



Lupapiste verkkopalvelun tietovarasto- ja raportointiratkaisu

Ville Matikainen

Opinnäytetyö
Tammikuu 2015
Tietojenkäsittelyn
koulutusohjelma

TAMPEREEN AMMATTIKORKEAKOULU
Tampere University of Applied Sciences

TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Tietojenkäsittelyn koulutusohjelma

MATIKAINEN, VILLE:

Lupapiste verkkopalvelun tietovarasto- ja raportointiratkaisu

Opinnäytetyö 51 sivua, joista liitteitä 4 sivua
Tammikuu 2015

Opinnäytetyön tavoitteena oli kehittää toimeksiantajayritys Solita Oy:n sisäiseen käyttöön tulevaa raportointia. Lupapiste on asiointipalvelu, jossa kansalaiset, viranomaiset ja yritykset voivat hoitaa sähköisesti rakentamiseen liittyvät lupa-asiansa. Tarkoitus oli toteuttaa Solita Oy:n sisäiseen käyttöön tuleva tietovarasto- ja raportointiratkaisu Lupapisteen tarpeisiin.

Opinnäytetyössä käytettiin konstruktivistista ja toiminnallista tutkimusotetta. Tässä opinnäytetyössä Business intelligence -arkkitehtuuri koostuu lähdejärjestelmistä, ETL-prosessista, tietovarastosta ja raportoinnista. ETL-prosessissa poimittiin ensin raportoinnissa tarvittavat tiedot Lupapisteen lähdejärjestelmästä, jonka jälkeen tiedot yhtenäistettiin ja lopuksi ladattiin tietovarastoon. Raportointiratkaisu toteutettiin vakio- ja parametriraportointina, jossa loppukäyttäjä voi rajata hakuehdoin raportin palauttamaa tietosisältöä. Solita Oy käyttää vakioraporttien tietosisältöä asiakkaiden laskutusperusteena, mikä oli yksi projektin tavoitteista.

Projektin lopputuloksena käyttöön otettiin keskitetty tietovarasto- ja raportointiratkaisu, joka täytti projektille asetetut tavoitteet. Tietovaraston tiedoista muodostettiin kolme vakioraporttia. Toteutus vastasi alussa asetettuihin raportointitarpeisiin ja se toteutettiin sovitussa aikataulussa. Raportointia on mahdollisuus kehittää ja laajentaa tulevaisuudessa kuntien viranomaisorganisaatioiden ja ympäristöministeriön käyttöön. Uusia raportteja voidaan jatkossa muodostaa mittaamalla hakemusten käsittelyaikojen pituuksia ja seuraamalla käsittelyaikojen kehityssuuntia. Myöhemmin Lupapisteen tietovarastosta voitaisiin tuottaa julkista tietoa, josta julkisen sektorin organisaatiot voisivat olla kiinnostuneita.

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Degree Programme in Business Information Systems

MATIKAINEN, VILLE:

Data warehouse and reporting solution based on Lupapiste web service

Bachelor's thesis 51 pages, appendices 4 pages

January 2015

The objective of this thesis was to develop an internal business intelligence solution to the client, Solita Oy. Lupapiste is a web service where citizens and companies can apply for a construction permit. The purpose of this thesis was to implement customized data warehouse and business intelligence solution for Lupapiste.

The thesis was completed using constructive and practice-based research methods. In this project, the components of business intelligence architecture used are source systems, ETL process, data warehouse and reporting. First in ETL process, the data was extracted from source systems to staging area. Secondly the data was standardized and finally loaded from the staging area to the enterprise data warehouse. Reporting solution consist of constant and parameterized reports where end users can define the outcome of the reports. Solita Oy invoices its customers based on reports data content.

The outcome of the project was an implemented data warehouse and a reporting system. Three constant reports were created based on data from the data warehouse. The project was successful and implemented on time. In the future, there is a possibility to develop and expand reporting to the needs of the Ministry of Environment. Later on the data warehouse of Lupapiste could produce open data to the organizations of public sector.

Key words: business intelligence, data warehouse, ETL, reporting

SISÄLLYS

1	JOHDANTO.....	8
2	BUSINESS INTELLIGENCE	10
	2.1 Määritelmä.....	10
	2.2 BI-arkkitehtuuri	11
3	LUPAPISTE.....	13
	3.1 Lupapisteen käsitteet ja käsitemalli	14
	3.2 Nykytilanne ja raportointitarpeet	15
	3.3 Lupapisteen BI-arkkitehtuuri.....	18
4	LÄHDEJÄRJESTELMÄN KUVAUS.....	19
	4.1 Lähdejärjestelmät ja tietolähteet	19
	4.1.1 Operatiiviset järjestelmät	19
	4.1.2 Strukturoitu ja strukturoimaton tieto.....	20
	4.2 Lupapisteen lähdejärjestelmä.....	20
	4.2.1 Hakemukset.....	21
	4.2.2 Organisaatiot	22
5	TIETOVARASTOINTI.....	23
	5.1 Tietovarastotyypit	24
	5.2 Tietovaraston mallinnus.....	25
	5.2.1 Fakta- ja dimensiotaulut.....	25
	5.2.2 Tähti- ja lumihiutalemalli.....	26
	5.2.3 Data vault	27
	5.3 Lupapisteen tietovarasto	28
	5.3.1 Tietokantamalli	28
	5.3.2 Julkaisumalli	29
6	ETL-PROSESSI.....	31
	6.1 ETL-prosessin menetelmät ja vaiheet.....	31
	6.1.1 Poiminta (Extract)	32
	6.1.2 Muokkaus (Transform)	33
	6.1.3 Lataus (Load)	33
	6.2 Lupapisteen ETL-Toteutus	34
	6.2.1 Poiminta (Extract)	35
	6.2.2 Muokkaus (Transform)	36
	6.2.3 Lataus (Load)	38
7	RAPORTOINTI	39
	7.1 Lupapisteen Neuvontapyynnöt ja hakemukset -raportin muodostaminen.....	40

7.2 Datan rajaus ja valmis raportti	41
8 POHDINTA.....	44
LÄHTEET	46
LIITTEET	48
Liite 1. Hakemukset-rajapinnan tietosisältö	48
Liite 2. Organisaatiot-rajapinnan rakenne	50
Liite 3. Lista tietovaraston tauluista	51

LYHENTEET JA TERMIT

BI	Business intelligence, liiketoimintatiedon hallinta
CRM	Customer Relationship Management, asiakkuudenhallinta
CSV	Comma-separated Values on tiedostomuoto, jolla voi tallentaa taulukkomuotoista tietoa tekstitiedostoon
Data	Data on symboleita, lukuja, kirjaimia ilman kontekstia
Data mart	Paikallistietovarasto
Data vault	Tietovarastietokannan mallinnusmenetelmä
EDW	Enterprise Data Warehouse on keskitetty yritystason tietovarasto
ETL	Extract, transform, load (poiminta, muokkaus ja lataus) on tietovarastoinnin prosessi, jossa tiedot ladataan tietolähteistä tietovarastoon
ERP	Enterprise Resource Planning, toiminnanohjausjärjestelmä
HTTP	Hypertext Transfer Protocol on protokolla, jota selaimet ja www-palvelimet käyttävät tiedonsiirtoon
HTTP autentikointi	HTTP Basic authentication on tunnistautumismenetelmä, joka on yksinkertainen tapa todentaa käyttäjä http-palveluun
Informaatio	Tietoa tarkastellessa pystytään ymmärtämään mitä se kuvaa ja sen pohjalta tekemään johtopäätöksiä

JSON	JavaScript Object Notation on yksinkertainen avoimen standardin tiedostomuoto tiedonvälitykseen
Lumihiutalemalli	Snowflake schema on tietovarastotietokannan suunnittelumenetelmä
Lähdejärjestelmä	Tietolähde, josta ladataan tietoa esimerkiksi tietovarastoon
ODS	Operational Data Store on operatiivisten tietokantojen ja tietovaraston väliin rakennettu erillinen välivarasto
OLAP	Online Analytical Processing, moniulotteinen analysointi
OLTP	Online Transaction Processing on tapahtumankäsittelyjärjestelmä, joka pystyy käsittelemään isoja määriä reaaliaikaisia tapahtumia hyvällä vasteajalla
Operatiivinen järjestelmä	Perussovelluksen käyttöä varten suunniteltu tietokanta, johon tallennetaan liiketoimintaan liittyviä transaktiotapahtumia
Rajapinta	Määritelmä, jonka avulla sovellukset voivat vaihtaa tietoja keskenään
Tieto	Data, jolla on konteksti
Tietovarasto	DW, Data Warehouse on monien käyttäjäryhmien yhteiskäyttöinen tietokanta
Tietovarastointi	DW, Data Warehousing on yrityksen tietoresurssien saattaminen helposti saataville raportointia ja analysointia varten.
XML	Extensible Markup Language on xml-merkinäkielellä kirjoitettu rakenteinen tekstidokumentti

1 JOHDANTO

Tieto on yritykselle arvokasta pääomaa ja nykypäivänä sen oikeanlainen hyödyntäminen liiketoiminnassa on haasteellista. Tieto on usein hajallaan ympäri organisaatiota, eikä eri tietojärjestelmissä olevia tärkeitä tietoja osata hyödyntää. Yksi tiedon hyödyntämismenetelmä on tietovarastointi, jossa tiedot tallennetaan keskitetysti tietovarastoon. Tieto itsessään ei vielä tuota lisäarvoa organisaatiolle, vaan tiedosta täytyy jalostaa informaatiota päätöksenteon tueksi. Business intelligence (BI) on systemaattista yrityksen suorittamaa liike-elämän tietojen keräämistä, tallennusta ja analysointia. Tässä opinnäytetyössä käytetään termiä BI. BI-ratkaisut ovat yksi menetelmä tuottaa suuresta datamassasta hyödyllistä tietoa visuaalisessa muodossa. ”Yritys, joka pystyy hallitsemaan ja hyödyntämään tietonsa saa merkittävän kilpailuedun”. (Hovi, Koistinen & Ylinen, 2001, 19).

Opinnäytetyön toimeksiantaja Solita Oy on 1996 perustettu digitaalisen liiketoiminnan asiantuntijayritys, joka luo uutta liiketoimintaa ja palveluja yrityksille ja julkishallinnolle. Siinä yhdistyy digitaalinen teknologia, liiketoimintaprosessit ja sisällöt uudella tavalla. Solita Oy luo digitaalisia ratkaisuja ja verkkopalveluita sähköiseen liiketoimintaan ja asiointiin sekä tiedolla johtamiseen. Päivittäin arjessa Solita Oy näkyy esimerkiksi vähittäiskaupan verkkopalveluiden tai julkishallinnon sähköisten palveluiden kautta. Toimipisteitä on Tampereella, Helsingissä ja Oulussa, jossa työskentelee yli 300 digitaalisen liiketoiminnan asiantuntijaa. (Solita Oy 2014.)

Opinnäytetyön tavoitteena on kehittää Solita Oy:n sisäiseen käyttöön tulevaa raportointia. Tarkoituksena on toteuttaa Lupapiste verkkosovellukselle tietovarasto, jonka tiedoista saadaan muodostettua valmiita raportteja BI-raportointityökaluilla. Valmiiden raporttien tietojen perusteella Solita Oy voi laskuttaa asiakkaita. Tässä opinnäytetyössä projektilla tarkoitetaan Lupapisteen toteutusprojektia. Opinnäytetyön toteutuksessa käytettiin tutkimusmenetelmänä konstruktivistista- ja toiminnallista tutkimusotetta.

Tyypilliseen BI-arkkitehtuuriin kuuluu lähdejärjestelmät, tietovarastot, ETL (Extract, Transform, Load), raportointi ja niihin liittyvät menetelmät. Opinnäytetyön BI-arkkitehtuurin toteutus on räätälöity Lupapisteen tarpeisiin. Tässä opinnäytetyössä ei

käsitellä metamallinnusta, tietojen suunnittelua ja jatkojalostusta, eikä projektinhallintaa.

Toisessa luvussa määritellään Business intelligence käsitteenä, sekä avataan BI-arkkitehtuuriin kuuluvat prosessit. Opinnäytetyön luvussa kolme esitellään Lupapisteelle toteutettavan projektin tietovarasto ja raportointitarpeet. Luvusta neljä alkaen jokainen luku sisältää teoriaosuuden ja sitä vastaavan toteutuksen Lupapisteen projektista. Luvussa neljä lukijalle selvennetään millaisista lähteistä yritys pyrkii tietoa keräämään. Luvussa esitellään myös Lupapisteen tietomalli, joka toimii tietovaraston tietolähteenä.

Luvussa viisi määritellään tietovarastointi ja yleisimmät tietovarastotyypit. Lisäksi luvussa käydään läpi tietovaraston mallinnus, jonka tarkoitus on kuvata loogisella tasolla tietovaraston rakenne. Teoriaosuudessa käydään läpi tietovaraston kolme yleisintä tietovarastomallia. Kuudes luku sisältää ETL-prosessissa käytettävät menetelmät ja vaiheet.

Luvussa seitsemän esitellään BI-arkkitehtuurin viimeinen vaihe, johon kuuluu raportointi ja tiedon analysointi. Tässä luvussa tuodaan esille raportointi- ja analysointimenetelmiä. Pääpaino on raportoinnissa, koska Lupapisteen projektissa halutaan kehittää raportointiratkaisua. Toteutuksessa kuvataan yhden vakioraportin muodostuminen.

Lopuksi opinnäytetyössä pohditaan Lupapisteen projektin toteutuksen onnistumista, sekä analysoidaan projektissa kohdattuja haasteita. Opinnäytetyön onnistumista tarkastellaan kahdesta näkökulmasta, vastaako opinnäytetyön empiirinen osuus esitettyä teoriaa, sekä vastaako projekti asetettuja tavoitteita. Lopuksi esitetään kehitysehdotuksia, kuinka raportointitarpeita voisi tulevaisuudessa kehittää.

2 BUSINESS INTELLIGENCE

Tiedon määrän kasvaessa yritykset tarvitsevat työkaluja kehittääkseen tiedonhallintaa. Tiedonhallinnan haasteena on löytää kasvavasta datamäärästä oleellinen tieto liiketoiminnan kannalta. Business intelligence mahdollistaa suurten datamäärien keräämisen sekä tuottamaan siitä hyödyllistä informaatiota päätöksenteon tueksi.

2.1 Määritelmä

Business intelligence -termi määritellään eri yhteyksissä eri tavoilla, jotka vaihtelevat lähteestä riippuen. Yksinkertaisimmillaan BI on systemaattista yrityksen suorittamaa liike- tai muusta toiminnasta syntyvien tietojen keräämistä, tallennusta ja analysointia. BI:stä käytetään myös esimerkiksi synonyymia liiketoimintatiedon hallinta, joka koetaan suppeimmasta näkökulmasta lähinnä raportoinniksi, kun taas laajemmasta näkökulmasta katsottuna siihen liitetään myös muita johtamiskäsitteitä ja –työkaluja, kuten suorituskyvyn mittaaminen ja tietojohdaminen. Määritelmiä on monia, mutta usein liiketoimintatiedon hallinta käsitetään yrityksen laajuiseksi strategiatyökaluksi, joka tukee raportointia, analysointia sekä päätöksentekoa useilla eri organisaatiotasolla. (Hovi, Hervonen & Koistinen 2009, 80-81). Yrityksmaailmassa käytetään termiä Business intelligence tai BI. Termin sisällön ja merkityksen määritelmä on jakaantunut vuosien varrella kahteen ajattelutapaan.

Kvantitatiivinen näkemyksessä yritys pyrkii hyödyntämään ja hallitsemaan sen sisällä syntyvää liiketoimintatietoa. Alkuperäisenä lähteenä tiedolle toimii yrityksen sisäiset tietokannat ja tietojärjestelmät. Tieto on relaatiotietokantoihin helposti tallennettavaa, pääosin numeerista dataa. (Hovi ym. 2009, 78)

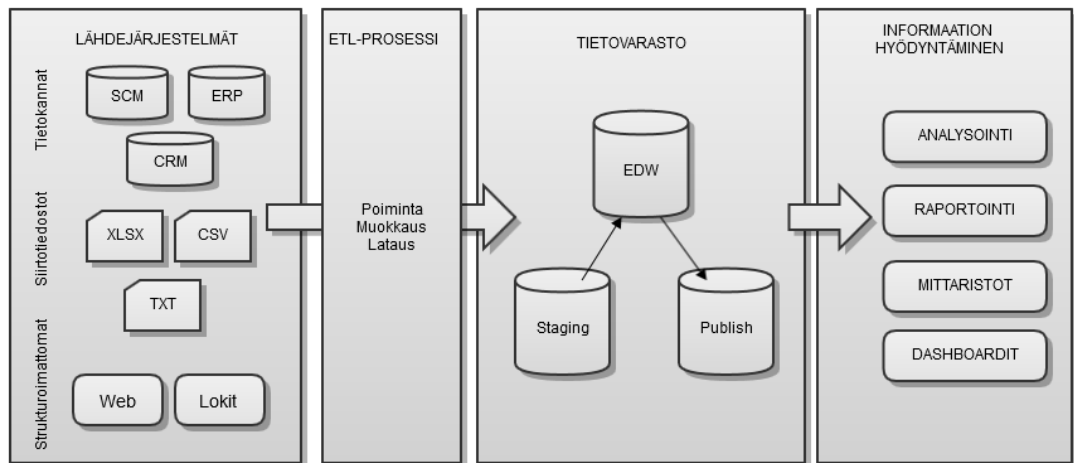
Kvalitatiivisessa näkemyksessä yritys pyrkii keräämään tietoa kilpailijoista ja markkinoista, jota hyödynnetään ja hallitaan. Näitä tietoja kerätään yrityksen ulkopuolisilta tietopankeilta, uutistoimistoilta ja julkisista tietolähteistä. Tieto on yleensä asiakirja- ja dokumenttipohjaista laadullista aineistoa. (Hovi ym. 2009, 78)

Näkemyksissä löytyy paljon yhdenmukaisuuksia, koska molemmissa tulkinnoissa tavoitteena on tarjota loppukäyttäjälle parempaa tietoa päätöksenteon tueksi. Turban, Sharda, Delen ja King (2011, 8) määrittelevät BI:n niin, että BI sisältää alikäsitteitä, joita ovat arkkitehtuuri, työkalut, tietokannat, analysointi, sovellukset ja metodologiat. Alikäsitteet ovat keskeisessä roolissa, kun yritys kerää tietoa operatiivisista järjestelmistä nykyhetkestä ja historiasta. Tietoa voidaan hyödyntää ja jalostaa siten, että se tukee päätöksentekoa ja tekee siitä laadukkaampaa ja tehokkaampaa. (Turban ym. 2011, 8)

Teachtarget (2014) määrittelee BI-käsitteen teknisestä näkökulmasta laajaksi ryhmäksi erilaisia sovelluksia ja teknologioita, joilla poimitaan, muokataan, ladataan ja esitetään yrityksen liiketoiminnan synnyttämää tietoa. Lähdejärjestelmistä koottu tieto yhdenmukaistetaan ja jalostetaan siten, että yrityksen eri käyttäjätasot saavat hyödynnettyä tietoa parhaalla mahdollisella tavalla, kuten raportoinnissa ja analysoinnissa. BI-ratkaisussa pyritään kuvaamaan ne järjestelmät ja toteutukset, joiden avulla loppukäyttäjät pääsevät hyödyntämään liiketoiminnasta syntyvää tietoa. (Teachtarget 2014.)

2.2 BI-arkkitehtuuri

Tietovarastointi ja BI ovat kehittyneet viime vuosina tietoteknisestä näkökulmasta. Yritykset ja julkisen sektorin organisaatiot ovat huomanneet, että operatiivisissa järjestelmissä olevat tiedot ovat vaikeasti saatavilla analysointeja ja raportointia varten. Tässä opinnäytetyössä BI-arkkitehtuuri koostuu seuraavista komponenteista: lähdejärjestelmät, ETL-prosessi, tietovarasto ja informaation hyödyntäminen. Tietovarastoinnilla tarkoitetaan tietojen yhdenmukaistamista ja yhdistämistä eri lähteistä yhteiseen tietovarastoon. (Hovi ym. 2009, XI). BI-arkkitehtuuri koostuu tietojen yhdistämisestä lähdejärjestelmistä tietovarastoon ja informaation hyödyntämisestä (kuvio 1).



Kuvio 1. BI-arkkitehtuuri

Tieto siirtyy lähdejärjestelmistä ETL-prosessien kautta tietovarastoon, josta tietoa voidaan hyödyntää, esimerkiksi muodostamalla erilaisia raportteja. Hovin (2009, 86) mukaan BI-arkkitehtuuri koostuu seuraavista teknologiakomponenteista:

1. Useimpien BI-ratkaisujen tietolähteinä toimivat yrityksen liiketoiminnan prosesseja tukevat operatiiviset järjestelmät, siirtotiedostot tai strukturoimattomat tietolähteet.
2. ETL-prosessi (Extract, Transform, Load) on keskeinen tietojen integroinnin vaihe, jossa käytetään tietojen yhdistämiseen tarkoitettuja työkaluja. Prosessissa tiedot poimitaan lähdejärjestelmistä, muokataan ja yhtenäistetään, jonka jälkeen tiedot ladataan tietovarastoon.
3. Tietovarastoon kootaan lähdejärjestelmien tieto yhtenäiseen muotoon, jotta tietoa voidaan hyödyntää raportoinnissa ja analysoinnissa.
4. Informaation hyödyntäminen on BI:n viimeinen vaihe. Siinä käytetään menetelmiä, joilla loppukäyttäjät saavat BI-ratkaisun tiedot päätöksenteon tueksi. (Hovi ym. 2009, 86)

3 LUPAPISTE

Lupapiste on rakentamisen lupien sähköinen asiointipalvelu, joka helpottaa käyttäjien ja viranomaisten välistä tiedonkulkua ja arkea. Kuntien rakennusvalvonnassa palvelaan vuosittain 1,5 miljoonaa asiakasta ja rakentamiseen liittyviä lupa-asioita hoidetaan noin 100 000. Lupa-asiat hoidetaan nykyisin virastojen palvelupisteissä pääosin puhelimen ja sähköpostin välityksellä. Hakemukset ja materiaalit toimitetaan yleensä paperimuotoisena, joiden tiedot syötetään viranomaisjärjestelmiin käsin.(Solita Oy 2014c.)

Uudessa Lupapiste-palvelussa kansalaiset, viranomaiset ja yritykset hoitavat lupa- ja ilmoitusasiansa, sekä niissä tarvittavat hakemus-, liite-, ja suunnittelutiedot sähköisesti verkkopalvelussa. Lupapiste kokoaa rakentajan, suunnittelijan ja kunnan viranomaiset samaan palveluun. Palvelu mahdollistaa heidän välisen vuorovaikutuksen hakuprosessin aikana. Lupapiste on riippumaton virastojen aukioloajoista, koska asiointi onnistuu ympäri vuorokauden. Kaikki lupaprosessin asianosaiset, kuten rakennuttaja, pääsuunnittelija tai rakennusvalvoja ovat koko ajan tietoisia prosessin etenemisestä. Tämä nopeuttaa käyttäjien ja viranomaisten välistä tiedonkulkua, mahdollistaen miljoonien eurojen säästöpotentiaalin vuosittain.(Solita Oy 2014a.)

Ympäristöministeriö vastaa yhteistyössä Solita Oy:n kanssa Lupapiste palvelusta. Solita Oy:n vastuulla on Lupapiste-palvelun kehittäminen ja käyttöönotto. Käyttöönoton jälkeen Solita Oy ja yhteistyökumppanit vastaavat palvelun ylläpidosta, markkinoinnista ja kehittämisestä.(Solita Oy 2014a.)

Palvelu otettiin käyttöön muutamassa pilottikunnassa keväällä 2013, ja palvelua käyttävien kuntien lukumäärä on kasvanut moninkertaisesti. Mukana olevat pilottikunnat ovat asukasmäärältään erikokoisia ja edustavat laajaa maantieteellistä otosta. Aluksi Lupapisteen palvelulla hoidettiin rakennusvalvonnan lupakäsittelyt, mutta vuoden 2013 loppuun mennessä Lupapiste laajentui rakennusvalvonnan luvista kuntien ympäristö-, sijoitus-, kaivuu- ja yleisten alueiden käytön lupien palveluksi.(Solita Oy 2014a.) Vuoden 2014 aikana Lupapiste on jatkanut kasvuaan ja vuoden lopussa Lupapisteen käyttöön ottavien kuntien määrä on kasvanut sataan.

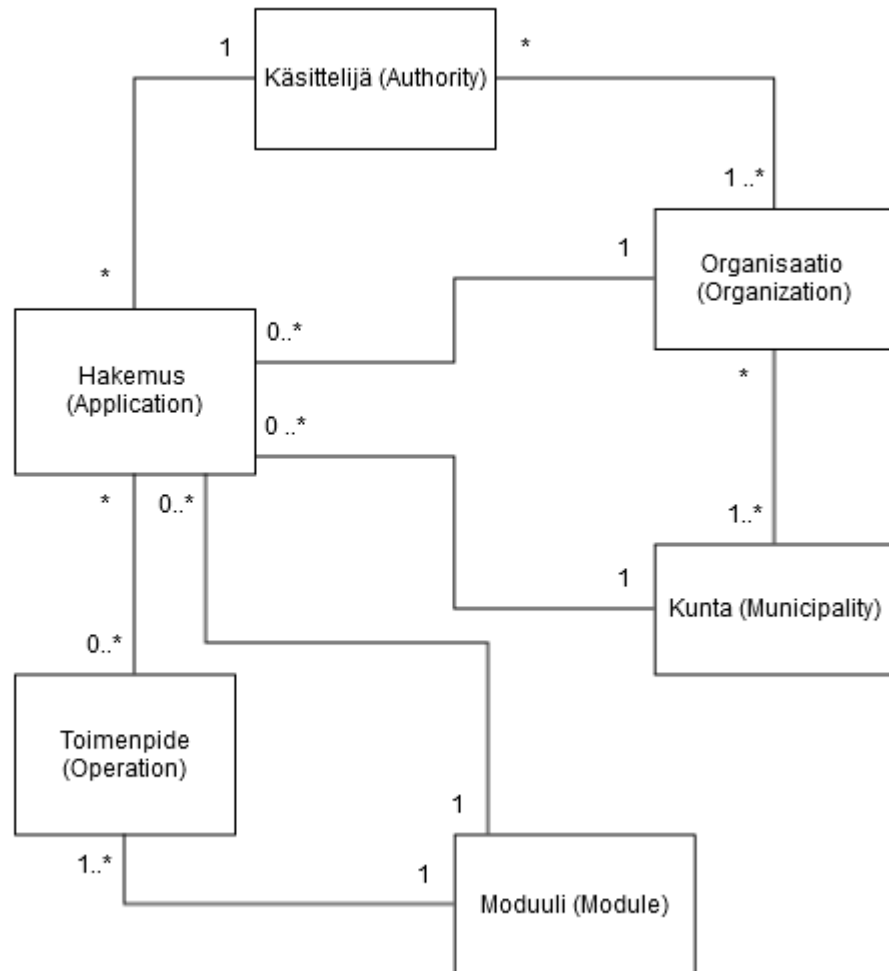
3.1 Lupapisteen käsitteet ja käsitemalli

Tietojenkäsittelyssä **käsite** kuvaa kohdealueen esinettä, asiaa, henkilöä, käsitettä tai paikkaa, josta halutaan tallentaa tietoa ja muodostaa taulu tietokantaan. (Hovi, Huotari & Lähdenmäki 2005, 35). Alla olevassa taulukossa on kuvattu Lupapisteen keskeisimmät käsitteet ja niiden selitteet.

TAULUKKO 1. Lupapisteen keskeisimmät käsitteet.

Käsite	Selite
Hakemus	<ol style="list-style-type: none"> 1. Asiointiprosessi, joka yleensä sisältää päätöksen. 2. Lupahakemus on asiakkaan tuottama tietorakenne tai asiakirja, johon asiakas täyttää lupahakemuksen käsittelyn kannalta tarpeellisen tiedon. Lupahakemus voi sisältää useita liitteitä.
Kunta	Kaikki hakemuksia käsittelevät kunnat.
Moduuli	Hankkeen lupatyyppi, esimerkiksi R = Rakennusvalvonta
Organisaatio	Viranomaisorganisaatio, esimerkiksi Ähtärin rakennusvalvonta.
Toimenpide	Hakemuksella on 0-n toimenpiteitä, esimerkiksi rakennuksen purku
Käsittelijä	Hakemuksen käsittelijä (viranomainen).

Käsiteanalyysi on tietokantoihin johtavan suunnitteluprosessin vaihe, jossa selvitetään mitkä ovat tietokantaan talletettavat tiedot ja niiden väliset yhteydet. Lopputuloksena syntyvä **käsitemalli** määrittelee pohjan tietokannan fyysiselle rakenteelle. Käsitemalli toimii projektin eri osapuolien kommunikaatiovälineenä, ja sitä tarkennetaan prosessin edetessä, kunnes lopulta voidaan toteuttaa itse fyysinen tietokanta tauluineen. (Hovi ym. 2005, 32). Kuviossa 2 on kuvattu Lupapisteen käsitteet ja käsitteiden väliset suhteet.



Kuvio 2. Lupapisteen käsitelmä.

Yhdellä hakemuksella voi olla useita tai ei yhtään toimenpidettä ja siihen liittyy yksi käsittelijä, moduuli, kunta ja organisaatio. Yhteen moduuliin voi liittyä useita toimenpiteitä. Organisaatio voi olla virasto, esimerkiksi rakennusvalvonta, joka käsittelee useamman kunnan hakemuksia. Käsittelijöitä voi olla enimmillään yksi, mutta he voivat kuulua useampaan organisaatioon.

3.2 Nykytilanne ja raportointitarpeet

Lupapisteelle on syntynyt erilaisia raportointitarpeita sähköisen rakentamisen lupien asiointipalvelun käytöstä. Solita Oy tarvitsee sisäiseen käyttöön tulevan tietovarastoratkaisun Lupapisteen raportointitarpeisiin. Lupapisteen jalostetuilla tiedoilla voidaan esimerkiksi seurata tietyn aikavälin neuvontapyyntöjen ja hakemusten määrää ja laskuttaa asiakkaita tämän perusteella. Alla olevissa kuvissa on määritelty

kolmen vakioraportin hakuehdot ja halutut tulokset. Ensimmäisen valmiin raportin on tarkoitus näyttää kuinka paljon Lupapisteeseen on perustettu uusia neuvontapyyntöjä ja hakemuksia kunnassa (kuvio 3).

Uudet neuvontapyynnöt ja hakemukset
Rajausehdot:
<ul style="list-style-type: none"> • Aikaväli • Moduuli
Halutut tulokset:
<ul style="list-style-type: none"> • Organisaation nimi • Kuukausi • Moduuli • Neuvontapyyntöjen määrä • Hakemusten määrä

Kuvio 3. Raportin 1 rajausehdot ja lopputulos

Rajausehtoina käyttäjä voi määrittää haulle aikavälin ja moduulin. Aikavälillä tarkoitetaan tässä yhteydessä, miltä vuodelta ja kuukaudelta tiedot halutaan. Moduuli tarkoittaa hankkeen lupatyyppiä. Raportin tietosisältö tulostetaan organisaation nimen, kuukauden ja moduulin mukaisessa järjestyksessä. Käyttäjä ei pysty valitsemaan raportille tulostettavien tietojen järjestystä. Toinen valmis raportti sisältää, kuinka paljon asiointeja Lupapisteessä on vireillä ja missä tiloissa ne ovat (kuvio 4).

Asiointien tilat
Rajausehdot:
<ul style="list-style-type: none"> • Moduuli • Organisaation nimi
Halutut tulokset:
<ul style="list-style-type: none"> • Organisaation nimi • Moduuli • Tila • Neuvontapyyntöjen määrä • Hakemusten määrä

Kuvio 4. Raportin 2 rajausehdot ja lopputulos

Rajausehtoina ovat moduuli ja organisaatio. Organisaation nimi tarkoittaa viranomaisorganisaatiota, joka käsittelee neuvontapyynnöt ja hakemukset. Raportin tietosisältö tulostetaan organisaation nimen, moduulin ja tilan mukaisessa järjestyksessä. Käyttäjä ei pysty valitsemaan raportille tulostettavien tietojen järjestystä. Kolmannessa raportissa lasketaan Lupapisteen neuvontapyyntöjen ja hakemusten kokonaismäärät (kuvio 5).

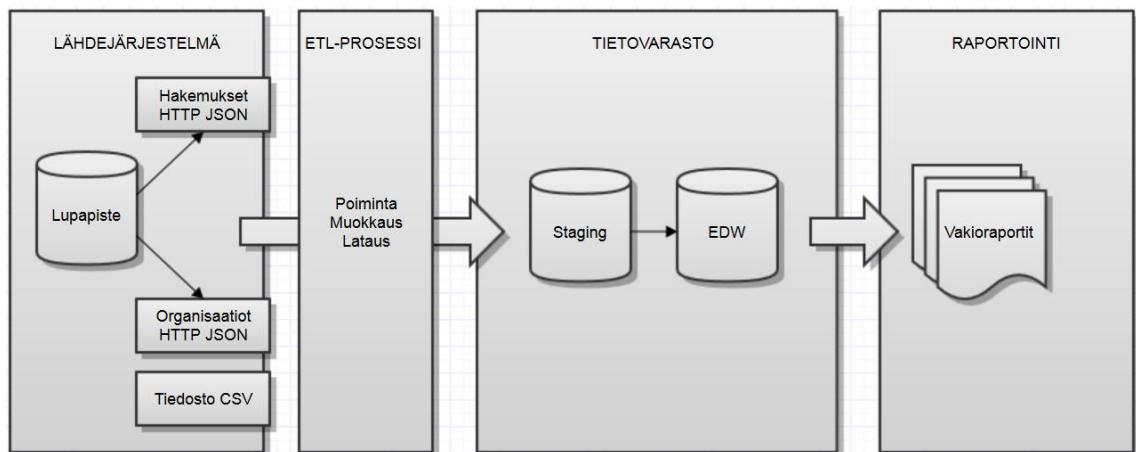
Kokonaismäärät
Rajausehdot:
<ul style="list-style-type: none"> • Aikaväli • Moduuli
Halutut tulokset:
<ul style="list-style-type: none"> • Organisaation nimi • Kuukausi • Moduuli • Neuvontapyyntöjen määrä • Hakemusten määrä

Kuvio 5. Kolmannen raportin rajausehdot ja lopputulos

Rajausehtoina käyttäjä voi määrittää hakemuksen aikavälin ja moduulin. Raportin tietosisältö tulostetaan organisaation nimen, kuukauden ja moduulin mukaisessa järjestyksessä. Käyttäjä ei pysty valitsemaan raportille tulostettavien tietojen järjestystä.

3.3 Lupapisteen BI-arkkitehtuuri

Projektin tavoitteena on kehittää Solita Oy:n sisäiseen käyttöön tulevaa raportointia. Tavoite saavutetaan toteuttamalla tietovarasto, josta saadaan muodostettua BI-raportointityökalulla valmiita raportteja. Tulevaisuudessa palvelua on tarkoitus laajentaa niin, että Lupapisteen käyttäjille tarjotaan mahdollisuus hakea tietoa raportoinnin avulla. Kuviossa 6 on kuvattu Lupapisteen tietovirta lähdejärjestelmästä raportointiin.



Kuvio 6. Lupapisteen BI-arkkitehtuuri.

Lähdejärjestelmistä tiedot haetaan ETL-prosessilla, mitkä tallennetaan tietovarastoon raportointia varten. BI-ratkaisussa raportointityökalulla tehdään kyselyjä keskitetystä tietovarastosta, joka palauttaa käyttäjän rajaamat tiedot muodostaen vakioraportin.

4 LÄHDEJÄRJESTELMÄN KUVAUS

Yrityksen päättäjät tarvitsevat reaaliaikaisia yhteenvedoja ja tilastoja päätöksenteon tueksi eri lähdejärjestelmissä olevista tiedoista. Yleensä tietoja poimitaan suoraan operatiivisten järjestelmien tietokannoista tai erillisistä siirtotiedostoista, jotka molemmat sisältävät tietoa yrityksen jokapäiväisestä toiminnasta. Yrityksen operatiiviset järjestelmät sisältävät tärkeitä liiketoimintaa kuvaavia tietoja, mutta usein tieto on hajallaan erilaisissa lähdejärjestelmissä. Usein operatiivisissa järjestelmissä olevia tietoja ei ole kuitenkaan kuvattu riittävästi tai niiden tietorakenteet ovat hankalia. Tiedot ovat yritykselle arvokas resurssi, joten tiedonhallinta ja tiedon hyödyntäminen antaa merkittävän kilpailuedun kiristyvässä kilpailussa. (Hovi ym. 2009, IV-XI).

4.1 Lähdejärjestelmät ja tietolähteet

Yritykset hyödyntävät sisäisesti kerättyä liiketoimintatietoa, jossa toimivat lähteenä yrityksen omat tietokannat ja operatiiviset järjestelmät. (Hovi ym. 2009, 18). Liiketoimintaan liittyviä tietoja ovat esimerkiksi henkilötiedot, taloustiedot, tieto prosesseista sekä asiakastieto. Yritykset pyrkivät hyödyntämään myös kilpailijoista ja markkinoista saatavaa tietoa. Tiedon lähteinä toimivat yrityksen ulkopuoliset tietokannat ja julkiset tietolähteet. Näitä ovat esimerkiksi Väestörekisterikeskus ja Liikenteen turvallisuusvirasto, jotka sisältävät väestö- ja ajoneuvotietoja. Internet sisältää paljon tilasto- ja viranomaistietoja, joten ulkoisia lähteitä hyödynnettäessä pitää arvioida tietolähteen luotettavuus ja tiedon oleellisuus. Liiallinen internet-lähteiden käyttö voi luoda yritykselle tietotulvan, josta on enemmän haittaa kuin hyötyä. (Hovi ym. 2001, 77).

4.1.1 Operatiiviset järjestelmät

Operatiivinen tietokanta on sovelluksen käyttöä varten suunniteltu tietokanta, johon tallennetaan liiketoimintaan liittyviä transaktiotapahtumia. Operatiivisten järjestelmien tehtävänä on automatisoida käyttäjien toimintoja. OLTP (Online Transaction Processing) on tapahtumankäsittelyjärjestelmä, joka pystyy käsittelemään isoja määriä reaaliaikaisia tapahtumia lyhyellä vasteajalla. Muita esimerkkejä operatiivisista

järjestelmistä on toiminnanohjauksen (ERP, Enterprise Resource Planning), asiakkuudenhallinnan (CRM, Customer Relationship Management) ja taloushallinnon järjestelmät (Financial Management Systems). (Hovi ym. 2009, 22).

4.1.2 Strukturoitu ja strukturoimaton tieto

Lähdejärjestelmissä oleva data voi olla strukturoidussa tai strukturoimattomassa muodossa. **Strukturoiduksi eli määrämuotoiseksi tiedoksi** kutsutaan tietoa, joka on kiinteän mittaista ja selkeästi määritelty. Tällaisia ovat esimerkiksi merkki-, numero- tai päivämäärämuotoiset tiedot, jotka ovat mitattavissa. Esimerkiksi asiakkaan ja tuotteen nimi sisältää merkkimuotoista dataa, kun taas päivämäärät ja tuotteen hinta sisältävät numeerista dataa. (Hovi ym. 2001, 49) Kaikkea tietoa ei saada operatiivisista järjestelmistä, joten datan tallentamiseen käytetään myös CSV-tiedostoja (Comma-separated values) tai muun tyyppisiä siirtotiedostoja. CSV on tiedostomuoto, jolla voi tallentaa taulukkomuotoista tietoa tekstitiedostoon (Shafranovich 2005, 1).

Strukturoimatonta tietoa ovat dokumentit, sähköpostit, valokuvat tai videot, jotka sisältävät multimediatyypistä tietoa. Tätä tietoa ei voi mitata, vaan niitä käsitellään yleensä omilla sovelluksilla. Strukturoimatonta tietoa säilytetään erillisissä tiedostoissa tietovaraston sijasta. Tulevaisuudessa tavoitteena on tallentaa multimediatiedostot tietovarastoon yhdessä strukturoidun datan kanssa. Näin yritys saa kattavamman näkemyksen asiakkaastaan, kun kaikki asiakasta koskevat dokumentit ovat käytettävissä yhdessä tietovarastossa. (Hovi ym. 2009, 18)

4.2 Lupapisteen lähdejärjestelmä

Lupapiste-sovellus tarjoaa raportointia varten hakemukset ja organisaatiot -HTTP JSON-rajapinnat, joista tietovarastoon poimittava data on aina luettavissa ja ajan tasalla. Lisäksi hinta- ja kuntatiedot poimitaan erillisistä CSV-tiedostoista. Rajapintojen avulla sovellukset voivat vaihtaa tietoja keskenään. JSON (JavaScript Object Notation) on yksinkertainen tiedonvälitykseen käytettävä avoimen standardin tiedostomuoto (Bray 2014, 2). Tunnistautumismenetelmänä käytetään HTTP basic autentikaatiota, joka on yksinkertainen tapa todentaa käyttäjä HTTP-palveluun. Käyttäjän pyynnön mukana palvelimelle välitetään käyttäjätunnus ja salasana (Franks 1999, 5).

4.2.1 Hakemukset

Kutsuttaessa hakemukset-rajapintaa voidaan määritellä palauttaako rajapinta kaikki hakemukset, jotka sovellukseen on tallennettu. Sovelluksen rajapinnan verkko-osoite voi myös palauttaa hakemukset, joiden muokkausaikaleima on suurempi tai yhtä suuri kuin annettu aikarajausparametri. Näin rajapinnan palauttamaa tulosjoukkoa voidaan rajata esimerkiksi seitsemän edellisen päivän aikana luotuihin tai muokattuihin hakemuksiin. Kuviossa 7 on hakemukset-rajapinnan JSON-dokumentti.

```

1
2 "ok": true,
3 "applications": [{
4   "sent": 1415969794229,
5   "authority": {
6     "role": "authority",
7     "lastName": "Mental",
8     "firstName": "Mike",
9     "username": "mental.mike@esimerkki.com",
10    "id": "1234567"
11  },
12  "submitted": 1415969794124,
13  "state": "verdictGiven",
14  "permitSubtype": null,
15  "convertedToApplication": null,
16  "organization": "753-R",
17  "started": null,
18  "closed": null,
19  "operations": [{
20    "displayNameSv": "Uppförande av bostadshus",
21    "displayNameFi": "Asuinrakennuksen rakentaminen",
22    "priceClass": "C",
23    "id": "5465fc01e4b0ce9e361d9453",
24    "name": "asuinrakennus",
25    "description": null,
26    "created": 1415969793310
27  }],
28  "infoRequest": false,
29  "openInfoRequest": false,
30  "opened": 1415969793310,
31  "applicant": "Erkki Esimerkki",
32  "created": 1415969793310,
33  "propertyId": "75312312341234",
34  "modified": 1415969795914,
35  "address": "Paatoskuja 12",
36  "permitType": "R",
37  "id": "LP-753-2014-00009",
38  "municipality": "753"
39 },

```

Kuvio 7. Hakemukset-rajapinnan palauttama JSON-dokumentin tietosisältö.

Kuviossa 7 esitetään JSON-dokumentissa olevat hakemuksen tiedot, jotka ovat tallentuneet Lupapiste verkkopalvelun tietokantaan. Tekstitiedosto palauttaa hakemuksen tietosisällön JSON-dokumentin muodossa. Tarkempi kuvaus hakemusten tietosisällöstä löytyy liitteestä 1.

4.2.2 Organisaatiot

Rajapinta palauttaa aina kaikki sovellukseen tallennetut organisaatiot. Sovelluksen verkkorajapintaa kutsumalla palautuu JSON-dokumentti, jonka tietosisältö ladataan tietovarastoon. Kuviossa 8 kuvataan organisaatiot-rajapinnan JSON-dokumentti. Tarkempi kuvaus organisaatioiden tietosisällöstä löytyy liitteestä 2.

```
17 {
18   "organizations":
19     "id": "837-R",
20     "name": {
21       "fi": "Tampereen rakennusvalvonta",
22       "sv": "Tammerfors byggnadstillsyn"
23     },
24     "scope": [{
25       "new-application-enabled": true,
26       "inforequest-enabled": true,
27       "permitType": "R",
28       "municipality": "837"
29     }]
30 }
```

Kuvio 8. Organisaatiot-rajapinnan palauttama JSON-dokumentin tietosisältö.

Tekstitiedosto sisältää objektin name, joka sisältää organisaation nimen suomeksi ja ruotsiksi. Toinen objekti on scope, joka sisältää konfiguraatitiedon siitä, minkä tyyppisiä hakemuksia ja minkä kunnan alueella organisaatio käsittelee hakemuksia. Esimerkiksi scope.permitType kenttä kuvaa hakemuksen lupatyyppin eli moduulin, joka kuvassa on R = Rakennusvalvonta. Scope.municipality kenttä on kolminumeroinen kuntatunnus, joka on yllä olevassa kuviossa Tampere. (Tilastokeskus 2014).

5 TIETOVARASTOINTI

Tietovarastoasiantuntija Ari Hovin (2001, 32) mukaan ”Tietovarastointi tarkoittaa joukkoa ajatusmalleja, suunnittelumenetelmiä, organisointia ja työkaluja, joiden avulla erillisistä operatiivisista tietokannoista saadaan tiedot yhdistettyä yhtenäiseksi, integroiduksi tietovarastoksi, jota voi käyttää monenlaisilla kysely- ja raporttivälineillä.”

Yrityksen sisällä syntyy paljon arvokasta tietoa, joka on yritykselle arvokas resurssi oikein hyödynnettynä. Operatiiviset järjestelmät eivät sovi hyvin tietojen analysointiin ja raportointiin, joten niitä varten on hyvä luoda tietovarasto (Hovi ym. 2001, 29). Tietovarastoon kootaan erilaisten operatiivisten järjestelmien tuottama tieto yhtenäiseen muotoon, jotta sitä voidaan hyödyntää tehokkaasti raportoinnissa ja analysoinnissa.

Tietovarastolle on useita eri määritelmiä. Yhden niistä on määritelty yhdysvaltalainen Bill Inmon, joka on kirjoittanut aiheesta useita kirjoja. Inmonia pidetään tietovarastokäsitteen keksijänä. (Hovi ym. 2009, 11). Tietovaraston toteutuksessa suositetaan tapaa, jossa esimerkiksi yrityksen yhdessä tai useammassa järjestelmässä olevat asiakastiedot yhdistetään yhteen keskitettyyn tietovarastoon (Data Warehouse Definition 2014). Inmonin (2010, 7) mukaan tietovarasto on aihe-suuntautunut, integroitu, aika-riippuvainen ja pysyvä kokoelma tietoja, jotka tukevat johdon päätöksentekoa. Edellä mainittuja ominaisuuksia voidaan kuvata seuraavasti:

- Tietovarastossa data organisoidaan aihealueen mukaan, ei yksittäisten tapahtumien perusteella
- Integroinnissa yhtenäistetään useiden eri lähteiden tiedot haluttuun muotoon tietovarastoon. Puhutaan yhdestä totuudesta, jolloin sama tieto esitetään vain yhdellä tavalla.
- Aika-riippuvuus tarkoittaa, että tiedoista jää aikaleima. Näin saadaan tieto, milloin tietovarastoa on esimerkiksi muokattu tai päivitetty
- Pysyvyys tarkoittaa, että tietovarastosta ei poisteta tietoja, vaan tiedot historioidaan uusien tietojen päivitysten yhteydessä. (Data Warehouse Definition 2014).

5.1 Tietovarastotyypit

Yrityksellä voi olla käytössä eri tietovarastotyyppisiä liiketoiminnan tarpeisiin. Tietovaraston tyyppistä riippuen se voi olla yrityksen kattava tietovarasto tai jaettu paikallisvarastoiksi. Seuraavassa esitellään yleisimmät tietovarastotyypit.

Keskitetty tietovarasto (EDW, Enterprise data warehouse) on koko yritystason tietovarasto, joka sisältää monen liiketoiminta-alueen tietoja. Tavoitteena on kerätä ja integroida yrityksen tiedot yhteen tai muutamaankin isoon tietokantaan. (Hovi ym. 2001, 67). Parhaimmillaan tiedot keskitetään EDW:n, johon ei tehdä kyselyjä. Kyselyt ja raportointi tehdään näistä johdetuista paikallisvarastoista ja kuutioista.

Paikallistietovarasto (Data Mart, Datamartti) on pienempi tietovarasto, joka on suunniteltu vain yrityksen tietyille osa-alueille tai käyttäjille, kuten talous- tai henkilöstöosastolle. Datamartin tehtävä on tarjota erilaisissa kyselyissä ja raportoinnissa käsiteltävää dataa helpommin ja nopeammin käyttäjän hyödynnettäväksi. Datamartti voi olla osa keskitettyä tietovarastoa tai toimia itsenäisenä kokonaisuutena, jolloin tiedot saadaan yhdestä lähdejärjestelmästä. (Hovi ym. 2009, 24)

Tietovarastointiprosessissa voidaan käyttää myös muitakin kuin pelkästään operatiivisia järjestelmiä tai itse tietovarastoa. Tällainen on **Staging area (välitietokanta)**, johon rajapinnoista kerättävät tiedot tallennetaan muokkausta varten. Tietovaraston ETL-prosessin latausvaiheen yhteydessä tiedot usein poimitaan operatiivisista järjestelmistä välitietokantaan ennen latausta varsinaiseen tietovarastoon. Tietojen rakenne pysyy samana ja varsinainen muokkaus tapahtuu ETL-prosessissa ennen tietovarastoon lataamista. Välitietokannan tehtävä on avustaa ETL-prosesseja ja siksi sen luonne on hyvin tekninen. (Hovi ym. 2009, 25). **Tilannekanta (Operational Data Store, ODS)** on myös operatiivisten tietokantojen ja tietovaraston väliin rakennettu erillinen välivarasto, joka yhdistää useamman operatiivisen järjestelmän tietoja. Tilannekanta päivittyy useammin kuin tietovarasto ja se sisältää vain ajan tasalla olevaa dataa. Tilannekanta ei ole varsinainen tietovarasto, vaan sitä käytetään operatiivisten järjestelmien tiedon yhdistämisessä, osana raportointia ja Master datan hallintaan. (Hovi ym. 2009, 25).

5.2 Tietovaraston mallinnus

Tietovaraston mallinnuksen tavoitteena on kuvata loogisella tasolla tietovaraston rakenne. Tietovaraston mallinnuksen kolme yleisintä menetelmää ovat tähti- ja lumihuutalemalli sekä Data vault. Mallinnuksessa taulut jaetaan kahteen eri tyyppiin, fakta- ja dimensiotauluihin.

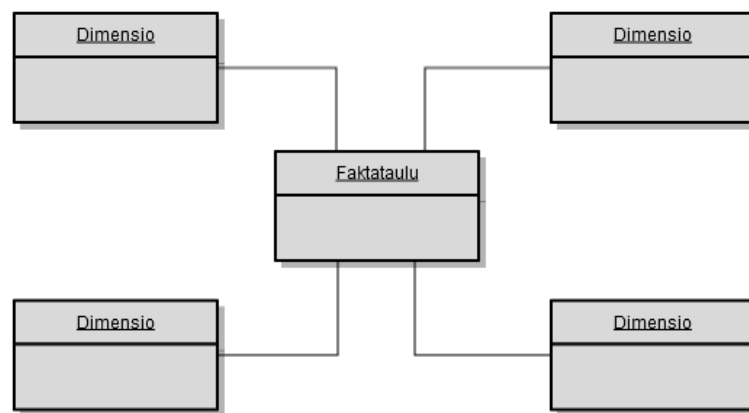
5.2.1 Fakta- ja dimensiotaulut

Moniulotteisessa mallinnuksessa **faktataulu** sisältää liiketoiminnan prosesseista syntyvää tapahtumatyyppistä tietoa, kuten myynti- tai tilaustapahtumia. Faktoiksi kutsutaan taulussa numeroarvoja sisältäviä sarakkeita. Faktataulun numeroarvoja voidaan summata eri dimensioista saaduilla tasoilla. Faktataulun pääavain (primary key) muodostuu dimensiotaulujen pääavaimista yhdisteltynä. (Hovi ym. 2005, 135). Faktataulusta löytyy yleensä kaksi tai useampi viiteavainta (foreign key), jotka ovat liitoksissa dimensiotaulujen pääavaimiin. Faktataulujen rivimäärä kasvaa tietokannassa rivien lisäysten yhteydessä, koska historiatiedot halutaan säilyttää eikä vanhoja rivejä päivitetä. Koska faktataulu kasvaa koko ajan, se vie moniulotteisissa tietokannoissa 90% kokonaistilasta. (Kimball & Margy 2013, 10-12).

Poikkeuksena faktatauluihin, **dimensiotaulujen eli ulottuvuustaulujen** tiedot pysyvät useimmiten melko muuttumattomina ja niistä saadaan erilaisia ryhmittelyjä ja hakukenttiä faktataulun rivejä summaaville kyselyille. Dimensiotaulut ovat selvästi pienempiä kuin faktataulut, mutta sarakkeiden ja rivien määrä niissä on yleensä isompi. (Hovi ym, 2005, 135). Dimensiotaulut tekevät yleensä tietovarastosta ymmärrettävän ja käytettävän, koska ne sisältävät faktatauluissa esiintyvien lukujen tekstimuotoiset kuvaukset. (Kimball & Margy 2013, 14)

5.2.2 Tähti- ja lumihiutalemalli

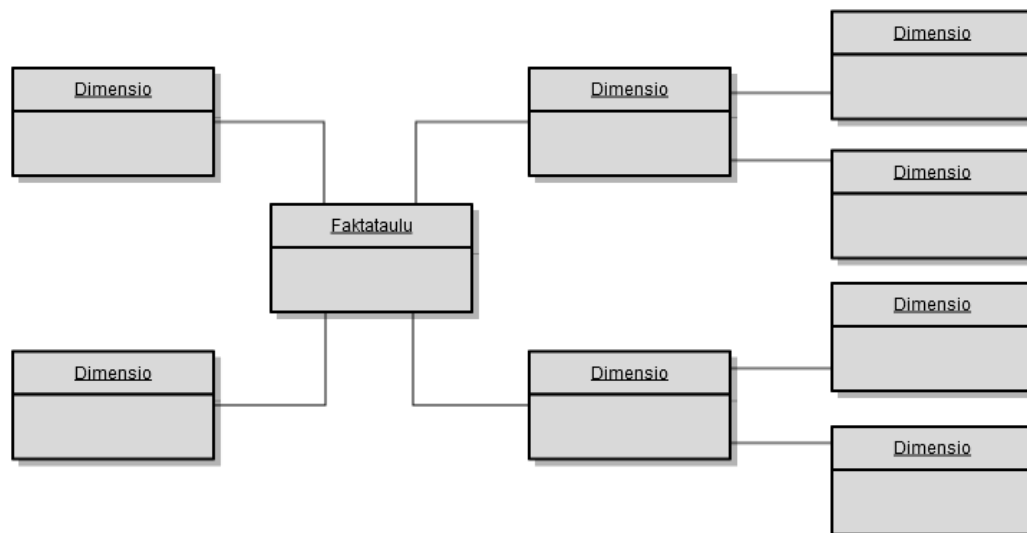
Tähtimalli (Star Schema) on tietovarastojen yhteydessä käytettävä moniulotteinen suunnittelumenetelmä, jonka arkkitehtuurin Ralph Kimball on kehittänyt (Hovi ym. 2009, 11). Tähtimalli koostuu keskellä olevasta faktataulusta ja sen ympärillä olevista dimensiotauluista. Tähtimalli muistuttaa nimensä mukaisesti tähteä sakaroineen ja se tukee hyvin moniulotteista ja numeerista tietoa. (Hovi ym. 2009, 36-37) Kuviossa 9 on esimerkki neljän dimensiotaulun tähtimallista.



Kuvio 9. Esimerkki tähtimallista.

Tähtimallin avulla voidaan jäljitellä moniulotteisuutta käyttäen kuitenkin perinteistä relaatiotietokantaa. Yhdeksi standardiksi muodostunut tähtimalli sopii hyvin tietovarastojen suunnittelumenetelmäksi, koska sen tavoite on tehdä kyselyistä ja raportoinnista mahdollisimman helppoa ja suorituskykyistä. (Hovi ym. 2009, 39). Faktataulut ovat yleensä normalisoituja, joten se ei sisällä tuplarivejä eli samaa tietoa kahteen kertaan. Dimensiotaulut puolestaan ovat denormalisoituja, jossa tietoa toistetaan. (Hovi ym. 2005, 135).

Kuviossa 10 dimensiotaulujen tieto voidaan myös normalisoida, jolloin tietoja ei toisteta ja tähtimalli muuttuu **lumihutalemalliksi (Snowflake Schema)**.



Kuvio 10. Esimerkki lumihutalemallista.

Tässä mallissa dimensiotaulut jakaantuvat useammaksi peräkkäiseksi dimensioksi. Vaikka tähtimallia pidetään yksinkertaisena ja paremman suorituskyvyn omaavana menetelmänä, BI-ratkaisussa asia ei ole yksiselitteinen. Osa BI-työkaluista on yhteensopiva tähtimallin ja toiset lumihutalemallin kanssa. (Hovi ym. 2009, 39-40).

5.2.3 Data vault

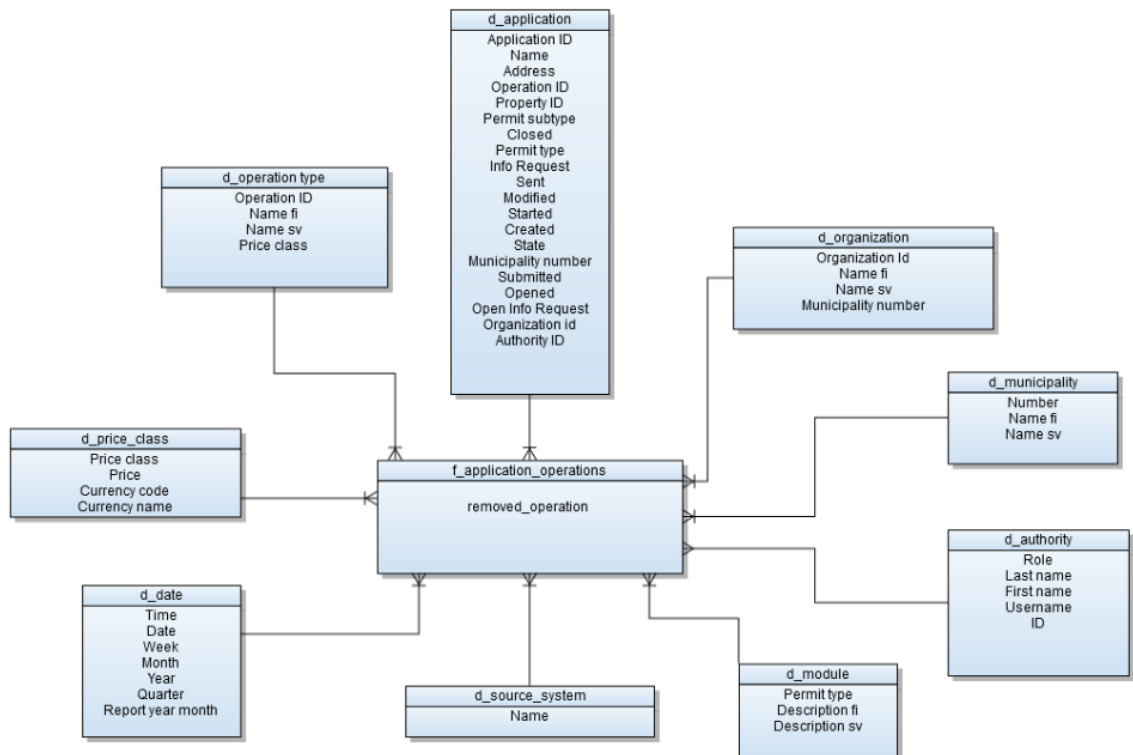
Data vault on 2000-luvun alussa kehitetty tietovarastointimalli, joka on joustava, skaalautuva ja johdonmukainen. Sitä kutsutaan hybridimalliksi, koska siinä yhdistetään kolmannen normaalin mallin ja tähtimallin parhaita osia. Data vault on mallinnusmenetelmä, joka on optimoitu tiedon integroinnin ja varastoinnin vaatimuksiin (Hultgren 2012, 20). Yritysten tarve hyödyntää tietoa muuttuu jatkuvasti ja tietovaraston täytyy olla valmis vastaamaan nykypäivän haasteisiin. Vanhoissa malleissa ongelmana on niiden jäykkyys ja hitaus muutoksille. (Linstedt 2013.)

5.3 Lupapistein tietovarasto

Projektissa on käytetty keskitettyä EDW:tä, johon historioidaan Lupapisteen raportointiin liittyvät tiedot. Tässä projektissa tiedot saadaan kahdesta lähdejärjestelmästä, joita ovat Lupapiste verkkosovellus ja CSV-tiedostot. Lähdejärjestelmistä tiedot viedään staging-alueen välitietokannan kautta keskitettyyn EDW:hen. Tietokannan mallinnustyö on tehty Model Right-mallinnustyökalulla, koska työkalu oli käytettävyydeltään hyvä, lisenssi oli edullinen ja aikaisemmat kokemukset puolsivat valintaa.

5.3.1 Tietokantamalli

Lupapisteen EDW-tietokantamallissa (kuvio 11) on yksi faktataulu `f_application_operations`, johon kaikki dimensiotaulut liittyvät. Tietovaraston taulujen nimeämisessä käytettiin yleistä menetelmää, missä faktataulujen nimen eteen tulee aina ”f” (esimerkiksi `f_application_operations`) ja dimensiotaulun nimen eteen ”d” (esimerkiksi `d_authority`).

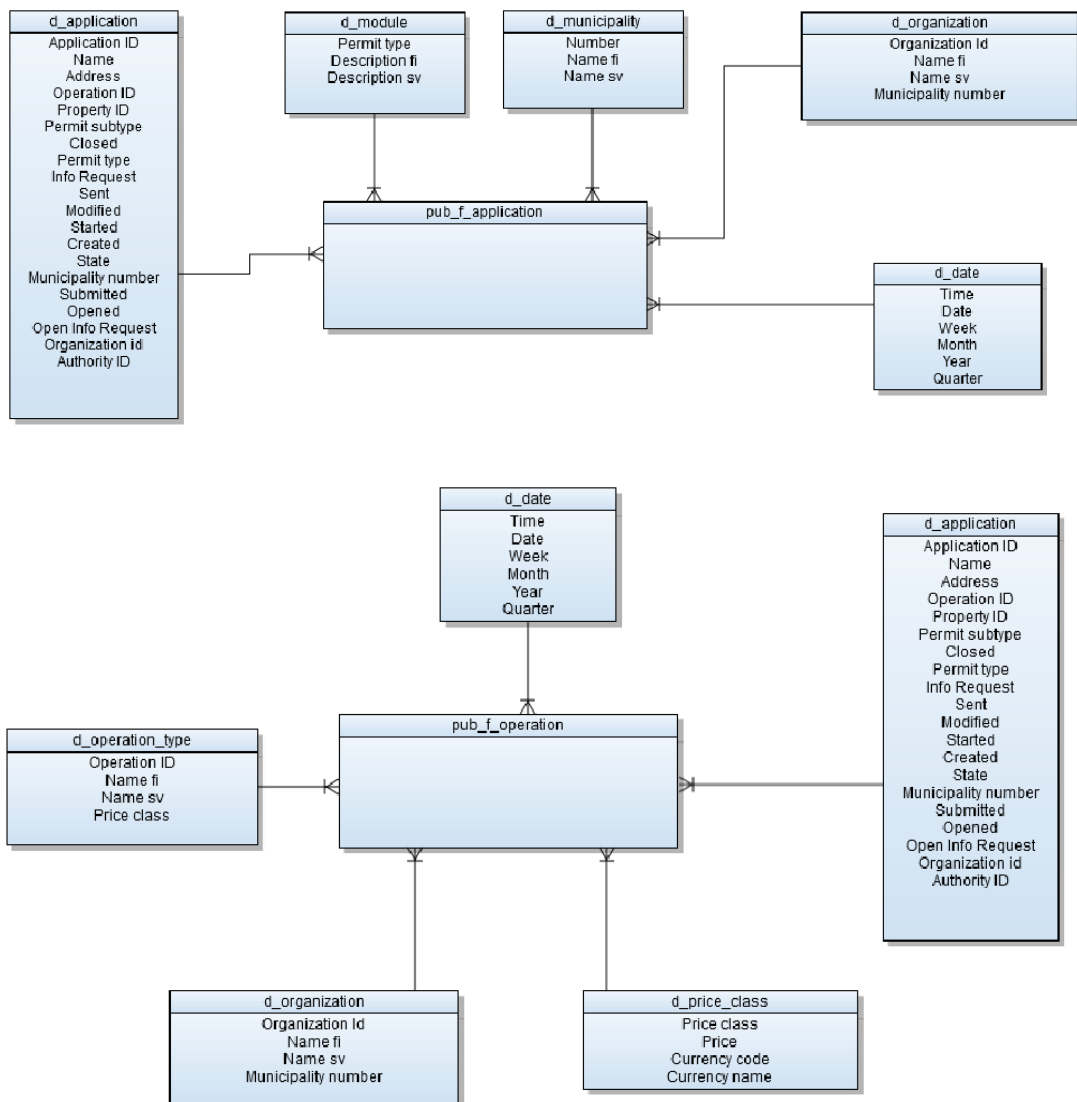


Kuvio 11. Lupapisteen tähtimallin mukainen EDW-tietokantakaavio

Tietovaraston mallinnuksessa on noudatettu tähtimallia ja toteutus on tehty PostgreSQL-relaatiotietokantaan. Lupapisteen osalta tähtimalliin päädyttiin sen helppokäyttöisyyden, nopeiden kyselyjen ja hallittavuuden takia.

5.3.2 Julkaisumalli

Raportointia varten on julkaistu erilliset faktataulut `pub_f_application` ja `pub_f_operation`, joilla on omat käyttötarkoituksensa, mutta niillä ei ole erillistä julkaisuskeemaa. Dimensiot ovat samoja kuin edellä mainitussa EDW-tähtimallissa. Raporttien kyselyt kohdistetaan näihin julkaisufaktoihin, jotka on kuvattu kuviossa 12.



Kuvio 12. Raportoinnissa käytettävät fakta- ja dimensiotaulut

Pub_f_application sisältää tarpeelliset tiedot neuvontapyyntöjen ja hakemusten määristä. Pub_f_operation koostuu toimenpiteiden oleellisista tiedoista. Julkaisufaktat kehitettiin helpottamaan raportointia yksinkertaistamalla niiden kyselyitä.

6 ETL-PROSESSI

Lyhenne ETL, eli extract, transform, load (poiminta, muokkaus ja lataus), on tietojen jalostusketju operatiivisista järjestelmistä tai muista järjestelmistä tietovarastoon. ETL-prosessin tehtävä on poimia dataa operatiivisista järjestelmistä tai siirtotiedostoista, muokata niitä ja lopuksi ladata data tietovarastoon. Tietomäärät ovat usein suuria, joten latausten ajankohdat ja kestot pyritään määrittelemään siten, etteivät ne kuormittaisi lähdejärjestelmiä niiden päivittäisen käytön aikaan. (Turban ym. 2011, 39) Muokausvaihe sisältää pienempiä osaprosesseja kuten virheiden tarkistusta, koodien muunnoksia ja tietojen historiointia. ETL-toteutus on tietokantapohjaista eräajosovelluksen ohjelmointia. Suurten datamassojen siirrossa ETL-prosessi suoritetaan eräajona yleensä kerran yössä. Tavoitteena on saada tiedot ladattua tietovarastoon, jotta päiväsaikaan tehtävät kyselyt BI-raportointityökaluilla olisivat nopeampia. (Hovi ym. 2009, 48)

6.1 ETL-prosessin menetelmät ja vaiheet

ETL-prosessissa käytetään työntö- ja vetomenetelmiä. **Työntömenetelmässä** halutut yksittäiset tiedot poimitaan operatiivisista järjestelmistä ja tallennetaan ne tiedostoihin. Työntöintegraatio siirtää tiedostot kohteen luettavaksi, jossa tiedot luetaan sisään, muokataan ja lopuksi kirjoitetaan tietovarastoon. (Hovi ym. 2009, 50)

Työntömenetelmässä on omat etunsa. Operatiivisen järjestelmän rakenteet ovat monimutkaisia, joten poiminnan vastuu on yleensä asiantuntijalla, kenelle rakenteet ovat tuttuja. Poiminnassa aloite tulee operatiivisista järjestelmistä, jolloin tietojen lukemiseen sopiva hetki voidaan määrittää tarkasti eikä tule päällekkäisyyttä esimerkiksi päivityseräajon aikana. Tiedostojen luomisen ansiosta esimerkiksi katkenneet latausajot voidaan uusia nopeammin kuin suoraan operatiivisen järjestelmän tietokannasta. Siirtotiedostossa olevat tiedot ovat pysyneet muuttumattomina, joten uusi lataus voidaan aloittaa tiedoista heti, kun virhe edellisessä ajossa on paikallistettu. Yritys ei ole riippuvainen operatiivisesta järjestelmästä, koska siirtotiedostot ovat tallessa eri ympäristössä. Operatiivista järjestelmää uusittaessa tarvitsee tilata

toimittajalta samanlaiset siirtotiedostot kuin korvattavassa järjestelmässä ja tietovaraston käyttö voi jatkua ilman keskeytyksiä. (Hovi ym. 2009, 50-51)

Työntömenetelmän ongelmaksi muodostuu usein tietojen integrointi. ETL-työkaluilla tietojen lukeminen tiedostoista on haastavampaa kuin suoraan operatiivisesta kannasta. Lisäksi tietokannan tietokuvauksien, kuten nimien ja tietotyyppien siirtäminen on helpompaa tietokantaan kuin tiedostoon. Työntömenetelmä sisältää myös enemmän vaiheita sekä mahdollisia kustannuksia kuin suoraan tietokannasta luettaessa. (Hovi ym. 2009, 51). Vastuu tiedon siirtämisestä on operatiivisella järjestelmällä. Ongelmatilanteissa tiedot saattavat jäädä siirtymättä, koska kaikissa ratkaisuisa ei varmisteta, että ajantasainen tieto saadaan siirrettyä tietovarastoon asti.

Vetomenetelmässä tiedot poimitaan suoraan operatiivisen järjestelmän tietokannan rajapinnasta tai siirtotiedostoista ETL-työkalulla. ETL-työkalu hakee ajastetussa latausajossa tiedot operatiivisesta tietokannasta, jolloin ohjelmointivaiheessa tietojen nimet ja kuvaukset saadaan siirrettyä välitietokantaan.

Vetomenetelmän etuja ovat yksinkertainen toteutus, joustavuus ja nopeus. Vetomenetelmällä on myös omat ongelmansa. Tietovarastoon siirtyy keskeneräisiä tietoja esimerkiksi kesken järjestelmäpäivityksen, jolloin uusintalatauksen suunnittelu ja toteutus on paljon hankalampaa kuin työntömenetelmässä. Pahimmassa tapauksessa tietoja ei saada ladattua enää tietovarastoon, koska ne ovat muuttuneet tai poistuneet kokonaan operatiivisesta tietokannasta. Jos yritys vaihtaa operatiivista järjestelmää tai sen sisäinen toteutus muuttuu, käytännössä koko ETL-prosessi on tehtävä uudestaan, mikä aiheuttaa lisätyötä sekä kuluja. (Hovi ym. 2009, 52)

6.1.1 Poiminta (Extract)

ETL-prosessin ensimmäistä vaihetta kutsutaan poimintavaiheeksi. Tietojen poimintaan käytetään usein samoja toimintoja kuin työntö- ja vetomenetelmissä. Poimintavaiheessa tiedot siirretään yhdestä tai useammasta lähdejärjestelmästä ETL-järjestelmän staging-alueen välitietokantaan. (Albrecht & Naumann 2008. 1). Poiminnassa tiedot luetaan suoraviivaisesti operatiivisten järjestelmien tietokannoista. Joskus tietojen muokkausta

saatetaan tehdä jo poimintavaiheessa, mutta yleensä tarkistukset ja tiedon yhtenäistäminen tehdään muokausvaiheessa. (Hovi ym. 2001, 80)

6.1.2 Muokkaus (Transform)

Tiedon muokausvaihetta voidaan pitää koko ETL-prosessin keskeisimpänä vaiheena. (Albrecht & Naumann 2008, 1) Poiminnan jälkeen lähdejärjestelmästä tiedot siirretään muokkausta varten välitietokantaan tietokannan edellyttämän rakenteen mukaisesti. Tätä siirtoa varten on olemassa erilaisia ETL-työkaluja, joiden avulla tiedon siirtäminen voidaan ajoittaa tai käynnistää automatisoidusti.

Siirron valmistuttua siirrytään muokausvaiheessa tiedon tarkastamiseen ja muokkaamiseen. Tarkastuksessa pyritään löytämään esimerkiksi tuplarivejä, jotka korjataan ja poistetaan. Tässä vaiheessa korjataan myös tyhjiksi jääneet ja tarkistetaan pakolliseksi määritetyt sarakkeet, sekä tehdään lukumäärätarkistus. Lukumäärätarkistuksella varmistetaan, että kaikki lähdejärjestelmästä siirretyt rivit siirtyvät välitietokantaan. (Hovi ym. 2009, 56-57)

Muokausvaiheen tehtävä ETL-prosessissa on muokata operatiivisesta järjestelmästä tullut tieto helppokäyttöiseen ja raportointia tukevaan muotoon. Käsiteltävän tiedon muokkaamiseen kuuluu esimerkiksi yhdistelmätietojen purkaminen, jossa sosiaaliturvatunnuksesta voidaan erotella syntymäpäivä, sukupuoli ja ikä omiin sarakkeisiin. (Hovi ym. 2009, 56-57).

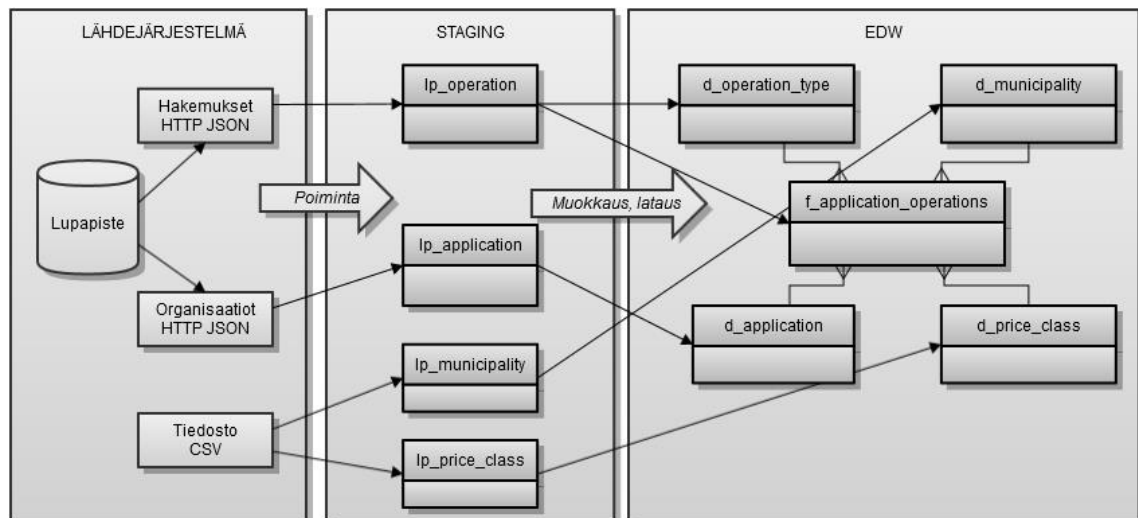
6.1.3 Lataus (Load)

ETL-prosessin latausvaiheessa käsiteltävä tieto ladataan tietovarastoon. Tietovarastoon ladataan tietoa, joka on validoitu ja muunnettu tietovaraston tietomallin rakenteen mukaiseksi. (Albrecht & Naumann 2008, 1). Tapahtuma- tai faktatyypiset rivit lisätään tietovarastossa yleensä suoraan vanhojen ja yhdenmukaisten tietojen perään. Muuttumattomien ja mitattavien taulujen tiedot korvataan uusilla tai ne voidaan historioida tietovarastoon. (Hovi ym. 2009, 58)

ETL-prosessin latausvaihe voidaan toteuttaa käytännössä useammalla tavalla. Latausvaihe voidaan suorittaa SQL-kielen INSERT-komennolla, joka sijoittaa tiedon tietovarastotietokannan riveille. Toinen vaihtoehto on hyödyntää tiedonsiirrossa käytettävän tietokantasovelluksen omia tehokkaita latausominaisuuksia. Yleensä latauksessa käytetään ETL-työkaluja, koska ne pystyvät kirjoittamaan suoraan tietokantaan tai hyödyntämään tietokantatuotteiden latausohjelmia. (Hovi ym. 2009, 58)

6.2 Lupapisteen ETL-Toteutus

Kolmivaiheisessa latausprosessissa (kuvio 13) Lupapisteen hakemukset- ja organisaatiot-rajapinnoista tiedot poimitaan vetomenetelmällä tietovaraston staging-välitietokannan tuontitauluihin. Samalla poimitaan tietoja myös csv-tiedostoista, jotka sisältävät esimerkiksi hintatiedot (price_class) ja kunnanumerot (municipality). Tuontitauluista tiedot muokataan yhtenäisiksi ja ladataan EDW skeeman tähtimalliin.

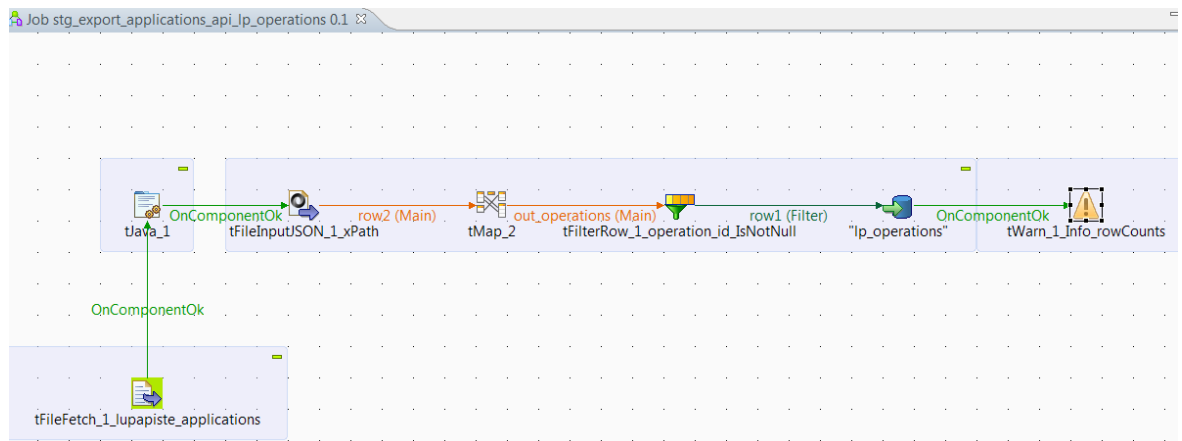


Kuvio 13. Lupapisteen ETL-prosessikaavio.

Työhön valittiin avoimen lähdekoodin ETL-työkalu Talend Open Studio. Tähän työkaluun päädyttiin, koska toteutuksessa ei haluttu investoida kalliisiin lisensseihin ja kyseessä olevasta työkalusta on aikaisemmin saatu hyviä kokemuksia. Talend Open Studio generoi suoritettavan java-koodin, joka tekee ETL muokkausoperaation muistinvaraisesti.

6.2.1 Poiminta (Extract)

Lupapisteen HTTP JSON-rajapinnosta JSON dokumentit ja CSV-tiedostot ladataan palvelimelle väliaikaisesti tallennetuiksi tiedostoiksi, jotka ladataan staging-alueen tuontitauluihin välitietokantaan. Kuviossa 14 on kuvattuna toimenpiteiden haku Lupapisteen hakemukset-rajapinnasta lp_operations tuontitauluun.

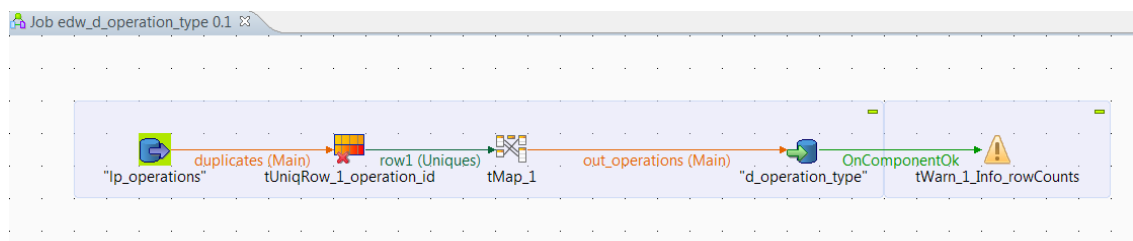


Kuvio 14. Rajapinnasta poiminta tuontitauluun lp_operations.

Komponentissa tFileFetch_1 haetaan Lupapisteen hakemukset-rajapinnasta JSON-dokumentti palvelimelle. tFileInput_1 lukee hakemuksesta ainoastaan toimenpiteen tiedot. Komponentissa tMap_2 yhdistetään luetut tiedot tuontitaulun kenttiin. Seuraavaksi ajoketjussa tFilterRow_1 jättää pois kaikki rivit, joiden toimenpide ID on tyhjä (null). Lopuksi tiedot lisätään lp_operations tuontitauluun. ETL-prosessin lokiin saadaan tieto lisätyistä riveistä tWarn_1_Info komponentin avulla seuranta varten.

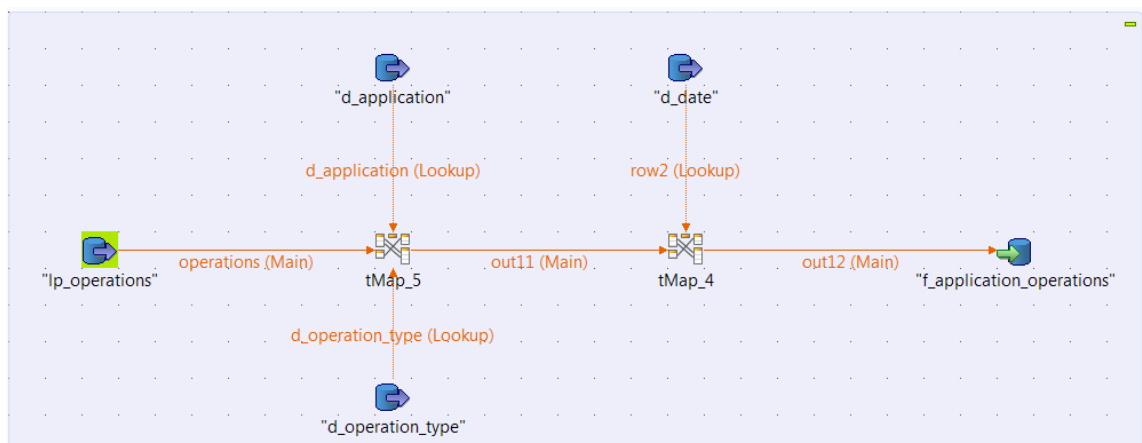
6.2.2 Muokkaus (Transform)

Muokkaus tapahtuu ETL-prosessissa muistinvaraisesti. Staging-alueen tuontitaulusta ladataan tiedot muistiin, josta tiedot prosessoidaan EDW-skeeman tauluihin. Ensin ladataan dimensiotaulut, jonka jälkeen ladataan faktataulut, koska faktataulut viittaavat dimensiotauluihin. Tällä tavoin tiedot muunnetaan yhdenmukaisiksi tähtimalliin. Kuviossa 15 on kuva dimensiolatauksesta.



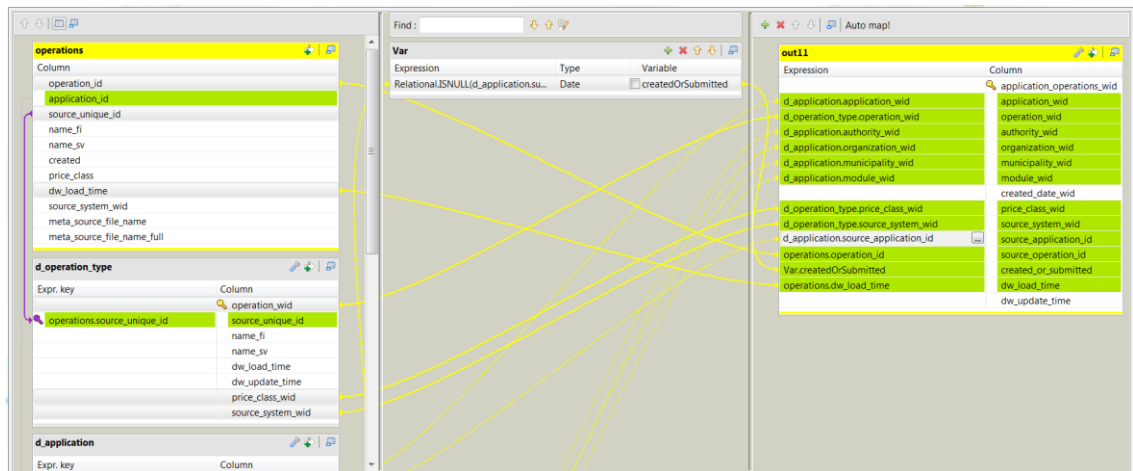
Kuvio 15. Dimensiolataus d_operations_type.

Dimensiolatauksessa d_operation_type haetaan kaikki operaatiot tuontitaulusta lp_operations. Dimensioon d_operation_type halutaan vain eri esiintymät toimenpiteistä. Esimerkiksi useassa eri hakemuksessa toistuu sama toimenpide (purkaminen), joten tUniqRow_1 komponentilla poistetaan näistä tuplarivit. Faktalatauksessa (kuvio 16) haetaan tuontitaulusta lp_operations kaikki toimenpiteet.



Kuvio 16. Faktalataus f_application_operations.

Komponentissa tMap_5 d_operation_type- ja d_application -dimensioiden tiedot liitetään operaatioihin. Tuontitaulu lp_operations sisältää hakemusten toimenpiteet, jotka liitetään hakemuksen tietoihin. Seuraavaksi aikadimensio d_date liitetään tMap_4 komponentissa hakemuksen luontiaikaleimaan. Tiedot lisätään tai päivitetään f_application_operations faktatauluun. Kuviossa 17 komponenttiin tMap_4 tehdään liitos, jossa haetaan dimensioviite.



Kuvio 17. Komponentti tMap_4:n sisältö

Komponentin tMap_4 sisällä hakemuksen operaatioihin liitetään operaation tyyppi (d_operation_type) operaation luonnollisella avaimella source_unique_id. Hakemuksen tiedot haetaan dimensiosta d_application id:lla (source_application_id).

6.2.3 Lataus (Load)

Latausvaiheessa dimensio- ja faktatauluun lisätään uudet rivit tai päivitetään olemassa olevia rivejä. Luonnollisena avaimena `d_operation_type` dimensiotaulussa on `source_unique_id`, joka sisältää toimenpiteen muuttumattoman teknisen nimen. Surrogaatti-pääavain on `operation_wid`. Kuviossa 18 on dimensiotaulun `d_operation_type` rivien lisäys tai päivitys.

The screenshot shows the configuration for the table `"d_operation_type"` (tPostgresqlOutput_1). The **Advanced settings** tab is selected, showing the following configuration:

Additional columns:

Name	SQL expression	Position	Reference column
operation_wid	nextval('edw.d_operation_seq'...)	Replace	operation_wid

Field options:

Column	<input type="checkbox"/> Update Key	<input type="checkbox"/> Deletion key	<input checked="" type="checkbox"/> Updatable	<input checked="" type="checkbox"/> Insertable
operation_wid	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
source_unique_id	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
name_fi	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
name_sv	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
dw_load_time	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
dw_update_time	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
price_class_wid	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
source_system_wid	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Other options: Enable debug mode, Support null in "SQL WHERE" statement, tStatCatcher Statistics

Kuvio 18. Dimensiotaulu `d_operation_typen` avaimet ja attribuutit

Kuviossa 18 luonnollinen avain `source_unique_id` on merkitty päivitysavaimeksi sarakkeessa "Update key". Luonnollinen avain ei ole päivitettävissä, eli sillä ei ole merkintää sarakkeessa "Updatable" kuten ei ole myöskään pääavaimella `operation_wid`. Nämä attribuutit saavat arvonsa uuden rivin luomisen yhteydessä. Surrogaatti pääavain `operation_wid` saadaan tietokannan sekvenssistä uutta riviä lisättäessä. Jos dimensiotaulusta löytyy jo ennestään luonnollisella avaimella vastaava rivi, päivitetään ne kentät, jotka on merkitty päivitettäväksi kohdassa "Updatable".

7 RAPORTOINTI

Raportointi ja analysointi on BI-arkkitehtuurin viimeinen vaihe, joka on loppukäyttäjän kannalta kaikkein oleellisin. BI-ratkaisujen avulla tieto saadaan loppukäyttäjille helppokäyttöisinä ja visuaalisina raportteina tai tunnuslukumittareina, joiden perusteella voidaan tehdä liiketoimintaa hyödyntäviä tärkeitä päätöksiä. Tietovarastosta kyselyjä tietoja analysoidaan ja raportoidaan BI-työkaluilla. Raportointia pidetään edelleen yhtenä keskeisimmistä BI-ratkaisujen tiedon hyödyntämismuodoista. Erityistä raportoinnille on reaaliaikaisuus, koska tiedon pitää olla luotettavaa ja entistä nopeammin saatavilla. (Hovi ym. 2009, 87) Raportointia toteutetaan ja julkaistaan yleensä selainpohjaisessa ympäristössä, jolloin raportit saadaan helposti laajaan jakeluun (Hovi ym. 2001, 117).

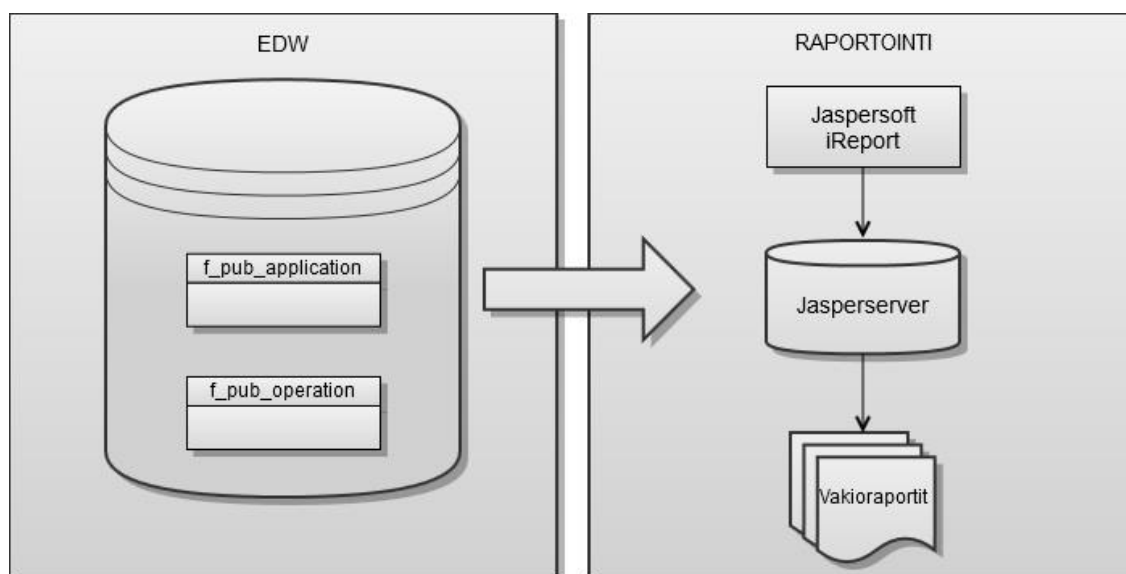
Kun raportteja tarvitaan säännöllisesti kuukausittain tai ne noudattavat vakiomuotoista rakennetta ja ulkoasua, puhutaan **vakioraporteista**. **Poikkeamaraporteiksi** kutsutaan raportteja, joille määritellään esimerkiksi kuukausimyyntiin tietyt rajat. Jos tietyn tuotteen myynti on alittanut kuukausirajan, saadaan tästä poikkeamasta valmis raportti päättäjille. **Parametroidun raportin** tarkoitus on rakentaa käyttäjille joukko vakiomuotoisia raportteja, jotka sisältävät runsaasti parametreja. Myös täsmäraportiksi kutsutussa parametroidussa raportissa käyttäjä määrittelee, mitä hakuehtoja halutaan käyttää raportin muodostamisessa. Parametriarvoja voivat olla esimerkiksi aika, myymälä tai tuote. **Ad hoc-kyselyksi** kutsutaan käyttäjän ennalta määrittelemättömiä kyselyjä ja raportteja. Ad hoc-kyselyjen teko vaatii käyttäjältä kyselytyökalun hallintaa, sekä tietämystä tietovaraston rakenteesta ja sisällöstä. (Hovi ym. 2001, 117-118)

Tietojen analysointiin käytetään monenlaisia menetelmiä. Aikaisemmin kuvatus raportoinnin lisäksi on tarjolla toinen teknologinen ratkaisu **OLAP (Online Analytical Processing)** eli moniulotteinen analysointi. OLAP on yleisin analysointityökalu tietovarastoratkaisuissa, joka tarjoaa moniulotteisen näkymän tietoon liiketaloudellista analysointia varten eli käyttäjä voi kerralla tarkkailla useampaa kuin yhtä dimensiota. OLAP-järjestelmän keskeisin osa kuvataan kuutiona, jossa summataan valmiiksi faktat jokaisilta dimensioiden tasoilta. (Hovi ym. 2009, 91). Erilaisten **mittaristojen** avulla

seurataan yrityksen tärkeimpiä tunnuslukuja, jolla saadaan realistinen kuva myynnin tämän hetkisistä luvuista. Yleisimpiä mittareita on esimerkiksi dashboardit ja scorecardit (Hovi ym. 2009, 95-96).

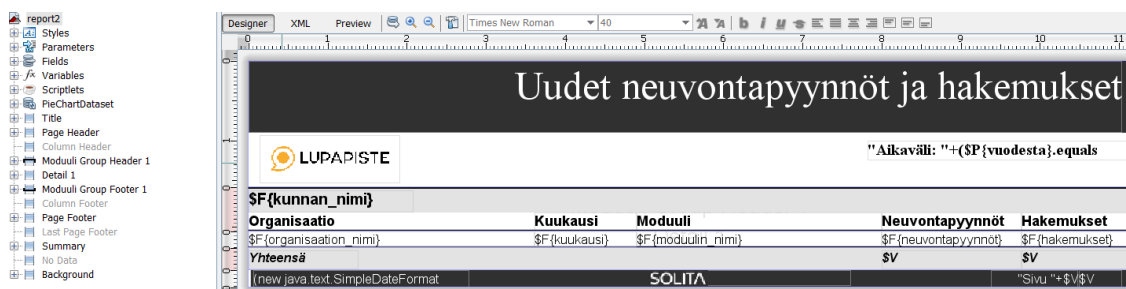
7.1 Lupapisteen Neuvontapyynnöt ja hakemukset -raportin muodostaminen

Luvussa kolme määritellyt raportointitarpeet on toteutettu mukailien vakio- ja parametriraportointia, jossa käyttäjän on mahdollista rajata raportin palauttamaa tietosisältöä parametrein. Vakioraportin asettelu ja esitettävät tietueet pysyvät samana, missä raportin tietosisältö vain muuttuu ajan- tai hakuparametrien suhteen. Valmiin raportin muodostamisessa on käytetty esimerkkinä uudet neuvontapyynnöt ja hakemukset –raporttia. Kuviossa 19 esitetään tietovaraston ja raportoinnin välinen suhde.



Kuvio 19. Lupapisteen raportoinnin prosessikaavio.

EDW:ssa on muodostettu julkaisufakta (f_pub_application), jossa tiedot on rajattu pelkkiin hakemuksiin, sekä (f_pub_operation), jossa on pelkästään toimenpiteisiin liittyvät tiedot. Jaspersoftin ilmaisella iReport-raportointityökalulla määritellään vakioraportin rakenne ja ulkoasu. XML-muotoinen raporttimäärittely tallennetaan JasperServer-palvelimelle, josta raportteja ajetaan. Projektiin valittiin JasperServerin ilmaisversio, koska lisensseihin ei haluttu investoida. Kuviossa 20 on esimerkki iReport-raportointityökalun käyttöliittymästä.



Kuvio 20. Uudet neuvontapyynnöt ja hakemukset -raportin rakenne.

Uudet neuvontapyynnöt ja hakemukset -raportti käyttää julkaisufakta f_pub_application taulua. Raportissa (kuvio 20) jaotellaan neuvontapyyntöjen ja hakemusten määrät kunnittain. Kuntajaottelun jälkeen neuvontapyyntöjen ja hakemusten määrät jaotellaan vielä organisaation, kuukauden ja moduulin mukaan. Kentät (\$F) määritellään raportin SQL-kyselyn SELECT-lausekkeessa. Neuvontapyyntöjen ja hakemusten summat kunnittain lasketaan muuttujien (\$V) kentissä. Aikaväli määritellään raporttia ajaessa käyttäjän syöttämistä arvoista, jotka välitetään parametreina (\$P) raportin SQL-kyselyyn.

7.2 Datan rajausta ja valmis raportti

Kuviossa 21 avautuu parametrinäyttö JasperReports palvelimen graafisessa käyttöliittymässä. Käyttöliittymässä käyttäjä voi asettaa raportin rajausehtoja ennen raportin ajamista. Käyttäjällä on mahdollisuus määrittää aikavälin vuodesta- ja vuoteen pudotusvalikoista. Moduulin valintalaatikosta käyttäjä voi valita hankkeen lupatyyppin. Organisaatio valintalaatikon alta käyttäjä voi valita kunnan ja viranomaisorganisaation, jonka neuvontapyyntöjä ja hakemuksia haluaa käsitellä. Moduulin ja organisaation valintalaatikoista voi valita yhden tai useamman valinnan, joiden tiedot käyttäjä haluaa nähdä.

Input Controls

* **vuodesta**
2014/01

* **vuoteen**
2014/10

Moduuli

Poikkeamisen hakeminen
 Rakentamisen lupa
 Yleisten alueiden lupa
All None Inverse

* **Organisaatio**

Hanko rakennusvalvonta
 Heinävesi rakennusvalvonta
 Helsinki - Rakennusvalvonta
All None Inverse

Apply OK Reset Cancel

Kuvio 21. Uudet neuvontapyynöt ja hakemukset–raportin datan rajausta.

Käyttäjä valitsee graafisen käyttöliittymän parametrilogista arvot, jolloin sovellus suorittaa tarvittavat kyselyt tietovarastoon ja valmis raportti näytetään käyttäjälle. Valmiissa vakioraportissa (kuva 22) käyttäjä on valinnut aikavälin 01/2014-10/2014. Moduuliksi eli lupatyypiksi käyttäjä on valinnut rakentamisen luvan. Organisaatio valintalaatikosta on valittu kunnaksi Helsinki ja Heinävesi, joiden viranomaisorganisaatio on rakennusvalvonta. Neuvontapyyntöjen ja hakemusten määrät näkyvät oikealla ja niiden yhteissumma alapuolella.

Uudet neuvontapyynnot ja hakemukset



Aikaväli: 2014/01 - 2014/10

Heinävesi				
Organisaatio	Kuukausi	Moduuli	Neuvontapyynnot	Hakemukset
Heinävesi rakennusvalvonta	2014/01	Rakentamisen lupa	0	4
Heinävesi rakennusvalvonta	2014/02	Rakentamisen lupa	0	1
Heinävesi rakennusvalvonta	2014/03	Rakentamisen lupa	0	2
Heinävesi rakennusvalvonta	2014/04	Rakentamisen lupa	0	3
Heinävesi rakennusvalvonta	2014/05	Rakentamisen lupa	0	4
Heinävesi rakennusvalvonta	2014/06	Rakentamisen lupa	1	2
Heinävesi rakennusvalvonta	2014/08	Rakentamisen lupa	0	6
Heinävesi rakennusvalvonta	2014/09	Rakentamisen lupa	0	3
Heinävesi rakennusvalvonta	2014/10	Rakentamisen lupa	0	1
Yhteensä			1	26
Helsinki				
Organisaatio	Kuukausi	Moduuli	Neuvontapyynnot	Hakemukset
Helsinki - Rakennusvalvonta	2014/03	Rakentamisen lupa	4	0
Helsinki - Rakennusvalvonta	2014/04	Rakentamisen lupa	1	0
Helsinki - Rakennusvalvonta	2014/05	Rakentamisen lupa	2	0
Helsinki - Rakennusvalvonta	2014/07	Rakentamisen lupa	1	0
Helsinki - Rakennusvalvonta	2014/08	Rakentamisen lupa	2	0
Helsinki - Rakennusvalvonta	2014/09	Rakentamisen lupa	3	0
Helsinki - Rakennusvalvonta	2014/10	Rakentamisen lupa	1	0
Yhteensä			14	0

tiistai 18. marraskuu 2014 12:52

SOLITA

Sivu 1/1

Kuvio 22. Ajettu raportti

Luvussa kolme määriteltiin raportointitarpeet, kuinka Lupapiste sovelluksen tietokantaan tallennetuista tiedoista halutaan hyötyä. Lupapisteen jalostettujen tietojen avulla voidaan seurata esimerkiksi neuvontapyyntöjen ja hakemusten määriä, minkä perusteella voidaan laskuttaa asiakasta. Raportointia voidaan käyttää myös asiakasorganisaation sisäiseen kehitykseen esimerkiksi mittaamalla hakemusten käsittelyaikojen pituuksia tai tutkimalla asiointien tiloja.

8 POHDINTA

Opinnäytetyön tavoite oli kuvata toteutus, kuinka Solita Oy:n sisäiseen käyttöön tulevaa raportointia kehitetään. Projektin tarkoitus oli toteuttaa Lupapiste sovellukselle tietovarasto, josta saadaan jalostettua valmiita vakioraportteja BI-työkalun avulla. Näiden valmiiden raporttien tietojen perusteella Solita Oy pystyy laskuttamaan asiakkaita, joita ovat erinäiset kunnalliset organisaatiot. Projektissa toteutettiin kolme vakioraporttia, joita olivat neuvontapyyntöjen ja hakemusten määrä, asiointien tilat ja kokonaismäärät. Opinnäytetyössä käytettiin tutkimusmenetelminä konstruktivistia- ja toiminnallista tutkimusotetta.

Lupapisteen hakemukset- ja organisaatiot-rajapinnoista sekä CSV-tiedostoista tiedot poimittiin kolmivaiheisessa ETL-latausprosessissa vetomenetelmällä staging-välitietokantaan muokkausta varten. Välitietokannassa tuontitaulujen tiedot muokattiin yhtenäisiksi ja ladattiin keskitettyyn EDW:hen, joka on toteutettu tähtimalliin. EDW koostuu kahdesta faktataulusta pub_f_application ja pub_f_operation, joihin on varastoitu Lupapisteen raportointiin liittyvät tiedot. Raportit ovat toteutettu mukaillen vakio- ja parametriraportointia, jossa käyttäjän on mahdollista rajata raportin palauttamaa tietosisältöä parametrein. Raportointipalvelin lähettää kyselyn tietovarastoon, joka palauttaa halutut tiedot ja generoi raporttipohjan mukaisen raportin käyttäjälle graafisessa käyttöliittymässä.

Projektin suurimpana haasteena oli uusien BI-työkalujen opettelu. Tietovaraston mallinnustyö on tehty Model Right-mallinnustyökalulla, ETL-työkaluna käytettiin avoimen lähdekoodin Talend Open Studiota ja raportointiin valittiin Jaspersoftin iReport-raportointityökalu. Käytettävät BI-työkalut valittiin sen mukaan, mitkä vastasivat parhaiten Lupapisteen tietovarasto- ja raportointiratkaisun tarpeita. Valintaperusteena olivat edulliset lisenssit ja ohjelmistojen käytettävyys. Solita Oy:ssa on toteutettu projekteja samoilla BI-työkaluilla, joten aikaisemmat positiiviset kokemukset puolsivat näiden työkalujen valintaa. Projektin toteutus edellä mainituilla työkaluilla osoittautui toteutuksen varrella tärkeäksi oman oppimisen ja kokemuksen laajentamisen kannalta.

Toteutetut vakioraportit ovat Solita Oy:n sisäisessä käytössä. Raportointia on tarkoitus tulevaisuudessa laajentaa kuntien ja ympäristöministeriön käyttöön. Kuntien viranomaiset näkisivät esimerkiksi tilastoja neuvontapyyntöjen ja hakemusten määristä. Ympäristöministeriöllä olisi mahdollisuus nähdä kaikkien kuntien tilastot. Jatkossa raportoinnin kehittämiskohteena voisi olla myös hakemuksen käsittelyajan keston tilastoiminen mittaamalla, kuinka kauan yhden hakemuksen käsittely vireillepanosta päätökseen kestää. Trendien seuraamisessa olisi mahdollisuus tarkastella esimerkiksi mihin kehityssuuntaan käsittelyajan pituus muuttuu eri kunnissa pitkällä aikavälillä. Tulevaisuudessa Lupapisteen sovelluksen tietovarastosta voitaisiin tuottaa Open dataa, jolloin EDW:ssa olevat tietoaineistot olisivat julkisia. Tästä voisi kiinnostua julkisen sektorin organisaatiot, kuten tilastokeskus, jota voisi kiinnostaa esimerkiksi eri ajanjaksoissa syntyneiden hakemusten määrät eri kunnissa.

Opinnäytetyön onnistumista voidaan pohtia kahdesta eri näkökulmasta. Ensimmäinen näkökulma on tutkia vastaako empiirinen osuus esitettyä teoriaa. Teoriassa määritelty BI-arkkitehtuuri on räätälöity Lupapisteen tarpeiden mukaan. Lupapisteen sovellus on teorianmukainen operatiivinen järjestelmä, joka käsittelee useita transaktioita. Teoriassa esitetyt ETL-prosessin menetelmät ja vaiheet on toteutettu Lupapisteen osalta kolmivaiheisena latausprosessina, jossa on käytetty vetomenetelmää. Tietovarastoinnin teoriassa tuotiin esille tietovarastotyyppjä ja -malleja, joista Lupapisteessä käytettiin keskitettyä EDW:tä, jonka rakenne oli tähtimalli. Projektin määrittelyn takia BI-ratkaisun menetelmänä käytettiin raportointia, joka on keskeisin tiedon hyödyntämismuoto nykypäivänäkin. Näin voidaan todeta, että projektin toteutus vastaa esitettyä teoriaa.

Toisesta näkökulmasta voidaan tutkia vastaako projekti asetettuja tavoitteita. Projekti oli onnistunut ja se toteutettiin sovitussa aikataulussa, sekä toteutus vastasi määriteltyjä raportointitarpeita. Projektin lopputuloksena käyttöön otettiin tietovarasto ja raportointiratkaisu, jotka täyttivät asetetut tavoitteet.

LÄHTEET

Albrecht, A. & Naumann, F. (2008). Managing ETL Processes [elektroninen aineisto]. Potsdam: University of Potsdam, Hasso-Plattner-Institute

Bray, T. (2014). The JavaScript Object Notation (JSON) Data Interchange Format.

Data Warehouse Definition.(2014).[viitattu 15.11.2014]. Saatavissa:

<http://www.1keydata.com/datawarehousing/data-warehouse-definition.html>

Franks, J., Hallam-Baker, P., Hostetler, J., Lawrence, S., Leach, P., Luotonen, A., & Stewart, L. (1999). HTTP authentication: Basic and digest access authentication.

Hovi, A., Hervonen, H., Koistinen, H. (2009). Tietovarastot ja business intelligence. Docendo. Porvoo.

Hovi, A., Huotari, J., Lähdenmäki, T.(2005). Tietokantojen suunnittelu ja indeksointi. Docendo. Porvoo.

Hovi, A., Ylinen, J., Koistinen, H.(2001). Tietovarastot liiketoiminnan tukena. Gummerus Kirjapaino Oy. Jyväskylä.

Hultgren, H. (2012). Modeling the Agile Data Warehouse with Data Vault. New Hamilton.

Inmon, W. H., Strauss, D., & Neushloss, G. (2010). *DW 2.0: The Architecture for the Next Generation of Data Warehousing: The Architecture for the Next Generation of Data Warehousing*. Morgan Kaufmann.

Kimball, R., & Ross, M. (2013). The data warehouse toolkit: The definitive guide to dimensional modeling. John Wiley & Sons.

Linstedt D. Data Vault Basics.(2013). [Viitattu 15.11.2014] Saatavissa:

<http://danlinstedt.com/about/data-vault-basic>

Shafranovich, Y. (2005). Common format and MIME type for Comma-Separated Values (CSV) files.

Solita Oy. 2014a. Lupapiste. [Viitattu 14.11.2014] Saatavissa:

<https://www.lupapiste.fi/>

Solita Oy. 2014b. Solita Oy. [Viitattu 6.11.2014] Saatavissa:

<http://www.solita.fi/>

Solita Oy. 2014. Solita Oy.[Viitattu 5.11.2014] Saatavissa:

<http://www.solita.fi/asiakkaat/case-lupapiste/>

Techtarget(2014). Business intelligence (BI). [Viitattu 8.11.2014] Saatavissa:

<http://searchdatamanagement.techtarget.com/definition/business-intelligence>

[Tilastokeskus.\(2014\). \[Viitattu 22.11.2014\] Saatavissa:
http://www.stat.fi/meta/luokitukset/kunta/001-2014/index.html](http://www.stat.fi/meta/luokitukset/kunta/001-2014/index.html)

Turban, E., Sharda, R., Delen, D., King, D.(2011). Business Intelligence – A managerial approach. Internal edition. Prentice Hall. United States

LIITTEET

Liite 1. Hakemukset-rajapinnan tietosisältö

1 (2)

TAULUKKO. Hakemukset-rajapinnan rakenne

Kenttä	Tietotyyppi	Kuvaus
id	String	Luonnollinen pääavain: asiointitunnus
address	String	Lähiosoite
applicant	String	Hakijan nimi denormalisoituna
authority	Object	Käsittelijä (viranomainen). Voi puuttua tai olla tyhjä, jos käsittelijää ei ole valittu vielä.
authority.id	String	Käyttäjän ID (viittaus surrogaatti-avaimeen)
authority.username	String	Käyttäjän käyttäjätunnus, tyypillisesti sama kuin sähköpostiosoite.
authority.firstName	String	Etunimi
authority.lastName	String	Sukunimi
authority.role	String	Käyttäjän rooli, tällä hetkellä aina "authority".
closed	Long	Hanke suljettu aikaleima
created	Long	Hanke luotu aikaleima
infoRequest	Boolean	Onko neuvontapyyntö? Neuvontapyynnöillä true, hakemuksilla false.
modified	String	Muokkausaikaleima
municipality	String	Kuntatunnus, minkä kunnan alueella hanke on. Kolmenumeroinen luku merkkijonona etunollilla.
opened	Long	Hanke siirtynyt valmisteille eli tullut kunnan työjonoon, aikaleima
openInfoRequest	Boolean	Neuvontapyyntö kunnassa joka ei ole ottanut Lupapistettä käyttöön
operations	Array	Toimenpiteet array, kustakin toimenpiteestä mm. nimi ja aikaleima

(jatkuu)

2 (2)

operations..id	String	Tämän toimenpide-instanssin surrokaattivain. Uniikki järjestelmässä.
operations..created	Long	Aikaleima, koska toimenpide on lisätty hakemukselle.
operations..displayNameFi	String	Suomenkielinen käyttöliittymässä näytettävä nimi
operations..displayNameSv	String	Ruotsinkielinen käyttöliittymässä näytettävä nimi
operations..name	String	Toimenpiteen tekninen avain. Operations-listassa voi olla useita toimenpiteitä joilla on sama name, ts. hakemus sisältää useita samoja toimenpiteitä, kuten rakennettaessa usea rakennus samalla hakemuksella.
operations..priceClass	String	Hintaluokka A – E
organization	String	Viranomaisorganisaation ID
propertyId	String	Kiinteistötunnus (tontin ID). 14 numeroa pitkä luku merkkijonona etunollilla.
permitSubtype	String	Hankkeen alityyppi, kun permitType=P: "poikkeamislupa" tai "suunnittelutarveratkaisu".
permitType	String	Hankkeen tyyppi-enumeraatio ("moduuli"). R, P, YA, YL, YI, MAL tai VVVL. R=rakennusvalvonta, P=poikkeusluvut jne.
sent	Long	Hanke siirretty kunnan taustajärjestelmään, aikaleima. (Lupapiste muodostanut siirtotiedoston, ei tiedossa koska siirtotiedosto oikeasti luetaan sisään kunnan järjestelmään)
started	Long	Rakentaminen aloitettu, aikaleima
state	String	Hankkeen tila
submitted	Long	Hanke jätetty vireille, aikaleima

Liite 2. Organisaatiot-rajapinnan rakenne

TAULUKKO. Organisaatiot-rajapinnan tietosisältö

Kenttä	Tietotyyppi	Kuvaus
id	String	Pääavain
name	Object	Sisältää organisaation nimen
name.fi	String	Nimi suomeksi
name.sv	String	Nimi ruotsiksi
scope	Object	Sisältää konfiguraatitiedon, minkä tyyppisiä hakemuksia ja minkä kunnan alueella organisaatio käsittelee.
scope.inforequest-enabled	Boolean	(Sovelluksen konfiguraatitietoa: neuvontapyyntöjen luonti käytössä?)
scope.municipality	String	Kuntatunnus, kolmenumeroinen luku merkkijonona etunollilla. Vrt. hakemuksen municipality.
scope.new-application-enabled	Boolean	(Sovelluksen konfiguraatitietoa: hakemusten luonti käytössä?)
scope.open-inforequest	Boolean	(Sovelluksen konfiguraatitietoa: ns. avointen neuvontapyyntöjen luonti luonti käytössä? Avoin neuvontapyyntö on annettu ilmaiseksi käyttöön kunnille jotka eivät ole varsinaisesti mukana Lupapisteessä.)
scope.open-inforequest-email	String	(Sovelluksen konfiguraatitietoa: sähköpostiosoite, johon lähetetään tieto uudesta avoimesta neuvontapyynnöstä.)
scope.permitType	String	Hakemuksen tyyppi eli moduuli. R, P, YA, YL, YI, MAL tai VVVL. Vrt. hakemuksen permitType

Liite 3. Lista tietovaraston tauluista

TAULUKKO. EDW-skeeman dimensio- ja faktataulut

Käsite	Taulu	Selite
Hakemus	d_application	Taulu sisältää kaikki hakemukset.
Käsittelijä	d_authority	Hakemusta käsittelevä viranomainen.
Aikadimensio	d_date	Tietovaraston tarvitsema dimensio, joka sisältää päivämäärät historiasta pitkälle tulevaisuuteen.
Moduuli	d_module	Hankkeen tyyppi-enumeratio ("moduuli"). R, P, YA, YL, YI, MAL tai VVVL. R=rakennusvalvonta, P=poikkeusluvut jne.
Kunta	d_municipality	Kaikki hakemuksia käsittelevät kunnat.
Toimenpiteen tyyppi	d_operation_type	Kaikki eri toimenpiteet, joita hakemuksella voi olla. Esimerkiksi asuinrakennus, purkaminen, laajentaminen.
Organisaatio	d_organization	Kunnan sisäinen organisaatio, esimerkiksi Rakennusvalvonta.
Organisaation moduuli	d_organization_module	Organisaation ja Moduulin muodostama monesta moneen suhde.
Hintaluokka	d_price_class	Hakemuksen ja operaatioiden hintatiedot, esimerkiksi hintaluokka, hinta, valuutta.
Lähdejärjestelmä	d_source_system	Tietovaraston tarvitsema dimensio, josta selviää mistä tieto on ladattu.
Hakemuksen toimenpiteet	f_application_operations	Faktataulu sisältää kaikki hakemuksiin sisältyvät toimenpiteet.
Julkaisufakta hakemuksista	pub_f_application	Raportoinnissa käytettävä faktataulu. Esimerkiksi halutaan tietää neuvontapyyntöjen tai hakemusten määrää.
Julkaisufakta toimenpiteistä	pub_f_operation	Raportoinnissa käytettävä faktataulu. Esimerkiksi halutaan tietää hakemukseen liittyvien toimenpiteiden määrää.