiden henkilöstö ja päätöksentekijät pääsevät käsiksi liiketoimintaa kuvaavaan informaatioon ja tämä informaatio auttaa heitä tekemään valistuneempia päätöksiä ja ohjaamaan toimintaa oikeaan suuntaan. Hyvin toteutetut liiketoimintatiedon hallinnan ratkaisut nostavat organisaation päätöksentekijöiden käyttöön tietojärjestelmien syvyyksissä uinuvan datan. (Hovi et al. 2009, s. 74)

Jotta organisaation data saadaan informaationa päätöstentekijöiden ulottuville, on sitä ensin jalostettava. Datan jalostuksen liiketoimintaa hyödyttävään muotoon voidaan ajatella olevan prosessi. Prosessia voidaan tarkastella aiemmin mainitun Giladin ja Giladin (1986, s. 53) määritelmän mukaan viitenä eri vaiheena. Vaiheet ja niiden yksityiskohdat on esitetty taulukossa 2.1

Taulukko 2.1. *Liiketoimintatiedon hallinnan prosessi (vaiheet lähteestä Gilad & Gilad 1986; yksityiskohdat lähteestä Hovi et al. 2009)*

Vaihe	Yksityiskohdat
Raakadatan kerääminen	Tiedon tunnistaminen operatiivisista järjestelmistä
Datan luotettavuuden ja oikeellisuuden varmistaminen	Hankitun tiedon muokkaaminen ja jäsentäminen, sekä väärän tiedon eliminointi
Datan ja information analysointi ja esittäminen	Tiedon arviointi, sopivien kaavojen ja mallien etsintä sekä skenaarioiden muodostaminen, raportointi- ja analysointiratkaisut
Informaation varastointi	Informaation varastointi sopivalla tasolla
Analysoidun information jakaminen päätöksentekijöille	Informaation visualisointi, kommunikointi

Liiketoimintatiedon hallinnan prosessin ensimmäinen vaihe on **raakadatan kerääminen** (Gilad & Gilad 1986, s. 53). Datan keräämisessä on otettava huomioon tietotarpeet, joihin dataa tarvitaan, ja määriteltävä näiden pohjalta kerättävä data ja keräyksen laajuus. Kun tarvittavan raakadatan määrittely on tehty, on tarvittava tieto etsittävä lähdejärjestelmistä. (Hannula et al. 2002, s. 84) Yleisimpiä raakadatan lähteitä ovat organisaation prosesseja tukevat operatiiviset tietojärjestelmät, kuten asiakkuudenhallinnan, toiminnanohjauksen ja taloushallinnon järjestelmät. Lähteet voivat olla myös organisaation ulkopuolisia lähteitä, kuten markkina- tai kilpailijatietoa. (Hovi et al. 2009, s. 86) Oleellista on tietysti, että organisaation tuottama ja tarvitsema tieto on tallennettu datana tietojärjestelmiin, joista sitä voidaan hyödyntää (Sydänmaanlakka 2007, s. 184).

Raakadatan tunnistamista ja keräämistä seuraa **datan luotettavuuden ja oikeellisuuden varmistaminen**. Kun raakadata on tunnistettu ja kerätty lähdejärjestelmistä, tulee tieto jäsennellä rakenteellisesti järkeviksi kokonaisuuksiksi ja samalla eliminoida epäolennainen ja väärä tieto (Hannula et al. 2002, s. 84). Tätä kutsutaan tiedon integroinniksi tai yhdistelyksi, sillä liiketoimintatiedon hallinta ei keskity ainoastaan yhteen tietolähteeseen, vaan se yhdistelee dataa eri lähteistä (Thierauf 2001, s. 125). Perinteisesti organisaation data tallentuu operatiivisiin perusjärjestelmiin. Perusjärjestelmät ovat kuitenkin usein hajallaan ja niiden tietorakenteet eivät ole yhteneviä. Näin ollen tieto ei välttämättä ole helposti saatavilla analysointia ja raportointia varten, ja siksi tarvitaan integrointiprosessi, jonka avulla data voidaan valjastaa informaatioksi ja tietämykseksi tukemaan organisaation päätöksentekoa. (Hovi et al. 2009, s. xi)

Tiedon integrointi tapahtuu perinteisesti ETL-prosessin (Extract-Transform-Load) kautta, jossa data haetaan, muokataan ja ladataan lähdejärjestelmistä eteenpäin tietovarastointiratkaisuihin. Tämä vaihe toteutetaan tehokkailla tiedon integroinnin ohjelmistoilla, joihin on määritelty datan muokkaamistarpeet ja latausmuoto. ETL-

prosessissa raportointia ja analysointia varten kerättävä data tallennetaan tietokantapohjaiseen varastointiratkaisuun. (Thierauf 2001, ss. 68-69; Hovi et al. 2009, s. 86) ETL-prosessin avulla suuret datamäärät voidaan jäsennellä rakenteellisesti järkeviksi kokonaisuuksiksi ja epäolennainen tai väärä tieto voidaan eliminoida (Hannula et al. 2002, s. 84). Jäsentelyn tuloksena dataa saadaan ryhmiteltyä informaatioksi. Liiketoimintatiedon hallinnan järjestelmissä datan laatu ja oikeellisuus on erittäin tärkeä huoli, ja integrointivaiheessa tulee kiinnittää huomiota, että datan muokkaus ja jäsentely tapahtuvat hallitusti eivätkä aiheuta laatuongelmia (Thierauf 2001, s. 125).

Kun operatiivisten järjestelmien data integroidaan, tallennetaan se yleensä tietovarastoon. Tietovarastoinnin tehtävänä on yhdistää ja yhdenlaistaa tietoa eri lähteistä hyvin suunniteltuun tietokantaan, josta tieto on sitten liiketoimintatiedon hallinnan työkalujen avulla helppokäyttöisesti ja selkeästi saatavilla käyttäjille. (Hovi et al. 2009, s. xiii) Data tallennetaan tietokantaan, joka ylläpitää organisaation dataa ja informaatiota sekä mahdollistaa pääsyn tähän tietopääomaan (Thierauf 2001, ss. 68-69).

Datan integroinnin jälkeen tapahtuu liiketoimintatiedon hallinnan prosessin olennaisin vaihe eli **datan ja informaation analysointi**, jonka avulla jäsenneltyä dataa ja informaatiota halutaan jalostaan päätöksenteon tietotarpeita vastaavaksi informaatioksi. Tietovaraston datan analysoinnissa ja tutkimisessa tarvitaan monenlaisia näkökulmia. (Hovi 1997, s. 90) Liikevaihtodataa voidaan esimerkiksi tarkastella sekä asiakas- että vähittäismyyjä tasolta ja eri aikaväleiltä. Analysoinnissa on otettava huomioon nämä eri näkökulmat ja toteutettava analysoitua sisältöä kaikille sitä tarvitseville. Analysoidun informaation perusteella voidaan luoda yhteys ja merkitys yrityksen ja senhetkisen päätöksentekotilanteen välille (Hovi et al. 2009, ss. 85-86). Informaation analysoinnissa informaation arvioinnin, sopivien kaavojen ja mallien etsimisen ja hyödyntämisen sekä skenaarioiden muodostamisen kautta on tarkoitus tuottaa arvokasta ja käyttökelpoista informaatiota, jota päätöksentekijät voivat hyödyntää päätöksentekoprosessissa (Hannula et al. 2002, s. 84).

Analysoitu informaatio tulee esittää loppukäyttäjälle sopivassa muodossa. Perinteisesti informaatio halutaan esittää raportteina ja taulukoina, toisaalta taas esimerkiksi kehitystä on myös helppo seurata graafisina kuvaajina (Hovi et al. 2009, s. 87; Kubina et al. 2015). Yleensä organisaatiossa on tarve sekä vakioraporteille että nopeille, uusille ja määrittelemättömille kyselyille, joiden avulla voidaan tietokannasta nopeasti tarkistaa jokin informaatiossa kiinnostava tai askarruttava asia. (Hovi 1997, s. 90) Hyvin tuotettujen liiketoimintatiedon hallinnan ratkaisujen myötä loppukäyttäjien ei tarvitse tuntea tietokantojen tai -järjestelmien rakenteita vaan heidän tarvitsemansa informaatio on helposti käytettävissä, silloin kun he sitä tarvitsevat. (Hovi et al. 2009, s. 74) Raportointia ja raportointityökaluja tarkastellaan tarkemmin luvussa 2.4.

Analysointiprosessin tuloksena syntynyt **informaatio varastoidaan** sopivaan rakenteeseen, josta loppukäyttäjät pääsevät analyysia hyödyntämään. Analysoitu

informaatio voidaan varastoida esimerkiksi erilliselle palvelimelle, jonka sisältöön loppukäyttäjä pääsee käsiksi selainkäyttöliittymän avulla. Valmiiden analyysien tallennuspaikkana saattaa olla myös jaettu verkkolevy, josta analyysin voi noutaa omaan käyttöönsä. Tärkeää on, että liiketoimintatiedon hallinnan prosessina syntyneet tuotteet ovat oikea-aikaisesti loppukäyttäjän saatavilla ja, että ne tukevat myös mahdollisia uusia tietotarpeita (Hannula et al. 2002, s. 84). Analysointiprosessin tuotosten tulisi usein olla saatavilla myös organisaation ulkoisessa verkossa, jolloin joudutaan kiinnittämään erityistä huomiota tallennusalustan tietoturvallisuuteen, sillä analysoitu informaatio on erittäin tärkeää organisaation toiminnan ja kilpailukyvyn kannalta (Hannula et al. 2002, s. 85; Hovi et al. 2009, s. 74).

Lopulta on vuorossa **analysoidun informaation jakaminen päätöksentekijöille**. Oleellista analysoidun informaation jakamisessa on, että oikea tieto on saatettava oikeille henkilöille oikea-aikaisesti (Hovi et al. 2009, s. 73). Liian myöhäinen kommunikointi saattaa viedä arvon koko liiketoimintatiedon hallinnan prosessilta, sillä silloin päätöksiä joudutaan tekemään ilman kattavaa taustainformaation analysointia. (Hannula et al. 2002, s. 84) Tältä ongelmalta voidaan välttyä, jos päätöksentekotarpeet on määritelty hyvin ja liiketoimintatiedon hallinnan prosessi toimii määrittelyn pohjalta oikea-aikaisesti. On siis määriteltävä, ketkä tarkastelevat analysoitua informaatiota, mistä sitä luodaan, millä se esitettään, ja millä aikavälillä ja käynnisteellä tieto päivittyy sekä miten käyttäjille ilmoitetaan päivittyneestä informaatiosta. Analysoidun informaation jakaminen päätöksentekijöille on liiketoimintatiedon hallinnan prosessin perimmäisin ja tärkein tavoite. Niinpä on tärkeää kiinnittää huomiota jakeluprosessiin. (Kubina et al. 2015) Jakelussa välineenä toimivat tietotekniset ratkaisut, joissain tapauksissa toimii sähköposti, mutta lähtökohtaisesti olisi suotavaa, että analyyseihin pääsisi käsiksi oikea-aikaisesti esimerkiksi selainkäyttöliittymän kautta. Näin organisaatioiden henkilöstö ja päätöksentekijät pääsevät käsiksi liiketoimintaa kuvaavaan informaatioon ja tämä informaatio auttaa heitä tekemään valistuneempia päätöksiä ja ohjaamaan toimintaa oikeaan suuntaan (Hovi et al. 2009, s. 74).

Nykyajan jatkuvasti muuttuvassa liiketoimintaympäristössä liiketoimintatiedon hallinnan järjestelmille on yhä enemmän tarvetta ja koko organisaation laajuiset liiketoimintatiedon hallinnan järjestelmät ovat nykypäivää. Nämä järjestelmät ja ratkaisut haastavat organisaatioita järjestelemään tiedonhallintansa uudestaan ja niiden kehittäminen ja käyttöönotto organisaatiossa vaatii näkemystä, kärsivällisyyttä ja investointeja. (Thierauf 2001, s. 3; Liebowitz 2006 s. 43) On myös huomioitava, että liiketoimintatiedon hallinnan ratkaisu ei ole yleensä kertainvestointi. Organisaation tietotarpeisiin tulee muutoksia ja päätöstenteon aikaväli lyhenee. Raportoinnissa on siirryttävä kvartaali- tai kuukausiraportoinnista päivä- tai tuntitasoiseen raportointiin. Nopeampien ja entistä monipuolisempien analyysien teko myös lisää tiedon moninaisuutta. Informaatiota koostetaan useasta eri lähteestä ja integrointimenetelmät ja tallennustietokannat joutuvat koetukselle. Tarvitaan siis entistä tehokkaampia tietokanta- ja liiketoimintatiedon

hallinnan ratkaisuja, joten organisaatioiden on oltava valmiita investoimaan myös tulevaisuuden tarpeisiin. (Hovi et al. 2009, s. 76)

2.3 Raportointityökalut

Tiedosta ja tiedonhallinnasta on viime vuosikymmeninä tullut yhä tärkeämpi organisaatioiden resurssi ja kilpailutekijä (Alavi & Leidner 2001, s. 107; Sydänmaanlakka 2007, s. 175). Tietopääoma on resurssina tärkeä, mutta keskiössä ei ole tiedon omistaminen, vaan miten organisaatio osaa hyödyntää tietoaan (Sydänmaanlakka 2007, s. 175). Informaation todellinen voima on sen täsmällisesti määritellyssä ja hyödynnetyssä muodossa (Ståhle & Grönroos 1999, ss. 140-141). Tiedon hyödyntämisessä apuna käytetään liiketoimintatiedon hallinnan työkaluja, joista raportointityökalut ovat perinteisimmät loppukäyttäjän työkalut (Hovi et al. 2009, s. 4).

Tietokannan datan analysoinnissa ja tutkimisessa tarvitaan monenlaisia näkökulmia. Perinteisesti tieto halutaan raportteina. (Hovi 1997, s. 90) Raporteiksi mielletään liiketoimintatiedon hallinnan ratkaisuissa analysoidun tiedon yleisimmät esitysmuodot. Raportointityökalujen avulla voidaan luoda erilaisia raportteja. Perinteiset raportointityökalut esittävät analysoidun tiedon taulukkomuodossa tai pylväs- ja viivakuvaajina sopivilla akseleilla. (Hovi et al. 2009, s. 87) Näiden lisäksi suosiotaan ovat nostaneet laajennetut raportointityökalut, joiden avulla voidaan luoda visuaalisia mittaristoja (engl. dashboard) ja toiminnan kriittisiä lukuja esittäviä tuloskortteja (engl. scorecard) tai muita visuaalisia esityksiä (Turban et al. 2008, s. 97; Hovi et al. 2009, s. 85). Raporttien ja visualisointien avulla voidaan esittää ja jakaa analysoitua informaatiota loppukäyttäjille, eli perinteisesti päätöksentekijöille.

Raportointityökaluilla voidaan muodostaa yhteys haluttuun tietokantaan, ja muodostaa sen datasta erilaisia visualisointeja sekä uudelleen nimetä rivitietojen tunnisteita, jotta lopputulos on mahdollisimman selkeä raportin lukijan näkökulmasta. (Hovi et al. 2009, s. 87) Raportointityökalujen avulla muodostetaan siis datasta merkityksellistä informaatiota ryhmittelemällä datajoukkoja ja esittämällä niiden välisiä suhteita ja vaikutuksia. Kun nämä vielä esitetään visualisesti mahdollisimman selkeästi, niin voidaan tuottaa merkityksellistä informaatiota päätöksenteon tueksi.

Perinteisessä taulukkolaskennassa ja -raportoinnissa on pitkään hyödynnetty Microsoft Excelin työkirjoja. Ne sopivat hyvin taulukkojen muodostamiseen ja esittämiseen, mutta ongelmaksi muodostuu taustatiedon oikea-aikainen integrointi työkirjaa varten ja työkirjan tallentaminen ja jakaminen. Nykyaikaisten raportointityökalujen taustalla ja pohjalla olevat liiketoimintatiedon hallinnan ratkaisut auttavat tiedon integroinnissa ja tehokkaassa tallentamisessa ja jakamisessa. Liiketoimintatiedon hallinnan avulla voidaan myös olla varmoja, että informaatio on oikeellista ja se on oikea-aikaisesti saatavilla. (Hovi et al. 2009, ss. 86-87) Nykyaikaiset raportointityökalut mahdollistavat lisäksi

taulukointia laajemmat tiedon esittämismuodot, paremman pääsyn raportointirajapintaan sekä rajapinnan ja siihen pääsyn kontrolloinnin (Few 2009, s. 3).

Perinteisesti raportointitarve on ennalta määrätty ja raportointitarpeisiin voidaan vastata vakioraporttien avulla (Turban et al. 2008, s. 96). Yleisin raporttimuoto on parametrisoidut raportit. Käyttäjille rakennetaan joukko vakiomuotoisia raportteja, jotka sisältävät runsaasti parametreja, joita muuttamalla käyttäjä saa raportin esimerkiksi haluamaltaan ajanjaksolta ja tietyltä liiketoiminnan alueelta. (Hovi 1997, s. 91) Parametrit toimivat hakuehtoina informaatiolle, jota raportti esittää, joten yksi vakioraportti voi palvella montaa eri käyttäjää ja näkökulmaa (Hovi et al. 2009, s. 90). Vakioraporttien raportointiväli voi olla vuosi-, kuukausi-, viikko-, päivä- tai jopa tuntitasoinen. Raportointiväli riippuu raportin sisällön vaikutuksesta liiketoimintaan ja sen johtamiseen. Nopea raportointitahti tarkoittaa, että raportoidun informaation on oltava reaaliaikaista ja tarvitaan reaaliaikainen yhteys tietolähteeseen. Vakioraportit esitetään yleensä taulukkomuotoisina ratkaisuina, tai visualisoituina kuvaajina ja pylväs- tai viivakaavioina (Few 2012, s. 1). Raportoinnin visuaalinen esitysvoima on käsitelty tarkemmin luvussa 2.4.2.

Vakioraporttien lisäksi organisaatiossa on usein tarve luoda myös ennalta määrittelemättömiä raportteja, joita voidaan luoda tarvittaessa nopeasti (Turban et al. 2008, s. 96). Ennalta määrittelemättömiä raportointitarpeita voivat olla esimerkiksi nopea kyselytarve tai päätöksentekijän intuition vahvistaminen. Liiketoimintatiedon hallinnan ratkaisujen avulla voidaan luoda ensin tietovarasto, josta voidaan koostaa reaaliaikaisia raportteja käyttäjille tai julkaista tietolähteitä, joiden pohjalta käyttäjät voivat itse luoda tarvitsemiaan raportteja. Nykyaikaiset raportointityökalut mahdollistavat loppukäyttäjien toteuttamien, ennalta määrittelemättömien raporttien nopean ja helpon luonnin. (Hovi 1997, s. 90; Hovi et al. 2009, s. 86) Tällöin on tärkeää, että raportointityökalut pystyvät hyödyntämään reaaliaikaista informaatiota ja käyttöliittymä on helposti saavutettava. Saavutettavuutensa puolesta selainkäyttöliittymä on helppo ja suosittu vaihtoehto, jota nykyaikaiset raportointityökalut hyödyntävät. (Hovi et al. 2009, s. 87)

Raportointityökaluja on useita erilaisia eri toimittajilta. Perinteiset raportointityökalut pitävät yhä pintansa markkinoilla, mutta markkinoille astuu jatkuvasti uusia, innovatiivisia toimijoita, jotka tarjoavat moderneja työkaluja moderneihin tarpeisiin. Perinteisinä raportointityökaluina pidetään esimerkiksi Cognos-työkalua sekä BusinessObjectsia (Hovi et al. 2009, s. 118). Uudempia tulokkaita ovat esimerkiksi visualisuuteen panostava QlikTechin QlikView ja Qlik Sense, sekä monipuolinen Tableau ja avoimen lähdekoodin Pentaho. Myös suuret pilvipalveluita tarjoavat toimijat ovat vasta esitelleet omat raportointityökalunsa, jotka on erityisesti suunnattu heidän omien pilvipalveluidensa päälle, tästä esimerkkeinä ovat Microsoftin Power BI ja Amazonin QuickSight (Microsoft 2015; Amazon 2015).

2.3.1 Raportoinnin tarpeellisuus ja hyödyntäminen

Raportointityökalut ovat tärkein liiketoimintatiedon hallinnan ratkaisuissa esiintyvä tiedon hyödyntämismuoto (Hovi et al. 2009, s. 87). Raportointityökalut ovat hyödyllisiä vertailujen tekemisessä, trendien ja kaavojen esittämisessä ja analysoinnissa sekä historiatiedon sekä ajankohtaisen informaation esittämisessä päätöksentekijöille (Thierauf 2001, s. 4). Raportoinnin avulla saadaan liiketoimintatiedon hallinnan prosessissa syntynyt analysoitu informaatio esitettyä tavalla, joka tukee tietämyksen syntymistä ja kehittymistä mahdollisimman hyvin.

Raportointityökalut tukevat organisaation päätöksentekoprosessia. Raportoinnin avulla pyritään perinteisesti esittämään organisaatiolle tärkeitä liiketoiminnan lukuja, jotta päätökset voidaan perustaa tietoon (Pirttimäki 2007, s. v). Raportointi siis tukee vahvasti tiedolla johtamista. Muun muassa operatiivinen johto suorittaa seurantaa yrityksen tehokkuudesta, jolloin tarvitaan tietoa lyhyen aikavälin aikaansaannosten suhteesta käytettyihin voimavaroihin (Thierauf 2001, s. 66). Raportointia voidaan tehdä takautuvasti ja seurata esimerkiksi kumulatiivista liikevaihtoa, tai ennustavasti ja perustaa liikevaihtoennuste esimerkiksi viime vuoden toteutuneeseen liikevaihtoon (Hovi et al. 2009, s. 111). Raportointia voidaan tehdä päätöksenteon tueksi ja tulosten seuraamiseksi, ja hyödyntää saavutettua tietämystä liiketoiminnan ohjauksessa (Hovi et al. 2009, s. xii). Toisaalta raportointityökaluja voidaan hyödyntää myös ennakoivassa päätöksenteossa. Historiatiedon ja ennusteiden pohjalta voidaan luoda tietoa lisääviä raportteja, jotka ennakoivat muutoksia, ja jotka on helppo havaita graafisesta esityksestä.

Raportointityökalut eivät tue ainoastaan organisaation sisäistä päätöksentekoa ja johtamista, vaan myös ulkoisten tahojen tarpeita. Yritys on pääomahuoltonsa suhteen usein riippuvainen ulkopuolisista tahoista kuten rahoittajista ja sijoittajista. Tämän vuoksi sen on säännöllisesti julkaistava kaikki keskeiset tunnuslukunsa, jotka koskevat sen tuloja, menoja ja pääomaliikettä. (Ståhle & Grönroos 1999, s. 59) Raportointityökalujen avulla näistä tiedoista voidaan tehdä julkaistavia raportteja. Osakeyhtiöillä ja julkishallinnon organisaatioilla on myös lain asettama raportointivelvollisuus, joten osakeyhtiöille tunnuslukujen julkinen raportointi ei ole vapaaehtoista vaan lain asettama velvollisuus (Hovi et al. 2009, s. xii). Ulkoisille toimijoille raportoitaessa tulee kiinnittää erityistä huomiota tiedon oikeellisuuteen, sillä kyseessä on yrityksen maine sen ulkoisten toimijoiden silmissä ja lisäksi virheellisten tietojen ilmoittaminen voi johtaa oikeustoimiin (Hovi et al. 2009, s. xiii).

Systemaattinen tiedon analysointi, kommunikointi ja varastoiminen vähentävät päällekkäisen tiedon hankintaa ja käsittelyä yrityksessä (Hannula et al. 2002, s. 86). Raportoinnin avulla voidaan muodostaa dokumentteja organisaation tietyistä tietosisällöistä ja julkaista dokumentit sopivassa mediassa, jotta kaikki sitä tarvitsevat saavat pääsyn informaatioon, eikä jokaisen tarvitse erikseen etsiä informaatiota organisaation tietovarannoista (Few 2009, s. 11). Raportointiprosessin tehokas

toteuttaminen kohentaa yrityksen kilpailukykyä ja lisää liiketoiminnan tuloksellisuutta (Hannula et al. 2002, s. 86). Raportointia voidaan hyödyntää myös organisaation strategian luonnissa ja seurannassa. Johdon vertaillessa esimerkiksi edellisen kauden myyntiraporttia nykyiseen tai uusimpaan raporttiin, vertailu muodostaa tietämystä, jota hyödynnetään strategisella tasolla muun muassa strategian muodostamisessa, toteuttamisessa ja seuraamisessa. (Thierauf 2001, s. 10) Strategiaa varten tarvitaan informaatiota sekä sisäisestä että ulkoisesta toimintaympäristöstä ja tiedon on oltava historiatiedon lisäksi myös tulevaisuutta ennakoivaa. Tulevaisuutta ennakoivaa näkökulmaa raportoinnissa voidaan käyttää myös muuhun suunnitteluun ja ennustamiseen sekä esimerkiksi budjetin luomiseen (Hovi et al. 2009, s. 111).

Raportointityökalut kehittyvät jatkuvasti ja tuovat myös lisää mahdollisuuksia raportoinnin hyödyntämiseen myös muissa yhteyksissä kuin puhtaasti päätöksentekoprosessissa, suunnittelussa tai ulkoisten tekijöiden kanssa kommunikoinnissa. Raportointityökalujen avulla voidaan esimerkiksi tarkistaa kehitysvaiheessa olevan tietokannan datan tila. Raportointityökaluista esimerkiksi Tableau ehdottaa suoraan raportin mallia sille osoitetun datan pohjalta. Yhdistämällä Tableau siis tietokantaan, voidaan datasta muodostaa raportti myös ilman varsinaisia käyttäjän raportointitaitoja. Tableaun avulla on myös mahdollista tarkastella rivitasoista dataa yhteysikkunan avulla. Näin voidaan esimerkiksi helposti tarkistaa, onko tietokannassa ylipäätään dataa ja mille suuruusluokalle se summautuu. Tämä on erittäin hyödyllistä silloin, kuin tietokantaan ei ole olemassa yksikäsitteistä kyselykieltä tai käyttäjä ei osaa itse kirjoittaa kyselyitä. Raportointityökalujen avulla voidaan mahdollistaa siis tietokannan datan tarkastelu ja testaus.

2.3.2 Graafinen esitystapa

Sanonta ”Yksi kuva kertoo enemmän kuin 1000 sanaa” pätee myös raportoinnin tapauksessa. Liiketoiminnan tilaa voidaan esittää tuhansin sanoin tai merkein, mutta yksi asian selittävä kuva on huomattavasti ymmärrettävämpi ja vetoavampi loppukäyttäjän näkökulmasta. Laajat taulukot ja dokumentit eivät siis ole yleensä loppukäyttäjän kannalta mielekkäimpiä, vaan informaation graafinen esitystapa kuvana vetoaa enemmän. Graafisen esitystavan avulla voidaan myös helpommin esittää informaation välisiä suhteita ja esimerkiksi trendit saadaan helposti näkyville. (Turban et al. 2008, s. 106) Graafisen esitystavan etuna on myös se, että tärkein informaatio saadaan välitettyä loppukäyttäjille tavalla, jonka avulla tärkein informaatio voidaan nähdä vain yhdellä vilkaisulla. Tämä säästää huomattavasti aikaa ja vähentää monitulkintaisuutta, jota sanallinen esitystapa saattaisi aiheuttaa. (Few 2006, s. 34) Monitulkintaisuuden välttäminen on tärkeää, sillä raportoinnin muodon on oltava yksiselitteinen, jotta sen avulla voidaan luoda turvallisuutta ja vakautta organisaatioon. (Ståhle & Grönroos 1999, ss. 140-141)

Raportoinnin tarkoituksena on saattaa analysoitu informaatio loppukäyttäjille mahdollisimman tehokkaasti ja todellista lisäarvoa tuottavasti (Hovi et al. 2009, s. 87). Tehokkuudeksi mielletään usein nopeus ja selkeys, ja siksi raportointityökalut ja niiden mahdollistamat informaation visualisoinnit sopivat informaation esittämiseen. Raportointityökalujen avulla voidaan luoda erilaisia graafisia esityksiä informaatiosta, jolloin rivitasoinen informaatio saadaan selkeästi esitettyä osana isompaa kokonaisuutta (Turban et al. 2008, s. 106). Graafiset esitykset ovat tehokkaita, sillä näkökyky on tehokkain ja vaikutusvaltaisin aistimme, kun ajatellaan informaation saamista ympäröivästä maailmasta. Näkemisen ja ymmärtämisen välillä on intiimi yhteys, jota hyödyntämällä voidaan tehokkaasti kasvattaa ymmärrystä. (Few 2012, s. 61) Ymmärrystä tulisi kuitenkin tukea mahdollisimman yksinkertaisella tavalla. Jos halutaan tarkastella yksittäistä lukua, on se tietysti helpoin esittää numerona, sillä kuvaaja saattaa vain hämmentää lukijaa ja viedä huomiota itse luvulta. Asioiden vertailussa kuvaaja puolestaan tuo lisäarvoa ja helpottaa lukijan tulkintaa. (Few 2006, s. 119)

Informaation esittämisessä on tärkeää löytää ja valita oikea esitysmuoto. Vakioitujen taulukoiden ja kuvaajien käyttö on yleinen käytäntö nykypäivän organisaatioissa kommunikoitaessa kvantitatiivista informaatiota. (Few 2012, s. 1) Toisaalta kaikkea tietoa ei välttämättä saada koottua yhteen taulukkoon, tai taulukon lukeminen on liian hidasta ja haastavaa. Jokainen lukutaitoinen osaa kyllä lukea taulukon rivejä ja sarakkeita, mutta tärkein informaatio saattaa jäädä piiloon tai tulkinnanvaraiseksi suuressa taulukossa. Yhdelle suurelle taulukolle vaihtoehdoksi voidaan harkita useampaa taulukkoa, yhtä tai useampaa kuvaajaa tai näiden yhdistelmää. (Few 2012, ss. 42-43) Toisaalta jos tietotarpeena on osoittaa yksi, määritelty luku ja sen vertailukohdat, voidaan hyödyntää myös visuaalisia mittaristoja, joiden avulla tärkein informaatio voidaan nähdä yhdellä vilkaisulla (Few 2006, s. 34). Olennaista raportointimuodon kannalta on siis ymmärtää tietotarve, johon raportilla halutaan vastata, tämän informaation luonne sekä kohderyhmän tarpeet ja toiveet, jotta raportista saadaan mahdollisimman selkeä ja tarkoituksenomainen (Few 2006, s. 119).

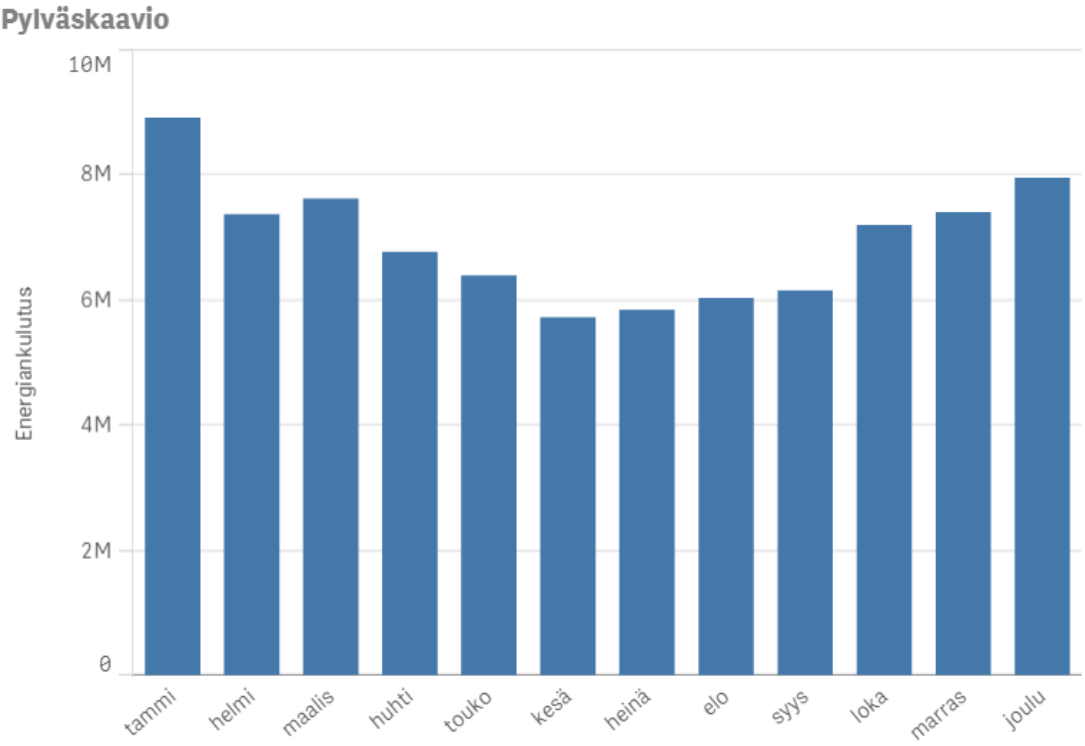
Taulukot mahdollistavat useiden tarkkojen arvojen esittämisen samanaikaisesti. Taulukot toimivat siis hyvin yksittäisen tai kahden asian tarkassa vertailussa sekä tarkkojen arvojen tarkistamisessa. Taulukoissa voidaan myös tehokkaasti esittää samalla sekä yksityiskohtainen että summatieto. (Few 2012, ss. 45-51) Taulukoiden informaatiosisältö saattaa kuitenkin helposti laajentua liian suureksi, jolloin on vaarana, että tärkein informaatio hautautuu muun informaation alle. Tällöin voidaan esimerkiksi kaventaa tietosisältöä ja luoda poikkeamaraportteja, jolloin raportille ilmestyy ainoastaan normaalista rajasta poikkeavat arvot. (Hovi 1997, ss. 90-91) Taulukon etu on siis täsmällisten arvojen esittämisessä ja vertailussa (Karjalainen & Karjalainen 2009, s. 10). Esimerkki raportointityökalu Qlik Sensen avulla muodostetusta taulukosta on esitetty kuvassa 2.2.

Taulukko

kk	Energiankulutus
Totals	**83416339**
tammi	8920608
helmi	7373299
maalis	7624441
huhti	6773655
touko	6393750
kesä	5726887
heinä	5846459
elo	6034304
syys	6154451
loka	7200071
marras	7409844
joulu	7958571

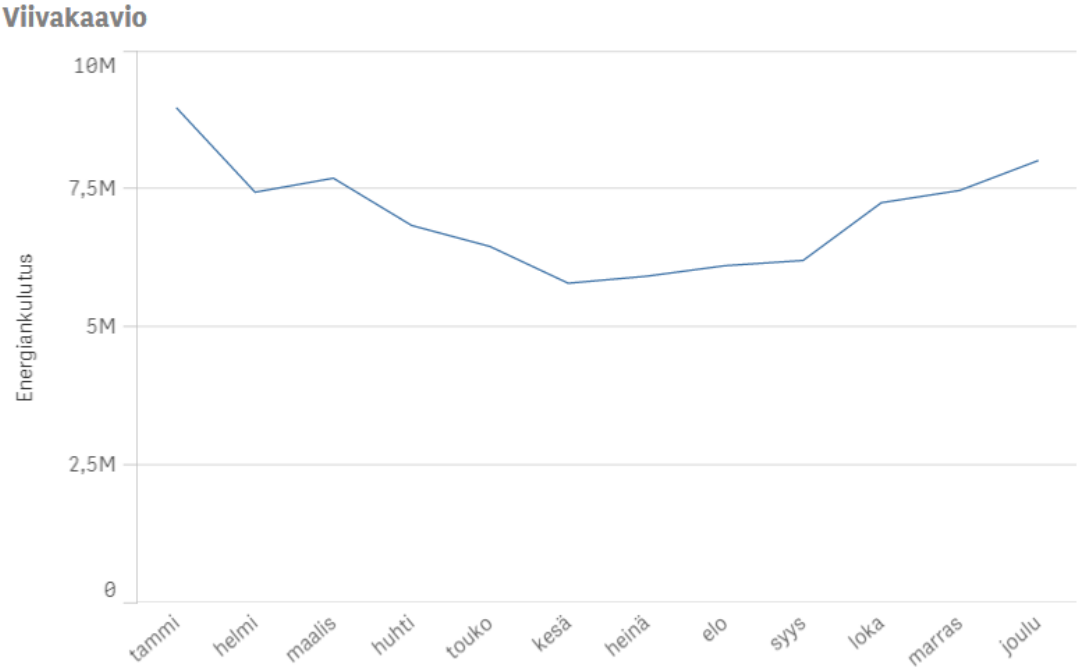
Kuva 2.2. *Energiankulutusta kuvaava taulukko*

Kuvan 2.2 taulukossa on kuvattu energiankulutusta kuukausitasolla. Taulukkomuodosta on helppo tarkistaa kulutuksen määrä kuukausi- ja vuositasolla. Vaikka taulukot perustuvat informaation selkeään esittämiseen, on niiden pohjalla kuitenkin oletus, että lukija osaa lukea ja tulkita numeroita. Erilaisissa kuvaajissa puolestaan riittää, että lukija käyttää vain näköaistiaan, ja siksi ne ovat tehokkaita. (Few 2012, s. 61) Perinteisiä kuvaajia ovat pylväskaaviot ja viivakaaviot. Pylväskaavio on tehokas keino havainnollistaa arvojen suuruuksia ja niiden muutoksia visuaalisina pylväinä, joiden kokoja voidaan helposti vertailla näkökyvyn avulla. Pylväskaaviosta on helppo nähdä suurin ja pienin arvo ja arvojen suuruusluokat on helppo tarkistaa. (Karjalainen & Karjalainen 2009, ss. 18-19) Esimerkki pylväskaaviosta on esitetty kuvassa 2.3.



Kuva 2.3. *Energiankulutusta kuvaava pylväskaavio*

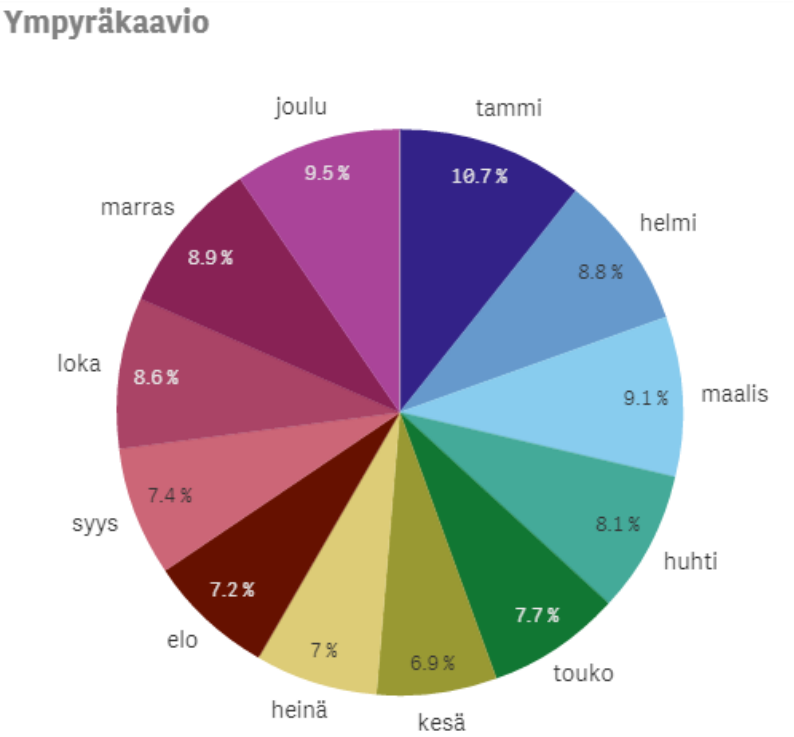
Kuvan 2.3 pylväskaavio havainnollistaa hyvin kuukausittaisia suuruusluokkia energiankulutukselle ja siitä pystytään myös näkemään kuukausittaista trendiä. Viivakaavio on kuitenkin tehokkaampi muoto esittää kehityspolkuja ja trendejä. Viivakaavion vaaka-akselilla on oltava tasavälisiä arvoja, kuten aikavälejä. Kun pystyakselille lisätään jatkuvaluonteinen ominaisuus, voidaan muodostaa viivakaavio, josta nähdään jatkuvaluonteisen ominaisuuden kehittyminen. Viivakaavion avulla voidaan siis luoda kehityskäyrä, jonka avulla voidaan seurata määrien tai muutosten kehitystä tasavälisellä ajalla. (Karjalainen & Karjalainen 2009, s. 32; 36) Esimerkki viivakaaviosta on esitetty kuvassa 2.4.



Kuva 2.4. *Energiankulutusta kuvaava viivakaavio*

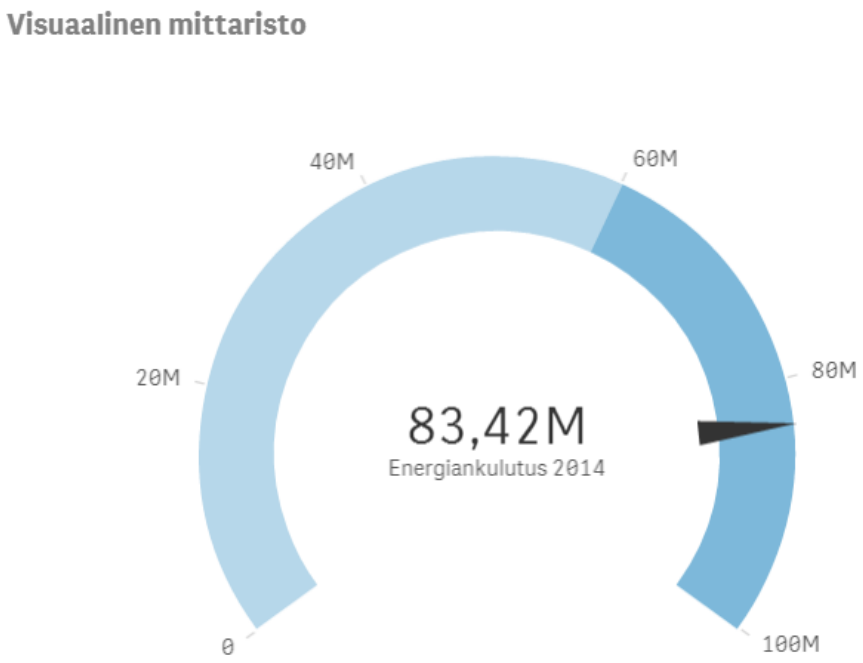
Kuvien 2.3 ja 2.4 pohjalla on täsmälleen sama data, energiankulutuksen määrä kuukausitasolla. Pylväskaaviosta voidaan helposti seurata kulutuksen määrää ja tarkistaa yksittäisen kuukauden kulutuksen määrä. Pylväiden kokoja voidaan myös vertailla keskenään ja näin huomata esimerkiksi, että kesällä kulutus on pienempää kuin talvella. Toisaalta viivakaaviosta nähdään heti kulutuksen kehittyminen kuukausittain. Viivakaaviosta voidaan heti huomata, että kulutuksen käyrä on laskeva kesää kohden ja nouseva talvea kohden. Vaikka molemmat kuvaajat siis esittävät saman informaation, on toisesta helpompi lukea kuukausittaiset arvot ja toinen taas esittää selkeämmin kulutuksen vaihtelun käyrän. Näin voidaan siis todeta, että molemmat esitystavat ovat hyödyllisiä, ja esitystapa tulisi valita tietotarpeen mukaan eli halutaanko lukea pääasiassa kuukausittaisia lukuja vai seurata kulutuksen kehitystä käyrältä.

Saman energiankulutuksen informaation pohjalta voidaan myös muodostaa esimerkiksi ympyräkaavio. Ympyräkaaviota voidaan kutsua myös sektorikaavioksi tai piirakaksi ja se on suosittu kaaviotyyppi etenkin suurelle yleisölle tarkoitetuissa esityksissä. Kaaviossa kokonaisuuden muodostava ympyrä jaetaan sektoreihin, jotka kuvaavat kokonaisuuden osia. Ympyräkaavio on siis melko epätarkka tiedon välittäjä, joka kuvaa yleensä suhteellisia osuuksia. (Karjalainen & Karjalainen 2009, s. 27) Energiankulutuksen ympyräkaaviosta voidaan nähdä kunkin kuukauden prosentuaalinen suuruus verrattuna koko vuoden energiankulutukseen. Näin voidaan esimerkiksi helposti huomata, että tammikuussa muodostuu yli 10 prosenttia koko vuoden kulutuksesta. Ympyräkaaviosta ei voida nähdä tarkkoja lukuja, ja kehityksenkin huomaaminen on haastavampaa, mutta aikavälin kulutuksen suhde koko vuoden kulutukseen on yksikäsitteinen ja helposti ymmärrettävissä. Ympyräkaavio on esitetty kuvassa 2.5.



Kuva 2.5. *Energiankulutusta kuvaava ympyräkaavio*

Kuvan 2.5. ympyräkaavio ei ole kovin informatiivinen, sillä se esittää ainoastaan kuukausittaisia osuuksia kokonaisuudesta ja jos prosenttimerkinnät otettaisiin pois, ei kaaviolla olisi juuri informaatioarvoa. Informaatioarvoltaan yksinkertainen kaavio on myös visuaalinen mittaristo. Jos halutaan seurata koko vuoden energiankulutuksen muodostumista suhteessa tavoiteltuun kulutukseen, voidaan muodostaa visuaalinen mittaristo, josta voidaan yhdellä silmäyksellä huomata nykyinen tulos suhteessa tavoiteltuun. Visuaalinen mittaristo muistuttaa auton kojelautaa, ja siitä voidaan helposti huomata tärkeät tunnusluvut. (Hovi et al. 2009, s. 95) Esimerkki visuaalisesta mittaristosta on esitetty kuvassa 2.6, jossa mittarina on koko vuoden energiankulutus. Mittari kuvaa asetettuja energiankulutuksen raja-arvoja ja toteutuneen kulutuksen suhdetta niihin.



Kuva 2.6. *Energiankulutusta kuvaava visuaalinen mittaristo*

Kuvien 2.2-2.6 taustalla on sama data, mutta se on esitetty ja summattu eri tavoin. Näin voidaan siis huomata, että sama data voi palvella montaa tarvetta, kunhan se osataan esittää analysoituna tietotarvetta palvelevalla tavalla. Jos tietotarpeena on esimerkiksi tietää, missä kuussa kulutus on suurinta, harva valitsisi taulukkoa, jossa käyttäjän on vertailtava kaikkien rivien informaatiota keskenään. Yhdellä vilkaisulla tämä tieto saadaan esille pylväskaaviosta, josta voidaan heti nähdä suurin pylväs. Paras informaation esitystapa määräytyy aina tarpeiden mukaan (Karjalainen & Karjalainen 2009, s. 36). Esitystapaa valittaessa huomioon on otettava informaation luonne, viestin luonne, kohderyhmän tarpeet ja toiveet, sekä mielipiteet ja preferenssit. Graafisia esitystapoja on useita ja tärkeintä on aina valita loppukäyttäjälle sopiva esitysmuoto. (Few 2006, s. 119)

3. NOSQL-TIETOKANNAT

Tutkimuksen kannalta on tärkeää ymmärtää, mitä NoSQL-tietokannat ovat, miksi ne ovat kehittyneet, ja mitkä ovat niiden erityispiirteet. Luku kolme keskittyy selvittämään NoSQL-tietokantoja kirjallisuuden avulla. Ensin käydään läpi NoSQL-tietokannan kehittymistä ja siihen vaikuttaneita tekijöitä, jonka jälkeen esitellään erityyppiset NoSQL-tietokannat.

3.1 NoSQL-tietokantojen kehittyminen

Toimintaympäristömme on jatkuvassa muutoksessa ja muutosnopeus on ennen näkemätöntä. Toimintaympäristön digitalisaation myötä tiedolla on yhä suurempi merkitys toiminnan mahdollistajana ja kehittäjänä, ja usein käytetään vertausta, että tieto on ”uusi öljy” ja tietokannat ovat tämän ”uuden öljyn” öljykenttiä, poria ja pumppuja (esimerkiksi Redmond & Wilson 2012, s. xi; Krishnan 2013, s. 255; Salo 2013 s. 24). Tietokantojen merkitys toimintaympäristössämme siis kasvaa jatkuvasti ja samalla toimintaympäristön muutokset haastavat tietokantoja kehittymään. Tietoja halutaan säilyttää yhä tarkemmalla tasolla ja yhä pidempään. Vaatimukset tietokannoille kasvavat jatkuvasti ja niiltä vaaditaan skaalautuvuutta ja myös strukturoimattoman tiedon tallennuskykyä (Hovi et al. 2009, s. xiii).

Relaatiomalliin perustuvat relaatiotietokannat ovat tietokantojen perinteinen muoto ja ovat vastanneet hyvin tiedon varastoinnin tarpeisiin lähes 30 vuoden ajan (Blanco 2013, s. 10; Altrafi et al. 2014). Relaatiotietokantojen teoreettinen pohja perustuu matematiikan joukko-oppiin. Relaatiotietokantojen vahvuus on strukturoitu kyselykieli eli SQL (Structured Query Language). SQL ei ainoastaan mahdollista kyselyjä, vaan myös tietokannan rakenteen määrittelyn ja muuttamisen, päivitykset, tapahtumakäsittelyn ohjaamisen, valtuuksien ja turvallisuuden hoitamisen sekä rajapinnat ohjelmointiin. (Hovi 2008, s. 5; 14) Vielä vuonna 2008 Hovi (2008, ss. 2, 265-266) totesi, että lähes kaikki tietokantatoteutukset perustuvat SQL-kieleen, eikä sille edes yritetä etsiä vaihtoehtoja, joten tulevaisuuteen voi panostaa opettelemalla SQL-kieltä ja relaatiomallin, sillä niille ei ole näkyvissä mitään tulevaa kilpailijaa.

Muutos on ollut kuitenkin erittäin nopeaa ja vaatimukset tietokannoille ovat kasvaneet lisääntyneen datamäärän johdosta. Kun yritykset kasvavat ja niiden tietotarpeet ja -varannot kasvavat, tulevat heidän nykyisten tietokanta-arkkitehtuuriensa rajat vastaan. Erityisesti relaatiotietokannan tapauksessa arkkitehtuurin laajentaminen on haasteellista datan määrän merkittävässä kasvussa. (Thierauf 2001, s. 121; Strauch 2011, s. 3) Relaatiotietokantojen tilalle ja rinnalle on siis täytynyt kehittää uudenlaisia, paremmin suureen datamäärään skaalautuvia ratkaisuja, joita on alettu kutsua NoSQL-

tietokannoiksi. NoSQL-termi valittiin löyhästi kuvaamaan ei-relaatiomallisia tyyppisiä tietokantoja, jotka eivät yleensä käytä SQL-kieltä. Termi on kuitenkin laajentunut tarkoittamaan ”Not only SQL” eli ”ei ainoastaan SQL”, sillä osa NoSQL-tietokannoista hyödyntää jossain määrin myös SQL-kieltä. (Strauch 2011, s. 1; Pokorny 2013; Planet Cassandra 2015)

NoSQL-tietokanta on ei-relaatiomallinen ja laajaa hajautusta tukeva tietokanta, joka mahdollistaa massiivisen ja keskenään erilaisen datamassan nopean ja ennalta määrittelemättömän järjestelyn ja analysoinnin. NoSQL-tietokannat on suunniteltu laajamittaiseen datan varastointiin ja massiivisen datajoukon käsittelyyn horisontaalisesti monella eri palvelimella samanaikaisesti. (Moniruzzaman & Hossain 2013; Planet Cassandra 2015) NoSQL-tietokannoista on tullut ensimmäinen varteenotettava vaihtoehto relaatiotietokannoille, sillä ne pystyvät vastaaman skaalautuvuuteen, saatavuuteen ja viansietokykyyn liittyviin haasteisiin (Xiang et al. 2010; Planet Cassandra 2015). NoSQL-tietokantojen hyödyntäminen on yleistynyt muutaman viime vuoden aikana. Tekijöitä ja tarpeita, jotka ovat vaikuttaneet NoSQL-tietokantojen syntyyn ja yleistymiseen ovat:

1) big data
2) tarve tietokannan skaalautuvuudelle
3) tarve jatkuvalle datan saatavuudelle
4) tarve joustaville datamalleille
5) suurten datamassojen analysointi- ja varastointitarpeet (Altrafi et al. 2014; Planet Cassandra 2015).

Valtavaa datan määrää ja monimuotoisuutta, sekä niiden nopeaa kasvua kuvataan termillä big data (Hurwitz et al. 2013, s. 14; Salo 2013; Anuradha & Ishwarappa 2015). Big data eli valtavat datamassat ovat olleet tärkein tekijä NoSQL-tietokantojen kehittymisen taustalla (Planet Cassandra 2015). Big data -ilmiö on aiheuttanut tarpeet skaalautuvuudelle, saatavuudelle, joustavuudelle ja analysointi- ja varastointitarpeille. Kaikki tekijät ja tarpeet on esitetty luvuissa 3.1.1-3.1.4

3.1.1 Big data

Jatkuvasti muuttuva ja kehittyvä toimintaympäristö ohjaa toimintaamme kohti digitalisaatiota. Digitaalinen maailma ja sen kompleksisuus kasvavat jatkuvasti kiihtyvällä nopeudella. Kompleksisuus kasvaa datan jatkuvasti moninkertaistuvan määrän, moninaisuuden ja lisääntymisnopeuden vuoksi (Moniruzzaman & Hossain 2013; Salo 2013). Tätä ilmiötä nimitetään yleisesti big dataksi, jolla siis tarkoitetaan suurten, järjestelemättömien ja jatkuvasti lisääntyvien datamassojen hallintaa (Hurwitz et al. 2013, s. 14). Big data ilmiössä haetaan Salon (2013, s. 24) mukaan vastausta ongelmaan: ”Miten siirtää, tallentaa, tarvittaessa yhdistää, monipuolisesti analysoida ja ennen kaikkea tehokkaasti hyödyntää kaikkea käsillä olevaa dataa”. Jotta kaikkea käsillä olevaa dataa

voidaan hyödyntää, on ensin pystyttävä siirtämään, tallentamaan ja analysoimaan se. Suorituskyvyltään tehokkaat NoSQL-tietokannat ovat syntyneet vastaamaan tähän big data -haasteeseen.

Big datalla tarkoitetaan valtavia määriä ryhmittelemätöntä dataa, jota muodostuu huipputehoisten sovellusten pyörittämisen ohella ja tuloksena. Näitä sovelluksia ovat esimerkiksi tieteelliset laskentajärjestelmät, sosiaalinen media ja e-palvelut, kuten sähköinen terveydenhoitojärjestelmä. Näillä järjestelmillä on yhteistä se, että niissä on laajamittaista dataa, skaalautuvuus on ongelma ja datan saattaminen informaatioksi ja sen esittäminen voi olla haasteellista, mutta lopputulokset ovat palkitsevia. (Cuzzocrea et al. 2011; Chen et al. 2012; Anuradha & Ishwarappa 2015) Big dataa synnyttävät sekä koneet että ihmiset. Ihmiset luovat ja julkaisevat digitaalista sisältöä, kuten videoita ja sosiaalisen median päivityksiä, ja näistä muodostuu massiivinen määrä dataa. Sosiaalisen median palveluita pidetäänkin yhtenä pääsyynä big data -ilmiön muodostumisessa. Ihmisen lisäksi myös koneet luovat dataa, jota ei synny ainoastaan tietokoneiden avulla, vaan myös esimerkiksi hyvin perinteiset teollisuuskoneet voivat kerätä dataa niihin asetettujen anturien avulla tai sääasemat voivat olla suoraan yhteydessä tietokantaan, johon voidaan tallentaa mittarilukemia tietyin aikavälein. (Salo 2013, ss. 20-21)

Big dataa luonnehditaan yleisesti kolmen V:n avulla. Big datalle ominaista ovat volyymi (engl. volume), vauhti (engl. velocity) ja vaihtelevuus (engl. variety). (Moniruzzaman & Hossain 2013; Salo 2013, ss. 21-23) Volyymi on big datan merkittävin luonteenpiirre (Anuradha & Ishwarappa 2015). Datan määrä maailmassa kasvaa eksponentiaalisesti, eikä kasvulle näy loppua. Dataa syntyy ja tuotetaan yhä enemmän ja nykyaikaiset, edulliset datan varastointitavat mahdollistavat myös laajemman datankeruun ja tallentamisen. (Salo 2013, ss. 20-23). Datan määrän kasvulle sekä syy että seuraus on myös datan vauhti. Uutta dataa syntyy ja sitä syötetään järjestelmiin kiihtyvällä nopeudella, mikä haastaa tallennetun datan jatkojalostusta ja hyödyntämistä, sillä myös sen on oltava entistä nopeampaa ja tehokkaampaa. (Salo 2013, s. 21) Datan volyymia ja vauhtia kasvattaa myös nykyajan nopea päätöksentekoväli. Tiedon määrä kasvaa, koska nykypäivän organisaatioissa on tehtävä päätöksiä entistä nopeammalla aikataululla ja lyhemmällä aikavälillä. Raportoinnissa on siirryttävä kvartaali- tai kuukausiraportoinnista päivä- tai tuntitasoiseen raportointiin. (Hovi et al. 2009, s. 76)

Nopeampien ja entistä monipuolisempien ja laajempien analyysien teko myös lisää datan moninaisuutta ja vaihtelevuutta. Informaatiota koostetaan useasta eri lähteestä ja integrointimenetelmät ja tallennustietokannat joutuvat koetukselle. (Hovi et al. 2009, s. 76) Datalähteiden monipuolisuus lisää datan heterogeenisyyttä eli data muuttuu entistä vaihtelevammaksi ja siirrytään strukturoidusta datasta myös semi-strukturoituun ja strukturoimattomaan dataan (Salo 2013, ss. 21-23). Strukturoitu data on dataa, jonka täytyy olla järjestettävissä tauluihin, joissa on rivejä ja sarakkeita eli datalla on selkeä rakenne. Relaatiotietokannan datan on siis oltava tällaista ja se voi olla esimerkiksi organisaation yhteystieto- tai myyntidataa. (McFadden et al. 1999, s. 208)

Strukturoimattomalla datalla tarkoitetaan päinvastaisesti dataa, jolla ei ole rakennetta. Strukturoimatonta dataa ovat esimerkiksi videot, pdf-dokumentit sekä sosiaalisen median julkaisut. Strukturoidun ja strukturoimattoman datan välimuoto on semi-strukturoitu data. Esimerkiksi video on strukturoimatonta dataa, mutta kun siihen liitetään avainsanoja, muodostavat avainsanat struktuurin, joten näin videosta tulee semi-strukturoitua dataa. (Salo 2013, s. 25)

Datan heterogeenisyys haastaa erityisesti relaatiotietokantoja, sillä ne on suunniteltu ainoastaan strukturoitua dataa varten. Perinteisen relaatiomaailman tietovaraston teho perustuu pääosin numeerisen ja tiukasti ennalta määritellyn relaatiomuotoisen datan varastointiin sellaiseen muotoon, josta optimoitujen SQL-kyselyjen avulla voidaan nopeasti ja helposti kaivaa tarvittavat tiedot. Big data ei ole välttämättä ole saatavissa tai järjestettävissä relaatiomuotoiseksi dataksi, joka voitaisiin varastoida relaatiotietokantaan ja sieltä hakea optimoiduilla SQL-kyselyillä. (Salo 2013, s. 65) SQL-kieli itsessään ei myöskään sovellu strukturoimattoman ja semi-strukturoidun datan tehokkaaseen käsittelyyn, joten SQL-kieleen perustuville relaatiotietokannoille on kehitetty big datan näkökulmasta tehokkaampi tietokantarakenne, NoSQL-tietokanta (Altrafi et al. 2014; Anuradha & Ishwarappa 2015). NoSQL-tietokantaan voidaan kuitenkin tallentaa myös relaatiomallista, strukturoitua dataa, ja osittain siksi termin NoSQL katsotaan tarkoittavan ”*Not only SQL database”* (Loshin 2013, s. 83; Salo 2013, s. 65).

Big data -ilmiöllä ei tarkoiteta pelkästään massiivista datamäärää ja havaintoa siitä, että dataa on koko ajan enemmän ja sen muoto ja laatu vaihtelevat suuresti. Big data -ilmiöön kuuluvat myös ratkaisut, joilla tuohon haasteeseen tartutaan. (Barbierato et al. 2014; Salo 2013, s. 28) Big data -ratkaisujen täytyy tukea skaalautuvuutta ja datan vaihtelevuutta, ja big data -ilmiön myötä on tullut markkinoille monia skaalautuvia tietokantatyökaluja ja -tekniikoita (Anuradha & Ishwarappa 2015). Datan luonteen vuoksi big data -ratkaisut eivät pääasiassa sovi relaatiotietokantoihin, sillä relaatiotietokannat haluavat strukturoitua tietoa ja niiden skaalautuvuus on rajallista. Big data ja sen sovellukset edellyttävät skaalautuvuutta, datan tehokasta prosessointia ja viansietoa, joihin NoSQL-tietokannoilla on mahdollisuus vastata. (Barbierato et al. 2014)

3.1.2 Skaalautuvuus ja saatavuus

Moninkertaistunut ja edelleen kasvava datamäärä haastaa tietokantoja myös niiden skaalautuvuuden eli laajentumiskyvyn kannalta. Kun dataa syntyy jatkuvasti lisää, pitää tietokantojen myös pystyä vastaamaan tähän kasvuun kasvamalla itse. Kasvun on tapahduttava niin, että se ei aiheuta vikatiloja järjestelmässä tai näy loppukäyttäjälle. (Pokorny 2013) Tästä tullaan toiseen tärkeään tekijään, eli datan jatkuvaan saatavuuteen.

Massiivinen datamäärä ja big data -ilmiö vaativat tietokannoilta skaalautuvuutta, ja yhtenä syynä NoSQL-konseptin ja -tietokantojen syntyyn on ollut perinteisten relaatiotietokantojen skaalautuvuuden riittämättömyys (Altrafi et al. 2014).

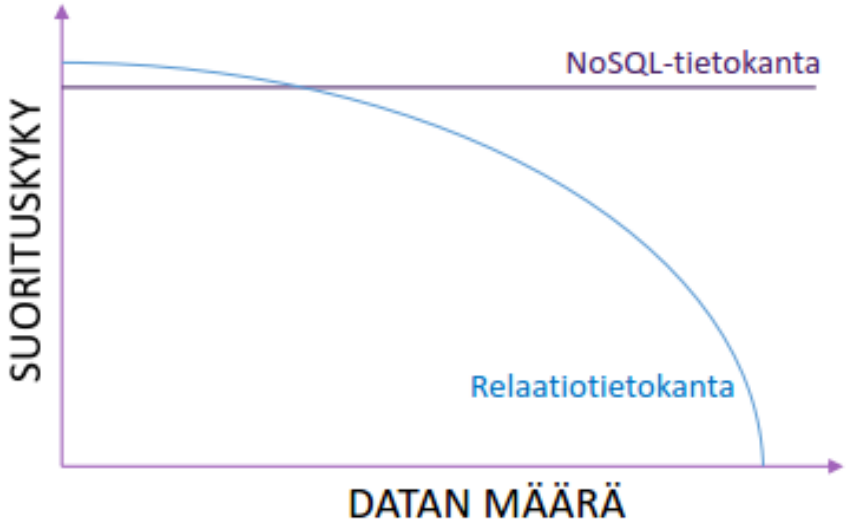
Skaalautuvuudella tarkoitetaan järjestelmän laajennettavuutta eli mahdollisuutta kasvattaa teknistä ympäristöä ilman, että toiminta häiriintyy. Datan määrä kasvaa jatkuvasti eksponentiaalisesti ja tietokantojen suorituskyvyn tulisi kasvaa datan määrän mukana. Tarkkaan strukturoidut relaatiotietokannat eivät tähän pysty, joten on ollut tarpeellista kehittää paremmin skaalautuva ja kustannustehokas vaihtoehto niille. (Altrafi et al. 2014; Salo 2013, s. 20, 65) NoSQL-tietokanta skaalautuu relaatiotietokantaa paremmin, sillä NoSQL-tietokannan datamassat voidaan tehokkaasti partitioida eli jakaa nippuihin ja säilöä niput horisontaalisesti skaalautuvan hajautetun järjestelmän eri osissa (Floratou et al. 2012). Relaatiotietokannan rakenne datan säilyttämiseen on puolestaan erittäin muodollinen, ja taulumainen, joten dataa on vaikea hajauttaa (Leavitt 2010).

Relaatiotietokannat sijaitsevat perinteisesti yhdellä palvelimella, jota voidaan skaalata laajemmaksi lisäämällä prosessoreita, muistia ja ulkoista varastointitilaa. Perinteinen relaatiotietokanta vaatii siis laajentuakseen fyysistä laitteistoa, ja tämä on tehotonta ja kallista. (Pokorny 2013; Altrafi et al. 2014) Nykyajan dataintensiivisillä palveluilla on tarve dynaamisemmin skaalautuville tietokannoille, ja NoSQL-tietokannat ovat tällaisia. NoSQL-tietokannat skaalautuvat lähes lineaarisesti käytettyjen palvelimien määrän mukaan. (Lo 2015) Tämän mahdollistaa se, että tietokannan data voidaan partitioida eli jakaa nippuihin. Tämä horisontaalinen datan jakaminen mahdollistaa laskentaprosessin jakamisen useaksi samanaikaisesi prosessoitavaksi tehtäväksi hajautetun järjestelmän eri osissa. (Pokorny 2013)

Myös NoSQL-tietokanta voi sijaita vain yhdellä palvelimella, mutta yleensä se on suunniteltu toimimaan pilvipalveluympäristössä monen palvelimen päällä hajautetusti. Big datan kanssa pilvipalvelut ovat hyödyllisiä, sillä ne skaalautuvat kohtalaisen helposti ja niiden tallennustila on verrattain halpaa. Lisäksi raakadatan suhteen tietoturvallisuusvaatimukset eivät ole aivan yhtä korkealla tasolla kuin liiketoimintakriittisen informaation tapauksessa. (Cuzzocrea et al. 2011) Hajautetut NoSQL-tietokannat ovat dynaamisesti skaalautuvia. Dynaaminen skaalautuvuus eli palvelimien jatkuva lisääminen tai vähentäminen, on hyödyllistä, sillä silloin voidaan vastata muuttuvaan käyttäjämäärään tai datan volyymiin. Tällöin koneet ja skaalautuvuus on helppoa, tehokasta ja halpaa. Sosiaalisessa verkostoissa ja mediassa dynaaminen ja horisontaalinen skaalautuvuus on tärkeää. (Pokorny 2013) Myös relaatiotietokanta voi jossain määrin skaalautua horisontaalisesti, mutta niitä ei ole rakennettu siihen tarpeeseen, ja niiden horisontaalinen skaalautuvuus on työlästä ja tehotonta, sillä ratkaisuun joudutaan yleensä lisäämään ylimääräisiä ohjelmisto- ja laiteosia (Altrafi et al. 2014).

Tietokannan skaalautuvuus vaikuttaa myös sen tehokkuuteen. Jos suorituskykyä voidaan skaalata suuremmalle määrälle laskentatehoa, on tietokannan suorituskyky myös tehokkaampi. Koska relaatiotietokantojen skaalautuvuus on huono, laskee niiden suorituskyky datan määrän kasvaessa. NoSQL-tietokantojen hyvän skaalautuvuuden vuoksi suorituskyky voidaan pitää lähes vakiona datan määrän kasvaessa, ja siksi

NoSQL-tietokantoja suositaankin suurten datamäärien ja big datan yhteydessä. (Lo 2015) Datan määrän vaikutuksen tietokantojen suorituskykyyn on havainnollistettu kuvassa 3.1.



Kuva 3.1. *Datan määrän vaikutus tietokannan suorituskykyyn (mukailtu lähteestä Lo 2015)*

Kuten kuvasta 3.1. voidaan nähdä, pystyvät NoSQL-tietokannat vastaamaan skaalautuvuutensa vuoksi lisääntyvään datan määrään ilman vaikutuksia suorituskyvyssä. Relaatiotietokantojen skaalautuus on puolestaan huono, joten niitä ei voida skaalata loputtomiin, ja näin ollen relaatiotietokannan suorituskyvyn raja tulee jossain vaiheessa vastaan. Relaatiotietokanta on siis edelleen hyvä ratkaisu tilanteisiin, jossa datan määrä on helposti hallittavissa, mutta tilanteisiin, joissa datan määrä on merkittävästi suurempi tai kasvaa jatkuvasti, on NoSQL-tietokanta skaalautuvuutensa takia usein tehokkaampi ratkaisu.

NoSQL-tietokannat keskittyvät massiivisten datamäärien analyyttiseen prosessointiin ja varastointiin sekä laajentuneeseen laitteistosta riippumattomaan skaalautuvuuteen. (Moniruzzaman & Hossain 2013) Skaalautuvuuden lisäksi nykyaikaisilta tietokannoilta vaaditaan myös datan jatkuvaa saatavuutta. Datan jatkuva saatavuus on erittäin tärkeää nykyajan nopeasti muuttuvilla, dataintensiivisillä markkinoilla. Jos organisaation käytössä oleva palvelin tai palvelimet ovat alhaalla, voivat vaikutukset olla kohtalokkaita sekä liiketoiminnan että organisaation maineen kannalta. Skaalautuvat ja hajautetut järjestelmät takaavat osaltaan yhden saatavuuden määritelmän toteutumisen: data on aina saatavilla, eikä yhden palvelimen kaatuminen vaikuta koko järjestelmän toimivuuteen. Tämä on mahdollista, sillä sama data on tallennettu usealle palvelimelle ja jos yksi hajautetun järjestelmän osa tai palvelin kaatuu, muut osat pystyvät jatkamaan toimintaa ilman datahäviöitä. Viansietokyky on siis erittäin hyvä. Horisontaalinen skaalautuvuus mahdollistaa myös järjestelmän päivittämisen ja modifioinnin ilman, että tietokantaa tarvitsee siirtää offline-tilaan, joten data on todellisesti koko ajan saatavilla ilman päivityskatkoja. (Planet Cassandra 2015)

3.1.3 Joustavat datamallit

Relaatiotietokannan rakenne ja datan tallentaminen on ennalta määritetty. Rakenne perustuu yleisesti kaavioihin eli skeemoihin (engl. schema). Skeema tarkoitta (tiedonhallinnassa) dokumentin rakenteen määritystä, joka esitetään tietokoneen ymmärtämässä muodossa. Skeemassa voidaan määritellä rakenteen lisäksi rakenteille attribuutteja ja niiden arvojoukkoja sekä pakollisuuksia. (Kaario & Peltola 2008, s. 161) Relaatiotietokantoihin voidaan tallentaa ja varastoida ainoastaan rakenteista dataa, joka on muodostettu ennalta annettujen sääntöjen ja suhteiden mukaisesti (Hannula et al. 2002, s. 92). Relaatiotietokannoilla on siis selkeä datamalli, jota sen sisällön tulee noudattaa ja tätä kutsutaan relaatiomalliksi. Relaatiomalli edellyttää, että datan täytyy olla rivi-sarake tyyppisesti järjestettävissä, datan manipuloinnin tulee olla mahdollista SQL-kielen avulla ja on oltava sääntöjä, joilla datan eheyttä voidaan ylläpitää (McFadden et al. 1999, s. 208). Relaatiomallinen datamalli perustuu datatauluihin ja yhteyksiin taulujen välillä. Taulujen väliset yhteydet liittyvät taulujen sarakkeisiin, jotka kuvaavat kategorioita, jotka on yksiselitteisesti järjestetty skeemassa. (Leavitt 2010; Planet Cassandra 2105) Esimerkiksi asiakasnumero-sarake voi löytyä monesta saman tietokannan taulusta, jolloin taulut ovat yhteydessä toisiinsa. Relaatiotietokannassa on siis tärkeää olla ennalta määrätty rakenne, jotta taulujen yhteyssuhteet ja datan eheys säilyvät myös dataa lisättäessä ja muutettaessa. Ennalta määrätty skeema määrittelee millaista dataa tietokantaan voidaan tallentaa ja miten se voidaan järjestää. (Moniruzzaman & Hossain 2013; Planet Cassandra 2015)

Joustamaton relaatiomallinen skeema on taakka erityisesti web-sovelluksille, kuten tekstistä, kuvista, videoista ja kommenteista koostuville blogeille, joiden kaikki sisältö täytyy tallentaa tehokkaasti. Web-sovellukset vaativat joustavaa skeeman määritystä ja relaatiotietokannat eivät pysty vastaamaan tehokkaasti tähän haasteeseen. (Moniruzzaman & Hossain 2013) Ei-relationaalisissa NoSQL-tietokannoissa ei ole ennalta määriteltyä rakennetta eli skeemaa, joka määrittäisi, missä muodossa datan tulisi olla. Tämä mahdollistaa joustavan käytön. (Salo 2013. s. 84) Joustavien datamallien ansiosta NoSQL-tietokantoihin voidaan tallentaa myös strukturoimatonta dataa ja tallennettavan datan muotoa ja yhteyttä muuhun dataan ei tarvitse määrittää etukäteen. (Padhy et al. 2011) NoSQL-tietokannoissa ei pääasiassa tarvitse päättää, mihin tauluun tai fyysiseen sijaintiin data menee, joten tallentaminen on tehokasta eikä vaadi uusien rakenteiden määrittämistä (Leavitt 2010). NoSQL-tietokannoissa on erilaisia joustavia datamalleja riippuen niiden tyypistä. Erityyppiset NoSQL-tietokannat on esitetty luvussa 3.2.

Se, että NoSQL-tietokannoista pääasiassa puuttuu selkeä, ennalta määritetty skeema, tarkoittaa sitä, ettei pystytä hyödyntämään relaatiotietokantojen vahvuutta eli strukturoitua kyselykieltä, SQL:a. NoSQL-tietokannoissa SQL-kielen käyttäminen vaikeutuu tai muuttuu mahdottomaksi ja erityisesti kyselyiden teko muuttuu haasteelliseksi. (Pokorny 2013) SQL-kieli on tehokas ja tunnettu kyselykieli, joten monessa NoSQL-tietokannassa on ohjelmointirajapinta, jonka avulla voidaan kääntää

käyttäjän kirjoittamia SQL-kyselyitä tietokannan omaksi kyselykieleksi (Moniruzzaman & Hossain 2013). NoSQL-tietokannoilla on myös omia, pelkistettyjä kyselykieliään mutta ongelmana on, että kaikki ratkaisut kehittävät omia komentojaan, eikä vaikuta siltä, että yhtä yhteistä kieltä tultaisiin kehittämään NoSQL-tietokantoja varten. Tämän on ongelma siinä mielessä, että käyttäjän on opeteltava uusia kieliä ja, että esimerkiksi raportointityökalujen olisi tuettava useaa eri kieltä. (Pokorny 2013) Tällä hetkellä yleisimmät raportointityökalut on kuitenkin suunniteltu toimimaan ainakin osittain SQL-kielen kanssa, joten NoSQL-tietokantojen SQL-tulkit ovat tärkeässä roolissa raportointia ajatellen (Krishnan 2013, s. 254).

3.1.4 NoSQL-tietokantojen käyttötarkoitukset

Tarve NoSQL-tietokannoille kehittyi dataintensiivisten, big dataa luovien ja käsittelevien suuryritysten tarpeista. Yrityksillä kuten Google, Amazon, Facebook ja LinkedIn oli haasteita käsitellä omaa massiivista ja jatkuvasti kasvavaa datanmääräänsä ja tavalliset relaatiokannat eivät voineet enää vastata haasteisiin. (Xiang et al. 2010; Moniruzzaman & Hossain 2013) Google, Amazon, Facebook ja LinkedIn kehittelivät kaikki omiin tarpeisiinsa oman NoSQL-tietokantansa, joiden jalanjäljissä markkinoille on astunut monia erilaisia NoSQL-tietokantatuotteita. (Pokorny 2013)

NoSQL-tietokannat ovat syntyneet vastamaan erityisesti big data -ilmiöön liittyviin tarpeisiin ja haasteisiin. Big datalle tyypillistä on datan strukturoimaton luonne, eli sitä ei voida varastoida tehokkaasti relaatiomallisiin tietokantoihin. Jopa 80 prosenttia organisaation datasta on strukturoimatonta ja myös strukturoimaton data olisi tärkeää saada mukaan organisaation liiketoimintatiedon hallinnan prosesseihin, jotta sitä voidaan arvioida yhdessä muun liiketoimintadatan kanssa. Liiketoiminnan raportoinnilta, analytiikalta sekä tietovarastoinnilta vaaditaan siis tehokasta taipumista myös strukturoimattomaan dataan. (Salo 2013, s. 65) Datan strukturoimattoman luonteen lisäksi relaatiotietokannat ovat haasteellisia, sillä niiden suorituskyky ei yleensä riitä erittäin suurten datamassojen analysointiin ja niiden suorituskyvyn kasvattaminen skaalaamalla on haasteellista ja kallista (Planet Cassandra 2015). Monimuotoinen liiketoimintatiedon hallinnan ratkaisukirjo sekä big data -analytiikka vaativat siis tehokasta tietokantaa datalleen ja NoSQL-tietokanta tukee näitä tarpeita hyvin (Moniruzzaman & Hossain 2013).

Big data -tyyppinen data, kuten mikä tahansa muukin data, on todella arvokasta vain jos sitä osataan hyödyntää. NoSQL-tietokantaan ei siis ainoastaan kerätä massiivisia määriä dataa, vaan tätä dataa pitäisi pystyä myös hyödyntämään sen jalostamisen avulla. (Salo 2013, s. 21) Big dataa voidaan jalostaa järjestelemällä ja analysoimalla sitä ja vertaamalla analysoitua dataa tunnistettuihin mittareihin ja tätä kutsutaan big data -analytiikaksi (Krishnan 2013, s. 251). Big datan yhdistelyssä ja analysoinnissa voi olla myös mukana innovatiivisia, matemaattisia algoritmeja ja analyyseja, joiden avulla voidaan tehdä ennustavaa päätöksentekoa ennusteiden esittämien skenaarioiden pohjalta (Salo 2013, s.

33). Big data -analytiikkaan kuuluu perinteisen analytiikan lisäksi datan louhinta ja mallintaminen sekä koneoppiminen, joita varten tarvitaan tehokkaita algoritmeja ja tilastollisia malleja. Big data -analytiikassa voidaan käyttää perinteisen strukturoidun datan lisäksi tekstiä, kuvia, audio-materiaalia, videoita ja konedataa. (Krishnan 2013, s. 251) NoSQL-tietokantaa voidaan hyödyntää big data -analytiikassa, sen lisäksi, että sitä hyödynnetään pääasiassa massiivisen datamäärän tallennuksessa sekä datan laajamittaisessa käsittelyssä ja analysoinnissa (Moniruzzaman & Hossain 2013).

Tietokantaan tallennettava data voi olla mitä tahansa organisaation toiminnan kannalta olennaista dataa. Perinteisesti datan lähteenä voi olla omat operatiiviset perusjärjestelmät, omat suunnittelujärjestelmät, sovelluspaketit, ulkopuolisina palveluina ostetut järjestelmät tai kokonaan ulkopuoliset lähteet (Hovi 1997, s. 48). NoSQL-tietokannan data syntyy yhä laajemmalla alueella. NoSQL-tietokantaan varastoitu data voi olla minkä luonteista tahansa. Se voi olla strukturoitua, semi-strukturoitua tai strukturoimatonta. NoSQL-tietokannalle tyypillistä dataa on big data. Big dataa on esimerkiksi sosiaaliseen mediaan tallentuva tieto ja mittalaitesensorien automaattisesti keräämä mittaustieto. (Altrafi et al. 2014) Data voi olla myös ns. varmuuden vuoksi varastoitavaa dataa, jota ei ole järkevää sijoittaa kalliiseen relaatiotietokantaan. Tällaista varmuuden vuoksi varastoitavaa dataa ovat esimerkiksi lokitiedostot ja mitta-antureiden syötteet. Mitta-anturit ovat yksikköhinnaltaan nykyään niin edullisia, että niitä voi sijoittaa esimerkiksi teollisuuskoneisiin ilman, että edes tiedetään miten anturien avulla kerättävää dataa tullaan hyödyntämään. Mutta koska anturi ja NoSQL-pohjainen tallennustila on niin edullista, voidaan dataa joka tapauksessa varastoida tulevaisuuden mahdollisia käyttötarpeita varten. (Salo 2013, s. 13)

NoSQL-tietokantojen kehittämisen taustalla on datamäärän massiivinen lisääntyminen, johon ovat vaikuttaneet kaksi merkittävää trendiä. Ensinnäkin käyttäjät, järjestelmät ja sensorit luovat uutta dataa massiivisia määriä ennen näkemättömällä nopeudella, ja toiseksi Internet, sosiaalinen media ja useat avoimet ja hajautetut tietolähteet luovat monimutkaisia ja laajenevia riippuvuussuhteita. Sen lisäksi, että varastoitava datamäärä on massiivinen, tarvitaan myös sen nopeaa käsittelyä, sekä samanaikaista pääsyä tietoon ja mahdollisuutta kirjoittaa ja lukea dataa samanaikaisesti ja rinnakkaisesti. (Xiang et al. 2010) Näihin massiivisten datamäärien aiheuttamiin ongelmiin on relaatiotietokantojen yhä useammissa tapauksissa vaikea tai mahdotonta vastata sekä datan mallintamisen että horisontaalisen skaalautuvuuden puutteen vuoksi. NoSQL-tietokannat pystyvät vastaamaan paremmin sekä massiivisten datamäärien tallentamiseen että analysointiin. Big data -analytiikan, liiketoimintatiedon hallinnan ja sosiaalisen median sovellusten laskennalliset vaatimukset ja varastointitarpeet ovat asettaneet tarpeillaan markkinaraon NoSQL-tietokannoille. (Moniruzzaman & Hossain 2013) NoSQL-tietokantoihin voidaan tallentaa tätä haastavaa dataa, mutta sen analysoinnissa on käytettävä muita työkaluja. Analysoinnissa voidaan käyttää hyväksi esimerkiksi uudenlaisia koneellisen oppimisen

ratkaisuja tai perinteisempiä liiketoimintatiedon hallinnan työkaluja, kuten raportointityökaluja. (Krishnan 2013, ss. 251-254)

3.2 Erityyppiset NoSQL-tietokantaratkaisut

Koska dataa on monenluonteista, on myös eriluonteisia NoSQL-tietokantoja. NoSQL-tietokantojen kanssa ei päde ”one size fits all” -ajattelu eli erilaisiin tarpeisiin tarvitaan erilaisia NoSQL-tietokantoja. NoSQL-tietokantoja voidaan luokitella eri tavoin, mutta usein ne luokitellaan rakenteensa perusteella kolmeen eri luokkaan:

1) avain-arvo varasto (engl. key-value store)
2) sarakeorientoitunut tietokanta (engl. column-oriented database)
3) dokumenttipohjainen varasto (engl. document-based store). (Leavitt 2010; Moniruzzaman & Hossain 2013)

Avain-arvo varasto on yksinkertaisin NoSQL-tietokanta, joka muodostuu joukoista avain-arvo pareja. Avain on numero- ja/tai kirjaintyyppinen merkki tai jono ja osoittaa suoraan arvoon tai paikkaan, mihin arvo on tallennettu. Arvo voi olla yksinkertainen merkkijono tai monimutkaisempi lista tai kokonaisuus. Dataa voidaan yleensä hakea vain avaimen perusteella, ja silloinkin avaimen vastaavuuden haettuun tulee olla eksaktisti sama. (Loshin 2013, ss. 85-86; Moniruzzaman & Hossain 2013) Arvo voi olla strukturoitu tai strukturoimaton, avaimet auttavat järjestelemään kaiken muotoisen datan. Kun kaikille arvoille on avain, ei tarvita laajaa taulukkorakennetta, kuten relaatiotietokannoissa. Näin säästytään turhilta tyhjiltä kentiltä, eli NULL-kentiltä, sillä avain osoittaa arvoon, eikä riviin, jossa on sarakkeita. (Pokorny 2013) Esimerkki avain-arvo varastosta on havainnollistettu taulukossa 3.1.

Taulukko 3.1. *Avain-arvo varasto (mukailtu lähteestä Loshin 2013, s. 85)*

Avain	Arvo
…	
‘BMW’	{’1-Sarja’, ’3-Sarja’, ’5-Sarja’, ’5-Sarja GT’, ’7-Sarja’, ’X3’, ’X5’, ’X6’, ’Z4’ }
‘Mercedes’	{’A-Sarja’, ’B-Sarja’, ’C-Sarja’, ’S-Sarja’ }
‘Audi’	{’A1’, ’A3’, ’A4’, ’A6’, ’Q3’}
…	

Taulukon 3.1. avain-arvo varasto voisi kuvata autoliikkeen varastodataa. Avain on automerkki, jonka arvona on kyseessä olevalle automerkille saatavissa olevat mallit. Näin ollen, jos kutsutaan esimerkiksi avainta BMW, saadaan yksinkertaisesti vastauksena

listaus kaikista sen saatavilla olevista automalleista. Avain-arvo varastojen yksinkertaisuus tekee niistä erinomaisia ratkaisuja salamannopeille ja laajasti skaalautuville sovelluksille ja sovellusosille, joiden tarvitsee hakea yksittäisiä tietoja nopeasti. Tällaisia sovelluksia ovat esimerkiksi käyttäjä- tai tuoteprofiilin hallinta. (Loshin 2013, ss. 85-86) Avain-arvo varastoja ovat esimerkiksi Amazonin kehittelemä Dynamo ja LinkedInin kehittelemä Voldemort (Moniruzzaman & Hossain 2013).

Avain-arvo pareja voidaan myös yhdistellä ja muodostaa yhdistelmistä kokoelmia. Tällaisia tietokantoja kutsutaan sarakeorientoituneiksi NoSQL-tietokannoiksi. (Pokorny 2013) Sarakeorientoitunut NoSQL-tietokanta tarjoaa mahdollisuuden järjestellä dataa sarakemaisesti niin että yhdelle avaimelle voi olla monta attribuuttia (Moniruzzaman & Hossain 2013). Sarakeorientoituneisiin NoSQL-tietokantoissa dataa ei tallenneta riveinä vaan ne on suunniteltu tallentamaan dataa sarakkeiden lohkoihin (Planet Cassandra 2015). Sarakkeille määritetään sarakeperhe, eli sarakejoukko mihin ne kuuluvat. Sarakeperheet toimivat ikään kuin kategorioina sarakkeille, ja sarakeperheessä voi olla erilaisia ja eri määrä sarakkeita jokaisella rivillä. (Pokorny 2013) Sarakeorientoitunut NoSQL-tietokanta on esitetty taulukossa 3.2.

Taulukko 3.2. *Rivi sarakeorientoituneessa NoSQL-tietokannassa (mukailtu lähteestä Pokorny 2013)*

			Sarakeperhe 'Jälkeläiset'		
Avain	**Aikaleima**	**Sarakkeen nimi**	**1. Lapsi**	**2. Lapsi**	**3. Lapsi**
'nnnnn'	2010	'Eeva Jokinen'	'Joni'		
	2012	'Eeva Jokinen'	'Joni'	'Jenni'	
	2015	'Eeva Jokinen'	'Joni'	'Jenni'	'Jaakko'

Taulukko 3.2. kuvaa sarakeorientoituneen NoSQL-tietokannan yhtä riviä, jolle on yksi avain mutta useita sarakkeita dimensionaalisesti (Pokorny 2013). Taulukko 3.2. esittää sarakeorientoituneen mallin mukaisen kuvauksen henkilön Eeva Jokinen jälkeläisistä tietyllä ajanhetkellä. Sarakeperheeseen lisätään uusia sarakkeita, sitä mukaan kun niitä tarvitaan, eli tässä sitä mukaan kun jälkeläisiksi tulee lisää lapsia. Avain-arvo varastoon verrattuna siis sarakeorientoituneessa tietokannassa samalla avaimelle voidaan kohdistaa useita sarakkeita, mikä mahdollistaa monimutkaisemmat ja laajentuvat datarakenteet (Pokorny 2013). Sarakeorientoitunut NoSQL-tietokanta on suorituskyvyltään erittäin tehokas ja arkkitehtuuriltaan hyvin skaalautuva ja sopii siksi profiilien ja niiden muutosten hallintaan. Sarakkeisiin voidaan tallentaa myös strukturoimatonta dataa, joten sarakeorientoitunut NoSQL-tietokanta sopii myös esimerkiksi web-sisällön, kuten

blogien, wikien ja viestien tallentamiseen. (Pokorny 2013; Planet Cassandra 2015) Esimerkkejä sarakeorientoituneista NoSQL-tietokannoista ovat HBase ja Googlen kehittämä BigTable.

NoSQL-tietokantojen kolmas tyyppi on dokumenttipohjainen varasto, joka on nimensä mukaisesti dokumentteja varten suunniteltu varasto (Moniruzzaman & Hossain 2013). Dokumenttipohjainen varasto käyttää samaa ajatusta kuin avain-arvo varastokin. Avain-arvo varastossa avain osoittaa arvoon, mutta dokumenttipohjaisessa varastossa avain osoittaa dokumenttiin. Jokaisella dokumentilla on siis uniikki avain, jonka avulla dokumentti on helppo noutaa varastosta. (Planet Cassandra 2015) Dokumenttipohjaisen varaston etuna avain-arvo varastoon verrattuna on se, että dokumenttipohjaisessa varastossa dataa voidaan etsiä avaimen lisäksi myös arvon eli dokumentin sisällön perusteella (Moniruzzaman & Hossain 2013). Dokumenttipohjaisen varaston dokumentti on havainnollistettu kuvassa 3.2.

```
{'Etunimi': 'Eeva',
'Sukunimi': 'Jokinen',
'Osoite':{'Katuosoite': 'Lehmustie 6',
          'Postinumero': '00780',
          'Kaupunki': 'Helsinki' },
'Lapset': ['Joni':'2010', 'Jenni':'2012',
          'Jaakko':'2015']
}
```

Kuva 3.2. *Dokumentti dokumenttipohjaisessa varastossa (mukailtu lähteestä Pokorny 2013)*

Kuvassa 3.2. on esitetty henkilön Eeva Jokinen osoite- ja jälkeläistiedot dokumenttina, kuten se olisi dokumenttipohjaisessa varastossa. Dokumenttipohjaiset varastot perustuvat siihen, että dokumentit tallennetaan tiedonvälitykseen tarkoitettuun tiedostomuotoon. Tällaisia tiedostomuotoja ovat esimerkiksi JSON (Javascript Option Notation) ja BSON (Binary JSON). (Moniruzzaman & Hossain 2013) Kuvan 3.2. dokumentti on tallennettu JSON-muodossa, ja sen sisältö on sekä ihmisen että tiedonvälityksen ymmärrettävissä (Pokorny 2013). Dokumenttipohjaiset varastot on suunniteltu semi-strukturoidun datan varastointiin, noutamiseen ja hallintaan. (Planet Cassandra 2015) Dokumenttipohjainen varasto sopii hyvin massiivisten dokumenttimäärien varastoksi ja sieltä on tehokasta etsiä sisältöä avaimen tai sisällön perusteella (Tauro et al. 2012). Varastossa olevat dokumentit voivat olla kirjaimellisesti dokumentteja, kuten tekstidokumentteja tai sähköpostiviestejä, tai käsitteellisiä dokumentteja, kuten esityksiä tietokantakokonaisuuksista kuten tuote- tai asiakastiedoista (Moniruzzaman & Hossain 2013). Esimerkkejä dokumenttipohjaisista

varastoista ovat JSON-formaattia hyödyntävä CouchDB ja BSON-formaattia hyödyntävä MongoDB (Tauro et al. 2012; Moniruzzaman & Hossain 2013).

NoSQL-tietokanta ratkaisuja on erilaisia ja ne ovat syntyneet eri tarpeisiin. Tietokannan hyödyntäjän on osattava valita parhaiten omaan tarpeeseensa sopiva ratkaisu. Valittaessa sopivaa NoSQL-ratkaisua tulisi ottaa huomioon kyseessä olevan datan luonne ja sen käyttötarkoitus sekä miettiä myös, tulevatko nämä seikat muuttumaan tulevaisuudessa. Kun omat tarpeet on tunnistettu, voidaan niitä verrata markkinoilla oleviin ratkaisuihin. Lisäksi tulisi ottaa huomioon oman organisaation kyvykkyys pystyttää ja ylläpitää NoSQL-tietokantaa ja valita tarvittaessa ratkaisu, johon saa tukea palveluntarjoajalta tai ulkoiselta konsultilta. (Planet Cassandra 2015) NoSQL-tietokantoja on kehitetty yleisesti avoimen lähdekoodin ratkaisuina, joten niiden hyödyntämisen kokeileminen ja aloittaminen on kohtuullisen riskitöntä, sillä hyödyntäjän ei tarvitse sitoutua lisenssimaksuihin (Leavitt 2010). NoSQL-tietokannan pystyttäminen ja hallitseminen vaativat kuitenkin paljon työtä ja osaamista, ja siksi myös palveluna myytävien NoSQL-tietokantaympäristöjen suosio on kasvamassa (Planet Cassandra 2015).

4. TIETOKANTOJEN TESTAUS

Tietokanta on moniselitteinen ja jossain määrin tuotekohtainen käsite. Yleisesti tietokannalla tarkoitetaan loogisesti yhteenkuuluvien, tallennettujen tietojen joukkoa, jota voidaan käsitellä ja hallita tietoteknisin menetelmin. (Hovi et al. 2005, s. 4) Ennen kuin tietokannan dataa voidaan käsitellä ja hallita, tulee varmistua tietokannan toimivuudesta ja datan oikeellisuudesta testauksen avulla. Luvussa 4 käsitellään testauksen tarpeellisutta ja laajuutta sekä testauksen suunnittelua.

4.1 Testauksen tarpeellisuus

Tietojärjestelmän testaus on yleisin tapa todeta toimivuus, ja testaus onkin kiistaton kehitysprojektin osa-alue (Rapps & Weyuker 1985). Testausta on hyvä tehdä jokaisen kehitysvaiheen jälkeen, jotta seuraava vaihe voi alkaa mahdollisimman vakaalta pohjalta. Näin virheitä on myös helpompi jäljittää taaksepäin. Jos tuotteen kehityksen loppuvaiheessa havaitaan virhe, voidaan tarkastella vaihe kerrallaan taaksepäin ja etsiä testaustuloksista ja -raporteista löydettyyn virheeseen mahdollisesti viittaavia huomioita. (Haikala & Märijärvi 2006, ss. 51-53) Toisaalta kehitystä saatetaan tehdä eri henkilöiden toimesta eri vaiheissa, joten testausta olisi pystyttävä tekemään eri vaiheissa. Testausta voidaan tehdä myös dynaamisesti samalla kuin kehitys on vielä käynnissä, jolloin havaittujen virheiden korjaukset tulevat mukaan kehitykseen jo kehitysvaiheessa. (Souza et al. 2015)

Kaikki tärkeät tietojärjestelmät käyttävät tietokantatekniikkaa tietojen tallentamiseen. Tietokanta on siis yksinkertaistettuna paikka tallentaa dataa tai se osa tietojärjestelmästä, johon data tallentuu. On tärkeää, että tiedot on tallennettu järkevässä muodossa ja siten, että tiedoista on nopeasti saatavissa yhdistelmiä erilaisiin tarpeisiin. (Hovi et al. 2005, s. 4) Tietokannan toimivuus on erittäin tärkeää liiketoiminnan kannalta tai jopa elintärkeää ihmiselämän kannalta (Andreou & Sofokleous 2008). Liiketoiminnan saralla tietokantaongelma voi seisauttaa tuotannon tai jakelun ja aiheuttaa merkittäviä taloudellisia tappioita (Planet Cassandra 2015). Terveydenhuollossa tietokantojen toimiminen on vielä kriittisempää, sillä potilaan elintoiminnot voivat muuttua nopeasti ja virhe tai viive tietokannassa saattaa estää oikeanlaisen hoidon saamisen. Virhe tietokannassa saattaa myös aiheuttaa virheellisen datan hyödyntämisen organisaation toimintaa kuvaavien lukujen laskennassa. Jos esimerkiksi avoin osakeyhtiö julkaisee väärää tietoa toiminnastaan, rikkoo se lakia ja voi joutua kärsimään suuriakin sanktioita. (Andreou & Sofokleous 2008; Hambling et al. 2010, s. 11)

Tietojärjestelmän testaaminen on tapa varmistua järjestelmän oikeellisuudesta ja siitä, että toteutus täyttää kaikki määritellyt tavoitteet (Andreou & Sofokleous 2008) Näin

voidaan myös minimoida virheiden määrä. (Souza et al. 2015) Testaus on toiminnallinen tapa tarkistaa järjestelmän toteutuksen oikeellisuus ja tarkoituksellisuus kokeilemalla sitä. Kokeilulla tarkoitetaan yleisesti loppukäyttöä simuloivaa testauskäyttöä, jota voidaan pitää myös toteutuksen laadunvarmistuksena. (Rapps & Weyuker 1985; Ding et al. 2008) Toteutuksen laadunvarmistuksen tarkoitus on toisaalta estää virheiden pääsy tuotteeseen ja toisaalta auttaa löytämään tehdyt virheet mahdollisimman aikaisessa vaiheessa. Testaaminen on oleellinen osa laadunvarmistusta. (Haikala & Märijärvi 2006, s. 51) Kehitysvaiheessa olevan tietojärjestelmän toimivuus ja tarkoituksenmukaisuus on aina tarkistettava sekä kehitysprosessin aikana että ennen käyttöönottoa (Rapps & Weyuker 1985). Testaamisen tarkoitus on löytää virheitä, puutteita ja epäjohdonmukaisuuksia mahdollisimman aikaisessa vaiheessa kehitystä. Mitä aiemmassa vaiheessa virhe löydetään, sitä halvempi se on korjata ja vältytään kerrannaisvaikutuksilta.

Tietokanta on paikka tallentaa dataa, joten tietokannan virheet ilmenevät yleensä virheellisenä datana (Collins 2008). Jos käyttäjät eivät voi luottaa käyttämänsä ratkaisun datan oikeellisuuteen, eivät he voi käyttää ratkaisua ollenkaan. Vielä vakavampaa voi olla, jos käyttäjät eivät tiedä virheestä, ja käyttävät virheellistä dataa päätöksenteossa ja tekevät virheellisiä ja liiketoiminnan kannalta haitallisia päätöksiä ja myös organisaation imago kärsii. (Hovi et al. 2009, ss. 167-168) Datan tietokannassa täytyy olla oikein, sillä data ja siitä jalostettu informaatio on yksi tärkeimmistä tai tärkein yrityksen voimavara. Varsinkin dataintensiivisten organisaatioiden liiketoiminta häiriintyy pahasti, jos sen data on epäluotettavaa tai siihen ei ole pääsyä. (Chen et al. 2012) Tietokannan datan oikeellisuuden testaus on siis tärkeää datan luotettavan hyödyntämisen kannalta.

Erityisesti liiketoimintatiedon hallinnan järjestelmien kannalta datan laatu on tärkeä huoli. Jotta laadultaan huonoa tai väärää dataa ei joutuisi liiketoimintatiedon hallinnan järjestelmiin, on tärkeää eliminoida laatuvirheiden pääsy tietokantaan alun perinkään. Huonolaatuinen data aiheuttaa ongelmia ja saattaa jopa vääristää datajoukkoa siinä määrin, että sen pohjalta tehdään vääriä tai huonoja päätöksiä. (Thierauf 2001, s. 125) Huonon datan ja varsinkin sen seurauksena syntyneiden päätösten korjaaminen vaatii paljon aikaa ja resursseja, joten on erittäin tärkeää, että tietokannan data on laadukasta, eheää ja siihen voidaan luottaa. (Hovi et al. 2009, ss. 167-168) Datan korjaukseen ei kuulu ainoastaan datankeruuprosessin laadunvarmistus vaan luvassa voi olla erittäin haastava työ eliminoida huonolaatuinen data tietokannasta ja ajaa sinne uudestaan laadukas data rajatuilla ehdoilla. Tämän takia on myös tärkeää, että tietokantaan menee vain oikeaa tai oikeaksi luokiteltua dataa, jota tullaan jatkossa käyttämään. Tätä varten tarvitaan tiukka määritysprosessi ja -dokumentti. (Thierauf 2001, s. 125)

Tietokannan testausprosessiin liittyy myös muita aspekteja kuin datan oikeellisuus. Tietokannan datan lisäksi testataan usein myös suorituskykyä, integroitavuutta ja viansietokykyä. Näitä testejä varten on olemassa esimerkiksi automaattisia työkaluja, joita hyödyntämällä kehittäjä voi varmistua oman ratkaisunsa toimivuudesta ja tavoitteiden vastaamisesta. (Hambling et al. 2010, s. 45) Tietokannan tehtävä on toimia

paikkana, johon voidaan tallentaa dataa, joten datan oikeellisuus on pääsijalla testauksessa (Hovi et al. 2005, s. 4; Collins 2008). Lisäksi tämän tutkimuksen rajauksena on, että testaaja on datan oikeellisuutta testaava liiketoimintaorientoitunut loppukäyttäjä, joten testauksen käsittely tässä tutkimuksessa keskittyy ainoastaan datan oikeellisuuteen, ei teknisiin yksityiskohtiin.

4.2 Testauksen laajuus

Testaus on tärkeää kaikissa tietotekniikkaprojekteissa (Hovi et al. 2009, s. 166). Yleisesti testausskenaariona on se, että tietojärjestelmästä löytyy virheitä. Testauksen tavoitteena on siis löytää virhekäyttäytymistä järjestelmässä, jotta virheet voidaan korjata. (Myers et al. 2011) Tarvittavan testauksen määrää on kuitenkin vaikea arvioida ja testauksen lopettaminen onkin usein kompromissi tuotteessa olevien virheiden aiheuttamien kustannusten ja markkinoilta myöhästymisen aiheuttaman tuoton menetyksen välillä. (Haikala & Märijärvi 2006, s. 283) Testauksella on kuitenkin erittäin vaikea osoittaa absoluuttisesti, etteikö virheitä olisi olemassa – virheet vain eivät esiinny testitapausten kaltaisessa käytössä. Testauksella voidaan siis ainoastaan kasvattaa järjestelmän luotettavuutta löytämällä ja poistamalla virheitä, mutta absoluuttisesta virheettömyydestä ei voida olla varmoja. (Myers et al. 2011; Marcozzi et al. 2015)

Testaus on pääasiallinen tapa havaita ja etsiä virheitä toteutuksessa. Testattavat kokonaisuudet laajenevat tiedon määrän ja tallennustarpeen kasvaessa, joten testaus on entistä haasteellisempaa ja kustannustehokkuus on suuri huolenaihe. Manuaaliset testausprosessit kuluttavat paljon henkilöresursseja, ja lisäksi testaus on ihnimillisten tekijöiden varassa. Testauksen automatisaatio kasvattaakin suosiotaan. (Marcozzi et al. 2015) Oikein toteutetun automaattisen testauksen avulla voidaan myös kasvattaa toteutuksen laatua ja luotettavuutta, sillä manuaalisessa testauksessa syntyvät inhimilliset virheet voidaan välttää. (Do et al. 2015) Toisaalta automatisoidut testit vaativat paljon suunnittelua ja määrittelyä ja niiden voi olla vaikea vastata odottamattomiin virheisiin tai muutoksiin (Marcozzi et al. 2015). Myös automatisoinnin suunnittelu vaatii paljon resursseja, joten pienemmän mittakaavan tarkastelussa manuaalinen testaus pitää vielä pintansa (Rogstad & Briand 2016).

Sekä manuaalisessa että automatisoidussa testaukselle haasteita tuottaa oikean testausdatan ja -menetelmän löytäminen (Rogstad & Briand 2016). Kaikella saatavissa olevalla datalla testaaminen antaisi tietysti kattavimman kuvan tietokannan datan käyttäytymisestä, mutta yleisesti datamäärät ovat niin suuria, että koko määrällä testaaminen ei ole käytännöllistä tai edes kapasiteetillisesti mahdollista. (Rapps & Weyuker 1985; Myers et al. 2011; Marcozzi et al. 2015) On siis määritettävä erikseen, millä laajuudella tietokantaa testataan ja mitä dataa testauksessa tullaan käyttämään. Data olisi syytä valita vastaamaan todellisuutta ja kaikkia mahdollisia skenaarioita. Skenaarioiden muodostaminen perustuu usein tietokannan määrittelyehtojen pohjalta tunnistettuihin käyttötapauksiin, tai teknisiin toteutuksen osa-alueisiin. (Chen et al. 2004)

Tietokannassa testataan, ovatko osa-alueen tulokset sellaisia, miten määrittelyehdot ovat ne tarkoittaneet. Tätä varten on tunnistettava testaukseen soveltuva datajoukko, jonka käyttäytymistä kannassa voidaan seurata. Tällöin on oltava tiedossa datajoukon alkuperäinen muoto sekä ennalta määritelty, toivottu lopputulos. (Rapps & Weyuker 1985; Souza et al. 2015)

Testaukseen valittavalla aineistolla on suuri merkitys testauksen onnistumisessa (de la Riva et al. 2010). Tietokannat ovat niin laajoja kokonaisuuksia, että niiden joka ikisen datakentän testaaminen olisi taloudellisesti kannattamatonta tai toiminnallisesti jopa mahdotonta. Niinpä on tyydyttävä korkean tason testaukseen ja yksittäisiin testitapauksiin. Testauksen onnistumisen kannalta on erittäin tärkeää, että tunnistetaan testitapaukset, jotka mahdollisimman hyvin kuvaavat koko tietokannan datan luonnetta ja moninaisuutta. (Myers et al. 2011) Testitapausten joukosta tulisi löytyä mahdollisimman laajasti testattavan kokonaisuuden kattavia realistisia käyttötapauksia, erityistapauksia sekä tahallisia virhetapauksia. Jos vain mahdollista, niin testiaineistoksi valittavan datajoukon olisi hyvä olla oikeaa tuotantoaineistoa, sillä testiaineiston luominen käsin hidastuttaa projektia, ja on myös vaarana, että testiaineisto ei ole luonteeltaan täysin samanlaista kuin tuotantoaineisto. Tuotantoaineistoon voidaan kuitenkin tehdä tahallisia virhetapauksia, joita on hyvä testata, jotta voidaan varmistua, että järjestelmä käsittelee virheelliset tapaukset niin kuin on suunniteltukin ja, että testausmenetelmä tuo esiin virhetapaukset. (Hovi et al. 2009, ss. 168-172; de la Riva et al. 2010; Rogstad & Briant 2016)

Tietoa varastoivan tietokannan testaaminen on usein haastavampaa kuin operatiivisten järjestelmien, sillä loppukäyttäjälle näkyvä osa voi olla melko pieni projektin kokoon verrattuna. Tästä johtuen kokonaisvaltainen testaus voi olla haastavaa, jos loppukäyttäjän tekemässä testauksessa ei voida pureutua syvälle projektin tuotoksiin. (Hovi et al. 2009, s. 166) Tietokannan loppukäyttäjälle on tavallisesti tärkeintä, että tietokannan datan oikeellisuus on tarkistettu, eli että tietokanta on eheä ja sen dataan voi luottaa. Tietokanta on eheä, kun sen tiedot ovat oikein, ristiriidattomia ja vastaavat reaalimaailmaa. Eheys tarkoittaa, että datasisältöä ei ole muutettu ilman valtuuksia tai ettei se ole tahattomasti muuttunut ja mahdolliset muutokset voidaan todentaa. (Kamel 2009) Eheä data on aitoa, väärentämätöntä, sisäisesti ristiriidatonta, kattavaa, ajantasaista, oikeellista ja käyttökelpoista. (Kaario & Peltola 2008, s. 155) Eheyttä voidaan testata vertaamalla lähdejärjestelmien alkuperäistä datajoukkoa tietokannassa olevaan datajoukkoon (Chen et al. 2004; Myers et al. 2011).

Dataa tallentavan tietokannan yleisiä virhetapauksia ovat esimerkiksi puuttuvat datakentät, duplikaattidata sekä virheet desimaalimerkkien ja päivämäärien kanssa (Collins 2008). Virhetapaukset voidaan tunnistaa datasta vertaamalla tietokannan dataan odotettuun tulokseen. Odotettuun tulokseen vertaaminen antaa myös mahdollisuuden tarkastella toteutuksen vastaavutta todellisiin tarpeisiin. (Souza et al. 2015) Tietokannan virheet voivat olla myös seurausta väärin tehdystä tai ymmärretystä mallinnuksesta,

jolloin kannan datarakenteet käsittelevät dataa väärin tai esimerkiksi tallentavat datan väärään paikkaan. Myös mallinnuksesta johtuvat virheet on yleensä havaittavissa tietokannan vertaamisella odotettuun tulokseen. (Currim et al. 2014)

4.3 Testauksen suunnittelu

Kun organisaatio käynnistää tietojärjestelmiin liittyvän projektin, on sille yleensä ulkoinen toimittaja eli kehittäjä, ja tilaajaorganisaatio toimii projektissa asiakkaan roolissa. On tärkeää, että testausta tekee kehittäjän lisäksi myös toinen osapuoli, sillä kehittäjän on vaikea havaita omia virheitään ja hän on tehnyt ratkaisun mielestään valmiiksi. Ulkopuolisen tarkastelijan on helpompi huomata virheitä, epäjohdonmukaisuuksia tai epäselvyyksiä ratkaisuissa, ja näin testauksen tehokkuus ja luotettavuus kasvavat. (Hambling et al. 2010, s. 135) Kuitenkin on huomioitava, että mitä kauempana testaaja on kehittäjästä, sitä vieraammaksi ratkaisu muuttuu ja virheitä voi olla vaikea erottaa, jos testaaja ei tunne järjestelmää. Järjestelmän lopputuloksen kannalta on olennaista, että loppukäyttäjät kokevat järjestelmän sisältämän datan oikeelliseksi. Loppukäyttäjät ovat myös kehitysmielessä kaukana kehittäjästä, joten loppukäyttäjät sopivat hyvin testaajiksi. (Hovi et al. 2009, s. 167; Hambling et al. 2010, ss. 136-137)

Loppukäyttäjälle tietokannan datan oikeellisuus on tärkein toiminnallinen ominaisuus ja sen testaus on myös tärkeää (Hovi et al. 2005, s. 11; Hovi et al. 2009, s. 168). Usein loppukäyttäjä näkee lopputuloksen vasta valmiiksi luotuina raportteina, joiden valmistuminen on viimeisenä kehitysvaiheena. Olisi kuitenkin molempien osapuolien kannalta hyvä, että mahdolliset virheet datassa ilmenisivät jo data-alustan kehitysvaiheessa eikä vasta lopullisessa raportoinnissa. Valmiit raportit ovat myös aikataulullisesti huono vaihe huomata virheet, sillä valmiit raportit syntyvät vasta projektin loppuvaiheilla, ja silloin aikataulupaine on jo muutenkin tiukka eikä projektia haluttaisi venyttää. (Hambling et al. 2010, s. 17)

Suoritettiinpa testaus mistä näkökulmasta ja kenen toimesta tahansa, on testaus prosessina vaativa ja vaatii onnistuakseen määrätietoista suunnittelua ja toteutusta (Hambling et al. 2010, s. 142). Testaukseen liittyviä työvaiheita ovat testauksen suunnittelu, testiympäristön valinta ja luonti, testin suorittaminen, tulosten tarkastelu ja dokumentointi (Haikala & Märijärvi 2006, s. 283; Hambling et al. 2010, s. 21). Testauksen suunnittelulla tarkoitetaan yleisesti testaussuunnitelman tekoa ja testitapausten määrittelyä. Testitapausten määrittämisen tulisi olla suunnitelmallista, jotta testitapaukset täyttävät kaikki ennalta määrätyt tilanteet. (Hovi et al. 2009, s. 168) Muutamien tuntien huolellisesti valitulla ja rajatulla testitapausjoukolla voi johtaa parempaan tulokseen kuin päiviä kestävä umpimähkäinen kokeilu (Haikala & Märijärvi 2006, s. 283). Testaus tulisi suunnittelun lisäksi dokumentoida, jotta tuloksia voidaan käyttää testattavan ratkaisun kehittämiseen ja tätä varten luodaan yleensä testausraportti (Hovi et al. 2009, s. 172).

Testaussuunnitelma tehdään usein teknisen ja toiminnallisen vaatimusmäärittelyn pohjalta (Haikala & Märijärvi 2006, s. 287; Hambling et al. 2010, s. 143; Souza et al. 2015). Vaatimusmäärittelyn tarkoituksena on kartoittaa ja dokumentoida toimeksiantajan tarpeet ja toiveet toteutettavasta järjestelmästä. Vaatimusmäärittelystä selviää muun muassa mihin käyttöön järjestelmä tulee, ketkä ovat sen käyttäjiä, mitä toimintoja järjestelmältä vaaditaan, mitkä tiedot ja liittymät ovat kyseessä sekä mitä ei-toiminnallisilta ominaisuuksilta, kuten suorituskyky, viansietokyky ja skaalautuvuus, vaaditaan. (Hovi et al. 2005, s. 29) Määrittely tehdään tarpeiden mukaan ja siihen voivat vaikuttaa myös viranomaismääräykset tai lakipykälät. Tietokannan määritys ja toteutus ovat ihmisten tekemiä, joten inhimilliset virheet ovat aina mahdollisia. Inhimillisten virheiden lisäksi ongelmia voi aiheuttaa se, että määrittely on lähes aina puutteellinen ja syntyy ristiriitatilanteita. (Haikala & Märijärvi 2006, s. 287)

Vaatimusmäärittely toimii kuitenkin yleensä hyvin testaussuunnitelman pohjana (Souza et al. 2015). Testaussuunnitelma tehdään, jotta tiedetään, mitä tullaan testaamaan ja mitkä ovat odotetut ja hyväksytyt lopputulokset,. Testaussuunnitelma on tärkein dokumentti testausprosessissa, sillä siinä kerrotaan miten testaus aiotaan viedä läpi. (Hovi et al. 2009, s. 170) Testaussuunnitelmasta selviää muun muassa mitä testejä tehdään, milloin ne tehdään, miten ne järjestetään ja millaisia lopputuloksia odotetaan (Haikala & Märijärvi 2006, s. 299).

Testaussuunnitelman kohde eli ”mitä testataan” määrittelee sen osa-alueen tai kokonaisuuden, jonka testaus on kyseessä. Testaussuunnitelma tulisi yleensä tehdä yhdelle testattavalle kokonaisuudelle ja jos kokonaisuuksia on useita, jokaiselle kokonaisuudelle tulisi tehdä oma testaussuunnitelma. (Hambling et al. 2010, s. 143) Kun testaussuunnitelmia on jokaisen osa-alueen testaukselle omansa, syntyy pienempiä testauskokonaisuuksia, joita on helpompi hallita (Hovi et al. 2009, s. 170). Testaussuunnitelma voi näyttää esimerkiksi taulukon 4.1. mukaiselta kokonaisuudelta.

Taulukko 4.1. *Testaussuunnitelmamalli (Hovi et al. 2009, s. 171; Hambling et al. 2010, ss. 144-145)*

Tehtävä	Mitä tehdään?
Testauksen kohde	Mitä testataan?
Testauksen laajuus	Miten laajasti testataan?
Rajaus	Rajataanko testausta jollain tavalla?
Tavoitteet	Mitkä ovat testauksen halutut lopputulokset?
Testausaineisto	Millä aineistolla testataan?
Testitapaukset	Mitä testitapauksia käytetään?
Testausvälineet	Mitä välineitä testauksessa käytetään?
Testauksen resurssit	Mitä henkilö-, laitteisto- ja ohjelmistoresursseja käytetään?
Tarvittava osaaminen	Onko testaajilla vaadittava osaaminen ja tarvirtsevatko he koulutusta?
Testauksen lopputulokset	Miten testauksen lopputulokset kirjataan ja mihin ne tallennetaan?
Hyväksymiskriteerit	Mitkä ovat hyväksymiskriteerit?

Kuten myös taulukosta 4.1. voidaan huomata, on testaussuunnitelma yleensä selkokielinen dokumentti, josta ilmenee testauksen kohde, laajuus, tavoitteet, menetelmät, resurssit sekä lopputulosten käsittely. Testaussuunnitelman avulla voidaan testausprosessin lisäksi suunnitella myös resursointia. Testaussuunnitelma luodaan yleensä projektin alkuvaiheessa vaatimusmäärittelyn pohjalta, ja suunnitelmaa on syytä päivittää projektin etenemisen myötä. Kun testaussuunnitelma on tehty projektin alkuvaiheessa, voidaan jo ennakoivasti kartoittaa testaajia ja heidän osaamistaan, ja varmistaa että testaajat ovat heti valmiita, kun on testattavaa. (Hambling et al. 2010, ss. 142-143) On hyvä, että testaussuunnitelma ja testaajat ovat valmiina toimimaan, kun järjestelmä on testattavissa, sillä testaus on yleensä viimeinen asia ennen käyttöönottoa, jota halutaan yleensä kiirehtiä. Tämän takia on oltava selkeä suunnitelma testaukselle ja riittävästi aikaa ja muita resursseja. (Hovi et al. 2009, ss. 167-170)

Jotta testaus on toistettavissa ja sen tuloksiin voidaan palata, täytyy testaus ja sen tulokset dokumentoida hyvin (Hovi et al. 2009, s. 173). Testauksen dokumentointi tapahtuu esimerkiksi testausraportin tuottamisen avulla. Testausraportti on yleensä tiivistelmä testaussuunnitelman läpiviemisestä ja testauksen tuloksista. (Hambling et al. 2010, s. 154) Testausraportissa olisi hyvä kertoa:

- mitkä osat toteutuksesta on kyseisessä testauksessa testattu
- mitä virheitä havaittiin ja mitkä ovat niiden kriittisyysasteet
- onko testaus hyväksyttävissä testaajien mielestä
- mikä meni testauksessa hyvin ja mikä huonosti (Hovi et al. 2009, s. 173).

Testausraportti voi olla aikavälikohtainen tai osa-aluekohtainen, tai voi myös perustua testaussuunnitelman laajuuteen. Testausraportin luotettavuuden ja hyödyntämisen kannalta on tärkeää ilmaista selkeästi, mikä osa toteutuksesta on testattu. (Hambling et al. 2010, s. 154) Testausraporttiin merkitään testauksen tulokset, mutta on erityisen tärkeää kuvata selvästi testauksessa ilmenneet virheet, jotta ne voidaan korjata (Hovi et al. 2009, s. 173). Testausraporttiin merkitään testauksen tulokset, jotta muutkin kuin testaaja voivat seurata testauksen lopputuloksia. Testausraportilla on kaksi tehtävää. Ensinnäkin se kertoo mitä testauksessa on tapahtunut eli miten testaus on suoritettu testaussuunnitelman pohjalta, ja täyttyvätkö hyväksymiskriteerit. Toiseksi testausraportista ilmenee korjausehdotukset ja jatkotoimet. (Hambling et al. 2010, ss. 154-155)

Testausraportin pohjalta voidaan analysoida mitä virheitä järjestelmästä löytyy ja miten ne tulisi korjata, kuinka luotettava testaus oli ja onko tarpeellista tai taloudellisesti kannattavaa jatkaa testausta (Hambling et al. 2010, s. 154). Testauksen työvaiheisiin ja niihin läheisesti liittyvään virheiden jäljitykseen ja korjaukseen kuluu paljon tietojärjestelmäprojektin resursseista. Testauksen on määrä olla kompromissi käytettävissä olevien resurssien ja luotettavuudesta saavutetun varmuuden välillä. (Haikala & Märijärvi 2006, s. 283) Testausraportin pohjalta voidaan myös miettiä testausprosessin kehitystarpeita jatkoa ajatellen. Testausraportin avulla voidaan pohtia, oliko testaukselle asetetut tavoitteet sopivia, oliko testausmenetelmä ja -aineisto tarpeeksi kattava ja saatiinko testattua koko toivottu kokonaisuus. Nämä huomiot helpottavat tulevien testaussuunnitelmien muodostamista. (Hambling et al. 2010, s. 155)

5. TESTAUSSUUNNITELMA RAPORTOINTITYÖKALUJEN AVULLA TESTATTAVAA NOSQL-TIETOKANTAA VARTEN

Tässä luvussa esitellään lähtökohdat empiiristä tutkimusta varten. Ensin esitellään tutkittava tapaus ja sen erityispiirteet. Tämän jälkeen muodostetaan teorian pohjalta testaussuunnitelma, jonka yhteydessä esitellään myös käytettävä aineisto sekä testaustapaukset. Lopuksi esitellään vielä tutkimuksessa käytettävät työkalut.

5.1 Empiirisen tutkimuksen lähtökohdat

Tämän tapaustutkimuksen tausta-ajatuksena on, että liiketoimintaorientoituneen loppukäyttäjän tulisi testata NoSQL-tietokannan datan oikeellisuutta mahdollisimman tehokkaasti ja helpoin menetelmin. Tätä varten raportointityökaluja ymmärtävä loppukäyttäjä voi yhdistää raportointityökalun tietokannan dataan ja tarkastella raportointityökalujen tarjoamien ominaisuuksien avulla, onko tietokannan data oikeellista eli vastaako se odotettuja tuloksia. Tutkimus ottaa kantaa raportointityökalujen yleiseen soveltuvuuteen NoSQL-tietokannan datan testauksessa, minkä lisäksi tarkastellaan ja vertaillaan kolmea eri raportointityökalua tässä tarkoituksessa.

Testaajana toimivan liiketoimintaorientoituneen loppukäyttäjän ajatellaan olevan NoSQL-tietokantojen rakennetta ymmärtämätön tilaajaosapuolen edustaja, jota kiinnostaa ainoastaan datan oikeellisuus tietokannassa. Loppukäyttäjänä testaaja tulee käyttämään valmista tietokantaa, joten hänelle on tärkeää, että tietokannan data on oikeellista. Raportointityökalut on valittu testausvälineiksi, sillä loppukäyttäjällä on jonkin verran ymmärrystä raportoinnista, mutta NoSQL-tietokantaymmärrys puuttuu, ja raportointityökalujen avulla päästään helposti käsiksi dataan. Testaaja ei halua odottaa lopullisten raporttien valmistumista, sillä niissä on ulkoasuseikkoja ja muuta ylimääräistä, vaan haluaa tarkistaa datan jo ennen kun siitä aletaan toimittajan puolesta luomaan tilaajalle kalliita raportteja, joiden ”valmiin” datan korjaaminen on kallista.

Testaajana eli liiketoimintaorientoituneena loppukäyttäjänä ja tulosten dokumentoijana toimii tässä tutkimuksessa tutkija itse. Jottei tulos olisi liian subjektiivinen, suorittaa testauksen tutkijan lisäksi myös kaksi ulkopuolista koehenkilöä, molemmat koulutukseltaan tekniikan kandeja. Raportointi on koehenkilöille teoriassa tuttu asia, mutta he eivät ole aiemmin käyttäneet mitään kolmesta käytettävästä raportointityökalusta.

Koska testaajan ajatellaan olevan tilaajaosapuolen edustaja, osoittaa NoSQL-tietokannan kehittäjä testaajalle testattavan datan sijainnin. Tämän jälkeen loppukäyttäjä eli testaaja yhdistää raportointityökalut hänelle osoitettuun dataan ja suorittaa testaussuunnitelmassa määritellyt asiat. Testaussuunnitelma muodostetaan tutkimuksen lähtökohdan ja teoriaosuuden perusteella. Testaussuunnitelma ottaa kantaa NoSQL-tietokannan datan oikeellisuuden testaukseen raportointityökalujen avulla, ja siinä määritellään muun muassa käytettävät testitapaukset, -aineisto ja resurssit. Kun testaussuunnitelma on muodostettu, hyödynnetään sitä empiirisessä tutkimuksessa

5.2 Testaussuunnitelman esittely

Tämän empiirisen tutkimuksen suorittamisen askeleet ovat testitapauksen määrittely, tahallisten virheiden luominen dataan, odotetun tuloksen määrittäminen, työkalujen asennus, yhteyksien muodostaminen, testaus eri välineillä, dokumentointi sekä tulosten vertailu. Testaussuunnitelmassa otetaan kantaa testitapausten määrittelyyn, tahallisten virheiden luomiseen, odotettuihin tuloksiin, työkalujen käyttöönottoon sekä tulosten dokumentointiin. Testaussuunnitelman tarkoituksena on ohjata testausprosessia, jotta kaikki tarvittavat seikat tulee läpikäytyä ja testauksen tuloksiin voidaan luottaa. Tämän tutkimusken tarpeisiin luotu testaussuunnitelma on esitetty taulukossa 5.1.

Taulukko 5.1. *Sähködatan testaussuunnitelma*

Tehtävä	Miten suoritetaan?
Testauksen kohde	Testataan sähködatan oikeellisuutta Hbase-tietokannassa
Testauksen laajuus	Testataan vuoden 2014 sähkönkulutus- ja sähköntuotantodataa tuntitasolla, verrataan alkuperäiseen dataan.
Rajaus	Testataan datan oikeellisuutta, ei muita ominaisuuksia.
Tavoitteet	Tavoitteena löytää tahallisesti tehdyt virheet datasta ja varmistaa muilta osin oikeellisuus.
Testausaineisto	Energiateollisuus ry:n raportoima vuoden 2014 sähkönkulutus- ja tuotanto dataa tuntitasolla. Luodaan dataan virhetapausten mukaisia tahallisia virheitä.
Testitapaukset	Yleinen tapaus: Datan olemassaolon ja korkean tason tarkastelu. Virhetapaus: Duplikaatin, puuttuvan datan, virheellisen desimaalimerkin ja virheellisen päivämäärämuodon vaikutus kuukausisummiin. Erikoistapaus: Kesäajan muutos, tarkastelu tuntitasolla.
Testausvälineet	Raportointityökalut Tableau, Power BI ja Qlik Sense.
Testauksen resurssit	Liiketoimintaorientoitunut loppukäyttäjä, raportointityökalujen kokeilulisensseillä varustetut työkaluversiot.
Tarvittava osaaminen	Raportointityökalujen yleisten toimintojen tunteminen.
Testauksen lopputulokset	Lopputulokset dokumentoidaan sanallisten selitysten ja kuvakaappausten avulla.
Hyväksymiskriteerit	Testimenetelmällä huomataan puuttuva ja virheellinen data.

Testaussuunnitelmassa, eli taulukossa 5.1., testauksen kohde ja rajaus on sähködatan oikeellisuuden tarkastelu Hbase-tietokannassa. Tämä on tarkastelualue, jota liiketoimintaorientoitunut loppukäyttäjä voi tarkastella raportointityökalujen avulla. Testauksen laajuuden määrittelee tässä tapauksessa käytettävä testausaineisto, joka on esitetty luvussa 5.2.1. Tietokannassa olevaa testausaineistoa verrataan alkuperäiseen aineistoon ja pyritään löytämään tahallisesti tehdyt virheet datasta. Testitapaukset on jaettu yleisiin tapauksiin sekä virhe- ja erikoistapauksiin. Testitapaukset on esitetty tarkemmin luvussa 5.2.2.

Testauksen suorittaa liiketoimintaorientoitunut loppukäyttäjä, jolla on pintapuolinen ymmärrys raportointityökalujen käytöstä. Testaaja käyttää Tableauta, Power BI:ta ja Qlik

Sensea ja raportoi tulokset kuvakaappausten ja sanallisten selitysten avulla. Testauksen tavoitteena on löytää tahallisesti tehdyt virheet ja varmistua muilta osin datan oikeellisuudesta. Hyväksymiskriteeriksi voidaan asettaa, että testaus on hyväksytty kun kaikkien testitapausten tulos on oikeellinen. Lisäksi käytettäessä kolmea eri raportointityökalua, tavoitteena on arvioida ja vertailla työkalujen sopivuutta datan oikeellisuuden testaamiseen. Työkalut on esitelty tarkemmin luvussa 5.3.

Testauksen resurssit painottuvat henkilöstöresursseihin. Tässä tutkimuksessa testausta varten valitut teknologiat ovat kokeilulisenssin avulla ilmaisia käyttää tiettyyn pisteeseen asti, joten voidaan nähdä että perehtymiseen ja testaukseen käytetyt henkilötyötunnit ovat suurin kulutettava resurssi. Testauksen pohjana on ajatus, että loppukäyttäjä tuntisi raportointityökaluja yleisesti jonkin verran, mutta testaukseen on valittu käyttäjän kannalta helppoja raportointityökaluja, joten testausta voi tehdä myös erittäin lyhyen perehtymisen pohjalta.

5.2.1 Testausaineisto

Testauksen laajuuden määrittää sopivan testausaineiston saatavuus. Testausaineiston on oltava kokonaisuus, josta on saatavilla sekä tietokantaan tallennettu data että alkuperäinen data vertailutarkoituksiin. Tietokantaan tallennettuun testausaineistoon on myös pystyttävä tekemään tahallisia virheitä, jotta voidaan toteuttaa virheitä kuvaavat testitapaukset. Tahallisiksi virheiksi luodaan tietokannoille yleisiä datavirheitä, eli duplikaattidataa, puuttuvaa dataa sekä virheellisiä desimaalimerkkejä ja päivämäärämuotoja..

Tässä tutkimuksessa testausaineistona käytetään Energiateollisuus ry:n julkaisemaa sähködataa, joka kuvaa tuntitasoista sähköntuotanto- ja sähkönkulutusdataa Suomessa vuonna 2014. Data on ladattu Energiateollisuus ry:n (2015) internetsivuilta ja datasta tehdään testausta varten kaksi erilaista versiota. Ensimmäinen versio on alkuperäinen, muokkaamaton data, jota käytetään vertailuaineistona eli oikeellisena lopputuloksena. Toinen versio datasta on itse testausaineisto, johon on luotu tahallisia virheitä testausta varten. Tämä virheitä sisältävä data on tallennettu NoSQL-tietokanta HBaseen. Taulukossa 5.2. on kuvattu testausaineistoon tehdyt tahalliset virheet ja kuinka näiden virheiden tulisi näkyä testauksessa. Virheinä ovat virhellinen desimaalimerkki ja päivämäärä, puuttuva data sekä duplikaattidata. Taulukossa on lisäksi kuvattu tarkemmin, minkä päivämäärän datassa virhe ilmenee, sekä esitetty tarkempi kuvaus luodusta virheestä.

Taulukko 5.2. *Testausaineistoon tehdyt virheet*

Virhe	Missä ilmenee	Virheen tarkennus	Miten pitäisi näkyä lopputuloksessa
Virheellinen desimaalimerkki	21.1.2014 22. tunti	*Tuotanto* desimaalimerkki pilkun sijaan piste	*Tuotanto*-summa virheellinen tammikuussa
Virheellinen päivämäärä	11.5.2014 14. tunti	Päivämäärämuoto on muuttunut 5.11.2014 muotoon	Touko- ja marraskuun kuukausisummat virheellisiä
Puuttuva data	12.5.2014 14. tunti	Päivämäärätieto puuttuu	Lukemia, joitka eivät kohdistu millekään kuukaudelle, toukokuun summat virheellisiä
Puuttuva data	27.10.2014	Data tältä päivältä puuttuu	Lokakuun summat virheellisiä, lokakuussa vain 30 päivää
Puuttuva data	26.1.2014 23. tunti	*Vesivoima* ja *tuotanto* lukemat puuttuvat	Tammikuun *vesivoima*- ja *tuotanto*-summat virheellisiä
Duplikaattidata	2.7.2014 10. tunti	Data tältä tunnilta duplikaattina eli kahteen kertaan	Heinäkuun summat virheellisiä
Duplikaattidata	1.12.2014 9. tunti	Data tältä tunnilta duplikaattina eli kahteen kertaan	Joulukuun summat virheellisiä
Duplikaattidata	2.11.2014	Koko päivän data duplikaatteina eli kahteen kertaan	Marraskuun summat virheellisiä

Taulukosta 5.2. nähdään, että datavirheitä on tehty tammikuuhun, toukokuuhun, heinäkuuhun, lokakuuhun, marraskuuhun ja joulukuuhun eli näiden osalta kuukausitasoisissa summissa pitäisi olla eroavaisuuksia, ja muiden kuukausien osalta kuukausisummien pitäisi olla oikeellisia. Taulukon 5.2. virheiden lisäksi testausaineistosta huomioidaan erityistapauksena kesäaikaan siirtyminen. Vuonna 2014 kesäaika alkoi 30.3 klo 3.00 eli tuolloin kalenterista hävisi yksi tunti. Päivän 30.3. kohdalta siis puuttuu kaikki tiedot tunnilta numero 3. Tämä on erityistapaus, jonka kohdalla on aiheellista tarkistaa, näkyykö puuttuva tunti aineistossa ja toisaalta näkyisikö mikä tahansa puuttuva tunti kuukausitasoisessa tarkastelussa.

5.2.2 Testitapaukset

Virhetapausten lisäksi testitapauksiksi valitaan yleisiä tapauksia ja erikoistapauksia. Virhetapaukset ovat taulukossa 5.2. esitettyjen luotujen virheiden johdosta dataan syntyneitä virheitä. Yleiset tapaukset pyrkivät puolestaan kuvaaman useita yleisiä

tilanteita ja vastaavat normaalikäyttöä. Erikoistapaus on esimerkiksi kesä- ja talviajan vaihdos, joka saattaa aiheuttaa haasteita tietokantaan. Tämän tutkimuksen empiirinen osa eli sähködatan oikeellisuuden testaus on jaettu testaussuunnitelman ja testausaineiston virheiden pohjalta viiteen eri testitapaukseen:

1) yhteydenmuodostus
2) aloitus
3) datan olemassaolon tarkastelu
4) datan korkean tason tarkastelu
5) datan tuntitasoinen tarkastelu.

Ensimmäisenä testitapauksena on yhteydenmuodostus tietokannan ja raportointityökalun välillä. Yhteydenmuodostusta arvioidaan loogisuuden ja helppokäyttöisyyden perusteella. Myös testauksen aloitusta ja raportointityökalujen aloitusnäkymiä arvioidaan helppokäyttöisyyden ja loogisuuden perusteella. Helppokäyttöisyyden testaamiseksi aloitusnäkymässä valitaan kolme kenttää ja kokeillaan, miten helppoa visualisaation muodostaminen on.

Aloitusnäkymän testauksen jälkeen on vuorossa datan olemassaolon tarkistus. Olemassaolon tarkistuksen tavoitteena on tarkistaa nopeasti, onko testattava data olemassa, eli että dataa löytyy varmasti tietokannasta. Olemassaolon tarkistus on tehokas ja nopea keino huomata, onko data pää piirteittäin kunnossa, ja jos tässä vaiheessa huomataan suuria puutteita, ei testausta ole hyödyllistä jatkaa. Datan olemassaolon tarkistus tehdään muodostamalla datasta helppo ja mahdollisimman kattava kuvaaja. Sähködatan tapauksessa tahdotaan varmistaa, että dataa on varmasti kaikille kuukausille ja kaikille tuotanto- ja kulutuslajeille. Tässä testitapauksessa käytetään kuukausien tarkastelussa *Kuukausi*-kenttää, joka on merkkijono, eikä varsinaiseesti aikadimensio, joten samalla tulee testattua, osaavathan raportointityökalut erottaa merkkijonon dimensioksi.

Datan olemassaolon tarkistuksen jälkeen aloitetaan datan korkean tason tarkastelu. Korkeatasoista tarkastelua tehdään, jotta voidaan huomata, missä osa-alueilla on ongelmia oikeellisuuden kanssa eli mitä osa-alueita tulisi tarkastella tarkemmin. Sähködatan testauksessa korkean tason tarkastelu tehdään vertaamalla tietokannan kuukausittaisia summia kulutus- ja tuotantolajeittain alkuperäiseen dataan. Tässä kuukausitasona käytetään varsinaista aikadimensiota, *Päivämäärä*-kenttää, jotta voidaan varmistua, miten työkalut ymmärtävät aikadimension.

Datan tuntitasoista tarkastelua tehdään tuntitasoista virheiden ja puutteiden huomaamiseksi. Tässä tutkimuksessa tuntitasoisessa tarkastelussa on lähtökohtaisesti maaliskuu, sillä maaliskuussa on tarkasteltava erityistapaus – kesäaikaan siirtyminen. Tuntitasoisessa tarkastelussa pyritään tässä tutkimuksessa huomaamaan datasta puuttuvia tunteja.

5.3 Testauksessa käytettävien työkalujen esittely

Tutkimuksessa testattava data sijaitsee HBasessa, joka on Hadoop viitekehykseen kuuluva NoSQL-tietokanta. Hadoop on tunnetuin ja käytetyin big data -ratkaisu, joka tarjoaa puitteet hajautettuun suurien datamäärien varastointiin ja käsittelyyn (Anuradha & Ishwarappa 2015). Hadoop on Apache-lisensioitu eli sillä on avoimen lähdekoodin lisenssiehdot. Hadoop koostuu monista eri kokonaisuuksista ja sen tärkein osa on HDFS (Hadoop Distributed File System), joka tarjoaa hajautetun tiedostojärjestelmän, joka voidaan rakentaa edullisista levypalvelimista ja johon voidaan tallentaa mitä tahansa dataa. HDFS on erittäin hyvin skaalautuva ja viansietokykyinen, sillä dataa tallennetaan oletusarvoisesti vähintään kolmelle palvelimelle. (Hortonworks 2014) Apache-lisensioitu HBase on NoSQL-tietokanta, joka toimii HDFS:n päällä. Sen on hajautettu ja erittäin hyvin skaalautuva ei-relationaalinen tietokanta, joka tarjoaa reaaliaikaisen pääsyn HDFS:ssa olevaan dataan sarakemuodossa. HBase on siis sarakeorientoitunut NoSQL-tietokanta. (Anuradha & Ishwarappa 2015)

Liiketoimintatiedon hallinnassa hyödynnettäville raportointityökaluille yleistä on, että ne eivät pääasiassa osaa vielä tukea luontevasti muuta kuin selkeitä rivi-sarakerakenteita (Krishnan 2013, s. 254). Tätä varten NoSQL-tietokanta HBaseen yhdistetään ensin työkalu, jonka avulla dataa voidaan ryhmitellä raportointityökalujen luontevasti ymmärtämiksi riveiksi ja sarakkeiksi ja määritellä niille datatyyppi. Tässä apuna toimii Apache-lisensioitu Drill, joka on skeematon kyselymoottori, jonka avulla HBasen datasta voidaan luoda taulumaisia näkymiä, joita raportointityökalut ymmärtävät automaattisesti ja yksiselitteisesti. (Apache Drill 2015a) Näkymät ovat virtuaalisia tauluja, joihin ei tallennu mitään tietoa, mutta joiden avulla voidaan tarjota käyttäjälle käyttäjän tarpeille sopivia näkökulmia tietokantaan (Hovi et al. 2005, s. 14). Drillilla on oma ODBC (Open Database Connectivity) -ajuri, eli rajapinta jonka avulla sovellukset voivat kommunikoida tietokantapalvelimen kanssa. Raportointityökalut voidaan siis yhdistää ODBC-ajurilla Drilliin, joka on yhteydessä HBaseen, jotta HBasen datasta luotuja näkymiä voidaan hyödyntää raportointityökaluissa. Dataa ei siis tarvitse muokata enää raportointityökalussa, kun tarvittu datan valinta, järjestely ja datatyyppien määrittäminen on tehty Drillin avulla. Drill-ajurin voi ladata ilmaiseksi Apachen verkkosivuilta. (Apache Drill 2015b)

Tässä tutkimuksessa ei oteta kantaa sähködatan sijainnille HBase-tietokannassa, sillä tutkimuksen oletus on, että tietokanta ja sen data on olemassa ja testaaja suorittaa ainoastaan tietokannan datan testausta. Lähtökohtana siis on, että tietokannan toteuttaja on luonut valmiiksi testausdatasta näkymän, jonka luonti onnistuu muutamassa minuutissa. Näkymän myötä testaajan ei tarvitse ymmärtää tietokannan rakennetta tarkemmin, vaan hän voi keskittyä ainoastaan datan testaamiseen yhdistämällä raportointityökaluja valmiiseen näkymään datasta. Ote näkymästä sähködataan on havainnollistettuna liitteessä A.. Sähködataa esittävä näkymä on rivi-sarakemuodossa,

jolloin sekä inhimillisen tulkitsijan että tietoteknisten ratkaisujen on helppo lukea näkymän taulukkomaista sisältöä.

Tässä tutkimuksessa tietokannan dataa luetaan raportointityökalujen avulla. Tutkimuksessa on valittu tarkasteltaviksi kolme raportointityökalua: Tableau, Power BI ja Qlik Sense. Kaikki työkalut markkinoivat itseään analytiikan edelläkävijöinä, jotka tarjoavat ratkaisuja datan omatoimiseen visualisointiin (Microsoft 2015; Qlik 2016; Tableau Software 2016). Omatoimisen visualisoinnin ja laajan tunnettuuden lisäksi valintakriteerinä raportointityökaluille oli, että niistä on saatavilla ilmainen kokeiluversio. Lisäksi juuri näiden kolmen työkalun valintaan vaikutti se, että Gartner (2016) toteaa juuri Tableaun, Qlikin ja Microsoftin olevan kolme johtavaa toimittajaa liiketoimintatiedon hallinnan markkinoilla.

Tableau on yhdysvaltalaisen Tableau Softwaren vuonna 2005 julkaisema raportointi- ja visualisointityökalu (Tableau Software 2016). Tableau mahdollistaa intuitiivisen datan tarkastelun ja sen avulla on mahdollista luoda nopeasti interaktiivisia visuaalisia mittaristoja ilman laajoja raportointitaitoja. Tableau tunnistaa automaattisesti datalle todennäköisimmin sopivan visualisaation ja muodostaa sen vain muutaman painalluksen avulla. Tableaun vahvuuksia ovat myös hierarkioiden muodostaminen ja datan yhteyssuhteiden esittäminen. (Gartner 2015) Gartnerin selvityksessä (2015) asiakkaat arvostelivat Tableaun helppokäyttöisimmäksi raportointityökaluksi ja Tableaun raporttien luominen oli nopeinta riippumatta ratkaisun kompleksisuudesta. Tutkimuksessa käytetty Tableaun versio on tutkimushetkellä uusin saatavilla oleva versio, Tableau Desktop 9.2.1.

Power BI on yhdysvaltalaisen IT-jätti Microsoftin maailmanlaajuisesti vuonna 2015 julkaisema raportointi- ja visualisointityökalu, joka on suunniteltu erityisesti toimimaan yhdessä Microsoftin muiden tuotteiden kanssa. Power BI tarjoaa helppokäyttöistä ja interaktiivista visuaalisten mittaristojen luomista myös vähemmän kokeneille raporttien luojille ja ehdottaa valitun datan pohjalta todennäköisintä visualisaatiota. Power BI:n etuna on se, että se hyödyntää tuttuja Microsoft-ratkaisuja, kuten Excelia ja Power Pointia, joten kokeneen MS Officen käyttäjän on helppo aloittaa Power BI:n käyttäminen. (Microsoft 2015) Tutkimuksessa käytetty versio on Power BI:n tutkimushetkellä uusin saatavilla oleva versio, versio 2.31.4280.361.

Qlik Sense on puolestaan alun perin ruotsalaisen QlikTech-yhtiön vuonna 2014 julkaisema raportointi- ja visualisointityökalu (Qlik 2016). Qlik Sensen avulla liiketoimintaorientoituneet loppukäyttäjät voivat luoda omia, interaktiivisia visuaalisia mittaristoja, samalla kuin IT-toiminto ylläpitää ja hallitsee dataa niiden takana. Qlik Sense vaatii käyttäjältä kuitenkin enemmän osaamista ja datan ymmärtämistä kuin esimerkiksi Tableau. (Gartner 2015) Tutkimuksessa käytetty versio, on Qlik Sensen tutkimushetkellä uusin saatavilla oleva versio 2.1.1.

6. TESTAUKSEN TULOKSET JA VERTAILU

Tässä luvussa avataan tutkimuksen empiirisen osan tuloksia. Tulokset on jäsennetty tutkimuksen tutkimussuunnitelman testitapausten mukaisesti. Ensin käsitellään tutkijan näkökulmasta yhteydenmuodostusta ja testauksen aloitusta, sitten datan olemassaolon ja korkean tason tarkastelua ja lopuksi tuntien tarkastelua. Tämän jälkeen esitetään koehenkilöiden saamat testaustulokset ja arvioidaan tuloksia.

6.1 Testauksen tulokset

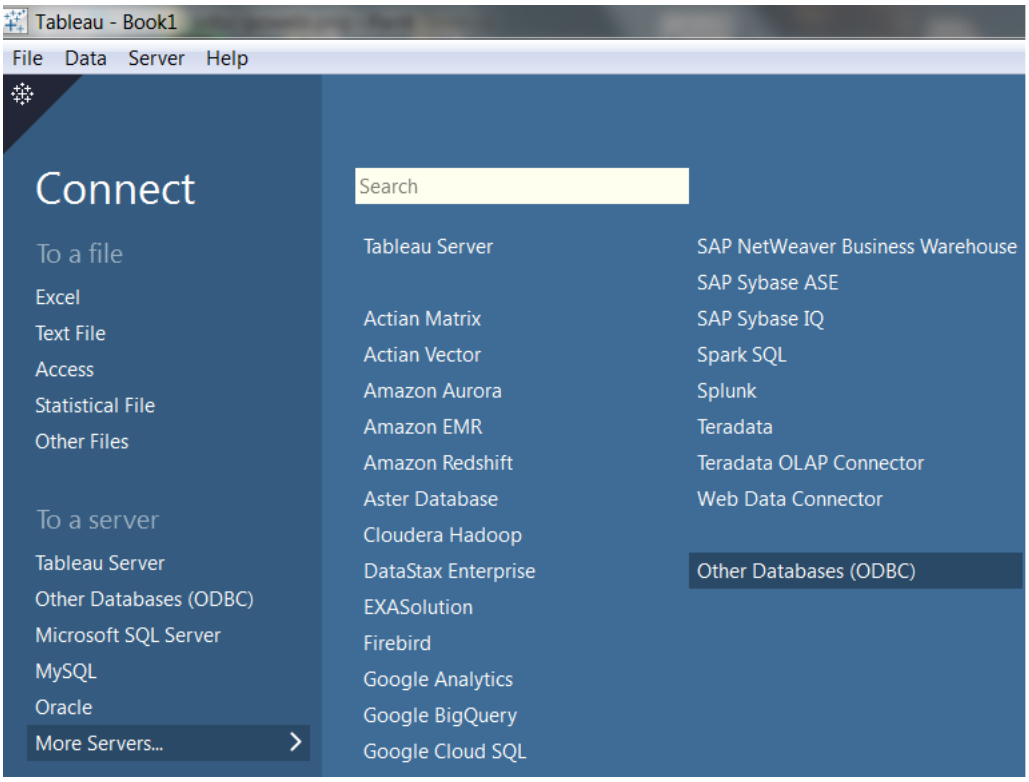
Testausprosessi ja sen tulokset on raportoitu kuvakaappauksien ja sanallisten selitysten avulla. Testauksessa käytettiin testaussuunnitelmassa määriteltyjä testitapauksia ja muita testisuunnitelman määrittelemiä asioita. Testaus suoritettiin testitapausten mukaan jaoteltuna viidessä osassa:

1) yhteydenmuodostus
2) aloitus
3) datan olemassaolon tarkastelu
4) datan korkean tason tarkastelu
5) datan tuntitasoinen tarkastelu.

Testitapaukset suoritettiin numerojärjestyksessä niin, että yksi tapaus suoritettiin kaikilla kolmella raportointityökalulla ja sen jälkeen siirryttiin seuraavaan tapaukseen. Luvuissa 6.1.1.-6.1.5 on esitetty testauksen tulokset suoritusjärjestyksessä. Testitapaus suoritettiin aina ensin Tableaulla, sitten Power BI:lla ja viimeiseksi Qlik Sensella.

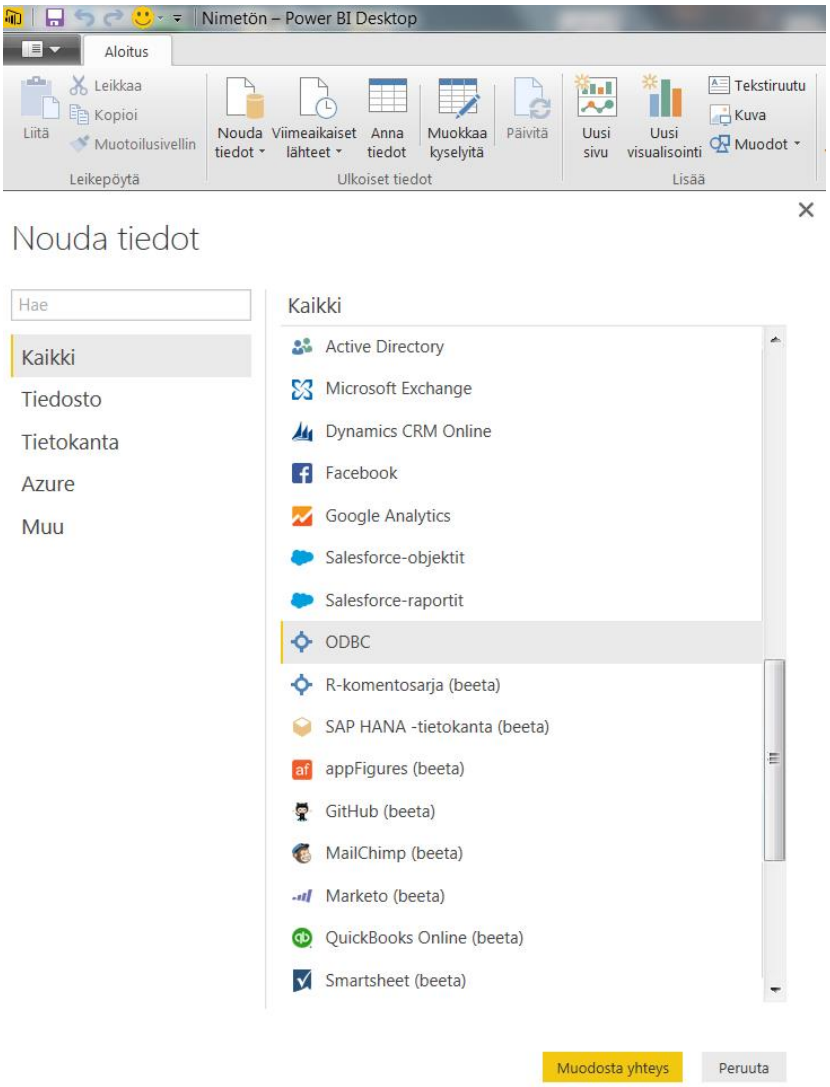
6.1.1 Yhteydenmuodostus

Raportointityökaluilla testaaminen alkaa muodostamalla yhteys tietokannan ja työkalujen välille. Testattavaan dataan on luotu ensin näkymä kehittäjän toimesta. Näkymä on havainnollistettu liitteessä A. Tässä tutkimuksessa yhteys muodostettiin MapR Drill ODBC-ajurin avulla. Ajurille oli määritelty Windows-käyttöjärjestelmässä kehittäjän tarjoamat yhteydenmuodostukseen tarvittavat tiedot eli osoite ja salasana. Sama ajuri toimi samoilla määrityksillä jokaisessa työkalussa. Kaikki testauksessa käytetyt raportointityökalut tukevat sovelluksen ja tietokannan välisiä ODBC-yhteyksiä ja tietokantayhteyden muodostaminen ODBC-yhteyden avulla on helppoa. Tableau-työkalun kanssa, aukeaa heti sovelluksen auetessa ”*Connect Data*”-ikkuna, eli ikkuna, jossa yhteydenmuodostus tehdään. Tableau tarjoaa monia yhteyksiä valmiiksi, ja listasta löytyy myös ODBC-yhteys. Kuvassa 6.1. on havainnollistettu Tableaun yhteydenmuodostusikkuna.

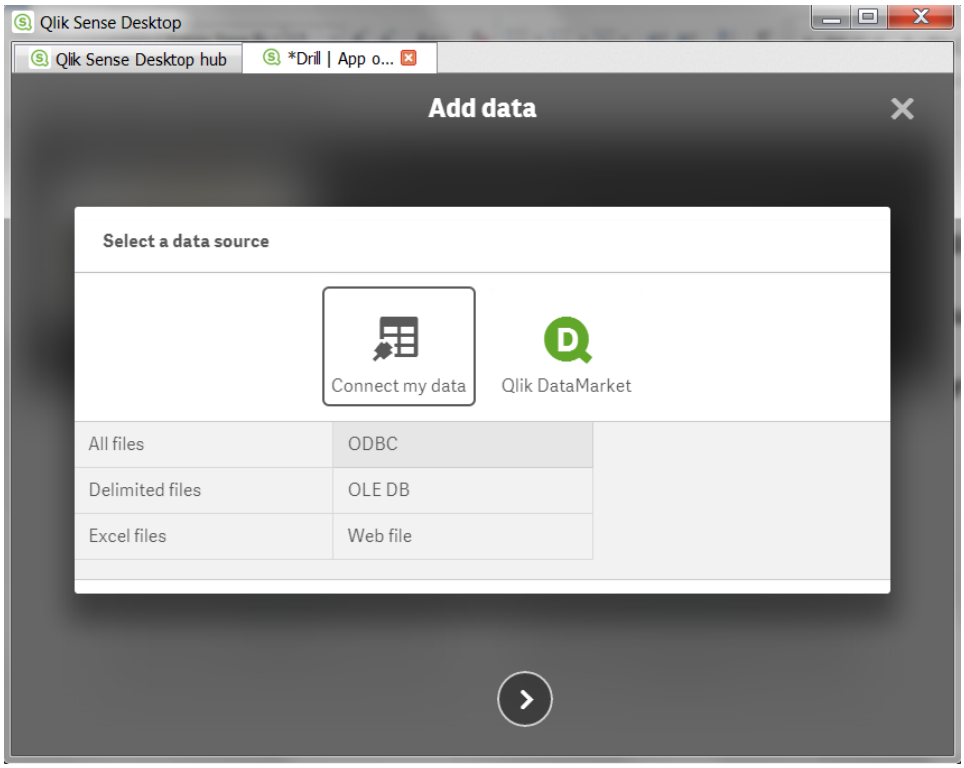


Kuva 6.1. *Datayhteyden muodostaminen Tableaussa*

Kun kuvan 6.1. yhteydenmuodostusikkunassa valitsee ODBC-vaihtoehdon, täytyy vain valita aiemmin määritelty ajuri ja antaa mahdollisesti salasana ja tämän jälkeen yhteydenmuodostus on valmis. Power BI:n ja Qlik Sensen yhteydenmuodostus toimii vastaavasti ja ne on kuvattu kuvissa 6.2. ja 6.3.



Kuva 6.2. *Datayhteyden muodostus Power BI:ssa*

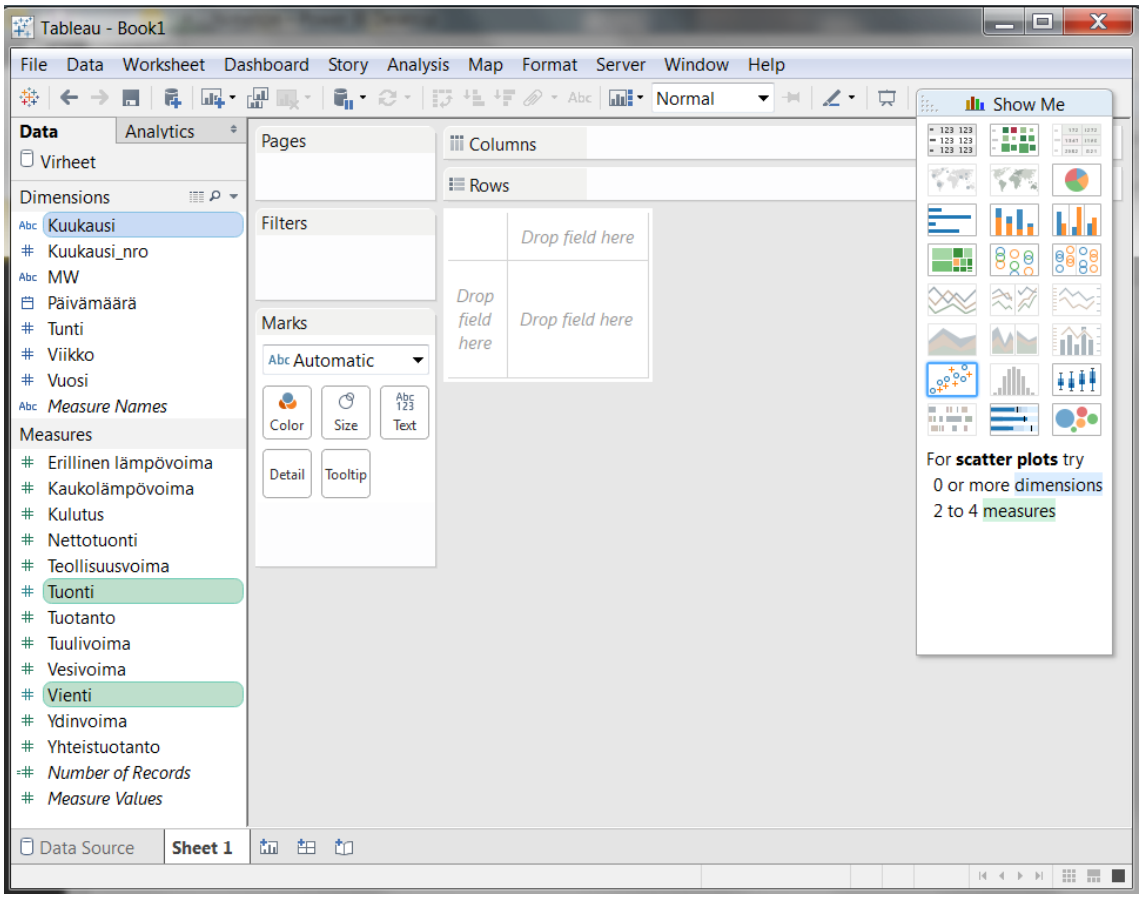


Kuva 6.3. *Datayhteyden muodostus Qlik Sensessa*

Kuvista 6.2. ja 6.3. voidaan nähdä, että ODBC yhteys on helppo muodostaa kaikissa raportointityökalussa vain valitsemalla oikean vaihtoehdon listasta. Tätä ennen yhteyden tiedot on tullut kuitenkin määrittää ODBC-ajurin määrittämisen yhteydessä, mutta sama ajuri toimii onneksi kaikissa raportointityökaluissa.

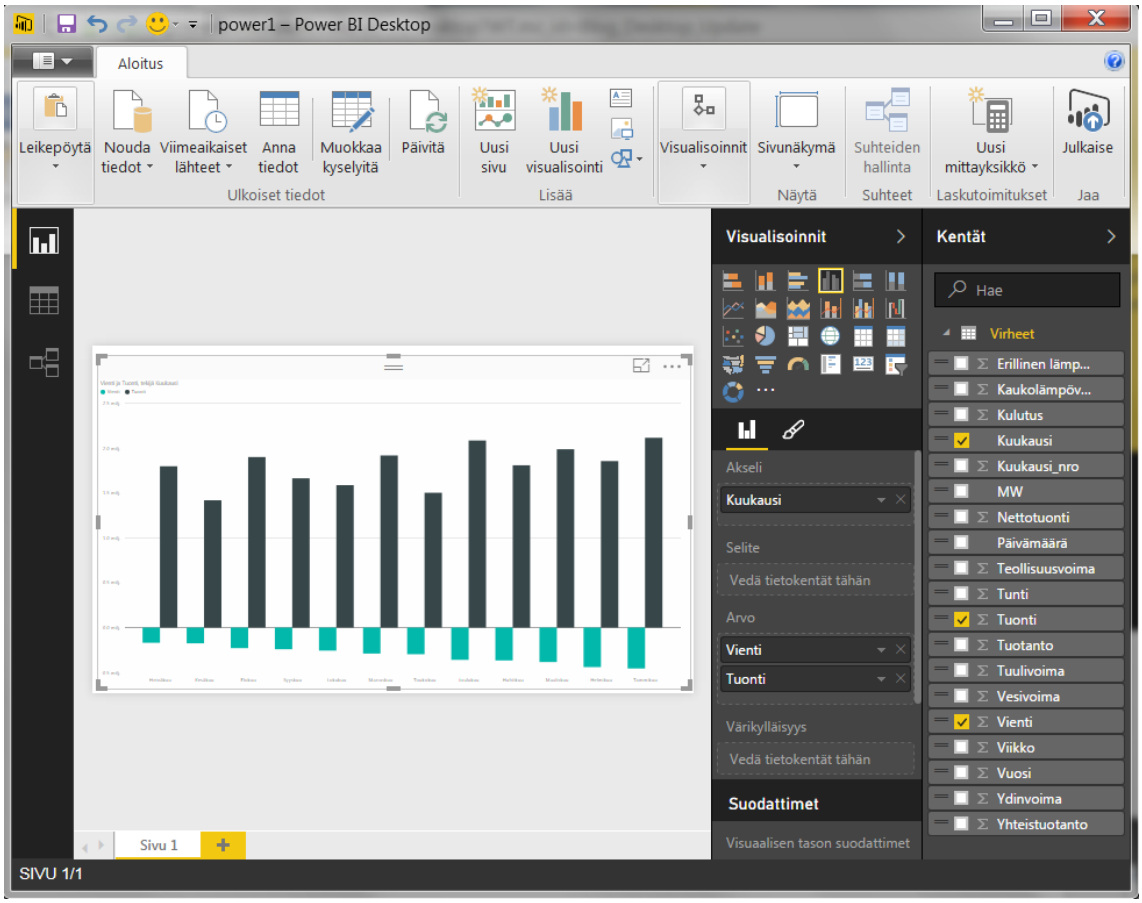
6.1.2 Aloitus

Kun raportointityökalut on yhdistetty datalähteeseen, voidaan aloittaa datan tarkastelu. Kokeillaan aluksi, miten voidaan valita datasta *Kuukausi*, *Tuonti* ja *Vienti* -kentät, ja katsotaan, ehdottaako työkalu suoraan visualisaatiota ja minkä laskennan visualisaatiota. Aloitettaessa raportointityökalu Tableaun käyttö tietokannan datan kanssa, näyttää ikkuna kuvan 6.4. mukaiselta.



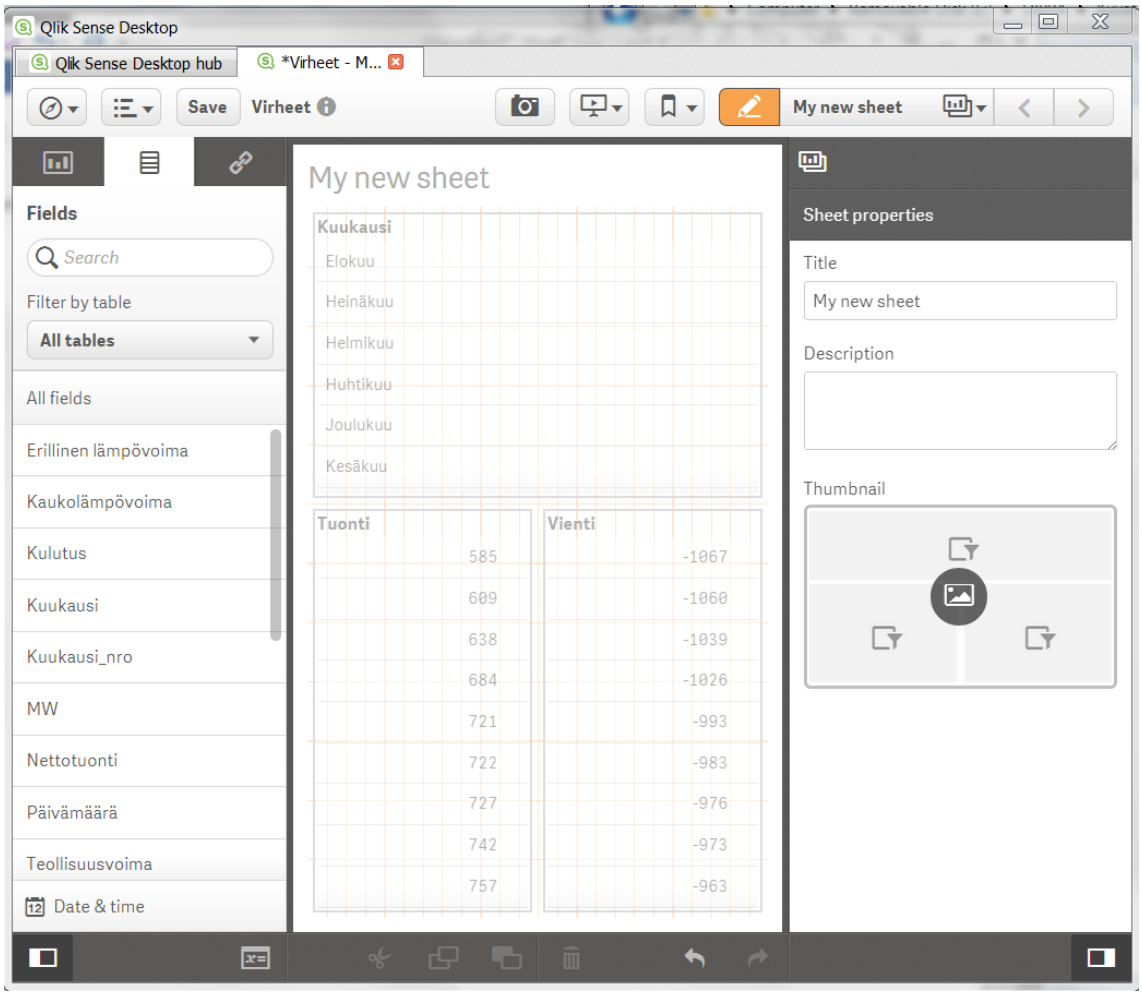
Kuva 6.4. *Aloitus Tableaussa*

Tableau tunnistaa kaikki numeromerkkiset kentät oletuksena mittaritiedoiksi eli ”*Measures*”-kentiksi, ja käyttäjän on itse siirrettävä dimensiokentät ”*Dimensions*”-laatikkoon. Kirjainmerkkiset kentät ovat automaattisesti dimensioita. Kuvassa 6.4. dimensioiksi on siirretty datassa olevat dimensiot numeromerkkiset *Kuukausi_nro*, *Tunti*, *Viikko* ja *Vuosi*. Kun dimensiot on määritelty, on Tableaun helpompi luoda sopivia visualisaatioita ja laskentoja. Tableaussa kenttiä voi käsitellä valitsemalla kentän listasta, jos haluaa valita useita kerralla, on käytettävä *Ctrl+valitse*. Kuvassa 6.4. on valittu *Kuukausi*, *Tuonti* ja *Vienti* -kentät ja Tableau ehdottaa ”*Show Me*”-toiminnossa, mitä visualisaatiota voidaan käyttää, mutta ei luo valmiiksi mitään visualisaatiota, vaan käyttäjän on valittava haluttu visualisaatio ehdotettujen joukosta. Ensisijainen ehdotus on kuitenkin korostettu sinisellä värillä ”*Show Me*”-laatikossa. Tableaun ehdottamat visualisaatiot perustuvat *Tuonti* ja *Vienti* kenttien summaukseen *Kuukausi*-tasolla. Aloitettaessa raportointityökalu Power BI:n käyttö tietokannan datan kanssa, näyttää ikkuna kuvan 6.5. mukaiselta.



Kuva 6.5. *Aloitus Power BI:ssa*

Kuten kuvasta 6.5. nähdään, näyttää Power BI kaikki kentät samassa listassa. Kenttiä voi valita rastittamalla valintalaatikon. Power BI tekee automaattisesti ehdottamansa visualisoinnin valittujen kenttien perusteella, sivuvalikosta voi myös muuttaa haluamaansa visualisointimallia. Kuvan 6.5 visualisaatio perustuu *Tuonti* ja *Vienti* kenttien summaukseen *Kuukausi*-tasolla. Aloitettaessa puolestaan raportointityökalu Qlik Sensen käyttö datalähteen kanssa, näyttää ikkuna puolestaan kuvan 6.6. mukaiselta.



Kuva 6.6. *Aloitus Qlik Sensessa*

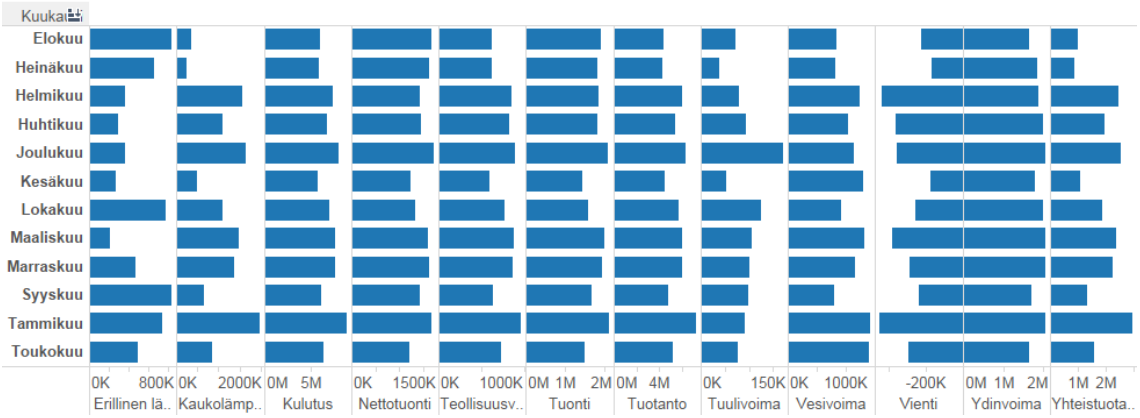
Kuten kuvasta 6.6. nähdään, Qlik Sense tuo kaikki datayhteyden kentät, joista täytyy itse tehdä haluamansa dimensiot. Qlik Sense ei ehdota mitään visualisaatiota, eikä kenttälistauksesta valitsemalla tapahdu mitään ja voi valita vain yhden kentän kerrallaan. Jos kentän raahaa työalueelle, tulee listamainen esitys siitä kentästä, mutta jos raahaa toisenkin niin tulee toinen lista. Jos haluaa visualisoinnin tai laskennan, käyttäjän täytyy ne itse tehdä.

Yhteydenmuodostuksen jälkeen esittävät kaikki raportointityökalut määritellyn datan kentät listamuodossa. Power BI ohjaa eniten käyttäjää sen suhteen, mikä visualisaatio sopisi valituille kentille ja muodostaa jo valmiiksi visualisaation. Kun halutaan tarkastella nopeasti ja helposti dataa, sopii Power BI siihen parhaiten, sillä käyttäjän ei tarvitse tehdä tai osata muuta kuin valita haluamansa kentät listasta ja valinnan perusteella syntyy valmis visualisaatio. Myös Tableau ehdottaa visualisaatiota valittujen kenttien pohjalta, mutta muodostaa sen vasta, kun käyttäjä valitsee haluamansa visualisaation ehdotettujen visualisaatioiden joukosta. Qlik Sense ei puolestaan ehdota mitään, ja käyttäjän on itse osattava määrittää dimensioita ja mittarilukuja ja niille sopivia laskentoja ja visualisaatiota. Qlik Sense siis suoriutuu heikoiten visualisointien ohjauksessa. Nopeaan ja helppoon datan tarkasteluun ja visualisointiin toimii Power BI siis parhaiten ja Tableau

lähes yhtä hyvin. Qlik Sense ei puolestaan mahdollista nopeaa tarkastelua ja vaatii käyttäjältä eniten osaamista.

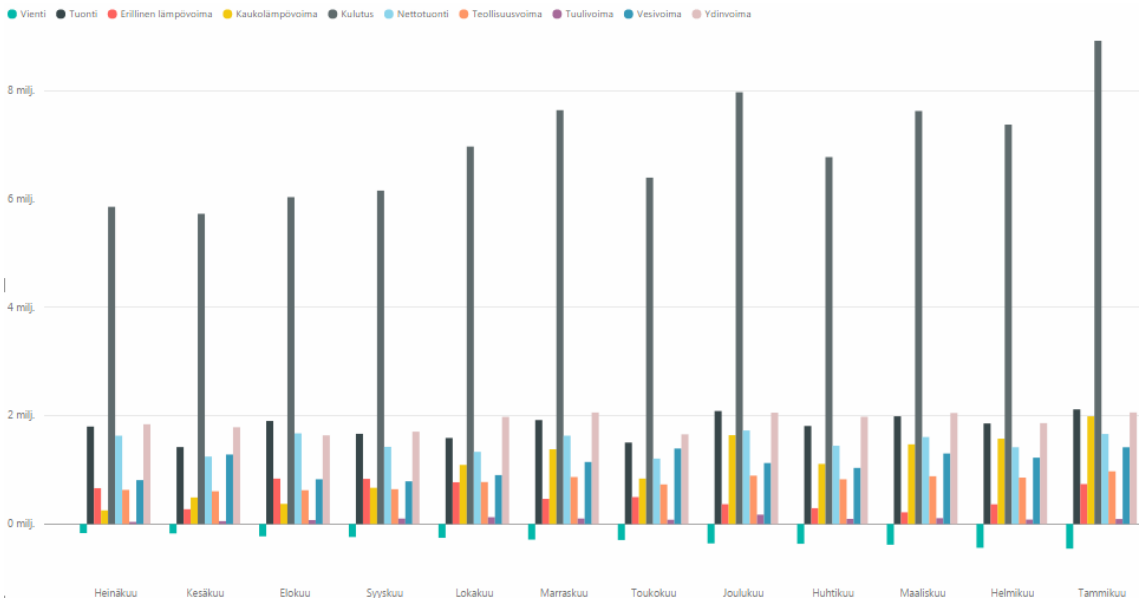
6.1.3 Datan olemassaolon tarkistus

Korkeatasoinen datan tarkistus tehdään ensin tarkistamalla datan olemassaolo eli, että data on yleensäkään paikallaan muodostamalla datasta helppo ja kattava kuvaaja. Tämän jälkeen voidaan jatkaa korkean tason tarkastelua vertaamalla kuukausittaisia summia. Ensin halutaan siis tarkistaa, että dataa on kaikille kuukausille ja kaikille sähköluokille. Tätä varten valitaan tarkasteltaviksi kaikki sähköluokat ja *Kuukausi*-kenttä ja käytetään raportointityökalun ehdottamaa visualisaatiota. *Kuukausi*-kenttä on merkkijono, eikä varsinaiseesti aikadimensio, joten samalla tulee testattua, osaavathan raportointityökalut erottaa merkkijonon dimensioksi. Tableaun ensisijaisesti ehdottama visualisaatio kuukausisummista on esitetty kuvassa 6.7.



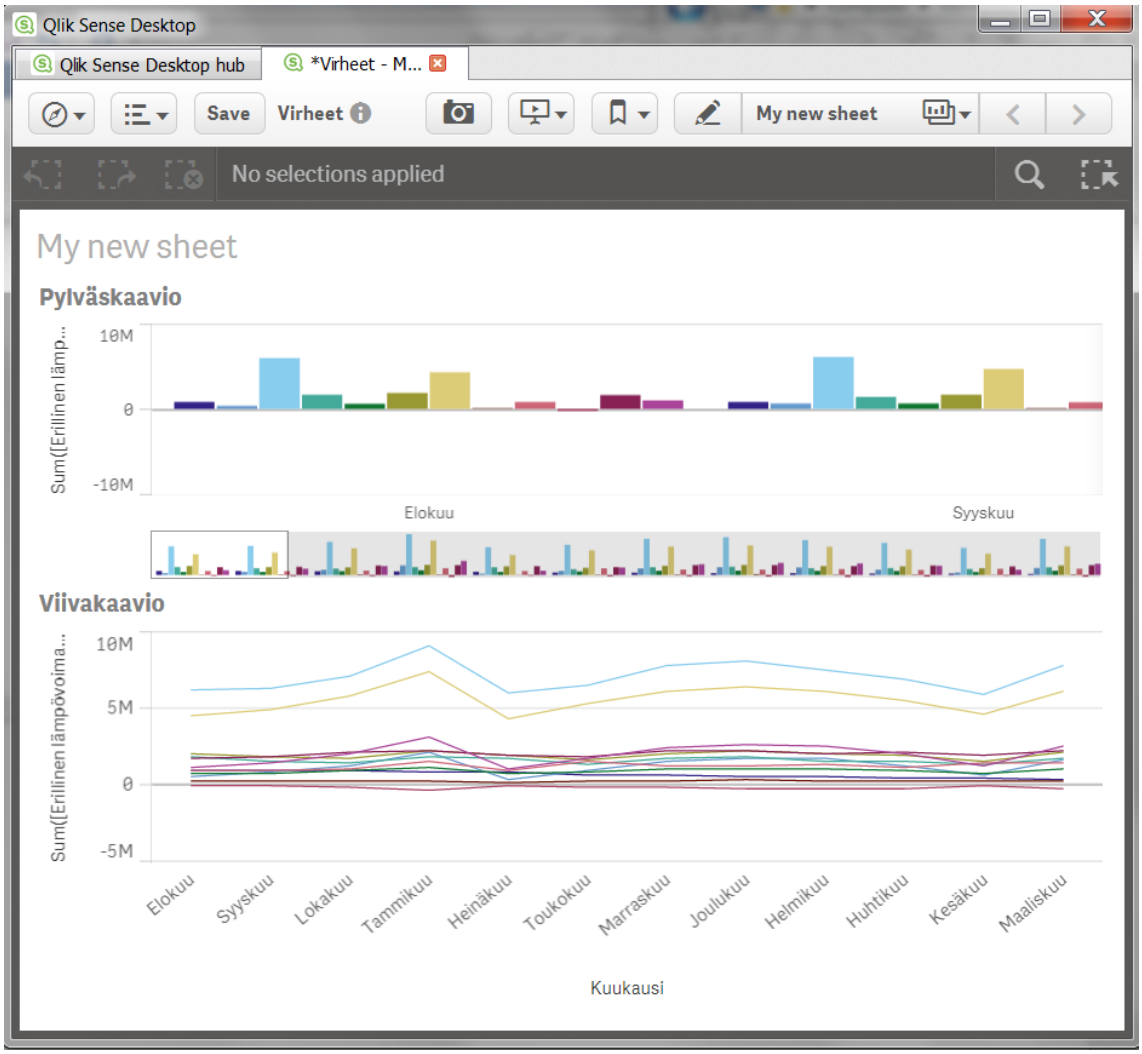
Kuva 6.7. *Tableaun ehdottama visualisaatio sähköluokkien summista kuukausitasolla*

Kuten kuvasta 6.7. huomataan, Tableau ehdottaa jokaiselle sähköluokille omaa pylväskaaviota kuukausitasolla. Kuvasta voidaan helposti nähdä, että joka kuukaudella ja jokaisella sähköluokalla on dataa. Palkkeja ei voida kuitenkaan vertailla eri luokkien kesken, sillä niillä on eri mitta-asteikot. Power BI ehdottaa samasta datasta puolestaan kuvan 6.8. mukaista visualisaatiota.



Kuva 6.8. *Power BI:n ehdottama visualisaatio sähköluokkien summista kuukausitasolla*

Kuten kuvasta 6.8. nähdään, ehdottaa Power BI kuukausitasoista pylväskaaviota, jossa kaikki sähköluokat ovat samassa kaaviossa eli voidaan nähdä, mitä luokkaa on eniten. Myös tästä kaaviosta nähdään kohtalaisen helposti olennainen eli, että kaikille kuukausille ja sähköluokille on dataa. Qlik Sense puolestaan ei ehdota valmiiksi visualisaatiota. Käyttäjän täytyy itse raahata työalueelle pylväskaavio ja valita siihen dimensioksi kuukausi ja sen jälkeen valita erikseen jokainen sähköluokka ja mitä laskentaa käytetään, tässä tapauksessa summausta. Kokeillaan muodostaa pylväskaavio ja vertailuksi viivakaavio. Qlik Sensen pylväs- ja viivakaavio on esitetty kuvassa 6.9.



Kuva 6.9. *Sähköluokkien kuukausisummat Qlik Sensessa*

Kuvan 6.9. pylväskaaviosta huomataan, että koko aikaväli ei näy kerralla, sillä pylväät ovat kohtuuttoman paksuja. Kokeillaan ”*Convert to line chart*”, eli vaihdetaan visualisaatioksi viivakaavio. Viivakaaviossa näkyy aikaväli, mutta viivojakin on vaikea tulkita. Nähdään kuitenkin, että kaikki kuukaudet ovat paikallaan, mutta sähköluokkia on hieman haastavampi tarkastella.

Kuvia 6.7., 6.8. ja 6.9. vertailemalla huomataan, että kuvan 6.7. Tableaun ehdottama visualisaatio kertoo helpoiten, että kaikille kuukausille on dataa kaikissa sähköluokissa. Power BI:n visualisaatio vaatii hieman enemmän tulkintaa, sillä joka kuukaudella pitäisi tarkistaa, onko jokaista sähköluokkaa vastaava pylväs kohdallaan. Qlik Sense puolestaan pakotti käyttäjän itse kokeilemaan, mikä visualisaatio toimisi ja tätä ennen kaikki sähköluokat piti määrittää summiksi, joten Qlik Sensen käyttö oli selvästi työläintä ja kuten kuvasta 6.9. nähdään, on kuukausittaisten sähkösummien seuraaminen myös selvästi haastavinta Qlik Sensessa.

6.1.4 Datan korkean tason tarkistus

Jatketaan korkean tason tarkastelua vertaamalla kuukausittaisia summia sähköluokista alkuperäiseen aineistoon. Alkuperäiset summat yhdeksän desimaalin tarkkuudella on esitetty liitteessä B. Käytetään kuukausittaisten summien muodostamiseen *Päivämäärä*-kentän kuukausitasoa, joka on raportointityökalujen ymmärtämä aikadimensio. Kuukausitasojen summien vertailulla voidaan huomata, missä kuukausissa on eroa alkuperäisen ja tietokannasta otetun datan välillä eli mitä kuukausia tulisi tarkastella tarkemmin. Valitaan halutut kentät ja käytetään esitysmuotona taulukkoa, sillä siitä nähdään yksittäiset luvut parhaiten. Tableaussa muodostettu taulukko kuukausisummista sähköluokittain on esitetty kuvassa 6.10.

Month of Päivämäärä	Erillinen lämpövoima	Kaukoläm..	Kulutus	Nettotuonti	Teollisuus..	Tuonti	Tuotanto	Tuulivoima	Vesivoima	Vienti	Ydinvoima	Yhteistuot..
Null	1 330	1 262	9 411	1 746	999	2 369	7 665	34	2 172	-623	1 868	2 260
tammikuu	736 353	1 987 453	8 920 608	1 660 142	971 264	2 115 405	7 240 634	90 549	1 415 681	-455 263	2 057 605	2 958 717
helmikuu	362 338	1 574 119	7 373 299	1 415 921	855 497	1 855 730	5 957 378	80 419	1 223 826	-439 809	1 861 179	2 429 616
maaliskuu	214 654	1 469 443	7 624 441	1 604 514	880 424	1 987 915	6 019 927	106 614	1 300 644	-383 401	2 048 147	2 349 867
huhtikuu	290 982	1 108 378	6 773 655	1 443 315	824 487	1 808 930	5 330 340	93 374	1 034 760	-365 615	1 978 358	1 932 866
toukokuu	493 879	835 301	6 376 364	1 203 596	727 430	1 498 625	5 172 768	77 464	1 389 064	-295 029	1 649 630	1 562 731
kesäkuu	271 244	488 761	5 726 887	1 244 287	603 153	1 419 085	4 482 599	52 183	1 281 929	-174 797	1 785 330	1 091 914
heinäkuu	660 821	250 293	5 855 527	1 629 175	628 722	1 797 953	4 226 352	37 471	808 922	-168 778	1 840 123	879 015
elokuu	836 753	370 473	6 034 304	1 672 310	622 344	1 900 790	4 361 993	72 181	824 760	-228 480	1 635 483	992 817
syyskuu	832 364	667 643	6 154 451	1 423 336	641 436	1 663 972	4 731 115	99 080	787 081	-240 636	1 703 511	1 309 078
lokakuu	769 570	1 089 991	6 967 641	1 331 621	773 361	1 587 643	5 636 020	124 358	901 944	-256 023	1 976 797	1 863 352
marraskuu	463 572	1 379 716	7 647 716	1 630 433	867 137	1 920 115	6 017 283	100 019	1 145 510	-289 682	2 059 567	2 246 852
joulukuu	364 208	1 639 123	7 970 692	1 725 900	891 593	2 084 867	6 244 793	171 504	1 124 548	-358 967	2 053 815	2 530 716

Kuva 6.10. *Kuukausisumma-taulukko Tableaussa*

Kuten kuvan 6.10. taulukosta voidaan huomata, tulee taulukosta todella leveä ja Tableau jättääkin esitystavasta desimaalit pois. Desimaaleja voidaan kuitenkin tarkastella erikseen, jos on tarve. Tableau tunnistaa ’31.01.2014’ muotoisen päivämääräkentän ja osaa automaattisesti muodostaa aikahierarkian *vuosi-neljännes-kuukausi-päivä*. Mitä tahansa hierarkiatasoa näistä voidaan käyttää dimensiona ja tässä kuvassa on käytetty kuukausidimensiota. Kun tarkastellaan kuva 6.10. taulukon dataa, huomataan että ensimmäisen rivin kuukausitietona on tyhjä kenttä eli ”Null” eli datasta löytyy kaikille sähköluokille lukemia, joille ei ole määritelty *Päivämäärä*-tietoa, eli *Päivämäärä*-kenttä puuttuu jostain kohtaa dataa. Vertaamalla kuukausisummia alkuperäiseen dataan, huomataan summien eroavan alkuperäisestä tammi-, touko-, heinä-, loka-, marras- ja joulukuussa, eli kaikkina kuukausina, joihin on tehty tahallisia virheitä. Esimerkiksi helmikuuhun ei puolestaan tehty virheitä ja sen data on kaikilla sähköluokilla oikein. Voidaan siis huomata, että jopa yhden tunnin virheellinen sähkölukema näkyy heti korkeamman tason summissa, joten loppukäyttäjä voi verrata datan oikeellisuutta tarkistamalla korkean tason summia.

Kuukausi	Yhteistuotanto	Ydinvoima	Vienti	Vesivoima	Tuulivoima	Tuotanto	Tuonti	Teollisuusvoima	Nettotuonti	Kulutus	Kaukolämpövoima	Erillinen lämpövoima
	2,260.44	1,867.69	-622.59	2,172.13	34.41	7,665.00	2,368.63	998.74	1,746.04	9,411.04	1,261.69	1,330.34
tammikuu	2,958,716.52	2,057,605.45	-455,263.29	1,415,681.19	90,548.76	7,240,634.39	2,115,404.99	971,263.84	1,660,141.70	8,920,607.79	1,987,452.68	736,352.95
helmikuu	2,429,616.27	1,861,178.67	-439,809.32	1,223,826.24	80,418.76	5,957,378.42	1,855,729.99	855,496.82	1,415,920.67	7,373,299.09	1,574,119.45	362,338.48
maaliskuu	2,349,867.46	2,048,147.24	-383,401.30	1,300,644.14	106,614.14	6,019,927.27	1,987,915.21	880,424.14	1,604,513.91	7,624,441.18	1,469,443.32	214,654.29
huhtikuu	1,932,865.73	1,978,358.14	-365,615.22	1,034,760.24	93,373.89	5,330,340.17	1,808,929.72	824,487.26	1,443,314.50	6,773,654.67	1,108,378.47	290,982.17
toukokuu	1,562,730.99	1,649,629.70	-295,028.66	1,389,063.90	77,464.47	5,172,767.93	1,498,624.85	727,430.04	1,203,596.19	6,376,364.12	835,300.95	493,878.86
kesäkuu	1,091,913.53	1,785,330.00	-174,797.03	1,281,929.32	52,182.75	4,482,599.39	1,419,084.51	603,152.66	1,244,287.48	5,726,886.87	488,760.87	271,243.78
heinäkuu	879,014.87	1,840,123.35	-168,778.42	808,921.62	37,471.27	4,226,351.90	1,797,953.33	628,722.04	1,629,174.91	5,855,526.81	250,292.83	660,820.80
elokuu	992,816.74	1,635,482.93	-228,480.31	824,759.84	72,180.73	4,361,993.42	1,900,790.49	622,344.14	1,672,310.18	6,034,303.60	370,472.60	836,753.18
syyskuu	1,309,078.40	1,703,511.44	-240,635.69	787,081.32	99,079.59	4,731,115.06	1,663,972.01	641,435.55	1,423,336.33	6,154,451.39	667,642.85	832,364.31
lokakuu	1,863,351.74	1,976,796.56	-256,022.67	901,944.49	124,357.94	5,636,020.37	1,587,643.47	773,360.54	1,331,620.80	6,967,641.17	1,089,991.20	769,569.64
marraskuu	2,246,852.42	2,059,567.34	-289,682.17	1,145,509.88	100,018.52	6,017,283.33	1,920,115.09	867,136.85	1,630,432.92	7,647,716.26	1,379,715.57	463,572.16
joulukuu	2,530,716.44	2,053,815.17	-358,967.22	1,124,548.27	171,504.18	6,244,792.52	2,084,867.04	891,592.96	1,725,899.83	7,970,692.34	1,639,123.48	364,208.46
Yhteensä	**22,149,801.54**	**22,651,413.68**	**-3,657,103.88**	**13,240,842.57**	**1,105,249.41**	**65,428,869.18**	**21,643,399.33**	**9,287,845.58**	**17,986,295.46**	**83,434,996.33**	**12,861,955.96**	**6,298,069.42**

Kuva 6.11. *Kuukausisumma-taulukko Power BI:ssa*

Kuvassa 6.11. on esitetty sama kuukausittaisia summia esittävä taulukko Power BI:n avulla tehtynä. Power BI:n taulukko on hyvin vastaava kuin Tableaunkin taulukko, erona ainoastaan. että Power BI esittää oletusarvoisesti luvut kahden desimaalin tarkkuudella. Myös Power BI tunnistaa ’31.01.2014’ muotoisen päivämääräkentän ja osaa automaattisesti muodostaa aikahierarkian *vuosi-neljännes-kuukausi-päivä*. Kun tarkastellaan kuvaa 6.11., voidaan myös huomata, että myös Power BI tunnistaa, että datan joukossa on sähkölukemia, joille ei ole *Päivämäärä*-tietoa ja sähkölukemat näkyvät tyhjän kuukausikentän kohdalla. Kuukausisummia voidaan verrata samalla tavalla kuin Tableaussakin ja huomataan summien eroavan alkuperäisestä tammi-, touko-, heinä-, loka-, marras- ja joulukuussa eli yhden tunnin virheellinen sähkölukema näkyy heti korkeamman tason summissa. Kuvassa 6.12. on puolestaan esitetty taulukko Qlik Sensella luotuna.

Kuukausi	Sum([Erillinen lämpövoima])	Sum([Kaukolämpövoima])	Sum(Kulutus)	Sum(Nettotuonti)	Sum(Teollisuusvoima)	Sum(Tuonti)	Sum(Tuotanto)	Sum(Tuulivoima)	Sum(Vesivoima)
Totals	**6298069**	**12861956**	**83434996**	**17986295**	**9287846**	**21643399**	**65439475**	**1105249**	**13240843**
Tammikuu	736353	1987453	8920608	1660142	971264	2115405	7251240	90549	1415681
Toukokuu	495845	837630	6393750	1206395	729446	1502806	5187355	77519	1392760
Maaliskuu	214654	1469443	7624441	1604514	880424	1987915	6019927	106614	1300644
Kesäkuu	271244	488761	5726887	1244287	603153	1419085	4482599	52183	1281929
Helmikuu	362338	1574119	7373299	1415921	855497	1855730	5957378	80419	1223826
Marraskuu	462936	1378649	7639741	1629380	866119	1918303	6010362	99999	1143986
Joulukuu	364208	1639123	7970692	1725900	891593	2084867	6244793	171504	1124548
Huhtikuu	290982	1108378	6773655	1443315	824487	1808930	5330340	93374	1034760
Lokakuu	769570	1089991	6967641	1331621	773361	1587643	5636020	124358	901944
Elokuu	836753	370473	6034304	1672310	622344	1900790	4361993	72181	824760
Heinäkuu	660821	250293	5855527	1629175	628722	1797953	4226352	37471	808922
Syyskuu	832364	667643	6154451	1423336	641436	1663972	4731115	99080	787081

Kuva 6.12. *Kuukausisumma-taulukko Qlik Sensessa*

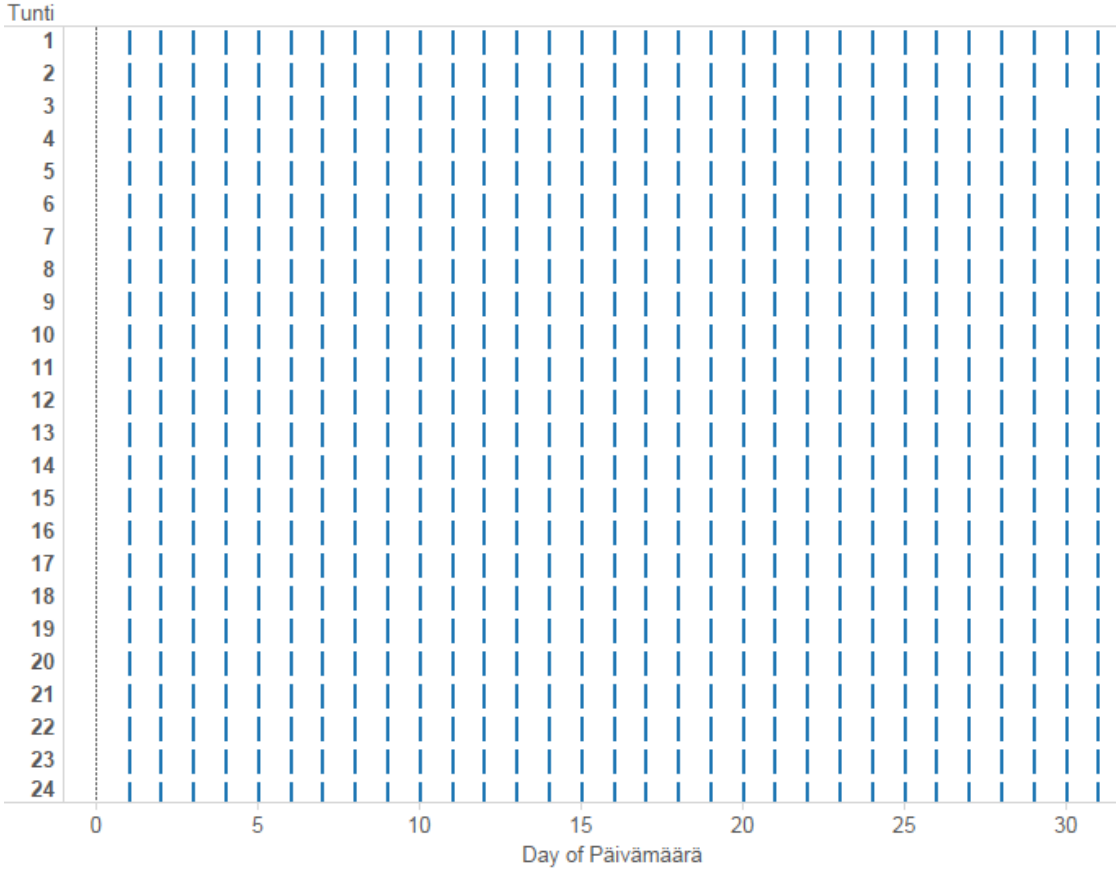
Kuvasta 6.12. huomataan, että Qlik Sensella tehty taulukko on todella leveä, eivätkä kaikki sähköluokat silti edes mahdu esitettävään taulukkoon, vaikka taulukossa ei ole desimaaleja. Jos haluttaisiin esittää kaikki luvut kerralla, tulisi luoda kaksi erillistä, allekkaista taulukkoa tai muokata taulukkoa, mikä olisi työlästä. Qlik Sense ei tunnista automaattisesti *Päivämäärä*-kenttää aikadimensioksi, joten peruskäyttäjä käyttää tässä tapauksessa merkkijonoista *Kuukausi*-kenttää, jolloin testauksessa jää huomaamatta, että *Päivämäärä*-kenttä puuttuu osasta dataa. Lisäksi jää huomaamatta kaikki muutkin virheet *Päivämäärä*-kentän osalta. Kokeneempi käyttäjä voisi itse luoda hierarkkisen aikadimension, mutta sekin olisi työlästä. Muuten kuukausisummista voidaan huomata samat virheet kuin Tableaun ja Power BI:n kohdallakin. Kuukausitasoisten summien tapauksessa Tableau ja Power BI ovat siis huomattavasti helppokäyttöisempiä ja informatiivisempia kuin Qlik Sense. Power BI:lla on pieni etu Tableauhun nähden siinä

mielessä, että se esittää luvut kahden desimaalin tarkkuudella, jolloin voidaan tehdä tarkempia vertailuja alkuperäiseen dataan.

6.1.5 Tuntitasoinen tarkastelu

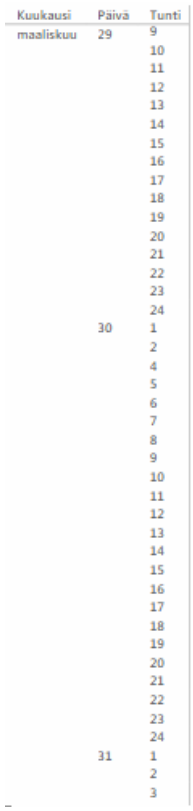
Testattavassa datassa on erityistapauksena huomioitu kesäaikaan siirtyminen, joka oli vuonna 2014 30. maaliskuuta klo 03.00. Tämän johdosta kalenterista hävisi yksi tunti ja päivän 30.3.2014 kohdalta puuttuu kaikki tiedot tunnilta numero 3. Tämä erityistapaus halutaan tarkistaa datasta, eli halutaan tietää, näkyykö puuttuva tunti myös testattavassa datassa puuttuvana ja eihän se vaikuta muihin tunteihin. Tätä testausta varten tehdään tuntitasoinen tarkastelu maaliskuulle, ja samalla voidaan testata, näkyisikö menetelmällä mikä tahansa puutuva tunti tietyn kuukauden tarkastelussa.

Luodaan testitapaus raportointityökaluilla valitsemalla *Tunti*-kenttä sekä *Päivämäärä*-kentän päivätaso ja asetetaan suodattimeksi maaliskuu. Käytetään raportointityökalun ehdottamaa visualisaatiota päivistä ja tunneista. Tableaulla suodattimen asettaminen on helppoa, raahataan vain haluttu suodatuskenttä ”*Filters*” laatikkoon ja valitaan haluttu suodatin, tässä tapauksessa maaliskuu. Tableaun ehdottama visualisaatio on esitetty kuvassa 6.13.



Kuva 6.13. *Päivä-tunti visualisaatio Tableaussa*

Kuten kuvasta 6.13. voidaan nähdä, esittää Tableau päivä-tunti datan visualisaationa, jossa päivä ja tunti muodostavat parin, ja jos pari on olemassa, niin sen kohdalla on sininen viiva. Kuvasta voidaan nähdä että pari ”tunti 3 - päivä 30” viiva puuttuu, joten kyseistä tuntia ei ole datan joukossa. Samalla voidaan tarkistaa, että kaikki muut parit löytyvät ja data on muiden tuntien osalta mukana datassa. Samat *Päivämäärä* ja *Tunti* valinnat ja niiden pohjalta ehdotettu visualisaatio Power BI:ssa on esitetty kuvassa 6.14. Power BI:ssa suodattimen asettaminen tapahtuu vastaavasti kuin Tableaussakin, raahaamalla haluttu kenttä ”*Suodattimet*” laatikkoon.



Kuukausi	Päivä	Tunti
maaliskuu	29	9
		10
		11
		12
		13
		14
		15
		16
		17
		18
		19
		20
		21
		22
		23
		24
	30	1
		2
		4
		5
		6
		7
		8
		9
		10
		11
		12
		13
		14
		15
		16
		17
		18
		19
		20
		21
		22
		23
		24
	31	1
		2
		3

Kuva 6.14. *Päivä-tunti visualisaatio Power BI:ssa*

Kuten kuvasta 6.14. nähdään, Power BI ei muodosta päivästä ja tunnista varsinaista visualisaatiota, vaan esittää ne ainoastaan taulukkomuodossa. Taulukko on pitkä ja jotta voidaan tarkastella kaikkia päiviä, on taulukkoa selattava ylös- ja alaspäin. Tiedettäessä mitä tuntia etsiä, nähdään taulukosta helposti, että 30.3.2014 tunti 3 puuttuu, mutta satunnainen tuntien tarkastelu on haastavampaa. Myös Qlik Sensella ainoa mahdollinen esitystapa kahdelle dimensiolle on taulukko ja se on esitetty kuvassa 6.15.

Päivämäärä	Tunti
29.3.2014	18
29.3.2014	19
29.3.2014	20
29.3.2014	21
29.3.2014	22
29.3.2014	23
29.3.2014	24
30.3.2014	1
30.3.2014	2
30.3.2014	4
30.3.2014	5
30.3.2014	6
30.3.2014	7
30.3.2014	8
30.3.2014	9
30.3.2014	10
30.3.2014	11

Kuva 6.15. *Päivä-tunti visualisaatio Qlik Sensessa*

Kuvasta 6.15. huomataan, että myös Qlik Sensessa visualisoitu muoto on pitkä taulukko, jota on selattava tietojen tarkastelemiseksi. Qlik Sensen taulukosta on kuitenkin Power BI:ta haastavampi huomata, milloin päivämäärä vaihtuu. Lisäksi automaattisen päivämäärä-hierarkia puuttuessa Qlik Sensesta, tuli sallittujen arvojen joukkoon suodattaa kaikki maaliskuun päivät erikseen ja lisätyötä tuotti myös taulukoon päätyminen visualisointivaihtoehtona ja sen luominen käyttäjän toimesta.

Vertaamalla kuvia 6.13, 6.14. ja 6.15., voidaan todeta, että Tableau tarjosi helpoimman keinon huomata puuttuvat tunnit. Yksittäisen tunnin pystyisi myös tarkistamaan Power BI:n tai Qlik Sensen taulukosta, mutta yleinen puuttuvien tuntien tarkastelu vaatisi taulukkomuodossa paljon tarkkuutta ja ylös-alas selailua. Tableaun visualisoidut päivä-tuntiparit loivat tehokkaan keinon visuaalisesti etsiä mahdollisesti puuttuvia arvoja ja mahdollistavat nopean ja helpon tuntien tarkastelun.

6.2 Raportointityökalujen vertailu

Kaikki tutkijan toimesta tutkitut raportointityökalut mahdollistivat pääsyn tutkittavaan dataan ja toimivat työkaluina tutkittavan datan visualisoinnissa. Raportointityökalujen välillä oli kuitenkin eroavaisuuksia helppokäyttöisyyden, nopeuden, loogisuuden ja graafisen esitysvoiman välillä. Raportointityökalujen vertailuksi tarkastellaan tuloksia testitapauksittain ja asetetaan raportointityökalut paremmuusjärjestykseen. Paremmuusjärjestys perustuu aiemmin mainittuihin attribuutteihin: helppokäyttöisyyteen, nopeuteen, loogisuuteen ja graafiseen esitysvoimaan. Paremmuusjärjestyksen määrittämiseksi testauksen suorittanut tutkija pisteyttää työkalut niin, että attribuuteiltaan paras työkalu saa kaksi pistettä, toiseksi paras yhden pisteen ja heikoin nolla pistettä. Tapauksessa, jossa raportointityökalut ovat tasavahvoja, saavat kaikki yhden pisteen. Tutkijan muodostama pisteytys testaustapauksittain on esitetty taulukossa 6.1.

Taulukko 6.1. *Raportointityökalujen pisteytys testitapauksittain*

Testitapaus	Tableau	Power BI	Qlik Sense
Yhteydenmuodostus	1	1	1
Aloitus	1	2	0
Datan olemassaolo	2	1	0
Korkean tason summatarkastelu	1	2	0
Tuntitarkastelu	2	1	0
Yhteensä	7	7	**1**

Taulukon 6.1. pisteytys on tehty luvussa 6.1 dokumentoitujen testaustulosten avulla. Kaikkien raportointityökalujen kohdalla yhteydenmuodostus oli helppoa, sillä datalähde valittiin vain listasta ja määritettiin yhteyden tiedot. Yhteydenmuodostus oli kaikilla raportointityökaluilla yhtä helppoa, joten kaikki työkalut saavat yhden pisteen.

Testauksen aloituksen ja aloitusnäkymän arvioinnissa työkaluilla oli huomattavia eroja. Tableaun käyttö oli aloitettava määrittelemällä tarvittavat numeromerkkiset kentät dimensioiksi. Tämän jälkeen kun valittiin kolme kenttää visualisoitaviksi, ehdotti Tableau visualisaatioita, joista käyttäjän oli valittava sopiva muoto. Power BI:lle ei puolestaan tarvinnut määrittää dimensioita erikseen ja se muodosti valmiin visualisaation valittujen kenttien pohjalta. Qlik Sensessa täytyi puolestaan määrittää itse kaikki dimensiot sekä visualisaatiot, joten se oli vaikeakäyttöisin ja työläin. Näin ollen Qlik

Sense oli vaikeakäyttöisin ja sai nolla pistettä, Tableau yhden pisteen, ja helpointen visualisaation muodostanut Power BI kaksi pistettä.

Datan olemassaolon tarkistuksessa Tableau ehdotti ensisijaisesti jokaiselle sähköluokalle omaa pylväskaaviota kuukausitasolla ja visualisaatiosta oli helppo huomata, että dataa oli jokaiselle kuukaudelle ja jokaiselle sähköluokalle. Power BI puolestaan ehdotti yhtä kuukausitasoista pylväskaaviota, jossa näkyvät kaikki sähköluokat eri värein ilmaistuina. Kaaviosta huomattiin heti, että kaikki kuukaudet ovat olemassa, mutta sähköluokkien tarkastelu vaati tarkkuutta eriväristen pylväiden kanssa. Qlik Sensessa visualisaation joutui puolestaan luomaan itse ja pylväskaavio oli epäselvä. Näin ollen Tableau suoriutui parhaiten, sillä Tableaun visualisaatiosta oli helpoin huomata datan olemassaolo. Tableau saa siis kaksi pistettä, Power BI yhden pisteen ja työläs ja epäselvä Qlik Sense nolla pistettä.

Datan korkean tason tarkastelussa verrattiin sähköluokkien kuukausittaisia summia tietokannan ja alkuperäisen datan välillä. Tableau ja Power BI tunnistivat *Päivämäärä*-kentän automaattisesti aikadimensioksi ja kun käytettiin tämän aikadimension kuukausitasoa, huomattiin, että on sähkölukemia, jotka eivät kohdistu millekään kuukaudelle, eli joilta puuttuu *Päivämäärä*-tieto. Qlik Sense ei puolestaan automaattisesti tunnistanut aikadimensiota, joten käytettiin merkkimuotoista *Kuukausi*-kenttää, jolloin puuttuva *Päivämäärä*-tieto jäi huomaamatta. Taulukko oli luontevin esitystapa kuukausitasoisille summille, ja kaikkien raportointityökalujen muodostamista taulukoista nähtiin eroavaisuudet kuukausisummissa verrattuna alkuperäiseen aineistoon. Power BI esitti summat tarkimmalla tasolla, joten se saa vertailussa kaksi pistettä, Tableau saa yhden pisteen ja automaattisen aikadimension puutteen vuoksi Qlik Sense saa nolla pistettä.

Tuntitasoisessa tarkastelussa Tableau oli selkeästi visuaalisesti vahvin esitysvoimaltaan. Tableaun visualisoinnissa pystyttiin silmämääräisesti etsimään puuttuvaa tietoa, ja kaikki tieto näkyi yhdellä vilauksella. Power BI ja Qlik Sense puolestaan esittivät päivän ja tunnin tarkastelun ainoastaan taulukkomuodossa, joten tuntien tarkastelua joutui tekemään selaamalla taulukkoa ylös ja alas. Power BI:n taulukko oli selkeämpi, sillä se vei vähemmän tilaa ja päivämäärän vaihtuminen oli helpommin huomattavissa. Näin ollen Tableau saa kaksi pistettä, Power BI yhden ja Qlik Sense jälleen nolla.

Kun testitapausten pisteytys lasketaan yhteen, asettuvat Tableau ja Power BI selvästi Qlik Sensen edelle, Qlik Sensen saadessa yhden pisteen ja muiden saadessa seitsemän pistettä. Tämän tutkimuksen puitteissa voidaan siis todeta, että Qlik Sensen sopivuus testaustyökaluksi on heikompi kuin Tableaun ja Power BI:n. Tableau ja Power BI ovat pisteytettyinä yhtä hyviä, mutta Tableau oli kuitenkin sekä datan olemassaolon tarkastelussa että tuntitasoisessa tarkastelussa selkeästi vahvin visuaalisesti, joten graafisen esitysvoiman ja nopeuden attribuuttien mukaan Tableau soveltuu hieman Power BI:ta paremmin NoSQL-tietokannan datan testaamiseen.

6.3 Koehenkilöiden suorittama testaus

Empiirisen testauksen suoritti tutkijan lisäksi myös kaksi ulkopuolista koehenkilöä. Koehenkilöille raportointi oli teoriassa tuttu asia, mutta he eivät olleet aiemmin käyttäneet mitään kolmesta testattavasta työkalusta. Koehenkilöiden tehtävänä oli suorittaa testausprosessi eli etsiä virheitä datasta, sekä arvioida ja pisteyttää työkalujen helppokäyttöisyys ja sopivuus kunkin testitapauksen yhteydessä.

Testaus suoritettiin niin, että tutkija antoi koehenkilöille testausssuunitelman ja testaustapaukset tarkasteltaviksi, joiden pohjalta he suorittivat testauksen samassa järjestyksessä kuin tutkijakin. Tutkija seurasi vierestä, kun koehenkilöt suorittivat testausta, mutta ei puuttunut testauksen kulkuun. Tutkija kirjoitti muistiinpanoina ylös testausten tulokset ja lisäksi koehenkilöt pisteyttivät työkalut kunkin testaustapauksen suhteen. Pisteytyksessä käytettiin samaa pisteytysjärjestelmää kuin taulukossa 6.1. Koehenkilöiden työkaluille antamat pisteet on esitetty taulukossa 6.2. niin, että koehenkilön 1 pisteet ovat tummanharmalla pohjalla, koehenkilön 2 pisteet keskiharmaalla ja tutkijan antamat pisteet vaaleimmalla pohjalla. Tutkijan antamat pisteet ovat samat kuin taulukossa 6.1. ja ne on esitetty myös tässä vertailun helpottamiseksi ja kokonaistuloksen esittämiseksi.

Taulukko 6.2. *Työkalujen kokonaispisteytys*

Testitapaus	Tableau			Power BI			Qlik Sense		
Yhteydenmuodostus	2	2	1	0	1	1	1	0	1
Aloitus	1	1	1	2	2	2	0	0	0
Datan olemassaolo	2	2	2	1	1	1	0	0	0
Korkean tason summatarkastelu	2	1	1	1	2	2	0	0	0
Tuntitarkastelu	2	2	2	1	1	1	0	0	0
Yhteensä	**9**	**8**	**7**	**6**	**7**	**7**	**1**	**0**	**1**
Kaikki yhteensä	**24**			**20**			**2**		

Ensimmäisen testitapauksen eli yhteydenmuodostuksen suhteen molemmat koehenkilöt totesivat työvaiheiden olleen helppoja kaikilla työkaluilla, ja yhteydenmuodostus onnistui lähes ongelmitta. Molemmat koehenkilöt kokivat Tableaun yhteydenmuodostuksen kaikkein loogisimmaksi ja selkeimmäksi, ja Tableau sai molemmilta kaksi pistettä. Power BI:n ja Qlik Sensen yhteydenmuodostus oli myös helppoa, mutta ei aivan yhtä suoraviivaista ja vaati useamman valinnan.

Toisen testitapauksen eli aloituksen suhteen koehenkilöt ja tutkija olivat yksimielisiä siitä, että Power BI:n aloitusnäkymä oli loogisin ja käytön aloittaminen helpointa. Toiseksi helpoin työkalu oli Tableau, ja Qlik Sensella oli kaikkien mielestä vaikeakäyttöisin aloitusnäkymä. Koehenkilöiden mielestä Power BI:n aloitusnäkymä oli loogisin, sillä siinä näkyi kaikki työkalut automaattisesti. Tableaulla visualisaatiot tuli itse klikata näkyviin ”*Show Me*” -toiminnon avulla. Toisaalta molemmat koehenkilöt preferoivat Tableaun tapaa esittää kaikki kentät vasemalla puolella, kun Power BI taas näytti kaikki kentät ja työkalut oikealla puolella aloitusnäkymää. Qlik Sensen aloitusnäkymä oli vaikeakäyttöisin, sillä koehenkilöiden oli vaikea löytää, missä kentät oli listattu ja miten datakenttiä sai visualisoitua.

Datan olemassaolon testauksessa koehenkilöt olivat tutkijan kanssa samaa mieltä siitä, että Tableau toimi datan tarkastuksessa parhaiten, sillä se esitti oletusarvoisesti jokaisen sähköluokan palkkeina omilla akseleillaan. Myös Power BI:n esittämät pylväät samalla akselilla osoittivat datan olemassaolon, mutta erityisesti pienempiä pylväitä oli vaikea havaita. Qlik Sensen kohdalla molemmat koehenkilöt hieman turhautuivat, sillä heidän oli itse päätettävä käytettävä visulisaatio ja sen muodostaminen vaati monta valintaa. Kaikilla työkaluilla pystyttiin kuitenkin toteamaan, että tietokannassa oli olemassa testattavaa dataa.

Datan korkean tason tarkastelussa toinen koehenkilö koki Tableaun parhaaksi välineeksi ja toinen Power BI:n. Kuukausitason valinta molemmilla työkaluilla vaati hieman etsintää ja oli hieman makuasia, oliko kuukausitason määrittäminen helpompaa Tableaulla vai Power BI:lla. Näillä molemmilla työkaluilla molemmat koehenkilöt kuitenkin heti huomasivat, että päivämäärätiedoissa oli puutteita, sillä osa sähköstä ei kohdistunut millekään kuukaudelle. Qlik Sense sai jälleen molemmilta koehenkilöiltä nolla pistettä, sillä puuttuva päivämäärätieto jäi kokonaan huomaamatta ja visualisaation muodostus vaati paljon työtä. Koehenkilöt muodostivat tarkasteltavasta datasta taulukkomuotoisen esityksen jokaisella työkalulla, ja vertailemalla taulukkoa alkuperäiseen dataan he huomasivat epäyhtäläisyyksiä luvuissa ja pystyivät näin päättelemään, että datassa oli jotain vikaa.

Puuttuvan tunnin testauksessa kaikki olivat yksimielisiä siitä, että puuttuva tunti oli helpoin huomata Tableaun avulla. Lisäksi maaliskuun asettaminen suodattimeksi oli yksimielisesti helpointa Tableaun avulla, ja siksi sekä koehenkilöt että tutkija antoivat Tableaulle korkeimmat pisteet. Kaikki olivat myös yksimielisiä siitä, että Qlik Sense oli vaikeakäyttöisin ja puuttuvan tunnin toteamiselle oli vaikea keksiä ja luoda selkeä esitystapa. Lopulta kaikkien työkalujen avulla oli kuitenkin mahdollista huomata puuttuva tunti, mutta Tableaun avulla se oli selvästi nopeinta.

Koehenkilöt pystyivät suorittamaan testausprosessin ennalta määriteltyjen testitapausten pohjalta. Koehenkilöt huomasivat kaikki dataan tehdyt virheet, vaikkeivat välttämättä ymmärtäneet mistä virheet johtuivat. Tavoitteena oli kuitenkin ainoastaan varmistaa, että

ulkopuoliset henkilöt löytävät virheet raportointityökalujen avulla, ja tässä onnistuttiin. Virheiden löytymisen lisäksi tavoitteena oli selvittää, mikä raportointityökalu soveltuu parhaiten tämän kaltaiseen datan testaukseen loppukäyttäjän näkökulmasta. Tätä tarkastelua varten koehenkilöt pisteyttivät työkalut helppokäyttöisyyden, nopeuden, loogisuuden ja graafisen esitysvoiman perusteella.

Koehenkilöiden antamat pisteet ovat kokonaisuudessaan todella lähellä tutkijan antamia pisteitä. Tutkijan pisteytyksessä Tableau ja Power BI jakoivat korkeimmat pisteet, mutta molempien koehenkilöiden pisteytyksessä Tableau sai korkeimmat pisteet ja Power BI yhden tai kaksi pistettä vähemmän. Tutkija ja koehenkilöt olivat varsin yksimielisiä siitä, että Qlik Sense soveltui huonosti tämän tyyppiseen datan tarkasteluun ja se ei yhteydenmuodostusta lukuunottamatta saanutkaan yhtään pistettä keneltäkään pisteiden antajista. Taulukosta 6.2 voidaan nähdä, että kun lasketaan kaikki työkaluille annetut pisteet yhteen, saa Tableau 24, Power BI 20 ja Qlik Sense ainoastaan 2 pistettä. Näin ollen voidaan todeta, että tämän kaltaisessa datan tarkastelussa Tableau soveltuu parhaiten loppukäyttäjän työkaluksi. Power BI:n soveltuvuus on lähes yhtä hyvä kuin Tableaun, mutta Qlik Sensen soveltuvuus jää tässä tapauksessa kauaksi näistä kahdesta.

6.4 Testaustulosten arviointi

Testaussuunnitelman toteuttaminen kolmella eri raportointityökalulla sujui odotusten mukaisesti. Testaaminen oli mahdollista kaikkien työkalujen avulla, mutta työkalujen välillä oli myös selkeitä eroja, ja Tableau ja Power BI olivat huomattavasti helppokäyttöisempiä ja loogisempia kuin Qlik Sense. Testaus siis osoitti, että NoSQL-tietokannan datan oikeellisuuden testaaminen on mahdollista raportointityökalujen avulla ja raportointityökaluissa on selkeitä toiminnallisia eroja. Teoreettisen näkökulman pohjalta muodostettu testaussuunnitelma oli mahdollista suorittaa raportointityökalujen avulla ja testaussuunnitelman määrittämät tavoitteet saavutettiin löytämällä virheet datasta.

Raportointityökalujen yhdistäminen NoSQL-tietokantaan oli yksinkertaista, tuli vain määrittää oikea ajuri ja yhteyden tiedot. Oletuksena kuitenkin oli, että tietokannan kehittäjä oli osoittanut testattavan datan luomalla näkymän siihen. Jos NoSQL-tietokannan päälle tulee joka tapauksessa raportointityökalu, ei tämä ole turha vaihe. Mutta jos raportointityökalua ei käytetä jatkossa, eikä näkymätyyppisille rakenteille ole tarvetta, tulee näkymän luonnista yksi ylimääräinen työvaihe ennen kuin testaus voidaan aloittaa. Toki on huomattava, että tutkimuksessa käytettiin ainoastaan HBase-tietokantaa, muiden ei-relationaalisten tietokantojen kanssa mahdollisuudet yhdistää raportointityökalut tietokannan dataan voivat olla erilaiset ja lisäksi HBaseen yhdistämiselle on Drillin lisäksi muitakin vaihtoehtoja. Siitä, että näkymän luonti on tarpeellista, voidaan huomata, että vaikka HBase NoSQL-tietokannan käsittely on mahdollista raportointityökalujen avulla, on raportointityökalujen kannalta tehokkaampaa ja luontevampaa esittää data varsin perinteisessä rivi-sarakemuodossa.

Tutkimuksessa tietokannassa sijainneeseen testausaineistoon oli tehty tahallisia virheitä, jotka oli tarkoitus löytää testaamisen avulla. Koska oli kyse loppukäyttäjän suorittamasta testauksesta, oli riittävää, että virheet huomattiin korkealla tasolla. Tämä tarkoittaa sitä, että testaajan kannalta ei ollut merkitystä, mistä syystä virheet johtuivat, riitti vain että data voitiin todeta ei-oikeelliseksi korkealla tasolla. Loppukäyttäjän edustaman tilaaja-osapuolen testauksen kannalta on riittävää löytää ylipäätään virheitä, joista voidaan ilmoittaa toimittajalle, ja on toimittajan tehtävä etsiä syy virheille. Näin ollen tilaajaosapuolen loppukäyttäjän testausvälineeksi korkean tason tarkastelu toimii hyvin, sillä huomataan heti, onko datassa virheitä tai puutteita ja niistä voidaan ilmoittaa toimittajalle, jonka tehtävänä on etsiä syy virheelle ja korjata se.

Visualisaatiot nopeuttivat datan tarkastelua huomattavasti. Tämä huomattiin esimerkiksi puuttuvan tunnin tarkastelussa, jossa visualisaatiosta voitiin nähdä puuttuva tunti yhden vilkaisun avulla, kun taulukkomuodosta puutetta joutui etsimään selaamalla pitkää taulukkoa ylös ja alas. Myös datan olemassaolon tarkistus hoitui erittäin nopeasti ja kattavasti visualisaation avulla. Toisaalta jos olisi tarkasteltu dataa, jolle ei olisi tarjolla yhteisiä ja yksiselitteisiä dimensioita, olisi visualisaatioiden muodostaminen haasteellisempaa, ja silloin testausta kannattaisi jakaa datan perusteella eri osa-alueisiin.

Raportointityökalujen etuna on, että käytettäessä niitä testauksessa, on tulokset helppo dokumentoida, sillä raportin visualisaatio saadaan tallennettua sellaisenaan tai kuvakaappausten avulla. Testit ovat myös helposti toistettavissa, sillä jos muodostettu raportti tallennetaan, voidaan päivittää vaan dataa raportin pohjalla ja katsoa näyttääkö raportti edelleen samalta. Visualisaatioita ehdottavien raportointityökalujen etuna on myös se, että työkalut saattavat ehdottaa datan pohjalta myös sellaisia kuvaajia ja analyyseja, joita raportointitarpeiden määrittelijälle ei ole tullut edes mieleen. Näin voidaan saada lisäarvoa olemassa olevasta datasta.

Tässä tutkimuksessa tietokannan dataa verrattiin alkuperäiseen dataan muodostamalla molemmista datakokonaisuuksista omat taulukkomuotoiset summat ja vertaamalla näitä summia konkreettisesti. Jos loppukäyttäjä kuitenkin olisi erittäin kokenut ja valmis muokkaamaan testausmenetelmää mahdollisimman optimaaliseksi, niin hän voisi tuoda raportille kaksi eri datalähdettä, alkuperäisen ja tietokannan, ja verrata raportilla suoraan näitä aineistoja ja muodostaa niin sanottuja poikkeamaraportteja. Poikkeamaraportit esittäisivät vain kohdat, joissa on eroavaisuuksia, joten konkreettiselta lukujen vertailulta vältyttäisiin. Tämä säästää aikaa virheiden löytämisessä, mutta tarkempi tarkastelu vaatii tarkkuutta, jotta aineistot eivät mene sekaisin ja silti virheiden löytyminen voi olla rivitasoisen tarkastelun takana. Poikkeamaraportit eivät ole myöskään informaatioarvoltaan jatkossa aivan yhtä arvokkaita. Testauksen dokumentoinnissa on tärkeää ilmoittaa, millaisella aineistolla on testattu ja millaisia lukemia testaustilanteessa on ollut, ja poikkeamaraportista nämä tiedot eivät välttämättä ilmene tarpeeksi kattavasti.

7. PÄÄTELMÄT

Työn viimeisessä luvussa kootaan yhteen tutkimuksen keskeiset johtopäätökset ja tulokset. Johtopäätösten lisäksi arvioidaan tutkimusta ja tuloksia sekä annetaan ehdotus mahdollisista jatkotutkimusaiheista. Lisäksi pyritään asettaamaan tämän tutkimus konstektiin suhteessa muihin tutkimuksiin.

7.1 Keskeiset johtopäätökset

Tutkimuksen tavoitteena oli muodostaa käsitys raportointityökalujen hyödyntämismahdollisuuksista NoSQL-tietokannan datan testauksessa. Tutkimuksen tuloksena oli, että raportointityökaluja voidaan hyödyntää datan testaamisessa muodostamalla korkean tason summia ja kuvaajia, sillä yksittäiset datavirheet näkyvät korkean tason tarkastelussa. Datan testauksessa näkökulmana oli datan oikeellisuuden testaaminen, sillä loppukäyttäjälle on tärkeintä, että data on oikeellista ja luotettavaa. Dataa tallennetaan tietokantoihin usein liiketoimintatiedon hallinnan prosesseja varten, ja näiden prosessien lopputuloksena tuotetaan yleisesti raportteja, joita loppukäyttäjät voivat hyödyntää liiketoiminnan seuraamisessa ja päätöksenteossa. Tietokannan dataa siis hyödynnetään lopputuotteena syntyvissä raporteissa, mutta yhtä hyvin raportointityökaluja voitaisiin hyödyntää datan tarkasteluun myös liiketoimintatiedon hallinnan prosessin muissa vaiheissa.

Liiketoimintatiedon hallinnan prosessiin kuuluu datan tallentaminen tietokantarakenteisiin. Tietokantarakenteet ovat perinteisesti perustuneet relaatiomalliin, mutta datamäärän jatkuvan kasvun ja moninaisuuden myötä on osittain siirrytty myös skaalautuvuutta ja joustavuutta tukeviin ei-relationaalisiin NoSQL-tietokantarakenteisiin. NoSQL-tietokannat eivät pääasiassa tue strukturoitua kyselykieltä SQL:aa, joten datan tarkastelu vaatii eirtysiosaamista. NoSQL-tietokantojen datan käsittely on vaihtelevampaa yhden selkeän kyselykielen puuttuessa, ja datan tarkastelemiseksi tulisi usein oppia uusi kyselykieli tai -metodi.

Tietokannan datan hyödyntäjä eli loppukäyttäjä näkee datan usein ainoastaan valmiina raportteina. Liiketoimintatiedon hallinnan loppukäyttäjä ei tarvitse tuntea tietokantarakenteita, varsinkaan monimutkaisempia ja vaihtelevampia NoSQL-rakenteita, tarkastellakseen tietokannan dataa raporteilta. Loppukäyttäjä on hyvä testaaja datan oikeellisuudelle, sillä hän yleensä tietää, mitä datalta odotetaan. Loppukäyttäjä voi tarkastella datan oikeellisuutta raportointityökalujen avulla, jolloin hänen ei tarvitse tuntea tietokannan teknistä toteutusta tarkastellakseen sen dataa. Raportointityökaluja voidaan siis hyödyntää jo datan oikeellisuuden testaukseen ja voidaan helposti varmistua datan oikeellisuudesta jo testausvaiheessa.

Jotta testaus etenee hallitusti ja suunnitellusti, on testauksen pohjana hyvä käyttää testaussuunnitelmaa, jossa määritellään muun muassa testauksen rajaus, resurssit, tavoitteet ja testitapaukset. Tässä tutkimuksessa testaussuunnitelma muodostettiin teoriaosuuden pohjalta. Testaussuunnitelma tuki ja ohjasi testauksen toteuttamista ja antoi testausprosessille selkeän rakenteen. Testaussuunnitelman tärkeimmät kohdat olivat testausaineiston ja testitapausten määrittely. Selkeästi määritelty ja muodostettu testausaineisto mahdollisti testauksen ja auttoi tavoitteiden saavuttamisessa, ja lisäksi testaus oli helppo jakaa osiin testitapausten avulla. Testaussuunnitelman määrittämät tavoitteet saavutettiin löytämällä virheet datasta.

Tutkimuksen empirian tarkoituksena oli varmistua raportointityökalujen soveltuvuudesta NoSQL-tietokannan datan testaukseen. Empirian tavoitteena oli lisäksi selvittää, miten virheet tietokannan datassa näkyvät raportointityökaluilla ja miten eri raportointityökalut soveltuvat tämän tyyppiseen testaukseen. Tuloksena oli, että raportointityökalut sopivat testausvälineiksi, sillä virheet datassa ilmenivät korkean tason tarkastelussa, työkalut olivat loppukäyttäjälle helppokäyttöisiä ja raportointityökalujen graafiset esitystavat helpottivat datan tarkastelua. Lisäksi testaussuunnitelma ja siinä määritellyt testitapaukset ohjasivat testausta hyvin, joten ainakin tämän tutkimuksen kannalta testaussuunnitelman hyödyllisyys voitiin todeta. Lisäksi empirian perusteella voitiin tunnistaa, että Tableau ja Power BI toimivat Qlik Sensea paremmin tämän kaltaisessa käytössä.

Tässä tutkimuksessa käytettyjä raportointityökaluja on vertailtu muissa tutkimuksissa esimerkiksi raporttien luomisen ja liiketoimintaympäristöön valitsemisen näkökulmasta (esimerkiksi Gartner 2015; Xiang 2015), mutta puhtaasti loppukäyttäjän testausvälineinä aiempaa tutkimusta ei ole tehty. Tämä tutkimus tuo uuden näkökulman raportointityökalujen vertailuun vertailemalla työkalujen hyödynnettävyyttä strukturoidun NoSQL-datan tarkastelussa. Tämän tutkimuksen puitteissa Tableau soveltui parhaiten testauskäyttöön, kun arvioitiin helppokäyttöisyyttä, loogisuutta ja graafista esitysvoimaa. Myös Gartnerin selvityksessä (2015) asiakkaat arvostelivat Tableaun helppokäyttöisimmäksi raportointityökaluksi ja Tableaun raporttien luominen oli nopeinta riippumatta ratkaisun kompleksisuudesta, joten nämä havainnot tukevat toisiaan.

Tässä tutkimuksessa raportointityökaluja vertailtiin kokemattoman loppukäyttäjän näkökulmasta, eli helppokäyttöisyys nousi tärkeimmäksi attribuutiksi. Tutkimuksen pohjalta ei siis oteta kantaa, mikä työkaluista on kokonaisuudessaan paras, sillä tässä käsiteltiin ainoastaan pientä osa-aluetta ja lisäksi työkalut vastaavat muutenkin hieman eri tarkoituksiin. Lisäksi, jos koehenkilöillä olisi ollut asiantuntemusta esimerkiksi Qlik Sensesta, olisivat tulokset olleet erilaiset. Helppokäyttöisyyden näkökulmasta voidaan kuitenkin todeta, että Tableau ja myös Power BI ovat Qlik Sensea helppokäyttöisempiä.

7.2 Tutkimuksen ja tulosten arviointi

Hyvään tutkimukseen kuuluu olennaisena osana myös tutkijan oma arviointi, joka osoittaa tutkimuksen toteuttajan omaa kriittistä arviointia työn onnistuneisuutta ja uskottavuutta kohtaan (Olkkonen 1993, s. 111). Tämä tutkimus tuntuu onnistuneelta, sillä tutkimuskysymyksiin voitiin vastata, mutta tutkimuksen yleistetävyys on sidonnaista.

Tutkimuksen yleistettävyys ja luotettavuus voidaan jossain määrin kyseenalaistaa, sillä tutkimus suoritettiin ainoastaan yhden NoSQL-tietokannan perusteella ja valitut raportointityökalut myös ohjasivat tuloksia. Lisäksi tietokannassa ollut testattava data oli varsin perinteistä strukturoitua tuotanto- ja kulutusdataa, joten tuloksia ei voida yleistää kaiken datan kannalta. NoSQL-tietokannan data voi yhtä hyvin olla strukturoimatonta tai semi-strukturoitua dataa, ja näiden käyttäytymiseen raportointityökaluissa ei voida ottaa kantaa tämän tutkimuksen perusteella. Lisäksi testitapausten suunnittelu erilaisen datan kanssa voi olla haasteellisempaa kuin tässä tutkimuksessa.

Tämä tapaustutkimus pohjautuu tiettyyn tapaukseen, joka tässä oli NoSQL-tietokannan testaaminen loppukäyttäjän näkökulmasta. Tutkimustuloksissa ei siis suoranaisesti poissuljeta mahdollisuutta, että raportointityökalujen avulla testattava tietokanta voisi olla myös relaatiotietokanta. Tässä tutkimuksessa oli kuitenkin tapaustutkimuksen puitteissa kyseessä ainoastaan NoSQL-tietokanta, ja tutkimuksen tausta-ajatuksena oli, että NoSQL-tietokannan datan tarkasteluun tulisi tarjota liiketoimintaorientoituneelle loppukäyttäjälle sopiva työkalu, kun strukturoitua kyselykieltä ei ole saatavilla. Samat menetelmät kuitenkin oletettavasti toimisivat myös relaatiotietokantojen yhteydessä, jos testattava data olisi sama. Tämän tutkimuksen lähestymistapa on siis suunniteltu NoSQL-tietokannoille, mutta tutkimuksen perusteella ei voida todeta, etteikö sama lähetymistapa olisi soveltuva myös relaatiotietokantojen yhteydessä.

Datan tallennuspaikkana NoSQL-tietokanta on relaatiotietokantaa mukautuvampi, jolloin voidaan toteuttaa paremmin skaalautuvia, joustavia ja kustannustehokkaita ratkaisuja. Loppukäyttäjälle tulos on samannäköinen, käytettiin sitten relaatiotietokantaa tai NoSQL-tietokantaa. Raportointityökalujen hyödyntäminen mahdollistaa sen, että loppukäyttäjän ei tarvitse tietää datan tallennusrakenteista yhtään mitään. Kehitysnäkökulmasta NoSQL tuo kuitenkin lisäarvoa laajuutensa ja mukautuvuutensa vuoksi, joten datan tallennus NoSQL-tietokantaan on monissa yhteyksissä perusteltua. Vaikka loppukäyttäjälle datan tallennuspaikalla ei olekaan väliä, on relaatio- ja NoSQL-tietokantojen kehittäminen kuitenkin erilainen operaatio, joka vaati kehittäjältä erityisosaamista.

Tässä tutkimuksessa keskityttiin ainoastaan datan oikeellisuuden testaukseen, sillä se on loppukäyttäjän näkökulmasta tärkein osa toimivuutta ja käyttöönottoa. Raportointityökalujen avulla käsitellään tietokannan dataa, joten raportointityökaluilla testaus myös keskittyy dataan. Mutta myös tietokannan suorituskyky, skaalautuvuus ja

viansieto ovat tärkeitä testauskohtia, mutta näiden tekijöiden testaus ei ole liiketoimintaorientoituneen loppukäyttäjän vastuulla, joten ne on siksi rajattu pois.

Tutkimusta ja testausta suoritti tutkija lisäksi kaksi ulkopuolista koehenkilöä, jotka saivat tutkijan kanssa yhtenevät tulokset. Näin ollen voidaan nähdä, että tutkijan tulokset eivät olleet ainoastaan subjektiivisia, ja samat tulosket syntyivät myös ulkopuolisten tarkastelijoiden suorittamassa tutkimusosiossa. Tämä lisää tutkimuksen luotettavuutta. Tutkimus on melko helposti toistettavissa, sillä tutkimuksessa edettiin teoreettisen tarkastelun pohjalta luodun testaussuunnitelman mukaisesti. Testaussuunnitelmaa ja testitapauksia noudattaen empiirinen tutkimus olisi toistettavissa vastaavanlaisen datan tapauksessa.

Raportointityökalut asetettiin järjestykseen yksinkertaisen pisteytyksen avulla, jossa kaikki testitapaukset olivat yhtä merkittäviä kokonaispisteiden kannalta. Jos kuitenkin keskitytään puhtaasti datan testaamiseen, voisi olla hyödyllistä painottaa pisteitä niin, että datan testaus olisi merkittävämpi kuin esimerkiksi yhteydenmuodostus. Myös helppokäyttöisyyttä, loogisuutta ja graaafista esitysvoimaa voitaisiin pisteyttää ja tarkastella erikseen. Kokonaispisteet eivät siis kerro absoluuttisesti työkalujen paremmuudesta, vaan ovat ainoastaan kolmen henkilön mielipiteet helppokäytöisyyden, loogisuuden ja graafisen esitysvoiman suhteen. Kokonaispisteet ovat kuitenkin selvästi suuntaa antavia siitä, että Tableaun ja Power BI:n soveltuvuus datan testaukseen on parempi kuin Qlik Sensen, kun loppukäyttäjällä ei ole aiempaa kokemusta työkaluista.

7.3 Mahdollisia jatkotutkimuksen kohteita

Aiheen tarkastelu tässä tutkimuksessa on pintapuolinen ja suuntaa-antava, joten aiheen syvällisempi ja laajempi tarkastelu olisi tarpeellista. Tämän tutkimuksen perusteella raportointityökalut toimivat NoSQL-tietokannan datan testauksessa, mutta yleistettävyyttä voitaisiin parantaa tutkimalla aihetta laajemmin esimerkiksi eri tietokannan, eri raportointityökalujen ja erityyppisen testausaineiston perusteella, sekä ottamalla huomioon myös semi-strukturoitu ja strukturoimaton data. Tutkimuksessa käytetty testausaineisto ei ollut niin massiivinen kuin NoSQL-tietokannoissa yleensä ja lisäksi data oli strukturoitua. Olisi siis myös hyödyllistä suoritta vastaava tutkimus huomattavasti suuremmalla ja monimuotoisemmalla datajoukolla ja seurata samalla myös raportointityökalujen suorituskykyä.

Tämä tutkimus ei ota kantaa kaikkiin eri tapoihin, joilla NoSQL-tietokantojen data voidaan yhdistää raportointityökaluihin, joten näiden mahdollisuuksien tutkiminen on yksi mahdollinen tutkimuskohde. Tämän pohjalta voitaisiin myös muodostaa laajempi näkemys raportointityökalujen sopivuudesta NoSQL-tietokantojen testausvälineiksi. Lisäksi saman testausprosessin voisi toteuttaa relaatiotietokannan datalle, jolloin voitaisiin varmistua, toimiiko sama lähestymistapa NoSQL-tietokannan lisäksi myös relaatiotietokannan tapauksessa.

NoSQL-tietokantojen datan testausta ei ole juuri käsitelty kirjallisuudessa. Big data -ilmiön ja moninkertaistuvan datamäärän myötä NoSQL-tietokantojen yleisyys tulee kuitenkin lisääntymään ja on tärkeää löytää mahdollisia testauskeinoja NoSQL-tietokantojen ja niiden datan oikeellisuuden varmistamiseksi. Laajempaa tarkastelua voitaisiin siis kohdistaa NoSQL-tietokannan ja sen datan testaukseen yleensäkin.

LÄHTEET

Alavi, M. & Leidner, D. 2001. Review: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues. MIS Quarterly, Vol. 25, No. 1, ss. 107-136.

Altrafi, O., Ismail, M. & Mohamed, M. 2014. Relational vs. NoSQL Databases: A Survey. International Journal of Computer and Information Technology. Vol. 3, No. 3, ss. 598-601.

Amazon. 2015. Amazon QuickSight. https://aws.amazon.com/quicksight/. Viitattu 12.12.2015.

Andreou, A. & Sofokleous, A. 2008. Automatic, evolutionary test data generation for dynamic software testing. The Journal of Systems and Software. Vol. 81, ss. 1883-1898.

Anuradha, J. & Ishwarappa. 2015. A Brief Introduction on Big Data 5Vs Characteristics and Hadoop Technology. Procedia Computer Science. Vol. 48, No. C, ss. 319-324.

Apache Drill. 2015a. Apache Drill. https://drill.apache.org/. Viitattu 20.1.2016.

Apache Drill. 2015b. Tableau Examples. https://drill.apache.org/docs/tableau-examples/. Viitattu 20.1.2016.

Barbierato, E., Gribaudo, M. & Iacono, M. 2014. Performance evaluation of NoSQL big-data applications using multi-formalism models. Future Generation Computer Systems. Vol. 37, ss. 345-353.

Blanco, P. 2013. NoSQL databases in cross-platform development. Master of Science Thesis. Tampere, Tampere University of Technology, Department of Information Technology, 50 s.

Chen, H., Chinag, R. & Storey, V. 2012. Business Intelligence and Analytics: From Big Data to Big Impact. MIS Querterly, Vol. 36, No. 4, ss. 1165-1188.

Chen, T.Y., Poon, P-L., Tang, S-F. & Tse, T.H. 2004. On the identification of categories and choices for specification-based test case generation. Information and Software Technology. Vol. 46, ss. 887-898.

Choo, C. 1998. The Knowing Organization - How Organizations Use Information to Construct Meaning, Create Knowledge, and Make Decisions. New York, New York, Oxford University Press Inc, 298 s.

Collins, D. 2008. Data quality: Types of error in database data. http://dataquality.origma.co.uk/uploads/fckeditor/file/200805/20080520155902.pdf. Viitattu 28.1.2016.

Creswell, J.W. 2003. Research Design, Qualitative, Quantitative, and Mixed Methods Approaches. 2. painos, Thousand Oaks, Kalifornia, SAGE Publications, 26 s.

Currim, S., Ram, S., Durcikova, A. & Currim, F. 2014. Using a knowledge learning framework to predict errors in database design. Information Systems. Vol. 40, ss. 11-31.

Cuzzocrea, A., Song, I. & Davis, K. 2011. Analytics over Large-Scale Multidimensional Data: The Big Data Revolution! ACM Fourteenth International Workshop on Data Warehousing and OLAP (DOLAP'11), 28.10.2011, Glasgow, Skotlanti, ss. 101-103.

Do, T., Khoo, S-C., Ming Fong, A., Pears, R. & Quan, T. 2015. Goal-oriented dynamic test generation. Information and Software Technology. Vol. 66, ss. 40-57.

Ding, Z., Zhang, K. & Hu, J. 2008. A rigorous approach towards test case generation. Information Sciences. Vol. 178, No. 21, ss. 4057-4079.

Energiateollisuus ry. 26.3.2015. Sähkön tuntidata. http://energia.fi/tilastot-ja-julkaisut/sahkotilastot/sahkon-tuntidata. Viitattu 30.8.2015.

Few, S. 2006. Information Dasboard Design: The Effective Visual Communication of Data. 1. painos, Sebastopol, Kalifornia, Yhdysvallat, O'Reilly Media Inc, 211 s.

Few, S. 2009. Now You See It – Simple Visualization Techniques for Quantitative Analysis. 1. painos, Oakland, Kalifornia, Yhdysvallat, Analytics Press, 327 s.

Few, S. 2012. Show Me the Numbers: Designing Tables and Graphs to Enlighten. 2. painos, Burlingame, Kalifornia, Yhdysvallat, Analytics Press, 351 s.

Floratou, A., DeWitt, D., Patel, J., Teletia, N. & Zhang, D. 2012. Proceedings of the VLDB Endowment. Vol. 5, No. 12, ss. 1712-1723.

Gartner. 2015. Critical Capabilities for Business Intelligence and Analytics Platforms. http://www.gartner.com/technology/reprints.do?id=1-2F5GQXN&ct=150513&st=sb. Viitattu 6.8.2015.

Gartner. 2016. Magic Quadrant for Business Intelligence and Analytics Platforms. https://www.gartner.com/doc/reprints?id=1-2XXET8P&ct=160204. Viitattu 4.2.2016.

Gilad, B. & Gilad, T. 1986. Business intelligence – the quiet revolution. Sloan Management Review. Vol. 27, No. 4, ss. 53-61.

Gummesson, E. 2000. Qualitative Methods in Management Research. 2. painos, Thousand Oaks, Kalifornia, Yhdysvallat, Sage Publications, 250 s.

Hakala, J. 2006. Informaatiohyöky – Tiedon ja osaamisen hallinta työelämässä. Helsinki, Gaudeamus Kirja / University Press Finland Ltd, 264 s.

Haikala, I. & Märijärvi, J. 2006. Ohjelmistotuotanto. 11. painos, Helsinki, Talentum Media Oy, 440 s.

Hambling, B., Morgan, P., Samaroo, A., Thompson, G. & Williams, P. 2010. Software Testing – An ISTQB-ISEB Foundation Guide. 2. painos, Swindon, Iso-Britannia, British Informatics Society Limited, 223 s.

Hannula, M., Leinonen, M., Lönnqvist, A., Mettänen, P., Miettinen, A., Okkonen, J. & Pirttimäki, V. 2002. Nykyaikaisen organisaation suorituskyvyn mittaus. Tampere, Tampereen teknillinen korkeakoulu, Tuotantotalouden osaston tutkimusraportti numero 1/2002. 190 s.

Hirsjärvi, S., Remes, P. & Sajavaara, P. 2007. Tutki ja kirjoita. 13. painos, Helsinki, Kustannusosakeyhtiö Tammi, 448 s.

Hortonworks. 2014. Hortonworks Data Platform - Buyer's Guide. http://hortonworks.com/wp-content/uploads/2014/05/Product-Guide-HDP-2.1-v1.0.pdf. Viitattu 18.1.2016.

Hovi, A. 1997. Data Warehousing – Tietovarastotekniikka. Espoo, Suomen ATK-kustannus Oy, 124 s.

Hovi, A. 2008. SQL-opas. 5. painos, Jyväskylä, Docendo Finland Oy, 280 s.

Hovi, A., Hervonen, H. & Koistinen, H. 2009. Tietovarastot ja Business Intelligence. Jyväskylä, WSOYpro/Docendo-tuotteet, 196 s.

Hovi, A., Huotari, J. & Lahdenmäki, T. 2005. Tietokantojen suunnittelu & indeksointi. 1. painos, Jyväskylä, Docendo Finland Oy, 365 s.

Hurwitz, J., Nugent, A., Halper, F. & Kaufman, M. 2013. Big Data for Dummies. Hoboken, New Jersey, Yhdysvallat, John Wiley & Sons, Inc., 336 s.

Kaario, K. & Peltola, T. 2008. Tedon hallinta – Avain tietotyön tuottavuuteen. 1. painos, Jyväskylä, WSOYpro/Docendo-tuotteet, 164 s.

Kamel, I. 2009. A schema for protecting the integrity of databases. Copmuters & Security. Vol. 28, No. 7, ss. 698-709.

Karjalainen, L. & Karjalainen, J. 2009. Tilastojen graafinen esittäminen. 1. painos, Ristiina, Pii-Kirjat Ky, 183 s.

Koskinen, A., Pirttimäki, V. & Hannula, M. 2005. Liiketoimintatiedon hallinnta suomalaisissa suuryrityksissä vuosina 2002-2005. Tampere, e-Business Research Center, Tampereen teknillinen yliopisto ja Tampereen yliopisto, Research. Reports 21. 37 s.

Krishnan, K. 2013. Data Warehousing in the Age of Big Data. 1. painos, Waltham, Massachusetts, Yhdysvallat, Morgan Kaufmann, 346 s.

Kubina, M., Koman, G. & Kubinova, I. 2015. Possibility of Improving Efficiency within Business Intelligence Systems in Companies. Procedia Economics and Finance. Vol. 26, ss. 300-305.

Leavitt, N. 2010. Will NoSQL Databases live up to their promise? Computer. Vol. 34, No. 2, ss. 12-14.

Liebowitz, J. 2006. Strategic Intelligence – Business Intelligence, Competitive Intelligence, and Knowledge Management. Boca Raton, Florida, Yhdysvallat, Auerbach Publications, 223 s.

Lo, F. 2015. Big Data Technology: What is Hadoop? What is MapReduce? What is NoSQL? https://datajobs.com/what-is-hadoop-and-nosql. Viitattu 14.9.2015.

Loshin, D. 2013. Big Data Analytics: From Strategic Planning to Enterprise Integration with Tools, Techniques, NoSQL and Graph. 1. painos, Waltham, Massachusetts, Morgan Kaufmann, 142 s.

Marcozzi, M., Vanhoof, W. & Hainaut, J. 2015. Relational symbolic execution of SQL code for unit testing of database programs. Science of Computer Programming. Vol. 105, ss. 44-72.

McFadden, F., Hoffer, J. & Prescott, M. 1999. Modern Database Management. 5. painos, Yhdysvallat, Addison-Wesley Educational Publishers Inc, 622 s.

Microsoft. 2015. Power BI. https://powerbi.microsoft.com/en-us/. Viitattu 12.12.2015.

Moniruzzaman, A. & Hossain, S. 2013. NoSQL Database: New Era of Databases for Big data Analytics – Classification, Characteristics and Comparison. International Journal of Database Theory and Application. Vol. 6, No. 4, ss. 1-14.

Myers, G., Sandler, C. & Badgett, T. 2011. The art of software testing. 3. painos, Hoboken, New Jersey, Yhdysvallat, John Wiley & Sons, Inc., 240 s.

Olkkonen, T. 1993. Johdatus teollisuustalouden tutkimustyöhön. Tampere, Teknillinen korkeakoulu, teollisuustalous ja työpsykologia, Raportti 152. 143 s.

Padhy, R., Patra, M. & Satapathy, S. 2011. RDBMS to NoSQL: Reviewing Some Next-Generation Non-Relational Database's. International Journal of Advanced Sciences and Technologies. Vol. 11, No. 1, ss. 15-30.

Pirttimäki, V. 2007. Business Intelligence as a Managerial Tool in Large Finnish Companies. Thesis for the degree of Doctor of Technology. Tampere, Tampere University of Technology, Department of Industrial Engineering and Management. 129 s.

Planet Cassandra. 2015. NoSQL Databases Defined and Explained. http://www.planetcassandra.org/what-is-nosql/. Viitattu 9.1.2016.

Pokorny, J. 2013. NoSQL databases: a step to database scalability in web environment. International Journal of Web Information Systems. Vol. 9, No. 1, ss. 69-82.

Qlik. 2016. Qlik Sense – Mitä oivelluksia dataan kätkeytyy? http://global.qlik.com/fi. Viitattu 2.2.2016.

Rapps, S. & Weyuker, E. 1985. Selecting Software Test Data Using Data Flow Information. IEEE Transactions on Software Engineering. Vol. SE-11, No. 4, ss. 367-375.

Redmond, E. & Wilson, J. 2012. Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement. 1. painos, Dallas, Texas, Yhdysvallat, The Pragmatic Bookshelf, 352 s.

de la Riva, C., Suarez-Cabal, M. & Tuya, J. 2010. Constraint-based Test Database Generation for SQL Queries. ACM Fifth International Workshop on Automation of Software Test (AST'10), 3.-4.5.2010, Kapkaupunki, Etelä-Afrikka, ss. 67-74.

Rockstad, E. & Briand, L. 2016. Cost-effective strategies for the regression testing of database applications: Case study and lessons learned. Journal of Systems and Software. Vol. 113, ss. 257-274.

Salo, I. 2013. Big data – Tiedon vallankumous. 1. painos, Jyväskylä, Docendo Finland Oy, 147 s.

Souza, E., Falbo, R. & Vijaykumar, N. 2015. Knowledge management initiatives in software testing: A mapping study. Information and Software Technology. Vol. 57, ss. 378-391.

Strauch, C. 2011. NoSQL Databases. Stuttgart Media University, 149 s. www.christof-strauch.de/nosqldbs.pdf. Viitattu 9.1.2016.

Ståhle, P. & Grönroos, M. 1999. Knowledge Management – tietopääoma yrityksen kilpailutekijänä. 2. painos, Porvoo, WSOY, 218 s.

Sydänmaanlakka, P. 2007. Älykäs organisaatio. 8. painos, Helsinki, Talentum Media Oy, 299 s.

Tableau Software. 2016. Answer questions as fast as you can think of them. http://get.tableau.com/trial/tableau-software.html. Viitattu 2.2.2016.

Tauro, C., Aravindh, S. & Shreeharsha, A.B. 2012. Comparative Study of the New Generation, Agile, Scalable, High Performance NoSQL Databases. International Journal of Computer Applications. Vol. 48, No. 20, ss. 1-4.

Thierauf, R. 2001. Effective Business Intelligence Systems. Westport, Connecticut, Yhdysvallat, Quorum Books, 330 s.

Turban, E., Sharda, R., Aronson, J. & King, D. 2008. Business Intelligence: A Managerial Approach. 1. painos, New Jersey, Yhdysvallat, Pearson Education Inc., 233 s.

Xiang, L. 2015. The Comparison of Qlik and Tableau: A Theoretical Approach Combined with Practical Experiences. Master's thesis. Hasselt, Universiteit Hasselt, Faculty of Business Economics, 53 s.

Xiang, P., Hou, R. & Zhiming, Z. 2010. Cache and Consistency in NOSQL. 3rd IEEE International Conference on Computer Science and Information Technology, Peking, Kiina, 9.7-11.7.2010, IEEE Conference Publications, 4 s.

Yin, R. 2003. Case Study Research – Design and Methods. 3. painos, Thousand Oaks, Kalifornia, SAGE Publications, 181 s.

LIITE A: OTE NÄKYMÄSTÄ SÄHKÖDATAAN

ID	Vuosi	Kuukausi	Kk_nro	Pvm	Tunti	Viikko	Tuotanto	Vesivoima	Tuuliv oima	Ydinvoima	Yhteistuotanto	Kaukolämpövoima	Teollisusvoima	Erillinen lämpövoima	Tuonti	Vienti	Kulutus	Nettotuonti
2014010101	2014	Tammikuu	1	1.1.2014	1	1	7887	1650	141	2769	3090	1901	1189	236	2205	-367	9725	1838
2014010102	2014	Tammikuu	1	1.1.2014	2	1	7762	1679	134	2769	2910	1737	1172	270	2135	-609	9288	1526
2014010103	2014	Tammikuu	1	1.1.2014	3	1	7606	1591	123	2769	2857	1688	1169	266	2028	-677	8957	1351
2014010104	2014	Tammikuu	1	1.1.2014	4	1	7544	1546	115	2767	2845	1683	1162	271	1908	-701	8752	1208
2014010105	2014	Tammikuu	1	1.1.2014	5	1	7513	1517	102	2769	2852	1684	1168	273	1883	-707	8689	1176
2014010106	2014	Tammikuu	1	1.1.2014	6	1	7544	1532	91	2768	2883	1708	1175	270	1911	-707	8748	1204
2014010107	2014	Tammikuu	1	1.1.2014	7	1	7609	1588	83	2769	2901	1727	1174	268	2109	-768	8950	1341
2014010108	2014	Tammikuu	1	1.1.2014	8	1	7683	1635	70	2769	2954	1777	1177	255	2071	-768	8985	1302
2014010109	2014	Tammikuu	1	1.1.2014	9	1	7855	1710	74	2769	3054	1871	1183	248	2127	-798	9185	1329
2014010110	2014	Tammikuu	1	1.1.2014	10	1	7977	1764	67	2769	3133	1955	1178	245	2066	-794	9249	1272
2014010111	2014	Tammikuu	1	1.1.2014	11	1	8083	1835	63	2770	3178	1992	1186	237	2051	-839	9295	1212
2014010112	2014	Tammikuu	1	1.1.2014	12	1	8111	1863	63	2769	3185	1996	1188	232	2236	-877	9470	1359
2014010113	2014	Tammikuu	1	1.1.2014	13	1	8181	1918	59	2768	3203	2017	1186	233	2261	-872	9570	1389
2014010114	2014	Tammikuu	1	1.1.2014	14	1	8226	1936	51	2769	3242	2052	1190	229	2267	-871	9622	1396
2014010115	2014	Tammikuu	1	1.1.2014	15	1	8229	1942	48	2769	3243	2059	1184	227	2389	-884	9734	1506
2014010116	2014	Tammikuu	1	1.1.2014	16	1	8336	2025	48	2769	3250	2071	1179	244	2721	-900	10156	1821
2014010117	2014	Tammikuu	1	1.1.2014	17	1	8577	2149	49	2768	3272	2090	1182	340	2731	-904	10404	1827
2014010118	2014	Tammikuu	1	1.1.2014	18	1	8585	2092	49	2769	3299	2116	1183	376	2724	-877	10431	1846
2014010119	2014	Tammikuu	1	1.1.2014	19	1	8611	2118	49	2769	3325	2148	1177	350	2736	-863	10483	1872
2014010120	2014	Tammikuu	1	1.1.2014	20	1	8585	2091	45	2768	3322	2138	1184	359	2760	-860	10485	1900
2014010121	2014	Tammikuu	1	1.1.2014	21	1	8453	1968	46	2768	3328	2143	1185	342	2694	-884	10264	1811
2014010122	2014	Tammikuu	1	1.1.2014	22	1	8328	1917	48	2768	3335	2151	1184	259	2465	-903	9890	1562
2014010123	2014	Tammikuu	1	1.1.2014	23	1	8266	1898	48	2769	3307	2130	1177	244	2777	-890	10152	1887
2014010124	2014	Tammikuu	1	1.1.2014	24	1	8166	1818	46	2769	3296	2112	1184	237	2706	-896	9975	1810

LIITE B: ALKUPERÄISEN AINEISTON TUOTANTOLAJIEN SUMMAT KUUKAUSITASOLLA

	Erillinen lämpövoima	Kaukolämpövoima	Kulutus	Nettotuonti	Teollisuusvoima	Tuonti
tammikuu	736 352,946934687	1 987 452,68072413	8 920 607,78659287	1 660 141,7	971 263,836037658	2 115 404,992
helmikuu	362 338,480118511	1 574 119,45209532	7 373 299,08742929	1 415 920,67	855 496,821201198	1 855 729,987
maaliskuu	214 654,294917087	1 469 443,31894729	7 624 441,1841442	1 604 513,913	880 424,144549217	1 987 915,209
huhtikuu	290 982,165120743	1 108 378,46764931	6 773 654,66635887	1 443 314,501	824 487,257646624	1 808 929,723
toukokuu	495 845,185816899	837 629,640617862	6 393 749,92330673	1 206 395,23	729 446,311057228	1 502 805,532
kesäkuu	271 243,78483926	488 760,870556871	5 726 886,87262844	1 244 287,482	603 152,66038743	1 419 084,509
heinäkuu	659 929,748932314	249 806,852144053	5 846 459,05508653	1 627 086,475	627 878,290450515	1 795 514,198
elokuu	836 753,181380767	370 472,596015687	6 034 303,60113475	1 672 310,182	622 344,142708467	1 900 790,49
syyskuu	832 364,313083504	667 642,850024001	6 154 451,38632036	1 423 336,325	641 435,551570095	1 663 972,014
lokakuu	800 914,452347209	1 122 379,19376306	7 200 070,71721134	1 368 621,503	800 307,267635668	1 632 075,461
marraskuu	454 927,019847483	1 340 412,89988876	7 409 843,70055705	1 576 672,859	837 586,798680695	1 857 282,992
joulukuu	363 027,111036295	1 636 799,94564914	7 958 570,86243847	1 723 591,134	890 319,722607498	2 082 089,276

	Tuotanto	Tuulivoima	Vesivoima	Vienti	Ydinvoima	Yhteistuotanto
tammikuu	7 260 466,08659288	90 548,763610613	1 417 242,41321261	-455 263,292	2 057 605,44607318	2 958 716,51676178
helmikuu	5 957 378,41742929	80 418,756392900	1 223 826,237177	-439 809,317	1 861 178,67044437	2 429 616,27329652
maaliskuu	6 019 927,2711442	106 614,135527915	1 300 644,13900955	-383 401,296	2 048 147,23819315	2 349 867,46349651
huhtikuu	5 330 340,16535887	93 373,893846089	1 034 760,24336371	-365 615,222	1 978 358,1377324	1 932 865,72529593
toukokuu	5 187 354,69330674	77 518,583240149	1 392 760,15459516	-296 410,3019	1 654 154,81797945	1 567 075,95167509
kesäkuu	4 482 599,39062845	52 182,747077036	1 281 929,32344482	-174 797,027	1 785 330,00432303	1 091 913,5309443
heinäkuu	4 219 372,58008652	37 244,530069711	807 123,689830772	-168 427,723	1 837 389,46865916	877 685,142594569
elokuu	4 361 993,41913474	72 180,729846604	824 759,842710642	-228 480,308	1 635 482,92647258	992 816,738724153
syyskuu	4 731 115,06132035	99 079,586856690	787 081,316545521	-240 635,689	1 703 511,44324054	1 309 078,4015941
lokakuu	5 831 449,21421133	133 618,398833339	931 234,289979351	-263 453,958	2 042 995,6116527	1 922 686,46139873
marraskuu	5 833 170,84155704	93 040,374956618	1 116 435,05342359	-280 610,133	1 990 768,6947599	2 177 999,69856945
joulukuu	6 234 979,72843846	171 416,494931061	1 122 368,31841987	-358 498,142	2 051 048,13579461	2 527 119,66825663