



LAUREA
AMMATTIKORKEAKOULU
Yhdessä enemmän

Talousraportoinnin toteutus SharePoint-ympäristössä
Laine, Markus

2014 Leppävaara



Laurea-ammattikorkeakoulu
Leppävaara

Talousraportoinnin toteutus SharePoint-ympäristössä

Markus Laine
Tietojenkäsittelyn koulutusohjelma
Opinnäytetyö
Joulukuu, 2014

Markus Laine

Talousraportoinnin toteutus SharePoint-ympäristössä

Vuosi	2014	Sivumäärä	35
-------	------	-----------	----

Tämän toiminnallisen opinnäytetyön tavoitteena on kuvata Energiaviraston talousraportointijärjestelmän toteutusta SharePoint-ympäristössä. Opinnäytetyössä kuvataan osiin jaetultuna raportointijärjestelmän eri osa-alueet esimerkkejä käyttäen. Jokaista järjestelmän erillistä osaa ei näytetä, sillä tämä johtaisi liialliseen määrään liitetiedostoja, sen sijaan esimerkeissä ja liitteissä keskitytään näyttämään järjestelmä suuremmassa mittakaavassa.

Opinnäytetyö toteutettiin luomalla Energiavirastolle talousraportointijärjestelmä ja kirjoittamalla siitä toteutuksen kuvaus. Ennen kuin työssä kuvattuun järjestelmään päädyttiin, harkittiin myös toista mahdollista toteutusvaihtoehtoa, joka sitten hylättiin. Tästä on myös lyhyt kuvaus.

Opinnäytetyön alku sisältää keskeiset käsitteet, työn taustan ja toimeksiantajan kuvauksen. Loppu sisältää hylättyjen vaihtoehtojen kuvaukset.

Markus Laine

Implementation of Budget Reporting in SharePoint-environment

Year	2014	Pages	35
------	------	-------	----

The purpose of this thesis is to describe the Energy Authority's financial reporting solution in SharePoint-environment. The thesis holds information describing all the sections of the reporting solution through examples, split in parts. Not every part of the reporting solution is described in detail due it resulting in excess amounts of appendixes. Instead, the examples and included appendixes focus on describing the solution on larger scale.

The project was carried out by creating a budget reporting solution for Energy Authority and writing this document about it. Before choosing the solution described here, one other solution was considered which was then abandoned. A short description of this is also included.

The thesis starts with important terms, background information and description of the client and ends with the descriptions of the abandoned concepts.

Keywords: Financial reporting, Energy Authority, SharePoint, SQL, SQL Server Report Builder

Sisällys

1	Johdanto	6
2	Toimeksiantaja ja toteutusympäristö	7
3	Tietoperusta	7
	3.1 Käytetty menetelmä	8
	3.2 Keskeiset termit	8
4	Järjestelmän osat	9
	4.1 Tietokanta	9
	4.2 Raportointi	12
	4.3 SharePoint	13
	4.4 Esitettävä raportti	13
5	Hylätyksi päätynyt ratkaisuvaihtoehto	17
6	Energiaviraston taloushallinnon arvio	17
7	Yhteenveto	19
	Lähteet	20
	Kuvat	21
	Liitteet	22

1 Johdanto

Budjetissa pysyminen on epäilemättä yksi tärkeimmistä tavoitteista julkisella sekä yksityisellä sektorilla. Tämän avuksi on usein erilaisia dokumentteja ja esityksiä, jotka linjaavat budjetin ja sen käyttötarkoituksen. Näistä pitäisi käydä ilmi, kuka on vastuussa mistäkin budjetin osaluueesta ja kuinka paljon rahaa näihin on varattu. Toisin sanoen budjetin valmistelun ja toteutumisen välinen suhde pitää olla läheinen. Ongelmana on, että vaikka tämä tieto on saatavilla taloushallinnolle, niin organisaation muille jäsenille nämä tiedot eivät välttämättä ole yhtä helposti saatavilla.

Tässä opinnäytetyössä kuvatus järjestelmän toimenantona oli auttaa organisaation taloustietojen saantia koskevan ongelman ratkaisussa. Esittämällä taloustiedot yleisesti näkyvillä olevalla alueella, jossa kaikki organisaation jäsenet näkevät ne, pyrittiin helpottamaan heidän tiedonsaantiaan. Lisäksi haluttiin antaa taloushallinnolle mahdollisuus hoitaa mahdollisimman suurissa määrin järjestelmän tietojen ylläpito.

Opinnäytetyön lähtökohtana oli talousraportoinnin toteutus SharePoint-ympäristössä. Opinnäytetyö kuvaa talousraportoinnin toteutuksen tietokannasta SharePointissa julkaistavaan raporttiin. Tämän talousraportin tarkoituksena on antaa Energiaviraston työntekijöille ja etenkin esimiehille helppo tapa seurata budjetin käyttöä ja arvioida oman ryhmänsä kuluja ja sitä kuinka paljon heillä on rahaa jäljellä käytettäväksi. Opinnäytetyö kuvaa samalla talousraportointijärjestelmän toimintaa.

Opinnäytetyössä ei selitetä termejä tarkemmin kuin mitä on kuvattu keskeiset termit - osiossa. SQL- tietokantojen, SharePointin tai Windows-pohjaisen tietojärjestelmän toimintoja tai termejä ei selitetä, elleivät ne liity suoraan raportointijärjestelmän toimintaan. Nämä jätetään pois siksi, että ne vain turhaan kasvattaisivat dokumentin pituutta.

Työssä käytetyissä kuvista sekä liitteistä on poistettu joitain tietoja, jotka saattaisivat vaarantaa järjestelmän tietoturvan. Lisäksi kuvissa esiintyviä mahdollisesti keskeneräisiä budjetitietoja on poistettu. Mikäli näitä halutaan tarkistella, ne ovat julkisesti näkyvillä niiden vahvistamisen jälkeen, kun budjetista tulee julkinen.

Talousraportointi toteutettiin yhteistyössä Peaches Industries -yrityksen kanssa muun Energiaviraston intraverkon kehitystyön yhteydessä.

2 Toimeksiantaja ja toteutusympäristö

Energiavirasto edistää ja valvoo energiamarkkinoita, päästöjen vähentämistä, energiatehokkuutta sekä uusiutuvan energian käyttöä. Työ- ja elinkeinoministeriön hallinnonalaan kuuluva Energiavirasto huolehtii yhdentyneiden sähkö- ja kaasumarkkinoiden toimivuudesta sekä verkkopalveluiden hinnoittelun kohtuullisuudesta. Toimiva päästökauppa, uusiutuvan energian käyttö ja energiatehokkuus edistävät kustannustehokkaasti ilmastotavoitteiden saavuttamista.

Energiavirastossa on töissä kirjoitushetkellä 72 työntekijää jaettuna seitsemään ryhmään. Nämä ryhmät ovat johto, hallinto, markkinat, uusiutuva energia, verkot, energiatehokkuus ja päästökauppa.

Talousraportointi toteutettiin Energiaviraston työ- ja tietoliikennenympäristössä, johon tästä lähtien viitataan vain Energiaviraston ympäristönä. Tietokanta varastoidaan SQL Server tietokannassa. Raportointiosio talousraportoinnista esitetään Energiaviraston intraverkossa, joka on toteutettu käyttäen Microsoftin SharePoint 2013 ohjelmistoa.

3 Tietoperusta

Talousraportointijärjestelmän rakennusprojekti toteutettiin käyttäen soveltaen Scrum-viitekehystä. Scrum-viitekehyksessä on kolme roolia: ScrumMaster on vastuussa siitä, että prosessia ymmärretään ja noudatetaan, Product Owner vastaa tiimin jäsenten työn arvon maksimoinnista sekä Team, joka tekee varsinaisen työn. Toinen suuri elementti SCRUM-viitekehystä on säännöllisesti pidetyt kokoukset, joista SCRUMin sydämessä on Sprint, joka on kerran kuukaudessa tai useammin pidettävä kokous, jossa tiimi ja osakkaat käyvät läpi mitä on tehty. (Schwaber, Beedle 2002)

Toinen osa Scrumia on päivittäin pidettävä lyhyt ei-muodollinen noin 15 minuuttia kestävä kokous nimeltään päivittäinen Scrum (Daily Scrum). Tässä kokouksessa jokainen tiimin jäsen kertoo lyhyesti mitä on tehnyt viimeksi, mitä aikoo tehdä seuraavaksi ja mitä esteitä hän kohtaa tai on kohdannut. (Schwaber, Beedle 2002)

Scrumin tavoitteena on rakentaa jokaisen Sprintin aikana toiminnollisuutta lisää projektiin. Tämän toiminnollisuuden pitäisi olla mahdollisesti julkaisukelpoista, koska Product Owner voi päättää ottaa toiminnollisuuden käyttöön. (Schwaber, Beedle 2002)

Muista Scrumin käsitteitä ovat Sprint Backlog, joka tarkoittaa asioita jotka syystä tai toisesta ovat jääneet kesken. Sprint Backlog Burndown on kaavio, joka arvioi aikaa, joka menisi kaik-

kien Sprint Backlog asioiden loppuun saattoon. On Scrumille luonnollista, että Sprint Backlog -lista elää Sprintin aikana ja suositellaan, että tämä lista pidetään selkeästi esillä muistuttamassa sen tilasta. (Schwaber, Beedle 2002)

3.1 Käytetty menetelmä

Päädymme käyttämään Scrumia soveltaen johtuen tiimin pienestä kahden henkilön koosta sekä siitä, että kooltaan projekti oli melko pieni, sillä se sisälsi vain yhden raportin alaraportteineen sekä tukijärjestelmineen. Virallisesti tiimillä ei ollut ScrumMasteria. Product Ownerina toimi Energiaviraston taloushallinnon edustaja, joka oli myös projektin toimeksiantaja.

Emme pitäneet päivittäisiä kokouksia normaalin Scrum tavan mukaan. Sen sijaan pidimme Daily Scrum- kokouksia vastaavia hieman pidempiä kokouksia johtuen työtehtävien aiheuttamista aikatauluvaatimuksista noin kerran viikossa, joskus useammin. Sprint kokouksia pidimme noin kerran kuukaudessa, näihin kokouksiin osallistui tiimin lisäksi Product Owner sekä joskus harvemmin muitakin Energiaviraston työntekijöitä, joilla oli jotain osakkuutta projektiin.

3.2 Keskeiset termit

SharePoint 2013 on Microsoftin uusin isompi versio SharePointista. SharePoint mahdollistaa dokumenttien, kontaktien ja tiedon helpon jaon käyttäjien kesken. SharePointin tarkoituksena on keskittää data yhteen sijaintiin, pois erinäisistä tietokannoista sekä muista tiedostojasijainneista. SharePoint on tuotteena hyvin suuri, sen eri osista jokaisesta voisi kirjoittaa oman kirjansa. (Shafiq 2014, 1 - 4)

SQL Server on Microsoftin lippulaivatietokantatuote. SQL Server 2012 sisältää sen ytimessä olevan tietokantamoottorin, joka mahdollistaa suurten tietomäärien varastoinnin ja noudon relationaalisesti. Lisäksi SQL Server 2012 pitää sisällään Query Optimizerin, joka optimoi kyselyitä valitsemalla optimoidumman tavan hakea tietoa. SQL Server -tietokanta on kokoelma objekteja, jotka pitävät sisällään ja käsittelevät dataa. Yleensä yksi SQL Server -instanssi pitää sisällään vain kourallisen tietokantoja, mutta useampien tietokantojen instanssit eivät ole ennenkuulumattomia. Joissakin erikoistapauksissa kuten SQL Server -järjestelmätietokannassa ja tietokantashototeissa tietokannat eivät ole objektipohjaisia. (Delaney 2013)

SQL Server Reporting Services 2012 (SSRS) on uusin versio, joka on osa SQL Server 2012:sta. SSRS tarjoaa kokoelman työkaluja raporttien luomiseen, käyttöönottoon ja hallintaan. Versiossa 2012 tuli useita uusia ominaisuuksia, joiden kaikkien listaaminen veisi useita sivuja. Suurimmista muutoksista tai lisäyksistä ensimmäinen oli siirtyminen Visual Studio Business In-

terlligencestä (BI) SQL Server Data Toolsiin (SSDT), mikä tarkoitti sitä, että erillistä lisenssiä Visual Studiolle ei enää tarvita. Toinen tärkeä muutos uudessa versiossa oli ad-hoc- raportoinnin parantaminen yhdistämällä Business Interlligence Semantic Model (BISM) SharePoint Server:iin auttamaan Data Alerttien ja Power Viewien esittämisessä. (Krishnaswamy, 2014, 1 - 13)

4 Järjestelmän osat

Toteutetussa raportointisovelluksessa on kolme helposti toisistaan eroteltavaa osaa: tietokanta, raportointi ja SharePoint. Lyhyesti: tietokanta pitää sisällään raportin sisältämän datan, raportointi käsittelee sen haluttuun muotoon ja data esitetään SharePointissa.

4.1 Tietokanta

Raakadata toimitetaan csv- tiedostona Palkeet- pääkirjanpidosta Energiaviraston sftp-palvelimelle. Täältä se siirretään tietokantapalvelimelle automaattisesti skriptillä, joka on ajastettu käyttäen Windowsin Task Scheduleria. Tämä skripti myös muokkaa raakadatan muotoon, joka on helpommin luettavissa tietokantaan. Tietokanta lukee tämän datan myöhemmin ajastetulla tehtävällä. Kuva 1 sisältää tiedonsiirrosta vastaavan skriptin. Tietokanta lukee tämän datan käyttäen proseduuria Read_TAP-File_IntoDB, katso kuva 3 proseduurien listausta varten.

Tietokantaa rakentaessa ei seurattu yleisiä parhaita käytäntöjä koskien tietovarantoja (Data Warehousing) ja suuren mittakaavan relationaalisia tietovarantoja (Large Scale Relational Data Warehouse). Joitain parhaita käytäntöjä koskien tietovarantoja on suurien taulujen osiointi, vanhan datan tehokas poisto sekä varmuuskopioinnin tehostaminen esimerkiksi käyttämällä SAN- teknologiaa. (Top 10 Best Practices for Building a Large Scale Relational Data Warehouse 2013)

Nämä käytännöt eivät olleet tarpeen, koska vaikka taloustietokantaan näyttää menevän paljon dataa, se ei ole tietokantojen mittakaavassa kovinkaan paljon. Tauluja ei ole kuin yhdeksän eikä niissäkään ole paljoa kenttiä. Koska tietomäärä ei ole niin suuri, niin ei ole tarpeen ottaa käyttöön edellä mainittuja toimenpiteitä tai vastaavia, sillä raportin käyttämän tietokannan koko tai rasitus tietokantapalvelimelle ei ole niin suurta.

```

set-strictmode -version "2.0"

function SevenToEight {
    param($c)
    # 7/8 bittisen ASCII:n konversio. Vaihetaan ketjutetulla Replacella ohjausmerkit ääkköksi. Paljon nopeampaa kuin konvertoida char[] -muotoon.
    return $c.Replace('[', 'Ä').Replace('\', 'Ö').Replace(']', 'Å').Replace('^', 'Ü').Replace('{', 'ß').Replace('|', 'ø').Replace('}', 'å').Replace('~', 'ü')
}

new-variable -option readonly -name LOGFILE -value "POISTETTU.log"

add-content $LOGFILE "--- START ---"
add-content $LOGFILE (Get-Date).ToString("u")

# Raintancen lähdetiedosto SFTP-palvelimella
$sourceShare = "POISTETTU"
net use $sourceShare "POISTETTU"

$sourceFile = "POISTETTU"
$destinationFile = "POISTETTU"

# Jos ei ole käytettävissä, ei tehdä mitään
if(-not (test-path $sourceFile) ) {
    write-host "$sourceFile ei ole käsiteltävissä."
    add-content $LOGFILE "$sourceFile ei ole käsiteltävissä."
    net use $sourceShare /delete
    return
}

# Luetaan otsakerivi erikseen, koska se ei ole yhtenevä datasisällön kanssa
$sr = new-object IO.StreamReader($sourceFile)
$headerRow = $sr.ReadLine()
$headerRow = SevenToEight $headerRow

$sb = new-object Text.StringBuilder
[void]$sb.AppendLine($headerRow)
$rowCount = 0
while( ($line = $sr.ReadLine()) -ne $null ) { |
    [string]$data = SevenToEight $line

    # Pilkotaan string taulukoksi puolipisteen kohdalta. Jos on 18 solua, kaikki OK
    # Muuten selite -kentässä on mukana puolipiste(itä)
    $arr = $data.Split(";")

    if($arr.Count -lt 18) {
        # Oops, ei tarpeeksi sarakkeita?!
        write-host "Rivillä oli vain $arr.count saraketta. Pitäisi olla ainakin 18, ohitetaan rivi."
        add-content $LOGFILE "Rivillä oli vain $arr.count saraketta. Pitäisi olla ainakin 18, ohitetaan rivi."
    } else {
        # Jos on vähintään 18 saraketta, käsitellään vain ne.
        # "ylimääräiset", eli sarakkeet joiden datassa on puolipiste,
        # jätetään käsittelemättä
        for($i = 0; $i -lt 18; $i++) {
            $arr[$i] = $arr[$i].Trim()
        }

        # Lopuksi trimataan lopputulos, jotta saadaan pois viimeisessä
        # solussa mahdollisesti oleva whitespace
        $data = ($arr -join ";").Trim()
        [void]$sb.AppendLine($data)
    }
    ++$rowCount
}

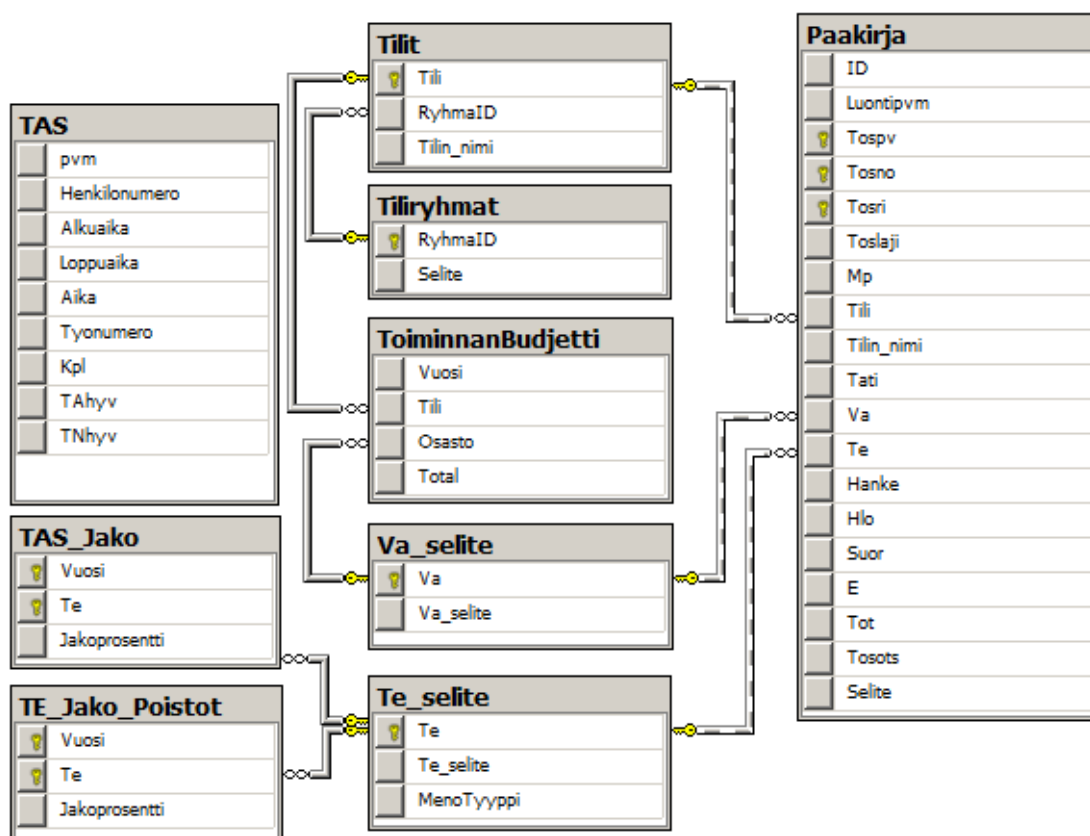
# Suljetaan lähde
$sr.Close()
$sr.Dispose()
set-content -encoding unicode -path $destinationFile -value $sb.ToString()

net use $sourceShare /delete
add-content $LOGFILE "$rowCount riviä käsitelty."
add-content $LOGFILE (Get-Date).ToString("u")
add-content $LOGFILE "--- STOP ---"

```

Kuva 1: Skripti tiedostojen siirtoon sftp-palvelimelta

Tietokannan raportoinnin käyttämiä tietokantayhteyksiä (Data Sources) muodostaessa käytetään paikallista tietokantakäyttäjätunnusta. Tämä tunnus on luotu raportin käyttämän tietokannan instanssiin. Käyttäjälle on määriteltävy CRUD-oikeudet (tässä tapauksessa: Connect, Read, Update, Delete) kyseiseen tietokantaan.



Kuva 2: Tietokannan taulujen riippuvuudet

TAS- ja TAS_Jako -taulut pitävät sisällään kerran vuodessa manuaalisesti syötettäviä tietoja työajanseurannasta. Paakirja-taulu pitää sisällään päivittäin Raindancesta aikaisemmin mainitut tiedot. Tilit-taulu pitää sisällään raportoinnissa käytettävät tilit. Tämä on tärkeä taulu, sillä jos tili ei mukana tässä taulussa, se ei myöskään ole mukana SharePointin puolella olevassa raportissa. Tiliryhmät-taulu sisältää raportoinnissa käytettävien tilien ryhmät, joihin jokainen tili kuuluu. Va_selite- ja Te_selite -taulut kertovat Va- ja Te-tiedon selitteen. Nämä taulut ovat tarpeellisia siksi, että Raindancesta saapuvassa tiedostossa tulee vain integer-tyyppin numero ilman selityksiä. TE_Jako_Poistot syötetään kerran vuodessa manuaalisesti tähän tauluun. ToiminnanBudjetti-taulu pitää sisällään vuositason budjettitiedot, jotka syötetään yleensä kerran vuodessa. Poikkeuksena tähän on se, kun otetaan käyttöön uusi TA- tai LKP-tili, mikä vaatii isompia töitä tietokannan puolella.

Tietokantaa rakentaessa pyrittiin pitämään se mahdollisimman siistinä vähentäen turhan tiedon määrää. Tähän pyrittiin käyttämällä suodatusta, jotka tulivat kahdesta eri lähteestä: kuvassa 1 esitellystä skriptistä sekä proseduurilla, joka lukee datan tietokantaan. Joitain uusia tauluja ja tietokenttiä oli kuitenkin tarpeen luoda järjestelemään dataa muotoon, jota raportti voi käyttää. Näitä luotuja tauluja ovat Tiliryhmat, Va_selite sekä Te_selite. Tässä työs-

kenneltiin yhdessä taloushallinnon kanssa, jotta ne pitävät oikeat tilit sisällään ja tulevaisuudessa taloushallinto ylläpitää niitä.

Näistä edellä mainituista tauluista Tilit, Va_selite, Te_selite sekä ToiminnanBudjetti tauluja voidaan ylläpitää myös SharePointin puolelta, mistä lisää kappaleessa 4.3.

4.2 Raportointi

Raportointipuoli talousraportointijärjestelmästä koostuu kahdesta osasta, tietokannassa olevista proseduureista sekä SQL Server Report Builder -ohjelmalla toteutetusta raportista. Alla olevassa kuvassa listatut proseduurit tekevät muutakin kuin raporttia, mutta ensisijaisesti ne laskevat ja palauttavat raportin pyytämän datan. Jokaiselle raportille on ainakin yksi proseduri, joka palauttaa dataa vain tälle raportille. Näin ollen laskentoja voidaan vaivattomasti muokata tietokannan päässä ilman erinäisiä sovelluksia, joita raportointi edellyttäisi, jos laskenta olisi tehty raportissa itsessään. Myöskään aivan kaikkea ei olisi voitu toteuttaa raporteissa, joten oli luontevaa siirtää kaikki laskenta tietokantaan. Katso liite 1 proseduurien riippuvuuksia varten.

Name	Schema	Owner
System Stored Procedures		
calc_PercentOfTimePassedOfYear	dbo	dbo
get_KohdistamattomatTilit	dbo	dbo
get_Menoseuranta	dbo	dbo
get_MenotTileittain	dbo	dbo
get_Paakirja_MaxTosPv	dbo	dbo
get_Te_Tulosaluekohtaiset	dbo	dbo
get_Va_RyhmakehtainenBudjettiJaMenot	dbo	dbo
get_Va_RyhmakehtaisetSummat	dbo	dbo
get_Va_RyhmakehtaisetSummatSub	dbo	dbo
get_Va_RyhmienMenoseuranta	dbo	dbo
Read_TAP-File_IntoDB	dbo	dbo

Kuva 3: Proseduurit

Funktiot tukevat järjestelmän tiedon esittämistä oikeassa muodossa, sallivat hakujen laajentamisen ja tekevät proseduureista selkeämpiä kokonaisuuksia. Funktioita on kahta eri tyyppiä: skalaarisia ja taulukko-funktioita. Taulukko-funktiota käytetään tässä järjestelmässä palauttamaan kyselyissä jo valmiiksi pääkirjasta suodatettu data, joten tätä ei tarvitse kirjoittaa jokaiseen kyselyyn erikseen ja suodatuksen hallinta on vain yhdessä paikassa. Skalaariset funktiot palauttavat aina jonkin yksittäisen arvon takaisin kyselylle. Katso liite 2 funktioiden riippuvuuksia varten.

4.3 SharePoint

SharePointissa oleva raportti käyttää SQL Server Reporting Services (SSRS) -palvelua hyödyntäen olemassa olevaa Energiaviraston Active Directoryä (AD). SharePoint Central Admin sivuston kautta on tehty raportoinnille Application Pool, jota käytetään raportoinnin palvelun ajamisen alustana nimeltään Service Application Pool - SSRSAppPool.

Suoraan käyttöönoton jälkeen SharePoint ei täysin tue ulkopuolisia data-lähteitä, mikä oli haasteena raportointijärjestelmälle. Mikäli raportti olisi haluttu esittää pelkästään Energiaviraston intraverkossa, perustoiminnot olisivat riittäneet. Mutta koska haluttiin mahdollisuus muokata raportin tietoja suoraan, se ei riittänyt. Tämän ongelman selvittämiseen käytettiin Business Data Connection-palvelua. Business Data Connection -palvelu pitää sisällään seuraavat tietoyhteydet, joista jokainen tuo yhden tietokannan taulun näkyviin Intraan, jossa sen tietoja voi muokata suoraan ilman tietokantatyökaluja.

Nimi	Taulun nimi tietokannassa	Tietolähteen tarkoitus ja ylläpito
Budjetti	ToiminnanBudjetti	Ylläpito syöttää manuaalisesti kerran vuodessa suoraan tietokantaan tai taloushallinto intrassa rivi kerrallaan.
Poistot	Te_Jako_Poistot	Taloushallinto laskee poistot kerran vuodessa ja lisää manuaalisesti ne tietokantaan Intran kautta.
Ryhmät	Va_selite	Taloushallinto ylläpitää manuaalisesti tätä Intrasta.
TAS-jako	TAS_Jako	Taloushallinto laskee TAS-jaon kerran vuodessa ja lisää sen manuaalisesti tietokantaan Intran kautta.
Tiliryhmät	Tiliryhmat	Taloushallinto ylläpitää manuaalisesti tätä Intrasta.
Tilit	Tilit	Taloushallinto ylläpitää manuaalisesti tätä Intrasta.
Työryhmät	Te_selite	Taloushallinto ylläpitää manuaalisesti tätä Intrasta.
Tilit	Tilit	Jos tili ei ole Tilit-taulussa, sitä tiliä ei oteta huomioon budjettilaskennoissa. Taloushallinto ylläpitää manuaalisesti tätä Intrasta.

Kuva 4: Käytetyt ulkopuoliset tietolähteet

Taloushallinnosta koulutettiin yksi henkilö ylläpitämään kuvassa 4 mainittuja tauluja, joita ei voida päivittää automaattisesti pääasiallisesta tietolähteestä. Nämä muutokset voidaan tehdä joko suoraan tietokantaan tai sitten intraverkosta löytyvästä lomakkeesta. Vaihtoehtoisesti, myös tietohallinnolla on oikeudet tehdä muutoksia tietokantaan intran kautta, mutta ensisijaisesti tämä on taloushallinnon tehtävä.

4.4 Esitettävä raportti

Kuten aikaisemmin mainittiin, itse esitettävä raportti on luotu käyttäen SQL Server Report Builder- ohjelmaa. Rakenteeltaan raportit on järjestetty puumaisesti, tulot ja menot- yhteen- veto raportin ollessa päällimmäinen. Tämä päällimmäinen raportti on myös se, joka esitetään

oletuksena intrassa. Tästä raportista voidaan pureutua sisemmälle tarkastelemaan muita puun raportteja, jotka antavat tietoa jaettuna ryhmittäin, tulosalueittain tai vastuualueittain.

Kaikkea tietokannassa listattua dataa ei tarvita raportin esittämistä varten. Osa tietokannassa olevista tauluista ja kentistä on vain tiedon sisäistä järjestelyä varten, eikä niitä ikinä tarvita tietokannan ulkopuolella. Liitteessä 5 oleva funktio palauttaa vain raportointia varten tarvittavan datan, jota liitteissä 3 ja 4 olevat proseduurit käyttävät esittääkseen sen halutussa muodossa.

The screenshot shows the Report Builder interface. On the left, a tree view lists data sources and fields under categories like 'Sisäiset kentät', 'Parametrit', 'Tietolähteet', 'Tietojoukot', and 'Ryhmakohittaiset'. The main design area displays a report layout with the following table structure:

«Lauseke»				«Lauseke»			
TULOT				«Lauseke»			
Tili	Tilin nimi	Toteuma	«Lauseke»				
[RyhmaSelite]	[Tili]	[Tilin_nimi]	«Lauseke»				
Yhteensä:			«Lauseke»				
MENOT				«Lauseke»			
Tili	Tilin nimi	Toteuma	Budjetti	Hakualueen budjetti	Hakualueen käyttö	Koko vuoden erotus	
[RyhmaSelite]	[Tili]	[Tilin_nimi]	«Lauseke»	«Lauseke»	«Lauseke»	«Lauseke»	«Lauseke»
Yhteensä:			[Sum(Tot)]	Sum(Budjetti]	ueenBudjetti]]	«Lauseke»	«Lauseke»
Yhteenveto			Kaikki yhteensä				
[TuloMenIndicator]			[Yhteensa]				
Yhteensä:			[Yhteensa]]				

Kuva 5: Tulot ja Menot -yhteenveto Report Builderissa

Koska tietokantaan saapuu data päivittäin, näin myös raporttien käsittelemä tietokin päivittyy päivittäin. Tai mikäli tietokantaan tehdään muutoksia, niin raporttien tieto päivittyy saman tien.

The screenshot shows the Report Builder interface. On the left, a tree view lists various report fields and parameters, including 'Sisäiset kentät' (Internal fields) and 'Parametrit' (Parameters). The main area displays a report preview with a table titled '«Lauseke»'. The table has columns for 'Tili' (Account), 'Tilin nimi' (Account name), 'Toteuma' (Actuals), 'Budjetti' (Budget), 'Hakualueen budjetti' (Budget area budget), 'Hakualueen % käyttö' (Budget area % usage), 'Budjettierotus €' (Budget variance €), and 'Budjettierotus %' (Budget variance %). The table includes summary rows for 'Yhteensä' (Total) and 'Budjettierotus % koko vuosi' (Budget variance % for the whole year).

Tili	Tilin nimi	Toteuma	Budjetti	Hakualueen budjetti	Hakualueen % käyttö	Budjettierotus €	Budjettierotus %
[Tili]	[Tilin_nimi]	«Lauseke»	«Lauseke»	«Lauseke»	«Lauseke»	«Lauseke»	«Lauseke»
Yhteensä:		[Sum(Total)]	[TotalBudjetti]	[ueenBudjetti]	«Lauseke»	«Lauseke»	«Lauseke»
Budjettierotus €:		«Lauseke»					
Budjettierotus % koko vuosi:		«Lauseke»					
«Lauseke»		«Lauseke»					

Kuva 6: Sisään pureutuminen Report Builderissa

Kuvat 5 ja 6 näyttävät suunnittelupuolen näkymän Report Builderista ja kuten näkyy, raportit luottavat paljon lausekkeisiin tiedon esittämisessä. Valittaessa ryhmän ylemmässä raportissa lauseke lataa alemman raportin käyttäen valittua ryhmää suodattaen pelkästään sen sisältämät tiedot. Mikä tili kuuluu mihinkin ryhmään ja tulosalueeseen määritellään tietokannan puolella, jota ylläpidetään käsin taloushallinnon toimesta. Hankeraportit, joita ei ole tässä esimerkkinä, on mahdollista avata myös pääkirjatasolle asti ja tarkastella esimerkiksi sitä, että mitkä erät ovat jo kirjanpidossa.

Esittäessä taloudellista dataa on oleellista, että esitetystä tiedosta käy ilmi organisaation tulot ja menot. Tärkeää on, että esitetystä tiedosta käy ilmi kuinka rahaa käytetään ja kuinka paljon sitä on saatavilla. (Beginners' guide to Financial Statements 2014)

Raportti esittää toteuman lisäksi vuosibudjetin sekä hakualueen budjetin. Hakualuetta voi käyttäjä muokata itse saadakseen halutun tiedon esille. Oletuksena hakualue näyttää joulukuun ensimmäisenä päivänä toteumat tammi-marraskuu suhteessa saman aikajakson budjettiin eli se on kalenterin mukainen. Tämä auttaa etenkin vuoden alussa, sillä se kertoo ovatko menot tulleet etu- vai takapainotteisesti.

Energiaviraston toiminnan tulot ja menot yhteenvetona:

Toiminnot |  | 

Tulot ja menot, 01.01.2013 - 30.12.2013

TULOT	Toteuma	Budjetti	Hakualueen budjetti	Hakualueen % käyttö	Erotus	Käyttö
Toiminnan tulot						
Yhteensä:						
MENOT	Toteuma	Budjetti	Hakualueen budjetti	Hakualueen % käyttö	Erotus	Käyttö
Ostetut tarvikkeet						
Henkilöstömenot						
Vuokrat						
Ostetut palvelut						
Investoinnit						
Matkat						
Muut menot						
Yhteensä:						

Ali/Ylijäämä	
Käytettävissä oleva määräraha	
Jäljellä	
Budjettierotus €	
Budjettierotus %	

Raportin lukuohje:

Toteuma = syötetyn hakualueen välinen yhteenlaskettu tulo tai meno.

Budjetti = Koko vuoden budjetti valitun alkupäivämäärän mukaan.

Hakualueen budjetti = syötetyn loppupäivämäärän perusteella laskettu euromääräinen prosenttiosa budjetista, esim. 1.7.2013 = 50% budjetista ja 31.12.2013 = 100%.

Hakualueen % käyttö = syötetyn loppupäivämäärän perusteella laskettu prosenttiosa budjetista joka on käytetty syötetyn loppupäivämäärän hetkellä.

Erotus = Budjetti - Toteuma

Käyttö = Toteuma / Budjetti %

Ali/Ylijäämä = Tulot yhteensä - Menot yhteensä

Budjettierotus = menojen Budjetti yhteensä - Toteuma yhteensä

Budjettierotus % = budjettierotuksen suhde budjettiin

Kuva 7: Tulot ja Menot -yhteenveito intranetissa, luvut poistettu

Raporteissa on mahdollista pureutua sisään, kuten aikaisemmin mainittiin sekä rakenneosiossa kuvattiin. Kuvassa 8 esitetään esimerkkinä tätä toimintoa käyttäen ryhmää investoinnit.

Yhteenveito, Investoinnit, 01.01.2013 - 30.12.2013

Tili	Tilin nimi	Toteuma	Budjetti	Hakualueen budjetti	Hakualueen % käyttö	Budjettierotus €	Budjettierotus %
	Ostetut atk-ohjelmistot						
	Itsevalm. ja teetetyt atk-ohj.						
	Keskeneräiset aineettomat ktö						
	Yhteensä:						

Budjettierotus €:	
Budjettierotus % koko vuosi:	
Budjettierotus % vuodesta:	

Raportin lukuohje:

Toteuma = syötetyn hakualueen välinen yhteenlaskettu tulo tai meno.

Budjetti = Koko vuoden budjetti valitun alkupäivämäärän mukaan.

Hakualueen budjetti = syötetyn loppupäivämäärän perusteella laskettu euromääräinen prosenttiosa budjetista, esim. 1.7.2013 = 50% budjetista ja 31.12.2013 = 100%.

Hakualueen % käyttö = syötetyn loppupäivämäärän perusteella laskettu prosenttiosa budjetista joka on käytetty syötetyn loppupäivämäärän hetkellä.

Erotus = Budjetti - Toteuma

Käyttö = Toteuma / Budjetti %

Ali/Ylijäämä = Tulot yhteensä - Menot yhteensä

Budjettierotus = menojen Budjetti yhteensä - Toteuma yhteensä

Budjettierotus % = budjettierotuksen suhde budjettiin

Kuva 8: Sisään pureutuminen intraverkossa, luvut poistettu

Kuvissa 7 ja 8 esitetään vertailuna kuviin 5 ja 6, miltä valmis raportti näyttää esitettynä intraverkossa. Raportin visuaalisessa suunnittelussa pyrittiin esittämään tieto selkeästi ja yksinkertaisesti. Taloushallinnolla oli omat kriteerinsä sille, miten tietoa piti saattaa esille. Pohjana raportin suunnittelulle käytettiin jo olemassa olevaa Excel- taulukkoa, johon aikaisemmin taloushallinto syötti käsin sen, mitä tämä raportti esittää automaattisesti.

5 Hylätyksi päätynyt ratkaisuvaihtoehto

Ratkaisuvaihtoehto, joka päättyi hylätyksi, oli raportoinnin toteutus käyttäen pelkästään Excel- ohjelmaa. Ajatuksena oli hyödyntää Excelin PowerPivot lisäosaa vetämään tietoja joko tietokannasta tai sitten toiselta välilehdeltä Excelistä. Välilehtiversiossa tieto tuotaisiin manuaalisesti sinne, mutta koska tämä tehtäisiin vain kerran vuodessa, sen ei ajateltu tuottavan liikaa vaivaa. Lisäksi aikomuksena oli tehdä tästä SharePointtiin myös oma versio, jotta tieto saataisiin esitettyä organisaation jäsenille.

Sytä Excel-pohjaisesta ratkaisusta luopumiselle oli useita. Ensinnäkin tässä ratkaisussa tuli ilmi useita teknisiä ongelmia. Hyvän tiedonhallintatavan mukaan tiedon on oltava saatavilla, käytettävissä, suojattu sekä eheää. (Hyvän tiedonhallintatavan määrittäminen 2000)

Käyttäessä Exceliä tähän tarkoitukseen, ongelmaksi nousi tiedon eheys. Mikäli Exceliin tuotava tieto muuttui rakenteellisesti, ei ollut varmuutta osaisiko raportti käsitellä sen oikein. Tämän todettiin vaarantavan tiedon eheyden. Tämän olisi mahdollisesti pystynyt kiertämään ajamalla saapuvan raakadatan jonkinlaisen suodatus-skriptin läpi ensin. Katso kuva 1 esimerkkinä minkälainen suodatus-skripti olisi ollut mahdollinen

Toiseksi, koska raportista oltaisiin tekemässä joka tapauksessa versio SharePointtiin, heräsi kysymys, miksi koko raporttia ei siirrettäisi sinne, mikä helpottaisi tiedon esittämistä organisaatiossa. Samalla heräsi ajatus, että mikäli tieto joudutaan suodattamaan sen eheyden varmistamiseksi, niin miksi kaikkea dataa ei siirrettäisi SQL-tietokantaan, mikä tekisi sen varastoinnista siistimpää kuin sen pitämisessä täysin Excel-taulukossa.

6 Energiaviraston taloushallinnon arvio

Seuraava kommentti on Energiaviraston taloushallinnon arvio talousraportointijärjestelmän toimivuudesta ja siitä vastasiko se tarvetta. Arviosta on poistettu sen esittäjän luvalla joitain kohtia tietoturvan sekä muotoilun vuoksi.

”Virasto tarvitsi taloutensa seurantaan käyttökelpoisempia raportteja kuin ne valmisraportit, joita taloushallintofunktio pystyy kirjanpitojärjestelmästä (valtion Raindance) ajamaan. Li-

säksi tarvittiin raportteja, joissa on mukana budjettiseuranta. Yleisen budjettitietoisuuden lisäämiseksi haluttiin, että raportit olisivat kaikkien nähtävillä intrassa. Tulosalueiden kustannusvastaavuuden seurantaan tarvittiin raportti, joka ottaa työajan kohdentamisen huomioon, ja tätä tietoa ei saa peruskirjanpidosta ollenkaan. Koska työajan kohdentaminen toimii ajurina sekä palkkojen että yhteisten kulujen vyöryttämisessä, oli kyse varsin mittavasta osasta kustannuksia, jota ei pystynyt pelkän kirjanpidon avulla seuraamaan.

Intraan viedyt raportit vastaavat hyvin ainakin hallinnon tarpeita, ja vähentävät oleellisesti tarvetta käydä Raindancerista hakemaan ad hoc -raportteja. Excelin käyttö raportoinnissa on vähentynyt, mikä vähentää virheiden riskiä ja taloushallinnon työtaakkaa. Tulot ja menot -raporteilla voidaan seurata sekä koko viraston, että ryhmien budjettien toteutusta. Tulosalueiden tuotot ja kulut -raportti näyttää välittömien kulujen lisäksi myös tulosalueille kirjatun työajan kustannukset sekä tulosalueiden osuuden yleiskustannuksista. Raportti ei näytä poistoja, koska Raindance ei kirjaa käyttöomaisuutta tulosalueille, joten poistot jaetaan manuaalisesti tilinpäätöksen yhteydessä.

Raportit voidaan lisäksi tarpeen vaatiessa viedä Exceliin jatkokäsittelyä varten. Raportit käyttävät edellisen päivän kirjanpidon dataa, joten ne ovat melkein reaaliaikaisia. Raportit löytyvät intrasta ylänauhan Raportit-otsikon alta. Samassa paikassa on mm tietoja substanssialueiden toimivuudesta.

Raportit näyttävät myös toteuman prosentteina, ja värikoodattuna alitukset ja ylitykset. Näin ollen voin yhdellä silmäyksellä nähdä miten budjetti on toteutumassa. Raportit ovat helppokäyttöisiä, ja niissä on selkeät Päivitä ja Palaa -napit. Näytön koosta riippuen osa raportista saattaa jäädä ulkopuolelle, mikä hieman haittaa käytettävyyttä.

Kaikki raportit käyttävät kirjanpidon pääkirjamaista TAP raporttia. Tästä samasta raportista suodatetaan tiedot eri seurantakohteen mukaan: ryhmä, toiminto tai hanke.

Raporteista ei tule erillistä ilmoitusta käyttäjille. Poikkeuksena tähän on siirtoraportti, joka tulee taloushallintoon kerran vuorokaudessa ja kertoo, että TAP raportti on onnistuneesti siirretty viraston palvelimelle. Käyttäjien tulee siis itse käydä raporttia katsomassa. Tätä ei vielä riittävässä määrin tehdä. Tämä on kuitenkin enemmän viestinnällinen haaste.

Raporteissa on käytetty viraston visuaalinen ilmeen värejä jonkun verran. Värikoodit (sininen=alle budjetoidun, vahva oranssi=yli budjetin enemmän kuin 10 %) eivät kuitenkaan oikein sovi visuaaliseen ilmeeseen. Koska tulo- ja menoraportti näyttää sekä hakualueen budjetti + käyttö % että koko vuoden budjetin + käyttö %, se voi olla tottumattomalle sekava. Raportti ei kuitenkaan ole saanut kielteistä palautetta.” (Wasström, 2014)

7 Yhteenveto

Opinnäytetyön tavoitteena oli rakentaa Energiavirastolle talousraportointijärjestelmä jakamaan budjetti- ja taloustietoa organisaation sisällä. Opinnäytetyön tavoite eli talousraportointijärjestelmän toteuttaminen tavoitettiin ja tuloksena oli tässä dokumentissa kuvattu järjestelmä.

Joitain ongelmia, mitä tuli esille projektin aikana oli, että tiimi ei hallinnut taloustermistöä ja taloushallinto ei hallinnut IT- alan termistöä, mikä johti erinäisiin väärinkäsityksiin palautteen annossa työn edistymisestä sekä määritellessä ja kuvaillessa järjestelmän toimintoja ja toimivuutta.

Talousraportointijärjestelmän käyttöönotto on helpottanut taloushallinnon työtä. Järjestelmään on oltu tyytyväisiä. Lukuun ottamatta joitain julkistuksen jälkeisiä pieniä bugeja ja virheitilanteita järjestelmä on toiminut niin kuin suunniteltiin.

Jatkokehitys raportointijärjestelmälle on hieman rajattua, sillä kyseessä on kuitenkin taloushallinnon järjestelmä, mutta raportointia on mahdollista laajentaa sisältämään muitakin erilaisia raportteja. Näin on itse asiassa tapahtunut raportointijärjestelmän julkaisun jälkeen, raportointiin lisättiin uusi osa-alue eli tiedon esittäminen hankkeittain. Lisäksi, järjestelmän toteutusta voidaan soveltaa muunkinlaisiin raportteihin, kunhan tehdään tarvittavia muutoksia.

Lähteet

Kirjalähteet

Delaney, K. Microsoft SQL Server 2012 Internals. 2013. O'Reilly Media Inc

Krishnaswamy, J. Learning SQL Server Reporting Services 2012. 2014. Packt Publishing

Schwaber, K., Beedle, M. Agile Software Development with SCRUM. 2002. Prentice Hall

Shafiq, S. 2014. A practical guide to SharePoint 2013. Wali Systems Inc.

Sähköiset lähteet

Beginners' Guide to Financial Statements. 2014. US Gov. Viitattu 18.11.2014.

<http://www.sec.gov/investor/pubs/begfinstmtguide.htm>

Hyvän tiedonhallintatavan määrittäminen. 2000. Valtiovarainministeriö. Viitattu 20.10.2014.

https://www.vm.fi/vm/fi/04_julkaisut_ja_asiakirjat/01_julkaisut/04_hallinnon_kehittaminen/4127/4128_fi.pdf

Top 10 Best Practices for Large Scale Relational Data Warehouse. 2013. Viitattu 3.12.2014

<http://blogs.msdn.com/b/sqlcat/archive/2013/09/16/top-10-best-practices-for-building-a-large-scale-relational-data-warehouse.aspx>

Julkaisemattomat lähteet

Wasström, P. Energiaviraston taloushallinnon arviointi raportointijärjestelmästä 3.12.2014. Energiavirasto. Helsinki

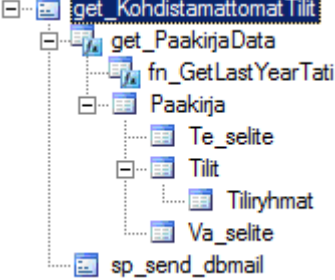
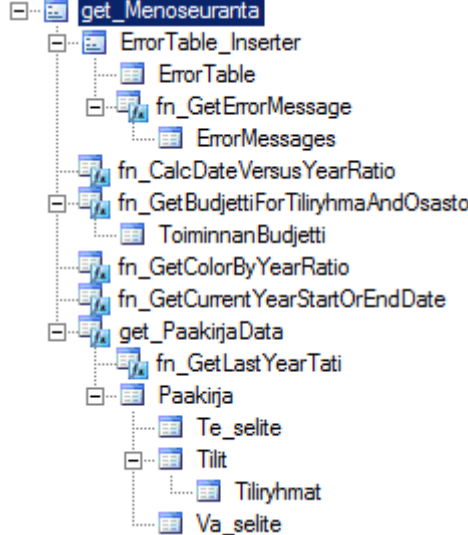
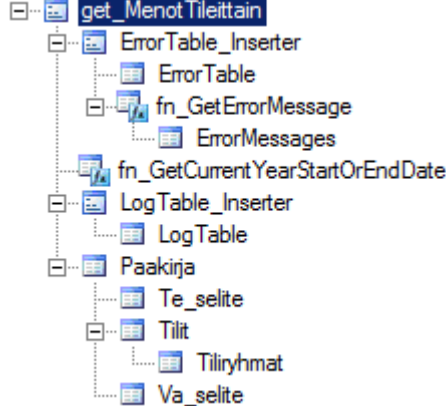
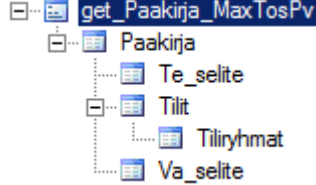
Kuvat

Kuva 1: Skripti tiedostojen siirtoon sftp-palvelimelta	10
Kuva 2: Tietokannan taulujen riippuvuudet.....	11
Kuva 3: Proseduurit.....	12
Kuva 4: Käytetyt ulkopuoliset tietolähteet	13
Kuva 5: Tulot ja Menot -yhteenveto Report Builderissa	14
Kuva 6: Sisään pureutuminen Report Builderissa.....	15
Kuva 7: Tulot ja Menot -yhteenveto intranetissa, luvut poistettu	16
Kuva 8: Sisään pureutuminen intraverkossa, luvut poistettu	16

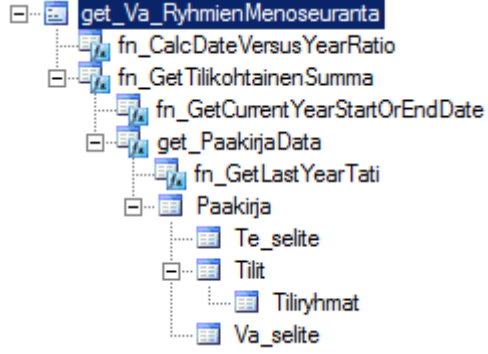
Liitteet

Liite 1: Proseduurien riippuvuudet	23
Liite 2: Funktioiden riippuvuudet	26
Liite 3: Päällimäisen raportin muodostava proseduuri	28
Liite 4: Tilikohtaista pureutumista varten tiedot hakeva proseduuri.....	32
Liite 5: Raportointia varten haettava data funktio	34

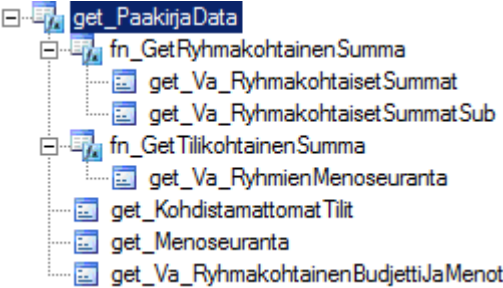
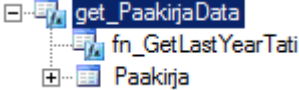
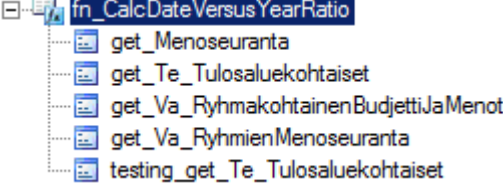
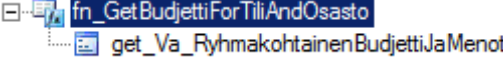
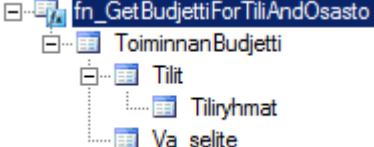
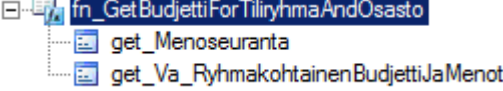
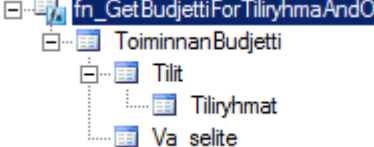
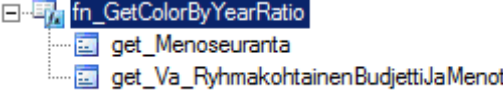
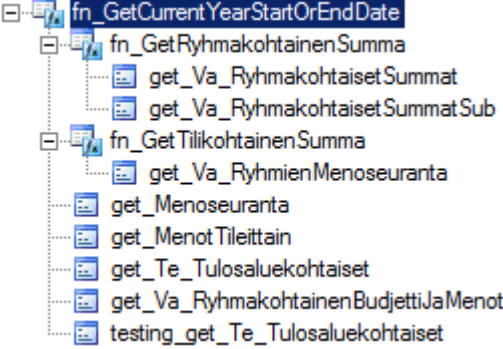
Liite 1: Proseduurien riippuvuudet

Tietokantaobjektit jotka ovat riippuvaisia tästä tietokantaobjekteista ja järjestelmän ominaisuuksista	Proseduuri on riippuvainen näistä tietokantaobjekteista ja järjestelmän palveluista
	 <p>get_KohdistamattomatTilit get_PaakirjaData fn_GetLastYearTati Paakirja Te_selite Tilit Tiliryhmat Va_selite sp_send_dbmail</p>
	 <p>get_Menoseuranta ErrorTable_Inserter ErrorTable fn_GetErrorMessage ErrorMessages fn_CalcDateVersusYearRatio fn_GetBudjettiForTiliryhmaAndOsasto ToiminnanBudjetti fn_GetColorByYearRatio fn_GetCurrentYearStartOrEndDate get_PaakirjaData fn_GetLastYearTati Paakirja Te_selite Tilit Tiliryhmat Va_selite</p>
	 <p>get_MenotTilittain ErrorTable_Inserter ErrorTable fn_GetErrorMessage ErrorMessages fn_GetCurrentYearStartOrEndDate LogTable_Inserter LogTable Paakirja Te_selite Tilit Tiliryhmat Va_selite</p>
	 <p>get_Paakirja_MaxTosPv Paakirja Te_selite Tilit Tiliryhmat Va_selite</p>

	<ul style="list-style-type: none"> get_Te_Tulosaluekohtaiset <ul style="list-style-type: none"> fn_CalcDateVersusYearRatio fn_GetCurrentYearStartOrEndDate Paakirja <ul style="list-style-type: none"> Te_selite Tilit <ul style="list-style-type: none"> Tiliryhmat Va_selite TAS_Jako TE_Jako_Poistot
	<ul style="list-style-type: none"> get_Va_RyhmakohtainenBudjettiJaMenot <ul style="list-style-type: none"> fn_CalcDateVersusYearRatio fn_GetBudjettiForTiliAndOsasto <ul style="list-style-type: none"> ToiminnanBudjetti <ul style="list-style-type: none"> Tilit <ul style="list-style-type: none"> Tiliryhmat Va_selite fn_GetBudjettiForTiliryhmaAndOsasto fn_GetColorByYearRatio fn_GetCurrentYearStartOrEndDate get_PaakirjaData <ul style="list-style-type: none"> fn_GetLastYearTati Paakirja <ul style="list-style-type: none"> Te_selite
	<ul style="list-style-type: none"> get_Va_RyhmakohtaisetSummat <ul style="list-style-type: none"> fn_GetRyhmakohtainenSumma <ul style="list-style-type: none"> fn_GetCurrentYearStartOrEndDate get_PaakirjaData <ul style="list-style-type: none"> fn_GetLastYearTati Paakirja <ul style="list-style-type: none"> Te_selite Tilit <ul style="list-style-type: none"> Tiliryhmat Va_selite
	<ul style="list-style-type: none"> get_Va_RyhmakohtaisetSummatSub <ul style="list-style-type: none"> fn_GetRyhmakohtainenSumma <ul style="list-style-type: none"> fn_GetCurrentYearStartOrEndDate get_PaakirjaData <ul style="list-style-type: none"> fn_GetLastYearTati Paakirja <ul style="list-style-type: none"> Te_selite Tilit <ul style="list-style-type: none"> Tiliryhmat Va_selite

	 <ul style="list-style-type: none">get_Va_RyhmiemMenoseuranta<ul style="list-style-type: none">fn_CalcDateVersusYearRatiofn_GetTilikohtainenSumma<ul style="list-style-type: none">fn_GetCurrentYearStartOrEndDateget_PaakirjaData<ul style="list-style-type: none">fn_GetLastYearTatiPaakirja<ul style="list-style-type: none">Te_seliteTilit<ul style="list-style-type: none">TiliryhmatVa_selite
	 <ul style="list-style-type: none">Read_TAP-File_IntoDB<ul style="list-style-type: none">fn_GetDateString_yyMMddfn_GetDateString_yyyy-mm-ddPaakirja<ul style="list-style-type: none">Te_seliteTilit<ul style="list-style-type: none">TiliryhmatVa_selitesp_send_dbmailTAPImportLog

Liite 2: Funktioiden riippuvuudet

Tietokantaobjektit jotka ovat riippuvaisia funktiosta	Funktio on riippuvainen näistä tietokantaobjekteista:
<p>get_PaakirjaData suodattaa paakirja-taulusta kaikki raportoinnille oleelliset rivet.</p>  <pre> graph TD A[get_PaakirjaData] --- B[fn_GetRyhmakehtainenSumma] A --- C[fn_GetTilikohtainenSumma] B --- D[get_Va_RyhmakehtaisetSummat] B --- E[get_Va_RyhmakehtaisetSummat.Sub] C --- F[get_Va_Ryhmiemenoseuranta] C --- G[get_KohdistamattomatTilit] C --- H[get_Menoseuranta] C --- I[get_Va_RyhmakehtainenBudjettiJaMenot] </pre>	 <pre> graph TD A[get_PaakirjaData] --- B[fn_GetLastYearTati] A --- C[Paakirja] </pre>
 <pre> graph TD A[fn_CalcDateVersusYearRatio] --- B[get_Menoseuranta] A --- C[get_Te_Tulosaluekohtaiset] A --- D[get_Va_RyhmakehtainenBudjettiJaMenot] A --- E[get_Va_Ryhmiemenoseuranta] A --- F[testing_get_Te_Tulosaluekohtaiset] </pre>	
 <pre> graph TD A[fn_GetBudjettiForTiliAndOsasto] --- B[get_Va_RyhmakehtainenBudjettiJaMenot] </pre>	 <pre> graph TD A[fn_GetBudjettiForTiliAndOsasto] --- B[ToiminnanBudjetti] B --- C[Tilit] C --- D[Tiliryhmat] C --- E[Va_selite] </pre>
 <pre> graph TD A[fn_GetBudjettiForTiliryhmaAndOsasto] --- B[get_Menoseuranta] A --- C[get_Va_RyhmakehtainenBudjettiJaMenot] </pre>	 <pre> graph TD A[fn_GetBudjettiForTiliryhmaAndOsasto] --- B[ToiminnanBudjetti] B --- C[Tilit] C --- D[Tiliryhmat] C --- E[Va_selite] </pre>
 <pre> graph TD A[fn_GetColorByYearRatio] --- B[get_Menoseuranta] A --- C[get_Va_RyhmakehtainenBudjettiJaMenot] </pre>	
 <pre> graph TD A[fn_GetCurrentYearStartOrEndDate] --- B[fn_GetRyhmakehtainenSumma] A --- C[fn_GetTilikohtainenSumma] B --- D[get_Va_RyhmakehtaisetSummat] B --- E[get_Va_RyhmakehtaisetSummat.Sub] C --- F[get_Va_Ryhmiemenoseuranta] C --- G[get_Menoseuranta] C --- H[get_MenotTileittain] C --- I[get_Te_Tulosaluekohtaiset] C --- J[get_Va_RyhmakehtainenBudjettiJaMenot] C --- K[testing_get_Te_Tulosaluekohtaiset] </pre>	

<ul style="list-style-type: none"> [-] fn_GetDateString_yyMMdd <ul style="list-style-type: none"> [-] Read_TAP-File_IntoDB 	
<ul style="list-style-type: none"> [-] fn_GetDateString_yyyy-mm-dd <ul style="list-style-type: none"> [-] Read_TAP-File_IntoDB 	
<ul style="list-style-type: none"> [-] fn_GetLastYearTati <ul style="list-style-type: none"> [+] get_PaakirjaData 	
	<ul style="list-style-type: none"> [-] fn_GetRyhmalID <ul style="list-style-type: none"> [-] Tilit <ul style="list-style-type: none"> [-] Tiliryhmat
<ul style="list-style-type: none"> [-] fn_GetRyhmakehtainenSumma <ul style="list-style-type: none"> [-] get_Va_RyhmakehtaisetSummat [-] get_Va_RyhmakehtaisetSummatSub 	<ul style="list-style-type: none"> [-] fn_GetRyhmakehtainenSumma <ul style="list-style-type: none"> [-] fn_GetCurrentYearStartOrEndDate [-] get_PaakirjaData <ul style="list-style-type: none"> [-] fn_GetLastYearTati <ul style="list-style-type: none"> [-] Paakirja <ul style="list-style-type: none"> [-] Te_selite [-] Tilit <ul style="list-style-type: none"> [-] Tiliryhmat [-] Va_selite
<ul style="list-style-type: none"> [-] fn_GetTilikohtainenSumma <ul style="list-style-type: none"> [-] get_Va_Ryhmiemenoseuranta 	<ul style="list-style-type: none"> [-] fn_GetTilikohtainenSumma <ul style="list-style-type: none"> [-] fn_GetCurrentYearStartOrEndDate [-] get_PaakirjaData <ul style="list-style-type: none"> [-] fn_GetLastYearTati <ul style="list-style-type: none"> [-] Paakirja <ul style="list-style-type: none"> [-] Te_selite [-] Tilit <ul style="list-style-type: none"> [-] Tiliryhmat [-] Va_selite

Liite 3: Päällimäisen raportin muodostava proseduuri

```

USE [POISTETTU]
GO
/***** Object:  StoredProcedure [dbo].[get_Menoseuranta]    Script Date:
2.12.2014 13:38:24 *****/
SET ANSI_NULLS OFF
GO
SET QUOTED_IDENTIFIER OFF
GO

-- =====
-- Author:                P. Mäkelä, M. Laine, Energiamarkkinavirasto
-- Create date: 03.07.2013
-- Description:          Palauttaa "tiliryhmien taloustiedot vs budjetti" tai
-- "tili/tiliryhmäkohtaiset taloustiedot" SQL Reporting Service raportille
-- xxxxx (SharePointissa) perustuen annettuihin parametreihin.
-- =====

ALTER PROCEDURE [dbo].[get_Menoseuranta]
(
    @RyhmaID                INT --- NULL
    ,@AlkuPvm                DATE --- NULL
    ,@LoppuPvm                DATE --- NULL
)
AS
BEGIN
    --DECLARE @CurrentProcedureName VARCHAR(50) = OBJECT_NAME(@@PROCID);
    --DECLARE @CustomErrorMessage VARCHAR(500) = NULL;

    -- If date parameters are empty, too short or NULL,
    -- those are changed into:
    -- AlkuPvm = The first date of current year
    -- LoppuPvm = The last date of current year
    SELECT @AlkuPvm =
[dbo].[fn_GetCurrentYearStartOrEndDate]('StartDate', @AlkuPvm)
    SELECT @LoppuPvm =
[dbo].[fn_GetCurrentYearStartOrEndDate]('EndDate', @LoppuPvm)

    -- Get max date from Paakirja
    DECLARE @MaxDate AS VARCHAR(10)
    SELECT @MaxDate = CAST(MAX([Tospv]) AS VARCHAR(10)) FROM [POISTET-
TU].[dbo].[ POISTETTU]
    PRINT @MaxDate
    -- And if LoppuPvm is greater than Max date in Paakirja, we only
query based on smallest value
    IF @LoppuPvm > @MaxDate SET @LoppuPvm = @MaxDate
    PRINT @LoppuPvm

    -- Fetch End Date Versus Year Ratio
    DECLARE @Ratio AS DECIMAL(5,2)
    SET @Ratio = dbo.[fn_CalcDateVersusYearRatio](CONVERT(VARCHAR(10),
@LoppuPvm, 120))*100
    PRINT ISNULL(CAST(@Ratio AS VARCHAR(10)), 'null')

    IF @RyhmaID = NULL
    BEGIN
        BEGIN TRY
            PRINT 'RyhmaID = NULL'
            SELECT

```

```

        x.RyhmaID
        , x.RyhmaSelite
        , x.Total
    )
    dbo.fn_GetBudjettiForTiliryhmaAndOsasto(x.RyhmaID,999,YEAR(@AlkuPvm)) AS Total-
    Budjetti
    ,
    dbo.fn_GetBudjettiForTiliryhmaAndOsasto(x.RyhmaID,999,YEAR(@AlkuPvm))*(@Ratio/100
    ) AS HakualueenBudjetti
    , CASE WHEN
    dbo.fn_GetBudjettiForTiliryhmaAndOsasto(x.RyhmaID,999,YEAR(@AlkuPvm)) = 0
        OR @Ratio = 0
        THEN
    NULL
        ELSE
    (-1)*x.Total / (((-
    1)*dbo.fn_GetBudjettiForTiliryhmaAndOsasto(x.RyhmaID,999,YEAR(@AlkuPvm)))*(@Ratio
    /100.00))
        END HakualueenKaytto
    , @Ratio AS DateVersusYearRatio
    ,[dbo].[fn_BudgetUseHighlightColor]((-
    1)*dbo.fn_GetBudjettiForTiliryhmaAndOsasto(x.RyhmaID,999,YEAR(@AlkuPvm)),x.Total,
    @Ratio) AS Color
    FROM (
        SELECT
            --t.Tili
            t.RyhmaID
            ,MIN(r.Selite) AS
    RyhmaSelite
            ,SUM(p.Tot) AS To-
    tal
        FROM
    dbo.get_PaakirjaData(@AlkuPvm, @LoppuPvm) p
        LEFT JOIN dbo.Tilit t ON t.Tili
    = p.Tili
        LEFT JOIN dbo.Tiliryhmat r ON
    r.RyhmaID = t.RyhmaID
        GROUP BY t.RyhmaID) x
    ORDER BY x.RyhmaID
    END TRY
    -- Testausta varten
    BEGIN CATCH
        PRINT 'Error: ' + ERROR_MESSAGE()
    END CATCH
    END --IF @RyhmaID = NULL
    ELSE
    BEGIN
        BEGIN TRY
            PRINT 'ELSE, ' + CAST(@RyhmaID AS VAR-
    CHAR(4))
            -- SELECT palauttaa tulosjoukon Tilin mukaan
            ryhmiteltynä kun RyhmaID on jotain muuta kuin NULL
            --SELECT t.Tili, MIN(t.Tilin_nimi) AS
    Tilin_nimi, SUM(p.Tot) AS Total, SUM(b.Total) AS TotalBudjetti
            SELECT
                x.Tili
                ,x.Tilin_nimi
                ,x.Total
                ,x.TotalBudjetti

```

```

AS HakualueenBudjetti
    ,x.TotalBudjetti*(@Ratio/100)
    ,CASE WHEN x.TotalBudjetti = 0
        THEN NULL
        ELSE (-
1)*x.Total / ((-1)*x.TotalBudjetti*(@Ratio/100))
        END HakualueenKaytto
    ,@Ratio AS DateVersusYearRatio
    --,
dbo.fn_GetColorByYearRatio((-1)*x.Total, ((-1)*x.TotalBudjetti*(@Ratio/100) )) AS
Color
    ,[dbo].[fn_BudgetUseHighlightColor]((-
1)*x.TotalBudjetti,x.Total,@Ratio) AS Color
    ,x.Vuosi
FROM (
SELECT
    t.Tili
    , MIN(t.Tilin_nimi)
    , SUM(p.Tot) AS
    , SUM(b.Total) AS
    , MIN(b.Vuosi) AS
AS Tilin_nimi
Total
TotalBudjetti
Vuosi
FROM [dbo].[Tilit] t
LEFT JOIN
(
SELECT
    Tili
    , '999' AS Osasto
    , SUM(Total) AS
    , MIN(Vuosi) AS
FROM [dbo].[ToiminnanBudjetti]
WHERE
    Vuosi =
    AND Osasto < 999
GROUP BY Tili
) b ON t.Tili = b.Tili
LEFT JOIN
(
SELECT
    Tili
    , SUM(Tot) AS Tot
FROM
    GROUP BY Tili
) p ON p.tili = t.Tili
WHERE t.RyhmaID = @RyhmaID
GROUP BY t.Tili
) x
END TRY
-- Testausta varten
BEGIN CATCH
PRINT 'Error: ' + ERROR_MESSAGE()
END CATCH

```

END

END

Liite 4: Tilikohtaista pureutumista varten tiedot hakeva proseduuri

```

USE [POISTETTU]
GO
/***** Object:  StoredProcedure [dbo].[get_MenotTileittain]    Script Date:
2.12.2014 16:55:36 *****/
SET ANSI_NULLS OFF
GO
SET QUOTED_IDENTIFIER OFF
GO

-- =====
-- Author:          P. Mäkelä, M. Laine, Energiamarkkinavirasto
-- Create date: 10.07.2013
-- Description:     Palauttaa taloustiedot per valittu tili
-- =====

ALTER PROCEDURE [dbo].[get_MenotTileittain]
(
    @Tili          INT = NULL,
    @AlkuPvm      DATE = NULL,
    @LoppuPvm     DATE = NULL
)
AS
BEGIN
    --DECLARE @CurrentProcedureName VARCHAR(50) = OBJECT_NAME(@@PROCID);
    --DECLARE @CustomErrorMessage VARCHAR(500) = NULL;

    --BEGIN TRY
    --    EXEC [dbo].[LogTable_Inserter] @User, @ReportName
    --END TRY
    --BEGIN CATCH
    --    SELECT @CustomErrorMessage = dbo.fn_GetErrorMessage(1)
    --    EXEC [dbo].[ErrorTable_Inserter] @CurrentProcedureName,
@CustomErrorMessage, @@ERROR
    --END CATCH

    BEGIN TRY
        -- If date parameters are empty, too short or NULL,
        -- those are changed into:
        -- AlkuPvm = The first date of current year
        -- LoppuPvm = The last date of current year
        SELECT @AlkuPvm =
[dbo].[fn_GetCurrentYearStartOrEndDate]('StartDate', @AlkuPvm)
        SELECT @LoppuPvm =
[dbo].[fn_GetCurrentYearStartOrEndDate]('EndDate', @LoppuPvm)

        IF @AlkuPvm > @LoppuPvm
        BEGIN
            SELECT 0 AS Tili, 0 AS Total, 0 AS Total-
Budjetti, dbo.fn_GetErrorMessage(2) AS RyhmaSelite
        END
    END TRY
    -- Testausta varten
    BEGIN CATCH
        PRINT 'Error: ' + ERROR_MESSAGE()
    END CATCH

    BEGIN TRY
        IF @Tili != NULL

```



```
begin
    PRINT 'Tili = ' + CAST(@Tili AS VARCHAR(10))
    SELECT t.Tili
           ,MIN(p.[Tilin_nimi]) AS Tilin_nimi
           ,SUM(p.Tot) AS Total
    FROM Paakirja p
         INNER JOIN Tilit t ON t.Tili = p.Tili
    WHERE t.Tili = @Tili
           AND (Tati = 326001 OR Tati =
411326001)
           AND ((@AlkuPvm IS NULL AND
@LoppuPvm IS NULL)
           OR (Tospv BETWEEN
@AlkuPvm AND @LoppuPvm))
    GROUP BY t.tili
end
ELSE
begin
SELECT 0000, 'Ei tiliä parametrina!', 0
end
END TRY
-- Testausta varten
BEGIN CATCH
PRINT 'Error: ' + ERROR_MESSAGE()
END CATCH
END
```

Liite 5: Raportointia varten haettava data funktio

```
USE [POISTETTU]
GO
/***** Object: UserDefinedFunction [dbo].[get_PaakirjaData]    Script Date: 2.12.2014
17:10:12 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

-- =====
-- Author:                P.Mäkelä, Peaches Industries
-- Create date: 06.08.2013
-- Description:           Palauttaa pääkirjasta vain raportoinnissa
-- tarvittavat rivit
-- =====

ALTER FUNCTION [dbo].[get_PaakirjaData]
(
    -- Add the parameters for the function here
    @AlkuPvm                DATE = NULL,
    @LoppuPvm                DATE = NULL
)
RETURNS TABLE
AS
RETURN
(
    SELECT
        [Tospv]
    , [Tosno]
    , [Tosri]
    , [Toslaji]
    , [Mp]
    , [Tili]
    , [Tilin_nimi]
    , [Tati]
    , [Va]
    , [Te]
    , [Hanke]
    , [Hlo]
    , [Suor]
    , [E]
    , [Tot]
    , [Tosots]
    , [Selite]
    FROM [EMV_ReportingData].[dbo].[Paakirja]
    WHERE
        (Tati = 326001 OR Tati = [dbo].[fn_GetLastYearTati](YEAR(@AlkuPvm)))
        AND (Tili IN (1191,1140,1120)
            OR Tili BETWEEN 3000 AND 4800
            OR Tili BETWEEN 4900 AND 9000
            OR Tili BETWEEN 40001 AND 46000)
        AND ((@AlkuPvm IS NULL AND @LoppuPvm IS NULL)
            OR (Tospv BETWEEN @AlkuPvm AND @LoppuPvm))

```

)