

SVEUČILIŠTE U SPLITU  
FAKULTET ELEKTROTEHNIKE, STROJARSTVA I BRODOGRADNJE

*Kolegij: Skladišta podataka*

SEMINARSKI RAD

DW - AIRPORT

|  |  |
| --- | --- |
| Student: | Meri Lovrić |

# Sadržaj

[Sadržaj 2](#_Toc42281179)

[1. Sažetak i ključne riječi 1](#_Toc42281180)

[1.1. Sažetak 1](#_Toc42281181)

[1.2. Ključne riječi 1](#_Toc42281182)

[2. Uvod 1](#_Toc42281183)

[2.1. Osnovni pojmovi 1](#_Toc42281184)

[2.2. Značaj skladišta podataka 1](#_Toc42281185)

[2.3. Ideja projekta 1](#_Toc42281186)

[3. Transakcijska baza 4](#_Toc42281187)

[3.1. Relacijska baza podataka 4](#_Toc42281188)

[3.2. Dijagram transakcijske baze 4](#_Toc42281189)

[4. Logički model skladišta podataka 6](#_Toc42281190)

[4.1. Dimenzijski model 6](#_Toc42281191)

[4.1.1. Tablica činjenica 8](#_Toc42281192)

[4.1.2. Tablice dimenzija 8](#_Toc42281193)

[5. Punjenje skladišta podacima 9](#_Toc42281194)

[5.1. SQL Server Data Flow 9](#_Toc42281195)

[5.1.1. Connection manager 10](#_Toc42281196)

[5.1.2. Data Flow Task (DFT) za punjenje tablica dimenzija 12](#_Toc42281197)

[5.1.3. Definiranje pravila za dimenzijske tablice 12](#_Toc42281198)

[5.1.4. Definiranje pravila za popunjavanje tablice činjenica 14](#_Toc42281199)

[6. Izgradnja kocke 18](#_Toc42281200)

[6.1. Kreiranje SQL Server Analysis Services (SASS) projekta 18](#_Toc42281201)

[6.2. Definiranje pogleda 19](#_Toc42281202)

[6.3. Kreiranje kocke 21](#_Toc42281203)

[6.4. Dodavanje atributa i kreiranje hijerarhija u dimenzijama 26](#_Toc42281204)

[6.4.1. Razvijanje (deploy) kocke 27](#_Toc42281205)

[7. Izvješća (Cube Player) 29](#_Toc42281206)

[7.1. Spajanje na SQL Server Analysis Services Server 29](#_Toc42281207)

[7.2. Kreiranje izvješća 30](#_Toc42281208)

[8. Zaključak 34](#_Toc42281209)

[9. Prilozi 35](#_Toc42281210)

[9.1. Kazalo slika 35](#_Toc42281211)

[9.2. Kazalo upita 36](#_Toc42281212)

[9.3. Popis oznaka i kratica 36](#_Toc42281213)

[10. Literatura 37](#_Toc42281214)

# Sažetak i ključne riječi

## Sažetak

U ovom seminarskom radu detaljno će se opisati uloga skladištenja podataka za potrebe praćenja prometa zrakoplovnih luka te će se isto pokazati na konkretnim podacima implementiranim u projektu. Kako bi što bolje razumjeli potrebu za skladištenjem podataka i funkcioniranje izvještaja kao početak dan je kratak pregled značenja skladišta podataka i svih povezanih pojmova.

Daljnja poglavlja odnose se na konkretnu realizaciju projekta kroz niz sekvencijalnih koraka koji su se realizirali u sklopu laboratorijskih vježbi. Za početak je izvedena transakcijska baza podataka čije relacije predstavljaju logički model podataka preko kojih su odabrani podaci koji se žele vremenski pratiti i opisano je zašto i na koji način će se podaci prevoditi iz relacijskog modela u jednostavnije skladište podataka.

Prijelaz podataka iz relacijskog modela u logički model podataka ostvaren je preko novije inačice BIDS softvera, *SQL Server Integration Services* alata. Temelj prijelaza je odabir dimenzija koje želimo pratiti i njihovo prevođenje uz pomoć softvera. Uz odabir dimenzija, potrebno je napraviti tablicu činjenica koja predstavlja sponu dimenzija. Uloga dimenzija i tablice činjenica bit će detaljnije objašnjena u daljnjim poglavljima.

Idući korak je stvaranje modela kocke pomoću SSAS softvera kojim se realizira logički model kocke koja će se potom koristiti za izradu izvještaja u *Cube Player* softveru. Izrada izvještaja je srž projekta jer u izvještajima prezentiramo podatke koje smo odabrali kao dimenzije i mjere poslovnog procesa te ih prezentiramo korisniku na pregledan i sistematičan način. Izvještaj je okosnica poslovnog procesa i služi za kvalitetnu statističku analizu i praćenje prometa.

## Ključne riječi

Ključne riječi čije poznavanje je potrebno za razumijevanje projekta, a koje će biti objašnjene kroz projekt su:

* Skladište podataka
* Transakcijska baza podataka
* Tablica činjenica
* Tablica dimenzija
* Kocka

# Uvod

Prvi dio dokumentacije bavi se teoretskom osnovom skladišta podataka čije je razumijevanje potrebno za shvaćanje ideje projekta i važnosti korištenja skladišta podataka za praćenje poslovnog procesa.

## Osnovni pojmovi

Osnovni pojam koji će se prvo objasniti je skladište podataka. Skladište je podataka posebna vrsta baze podataka za skupljanje podataka od više različitih izvora koja omogućava korisnicima jednostavnije izvršavanje inače kompleksnih upita o podacima. Uglavnom se koristi u poslovnoj inteligenciji gdje se podaci iz prodaje, marketinga i raznih drugih odjela neke tvrtke skupljaju kako bi omogućili donošenje strateških odluka na temelju prethodnih podataka. Temeljna razlika u odnosu na klasičnu bazu podataka je manji broj veza između entiteta/dimenzija jer je fokus na analizi podataka, ne *real-time* ažuriranju. Podatke prema kojima želimo raditi izvještaje grupiramo u tablice koje nazivamo tablice dimenzija. Činjenice koje proizlaze iz mjerljivih podataka grupiramo u tablici činjenica i nazivamo mjerama.

## Značaj skladišta podataka

Ovime se postiže lakše izvršavanje inače kompleksnih upita u bazu i stvaranje izvještaja. Izvještaji se, naravno, mogu generirati i iz robusne relacijske baze, ali su onda potrebni puno kompliciraniji upiti koji se sastoje od pretraživanja i spajanja mnogih tablica kako bi se došlo do osnovnih informacija o poslovnom procesu. Samim time što se rabe složeniji upiti i generiranje zahtjeva više vremena što može biti osobiti problem ukoliko je u pitanju velika baza podataka.

Zbog navedenih razloga preporuča se izrada i korištenje skladišta podataka kako bi se izbjegli kompleksni upiti i upravljanje ogromnom količinom podataka na način da se iz transakcijske baze prebace samo podaci koji će služiti za izvještaje.

## Ideja projekta

Osnovna ideja seminarskog rada je napraviti skladište podataka za praćenje zrakoplovnog prometa. Struktura rješenja funkcioniranja skladišta podataka dana je na slici 2‑1. Početak realizacije je stvaranje transakcijske baze koja će sadržavati sve podatke koji će se potom prebaciti u dimenzije skladišta podataka. Dimenzijski podaci prenose se na model kocke iz kojeg se generiraju izvješća. Korištene tehnologije su: MS SQL server 2014 za transakcijsku bazu i skladište, SQL Server Integration Services (SSIS) za prijenos i punjenje skladišta podacima te SQL Server Analysis Services i *Cube Player* za analizu podataka.



Korisnik

Transakcijska baza

Skladište podataka

Analiza podataka

Slika 2‑1 Idejna izvedba skladišta podataka

Primjena i korisnost skladišta podataka za praćenje prometa proizlazi iz korištenja skladišta kao centralnog mjesta za generiranje izvještaja nad raznovrsnim i robusnim podacima kakvi se generiraju iz poslovnih procesa zrakoplovnog prometa. Takve podatke bi bilo teško i nepraktično na sistematičan način predstaviti korisniku iz transakcijske baze.

Primjeri poslovnih procesa koji će se pratiti:

* Prodaja karata kroz vrijeme po kategorijama
  + Kategorije se mogu odnositi na putnika – dob, nacionalnost, itd.
  + Kategorije se mogu odnositi na prodavača – dob, nacionalnost, radno mjesto, itd.
  + Kategorije se mogu odnositi na zrakoplovnu agenciju
  + Kategorije se mogu odnositi na zrakoplovnu luku
* Frekventnost letova kroz vrijeme
* Frekventnost prometa po pojedinim zrakoplovnim lukama
* Prodaja u odnosu na cijenu karata
* Popunjenost pojedinih letova i sl.

Po dobroj razredi poslovnih procesa koji će se pratiti i pravilno odabranih podataka prelazi se na dizajn skladišta podataka. Dizajnu skladišta podataka prethodi implementacija klasične transakcijske baze čija je struktura objašnjena u idućem poglavlju.

# Transakcijska baza

Ovo poglavlje odnosi se na predstavljanje izgleda i implementacije transakcijske baze te analize njene uloge u kontekstu projekta.

## Relacijska baza podataka

Relacijske se baze u osnovi sastoje od entiteta i relacija među entitetima. Entiteti predstavljaju objekte ili koncepte informacija koje želimo zapisati. Informacije o objektima zovemo atributima entiteta. Više entiteta povezano je relacijom koja ima dva bitna svojstva: kardinalnost i obvezu članstva. Kardinalnost govori u kakvom su omjeru entiteti, gdje postoje tri vrste kardinalnost: 1-1 (jedan na jedan), 1-n (jedan na više) i n-m (više na više). Obveza članstva odnosi se na postojanje veze jer u nekim slučajevima logika rada baze može funkcionirati i bez jednog entiteta u relaciji. U konkretnom primjeru ovog projekta sve relacije imaju obavezu postojanja. Relacijska baza će poslužiti kao transakcijska baza iz koje će se povlačiti podaci u skladište.

## Dijagram transakcijske baze

Izgled baze podataka dan je na slici 3‑1, izdvojeni entiteti su:

* dbo.Aircraft: informacije o pojedinom modelu zrakoplova – kapacitet zrakoplova (AirportCapacity) i ime agencije koja je vlasnik zrakoplova (AirlineID)
* dbo.Airline: informacije o pojedinoj zrakoplovnoj agenciji – naziv agencije (AirlineName), matična država (AirlineCountry), ukupan broj odrađenih letova (TotalFlights), ukupan prihod od svih letova (TotalRevenue)
* dbo.AirportInstance: informacije o pojedinoj zrakoplovnoj luci – naziv zrakoplovne luke (AirportName), matična država (AirportCountry) i grad (AirportCity) te univerzalni kod od tri znaka za imenovanje luke (AirportCode)
* dbo.Employee: informacije o pojedinom zaposleniku – ime (EmployeeName), prezime (EmployeeSurname), spol (EmployeeGender) te rođendan (EmployeeBirthday) zaposlenika i mjesto zaposlenja tj. matična zrakoplovna luka (AirportID)
* dbo.Flight: informacije o pojedinom letu – odredišna zrakoplovna luka (FlightDestinationAirport), model zrakoplova (AircraftID), popunjenost leta (FlightVacancy), cijena leta (FlightPrice), broj prodanih karata (FlightTicketsSold), zarada od prodanih karata (FlightRevenue), vrijeme leta (FlightTime) te izvorišna zrakoplovna luka (FlightSourceAirportID)
* dbo.Luggage: informacije o prtljazi – vlasnik prtljage (PassengerID), vrsta prtljage (LuggageType) te težina prtljage (LuggageWeight)
* dbo.Passenger: informacije o pojedinom putniku – ime (PassengerName), prezime (PassengerSurname), rođendan (PassengerBirthday), spol (PassengerGender) i nacionalnost putnika (PassengerNationality)
* dbo.Ticket: informacije o pojedinoj prodanoj karti – putnik koji je kupio kartu (PassengerID), zaposlenik koji je prodao kartu (EmployeeID), let za koji je kupljena karta (FlightID) te vrijeme leta (TicketDepartureTime)

Uz sve navedene atribute svaki entitet ima i primarni ključ definiran u formatu: **EntityNameID(PK, bigint, not null)**.



Slika 3‑1 Dijagram transakcijske baze

# Logički model skladišta podataka

Ovo se poglavlje bavi opisom logičkog modela skladišta podataka za primjer ovog projekta te će se objasniti značenje i uloga tablice činjenica i tablica dimenzija koje su sastavni dio skladišta.

## Dimenzijski model

Logička shema modela skladišta je zvjezdasta. Model zvijezde je set relacijskih tablica u skladištu podataka koje predstavljaju najjednostavniju višedimenzionalnu shemu. Uz model zvijezde, na predavanjima je spomenut i model pahuljice, ali je za izradu projekta odabran model zvijezde zbog jednostavne strukture, razumljivog modela, jednostavnijih upita i podržanosti od strane velikog broja alata. Ipak, postoji nedostatak modela zvijezde, a to je da je potrebno relativno dugo vrijeme učitavanja podataka u dimenzijske tablice zbog denormalizacije. Za potrebe projekta u dimenzijske tablice je upisan mali broj podataka stoga navedeni nedostatak ne dolazi do izražaja. Izgled konkretnog dimenzijskog modela projekta dan je na slici 4‑1. Dimenzijski model se sastoji od središta koje predstavlja tablica činjenica – *fact table* (FactSale) i 7 dimenzija koje su odabrane kao relevantne za generiranje izvještaja:

* DimDate – dimenzija vremenskih raspona
* DimPassenger – dimenzija s informacijama o pojedinom putniku
* DimAirport – dimenzija s informacijama o pojedinoj zrakoplovnoj luci
* DimFlight – dimenzija s informacija o pojedinom letu
* DimEmployee – dimenzija s informacijama o pojedinom zaposleniku
* DimAirline – dimenzija s informacijama o pojedinoj zrakoplovnoj agenciji
* DimTicket – dimenzija s informacijama o pojedinoj prodanoj karti



Slika 4‑1 Logički model skladišta podataka

### Tablica činjenica

Tablica činjenica je temeljna tablica u DW-u. Sastoji se od poslovnih činjenica koje nazivamo mjerama (*measures*) i stranih ključeva (*foreign keys*) koji predstavljaju identifikaciju u tablicama dimenzija. Mjere su brojčane vrijednosti koje su zanimljive analitičarima za praćenje poslovnog procesa. Za projekt su to cijena leta SaleFlightPrice i automatski generirana Fact Sale Count koja prati sve letove. Svi strani ključevi unutar tablice činjenica zajedno čine složeni primarni ključ tablice činjenica. Uz složeni primarni ključ u tablicu činjenica dodan je i obični primarni ključ SaleKey. Općenito su tablice činjenica puno veće i robusnije od dimenzijskih tablica te mogu činiti do 90% DW-a [1].

### Tablice dimenzija

Tablice dimenzija (*dimension tables*) sadrže podatke o svim dimenzijama te na taj način nadopunjuju kontekst činjenica. Sadrže ključeve dimenzija, vrijednosti i atribute. Podaci o dimenzijama su najčešće opisne vrijednosti koje mogu biti strukturirane u više hijerarhija. Ukoliko dimenzijska tablica nema hijerarhija tada je zovemo ravna dimenzija ili lista. Dimenzije definirane u projektu navedene su u odlomku 4.1, a zajednička dimenzija svim skladištima podataka je dimenzija vremena jer se većina željenih izvještaja odnosi na pregled tokom vremena.

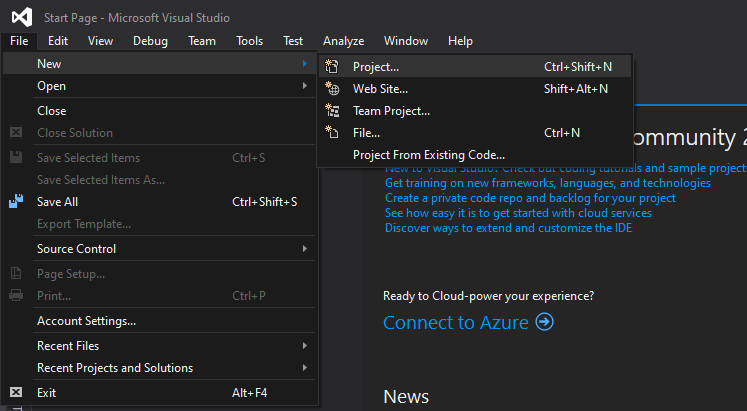
Po izradi logičkog modela skladišta podataka prelazi se na punjenje skladišta konkretnim podacima. Podaci se inicijalno pune odjednom, no taj proces se može i mora periodički ponavljati kako bi podaci bili ažurni u odnosu na poslovni proces. Skladište se puni na automatiziran način korištenjem BIDS (Bussiness Intellignece Development Studio) alata tj. SQL Server Integration Services alata unutar Visual Studio razvojnog okruženja.

# Punjenje skladišta podacima

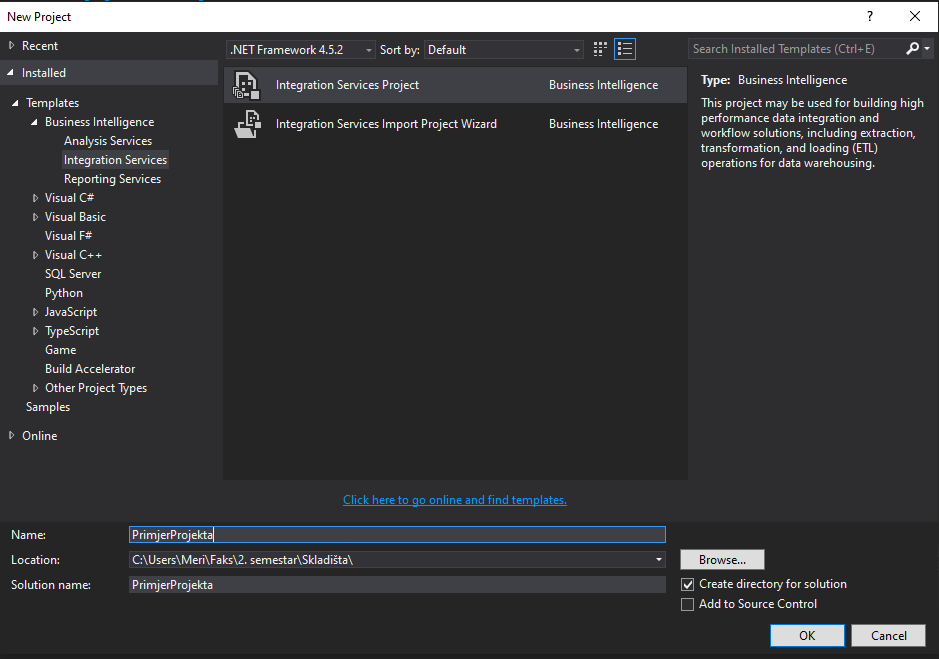
Ovo se poglavlje bavi prikazom procesa punjena skladišta podataka pomoću SQL Server Integration Services alata. SISS povlači podatke iz transakcijske baze i kopira ih na odgovarajuća mjesta unutar skladišta podataka.

## SQL Server Data Flow

SQL Server Dana Flow je naziv za noviju verziju SQL Server Integration Service (SISS). SISS je platforma za integraciju podataka i stvaranje aplikacijskog poslovnog toka. Jedna od značajki je alat za ekstrakciju, transformaciju i učitavanje (*extraction, transformation and loading – ETL*) podataka u skladištu podataka. Alat se također koristi za automatiziranje održavanja SQL Server baze podataka i ažuriranje višedimenzionalnih kocki. Postupak kreiranja novog projekta radi se u nekoliko koraka prikazanih na slici 5‑1 i slici 5‑2. Odabire se *New → Project → Integration Services Project*.



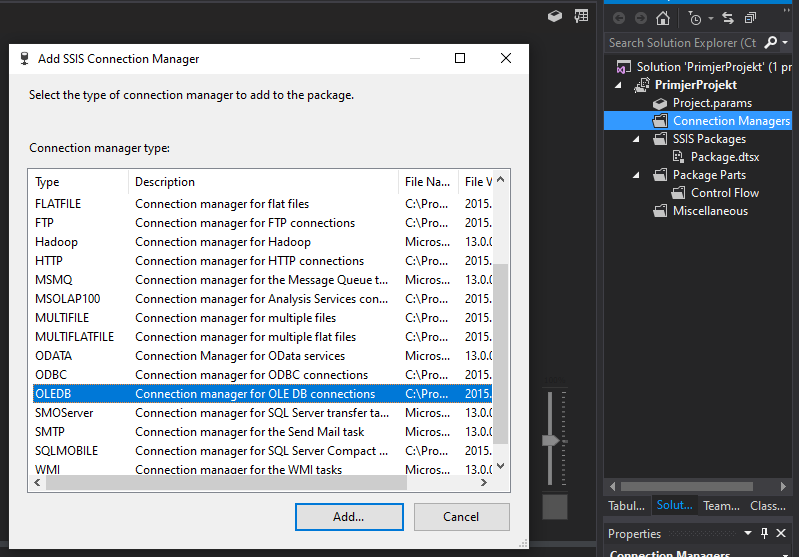
Slika 5‑1 Kreiranje novog projekta



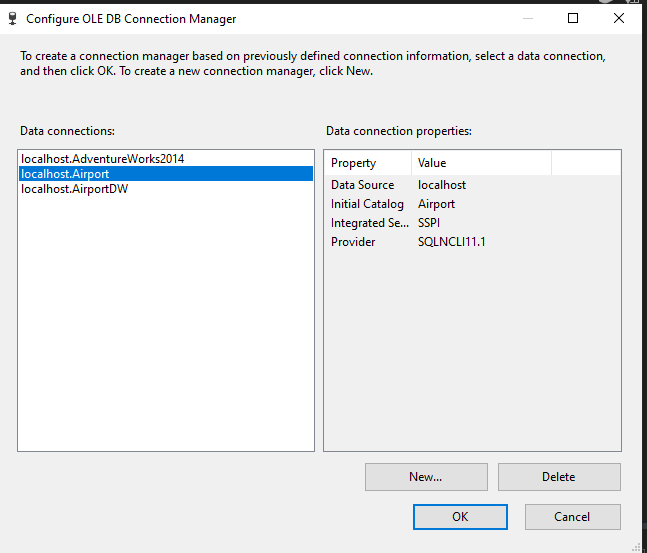
Slika 5‑2 Odabir Integration Services projekta

### Connection manager

Prvi korak je spajanje SQL Server Dana Flow platforme s SQL Server Management Studio. U ovom koraku povezujemo transakcijsku bazu Airport i skladište podataka AirportDW s platformom za integraciju podataka. Kao *Connection Manager* odabiremo OLEDB manager kao prikazano na slici 5‑3.



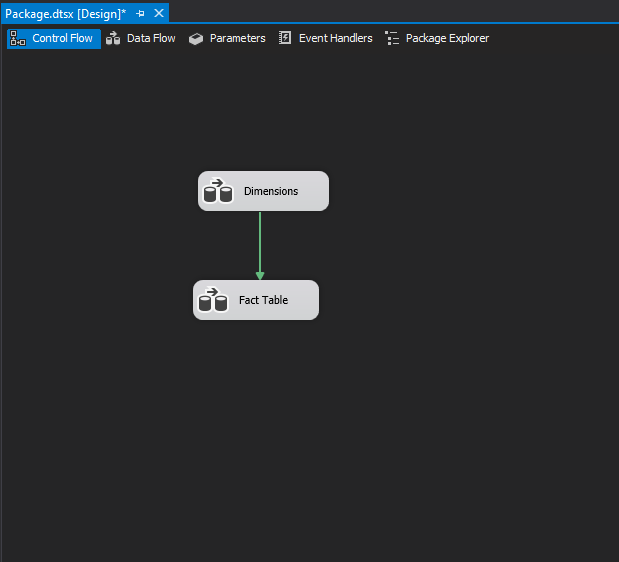
Slika 5‑3 Postavljanje connection managera



Slika 5‑4 Odabir DB i DW

### Data Flow Task (DFT) za punjenje tablica dimenzija

Drugi korak je postavljanje DataFlowTask kontrole unutar *control flow* ploče. Postavljamo dvije DataFlowTask kontrole, jednu za dimenzijske tablice i jednu za tablicu činjenica jer je potrebno poštivati redoslijed punjenja podataka tako da se prvo dimenzijske tablice pune podacima, a potom tablica činjenica jer se ona puni korištenjem nekih podataka iz dimenzija. Kontrole su stoga preimenovane u „Dimensions“ i „Fact Table“.



Slika 5‑5 DataFlowTask za dimenzije i činjenice

### Definiranje pravila za dimenzijske tablice

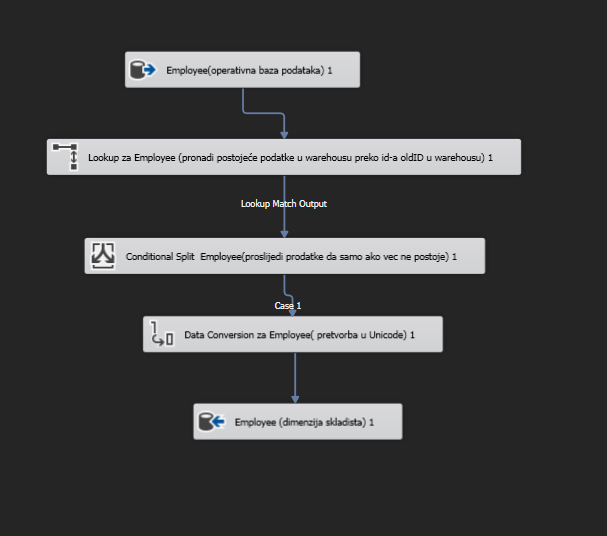
U idućem koraku, kao što je prije navedeno, punimo dimenzijske tablice prije tablice činjenica. Kako bi to napravili postavljamo pravila za dimenzije:

* OLE DB Source – izvor podataka (OLTP baza)
* Lookup – povezivanje PrimaryKey od transakcijske baze s oldID dimenzijske tablice
* Conditional Split – provjera postojanja podataka kako bi se spriječilo kopiranje postojećih podataka tako što ispituje ISNULL(OldID)
* OLED DB Destination. - odredište podataka (DW baza)

Podaci iz OLTP baze se mapiraju u DW tako što ID primarnog ključa transakcijske baze prelazi u OldID dimenzijske tablice, dok ID primarnog ključa dimenzijske tablice ostavljamo praznim te postavljamo na AutoIncrement opciju tako da se pri punjenju dimenzijske tablice automatski generira inkrement ključa.

Sve definirane dimenzijske tablice su jednostavne te podatke povlače iz jedne tablice u transakcijskoj bazi.

Primjer Data Flow Task za punjenje podataka Employee dimenzijske tablice prikazan je na Slika 5‑6.

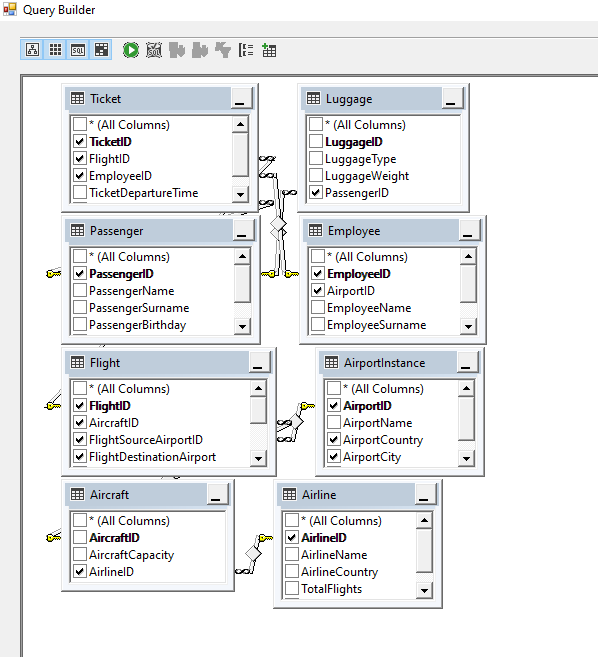


Slika 5‑6 Data Flow Employee dimenzijske tablice

### Definiranje pravila za popunjavanje tablice činjenica

Idući korak odnosi se na definiranje pravila za popunjavanje tablice činjenica. U odnosu na popunjavanje dimenzijskih tablica postupak je sličan, ali nije identičan. Također je bitno naglasiti da ne postoji *one-size fits all* rješenje već je postupak personaliziran za svako skladište podataka. Iznova definiramo izvorište i odredište podataka te pravila toka podataka, ali s izmijenjenom logikom. Neki od podataka potrebnih za popunjavanje tablice činjenica dodaju se u dimenzijske tablice i stoga, kao što je prije naglašeno, korak popunjavanja dimenzija mora prethoditi koraku punjenja tablice činjenica. Komponente toka podataka za popunjavanje tablice činjenica su:

* OLE DB Source – izvor podataka (OLTP baza), ali se ovaj put kao vrsta pristupa odabire *SQL command*, te radimo SQL upit (Slika 5‑7 i Upit 5‑1) za sve tablice čiji podaci sudjeluju u kreiranju tablice činjenica. Ovdje je bitno napomenuti korištenje CONVERT naredbe koja je korištena kako bi vremenske podatke pretvorili u *bigint* oblik s kojim je lakše manipulirati.

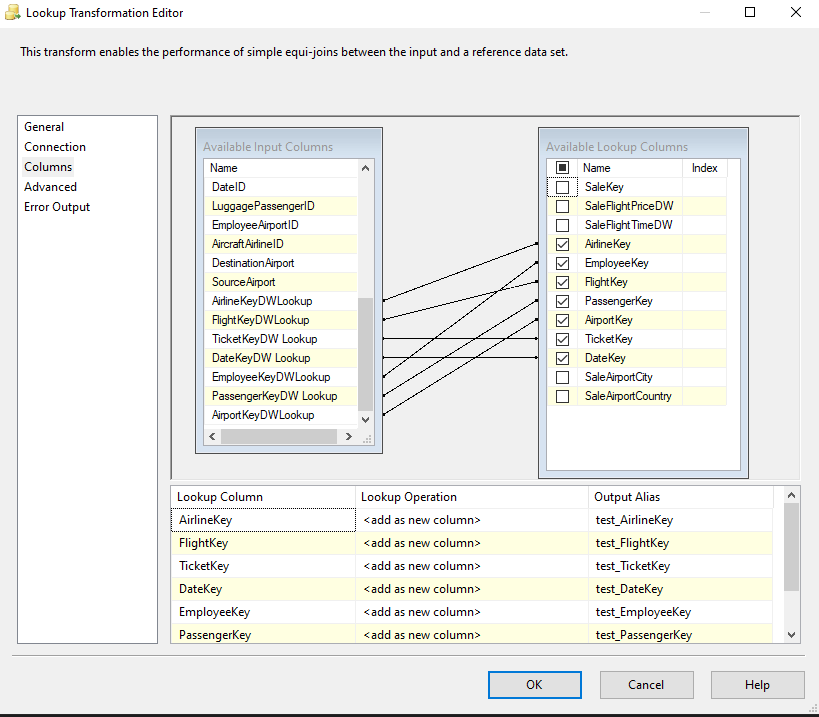


Slika 5‑7 Tablice koje sudjeluju u kreiranju Fact tablice



Upit 5‑1 Sve tablice čiji podaci ulaze u tablicu činjenica

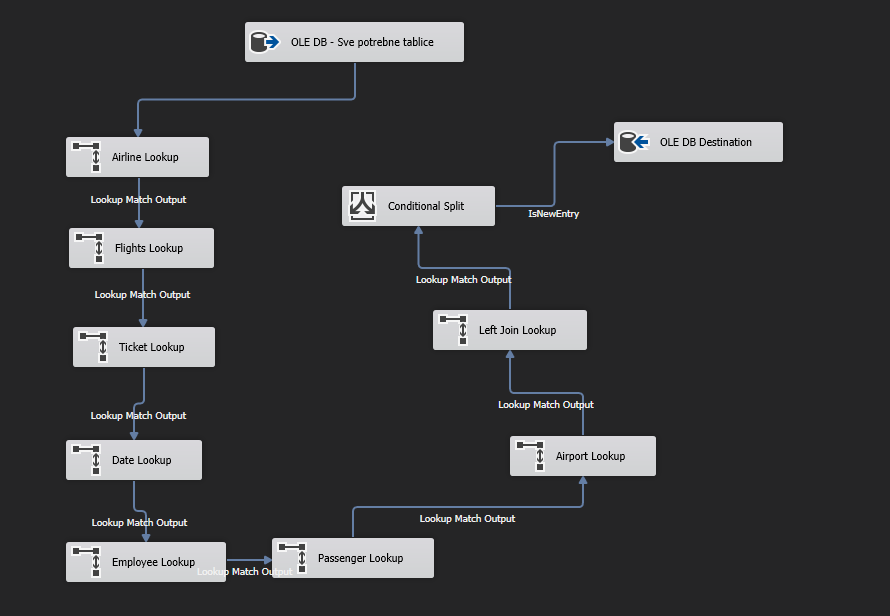
* Lookup – služi za prikupljanje ID iz skladišta podataka, dok ID podatka iz OLTP baze već imamo u dimenzijskoj tablici, koje ćemo koristiti za povezivanje podataka u tablici činjenica
* Left Join Lookup – provjera postojanja podataka u tablici činjenica tako što se uspoređuju ID iz tablice dimenzija s onim u tablici činjenica (Slika 5‑8)



Slika 5‑8 Left Join Lookup

* Conditional Split – provjera postojanja podataka kako bi se spriječilo kopiranje postojećih podataka tako što ispituje ISNULL(ID)
* OLED DB Destination. - odredište podataka tj. tablica činjenica (DW baza)

Na slici 5‑9 dan je prikaz toka podataka za popunjavanje tablice činjenica.



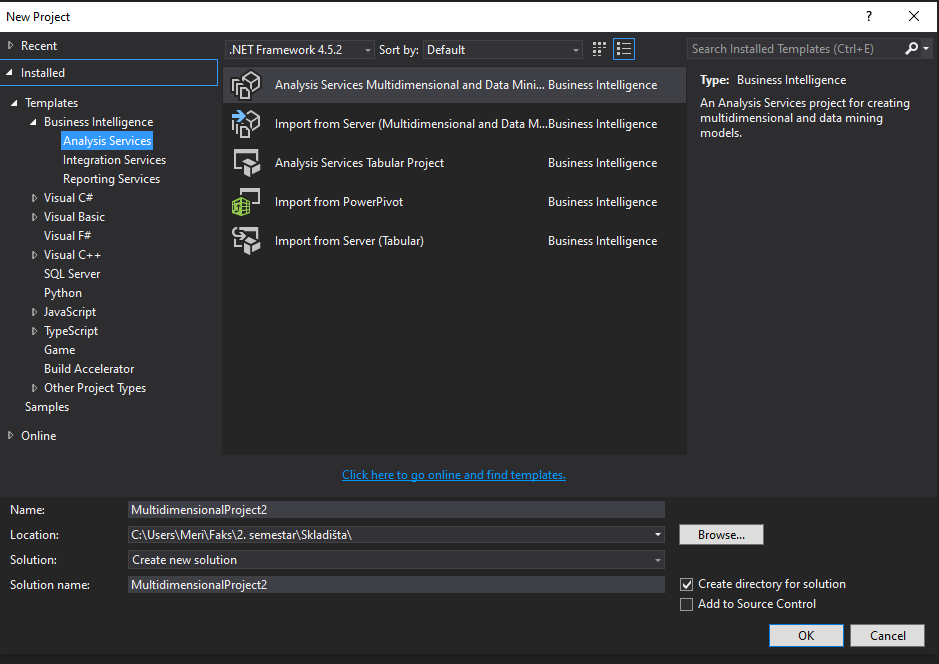
Slika 5‑9 Data Flow popunjavanja Fact tablice

# Izgradnja kocke

Nakon što se skladište podataka napuni podacima kreiramo kocku u SQL Server Analysis Services. Kocka je struktura optimizirana za brzu analizu podataka, a sastoji se od brojčanih činjenica koje nazivamo mjerama koje se kategoriziraju u dimenzije. Operacije nad podacima se najčešće rade u obliku *spreadsheet* formata – dvodimenzionalni podaci u redovima i stupcima. No, kocka nam omogućava spremanje višedimenzionalnih podataka. Također, svaka dimenzija može sadržavati hijerarhiju razina za specifikaciju kategorizacije. Na primjer, vremenska dimenzija sadrži hijerarhiju razina: godina, kvartal, mjesec, tjedan i dan.

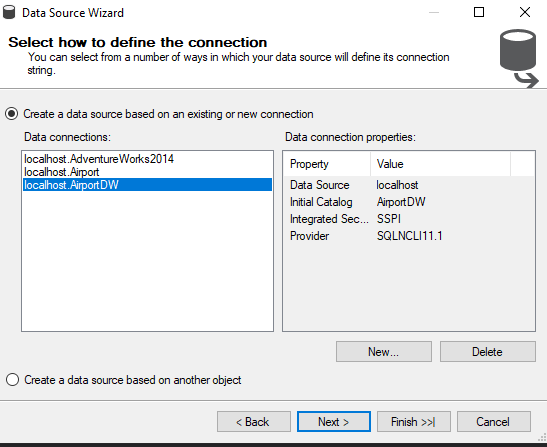
## Kreiranje SQL Server Analysis Services (SASS) projekta

Odabire se *New → Project → Analysis Services Multidimensional Project* (Slika 6‑1).



Slika 6‑1 Pokretanje novog Analysis Services projekta

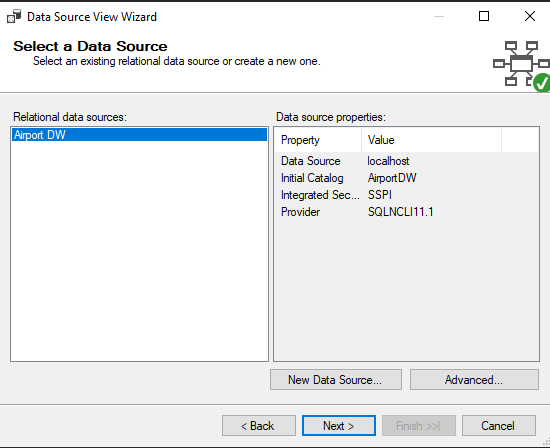
Prvi korak je dodavanje skladišta podataka kao izvor podataka: *Data Sources → New Data Source* te se prate koraci čarobnjaka (Slika 6‑2).



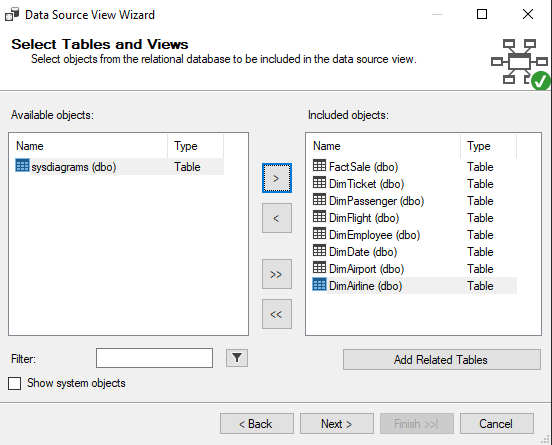
Slika 6‑2 Čarobnjak za postavljanje izvora podataka

## Definiranje pogleda

Definiramo dimenzije koje će se iz izvora podataka (skladišta podataka) prebaciti u kocku. *Data Source View* je pogled kojim definiramo koje podatke želimo prebaciti. Izgled čarobnjaka za postavljanje *Data Source View* i sučelje za odabir dimenzijskih tablica koje prelaze u kocku prikazani su na slici 6‑3 i slici 6‑4. Prvo se odabire tablica činjenica potom tablice dimenzija.

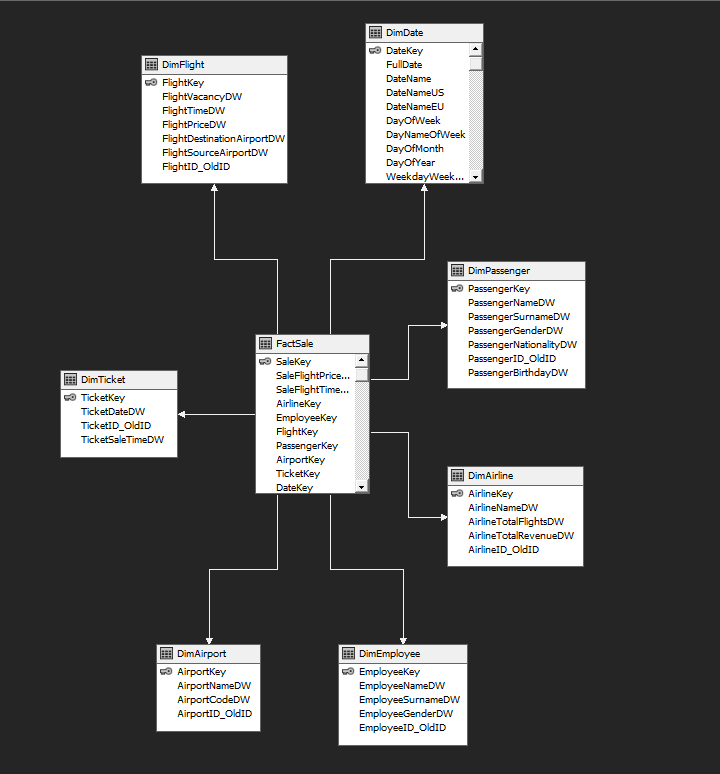


Slika 6‑3 Čarobnjak za postavljanje Data Source View



Slika 6‑4 Odabir tablica koje prebacujemo u dimenzije kocke

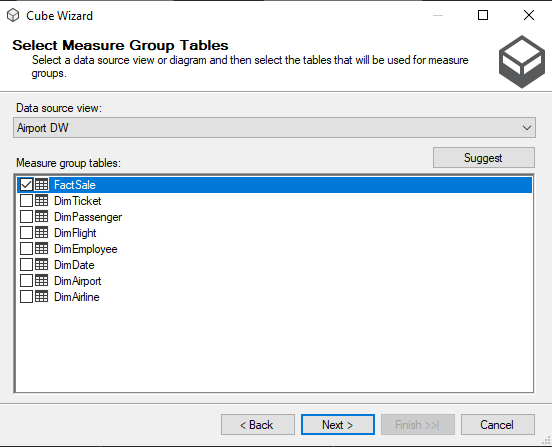
Dobiveni pogled prikazan je na slici 6‑5.



Slika 6‑5 Data Source View

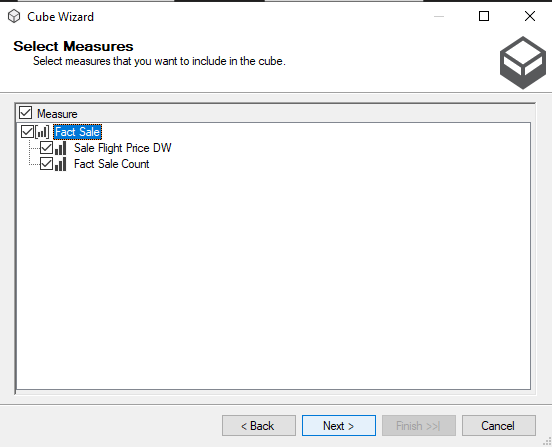
## Kreiranje kocke

Pogled će poslužiti kao predložak za definiranje kocke: *Cube → New Cube* u *Solution Exploreru*. Odabiremo opciju korištenja postojećih podataka te iz popisa tablica odabiremo tablicu činjenica (važno je naglasiti da može postojati više tablica činjenica, no u sklopu ovog projekta napravljena je samo jedna) kao što je prikazano na slici 6‑6.



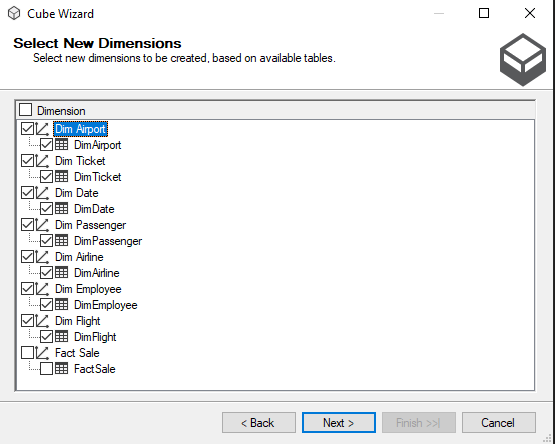
Slika 6‑6 Odabir činjenica

Potom se odabiru mjere kao što je prikazano na slici 6‑7.



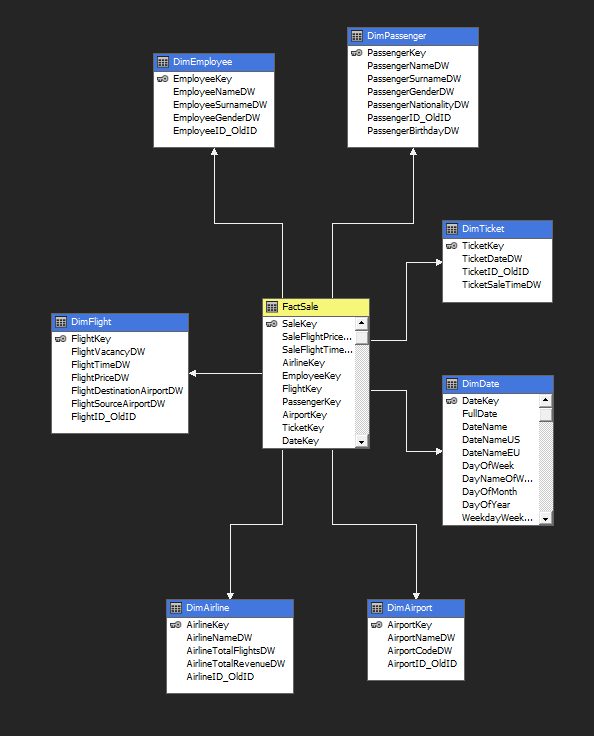
Slika 6‑7 Odabir mjera

Posljednje, odabiru se dimenzije kao što je prikazano na slici 6‑8.

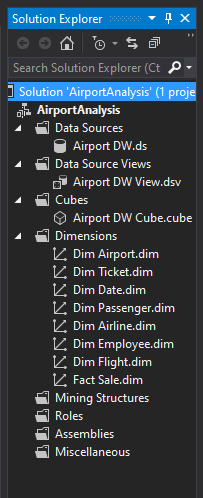


Slika 6‑8 Odabir dimenzija

Dobivena kocka i pripadni *Solution Explorer* prikazani su na slici 6‑9 i slici 6‑10 .



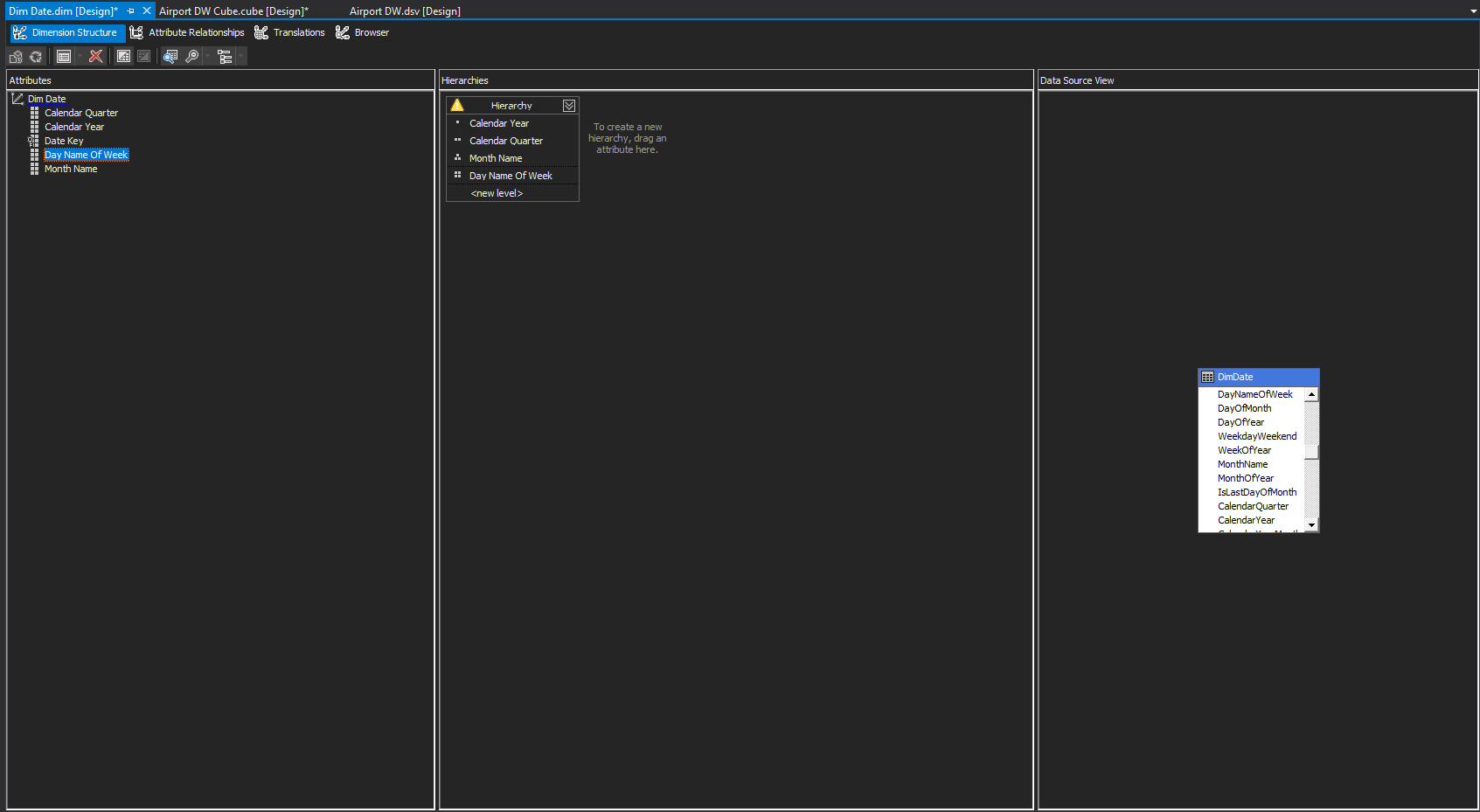
Slika 6‑9 Kocka



Slika 6‑10 Solution Exploerer

## Dodavanje atributa i kreiranje hijerarhija u dimenzijama

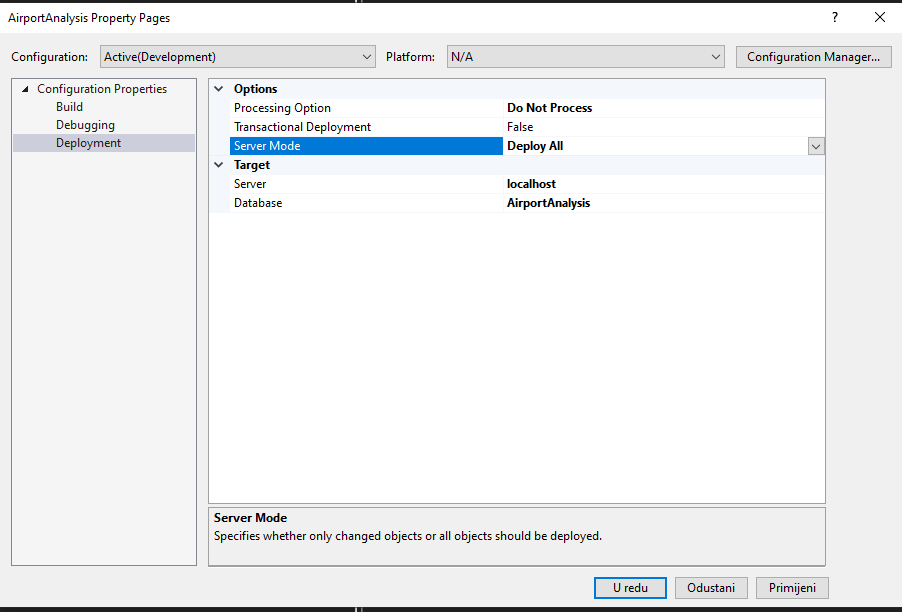
Ako unutar skladišta postoje dimenzije koje nisu ravne tzv. *flat*, za njih kreiramo zasebne hijerarhije dimenzija. Prigodan i jednostavan primjer za pokazati kreiranje hijerarhije je vremenska dimenzija DimDate za koju ćemo u pogledu definirati da uzima atribute *CalendarYear, CalendarQuarter, MonthName* i *DayNameOfWeek* (Slika 6‑11).



Slika 6‑11 Hijerarhija vremenske dimenzije

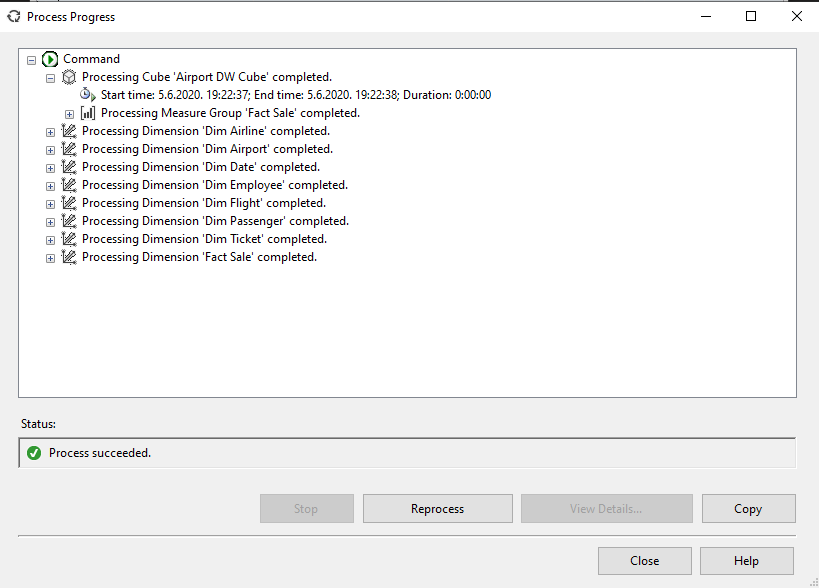
### Razvijanje (deploy) kocke

Nakon što se definiraju sve potrebne hijerarhije, kocka se razvija (*deploy*) na SQL Server Analysis Services Server kako bi je mogli koristiti u sklopu alata *CubePlayer*. Prije samog razvijanja potrebno je namjestiti postavke prikazane na slici 6‑12.



Slika 6‑12 Postavke deploymenta

Slika 6‑13 prikazuje uspješno izvršavanje *deploymenta* kocke.



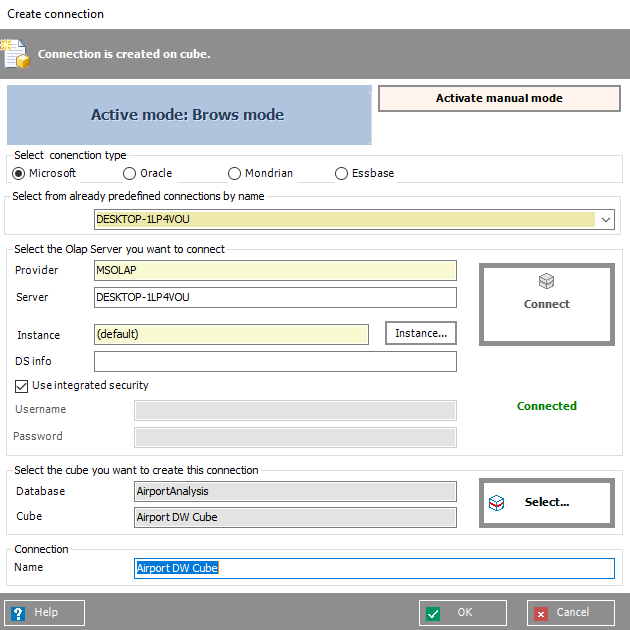
Slika 6‑13 Rezultat deploymenta

# Izvješća (Cube Player)

Nakon što se kreira kocka i razvije na server koristimo *Cube Player* alat za povezivanje na server i kreiranje željenih izvješća.

## Spajanje na SQL Server Analysis Services Server

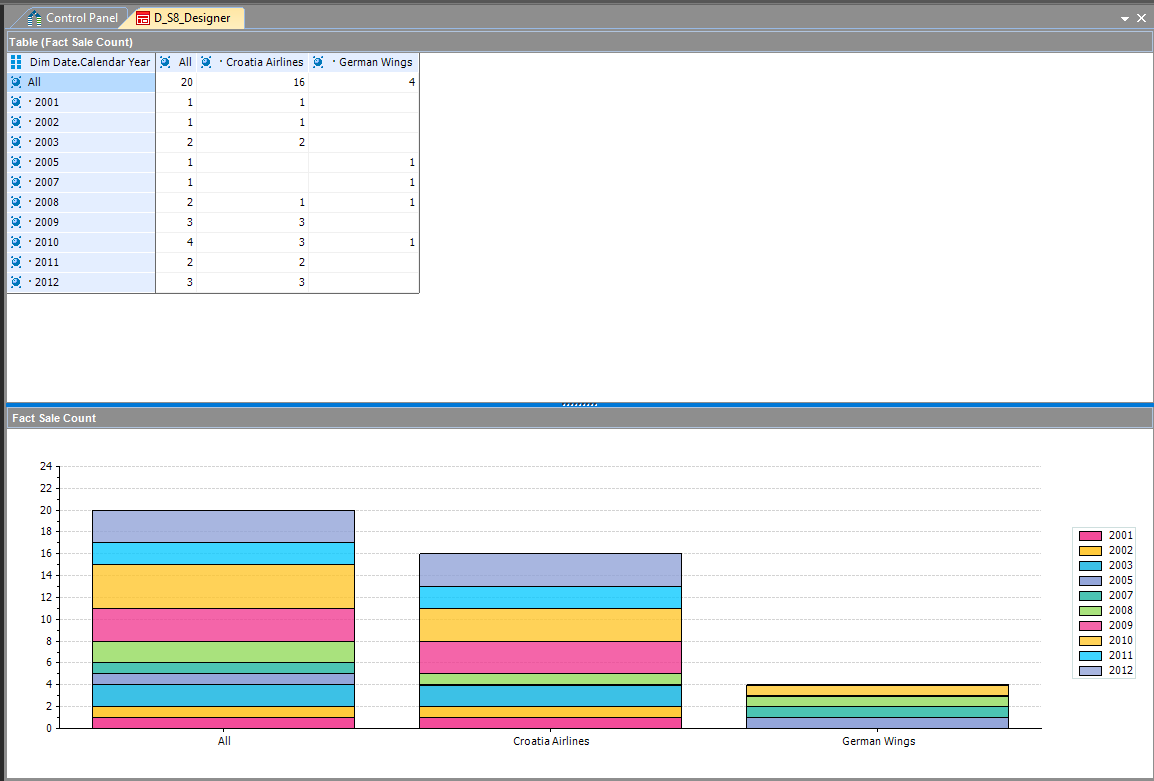
Kao što je navedeno u uvodu potrebno je spojiti *Cube Player* alat na server tako što postavljamo novu konekciju odabirom opcije *Connections*. Postavke za spajanje na server i kocku prikazane su na slici 7‑1.



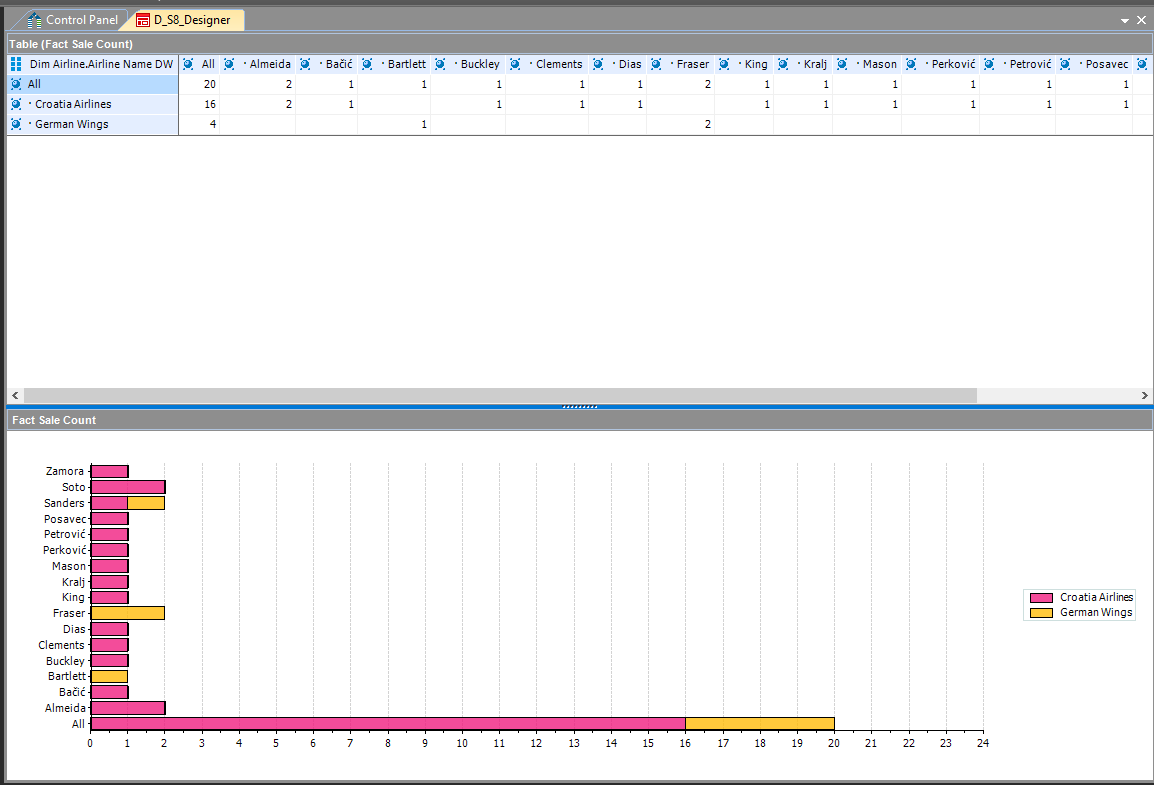
Slika 7‑1 Kreiranje konekcije na server i odabir kocke

## Kreiranje izvješća

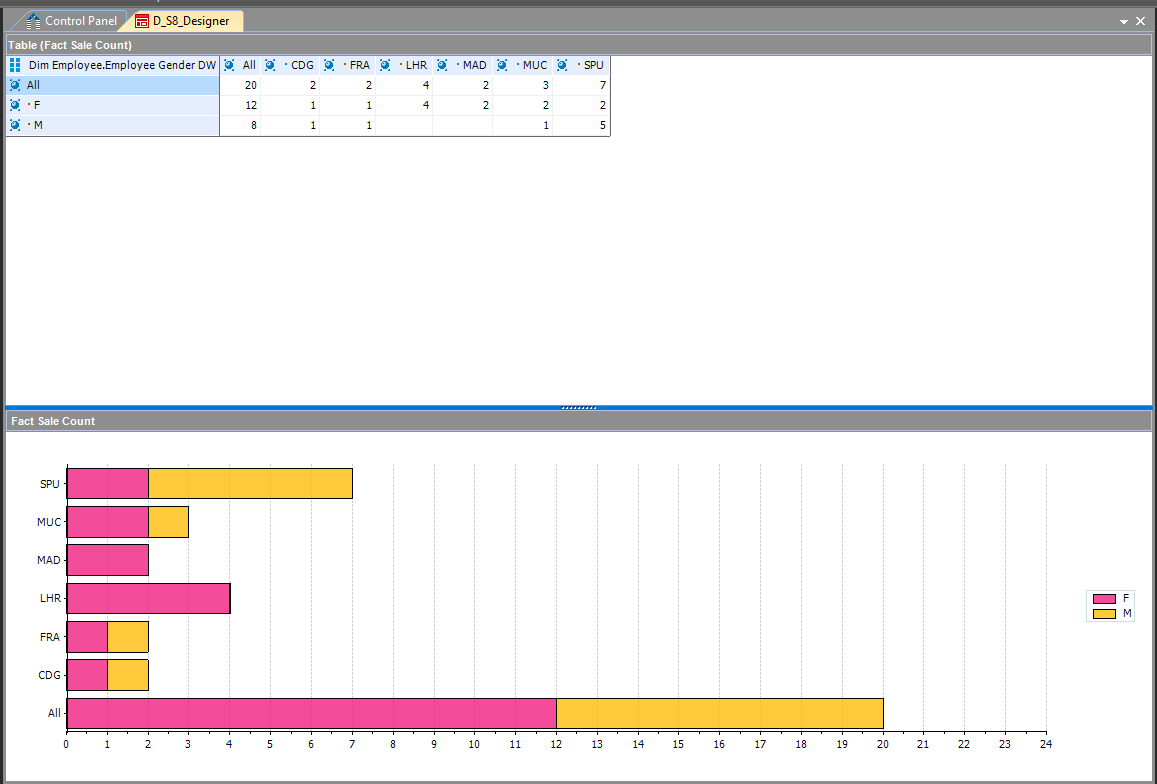
Projekt se u osnovi bazirao na generiranju izvješća, te su u ovom odlomku prikazana neka izvješća dobivena kao rezultat analize poslovnog procesa.



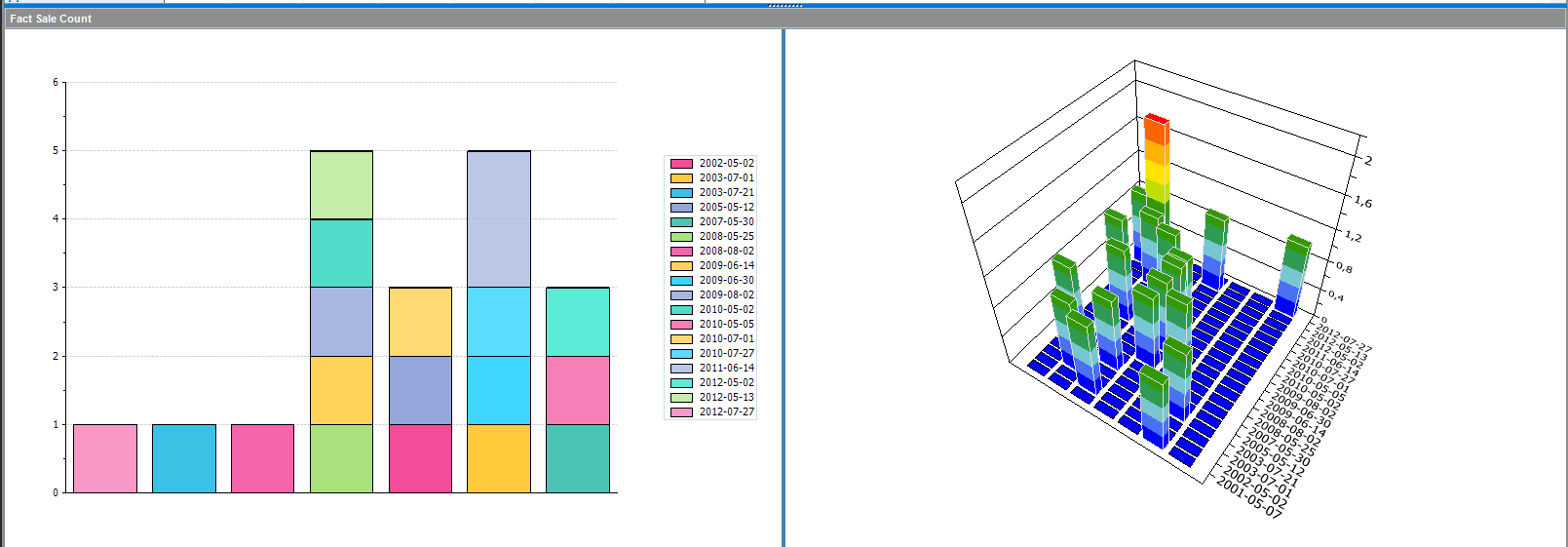
Slika 7‑2 Broj letova zrakoplovne agencije po godinama

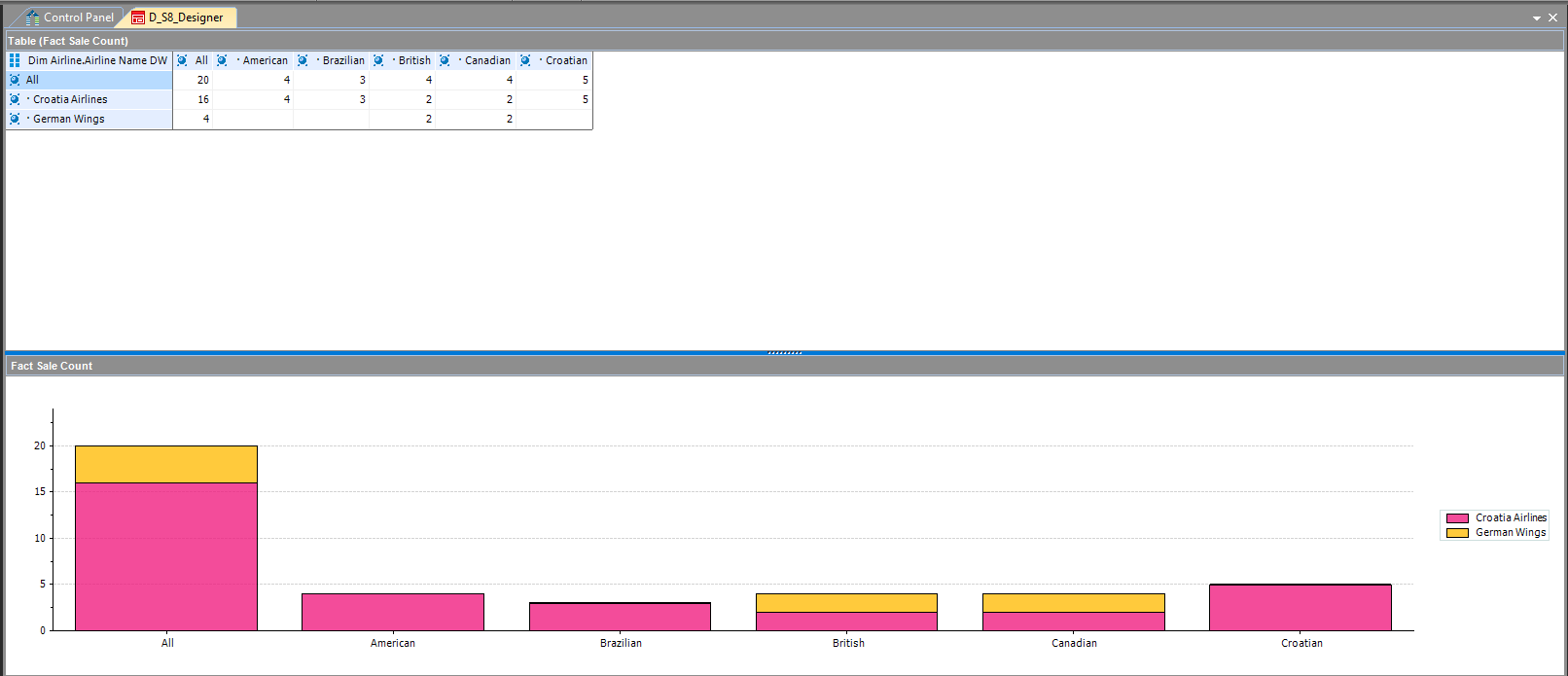


Slika 7‑3 Broj kupljenih karata po kupcu za određenu zrakoplovnu agenciju

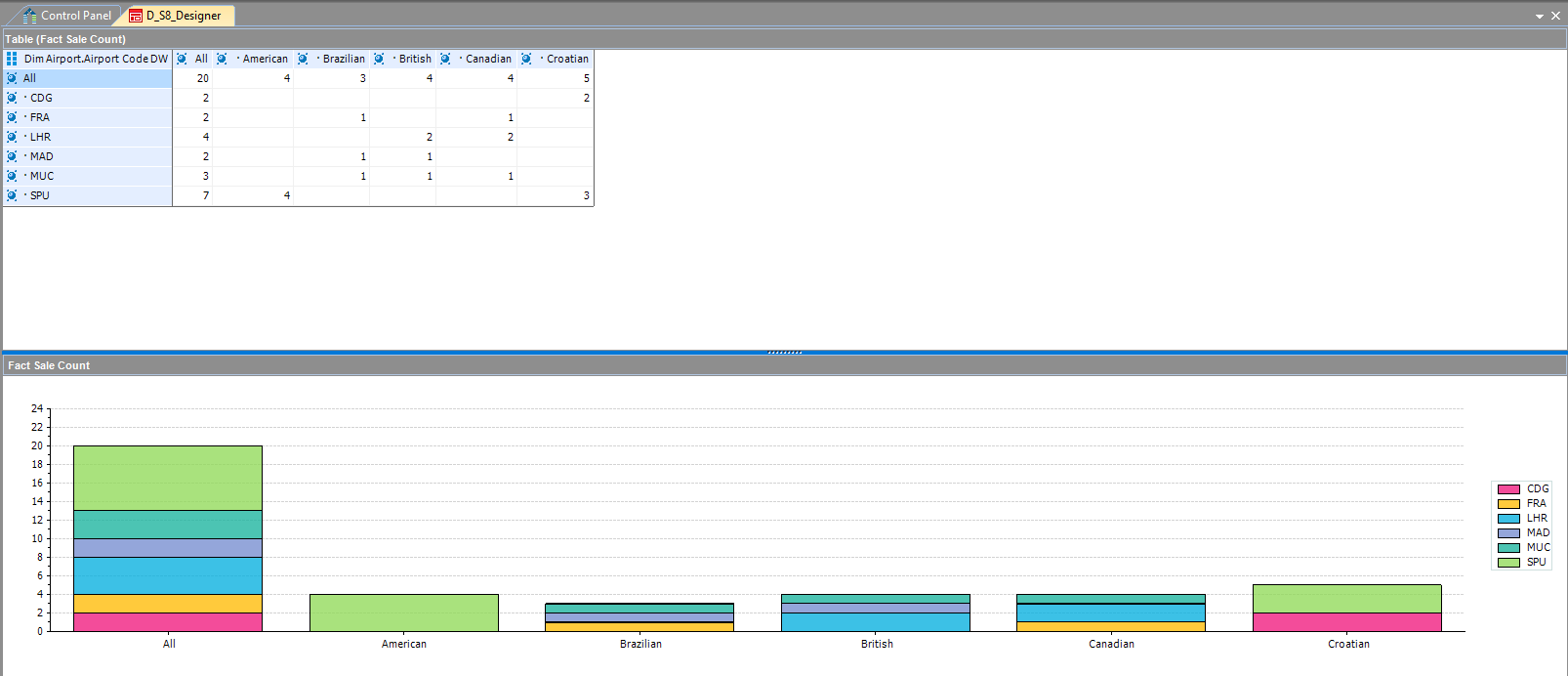


Slika 7‑4 Izvještaj omjera spola zaposlenika svake zrakoplovne luke

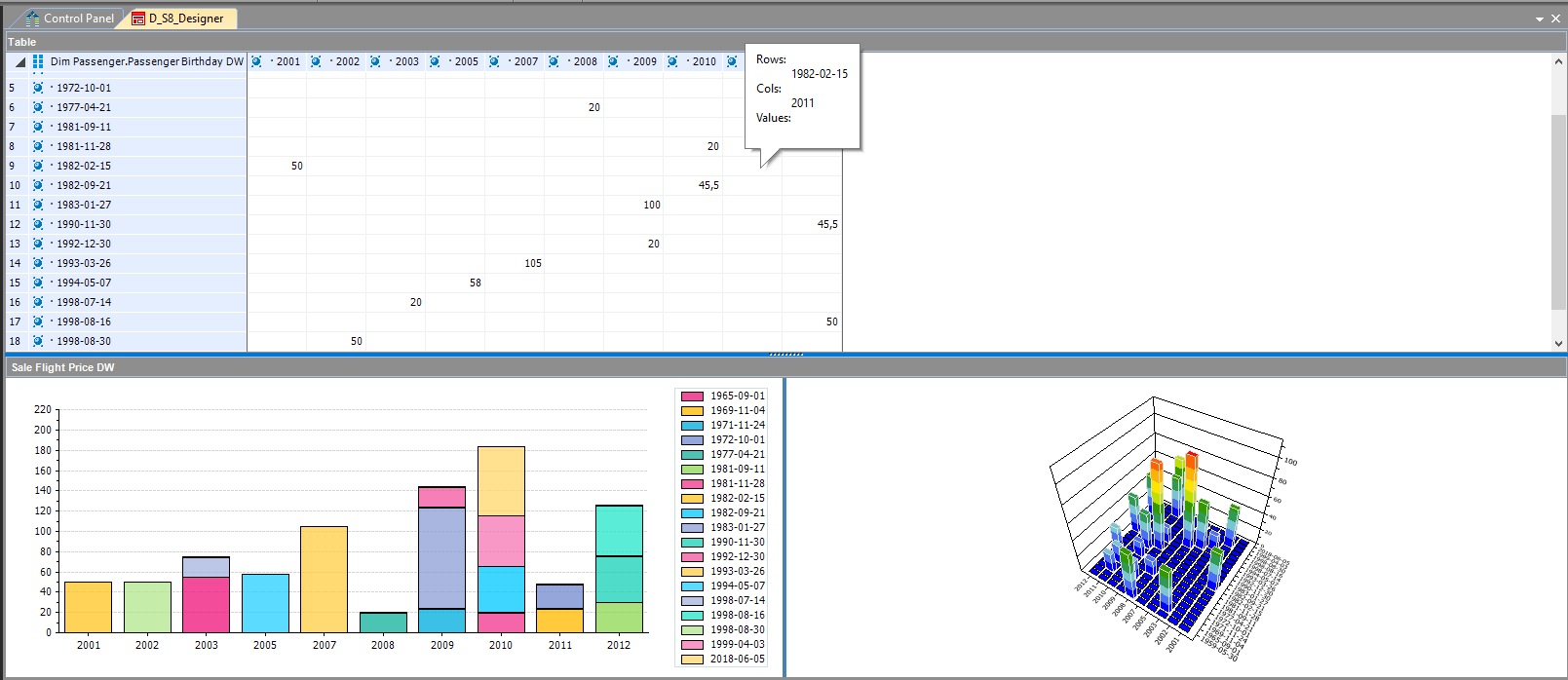
Slika 7‑5 Izvještaj prodaje karata po danima u tjednu



Slika 7‑6 Izvještaj frekventnosti korištenja usluga zrakoplovne agencije ovisno o nacionalnosti putnika



Slika 7‑7 Izvještaj putnika po nacionalnosti za svaku zrakoplovnu luku



Slika 7‑8 Izvještaj putnika po starosti po godinama poslovanja

# Zaključak

Ovaj seminarski rad na konkretnom primjeru objašnjava primjenu skladišta podataka. Realizacija se radila sekvencijalno prateći tjedne laboratorijske vježbe. Sama primjena skladišta podataka fokusirana je na generiranje izvještaja, ne toliko na efikasno spremanje podataka. Stoga je bitno prije početka projekta dobro analizirati i procijeniti što je glavni fokus projekta i je li skladište podataka pravi izbor. Ukoliko je bitnija analiza podataka i generiranje čitkih izvještaja od fleksibilnosti upravljanja podacima tada je skladište podataka dobar izbor.

Za primjer ovog projekta i praćenja poslovnih procesa zrakoplovnih luka skladište podataka uvelike olakšava snalaženje u potencijalno velikoj i raznovrsnoj količini podataka koja se brzo generira. S obzirom na frekventnost zrakoplovnog prometa i preciznost analize koju zahtjeva korištenje skladište podataka daje efikasan i relativno brz *feedback* poslovnih procesa. Na taj način svaka zrakoplovna kompanija ima pregledan uvid u svoj promet prema raznim kategorijama i može procijeniti potrebu za smanjenjem ili povećanjem prometa na određenim područjima ili za određene termine. Također, može se pratiti rad zaposlenika te procijeniti koji zaposlenici više pridonose prodaji. Što se tiče putnika bitno je imati uvid u demografiju koja gravitira određenim lokacijama i tako omogućiti prilagođavanje cijena i frekventnosti letova. Uvid u prodaju po putniku također omogućava eventualno stvaranje *loyalty* programa.

Zaključak je da proces skladištenja podataka uvelike olakšava praćenje poslovnih procesa zbog pomne selekcije podataka za analizu tako da se redundantni podaci ne prenose u logički model. Ukoliko bi koristili neki drugi oblik spremanja podataka zbog količine informacija koja se pohranjuje analiza bi se eksponencijalno zakomplicirala. Pretraživanje relacijskih baza zahtjeva spajanje nekoliko razina tablica samo kako bi se došlo do jednog podatka što može bit zahtjevno i sporo i za programera i za računalo. Skladište podataka zbog jednostavne strukture omogućava jednostavne upite i izoliranje bitnih podataka. Vidimo na prethodno objašnjenom primjeru teme ovog projekta da korištenje skladišta podataka omogućava brzu reakciju na promjene u poslovnom procesu i prilagođavanje poslovanja dobivenim rezultatima kako bi poslovanje teklo što brže i efikasnije.

# Prilozi

## Kazalo slika

[Slika 2‑1 Idejna izvedba skladišta podataka 2](#_Toc42269423)

[Slika 3‑1 Dijagram transakcijske baze 5](#_Toc42269424)

[Slika 4‑1 Logički model skladišta podataka 7](#_Toc42269425)

[Slika 5‑1 Kreiranje novog projekta 9](#_Toc42269426)

[Slika 5‑2 Odabir Integration Services projekta 10](#_Toc42269427)

[Slika 5‑3 Postavljanje connection managera 11](#_Toc42269428)

[Slika 5‑4 Odabir DB i DW 11](#_Toc42269429)

[Slika 5‑5 DataFlowTask za dimenzije i činjenice 12](#_Toc42269430)

[Slika 5‑6 Data Flow Employee dimenzijske tablice 13](#_Toc42269431)

[Slika 5‑7 Tablice koje sudjeluju u kreiranju Fact tablice 14](#_Toc42269432)

[Slika 5‑8 Left Join Lookup 16](#_Toc42269433)

[Slika 5‑9 Data Flow popunjavanja Fact tablice 17](#_Toc42269434)

[Slika 6‑1 Pokretanje novog Analysis Services projekta 18](#_Toc42269435)

[Slika 6‑2 Čarobnjak za postavljanje izvora podataka 19](#_Toc42269436)

[Slika 6‑3 Čarobnjak za postavljanje Data Source View 20](#_Toc42269437)

[Slika 6‑4 Odabir tablica koje prebacujemo u dimenzije kocke 20](#_Toc42269438)

[Slika 6‑5 Data Source View 21](#_Toc42269439)

[Slika 6‑6 Odabir činjenica 22](#_Toc42269440)

[Slika 6‑7 Odabir mjera 23](#_Toc42269441)

[Slika 6‑8 Odabir dimenzija 24](#_Toc42269442)

[Slika 6‑9 Kocka 25](#_Toc42269443)

[Slika 6‑10 Solution Exploerer 26](#_Toc42269444)

[Slika 6‑11 Hijerarhija vremenske dimenzije 27](#_Toc42269445)

[Slika 6‑12 Postavke deploymenta 27](#_Toc42269446)

[Slika 6‑13 Rezultat deploymenta 28](#_Toc42269447)

[Slika 7‑1 Kreiranje konekcije na server i odabir kocke 29](#_Toc42269448)

[Slika 7‑2 Broj letova zrakoplovne agencije po godinama 30](#_Toc42269449)

[Slika 7‑3 Broj kupljenih karata po kupcu za određenu zrakoplovnu agenciju 30](#_Toc42269450)

[Slika 7‑4 Izvještaj omjera spola zaposlenika svake zrakoplovne luke 31](#_Toc42269451)

[Slika 7‑5 Izvještaj prodaje karata u vremenu 31](#_Toc42269452)

[Slika 7‑6 Izvještaj frekventnosti korištenja usluga zrakoplovne agencije ovisno o nacionalnosti putnika 32](#_Toc42269453)

[Slika 7‑7 Izvještaj putnika po nacionalnosti za svaku zrakoplovnu luku 32](#_Toc42269454)

[Slika 7‑8 Izvještaj cijena letova za sve zrakoplovne agencije kroz godine 33](#_Toc42269455)

## Kazalo upita

[Upit 5‑1 Sve tablice čiji podaci ulaze u tablicu činjenica 15](#_Toc42269466)

## Popis oznaka i kratica

DW Data Warehouse

BIDS Business Intelligence Development Studio

SSIS SQL Server Integration Services

SSAS SQL Server Analysis Services

OLTP Online Transaction Processing

OLAP Online Analytical Processing

OLE DB Object Linking and Embedding Database

# Literatura

1. <https://elearning.fesb.unist.hr/mod/resource/view.php?id=70736> Skladišta Podataka – predavanja 2019/20, doc.dr.sc. S. Čelar, ljetni semestar ak. godina 2019/2020
2. <https://elearning.fesb.unist.hr/mod/resource/view.php?id=59582> Skladišta Podataka – upute za laboratorijske ver. 2.0.1, doc.dr.sc. S. Čelar, ljetni semestar ak. godina 2015/2016
3. <https://www.slideshare.net/antoniosch/building-data-warehouse-in-sql-server> Building Data Warehouse in SQL Server, Antonios Chatzipavlis, 2015