



Osaamista
ja oivallusta
tulevaisuuden
tekemiseen

Sebastian Gymer

Tietojen luotettavuuden parantaminen yrityksen sisäisessä käytössä olevilla Qlik Sense -sovelluksilla

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tieto- ja viestintäteknikka

Insinööriyö

1.12.2020

Tekijä Otsikko	Sebastian Gymer Tietojen luotettavuuden parantaminen yrityksen sisäisessä käytössä olevilla Qlik Sense -sovelluksilla
Sivumäärä Aika	59 sivua + 2 liitettä 1.12.2020
Tutkinto	Insinööri (AMK)
Tutkinto-ohjelma	Tieto- ja viestintätekniikka
Ammatillinen pääaine	Hyvinvointi- ja terveysteknologia
Ohjaajat	Sakari Lukkarinen, Lehtori Ossi Erkkilä, Konsultoinnin johtaja
<p>Tämän insinöörityön tavoitteena oli parantaa tietojen luotettavuutta Cubiq Analytics Oy:n sisäisessä käytössä olevilla Qlik Sense -sovelluksilla. Tarkoituksena oli parantaa yrityksen tiedolla johtamisen edellytyksiä ja vähentää manuaaliseen työhön käytettävää aikaa. Tietojen luotettavuuteen vaikuttaa erityisesti kolme asiaa: tiedon ajantasaisuus, oikeellisuus ja yhdenmukaisuus eri sovellusten välillä. Sovellusten pääasiallisena datalähteenä oli yrityksen käyttämän ValueFrame-toiminnanohjausjärjestelmän taustalla toimiva tietokanta. Tarve sovellusten kehittämiseksi huomattiin keväällä 2018, kun yritykselle tehtiin uutta Qlik Sense -sovellusta olemassa olleiden sovellusten pohjalta. Aiemmillä sovelluksilla ei ollut noudatettu yrityksen määrittämiä hyviä käytäntöjä, ja lisäksi tiedot olivat joiltain osin väärin tai epäyhdenmukaiset eri sovellusten välillä.</p> <p>Tiedolla johtaminen tarkoittaa oikean tiedon hyödyntämistä päätöksenteossa. Oikea tieto saadaan esimerkiksi analysoimalla saatavilla olevaa dataa erilaisilla työkaluilla, kuten Qlik Sense -ohjelmistolla. Jos dataa analysoidaan väärin, siitä ei saada jalostettua luotettavaa tietoa, jota voidaan tehokkaasti hyödyntää päätöksenteossa. Tässä työssä data-analytiikan mahdollisuuksiin tutustuttiin Qlik-yrityksen ja sen tarjoamien tuotteiden, kuten Qlik Sense -ohjelmiston kautta.</p> <p>Työn tekeminen aloitettiin kesällä 2018. Ensimmäisessä vaiheessa kartoitettiin sovellusten senhetkinen tilanne ja määriteltiin ne asiat, joita voisi kehittää. Kehitettäviä asioita löytyi runsaasti ja niistä valittiin tärkeimmät, joita parannettiin seuraavan kesän aikana. Olemassa olleisiin sovelluksiin tehtiin muutoksia ja lisäksi luotiin uusia sovelluksia ja otettiin käyttöön yrityksen määrittämiä hyviä käytäntöjä sovelluskehityksessä. Tuloksena syntyi kokonaisuus, jossa ValueFramesta luettavan datan jalostaminen, mittareiden määrittelyt sekä erillisten tietojen ylläpitäminen on hallittu keskitetympin kuin aiemmin. Työntekijöille tehdyn kyselyn perusteella sovellusten tiedot ovat kehitysvaiheen jälkeen olleet pääasiassa oikein, ajan tasalla ja yhdenmukaisia eri sovellusten välillä. Kehitysvaiheen jälkeen kokonaisuuteen ei ole tarvinnut kahden vuoden aikana tehdä suuria muutoksia. Yrityksen tiedolla johtamisen edellytykset ovat tämän kokonaisuuden osalta nyt hyvällä tasolla.</p>	
Avainsanat	Tiedolla johtaminen, tieto, misinformaatio, disinformaatio, data-analytiikka, dataintegraatio, Qlik, Qlik Sense, QlikView, data, mallintaminen, jalostaminen, valmistelu, visualisointi

Author Title Number of Pages Date	Sebastian Gymer Improving Reliability of Information Displayed on Qlik Sense Applications Used Internally in Company 59 pages + 2 appendices 1.12.2020
Degree	Bachelor of Engineering
Degree Programme	Information and Communication Technology
Professional Major	Health Technology
Instructors	Sakari Lukkarinen, Senior Lecturer Ossi Erkkilä, Consulting Director
<p>The main objective of the study was to improve the reliability of information displayed on Cubiq Analytics Oy's internal Qlik Sense applications. The purpose was to enhance the conditions for knowledge-based management in the company and to reduce the time spent on manual labour. The reliability of information was determined to be affected by three main factors: the timeliness, correctness, and consistency of the information between different applications. The main data source for the applications was the database behind the ValueFrame Enterprise Resource Planning (ERP) system used by the company. The need for application development was noticed in the spring of 2018, when a new Qlik Sense application was being developed for the company based on the existing applications. Previous applications had not followed the good practices identified by the company and, in addition, the information was in some respects incorrect or inconsistent between the different applications.</p> <p>Knowledge-based management means utilizing the right information in decision-making. The right information is obtained, for example, by analyzing the available data with various tools, such as Qlik Sense software. If data is mis-analyzed, it will not provide reliable information that can be effectively utilized in decision-making. In the study, the possibilities of data analytics were explored through Qlik and the products it offers, e.g. Qlik Sense software.</p> <p>Within the study, the current state of the applications was examined and the features that could be developed were identified. There were many things to develop from which the most important ones were selected and improved over the summer of 2018. Modifications were made to the existing applications and new applications were created. Good practices in application development defined by the company were also introduced. The result was an entity in which the processing of data read from ValueFrame, the definition of metrics and the maintenance of separate data are all managed more centrally than before. Based on a survey carried out amongst the Cubiq Analytics employees, the information displayed on the applications after the development phase was found to be mainly correct, up-to-date, and consistent across applications.</p>	
Keywords	Knowledge-based management, information, data, misinformation, disinformation, data analytics, data integration, Qlik, Qlik Sense, QlikView, data, modelling, processing, preparing, visualizing

Sisällys

Lyhenteet

1	Johdanto	1
2	Taustat ja tarve analytiikan kehittämiseksi Cubiqilla	3
3	Tiedolla johtaminen ja data-analytiikka	6
4	Qlik	13
4.1	Qlik yrityksenä	13
4.2	Qlikin tuotteet	13
5	Qlik Sense	17
5.1	Ohjelmisto	17
5.2	Käyttöliittymä	18
5.3	Assosiativinen moottori	19
5.4	Sovellukset	24
5.4.1	Loppukäyttäjän näkökulma	24
5.4.2	Sovelluskehittäjän näkökulma	29
6	Analytiikan kehittämisen työvaiheet Cubiqilla	35
6.1	Ongelman ja tavoitteen määrittely ja rajaukset	35
6.2	Kartoitusvaihe	36
6.2.1	Sovellukset ja datalähteet	37
6.2.2	Latausskriptit	38
6.2.3	Datamallit	39
6.2.4	Visualisoinnit, Master items -kirjastot ja muuttujat	40
6.3	Kehitysvaihe	41
6.3.1	Kerroksellisen tietoarkkitehtuurin rakentaminen	43
6.3.2	Datan mallintaminen	45
6.3.3	Mittareiden määrittelyt ja keskitetty hallinta	47
6.3.4	Erikseen ylläpidettävien tietojen ja tiedostojen määrän minimointi	47
6.3.5	Sovelluksilla näytettävien tietojen ajantasaisuuden parantaminen	47

6.3.6	Dokumentaatio ja yhteiset käytännöt	48
7	Saavutetut tulokset	50
8	Pohdinta	54
	Lähteet	56

Liitteet

Liite 1. Kysely Qlik Sense -sovellusten tietojen luotettavuudesta edellisten kahden vuoden aikana

Liite 2. Kyselyn vastaukset

Lyhenteet ja termit

CEBMa	<i>Center for Evidence-Based Management</i> . Tiedolla johtamisen keskus.
ERP	<i>Enterprise Resource Planning</i> . Yrityksen toiminnanohjausjärjestelmä.
Hub	<i>Qlik Sense Hub</i> . Qlik Sense -ohjelmiston näkymä, jossa voidaan kehittää uusia sovelluksia ja käyttää valmiita sovelluksia.
PostgreSQL	Avoimena lähdekoodina jaettava tietokannan hallintajärjestelmä.
QMC	<i>Qlik Management Console</i> . Qlik Sense -ohjelmiston hallintapaneeli.
QSEoK	<i>Qlik Sense Enterprise on Kubernetes</i> . Kubernetes-ympäristössä käytettävä versio Qlik Sense -ohjelmistosta.
QSEoW	<i>Qlik Sense Enterprise on Windows</i> . Windows-ympäristössä käytettävä versio Qlik Sense -ohjelmistosta.
QVD	<i>QlikView Data</i> . Tiedostomuoto, joka sisältää Qlik Sense- tai QlikView-ohjelmistolla tallennettua dataa.
RAM	<i>Random Access Memory</i> eli hajasaantimuisti on muistityyppi tietotekniikassa. RAM-muistia ovat muun muassa tietokoneen välimuisti ja keskusmuisti.
SQL	<i>Structured Query Language</i> (SQL) on IBM:n kehittämä standardoitu kyselykieli, jolla relaatiotietokantaan voi tehdä erilaisia hakuja, muutoksia ja lisäyksiä.
TLS	<i>Transport Layer Security</i> (TLS), aiemmin tunnettu nimellä <i>Secure Sockets Layer</i> (SSL) on salausprotokolla, jolla voidaan suojata Internet-sovellusten tietoliikenne IP-verkkojen yli. Se on nykyisin yksi tavallisimmista tavoista suojata tietoliikennettä.

1 Johdanto

”Tiedolla johtaminen” on termi, johon moni meistä todennäköisesti törmää useammankin kerran työuransa aikana. Erilaisilla organisaatioilla ja yksittäisillä ihmisilläkin on nykyisin pääsy isoon ja eksponentiaalisesti kasvavaan määrään dataa ja täten mahdollisuus käyttää sitä hyödyksi toimintansa johtamisessa. Monilla yrityksillä on käytössään erilaisia data-analytiikkaohjelmistoja osana liiketoimintansa analysointia ja kehittämistä. Tavoitteena on jalostaa datasta tietoa, jota voidaan käyttää päätöksenteon tukena yrityksen johtamisessa. Mutta entä jos datasta jalostettu tieto, jonka perusteella päätöksiä tulisi tehdä, ei vaikutakaan luotettavalta?

Tämän työn tilaaja on suomalainen data-analytiikka-alan yritys Cubiq Analytics Oy. Cubiq kertoo tehtäväkseen toteuttaa suomalaisissa organisaatioissa ennennäkemättömiä läpimurtoja datan avulla. Iso osa näistä läpimurroista tehdään käyttäen Qlikin tuotteita, kuten Qlik Senseä. Cubiqin asiakkaat ovat Suomen johtavia yrityksiä ja julkisorganisaatioita, joiden tiedolla johtamista ja dataohjautuvuutta pyritään tiiviin yhteistyön kautta parantamaan jatkuvasti ja näkyvästi. [1.] Oman jokapäiväisen yritystoimintansa ohjaamisen apuna Cubiq käyttää Visman ValueFrame-toiminnanohjausjärjestelmää. ValueFrameissa ylläpidetään esimerkiksi työntekijöiden tuntikirjauksia, laskutukseen liittyvää tietoa, ynnä muuta. ValueFramesta tiedot luetaan analysoitavaksi Cubiqin Qlik Sense -ympäristöön ja siellä oleville sovelluksille. Sovellusten tärkein tehtävä on tarjota Cubiqin avainhenkilöille ajantasainen ja oikea tieto yrityksen toiminnasta sekä mahdollistaa tiedon hyödyntäminen päätöksenteossa.

Aloitin työharjoittelun Cubiqilla keväällä 2018 ja olen työskennellyt yrityksessä täysipäiväisesti siitä asti. Työharjoittelun aikana huomasin, ettei yrityksen oman toiminnan analysoinnissa käytettyjä Qlik Sense -sovelluksia ollut rakennettu yrityksen määrittelemien hyvien käytäntöjen mukaisesti. Lisäksi sovelluksilla olleet tiedot eivät olleet aina ajan tasalla, tai ne olivat joiltain osin jopa väärin tai puutteelliset. Yritysjohdanto ei luottanut täysin sovelluksilta saamaansa tietoon, vaan lukuja jouduttiin tarkistamaan usein manuaalisesti. Tämän työn tekemisen tavoitteena oli varmistaa, että työntilaajayrityksen liiketoiminnan analysoinnissa käytettävät Qlik Sense -sovellukset sisältäisivät mahdollisimman

ajantasaista ja virheetöntä tietoa, joka ei ole ristiriidassa eri sovellusten välillä. Pyrkimyksenä oli siis parantaa sovelluksilla näytettävän tiedon luotettavuutta, jotta sen perusteella voidaan tehdä oikeita päätöksiä yrityksen johtamisessa. Tavoitteeseen pääsy vaati melko syvällistä tutustumista yrityksen ValueFrame- ja Qlik Sense -ympäristöihin sekä olemassa olleisiin sovelluksiin. Lisäksi täytyi muodostaa ymmärrys tiedolla johtamisen periaatteista ja varsinkin Qlik Sense -ohjelmiston ja sovellusten toiminnasta yleisellä tasolla.

Tässä opinnäytetyössä taustoitan ensin tarkemmin sitä, miten kehitystarpeet tulivat ilmi työharjoittelujaksoni aikana. Taustoituksen jälkeen kerron tiedolla johtamisesta, Qlik-yrityksestä ja sen tuotteista sekä Qlik Sense -ohjelmistosta ja -sovelluksista. Lopuksi käyn läpi käytännön työvaiheet ja esittelen saavutetut tulokset.

Tässä opinnäytetyössä keskitytään vain Qlik Sense Enterprise on Windows -ohjelmistoon (QSEoW), sillä tätä versiota käytetään myös Cubiqilla. Qlik Sensestä on olemassa myös esimerkiksi Qlik Sense on Kubernetes (QSEoK), Qlik Sense Cloud ja Qlik Sense Desktop -versiot, mutta tässä opinnäytetyössä Qlik Sensellä tarkoitetaan nimenomaan QSEoW-versiota, ja kaikki esimerkit ovat tällä versiolla tehtyjä.

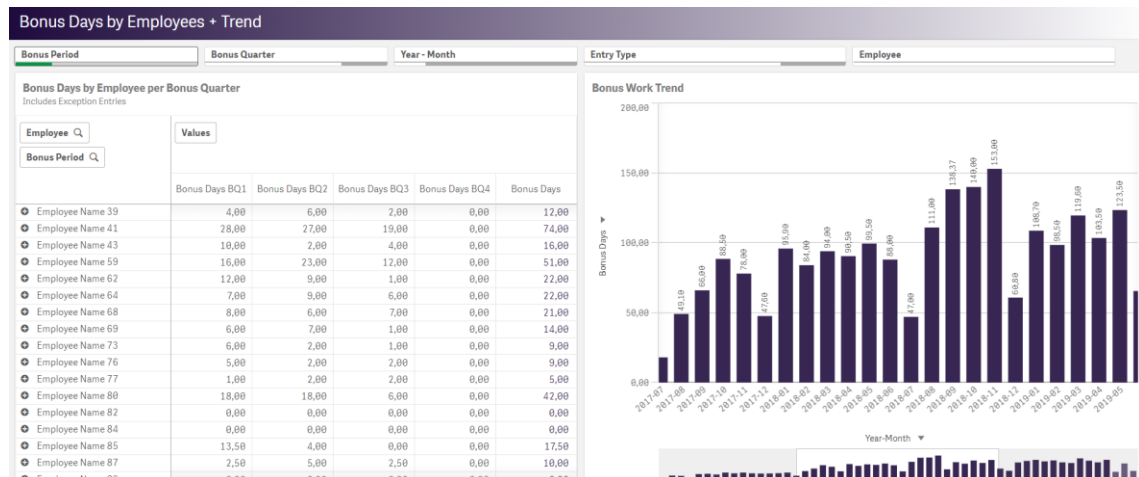
2 Taustat ja tarve analytiikan kehittämiseksi Cubiqilla

Keväällä 2018 aloitin työharjoittelun Cubiqilla yhdessä neljän muun henkilön kanssa. Neljän kuukauden pituisen työharjoittelujakson tarkoituksena oli kouluttaa data-analytiikasta kiinnostuneista korkeakouluopiskelijoista tai opintonsa hiljattain päättäneistä uusia täyspäiväisiä työntekijöitä yritykselle. Työharjoittelun ensimmäiset kuukaudet sisälsivät kymmeniä koulutuksia ja intensiivistä itseopiskelua data-analytiikka-alaan ja erityisesti Qlik Sense -ohjelmistoon ja -sovelluskehitykseen liittyen. Koulutuksia pitivät enimmäkseen Cubiqin kokeneemmat konsultit, mutta näiden lisäksi kouluttajia oli myös yrityksen ulkopuolelta. Koulutusten tavoitteena oli opettaa meitä tekemään asiat alalla yleisesti hyväksytyjen ja yrityksen määrittämien hyvien käytäntöjen mukaisesti. Työharjoittelun loppupuolella saimme toteutettavaksemme erilaisia yrityksen sisäisiä projekteja, joissa pääsimme hyödyntämään koulutuksissa oppimiamme asioita. Harjoitusprojektien tarkoituksena oli valmistaa meitä asiakkaille tehtäviä ”oikeita” projekteja varten.

Omassa harjoitusprojektissani (yhteistyössä toisen harjoittelijan kanssa) sain tehtäväkseni laatia yrityksen käyttöön uuden Qlik Sense -sovelluksen, jolla voidaan analysoida työntekijöiden kirjaamia työtunteja. Työtunnit kirjattiin ValueFrame-toiminnanohjausjärjestelmään, jonka taustalla olevasta tietokannasta ne luettiin Qlik Senseen. Tavoitteena oli vähentää sekä yrityksen talousosaston tekemän manuaalisen työn määrää että työntekijäkohtaisten laskutusbonusten laskemiseen kuluva aikaa. Tavoitteena oli lisäksi tuoda laskentojen perusteena käytettävät tiedot läpinäkyvästi myös työntekijöiden saataville. Työntekijäkohtaisia bonuksia maksettiin neljä kertaa vuodessa, ja maksettavien bonusten määrä riippui kunkin työntekijän tekemistä bonukseen oikeuttavista työpäivistä. Ennen sovelluksen luontia yrityksen talousosasto laski bonusten määrät kullekin työntekijälle erikseen jokaisen kvartaalin jälkeen, eivätkä työntekijät päässeet itse näkemään kaikkia laskennassa käytettyjä tarkkoja lukuja mistään.

Harjoitusprojektin tilaajana toimi Cubiqin toimitusjohtaja, joka määritteli sovelluksen toivotun ulkoasun melko tarkkaan – tärkeintä oli, että sovelluksella näytettävät tiedot olisivat oikein, jotta samoja asioita ei tarvitsisi laskea erikseen käsin. Bonus-sovelluksen pohjana käytettiin aiempien työharjoittelijoiden tekemää Raportointi-sovellusta, jolla samoja tuntikirjaustietoja analysoitiin hieman eri näkökulmasta. Bonus-sovelluksen ensimmäinen versio sisälsi paljon samoja latausskriptin osia sekä laskennoissa käytettyjä kaavoja

kuin Raportointi-sovellus, mutta siihen lisättiin bonuslaskennassa tarvittavaa dataa erillisestä Excel-tiedostosta. Excel-tiedostosta tuotua dataa lukuun ottamatta, data ja useat laskentakaavat sovellusten välillä olivat tässä kohtaa identtisiä. Kuvassa 1 on näkymä harjoitusprojektin lopputuloksena syntyneestä, niin sanotusta Bonus-sovelluksesta.



Kuva 1. Uuden Bonus-sovelluksen näkymä (työntekijöiden nimet ja luvut sotkettu).

Bonus-sovellukseen luotiin aluksi vain yksi näkymä (kuva 1), josta pystyi näkemään bonuskeihin oikeuttavien työpäivien määrän työntekijöittäin per tilikauden kvartaali (vasemman reunan taulukko). Lisäksi välilehdellä oli suodattimia (yläreunan painikkeet) sekä kuvaaja, josta näki trendiä ajan suhteen (pylväsdiagrammi). Sovelluksen avulla työntekijäkohtaisten bonusten laskentaa saatiin nopeutettua ja helpotettua huomattavasti. Lisäksi työntekijät pääsivät sovelluksen kautta näkemään, mitä lukuja laskennassa oli käytetty.

Samaan aikaan kun laadimme Bonus-sovellusta, myös muut työharjoittelijat työstivät harjoitteluprojekteissaan uusia Qlik Sense -sovelluksia yrityksen sisäiseen käyttöön. Näillä sovelluksilla käytettiin ainakin osittain samaa dataa ja samoja laskentakaavoja kuin Bonus- ja Raportointi-sovelluksilla. Vaikka sovelluksilla oli paljon yhteistä sisältöä, niin niille oli myös eri käyttötarkoitukset, joten niistä ei haluttu muodostaa yhtä sovellusta, johon ”ahdettaisiin” kaikki mahdolliset näkymät. Sovellukset haluttiin pitää erillään, omina kokonaisuuksinaan, eli samaa dataa ja laskentakaavoja jouduttiin hallitsemaan usealla eri sovelluksella. Ymmärsin pian, että tiedot tulevat ennemmin tai myöhemmin olemaan

ristiriidassa tai eri tarkkuudella näiden välillä, esimerkiksi kun latausskriptiä tai tiettyä laskentakaavaa muutetaan jollain sovelluksella. Edellä mainittujen asioiden lisäksi huomasin, että alkuperäisten sovellusten data oli puutteellista, vaikeasti ylläpidettävää tai jopa joiltain osin kokonaan väärää, ja että koulutuksissa opetettuja hyviä käytäntöjä ei ollut noudatettu esimerkiksi datan mallintamisessa.

Esittelin harjoittelutyön aikana tekemiäni huomioita yritysjohdolle ja ehdotin, että lähtisin parantamaan yrityksen sisäisessä käytössä olevia Qlik Sense -sovelluksia. Yritysjohto suhtautui ehdotukseen erittäin myönteisesti, ja sain hyväksynnän ehdotukselle. Myöhemmin sovittiin, että tekisin myös opinnäytetyöni aiheen tiimoilta.

Seuraavissa luvuissa pohjustetaan työn aihetta kertomalla tiedolla johtamisesta sekä data-analytiikasta ja erityisesti Qlik Sense -ohjelmistosta ja sovelluksista yleisellä tasolla. Luvusta 6 alkaen palataan Cubiqin toimintaympäristöön ja kerrotaan käytännön työvaiheista sekä saavutetuista tuloksista.

3 Tiedolla johtaminen ja data-analytiikka

Tiedolla johtaminen ja data-analytiikka ovat trendikkäitä termejä, joihin liittyen on lähi-vuosina kirjoitettu valtava määrä opinnäytetöitä ja muita julkaisuja. Esimerkiksi haulla ”tiedolla johtaminen” löytyy Theseus-palvelusta jo yli 4400 julkaisua [2]. Tiedolla johta-misen termille löytyy muun muassa edellä mainituista julkaisuista useita hieman erilaisia määritelmiä. Valtiohallinnon #Tietokiri-hankkeessa tiedolla johtaminen on määritelty tie-toiseksi johtamis- ja toimintamalliksi, jossa tietoa hyödynnetään ja analysoitua tietoa ja dataa tuodaan osaksi päätöksentekoprosessia [3]. Yksinkertaisimmillaan tiedolla johta-minen tarkoittaa oikeaan tietoon perustuvaa päätöksentekoa, jossa oikea tieto saadaan dataa analysoimalla [4]. Määritelmien mukaan data-analytiikka on siis osa tiedolla johta-mista eikä siitä irrallinen asia. Digitaalisen datan määrän kasvaessa alati kiihtyvään tah-tiin, on tiedolla johtaminen käytännössä mahdotonta ilman data-analytiikkaa.

Vaikka tiedolla johtamiselle ei ole yhtä yleisesti hyväksyttyä täsmällistä määritelmää, ja eri organisaatiot määrittelevät termin hieman eri tavoin, on perimmäinen ajatus silti sama: toimintaa johdetaan ja päätöksiä tehdään systemaattisesti kerättyyn tietoon pe-rustuen, eikä esimerkiksi tunteiden tai luulojen pohjalta. Tiedolla johtamisen päämääränä on tukea yrityksen tai organisaation strategian toteutumista hyödyntämällä tehokkaasti olemassa olevaa tietoa [5]. Tiedolla johtaminen mahdollistaa esimerkiksi sen, että erilai-set näkökulmat, vaihtoehdot ja riittävä informaatio otetaan johtamisessa ja päätöksente-ossa huomioon [6].

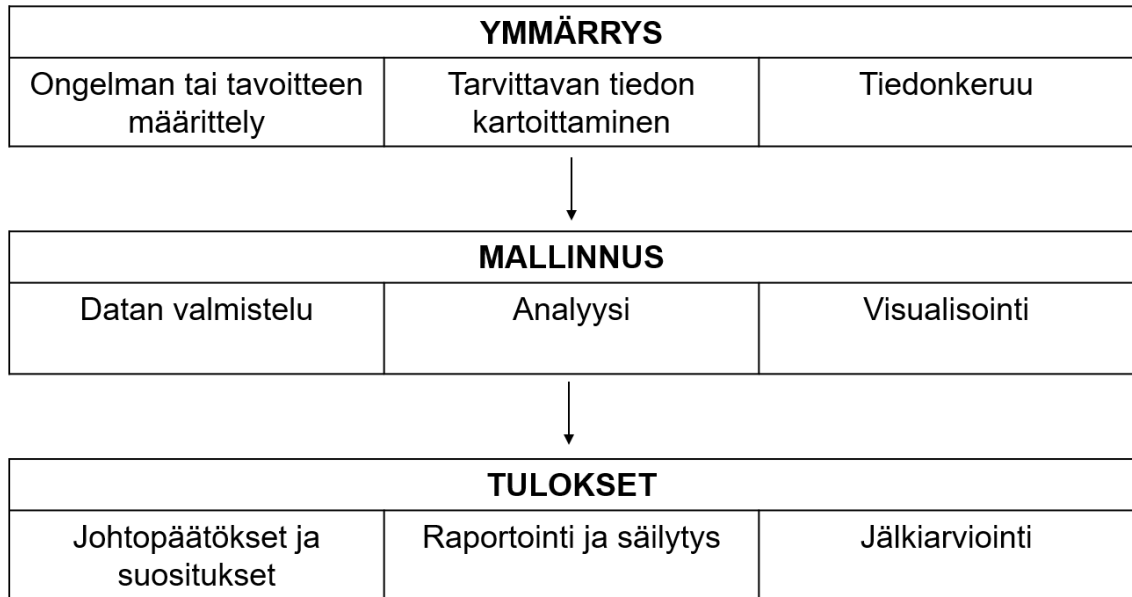
CEBMa:n (Center for Evidence-Based Management) julkaisussa Evidence Based Ma-nagement – The Basic Principles [7] tiedolla johtamista kuvataan iteroivana prosessina, joka sisältää kuusi päävaihetta, jotka on kuvattu taulukossa 1.

Taulukko 1. Tiedolla johtamisen prosessin vaiheet (CEBMa) [7].

Vaihe	Selite
1. Kysyminen (Asking)	Päätettävän asian tunnistaminen vastattavissa olevaksi kysymykseksi.
2. Hankinta (Acquiring)	Tietojen systemaattinen etsiminen ja hankinta.
3. Arviointi (Appraising)	Tietojen luotettavuuden ja merkityksellisyyden kriittinen arviointi.
4. Koostaminen (Aggregating)	Tietojen koostaminen yhteen ja tarkastelu.
5. Soveltaminen (Applying)	Tietojen soveltaminen ja hyödyntäminen osana päätöksentekoprosessia.
6. Arviointi (Assessing)	Tietojen perusteella tehtyjen päätösten vaikutusten arviointi.

Tiedolla johtamisen prosessi alkaa muotoilemalla tutkittava asia tai ongelma kysymykseksi, johon voidaan vastata, esimerkiksi ”Onko yrityksemme toiminta taloudellisesti kannattavaa?” Seuraavissa vaiheissa hankitaan aiheeseen liittyvää tietoa, arvioidaan sen luotettavuutta ja koostetaan tiedot yhteen. Lopuksi hyödynnetään hankittua tietoa päätöksenteossa ja arvioidaan tietojen perusteella tehdyn päätöksen vaikutuksia. Viimeisen vaiheen jälkeen voidaan aloittaa uudestaan ensimmäisestä vaiheesta ja pyrkiä tekemään vielä parempia päätöksiä toiminnan johtamisessa.

Kaakkois-Suomen ammattikorkeakoulun digitaalisen tiedonhallinnon tki-asiantuntija Miia Kosonen kuvailee tiedolla johtamisen prosessin sisältävän kuvassa 2 esitetyt yhdeksän vaihetta [8, s.10]. Kuvaa luetaan ylhäältä alaspäin, vasemmalta oikealle.



Kuva 2. Tiedolla johtamisen prosessin yhdeksän vaihetta [mukaillen 8, s. 10].

Kuvassa 2 on kuvattu Kososen esittelemä tiedolla johtamisen prosessikuvaus [8]. Tässä määritelmässä prosessi jakaantuu kolmeen päävaiheeseen, joista kukin jakaantuu vielä kolmeen eri vaiheeseen. Kososen prosessikuvaus sopii hyvin juuri tässä työssä esiteltävään aiheeseen, sillä siinä on erikseen eroteltu esimerkiksi vaiheet ”datan valmistelu” ja ”visualisointi”. Kososen määrittelemät yhdeksän vaihetta voidaan kuitenkin nähdä myös osana aiemmin esiteltyä CEBMa:n prosessia, esimerkiksi seuraavasti:

Taulukko 2. Tiedolla johtamisen prosessit rinnakkain [7; 8].

Päävaihe	Vaihe	Vaihe (CEBMa)
Ymmärrys	Ongelman tai tavoitteen määrittely	1. Kysyminen
	Tarvittavan tiedon kartoittaminen	1. Kysyminen / 2. Hankinta
	Tiedon keruu	2. Hankinta
Mallinnus	Datan valmistelu	3. Arviointi
	Analyysi	4. Koostaminen
	Visualisointi	4. Koostaminen
Tulokset	Johtopäätökset ja suositukset	5. Soveltaminen
	Raportointi ja säilytys	4. Koostaminen / 5. Soveltaminen
	Jälkiarviointi	6. Arviointi

Taulukossa 2 esitetyt prosessikuvaukset sisältävät kuta kuinkin samat vaiheet, eikä niiden eri vaiheille ole määritetty eri painoarvoja tai tärkeysjärjestystä. Kosonen toteaa kuitenkin mallinnusvaiheen ja eritoten datan valmisteluvaiheen olevan yksi työläimmistä ja aikaa vievimmistä vaiheista [8, s. 10]. Tässä opinnäytetyössä tiedolla johtamisen prosessi nähdään Kososen esittelemän mallin mukaisena.

Suomessa esimerkiksi julkista hallintoa on ollut tapana kehittää ja uudistaa tietoa hyväksi käyttäen [9]. Muun muassa valtiohallinnossa perustetussa #Tietokiri-hankkeessa tiedolla johtamisen prosessi on johdettu ensin mainitusta kuusivaiheisesta prosessikuvauksesta. #Tietokiri-hankkeen tavoitteena on edistää valtiohallinnon tiedolla johtamisen kulttuuria ja tehdä näkyväksi tiedolla johtamisen tuloksia ja jakaa hyväksi havaittuja työkaluja ja oppeja. Hankkeessa on mukana muun muassa valtionvarainministeriö, valtiokonttori ja maanmittauslaitos sekä useita muita toimijoita.

Sen lisäksi, että tiedolla johtamisen määritelmiä ja prosessikuvauksia on erilaisia, niin myös tiedolla johtamisen tavoitteet voivat olla erilaisia. Tavoitteet voidaan luokitella esimerkiksi taulukon 3 mukaisesti.

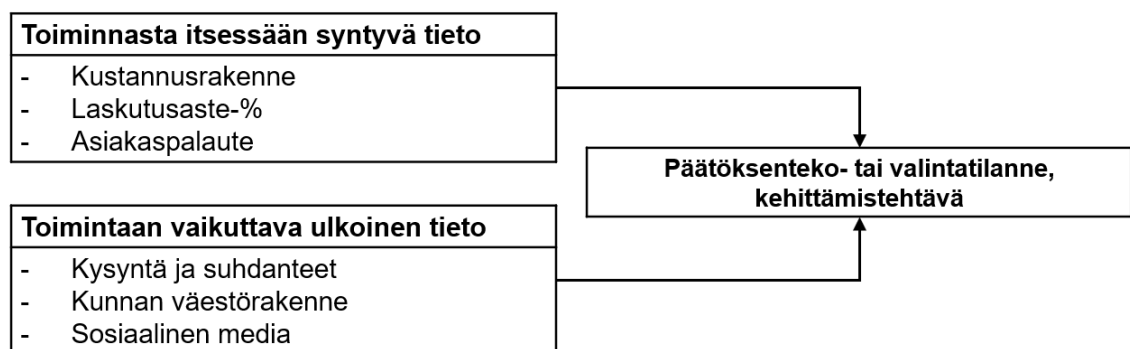
Taulukko 3. Tiedolla johtamisen erilaisia tavoitteita [8, s. 10–11].

Tavoite	Kuvaus
Kuvailu	Selvitetään, mitä on tapahtunut.
Diagnoosi	Selvitetään, miksi jotain on tapahtunut.
Ennakointi	Arvioidaan, mitä mahdollisesti tulee tapahtumaan.
Ohjailu	Laaditaan suosituksia siitä, mitä pitäisi tehdä.

Taulukossa 3 esitellyt tiedolla johtamisen tavoitteet ohjaavat tiedolla johtamisen prosessia. Esimerkiksi, jos tavoite on kuvailu tai diagnosointi, niin voidaan keskittyä keräämään tietoa jo tapahtuneesta. Kun taas pyritään vastaamaan tuleviin haasteisiin, voidaan ajatella tiedolla johtamisen tavoitteiden olevan ennakoivia tai ohjailevia. Tällöin myös kerättävä tieto on todennäköisesti hyvin erilaista ja mallinnusvaiheessa sitä saatetaan käsitellä eri tavalla. Tiedolla johtamisella on erilaisia tavoitteita eri tilanteissa, ja oikeastaan kaikki neljä edellä mainittua tavoitetta voi toteutua samassa organisaatiossa. Esimerkiksi Cubiqilla tavoitteena voi olla selvittää, kuinka paljon työtunteja on tehty edellisenä vuotena ja onko kehitys ollut nousevaa vai laskevaa sinä aikana. Toisaalta sen perusteella

seuraavana tavoitteena on mahdollisimman tarkasti ennakoida työntekijöiden määrän tarvetta tulevaisuudessa ja tarvittaessa ohjata esimerkiksi rekrytointiin lisään resursseja.

Valtionvarainministeriön toimittamassa Tiedolla johtamisen käsikirjassa todetaan, että kunnat voivat vastata tuleviin haasteisiinsa vain kehittämällä voimakkaasti omaa toimintaansa ja ”toiminnan tehokas kehittäminen on mahdollista vain, jos päätöksenteossa on käytettävissä oikea, luotettava, riittävän laadukas ja ajantasainen tieto omasta toiminnasta ja toimintaympäristöstä” [10, s. 9]. Edellä mainittu virke sisältää tärkeän huomion siitä, että tiedolla johtaminen edellyttää sekä toiminnasta itsessään syntyvää tietoa että toimintaan vaikuttavaa ulkoista tietoa. Tätä on havainnollistettu kuvassa 3.



Kuva 3. Päätöksentekotilanteessa hyödynnettävät tiedot. [mukaillen 8, s. 7]

Kuvassa 3 on havainnollistettu sitä, että päätöksenteossa tarvitaan sekä sisäistä että ulkoista tietoa. Sisäinen tieto (kuva 3) voi liittyä esimerkiksi yrityksen kustannusrakenteeseen, toimintaan vaikuttava ulkoinen tieto taas voi liittyä esimerkiksi kysyntään ja suhdanteisiin. [8.] Esimerkiksi kunnan väestörakenteen muutosten huomioiminen kunnan palvelujen tarjonnassa on mahdollista vain, jos tietoa kerätään, analysoidaan ja tuodaan osaksi päätöksentekoprosessia. Espoossa yli 64-vuotiaiden osuus kaikista kunnan asukkaista on yli kaksinkertaistunut edellisten reilun kolmenkymmenen vuoden aikana [11]. Muun muassa edellä mainitun kaltaista ulkoista tietoa voidaan käyttää yhdessä kunnan toiminnasta syntyvän sisäisen tiedon kanssa ja hyödyntää sitten Espoon toiminnan johtamisessa. Cubiq hyödyntää niin ikään sekä yrityksen omasta toiminnasta syntyvää tietoa että ulkopuolista tietoa tiedolla johtamisessa. Asiakaspalautteen lisäksi esimerkiksi sosiaalisesta mediasta kerättyä tietoa hyödynnetään päätöksenteossa. Päätös voi liittyä

esimerkiksi siihen, että minä päivänä ja mihin aikaan on järkevintä julkaista tiedote tai mainos, jolle halutaan mahdollisimman suuri näkyvyys.

Tiedolla johtamisen haasteita on myös tutkittu jonkin verran viime vuosina. Esimerkiksi Sitran asiantuntijat Hellström ja Ikäheimo toteuttivat kesällä 2017 Tieto päätöksenteossa -kyselyn [12]. Tutkimuksessa ongelmia tunnistettiin sekä tiedontuotannossa että päätöksenteossa. Kyselyn tulosten mukaan vain noin joka neljäs vastaaja (23 %) oli sitä mieltä, että tiedontuotannon ja yhteiskunnallisen päätöksenteon yhteistyö ja vuorovaikutus toimii Suomessa hyvin. Tiedon hyödyntämiselle päätöksenteossa tunnistettiin kuusi merkittävää haastetta:

- kiire
- tarkoitushakuisuus
- yksipuolisuus
- digitalisaatio
- tiedonlukutaito
- kilpaileva tieto. [12.]

Kyselyn vastaajien mukaan päättäjillä ei ole tarpeeksi aikaa tutustua päätöksiin tarvittavaan tietoon, tietoa käytetään tarkoitushakuisesti tai yksipuolisesti, eikä digitalisaation tarjoamia erilaisia mahdollisuuksia hyödynnetä riittävän monipuolisesti. Lisäksi haasteiksi tunnistettiin, että päättäjät eivät osaa riittävän hyvin arvioida saamaansa tietoa tai sen luotettavuutta ja tekemään sen perusteella oikeita johtopäätöksiä. Haasteeksi nähtiin myös kilpailevan tiedon suuri määrä. [12.]

Kilpailevan tiedon määrän lisäksi tiedolla johtamiselle aiheuttaa haasteita saatavilla oleva väärä tieto. Tiedon määrän kasvaessa myös tahattomasti (misinformaatio) ja tahallisesti tuotetun väärän tiedon (disinformaatio) määrä kasvaa väistämättä. Esimerkiksi Facebook joutui hiljattain kieltämään kaiken sisällön, jolla pyritään kiistämään holokausti (juutalaisten kansanmurha toisen maailmansodan aikana), sillä aiheesta levitettiin paljon disinformaatiota [13]. Myös erityisesti koronaviruksen ja Yhdysvaltain presidentinvaalien yhteydessä uutisotsikoissa näkee useasti puhuttavan disinformaatiosta. Esimerkiksi Helsingin sanomien uutisessa [14] viitattiin Cornellin yliopistossa tehtyyn tutkimukseen,

jonka mukaan mukaan Yhdysvaltain presidentti Donald Trump on suurin yksittäinen tekijä ("driver"), jonka kautta koronavirukseen liittyvä "infodemia" eli väärä tieto on päässyt leviämään. Disinformaation aikakaudella oikean tiedon erottaminen väärästä tiedosta on haastavaa. Tämä voi ilmetä myös niin, että erilaisten tutkimusten tuloksia lukiessa tulee kyseenalaistaneeksi tutkijoiden motiiveja eikä luota tutkimusten tuloksiin. Toiminnan johtamista saatetaan sitten jatkaa tunteiden ja vanhojen luulojen pohjalta.

Vaikka datan käsittelyyn, analysointiin ja visualisointiin tarkoitetut työkalut ovat kehittyneet valtavasti viimeisten vuosien aikana, niin myös validin ja oikean tiedon saaminen oikea-aikaisesti on osoittautunut haastavaksi. Esimerkiksi saatavilla oleva disinformaatio aiheuttaa haasteita oikean tiedon erottamiselle. Toisaalta datan suuri määrä itsessään asettaa data-analytiikkaohjelmistoille kasvavia vaatimuksia. Ohjelmistot tarjoavat kuitenkin yhä parempia keinoja tiedon tehokkaaseen käsittelyyn ja analysointiin. [5.]

4 Qlik

Tässä työssä data-analytiikasta ja sen tarjoamista mahdollisuuksista kerrotaan Qlik-yrityksen ja sen tuotteiden kautta.

4.1 Qlik yrityksenä

Qlik perustettiin vuonna 1993 Lundissa, Ruotsissa – alun perin nimellä QlikTech [15]. Nyt Qlik on yksi maailman johtavista data-analytiikkaratkaisuihin erikoistuneista teknologiayrityksistä [16]. Qlikin pääkonttori sijaitsee Pennsylvaniassa, Yhdysvalloissa ja yritys työllistää yli 2000 työntekijää maailmanlaajuisesti. Qlikillä on yli 50 tuhatta asiakasta yli sadassa eri maassa ja yli 1700 yrityspartneria, kuten Cubiq. Yrityksen omistaa yksityinen pääomasijoitusyhtiö Thoma Bravo, joka omistaa myös useita muita teknologiayrityksiä [17]. Qlikin asiakkaita ovat muun muassa yritykset kuten Samsung, Lenovo ja Cisco [18]. Merkittäviä teknologiapartnereita ovat esimerkiksi Microsoft, SAP ja Snowflake [19]. Toisaalta Microsoftilla ja SAP:illa on myös omat data-analytiikkatuotteensa, joten ne ovat samaan aikaan myös Qlikin kilpailijoita. Muita merkittäviä kilpailijoita ovat esimerkiksi Tableau ja ThoughtSpot [16].

Qlikin yhtenä visiona on datalukutaitoinen maailma, jossa jokainen voi hyödyntää dataa ja analytiikkaa päätöksenteon parantamiseksi ja haastavimpienkin ongelmien ratkomisessa [20]. Datalukutaidolla tarkoitetaan esimerkiksi kykyä lukea, analysoida ja hyödyntää dataa sekä kommunikoida sen perusteella [21]. Qlik pyrkii tarjoamaan vision mahdollistamiseen tarvittavia työkaluja, joista kerrotaan lisää seuraavaksi.

4.2 Qlikin tuotteet

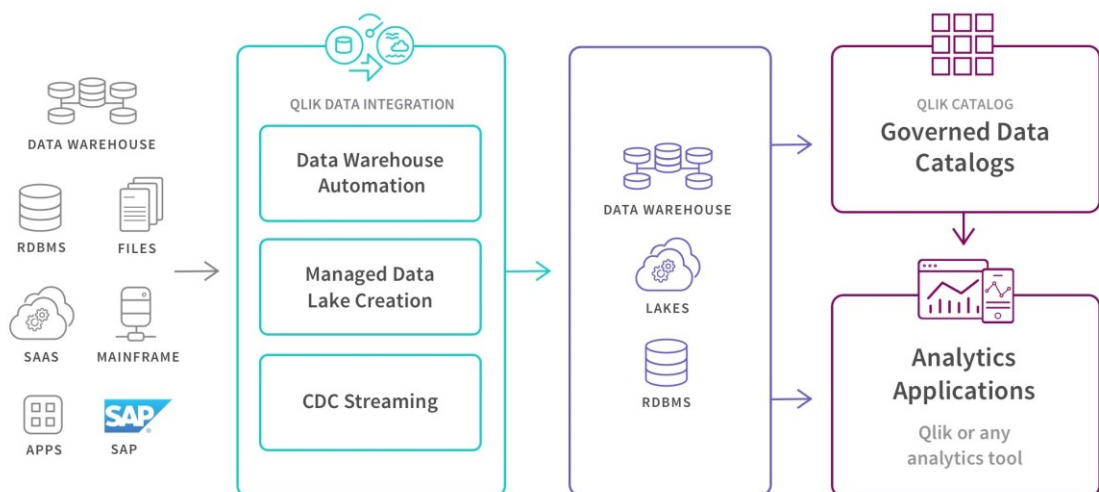
Qlikin tarjoamat ohjelmistotuotteet voidaan jakaa karkeasti kahteen eri osa-alueeseen, joita ovat dataintegraatio ja data-analytiikka.

Dataintegraatiotuotteet

Digitaalisen ajan johtamisen kannalta on tärkeää, että kaikilla liiketoiminnan henkilöillä on helppo pääsy uusimpaan ja tarkimpaan dataan [22]. Tätä varten Qlik tarjoaa niin sanottuja dataintegraatiotuotteita, joilla voidaan esimerkiksi automatisoida ja tehostaa:

- tietovarastointia
- datan replikointia
- datan jalostamista
- datan luettelointia
- datan julkaisua. [22.]

Qlikin dataintegraatiotuotteita ovat esimerkiksi Qlik Replicate, Qlik Compose ja Qlik Catalog. Tuotteiden avulla saadaan esimerkiksi jalostettua niin sanotusta ”raakadatasta” valmiimpaa, ”analytiikkakelpoista” dataa, jota voidaan samalla myös siirtää paikasta toiseen ja sitten hyödyntää erilaisilla data-analytiikkaohjelmistoilla, kuten Qlik Sensellä. Kuvassa 4 on esitelty yksinkertaistettu esimerkki datan kulusta lähdejärjestelmistä dataintegraatiotyökalujen kautta aina analytiikkaohjelmistoon saakka. [22.]



Kuva 4. Esimerkki dataintegraatiotuotteiden toiminnasta [22].

Kuvan 4 vasemmanpuoleisessa sarakkeessa on kuvattu organisaation dataa sisältäviä järjestelmiä, kuten tietovarastot, tietokannat, tiedostot, erilaiset sovellukset ja muut palvelut. Nämä järjestelmät voivat sijaita ympäri maailmaa ja olla täysin toisistaan erillisiä. Järjestelmien sisältämä data voi olla täysin jalostamatonta ja hyvinkin vaihtelevassa

muodossa eri järjestelmien välillä. Toisessa sarakkeessa on kuvattuna dataintegraatio-työkaluilla tehtäviä toimenpiteitä, joita voivat olla esimerkiksi tietovarastoautomaatio, data-altaan (*data lake*) hallittu luonti ja muuttuneen datan reaaliaikainen replikointi järjestelmästä toiseen. Kolmannessa sarakkeessa on kuvattu tilanne, jossa data on eri toimenpiteillä saatu jalostettua ”analytiikkakelpoisempaan” muotoon ja säilötyä kohdejärjestelmiin kuten tietovarastoon, data-altaisiin ja relaatiotietokantoihin. Viimeisen sarakkeen ylempi osio kuvastaa niin sanottua ”dataluettelo”, johon on koostettu valmiita ”datapaketteja” aiemmissa vaiheissa käsitellystä datasta. Ajatuksena on, että dataan päästään käsiksi yhdestä keskitetystä ja turvallisesta paikasta, jossa data on jalostettu analytiikkakelpoiseen muotoon ja se on organisoitu sekä dokumentoitu järkevästi. Dataa ei välttämättä ole replikoitu moneen paikkaan, mutta luettelon kautta tarvittava data saadaan ladattua sieltä, missä se sijaitseekaan. Lopulta data päätyy analysoitavaksi erilaisiin analytiikkasovelluksiin, kuten Qlik Senseen. [23.]

Data-analytiikkatuotteet

Organisaation muuntautumiseen datan avulla tarvitaan muutakin kuin pelkkä visualisointityökalu [24]. Qlikin data-analytiikkaohjelmistojen avulla käyttäjät pystyvät osaamisestaan riippumatta tutkimaan dataa vapaasti tekoälyn avustamana sekä tekemään parempiin lopputuloksiin johtavia ja viisaampia päätöksiä. Qlik tarjoaa data-analytiikkaan useita erilaisia ohjelmistoja, joilla on muun muassa seuraavia kyvykkyksiä:

- datan yhdistely, lataus ja visualisointi
- interaktiivisten sovellusten kehittäminen ja jakelu
- visualisointien upottaminen muihin järjestelmiin
- erilaisten staattisten raporttien luominen ja jakelu
- dataan perustuvien hälytysten lähettäminen
- paikkatiedon ja karttojen hyödyntäminen. [24.]

Qlikin tunnetuimmat ja tärkeimmät data-analytiikkaohjelmistot ovat QlikView ja Qlik Sense, joiden molempien taustalla toimii Qlikin assosiativinen moottori. Qlik Sense -ohjelmistosta ja assosiativisesta moottorista kerrotaan tarkemmin seuraavassa luvussa.

Edellä mainittujen ohjelmistojen lisäksi Qlikillä on myös muita arvoa lisääviä tuotteita, joilla voidaan muun muassa jakaa staattisia raportteja (Qlik NPrinting), lähettää dataan

perustuvia hälytyksiä (Qlik Alerting) sekä hyödyntää paikkatietoa ja karttoja monipuolisesti (Qlik GeoAnalytics).

5 Qlik Sense

Tässä työssä Qlik Sensellä tarkoitetaan Qlik Sense Enterprise on Windows -versiota. Qlik Sense on moderni ja responsiivinen web-pohjainen data-analytiikkaohjelmisto, jolla voidaan luoda näyttäviä ja interaktiivisia visualisointeja [25], joita voidaan myös upottaa muihin järjestelmiin [26]. Visualisoinnin lisäksi ohjelmistolla voidaan ladata dataa useista eri lähteistä ja dataa voidaan valmistella monipuolisesti, myös tekoälyä hyödyntäen [27].

5.1 Ohjelmisto

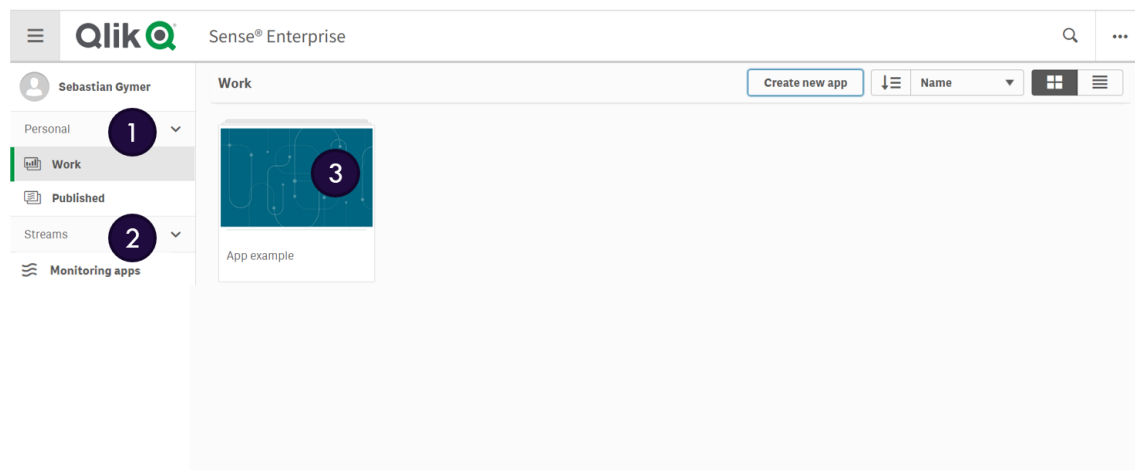
Qlik Sense -ohjelmisto koostuu useista Windows-palveluista, yhdestä keskitetystä PostgreSQL-tietokantasäilöstä sekä yhdestä jaetusta verkkolevykansioista [28]. Jaetussa verkkolevykansiossa säilytetään muun muassa Qlik Sense -sovelluksia, sovellusten sisältämiä kuvia ja kuvakkeita sekä lokitiedostoja. PostgreSQL-tietokannassa ylläpidetään lähes kaikkea ympäristöön liittyvää tietoa, kuten sitä missä sovellukset sijaitsevat verkkolevykansiossa ja kenellä käyttäjällä on oikeudet mihinkin sovellukseen. Qlik Sensen Windows-palveluista jokaisella on omat tehtävänsä. Eri palveluiden tehtäviä ovat esimerkiksi käyttäjien autentikointi, laskennan suorittaminen sovelluksilla tai ajastettujen tehtävien käynnistäminen. Palvelut voidaan asentaa yhdelle tai useammalle palvelimelle. Kaikkia palveluita tarvitaan, mutta joitain palveluita voidaan asentaa useita ja useille eri palvelimille. [29.] Palvelut toimivat yhteistyössä, ja ne kommunikoivat keskenään turvallisesti Transport Layer Security (TLS) -salausprotokollaa hyödyntäen [30].

Mitä suuremmat vaatimukset ympäristölle asetetaan, sitä useammalle palvelimelle palvelut yleensä jaetaan. Vaatimuksia ovat esimerkiksi sovellusten koko (datan määrä), sovellusten määrä sekä käyttäjien määrä. Erityisesti samanaikaisten käyttäjien määrällä on suuri vaikutus. [31.] Yksinkertaisimmassa Qlik Sense -ympäristössä kaikki palvelut, PostgreSQL-tietokanta sekä jaettu verkkolevykansio löytyvät kaikki yhdeltä palvelimelta. Yhden palvelimen ympäristössä kaikki vaatimukset asetetaan yhdelle palvelimelle, eikä kuormaa voida jakaa muiden palvelimien kanssa. Suositus on, että palvelin tai palvelimet on varattu Qlik Sensen käyttöön, sillä ohjelmisto hyödyntää palvelimen tai palvelinten laskentatehosta ja muistista tarvittaessa kaiken mahdollisen, esimerkiksi kun dataa la-

dataan sovelluksille, laskentoja suoritetaan tai sovelluksia ja laskentojen tuloksia säilytetään muistissa. Qlik Sense -ohjelmiston asennukselle on liuta muitakin järjestelmävaatimuksia, mutta ne eivät ole tämän työn kannalta olennaisia.

5.2 Käyttöliittymä

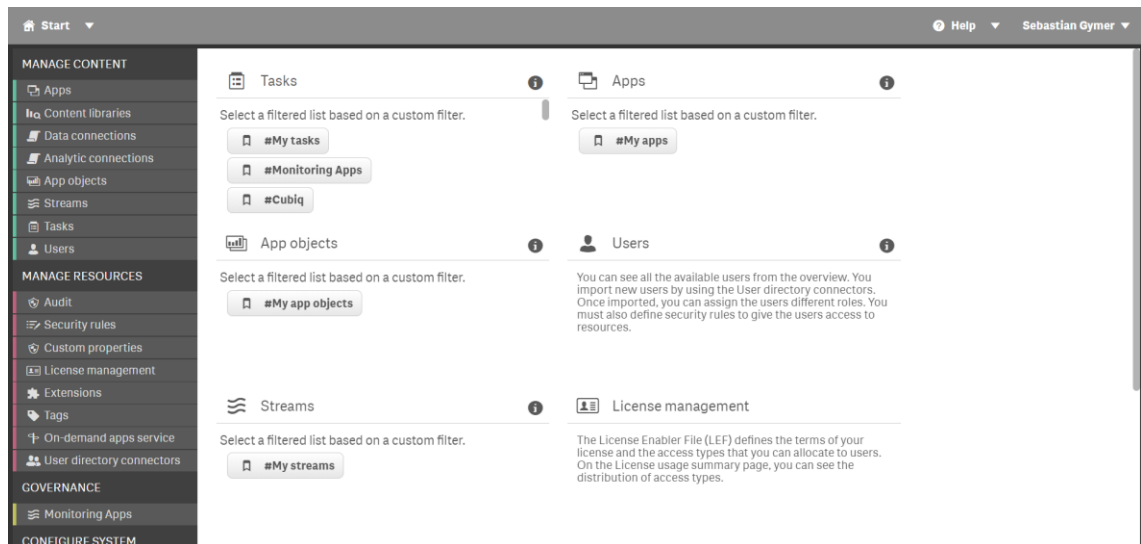
Qlik Senseä käytetään internetselaimella ja sitä voidaan käyttää useilla eri laitteilla, kuten tietokoneella, älypuhelimella tai tabletilla. Tämän työn kannalta oleellimmat näkymät ovat Qlik Sense Hub (Hub) ja Qlik Management Console (QMC). Qlik Sense -sovellus avataan Hubista, jossa käyttäjä näkee omat sovelluksensa, sekä ne muiden julkaisemat sovellukset, joihin hänellä on lukuoikeus. Kuvassa 5 esitellään Qlik Sense Hubia.



Kuva 5. Qlik Sense Hub -näkyvä.

Kuvassa 5 Hubin vasemmanpuoleinen valikko on jaettu Personal-osioon (kuva 5, kohta 1) ja Streams-osioon (kuva 5, kohta 2). Personal-osion alta löytyy Work-osio, jossa näkyvät käyttäjän omat julkaisemattomat sovellukset ja siellä voidaan kehittää sovelluksia. Streams-osion alla näkyvät ne streamit, joihin käyttäjällä on lukuoikeus. Streamit pitävät sisällään julkaistuja sovelluksia. Sovellus avataan painamalla sovelluksen kuvaketta (kuva 5, kohta 3).

QMC:n kautta hallitaan koko Qlik Sense -ympäristöä ja sen asetuksia. QMC sisältää useita eri osioita, joiden kautta voidaan esimerkiksi luoda ja muokata streameja, sovellusten lukuoikeuksiin liittyviä sääntöjä ja sovellusten ajastettuja lataustehtäviä. QMC:n näkymä on kuvattu kuvassa 6.



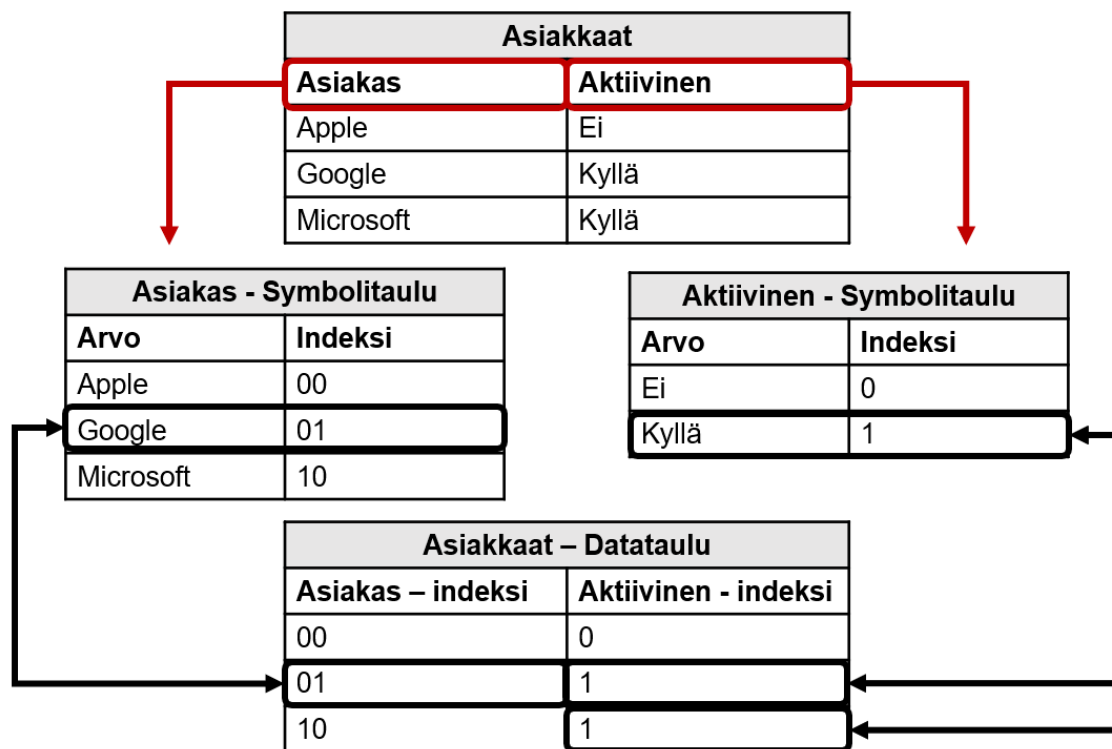
Kuva 6. Qlik Management Console -näkymä.

Kuvassa 6 näkyy QMC:n aloitussivu, jonka vasemmanpuoleisesta valikosta pääsee hallinnoimaan erilaisia asetuksia. Käyttöoikeuksista riippuen voi olla oikeus muokata kaikkia ympäristöön liittyviä asetuksia, osaa niistä tai ei mitään niistä. Tavallisilla käyttäjillä ei yleensä ole oikeutta mihinkään osioon QMC:ssa, jolloin valikon osiot näkyvät harmaina, eikä käyttäjä pääse niihin käsiksi. Täydet tai osittaiset oikeudet QMC:hen on yleensä muutamalla järjestelmänvalvojalla ja sovelluskehittäjällä.

5.3 Assosiativinen moottori

Ohjelmiston taustalla toimii Qlikin assosiativinen moottori, joka erottaa sen monista muista data-analytiikkaohjelmistoista, jotka ovat usein kyselypohjaisia. Qlikin assosiativisen moottorin tehokkuus perustuu muun muassa datan indeksointiin ja muistinvaraisuuteen. Muistinvaraisuudella tarkoitetaan, että sovelluksella oleva data säilytetään palvelimen RAM-muistissa. Indeksointi liittyy assosiativisen moottorin sisäisen tietokannan

tapaan säilyttää dataa symboli- ja datatauluissa. Kuvassa 7 on yksinkertaistettu esimerkki, jossa kuvitteelliselle Qlik Sense -sovellukselle on ladattu asiakastietoja sisältävä taulu, jossa on asiakkaan nimi ja tieto asiakkuuden tilasta.



Kuva 7. Datan säilytys assosiatiivisen moottorin sisäisessä, muistinvaraisessa tietokannassa [mukaillen 32].

Kuvassa 7 ylimpänä näkyy Qlik Sense -sovellukselle luettu Asiakkaat-taulu siinä muodossa kuin se näytetään sovelluksen käyttäjälle visualisoinneissa tai sovelluskehittäjälle datamallinäkyessä. Alimpana oleva Asiakkaat-datataulu kuvastaa sitä, miten Asiakkaat-taulu säilytetään moottorin muistissa. Sovelluksen skriptissä on määritelty Asiakkaat-taulu luettavaksi sisään sovellukselle esimerkiksi jostain tietokannasta, tiedostosta tai muusta lähteestä. Kun sovelluksen lataus valmistuu, eli kun skriptin suoritus päättyy, data- ja symbolitaulut luodaan automaattisesti assosiatiivisen moottorin sisäiseen tietokantaan. Sovellukselle luettujen taulujen jokaisesta kentästä muodostetaan omat symbolitaulut, jotka sisältävät kentän kaikki uniikit arvot kerran (esimerkiksi "Google") sekä sitä vastaavan indeksiluvun (esimerkiksi "01"). [32.]

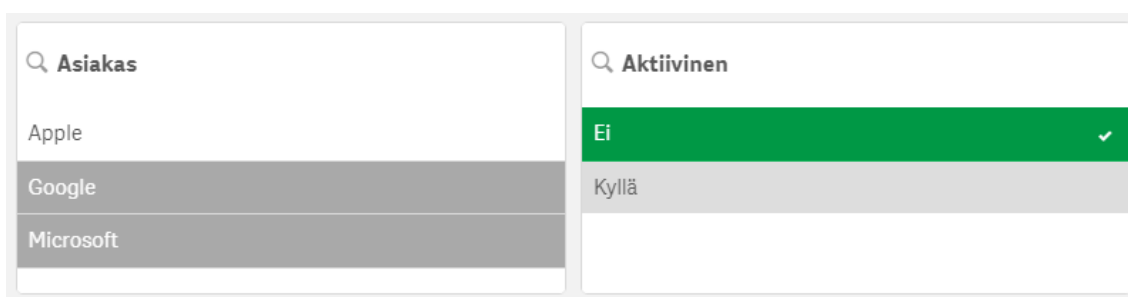
Datataulussa säilytetään vain indeksiluvut, jotka toimivat osoittimina symbolitauluihin. Yksi kenttä voi sisältää enintään 2147483648 (2^{31}) uniikkia arvoa [33]. Indeksit säilötään binäärijärjestelmän lukuna mahdollisimman vähillä biteillä esitettynä. Mitä useamman uniikin arvon kukin kenttä sisältää, sitä useammalla bitillä indeksi muodostetaan. Indeksointiin käytetään siis vain se määrä bittejä kuin tarvitsee, mikä tekee siitä mahdollisimman tehokasta (*bit-stuffed index*). Esimerkiksi Aktiivinen-kenttä sisältää vain arvot ”Kyllä” ja ”Ei”, joten tieto voidaan esittää yhdellä bitillä (0 tai 1). Tehokkuudella tarkoitetaan tässä tapauksessa esimerkiksi sitä, että käytetään mahdollisimman vähän palvelimen RAM-muistia arvojen säilömiseen. Tarvittavien bittien määrä voidaan ratkaista kaavasta:

$$2^n \geq x$$

x on kentän uniikkien arvojen määrä
n on tarvittavien bittien määrä kokonaislukuna

[32].

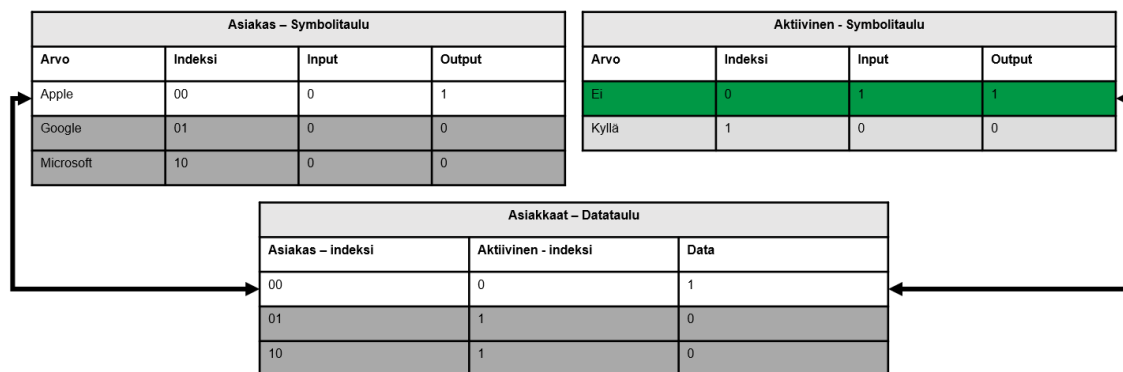
Qlikin assosiativinen moottori mahdollistaa suurten datamäärien tehokkaan käsittelyn, tutkimisen sekä erilaisten yhteyksien ja riippuvuuksien ymmärtämisen. Kyselypohjaiset ohjelmistot kykenevät vastaamaan vain ennalta määriteltyihin kysymyksiin (esimerkiksi tietty SQL-kysely), mutta kaikki muu data ”jää pimentoon”. Monilla ohjelmistoilla voidaan esimerkiksi löytää datasta yrityksen parhaiden myyneet tuotteet tai eniten ostaneet asiakkaat ja visualisoida tiedot näyttävästi, mutta Qlik Sensellä pystytään lisäksi löytämään myös ne tiedot, jotka jäisivät kyselyn ulkopuolelle. Kuvan 8 esimerkissä on käytetty edellisen esimerkin dataa havainnollistamaan tätä:



Kuva 8. Qlik Sense -sovelluksella olevat kaksi suodatinobjektia (*filter pane*).

Vasemmanpuoleisella suodattimella (kuva 8) voidaan suodattaa sovelluksella näytettäviä tietoja Asiakas-dimension perusteella. Oikeanpuoleinen on suodatin Aktiivinen-dimensiolle. Käyttäjän tekemä suodatusvalinta (Aktiivinen = ”Ei”) näkyy vihreällä pohjalla. Valintaan liittyvät mahdolliset Asiakas-kentän arvot näkyvät valkoisella pohjalla (”Apple”) ja valinnan ulkopuolelle rajautuvat arvot tumman harmaalla pohjalla (”Google”, ”Microsoft”). Vaalean harmaalla pohjalla näkyvät niin sanotut vaihtoehdot, eli arvot, jotka rajautuvat mahdollisten arvojen ulkopuolelle vain samaan kenttään tehtyjen valintojen perusteella (”Kyllä”). [32.] Datan analysointi eri näkökulmista on nopeaa ja helppoa ja kiinnostavia asioita ja yhteyksiä voi usein löytää jopa hieman yllättäen. Tietysti esimerkiksi SQL-kyselyllä voidaan yhtä lailla hakea niiden asiakkaiden tiedot, jotka ovat aktiivisia, mutta kysely tuottaa vastauksen vain juuri tuohon, eikä ulkopuolelle jäävää tietoa päästä tutkimaan yhtä helposti. Kyselypohjaisuus aiheuttaa sen, että kyselyitä joudutaan aina muokkaamaan uudestaan ja vastausta joudutaan odottamaan – puhutaan niin sanotusta ”kysy, odota, vastaa” -syklistä (*ask, wait, answer cycle*). Qlik Sensellä dataa pystytään tutkimaan ja analysoimaan ketterästi ja kysymystä voidaan muokata nopeasti, eikä vastausta tarvitse odottaa kovin kauaa. [34.]

Edellisen esimerkin valintaa tehdessä Qlik Sensen moottori lisäsi tauluihin tietoa kunkin taulun rivin tilasta (*state vectors*) [32]. Kuvassa 9 on havainnollistettu tätä ja selkeyden vuoksi käytetty myös samoja värejä kuin edellisessä esimerkissä.



Kuva 9. Symboli- ja datataulut, joihin moottori on lisännyt tietoa rivin tilasta [mukaillen 32].

Kuvasta 9 nähdään, että symbolitaulun Input-kenttään kirjoitetaan arvo 1, jos käyttäjä on valinnut kyseisen rivin arvon ja 0 jos ei. Output-kentän arvo 1 kertoo, että kentän arvo

yhdistyy käyttäjän tekemiin valintoihin, kun taas 0 tarkoittaa, että arvo rajautuu pois. Arvoja hyödynnetään suodatinobjektien kenttien arvojen värjäämisessä (ks. kuva 8) ja niiden perusteella määrittyy lopulta myös, mitkä datataulun rivit huomioidaan laskennoissa. Tuo ilmaistaan datataulussa Data-kentän arvoilla 1 (huomioidaan) ja 0 (ei huomioida). Esimerkiksi, jos sovellus sisältäisi lisäksi tietoa asiakkuuteen kohdistuvista töistä, niin kyseisellä valinnalla kaikki visualisoinnit päivittyisivät kuvastamaan niitä asiakkaita, joiden Aktiivinen-kentän arvo on ”Ei”. Moottori päättelee Input-, Output- ja Data-kentille oikeat arvot heti, kun käyttäjä on tehnyt valintoja ja laskee sen jälkeen kaikkiin visualisointeihin oikeat arvot uudestaan. Vasteajan parantamiseksi laskentojen tuloksia säilytetään tietyn aikaa välimuistissa, eli jos hetken päästä tehdään sama valinta uudestaan, niin laskennan tulokset haetaankin välimuistista, eikä niitä lasketa uudestaan. [32.]

Cubiqin tapauksessa saatetaan esimerkiksi valita Viikko-kentästä edellinen täysi viikko, kun halutaan tarkistaa, ovatko kaikki työntekijät jo kirjanneet tekemänsä työt toiminnanohjausjärjestelmään. Tiedon perusteella osataan esimerkiksi odottaa, kunnes kaikki työntekijät ovat kirjauksensa tehneet ennen kuin tehdyistä töistä lähetetään lasku asiakkaalle. Toisaalta valitsemalla ne työntekijät, joilta kirjauksia puuttuu, voidaan myös saada listaus henkilöistä, joita tulisi kehottaa tekemään kirjauksensa. Olisi myös mahdollista automatisoida tämä vaihe ja vähentää manuaalisen työn määrää entisestään.

Yhtenä osana tämän opinnäytetyön käytännön työvaiheista on datan mallintaminen, ja edellä mainittujen esimerkkien ymmärtäminen auttaa siinä. Esimerkiksi millisekunnin tarkkuudella olevia aikaleimoja ei kannata turhaan käyttää, jos tietoja analysoidaan vain päivämäärän tarkkuudella. Päivämäärän ja kellonajan tai henkilön etunimen ja sukunimen jakaminen omiin kenttiinsä voi jo parantaa sovelluksen suorituskykyä ja sen vaatimaa muistin määrää (vähemmän uniikkeja arvoja). Tämän työn varsinaisena tavoitteena ei kuitenkaan ole Qlik Sense -sovelluksen tai sen sisältämän datamallin suorituskyvyn optimointi, vaan tiedon oikeellisuuden ja ajantasaisuuden varmistaminen. Optimoimatonkin sovellus voi toimia täysin hyvin, jos data on mallinnettu muuten järkevästi eli oikeat rivit yhdistyvät toisiinsa. Optimointitarpeet kasvavat yleensä datan määrän ollessa todella suurta, sillä samalla kasvavat myös palvelimen laskentatehon ja muistin tarve. Edellä mainitut esimerkit eivät vielä täysin kuvaa Qlikin assosiativisen moottorin toimintaa, mutta todennäköisesti kuitenkin riittävällä tarkkuudella, jotta seuraavassa Qlik

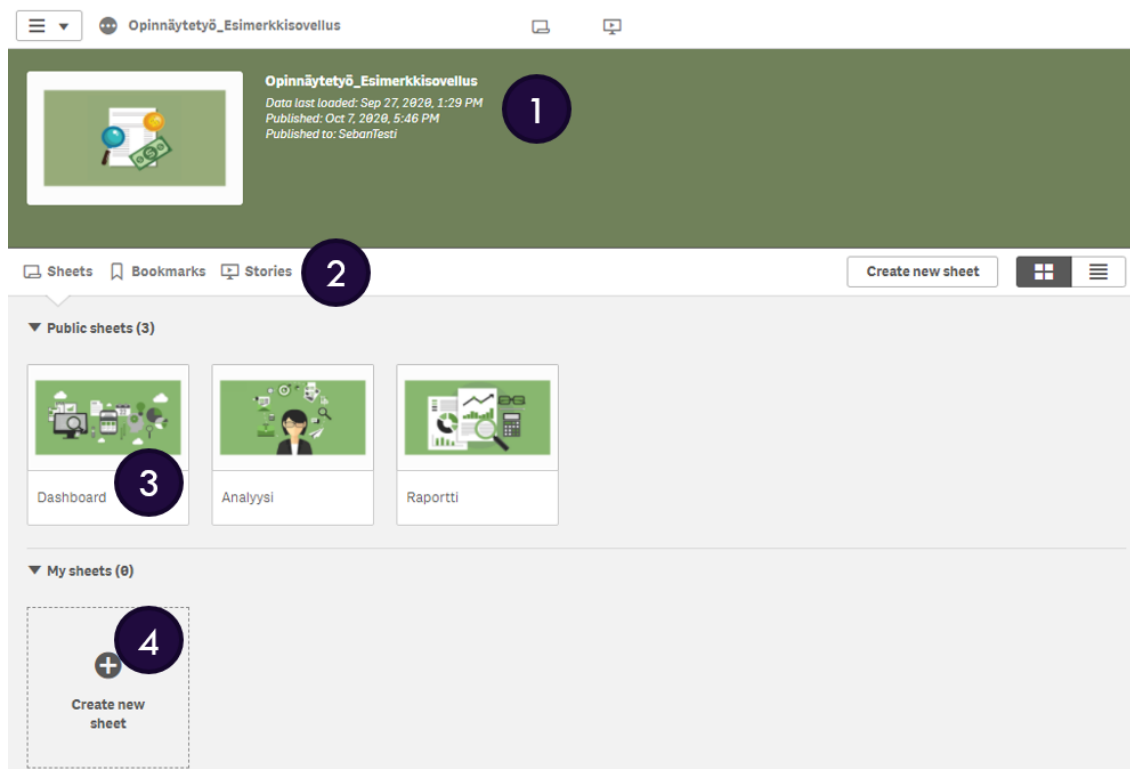
Sense -sovelluksia käsittelevässä osassa pystyy hahmottamaan, mitä taustalla tapahtuu.

5.4 Sovellukset

Esittelen ensin Qlik Sense -sovelluksen loppukäyttäjän näkökulmasta ja sitten sovelluskehittäjän näkökulmasta. Tarkoitukseni ei ole eritellä kaikkia ominaisuuksia, vaan antaa käsitys Qlik Sense -sovellusten käyttämisestä ja niiden kehittämisestä. Pyrin esimerkkien kautta tuomaan ilmi erityisesti sitä, kuinka monella eri tapaa sovelluskehittäjä voi vaikuttaa sovelluksella näkyvään tietoon, jota loppukäyttäjä hyödyntää. Toisaalta näiden asioiden erittely ja ymmärtäminen ovat edellytyksiä työn myöhemmille vaiheille, joissa parannetaan Cubiqin sovellusten tietojen luotettavuutta. Tämän työn kannalta ei ole oleellista, missä ja miten sovellusta teknisesti ottaen säilytetään palvelimella, joten en esittele näitä asioita.

5.4.1 Loppukäyttäjän näkökulma

Loppukäyttäjän näkökulmasta Qlik Sense -sovellus on yksinkertaisimmillaan vain kokonaisuus erilaisia interaktiivisia visualisointeja, kuten esimerkiksi pylväs- ja viivadiagrammeja, taulukoita ja avainlukumittareita. Sovellus sisältää yhdestä tai useammasta lähteestä ladattua dataa, josta on sovelluskehittäjän toimesta muodostettu datamalli, jossa oikeat tiedot yhdistyvät toisiinsa. Jos sovellukseen ladataan taustalla uutta dataa, visualisoinnit päivittyvät vastaamaan uutta sisältöä. Sovelluksen loppukäyttäjä ei yleensä pääse näkemään sovelluksen taustalle rakennettua datamallia tai latausskriptiä ollenkaan, saati ymmärtää niiden sisältöä tarkemmin. Kuvissa 11–14 esitellään esimerkkitsovellusta, joka sisältää tietoa kuvitteellisen yrityksen työntekijöiden kirjaamista työtunneista, yrityksen asiakkaista ja projekteista. Sovellus avataan Hubista sovelluskuvaketta painamalla, jolloin käyttäjälle avautuu ensin App Overview -näkyvä.

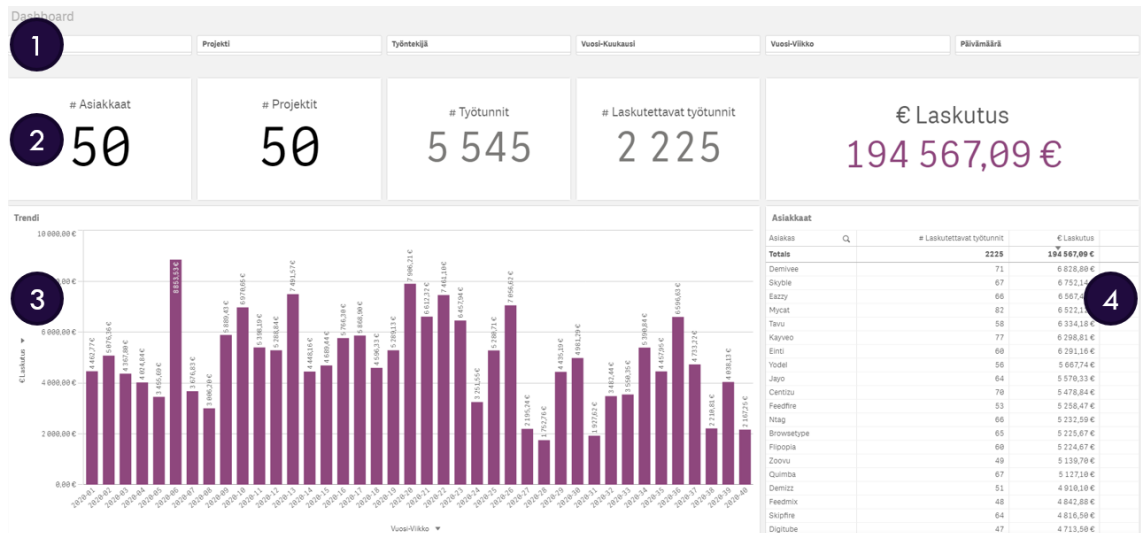


Kuva 10. Esimerkkisovelluksen App Overview -näkyvä.

Kuvan 10 App Overview -näkyvässä käyttäjä näkee sovelluksen perustiedot, kuten:

- 1 = sovelluksen nimen ja kuvauksen, viimeisimmän datalatauksen ajankohdan, sovelluksen julkaisuajankohdan ja streamin, johon sovellus on julkaistu
- 2 = valikon, josta voi valita tarkasteleeko sovelluksessa olevia välilehtiä, kirjanmerkkejä vai tarinoita
- 3 = sovelluksen välilehdet, joihin käyttäjällä on lukuoikeus
- 4 = käyttäjän lisenssistä ja käyttöoikeuksista riippuen käyttäjällä voi olla mahdollisuus luoda sovellukselle omia välilehtiä, jotka ei näy muille käyttäjille.

Kuvassa 11 on esitelty saman sovelluksen Dashboard-välilehti.



Kuva 11. Dashboard-välilehden näkymä.

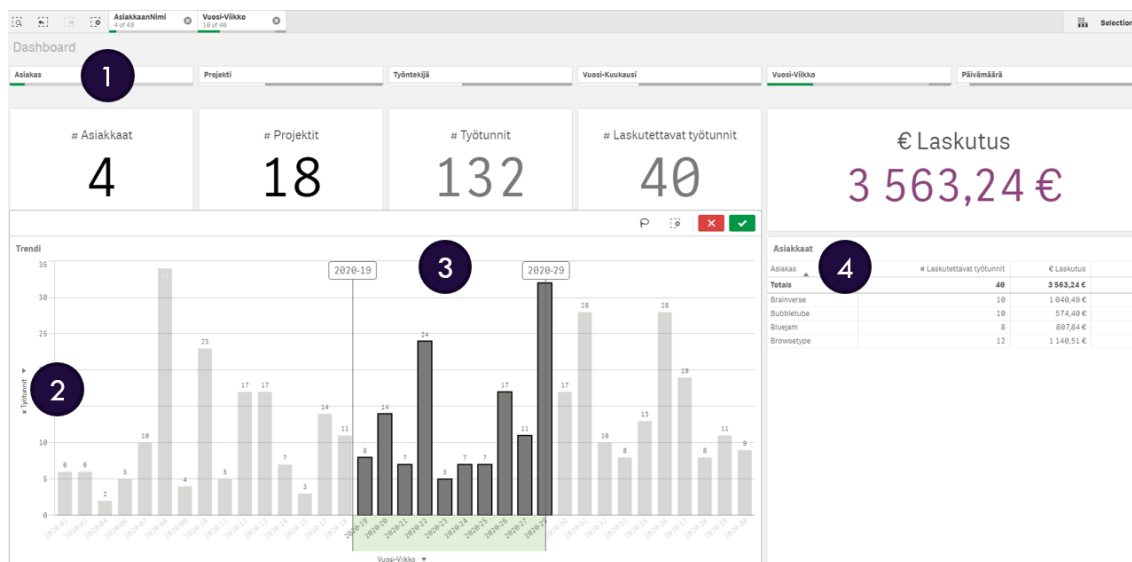
Kuvan 11 Dashboard-välilehti sisältää:

- 1 = kuusi suodatinta (*filter pane*)
- 2 = viisi avainlukumittaria (*KPI*)
- 3 = yhden pylväsdiagrammin (*bar chart*)
- 4 = yhden taulukon (*table*).

Kaikki visualisoinnit ovat interaktiivisia, eli käyttäjä voi klikata niitä, ja sovelluksen näkymä päivittyy klikkausten mukaan. Interaktiivisuus voi olla esimerkiksi seuraavanlaista:

- Käyttäjä voi valita suodatimesta dimensiot, joita haluaa tarkastella, esimerkiksi tietyt asiakkaat tai tietyn projektin.
- Käyttäjä voi klikata pylväsdiagrammin eri osia, esimerkiksi pylväsdiagrammin pylväitä, ja valita esimerkiksi tietyt viikot.
- Käyttäjä voi vaihtaa visualisoinneissa tarkasteltavia asioita (dimensioita) kaavion reunalta löytyvistä valikoista.
- Käyttäjä voi järjestellä taulukon rivit tietyn sarakkeen mukaan nousevasti tai laskevasti.

Kuvassa 12 on esimerkki samasta välilehdestä, kun käyttäjä on klikannut visualisoinneissa oleviin osiin, eli tehnyt tiettyjä valintoja ja vaihtanut visualisoinneissa käytettäviä dimensioita.



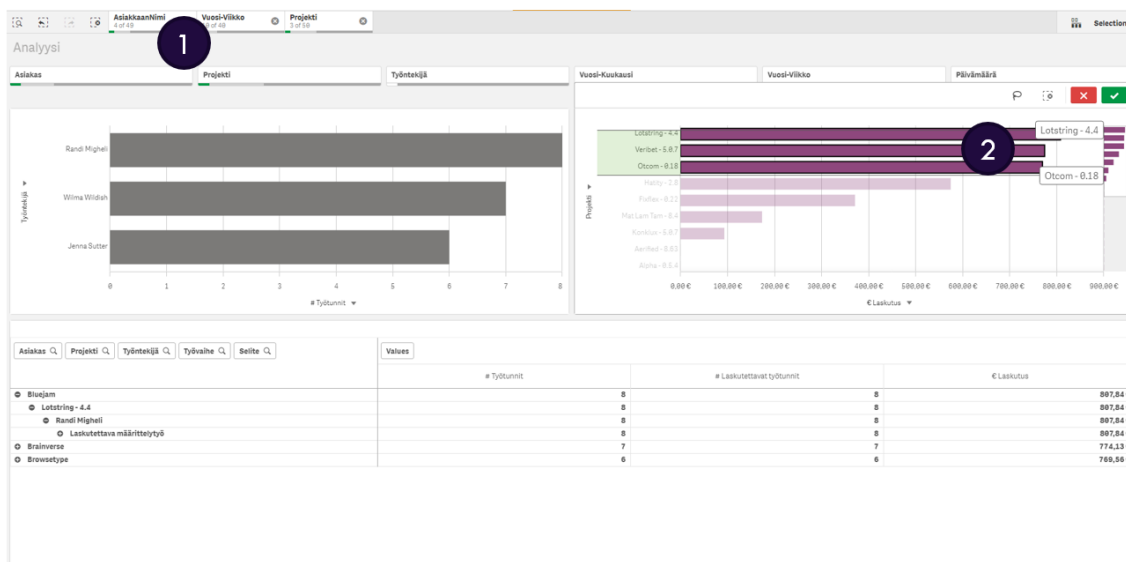
Kuva 12. Dashboard-välilehden näkymä, kun käyttäjä on tehnyt valintoja, sekä vaihtanut visualisoinneissa näytettäviä dimensioita.

Kuvassa 12 on sama Dashboard-välilehti kuin aiemmassakin esimerkissä, mutta käyttäjä on hyödyntänyt visualisointien interaktiivisuutta seuraavasti:

- 1 = Käyttäjä on valinnut Asiakas-suodattimesta neljä asiakasta tarkempaan tarkasteluun.
- 2 = Käyttäjä on vaihtanut pylväsdiagrammin y-akselille työtunnit laskutettujen eurojen sijaan.
- 3 = Käyttäjä on valinnut pylväsdiagrammista tietyt 10 viikkoa.
- 4 = Käyttäjä on järjestänyt taulukon rivit asiakkaan nimen mukaan aakkosjärjestykseen.

Kuvasta 12 voidaan huomata, että kaikki visualisoinnit ovat päivittyneet edustamaan käyttäjän tekemiä valintoja. Avainlukumittarien näyttämät luvut ovat päivittyneet, pylväsdiagrammi on päivittynyt, taulukossa näkyvät nyt vain valittujen neljän asiakkuuden tiedot, ja yläreunan valikkoon on ilmestynyt näkyviin käyttäjän tekemät valinnat. Qlik Senseen assosiatiivinen moottori on jokaisen valinnan jälkeen laskenut taustalla kaikki laskennat uudestaan vastaamaan käyttäjän valitsemaa osaa datasta [34, s. 5; luku 5.3]. Tämän esimerkin tavoitteena on kuvastaa sitä, että käyttäjä pystyy jo yhden välilehden sisällä analysoimaan dataa hyvinkin monipuolisesti ja näkemään tekemiensä valintojen vaikutukset heti sekä tutkimaan juuri sitä osaa datasta, joka häntä kiinnostaa.

Kaikki käyttäjän tekemät valinnat vaikuttavat kaikkiin sovelluksella näkyviin visualisointeihin myös muilla välilehdillä. Käyttäjä voi siis tehdä esimerkiksi edellä mainitut valinnat Dashboard-välilehdellä ja siirtyä sitten toiselle välilehdelle tarkastelemaan tietoja erilaisien visualisointien avulla. Tästä on esimerkki kuvassa 13.



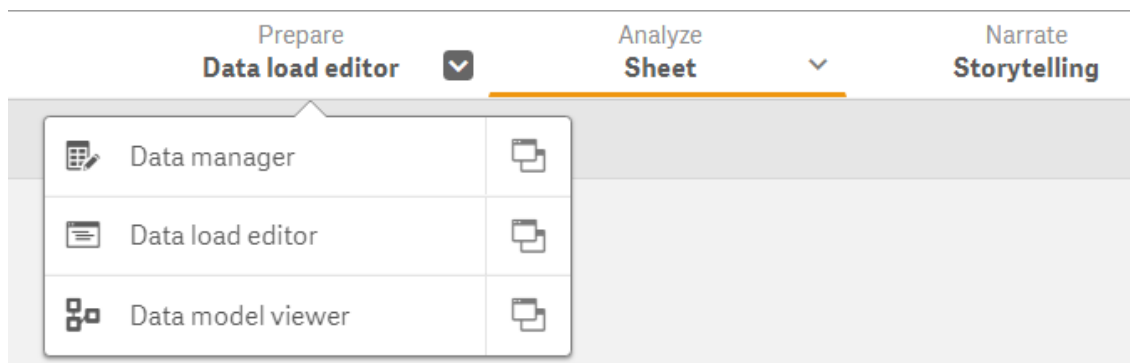
Kuva 13. Analyysi-välilehden näkymä.

Kuvassa 13 näkyy, että käyttäjän Dashboard-välilehdellä tekemät valinnat ovat säilyneet (kuva 13, kohta 1). Lisäksi käyttäjä on päättänyt valita Analyysi-välilehden toisesta pylväsdiagrammista tietyt kolme projektia tarkempaan tarkasteluun (kuva 13, kohta 2). Kaikki visualisoinnit edustavat nyt sitä osaa datasta, josta on valittu tietyt neljä asiakasta, tietyt 10 viikkoa sekä tietyt kolme projektia. Käyttäjä voi nyt tarkastella valintoihin liittyviä tietoja vielä tarkemmin esimerkiksi alaosan pivot-taulukossa.

Käyttäjän oikeuksista riippuen hänellä voi myös olla oikeus luoda itse uusia välilehtiä sovellukselle käyttäen sovelluksen taustalla olevaa dataa ja datamallia sekä sovellukselle määritettyä Master items -kirjastoa. Master items -kirjastosta kerrotaan tarkemmin seuraavassa osiossa, jossa esitellään sovellusta ja sen kehittämistä sovelluskehittäjän näkökulmasta.

5.4.2 Sovelluskehittäjän näkökulma

Sovelluskehittäjän näkökulmasta yksi Qlik Sense -sovellus sisältää huomattavasti enemmän asioita kuin mitä loppukäyttäjä näkee. Loppukäyttäjää kiinnostaa yleensä enimmäkseen vain visualisoinnit eli sovelluksen *Analyze*-osio, kun taas sovelluskehittäjää kiinnostaa sen lisäksi erityisesti datan valmisteluun liittyvä *Prepare*-osio, joka ei julkaistujen sovellusten loppukäyttäjille yleensä näy.



Kuva 14. Qlik Sense -sovelluksen eri osiot, joista *Prepare*-osion valikko on laajennettu näkyviin.

Kuvassa 14 näkyvistä osioista latausskriptinäköymä (*Data load editor*) ja datamallinäköymä (*Data model viewer*) ovat visualisointien lisäksi sovelluskehittäjän tärkeimmät työkalut. Tekoälyä hyödyntävä *Data manager* -osio on käytännössä visuaalinen käyttöliittymä latausskriptin muodostamiseen ja datamallin rakentamiseen, mutta tässä työssä ei keskitytä siihen. Sovellusten kehittäminen on yleensä iteroivaa työtä, eli niitä kehitetään vaiheissa ja parannellaan hiljalleen, kunnes päästään haluttuun lopputulokseen. Yleensä prosessin ensimmäinen vaihe on latausskriptin kirjoittaminen *Data load editor* -näkylässä.

Latausskripti

Latausskriptissä käytännössä kerrotaan, mitä dataa halutaan ladata ja mistä, sekä miten dataa tulisi käsitellä, kuten esimerkkikoodissa 1.

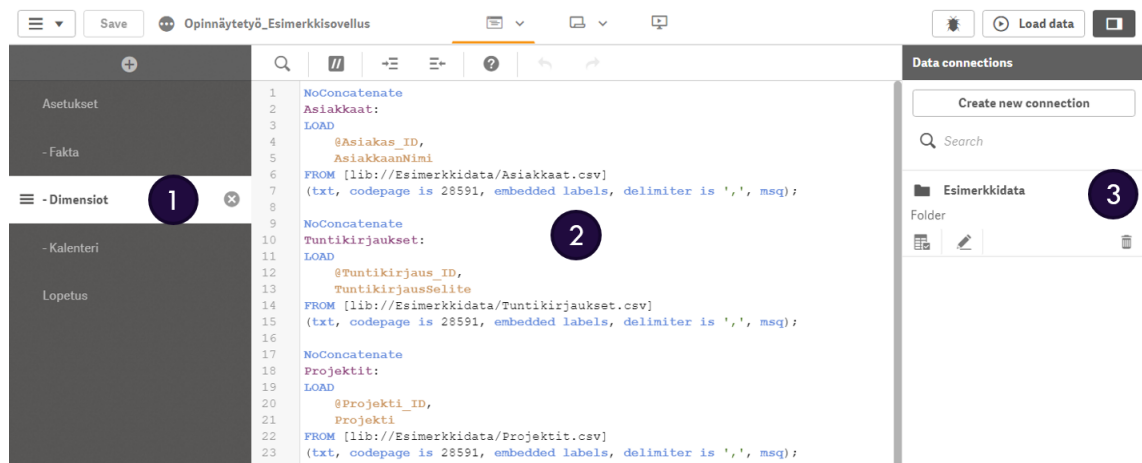
```

Fakta:
LOAD
    Tuntikirjaus_ID,
    Työntekijä_ID,
    Asiakas_ID,
    Tuntimäärä,
    Päivämäärä
FROM [lib://Esimerkkidata/Tuntikirjaukset.qvd] (qvd);

```

Esimerkkikoodi 1. Ote esimerkkisovelluksen latausskriptistä ja datan lukemisesta sovellukselle.

Esimerkkikoodi 1:ssä ”Fakta”-nimiseen tauluun luetaan viisi kenttää erillisestä ”Tuntikirjaukset.qvd”-tiedostosta hyödyntäen Esimerkkidata-nimistä datayhteyttä. Esimerkissä kentät luetaan lähdetiedostosta sellaisinaan, eli esimerkiksi kenttien nimiä ei muuteta. Latausskriptinäkömä on esitelty kuvassa 15.



Kuva 15. Qlik Sense -sovelluksen latausskriptinäkömä (*Data load editor*) ja sen eri osat.

Kuvan 15 latausskriptinäkömän vasemmassa reunassa näkyvät latausskriptin sisältämät osiot (kuva 15, kohta 1). Skriptin (kuva 15, kohta 2) voi jakaa useaan osioon ja kokonaisuudessaan skripti voi olla satoja tai tuhansia rivejä pitkä. Yhden latausskriptiosion suositellaan olevan enintään 50 000 merkkiä pitkä [33]. Eri datalähteisiin yhdistämiseen käytetään oikeassa reunassa näkyviä datayhteyksiä (kuva 15, kohta 3), jotka määritetään erikseen. Datayhteys voidaan määritellä viittaamaan esimerkiksi tiettyyn kansioon Qlik Sense -palvelimella, johonkin tietokantaan (kuten ValueFramen tietokanta) tai muuhun

lähteeseen. Usealla eri sovelluksella voidaan käyttää samaa datayhteyttä. Lisäksi skriptissä voidaan määritellä käyttäjille ja käyttäjäryhmille oikeudet tiettyyn osaan datasta muodostamalla *Section Access* -osion [35]. Skriptissä voidaan myös määritellä muuttujia, joita voidaan käyttää apuna skriptin logiikan rakentamisessa tai visualisoinneissa. Kun uusi sovellus luodaan, niin se sisältää automaattisesti tiettyjen muuttujien määrittäykset, kuten esimerkkikoodissa 2.

```
SET MoneyFormat='# ##0,00 €;-# ##0,00 €';
SET TimeFormat='h.mm.ss';
SET DateFormat='D.M.YYYY';
```

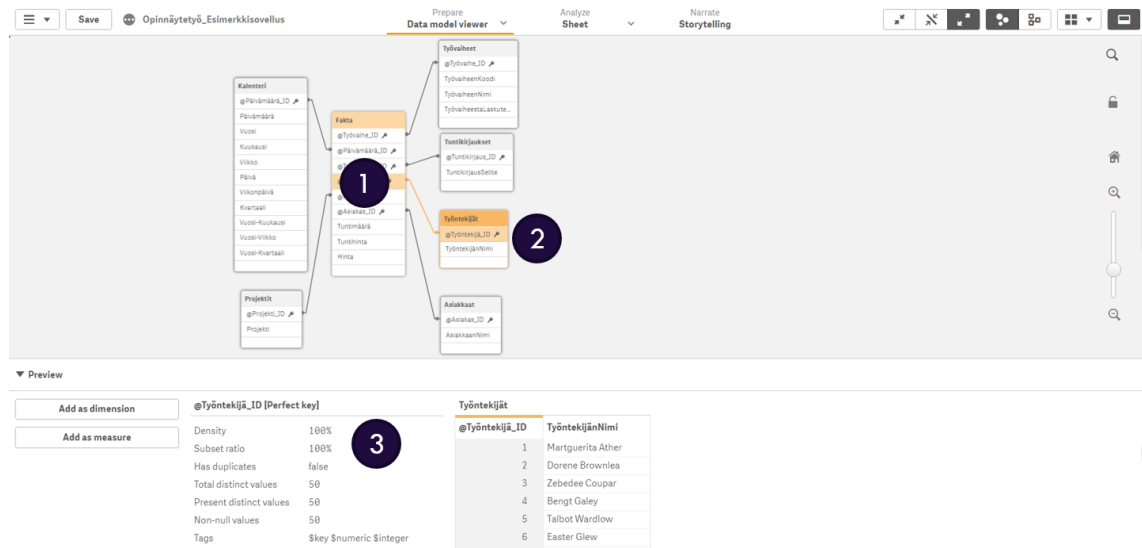
Esimerkkikoodi 2. Ote esimerkisovelluksen latausskriptistä ja muuttujien määrittämisestä.

Esimerkkikoodissa 2 esiteltyjen muuttujien oletusarvot määrittyvät palvelimen käyttöjärjestelmän mukaan. Kyseisiä muuttujia hyödynnetään esimerkiksi lukujen ja päivämäärien näyttämässä tietyssä muodossa. Jos sovelluksella ei määritellä niin sanottuja perusmuuttujia, eli määrittäykset poistetaan skriptistä, niin sovellus käyttää oletusarvoja. [36.] Näiden muuttujien arvoja voidaan muokata omiin tarkoituksiin sopiviksi tai rinnalle voidaan lisätä uusia. Muuttujia voidaan määritellä latausskriptin lisäksi myös sovelluksen *Analyze*-osiossa.

Datan mallintaminen

Sovelluskehittäjän ehkä tärkeimpänä tehtävänä on varmistaa, että loppukäyttäjä saa sovellukselta oikeaa tietoa, jota hän voi hyödyntää päätöksenteon tukena. Käytännössä tämä tarkoittaa sitä, että data on mallinnettava oikein eli se sisältää tarvittavat tiedot, jotka myös yhdistyvät toisiinsa oikein. Tietyt kentät yhdistetään toisiinsa avainkentillä, jotka muodostetaan yksinkertaisesti nimeämällä eri taulujen toisiaan vastaavat kentät samalla nimellä, esimerkiksi "@Asiakas_ID". Avainkentissä käytetään usein etumerkinä esimerkiksi @-merkkiä, %-merkkiä tai jotain muuta erikoismerkkiä. Kentät voidaan piilottaa loppukäyttäjältä asettamalla *HidePrefix*-muuttujan arvoksi käytetty etumerkki. Avainkenttien käyttäminen laskennoissa voi tuottaa joissain tapauksissa vääriä tuloksia, joten ne yleensä piilotetaan loppukäyttäjältä. Latausskriptillä määritellään, miltä tuleva datamalli tulee näyttämään. Mikä on oikein kussakin tilanteessa voi vaihdella, ja samaan

lopputulokseen voi päästä monella eri tavalla. Latausskriptin suorittamisen jälkeen (kun data on ladattu sovellukselle) sovelluskehittäjä voi tutkia syntynyttä datamallia datamallinäkössä (*Data model viewer*), joka on esitelty kuvassa 16.

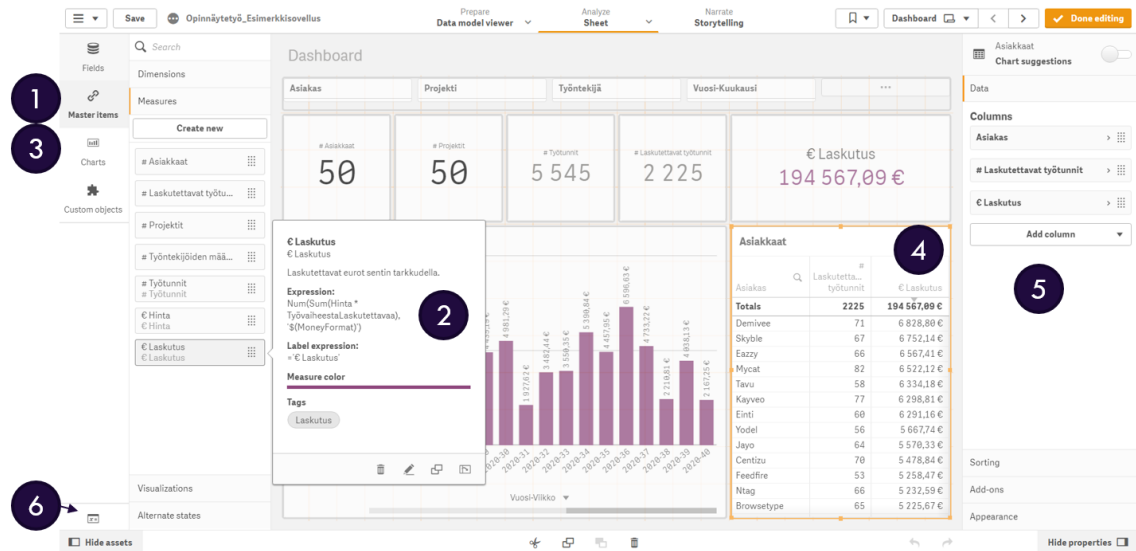


Kuva 16. Qlik Sense -sovelluksen datamallinäkö (Data model viewer) ja sen eri osat.

Kuvan 16 datamallinäkössä nähdään sovellukselle luetut datataulut ja näiden väliset yhteydet (kuva 16, kohdat 1 ja 2). Taulujen sisältöä ja sisältöön liittyvää kuvailevaa tietoa (metadataa) voidaan tutkia *Preview*-osiossa (kuva 16, kohta 3). Datamallin muodostamisessa parhaaseen lopputulokseen päästään yleensä pyrkimällä niin sanottuun ”tähtimalliin”, kuten kuvassa 16. Tähtimallissa pyritään kasaamaan kaikki kvantitatiivinen data, eli mitattavat asiat kuten tuntimäärä ja hinta, yhteen tai useampaan niin sanottuun faktatauluun. Faktataulu sisältää lisäksi avaimet ympärillä oleviin dimensiotauluihin, joihin kasataan kvalitatiivinen data, eli kuvailevat tiedot kuten työntekijän nimi. Optimaalisessa tilanteessa dimensiotaulusta löytyy kaikki faktataulusta löytyvät avainkentän arvot, eikä dimensiotaulun avainkentässä ole samaa arvoa usealla eri rivillä. Näin jokainen faktataulun rivi yhdistyy yhteen ja vain yhteen dimensiotaulun riviin. [32.] *Preview*-näkömään tiedot auttavat tuohon pyrkimisessä ja ongelmien vianmäärittämisessä. Kuvassa 16 korostettu kenttä ”@Työntekijä_ID” on niin sanottu täydellinen avain (*Perfect key*), sillä se täyttää edellä mainitut ehdot. Kuten luvun 3 teoriaosuudessa todettiin, niin datan valmistelu ja mallintaminen on usein aikaa vievin vaihe ja on myös todennäköistä, että latausskripti- ja datamallinäkömien välillä joudutaan siirtymään edes takaisin useita kertoja sovellusta kehitettäessä.

Visualisoinnit, Master items -kirjasto ja muuttujat

Edellä mainittujen näkymien lisäksi sovelluskehittäjällä on tärkeitä tehtäviä myös *Analyze*-osion välilehtinäköymässä (*Sheet*), jossa määritellään muun muassa sovelluksen visualisoinnit, *Master items* -kirjaston sisältö ja muuttujat.



Kuva 17. Qlik Sense -sovelluksen *Analyze*-osion välilehtinäköymä sovelluskehittäjän näkökulmasta.

Master items -kirjasto (kuva 17, kohta 1) on valikoima ennalta määriteltäviä mittareita (*Measure*), dimensioita (*Dimension*), visualisointeja (*Visualization*) ja vaihtoehtoisia tiloja (*Alternate state*). Kirjastoon määritely mittari (kuva 17, kohta 2) voi olla esimerkiksi ”€ Laskutus”, jolle voidaan määrittää nimen lisäksi laskentakaava (*Expression*), kuvaus (*Description*), nimike (*Label expression*) ja tarvittaessa myös väri ja tunnisteet (*Tags*). Dimensio voi olla esimerkiksi ”Asiakas” eli kuvailevaa tietoa. Näitä voidaan hyödyntää esimerkiksi visualisoinnissa, jossa esitetään laskutus asiakkaittain. Master items -kirjaston rakentaminen on yksi sovelluskehittäjän tehtävistä. Kirjaston tarkoitus on, että mittareita ja visualisointeja hallitaan keskitetysti. Kun kirjastossa olevan mittarin määrittystä muutetaan, uusi määrittely päivittyy automaattisesti kaikille visualisoinneille, joissa sitä on käytetty. Toisaalta myös loppukäyttäjät pystyvät hyödyntämään kirjaston objekteja omissa visualisoinneissaan, jos heillä on oikeus luoda sovellukselle omia näkymiä.

Visualisoinnit lisätään sovellukselle valitsemalla ne vasemman puolen valikon kohdasta *Charts* (kuva 17, kohdat 3 ja 4) ja raahaamalla ne sopivaan kohtaan. Visualisoinnin sisältämää tietoa ja ulkoasua hallitaan oikean puolen valikosta (kuva 17, kohta 5). Visualisoinneissa voi ja kannattaa käyttää Master items -kirjastoon määritettyjä valmiita mittareita ja dimensioita. Esimerkiksi kuvan 17 kohdassa 4 on taulukko, jossa on dimensiona ”Asiakas” ja mittareina ”Laskutettavat työtunnit” sekä ”€ Laskutus”.

Lisäksi sovelluskehittäjä voi määritellä muuttujia (kuva 17, kohta 6), joita voi hyödyntää esimerkiksi Master items -kirjaston mittareissa ja dimensioissa, tai suoraan visualisoinneissa. Muuttujia voi sisältää esimerkiksi tietyn laskentakaavan tai osan siitä. Muuttujien ja Master items -kirjaston hyödyntäminen vähentää sovellusten ylläpitoon kuluvaa aikaa ja vaivaa, sillä tietoa hallitaan keskitetympin. Esimerkiksi, jos tietyn mittarin laskentakaavaa halutaan muuttaa, niin sitä ei tarvitse muuttaa jokaiselle visualisoinnille erikseen.

Tässä luvussa esitettyjen esimerkkien kautta on saatu muodostettua kattava näkemys Qlik Sense -sovelluksista ja erityisesti niistä osista, joissa vaikutetaan käyttäjälle näytettävään tietoon. Seuraavassa luvussa kerrotaan käytännön työvaiheista, jotka Cubiqilla tehtiin, jotta sovelluksilla näytettävien tietojen luotettavuutta saatiin parannettua.

6 Analytiikan kehittämisen työvaiheet Cubiqilla

Analytiikan kehittämisen tavoitteena oli parantaa Cubiqin tiedolla johtamisen edellytyksiä parantamalla tietojen luotettavuutta Qlik Sense -sovelluksilla. Seuraavassa alaluvussa on kuvattu tavoitteet ja rajaukset tarkemmin.

6.1 Ongelman ja tavoitteen määrittely ja rajaukset

Kososen kuvaileman tiedolla johtamisen prosessin ensimmäinen vaihe on ongelman tai tavoitteen määrittely (ks. luku 2.1) [8]. Siitä lähdettiin liikkeelle myös analytiikan kehittämisessä. Ratkaistava ongelma ja tavoitteet määritettiin Cubiqin johtoryhmään kuuluvien henkilöiden kanssa. Ongelmaksi määritettiin sovelluksilta saatavan tiedon huono luotettavuus ja tärkeimmäksi tavoitteeksi määritettiin sen parantaminen. Merkittävimmiksi tietojen luotettavuuteen vaikuttaviksi tekijöiksi määritettiin:

- ajantasaisuus
- oikeellisuus
- yhdenmukaisuus eri sovellusten välillä.

Esimerkiksi, jos uusien sovellukselle luettu data on viikon takaa, niin sovellukselta saatava tieto ei ole luotettavaa, jos tarkoituksena on analysoida koko edellisen kuukauden tuntikirjauksia. Toisaalta ajan tasalla oleva data, joka on mallinnettu väärin, voi tuottaa virheellistä tietoa näytettäväksi sovelluksilla. Jos taas tuntikirjaus linkittyy väärään projektiin, niin sovellukselta saadaan väärää tietoa. Luotettavuuteen todettiin olennaisesti vaikuttavan myös sen, saadaanko kaikilta sovelluksilta sama tieto samaan aikaan. Esimerkiksi, jos Bonus-sovellukselta ja Raportointi-sovellukselta katsotaan tietylle asiakkuudelle kirjattujen työtuntien määrää samaan aikaan, niin saadaanko molemmilta sovelluksilta myös sama tieto.

Päätavoitteen määrittämisen lisäksi sovittiin, että sovelluskehityksessä noudatettaisiin yrityksen määrittämiä hyviä sovelluskehityskäytäntöjä ja ratkaisusta yritettäisiin rakentaa mahdollisimman helposti ylläpidettävä. Rajauksena määritettiin, että kehitystä tehtäisiin niille sovelluksille, jotka käyttävät tärkeimpänä datalähteenään ValueFrame-toiminnan-ohjausjärjestelmän tietokantaa. Tiedolla johtamisen edellytyksiä oli siis määrä parantaa

yrittäjien toiminnasta itsestään syntyvää, sisäistä tietoa koskien. Yritysjohdon kanssa sovittiin lisäksi, että ensin kartoitettaisiin sovellusten nykytila ja kartoitusvaiheen löydösten perusteella tavoitteita tarkennettaisiin.

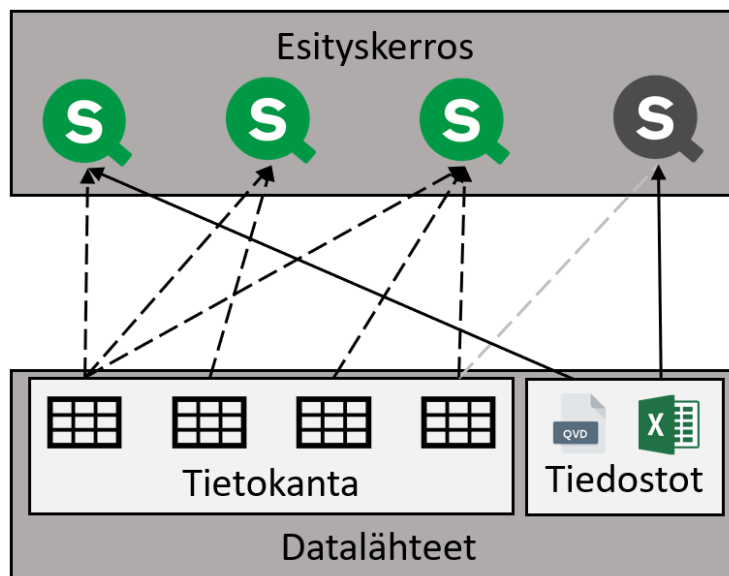
6.2 Kartoitusvaihe

Tiedolla johtamisen prosessissa oleellisina vaiheina on kartoittaa, mitä tietoa tarvitaan ja kerätä sitä [8]. Kartoitusvaiheessa kerättiin tietoa erityisesti olemassa olleisiin ValueFrameen liittyviin Qlik Sense -sovelluksiin liityen. Lisäksi pyrittiin löytämään asioita, jotka vaikuttivat negatiivisesti tiedon luotettavuuteen tai olivat yrityksen määrittämien sovelluskehityskäytäntöjen vastaisia tai vaikeasti ylläpidettäviä ratkaisuja. Tavoitteena oli siis muodostaa kokonais käsitys tilanteesta ja samalla tunnistaa kehityskohteita. Kartoitusvaiheessa tutustuttiin lisäksi tarkemmin Qlik Sense -ohjelmistoon ja sovelluksiin yleisellä tasolla, jotta ymmärrys niiden toiminnasta kasvaisi ja kehityskohteita olisi helpompi tunnistaa opittujen asioiden perusteella. Ohjelmistosta ja sovelluksista on kerrottu aiemmin tämän työn luvussa neljä, jotta tässä osiossa kerrottavat asiat olisivat jo tuttuja.

Kartoitusvaiheessa ilmeni, että yrityksen sen hetkiset sovellukset olivat aiempien työharjoittelijoiden tekemiä. Sovellusten tekohetkellä yritys ei ollut vielä määritellyt yhteisiä toimintatapoja sovelluskehityksessä. Sovellukset oli luotu vastaamaan yritysjohtoa sillä hetkellä kiinnostaneeseen asiaan mahdollisimman nopeasti. Analysointitarpeiden muuttua ja kasvettua ne eivät kuitenkaan enää täysin palvelleet käyttäjiänsä. Edellä mainitun tiedon perusteella oli sopivaa olettaa, että kehityskohteita löytyisi paljon. Toisaalta olemassa olleiden sovellusten alkuperäisten kehittäjien tunnistaminen auttoi myöhemmässä vaiheessa, kun tiedettiin keneltä voisi tarvittaessa kysyä sovelluksiin liittyviä asioita. Tiedon luotettavuuteen vaikuttavista tekijöistä ajantasaisuuden todettiin olevan melko hyvällä tasolla jo lähtötilanteessa. Sovellukset päivittyivät ajastetusti tunnin tai kahden välein. Yhdenmukaisuus ja oikeellisuus sen sijaan vaativat enemmän kehittämistä – sovelluksilta sai tietyissä tilanteissa vahingossa tuotettua väärää tietoa eli misinformaatiota. Seuraavana on tarkemmin tietoa kartoitusvaiheesta.

6.2.1 Sovellukset ja datalähteet

Olemassa oleviin sovelluksiin tutustuminen aloitettiin listaamalla kaikki ne sovellukset, jotka lukivat ValueFramen tietokannasta (jäljempänä pelkkä *tietokanta*) dataa. Lisäksi kerättiin tietoa sovellusten käyttämistä muista datalähteistä ja eriteltiin tietokannan taulut, joista dataa luettiin sovelluksille. Sovelluksia löytyi kolme, ja niihin luettiin dataa yhteensä 20 eri taulusta tietokannasta. Lisäksi sovelluksille luettiin dataa useista erillisistä tiedostoista, kuten Excel-tiedostoista ja QVD-tiedostoista (*QlikView Data*). QVD-tiedosto on QlikView'n ja Qlik Sengen käyttämä tiedostomuoto, joka sisältää taulukkomuodossa olevaa dataa (yhden taulun). Tiedostoformaatti on optimoitu datan säilyttämiseen kompaktissa muodossa, josta se voidaan erittäin nopeasti lukea esimerkiksi Qlik Sense -sovellukselle. Tiedon lukeminen QVD-tiedostosta on tyypillisesti 10–100 kertaa nopeampaa kuin muista datalähteistä. [37.] Kuvassa 18 on visualisoitu lähtötilannetta kerätyn ja analysoidun tiedon perusteella.



Kuva 18. Datan kulku datalähteistä esityskerroksen sovelluksiin lähtötilanteessa.

Kuvan 18 yläosan kuvakkeet kuvastavat Qlik Sense -sovelluksia, alareunan kuvakkeet kuvastavat datalähteitä. Kuvakkeiden määrä ei vastaa tietokannan taulujen tai erillisten tiedostojen todellista määrää. Kuvan tarkoituksena on havainnollistaa sitä, että samaa dataa luettiin useille sovelluksille ja että datalähteitä oli ValueFramen tietokannan lisäksi

myös erilliset tiedostot. Kuvan 18 vasemmassa alareunassa olevasta tietokannan taulusta luettiin dataa kaikille kolmelle sovellukselle (esimerkiksi tuntikirjaukset). Toisaalta jokaiselle sovellukselle luettiin myös jotain sellaista dataa, jota ei luettu muille sovelluksille. Lisäksi löydettiin myös yksi sovellus (kuvan 18 oikea yläreuna), joka luki dataa erillisestä tiedostosta, vaikka sama data olisi ollut saatavilla myös tietokannasta, mistä sitä luettiin toiselle sovelluksille. Kyseinen sovellus luki työntekijöihin liittyviä tietoja (kuten nimi ja osoite) erillisestä tiedostosta, jossa niitä ylläpidettiin ennen ValueFramen käyttöönottoa.

Datan lukeminen sovelluksille suoraan tietokannasta tai muusta tietolähteestä on nopea tapa päästä analysoimaan sitä, mutta tässä tapauksessa se tunnistettiin kehityskohteeksi, sillä saman datan valmistelulogiikkaa hallittiin jokaisella sovelluksella erikseen. Tämä todettiin myös yhdeksi syyksi sille, miksi eri sovelluksilla esiintyi ajoittain ristiriitaisuuksia. Sen lisäksi, että datan lukeminen suoraan tietokannasta ja sen valmistelu jokaisella sovelluksella erikseen asetti hallittavuudelle haasteita, on se myös Cubiqin määrittelyjen käytäntöjen vastaista. Käytäntöjen mukaan datan luku- ja valmisteluvaiheiden tulisi olla esityskerroksesta erillisiä omia kerroksiaan tietoarkkitehtuuria rakentaessa. Näistä kerrotaan tarkemmin kehitysvaihetta koskevassa alaluvussa 6.3. Sovellusten ylläpidettävyyden ja tietojen yhdenmukaisuuden varmistamiseksi päätettiin lisäksi, että erikseen ylläpidettävien tiedostojen määrää pyritään vähentämään.

6.2.2 Latausskriptit

Sovellusten ja niiden käyttämien datalähteiden listaamisen jälkeen tutustuttiin tarkemmin sovellusten latausskripteihin. Latausskriptit näyttivät visuaalisessa tarkastelussa melko lähelle siltä, miltä niiden kuuluisikin näyttää Cubiqin määrittämien hyvien käytäntöjen mukaan. Latausskriptit sisälsivät siis lukemista helpottavia kommentteja (rivejä, jotka eivät vaikuta skriptin suorittamiseen), latausskriptiä oli jaettu aihekokonaisuuksittain omiin osioihinsa (ks. kuva 15), eivätkä osiot sisältäneet yli 50 000 merkkiä.

Ongelmaksi tunnistettiin se, että samaa dataa valmisteltiin eri sovellusten latausskripteissä hieman eri tavoin. Latausskripteissä määriteltiin esimerkiksi, mitkä tuntikirjaukset ovat Cubiqin sisäistä työtä ja mitkä taas asiakkaalta laskutettavaa työtä. Tämä aiheutti kokonaisuuden hallintaan haasteita, sillä latausskriptin muutokset täytyi muistaa

tehdä monelle sovellukselle, jos esimerkiksi sisäiseksi työksi luokiteltavan tuntikirjauksen määritelmä muuttui. Kehityskohteeksi tunnistettiin lisäksi se, että jokaisella sovelluksella määritettiin niin sanotut perusmuuttujat erikseen (ks. 5.4.2). Cubiqin käytäntöjen mukaan muuttujat kannattaa määritellä erillisessä tiedostossa, josta ne voidaan lukea kaikille sovelluksille. Näin varmistutaan siitä, että ne on määritelty samoin kaikkien sovellusten välillä, ja esimerkiksi päivämäärät näytetään aina samassa formaatissa. Vähemmän merkityksellisenä asiana huomattiin lisäksi, ettei kenttiä oltu nimetty kovin käyttäjäystävällisesti, eikä muuttujien nimeämisessä ollut käytetty suositeltuja käytäntöjä.

6.2.3 Datamallit

Datamallit näyttivät sovelluksilla asianmukaisilta. Cubiqin käytäntöjen mukaan datan mallintamisessa tulisi mahdollisuuksien mukaan pyrkiä muodostamaan niin sanottu ”tähtimalli” (ks. 5.4.2). Visuaalisesti tarkasteltuna tässä olikin pääasiassa onnistuttu, mutta vaikka datamallit muistuttivat tähtimallia (keskellä yksi taulu, jonka ympärillä muut taulut), niin taulujen sisältö oli vääränlaista. Faktataulut sisälsivät paljon kuvailevaa dataa, joka olisi kuulunut dimensiotauluihin, kun taas dimensiotauluissa oli laskennoissa käytettävää dataa. Oikein rakennettu tähtimalli on etenkin suurten datamäärien kanssa yleensä paras tapa mallintaa data, esimerkiksi palvelimen resurssien käytön kannalta.

Vaikka datamalli voi näyttää monenlaiselta, ja tiedot voivat silti yhdistyä oikein ja tuottaa laskennoissa oikeat tulokset, niin sovelluksilta löydettiin käytännön ongelmia ja virheitä, joiden perusteella voidaan sanoa, että data oli mallinnettu väärin tai ainakin puutteellisesti. Yhdeksi ongelmaksi tunnistettiin esimerkiksi se, että eri kirjauksille oli lisätty kuvailevaa aikatietoa vaihtelevasti. Esimerkiksi tuntikirjaus sisältää aina tiedon päivästä, johon se kohdistuu. Tuntikirjauksen sisältävää riviä oli rikastettu latausskriptissä lisäämällä tiedot viikosta, kuukaudesta ja vuodesta, jolle päivämäärä kuuluu, kuten esimerkkikoodissa 3.

```
Fakta:
LOAD
    Tuntikirjaus_ID,
    Työntekijä_ID,
    Asiakas_ID,
    Tuntimäärä,
    Päivämäärä,
    Week(Päivämäärä) As Viikko,
    Month(Päivämäärä) As Kuukausi,
    Year(Päivämäärä) As Vuosi
FROM [lib://Esimerkkidata/Tuntikirjaukset.qvd] (qvd);
```

Esimerkkikoodi 3. Ote esimerkkisovelluksen latausskriptistä, jossa Päivämäärä-kentästä on johdettu uusia kenttiä.

Esimerkkikoodissa 3 tuntikirjaukset luetaan QVD-tiedostosta Fakta-tauluun. Tuntikirjausrivillä olevasta Päivämäärä-kentästä johdetaan uusia kenttiä skriptifunktioiden avulla. Näin tekemällä tuntikirjausrivi sisältää päivämäärän lisäksi tiedon siitä, mihin viikkoon, kuukauteen ja vuoteen se liittyy. Ongelma syntyi, kun myöhemmin Fakta-tauluun lisättiin esimerkiksi laskutukseen liittyvää dataa, jonka riveille oli määritetty päivämäärän lisäksi vain kuukausi- ja vuositiedot. Näin ollen, jos sovelluksella valittiin jokin tietty viikko, niin mikään laskutukseen liittyvistä riveistä ei liittynyt valintaan. Päivämäärä-, Kuukausi- ja Vuosi-kenttiin tehtyjen valintojen kohdalla kaikki toimi oikein. Voidaan sanoa, että tämä ongelma liittyi jollain tapaa enemmän latausskriptiin, mutta tuloksena syntyi puutteellinen datamalli, jossa oikeat asiat eivät yhdistyneet toisiinsa. Cubiqin käytäntöjen mukaan päivämäärätietoa kuvailevat muut kentät kannattaa yleensä myös pitää Fakta-taulusta erillään, erillisessä dimensiotaulussa (Kalenteri-taulu). Näin ollen ongelma luokiteltiin datamalliin liittyväksi. Kalenteri-taulusta lisää kehitysvaiheen alaluvussa (6.3) sekä tuloksissa (luku 7).

6.2.4 Visualisoinnit, Master items -kirjastot ja muuttujat

Sovellusten visuaaliseen ilmeeseen oltiin pääasiassa melko tyytyväisiä, eikä siihen kaihattu kovin suuria muutoksia. Tämän työn kannalta visualisointien ulkonäkö ei ole yhtä merkityksellinen kuin niiden tarjoama tieto. Kehityskohteiksi tunnistettiin seuraavat asiat:

- Jokaisella visualisoinnilla on mahdollista näyttää lisätietoja siinä käytetyistä mittareista ja dimensioista, mutta tätä ei ollut hyödynnetty kuin osassa visualisointeja.

- Osassa visualisoinneista ei ollut käytetty Master items -kirjaston mittareita eikä dimensioita.
- Osassa visualisointeja, Master items -kirjaston mittareita ja muuttujia oli käytetty niin sanottua ”kovakoodausta”. Esimerkiksi kuluvaa ja edellistä vuotta vertailevissa laskentakaavoissa oli käytetty arvoja ’2019’ ja ’2018’. Nämä olisi voitu ilmaista esimerkiksi ’Year(Today())’- ja ’Year(Today())-1’ -muodoissa, jolloin niitä ei tarvitsisi muuttaa vuoden vaihtuessa.
- Laskentakaavojen määrittelyissä oli pieniä eroja eri sovellusten välillä, osittain johtuen yritysjohdon eriävistä näkemyksistä aiempaa sovelluskehittäjä ohjeistaessa, osittain vahingoista tai unohduksista.

Edellä mainittujen kehityskohteiden osalta sovittiin, että jatkossa lisätiedot näytettäisiin kaikissa visualisoinneissa, Master items -kirjaston objekteja hyödynnettäisiin niin ikään kaikissa visualisoinneissa ja kovakoodausta vältettäisiin. Tärkeimpänä koettiin, että Master items -kirjaston mittareita saataisiin hallittua keskitetysti, jotta ne olisi määritelty samoin jokaisella sovelluksella.

6.3 Kehitysvaihe

Kehitysvaiheen aluksi yritysjohdon kanssa katselmoitiin kartoitusvaiheessa tehdyt huomiot ja valittiin kehitettävät kohteet. Kehityskohteet luokiteltiin myös tiedolla johtamisen prosessin vaiheiden mukaisesti ja niille määritettiin tärkeysjärjestys taulukon 4 mukaisesti. Taulukossa ensimmäisenä on tärkein kehityskohde.

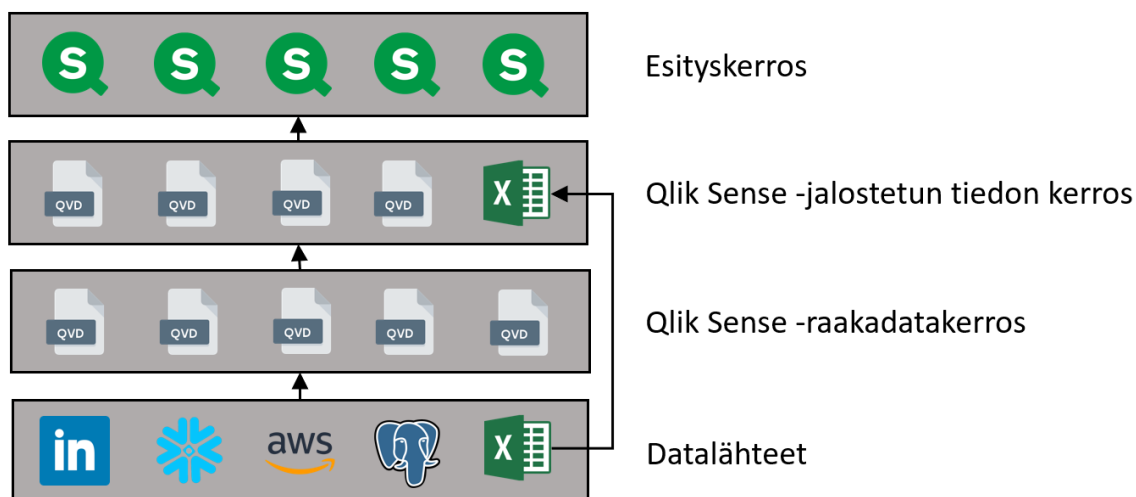
Taulukko 4. Kehityskohteet tärkeysjärjestyksessä ja luokiteltuna Kososen kuvaileman tiedolla johtamisen prosessin vaiheisiin [8].

Kehityskohde	Tiedolla johtamisen prosessin vaiheet
Kerroksellisen tietoarkkitehtuurin rakentaminen eli datan luku- ja valmistelukerrosten erottaminen esityskerroksesta.	Ymmärrys: Tarvittavan tiedon kartoittaminen ja tiedonkeruu. Mallinnus: Datan valmistelu. Tulokset: Raportointi ja säilytys.
Datan mallintaminen paremmin ja mahdollisimman oikeaoppisesti tähtimalliin. Tavoitteena on parantaa tietojen oikeellisuutta yhdistämällä oikeat asiat toisiinsa ja täydentämällä puuttuvia tietoja, esimerkiksi lisäämällä erillisen Kalenteri-dimensiotaulun.	Mallinnus: Datan valmistelu.
Laskentakaavojen (mittareiden) määrittämisen tarkistaminen ja niiden keskitetty säilytys paikassa, josta ne voidaan lukea sovelluksille. Määrittämisen näyttämisen visualisointien lisätietiedoissa. Tavoitteena parantaa tietojen yhdenmukaisuutta ja läpinäkyvyyttä (miten tietty asia lasketaan).	Mallinnus: Datan valmistelu. Mallinnus: Visualisointi.
Erikseen ylläpidettävien tietojen ja tiedostojen määrän minimointi.	Ymmärrys: Tarvittavan tiedon kartoittaminen ja tiedonkeruu. Mallinnus: Datan valmistelu ja analyysi.
Sovelluksilla näytettävien tietojen ajantasaisuuden parantaminen.	Ymmärrys: Tiedon keruu. Mallinnus: Datan valmistelu ja visualisointi.
Yhteisten käytäntöjen käyttöönotto esimerkiksi sovellusten ja muuttujien nimeämisessä ja kansiorakenteessa. Dokumentaation parantaminen Qlik Sense -ympäristöön liittyen.	Ymmärrys: Tiedonkeruu. Tulokset: Raportointi ja säilytys.

Suurin osa taulukossa 4 esitellyistä kehityskohteista liittyi tiedolla johtamisen prosessin mallinnus-päävaiheeseen [8]. Kehitysvaiheessa tehdyistä käytännön toimenpiteistä kerrotaan alla tärkeimmästä alkaen.

6.3.1 Kerroksellisen tietoarkkitehtuurin rakentaminen

Cubiqilla kerroksellinen tietoarkkitehtuuri tarkoittaa sitä, että dataa valmistellaan kerroksissa ennen sen esittämistä loppukäyttäjille. Tämä toteutetaan Qlik Sensessä luomalla omat sovellukset datan hakemiseen lähdejärjestelmästä, datan valmisteluun ja tiedon esittämiseen. Kuva 19 havainnollistaa tätä:



Kuva 19. Esimerkki kerroksellisesta tietoarkkitehtuurista.

Kuvassa 19 on esimerkki kerroksellisesta tietoarkkitehtuurista. Alimmassa kerroksessa ovat erilaiset datalähteet kuten tietokannat, sosiaalinen media ja tiedostot. Raakadatakeros muodostetaan Qlik Sense -sovelluksilla, joiden tehtävänä on lukea data lähdejärjestelmästä ja tallentaa se muokkaamattomana QVD-tiedostoihin. Cubiqin käytäntöjen mukaan jokaista lähdejärjestelmää kohden luodaan oma sovellus. Jos datan määrä on suuri, sovelluksia voi olla useampiakin. Jalostetun tiedon kerros muodostetaan tähän tarkoitukseen luoduilla sovelluksilla, jotka lukevat raakadatakerosen QVD-tiedostoista dataa, valmistelevat sitä ja tallentavat sitten uudet QVD-tiedostot. Esityskerroksessa dataa ei enää käsitellä, vaan se luetaan sovelluksille edellisestä kerroksesta valmiiksi valmisteltuna. Erillisiä tiedostoja, kuten Excel-tiedostoja, ei välttämättä tarvitse erikseen valmistella ollenkaan ennen esityskerrosta.

Cubiqin käytäntöjen mukaan datan valmistelua kannattaa tehdä mahdollisuuksien mukaan jo lähdejärjestelmissä. Valmistelua voidaan tehdä isommassa mittakaavassa esimerkiksi Qlikin dataintegraatiotuotteita hyödyntämällä ennen kuin dataa luetaan Qlik

Senseen. Näin toimimalla ei tarvitse käyttää yhtä paljon Qlik Sense -ympäristön palvelimien resursseja datan valmisteluun, jolloin käyttäjät voivat saada esimerkiksi lyhyemmällä vasteajalla toimivia sovelluksia käyttöönsä. Hyvin valmisteltu data voidaan lukea esimerkiksi yrityksen tietovarastosta suoraan jalostetun tiedon kerrokseen. Toisaalta valmistelu voi olla niinkin pientä, että erilaisille työläjelle annetaan ValueFrame-järjestelmässä sellaiset numerokoodit, joiden perusteella ne voidaan luokitella Qlik Sensessä dynaamisesti muokkaamatta sovelluksia tai mittareiden määrittäjiä joka kerta uudestaan. Kerroksellisuuden rakentaminen tehdään esimerkiksi siksi, että datan valmistelua voidaan hallita mahdollisimman keskitetysti. Esimerkiksi ValueFramesta luettua dataa esitetään loppukäyttäjille usealla sovelluksella, joten se tulee valmistella yhdenmukaisesti, jotta tiedot eivät ole ristiriidassa keskenään eri sovellusten välillä. Toisaalta esimerkiksi datan mallintamisvaiheessa on huomattavasti nopeampaa lukea dataa sovellukselle raakadatakerroksen QVD-tiedostoista kuin suoraan tietokannasta.

Cubiqin käytäntöjen mukaisen tietoarkkitehtuurin rakentaminen vaati tässä tapauksessa vähintään kahden uuden sovelluksen luomisen: yksi datan lukemiseen ValueFramen tietokannasta, toinen sen valmisteluun esityskerrosta varten. Lähtötilanteessa data luettiin lähdejärjestelmästä ja valmisteltiin suoraan esityskerroksessa (ks. kuva 18). Edellä mainittujen kahden sovelluksen lisäksi päätettiin rakentaa vielä kolmas sovellus. Kolmannen sovelluksen tarkoituksena on lukea jalostetun tiedon kerroksen valmiista QVD-tiedostoista data ja pitää sisällään valmis datamalli. Loppukäyttäjien esityskerroksessa käyttämät sovellukset lukevat datamallin itseensä hyödyntämällä Qlik Sengen *Binary*-komentoa. Binary-komennolla sovellukselle voidaan lukea kaikki toisen sovelluksen sisältämä data (Qlikin virallisen dokumentaation mukaan tämä ei ole mahdollista Qlik Sense Enterprise -ympäristössä, mutta toimii silti). [36.] Toinen vaihtoehto olisi, että esityskerroksen sovellukset lukisivat jalostetun tiedon kerroksen QVD-tiedostoista tiedot, mutta tämä tapa valittiin, jotta tietoa ei enää vahingossa käsiteltäisi eri tavoin lukuvaiheessa. Toisaalta ajoittain on esiintynyt ongelmia, jos useat sovellukset lukevat samoja QVD-tiedostoja samaan aikaan tai kun toinen sovellus on tallentamassa tietoa niihin.

Raakadatakerroksen synnyttävän sovelluksen luominen oli helppoa; data luettiin niistä ValueFramen tietokannan tauluista, jotka olivat kartoitusvaiheessa tunnistettu alkupeuraisten sovellusten datalähteiksi, ja tallennettiin sitten QVD-tiedostoihin. Jalostetun tie-

don kerroksen rakentaminen oli haastavampaa, sillä ValueFramen datan lisäksi esityskerrokseen haluttiin dataa erillisistä tiedostoista, jotka olivat peräisin ajalta ennen kuin Cubiqilla otettiin ValueFrame käyttöön. Datamallin sisältävän sovelluksen luominen oli yksinkertaista, kun jalostetun tiedon kerros oli ensin rakennettu. Jalostetun tiedon kerroksen rakentamisesta kerrotaan alla tarkemmin.

6.3.2 Datan mallintaminen

Datan mallintamisvaihe, eli jalostetun tiedon kerroksen rakentaminen, oli ylivoimaisesti aikaa vievin ja vaikein osuus, koska aiempaa kokemusta ei juurikaan ollut. Mallintamista tehtiin iteroivasti osa kerrallaan ja iteroitien välissä lukuja tarkistettiin moneen kertaan, kunnes ne saatiin oikein. ValueFramesta luetun datan valmistelu oli helpompaa, erillisistä tiedostoista luetun datan valmistelu vaikeampaa. Suurimman haasteen tuotti se, että alkuperäisillä sovelluksilla ValueFramen data oli valmisteltu ja mallinnettu siihen muotoon, kuin sitä säilytettiin erillisissä Excel-tiedostoissa yrityksen toiminnan alkuaikoina, joten lähes kaikkea piti muuttaa.

Sovellukset oli ilmeisesti alun perin luotu pelkästään erillisten Excel-tiedostojen dataa käyttämällä, ja näille oli sittemmin lisätty ValueFramen dataa. ValueFramesta luettu data oli valmisteltu sovelluksille suurin piirtein samaan muotoon kuin se oli alkuperäisissä Excel-tiedostoissa ollut, esimerkiksi tuntikirjausten osalta. Todennäköisesti tämä oli sillä hetkellä nopein tapa saada aiemmin hyvin toimineelle sovellukselle uutta tietoa uudesta datalähteestä. Vanhoissa, erillisissä tiedostoissa esimerkiksi yrityksen sisäistä työtä koskevat tuntikirjaukset olivat omassa sarakkeessaan, laskutettavat tunnit ja lomatunnit omissaan. Alkuperäisille sovelluksille oli esimerkiksi kovakoodattu tietyt työlajit, joille kirjatut työtunnit luettiin ”Laskutettavat työtunnit” -kenttään, kuten esimerkikoodissa 4.

```
Fakta:
LOAD
    Tuntikirjaus_ID,
    Työntekijä_ID,
    Asiakas_ID,
    If(Työlaji_ID = '1', Tuntimäärä, 0) As [Laskutettavat työtunnit],
    If(Työlaji_ID = '2', Tuntimäärä, 0) As [Sisäiset työtunnit]
    Päivämäärä
FROM [lib://Esimerkkidata/Tuntikirjaukset.qvd] (qvd);
```

Esimerkkikoodi 4. Esimerkki työlajien jaottelusta latausskriptissä.

Esimerkkikoodissa 4 Tuntimäärä-kentän arvo tallennetaan joko Laskutettavat työtunnit -kenttään tai Sisäiset työtunnit -kenttään riippuen Työlaji_ID-kentän arvosta. Jos arvo on jokin muu kuin 1 tai 2, tuntimäärää ei lueta mihinkään kenttään. ValueFramessa tuntikirjauksen tuntimäärä säilytetään yhdessä tietokannan kentässä ja kirjaukseen liittyvien muiden tietojen perusteella se voidaan luokitella sisäiseksi työksi, laskutettavaksi työksi tai lomaksi. Kun ValueFrameen lisättiin uusia laskutettavaksi tarkoitettuja työlajeja, ja niille kirjattiin työtunteja, niin tuntimäärät eivät enää menneet mihinkään kenttään, koska kovakoodatut ehdot eivät täytyneet näiden osalta. Kehitysvaiheessa tätä muutettiin niin, että eri työlajit luetaan erilliseen dimensiotauluun, joka yhdistyy tuntikirjaukseen työlajin yksilöivän ID-kentän kautta (Työlaji_ID). Työlajit-dimensiotaulu sisältää työlajia koskevaa kuvailevaa tietoa, kuten ValueFramessa määritetyn nelinumeroisen koodin, jonka perusteella päätellään työlajin luokitus esimerkiksi laskutettavaksi tai bonuskertymää kasvattavaksi kirjaukseksi. Eli kun työlajeja lisätään, niin Qlik Sensen päässä kaikki toimii ilman muutoksia, jos ValueFramen päässä työlajille on määritelty oikea koodi. Koodin muuttaminen ValueFramessa on huomattavasti helpompaa kuin usean sovelluksen latausskriptin päivittäminen.

Kehitysvaihetta käynnistettäessä Cubiq oli ollut toiminnassa vasta pari vuotta eikä dataa ollut kertynyt kovin pitkältä ajalta. Siksi myös erillisten tiedostojen sisältämä historia haluttiin säilyttää. Jouduttiin siis valmistelemaan niiden tiedostojen sisältämää dataa vastaamaan uutta tarvetta. Käytännössä siis tehtiin käänteisesti se, mitä aiemmat sovelluskehittäjät olivat tehneet. He valmistelivat uuden datan vanhaan muotoon, nyt vanha data piti valmistella uuteen muotoon. Esimerkiksi niille työtunneille, jotka olivat Excel-tiedostossa merkitty ”Laskutettavat työtunnit” -sarakeeseen, annettiin tietty Työlaji_ID, jotta luokittelu toimi samoin kuin ValueFramesta haettavien uusien tietojen kohdalla. Tähän tarkoitukseen luotiin vielä yksi erillinen ja ”kertakäyttöinen” sovellus, jolla tuotettiin yksi QVD-tiedosto. Tuosta tiedostosta data voidaan lukea valmisteltavaksi samoin kuin tietokannasta luettu raakadata. Mallintaminen tehtiin parhaan osaamisen mukaan mahdollisimman oikeaoppisesti. Datamallin taulut muodostava sovellus tallentaa valmiit taulut jalostetun tiedon kerrokseen QVD-tiedostoihin.

6.3.3 Mittareiden määrytykset ja keskitetty hallinta

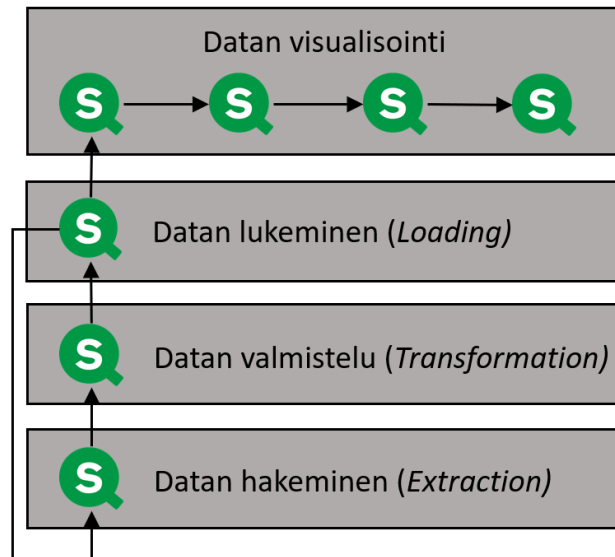
Mittareiden määrytykset tarkistettiin yritysjohtoon kanssa. Tiettyjen mittareiden kohdalla oli isojakoin näkemuseroja. Erityisesti yhden mittarin määrytyksen tiimoilta käytiin pitkiä keskusteluja, mutta lopulta kaikki tärkeimmät mittarit saatiin määrytettyä. Mittareita olivat esimerkiksi bonuksiin oikeuttavien työtuntien määrä ja työntekijän laskutusaste prosentti. Mittarit kerättiin yhteen Excel-tiedostoon, johon kirjattiin laskentakaavan lisäksi ainakin ne tiedot, jotka voidaan määrytellä Master items -kirjaston Measure-objektille (ks. 5.4.2). Määrytykset luetaan sovelluksille latausskriptissä ja sitä kautta Master items -kirjastoon.

6.3.4 Erikseen ylläpidettävien tietojen ja tiedostojen määrän minimointi

Sen lisäksi, että mittareiden määrytyksien hallintaa keskitettiin, niin myös erikseen ylläpidettävien tietojen ja tiedostojen määrää vähennettiin. Tiedostojen määrää saatiin vähennettyä esimerkiksi hyödyntämällä ValueFramea paremmin ja ylläpitämällä kaikki mahdollinen tieto siellä. Esimerkiksi työntekijöiden puhelinnumeroita ja osoitetietoja hallittiin aiemmin Excel-tiedostossa, mutta ne siirrettiin ValueFrameen. Kaikkea tarvittavaa tietoa ei saatu siirrettyä ylläpidettäväksi ValueFrameen, mutta iso osa erikseen ylläpidettävästä tiedosta saatiin kerättyä yhteen Excel-tiedostoon. Tuo tiedosto sijoitettiin pilvipalveluun, josta se saadaan luettua Qlik Sense -ympäristöön. Pilvipalvelun kautta useampi henkilö pääsee muokkaamaan sitä tarvittaessa mistä tahansa.

6.3.5 Sovelluksilla näytettävien tietojen ajantasaisuuden parantaminen

Sovelluksilla näytettävien tietojen ajantasaisuutta parannettiin entisestään tihentämällä sovellusten ajastettujen lataustehtävien (*Task*) aikaväliä ja ketjuttamalla tehtävät toistensa perään, niin että ne muodostavat kehän, joka käynnistyy aina heti uudestaan, kun tietty vaihe valmistuu. Tehtävä tarkoittaa, että siihen liitetyn sovelluksen latausskripti suoritetaan. Kuvassa 20 on pyritty havainnollistamaan tätä.



Kuva 20. Lataustehtävien ketjutukset.

Kuvan 20 laatikoiden sisällä olevat kuvakkeet kuvastavat sovellusten lataustehtäviä. Kuvassa alimman, ValueFramesta dataa hakevan tehtävän valmistuttua käynnistyy dataa valmistelevan sovelluksen lataus, jonka jälkeen käynnistyy datamallin lukevan sovelluksen lataus. Datamallin lukevan sovelluksen latauksen valmistuttua käynnistyy sekä esitysgerroksen sovellusten lataukset että dataa hakevan sovelluksen lataus uudestaan. Tehtävät toistuvat ikuisessa silmukassa, kunnes ne pysäytetään manuaalisesti tai jokin tehtävistä epäonnistuu ja ketju katkeaa. Ketjun ensimmäinen tehtävä käynnistetään varmuuden vuoksi ajastetusti joka aamu siltä varalta, että se olisi päässyt jostain syystä katkeamaan esimerkiksi yön aikana. Lisäksi lataustehtävien valvontaan rakennettiin automaattinen hälytysjärjestelmä, joka lähettää kaatuneesta latauksesta automaattisesti hälytyksen yrityksen tukiosoitteeseen ja siitä kirjautuu myös tukipyyntö yrityksen toiseen järjestelmään. Näin vika saadaan korjattua mahdollisimman pian.

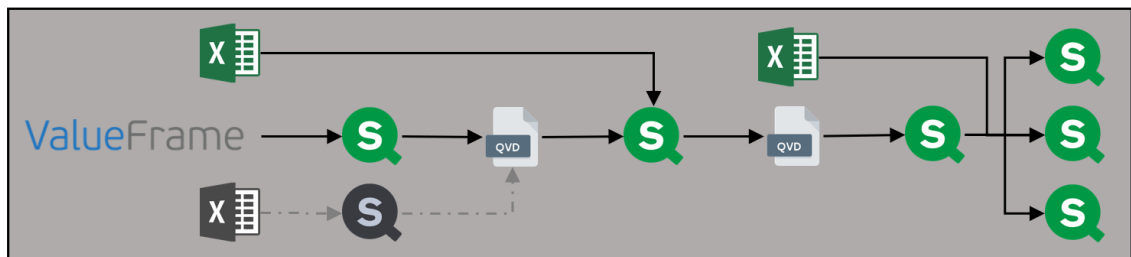
6.3.6 Dokumentaatio ja yhteiset käytännöt

Sovellukset ja niillä käytettävät muuttujat sekä sovellusten tuottamat QVD-tiedostot ja sovellusten käyttämät datayhteydet pyrittiin nimeämään yhteisten käytäntöjen mukaisesti. Tiedostot myös tallennettiin tiettyyn kansiorakenteeseen yhteisten käytäntöjen mukaisesti. Tarkoituksena on, että Cubiqin työntekijät käyttäisivät samanlaisia käytäntöjä tehdessään töitä mille tahansa asiakkaalle. Näin ollen siirtyminen asiakkuudesta toiseen

on mahdollisimman sujuvaa ja palvelu on myös tasalaatuista. Dokumentaatiota parannettiin esimerkiksi lisäämällä sovellusten latauskripteihin enemmän kommentointia. Jokaisen sovelluksen latauskripteihin tehtiin lisäksi ylimmäksi oma osio, jossa kerrotaan sovelluksesta lisätietoa ja ylläpidetään muutoshistoriaa. Tavoitteena on helpottaa sovellusten jatkokehittämistä muiden työntekijöiden toimesta. Qlik Sense -ympäristöstä tuotettiin myös erillinen dokumentaatio yrityksen toiseen järjestelmään.

7 Saavutetut tulokset

Cubiqin sisäisen Qlik Sense -analytiikan kehittämisprojektin tuloksena syntyi kokonaisuus, jossa on pyritty minimoimaan tiedon luotettavuuteen negatiivisesti vaikuttavat tekijät. Sovelluksilla näytettävä tieto on lähes poikkeuksetta enintään puoli tuntia vanhaa, tiedot ovat oikein ja yhdenmukaisesti esitetty eri sovellusten välillä. Tärkeimpänä tekijänä oli kerroksellisen tietoarkkitehtuurin rakentaminen, jolla saavutettiin se, että dataa valmistellaan vain yhdellä tapaa ennen sen lukemista esityskerrokseen yhdestä keskitetystä lähteestä. Muina tärkeitä tekijöinä oli, että muodostetussa datamallissa tiedot yhdistyvät toisiinsa oikein, mittareita ja perusmuuttujia hallitaan keskitetysti ja lataustehtäviä ajetaan niin usein kuin mahdollista (ks. kuva 20). Ratkaisu on toteutettu pääosin Cubiqin yhteisten käytäntöjen mukaisesti, mikä mahdollistaa sen, että kenen tahansa työntekijän pitäisi pystyä tarvittaessa tekemään jatkokehitystä kokonaisuuteen kuuluville sovelluksille. Kuva 21 havainnollistaa datan kulkua lopullisessa ratkaisussa.



Kuva 21. Datan kulkeminen ValueFramesta ja erillisistä tiedostoista loppukäyttäjien käyttämille sovelluksille.

Kuvassa 21 on kuvattu datan kulkeminen esityskerroksen sovelluksille. Kuvasta voidaan nähdä, että kaikki ValueFramesta luettava data kulkee nyt yhtä ja samaa reittiä pitkin. Yksi Qlik Sense -sovellus hakee datan ja tallentaa sen raakadatakerroksen QVD-tiedostoihin, toinen sovellus valmistelee datan jalostetun tiedon QVD-tiedostoihin, joilta se luetaan niin sanotulle Datamalli-sovellukselle. Dataa ei siis valmistella eri tavoin eri sovelluksille, kuten lähtötilanteessa (vrt. kuva 18). Datamalli-sovellukselta data luetaan esityskerrokseen esimerkiksi Raportointi-, Bonus- ja Lead konsultit -sovelluksille. Eri sovelluksilla on eri käyttötarkoitukset ja niillä visualisoidaan tietoja eri tavoin, mutta kaikilla on sama datasisältö. Erikseen ylläpidettävien tiedostojen (kuvan vasemman yläkulman Excel-tiedosto) määrä on minimoitu ja sieltä luettava data käsitellään vain yhdellä tapaa, ennen sen päätymistä loppukäyttäjien sovelluksille. Tärkeimpien mittareiden määrittämiä

hallitaan yhdessä tiedostossa (oikean yläkulman Excel-tiedosto), jolta ne luetaan esityskerroksen sovelluksille ja niiden Master items -kirjastoihin. Erikseen ylläpidettävien tiedostojen minimointi vähentää niiden ylläpitämiseen kuluvaan aikaan ja pienentää virherkkyyttä, joka taas parantaa tiedon laatua ja luotettavuutta sovelluksilla. Lisäksi kuvan 21 vasemmassa alareunassa on kuvattu kertakäyttöinen sovellus, joka valmisti vanhoja tuntikirjauksia sisältäneen datan siihen muotoon kuin se nyt saadaan ValueFramesta.

Kehitysvaiheen tehtävät tehtiin pääasiassa kesän 2018 aikana, jonka jälkeen kokonaisuus on esityskerrosta lukuun ottamatta pysynyt lähes muuttumattomana. Tarpeet ovat toki muuttuneet ja kasvaneet reilun kahden vuoden aikana, mutta ratkaisun runkoon ei ole tarvinnut tehdä isoja muutoksia. Sovelluksia käytetään aktiivisesti päivittäin useiden yrityksen työntekijöiden toimesta ja kaikilla työntekijöillä on pääsy niille. Kahden vuoden aikana tietojen oikeellisuutta on epäilty vain muutaman kerran, jolloin asiaa on tutkittu ja tarvittaessa esimerkiksi tietyn mittarin määrittäminen on muutettu, jos on todettu, että laskenta tuottaa väärä tuloksia. Ironisesti määrittäminen on jouduttu muuttamaan kahdesti juuri siihen mittariin, jonka määrittämisessä kesti alun alkaenkin kauiten aikaa. Monesti, kun tietojen oikeellisuutta on epäilty, onkin käynyt ilmi, että tiedot oli kirjattu ValueFrameen väärin, tai että tietokannassa oli ollut ongelma, jonka johdosta tiettyä dataa ei ollut saatu haettua. Useimmin toistunut ongelma on liittynyt lataustehtävän kaatumiseen syystä tai toisesta, jolloin latausketju (ks. kuva 20) on myös katkennut eivätkä tiedot ole päivittyneet sovelluksille hetkeen. Järjestelmän tärkeyttä kuvaa se, että jos ongelmaa ei ole saatu korjattua parissa tunnissa, niin käyttäjät huomaavat tämän ja kysyvät asiasta. Erityisesti yrityksen toimitusjohtaja käyttää sovelluksia työnsä tukena päivittäin ja huomaa käytännössä heti, jos tiedot eivät ole päivittyneet vähään aikaan. Jatkokehitystehtävänä on sovittu, että kaatuneista lataustehtävistä (sovellusten tiedot eivät ole ajan tasalla) tiedotetaan paremmin ja automatisoidusti.

Se, että kokonaisuus on säilynyt käytössä ilman merkittäviä muutoksia yli kahden vuoden ajan, on jo itsessään hyvä indikaattori ratkaisun laadusta ja hyödyllisyydestä. ValueFramen dataa käyttävien sovellusten tiedon luotettavuuteen liittyviä mielipiteitä kysyttiin kuitenkin vielä suoraan yrityksen työntekijöiltä Microsoft Forms -kyselyllä (Liitteet 1 ja 2). Kysely toteutettiin anonymisti ja vastauksia analysoitiin Qlik Sensellä. Kyselyyn vastasi 27 työntekijää, eli melkein puolet henkilöstöstä. Kaikki kyselyyn vastanneet henkilöt

kertoivat työskennelleensä Cubiqilla vähintään vuoden ajan. Valtaosa vastaajista kertoi käyttävänsä sovelluksia päivittäin (37 %) tai viikoittain (41 %). Loput vastaajat kertoivat käyttävänsä sovelluksia kuukausittain (7 %) tai harvemmin tai ei ollenkaan (15 %). Vähintään kuukausittain sovelluksia käyttäviä vastaajia (85 %) pyydettiin arvioimaan sovelluksilta saadun tiedon luotettavuutta edellisten kahden vuoden ajalta, eli ajalta kun kehitysvaiheessa tehdyt muutokset ovat olleet käytössä. Tietojen luotettavuutta arvioitiin vastaamalla kolmeen väittämään. Vastaukset annettiin asteikolla 1–5, jossa 1 = Täysin eri mieltä ja 5 = Täysin samaa mieltä. Väittämät ja vastausten keskiarvot olivat:

- Viimeisten kahden vuoden aikana sovelluksilta saamani tieto on ollut pääosin **ajan tasalla**. (4,74)
- Viimeisten kahden vuoden aikana sovelluksilta saamani tieto on ollut pääosin **oikein**. (4,74)
- Viimeisten kahden vuoden aikana sovelluksilta saamani tieto on ollut pääosin **yhdenmukaista** eri sovellusten välillä. (4,52)

Cubiqin johtoryhmään kuuluvien henkilöiden vastaukset olivat linjassa muiden vastausten kanssa (keskiarvo kaikissa väittämässä 4,67). Käyttäjiä pyydettiin lisäksi valitsemaan ne sovellukset, joiden tietoja he pitävät epäluotettavina, mutta yhtään sovellusta ei valittu. Vastaajien joukosta löytyi myös yksi täysin muista poikkeava vastaaja, jonka voidaan arvella vastanneen kyselyyn käänteisellä asteikolla. Kaikkien muiden vastaajien keskuudessa huonoin annettu vastaus edellä mainittuihin väittämiin oli 3 (2 kpl). Kaikki muut vastaukset olivat joko 4 (9 kpl) tai 5 (55kpl). Poikkeavan vastaajan vastaukset olivat 2, 1 ja 2. Jos kyseisen vastaajan vastaukset jätetään tuloksista pois tai tulkitaan käänteisellä asteikolla, niin vastausten keskiarvot ovat:

- Viimeisten kahden vuoden aikana sovelluksilta saamani tieto on ollut pääosin **ajan tasalla**. (4,83–4,86)
- Viimeisten kahden vuoden aikana sovelluksilta saamani tieto on ollut pääosin **oikein**. (4,91)
- Viimeisten kahden vuoden aikana sovelluksilta saamani tieto on ollut pääosin **yhdenmukaista** eri sovellusten välillä. (4,61–4,64)

Yhdenmukaisuuden osalta tulos oli hieman heikompi kuin ajantasaisuuden ja oikeellisuuden osalta. Valitettavasti alkuperäisestä tilanteesta ei kerätty vertailudataa, joten kyselyn tulosten perusteella ei voida arvioida tilanteen parantuneen tai huonontuneen, vaan ne pelkästään kuvaavat nykytilannetta. Kaiken kaikkiaan tulokset kertovat, että

näillä kolmella mittarilla mitattuna työntekijät ovat pitäneet sovelluksilta saamansa tietoa varsin luotettavina edellisten kahden vuoden aikana.

8 Pohdinta

Omasta näkökulmastani kehitysvaiheen lopputuloksena syntynyt kokonaisuus (sovellukset, lataustehtävät, mittareiden määrittysten keskitetty hallinta ja muut parannukset) on melko yksinkertainen, eikä sen hallinta ole kovin työlästä tai vaikeaa. Isoimman käytännön haasteen on aiheuttanut se, että sisäisille sovelluksille jatkokehitystä tekevät yleensä harjoitustehtävinä yrityksen kokemattomimmat työntekijät, joille kokonaisuuden hahmottaminen voi olla haastavaa. Ratkaisu koostuu useista sovelluksista, tiedostoista, lataustehtävistä ja muista asioista, jotka kaikki liittyvät jollain tapaa toisiinsa. Toisaalta Cubiqin toimitusjohtaja näkee tämän haasteen myös positiivisena asiana. Hän muotoili asian seuraavasti:

”Mielestäni tämä on paras mahdollinen paikka ajaa uusia harjoittelijoita sisään – pitää ymmärtää liiketoimintaa, kaivaa tietoa eri ihmisiltä (kommunikointi), ymmärtää sovelluslogiikka ja riippuvuudet ja tehdä toimiva ja liiketoiminnan tarpeita vastaava lopputulos. Jos tämä onnistuu, niin harjoittelijasta on potentiaalia kasvaa hyväksi konsultiksi Cubiqilla.” [37.]

Joitain ratkaisuun liittyviä asioita tekisin varmasti nykyisellä kokemuksella ja osaamisella eri tavalla tai paremmin, jos lähtisin tekemään samaa uudestaan. Esimerkiksi Datamall-sovellus, jolta tiedot luetaan esityskerroksen sovelluksille, sisältää todella paljon sellaista tietoa, jota ei hyödynnetä sovelluksilla ollenkaan. Eli ylimääräistä, epäoleellista tietoa karsisin runsaasti pois. Olen kuitenkin todella ylpeä siitä, mitä sain aikaan, erityisesti kun otetaan huomioon, että lähdin toteuttamaan ratkaisua vain muutaman kuukauden kokemuksella data-analytiikka-alasta ja Qlik Sensestä. Tiedän myös, että ratkaisusta on ollut hyötyä yritykselle ja sen työntekijöille. Jos voisin aloittaa opinnäytetyön tekemisen uudestaan, niin määrittelin aiheen ja työn tavoitteet tarkemmin ja tekisin selkeämmän suunnitelman. Esimerkiksi pelkästään kerroksellisen tietoarkkitehtuurin rakentamisesta olisi varmasti saanut kirjoitettua tarpeeksi ja siitä olisi voinut kertoa yksityiskohtaisemmin. Toisaalta koen, että opinnäytetyön tekemisen tarkoituksena on myös oppia tällaisia asioita ja osata jatkossa toimia tarvittaessa toisin. Kaiken kaikkiaan opinnäytetyön tekeminen oli hyödyllistä ja opin todella paljon erityisesti Qlik Sense -ohjelmistosta.

Cubiqin näkökulmasta katsottuna tiedolla johtamisen edellytykset ovat tämän työn myötä nousseet uudelle tasolle. Päätösten tukena hyödynnettävään tietoon voidaan luottaa ja sitä voidaan hyödyntää tehokkaasti päätöksenteon tukena ja toiminnan ohjaamisessa.

Aikaa ei tarvitse myöskään käyttää niin paljoa erilaisten lukujen tarkistamiseen tai manuaaliseen laskemiseen. Työaikaa voidaan yhä tehokkaammin käyttää liiketoiminnan kehittämiseen ja ennennäkemättömien dataläpimurtojen tuottamiseen yrityksen asiakkaille.

Lähteet

- 1 Cubiqin kotisivut. 2020. Verkkoaineisto. <<https://cubiq.fi/>>. Luettu 1.12.2020.
- 2 Hakusanat "tiedolla" ja "johtaminen". 2020. Verkkoaineisto. Theseus - ammatti-korkeakoulujen opinnäytetyöt ja julkaisut verkossa. <<https://www.theseus.fi/discover?query=tiedolla+johtaminen>>. Luettu 1.9.2020.
- 3 Mitä on tiedolla johtaminen? 2020. Verkkoaineisto. #Tietokiri-hanke. <<https://tietokiri.fi/tiedolla-johtaminen/tiedolla-johtaminen-nain-se-tapahtuu/>>. Luettu 4.7.2020.
- 4 Mitä on tiedolla johtaminen? 2020. Verkkoaineisto. Advian. <<https://www.advian.fi/mita-on-tiedolla-johtaminen>>. Luettu 6.10.2020.
- 5 Blomqvist, Niklas. 2019. Tiedolla johtamisen lyhyt oppimäärä. Verkkoaineisto. <<https://aureolis.com/tiedolla-johtaminen/tiedolla-johtamisen-lyhyt-oppimaara/>>. Luettu 1.9.2020.
- 6 Salenius, Maria & Nieminen Ville. 2019. Tiedolla johtaminen ja sen työvälineet. Verkkoaineisto. <<https://www.kuntaliitto.fi/johtaminen-ja-kehittaminen/tiedolla-johtaminen-ja-sen-tyovalineet>>. Luettu 1.9.2020.
- 7 Barends, Eric & Rousseau, Denise M. & Briner, Rob B. 2014. Evidence-Based Management – The Basic Principles. Verkkoaineisto. Amsterdam: Center for Evidence-Based Management. <<https://www.cebma.org/wp-content/uploads/Evidence-Based-Practice-The-Basic-Principles-vs-Dec-2015.pdf>>. Luettu 4.9.2020.
- 8 Kosonen, Miia. 2019. Tiedolla johtamisen käsikirja (ISBN: 978-952-344-183-5). Kaakkois-Suomen ammattikorkeakoulu.
- 9 Virtanen, Petri & Stenvall, Jari & Rannisto Pasi-Heikki. 2015. Tiedolla johtaminen hallinnossa. Tampere: Tampere University Press.
- 10 Tiedolla johtamisen käsikirja. Verkkoaineisto. <<https://wiki.julkict.fi/julkict/projektit/sote-tietojohdaminen/tiedolla-johtamisen-kasikirja-pdf/view>>. Luettu 5.7.2020.
- 11 Kuntien avainluvut. 2020. Verkkoaineisto. Tilastokeskus. <<https://www.stat.fi/tup/alue/kuntienavainluvut.html>>. Luettu 4.10.2020.
- 12 Hellström, Eeva & Ikäheimo, Hannu-Pekka. 2017. Tieto päätöksenteossa (ISBN 978-952-347-011-8).

- 13 Facebook bans Holocaust denial content. 2020. Verkkoaineisto. <<https://www.bbc.com/news/technology-54509975>>. Luettu 15.10.2020.
- 14 Räsänen, Piritta. 2020. Verkkoaineisto. Tutkimus: Donald Trump on suurin yksittäinen tekijä, jonka kautta koronaviruksesta on levinnyt virheellistä tietoa. <<https://www.hs.fi/nyt/art-2000006656142.html>>. Luettu 3.10.2020.
- 15 Christiansen, Emily. 2014. The Success Stories You Probably Haven't Heard Of: Qlik. Verkkoaineisto. <<https://oresundstartups.com/success-stories-might-missed-qlik/>>. 27.4.2014. Luettu 19.9.2020.
- 16 Richardson, James & Sallam, Rita & Schlegel, Kurt & Kronz, Austin & Sun, Julian. 2020. Magic Quadrant for Analytics and Business Intelligence Platforms. Verkkoaineisto. <<https://www.gartner.com/doc/reprints?id=1-1XYUYQ3I&ct=191219&st=sb>>. 11.2.2020. Luettu 19.9.2020.
- 17 Thoma Bravo – Companies. 2020. Verkkoaineisto. <<https://www.thomabravo.com/companies>>. Luettu 19.9.2020.
- 18 Customer Success Stories. 2020. Verkkoaineisto. QlikTech International AB. <<https://www.qlik.com/us/solutions/customers/customer-stories>>. Luettu 19.9.2020.
- 19 Featured Technology Partners. 2020. Verkkoaineisto. QlikTech International AB. <<https://www.qlik.com/us/products/technology/featured-partners>>. Luettu 19.9.2020.
- 20 Accelerate Business Value Through Data and Analytics - An Approach to Real-time Active Intelligence. 2020. Verkkoaineisto. QlikTech International AB. <<https://www.qlik.com/us/-/media/files/resource-library/global-us/direct/datasheets/ds-accelerate-business-value-through-data-and-analytics-en.pdf>>. Luettu 19.9.2020.
- 21 Bhargava, Rahul & D'Ignazio, Catherine. 2016. Designing Tools and Activities for Data Literacy Learners. Verkkoaineisto. <<https://dam-prod.media.mit.edu/x/2016/10/20/Designing-Tools-and-Activities-for-Data-Literacy-Learners.pdf>>. Luettu 1.10.2020.
- 22 Qlik Data Integration Platform. 2020. Verkkoaineisto. QlikTech International AB. <<https://www.qlik.com/us/products/data-integration-products>>. Luettu 19.9.2020.
- 23 Enabling Analytics with Trusted, Business-Ready Data. 2020. QlikTech International AB. Verkkoaineisto. <<https://www.qlik.com/us/-/media/files/resource-library/global-us/direct/datasheets/ds-qlik-catalog-en.pdf>>. Luettu 19.9.2020.

- 24 Data Analytics Products. 2020. QlikTech International AB. Verkkoaineisto. <<https://www.qlik.com/us/products/analytics-products>>. Luettu 19.9.2020.
- 25 Qlik Sense: webbipohjainen BI-työkalu visuaaliseen analytiikkaan. 2020. Verkkoaineisto. <<https://www.climber.fi/tarjontamme/tyokalumme/qliksense/>>. Luettu 1.10.2020.
- 26 Embedded Analytics. 2020. Verkkoaineisto. <<https://www.qlik.com/us/bi/embedded-analytics>>. Luettu 1.10.2020.
- 27 Augmented intelligence. 2020. Verkkoaineisto. <<https://www.qlik.com/us/bi/augmented-intelligence-and-analytics>>. Luettu 1.10.2020.
- 28 Qlik Sense for administrators - Qlik Sense Enterprise on Windows architecture. 2020. Verkkoaineisto. <https://help.qlik.com/en-US/sense-admin/September2020/Subsystems/DeployAdministerQSE/Content/Sense_DeployAdminister/QSEoW/Deploy_QSEoW/Architecture.htm>. Luettu 1.10.2020.
- 29 Qlik Sense for administrators - Services. 2020. Verkkoaineisto. <https://help.qlik.com/en-US/sense-admin/September2020/Subsystems/DeployAdministerQSE/Content/Sense_DeployAdminister/QSEoW/Deploy_QSEoW/Services.htm>. Luettu 1.10.2020.
- 30 Qlik Sense for administrators – Protecting the platform. 2020. verkkoaineisto. <https://help.qlik.com/en-US/sense-admin/June2020/Subsystems/DeployAdministerQSE/Content/Sense_DeployAdminister/QSEoW/Deploy_QSEoW/Server-Security-Protect-Platform.htm>. Luettu 1.10.2020.
- 31 Qlik Sense for administrators - Qlik Sense Enterprise on Windows on-premises. 2020. Verkkoaineisto. <https://help.qlik.com/en-US/sense-admin/June2020/Subsystems/DeployAdministerQSE/Content/Sense_DeployAdminister/QSEoW/Deploy_QSEoW/Enterprise-deployment.htm>. Luettu 1.10.2020.
- 32 Mahler, Martin & Vitantonio, Juan Ignacio. 2018. Mastering Qlik Sense. Packt Publishing.
- 33 Qlik Sense on Windows – Guidelines for data and fields. 2020. Verkkoaineisto. <https://help.qlik.com/en-US/sense/September2020/Subsystems/Hub/Content/Sense_Hub/Introduction/guidelines-data-fields.htm>. Luettu 1.10.2020.
- 34 The associative difference. 2020. Verkkoaineisto. <<https://www.qlik.com/us/-/media/files/resource-library/global-us/direct/datasheets/ds-the-associative-difference-en.pdf>>. Luettu 1.10.2020.

- 35 Qlik Sense on Windows – Managing security with section access. 2020. Verkkoaineisto. <https://help.qlik.com/en-US/sense/September2020/Subsystems/Hub/Content/Sense_Hub/Scripting/Security/manage-security-with-section-access.htm>. Luettu 1.10.2020.
- 36 Qlik Sense on Windows - Working with variables in the data load editor. 2020. Verkkoaineisto. <https://help.qlik.com/en-US/sense/September2020/Subsystems/Hub/Content/Sense_Hub/Scripting/work-with-variables-in-data-load-editor.htm>. Luettu 1.10.2020.
- 37 Qlik Sense on Windows - Working with QVD files. 2020. Verkkoaineisto. <https://help.qlik.com/en-US/sense/September2020/Subsystems/Hub/Content/Sense_Hub/Scripting/work-with-QVD-files.htm>. Luettu 1.10.2020.
- 38 Qlik Sense on Windows – Binary. 2020. Verkkoaineisto. <https://help.qlik.com/en-US/sense/September2020/Subsystems/Hub/Content/Sense_Hub/Scripting/ScriptRegularStatements/Binary.htm>. Luettu 1.10.2020.
- 39 Pekkala, Mikko. 2020. Sähköpostikeskustelu.

Kysely Qlik Sense -sovellusten tietojen luotettavuudesta edellisten kahden vuoden aikana

Tässä liitteessä esitellään Cubiqin työntekijöille tehty kysely, jolla pyrittiin selvittämään niiden Qlik Sense -sovellusten tietojen luotettavuutta, jotka käyttävät tärkeimpänä data-lähteenään ValueFramen tietovarastosta saatavaa dataa.

ValueFramen dataa käyttävien Cubiqin sisäisten Sense-sovellusten tietojen luotettavuus

Tämän kyselyn tavoitteena on selvittää, miten yrityksen työntekijät arvioivat tiedon luotettavuutta sisäisillä Qlik Sense -sovelluksilla, jotka käyttävät pääasiallisena tietolähteenään ValueFramea.

Vastaukset on anonymoija.

HUOM!

Kyselyssä on tarkoitus arvioida VAIN seuraavia sovelluksia:

- Reporting
- Bonus
- Lead Consultants

* Pakollinen

Taustatiedot

1

Kuinka kauan olet työskennellyt Cubiqilla? *

- Alle vuoden
- 1-2 vuotta
- Yli 2 vuotta

2

Kuulutko johtoryhmään? *

- Kuulun
- En kuulu

3

Arvioi kuinka usein keskimäärin käytät Reporting-, Bonus- tai Lead Consultants -sovellusta. *

- Päivittäin
- Viikottain
- Kuukausittain
- Harvemmin tai en ollenkaan

Seuraava

Sivu 1/3

Arvioi sovelluksilta saamiesi tietojen luotettavuutta edellisten KAHDEN VUODEN ajalta.

Luotettavuutta mitataan tässä kyselyssä kolmella mittarilla, joita ovat: tiedon ajantasaisuus, oikeellisuus ja yhdenmukaisuus sovellusten välillä.

HUOM!

Kyselyssä on tarkoitus arvioida vain seuraavia sovelluksia:

- Reporting
- Bonus
- Lead Consultants

(Jos olet työskennellyt Cubiqilla vähemmän aikaa, arvioi sen ajan perusteella.)

4

Viimeisten kahden vuoden aikana sovelluksilta saamani tieto on ollut pääosin AJAN TASALLA.

1 = Täysin ERI mieltä.

5 = Täysin SAMAA mieltä. *

1 2 3 4 5

5

Viimeisten kahden vuoden aikana sovelluksilta saamani tieto on ollut pääosin OIKEIN.

1 = Täysin ERI mieltä.

5 = Täysin SAMAA mieltä. *

1 2 3 4 5

6

Viimeisten kahden vuoden aikana sovelluksilta saamani tieto on ollut pääosin YHDENMUKAISTA ERI SOVELLUSTEN VÄLILLÄ.

1 = Täysin ERI mieltä.

5 = Täysin SAMAA mieltä. *

1 2 3 4 5

7

Onko jokin sovellus, jonka tietoja et pidä luotettavina? (Ei pakollinen.)

- Reporting
- Bonus
- Lead Consultants

8

Avoin palaute sovelluksiin liittyen: (Ei pakollinen.)

Kirjoita vastaus

Edellinen

Seuraava

Sivu 2/3

Kiitos vastauksista!

Muista painaa Lähetä-nappia!



Edellinen

Lähetä

Sivu 3/3

Kyselyn vastaukset

Tässä liitteessä esitellään liitteessä 1 esiteltyyn kyselyyn saadut vastaukset.

ID	Alkamisaika	Valmistumisaika	Sähköposti	Nimi	Kuinka kauan olet työskennellyt Cubiqilla?	Kuulutko johtoryhmään?	Arvioi kuinka usein keskimäärin käytät Reporting-, Bonus- tai Lead Consultants -sovellusta.
1	10.24.20 15:16:46	10.24.20 15:17:17	anonymous		Yli 2 vuotta	En kuulu	Päivittäin
2	10.24.20 15:45:32	10.24.20 15:46:46	anonymous		1-2 vuotta	En kuulu	Päivittäin
3	10.24.20 15:46:27	10.24.20 15:47:23	anonymous		Yli 2 vuotta	En kuulu	Viikottain
4	10.24.20 15:45:29	10.24.20 15:48:56	anonymous		Yli 2 vuotta	Kuulun	Päivittäin
5	10.24.20 15:53:03	10.24.20 15:53:16	anonymous		1-2 vuotta	En kuulu	Harvemmin tai en ollenkaan
6	10.24.20 15:51:05	10.24.20 15:53:22	anonymous		1-2 vuotta	En kuulu	Viikottain
7	10.24.20 15:52:22	10.24.20 15:54:04	anonymous		Yli 2 vuotta	En kuulu	Kuukausittain
8	10.24.20 15:57:15	10.24.20 15:58:12	anonymous		1-2 vuotta	En kuulu	Viikottain
9	10.24.20 15:55:27	10.24.20 15:59:23	anonymous		Yli 2 vuotta	En kuulu	Viikottain
10	10.24.20 16:00:52	10.24.20 16:01:27	anonymous		1-2 vuotta	En kuulu	Harvemmin tai en ollenkaan
11	10.24.20 16:01:14	10.24.20 16:01:33	anonymous		Yli 2 vuotta	En kuulu	Harvemmin tai en ollenkaan
12	10.24.20 16:00:39	10.24.20 16:02:24	anonymous		Yli 2 vuotta	En kuulu	Viikottain
13	10.24.20 16:01:55	10.24.20 16:02:53	anonymous		Yli 2 vuotta	Kuulun	Päivittäin
14	10.24.20 16:28:25	10.24.20 16:28:56	anonymous		Yli 2 vuotta	En kuulu	Päivittäin
15	10.24.20 16:42:45	10.24.20 16:44:20	anonymous		Yli 2 vuotta	Kuulun	Päivittäin
16	10.24.20 16:47:44	10.24.20 16:48:29	anonymous		Yli 2 vuotta	En kuulu	Päivittäin
17	10.24.20 17:49:21	10.24.20 17:51:54	anonymous		Yli 2 vuotta	En kuulu	Viikottain
18	10.24.20 23:49:31	10.24.20 23:51:42	anonymous		Yli 2 vuotta	En kuulu	Päivittäin
19	10.25.20 8:04:18	10.25.20 8:05:26	anonymous		Yli 2 vuotta	En kuulu	Harvemmin tai en ollenkaan
20	10.25.20 10:29:40	10.25.20 10:29:51	anonymous		Yli 2 vuotta	En kuulu	Päivittäin
21	10.25.20 11:06:10	10.25.20 11:07:01	anonymous		Yli 2 vuotta	En kuulu	Viikottain
22	10.25.20 15:05:13	10.25.20 15:06:25	anonymous		Yli 2 vuotta	En kuulu	Viikottain
23	10.25.20 16:08:47	10.25.20 16:10:49	anonymous		Yli 2 vuotta	En kuulu	Päivittäin
24	10.25.20 20:32:12	10.25.20 20:33:10	anonymous		1-2 vuotta	En kuulu	Viikottain
25	10.26.20 8:10:56	10.26.20 8:12:08	anonymous		1-2 vuotta	En kuulu	Viikottain
26	10.26.20 8:24:15	10.26.20 8:26:27	anonymous		Yli 2 vuotta	En kuulu	Kuukausittain
27	10.26.20 8:36:32	10.26.20 8:37:17	anonymous		1-2 vuotta	En kuulu	Viikottain

ID	Viimeisten kahden vuoden aikana sovelluksilta saamani tieto on ollut pääosin AJAN TASALLA.	Viimeisten kahden vuoden aikana sovelluksilta saamani tieto on ollut pääosin OIKEIN.	Viimeisten kahden vuoden aikana sovelluksilta saamani tieto on ollut pääosin YHDENMUKAISTA ERI SOVELLUSTEN VÄLILLÄ.	Onko jokin sovellus, jonka tietoja et pidä luotettavina? (Ei pakollinen.)
	1 = Täysin ERI mieltä. 5 = Täysin SAMAA mieltä.	1 = Täysin ERI mieltä. 5 = Täysin SAMAA mieltä.	1 = Täysin ERI mieltä. 5 = Täysin SAMAA mieltä.	
1	5	5	5	5
2	5	5	5	5
3	5	5	5	5
4	5	4	4	4
5				
6	5	5	5	4
7	5	5	5	5
8	5	5	5	4
9	5	5	5	5
10				
11				
12	5	5	5	5
13	4	5	5	5
14	5	5	5	5
15	5	5	5	5
16	5	5	5	5
17	5	5	5	5
18	5	4	4	4
19				
20	5	5	5	5
21	5	5	5	5
22	2	1	1	2
23	4	5	5	3
24	5	5	5	5
25	5	5	5	3
26	5	5	5	5
27	4	5	5	5