Integroitu malli holistisen näkemyksen muodostamiseksi teollisen internetin kontekstissa yrityksen tietovarastoa ja suosittelujärjestelmää hyödyntäen

Janne Hyvönen

Tampereen yliopisto
Informaatiotieteiden yksikkö

Tietojenkäsittelytieteiden tutkinto-ohjelma
Pro gradu -tutkielma
Ohjaaja: Marko Junkkari, Kati Iltanen
Lokakuu 2016

Tampereen yliopisto
Informaatiotieteiden yksikkö
Tietojenkäsittelytieteiden tutkinto-ohjelma
Tekijän Nimi: Janne Hyvönen
Pro gradu -tutkielma, 41 sivua
Lokakuu 2016

Tässä opinnäytetyössä tutkitaan kirjallisuustutkimuksen keinoin, miten teollinen internet vaikuttaa yrityksen tietovaraston vaatimuksiin ja toteutustapoihin ja miten tietovarasto voisi tukea holistisen näkemyksen muodostumista yrityksessä, ettei yritys menettäisi liiketoimintamahdollisuuksia. Asiaa on tutkittu erityisesti valmistavan teollisuuden näkökulmasta.

Opinnäytetyön tuloksena esitetään integroitu malli, joka tukee holistisen näkemyksen muodostumista yrityksessä. Mallin perustana on kirjallisuudessa esitelty kerroksellinen arkkitehtuuri, joka mahdollistaa ison datan kustannustehokkaasti skaalautuvan tallentamisen ja ison datan analysoimisen. Tietovaraston arkkitehtuuri on yhdistetty DMAIC-iteraatiota (define, measure, analyze, improve, control) noudattavaan datan hyödyntämisprosessiin, joka tukee yrityksen toiminnan kehittämistä jatkuvan parantamisen periaatetta noudattaen. Esitetyssä integroidussa mallissa lisäksi ehdotetaan sisältöperusteisen suosittelujärjestelmän käyttämistä holistisen näkemyksen edistämiseksi yrityksessä ehdottamalla käyttäjille tietovarastoon tallennettuja aineistoja.

Avainsanat ja -sanonnat: teollinen internet, yrityksen tietovarasto, sisältöperusteinen suosittelujärjestelmä, holistinen näkemys, iso data, jatkuva kehitys, DMAIC, integroitu malli

Sisällys

1. Johdanto

Nykyaikaisessa globaalissa ja kilpaillussa toimintaympäristössä teollisten yritysten on oleellista kyetä tunnistamaan muutoksia toimintaympäristössä ja kehittämään yrityksen omaa toimintaa pystyäkseen vastaamaan muutosten asettamiin haasteisiin. Valmistavassa teollisuudessa eräs selvä käynnissä oleva muutos on informaatioteknologian tuleminen entistä tiukemmin integroiduksi osaksi operatiivista toimintaa sekä sen kehittämistä ja tuotteita. Usein tästä muutoksesta käytetään termiä teollisuus 4.0 (Industrie 4.0, mm. Posada ja muut [2015]). Teollisuus 4.0 sisältää käsitteen teollinen internet, jolla tarkoitetaan laitteiden ja tuotteiden muuttumista "älykkäämmiksi" sekä niiden kykyä kommunikoida keskenään ja tallentaa dataa.

Toimintaympäristön muuttuessa myös yritysten tietovarastojen (data warehouses) täytyy muuttua. Tuotteet ja laitteet muuttuvat älykkäämmiksi (esimerkiksi Bruner [2013] ja Ivanov ja muut [2014]) teollisen internetin myötä, jolloin datan määrä, nopeus, jolla uutta dataa tuotetaan, sekä erilaisten dataformaattien määrä kasvavat ja voidaan puhua isosta datasta (big data) (mm. Laney [2001], Wu ja muut [2014]). Nykyaikaiset yritysten tietovarastot perustuvat pääasiassa relaatiotietokantoihin, jotka eivät pysty skaalautumaan riittävän kustannustehokkaasti ison datan tallentamiseen ja analysoimiseen (mm. Kimball ja Ross [2013], Krishnan [2013]). Datan määrän kasvaessa ja formaattien lisääntyessä ja muuttuessa monimutkaisemmiksi myös datan analysoimiseen tarvittavat menetelmät muuttuvat ja tietovaraston on pystyttävä tukemaan tehokkaasti näitä uusia menetelmiä.

Analyysimenetelmien kehittymisen lisäksi tietovaraston on kyettävä tukemaan holistista, eli kokonaisvaltaista, päätöksentekoa ja muuta operatiivista toimintaa yrityksessä [Zink 1998] sekä vastattava yrityksen organisaation informatiivisiin tietotarpeisiin, jotta organisaatio kykenee hyödyntämään liiketoimintamahdollisuuksia. Informatiivisilla tietotarpeilla tarkoitetaan esimerkiksi toimintaohjeiden ja tuotetiedon kaltaisen datan tarvitsemista. Jos relevantti informaatio ei välity oikea-aikaisesti oikeille henkilöille, voi yritys menettää liiketoimintamahdollisuuksiaan. Informatiivisiin tietotarpeisiin vastaaminen on nykyaikaisessa globaalissa ja verkostoituneessa toimintaympäristössä entistä haastavampaa ja muuttuu vielä haastavammaksi, kun neljäs teollinen vallankumous etenee ja yritysten operatiivinen toiminta muuttuu entistä virtuaalisemmaksi [Nagabushanan 2006].

Tässä opinnäytetyössä käsitellään yrityksen tietovarastoja (mm. Kimball [1996]), teollisen internetin niihin kohdistamia muutostarpeita, arkkitehtuurisia ja teknologisia ratkaisuja sekä prosesseja, joilla tietovarastot voisivat vastata näihin muutostarpeisiin. Lisäksi opinnäytetyössä käsitellään teknologioita joiden avulla yrityksen tietovarasto voisi tukea entistä paremmin informatiivisia tietotarpeita edistäen holistista näkemystä

yrityksessä. Kirjallisuudessa on käsitelty laajasti datan analysointimenetelmiä teollisuudessa ja ison datan aiheuttamia muutostarpeita tietovarastoille, mutta erityisesti teollisuuden prosessien ja informatiivisten tietotarpeiden vaatimaa tukea tietovarastolta ei ole juuri käsitelty. Teollisuus 4.0 -termin alla painopiste tutkimuksessa on ollut koneiden ja tuotteiden muuttamisessa älykkäämmiksi, moderneissa valmistusmenetelmissä ja datan analysointimenetelmissä, mutta ison datan tallentaminen tietovarastoon ja tietovarastolta vaadittava tuki ison datan analysoimiselle on jäänyt vähälle huomiolle. Lisäksi tietovaraston tukea holistisen näkemyksen muodostumiselle teollisessa yrityksessä on tutkittu vähän.

Tämän opinnäytetyön tavoite on luoda viitekehys, jonka avulla olisi mahdollista lähteä toteuttamaan modernia tietovarastoa, joka tukisi ison datan tallennusta ja analysoimista sekä holistisen näkemyksen muodostumista teollisessa yrityksessä. Opinnäytetyön tuloksena esitellään kirjallisuustutkimukseen perustuva integroitu malli, jonka perustana on kirjallisuudessa esitelty kerroksellinen arkkitehtuuri (Krishnan [2013], Pokorný [2014]), joka mahdollistaa ison datan kustannustehokkaasti skaalautuvan tallentamisen ja ison datan analysoimisen. Kerroksellisessa arkkitehtuurissa hyödynnetään Hadoop:ia ja NoSQL-teknologioita. Koska Hadoop on kirjallisuudessa yleisimmin käytetty hajautettu tiedostojärjestelmä, tässä opinnäytetyössä ei käsitellä muita hajautettuja tiedostojärjestelmiä. Tietovaraston arkkitehtuuri on yhdistetty DMAIC-iteraatiota [Singh and Khanduja 2015] noudattavaan datan hyödyntämisprosessiin, joka tukee yrityksen toiminnan kehittämistä jatkuvan parantamisen periaatteella [Hopp and Spearman 2000]. Esitetyssä integroidussa mallissa on lisäksi ehdotettu sisältöperusteisen suosittelujärjestelmän [Lops *et al*. 2011] käyttämistä holistisen näkemyksen edistämiseksi yrityksessä [Bhutta *et al*. 1999] ehdottamalla käyttäjille tietovarastoon tallennettuja aineistoja.

2. Yrityksen tietovarasto

Tässä luvussa esitellään mikä yrityksen tietovarasto on, mihin sitä tarvitaan ja mitä ohjelmistokomponentteja perinteiseen tietovarastoon kuuluu. Lisäksi esitellään muutama yleisesti käytetty arkkitehtuuri yrityksen tietovarastolle.

2.1 Mikä yrityksen tietovarasto on?

Bill Inmonin yleisesti hyväksytyn määritelmän mukaan yrityksen tietovarasto (data warehouse) on sovellusalueeseen ja aikaan sidottu, vakaa, integroitu kokoelma dataa, jota käytetään päätöksenteon tukena. Sovellusalueeseen sidottu tarkoittaa, että tietovaraston avulla voidaan analysoida tiettyä sovellusaluetta, kuten vaikkapa markkinointia. Vakaa tarkoittaa, että tietovarastoon tallennettua dataa ei pitäisi myöhemmin muuttaa. Integroitu merkitsee eri lähteistä yhdistetyn datan yhteensopivuutta. Aikaan sidottu tarkoittaa, että yrityksen tietovarastoon on tallennettu historiatietoa. [Krishnan 2013]

Ralph Kimballin määritelmän mukaan tietovarasto on kopio transaktiodatasta, joka on strukturoitu erityisesti hakemista ja analysoimista varten. [Kimball 1996]

Krishnanin [2013] ja Nagabushananin [2006] mukaan yrityksen tietovaraston tarkoitus on olla yrityksen ainoa versio totuudesta (“single version of truth”). Ainoalla versiolla totuudesta tarkoitetaan, että kaikesta päätöksenteon ja operatiivisen toiminnan kannalta oleellisesta datasta on olemassa vain yksi versio, joka on käyttäjien saatavilla tietovarastossa ja jota kaikki käyttävät.

2.2 Mihin yrityksen tietovarastoa tarvitaan?

Yrityksen tietovaraston tulee vastata erilaisiin liiketoiminnan tarpeista seuraaviin tietotarpeisiin, joita voi olla kolmenlaisia [Nagabushanan 2006]:

- *Operatiivisia*: Järjestelmien tulee mahdollistaa sujuva päivittäisen toiminnan johtaminen ja kehittäminen sekä poikkeamien tunnistaminen ja korjaaminen poikkeamaraportoinnin ja työnkulunvalvonnan avulla. Operatiivisen toiminnan kannalta tietovarastosta saadun tiedon avulla on siis tarkoitus saada jokin liiketoimintaprosessi optimoitua tai suoritettua.

- *Päätöksenteollisia*: Järjestelmien on tuettava johdon päätöksentekoa ja pitkän aikavälin suunnittelua. Tietovarastosta haettua dataa hyödyntävien päätöksentekoa tukevien järjestelmien avulla voidaan tehdä esimerkiksi skenaarioperusteista mallinnusta, mitä jos -analyyseja ja analysoida trendejä.

- *Informatiivisia*: Järjestelmien tulee mahdollistaa esimerkiksi toimintaohjeiden, koulutusmateriaalien ja tuotetiedon kaltaisen tiedon jakaminen organisaatiossa sitä tarvitseville ihmisille.

Huonosti liiketoimintatarpeisiin vastaava tietovarasto voi johtaa organisaatiossa menetettyihin mahdollisuuksiin, joita voivat olla esimerkiksi [Krishnan 2013] :

- kilpailuedun saavuttaminen (Gaining competitive advantage),
- operatiivisten ja taloudellisten riskien pienentäminen (Reducing operational and financial risks),
- tulojen kasvattaminen (Increasing revenue),
- ydinliiketoimintojen tehokkuuden optimointi (Optimizing core business efficiencies) ja
- trendien analysoiminen ja ennakointi (Analyzing and predicting trends and behaviors).

Kilpailuetu voi esimerkiksi jäädä yrityksessä saavuttamatta yrityksen tietovaraston takia silloin, kun operatiivisen toiminnan kehittämisen kautta on saatu aikaan parempi tuotelaatu, mutta tieto parantuneesta laadusta ei saavuta asiakkaita, koska markkinoinnilla ei ole ollut tietoa parannuksen viestimiseksi asiakkaille. Tällöin tietovarasto ei ole vastannut organisaation informatiivisiin liiketoimintalähtöisiin tietotarpeisiin eikä oikea tieto ole saavuttanut oikeita henkilöitä oikeaan aikaan.

2.3 Yrityksen tietovaraston ohjelmistoarkkitehtuuri

Perinteisen yrityksen tietovaraston ohjelmistoarkkitehtuurin tarvitsemat ohjelmistokomponentit ovat kirjallisuuden perusteella varsin vakiintuneet. Ne esitellään alakohdassa 2.3.1. Lisäksi alakohdassa 2.3.2 esitellään kaksi yleisesti käytössä olevaa perusmallia yrityksen tietovaraston arkkitehtuurista, joilla tietovarastoja on käytännössä toteutettu monissa organisaatioissa.

2.3.1 Perinteisen yrityksen tietovaraston arkkitehtuurin tarvitsemat ohjelmistokomponentit

Nagabushananin [2006] mukaan tietovaraston ohjelmistokomponentit voidaan jakaa kolmeen kerrokseen:

- Tiedon keräily- ja muuntokerros (extraction & transformation).
- Tietovarastointiteknologiakerros (dw technology).
- Datan haku- ja käyttökerros (data access & retrieval).

Tiedon keräily- ja muuntokerroksessa tarvittava data kerätään lähdejärjestelmistä, muunnetaan yhteensopivaan muotoon, integroidaan, puhdistetaan ja ladataan tietovaraston yhteen tai useampaan tietokantatauluun.

Tiedon keräämisvaiheessa kootaan ja tulkitaan lähdedataa ja kopioidaan tarvittava data jatkokäsittelyä varten. Tästä hetkestä eteenpäin data on osa yrityksen tietovarastoa. [Kimball and Ross 2013]

Datan muuntamisessa yhteensopivaan muotoon lähdejärjestelmistä kerätty data muunnetaan tietovarastossa käytettäviin formaatteihin ja yksiköihin. Datan integroinnissa yhdistetään eri lähdejärjestelmistä kopioitua toisiinsa liittyvää dataa. [Nagabushanan 2006]

Datan puhdistamisessa esimerkiksi korjataan kirjoitusvirheitä datassa, poistetaan tai täydennetään puuttuvia arvoja ja poistetaan duplikaatteja. Keräily- ja muuntokerroksessa myös ratkaistaan eri lähdejärjestelmien datan kesken esiintyvät törmäävät nimeämiskäytännöt (resolving domain conflicts). Tiedon latausvaiheessa data strukturoidaan tietovaraston rakenteen mukaiseksi ja ladataan tietovarastoon dimensio- ja faktatauluihin. [Kimball and Ross 2013]

Nagabushananin [2006] mukaan tietovarastoteknologiakerroksessa ovat tallennettuina data ja siihen liittyvä metadata ja relaatiotietokannat soveltuvat hyvin isoille ja kasvaville tietovarastoille. Kimball ja Ross [2013] puolestaan esittävät, että datan tulisi olla tallennettuna relationaalisiin tähtikaavioihin (relational star schema) tai OLAP-kuutioihin (online analytical processing). Lisäksi heidän mukaansa tulisi tallentaa aggregaatiodatan lisäksi yksityiskohtainen data, jotta käyttäjän on mahdollista tarkastella syvällisemmin miten aggregaatiodata on muodostunut. Bill Inmonin lähestymistapa on, että data tulisi tallentaa tietovarastoon kolmannessa normaalimuodossa (3NF) (Kimball ja Ross [2013], Krishnan [2013]). Kolmannella normaalimuodolla tarkoitetaan, että tietokannan relaatiot ovat toisessa normaalimuodossa ja lisäksi attribuuteilla, jotka eivät ole avaimia, ei saa olla ei-triviaaleja funktionaalisia riippuvuuksia muihin kuin avainehdokkaiden superjoukkoon. Toinen normaalimuoto tarkoittaa, että jokainen relaation ominaisuus, joka ei ole avainominaisuus, on täydellisesti riippuvainen jokaisesta relaation avainehdokkaasta ja että relaatio on ensimmäisessä normaalimuodossa. Ensimmäinen normaalimuoto tarkoittaa, että jokaisen relaation sarakkeen arvojen tulee olla atomisia.

Datan haku- ja käyttökerros sisältää ohjelmistot, joita tarvitaan tietovaraston datan hakemiseksi, jakamiseksi ja esittämiseksi [Nagabushanan 2006]. Kimball ja Ross [2013] esittävät myös, että tässä kerroksessa olevat ohjelmistot voivat olla yksinkertaisia, kuten nopeaan hakemiseen tarkoitettu työkalu, tai toisaalta isoja ja monimutkaisia ohjelmistoja, joita voidaan käyttää esimerkiksi tiedonlouhintaan.

Krishnanin [2013], Nagabushananin [2006] ja Kimballin ja Rossin [2013] esittämät arkkitehtuurit ovat peruspiirteiltään samanlaisia keskenään. Suurin ero on terminologiassa. Krishnan esimerkiksi käyttää termiä valmistelualue (staging area) tietokannasta, jossa dataa muunnetaan, integroidaan ja puhdistetaan yrityksen tietovarastoon lataamista varten. Nagabushanan esitteli kyseiset toiminnot keräily- ja muuntokerroksen komponentteina. Kimball ja Ross puolestaan jakavat yrityksen tietovaraston kahteen kerrokseen:

- takahuoneeseen (backroom) ja
- etuhuoneeseen (front room)

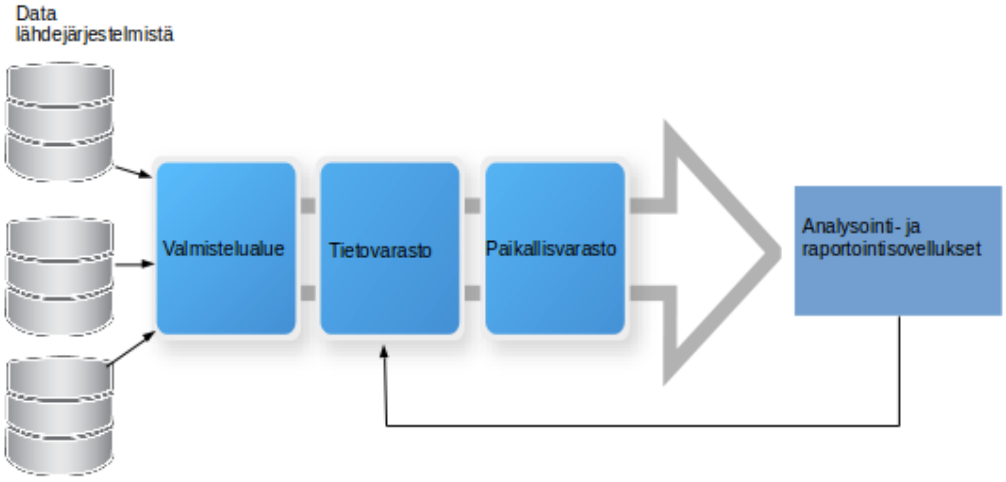
Takahuoneella Kimball ja Ross tarkoittavat samaa kuin Nagabushanan tiedon keräily- ja muuntokerroksella. Etuhuone-termi tarkoittaa samaa kuin Nagabushananin tietovarastointiteknologia ja datan haku- ja käyttökerros yhdessä.

Krishnan ja Nagabushanan esittelevät lisäksi operatiivisen datavaraston (operational data store, ODS) mahdollisena ohjelmistokomponenttina, joka voi Krishnanin mukaan samalla toimia valmistelualueena. ODS on tarkoitettu päivittäisessä operatiivisen toiminnan johtamisessa tarvittavan tilannedatan esittämiseen. ODS:n dataa päivitetään jatkuvasti eikä siellä säilytetä historiadataa, toisin kuin tietovarastossa.

Seuraavissa kappaleissa esitellään kaksi laajasti käytettyä arkkitehtuuria perinteisen yrityksen tietovaraston toteutukselle. Toinen esimerkkiarkkitehtuuri on Bill Inmonin esittelemä yrityksen informaatiotehdas ja toinen Ralph Kimballin esittelemä Kimball-arkkitehtuuri.

2.3.2 Yrityksen informaatiotehdas

Bill Inmonin esittämä yrityksen informaatiotehdas, cif (corporate information factory, Krishnan [2013] ja Kimball ja Ross [2013]) on arkkitehtuuri yrityksen tietovarastolle. Informaatiotehtaassa (kuva 1) lähdetään liikkeelle lähdejärjestelmistä ylhäältä alaspäin (top-down) lähestymistavalla. Data kerätään aluksi lähdejärjestelmistä tietovaraston valmistelualueelle, jossa dataa puhdistetaan ja sen laatu varmistetaan. Seuraavassa vaiheessa valmistelualueella käsitelty data muunnetaan tietovaraston mukaisiin formaatteihin ja data tallennetaan tietovarastoon. Tietovaraston tietokannan tulee olla informaatiotehdas-mallin mukaan kolmannessa normaalimuodossa (kuva 2), jolloin data voidaan tallettaa tietovarastoon mahdollisimman vähillä muunnoksilla. Informaatiotehdas-mallissa tietovarastossa on liiketoimintatarpeista riippuen useita näkyminä (views) ja aggregaatiotauluina toteutettuja virtuaalisia kerroksia, joita analysointi- ja raportointisovellukset voivat hyödyntää. Mallissa on useimmiten toteutettuna erillinen paikallisvarasto (datamart), jossa dataa voidaan jatkokäsitellä.



Kuva 1: Yrityksen informaatiotehdas.

Postinumero	Postitoimipaikka
33100	Tampere
36110	Ruutana

Opiskelija_ID	Etunimi	Sukunimi	Osoite	Postinumero
12345	Ossi	Onnekas	Olematonkatu 1 C 3	33100
54321	Ilona	Innokas	Puurtajanraitti 2	36110
12534	Nelli	Noheva	Esplanadi 3 B 2	33100

Kuva 2: Esimerkki tietokannan taulunäkymästä kolmannessa normaalimuodossa.

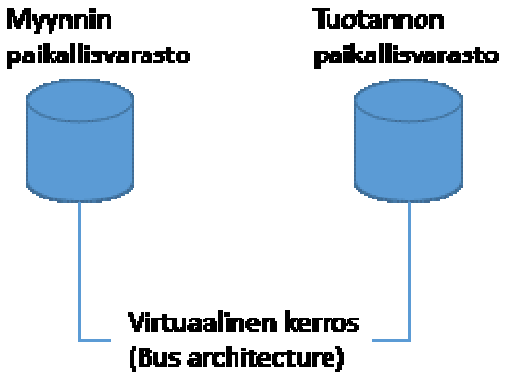
Krishnanin [2013] mukaan informaatiotehtaan vahvuuksia on, että se sisältää koko yrityksen laajuisesti dataa joka on yhdessä keskitetyssä tietokannassa, jolloin esimerkiksi poikkifunktionaalinen analysoiminen on helpompaa. Poikkifunktionaalinen analysoiminen tarkoittaa, että analysointia varten tarvitaan dataa useista yrityksen toiminnallisista osista, esimerkiksi myynnistä ja valmistuksesta. Lisäksi arkkitehtuuri, datan käsittelyssä käytettävät säännöt ja tietovaraston hallinnointi on toteutettu keskitetysti.

Krishnan [2013] kritisoi, että arkkitehtuurin heikkouksia ovat esimerkiksi ylläpidon kalleus, datan esikäsittelyn muodostuminen pullonkaulaksi, jos dataa pitää esimerkiksi

puhdistaa paljon ja korkea häiriöriski. Kimball ja Ross [2013] puolestaan toteavat informaatiotehtaan heikkouden olevan, että jos datan käsittelyssä ennen tallennusta tietovarastoon tehdään muunnoksia, esimerkiksi muutetaan data-attribuuttien nimiä tai tehdään erilaisia laskutoimituksia osastoittain, niin voi olla hankalaa yhdistää muunnoksien tuloksia atomiseen dataan haku- ja käyttökerroksen analysoimisessa.

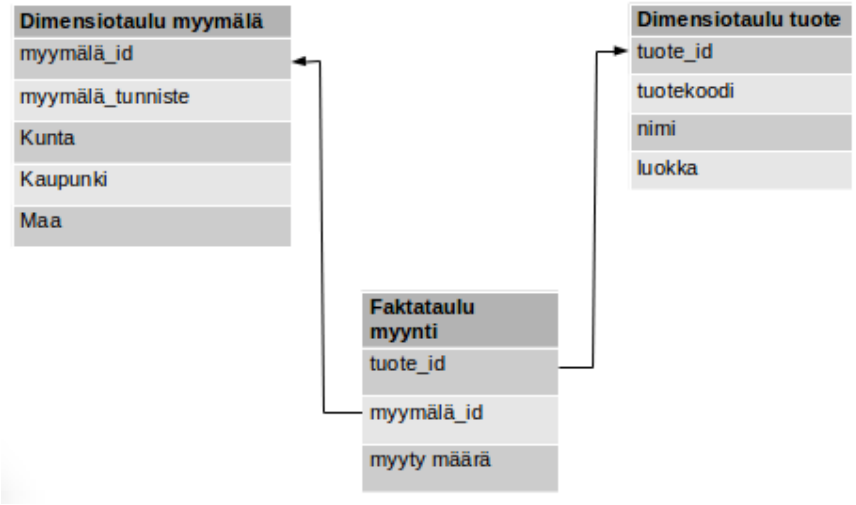
2.3.3 Kimball-arkkitehtuuri

Kimball-arkkitehtuurissa yrityksen tietovarastoa rakennetaan alhaalta ylöspäin (bottom up). Arkkitehtuuri perustuu tiukasti integroituihin paikallisvarastoihin (kuva 3), jotka on toteutettu dimensionaalisella datamallilla. [Krishnan 2013]



Kuva 3: Kimball-arkkitehtuuri.

Dimensionaalisessa datamallissa data on esitetty dimensiorelaatioissa ja faktarelaatioissa, jotka yhdistävät dimensiorelaatioita toisiinsa. Kuvassa 4 on esitetty dimensionaalisen datamallin toteutus tähtikaaviolla. Arkkitehtuurin avulla yritys voi toteuttaa ensin paikallisvarastoja eri liiketoiminta-alueille ja -funktioille, kuten myynnille ja tuotannolle ja yhdistää myöhemmin paikallisvarastot niiden yhteisten dimensioiden avulla yrityksen laajuiseksi tietovarastoksi. Eräs myynnin ja tuotannon paikallisvarastoja yhdistävä dimensio voisi olla asiakasdata. [Krishnan 2013]



Kuva 4: Esimerkki tähtikaaviona toteutetusta tietokannan taulunäkymästä.

Data kerätään aluksi lähdejärjestelmistä, puhdistetaan, integroidaan, muunnetaan ja tallennetaan liiketoimintaprosessien mukaisiin paikallisvarastoihin niiden dimensionaalisen rakenteen mukaisesti. Tämän vaiheen tarkoituksena on varmistaa paikallisvarastoihin tallennettavan datan yhtenäisyys ja yhdenmukaisuus. [Kimball and Ross 2013]

Tietovarasto on toteutettu Kimball-arkkitehtuurissa virtuaalisena kerroksena (BUS architecture), jonka kautta paikallisvarastojen dataa voidaan yhdistää niiden yhteisten dimensioiden perusteella. Raportointi- ja analysointisovellukset ja käyttäjät voivat käyttää virtuaalikerrosta datan hakemiseen ja yhdistämiseen eri paikallisvarastoista. [Krishnan 2013]

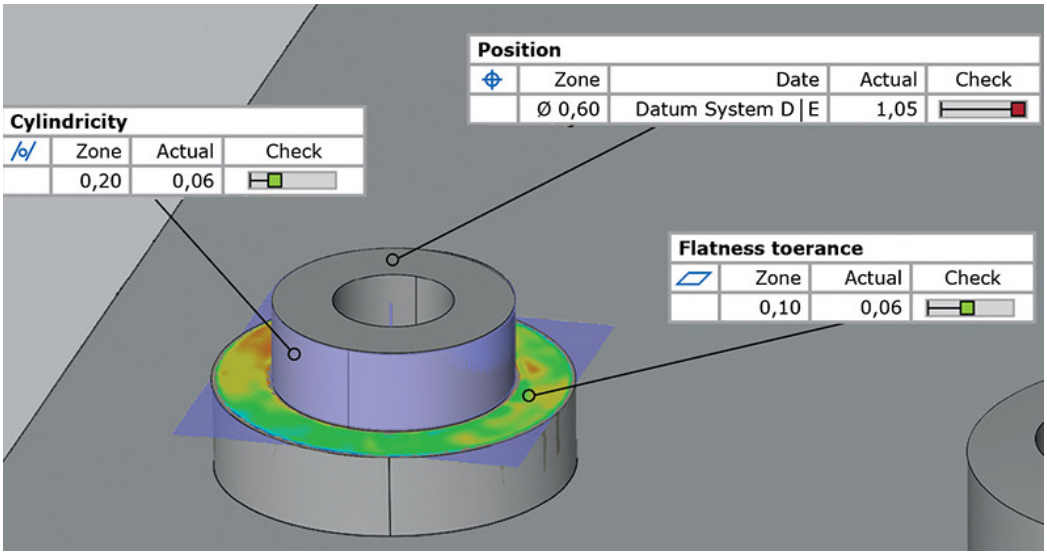
Kimball-arkkitehtuurin etuja ovat toteutettavuus useana hallittavana moduulina (paikallisvarastot) ja paikallisvarastojen yksinkertaisempi rakenne. Arkkitehtuurin heikkouksiksi Krishnan mainitsee paikallisvarastojen arkkitehtuurin redundanttiuden ja vaikeuden tehdä komplekseja liiketoiminta-analyyseja, joissa dataa tarvitaan laajasti useista paikallisvarastoista. [Krishnan 2013]

3. Teollinen internet ja sen vaikutus yrityksen tietovarastointiin

Seuraavaksi esitellään mitä teollisella internetillä tarkoitetaan ja teollisen internetin aiheuttamia muutostarpeita tietovarastoon. Lisäksi esitellään teknologioita ja arkkitehtuureita uudenlaisen ison datan tallentamiseen ja analysoimiseen soveltuvan tietovaraston toteuttamiseksi.

3.1 Teollinen internet

Bruner [2013] esittää, että teollinen internet tarkoittaa tietoverkkoja, joissa "älykkäät" koneet ja laitteet voivat viestiä käyttäen avoimia protokollia. Tällöin lähes jokaisesta koneesta voi tulla sensori. Siitä seuraa internetin kaltainen verkosto, jossa koneet julkistavat dataa autorisoiduille tahoille ja ottavat vastaan komentoja autorisoiduilta tahoilta. Ivanovin ja muiden [2014] mukaan tulevaisuuden versiossa teollisista tietoverkoista koneiden ja laitteiden lisäksi myös tuotteet voivat vuorovaikuttaa toistensa kanssa ilman ihmisen ohjausta. Samalla koneet tuottavat huomattavasti enemmän dataa, kuin perinteiset tuotantolaitteet, joilla on ollut kirjausten tekemisestä vastaava käyttäjä. Bruner käyttää esimerkkinä matkustajakonetta, jossa kokenut lentäjä voi kerätä 10 000...20 000 tuntia lentokokemusta uransa aikana, mutta lentäjättömän lentokoneen ohjausjärjestelmä voisi kerätä vuodessa järjestelmää käyttävistä koneista satojen tuhansien tuntien verran dataa. Posadan ja muiden [2015] mukaan yhdessä monimutkaisessa laitteessa voi olla muutamia tuhansia antureita, joista pitäisi lukea arvot joissain tapauksissa muutaman millisekunnin välein, jolloin vuodessa kertyisi tallennettavaa dataa useita miljardeja arvoja. Lisäksi he esittävät, että datan suuri määrä ei ole ainoa ongelma, vaan lisäksi data pitää pystyä organisoimaan niin, että ihminen pystyy ymmärtämään sen ja tekemään päätöksiä sen perusteella. Lisäksi he mainitsevat, että tuotannossa tarvittavan datan muoto voi olla monimutkainen. Monimutkaisessa muodossa olevaa dataa ovat esimerkiksi valmistettavien kappaleiden 3d-mallit, mittaustulokset, joissa valmistettua kappaletta on verrattu 3d-malliin (kuva 5) tai valmistukseen käytettävät CAM-mallit (computer-aided manufacturing, tietokoneavusteinen valmistus).



Kuva 5: Mittaustulos, jossa valmistettua kappaletta on verrattu 3d-malliin (http://www.gom.com/3d-software/gom-inspect/features.html, viitattu 10.03.16).

Nagabushanan [2006] esittelee virtuaalisen yrityksen käsitteen, jossa yrityksen toiminta on hajaantunut useisiin toimipaikkoihin ja lisäksi yritykset muodostavat strategisia verkostoja, jolloin tiedon on myös siirryttävä yritysten välillä sujuvan operatiivisen toiminnan mahdollistamiseksi. Posadan ja muiden [2015] mukaan tulevaisuudessa tuotantolaitteet voivat kommunikoida itsenäisesti keskenään ja mahdollistaa saumattoman yhteydenpidon eri paikoissa sijaitsevien tehtaiden välillä. Tällöin tuotantojärjestelmä voi esimerkiksi etsiä alihankintaverkostosta valmistettavan kappaleen seuraavan työvaiheen tekemiseen soveltuvimman tuotantolaitteen, jolla on tarjolla kapasiteettia ja ohjata kappaleen lähetettäväksi sinne.

3.2 Teollisen internetin vaikutus yrityksen tietovarastointiin ja datan analysointiin

Edellä on esitetty, että teollinen internet lisää datan määrää yrityksessä, uutta dataa syntyy entistä nopeammin ja samalla datasta tulee monimutkaisempaa. Älykkäämmät koneet ja laitteet tarvitsevat aikaisempaa monimutkaisempaa dataa toimiakseen. Lisäksi koneet ja laitteet tallentavat dataa automaattisesti ilman ihmisen tulkintaa, jolloin datan organisointi ja analysointi on monimutkaisempaa, kuin jos ihminen olisi jo tulkinnut dataa kirjauksia tehdessään. Jaroslav Pokorný [2014] määrittelee, että isolla datalla (big data) tarkoitetaan dataa, jota on määränsä takia vaikea tai mahdoton kerätä, prosessoida, hakea, hallita ja analysoida perinteisillä tietokantatyökaluilla. Pokornýn ja monen muun (mm. Douglas Laney [2001] ja Krishnan [2013]) mukaan isoa dataa voidaan luonnehtia kolmella tai neljällä tekijällä määrittelijästä riippuen (jatkossa 3-4 V:n määritelmä sanojen Volume, Velocity, Variety ja Veracity mukaan) :

- *Volyymi (Volume)*: Datan suuri määrä.

- *Nopeus (Velocity)*: Uutta dataa syntyy lisää nopeasti.

- *Moninaisuus (Variety)*: Dataa on lukuisissa erilaisissa muodoissa, esimerkiksi strukturoituna, puolistrukturoituna ja strukturoimattomana, kuten tekstinä ja kuvatiedostoina. Lisäksi teollisessa ympäristössä tarvitaan usein esimerkiksi 3d-malleja valmistettavista tuotteista ja CAM-tiedostoja tuotteiden valmistamiseksi.

- *Epävarmuus (Veracity)*: Kerättyyn dataan liittyvä epävarmuus, kuten tehtyjen mittausten mittausepävarmuus.

Toinen ehdotettu lähestymistapa ison datan tunnusmerkistöksi on Wun ja muiden [2014] esittämä HACE-teoreema. HACE-teoreeman mukaan isolla datalla tarkoitetaan isoa datan volyymia heterogeenisistä ja autonomisista datalähteistä hajautetulla ja keskittämättömällä hallinnolla ja josta pyritään löytämään monimutkaisia ja kehittyviä data-suhteita.

HACE-teoreemassa on yhteneväisyyksiä 3-4 V:n määritelmään. Molemmissa tunnistetaan ison datavolyymin merkitys ja datan moninaisuus, joka seuraa itsenäisistä ja autonomisista datalähteistä, joita ei hallita keskitetysti. Valmistavassa teollisuudessa tällaisia heterogeenisiä ja autonomisia datalähteitä ovat Nagabushananin [2006] esittelemään virtuaaliseen yritykseen liittyvät itsenäiset yhteistyökumppanit, joiden kanssa organisaatio muodostaa verkoston. Tällaisen verkoston sujuva ja tehokas toiminta edellyttää datan vaihtamista, mutta kaikilla toimijoilla voi olla omat järjestelmät ja toimintatavat, jolloin data on heterogeenistä. Eroavaisuutena 3-4 V:n määritelmään on, että HACE-teoreemassa mainitaan lisäksi monimutkaisten ja kehittyvien data-suhteiden löytäminen. Virtuaalisen yrityksen toiminnassa autonomisten toimijoiden tuottamaa dataa täytyy integroida, jolloin data-suhteiden tunnistaminen on oleellista. Toisin kuin HACE-teoreemassa virtuaalinen yritys saattaa pystyä hallitsemaan keskitetysti verkoston tuottamaa dataa ainakin jossain määrin.

Kirjallisuudessa on viime vuosina yleisesti tunnistettu (mm. Pokorný [2014], Krishnan [2013], Hu ja Kaabouch [2013]), että perinteisillä, relaatiotietokantoihin ja SQL:ään perustuvilla, tietokantateknologioilla ei pystytä vastaamaan kustannustehokkaasti, skaalautuvasti ja moderneja analysointitekniikoita, kuten koneoppimista, tukevasti ison datan asettamiin haasteisiin. Krishnanin [2013] mukaan esimerkiksi cif- ja Kimball-arkkitehtuurit eivät pysty vastaamaan ison datan asettamiin haasteisiin, koska:

- Ne ovat riippuvaisia relaatiotietokannoista. Relationaalinen datamalli rajoittaa mahdollisuuksia toteuttaa joustavia arkkitehtuureja.

- Ne perustuvat kaikki jaettu -arkkitehtuuriin (shared everything architecture), jolloin suorituskyky ja suorituskyvyn lisäämisen korkea kustannus muodostuvat ongelmiksi.

- Suorituskyky- ja skaalautuvuusvaatimukset kasvavat jatkuvasti.

- Ne eivät pysty tukemaan ison datan analysoimista (big data analysis).

Toisaalta Carlos Ordonez [2013] esittää, että vaikka hänenkin mukaansa relaatiotietokantojen skaalaaminen on hankalaa ja kallista, niin ison datan analysointi relaatiotietokannoissa tarjoaa myös etuja. Ordonezin mukaan näitä etuja ovat esimerkiksi turvallisuus, pienempi datan epäyhteneväisyys (data inconsistency) ja ennen kaikkea se ettei datan hakeminen relaatiotietokannoista muodostu pullonkaulaksi.
Krishnanin [2013] mukaan yrityksen tietovaraston tarve ja tarkoitus ei muutu, mutta asioita kyetään tekemään entistä paremmin uudenlaisia arkkitehtuureja ja teknologioita hyödyntämällä. Sekä Krishnan [2013] että Pokorný [2014] esittävät, että ison datan käsittelyyn soveltuvan arkkitehtuurin tulisi olla kerroksellinen erilaisten dataformaattien käsittelyä varten ja hyödyntää perinteisten relaatiotietokantojen lisäksi esimerkiksi NoSQL- tai Hadoop-teknologioita. Myös Wu ja muut [2014] esittävät esimerkiksi Hadoopin ja Wekan tai R:n integroimista ison data analysoimista varten.

3.3 Hajautetut tiedostojärjestelmät ja NoSQL teknologiat

Luvussa 3.3.1 esitellään hajautettuja tiedostojärjestelmiä ja erityisesti Hadoopia, joka on laajasti käytetty hajautettu tiedostonhallintajärjestelmä. Alakohdassa 3.3.2 esitellään NoSQL-teknologioita.

3.3.1 Hajautetut tiedostojärjestelmät ja Hadoop

Hajautetut tiedostojärjestelmät (distributed file systems) eroavat perinteisistä tietoverkkoyhteydellä yhdistetyistä tiedostojärjestelmistä, kuten UNIXista, esimerkiksi hyödyntämällä tiedostojen ositusta (file partitioning) ja monistusta (replications). Tunnetuin on Hadoopin hajautettu tiedostojärjestelmä. [Pokorný 2014]

Apache Hadoop on ohjelmistokehys (framework), joka mahdollistaa isojen data-aineistojen hajautetun tallentamisen ja käsittelyn jopa tuhansien tietokoneiden muodostamilla klustereilla yksinkertaisia ohjelmointimalleja käyttäen. Hadoopin

lähtökohta on, että halpoja vakiolaitteistoja käytettäessä laitteistosta johtuvat virhetilanteet ovat todennäköisiä ja kaikki Hadoopin moduulit on suunniteltu niin, että laitteistoviasta johtuvat virheet käsitellään automaattisesti ohjelmistokirjaston sisällä. [Hadoop]

Hadoop mahdollistaa ison datan käsittelyn hajautetusti kustannuksilla, jotka relaatiotietokantapohjaisiin ratkaisuihin verrattuna ovat pienempiä. [Krishnan 2013]

Hadoopin perusmoduulit ovat [Hadoop]:

- *Hadoop yleinen (Hadoop Common)*: Yleiset palvelut, joita muut moduulit tarvitsevat toimiakseen.

- *Hadoop hajautettu tiedostojärjestelmä (Hadoop Distributed File System, HDFS):* Hajautettu tiedostojärjestelmä, joka tarjoaa tehokkaan pääsyn dataan.

- *Hadoop YARN*: Ohjelmistokehys (framework) kuormanjakamiseen (job scheduling) ja klusteriresurssien hallintaan (cluster resource management).

- *Hadoop MapReduce*: YARN:iin perustuva järjestelmä isojen data-aineistojen hajautettuun käsittelyyn.

Muita tunnettuja Hadoop:iin perustuvia tai sitä hyödyntäviä ja laajasti käytettyjä ohjelmistoja ovat mm. Hive ja Pig [Krishnan 2013]. Näiden yksityiskohtainen tarkastelu sivuutetaan tässä tutkielmassa.

3.3.2 NoSQL teknologiat

NoSQL (Not only SQL) tarkoittaa tietokantateknologioita, jotka eivät perustu relaatiotietietokantoihin eivätkä SQL:ään (structured query language). NoSQL teknologioiden etu ison datan käsittelyssä on kyky skaalautua datan volyymin, nopeuden ja moninaisuuden kasvaessa. [Krishnan 2013]

NoSQL -tietokantojen suurin ero relaatiotietokantoihin verrattuna on kaavioton datamalli (schemaless data model), jota voidaan muokata joustavasti ilman tietokannan palvelukatkoksia. Relaatiotietokannalla on määrätty kaavio, jonka muuttaminen vaatii ylläpitotoimenpiteitä, joiden aikana tietokanta ei ole käytettävissä. Lisäksi NoSQL -tietokannat on suunniteltu alusta asti ison datan hajautettua tallentamista ja hakemista varten ja niiden joustava kyky skaalautua datan volyymin mukana perustuu kuorman jakamiseen useille palvelimille. Siksi NoSQL -tietokannat tarjoavat relaatiotietokantoja halvemman toteutuksen ison datan tallentamiseen. [Bhogal and Choksi 2015]

Krishnan [2013] ja Bhogal ja Choksi [2015] määrittelevät, että NoSQL -tietokannanhallintajärjestelmät voidaan jakaa karkeasti neljään luokkaan:

- *Avain-arvo -parit (key-value pairs)*: Tietokanta tallentaa avain-arvo -pareja. Avaimen tulee olla uniikki ja sen perusteella voidaan hakea sitä vastaava arvo.

- *Sarakeorientoituneet (Column family stores)*: Data tallennetaan tietokantaan sarakkeittain eikä riveittäin, kuten relaatiotietokannoissa useimmiten tehdään.

- *Dokumenttitietokannat (Document databases):* Data tallennetaan dokumenttina, joka useimmiten esitetään JSON (JavaScript object notation)- tai XML (Extensible Markup Language) -muodossa.

- *Graafitietokannat (Graph databases):* Tietokannat perustuvat graafiteoriaan.

3.3.3 Hadoopin ja NoSQL:n hyödyt

Hadoopin etuna on mahdollisuus toteuttaa datan analysoimiseen ydin, joka mahdollistaa ison datan hajautetun käsittelyn ja analysoimisen esimerkiksi koneoppimisen avulla. NoSQL -teknologiat mahdollistavat yrityksille helposti kehitettävän ison datan tallennuksen ja lisäksi NoSQL -teknologiat ovat joustavasti skaalautuvia. Lisäksi Hadoopin ytimeen voidaan yhdistää NoSQL -teknologiaa avain-arvo -parien hakemiseen ja analysoimiseen. [Kumar *et al*. 2014]

Jagtapin ja Patilin [2014] artikkelin perusteella NoSQL:n avulla voidaan tallentaa valtavia datajoukkoja tehokkaasti, mutta datasta on hyötyä ainoastaan, jos sitä pystytään hyödyntämään. Heidän mukaansa Hadoop mahdollistaa tällaisten suurten datajoukkojen hyödyntämisen. Hadoop- ja NoSQL-teknologioita voidaan siis käyttää yhdessä tai erikseen niin, että NoSQL mahdollistaa ison datan tehokkaan tallentamisen ja hakemisen skaalautuvalla tavalla ja Hadoop mahdollistaa datan analysoimisen tehokkaasti.

3.4 Moderneja arkkitehtuureita tietovaraston toteuttamiseksi teollisen internetin ympäristössä

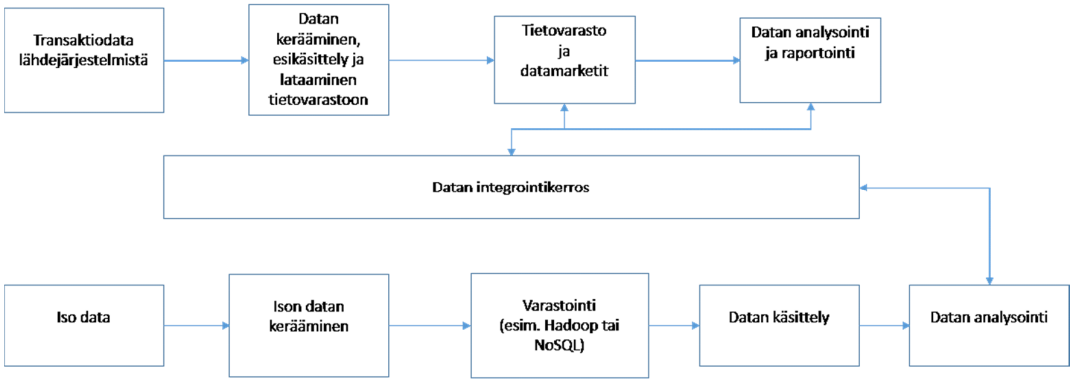
Krishnan [2013] esittelee kaksi arkkitehtuuria, jotka selviytyvät teollisesta internetistä seuraavassa ison datan ympäristössä ja joissa yrityksen mahdollisesti olemassa olevan tietovaraston rakenne voidaan säilyttää ennallaan:

- Ulkoinen dataintegraatioarkkitehtuuri (external data integration) ja
- Integraatiolähtöinen arkkitehtuuri (integration-driven approach).

Muissa Krishnanin ehdottamissa arkkitehtuureissa olemassa olevaa tietovarastoa jouduttaisiin muuttamaan sisäiseltä rakenteeltaan. Joko tiedon keräily- ja muuntokerroksen ja tietovarastointiteknologiakerroksen tai sitten tietovarastointiteknologiakerroksen ja datan haku- ja käyttökerroksen väliin tarvittaisiin uusi komponentti. Yritysten investoitua merkittäviä määriä rahaa ja muita resursseja olemassa oleviin tietovarastoihin ei niiden rakennetta kannata muuttaa, jos syynä on ainoastaan tarve selvitä ison datan ympäristön mukanaan tuomista haasteista. Tällöin kaikki resurssit voidaan keskittää uusien ominaisuuksien integroimiseen olemassa olevaan arkkitehtuuriin. Jos yrityksen olemassa oleva tietovarasto vaatii muutenkin muutoksia, voivat muutkin arkkitehtuurit olla hyviä vaihtoehtoja. Tässä opinnäytetyössä oletuksena on, että olemassa olevaa tietovarastoarkkitehtuuria laajennetaan, mutta olemassa olevan ratkaisun sisäistä rakennetta ei muuteta arkkitehtuuritasolla.

3.4.1 Ulkoinen dataintegraatioarkkitehtuuri

Ulkoisessa dataintegraatioarkkitehtuurissa (kuva 6) olemassa oleva tietovarasto voidaan säilyttää ennallaan ja sen rinnalle voidaan toteuttaa ison datan keräämiseen, tallentamiseen, käsittelyyn ja analysointiin soveltuva järjestelmä. Kuvan 6 nuolet esittävät datan liikkumista arkkitehtuurin komponenttien välillä. Datan integrointi näiden kerrosten kesken toteutetaan erillisellä kerroksella, jonka tekemisessä hyödynnetään metadataa ja semanttisia teknologioita.



Kuva 6: Ulkoinen dataintegraatioarkkitehtuuri [Krishnan 2013].

Metadata on dataa datasta. Sen avulla voidaan liittää esimerkiksi tietokannan attribuuttiin sen merkitys. Semanttisten teknologioiden avulla voidaan esittää asioiden merkityksiä erillään datasta ja tiedostoista ja ne mahdollistavat asioiden suhteiden ymmärtämisen. Datan integrointikerros on erikseen räätälöitävä ja rakennettava jokaista tietovarastoon integroitavaa järjestelmää varten.

Ison datan käsittelyyn tarkoitetussa kerroksessa datan tallentamiseen tulisi käyttää Hadoopin ja NoSQL:n kaltaisia teknologioita, jotka soveltuvat hyvin ison datan

tallentamiseen (Krishnan [2013], Pokorný [2014]). Perinteisemmän datan tallentamiseen esim. ERP (Enterprise Resource Planning) ja CRM (Customer Relations Management) -järjestelmistä voidaan käyttää relaatiotietokantoja, jotka soveltuvat hyvin tähän tarkoitukseen.

Arkkitehtuurin etuna on, että datan kerääminen, esikäsitteleminen, tallentaminen ja analysointi voidaan tehdä eri kerroksissa niillä teknologioilla ja tekniikoilla, jotka sopivat parhaiten tehtävään. Hadoop- ja NoSQL -teknologiat mahdollistavat ison datan tehokkaan tallentamisen ja käsittelyn, koska mm. dokumentit voidaan monistaa ja tallentaa useisiin paikkoihin ilman, että joudutaan investoimaan redundantteihin relaatiotietokantoihin ja niiden integroimiseen. Toisaalta relaatiotietokannat mahdollistavat tehokkaan raportoinnin ja analysoimisen perinteisemmälle transaktiodatalle, kuten kohdassa 1.3 on esitetty. Näiden etujen ansiosta arkkitehtuuri on joustava ja skaalautuu hyvin ja kustannustehokkaasti erilaisille dataformaateille myös tallennetun datan määrän kasvaessa ajan mittaan.

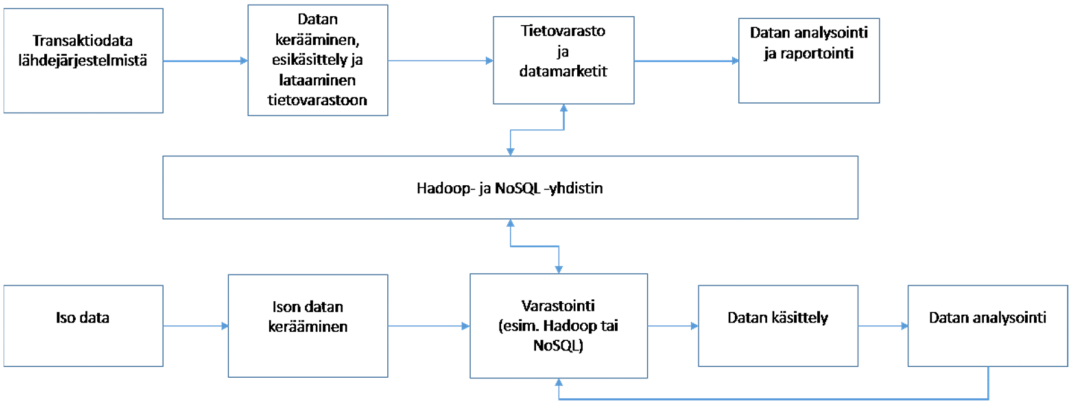
Arkkitehtuurin heikkous on datan integrointiin käytettävän kerroksen mahdollinen muodostuminen monimutkaiseksi. Lisäksi datan integrointi voi muodostua pullonkaulaksi ajan mittaan, kun tietovarastoon tallennettavien dataformaattien määrä ja tallennetun datan määrä kasvavat. Integrointiin käytettävän kerroksen onnistuneen toteutuksen kannalta on myös oleellista, että metadatan arkkitehtuuri on organisaatiossa hyvä ja selkeä. Näistä syistä johtuen ulkoinen dataintegraatioarkkitehtuuri soveltuu Krishnanin [2013] mukaan parhaiten organisaatioille, joissa data pysyy vakaana ajan mittaan, kuten antureista luettu data. Teollisessa ympäristössä koneiden ja laitteiden ohjelmistot, fyysinen toteutus ja tiedonsiirtostandardit asettavat rajoitteita dataformaateille ja niiden muutoksille, vaikka tuotettavan datan määrä voikin olla suuri ja tallennettavia ja analyyseissa käytettäviä dataformaatteja voi olla useita. Siksi ulkoinen dataintegraatio on teollisen internetin aikana soveltuva arkkitehtuuri yrityksen tietovaraston toteuttamiseksi.

3.4.2 Integraatiolähtöinen arkkitehtuuri

Integraatiolähtöisessä arkkitehtuurissa relaatiotietokantaan perustuva perinteinen tietovarastokerros liitetään ison datan käsittelykerrokseen yhdistimellä (connector), jonka kautta tietoa voidaan siirtää kerrosten välillä (kuva 7). Kuvassa 7 nuolet esittävät datan liikkumista arkkitehtuurin komponenttien välillä. Yhdistin on verrattavissa toiminnaltaan JDBC:hen (Java Database Connectivity), joka on Java-ohjelmointikielen rajapinta, joka määrittää standardisoidun tavan sovellukselle käyttää tietokantaa.

Integraatiolähtöisen arkkitehtuurin etuja on, että se skaalautuu hyvin eikä ole yhtä riippuvainen selkeästä ja hyvästä metadata-arkkitehtuurista kuin ulkoinen

dataintegraatioarkkitehtuuri. Integraatiolähtöisen arkkitehtuurin heikkous on yhdistin, koska datan siirtämisessä kaistanleveys (bandwidth) voi muodostua pullonkaulaksi.



Kuva 7: Integraatiolähtöinen arkkitehtuuri [Krishnan 2013].

4. Yrityksen tietovarasto ja holistinen näkemys yrityksessä

Seuraavaksi esitellään mitä tarkoitetaan holistisella näkemyksellä yrityksessä ja LEAN, joka on esimerkiksi tuotannollisissa yrityksissä laajasti käytetty kehittämisfilosofia. Lisäksi käsitellään suosittelujärjestelmiä ja tapoja, joilla tietovarastoon yhdistetty suosittelujärjestelmä voi edesauttaa holistisen näkemyksen muodostumista yrityksessä.

4.1 Holistinen näkemys yrityksessä

Hopp ja Spearman [2000] tarkoittavat holistisella näkemyksellä valmistavassa teollisuudessa järjestelmäorientoitunutta näkemystä. Järjestelmäorientoitunut näkemys tarkoittaa, että tuotantojärjestelmän osia ja prosesseja tarkasteltaessa painotetaan niiden vuorovaikutusta tuotantojärjestelmän muiden osien kanssa ja niiden vaikutusta järjestelmän tavoitteiden saavuttamiseksi. Toisaalta modernin globaalin kilpailun ympäristössä päätöksentekijät tarvitsevat yksityiskohtaista, myös teknisiä yksityiskohtia sisältävää, ymmärrystä liiketoiminnastaan organisaation menestymiseksi.

Zinkin [1998] mukaan modernissa liiketoimintaympäristössä tarvitaan holistista, eli kokonaisvaltaista johtamista yrityksen menestymiseksi. Zink esittää, että tällaisen integroivan ja systemaattisen lähestymistavan tunnusmerkkejä ovat:

- yritykset, jotka ovat ympäristöönsä tiukasti sidoksissa olevia avoimia järjestelmiä,
- lähestymistapa perustuu analyyttiseen ja synteettiseen ajatteluun, ja tunnistaa verkostojen rakenteen merkityksen,
- järjestelmäorientoitunut ajattelutapa,
- osastojen ja funktionaalisuuksien rajat ylittävä ajattelu ja yhteistyö ja
- prosessit ja rakenteet, joiden avulla selvitään kasvavasta informaation tärkeydestä.

Sekä Hopp ja Spearman [2000] ja Zink [1998] siis esittävät, että modernissa teollisessa ympäristössä yritysten tulee pystyä toisaalta hyödyntämään yrityksen sisältä ja ulkoa tulevaa dataa menestykseen tarvittavan ymmärryksen muodostamiseksi ja toisaalta pystyttävä hahmottamaan kokonaisvaltaisesti asioiden suhteet poikkifunktionaalisesti ja -osastollisesti. Epäonnistuminen näissä tavoitteissa voi johtaa Krishnanin [2013] esittämiin menetettyihin liiketoimintamahdollisuuksiin.

Eräs tunnettu holistinen lähestymistapa yrityksen toiminnan kehittämiseen ja tehostamiseen on LEAN. Slackin ja muiden [2004] mukaan LEAN on Toyotan tuotantojärjestelmästä alkunsa saanut kehittämisfilosofia, jonka ydin on hukan vähentämisessä kaikessa toiminnassa, koko henkilöstön ja kaikkien prosessien mukaanottamisessa kehitystoimintaan ja jatkuvassa parantamisessa. Hukan vähentäminen tarkoittaa, että kaiken toiminnan tulisi tuottaa lisäarvoa asiakkaalle. Koko henkilöstön ja kaikkien prosessien mukaan ottaminen merkitsee, että jokaisen henkilön ja prosessin on osallistuttava systemaattisesti hukan poistamiseen ja jatkuvaan kehittämiseen. Jatkuvalla kehittämisellä tarkoitetaan, että vaikka prosesseja ei tällä hetkellä saataisi vastaamaan organisaation asettamia tavoitteita asioiden ideaalitilasta, niin silti organisaation tulisi koko aika pyrkiä askel kerrallaan saavuttamaan tämä ideaalitila. LEAN-filosofian mukaan tärkeämpää kuin inkrementaalisten kehitysaskeleiden ottamistahti on systemaattinen tapa etsiä jatkuvasti keinoja parantaa toimintaa [Slack *et al*. 2004]. Eräs työkalu tällaiseen iteratiiviseen kehittämiseen on DMAIC (define, measure, analyze, improve, control) [Singh and Khanduja 2015]. DMAIC:ssä tunnistetaan ongelma, määritellään vaatimukset, asetetaan tavoitteet, mitataan oleellisia parametreja tai muuten kerätään dataa parannettavasta prosessista, analysoidaan mittaustuloksia, suunnitellaan ja toteutetaan parannusohjelma ja lopuksi varmistetaan ongelman poistuminen ja sekä parantuneen tilanteen säilyminen.

Coplien ja Bjørnvig [2010] tiivistävät LEANin ydinajatuksen lauseella kaikki mukana, kaikki yhdessä ja aikaisesta vaiheesta alkaen. Tällä Coplien ja Bjørnvig tarkoittavat, että kaikkien kehitysprojektiin tai sen tuloksiin liittyvien tahojen tulisi osallistua kehitysprojektiin, tai ainakin olla tietoisia siitä ja työskennellä yhdessä mahdollisimman aikaisesta vaiheesta alkaen. Tällöin aikaansaadaan holistinen näkemys organisaatiossa ja voidaan välttää liiketoimintamahdollisuuksien hukkaamista ja osaoptimointia, kun kehitysprojekteja kommentoidaan eri sidosryhmien näkökulmista.

4.2 Suosittelujärjestelmät

Ricci ja muut [2011] esittävät, että suosittelujärjestelmien tehtävä on ehdottaa käyttäjälle kohteita, joista käyttäjä voisi olla kiinnostunut. Pystyäkseen suosittelemaan kohteita järjestelmän täytyy pystyä arvioimaan mitkä kohteet käyttäjää kiinnostaisivat. Vaihtoehtoisia menetelmiä arvioida kohteen kiinnostavuutta tietylle käyttäjälle ovat [Ricci *et al.* 2011]:

- Kokoava suodatus (collaborative filtering) ja
- Sisältöön perustuva suodatus (content based filtering).

4.2.1 Kokoava suodatus

Kokoava suodatus tarkoittaa, että käyttäjälle suositellaan kohteita sen mukaan, mistä ovat kiinnostuneet käyttäjät, jotka ovat aikaisemman historian perusteella olleet kiinnostuneita samanlaisissa kohteista kuin suosittelun kohteena oleva käyttäjä. Käytännössä vertailuun voidaan käyttää läheisyysperustaisia (neighborhood-based) menetelmiä, kuten esimerkiksi k:n lähimmän naapurin (k-nearest neighbors) -algoritmia. Kokoava suodatus ei ole teollisessa ympäristössä hyvä lähestymistapa, koska työntekijöille suositeltu aineisto ei voi riippua siitä, että joku vastaavassa työtehtävässä työskentelevä henkilö on sattunut tutustumaan materiaaliin ja pitänyt sitä oleellisena työtehtävän kannalta. [Ricci *et al*. 2011]

4.2.2 Sisältöön perustuva suodatus

Lops ja muut [2011] käsittelevät sisältöön perustuvaa suodatusta tarkemmin. Heidän mukaansa sisältöön perustuvassa suodatuksessa aineistoa suositellaan käyttäjälle sillä perusteella, että aineisto muistuttaa sisältönsä perusteella aineistoa, jota käyttäjä on aikaisemmin pitänyt kiinnostavana. Tässä yhteydessä arkkitehtuurissa keskeiset komponentit ovat sisällönanalysoija (content analyzer), profiilinoppija (profile learner) ja suodatuskomponentti (filtering component).

Sisällönanalysoija analysoi kohteen ja esittää sen piirteet, kuten avainsanat, esimerkiksi vektorina. Menettely on yleensä samanlainen kuin tiedonhakujärjestelmissä (information retrieval systems).

Profiilinoppija on komponentti, joka käyttää ohjattuja oppimisalgoritmeja (supervised learning algorithms) muodostaakseen käyttäjän kiinnostuksenkohteita kuvaavan käyttäjäprofiilin. Algoritmin opettamiseen voidaan käyttää aineistoa, jonka käyttäjä on arvioinut itseään kiinnostavaksi tai käyttäjän antamaa eksplisiittistä kuvausta häntä kiinnostavista aineistoista. Usein suosittelujärjestelmissä on palautemekanismi, jonka avulla käyttäjä voi arvioida kuinka hyvin ehdotettu aineisto vastaa hänen tarpeitaan ja kiinnostuksenkohteitaan. Käyttäjän antaman palautteen perusteella profiilinoppija voi parantaa käyttäjän profiilia kuvaamaan käyttäjän kiinnostuksenkohteita entistä paremmin. Käyttäjän kiinnostuksenkohteet voivat myös muuttua ajan mittaan, jolloin palautejärjestelmä on välttämätön, että järjestelmä pystyisi vastaamaan käyttäjän kiinnostuksenkohteiden dynaamiseen luonteeseen.

Suodatuskomponentti käyttää jotain strategiaa arvioidakseen aineiston kohteiden kuvauksen vastaavuutta käyttäjän profiiliin. Vastaavuuden mukaan kohteet listataan sen mukaan, kuinka todennäköisesti aineiston kunkin kohteen arvioidaan kiinnostavan käyttäjää. Jos aineiston kohteiden piirteet ja käyttäjän profiili on esitetty vektoreina, niin eräs keino arvioida kohteen kiinnostavuutta käyttäjälle on laskea vektoreiden kosinivirhe. Mitä pienempi vektoreiden välinen kosinikulma on, niin sitä todennäköisemmin aineiston

kohde on käyttäjää kiinnostava ja sitä pienemmän järjestysluvun se saa listassa käyttäjälle suositeltavista kohteista.

Teollisessa yrityksessä sisältöön perustuva suodatus on mahdollinen tapa toteuttaa suosittelujärjestelmä, joka tukee holistisen näkemyksen muodostumista organisaatiossa. Suosittelujärjestelmän opetus voidaan tehdä joko työntekijäkohtaisesti tai profiilipohjaisesti. Työntekijäkohtaisessa opetuksessa työntekijä voi kerätä esimerkkikokoelman erilaisista raporteista ja muista dokumenteista, jotka kokee työnsä kannalta oleellisiksi ja käyttäjän profiili muodostetaan tämän kokoelman perusteella. Toinen vaihtoehto on, että työntekijä voi listata esimerkiksi avainsanoja, joita sisältävät dokumentit hän uskoo kokevansa hyödylliseksi. Profiilipohjaisesti opetus voidaan tehdä niin, että samaa työtä tekevät henkilöt keräävät kokoelman hyödyllisiä dokumentteja järjestelmän opettamista varten tai listaavat esimerkiksi avainsanoja, joita sisältävien dokumenttien he uskovat olevan hyödyllisiä työtehtävänsä kannalta. Työntekijälle tai saman profiilin omaaville käyttäjille voidaan tarjota mahdollisuus antaa palautetta suositelluista dokumenteista esimerkiksi arvioimalla asteikolla 1-5, kuinka hyödyllisiä suositellut dokumentit ovat olleet. Palautetiedon avulla palautejärjestelmän relevanttiuden arvioivaa suodatuskomponenttia voidaan opettaa entistä paremmaksi.

Sisältöön perustuvassa suosittelujärjestelmässä on kolme perustavanlaatuista heikkoutta [Lops *et al.* 2011] :

1. Suosittelujärjestelmä ei pysty erottelemaan käyttäjää kiinnostavia kohteita käyttäjää kiinnostamattomista, jos sisällönanalysoijan kohteesta tunnistamat piirteet eivät ole riittäviä erottelemiseksi. Sisällönanalysoijaa toteutettaessa usein tarvitaan paljon tietoa sovellusalueen kohteista (domain knowledge), jotta oleelliset kohdetta kuvaavat piirteet tulevat tunnistetuksi.

2. Järjestelmät ovat herkästi ylierikoistuneita, eli ne eivät osaa suositella käyttäjälle yllättäviä kohteita, joista käyttäjä kuitenkin voisi olla kiinnostunut. Sisältöperusteinen suosittelujärjestelmä suosittelee kohteita, jotka ovat suodatuskomponentin perusteella eniten käyttäjän profiilia vastaavia ja siksi käyttäjälle suositeltavat kohteet ovat keskenään hyvin samankaltaisia.

3. Uuden käyttäjän profiilia luotaessa tulee olla käytettävissä kyllin laaja aineisto tai kyllin laaja eksplisiittinen kuvaus käyttäjää kiinnostavista piirteistä, jotta suosittelujärjestelmä toimisi riittävällä tarkkuudella profiilinoppijan analysoitua aineiston tai kuvauksen.

Valmistavan teollisuuden kontekstissa ongelmat 1 ja 3 ovat kohtuullisen helposti hallittavissa, koska erilaisia analysoitavia aineistotyyppejä on hyvin rajallinen määrä verrattuna esimerkiksi verkkokauppaan, eivätkä aineistotyypit muutu yhtä nopeasti ja ennustamattomasti kuin verkkokaupassa. Teollisessa yrityksessä koneiden tuottama aineisto on hyvin vakiomuotoista ja ihmisten tuottamienkin aineistojen muotoa ja rakennetta voidaan kontrolloida esimerkiksi koulutuksen, vakioraporttipohjien ja toimintatapojen standardoinnin avulla. Uusia ja odottamattomia kohteita, joita pitäisi suositella käyttäjälle, ei pitäisi esiintyä usein, koska yritysten raportointi on yleensä varsin vakiintunutta ja standardoitua esimerkiksi laatujärjestelmien ohjeistuksen ansiosta. Esimerkiksi ISO9001 [2015] -laatujärjestelmä edellyttää, että organisaatio on tunnistanut liiketoiminnan kannalta tärkeät prosessit, kuvannut ne ja valvonut prosessien suorittamista kuvausten mukaisesti.

Teollisissa yrityksissä, joissa suurin osa käyttäjäprofiileista on ennemminkin työtehtäväkohtaisia kuin käyttäjäkohtaisia, on myös uuden käyttäjäprofiilin luominen helpompaa. Tyypillisesti yrityksessä samanlaista työtehtävää hoitaa monta ihmistä ja siksi profiilin laatimiseen osallistuu useita ihmisiä ja kattavan aineiston tai eksplisiittisen kuvauksen aikaansaaminen on helpompaa. Lisäksi kun useat ihmiset antavat järjestelmälle palautetta samalla profiililla on oletettavaa, että järjestelmä myös oppii nopeammin paremmaksi. Toisaalta ongelmaksi profiileja luotaessa voi muodostua rajaaminen kuinka monta erilaista profiilia tulee tehdä ja ketkä kaikki niihin kuuluvat. Esimerkiksi jos kaikille myyjille tehdään yksi profiili, niin kaikki voisivat saada suosituksia markkina-analyyseistä kaikilta maantieteellisiltä alueilta, vaikka vastuualueet olisikin jaettu esimerkiksi maittain. Eräs tapa välttää tämä ongelma voisi olla, että profiili tehdään työtehtäväkohtaisesti, mutta käyttäjälle annetaan mahdollisuus lisätä profiiliin suodatusehtoja ja toisaalta avainsanoja ja muita piirteitä, joista hän on kiinnostunut. Lisäksi palautetta kannattaisi kerätä paitsi profiili- myös käyttäjäkohtaisesti, jolloin järjestelmä voisi oppia ajan mittaan huomioimaan käyttäjäkohtaisia eroja.

Erääksi ongelmaksi suosittelujärjestelmien käytössä voi muodostua tietoturva. Organisaation kannalta on oleellista, että käyttäjät eivät voi itse vapaasti määrittää, mihin dataan he pääsevät käsiksi valitsemalla esimerkiksi sopivia avainsanoja käyttäjäprofiiliinsa, koska kaikkea dataa ei ole tarkoitettu kaikkien organisaation jäsenten saatavaksi. Tästä syystä yrityksen tietovaraston tulee suositella käyttäjälle ainoastaan sellaista dataa, jota käyttäjällä on oikeus saada haltuunsa.

4.3 Miten yrityksen tietovarasto voi tukea holistisen liiketoimintanäkemyksen muodostumista

Yrityksen tietovarastoon yhdistetty suosittelujärjestelmä voi tukea työtehtävien kannalta oleellisen tiedon tulemista työntekijöiden tietoon oikea-aikaisesti, eli Nagabushanan

[2006] esittelemän liiketoimintatarpeesta lähtöisin olevan informatiivisen tietotarpeen täyttymistä. Informatiivisen liiketoimintalähtöisen tietotarpeen täyttäminen auttaa yritystä välttämään Krishnanin [2013] esittämiä menetettyjä liiketoimintamahdollisuuksia.

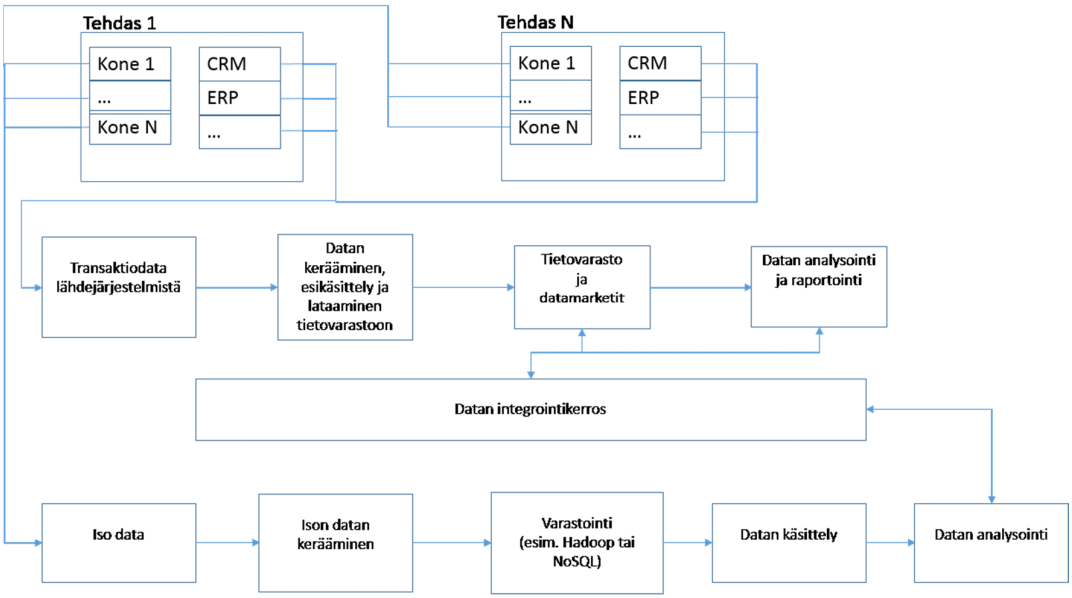
Kun yrityksessä tehdään kehitysprojekteja, strategisia päätöksiä tai esimerkiksi muutoksia toimintatavoissa, niin holistisen näkemyksen kannalta on oleellista, että tieto niistä välittyy oikea-aikaisesti niille työntekijöille, joiden työn kannalta tieto on tarpeellista. Isoissa organisaatioissa ja erityisesti, jos toiminta on hajautettu maantieteellisesti, voi työntekijän olla vaikea itsenäisesti tietää dokumenteista ja muusta datasta, joihin hänen olisi hyödyllistä tutustua. Toisaalta dataa tallentavan tai tallennuksesta ja tallennetusta datasta tiedottamisesta vastaavien työntekijöiden voi olla vaikea hahmottaa keille kaikille organisaatiossa tieto tulisi välittää. Tässä epäkohdassa suosittelujärjestelmä voi tukea organisaation toimintaa, koska datan tallennuksen yhteydessä voidaan keskittyä liittämään dataan suositteluun tarvittavaa metadataa ja käyttämään organisaation standardien mukaisia toimintatapoja suosittelujärjestelmän tehokkaan toiminnan edesauttamiseksi. Dataa tarvitseva työntekijä puolestaan voi keskittyä käyttäjäprofiilinsa jatkokehittämiseen antamalla tehdyistä suosituksista palautetta järjestelmälle sekä mahdollisesti suoraan eksplisiittistä profiilin kuvausta täydentämällä.

5. Tietovarasto osana teollisen yrityksen tietämyksenmuodostamisprosessia ison datan ympäristössä ja tukemassa holistisen näkemyksen muodostumista

Seuraavaksi esitellään, miten luvussa 3 esitelty tietovaraston arkkitehtuuri mahdollistaa datan tallentamisen, analysoimisen LEANin mukaisella jatkuvan kehittämisen filosofialla ison datan ympäristössä ja edistää holistisen näkemyksen muodostumista teollisessa yrityksessä suosittelujärjestelmän avulla.

5.1 Datan tallentaminen tietovaraston arkkitehtuurin kerroksiin

Datan tallentaminen tietovaraston arkkitehtuurin kerroksiin tarkoittaa, että aluksi on tunnistettava eri lähteistä tallennettavan datan luonne. Jos data on luonteeltaan transaktiodataa, joka ei täytä ison datan määritelmää ja jonka käsittelyyn perinteinen yrityksen tietovarasto soveltuu, niin datan tallentamiseen tietovarastoon kannattaa käyttää perinteisen tietovaraston arkkitehtuuria ja prosesseja. Kuvassa 8 on esitetty, kuinka eri tehtaiden tuottama transaktiodata ERP:n (tuotannonohjausjärjestelmä) ja CRM:n (asiakastiedon hallintajärjestelmä) kaltaisista järjestelmistä tallennetaan ulkoisen dataintegraatioarkkitehtuurin perinteisellä toteutusarkkitehtuurilla. Jos tallennettava data täyttää ison datan määritelmän, niin data kannattaa tallentaa tietovarastoon käyttäen luvussa 3 esitettyä ulkoisen dataintegraatioarkkitehtuurin ison datan käsittelyyn soveltuvaa alempaa kerrosta. Kuvassa 8 on lisäksi esitetty, kuinka eri tehtaissa sijaitsevista tuotantokoneista tallennetaan ison datan määritelmän täyttävää dataa tietovarastoon. Kuvassa 8 esitetyt tehtaat tai muut datalähteet voivat olla organisaation sisäisiä tai ulkoisia toimijoita. Ulkoisella toimijalla tarkoitetaan esimerkiksi alihankkijaa, joka kuuluu organisaation valmistusketjuun, mutta ei ole osa organisaatiota tai muita Nagabushananin [2006] virtuaalisen yrityksen sidosryhmiin kuuluvia toimijoita, jotka tuottavat dataa yrityksen tietovarastoon tai tarvitsevat sieltä dataa. Tällainen toimija vastaa myös HACE-teoreemassa mainittua heterogeenistä ja autonomista lähdettä.



Kuva 8: Datan tallentaminen tietovarastoon eri tehtaiden laitteista ja tietojärjestelmistä.

Formaatti, jossa data siirretään yrityksen tietovarastoon eri tehtaiden eri lähdejärjestelmistä ja koneista, voi riippua useista tekijöistä. Tällaisia tekijöitä ovat esimerkiksi formaatti, jossa data on tallennettu lähdejärjestelmään ja siirrettävän datan määrästä. Lau ja muut [2009] esittelivät artikkelissaan IQMS-järjestelmän (intelligent quality management system, älykäs laadunjohtamisjärjestelmä), jossa eri liiketoimintafunktioiden, kuten myynti- ja tuotanto-osastojen, yhteisesti tarvitsema data siirretään internetin välityksellä tietovarastoon tai tietovarastosta muihin järjestelmiin käyttäen XML-tiedostoja. Heidän esittämässään lähestymistavassa siirrettävä data on kvantitatiivista ja esitelty järjestelmä on erikoistunut laatuun vaikuttavien sääntöjen louhimiseen datasta assosiaatioanalyysin avulla.

Modernissa valmistavassa teollisuudessa kaikkea organisaation esittämää dataa, jota voidaan tarvita operatiivisessa toiminnassa ja toiminnan kehittämisessä, strategiatyöskentelyssä tai tiedottamisessa ei kannata esittää XML-muodossa. Esimerkiksi eräs tuotetiedon esittämiseen ja välittämiseen yleisesti käytettävä formaatti perustuu standardiin ISO 10303 [1994] ja tunnetaan kutsumanimellä STEP (Standard for the Exchange of Product model data). STEP-tiedostojen avulla voidaan välittää dataa eri tietokoneavusteisten suunnittelu-, valmistus- ja tiedonhallintajärjestelmien välillä. STEPin avulla asiakas voi välittää CAD-mallin (computer aided design, tietokoneavusteinen suunnittelu) valmistajalle, joka voi hyödyntää mallia CAM-ohjelmistossaan (computer aided manufacturing, tietokoneavusteinen valmistuksen suunnittelu) tuotteen valmistuksen suunnittelemiseksi. CAM-mallia on mahdollista hyödyntää tuotantolaitteiden ohjelmoinnissa kappaleen valmistamiseksi. Koska STEP-

tiedostojen avulla voidaan valmistaa tuotteita maantieteellisesti eri paikoissa sijaitsevissa tehtaissa, on STEP-tiedostoja pystyttävä tallentamaan yrityksen tietovarastoon. Lisäksi tietovarastoon tallennettuja STEP-tiedostoja tulisi tulevaisuudessa pystyä analysoimaan, koska tiedostot määrittelevät tuotteen piirteet ja ominaisuudet ja voisivat olla hyödyksi analysoitaessa esimerkiksi tuotepiirteiden vaikutusta laatuun tai valmistuskustannuksiin.

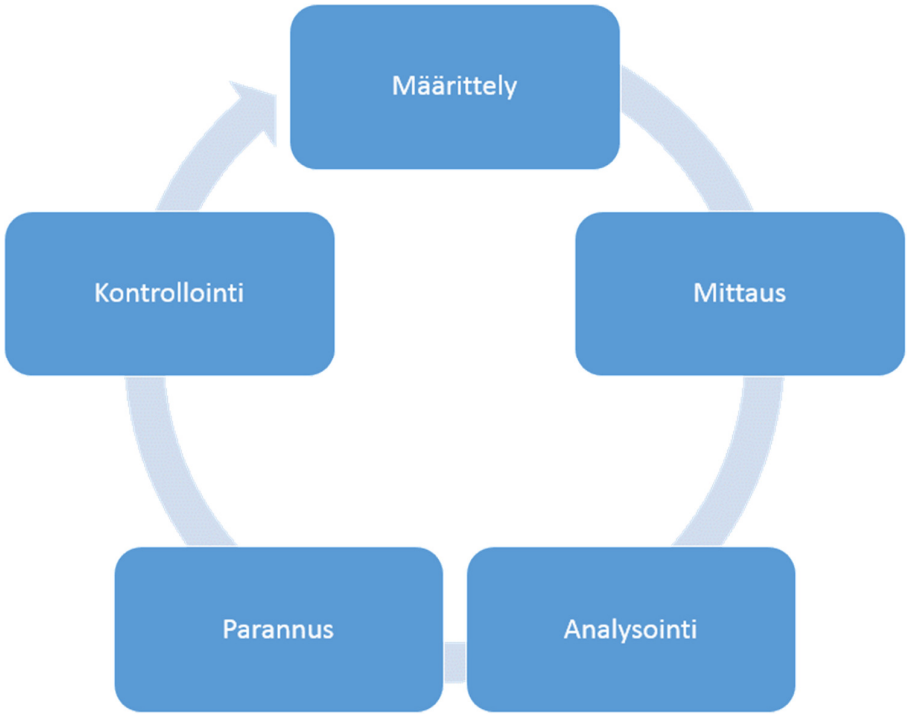
STEP-tiedostot ovat esimerkki ison datan määritelmän täyttävästä datasta. Tiedostot voivat olla kooltaan jopa kymmeniä tai satoja megabittejä. Jos tuotteiden valmistus edellyttää CAD- ja CAM-malleja, niin jokaisesta tuotteesta on oltava oma malli tallennettuna. Lisäksi tiedostoformaatti on monimutkainen verrattuna esimerkiksi XML:ään, ja siksi tarpeellisen tiedon louhiminen tiedostosta on monimutkaisempaa. STEP-tiedostot siis täyttävät ison datan määritelmän ja siksi ne tulisi tallettaa ison datan käsittelyyn tarkoitettuun kerrokseen tietovaraston arkkitehtuurissa. Hadoopin ja NoSQL:n avulla järjestelmä skaalautuu kustannustehokkaasti datan määrän ja nopeuden kasvaessa, koska tallennuskapasiteettia saadaan lisättyä lisäämällä halpoja tietokoneita klustereihin. Lisäksi STEP-tiedostoja voidaan monistaa ja tarvittaessa tiedostoja voidaan osittaa, jolloin tiedostojen sisältämää dataa voidaan analysoida tehokkaasti käytettävissä olevilla tietokoneklustereilla. Yhtä tiedostoa ei siis välttämättä tarvitse jäsentää yhdellä tietokoneella sen sisältämän datan analysoimista varten, vaan tiedosto voidaan osittaa useammalle koneelle, analysoida tiedoston eri osat ja lopuksi tallettaa tulokset tiedostoon yrityksen tietovarastoon.

Kun dataa on tallennettu datan luonteen perusteella ulkoisen dataintegraatioarkkitehtuurin kerroksiin, on kerrosten data mahdollista integroida datan integrointikerroksen avulla ja hyödyntää raportoinnissa ja datan jatkoanalysoinnissa. Seuraavassa luvussa esitellään DMAIC:hen perustuva lähestymistapa, jolla tietovaraston dataa voidaan käyttää systemaattisesti tietämyksen muodostamiseen organisaatiossa.

5.2 DMAIC:n mukainen prosessi tietämyksen muodostamiseksi

DMAIC (kuva 9) tarjoaa systemaattisen mallin, jonka avulla tietämyksenmuodostamisprosessia organisaatiossa voidaan tarkastella. Lau ja muut [2009] esittivät laadunkehityksen tukemiseksi assosiaatiosääntöjä louhimalla prosessia, joka muistuttaa monelta osin DMAIC-prosessia, mutta on yhteen käyttötarkoitukseen sovellettu erikoistapaus. Heidän mallissaan prosessin vaiheet ovat hajautetun prosessidatan kerääminen (collection of distributed process log), kategorisointi (categorize), tietämyksen muodostaminen (knowledge discovery) ja kehittäminen (improvement). Mallin ongelma yleisen kehittämisongelman kohdalla on, että malli ei sovellu kaikenlaisiin kehittämistehtäviin eikä esimerkiksi huomioi, että kenties kaikkea oleellista dataa ei ole kerätty tietovarastoon ja joukko assosiaatiosääntöjä jää siksi löytymättä. DMAIC on prosessimalli yleisen kehitysongelman pilkkomiseen hallittaviksi

ja toteutettaviksi vaiheiksi ja sitä on käytetty laajasti valmistavassa teollisuudessa hyvällä menestyksellä (esimerkiksi Singh ja Khanduja [2015]).



Kuva 9: DMAIC-iteraatio.

DMAIC -prosessin ensimmäisessä vaiheessa organisaatiolla on jokin ongelma tai tavoite, joka halutaan ratkaista tai saavuttaa. Aluksi ongelma tai tavoite kuvataan, jotta saadaan muodostettua yhteinen käsitys siitä, mikä ratkaistava ongelma on ja mitkä vaatimukset ratkaisun on täytettävä ideaalitilanteessa. Kun prosessin lopussa arvioidaan kehitettyä ratkaisua ja sen vaikutuksia, arvioinnin pohjana on tämä ensimmäisessä vaiheessa muodostettu määritelmä. Singhin ja Khandujan [2015] mukaan tämä vaihe on erittäin tärkeä, koska jos projektin tavoite ei ole selkeä, käy helposti niin, että mittausvaiheessa löydetään paljon ongelmakohtia. Tästä voi seurata, että projekti epäonnistuu, koska ei enää keskitytä alkuperäisen ongelman ratkaisemiseen.

Prosessin toisessa vaiheessa määritellään, onko kaikki tutkimukseen tarvittava data saatavilla tietovarastosta, vai täytyykö prosessissa etenemiseksi kerätä lisää dataa. Jos kaikkea tarvittavaa dataa ei ole saatavilla, tarvittavaa puuttuvaa dataa tulee kerätä ja se tulee integroida osaksi yrityksen tietovarastoa. Eräs metodi analysoida ongelman ratkaisuun liittyviä tekijöitä ja niihin liittyvää dataa on kalanruotokaavio (mm. Singh ja Khanduja [2015]). Kalanruotokaaviossa ongelma jaetaan siihen liittyviin tekijöihin ja

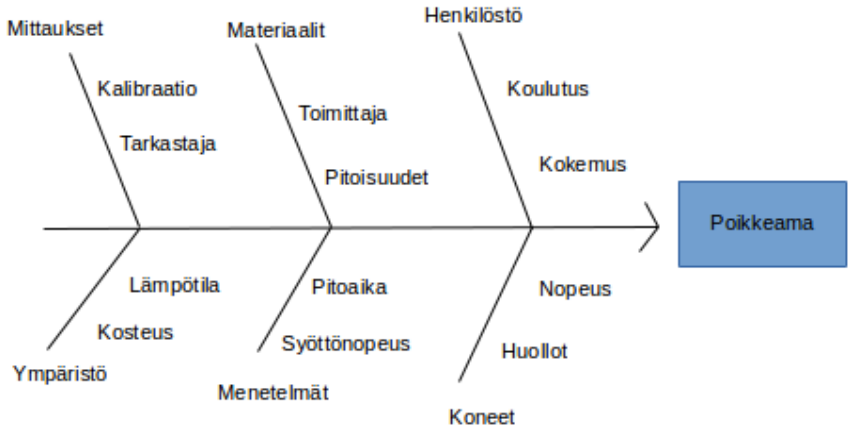
tekijöihin liittyvä data kuvataan kalanruotoa muistuttavalla kaaviolla (kuva 10). Kun ongelman analysoimisen kannalta tarpeelliseksi katsottu data on saatavilla tietovarastosta, voidaan edetä prosessin kolmanteen vaiheeseen eli datan analysoimiseen.

Datan analysointivaiheessa kerättyä dataa analysoidaan soveltuvalla analysointimenetelmällä. Erityisesti ison datan ympäristössä tiedonlouhintamenetelmät ovat usein hyödyllisiä. Esimerkiksi Lau ja muut [2009] ja Kamsu-Foguem ja muut [2013] ovat soveltaneet assosiaatiosääntöjen louhintaa laadunparannusprojekteihin valmistavassa teollisuudessa. Singh ja Khanduja [2015] puolestaan mainitsevat analysointivaiheessa usein käytetyiksi menetelmiksi esimerkiksi regressio- ja korrelaatioanalyysin, prosessin mallinnuksen ja simuloinnin sekä khiin neliö -testin (chi-square test). Analysointivaiheen tarkoitus on oppia ymmärtämään tarkasteltavaa prosessia entistä paremmin ja löytää ongelmien juurisyyt, joiden poistaminen parantaa prosessia. Tämän analyysin pohjalta kehitetään seuraavassa vaiheessa parannuskeinot.

Parannusvaiheessa aluksi suunnitellaan ja määritellään analyysien pohjalta kehitystoimenpiteet, jotka poistavat ongelman juurisyyt. Suunnittelun ja määrittelyn jälkeen suunnitelmat toteutetaan määritelmien mukaisesti tavoitteena ongelmien ratkaiseminen.

DMAIC-iteraation viimeisessä vaiheessa todetaan kehitysprojektin toimenpiteiden vaikutus ja tarvittaessa määritellään korjaavat toimenpiteet. Voidaan esimerkiksi aloittaa uusi iteraatio ja toteuttaa tarvittava seuranta sekä raportointi prosessin valvomiseksi jatkossa. Yrityksen tietovaraston kannalta tämä tarkoittaa, että tietovaraston tiedon haku- ja käyttökerrokseen toteutetaan tarvittavat raportointi- ja analysointityökalut. Usein tarvittavat analysointi- ja raportointityökalut on suunniteltu ja toteutettu jo iteraation analysointivaiheessa ja ne täytyy vain ottaa käyttöön ja vakiinnuttaa osaksi organisaation normaaleja analysointi- ja raportointikäytäntöjä. Lisäksi DMAIC-iteraatiolla aikaansaadut tulokset ja tehdyt kehitystoimenpiteet niiden aikaansaamiseksi tulee dokumentoida, jotta tieto on tarpeellisten sidosryhmien saatavilla.

Lisäksi kehitysprojektin päätyttyä on oleellista, että projektin tulokset välitetään kaikkien oleellisten sidosryhmien tietoon, eli täytetään informatiivinen tietotarve. Ilman kehitysprojektien tuloksista tiedottamista voidaan menettää liiketoimintamahdollisuuksia. Seuraavassa kohdassa esitellään, miten yrityksen tietovarastoon integroitu suosittelujärjestelmä voi tukea tällaisen holistisen näkemyksen muodostumista.



Kuva 10: Kalanruotokaavio.

5.3 Holistisen näkemyksen muodostumisen tukeminen organisaatiossa tietovarastoon integroidun suosittelujärjestelmän avulla

Kun tietovarastoon tallennetaan dataa, on oleellista, että tieto datasta tulee oikeiden henkilöiden tietoisuuteen, jotta organisaatiossa muodostuisi yhtenäinen näkemys, jonka mukaan toimitaan. Tällaista dataa, josta tulisi tiedottaa, voi olla esimerkiksi edellisessä luvussa esitellyn prosessin tuloksena muodostunut raportti, organisaation strategiatyön tuloksena muodostettu strategia tai vaikkapa asiakkailta saatu palaute. Modernissa verkostoituneessa toimintaympäristössä oleellisen datan suosittelu käyttäjille on erityisen tärkeää, jotta organisaatiossa kehitetyt parhaat toimintatavat saadaan otettua käyttöön eri toimipisteissä vaikka ne sijaitsisivatkin fyysisesti kaukana toisistaan. Bhutta ja muut [1999] esittävät artikkelissaan, kuinka esimerkiksi öljynjalostamot eivät keskustele tarpeeksi keskenään eivätkä parhaat käytännöt siksi välity käyttöön, jolloin sisäinen benchmarking ei toimi organisaatiossa parhaalla mahdollisella tavalla. Lisäksi he esittävät, että samassakin organisaatiossa eri osastoilla tai eri toimipisteissä työskentelevät työntekijät saattavat jättää tarkoituksellakin kertomatta löydetyistä parhaista toimintatavoista.

Luvussa 4 esitellysti suosittelujärjestelmän toiminta voi perustua datan sisällön analysoimiseen. Edellytys suosittelujärjestelmän toimimiselle organisaatiota hyödyttävällä tavalla on, että suosittelujärjestelmää käyttäville henkilöille on luotu työtehtävän tarpeisiin perustuva profiili. Luvussa 4 esitetyn lisäksi on hyödyllistä, että käyttäjä voi muokata omaa profiiliaan esimerkiksi antamalla järjestelmälle palautetta saamiensa suositusten hyödyllisyydestä itselleen. Palautteen antamisen avulla käyttäjä

voi muokata suosittelujärjestelmää vastaamaan entistä paremmin oman työtehtävänsä asettamia informatiivisia tarpeita.

Suosittelujärjestelmän toimintaa osana yrityksen tietovarastoa voidaan tarkastella Kamsu-Foguemin ja muiden [2013] artikkelin esittelemän laadunkehitysprojektin kontekstissa. He käyttivät tutkimuksessa assosiaatiosääntöjen louhintaa parantaakseen tuotantoprosessin laatua. Projektin lopputuloksena löydettiin useita sääntöjä, jotka edesauttoivat löytämään ja poistamaan juurisyitä, jotka olivat johtaneet poikkeamiin ja tuotantoajan menetyksiin. Artikkelissa ei kiinnitetty huomiota muun organisaation informoimiseen tehdyistä kehitystoimenpiteistä ja niiden vaikutukseen organisaation toimintaan.

Sisältöön perustuva suosittelujärjestelmä olisi voinut tukea edellä kuvatun kehitysprojektin tuloksien raportoinnin jälkeen organisaation holistisen näkemyksen muodostumista. Koska raportissa mainittiin kehitystoimenpiteiden vaikutus tuotannon häiriöiden takia menetettyyn tuotantoaikaan, olisi suosittelujärjestelmä voinut suositella raporttia organisaation tuotannonsuunnittelusta vastaaville henkilöille. On tyypillistä, että tuotannonsuunnittelussa laskennallinen kapasiteetti perustuu prosessin aikaisemman suorituskyvyn mukaan kehitettyyn malliin [Hopp and Spearmann 2000]. Jos prosessin suorituskyky muuttuu, on oleellista mukauttaa malleja, jotta lisääntynyt kapasiteetti pystytään hyödyntämään ja samalla tuotantokustannukset alenevat. Tuotannonsuunnittelun raportoidessa lisääntyneestä kapasiteetista suosittelujärjestelmä olisi voinut suositella raporttia organisaation laskentatoimelle, jonka tehtäviin kuuluu määrittää kustannus, joka tuotteen valmistamisesta seuraa. Lopulta suosittelujärjestelmä olisi voinut suositella myyjille dataa alentuneista tuotantokustannuksista, jotta myynti pystyisi viestimään asiakkaille alentuneista kustannuksista ja mahdollisesti saamaan lisää kauppoja. Lisäksi suosittelujärjestelmä olisi voinut välittää tiedon keinoista välttää häiriöitä organisaation muihin toimipisteisiin, joissa on käytössä samanlainen tuotantoprosessi. Ilman suosittelujärjestelmää tiedon välittyminen löydetyistä parhaista toimintatavoista on pahimmillaan sen varassa, että yksittäiset työntekijät tietävät muissa yksiköissä olevan käytössä samanlaisia prosesseja ja osaavat, muistavat ja haluavat välittää tiedon oikeille henkilöille.

6. Johtopäätökset

Tässä luvussa esitellään aiempiin lukuihin perustuen integroitu malli, jossa yrityksen tietovarasto on toteutettu ison datan tallentamiseen ja analysointiin soveltuvalla arkkitehtuurilla. Siihen on integroitu DMAIC:n mukainen jatkuvan kehittämisen prosessi datan analysoimiseksi ja hyödyntämiseksi organisaation prosessien kehittämisessä sekä sisältöön perustuva suosittelujärjestelmä, joka edesauttaa holistisen näkemyksen muodostumista organisaatiossa ja parhaiden käytäntöjen käyttöönottamista organisaation kaikissa osissa.

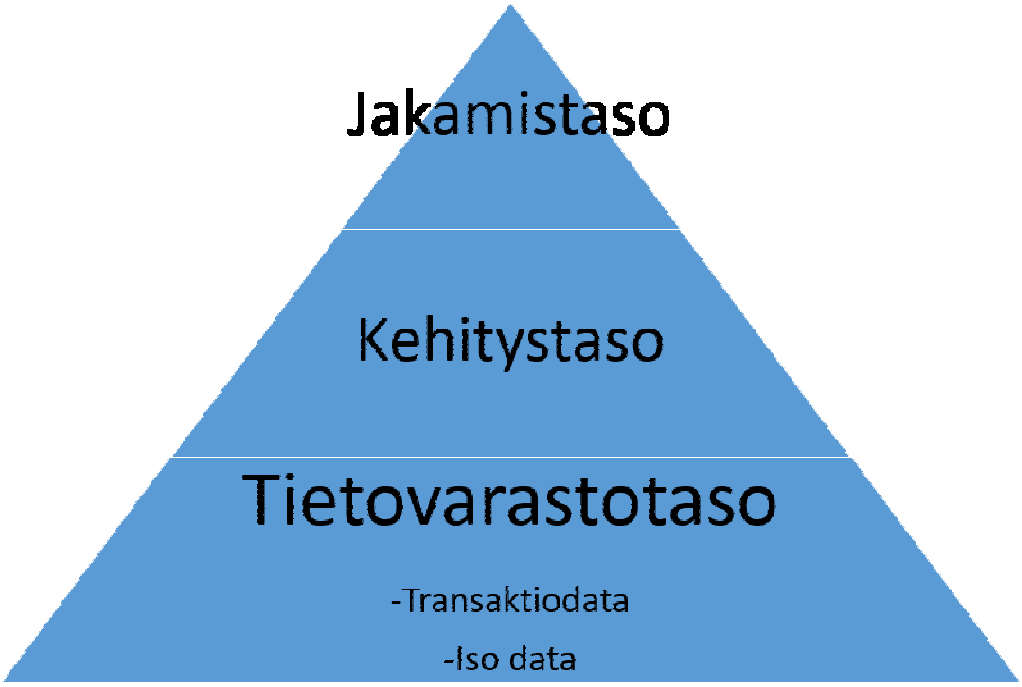
6.1 Holistinen malli tietovarastolle, sitä hyödyntävälle kehitysprosessille ja suosittelujärjestelmälle ison datan ympäristössä

Holistinen malli voidaan ajatella pyramidina, jossa on kolme tasoa (kuva 11) :

- Tietovarastotaso, joka on kerroksellinen. Data tallennetaan tietovaraston siihen kerrokseen, johon data luonteensa puolesta kuuluu.

- Kehitystaso, jossa dataa hyödynnetään ymmärryksen lisäämiseksi prosessista tai havaitun ongelman ratkaisemiseksi.

- Jakamistaso, jossa tietovarastoon tallennettuja raportteja ja muuta dataa suositellaan sen sisällön perusteella organisaation jäsenille, jotta organisaatio ei menettäisi liiketoimintamahdollisuuksia ja jotta parhaat menettelyt tulisivat kaikkialla organisaatiossa käyttöön (sisäinen benchmarking).



Kuva 11: Holistinen malli tietovarastolle, datan hyödyntämiselle kehitystoiminnoissa ja tiedon jakamiselle organisaatiossa.

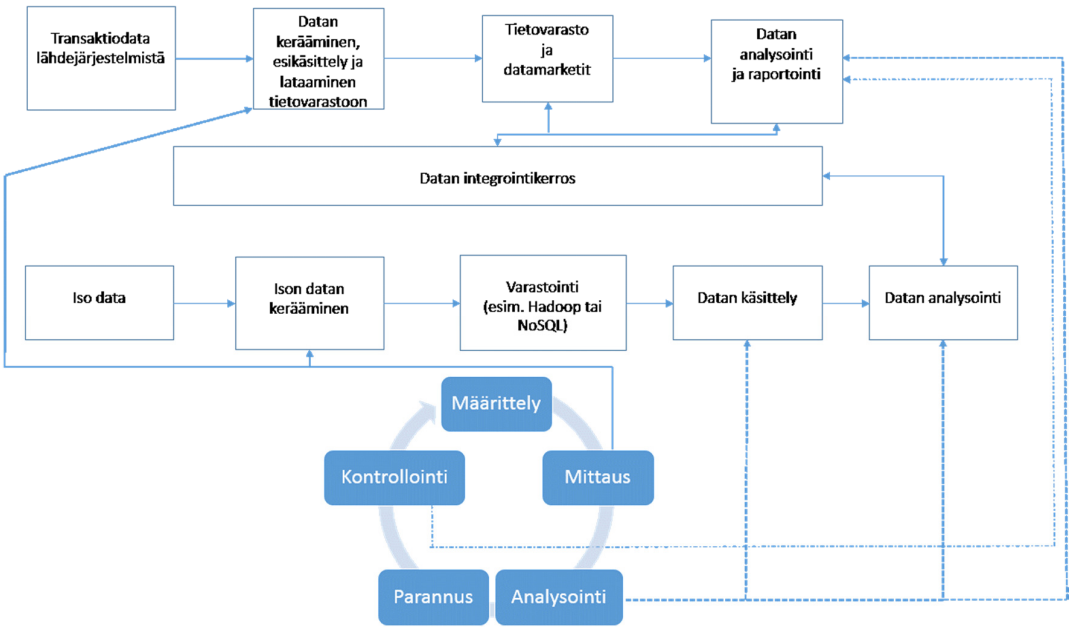
6.1.1 Tietovarastotaso

Pyramidin pohjalla tietovarastotasolla yrityksen tietovarasto on toteutettu luvussa 3 esitellyn kerroksellisen arkkitehtuurin periaatteella, niin että tietovarasto kykenee skaalautumaan kustannustehokkaasti datan volyymin kasvaessa, datan syntymisnopeuden ollessa suuri ja moninaisten tallennusformaattien ollessa käytössä. Lisäksi arkkitehtuuri kykenee aikaisemmissa luvuissa esitellysti tukemaan ison datan tehokasta analysoimista.

Tietovarastotaso luo pohjan pyramidin ylemmille tasoille, koska kehitystason toiminnan edellytyksenä on, että ymmärryksen syventämiseksi ja ongelmien ratkaisemiseksi tarvittavan datan on oltava saatavilla ja yhdistettävissä muuhun relevanttiin dataan sekä analysoitavissa esimerkiksi koneoppimista hyödyntäen. Ison datan ympäristössä aineiston ollessa iso on todennäköistä, että analysoinnissa on tarpeen hyödyntää hajautettua analysointia. Toisaalta myös tietovarastoon tallennetun aineiston sisältöperusteisen suosittelun edellytys on, että aineisto on suosittelujärjestelmän saatavilla ja analysoitavissa vaikka järjestelmän skaalautuvuus edellyttäisi hajautetun analysoimisen käyttöä esimerkiksi Hadoopia hyödyntäen.

6.1.2 Kehitystaso

Mallin kehitystasossa yrityksen tietovarastoon tallennettua dataa analysoidaan aiemmin esitellyllä DMAIC-prosessilla. DMAIC-prosessin vahvuus on, että tavoitteiden asettelussa sitoudutaan tietyn ongelman ratkaisemiseen, jolloin dataa analysoitaessa ja ongelmaa ratkaistaessa ei ajauduta helposti sivuraiteille. Lisäksi prosessi on iteratiivinen ja huomioi, että tietovarastossa ei välttämättä ole vielä tallennettuna kaikkea ongelman ratkaisuun tarvittavaa dataa, vaan sitä voidaan joutua keräämään ennen analysointivaihetta. Lisäksi jos prosessin tulos ei täytä määriteltyjä tavoitteita, prosessi voidaan iteratiivisesti aloittaa alusta. Prosessin analysointivaiheessa tietovaraston tulee aikaisemmissa luvuissa esitellysti ison datan ympäristössä tukea hajautetun laskennan ja tiedonlouhintamenetelmien käyttämistä (mm. Wu ja muut [2014]) datan analysoimiseksi. DMAIC-prosessin viimeinen vaihe kontrollointi tarkoittaa, että ratkaistun ongelman toistuminen tulevaisuudessa pyritään estämään. Käytännössä se voi tarkoittaa esimerkiksi sitä, että tietovarastoon lisätään automaattinen raportointi, josta selviää onko ongelma palaamassa. Tällaisesta raportoinnista ja analysoinnista on hyötyä ainoastaan, jos raportti tulee oikeiden henkilöiden tietoon oikea-aikaisesti. Kehitystason vuorovaikutus yrityksen tietovaraston kanssa on esitetty kuvassa 12.

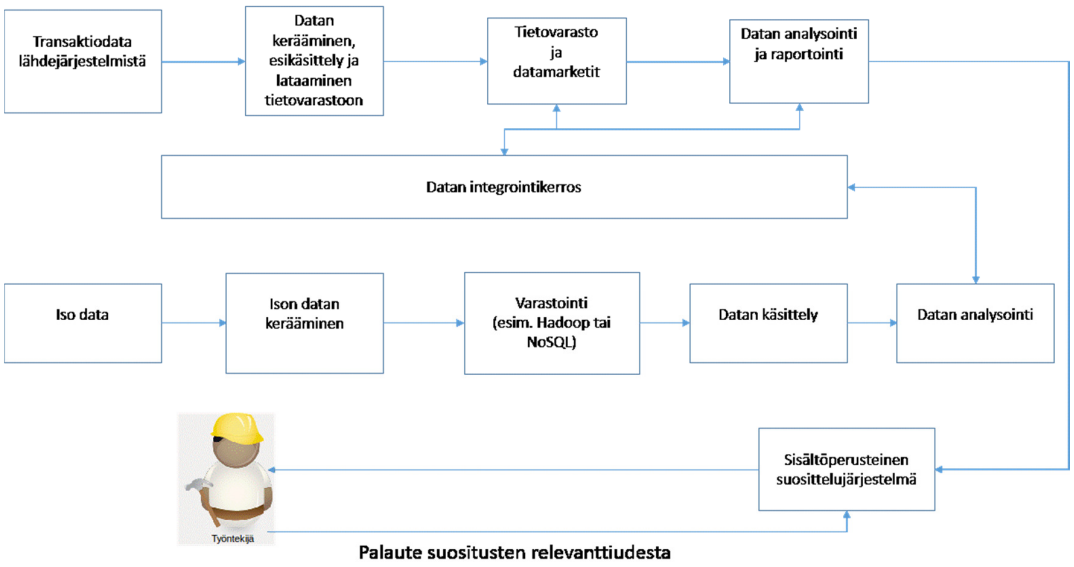


Kuva 12: Kehitystason vuorovaikutus yrityksen tietovaraston kanssa.

6.1.3 Jakamistaso

Jakamistason tarkoitus on edesauttaa tiedon kulkeutumista oikeille henkilöille oikeaan aikaan, jotta organisaatio ei menettäisi liiketoimintamahdollisuuksia. Lisäksi jakamistaso edistää parhaiden käytäntöjen käyttöönottamista kaikkialla organisaatiossa ja myös yhteistyökumppaneilla silloin, kun se tukee virtuaalisen yrityksen tavoitteiden toteutumista loukkaamatta sopimuksia ja kenenkään immateriaalioikeuksia. Jos esimerkiksi analysoimisen seurauksena organisaation ymmärrys lisääntyy tai löydetään parempia toimintatapoja, on oleellista, että tieto välittyy organisaation henkilöille, joille datasta on hyötyä [Bhutta *et al*. 1999]. Tietoa voi olla joissain tapauksissa tarpeen välittää myös yhteistyökumppanien työntekijöille.

Esitetyssä holistisessa mallissa tiedon välitys perustuu aikaisemmissa luvuissa esitetysti sisältöperusteiseen suosittelujärjestelmään (kuva 13), jossa työtehtäville luodaan rooliperusteinen profiili, jota voidaan myös muokata työntekijäkohtaisesti työntekijän antaman palautteen tai formaalin määrittelyn perusteella. Työtehtäväprofiili voi liittyä myös yhteistyökumppaniin, jolle halutaan välittää tietynlaista dataa yrityksen tietovarastosta. Tällöin tietoturvan ja immateriaalioikeuksien suojelemiseksi edellytyksenä on, että datan sisällön perusteella voidaan luotettavasti tunnistaa data, jota voidaan suositella yhteistyökumppanille. Eräs keino on liittää dataan metatietoa tai lisätä aineiston sisältöön aineiston julkisuudesta ja jakelusta maininta, jota suosittelujärjestelmä voi hyödyntää.



Kuva 13: Jakamistason suosittelujärjestelmän vuorovaikutus yrityksen tietovaraston ja työntekijöiden kesken.

7. Yhteenveto

Seuraavaksi kootaan yhteen tämän opinnäytetyön havainnot ja johtopäätökset sekä esitellään yhteenveto opinnäytetyön tuloksena kehitetystä integroidusta mallista. Mallin avulla ison datan ympäristössä voidaan toteuttaa yrityksen tietovarasto niin, että se tukee datan keräämistä ja jatkuvan kehittämisen periaatteen mukaista hyödyntämistä sekä parhaiden käytäntöjen jakamista organisaatiossa holistisen näkemyksen edistämiseksi.

7.1 Teollisen internetin vaikutus valmistavassa teollisuudessa

Kirjallisuustutkimuksen perusteella teollisen internetin myötä yritykset muodostavat entistä tiiviimpiä verkostoja, joista Nagabushanan [2006] käyttää termiä virtuaalinen yritys. Samalla koneiden, laitteiden ja tuotteiden muuttuessa entistä "älykkäämmiksi" ja niissä olevan anturoinnin lisääntyessä prosesseista syntyy entistä enemmän dataa entistä moninaisemmassa muodossa ja ihminen osallistuu prosessin operatiiviseen toimintaan entistä vähemmän.

Nämä muutokset tarkoittavat, että sekä 3 V:n määritelmän että HACE-teoreeman mukaan kyseessä on iso data. Tällöin myös yritysten tietovarastojen tulee muuttua niin, että ne pystyvät vastaamaan sekä perinteisen transaktiodatan tallentamiseen ja analysoimiseen että tällaisen suuremman, nopeammin kasvavan, moninaisemman ja vaihtelevamman datan tallentamiseen ja analysoimiseen. Datan analysoimisen merkitys korostuu, koska ihmisen osallistuessa prosessin operatiiviseen toimintaan entistä vähemmän tulee myös kehityskohteiden ja ongelmien juurisyiden tunnistaminen vaikeammaksi.

7.2 Integroitu malli datan tallentamiseen, hyödyntämiseen ja jakamiseen organisaatiossa

Esitetyssä integroidussa mallissa datan analysointi ja hyödyntäminen kehittämistasolla ja jakamistasolla nojaa tietovarastoon, joka pystyy tallentamaan isoa dataa ja tukemaan sen analysoimista. Kirjallisuustutkimuksen mukaan tällaisen tietovaraston tulee olla kerroksellinen, koska perinteisillä ratkaisuilla ei voida vastata toisaalta skaalautuvasti ja toisaalta kustannustehokkaasti ison datan tallentamiseen ja tallennetun datan hyödyntämisen vaatimuksiin. Perinteisten ratkaisujen suurin ongelma on huono horisontaalinen skaalautuvuus, joka johtuu relationaalisesta datamallista ja jaettu kaikki -arkkitehtuurista.

Kirjallisuustutkimuksen perusteella aikaisemmin ei ole tehty tutkimuksia, joissa olisi integroitu samaan malliin ison datan tallentamiseen ja analysoimiseen soveltuva yrityksen tietovarasto, systemaattinen prosessi datan analysoimiseksi ja suosittelujärjestelmä holistisen näkemyksen edistämiseksi ja olisi lisäksi huomioitu

teollisesta internetistä seuraavat erityispiirteet valmistavassa teollisuudessa. Eräs syy vastaavien tutkimuksien puuttumiseen on teollisen internetin uutuus. Teollisen internetin käsite on vanha, mutta vasta viime vuosina teknologinen kehitys on mahdollistanut käytännön sovellukset ja siksi kokonaisvaltaisen, integroidun mallin tarve on ollut vähäinen. Tulevaisuudessa tällainen malli voi toimia viitekehyksenä käytännön toteutuksille.

7.2.1 Ison datan tallentamiseen soveltuva yrityksen tietovarasto

Tällaisessa kerroksellisessa yrityksen tietovarastossa on kirjallisuustutkimuksen mukaan transaktiodatan tallentamiseen soveltuva kerros, joka vastaa yritysten tietovarastojen perinteisiä toteutusarkkitehtuureita noudattamalla relaatiotietokantoja, tähtikaaviota tai tietokuutioita ja jaettu kaikki -arkkitehtuuria. Lisäksi arkkitehtuurissa on ison datan tallentamiseen soveltuva kerros, jossa hyödynnetään NoSQL:n ja Hadoopin kaltaisia teknologioita, jotka tekevät arkkitehtuurista skaalautuvan ja kustannustehokkaan ja voivat tukea ison datan analysointia esimerkiksi assosiaatioanalyysia hyödynnettäessä. Tässä opinnäytetyössä esiteltiin esimerkiksi kaksi Kris Krishnanin [2013] esittämää kerroksellista arkkitehtuuria. Opinnäytetyön tuloksena esitellyssä integroidussa mallissa tietovarastotason arkkitehtuurina käytettiin Krishnanin esittelemää ulkoista dataintegraatioarkkitehtuuria.

7.2.2 Kehitystaso

Kehitystasossa esitettiin käytettäväksi DMAIC-kehitysympyrää. DMAIC-prosessin etuja mm. tietämyksen muodostamisprosessiin (Knowledge discovery from databases, KDD [Fayyad et al 1996]) ovat, että osana DMAIC:tä asetetaan projektille tavoitteet, jolloin ne on helpompi säilyttää projektin aikana mielessä eikä niistä harhauduta niin helposti. Toisaalta osana prosessia mietitään, millaista dataa ongelman ratkaisemiseksi tarvitaan ja jos oletusarvo on, että kaikkea tarvittavaa dataa ei ole saatavilla, sitä voidaan kerätä ennen analysointivaihetta. DMAIC-iteraatioon liittyy myös oleellisesti jatkuvan kehittämisen ajatus, eli jos kehitysprojektilla ei saavuteta asetettuja tavoitteita, on mahdollista aloittaa uusi iteraatio asetetun tavoitteen saavuttamiseksi. Lisäksi DMAIC-iteraation viimeisenä vaiheena on kontrollointi, jolloin tietovaraston raportointi-työkaluihin voidaan lisätä tarvittavat raportit, jotta ratkaistu ongelma ei toistuisi tai prosessi palaisi takaisin lähtötilanteeseen.

Teollisen internetin ympäristössä ihmiset osallistuvat kirjallisuustutkimuksen perusteella valmistusprosesseihin nykyistä vähemmän. Lisäksi toiminnan hajaannuttua verkostoissa laajemmalle on syy-seuraus-suhteiden muodostaminen ilman datan analysoimista entistä vaikeampaa. Siksi datan analysoiminen on vielä nykyistäkin

tärkeämpää, koska ihmiset eivät muuten osana työtään enää pysty löytämään syytä havaittuihin epäkohtiin tai tunnistamaan kehityskohteita.

7.2.3 Jakamistaso

Jakamistason tehtävänä on välittää tieto parhaista käytännöistä ja työtehtävien kannalta oleellisesta datasta eri osiin organisaatiota, ettei organisaatio menettäisi liiketoimintamahdollisuuksia. Datan määrän kasvaessa, datan monimuotoistuessa ja toiminnan automatisoituessa ilman suosittelujärjestelmää on suuri riski, että organisaation jäsenet ja yhteistyökumppanit eivät saa tietoonsa tarpeellista dataa.

Opinnäytetyössä on esitetty kirjallisuustutkimukseen ja omaan pohdintaan perustuen, että tällaisen suosittelujärjestelmän tulisi perustua tietovarastoon tallennetun datan sisältöön ja että työtehtäville tulisi muodostaa työtehtäväkohtaiset profiilit, joita yksittäiset työntekijät voisivat muokata joko formaalien määrittelyjen avulla tai antamalla palautetta heille suositellusta aineistosta. Aineiston suosittelussa on tunnistettu tärkeäksi huomioida myös tietoturva, ettei verkostoituneessa valmistusketjussa esimerkiksi loukata yhteistyökumppanin immateriaalioikeuksia jakamalla kilpailijalle luottamuksellista dataa.

Viiteluettelo

Bhogal, J., & Choksi, I. (2015, March). Handling Big Data using NoSQL. In *Advanced Information Networking and Applications Workshops (WAINA), 2015 IEEE 29th International Conference on* (pp. 393-398). IEEE.

Bhutta, K. S., & Huq, F. (1999). Benchmarking-best practices: an integrated approach. *Benchmarking: an International Journal*, *6*(3), 254-268.

Bruner, J. (2013). *Industrial Internet*. O'Reilly Media, Inc.

Coplien, J., & Bjørnvig, G. (2010). *Lean Architecture: for Agile Software Development*. John Wiley & Sons.

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From data mining to knowledge discovery in databases. *AI Magazine*, *17*(3), 37.

Hadoop. http://hadoop.apache.org/, viitattu 13.03.2016.

Hopp, W. J., & Spearman, M. L. (2000). *Factory Physics* (2nd ed.). McGraw Hill.

Hu, W. C., & Kaabouch, N. (2013). *Big Data Management, Technologies, and Applications* (1st ed.). IGI Global.

ISO9001:2015. *Laadunhallintajärjestelmät. Vaatimukset.*

ISO10303-1:1994. *Industrial Automation Systems and Integration –Product Data Representation and Exchange – Part 1: Overview and Fundamental Principles.*

Ivanov, D., Dolgui, A., Sokolov, B., Werner, F., & Ivanova, M. (2015). A dynamic model and an algorithm for short-term supply chain scheduling in the smart factory industry 4.0. *International Journal of Production Research, 54*(2), 386-402.

Jagtap, D. D., & Patil, B. K. (2014). Big Data using Hadoop. *International Journal of Engineering Research and General Science*, *2*(6), 467-472.

Kamsu-Foguem, B., Rigal, F., & Mauget, F. (2013). Mining association rules for the quality improvement of the production process. *Expert Systems with Applications*, *40*(4), 1034-1045.

Kimball, R. (1996). *The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouse*. John Wiley & Sons.

Kimball, R., & Ross, M. (2013). *The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling*. John Wiley & Sons.

Krishnan, K. (2013). *Data Warehousing in the Age of Big Data*. Morgan Kaufmann.

Kumar, R., Parashar, B. B., Gupta, S., Sharma, Y., & Gupta, N. (2014). Apache Hadoop, NoSQL and NewSQL solutions of Big Data. *International Journal of Advance Foundation and Research in Science & Engineering (IJAFRSE)*, *1*(6), 28-36.

Laney, D. (2001). 3D data management: controlling data volume, velocity and variety. *META Group Research Note*, *6*, 70.

Lau, H. C., Ho, G. T., Chu, K. F., Ho, W., & Lee, C. K. (2009). Development of an intelligent quality management system using fuzzy association rules. *Expert Systems with Applications*, *36*(2), 1801-1815.

Lops, P., De Gemmis, M., & Semeraro, G. (2011). Content-based recommender systems: state of the art and trends. In *Recommender Systems Handbook* (pp. 73-105). Springer US.

Nagabhushana, S. (2006). *Data Warehousing OLAP and Data Mining*. New Age International.

Ordonez, C. (2013). Can we analyze big data inside a DBMS?. In *Proceedings of the Sixteenth International Workshop on Data Warehousing and OLAP* (pp. 85-92). ACM.

Pokorný, J. (2014, November). How to store and process Big Data: Are today's databases sufficient?. In *IFIP International Conference on Computer Information Systems and Industrial Management* (pp. 5-10). Springer Berlin Heidelberg.

Posada, J., Toro, C., Barandiaran, I., Oyarzun, D., Stricker, D., de Amicis, R., ... & Vallarino, I. (2015). Visual computing as a key enabling technology for industrie 4.0 and industrial Internet. *IEEE Computer Graphics and Applications*, *35*(2), 26-40.

Ricci, F., Rokach, L., & Shapira, B. (2011). *Introduction to Recommender Systems Handbook* (pp. 1-35). Springer US.

Singh, J. T., Khanduja, D. (2015). Wrap *the Scrap with DMAIC: Strategic deployment of Six Sigma in Indian foundry SMEs*. Anchor Academic Publishing.

Slack N., Chambers S., & Johnston, R. (2004). *Operations Management, Fourth Edition*. Pearson education limited.

Wu, X., Zhu, X., Wu, G. Q., & Ding, W. (2014). Data mining with Big Data. *IEEE Transactions on Knowledge and Data Engineering*, *26*(1), 97-107.

Zink, K. J., (1998). *Total quality management as a holistic management concept, the European model for business excellence.* Springer-verlag Berlin.