Skladištenje podataka i poslovna inteligencija na platformi Microsoft Azure

Koprek, David

Master's thesis / Diplomski rad

2021

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Organization and Informatics / Sveučilište u Zagrebu, Fakultet organizacije i informatike**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:211:085386

SVEUČILIŠTE U ZAGREBU
FAKULTET ORGANIZACIJE I INFORMATIKE
VARAŽDIN

David Koprek

SKLADIŠTENJE PODATAKA I POSLOVNA INTELIGENCIJA NA PLATFORMI MICROSOFT AZURE

DIPLOMSKI RAD

Varaždin, 2021.

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

David Koprek

Matični broj: 44846/16–R

Studij: Baze podataka i baze znanja

SKLADIŠTENJE PODATAKA I POSLOVNA INTELIGENCIJA NA PLATFORMI MICROSOFT AZURE

DIPLOMSKI RAD

Mentor:

Prof. dr. sc. Kornelije Rabuzin

Varaždin, rujan 2021.

David Koprek

Izjava o izvornosti

Izjavljujem da je moj diplomski rad izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

Autor potvrdio prihvaćanjem odredbi u sustavu FOI-radovi

Sažetak

Rad se bavi teorijskim polazištima skladištenja podataka i poslovne inteligencije poput ciljeva takvih sustava, životnog ciklusa, komponenti i dimenzijskog modeliranja. Implementirano je skladište podataka pomoću *Azure SQL Database*-a koje obično služi za relacijske baze podataka ali donosi svoje prednosti kod skladištenja podataka kao i pomoću *Azure Synapse Analytics*-a, relativno novog Microsoft-ovog proizvoda za masovno skladištenje i analitiku podataka na razini poduzeća. Namjera je bila istražiti i pokazati mogućnosti izrade skladišta podataka na *Azure* platformi. Oba načina pokazala su se jednostavnim i efikasnim te jednostavnim za korištenje u alatu poslovne inteligencije, *Microsoft Power BI*-u. Za baratanje skladištima korišten je *SQL Server Management Studio* alat. Prije toga izrađen je model skladišta u skladu s prethodno objašnjenim konceptima i tehnikama dimenzijskog modeliranja. Korištenjem *PowerBI* alata napravljeni su izvještaji u svrhu poslovne inteligencije, a s namjerom prikazivanja nekih od mogućnosti alata i analize podataka iz domene.

Ključne riječi: skladišta podataka, Microsoft Azure, Synapse Analytics, poslovna inteligencija, Power BI, SQL pool, Azure SQL Database

Sadržaj

1. Uvod

Jedna od najvažnijih i najkorisnijih imovina bilo koje organizacije ili poduzeća jesu informacije. One se gotovo uvijek koriste u dvije svrhe. Jedna od njih je čuvanje zapisa o operacijama poduzeća, a druga za analitičko donošenje odluka. Za ovu drugu svrhu koriste se DW/BI sustavi (eng. *Data Warehouse/Business Intelligence*, Skladište podataka/Poslovna inteligencija) [1] koji su tema ovog rada. Korisnici takvih sustava analitički pregledavaju operacije i transakcije koje se bilježe prilikom rada organizacije ili poduzeća. Oni primjerice broje nove narudžbe i uspoređuju ih s prošlotjednim, pitaju se zašto je došlo do rasta broja novih kupaca, o čemu sve su se kupci žalili i slična konkretna pitanja koja su važna za poslovanje i daljnji rast organizacije. Iako su za odgovore na takva pitanja potrebni detaljni podaci, DW/BI korisnici se gotovo nikad ne bave pojedinim transakcijama. Sustavi koji se koriste u tu svrhu moraju biti optimizirani za upite visokih performansi koji uključuju stotine, tisuće ili čak milijune transakcija koje se pretvaraju u skup odgovora na prethodno postavljena pitanja. Kako bi se precizno mjerile performanse poslovanja kroz vrijeme u ovakvim sustavima često se traži očuvanje povijesnog konteksta (vremena transakcije i sl.). [1]

Učinci ideje o stvaranju skladišta podataka i implementiranje istih u poslovanje su brojni. Prema Inmonu et.al [2] učinak ideje skladišta podataka značajno je utjecao na poslovanja različitih organizacija. Neka od zahvaćenih područja uključuju:

- Zrakoplovne kompanije u središtu čijih *frequent flyer* programa jest najčešće njihovo centralno skladište podataka
- Analize prevara vezanih za kreditne kartice. Na temelju skladišta podataka kreiraju se potrošački profili (eng. *Spending profiles*) za svakog klijenta. Kad se dogodi transakcija neuobičajena za profil, kompanija može provjeriti događa li se kakva prijevara ili ilegalna radnja koristeći tu karticu.
- Upravljanje inventarom. U skladištima podataka detaljno se prati inventar, trendovi korištenja i prilike. Detaljnim razumijevanjem utroška dobara organizacije moguće je kvalitetno pratiti viškove i manjkove te ih držati na optimalnoj razini
- Profile kupaca. Organizacije koje žele bolje “upoznati” svoje kupce čuvaju njihove navike trošenja, stvari koje privlače pažnju i sl. Svi takvi podaci se čuvaju u skladištu podataka.

Tema ovog diplomskog rada jest skladištenje podataka - istraživanje teorijske podloge i temeljnih koncepata bitnih za razumijevanje i izradu skladišta podataka te mogućnosti implementacije istog koristeći Microsoft-ovu platformu *Azure*. Proći će se i kroz teorijske osnove poslovne inteligencije koja je usko povezana s područjem skladištenja podataka. Nakon implementacije skladišta na spomenutoj platformi na dva načina ono će se iskoristiti za kreiranje izvještaja poslovne inteligencije koristeći alat *PowerBI* kako bi se prikazale neke od njegovih mogućnosti.

2. Tehnike i metode rada

Proučavanjem stručne literature izvučeni su bitni koncepti i ideje skladištenja podataka i poslovne inteligencije. Nastojalo se prikazati stvari koje su relevantne i koje su se koristile u praktičnom dijelu rada. Čitanjem i praćenjem službene dokumentacije korištenih aplikacija i alata implementirani su prethodno objašnjeni koncepti iz teorije. Za pojašnjenja eventualnih nejasnoća i nedostataka iz tih izvora korišteni su dodatni *online* izvori.

Alati, tehnologije i aplikacije koje su se koristile u ovom radu:

- SQL Server Management Studio 18 za upravljanje skladištima (bazama) podataka
- Microsoft Azure Portal, resursi:
  - Data Lake Storage Gen 2 za pohranu datoteka
  - SQL Database za izradu skladišta podataka
  - Synapse Analytics za izradu skladišta podataka
  - Embedded Power BI za prikaz integracije sa Synapse Analytics-om
- Power BI (Desktop) za izradu interaktivnih izvještaja poslovne inteligencije

Za izradu i uređivanje LaTeX projekta korišten je *Overleaf* (*www.overleaf.com*)

3. Skladišta podataka

U ovom poglavlju proći će se kroz neke stavke i koncepte bitne kod izrade skladišta podataka poput ciljeva izrade koji usmjeravaju cijeli razvoj, životni ciklus razvoja te dimenzijskog modeliranja. Biti će pregledane komponente okruženja (eng. *environment*) skladišta podataka, ETL, shema zvijezde i njezine činjenične i dimenzijske tablice te koncept zrnatosti (eng. *granularity*) i zašto je on bitan. Prije svega, ukratko će se proći kroz povijest spremanja i korištenja podataka u organizacijama.

3.1. Kratki povijesni pregled

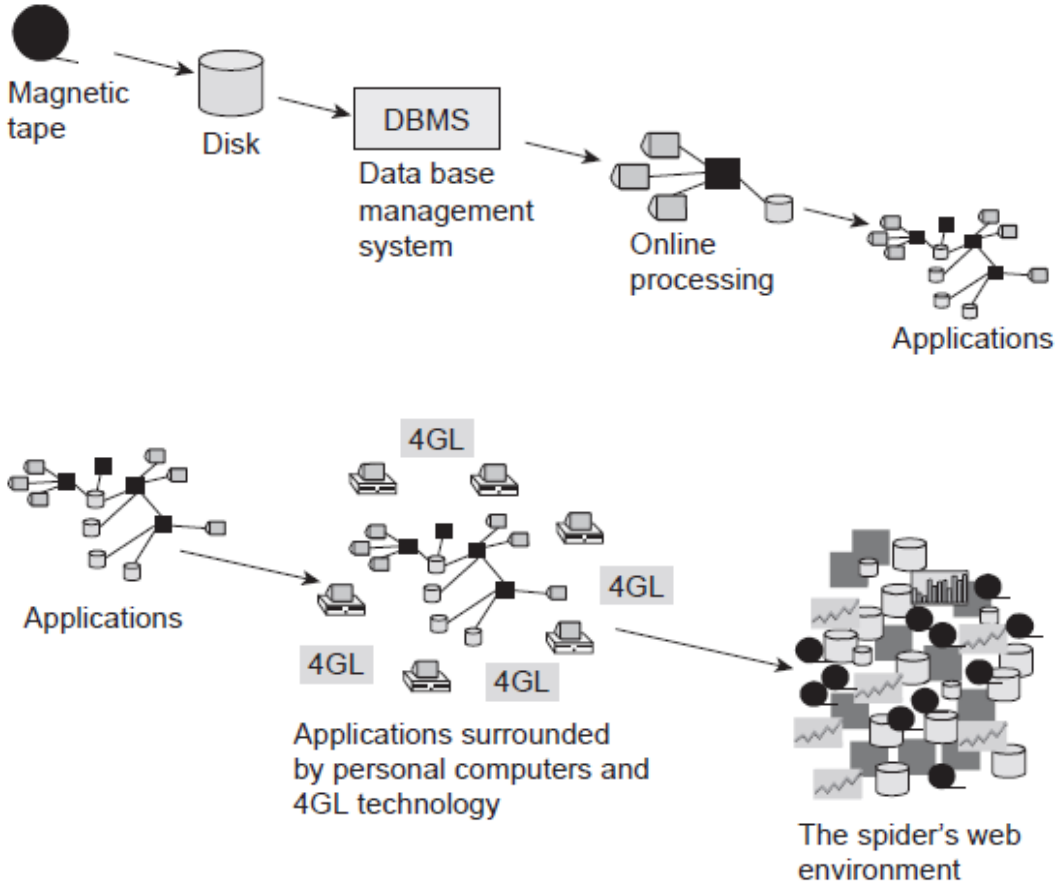
Prije govora o samim skladištima dobro je pogledati razvoj pohrane podataka kroz vrijeme čime će lakše shvatiti svrha skladišta i kako je uopće došlo do takvog koncepta. U svojoj knjizi, *DW 2.0: the architecture for the next generation of data warehousing*, Inmon et al. [2] daju dobar sažeti uvid.

Na samom početku čuvanja podataka mjesto za pohranu bilo je skupo i vrlo ograničeno. Pojavom magnetskih vrpci omogućeno je spremanje većih količina podataka na mnogo jeftiniji način. Ipak, čitanje podataka moralo se raditi isključivo sekvencijalno, a same vrpce nisu izrazito stabilan medij zbog podložnosti oštećenjima. [2]. Sekvencijalno čitanje podataka je neefikasno i relativno sporo jer se moraju iščitati svi podaci prije onog koji se traži da bi se došlo do njega. Sljedeći veliki iskorak bili su diskovi za spremanje podataka. Podacima se može pristupati direktno, a moguće je i zapisivati preko starih podataka. [2]

Takva diskovna pohrana podataka ubrzo je popraćena sa softverima koje nazivamo sustavima za upravljanje bazama podataka (eng. *Database management system*, DBMS) koji su naravno služili za manipulacijom podataka na disku. Pod tim se podrazumijevaju aktivnosti identifikacije lokacije podatka na disku, rješavanje konflikta u slučaju mapiranja različitih podataka na istu fizičku lokaciju na disku, brisanje podataka sa diska i dr. S mogućnošću direktnog pristupa podacima razvile su *online* aplikacije, aplikacije koje su ovisile o tome da računalo brzo i konzistentno pristupa podacima. Neke od komercijalnih *online* aplikacija koje su razvijene jesu bile za procesiranje bankovnog blagajništva, zrakoplovnih rezervacija, kontrole proizvodnje i mnoge druge. Njihova popularnost izazvala je nagli rast broja takvih aplikacija i uskoro se pojavio problem da korisnik zna da podaci postoje, ali ih je teško pronaći i nema garancije da su pronađeni točni podaci ako je korisnik i dospio do njih. [2] Prikaz razvoja do aplikacija prikazan je na gornjem dijelu slike 1.

Kao rješenje takvih problema pojavile su se dvije tehnologije - osobna računala i 4GL. Osobna računala dozvolila su da svatko obavlja vlastito procesiranje po želji. Dodatno, pojavili su se softveri s proračunskim tablicama. Ideja 4GL (eng. *Fourth-generation*, četvrta generacija) tehnologije je bila da se sustavski razvoj i programiranje pojednostave kako bi svatko mogao obavljati takve zadatke eliminirajući tako potrebu za IT sektorom. Otklanjanjem problema pristupa podacima korisnici su ubrzo shvatili nedostatke koje sami podaci nose sa sobom. Podaci kojima su pristupali nisu nužno bili točni, mogli su biti nepotpuni, nepravovremeni, postojalo

je više verzija istih podataka i sl. Nered koji je nastao se ponekad naziva okruženje paukove mreže (eng. *Spider web enviroment*) zbog velikog broja veza između velikog broja elemenata organizacije (osobnih računala, online aplikacija, 4GL-a i dr.). Takvo okruženje postajalo je nevjerojatno kompleksno i za njega nije bilo budućnosti pa se pojavila potreba za drugačijom informacijskom strukturom za koju će se ispostaviti da se temelji na skladištu podataka. [2] Grafički prikaz *spider web* okruženja i kako je do njega došlo od aplikacija nalazi se na donjem dijelu slike 1.



Slika 1: Ranija povijest pohrane podataka (Izvor: Kimball i Ross, 2013)

Uvođenje skladišta podataka predstavlja veliku promjenu u razmišljanju, iz ideje da baza podataka mora služiti svim svrhama vezano uz podatke u činjenicu da postoji mnogo različitih vrsta baza podataka. Skladište podataka je temelj za informacijsko procesiranje i sadrži integrirane zrnate povijesne podatke [2] i za razliku od baze podataka kod koje je redundantnost nepoželjna, skladište sadrži redudantne podatke [3]. Česti problem modernih poslovnih sustava jest da postoji mnogo baza podataka i aplikacija koje rade s njima, no mogućnost integracije tih podataka je skoro nepostojeća [3] što nije slučaj kod skladišta podataka. Integracija podataka dozvoljava organizaciji pogled na podatke na širini cijelog poslovanja. Još jedna prednost posljedica je zrnatosti podataka, a to je fleksibilnost podataka. Zbog fine razine detalja moguće je iste podatke gledati na različite načine, a ipak postoji samo jedna verzija istine. Još jedna prednost skladišta jest čuvanje povijesnih podataka jer je ono kvalitetan način čuvanja podataka i po nekoliko godina. [2]

Skladište podataka je spremište integriranih podataka pribavljenih iz različith izvora za specifičnu svrhu multidimezionalne analize podataka.[4] Ono je zapravo baza podataka koja sadrži povijesne i integrirane podatke iz različitih izvora i kao takva služi generiranju različitih izvještaja, provođenju OLAP analiza te rudarenju podataka. [3] Može se definirati i kao kolekcija subjektno orijentiranih, integriranih, nepromjenjivih podataka koji variraju u vremenu, a služe kao podrška menadžerskim odlukama.[4] Značenje ovih karakteristika je sljedeće:

- Subjektno orijentirano (eng. *Subject oriented*) znači da se skladišta podataka orijentiraju na analitičke potrebe različitih područja organizacije [4]
- Integrirani (eng. *Integrated*) znači da se podaci pribavljeni iz različitih operativnih i vanjskih sustava moraju spojiti što podrazumijeva rješavanje problema zbog razlika u definicijama i sadržaju podataka, formatu i kodiranju podataka, sinonimima (istim podacima pod različitim imenima), homonimima (podacima s istim imenom ali različitim značenjem), višestrukim pojavljivanjem podataka i sl. [4]
- Nepromjenjivi (eng. *Nonvolatile*) znači da je osigurana izdržljivost podataka onemogućavanjem modifikacija i brisanja. Time se postižu dugotrajniji podaci u odnosu na operativne sustave. [4]
- Variranje u vremenu (eng. *Time varying*) ukazuje na mogućnost zadržavanja različitih vrijednosti za istu informaciju kao i vrijeme kad su se dogodile promjene tih vrijednosti. [4]

3.2. Ciljevi DW/BI sustava

Prije modeliranja i konstruiranja skladišta podataka važno je predočiti si glavne ciljeve skladišta podataka i poslovne inteligencije. Potrebe za ovakvim sustavima mogu se lako iščitati iz izjava koje se mogu čuti od strane menadžmenta organizacije, kao što su: “Skupljamo mnogo podataka, ali nemamo im kako pristupiti”, “Pokaži mi samo ono što je bitno”, “Na sastancima raspravljamo o brojkama umjesto da donosimo odluke”, “Trebamo donijeti odluke na temelju činjenica” i dr. [1]

Ovo sve su problemi s područja skladištenja podataka i poslovne inteligencije iz kojih se mogu izvući zahtjevi za dobar i koristan DW/BI sustav.

DW/BI sustav mora učiniti informacije lako dostupnima. Sadržaji ovih sustavi moraju biti razumljivi, a podaci bi trebali biti intuitivni i očiti za korisnika, a ne samo za developera. Strukturiranje podataka i označavanje trebale bi imitirati tok misli i vokabular korisnika. Alati za poslovnu inteligenciju kojima se pristupa podacima moraju biti jednostavni i laki za korištenje. Također moraju vraćati rezultate upita s minimalnim čekanjem za korisnika. [1]

Prezentacija informacija DW/BI sustava mora biti konzistentna. Podaci u sustavu moraju biti vjerodostojni, skupljeni iz različitih izvora, pročišćeni i provjereni prije nego se objavljuju korisnicima na korištenje. Konzistentnost također podrazumijeva zajedničke oznake i definicije sadržaja unutar sustava.[1]

Nadalje, DW/BI sustavi moraju biti prilagodljivi promjenama. Zbog toga jer se potrebe korisnika, uvjeti poslovanja i sama tehnologija stalno mijenjaju DW/BI sustavi moraju biti dizajnirani tako da se mogu relativno "bezbolno" mijenjati u skladu s njima bez da invalidiraju postojeće podatke ili aplikacije. Postojeći podaci i aplikacije ne smiju se mijenjati ili poremetiti ako se postave nova pitanja ili se dodaju novi podaci.[1]

Informacije se iz DW/BI sustava moraju prezentirati pravovremeno. Što se intenzivnije koristi sustav za operativne odluke tako raste potreba da se sirovi podaci trebaju pretvoriti u korisne informacije unutar nekoliko sati, minuta pa čak i sekundi. Ipak, DW/BI tim i poslovni korisnici trebaju imati realna očekivanja oko toga kakva će biti dostava podataka kad postoji malo vremena za čišćenje i validaciju istih.[1]

Sljedeće, DW/BI sustav mora biti sigurno mjesto koje čuva informacijsku imovinu organizacije. Zbog toga što se u skladištu podataka čuvaju vrijedne informacije vezane uz poslovanje (u najmanju ruku primjerice što se prodaje, kome i po kojoj cijeni) koje su potencijalno štetne u posjedu neželjenih ljudi, sustav mora efektivno kontrolirati pristup takvim povjerljivim podacima.[1]

DW/BI sustav mora služiti kao autoritativan i pouzdan temelj za donošenje poslovnih odluka. Skladište mora sadržavati prave podatke da bi moglo služiti kao kvalitetna podrška za donošenje odluka koje su najbitniji output DW/BI sustava. Te odluke se donose na temelju analitičkih dokaza kojeg ovi sustavi prezentiraju. Stariji naziv za DW/BI sustave zapravo najbolje opisuje njegovu svrhu: sustav za podršku odlukama (eng. *Decision support system*).[1]

I na kraju, poslovno okruženje (zajednica) mora prihvatiti DW/BI sustav da bi ga se moglo proglasiti uspješnim. Bez obzira na elegantnost i kvalitetu rješenja, korištenje najboljih alata i platforma, ako sustav nije prihvaćen on se ne može smatrati uspješnim. Za razliku od operativnih sustavskih rješenja koja korisnici moraju prihvatiti, DW/BI rješenja nisu nužna za korištenje i ne moraju biti prihvaćena od strane korisnika. Ali, ako je taj sustav brz i jednostavan izvor kvalitetnih informacija korisnici će ga bez problema prihvatiti i koristiti.[1]

Svaki prethodno nabrojen zahtjev je bitan, ali su posljednja dva kritično važna. Stoga je na ovom području bitno osim samog snalaženja u tehničkom dijelu (sustavska arhitektura, modeliranje, administracija baze podataka) potrebno i razumijevanje teritorija poslovnih korisnika.[1] Već iz zahtjeva je jasno da je izrada skladišta podataka koje se radi za korištenje u poslovnoj inteligenciji interdisciplinarno područje koje zahtijeva barem djelomično ili potpuno razumijevanje svih aspekata poslovanja, poslovnog okruženja te poslovnih i strateških ciljeva organizacije.

3.3. Životni ciklus razvoja skladišta podataka

Klasični životni ciklus razvoja sustava (eng. *System development life cycle*, SDLC) koji podržava operativno okruženje nije isti ciklus kao onaj koji se koristi kod razvijanja skladišta podataka. Taj stariji pristup često se naziva i vodopadnim jer su aktivnosti od kojih se sastoji točno specificirane, a završetkom jedne aktivnosti okida se početak sljedeće. SDLC linearno prolazi kroz korake prikupljanja zahtjeva, analize, dizajna, programiranja, testiranja, integracije

i implementacije. [5] Uz vodopadni pristup s godinama su se pojavile modernije metodologije poput Scrum-a i DevOps-a. Gotovo nasuprot SDLC ciklusu nalazi se životni ciklus razvoja skladišta podataka zbog čega se ponekad naziva CLDS (SDLC obrnuto).

Glavna obilježja ovog pristupa jesu:

- Ekstenzivno korištenje prototipiranja
- Veliki zadaci/aktivnosti mogu se pojavljivati u različitim redoslijedovima
- Nije nužno čekati završetak jednog zadatka/aktivnosti da se počne raditi na sljedećem
- Potrebno je usvojiti drugačiju filozofiju upravljanja projektom i način razmišljanja u odnosu na klasični pristup
- Potrebna je promjena kulture u poslovanju u odnosu na klasični pristup [2]

I dok je prvi korak klasičnog ciklusa razvoja prikupljanje zahtjeva koji onda pogone čitavi projekt, CLDS započinje s podacima. Jednom kad su podaci dostupni moraju se integrirati i testirati. Programi su napisani u skladu s njima i potom analizirani pa se tek onda otkrivaju i razumiju zahtjevi sustava. Potom se rade potrebne izmjene i popravci nad sustavom pa ciklus počinje iznova za drugačiji skup podataka. Stoga se ovakav način razvoja naziva i spiralnom metodologijom razvoja. [5]

Zbog činjenice da se zadaci ne odvijaju nužno u istom redoslijedu i da svačiji posao može početi na drugačijem dijelu spirale razvoja potrebno je posebnu pažnju posvetiti organiziranju razvojnog tima. Prirodno se grupiraju tri vrste posla: *back-end, front-end* i rad s metapodacima. Rad s metapodacima je vrlo ovisan o drugim paralelnim zadacima pa stoga vodstvo projekta mora moći prepoznati međuovisnosti zadataka i efektno njima upravljati. Cilj je spiralne metodologije proizvesti inventar ponovo iskoristive imovine. Zato je moguće da organizacije koristeći takav pristup izgrade sustav sa ponovo iskoristivim komponentama koje se mogu preslagivati i nadograđivati u skladu sa potrebama poslovanja. [2]

Ukratko, klasični se razvoj sustava vodi unaprijed određenim zahtjevima, dok se razvoj skladišta vodi podacima. [5]

3.4. Tehnološke predispozicije

Neke tehnološke značajke su potrebne za zadovoljavajuće procesiranje skladišta podataka. To uključuje robusno jezično sučelje, podršku složenim ključevima i podacima varijabilne duljine kao i mogućnost upravljanja velikom količinom podataka, upravljanja podacima na različitim medijima, lako indeksiranje i nadzor podataka, suradnja sa velikim brojem tehnologija, paralelno spremanje i pristup podacima, kontrola nad metapodacima skladišta, efikasno učitavanje skladišta, efikasno korištenje indeksa, pohrana podataka na kompaktan način, selektivno isključivanje upravitelja zaključavanja, izvršavanje *index-only* procesiranja i brzo uspostava (povratak) podatke iz pohrane. [5] Moderni sustavi zadovoljavaju sve ove zahtjeve.

Osim toga, arhitekt podataka mora prepoznati razliku između DBMS-a temeljenog na transakcijama i DBMS-a temeljenog na skladištu podataka. Dok se prvi bavi efikasnim izvršavanjem transakcija i ažuriranjem, drugi ima fokus na efikasnoj obradi upita i upravljanjem napora učitavanja i pristupa. [5]

3.5. Varijacije skladišta

Iako koncepti skladišta podataka nisu zamišljeni da se podlažu pojedinim prodavačima i organizacijama, na tržištu je došlo do nekih "mutacija" pa su se tako pojavila "aktivna" (eng. *Active DW*) skladišta podataka, "federativna" (eng. *Federated DW*) skladišta podataka, skladišta podataka na shemu zvijezde (eng. *Star scheme DW*) i *data mart* skladišta podataka (eng. *Data mart DW*). [2]

Aktivno skladište podataka je ono u kojem je moguće izvršavati online procesiranje i ažuriranje podataka. Procesiranje transakcija sa visokim performansama je glavna značajka ovakvog skladišta. Neke mane uključuju poteškoće u održavanju integriteta podataka i transakcija, potreba da se osigura dovoljan kapacitet, teže obavljanje složenijih statističkih operacija i veći sveukupni troškovi. [2]

Federativni pristup skladištima podataka je pristup koji zapravo nema skladišta podataka. Zamisao je da se spajanjem različitih starih operativnih podataka dobije mjesto iz kojeg bi se svim tim bazama moglo pristupiti istovremeno. Pristup je privlačan jer se čini kao da organizaciji daje mogućnost da se izbjegne integracija podataka. Nažalost, takav pristup je više iluzija jer ima mnoge temeljne nedostatke, a neki od njih su vrlo loše performanse, manjak integracije podataka, potreba za složenim tehničkim mehanikama, ograničeni povijesni podaci, manjak ponovo iskoristivih upita, nasljeđena zrnatost podataka i dr. [2]

Skladišta podataka sa shemom zvijezde podrazumijevaju kreiranje činjeničnih i dimenzijskih tablica. Neke od mana ovakvog pristupa su mogućnost težeg prilagođavanja novim zahtjevima, donekle ograničena proširivost, uža ciljana "publika" (odjel, skupina ljudi), potreba za više shema zvijezda ako se trebaju udovoljiti potrebe velikog broja korisnika i dr. [2] Ovaj pristup je s vremenom poprimio najviše pažnje i često se koristi. Detaljnije će se sagledati u sljedećim poglavljima.

Data mart skladišta podataka su ideja da se napravi više različitih *data mart*-ova koji se onda nazivaju skladištem podataka umjesto da postoji "pravo" skladište. Takav pristup omogućuje da se poslovi organizacije odvijaju (npr. proizvodi se prodaju) bez procesa izgradnje čitavog skladišta. Problemi takvog pristupa su mnogobrojni, a uključuju: mogući nesklad u podacima (jer svaki odjel ima svoj data mart, odgovor na pitanje "Koliko je iznosila prodaja prošli mjesec?" bi mogao biti različit ovisno o tome koga pitate), broj ekstrahiranja podataka raste s brojem *data mart*-ova, propagacija potrebnih promjena (ako je potreba promjena jednog *data mart*-a onda je vjerojatno potrebno mijenjati sve ili više njih) i neproširivost (kad se mora izgraditi novi data mart kreće se otpočetka). [2]

3.6. Komponente u okruženju skladišta podataka

3.6.1. ODS

ODS (eng. *Operational data store*) je mjesto gdje se odvijaju online ažuriranja integriranih podataka sa OLTP (eng. *Online transaction processing*) vremenima odgovora. To je hibridno okruženje u kojem se aplikacijski podaci pretvaraju u integrirani format. Podaci u ODS-u su dostupni za procesiranje visokih performansi uključujući procesiranje ažuriranja. [2]

3.6.2. Data Mart

Data mart je mjesto gdje krajnji korisnik ima direktan pristup i kontrolu nad svojim analitičkim podacima. On se oblikuje prema skupu korisnikovih očekivanja o tome kako podaci trebaju izgledati, a često se grupiraju prema odjelu kojem korisnici pripadaju. Primjerice, marketinški odjel ima svoj *data mart*, računovodstvo ima svoj *data mart* itd. Izvor podataka za svaki data mart jest skladište podataka. *Data mart*-ovi uobičajeno sadrže značajne količine sumariziranih i agregiranih podataka. [2]

3.6.3. Istraživačko skladište

U istraživačko skladište (eng. *Exploration warehouse*) se krajnji korisnici dolaze baviti istraživačkim procesiranjem. Mnoge statističke analize se odvijaju ovdje. Većina ovakvih skladišta čuva podatke organizirane po projektima. Kad se projekt završi, briše se i istraživačko skladište. Ono preuzima na sebe iscrpno procesiranje potrebno za složenije statističke obrade čime "čuva" skladište podataka od takvih zadataka. [2]

3.6.4. ETL

ETL (eng. *Extract, transformation, load*) sustav DW/BI okruženja se sastoji od radnog prostora, instanciranih struktura podataka i skupa procesa. ETL sustav je sve između operativnih sustava kao izvora podataka i područja DW/BI prezentacije. [1]

ETL komponenta izvršava više funkcija u okruženju skladišta podataka, kao što su konverzija podataka, verifikacija domena, konverzija iz jednog sustava za upravljanje bazama podataka u drugi, kreiranje zadanih vrijednosti ako je to potrebno, agregacija podataka, dodavanje vremenskih vrijednosti ključu podataka, restrukturiranje ključa podataka, spajanje zapisa, brisanje nebitnih ili suvišnih podataka. [2] Pogreške u ovoj fazi izrade skladište podataka propagiraju se u kasnije faze što znači da primjerice greška u obradi podataka rezultira netočnim podacima koji onda završe na nekom izvještaju [3] zbog kojeg se onda može donijeti kriva odluka s menadžerske strane.

Ekstrakcija je dakle prvi korak u prebacivanju podataka u okruženje skladišta podataka. Pod ekstrakcijom se podrazumijeva čitanje i razumijevanje izvornih podataka i njihovo kopiranje u ETL sustav za daljnju manipulaciju. [1]

Nakon ekstrakcije podataka u ETL sustav slijedi transformacija koja može poprimiti razne oblike kao primjerice čišćenje podataka (ispravci pogrešno napisanih riječi/slova, rješavanje sukoba domena, bavljenje elementima koji nedostaju, parsiranje u standardne formate. . .), spajanje podataka iz više izvora i dr. Na taj se način podacima daje dodatna vrijednost. Dodatno, ove aktivnosti mogu stvarati i dijagnostičke metapodatke koji mogu voditi do eventualnog reinžinjeringa poslovnih procesa za poboljšanje kvalitete podataka u izvorima. [1]

Posljednji korak ETL procesa je fizičko strukturiranje i učitavanje podataka u dimenzijske modele ciljanog prezentacijskog područja. Kad se dimenzijske i činjenične tablice u dimezijskom modelu ažuriraju, indeksiraju, dodaju im se prikladni agregati i osigura se kvaliteta poslovnoj zajednici se javlja da su objavljeni novi podaci. [1]

Postoje slučajevi u kojima podaci stižu na prag ETL procesa u 3NF relacijskom formatu. U tim situacijama lakše je koristiti normalizirane strukture za čišćenje i transformaciju podataka. Iako je takav pristup prihvatljiv on podrazumijeva da se podaci potencijalno ekstrahiraju, transformiraju i učitavaju dvaput, jednom u normaliziranu bazu podataka i jednom u dimenzijski model. Takav proces dakle zahtijeva više vremena i ulaganja u razvoj, više vremena za periodičko učitavanje ili ažuriranje podataka, kao i veći kapacitet za čuvanje više kopija podataka. Ako u organizaciji ne postoje normalizirane tablice za korištenje u ETL sustavu onda postoje jednostavniji pristupi s manje troškova ako se podaci trebaju koristiti u DW/BI okruženju. Kreiranje normalizirane baze podataka kao podrška ETL-u je prihvatljivo rješenje, ali nije krajnji cilj i ona mora biti izvan dosega upita korisnika jer se inače krši originalni ciljevi razumljivosti i performanse DW/BI sustava. [1]

3.7. Dimenzijsko modeliranje

Dimenzijsko modeliranje (eng. *Dimensional modeling*) je općeprihvaćena tehnika za prezentiranje analitičkih podataka zbog toga jer istovremeno ispunjava dva zahtjeva, a to su dostavljanje podataka razumljivih korisniku i ostvarivanje brzih performansi upita. Ovakvo modeliranje je dugotrajna tehnika koja održava baze podataka jednostavnim što u produžetku omogućava lakše razumijevanje podataka i omogućava da softver brzo i efikasno navigira kroz podatke i dostavlja rezultate. [1]

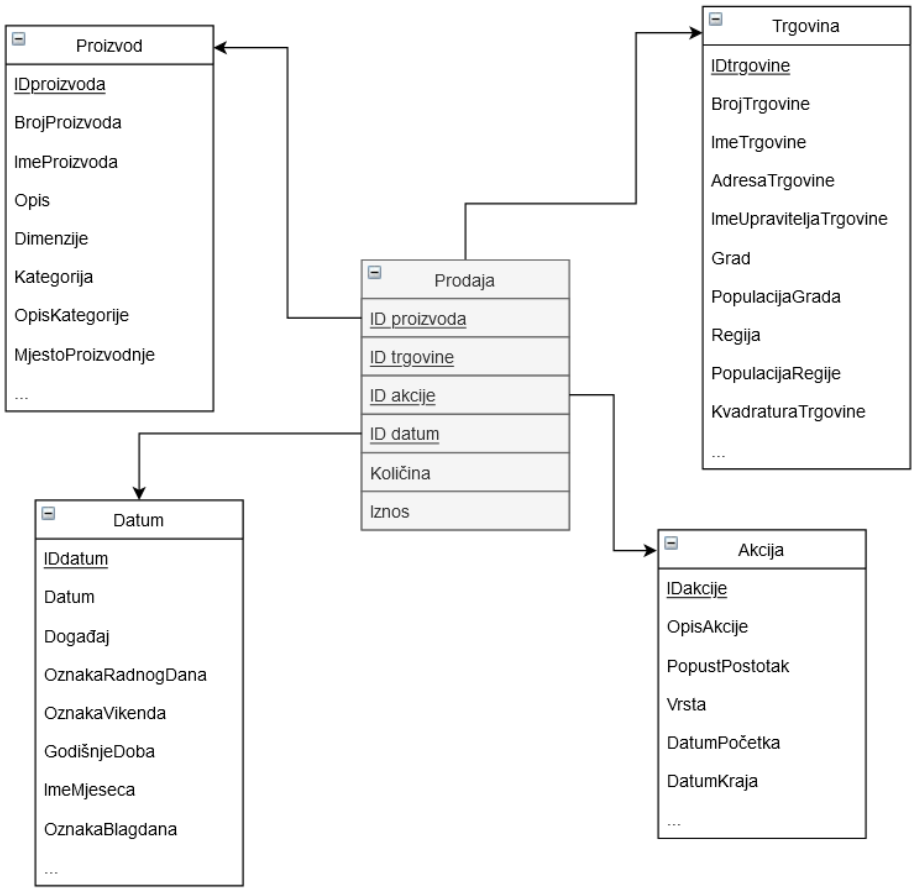
Iako se dimenzijski modeli često instanciraju u sustavima za upravljanje relacijskim bazama podataka oni se poprilično razlikuju od klasičnih modela u trećoj normalnoj formi (3NF) (eng. *Third normal form models*) koji teže uklanjanju redundacije podataka. Takvi modeli se još nazivaju ER (eng. *entity -relationship*, entitet - veza) modelima koji se prikazuju pomoću ERD-ova (eng. *ER Diagram*). Pomoću takvih dijagrama prikazuju se relacijske tablice i veze između njih. Ključna razlika između 3NF modela i dimenzijskih modela jest stupanj normalizacije. I dok su 3NF strukture izrazito korisne kod operativnih procesa, one su neefikasne za upite kakve traži poslovna inteligencija. [1]. Takvi upiti traže mnogo agregacija i sličnih operacija nad podacima što često znači prolaz kroz sve zapise baze [4] što nije optimalno. Složenost i nepredvidljivost korisničkih upita je prevelika za većinu sustava za upravljanje bazama podataka što rezultira lošim performansama upita. Nadalje, korisnici teže razumiju, navigiraju i pamte

takve složenije modele. Ovo su razlozi koji objašnjavaju kako je korištenje normaliziranih modela na području DW/BI suprotno samoj svrsi takvih modela koji služe za intuitivno prikupljanje i prikazivanje podataka s visokim performansama. Dobro je naglasiti i da dimenzijski modeli sadrže iste informacije kao normalizirani modeli, ali su podaci su strukturirani u formatu koji omogućava razumljivost korisnicima, dobre performanse upita i otpornost na promjene. [1]

Dimenzijski modeli implementirani u sustavima za upravljanje relacijskim bazama podataka se nazivaju shemama zvijezde zbog svoje zvjezdaste strukture. Dimenzijski modeli implementirani u multidimenzionalnim okruženjima baza podataka se nazivaju OLAP kockama (eng. *Online analytical processing cubes*). Oba ovakva pristupa imaju zajednički logički dizajn s prepoznatljivim dimenzijama, a razlika je u fizičkim implementacijama. [1]

OLAP (eng. *online analytical processing*) je paradigma specijalno orijentirana prema analizi podataka u organizacijskim bazama podataka za podršku odlučivanju. One su fokusirane na analitičke upite, a baze podataka usmjerene OLAP-u moraju podržavati velika opterećenja kod upita. Pojava potrebe za novom vrstom baze podataka za podršku OLAP-u dovela je do ideje o skladištima podataka. Ona se mogu dizajnirati i optimizirati za podršku OLAP upitima, ali i drugim analitičkim zadacima poput izvještavanja, rudarenja podacima i statističke analize.[4] Kad se podaci učitavaju u OLAP kocku oni se spremaju i indeksiraju korištenjem formata i tehnika specijalno dizajniranih za dimenzijske podatke. Agregacije performansi i pretkalkulirane sumacije tablica su kreirane od strane OLAP stroja. Zbog takvih pretkalkulacija, indeksiranja i sličnih optimizacija OLAP kocke nude bolje performanse upita nego skladišta sa shemom zvijezde iako su se naglim razvojem hardvera kroz godine razlike performansi značajno smanjile. [1]

Ipak, ovo poglavlje će biti posvećeno shemi zvijezde. Ono je dakle relacijska reprezentacija multidimenzionalnog modela u kojoj postoji jedna središnja činjenična tablica i skup dimenzijskih tablica (primjer na slici 2). U takvom modelu dimenzijske tablice općenito nisu normalizirane i stoga mogu sadržavati redundante podatke, pogotovo u prisutnosti hijerarhija [4] što je i prikazano u sljedećim potpoglavljima.



Slika 2: Primjer sheme zvijezde (prema Vaisman, 2014)

3.7.1. Činjenične tablice

Činjenična tablica (eng. *Fact table*) dimenzijskog modela čuva mjerenja performansi dobivenih odvijanjem poslovnih procesa organizacije. Podaci takvih mjerenja je uvjerljivo najveći skup podataka skladišta i zbog toga se ne bi smio replicirati na više mjesta. Dozvoljavanje poslovnim korisnicima pristup jednom centraliziranom repozitoriju za svaki skup podataka dobivenih mjerenjem osigurava konzistentnost podataka u poduzeću. Termin činjenica u nazivu ovih tablica se odnosi na mjeru poslovanja. Primjerice, ako bi se kod prodaje nekog proizvoda zapisavala količina proizvoda i iznos u valuti to bi bila mjerenja poslovanja koja bi se zapisivala u činjeničnu tablicu. Svaki red činjenične tablice korespondira s događajem koji se mjeri. Podaci svakog reda na određenoj razini detalja nazivaju se zrno (eng. *grain*) iz čega proizlazi bitan pojam zrnatosti (eng. *granularity*) dimenzijskog modela [1] kojemu je posvećeno kasnije poglavlje. Primjerice, razina detalja bi mogla biti jedan red po prodanom proizvodu u nekoj transakciji. Jedno od temeljnih načela dimenzijskog modeliranja jest da svi redovi činjenične tablice moraju imati istu zrnatost. Još jedan temeljni princip za dimenzijsko modeliranje jest ideja da mjerljivi događaj u stvarnom svijetu ima jedan na jedan vezu (eng. *one-to-one*) prema jednom redu činjenične tablice. [1]

Najkorisnije činjenice jesu one koje su numeričke i aditivne, primjerice iznos u valuti. Aditivnost je bitna jer aplikacije poslovne inteligencije rijetko koriste pojedinačne redove činje-

nične tablice već vraćaju stotine, tisuće ili milijune redova odjedanput stoga su najkorisniji rezultati najčešće zbrojeni podaci tih redova, na ovaj ili onaj način. Ipak, činjenice nekad mogu biti poluaditivne ili neaditivne. Poluaditivna činjenica je primjerice stanje računa koje se ne može zbrajati kroz vremensku dimenziju. Primjer neaditivne činjenice je jedinična cijena proizvoda za koje smo primorani koristiti prosjeke ili prebrojavanja. Teoretski je moguće da činjenica bude tekstualna iako se vrlo rijetko dogodi da postoji potreba za takvom činjenicom. U većini slučajeva tekstualna činjenica je opis nečega i izvlači se iz diskretne liste vrijednosti. Dizajner modela bi se trebao truditi da se tekstualni podaci stave u dimenzije (dimenzijske tablice) gdje se mogu efektivnije korelirati sa drugim tekstualnim atributima zauzimajući pritom manje mjesta. Redundatne tekstualne informacije ne smiju se nalaziti u činjeničnoj tablici. [1]

Neaktivnost se u činjeničnim tablicama ne označava nulama jer bi se tablica lako zasula nepotrebnim podacima. Unatoč tome, činjenične tablice obično zauzimaju 90-ak i više posto ukupnog prostora kojeg zauzima dimenzijski model. One imaju veliki broj redova, a manji broj stupaca. [1]

Sve činjenične tablice imaju dva ili više vanjskih ključeva kojima se spajaju na primarne ključeve dimenzijskih tablica. Jednostavan primjer je da se numerička oznaka proizvoda u redu činjenične tablice veže na jedinstvenu numeričku oznaku proizvoda u tablici proizvoda. Primarni ključ činjenične tablice je obično podskup vanjskih ključeva te tablice. Takav primarni ključ se naziva složenim ključem (eng. *Composite key*). Svaka tablica sa složenim ključem je činjenična tablica. Činjenične tablice dakle predstavljaju veze više na više (eng. *many-to-many*). Sve ostalo su dimenzijske tablice. [1]

3.7.2. Dimenzijske tablice

Dimenzijske tablice skladišta podataka sadrže tekstualni kontekst vezan uz mjereni poslovni događaj. Daju odgovore na pitanja tko, što, gdje, kada, kako i zašto vezana na sam poslovni događaj. [1] Dimenzije su perspektive koje se koriste za analizu podataka, a za svaku od njih trebala bi postojati jedna dimenzijska tablica.[4]. Dimenzijske tablice ne sadrže toliki broj redova kao činjenične tablice, ali su puno šire, odnosno imaju mnogo više atributa. Svaka dimenzijska tablica ima jedan jednostavni primarni ključ koja služi kao temelj za održavanje referencijalnog integriteta s bilo kojom činjeničnom tablicom s kojom je spojena. [1]

Attributi dimenzijskih tablica služe kao glavni izvor ograničenja nad upitima, grupiranja i oznaka u izvještajima. Zbog toga su dimenzijske tablice kritičan faktor u tome da se DW/BI sustav učini upotrebljivim i razumljivim. Atributi bi stoga trebali biti “prave” riječi, a ne skraćenice, akronimi, numeričke oznake i sl. Ako se u operativnom dijelu koriste kodovi potrebno je standardizirati dekodiranje za dimenzijske attribute kako bi se postigla konzistencija oznaka prilikom upita, u izvještajima i aplikacijama poslovne inteligencije. Ponekad operacijski kodovi imaju posebnu ulogu poslovanju, primjerice prvih nekoliko znamenki numeričke oznake označavaju mjesto proizvodnje, drugih nekoliko model proizvoda i sl. U tom slučaju se ta značenja moraju “izvaditi” iz oznake i prezentirati kao zasebni dimenzijski atributi koji se onda mogu lakše filtrirati, grupirati i prikazati umjesto da da korisnici sami pregledavaju ili filtriraju prema podnizu originalne oznake. [1]

Može se dakle reći da je skladište podataka onoliko dobro koliko su mu dobri atributi dimenzijskih tablica - analitička snaga DW/BI okruženja je direktno proporcionalna kvaliteti dimenzijskih atributa. [1]

Kad gradimo dimenzijski model prema dostupnim operativnim podacima, važna je i ponekad nejasna odluka radi li se o činjenici ili dimenzijskom atributu kad se radi o numeričkom podatku. Ako je stupac (točnije vrijednosti stupca) mjera koja poprima mnogo različitih vrijednosti i sudjeluje u kalkulacijama onda se zasigurno radi o činjenici. Ako je podatak više ili manje konstantan ili poprima vrijednosti iz diskretne liste i sudjeluje u ograničenjima upita i oznakama redova onda je to dimenzijski atribut. [1]

Dimenzijske tablice često predstavljaju hijerarhijske veze.[1] Hijerarhija sadrži nekoliko povezanih razina, niže se nazivaju djeca, a više roditelji. Tako se veze koje grade hijerarhiju nazivaju veze dijete-roditelj. Kardinalnost tih veza ukazuje na minimalni i maksimalni broj članova jedne razine koje može biti spojen na člana druge razine.[4] Primjerice, proizvodi imaju svoj opis, spadaju u neki brend proizvoda pa zatim u neku kategoriju proizvoda (primjer u tablici 1). U tom primjeru, dijete je "*Proizvod*", a roditelj "*Brend proizvoda*" i dijete je povezano na roditelja s kardinalnosti jedan na više (eng. *one-to-many*) što znači da dijete (proizvod) pripada samo jednom roditelju (brendu), ali roditelj (brend) može imati više djece (proizvoda).

Ovakve deskriptivne hijerarhijske informacije se spremaju u svrhu lakoće korištenja i performansi upita. Ovaj pristup je suprotan “klasičnom” građenju modela podataka u kojem se oni nastoje normalizirati i u kojem bi se primjerice napravile zasebne tablice za brendove proizvoda i za kategorije proizvoda. Zbog toga jer su dimenzijske tablice uobičajeno geometrijski manje od činjeničnih, ušteda prostora koja bi se postigla tim pristupom nema gotovo nikakav učinak na sveukupnu veličinu baze (skladišta) podataka. Isto tako se gotovo uvijek treba odabrati jednostavnost i pristupačnost ispred uštede prostora na ovaj način. [1]

Tablica 1: Primjer hijerarhijskih veza u dimenzijskoj tablici

ID proizvoda	**Proizvod**	**Brend proizvoda**	**Kategorija proizvoda**
1	Keksi čokolada 250g	Keksi	Slatkiši
2	Keksi čokolada 500g	Keksi	Slatkiši
3	Keksi čokolada 1000g	Keksi	Slatkiši
4	Keksi malina 250g	Keksi	Slatkiši
5	Keksi malina 500g	Keksi	Slatkiši
6	Čokolada lješnjak 90g	Čokolada	Slatkiši

3.8. Zrnatost

Važan aspekt dizajniranja skladišta podataka jest problem zrnatosti (eng. *granularity*). On prožima cijelu arhitekturu koja okružuje okoliš skladišta podataka. Zrnatost predstavlja razinu detalja ili sumarizacije nad jedinicama podataka u skladištu podataka.[5] Ono dozvoljava podacima da budu fleksibilni. [2] Što je više detalja to je niža razina zrnatosti. Primjerice, jednostavna transakcija ima nisku razinu zrnatosti dok sumarizacija svih transakcija u nekom određenom mjesecu ima visoku razinu zrnatosti.[5]

U ranijim operativnim sustavima zrnatost nije bila ozbiljno razmatrana. Kad su se ažurirali detaljni podaci gotovo se podrazumijevalo da oni budu na najnižoj razini zrnatosti. Ali kod skladišta podataka zrnatost se ne podrazumijeva već je jedna od kritičnih odluka kod dizajna. Razlog tome je zato što duboko utječe na volumen podataka koji će se nalaziti u skladištu i vrste upita na koje će skladište moći dati odgovor. Te su dvije stvari obrnuto proporcionalne. Niža razina zrnatosti znači mogućnost obavljanja svestranijih upita, a snižavanje razine zrnatosti smanjuje i svestranost potencijalnih upita.[5] Stoga Kimball i Ross [1] tvrde da se činjenične tablice moraju graditi na najnižoj razini zrnatosti za maksimalnu fleksibilnost i proširivost.

U gotovo svim slučajevima podaci koji stižu u skladište su na previsokoj razini zrnatosti. Ovo znači da developer mora utrošiti resurse dizajna i razvoja na razdvajanje podataka prije nego se oni mogu spremiti u skladište. Ipak, ponekad podaci stignu na preniskoj razini zrnatosti. Primjer toga je dnevnik Web podataka generiran od strane Web okruženja e-poslovanja (nazivaju se i *clickstream podaci*). Takav dnevnik, odnosno podaci u njemu se moraju urediti, filtrirati i sumarizirati prije nego dosegnu razinu zrnatosti prikladnu za spremanje u skladište podataka.[5]

3.8.1. Prednosti zrnatosti

Gradnja skladišta podataka nije jednostavno i može biti vrlo skupo, ali se mora izgraditi samo jednom, a kad se izgradi pruža temelj koji je ponovo iskoristiv i izrazito fleksibilan. Upravo su zrnati podaci ključ ponove iskoristivosti jer se takvi mogu koristiti na različite načine od strane različitih ljudi. Unutar organizacije se isti podaci mogu koristiti i zadovoljiti potrebe marketinga, prodaje, računovodstva itd. Svaki odjel gleda na iste bazične podatke, ali prodaju primjerice zanima iznos prodaje prema prodavaču, marketing iznos prodaje prema geografskom području, a računovodstvo iznos prodaje prema liniji proizvoda i sl. Skladište podataka im to omogućuje i dozvoljava da se na podatke gleda onako kako se želi. [5]

Još jedna prednost stvaranja ovakvog temelja jest da se eventualni nesklad jest da se lako može objasniti eventualni nesklad u analizama jednog ili više odjela, ako za tim postoji potreba. [5]

Fleksibilnost je još jedna stvar koju pruža niska razina zrnatosti. U slučaju da jedan odjel želi promijeniti način na koji gleda na podatke to se može postići jednostavno kad postoji temelj od zrnatih podataka. [5]

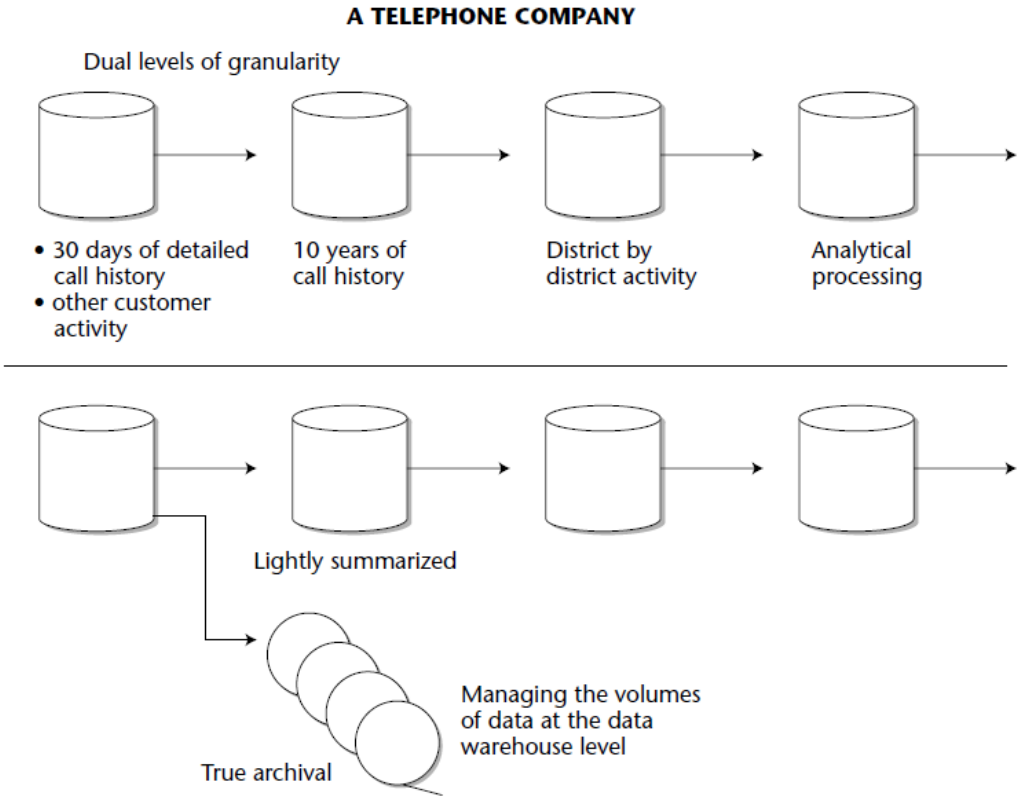
Čuvanje zrnatih podataka također znači da organizacija ima povijesne zapise o aktiv-

nostima i događajima. [5]

Vjerojatno najveća dobrobit koja proizlazi iz temelja u obliku skladišta podataka sa podacima određene zrnatosti jest sposobnost da se mogu ispuniti budući nepoznati zahtjevi. Ako postoji novi zahtjev za pogled nad podacima, donese se novi zakon, mijenjaju se pravila organizacije i sl., moguće je jednostavnije reagirati na takve promjene i na njih se prilagoditi. Uslijed novog zahtjeva i potrebe za informacijama skladište podataka je već spremno za analizu čime je organizacija pripremljena da barata s tim novim zahtjevima. [5]

3.8.2. Dvostruka razina zrnatosti

U većini slučajeva postoji potreba za efikasnošću spremanja i pristupanja podacima, ali i za sposobnost detaljnog analiziranja podataka. Zbog toga ima smisla razmatrati dvije ili više razine zrnatosti kad organizacija ima mnogo podataka u skladištu. Većinom je zapravo slučaj takav da bi dvostruka razina zrnatosti trebala biti zadana vrijednost za bilo koje poslovanje. Dobar primjer ovog bila bi telefonska kompanija (slika 2). Kod nje postoji velika količina detalja u operativnim podacima, a većina ih je potrebna u sustavima za naplaćivanje. Do trideset dana detalja se sprema na operativnoj razini.[5]



Slika 3: Primjer dvostruke zrnatosti (Izvor: Inmon, 2005)

Skladište podataka bi u ovom primjeru sadržavalo dvije vrste podataka - slabo sumarizirani podaci i “pravi arhivski”, detaljni podaci. Podaci u skladištu podataka čuvaju se i do deset godina, a oni iz skladišta idu prema različitim distriktima kompanije. Svaki distrikt analizira po-

datke neovisno o drugima. Lagano sumarizirani podaci su detaljni podaci obrađeni u malom obujmu, npr. informacije o pozivu jednog klijenta na mjesečnoj bazi kao što su prosječna dužina poziva mjesečno, broj poziva mjesečno i dr. Tako se postiže značajno manja količina podataka nego što ih ima u detaljnoj bazi, ali naravno postoji manja razina detalja kojoj se može pristupiti u tako sumariziranoj bazi podataka.[5]

Drugi red podataka u skladištu je onaj na najnižoj razini zrnatosti i sprema se na arhivskoj razini. Na toj razini se spremaju detalji koji dolaze iz operativnog okruženja. Kreiranjem dvije razine zrnatosti u istom skladištu postignute su dvije stvari kod potpore odlučivanju. Prva je da se većina procesiranja obavlja nad sumariziranim podacima koji su kompaktni i može im se efikasno pristupiti. Druga je da se u malom broju slučajeva (otprilike pet posto vremena ili manje) u kojima je potrebno pristupati većoj razini detalja to može napraviti. Iako je skuplje, sporije i složenije, ako se podacima mora pristupati na arhivskoj razini zrnatosti takva mogućnost postoji. Ukoliko traženje po najvišoj razini zrnatosti postane prečesto onda dizajner treba razmisliti o kreiranju novih polja podataka na sumariziranoj razini da se procesiranje prebaci na tu razinu.[5]

Zbog manjih troškova, efikasnosti, lakoće pristupa i mogućnosti odgovora na gotovo sve upite dvostruka zrnatost podataka je najbolji arhitekturalni izbor za detaljnu razina podataka u skladištu za većinu poslovanja. Jednostruku razinu jedino valja razmatrati ako se radi o relativno maloj količini podataka u okruženju skladišta podataka.[5]

4. Poslovna inteligencija

U tipičnoj tvrtki ili organizaciji mnogo informacija do menadžera dolazi u obliku statičnih izvještaja iz *ad hoc* prikupljanja i analize podataka. Tako bi primjerice viši menadžer dobio mjesečnu izjavu o profitu i gubicima, vidio da je prihod manji od predviđenog pa bi zatim morao okupiti ljude za analizu zaposlenika, srednji menadžment ili nekog funkcionalnog menadžera da se utvrde faktori koji su doprinjeli toj razlici. Takav način analize i izvori informacija su najvjerojatnije *ad hoc* i idiosinkratični. Malo je prilike i vremena za ekstenzivniju analizu scenarija i procjenjivanje alternativnih puteva postupanja. [6] Ovakav pristup je zastario i ne zadovoljava potrebe menadžmenta za kvalitetnom podrškom odlučivanju.

Druga mogućnost je da organizacija uloži u aplikaciju **poslovne inteligencije**. Takva aplikacija bi mogla gledati trendove prihoda po kupcu, po geografskoj regiji, po proizvodu i po prodavaču. Bila bi izdana na širini cijele organizacije, s online podukama o načinima korištenja aplikacije. Takva bi aplikacija uvelike olakšala i pojednostavila gore opisani postupak analize. [6] Njom bi se stvorila mogućnost analize trendova, uočavanja ključnih faktora poslovanja, ekstenzivnije analize scenarija i sl. čime se postigla bolja podrška kod donošenja menadžerskih odluka.

Poslovna inteligencija (BI, eng. *Business Intelligence*) je termin koji objedinjuje arhitekture, alate, baze podataka, analitičke alate, aplikacije i metodologije. [7] Poslovna inteligencija spaja produkte, tehnologije i metode da organizira ključne informacije potrebne menadžmentu za poboljšanje performansi i profita organizacije. Opširnije rečeno, poslovna inteligencija su poslovne informacije i poslovne analize u kontekstu ključnih poslovnih procesa koje vode prema odlukama i postupcima u svrhu poboljšanja poslovnih performansi. [6]

Poslovna inteligencija nije:

- Jedan proizvod; iako se razni proizvodi mogu koristiti za implementaciju poslovne inteligencije, ona sama nije proizvod koji se može kupiti i instalirati da se lagano riješe svi povezani problemi
- Tehnologija; iako se alati za skladišta podataka i tehnologije poput relacijskih baza podataka, alata za ETL, alati korisničkih sučelja poslovne inteligencije i poslužitelji koriste kao podrška poslovnoj inteligenciji, ona sama nije tehnologija. Implementiranje poslovne inteligencije zahtijeva tehnologiju, ali ona sama po sebi nije dovoljna. Uvođenje potrebne tehnologije bez uvođenja potrebnih promjena poslovnih procesa koji bi ju iskoristili nije se napravila nikakva promjena u organizaciji.
- Metodologija; snažna metodologija je potrebna za uspjeh sa poslovnom inteligencijom ali ju je potrebno spojiti sa prikladnim tehnološkim rješenjima i organizacijskim promjena za implementiranje cjelokupne poslovne inteligencije [6]

Glavni cilj poslovne inteligencije je omogućiti interaktivni pristup podacima, omogućiti manipulaciju podataka i dati poslovnim menadžerima i analitičarima sposobnosti obavljanja prikladnih analiza. Analiziranjem povijesnih i sadašnjih podataka, situacija i performansi do-

nositelji odluka dobivaju vrijedan uvid u poslovanje iz kojeg mogu donositi informiranije i bolje odluke. Proces poslovne inteligencije se temelji na transformaciji podataka u informacije, informacija u odluke i potom odluka u djelovanje. [7] Najjednostavnije rečeno, poslovna inteligencija je korištenje podataka za donošenje boljih odluka. U kontekstu poslovanja, bolje odluke su one koje će ga učiniti efektivnijim, efikasnijim i profitabilnijim. [8]

Potraga za ostvarivanjem dodatne vrijednosti organizaciji uvođenjem poslovne inteligencije se svodi na načine na koje organizacija može koristiti poslovnu inteligenciju za

- Poboljšanje procesa menadžmenta (planiranje, kontrola, mjerenje, promatranje, uvođenje promjena) kako bi se mogli povećati prihodi, smanjiti troškovi ili oboje
- Poboljšanje operativnih procesa (primjerice detekcija prevara, izvršavanje kampanja prodaje, procesiranje narudžbi kupaca, kupovinu i dr.) kako bi se mogli povećati prihodi, smanjiti troškovi ili oboje [6]

Ukratko, poslovna vrijednost poslovne inteligencije leži u njezinom korištenju kod menadžerskih procesa koji utječu na operativne procese ili kod samih operativnih procesa. [6]

Ipak, da bi uvođenje poslovne inteligencije (tj. aplikacije poslovne inteligencije) u poslovanje bilo uspješno mora postojati jasan plan kako bi se ona koristila unutar zajednice korisnika i plan za provođenje promjena i osiguravanje efikasnosti tih promjena potrebnih za ostvarivanje poslovne vrijednosti aplikacije. Drugim riječima, potrebno je imati ideje za strateško poravnanje (eng. *Strategic alignment*) [6]. Ključne stvari za ovaj proces jesu:

- Razumijevanje strateških pogonitelja i ciljeva organizacije
- Određivanje poslovnih pitanja na koje poslovna inteligencija mora odgovoriti kako bi se postigli ti ciljevi
- Identifikacija alata, metoda i analitičkih okvira za podršku odlukama i mjerenje performansi
- Dostavljanje informacija koje su potrebne organizaciji da poduzme akcije za poboljšanje perfomansi i podršku ciljevima organizacije [6]

4.1. Kratki povijesni pregled

Naziv BI (eng. *Business Intelligence*) smislila je tvrtka *Gartner Group* sredinom 1990-ih, iako je sam koncept mnogo stariji. Svoje korijene ima u MIS-ovima (eng. *Management Information System*) iz 1970-ih. Oni su bili statični, dvodimenzionalni i bez analitičkih sposobnosti. Početkom sljedećeg desetljeća rodio se koncept EIS-a (eng. *Executive Information System*) s poboljšanim mogućnostima poput dinamičkog višedimenzionalnog (*ad hoc* ili na zahtjev) izvještavanja, predviđanja, analize trendova, *drill-down* do detalja, pristupa statusu i kritičnim faktorima uspjeha. Ove značajke pojavljivale su se u mnogim proizvodima do sredine 1990-ih. Potom su se one, zajedno s nekim novima počele nazivati poslovnom inteligencijom. [7]

Današnji sustavi poslovne inteligencije sadrže sve informacije potrebne za menadžment. Razni alati i tehnike mogu biti uključeni u sustave poslovne inteligencije, a neki od njih su ETL, EIS, OLAP, rudarenje podacima i tekstom, prediktivne analize, proračunske tablice, DSS i dr. Najsofisticiraniji BI proizvodi sadrže mnogo ovih sposobnosti dok se drugi specijaliziraju samo na neke. [7]

4.2. Područja poslovne inteligencije

Deckler u svojoj knjizi [8] ugrubo je podijelio ključne koncepte poslovne inteligencije na pet područja. Ona su domena, podaci, model, analiza i vizualizacija.

4.2.1. Domena

Domena je jednostavno kontekst unutar kojeg se primjenjuje poslovna inteligencija. Većina poslovanja sastoji se od relativno standardnih poslovnih funkcija ili odjela poput prodaje, marketinga, proizvodnje, logistike, ljudskih resursa i sl. Svaka od njih predstavlja jednu domenu unutar koje poslovna inteligencija može biti korištena za odgovaranje na pitanja koja pomažu u donošenju boljih odluka. Domena pomaže u sužavanju fokusa na koja pitanja se mogu odgovoriti i koje odluke se moraju donositi. Određivanje domene u kojoj će se primijeniti poslovna inteligencije je ključan korak zato jer diktira na koja će se ključna pitanja odgovarati, potencijalne koristi, ali i koji podaci su potrebni za te odgovore. [8]

4.2.2. Podaci

Jednom kad se odluči o domeni potrebno je identificirati i pribaviti podatke primjerene toj domeni. To podrazumijeva identifikaciju izvora relevantnih podataka. Oni mogu biti interni ili eksterni s obzirom na organizaciju i mogu biti strukturirani, nestrukturirani ili polustrukturirani po prirodi. [8]

Interni podaci su oni koji se generiraju unutar same organizacije od strane njezinih procesa i operacija. Eksterni su dakle oni koji se generiraju izvan granica operacija organizacije, poput globalnog ekonomskog stanja vrste poslovanja kojim se bavi organizacija, cijene konkurenata i sl. Poslovna inteligencija je najefektnija kad se koristi kombinacija internih i eksternih podataka.[8]

Strukturirani podaci su podaci koji su u skladu formalnih specifikacija tablica sa redovima i stupcima. Izvori strukturiranih podataka su najlakši za alate poslovne inteligencije u smislu primanja i analize podataka. Strukturirani izvori su najčešće relacijske baze podataka poput *Microsoft SQL Servera*, *Azure SQL Database*-a, *Oracle*-a i dr. [8]

Nestrukturirani podaci su suprotno od strukturiranih, oni koji se ne mogu organizirati u tablice, poput audia, videa, slika i sl. Oni su najteži za učitavanje i analizu u alatima poslovne inteligencije. Takvi podaci također uključuju NoSQL baze podataka poput *MongoDB*-a, *Azure Cosmos DB*-a, *Neo4j*-a, *DynamoDB*-a i dr.[8]

Polustrukturirani podaci imaju strukturu ali ona nije u formalnom obliku kao strukturirani podaci, u obliku tablice s redovima i stupcima. Polustrukturirani podaci uključuju razgraničene (delimitirane) tekstualne datoteke, XML i druge jezike za označavanje poput HTML-a i XSL-a, JSON-a (eng. *JavaScript Object Notation*) i EDI (eng. *Electronic Data Interchange*). Polustrukturirani podaci uključuju i protokole za pristup podacima, poput OData protokola (eng. *Open Data Protocol*) i druge REST (eng. *Representational State Transfer*) API-e.[8]

Velika većina alata poslovne inteligencije su optimizirani za rukovanje strukturiranim i polustrukturiranim podacima. Dodatno, dizajnirani su tako da zaprimljene polustrukturirane podatke transformiraju u strukturirane.[8]

4.2.3. Model

Model podataka se odnosi na način na koji se jedan ili više izvora podataka organizira tako da podržava analizu i vizualizaciju. Modeli se grade transformiranjem i pročišćavanjem podataka pri čemu pomažu u definiranju tipa podataka unutar tih izvora i definiraju kategorije podataka za specifične tipove podataka. Modeli mogu biti jednostavni, kao primjerice jedna tablica, ali većinom to nije slučaj.[8]

Podaci dolaze u tablice iz više različitih izvora i moraju se povezati u smislenu cjelinu. To se radi definiranjem kako je svaki od različitih izvora podataka povezan s ostalima. Tehnologije za transformiranje i čišćenje podataka su zapravo ETL alati, a neki od primjera su *Microsoft SQL Server Integration Services (SSIS), Azure Data Factory, Oracle Data Integrator* i drugi.[8]

4.2.4. Analiza

Nakon odabira domene i spajanja izvora podataka u model vrijeme je za analizu podataka. Ovo je ključni proces u kojem se odgovara na pitanja relevantna za poslovanje. Analiza podataka može poprimiti razne oblike poput grupiranja podataka, kreiranje jednostavnih agregacija poput zbrajanja, prebrojavanja i prosjeka, ali i složenijih kalkulacija, identificiranja trendova, korelacija i predviđanja. Mnoge organizacije imaju ili žele imati ključne pokazatelje performanse poslovanja (KPI, eng. *Key performance indicators*). KPI može uključivati stvari poput broj novih kupaca mjesečno, bruto marža i sl. U nekim slučajevima se koriste napredni alati za analizu poput programskih jezika, strojnog učenja i umjetne inteligencije, rudarenja podataka, *streaming* analitike i nestrukturirane analize za dobivanje kvalitetnijih uvida. [8]

Česti programski jezici za ovu svrhu su *Python* i *R*, a česti sustavi za strojno učenje i rudarenje podacima su *Azure ML, DataRobot, Alteryx Analytics* i dr. Valja spomenuti i proračunske tablice kao česti alat kod analitike podataka, a najpoznatiji alat za to je svakako *Microsoft Excel* iako postoje i drugi poput *Google Sheets, Apple Numbers* i sl.[8]

4.2.5. Vizualizacija

Konačni ključni koncept poslovne inteligencije jest prezentacija provedene analize. Vizualizacija poprima oblike grafova, dijagrama i sličnih prikaza koji pomažu u objašnjavanju kon-

teksta i davanju značenja analizi. Vizualizacijom analitičar ili autor izvještaja gradi priču kojom odgovara na pitanja postavljena za pomoć pri donošenju poslovnih odluka.[8]

Alati poslovne inteligencije dozvoljavaju da se više individualnih tablica i dijagrama spoji na jednoj stranici ili izvještaju. Moderniji alati također podržavaju interaktivnost između individualnih vizualizacija za dodatnu pomoć u procesu otkrivanja i analize.[8]

5. Izrada skladišta na platformi Azure

U ovom će se poglavlju prikazati izrada skladišta podataka na Microsoftovoj platformi Azure na dva načina. Prije toga biti će opisana domena i korišteni podaci kao i korišteni model skladišta.

5.1. Opis domene, skupa podataka i modela

Domena u koju je smješteno skladište jest zamišljena tvrtka koja se bavi prodajom građevinskog materijala poput drvene građe, cementnih blokova, metalnih cijevi i sl. Preciznije, skladište je fokusirano na prodaju te tvrtke - izvršene prodaje kroz nekoliko različitih kraćih perioda od 2019. do 2021.

Podaci koji će se koristiti raspoređeni su u sedam CSV (eng. *comma separated value*) datoteka, dakle koriste se polustrukturirani podaci. Svaka datoteka odgovara jednoj tablici modela, a unutar svake pojedine datoteke podaci su podijeljeni po zaglavljima koja predstavljaju stupce tablica. Datoteke imaju nazive *dates, discounts, employees, marketings, products, sales* i *stores* u kojima su podaci o datumima izvršenih prodaja, popustima, zaposlenicima, marketinškim strategijama, proizvodima i samim prodajama.

5.1.1. Model izrađenog skladišta

Za izradu modela korišteno je dimenzijsko modeliranje. Model ima već spomenutu shemu zvijezde sa jednom centralnom činjeničnom tablicom, ***Sales*** i šest dimenzijskih tablica na koje je činjenična tablica povezana vanjskim ključevima. Dimenzijske tablice skladišta jesu: ***Products, Marketings, Discounts, Dates, Stores, Employees***. Model je prikazan na slici 4.

Dimenzijska tablica *Products* sadrži informacije o proizvodima koji su se prodavali. Njen primarni ključ je jedinstveni ID proizvoda (*PRODUCT_ID*) na kojeg je vanjskim ključem vezan istoimeni atribut činjenične tablice. U ovoj tablici prisutne su karakteristične hijerarhije dimenzijske tablice jer osim ID-a i imena proizvoda postoje još i atributi koji opisuju materijal od kojeg je napravljen proizvod, dimenzije proizvoda, opis proizvoda, kategoriju proizvoda, opis kategorije proizvoda i zemlju proizvodnje.

Sljedeća dimenzijska tablica, *Marketings* ima atribute koji opisuju medij koji se koristio za oglašavanje (TV, Radio), trajanje oglašavanja u danima, oglašavatelja i frekvenciju oglašavanja u satima. Primarni ključ tablice je ID marketinga (*MARKETING_ID*) na kojeg je povezan istoimeni vanjski ključ činjenične tablice.

Discounts je dimenzijska tablica koja ima popis akcija (popusta) koje je tvrtka imala. Atributi su opis akcije, postotak sniženja cijene, datumi početka i kraja akcije, ime mjeseca u kojem se odvijala akcija i trajanje akcije u danima. Primarni ključ je *DISCOUNT_ID* spojen s istoimenim vanjskim ključem u činjeničnoj tablici *Sales*.

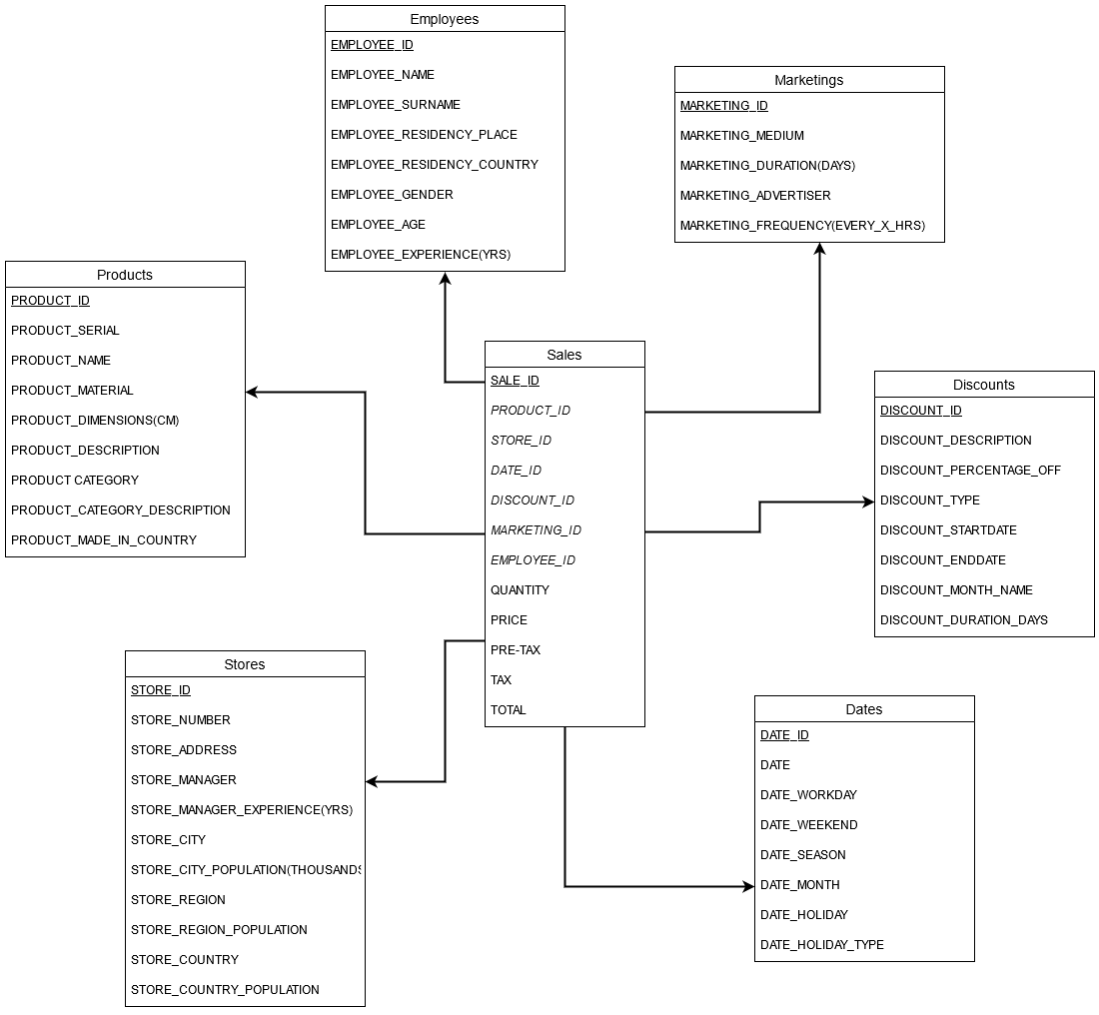
Dimenzijska tablica *Dates* sadrži atribute s informacijama o datumima kad se bilježila

prodaja. Atributi su zastavice koje označavaju je li zadani datum radni dan, vikend, praznik, a tu su i atributi za godišnje doba, ime mjeseca i vrstu praznika. Kao i kod ostalih, primarni ključ je jedinstvena numerička oznaka *DATE_ID* koji je povezan na vanjski ključ u činjeničnoj tablici.

Peta dimenzijska tablica jest tablica *Stores* koja čuva podatke o trgovinama. Sadrži atribute s informacijama o broju trgovine, adresi, imenu upravitelja, radnom stažu upravitelja u godinama, gradu u kojem je trgovina, broju stanovnika tog grada u tisućama, regiji u kojoj je trgovina, broju stanovnika te regije u tisućama, države u kojoj je trgovina te broju stanovnika te države u milijunima. Svaka trgovina ima svoj ID koji je primarni ključ (*STORE_ID*) i povezan je na istoimeni vanjski ključ u činjeničnoj tablici.

I posljednja dimenzijska tablica jest tablica sa zaposlenicima, *Employees*. Svaki zaposlenik ima svoj ID (*EMPLOYEE_ID*, povezan na vanjski ključ činjenične tablice), ime i prezime, mjesto stanovanja, državu stanovanja, spol, dob i radni staž u godinama.

Centralni dio modela je činjenična tablica *Sales*. Ona ima svoj primarni ključ, *SALE_ID* i spomenute vanjske ključeve: *PRODUCT_ID, MARKETING_ID, DISCOUNT_ID, DATE_ID* i *EMPLOYEE_ID*. Za svaku prodaju pamti se količina prodanog proizvoda, cijena proizvoda, iznos prije poreza, iznos poreza (15 posto iznosa prije poreza) i ukupni iznos s porezom.



Slika 4: Izrađeni model skladišta

5.2. Azure SQL Database

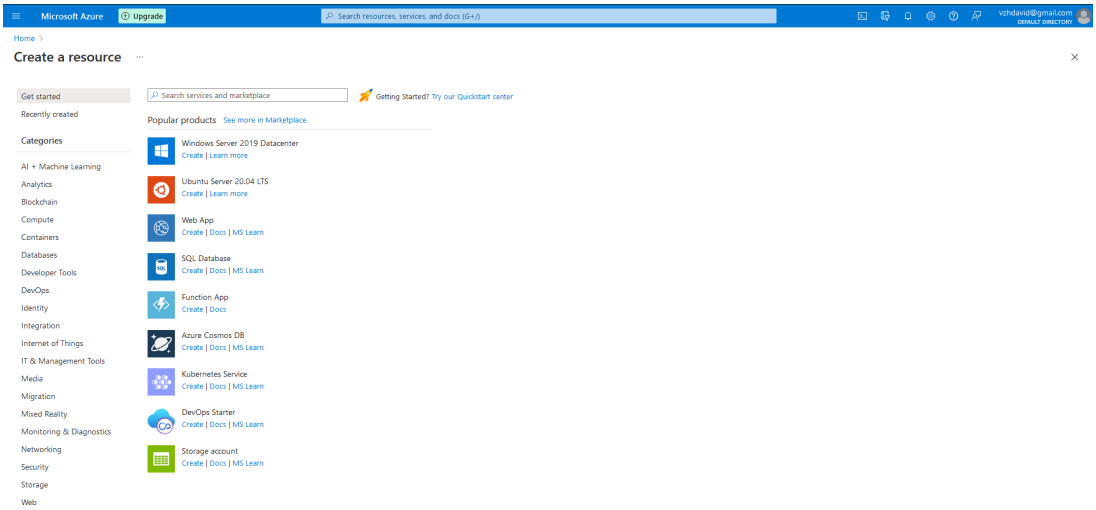
Kako je bilo navedeno, dimenzijski modeli poput sheme zvijezde kakva je izrađena (slika 4) se često instanciraju u sustavima za relacijske baze podataka. Stoga prvi način koji će se prikazati za izradu skladišta na Azure platformi koristi *Azure SQL Database*.

Azure SQL Database je stroj za baze podataka napravljen kao "platforma kao usluga" (eng. *Platform as a Service*, PaaS) koji obavlja većinu upravljačkih funkcija baza podataka poput nadogradnje, zakrpa, sigurnosnih kopija i nadzora bez uključenosti korisnika. [9] Ona je tehnologija baza podataka u oblaku (eng. *cloud hosted*) i može se na nju gledati kao PaaS verziju tradicionalne lokalne SQL baze podataka [10].

Azure SQL Database je OLTP sustav optimiziran za čitanje i pisanje podataka, ali ima nekoliko značajki koje ga čine pogodnim za skladištenje podataka. Može skalirati do 100 TB i pružiti dodatne *read-only* čvorove za računanje koji pružaju analitičke podatke. Također ima inteligentnu obradu upita i može biti vrlo reaktivan na promjene uvjeta u vremenu izvođenja čime se performanse održavaju i u vrhuncu opterećenosti u kritičnim vremenima. Konačno, ima više opcija za razvoj koje uključuju upravljane instance (eng. *Managed instances*) i elastične bazene (eng. *Elastic pools*). Prva opcija je instanca SQL servera na *cloud*-u i najbliža je postojećoj lokalnoj Microsoft SQL Server implementaciji. [10] Ova opcija biti će korištena u ovom poglavlju. Druga opcija koristi jedan bazen resursa za računanje (eng. *Compute resource*) što dozvoljava niži ukupni trošak vlasništva jer baze podataka mogu trošiti određenu količinu resursa iz zajedničkog bazena umjesto da svaka nezavisno skalira. [10]

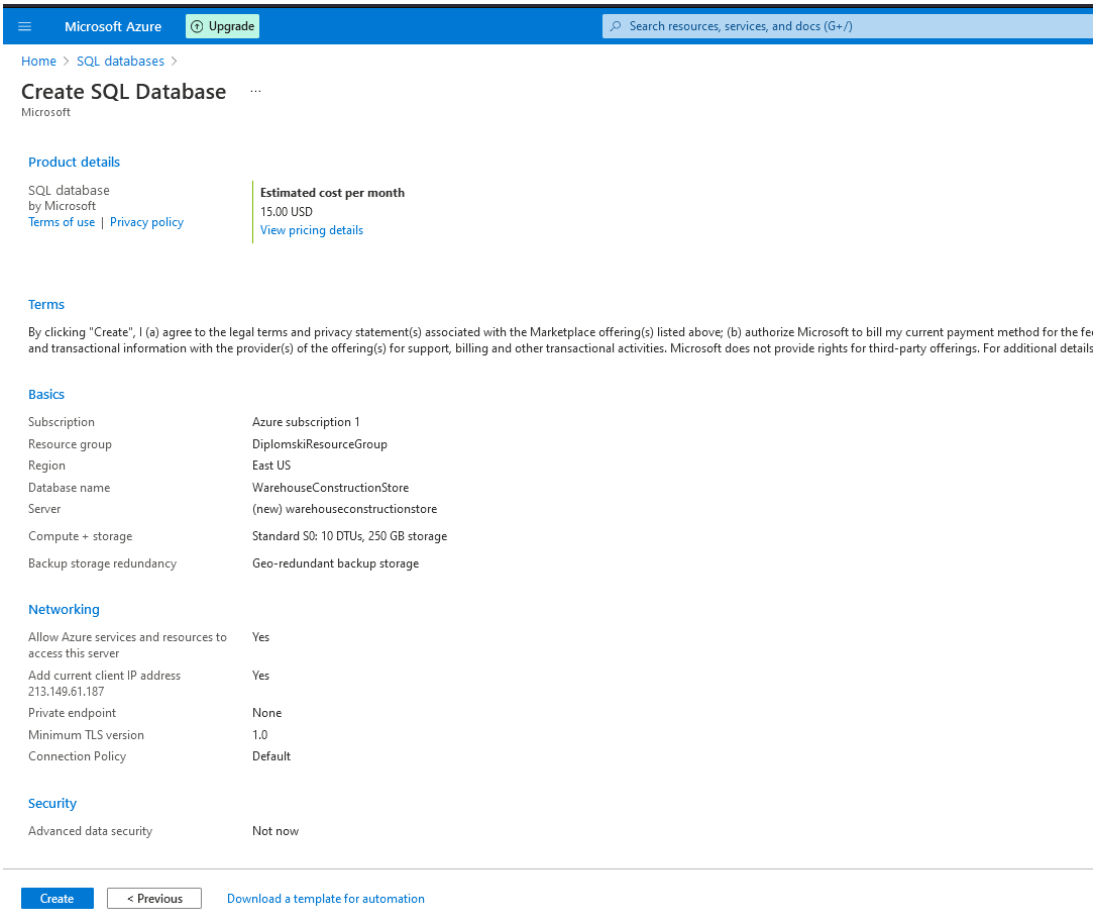
Azure SQL Database uvijek radi na najnovijoj stabilnoj verziji SQL Server stroja za baze podataka (eng. *database engine*) sa dostupnošću od 99.99%. [9] Standardna verzija *Azure SQL DB* ima gornju granicu na veličinu baze (u ovom slučaju skladišta) podataka od 4TB [10].

Prvi korak je kreirati jednu bazu podataka na Azure portalu [11]. Na početnoj stranici odabere se *Create resource* pa potom na stranici koja se otvara (slika 5) u tražilicu se upisuje *SQL Database* i zatim *Create*.



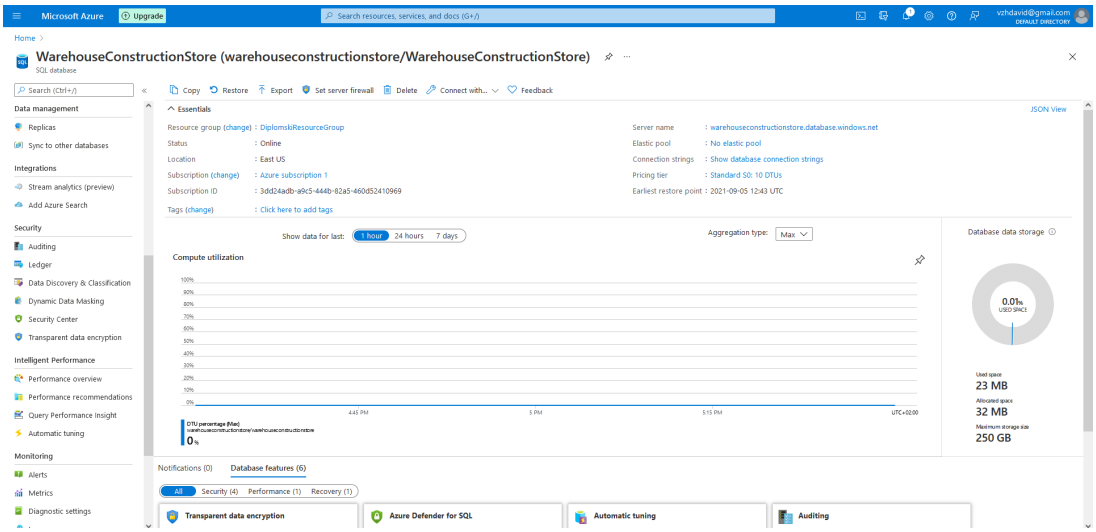
Slika 5: Stvaranje resursa na Azure portalu

Otvara se čarobnjak za stvaranje baze podataka (slika 6) u kojem se odabiru bitne postavke za instanciranje baze. Odabire se pretplata (eng. *subscription*), grupa resursa (eng. *resource group*), regija, ime baze podataka, poslužitelj (korisnik kreira novi ukoliko ga nema) te željene performanse obrade i veličina pohrane (eng. *Compute + storage*). Posljednja stavka direktno utječe na cijenu - najmanja dostupna veličina je 250GB čime dobivamo procijenjeni trošak od $15 mjesečno što se vidi na slici 6.



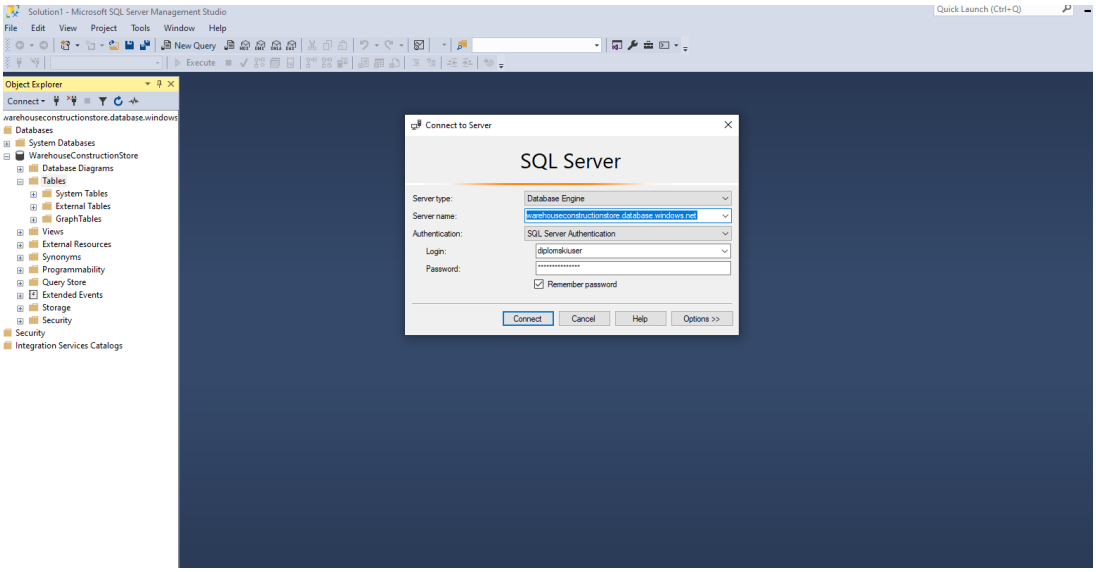
Slika 6: Kreiranje Azure SQL DB

Ovime je stvorena baza podataka čiji pregled se vidi klikom na taj resurs (*Azure SQL DB* na početnoj stranici. Osim pregleda (slika 7) postoje razne mogućnosti i značajke poput dnevnika aktivnosti, izvoza baze podataka, nadziranja, opcija sigurnosti, mijenjanja postavki pa i obavljanja upita nad bazom.



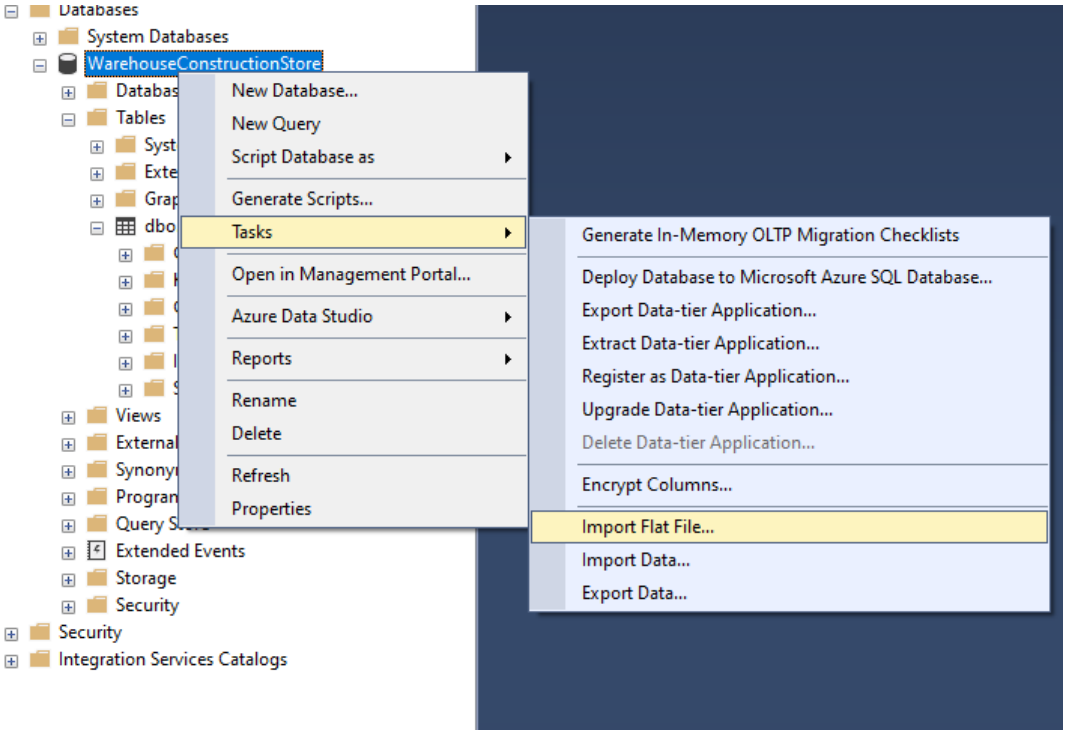
Slika 7: Pregled Azure SQL DB

Međutim, da bi stvorili potrebne tablice i implementirali prikazani model na stvorenu bazu podataka spaja se pomoću alata *Microsoft SQL Server Management Studio* (SSMS) kako je prikazano na slici 8. Upisuju se korisničko ime i lozinka prethodno stvorenog korisnika (administratora) i poveznica na poslužitelj s bazom podataka [12].

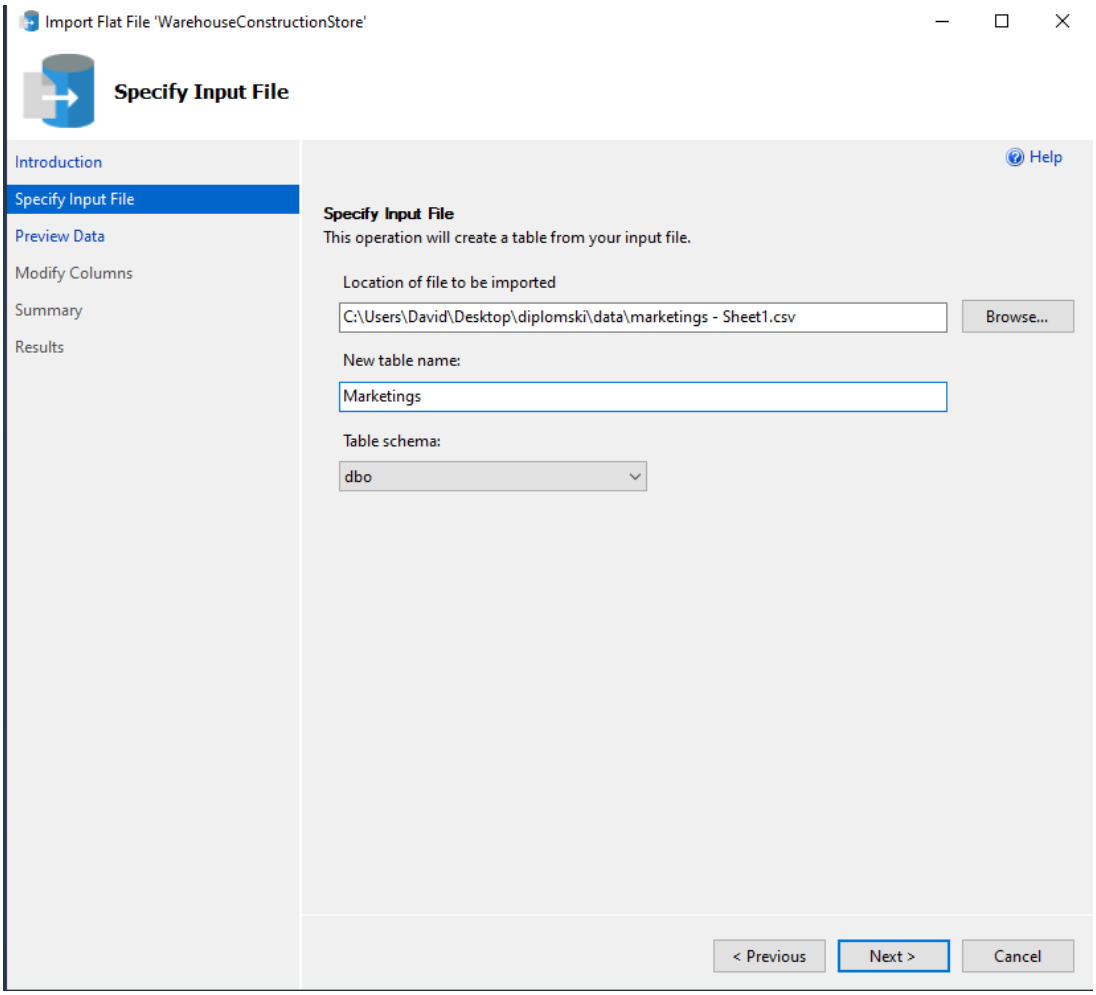


Slika 8: Spajanje na Azure SQL DB

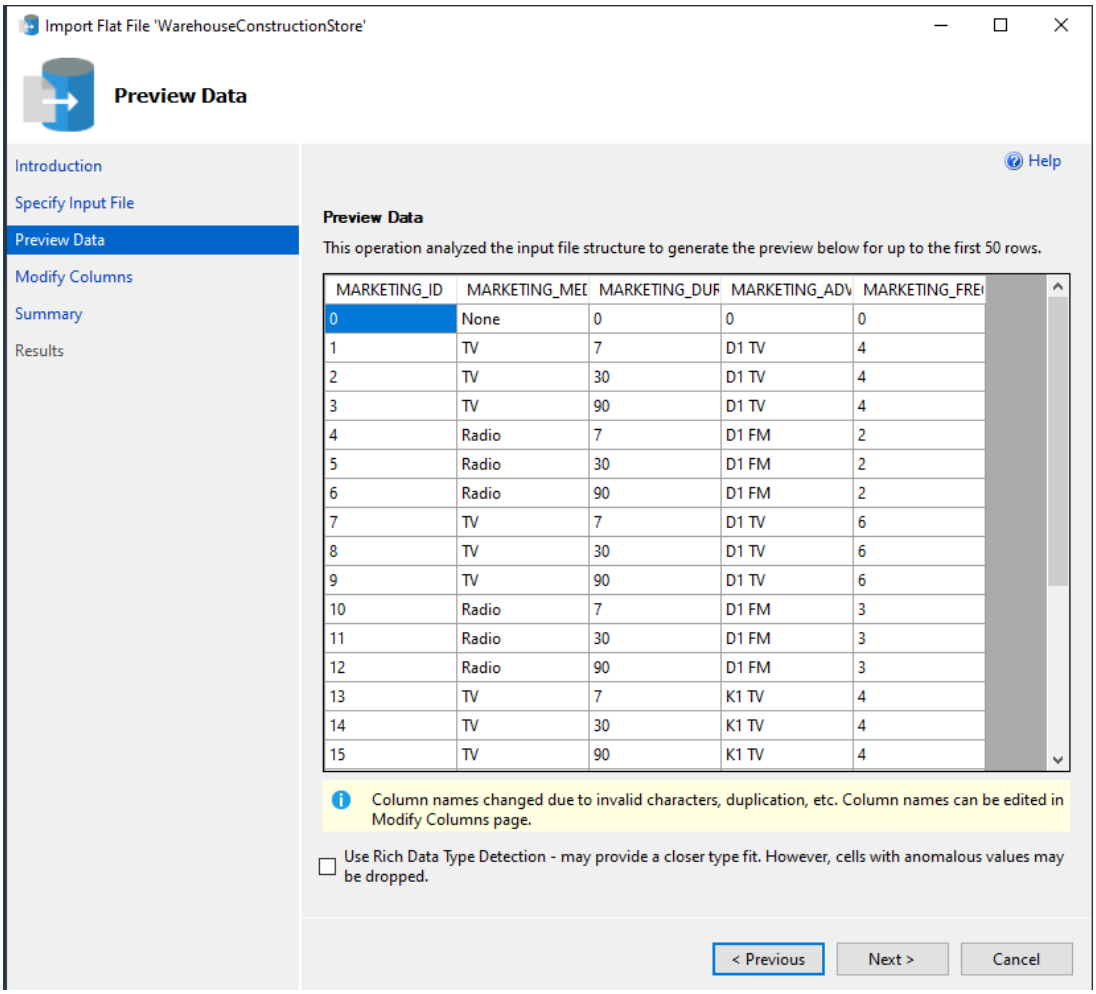
Nakon spajanja na stvorenu bazu podataka potrebno je stvoriti tablice modela i učitati podatke. Vrlo moćan alat kao što je *Management Studio* pruža jednostavan način za to - moguće je iz CSV datoteka učitati podatke pri čemu se automatski stvaraju tablice. Ovaj proces prikazan je na slikama 9-12. Na slici 9 odabire se zadatak (eng. *Task*) nad bazom podataka za učitavanje podataka, a na slici 10 odabire se datoteka i ime nove tablice. Na slici 11 vidi se kako će izgledati ti podaci u tablici. U koraku na slici 12 postavlja se tip podataka za atribute u tablici. Proces se ponavlja za svaku datoteku, a rezultat je sedam kreiranih tablica sa svim podacima iz datoteka.



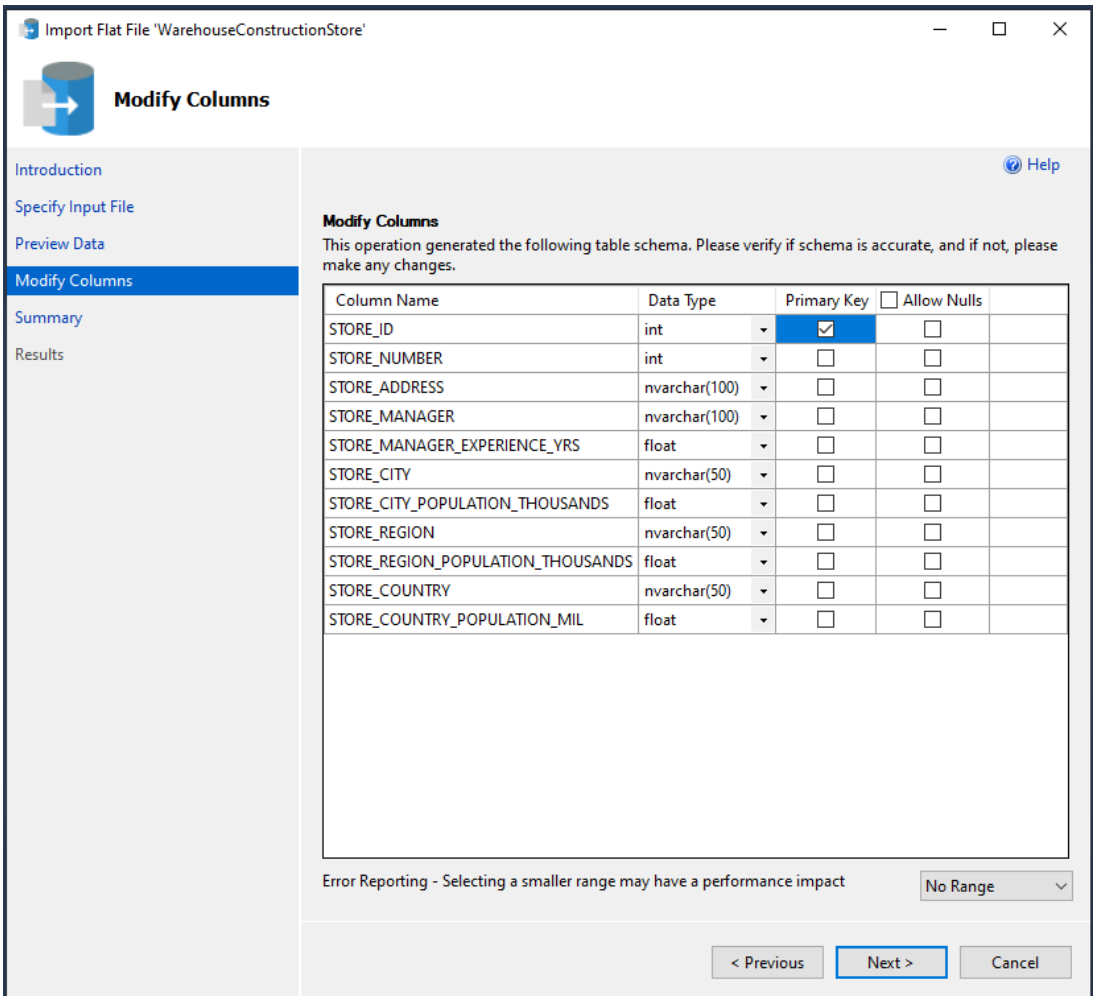
Slika 9: Učitavanje podataka u skladište (1)



Slika 10: Učitavanje podataka u skladište (2)

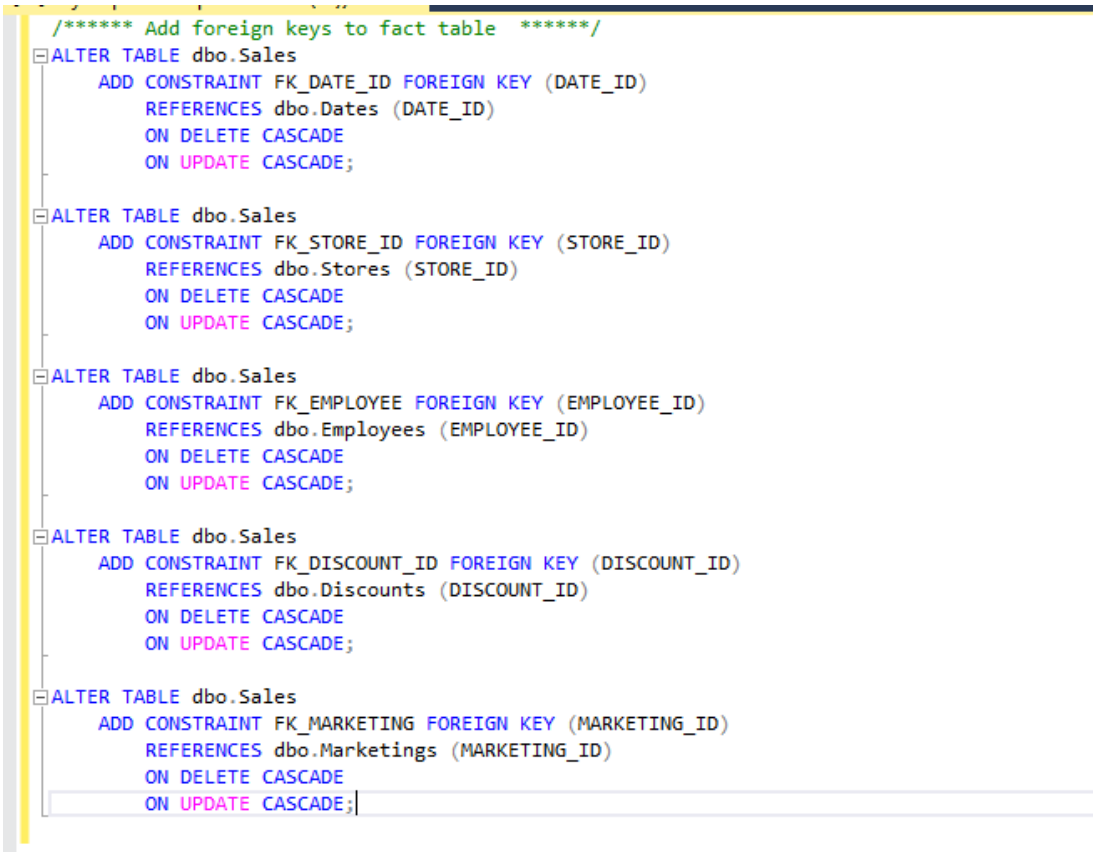


Slika 11: Učitavanje podataka u skladište (3)



Slika 12: Učitavanje podataka u skladište (4)

Sve što preostaje za potpuno zadovoljenje kreiranog modela je povezivanje tablica. Za povezivanje tablica koriste se primarni i vanjski ključevi pri čemu vrijednost vanjskog ključa mora odgovarati nekoj od vrijednosti primarnog ključa na kojeg se referencira [13]. Primarni ključevi definirani su kod stvaranja tablica, a jedina tablica s vanjskim ključevima je činjenična tablica *Sales*. Vanjske ključeve dodajemo naredbom *ALTER TABLE* koja omogućuje mijenjanje tablice nakon njezinog kreiranja [13] i dodavanjem ograničenja (*ADD CONSTRAINT*) [14] što je prikazano na slici 13. Dodaje se šest jednostavnih vanjskih ključeva što znači da se svaki sastoji od jednog stupca i može se implementirati kao ograničenje nad stupcem [13]. Osim SQL naredbom vanjski ključevi se u SSMS-u mogu dodati i preko *Table Designer*-a. [14]



Slika 13: Dodavanje vanjskih ključeva u činjeničnu tablicu

Ovim je kreirano skladište podataka na *Azure* platformi prema ranije izrađenom modelu s učitanim podacima iz polustrukturiranog izvora koje se dalje može koristiti u svrhu poslovne inteligencije što će biti prikazano u kasnijem poglavlju.

5.3. Azure Synapse Analytics

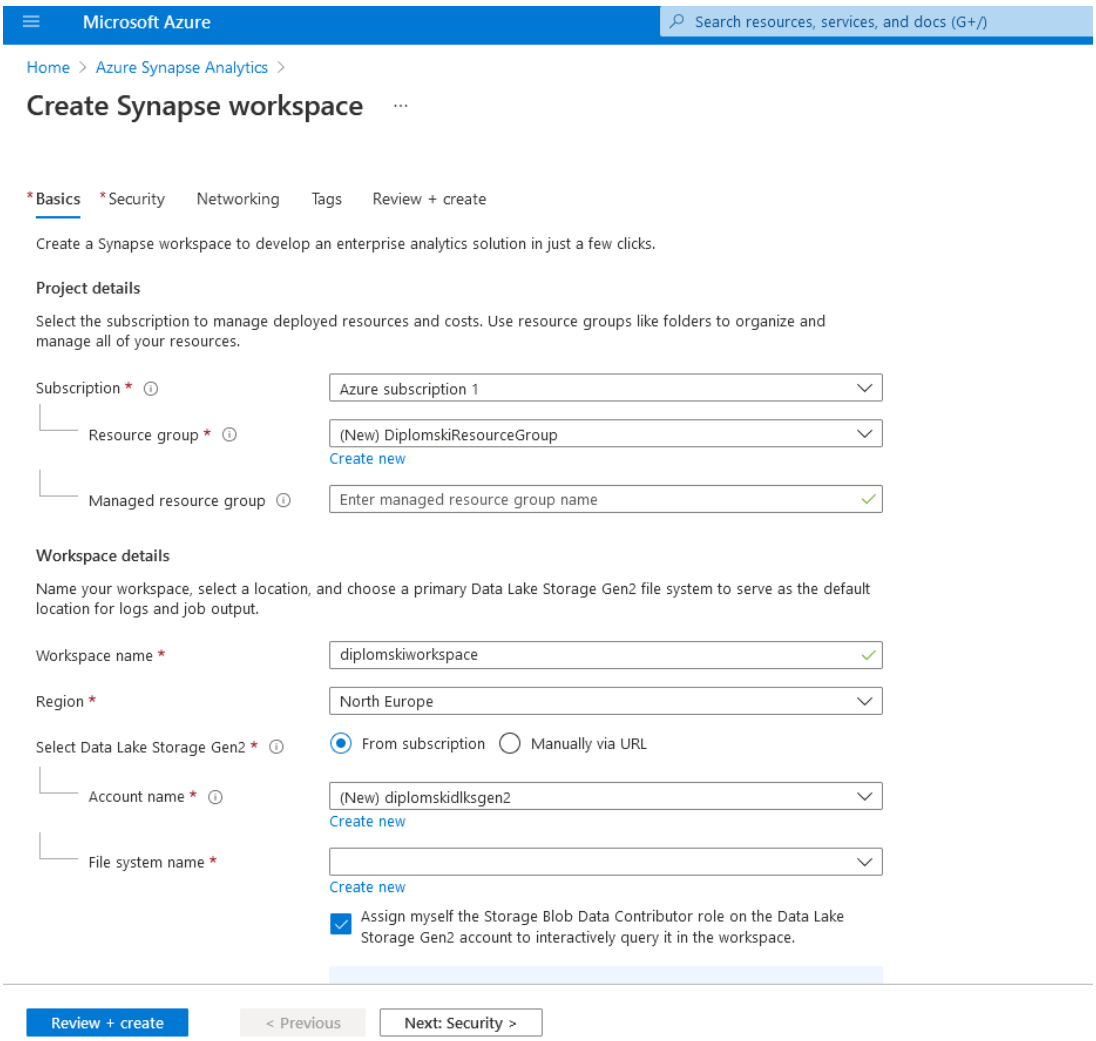
Drugi način na koji će se implementirati skladište podataka na *Azure* platformi jest koristeći *Azure Synapse Analytics*.

Porijeklo *Synapse Analytics*-a jest *Analytics Platform System* (ili *Parallel Data Warehouse*, PDW). Ono je bilo uređaj - prodavalo se kao fizički poslužitelj sa svojom verzijom *SQL server*-a. Porastom popularnosti *cloud*-a logični korak naprijed bilo je prenošenje ideja i funkcija PDW-a, poput jezgrenog SQL stroja, na njega. Rezultat je bio *Azure SQL Data Warehouse*. On je kasnije preimenovan u *Azure Synapse Analytics* bez značajnih promjena u funkcionalnostima, ali to nije ono što je *Synapse Analytics* danas. *Azure SQL DW* jest i dalje prisutan kao dio *Synapse Analytics*-a ali se sad naziva SQL bazen (eng. *SQL pool*) [15] koji će se izraditi u kasnijem poglavlju.

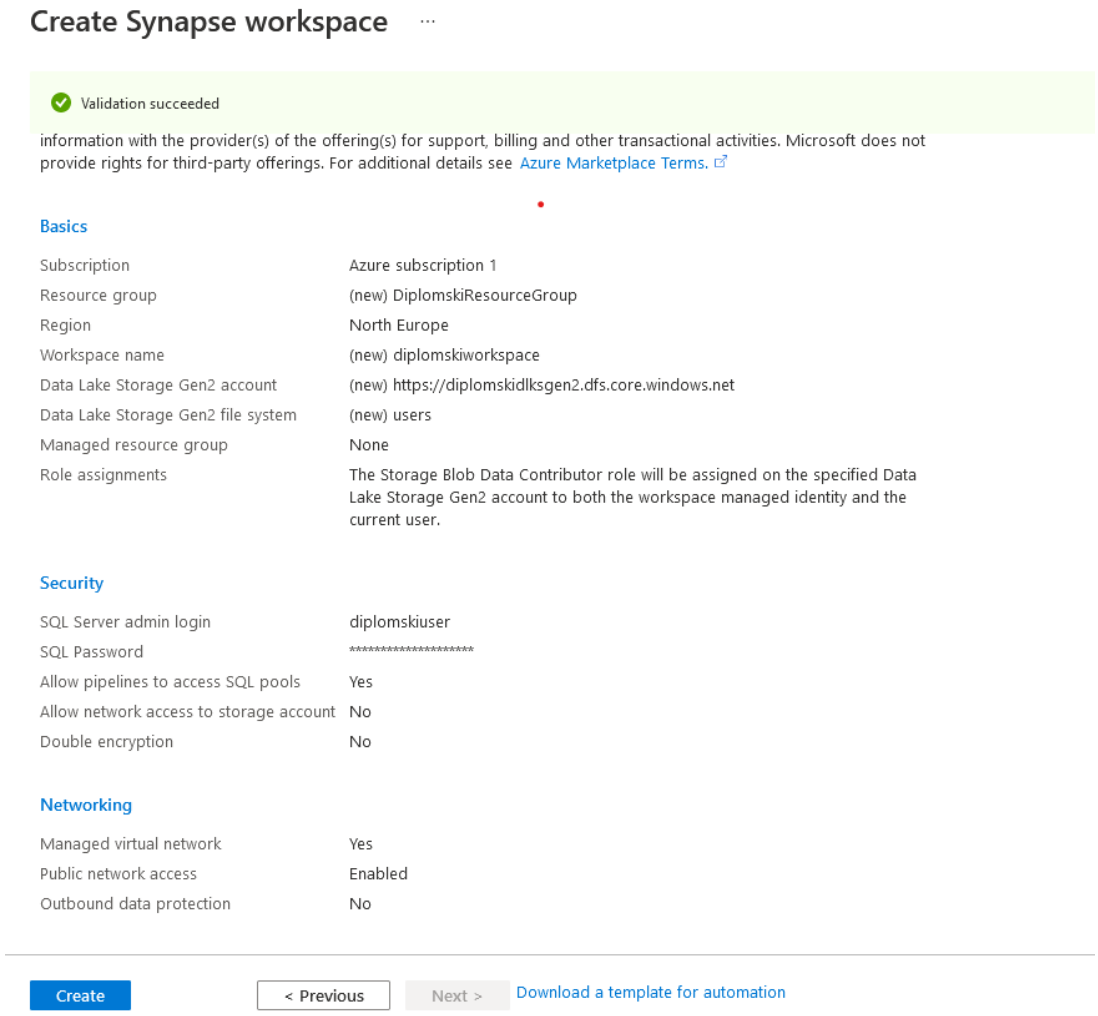
Azure Synapse je analitički servis za uvid kroz skladišta podataka i *big data* sustava. To je masovno paralelno procesirajući stroj (eng. *massively parallel processing engine - MPP engine*) koji se koristi za pohranu i obradu velike količine strukturiranih podataka u *Azure*-u

koristeći *SQL server* stroj na distribuiranom klasteru računala [10]. On spaja SQL tehnologije korištene u skladištenju podataka, Spark tehnologije korištene za *big data*, cjevovode (eng. *pipelines*) za integraciju podataka i ELT/ETL i integraciju s ostalim *Azure* servisima poput *Power BI*-a, *CosmosDB*-a i *AzureML*. [16]

Prvi korak je izraditi resurs *Synapse Analytics* na *Azure* portalu. To se radi klikom na *Create resource* kao na slici 5 i traženjem *Azure Synapse Analytics* u tražilici pa zatim klikom na *Create*. Otvara se postupak stvaranja *Synapse* radnog prostora (eng. *workspace*) kao na slici 14. Bira se pretplata, grupa resursa u koju spada novi *workspace*, njegovo ime, regija, ali i *Data Lake Storage Gen 2* resurs. Ono služi za pohranu *blob* podataka, a u ovom slučaju poslužit će kao mjesto za čuvanje CSV datoteka koje služe kao izvor podataka za skladište. Ispod *drop-down* izbornika za *Data Lake* postoji opcija za stvaranje novog takvog resursa.

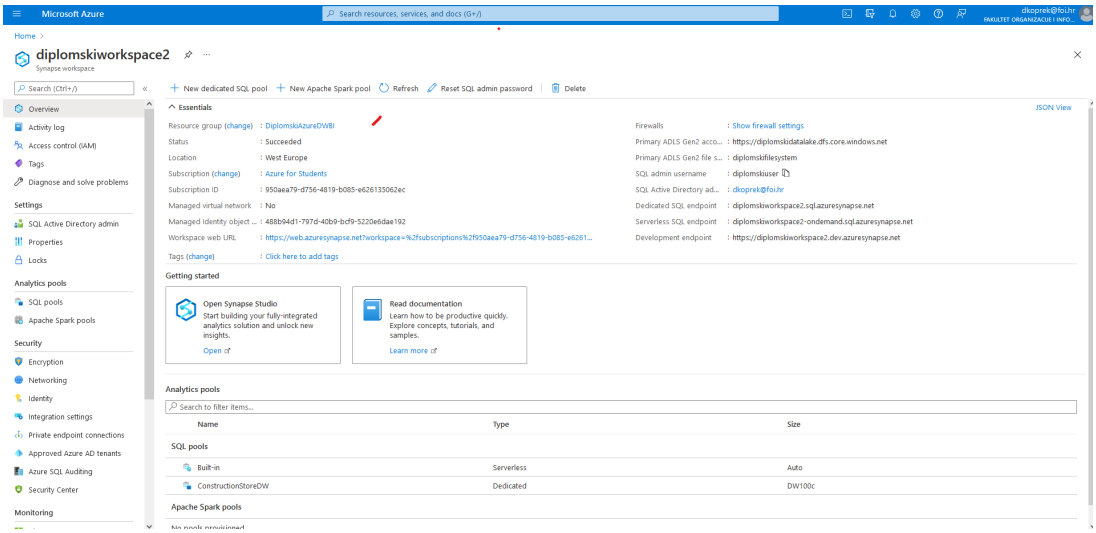


Slika 14: Kreiranje *Azure Synapse Analytics* resursa



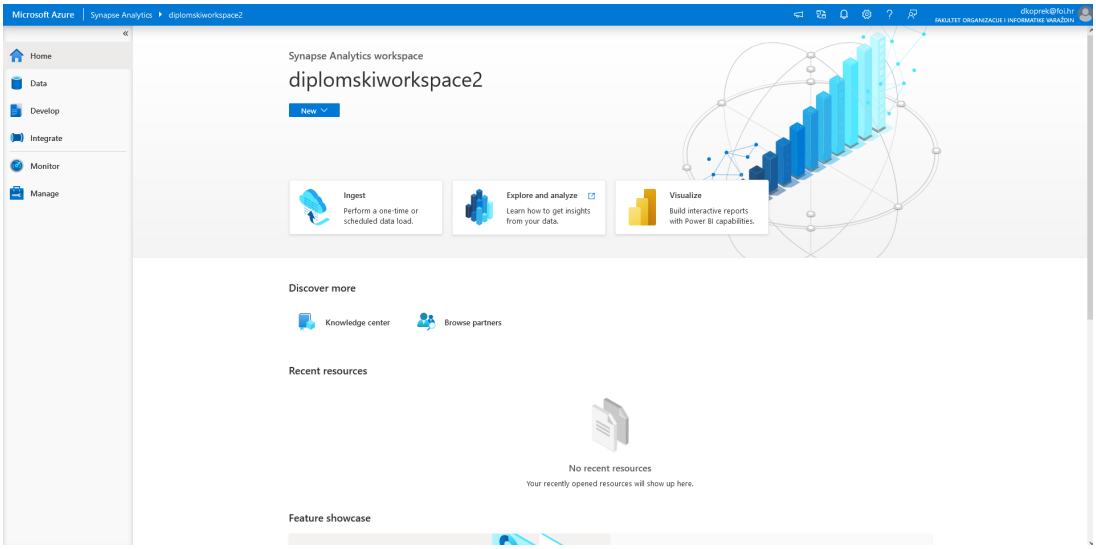
Slika 15: Pretpregled kreiranog *Synapse* radnog prostora

Na kraju procesa izrade je pretpregled odabranih postavki (slika 15) gdje je opcija *Create* klikom na koju se stvara novi resurs, *Synapse workspace*. Njegov pregled moguć je klikom na resurs na početnoj stranici *Azure portala*. U pregledu (slika 16) su vidljive ključne informacije poput SQL krajnih točaka (eng. *endpoints*), korisničko ime SQL administratora, web URL radnog prostora, primarni *ADLS Gen 2* račun (*Data Lake*) i sl. Na početnoj stranici resursa nalaze se i mogućnosti pregleda dnevnika aktivnosti, dijagnostika, postavke sigurnosti poput enkripcije i umrežavanja, nadzor nad resursom i dr.



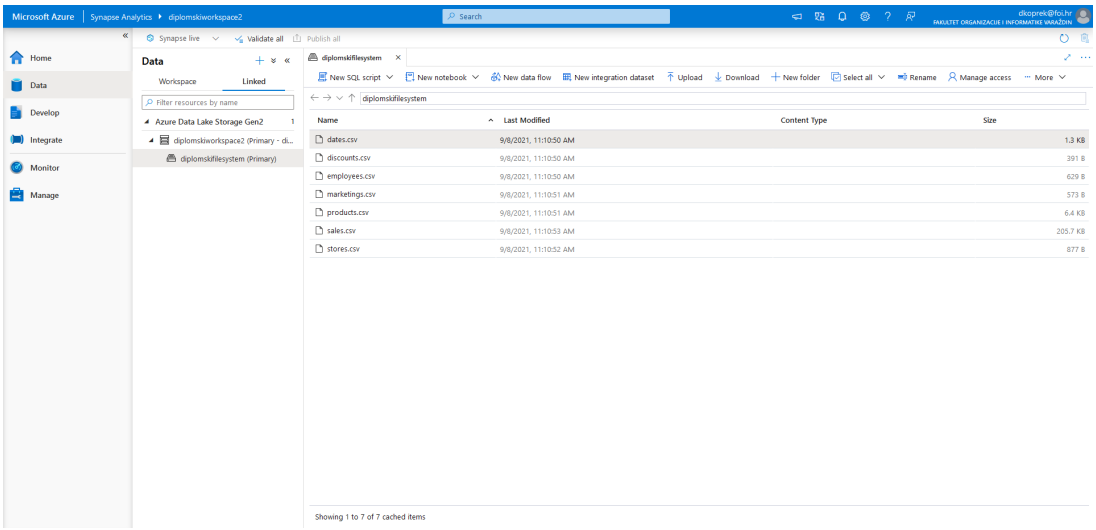
Slika 16: Pregled *Synapse workspace* resursa

Samom radnom prostoru pristupa se putem spomenutog web URL-a. On vodi na početnu stranicu (eng. *Home*), a u izborniku su još *Data, Develop, Integrate, Monitor* i *Manage* (slika 17).



Slika 17: Synapse Analytics workspace

Pod *Data* pa zatim *Linked* vidi se prethodno kreirana i povezana *Azure Data Lake Storage Gen2* pohrana gdje se učitavaju datoteke s podacima (slika 18).

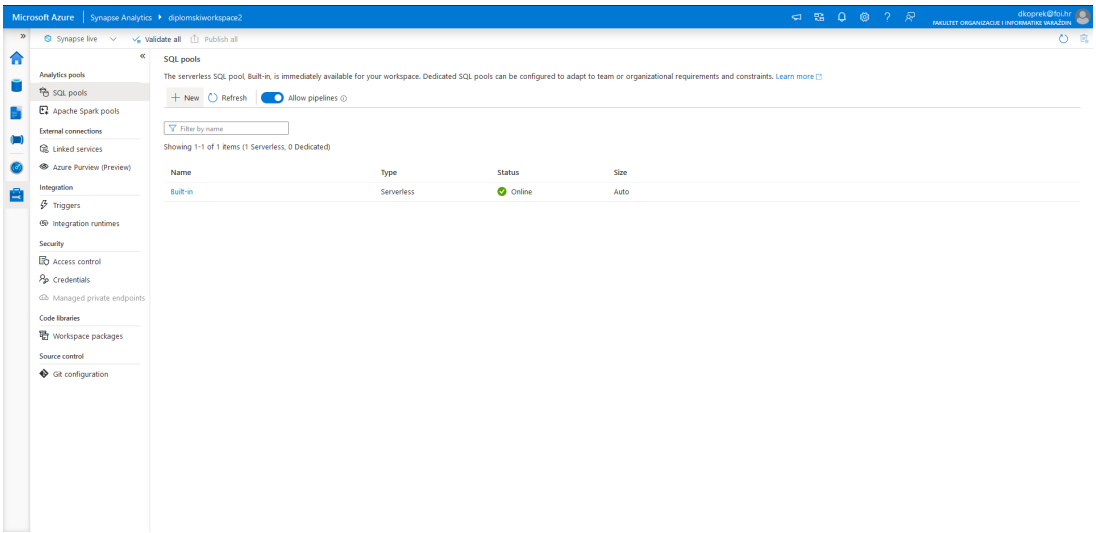


Slika 18: Datoteke učitane u *ADLS Gen2*

Sada je izvor podataka pripremljen i može se kreirati samo skladište podataka.

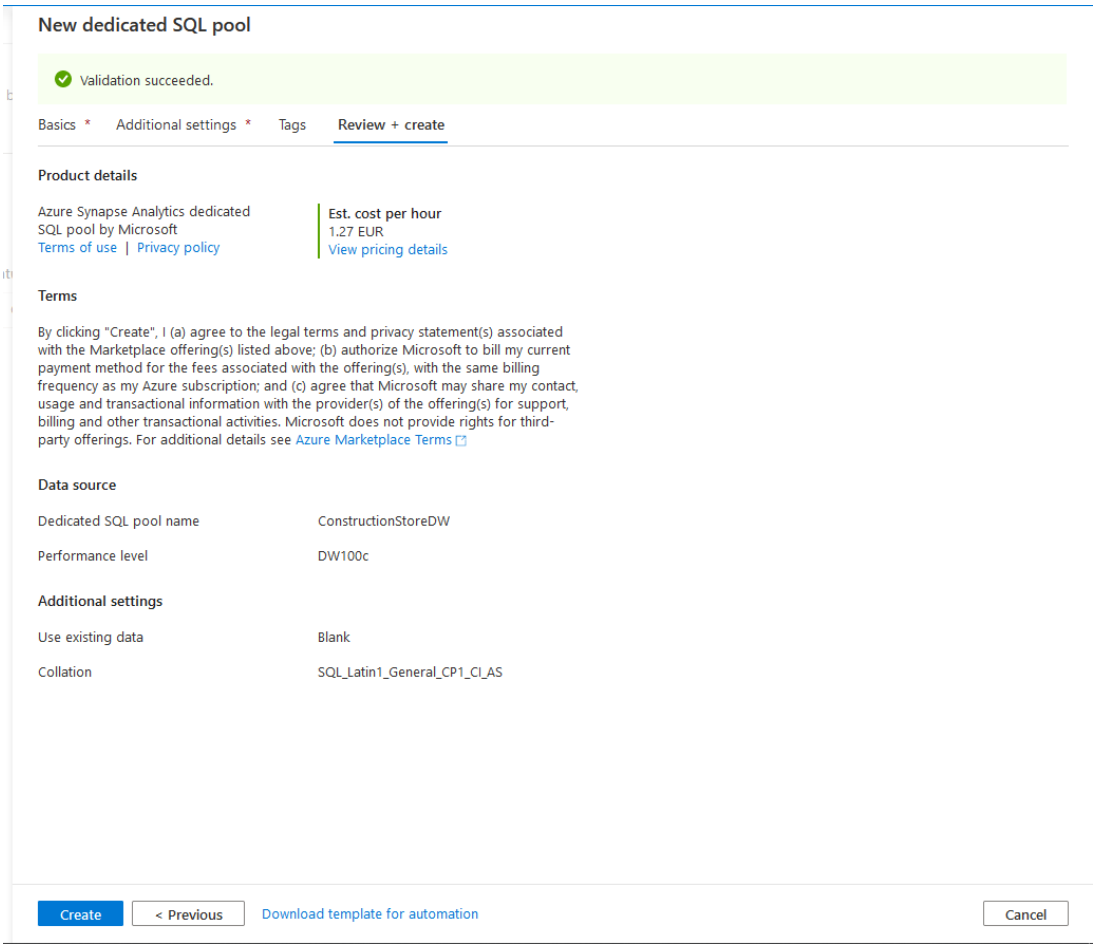
5.3.1. Izrada posvećenog SQL bazena

Skladište podataka biti će posvećeni SQL bazen (eng. ***dedicated SQL pool***) koji se nekad i doslovno zvao *Azure SQL Data Warehouse* [17]. Njega je moguće kreirati iz Azure portala klikom na *New dedicated SQl pool* (slika 16) ili iz Synapse radnog prostora pod *Manage -> SQL pools -> + New*. (slika 19).



Slika 19: Kreiranje SQL bazena (skladišta podataka) za Synapse(1)

Opet se pokreće proces kreiranja resursa, biraju se postavke i kreira se SQL bazen. Jedna od postavki je razina performansi skladišta koja direktno utječe na cijenu. U ovom slučaju skladište je malo i nisu potrebne performanse više od najniže moguće ponuđene razine. Takva razina daje trošak od €1.27 na sat (slika 20). Može se zaključiti da je takva cijena neisplativa za potrebe našeg skladišta u odnosu na trošak za *Azure SQL DB* iz prošlog poglavlja.



Slika 20: Kreiranje SQL bazena (skladišta podataka) za Synapse(2)

5.3.2. Izrada tablica skladišta

Ovime je stvoreno skladište na koje se moguće povezati sa *SQL Server Management Studio*-om jednako kao i na *Azure SQL DB* na slici 8, ali se naravno spaja na poslužitelj sa novostvorenim skladištem (u ovom slučaju to je *diplomskiworkspace2.sql.azuresynapse.net*). Prije toga treba provjeriti postavke vatrozida za bazu (skladište) i dodati IP klijenta s kojeg se spaja ako ga nema. [17] Sada je potrebno kreirati tablice u skladištu i u njih učitati podatke. Na poslužitelj je spojen SQL administrator, ali se preporučuje kreirati *login* i korisnika za učitavanje podataka (eng. *loader*). To se radi izvršavanjem sljedeće T-SQL naredbe u *master* bazi podataka [18]:

```
CREATE LOGIN LoaderRC20 WITH PASSWORD = 'a123STRONGpassword!';
CREATE USER LoaderRC20 FOR LOGIN LoaderRC20;
```

Potom je potrebno u skladištu kreirati istog tog korisnika i dati mu potrebne ovlasti [18]:

```
CREATE USER LoaderRC20 FOR LOGIN LoaderRC20;
GRANT CONTROL ON DATABASE::[ConstructionStoreDW] to LoaderRC20;
EXEC sp_addrolemember 'staticrc20', 'LoaderRC20';
```

S korisničkim imenom i lozinkom novokreiranog korisnika spaja se na poslužitelj sa skladištem podataka i može se započeti s kreiranjem tablica u koje će se učitati podaci. Ovo je

napravljeno izvršavanjem sljedećih T-SQL naredbi [18]:

```
CREATE TABLE [dbo].[Discounts]
(
    [DISCOUNT_ID] int NOT NULL,
    [DISCOUNT_DESCRIPTION] varchar(100) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
    [DISCOUNT_PERCENTAGE_OFF] float NULL,
    [DISCOUNT_TYPE] varchar(50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
    [DISCOUNT_STARTDATE] varchar(50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
    [DISCOUNT_ENDDATE] varchar(50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
    [DISCOUNT_MONTH_NAME] varchar(50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
    [DISCOUNT_DURATION_DAYS] float NULL
)
WITH
(
    DISTRIBUTION = ROUND_ROBIN,
    CLUSTERED COLUMNSTORE INDEX
);


CREATE TABLE [dbo].[Employees]
(
    [EMPLOYEE_ID] int NOT NULL,
    [EMPLOYEE_NAME] varchar(100) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
    [EMPLOYEE_SURNAME] varchar(100) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
    [EMPLOYEE_RESIDENCY_PLACE] varchar(100) COLLATE SQL_Latin1_General_CP1_CI_AS
       NULL,
    [EMPLOYEE_RESIDENCY_COUNTRY] varchar(100) COLLATE SQL_Latin1_General_CP1_CI_AS
       NULL,
    [EMPLOYEE_GENDER] varchar(50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
    [EMPLOYEE_AGE] float NULL,
    [EMPLOYEE_EXPERIENCE(YRS)] float NULL
)
WITH
(
    DISTRIBUTION = ROUND_ROBIN,
    CLUSTERED COLUMNSTORE INDEX
);


CREATE TABLE [dbo].[Marketings]
(
    [MARKETING_ID] int NOT NULL,
    [MARKETING_MEDIUM] varchar(100) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
    [MARKETING_DURATION(DAYS)] float NULL,
    [MARKETING_ADVERTISER] varchar(100) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
    [MARKETING_FREQUENCY(EVERY_X_HRS)] float NULL
)
WITH
(
    DISTRIBUTION = ROUND_ROBIN,
    CLUSTERED COLUMNSTORE INDEX
);
```

```
CREATE TABLE [dbo].[Products]
(
    [PRODUCT_ID] int NOT NULL,
    [PRODUCT_SERIAL] varchar(100) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
    [PRODUCT_NAME] float NULL,
    [PRODUCT_MATERIAL] varchar(100) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
    [PRODUCT_DIMENSIONS(CM)] varchar(100) NULL,
        [PRODUCT_DESCRIPTION] varchar(200) COLLATE SQL_Latin1_General_CP1_CI_AS NULL
            ,
        [PRODUCT_CATEGORY] varchar(100) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
        [PRODUCT_CATEGORY_DESCRIPTION] varchar(200) COLLATE
            SQL_Latin1_General_CP1_CI_AS NULL,
        [PRODUCT_MADE_IN_COUNTRY] varchar(100) COLLATE SQL_Latin1_General_CP1_CI_AS
            NULL
)
WITH
(
    DISTRIBUTION = ROUND_ROBIN,
    CLUSTERED COLUMNSTORE INDEX
);


CREATE TABLE [dbo].[Stores]
(
    [STORE_ID] int NOT NULL,
    [STORE_NUMBER] varchar(100) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
    [STORE_ADDRESS] varchar(100) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
    [STORE_MANAGER] varchar(100) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
    [STORE_MANAGER_EXPERIENCE(YRS)] float NULL,
    [STORE_CITY] varchar(50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
        [STORE_CITY_POPULATION(THOUSANDS)] float NULL,
        [STORE_REGION] varchar(50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
        [STORE_REGION_POPULATION(THOUSANDS)] float NULL,
        [STORE_COUNTRY] varchar(50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
        [STORE_COUNTRY_POPULATION(MILLIONS)] float NULL
)
WITH
(
    DISTRIBUTION = ROUND_ROBIN,
    CLUSTERED COLUMNSTORE INDEX
);


CREATE TABLE [dbo].[Dates]
(
    [DATE_ID] int NOT NULL,
    [DATE] varchar(100) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
    [DATE_WORKDAY] tinyint NULL,
    [DATE_WEEKEND] tinyint NULL,
    [DATE_SEASON] varchar(20) NULL,
        [DATE_MONTH] varchar(50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
        [DATE_HOLIDAY] tinyint NULL,
```

```
        [DATE_HOLIDAY_TYPE] varchar(50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL
)
WITH
(
    DISTRIBUTION = ROUND_ROBIN,
    CLUSTERED COLUMNSTORE INDEX
);

CREATE TABLE [dbo].[Sales]
(
    [SALE_ID] int NOT NULL,
        [PRODUCT_ID] int NOT NULL,
        [STORE_ID] int NOT NULL,
        [DATE_ID] int NOT NULL,
        [DISCOUNT_ID] int NOT NULL,
        [MARKETING_ID] int NOT NULL,
        [EMPLOYEE_ID] int NOT NULL,
        [QUANTITY] int NOT NULL,
        [PRICE] float NOT NULL,
        [PRE-TAX] float NOT NULL,
        [TAX] float NOT NULL,
        [TOTAL] float NOT NULL
)
WITH
(
    DISTRIBUTION = ROUND_ROBIN,
    CLUSTERED COLUMNSTORE INDEX
);
```

5.3.3. Učitavanje podataka u tablice

Podaci se učitavaju korištenjem *COPY INTO* T-SQL naredbe. Tradicionalni SQl bazeni (skladišta) koriste ETL proces za učitavanje podataka. Sa *Azure Synapse Analytics* moguće je koristiti ELT (eng. *Extract, Load, Transform*) proces čime se iskorištavaju ugrađene mogućnosti procesiranja distribuiranih upita i eliminira potreba za resursima potrebnih za transformaciju podataka prije učitavanja. ELT je proces u kojem se podaci izvlače iz izvornog sustava, učitavaju u posvećeni SQL bazen i onda transformiraju. [19]

Datoteke u kojima se nalaze podaci su u *ADLS Gen2* pohrani, a *Polybase* alat ugrađen u SQL Server i spomenuta *COPY* naredba omogućavaju pristup datotekama u *Azure Blob Storage*-u ili *Azure Data Lake*-u. [19] Upravo zbog toga se koristi spomenuta naredba, a unutar nje se podacima pristupa preko URL-a, primjerice adresa datoteke s podacima o zaposlenicima glasi:

https://diplomskidatalake.blob.core.windows.net/diplomskifilesystem/employees.csv

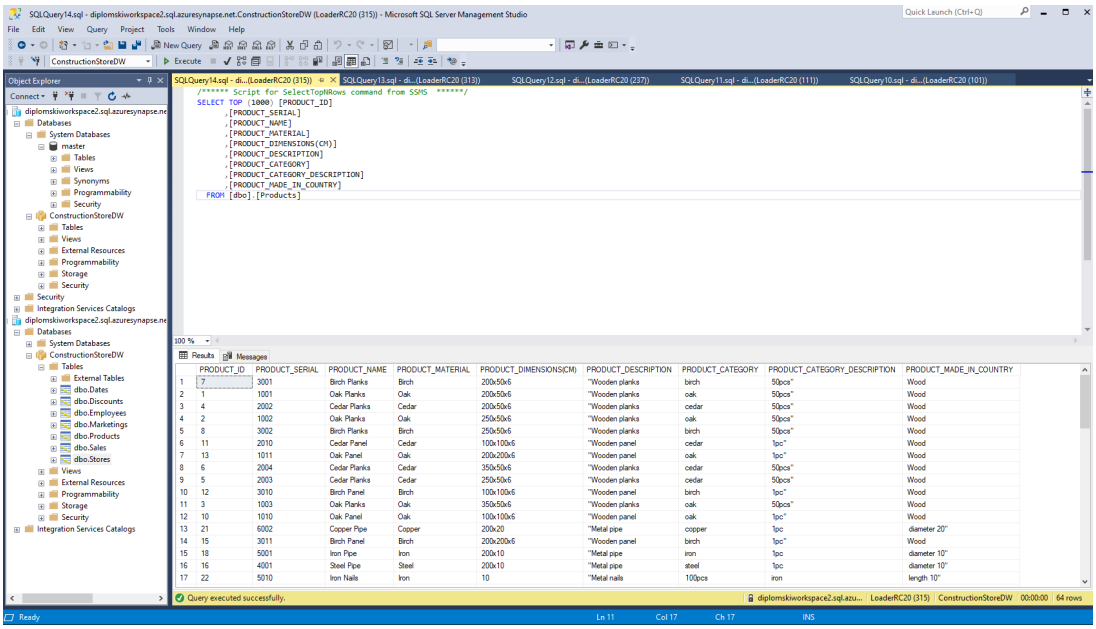
Sljedeće naredbe su izvršene za učitavanje podataka u tablice [18]:

```
COPY INTO [dbo].[Discounts]
FROM 'https://diplomskidatalake.blob.core.windows.net/diplomskifilesystem/discounts.
    csv'
```

```
WITH
(
    FILE_TYPE = 'CSV',
        FIELDTERMINATOR = ',',
        FIELDQUOTE = '',
        FIRSTROW = 2
)
OPTION (LABEL = 'COPY : Load [dbo].[Discounts] - Sales dataset');


COPY INTO [dbo].[Marketings]
FROM 'https://diplomskidatalake.blob.core.windows.net/diplomskifilesystem/marketings
    .csv'
WITH
(
    FILE_TYPE = 'CSV',
        FIELDTERMINATOR = ',',
        FIELDQUOTE = '',
        FIRSTROW = 2
)
OPTION (LABEL = 'COPY : Load [dbo].[Marketings] - Sales dataset');


COPY INTO [dbo].[Products]
FROM 'https://diplomskidatalake.blob.core.windows.net/diplomskifilesystem/products.
    csv'
WITH
(
    FILE_TYPE = 'CSV',
        FIELDTERMINATOR = ',',
        FIELDQUOTE = '',
        FIRSTROW = 2
)
OPTION (LABEL = 'COPY : Load [dbo].[Products] - Sales dataset');


COPY INTO [dbo].[Stores]
FROM 'https://diplomskidatalake.blob.core.windows.net/diplomskifilesystem/stores.csv
    '
WITH
(
    FILE_TYPE = 'CSV',
        FIELDTERMINATOR = ',',
        FIELDQUOTE = '',
        FIRSTROW = 2
)
OPTION (LABEL = 'COPY : Load [dbo].[Stores] - Sales dataset');


COPY INTO [dbo].[Dates]
FROM 'https://diplomskidatalake.blob.core.windows.net/diplomskifilesystem/dates.csv'
WITH
(
```

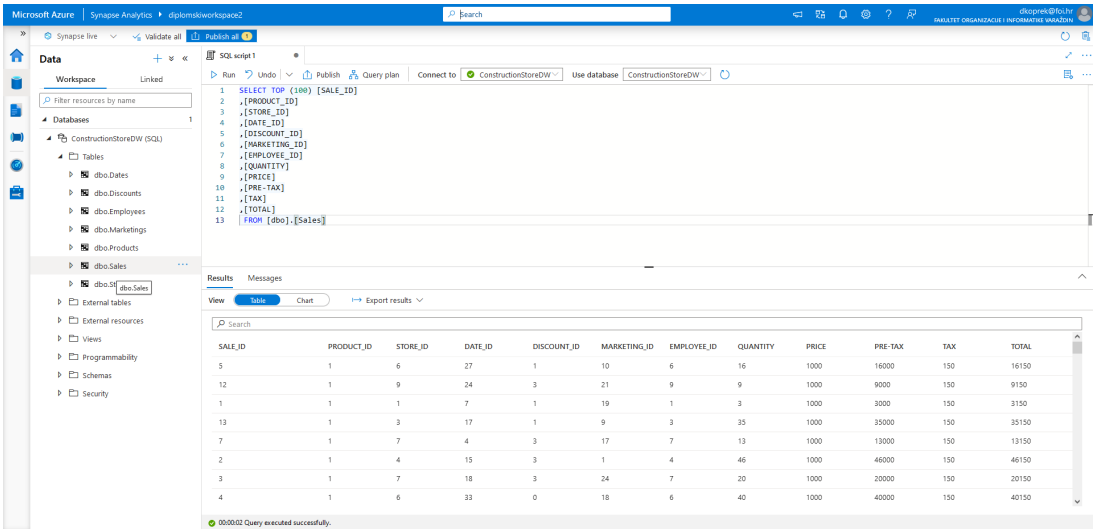
```
    FILE_TYPE = 'CSV',
        FIELDTERMINATOR = ',',
        FIELDQUOTE = '',
        FIRSTROW = 2
)
OPTION (LABEL = 'COPY␣:␣Load␣[dbo].[Dates]␣-␣Sales␣dataset');


COPY INTO [dbo].[Employees]
FROM 'https://diplomskidatalake.blob.core.windows.net/diplomskifilesystem/employees.
    csv'
WITH
(
    FILE_TYPE = 'CSV',
        FIELDTERMINATOR = ',',
        FIELDQUOTE = '',
        FIRSTROW = 2
)
OPTION (LABEL = 'COPY␣:␣Load␣[dbo].[Employees]␣-␣Sales␣dataset');


COPY INTO [dbo].[Sales]
FROM 'https://diplomskidatalake.blob.core.windows.net/diplomskifilesystem/sales.csv'
WITH
(
    FILE_TYPE = 'CSV',
        FIELDTERMINATOR = ',',
        FIELDQUOTE = '',
        FIRSTROW = 2
)
OPTION (LABEL = 'COPY␣:␣Load␣[dbo].[Sales]␣-␣Sales␣dataset');
```

Ako nema pogrešaka poput neslaganja tipova podataka (zbog toga je u *COPY* naredbama *FIRST ROW* argument postavljen na 2 [20], preskaču se zaglavlja stupaca koja su u prvom redu) podaci se uspješno učitaju u tablice čime je napravljeno i popunjeno skladište u *Azure Synapse Analytics*-u te je spremno za potrebe poslovne inteligencije. Na slici 21 vide se kreirane tablice i rezultati upita za prvih 1000 redova iz tablice *Products*.



Slika 21: Baratanje Synapse skladištem podataka u SSMS-u

Skladištu se može pristupiti i raditi upite nad njim u Synapse radnom prostoru, odabirom *Data* u izborniku, pa zatim *Workspace* -> *Databases* kako je prikazano na slici 22.



Slika 22: Skladište podataka u Synapse radnom prostoru

6. Poslovna inteligencija u Power BI alatu

Za potrebe poslovne inteligencije u ovom radu koristit će se *Microsoft Power BI*. *Power BI* je Microsoftov proizvod za vizualizaciju podataka i poslovnu inteligenciju koji se pojavio na tržištu 2014. godine. Tada je nudio odlične sposobnosti vizualizacije, a u međuvremenu se proširio mogućnostima ETL-a, interaktivnim funkcionalnostima i nizom ugrađenih konektora. [10] I iako se klasificira kao alat za vizualizaciju, *Power BI* nije jedan alat već kolekcija međupovezanih alata i servisa koji oblikuju čitav sustav poslovne inteligencije. Osim komponenti koje su specifične za *Power BI* prisutne su druge Microsoftove tehnologije koje rade interoperabilno s njim kao i integracije za treću stranu (eng. *third-party integrations*). [8] Oni pružaju platformu za vizualizaciju podataka i nudi *slice&dice* mogućnosti nad raznim vrstama informacija te izvrsne sposobnosti istraživanja podataka i kolaboracije. [10]

Power BI Free licenca je ona koja se koristi u ovom radu, za *Power BI Desktop* - stolnu aplikaciju besplatnu za preuzimanje, instalaciju i korištenje.

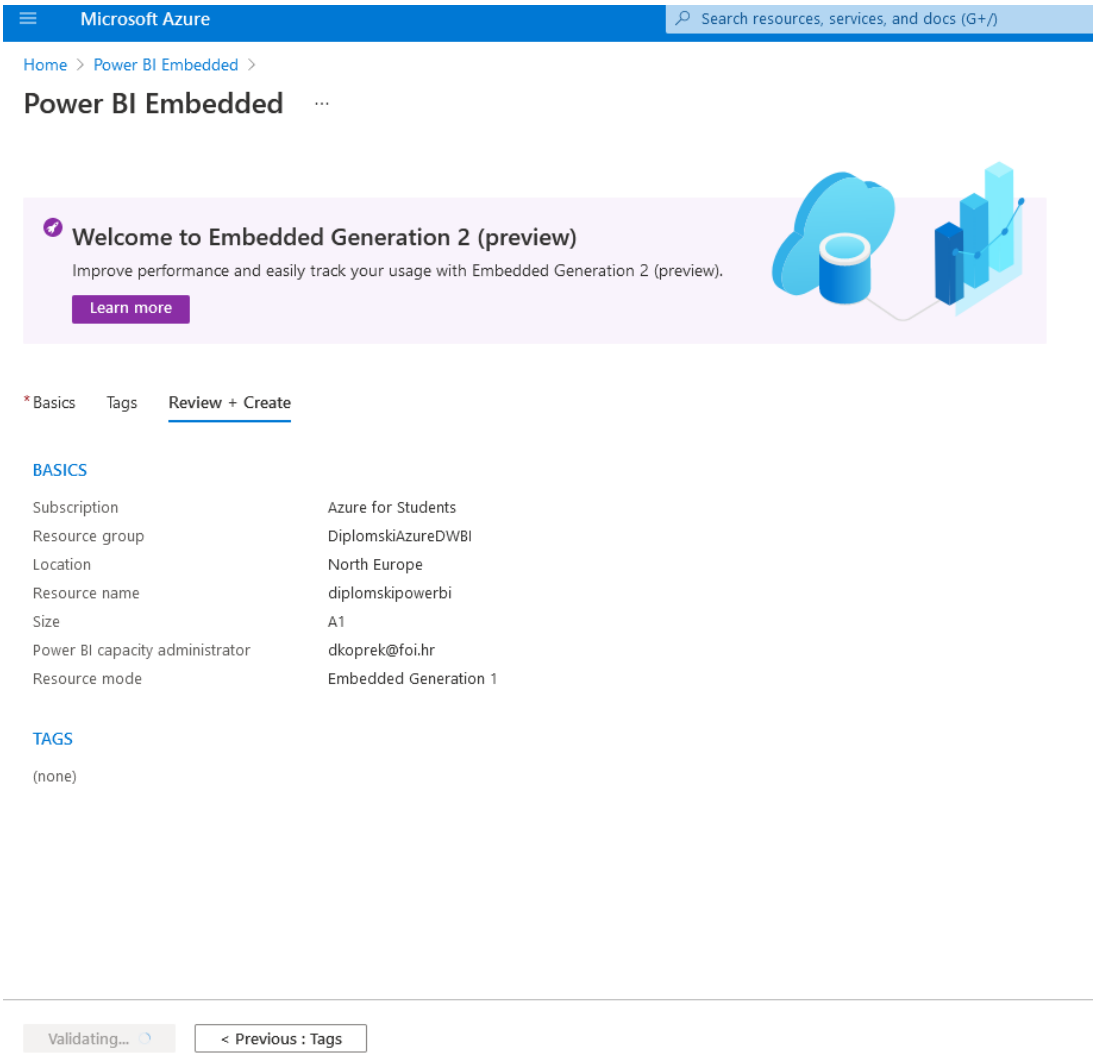
Ograničenja ove licence su ograničene mogućnosti kolaboracije (dostupno samo objavljivanje izvještaja na web, eng. *Publish to web*), mogućnost automatskog osvježavanja podataka samo kod online izvora podataka i to ograničeni broj puta na dan (osam puta s minimalnim razmakom od trideset minuta) i maksimalna veličina modela podataka koji se može objaviti od 1GB te maksimalna veličina svih podataka koji se mogu objaviti od 10GB. [8]

6.1. Power BI Embedded

Power BI Embedded ima licencu u obliku pretplate bazirane na kapacitetu (povećanja kapaciteta u obliku virtualnih CPU jezgri i memorije se kupuju po satu). Namijenjen je korištenju od strane developera za ugradnju Power BI vizualizacija, izvješća i nadzornih ploča (eng. dashboard) u svoje web aplikacije. [8]

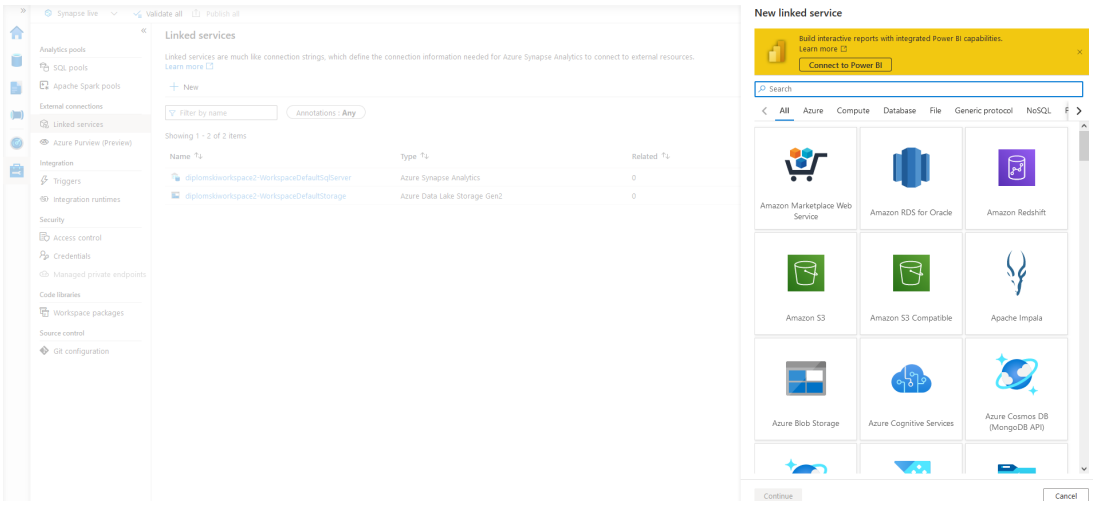
U ovom dijelu će se prikazati izrada *Power BI Embedded* resursa i povezivanje na *Synapse* radni prostor.

Za izradu potrebnog resursa ponovimo postupak sa slike 5, klik na *New resource* na početnoj stranici Azure portala, upisivanje traženog resursa u tražilicu pa potom klik na *Create*. Otvara se proces stvaranja gdje se odabiru opcije i na kraju se vidi pretpregled resursa koji će se stvoriti (slika 23).



Slika 23: Stvaranje Power BI resursa na Azure-u

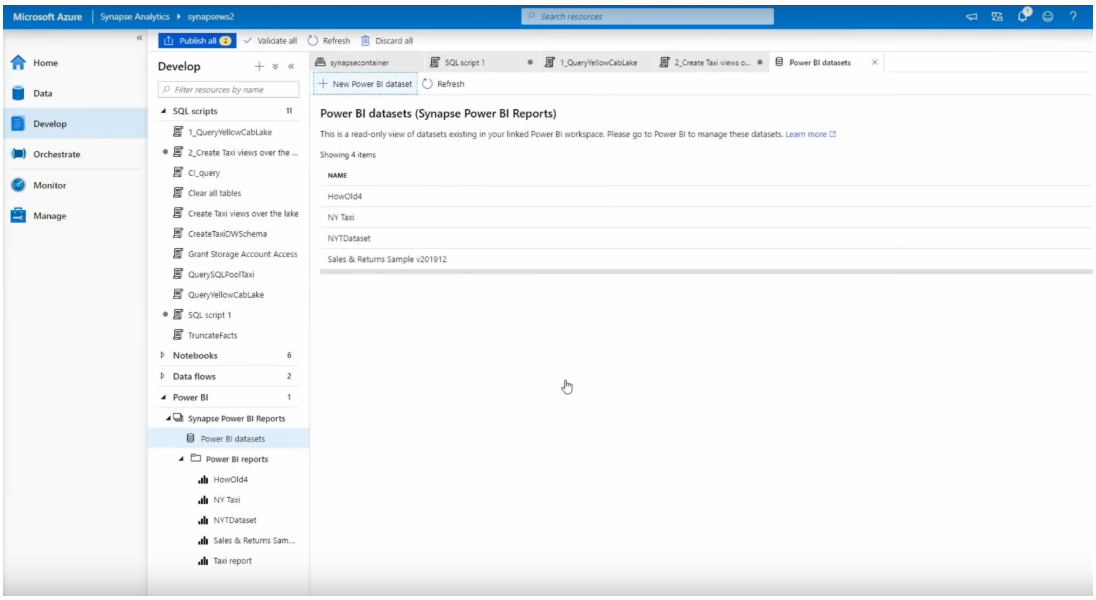
Stvoreni resurs možemo spojiti na postojeći *Synapse* radni prostor klikom na *Manage* u njegovom izborniku pa potom *Linked Services* -> *+ New* i pronalaskom Power BI resursa. U vrijeme pisanja mogućnost spajanja je ponuđena iznad tražilice nakon klika na *+ New* (slika 24).



Slika 24: Povezivanje Synapse Analytics-a i Power BI embedded

Rezultat spajanja je pojavljivanje *Power BI* komponente u *Synapse* radnom prostoru - pod *Develop* u glavnom izborniku vide se skupovi podataka (eng. *datasets*) i izvješća ako su prethodno kreirana.

Ovdje se može kreirati novi *dataset* klikom na *+ New Dataset* pri čemu se odabire bilo koja baza podataka u *Synapse* radnom prostoru nakon čega se stvara *.pbids* (*Power BI dataset*) datoteka. Ona se može otvoriti pomoću *Power BI Desktop* aplikacije čime će ona automatski biti spojena na odabrani izvor (bazu) podataka. [21]

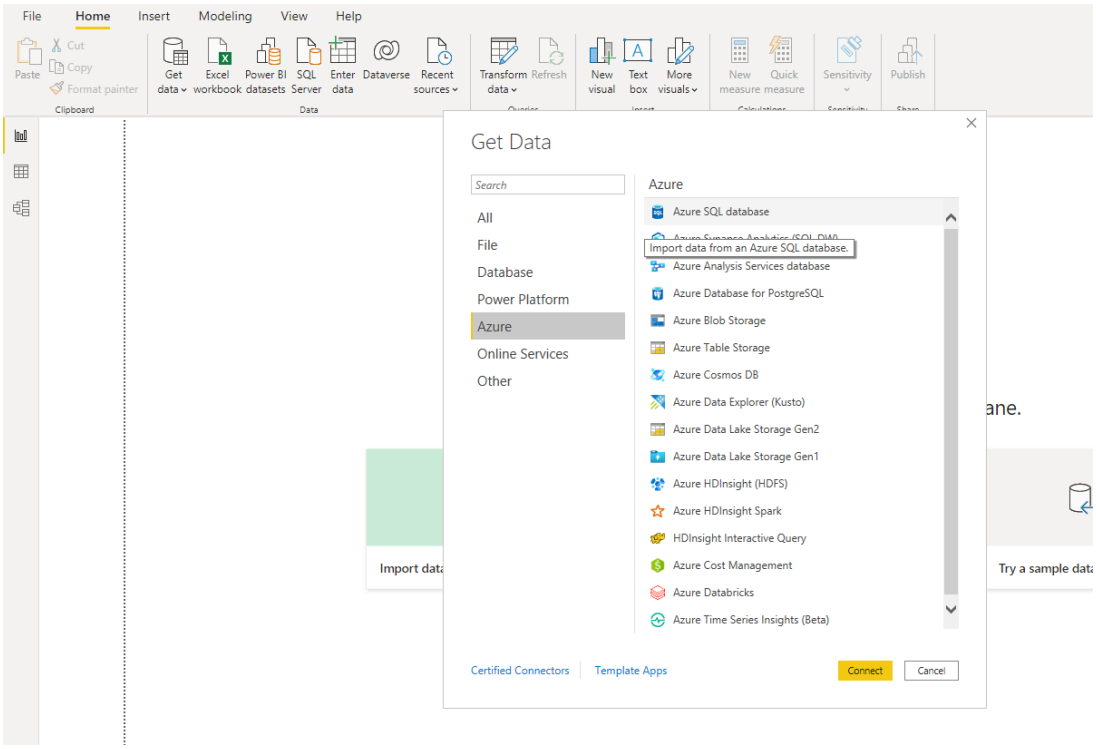


Slika 25: Embedded PowerBI u Synapse radnom prostoru (Izvor: [21])

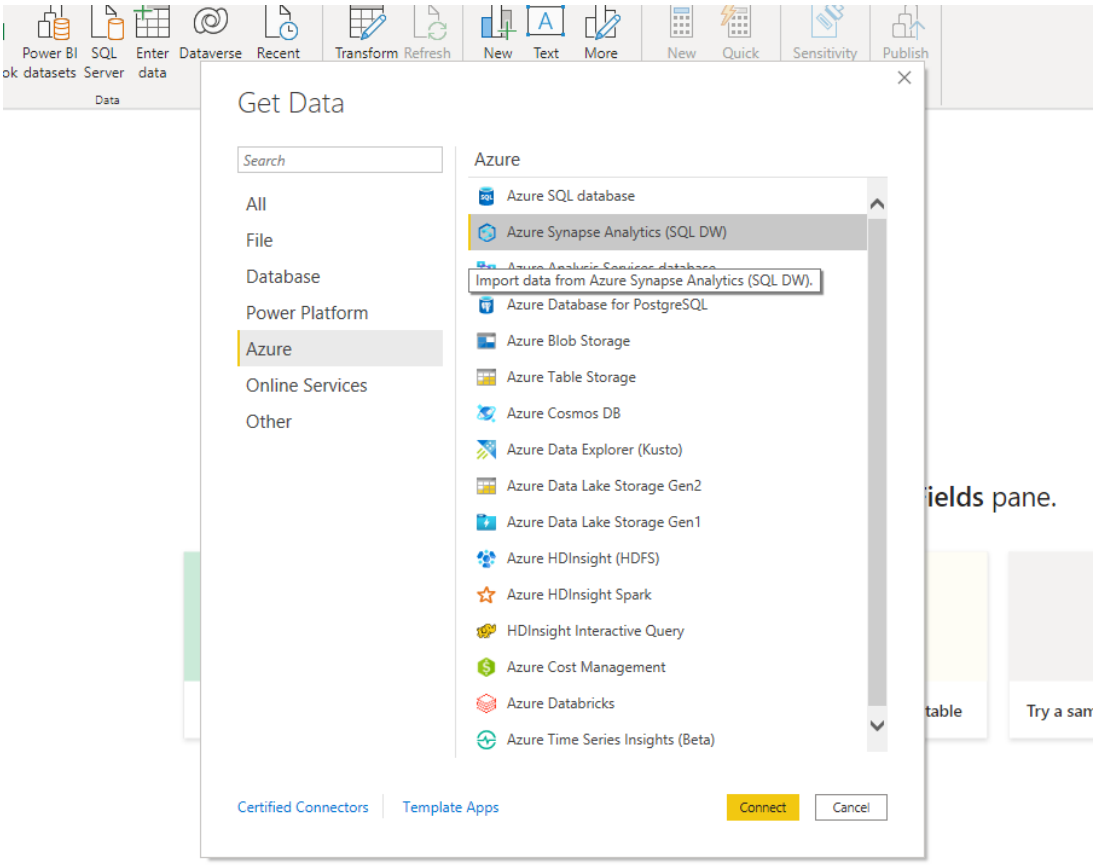
6.2. Kreiranje izvješća u Power BI Desktop

U ovom dijelu biti će prikazano korištenje izrađenih skladišta podataka u *Power BI Desktop* alatu za izradu interaktivnih poslovnih izvještaja.

Ako se ne koristi spomenuta *.pbids* datoteka potrebno se prvo spojiti na neki izvor podataka, odnosno učitati podatke u alat. U ovom slučaju to je jedno od izrađenih skladišta podataka. Odabere se opcija *Get Data* pa zatim se nađe *Azure SQL database* ako se spaja na prvo izrađeno skladište (slika 26) ili *Azure Synapse Analytics (SQL DW)* ako se spaja na drugo izrađeno skladište (slika 27).

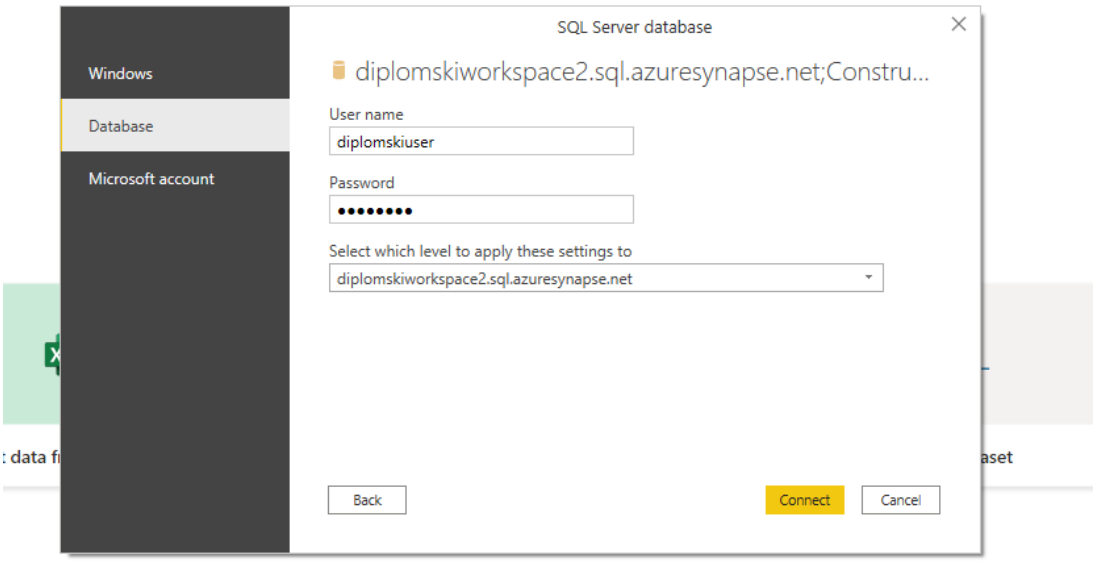


Slika 26: Dobavljanje podataka iz Azure SQL DB-a u Power BI



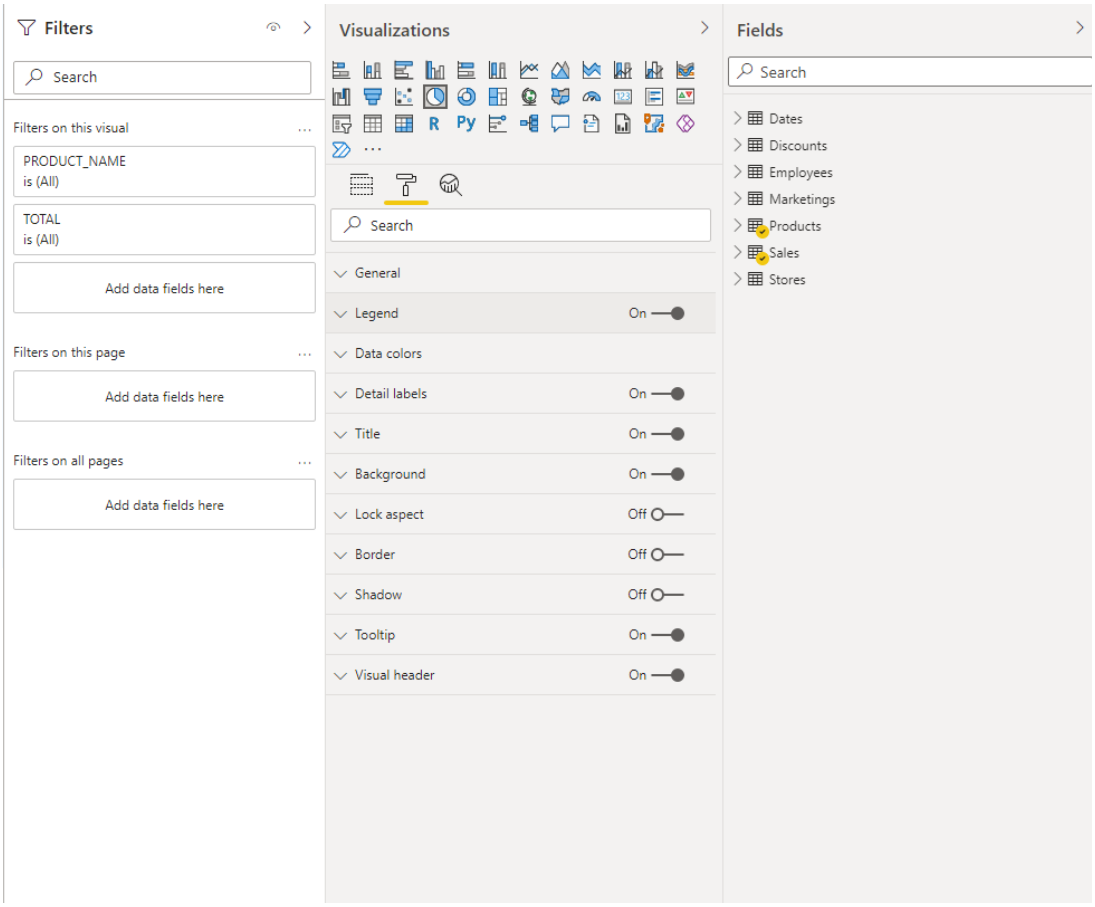
Slika 27: Dobavljanje podataka iz Synapse SQL bazena u Power BI

Upisuje se adresa poslužitelja na kojem je skladište i ime skladišta, a pristupa mu se korisničkim imenom i lozinkom SQL administratora *Azure SQL DB*-a ili *Synapse* radnog prostora (slika 28). Kako su oba skladišta napravljena prema istom modelu i u njih su učitani isti podaci, svejedno je koje se koristi jer će krajnji rezultat (izvještaji i informacije na njima) biti isti.



Slika 28: Spajanje na Azure SQL DB iz Power BI-a

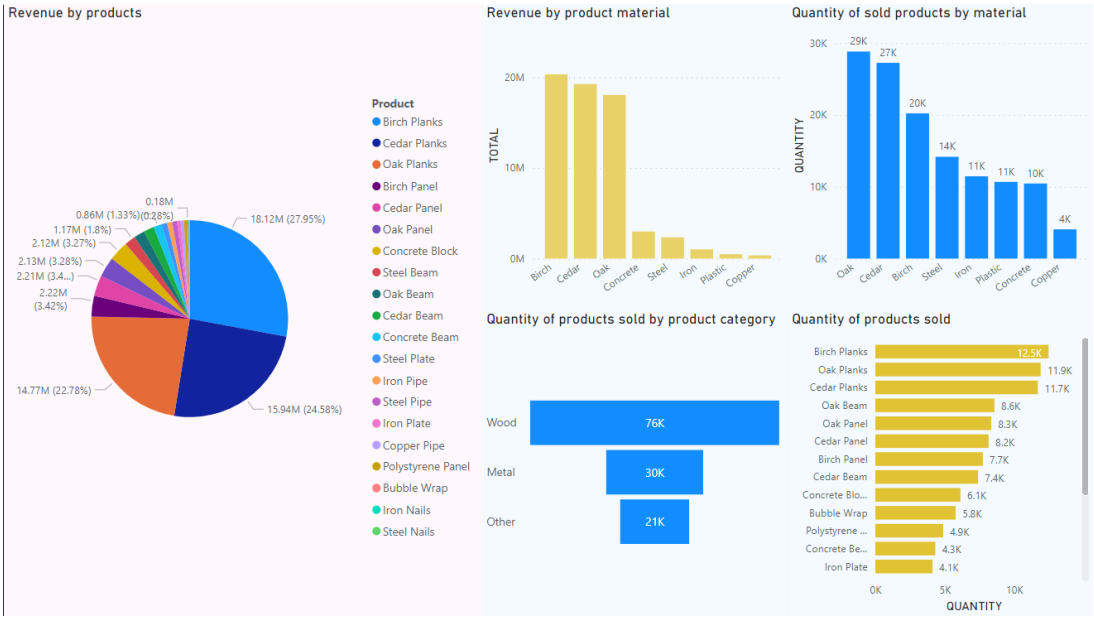
Kad su podaci učitani sve je spremno za izradu izvještaja. Započinje se odabirom neke od vizualizacija u izborniku na desnoj strani. Dostupno je mnogo vrsta koje uključuju stupčaste grafove, tortaste grafove, linearne grafove, dijagram rasipanja i dr. Kad se graf stvori potrebno ga je napuniti podacima. To se može *drag&drop* metodom iz *Fields* sekcije gdje su tablice i pripadni stupci na sam graf ili u polja *Axis, Legend* i *Values* u sekciji ispod izbornika s vizualizacijama. Ovisno o tome u koje od tih polja je postavljen stupac iz tablice će se na grafu pojaviti ti podaci. Podaci iz stupca u *Values* polju se pojavljuju na ordinati grafa, a oni iz stupca u *Axis* polju na apscisi grafa, ako je graf stupčast. Polja Polja *Axis, Legend* i *Values* postoje ovisno o vrsti grafa, npr. *Axis* ne postoji na tortnom grafu. Svaki graf se može formatirati pomoću opcija sa slike 29. Može se uključiti ili isključiti legenda, mijenjati naslov grafa, uključiti ili isključiti oznake podataka, mijenjati pozadina grafa, mijenjati boje stupaca i sličnih elemanata vizualizacije, dodavati granice i sjene, mijenjati fontovi, veličina i položaj teksta i dr. kako bi se grafovi napravili estetski što ugodnijim i/ili prilagodili nekoj temi. Ovo je jednostavan i intuitivan način za izradu i uređivanje poslovnog izvještaja na temelju izrađenog skladišta podataka.



Slika 29: Filteri na vizualizaciji, uređivanje vizualizacije i polja s podacima u Power BI alatu

Izrađen je interaktivni poslovni izvještaj, u slučaju primjera sa slike 30, o ukupnim prihodima i količini prodanih proizvoda razvrstanih po pojedinačnim proizvodima, kategorijama proizvoda i materijali proizvoda. Iz ovog izvješća se može vidjeti da su vodeći proizvod daske od brezovine sa 12.5 tisuća prodanih stavki i prihodom od 18.12 milijuna kuna. Može se i zaključiti da je najnepopularniji materijal bakar s gotovo dvostruko manje prodanih proizvoda od sljedećeg materijala, betona. Također, proizvodi koji se najslabije prodaju su čelični čavli i željezne cijevi. To su samo neke od informacija koje se mogu iščitati iz ovakvog izvještaja.

Kako je naglašeno izvještaj je interaktivan - primjerice može se kliknuti na *Birch* stupac u *Revenue by product material* grafu i na svim ostalim grafovima biti će naznačene one kategorije koje se odnose na brezovinu, dakle svi proizvodi od breze i udio proizvoda od breze u kategoriji svih proizvoda od drva. Podaci se mogu filtrirati ako se npr. žele prikazati proizvodi koji su prodani u količini većoj od neke zadane ili s prihodom većim ili manjim od nekog iznosa. Moguće je povećati jedan od grafova na puni ekran za bolju preglednost ili ga staviti pod reflektor (eng. *spotlight*) da ga se istakne.



Slika 30: Jedan od izrađenih izvještaja u Power BI alatu

Ukupno je izrađeno pet izvještaja koji su priloženi u PDF obliku uz ovaj rad.

7. Zaključak

U ovom radu na početku se prošlo se kroz teorijske osnove i polazišta iz područja skladištenja podataka i poslovne inteligencije. Dan je kratki povijesni pregled pohrane podataka, od magnetskih vrpca preko *spiderweb* okruženja do ideje o skladištima podataka. Skladište podataka je definirano kao temelj za informacijsko procesiranje i podloga za obavljanje složenih analiza nad podacima. Ono je kolekcija subjektno orijentiranih, integriranih, nepromjenjivih podataka koji variraju u vremenu, a služe za stvaranje informacija za podršku poslovnim (menadžerskim odlukama). Navedeni su neki širi ciljevi DW/BI sustava, životni ciklus razvoja skladišta podataka, neke varijacije skladišta podataka i neke komponente u okruženju skladišta podataka poput ETL-a, procesa koji uključuje ekstrakciju podataka iz izvora, transformaciju u potreban oblik i učitavanje u skladište podataka. Objašnjena je tehnika dimenzijskog modeliranja i sheme zvijezde kao modela skladišta te koncept zrnatosti skladišta i njegove prednosti. Poslovna inteligencija je prikazana kao termin koji objedinjuje arhitekture, alate, baze podataka, analitičke alate, aplikacije i metodologije koji služi da organizira ključne informacije potrebne menadžmentu za poboljšanje performansi i profita organizacije. Ključna područja poslovne inteligencije podijeljena su na domenu, podatke, model, analizu i vizualizaciju.

Alati koji su se koristili u praktičnom dijelu su *SQL Server Management Studio 18* za upravljanje skladištima podataka i *Power BI Desktop* za izradu interaktivnih poslovnih izvještaja. Koristili su se i resursi sa Microsoftove platforme *Azure - Synapse Analytics, SQL Database, Data Lake Storage Gen2*, a prikazan je i *Embedded Power BI* resurs.

U praktičnom dijelu izrađen je model skladišta koji će se implementirati koristeći znanje o dimenzijskom modeliranju i shemi zvijezde. Odabrana domena jest prodaja zamišljenog poduzeća koje se bavi građevinskim materijalom. Zatim su se koristeći različite resurse na *Azure*-u izradila dva jednaka skladišta podataka kako bi se pokazale različite mogućnosti koje ona nudi za skladištenje podataka. Jedan način bio je koristeći *Azure SQL DB*, rješenje koje je najsličnije izradi lokalne *SQL server* baze podataka (eng. *on-premise SQL server DB*). Iako namijenjeno klasičnim relacijskim bazama podataka, to rješenje nudi neke prednosti poput inteligentne obrade upita i dobre reaktivnosti na promjene uvjeta u vremenu izvođenja. Za manja skladišta poput ovog u radu pokazalo se i kao isplativije rješenje jer nisu potrebne velike količine resursa i procesorske moći što omogućuje odabir najjeftinije varijante koja u slučaju *Azure SQL DB*-a stoji $15 mjesečno, a u slučaju drugog korištenog načina, *Azure Synapse Analytics*-a $1.5 (€1.27) na sat. Kod drugog načina izrade skladišta na *Azure* platformi izrađen je *Azure Synapse Analytics* radni prostor zajedno sa *Azure Data Lake Storage Gen2* pohranom. *Azure Synapse Analytics* je masovno paralelno procesirajući stroj (eng. *massively parallel processing engine - MPP engine*) za pohranu i obradu velike količine podataka na *Azure* platformi. Unutar *Synapse* radnog prostora izrađen je posvećeni SQL bazen (*dedicated SQL pool*) koji je tako preimenovan od strane Microsofta iz svog prvotnog naziva *SQL Data Warehouse*. Na njega se, kao i na *Azure SQL DB* spojilo pomoću *SQL Server Management Studio* alata. U prvom skladištu tablice su kreirane i podaci učitani koristeći ugrađene mogućnosti alata koje dopuštaju uvoz iz polustrukturiranog izvora poput CSV datoteka, a u drugom skladištu su tablice kreirane preko SQL naredbi dok su podaci učitani korištenjem *COPY INTO* SQL naredbe. Rezultat su

dva jednaka skladišta napravljena na dva različita načina koristeći različite resurse dostupne na *Azure* platformi.

Za potrebe poslovne inteligencije iz tih skladišta učitani su podaci u *Power BI* alat i uz mogućnosti koje nudi taj Microsoftov alat za vizualizaciju podataka napravljeno je pet interaktivnih izvještaja kako bi prikazali informacije o prodaji. Iz njih možemo primjerice zaključiti da su najpopularniji drveni proizvodi (naročito brezovina), da je najviše prihoda među zaposlenicima donio Goran Goranić iako Helena Helenić ima najveći prosječni prihod po prodaji, da je najviše proizvoda prodano u regijama Sjeverna i Južna Hrvatska, da je od popusta najuspješniji bio popust za praznik rada s 33 tisuće prodanih proizvoda, da se najviše čeličnih greda prodaje u proljeće i još mnogo sličnih informacija koje bi mogle zanimati menadžment.

Tako se prošlo od izrade modela skladišta pa sve do konkretnih, lako čitljivih informacija koje mogu poslužiti u donošenju odluka za poboljšanje poslovanja koristeći usvojena teoretska znanja i mogućnosti *Azure* platforme.

Popis literature

[1] R. Kimball i M. Ross, *The data warehouse toolkit: the definitive guide to dimensional modeling*, Third edition. John Wiley i Sons, Inc, 2013.

[2] W. H. Inmon, D. Strauss i G. Neushloss, *DW 2.0: the architecture for the next generation of data warehousing*. Morgan Kaufmann, 2008.

[3] K. Rabuzin, *SQL: napredne teme*. Fakultet organizacije i informatike Sveučilišta u Zagrebu, 2014.

[4] A. Vaisman, *Data warehouse systems: design and implementation*. Springer, 2014., ISBN: 9783642546549.

[5] W. H. Inmon, *Building the data warehouse*, 4th ed. Wiley, 2005.

[6] S. Williams i N. Williams, *The profit impact of business intelligence*. Morgan Kaufmann, 2007.

[7] R. Sharda, D. Delen i E. Turban, *Business intelligence, analytics, and data science: a managerial perspective*, Fourth edition. Pearson, 2018.

[8] G. Deckler, *Learn Power BI: a beginner's guide to developing interactive business intelligence solutions using Microsoft Power BI*. Packt Publishing Ltd., 2019.

[9] Microsoft, *What is Azure SQL Database?* Pristupljeno: 7.9.2021. adresa: `https://docs.microsoft.com/en-us/azure/azure-sql/database/sql-database-paas-overview`.

[10] M. How, *The modern data warehouse in Azure: building with speed and agility on Microsoft's Cloud Platform*. Apress, 2020.

[11] Microsoft, *Azure Portal*. adresa: `https://portal.azure.com/#home`.

[12] ——, *Quickstart: Use SSMS to connect to and query Azure SQL Database or Azure SQL Managed Instance*, Pristupljeno: 7.9.2021. adresa: `https://docs.microsoft.com/en-us/azure/azure-sql/database/connect-query-ssms`.

[13] K. Rabuzin, *Uvod u SQL*. Fakultet organizacije i informatike Sveučilišta u Zagrebu, 2011.

[14] Microsoft, *Create Foreign Key Relationships*, Pristupljeno: 7.9.2021. adresa: `https://docs.microsoft.com/en-us/sql/relational-databases/tables/create-foreign-key-relationships?view=sql-server-ver15`.

[15] *Azure Synapse Analytics - Introduction & Overview*, [YouTube],objavio Advancing Analytics,16.6.2020. Pristupljeno 8.9.2021. adresa: `https://www.youtube.com/watch?v=2DX7dgR8cEw`.

[16] ——, *What is Azure Synapse Analytics?* Pristupljeno: 8.9.2021. adresa: https://docs.microsoft.com/en-us/azure/synapse-analytics/overview-what-is.

[17] ——, *Quickstart: Create and query a dedicated SQL pool (formerly SQL DW) in Azure synapse Analytics using the Azure portal*, Pristupljeno: 8.9.2021. adresa: https://docs.microsoft.com/en-us/azure/synapse-analytics/sql-data-warehouse/create-data-warehouse-portal#connect-to-the-server-as-server-admin.

[18] ——, *Tutorial: Load the New York Taxicab dataset*, Pristupljeno: 8.9.2021. adresa: https://docs.microsoft.com/en-us/azure/synapse-analytics/sql-data-warehouse/load-data-from-azure-blob-storage-using-copy.

[19] ——, *Data loading strategies for dedicated SQL pool in Azure Synapse Analytics*, Pristupljeno: 8.9.2021. adresa: https://docs.microsoft.com/en-us/azure/synapse-analytics/sql-data-warehouse/design-elt-data-loading.

[20] ——, *COPY (Transact-SQL)*, Pristupljeno: 8.9.2021. adresa: https://docs.microsoft.com/en-us/sql/t-sql/statements/copy-into-transact-sql?view=azure-sqldw-latest&preserve-view=true.

[21] *Azure Synapse Analytics & Power BI for BANANAS SCALE!* [YouTube],objavio Guy in a Cube,3.6.2020. Pristupljeno 8.9.2021. adresa: https://www.youtube.com/watch?v=vOSzw40TohM.

Popis slika

Popis tablica

1. Prilog 1: Korišteni podaci (CSV datoteke)

- dates.csv
- discounts.csv
- employees.csv
- marketings.csv
- products.csv
- sales.csv
- stores.csv

2. Prilog 2: Izrađeni poslovni izvještaji (PDF datoteka)

- Reports.pdf