

OLAP-tekniikka oppilastietojärjestelmässä

LAHTI UNIVERSITY OF APPLIED
SCIENCES
Bachelor's Degree Programme in
Business Information Technology

Thesis
22.1.2019
Ville Pohjonen

Lahti University of Applied Sciences
Bachelor's Degree Programme in Business Information Technology

POHJONEN, VILLE: OLAP-technology in student
information system

Bachelor's Thesis in Business Information Technology 32 pages

Spring 2019

ABSTRACT

The bachelor's thesis was commissioned by the information management section of the Lahti University of Applied Sciences. The aim of the thesis was to deepen my own knowledge of databases, data warehousing and OLAP solutions and to create guidelines for creating an OLAP-cube. The aim of the thesis was also to study how suitable the OLAP solution is for use in the student information system.

The thesis implementation section was carried out using Microsoft SQL Server Management Studio 17(SSMS) and Microsoft Visual Studio 2017. SSMS software was used to edit and create databases and the cOLAP cube was made with Visual Studio. The softwares and tools used were owned by the Lahti University of Applied Sciences.

Key words: OLAP, Business Intelligence, ETL

Lahden ammattikorkeakoulu
Tietojenkäsittelyn koulutus

POHJONEN, VILLE:

OLAP-tekniikka
oppilastietojärjestelmässä

Tietojenkäsittelyn koulutuksen opinnäytetyö, 32 sivua

Kevät 2019

TIIVISTELMÄ

Opinnäytetyön toimeksiantaja oli Lahden ammattikorkeakoulun tietohallinto. Opinnäytetyön tavoitteena oli syventää omaa osaamistani tietokantojen, tietovarastoinnin sekä OLAP-ratkaisujen parissa sekä luoda ohjeet OLAP-kuution luomiseen. Opinnäytetyön tavoitteena oli myös, tutkia kuinka OLAP-ratkaisu sopii käytettäväksi oppilastietojärjestelmässä.

Opinnäytteen toteutusosuus toteutettiin käyttäen Microsoft SQL Server Management Studio 17(SSMS) ja Microsoft Visual Studio 2017 - ohjelmistoja. Tietokantojen muokkaukseen ja luomiseen käytettiin SSMS - ohjelmistoa ja OLAP-kuution luomiseen Visual Studiota. Työssä käytetyt ohjelmistot ja työvälineet olivat Lahden ammattikorkeakoulun omistuksessa.

Asiasanat: OLAP, Business Intelligence, ETL

SISÄLLYSLUETTELO

1	JOHDANTO	4
2	TIETOVARASTOINTI	5
2.1	Tietovaraston määritelmä	5
2.2	Tietovarastoinnin historiaa	6
2.3	Tietovarastointiprosessi	6
2.4	ETL-prosessi	6
2.4.1	Extract	7
2.4.2	Transform	8
2.4.3	Load	8
3	BUSINESS INTELLIGENCE	9
3.1	Business intelligencen historia	9
3.2	Business intelligencen tavoitteet	11
4	OLAP	12
4.1	OLAP:in määrittely	14
4.2	Porautuminen	16
4.3	Moniulotteinen OLAP	17
4.4	Relationaalinen OLAP	17
4.5	Desktop OLAP	17
5	OPINNÄYTETYÖN TOTEUTUS	18
5.1	Opinnäytetyön tavoitteet	18
5.2	Opinnäytetyössä käytetyt ohjelmistot	18
5.2.1	Microsoft Visual Studio 2017	18
5.2.2	Microsoft SQL Server Management Studio 17 (SSMS)	18
5.2.3	Microsoft SQL Server Analysis Services (SSAS)	19
5.3	OLAP-kuutio	19
6	PÄÄTELMÄT	29
	LÄHTEET	31

1 JOHDANTO

Opinnäytetyö toteutettiin Lahden ammattikorkeakoulun tietohallinnolle ja opinnäytetyön tarkoituksena oli tehdä selkeät ohjeet OLAP-kuution luomiseen.

Opinnäytetyössä käytettävä data tuli myös Lahden ammattikorkeakoulun tietohallintolta, dataa oli käytössä kahden taulun verran, joissa rivejä oli yli 500 000. Taulut sisälsivät esimerkiksi kohtia kuten opiskeluoikeusavain, alkamis- ja päättymispäivät kursseille, kunnan, kotimaan, arvosanoja, yksikkö tietoja, koulutuksen laajuuksia sekä paljon muuta.

Kuution luomisessa käytin viittä eri taulua, jotka loin kahdesta jo valmiina olevasta taulusta. Tauluja olivat: koulutusala, joka sisälsi organisaatiokoodin, koulutusalan nimen, yksikön sekä ohjelman.

Laajuus, joka sisälsi rowkeyn, laajuuden, toiminna laajuuden, harjoittelun laajuuden sekä muiden tehtävien laajuuden.

Aika, joka sisälsi opiskeluoikeusavaimen, kurssien suoritus-, hyväksiluku-, alkamis- ja päättymispäivämäärät.

Kunta, joka sisälsi kuntakoodin ja kunnan. Sekä faktataulu, johon on koottu kaikkien edellämainittujen taulujen pääavaimet.

Taulujen luominen ja muokkaaminen tapahtui käyttäen Microsoft SQL Server Management Studiota ja OLAP-kuution luominen tapahtui ja Microsoft Visual Studio 2017 -ohjelmistolla.

Opinnäytetyössä käydään myös läpi aluksi tietovarastointia, joissa kerrotaan tietovarastoinnin historiasta ja itse tietovarastointiprosessista sekä business intelligenceä ja tietenkin myös OLAP-tekniologiaa ja sen eri muotoja.

2 TIETOVARASTOINTI

Tietovarastointi on tekniikka, jolla kerätään dataa esimerkiksi isomman yrityksen monista eri tietokannoista ja yhdistetään ne yhdeksi varastoksi, josta erilaisilla business intelligence -ohjelmistoilla voidaan hakea tietoa nopeasti. Tietovarasto on niinsanotusti business intelligence -ohjelmiston ydin, josta saadaan helposti haettua oikea aikaista tietoa ohjelmistojen avulla.

Tietovarastoon voidaan ladata tietoja monenlaisista tietolähteistä. Tavallisesti tiedot kuitenkin haetaan suoraan yritysten omista operatiivisista järjestelmistä. (Hovi, Hervonen, Koistinen. 2009.)

1.1 Tietovaraston määritelmä

Termille tietovarastointi ei ole "virallista" suomennosta, jota kaikki käyttäisivät, mutta hyvin kuvaavana voidaan pitää William H. (Bill) Inmonin määritelmää.

" a subject-oriented, integrated, time-variant and non-volatile collection of data in support of management's decision-making process". (Inmon 1996. Building the Data Warehouse)

Tietovarastointi on kokoelma dataa/tietoa, jota voidaan käyttää päätöksentekoprosessin tukena. Tietovarastoinnin tarkoitus on siis tarjota nopeasti ja oikea-aikaisesti sekä tietenkin helposti tietoa käyttäjien kyselyihin sekä valmiisiin raportteihin. Näihin edellämainittuihin raportteihin ja kyselyihin käytetään business intelligence -ohjelmistoja.

Tietovarastoista tai tietovarastoinnista käytetään usein myös seuraavia nimiä, Decision Support System (DSS), Executive Information System, Management Information System, Business Intelligence Solution, Analytic Application ja Data warehouse (DW tai DWH).

1.2 Tietovarastoinnin historiaa

Monet lähteet pitävät William H. (Bill) Inmon:ia tietovaraston keksijänä ja kuten ylempänä on mainittu hänen kehittämänsä määrittelemää tietovarastoinnista käytetään yleisesti.

Päätöksentekoa tukevia tietojärjestelmiä (Decision Support System, DSS) on kehitetty jo lähes siitä lähtien, kun ensimmäiset tietokoneet otettiin yrityksissä käyttöön. 1970 -luvulla alettiin kehittää johdon informaatiojärjestelmiä (Executive Information System, EIS) ja 1980 -luvulla kehitettiin jo ryhmäpäätöksenteko järjestelmiä (Group DSS) joita pidetään ensimmäisinä business Intelligence-järjestelminä. (Hovi et al. 2009.)

On-line analytical processing (OLAP) alkoi yleistymään 1990 -luvun loppupuolella ja myös ensimmäiset internet-selain pohjaiset BI-sovellusratkaisut tulivat markkinoille vuosituhannen lopussa.

1.3 Tietovarastointiprosessi

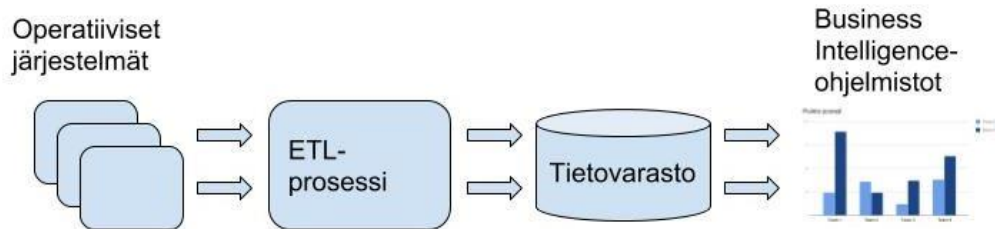
Tietovarastoista on mahdollista eritellä tietovaraston eri osa-alueita, omiksi paikallisvarastoiksi, joista käytetään nimitystä data mart. Yleensä data martit siis sisältävät vain yhden osa-alueen tiedosta, joka voi olla esimerkiksi tarkoitettu pelkästään tietylle osalle yrityksen työntekijöistä.

1.4 ETL-prosessi

ETL(Extract-Transform-Load), ETL-prosessilla kuvataan tietovarastointiprosessia, jossa tietoa tai dataa haetaan yhdestä tai useammista eri järjestelmiä ja siirretään uuteen tietovarastoon. Tietoa muokataan ja yhden mukaistetaan ennen tietovarastoon viemistä.

ETL-prosessia varten on kehitetty paljon erilaisia ominaisuuksia sisältäviä työkaluja, joiden avulla dataa voidaan siirtää isoja määriä, vaikka automatisoidusti. Kaikki ETL-prosessit eroavat yleensä vähintäänkin jollain

tavalla toisistaan, riippuen datan laadusta, määrästä sekä esimerkiksi käytettävistä ohjelmista ja myös riippuen mihin tarpeisiin ja miten informaatiota halutaan hyödyntää.



(Kuva 1. ETL-prosessi, mukaillen (Hovi, A. Hervonen, H. Koistinen, H. 2009.)

Esimerkki kuvassa kuvattuna ETL-prosessi, jossa ensin tiedot haetaan yrityksen tietokannoista tai järjestelmistä, sen jälkeen tietoa muokataan ja laitetaan yhden muokaiseen muotoon. Esimerkiksi joissain tietokannassa voidaan vuodet ja kuukaudet olla merkitty Y/M ja toisessa tietokannassa V/K, tällaiset tiedot on yhdenmukaistettava ETL-prosessissa. Tietoa on siis tarkoitus jalostaa kyselyitä sekä raportteja paremmin palvelevaan muotoon. (Hovi et al. 2009.)

Tämän jälkeen tiedot siirretään tietovarastoon ja sieltä loppukäyttäjien käsiin business Intelligence- ohjelmistoon.

1.4.1 Extract

Extract vaiheessa tietoja haetaan yrityksen eri tietokannoista, tietoa tai dataa voidaan oikeastaan hakea mistä vaan missä sitä yrityksessä sijaitsee, tiedon ei tarvitse olla suoraan tietokannassa.

Tietoa voidaan hakea esimerkiksi laskentaohjelmista, tietokannoista, yrityksen web-sivuilta, osoitekirjoista/yhteystiedoista, erilaisista henkilörekistereistä.

1.4.2 Transform

Transform vaihe on usein ETL-prosessin eniten työtä vaativin osa-alue. Transform vaiheessa eri lähteistä haettu tieto/data puhdistetaan ja yhden mukaistetaan. Tietotyyppejä muutetaan yhdenmukaiseen muotoon, NULL-arvoja poistetaan, päällekkäisyydet poistetaan ja kuten edellä aikasemmin tekstissä mainittiin, muokataan myös tietokantojen koodeja yhdenmukaiseen muotoon.

Kun datan muokkaamisvaiheessa dataa voi tulla paljon ja useasta eri lähteestä on oltava tarkkana, ettei tietokantaan tule duplikaatteja eli päällekkäisyyksiä, joissa sama tieto esiintyy moneen kertaan samanlaisena tai eri muodoissa.

Jos tietovarastoon jää datan muokkamisenkin jälkeen eriävyyksiä sekä duplikaatioita, vaikuttaa kaikki tämä Load vaiheen onnistumiseen sekä myöhemmin tietokannasta saataviin raporteihin sekä kyselyihin.

1.4.3 Load

Load vaiheessa tiedot siirretään yrityksen tietokantaan tai tietovarastoon. Load vaihe on yleensä ETL-prosessin nopein vaihe, Load vaiheessa pitää tarkistaa, että kaikki tieto on yhdenmukaistettu ja muokattu sopivaan muotoon aikasemmissä vaiheissa.

Tarkistuksen jälkeen tiedolle määritellään sijainnit eli pähkinänkuoressa kerrotaan mihin paikkaan mikäkin tieto halutaan.

2 BUSINESS INTELLIGENCE

Business intelligence -termille ei ole virallista suomenkielistä käännöstä, mutta yleisimmin käytetty termi on liiketoimintatiedon hallinta. Business intelligence -termiä käytetään usein, kun puhutaan tietovarastoinnin käyttäjäosuudesta, mutta myös termejä kuten yritystiedon rikastus, analyttinen tiedon hallinta, tiedon hallinta prosessi ja liiketoimintatiedon hallinta käytetään kuvaamaan business intelligenceä.

Kun tietovarastot ovat it-osaajien osaamisaluetta niin business intelligence ohjelmia hyödyntävät työssään esimerkiksi yrityksen liiketoiminnasta vastaavat henkilöt. Business intelligence -ohjelmien päätarkoitus on yleensä auttaa yrityksiä päätösten teossa ja liiketoiminnan suunnittelemisessa.

Monimuotoisista tiedoista muokataan uutta informaatiota, joka antaa enemmän tietoa päätösten tekoon ja auttaa menestymään peremmin kiristyvässä kilpailussa. Kaikille menetelmille on kuitenkin ominaista, että saadaan tuotettua oikea tieto oikeille henkilöille oikea-aikaisesti. (Hovi et al. 2009.)

Business intelligence -termin yhteydessä käytetään myös usein termejä Market intelligence ja Competitive intelligence. Näitä termejä käytetään, kun analysoidaan tietoja, joita on yrityksen kilpailijoista sekä markkinoista saatavilla, kun taas business intelligence -termiä käytetään, kun analysoidaan yrityksensisäistä tietoa yrityksen omista järjestelmistä.

2.1 Business intelligencen historia

Ensimmäisen kerran business intelligence on mainittu jo vuonna 1865, Richar Millar Devensin julkaisemassa kirjassa "Cyclopedia of Commercial and Business Anecdotes". Devens, käytti kirjassa termiä kuvatakseen kuinka pankkiiri Henry Furnese hyötyi ja teki voittoa tiedoilla, joita hän

keräsi kilpailijoistaan. (Devens. 1868. Cyclopedia of Commercial and Business Anecdotes)

Seuraavan kerran business intelligence mainittiin vuonna 1958 IBM-yhtiössä työskennelleen tietojärjestelmätietelijän Hans Peter Luhnin julkaisemassa artikkelissa. Luhn kuvaili artikkelissaan, joilla business intelligencen avulla voitaisiin kerätä tietoa teknologian avulla, analysoida tietoja ja muuttaa se hyödylliseksi informaatioksi ja käyttää sitä hyväksi kilpailussa, aivan kuten nykypäivänä. (Luhn. 1958. A Business Intelligence system)

Päätöksentekoa tukevia tietojärjestelmiä (Decision Support Systems, DSS) on rakennettu lähes siitä lähtien, kun ensimmäiset tietokoneet otettiin yritysten hyötykäyttöön. (Hovi et al. 2009.)

Business intelligence -ohjelmistoja alkoi ilmestyä enemmän Roomassa vuonna 1988 pidetyn ”The Multiway of Data Analysis Consortium” nimisen kokouksen jälkeen, jossa kerrottiin paljon business intelligencen hyödyistä analyyseissa ja myös, kuinka siitä voidaan tehdä käyttäjäystävällisempää. Tähän aikaan business intelligencen tietojen analysoinnissa oli kaksi pääpainopistettä, tiedon ja raporttien tuottaminen sekä sen organisointi ja visualisointi esitettävään ja helposti luettavaan muotoon.

Nykytulkinnan mukaan business intelligencen historia alkaa vuodesta 1989, kun Howard Dresner alkoi kuvata tiedon keräämis ja analysointi menetelmiä termillä business intelligence. (Dresner. 1989. Business Intelligence)

1990-luvun lopulta alkaen business intelligence järjestelmät ovat vain yleistyneet ja ne tarjoat koko ajan kehittyneempiä työkaluja, jotta käyttäjät/päätöksentekijät voivat olla itsenäisempiä ja tehokkaampia päätöstenteossa.

1990-luvulla SQL-kieli ja relaatiotietokannat yleistyivät kovaa vahtia ja saapui markkinoille ensimmäisiä liiketoimintaa tukevia tietovarastointiratkaisuja. Ensimmäisillä business intelligence ohjelmistoilla pyrittiinkin tekemään SQL-kyselyiden tekemisestä helpompaa yritystoiminnassa. (Hovi et al. 2009.)

2010 -luvulla suurimmat markkinaosuudet business intelligence -markkinoista omistavat yritykset kuten Microsoft, Oracle, IBM ja SAP.

2.2 Business intelligencen tavoitteet

Business intelligencen päätavoite on parantaa ja nopeuttaa yritysten ja päätöksentekijöiden kykyä tehdä päätöksiä. Tarkoitus on tuottaa käyttäjälleen mahdollisimman yksityiskohtaista ja oikea-aikaista informaatiota päätöksenteon tueksi.

Muita tavoitteita ovat vastata käyttäjien tietotarpeisiin oikea-aikaisesti, tukea organisaation/yrityksen strategiaa ja tavoitteisiin pääsyä, parantaa käyttäjien omatoimisuutta tietotarpeiden suhteen sekä vähentää kustannuksia ja parantaa operatiivista tehokkuutta. (Hovi et al. 2009.)

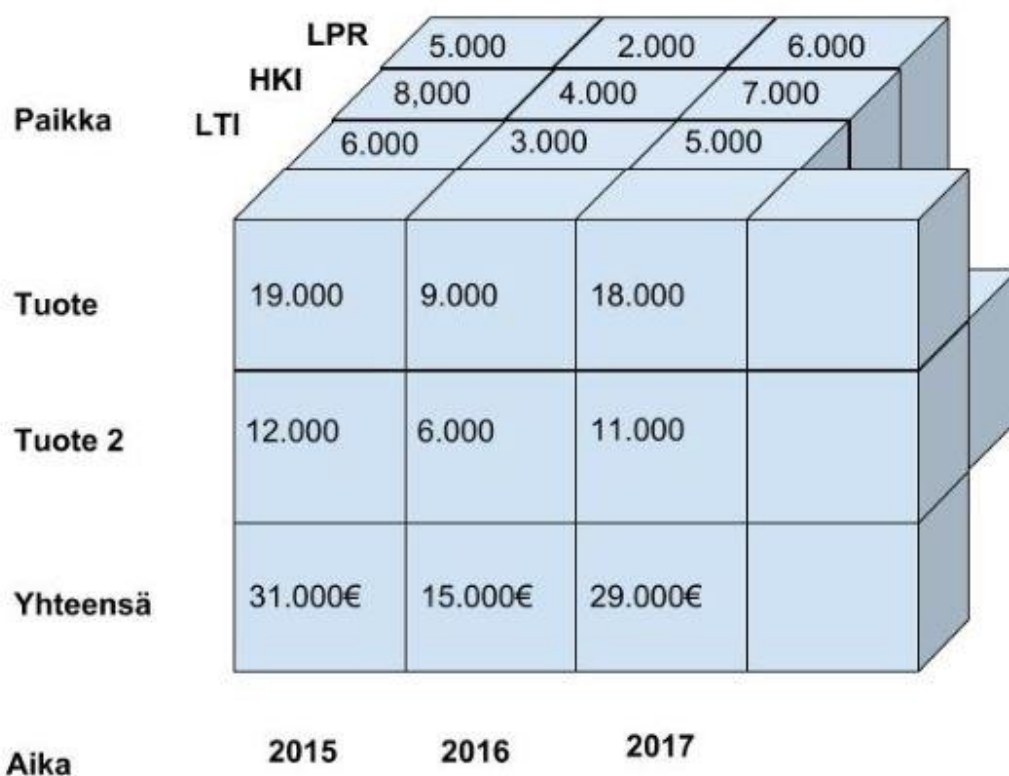
3 OLAP

Moniulotteinen analysointi eli On-line Analytical Processing (OLAP) on eräänlainen business intelligence-ratkaisu, jolla voidaan tarjota moniulotteinen näkymä erilaisista tietokannoista tai tietovarastoista tietojen tarkempaa analysoimista varten. OLAP-malli on suunniteltu käyttäen pohjana relaatiomallia sekä relaatiotietokantoja. (Hovi et al. 2009.)

OLAP-ratkaisulla on monia hyötyjä verrattuna muihin tietokantaratkaisuihin, joista ehkä suurimpia ovat sen nopeus ja käytön helppous ja tietojen luettavuus. OLAP-ratkaisuissa hyödynnetään hierarkioita sekä tietojen summaamista eli aggregointia, joka tarkoittaa esimerkiksi, että yrityksen tietokannassa voi olla suoraan luettavissa tieto yrityksen päivän myynnistä ilman erillistä kyselyä tietokannasta koska tiedot tai luvut ovat jo valmiiksi summattu yhteen OLAP- kuution sisällä.

Tärkeä asia OLAP-tietokantaratkaisuissa on, että käyttäjä pystyy analysoimaan tietoa suoraan ilman kyseilyä tietokannasta, eli niin sanotusti "On-line". OLAP-tietokannoissa tieto mallinnetaan usein kuution muotoon, joka on yleinen tapa kuvata OLAP-tietokannan sisältöä ja rakennetta. OLAP-tietokannassa on eri ulottuvuuksia joita voivat esimerkiksi olla aika, paikka, tuote yms.

Ulottuvuuksien arvot lasketaan yhteen jo tietokannan sisällä ja edellä mainituilla ulottuvuuksilla voidaan nähdä vaikka esimerkiksi tietyn tuotteen myyntimäärät Lahden alueella viimeisen kuukauden ajalta automaattisesti ilman kyseilyitä.



(Kuva 2. OLAP-kuutio esimerkki.)

Niin kuin aikasemmin tekstissä mainitsin, OLAP mallinnetaan usein kuution muotoon. Esimerkki kuvassa on kolme eri dimensiota eli ulottuvuutta, paikka, tuote ja aika. OLAP-kuutio laskee luvut yhteen jo tietokannan sisällä ilman erillistä kyselyä ja kuten esimerkki kuvasta näkyy, niin kuutiosta voidaan suoraan nähdä edellä mainituilla dimensioilla esimerkiksi, että kuinka paljon tuotetta numero yksi on myyty vuoden 2015 -aikana yhteensä kaikilla kuutiosta olevilla paikkakunnilla.

Tietoja tai informaatiota ei kuitenkaan yleensä lueta suoraan kuutiosta vaan se siirretään business Intelligence -ohjelmistoon, jonka avulla voidaan datasta tai informaatiosta tehdä erilaisia taulukkoja, raportteja sekä kyseilyitä.

Eri ulottuvuuksia OLAP-tietokannoissa on yleensä 3-8 kappaletta yhdessä kuutiossa, yli kymmenen ulottuvuuden tietokannat ovat yleensä hallinnaltaan haastavia. (Hovi et al. 2009.)

3.1 OLAP:in määrittely

Tietojenkäsittelytieteen tutkija Edgar F. Codd, joka loi relatiotietokantamallin 1970-luvulla on kehittänyt myös termin OLAP. Vuonna 1993 E.F Codd määritteli yhdessä S.B Codd ja C.T Salley:n kanssa OLAP-termille kirjassaan "Providing OLAP to User-Analyst: An IT Mandate" 12 sääntöä, joiden pitäisi toteutua OLAP-ratkaisussa.

Alla 12 sääntöä määriteltynä (Codd, Codd, Salley, 1993, Providing OLAP to User-Analyst: An IT Mandate)

1. Moniulotteinen, käsitteellinen näkymä tietoihin (**Multidimensional Conceptual View**)

Tiedot pitää pystyä esittämään moniulotteisena mallina, jossa tiedot on nimetty loogisesti/helposti ymmärrettävästi.

2. Läpinäkyvyys (**Transparency**)

Itse OLAP-kuutio pitää pysyä loppukäyttäjälle "näkymättömänä" eli käyttäjän ei tarvitse puuttua tiedon alkuperään tai kuutioon vaan voi käyttää pelkästään business Intelligence -ohjelmistoa.

3. Tietojen saatavuus (**Accessibility**)

Loppukäyttäjälle pitää pystyä esittämään tieto käyttäjälle sopivassa/halutussa muodossa.

4. Raportoinnin tasainen suorituskyky (**Consistent Reporting Performance**)

OLAP:in suorituskyky ei saa heikentyä kun dimensioiden/ulottuvuuksien määrää kasvatetaan.

5. Palvelintekniikka (**Client-Server Architecture**)

OLAP:in on oltava joustava ja siihen pitää olla mahdollista kytkeytyä erilaisilla työasematyökaluilla.

6. Yhtenäinen moniulotteisuus (**Generic Dimensionality**)

Ulottuvuuksien/dimensioiden tulee olla ominaisuuksiltaan yhdenmukaisia ja niihin kohdistettävien toimenpiteiden pitää toimia kaikissa ulottuvuuksissa samalla tavalla.

7. Matriisin dynaaminen käsittely (**Dynamic Sparse Matrix Handling**)

Tyhjät solut eivät saa huonontaa tietokannan tehokkutta tai vähentää sen tilaa.

8. Samanaikaisten käyttäjien tuki (**Multi-User Support**)

OLAP-ratkaisun pitää tukea samanaikaisesti useita eri käyttäjiä.

9. Rajoittamattomat dimensioiden väliset operaatiot (**Unrestricted Cross-dimensional Operations**)

Loppukäyttäjällä pitää olla mahdollisuus määritellä omia laskukaavuja sekä kyseilyitä, mutta OLAP-ratkaisun pitää tarjota myös valmiita vaihtoehtoja.

10. Joustava ja ymmärrettävä käyttöliittymä (**Intuitive Data Manipulation**)

OLAP-ratkaisun käyttäminen pitää olla yksinkertaista ja tiedonhaun helppoa loppukäyttäjälle.

11. Joustavuus (**Flexible Reporting**)

Loppukäyttäjällä pitää olla mahdollisuus siirrellä sarakkeita ja rivejä haluamaansa järjestykseen, jotta loppukäyttäjä voi analysida tietoja haluamallaan tavalla.

12. Rajoittamaton määrä ulottuvuuksia sekä koostetasoja (**Unlimited Dimensions and Aggregation Levels**)

OLAP-ratkaisun pitää mahdollistaa vähintään 15 dimensiota/ulottuvuutta. Jokaisessa ulottuvuudessa pitää voida olla rajoittamaton määrä koostetasoja.

3.2 Porautuminen

OLAP-ratkaisussa loppukäyttäjä voi halutessaan ”porautua” syvemmälle kuutioon tarkkempaa tietoa etsiessä. Esimerkiksi käyttäjä voi hakea tarkemmin vaikka pelkästään yhden tuotteen myyntilukuja, kuten kuvan 2. esimerkissä,

”Porautumisen tarkoituksena on siis selvittää ylemmän tason luvun muodostuminen yksityiskohtaisemmasta tasosta”

(Hovi et al. 2009.)

Loppukäyttäjä voi halutessaan porautua kumpaankin suuntaan tahansa, eli esimerkiksi yhdestä tuotteesta tai tasosta käyttäjä voi porautua ylöspäin katsomaan koko tuoteperheen myyntiä tai samalla tavalla toisin päin eli

käyttäjä voi porautua koko, tuoteperheestä tai ryhmästä vaikka yhden tuotteen myyntilukuihin.

3.3 Moniulotteinen OLAP

MOLAP (multidimensional OLAP), MOLAP-ratkaisussa tiedot tallennetaan suoraan moniulotteisiin taulukoihin. MOLAP-ratkaisussa jokaista taulukon solua kohtaan on jonkinlainen mitta-attribuutinarvo eikä tyhjiä arvoja ole.

Useasti MOLAP-ratkaisut ovat tuotekohtaisia ratkaisuja, jotka perustuvat tietokantoihin. Kuutiot ovat yleensä tiedostoja, joissa tiedot on jo valmiiksi laskettu, joka mahdollistaa nopeat vasteajat. (Hovi et al. 2009.)

3.4 Relationaalinen OLAP

ROLAP-ratkaisuissa tiedot tallennetaan tietovaraston relaatiotauluihin. ROLAP-ratkaisut tukevat normaalia suurempia tietomääriä, mutta myös siitä johtuen kyselyiden ja raporttien vasteajat ovat hitaita verrattuna muihin OLAP-ratkaisuihin.

ROLAP-ratkaisuihin tehdään useasti summatauluja hakujen nopeuttamiseksi. (Hovi et al. 2009.)

3.5 Desktop OLAP

DOLAP-ratkaisussa käytetään myös moniulotteista analysointia, mutta tällaisessa ratkaisussa kaikki tieto on tallennettuna paikallisesti käyttäjän tietokoneen muistiin. Tätä ratkaisua tulen käyttämään opinnäytetyössäni, koska kaikki data on minulla tallennettu paikallisesti työtietokoneelleni, näin myös voin muokata tietoa vapaammin ja tehdä erilaisia kokeilua, joutumatta huolehtimaan siitä, että jotain häviää tai sekoittuu järjestelmässä pysyvästi. Olimme opinnäytetyön toimenkiantajan kanssa samaa mieltä asiasta, että tämä on paras ratkaisu työn toteutukseen.

4 OPINNÄYTETYÖN TOTEUTUS

4.1 Opinnäytetyön tavoitteet

Opinnäytetyön tavoitteena on syventää omaa SQL-osaamistani, sekä syventää osaamistani työssä käytettävissä ohjelmistoissa. Tavoitteena on myös tiettenkin omalta osaltani oppia rakentamaan OLAP-kuutio sekä oppia tuntemaan koko prosessin kaikki vaiheet.

Opinnäytetyön tavoitteena toimeksiantajan puolesta on luoda OLAP-kuutio perustuen oppilastietokantohin ja tehdä selkeät ohjeet OLAP-kuution luomiseen. Tavoitteena on myös tutkia kuinka OLAP-kuutio -ratkaisu soveltuu oppilastietojärjestelmiin ja miten sitä voidaan hyödyntää tällaisessa ympäristössä.

4.2 Opinnäytetyössä käytetyt ohjelmistot

4.2.1 Microsoft Visual Studio 2017

Microsoft Visual Studio on Microsoftin kehittämä ohjelmakehitysympäristö. Visual studio tukee ohjelmointikieliä kuten C++, Visual Basic, C# ja F#. Visual Studioon on mahdollista lisätä eri valmistajien lisäominaisuuksia ja Visual Studiolla pystyy esimerkiksi tehdä erilaisia web- ja mobiilisovelluksia. Visual Studio 2017 on ohjelmiston uusin versio ja se julkaistiin 7. maaliskuuta 2017.

Opinnäytetyössäni käytän Visual Studiota OLAP-tietokantakuution luomiseen käyttäen Microsoft SQL Server Analysis Services (SASS) lisäosaa, joka on lisäosa Microsoft Server Management Studio 17 ohjelmistoon, mutta sitä käytetään Visual Studion kautta.

4.2.2 Microsoft SQL Server Management Studio 17 (SSMS)

Microsoft SQL Server Management Studio on ohjelmisto, jota käytetään Microsoft SQL server -komponenttien hallintaan ja konfigurointiin.

SSMS:ssä on graafinen käyttöliittymä, jolla voi hallita ja tehdä muutoksia tietokantoihin, mutta käyttäjä pystyy tekemään muutoksia ja kyseilyä tietokantoihin myös MYSQL-kielellä. SSMS:n ensimmäinen versio julkaistiin vuonna 2005 ja versio 2017 on julkaistu 2. lokakuuta 2017.

Opinnäytetyössäni käytän Microsoft SQL Server Management Studio 17 -ohjelmistoa tietokantojen hallintaan sekä niiden muokkamiseen.

4.2.3 Microsoft SQL Server Analysis Services (SSAS)

Microsoft SQL Server Management Studio sisältää paljon tietovarastointiin ja business Intelligenceen liittyviä työkaluja kuten, Integration Services (SSIS), Reporting Services (SSRS) ja Microsoft SQL Server Analysis Services (SSAS)

Microsoft SQL Server Analysis Services on OLAP ja datanlouhintatyökalu, joka toimii SSMS:n lisäosana. SSAS:ssin avulla voidaan rakentaa OLAP-tietokanta kuutio ja sen avulla yritykset pystyvät analysoimaan ja järjestelemään dataa mahdollisesti monista eri tietokannoista tai tiedostoista.

Opinnäytetyössäni käytän Microsoft SQL Server Analysis Services -ohjelmistoa OLAP-kuution luomiseen ja kuten edellä mainittu SSAS:sia käytetään Visual Studion kautta.

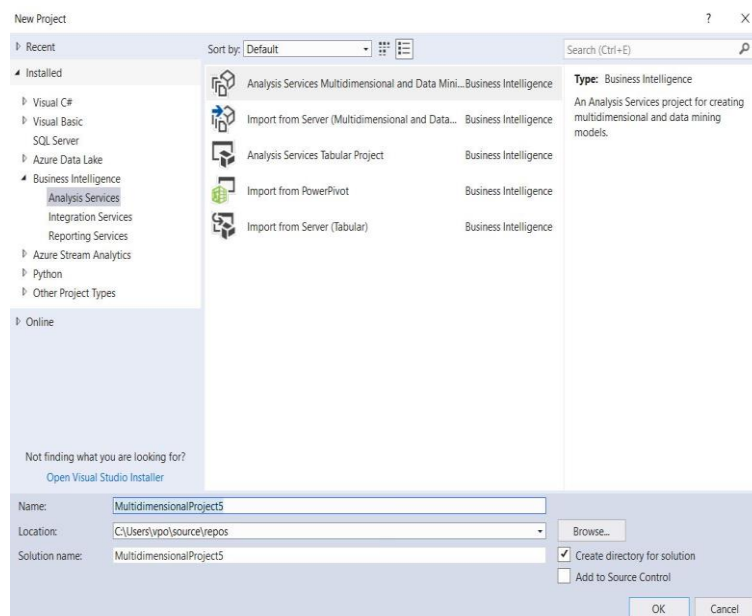
4.3 OLAP-kuutio

Kuten aikaisemmin tekstissä mainittiin, niin OLAP-kuution rakentamisen ensimmäinen vaihe on ETL-prosessi. Kun ETL-prosessi on tehty ja tiedot ovat oikeassa paikassa, halutussa muodossa niin voidaan alkaa rakentaa OLAP-kuutiota.

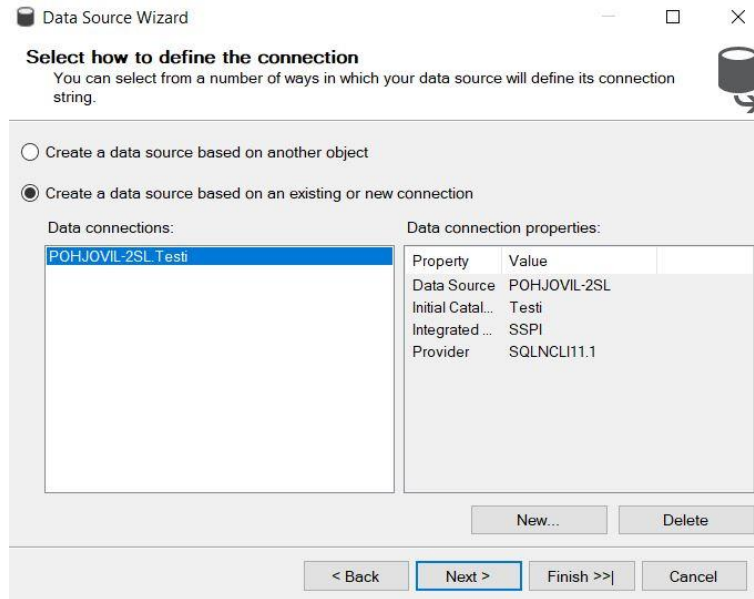
OLAP-kuution luominen tapahtuu Visual Studio –ohjelmistolla ja tässä osiossa käydään tarkemmin läpi, kuinka OLAP-kuution luominen Visual Studiolla tapahtuu.

Ennen kuution rakentamista pitää olla siis ETL-prosessi tehty, jotta kuutio voidaan rakentaa, sekä pitää olla asennettuna tarvittavat ohjelmistot Visual Studio sekä SSMS, jonka kautta tietokantoja hallitaan. Microsoft SQL Server Management Studio (SSMS) ja lisäosina pitää myös asentaa Microsoft SQL Server Analysis Services (SSAS).

OLAP-kuutio siis luodaan Visual Studiolla, joten ensimmäiseksi Visual Studiosta avataan uusi projekti (New Project) ja valitaan Business Intelligence –kohdan alta ”Analysis Services” ja sen alta ”Analysis Services Multidimensional and Data Mining Project”. Tämän jälkeen projektille valitaan nimi ja tallennus-sijainti ja sen jälkeen ”OK”.



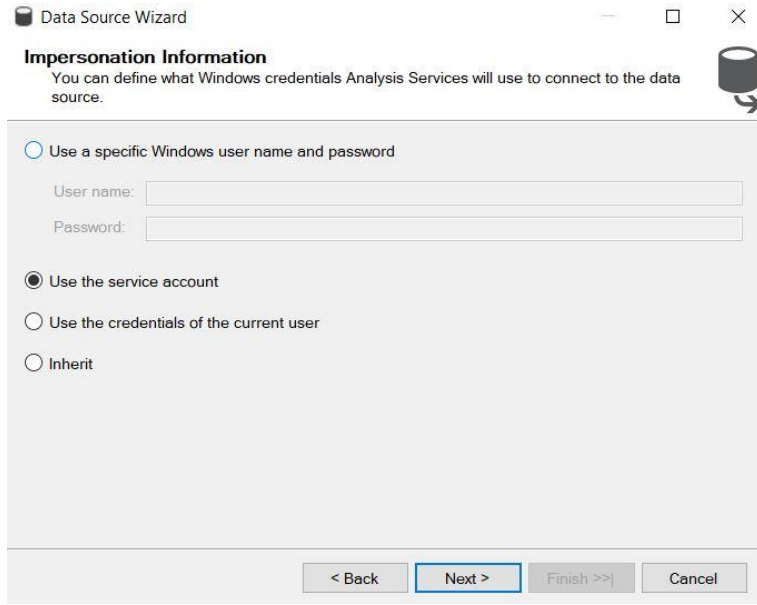
Seuraava vaihe on Datalähteen määrittäminen. Kun projekti on auennut, klikataan oikeasta yläkulmasta hiiren oikealla painikkeella ”Data Sources” kansion kuvaa ja valitaan ”New Data Source”. Tämän jälkeen määritetään yhteys tietokantoihin, esimerkissä käytän jo valmiiksi luotua yhteyttä, mutta jos yhteyttä ei ole määritelty niin voidaan uusi yhteys määritellä valitsemalla ”Creat a data source based on an existing or new connection” ja ”New” ja tämän jälkeen annetaan SQL-palvelimen nimi johon, yhteys halutaan muodostaa.



Jos et tiedä tai muista SQL- palvelimen nimeä niin alaspäinnoolta painamalla nimikentästä Visual Studio etsii automaattisesti palvelimia, joihin yhteys on mahdollista muodostaa, mutta jos oikeaa palvelinta ei löydy, voit tarkistaa sen SSMS:stä.

Sitten valitaan kirjautumistapa (Windows tai SQL –todennus) ja sitten valitaan tietokannan nimi, johon yhteys halutaan muodostaa. (Tässäkin vaiheessa alaspäinnoolella Visual Studio tarjoaa automaattisesti tietokantoja, joita aikasemmin valittu palvelin sisältää.) Tämän jälkeen käyttäjä voi vasemmasta alakulmasta painaa ”Test Connection” ja näin testata toimiiko yhteys. Kun yhteys toimii valitaan ”Next”.

Seuraavaksi valitaan käyttäjätiedot joilla kirjaudutaan datalähteeseen. Eri malleissa/esimerkeissa ja OLAP-kuutio ratkaisuihin käytetään eri vaihtoehtoja joten tähän ei ole niisanotusti oikeaa vastausta, että mikä kohta olisi paras valita. Tässä ohjeessa kuitenkin käytin vaihtoehtoa ”Use the service account”.



Data Source Wizard

Impersonation Information
You can define what Windows credentials Analysis Services will use to connect to the data source.

Use a specific Windows user name and password

User name:

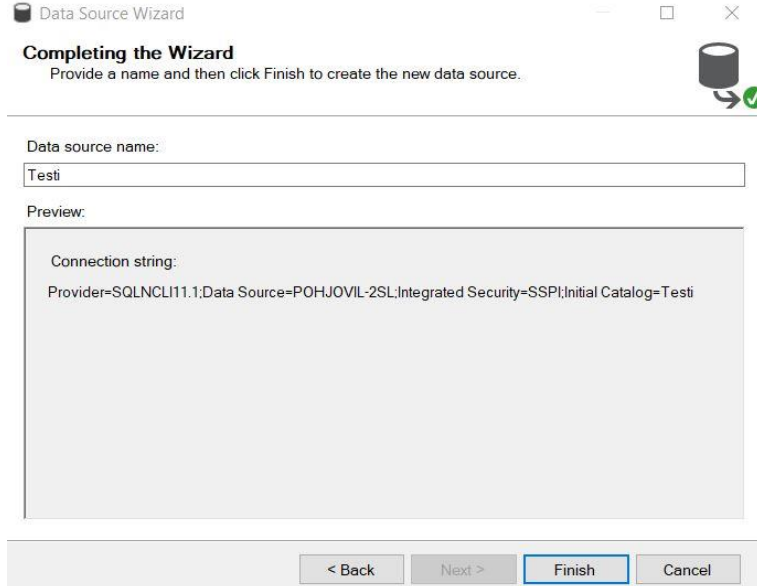
Password:

Use the service account

Use the credentials of the current user

Inherit

< Back Next > Finish >> Cancel



Data Source Wizard

Completing the Wizard
Provide a name and then click Finish to create the new data source.

Data source name:

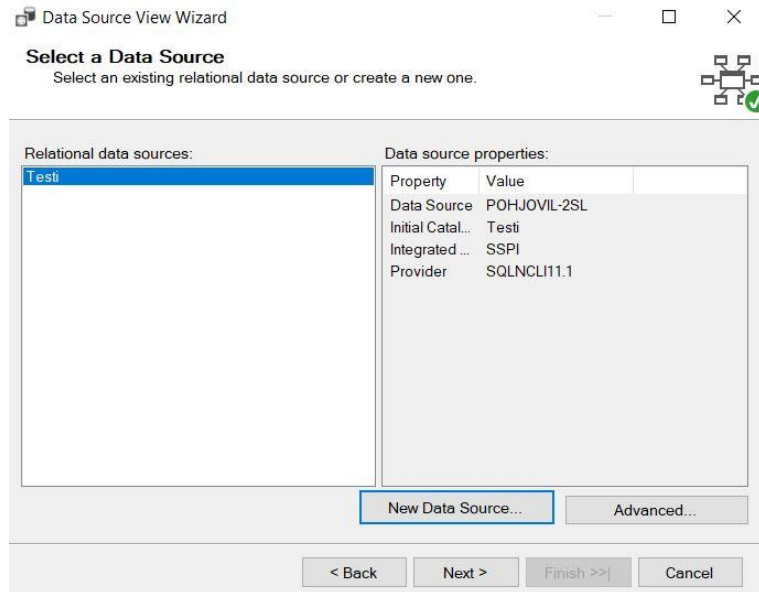
Preview:

Connection string:
Provider=SQLNCLI11.1;Data Source=POHJOVIL-2SL;Integrated Security=SSPI;Initial Catalog=Testi

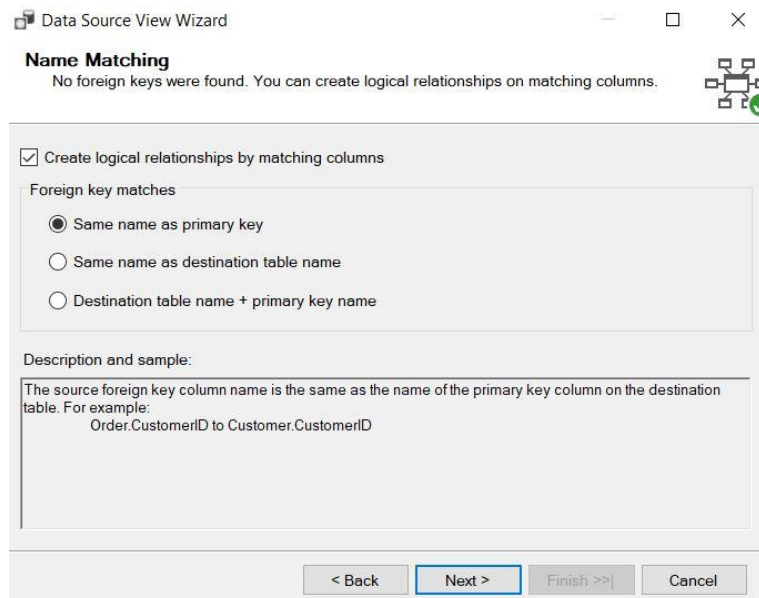
< Back Next > Finish Cancel

Tämän jälkeen valitaan nimi datalähteelle ja painetaan "Finish".

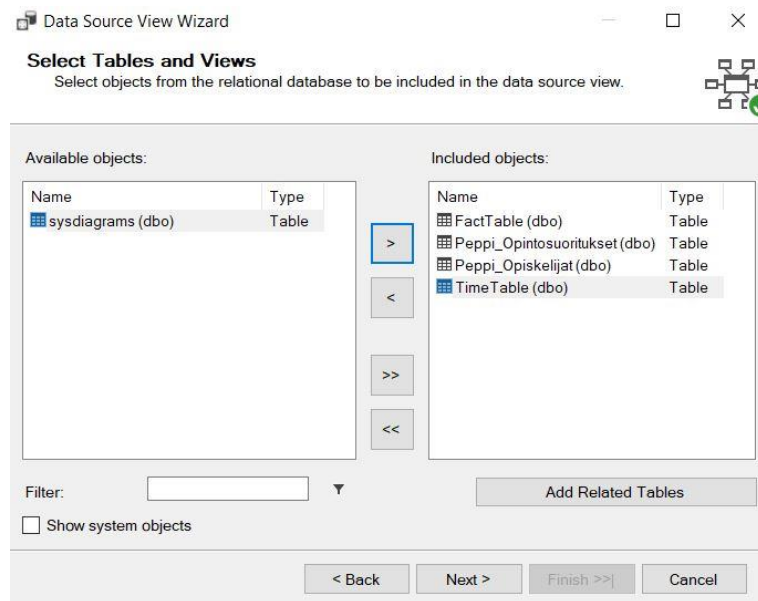
Seuraavaksi valitaan oikealta valikosta "Data Source Views" ja hiiren oikealla klikkaamalla "New data source view". Data source view kohdassa määrittellään, että mistä tietokannasta tietoja otetaan ja mitä tauluja halutaan käyttää. Ensin valitaan tietokanta joista tietoja halutaan.



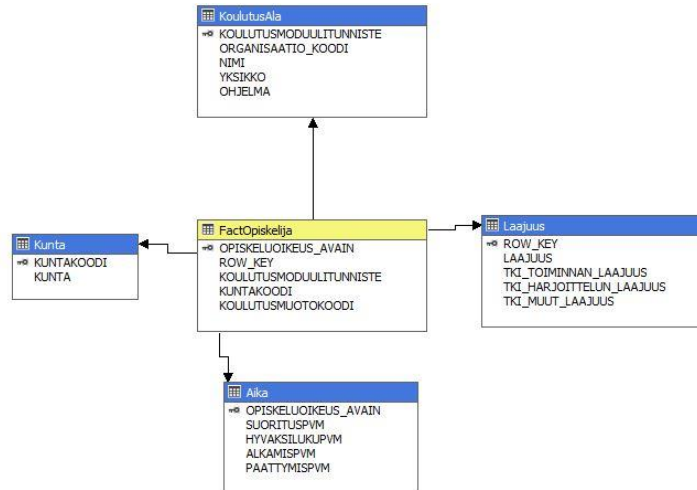
Jos et ole luonut yhteyksiä taulujen välille aikasemmin, voit antaa Visual Studion yrittää automaattisesti luoda loogisia yhteyksiä taulujen välillä eri vaihtoehdoin, tosin omien kokeilujeni perusteella Visual Studio ei luonut yhteyksiä vaikka niin valitsin, joten yhteydet taulujen välillä olisi hyvä määrittää jo ennakkoon ETL-prosessi vaiheessa.



Tämän jälkeen valitataan taulut joita OLAP-ratkaisussa halutaan käyttää. Kun olet valinnut yhden taulun voit klikata kohtaa "Add Related Tables" joka lisää automaattisesti kaikki taulut joilla on yhteyksiä jo valittuun tauluun. Seuraavaksi painetaan "Next" ja sitten nimetään datalähde ja vahvistetaan että käyttäjä on valinnut oikeat taulut.



Data source view muodostaa tauluista ER-kaavion, jota käyttäjä voi muokata, lisäämällä tauluja tai esimerkiksi luomalla uusia yhteyksiä taulujen välillä tai vaihtaa pääavaimia/viiteavaimia. Jos käyttäjä luo uusia pää tai –viiteavaimia niin on hyvä muistaa, että pää ja –viiteavaimien välille pitää muodostaa yhteys.



Tämän jälkeen luodaan itse kuutio, painamalla hiiren oikealla kohtaa "Cubes" ja sen jälkeen dropdown -valikosta "New Cube". Seuraavaksi valitaan, että halutaanko kuutiossa käyttää valmiita tauluja, luodaanko kuutio tyhjänä tai annetaanko Visual Studion yrittää luoda tauluja perustuen data lähteeseen, tosin tämä toimenpide ei yleensä toimi hirveän hyvin ja suosittelenkin käyttämään kuution luomisessa valmiita tauluja, jotka on jo aiemmin luotu SSMS:llä.

Cube Wizard

Select Creation Method
Cubes can be created by using existing tables, creating an empty cube, or generating tables in the data source.

How would you like to create the cube?

Use existing tables

Create an empty cube

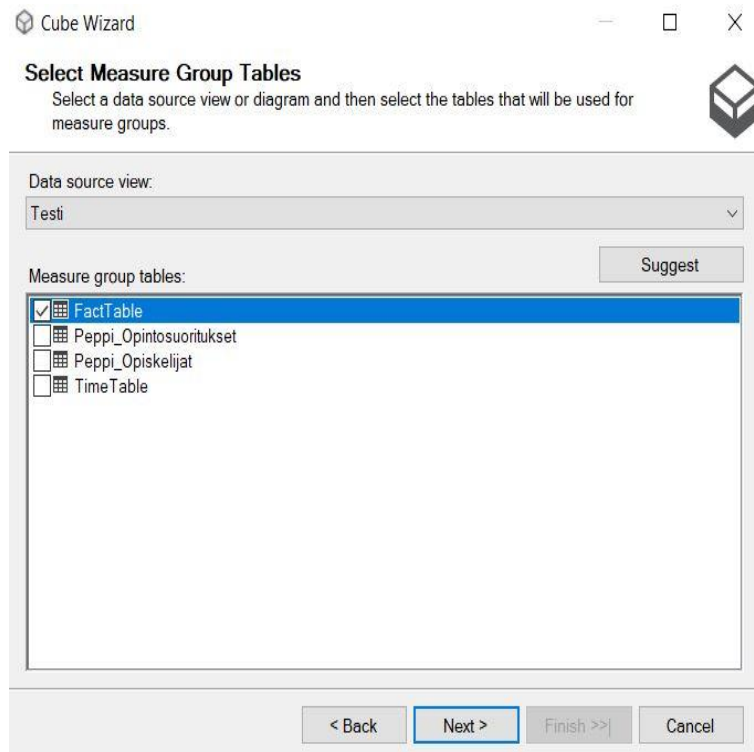
Generate tables in the data source

Template:
(None)

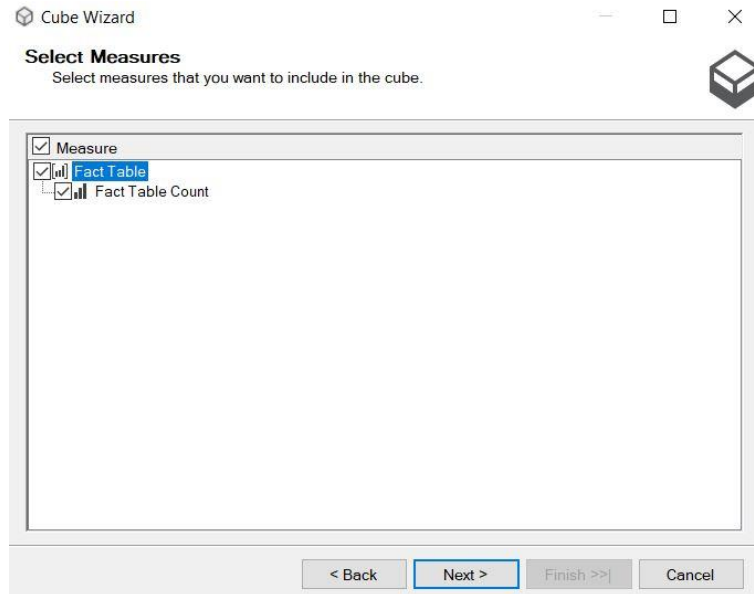
Description:
Create a cube based on one or more tables in a data source.

< Back Next > Finish >>| Cancel

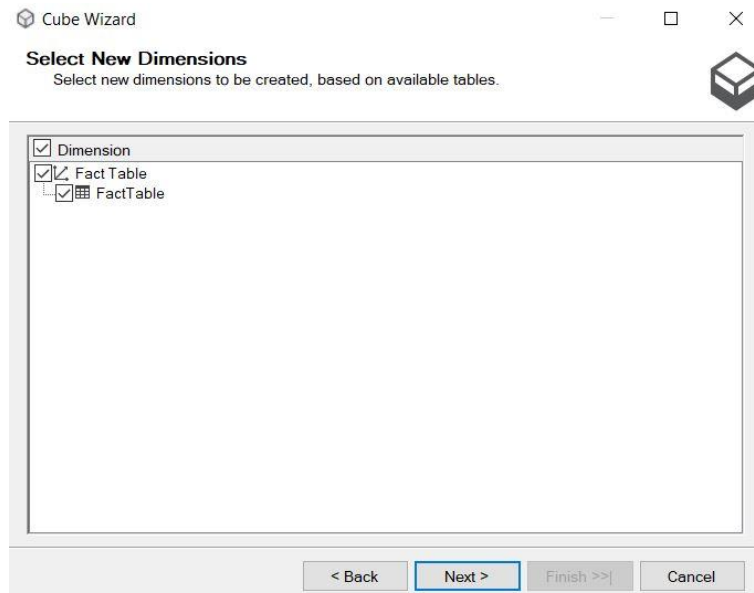
Seuraavaksi valitaan niin kutsuttu faktataulu, joka sisältää yleensä muitten taulujen pääavaimet. Jos olet luonut kuution jo aikasemmin luotujen taulujen perusteella, voit klikata ”Suggest” kohdasta ja Visual Studio ehdottaa oikeaa taulua. Taulun ehdotus toimii tosin vain, jos olet luonut taulujen väliset yhteydet jo ennen tätä ETL-prosessi vaiheessa.



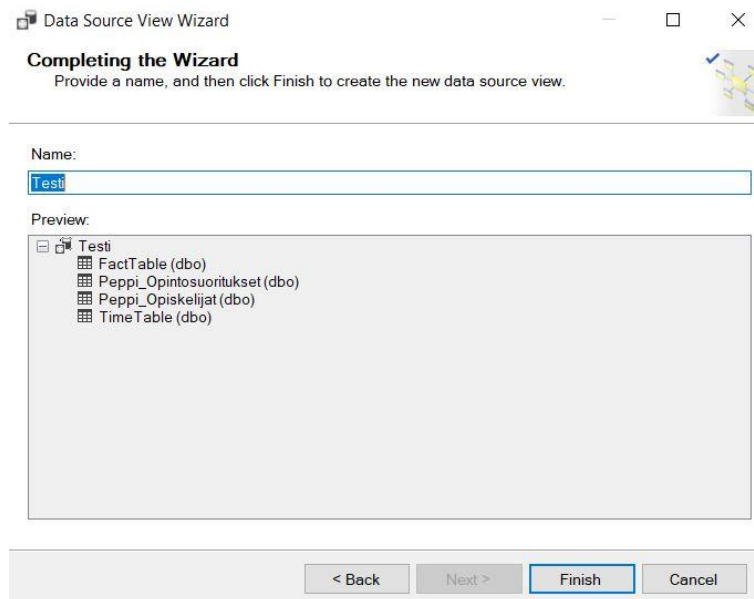
Sitten valitaan mitä kohtia faktataulusta halutaan ottaa kuutioon mukaan. Visual Studio ehdottaa tässä kohtaa jo yleensä oikeat valinnat eikä niitä tarvitse muuttaa.



Tämän jälkeen määritellään kuution dimensiot. Ja kuten aikasemmin työssä on kerrottu, kuutiossa on yleensä noin 3-8 dimensiota ja niitä olisi hyvä olla vähintään tuo kolme kappaletta. Ohjeen kuvassa ei näy kuin yksi dimensio, mutta omassa kuutio ratkaisussani dimensiona olivat: koulutusala, koulutuksen laajuus, aika sekä kunta.



Seuraavaksi näet mitä tauluja kuutio pitää sisällään ja voit nimetä kuution. Tämän jälkeen kuutio on lähes valmis.



Kun olet painanut "Finish" niin valitse Visual Studion ylävalikosta kohta "Build" ja valitse valikosta kohta "Build Solution", voit tehdä tämän myös näppäinyhdistelmällä Ctrl+Shift+B. Tämän jälkeen valitse vielä uudestaan kohta "Build" ja sen alta kohta "Deploy". Jos kaikki on niin kuin pitääkin niin Visual Studio ilmoittaa että, "Deployment succesfull" joka meinaa että, kuution luominen on onnistunut.

Kun kuutio on luotu onnistuneesti, niin se ilmestyy myös SQL Server Management Studioon ja voit halutessasi käyttää ja analysoida kuutiota myös sitä kautta. Visual Studiossa kuution onnistuneen luomisen jälkeen aukeaa uusi välilehti, jossa on nyt valmis OLAP-kuutio.

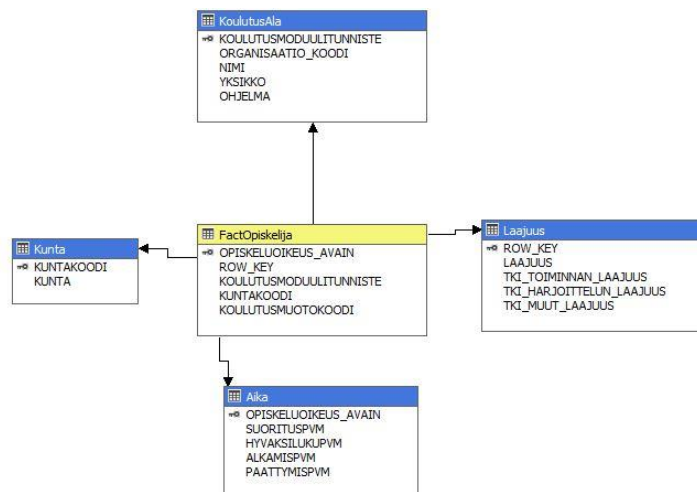
Käyttäjä voi tämänkin jälkeen vielä muokata kuutiota lisäämällä esimerkiksi uusia dimensiota tai muuttamalla hierarkioita. Käyttäjä voi esimerkiksi lisätä KPI:tä (Key Performance Indicators).

Kohdasta "Browser", joka löytyy myös SSMS:stä käyttäjä voi tarkastella kuutiota ja tehdä kyselyjä ja luoda esimerkiksi Pivot-taulukoita.

5 PÄÄTELMÄT

Suurimman osan ajasta opinnäytetyössäni käytin ETL-prosessiin ja kohtasinkin siinä paljon vaikeuksia, koska data ei ollut hirveän yhdenmukaista ja siitä piti muokata erilaisia tauluja, joissa data olisi samanlaista taulujen sisällä. Paljon vaikeuksia tuotti myös ETL-prosessissa pohdinta, että mitä tietoja OLAP-kuutioon kannattasi sisällyttää niin, että siitä saataisiin aikaseksi järkevä kokonaisuus, jota voitaisiin oppilastietojärjestelmässä hyödyntää.

Juuri tämä edellä mainittu kohta, jossa pohdin mitä dataa käytetään ja missä tauluissa vie mielestäni todella paljon aikaa. Lopulta päätin käyttää viittä taulua, jotka näkyvät tässä ER-kaaviossa.



OLAP-kuution luomisessa tämä edellä mainittu ETL-prosessi tuotti paljon vaikeuksia juurikin datan yhdenmukaistamisessa ja vasta kymmenien ja kymmenien yritysten jälkeen sain kuution luotua. Onnistuin kuitenkin luomaan kuution vain tyhjänä enkä ole tähän ongelmaan vielä löytänyt ratkaisua. Ongelmat luultavasti johtuvat ETL-prosessissa tehdyistä virheistä, kun dataa ei saatu yhdenmukaiseksi kaikilta osin. Kun taulut siirrettiin Visual Studioon ja kuutiota alettiin rakentamaan niin, jostain syystä tauluista hävisi aina dataa ja jokin sarake oli aina vain "NULL" ja näin ollen kuutiossa ei pystynyt tekemään kyselyjä tai laskelmia.

Asiaa konsultoin muutamalta tutuiltani, jotka työskentelevät tietokantojen parissa sekä tietenkin vastuuopettajaltani ja opinnäytetyöntilaajalta Lahden ammattikorkeakoulun tietohallinnosta, sekä myös tietenkin vietin erilaisten ohjeiden ja Googlen parissa todella paljon aikaa, mutta ongelmaan ei vain löytynyt ratkaisua.

Mielestäni OLAP-tietokantakuutio ei ole ehkä paras ratkaisu oppilastietojärjestelmään, koska ensinnäkin tieto/data ei ole usein sellaista, joista olisi helppoa kuutiota rakentaa, kuten esimerkiksi kaupan alalla se voisi olla paljon helpompaa, kun vaikka tuotteiden myyntiä olisi helppo seurata dimensiolla aika, paikka, tuote, josta siten selviäsi suoraan miten paljon ja missä mitäkin tuotetta on myyty. Oppilastietojärjestelmässä jouduin miettimään todella paljon, että mitä asioita voisi toisiinsa verrata, että niitä olisi järkevää lähteä rakentamaan kuutioksi.

Toisekseen datan pitäisi olla jo alusta alkaen yhdenmukaista, että siitä olisi helppo tehdä OLAP-kuutioratkaisuja ja luultavasti tämä vaatisi aika paljon muutoksia oppilastietojärjestelmiin, mutta jos järjestelmää, tietovarastoja sekä datamartteja ruvettaisiin rakentamaan alusta alkaen uusiksi ja jo tässä kohtaa pohdittaisiin OLAP-kuution mukaan ottamista tiedon analysoinnissa, niin kuutioden rakentaminen onnistuisi varmasti paljon helpommin.

OLAP-tietokantakuutio voisi siis olla hyvä ratkaisu oppilastietojärjestelmässä jos tieto/data olisi saatavilla yhdenmukaisessa muodossa ja järjestelmänylläpitäjät tai raportteja järjestelmästä tarvitsevat henkilöt määrittelisivät mitä asioita OLAP-kuutiosta tehtävän raportin pitäisi sisältää.

LÄHTEET

Hovi, A. Hervonen, H. Koistinen, H. 2009. Tietovarastot ja business intelligence WSOY, Jyväskylä.

E.F Codd, S.B Codd, C.T Salley. 1993. Providing OLAP to User-Analyst: An IT Mandate [viitattu14.10.2018]. Saatavissa:

http://www.arborsoft.com/essbase/wht_ppr/coddTOC.html

Elmasri, R. Navathe. S.B. 2007. Fundamentals of database systems. Boston [viitattu31.9.2018]. Saatavissa:

<http://iips.icci.edu.iq/images/exam/databases-ramaz.pdf>

Howard Dresner. 1989. Business Intelligence [viitattu15.9.2018].

Saatavissa:

<http://census.com.ar/download/Data%20Mining%20Scholars%20program.pdf>

HP Luhn. 1958. A Business Intelligence system [viitattu16.10.2018].

Saatavissa: <https://ieeexplore.ieee.org/abstract/document/5392644>

Keith D, Foote. 2017. A Brief History of Business Intelligence

[viitattu20.9.2018]. Saatavissa: <http://www.dataversity.net/brief-history-business-intelligence/>

OLAP Council. 1997. OLAP Council White Paper [viitattu16.10.2018].

Saatavissa: <http://www.olapcouncil.org/research/whtpapply.htm>

RM Devens. 1868. Cyclopedia of Commercial and Business Anecdotes [viitattu16.10.2018]. Saatavissa:

https://books.google.fi/books?hl=fi&lr=&id=vqBDAAAIAAJ&oi=fnd&pg=PR36&dq=Cyclopedia+of+Commercial+and+Business+Anecdotes&ots=E3ixhmUx7a&sig=jJH8GfWXW_MNNj3EoJPKk26XDOg&redir_esc=y#v=onepage&q=Cyclopedia%20of%20Commercial%20and%20Business%20Anecdotes&f=false

Surajit Chaudhuri. Umeshwar Dayal. 1997. An Overview of Data Warehousing and OLAP Technology [viitattu31.9.2018]. Saatavissa: <http://www.cs.sfu.ca/CourseCentral/459/han/papers/chaudhuri97.pdf>

William H. Inmon 1996. Building the Data Warehouse [viitattu14.9.2018]. Saatavissa: <http://fit.hcmute.edu.vn/Resources/Docs/SubDomain/fit/ThayTuan/DataWH/Bulding%20the%20Data%20Warehouse%204%20Edition.pdf>