Modeliranje i razvoj sustava potpore odlučivanju za upravljanje globalnom maloprodajom

Dudaković, Timon

Sveučilište Jurja Dobrile u Puli
Odjel za informacijsko-komunikacijske tehnologije

TIMON DUDAKOVIĆ

Modeliranje i razvoj sustava potpore odlučivanju za upravljanje globalnom maloprodajom

Završni rad

Pula, 28.06., 2020. godine

Sveučilište Jurja Dobrile u Puli
Odjel za informacijsko-komunikacijske tehnologije

TIMON DUDAKOVIĆ

Modeliranje i razvoj sustava potpore odlučivanju za upravljanje globalnom maloprodajom

Završni rad

JMBAG: 0303076148, redoviti student

Studijski smjer: Informatika

Kolegij: Sustavi poslovne inteligencije

Mentor: doc. dr. sc. Goran Oreški

Pula, 28.06., 2020. godine

IZJAVA O AKADEMSKOJ ČESTITOSTI

Ja, dolje potpisani Timon Dudaković, ovime izjavljujem da je ovaj završni rad rezultat isključivo mojega vlastitog rada, da se temelji na mojim istraživanjima te da se oslanja na objavljenu literaturu kao što to pokazuju korištene bilješke i bibliografija. Izjavljujem da niti jedan dio završnog rada nije napisan na nedozvoljen način, odnosno da je prepisan iz kojega necitiranog rada, te da ikoji dio rada krši bilo čija autorska prava. Izjavljujem, također, da nijedan dio rada nije iskorišten za koji drugi rad pri bilo kojoj drugoj visokoškolskoj, znanstvenoj ili radnoj ustanovi.

Student

U Puli, 28.06., 2020. godine

Sadržaj

1. Uvod

S obzirom na to da količina podataka pohranjenih u tvrtkama eksponencijalno raste, nije iznenađenje što se pronalazak pravog rješenja za upravljanje podacima nastavlja pojavljivati na popisu prioriteta glavnih službenika za informiranje. Podaci moraju biti sigurni i učinkovito distribuirani za važne ažurne poslovne odluke.

Tvrtke moraju prevesti podatke u informacije kako bi isplanirale buduće poslovne strategije. Za većinu tvrtki, vrijedni podaci pohranjuju se u ogromne proračunske tablice ili servere. U idealnom slučaju, ti podaci bi im trebali pružiti informacije o trendovima prodaje, ponašanju potrošača i raspodjeli resursa. Podaci tvrtke mogu ukazivati na održivost njihovog proizvoda i pomoći u planiranju njihovog budućeg rasta. Stoga, podaci mogu pomoći u povećanju prihoda i smanjenju troškova.

Sustav poslovne inteligencije kao rješenje pomaže u stvaranju točnih izvještaja vađenjem podataka izravno iz dotičnog izvora podataka. Sa današnjim sustavima poslovne inteligencije eliminira se dugotrajan proces ručne konsolidacije podataka. Budući da alati korišteni kod razvoja takvih sustava mogu proizvesti nedavne podatke, omogućuju menadžerima nadzor nad tvrtkama u stvarnom vremenu. Sustavi poslovne inteligencije izravno pružaju menadžerima izvješća u stvarnom vremenu s bilo koje lokacije. To pomaže smanjiti opseg pogrešaka pružajući menadžerima točne podatke za donošenje boljih odluka o onome što se događa i za predviđanje onoga što će se dogoditi. Sustavi poslovne inteligencije također su usmjereni na pružanje sigurnosti podataka koristeći postojeće sigurnosne infrastrukture za čuvanje podataka.

U ovom završnom radu promatrati ćemo razvoj i modeliranje sustava potpore odlučivanju, skladišta podataka koji podržava inkrementalan ETL proces i aplikacije za tokove podataka za upravljanje globalnom maloprodajom. Alati koji su se koristili kod praktičnog dijela završnog rada su MySQL kao baza podataka, Python programski jezik za izradu i punjenje baze podataka, Pentaho za integraciju podataka, OLAP alati za njihovu analizu i vizualizaciju te Apache Flink za tokove podataka.

2. Analiza podataka i opis završnog rada

2.1. Analiza podataka

Najvažniji aspekt svakog BI (Business Intelligence), BA (Business Analytics) ili "Data Science" sustava su podaci. Podaci nam daju uvid u poslovanje tvrtke te nam omogućuju definiranje problema koji se treba riješiti. Podaci predstavljaju skup činjenica o dotičnoj tvrtki. Osnovni aspekti podataka koji se moraju promotriti prilikom njihove analize su količina, tipovi podataka, raznolikost izvora te kvaliteta. Detaljnijom analizom tih aspekata možemo utvrditi nekakve karakteristike podataka koje određuju spremnost podataka za izradu modela poslovne inteligencije. Uz točnost sadržaja, njihovu potpunost, konzistentnost, granularnost i mnoge druge, važno nam je da postoji vremenska dimenzija kako bi mogli pratiti poslovanje tvrtke kroz vrijeme.

U ovom završnom radu ćemo promatrati poslovanje globalne maloprodaje u periodu od 2011. do 2014. godine. Skup podataka je javno dostupan (licenca), te je preuzet sa sljedećeg linka: https://www.kaggle.com/jr2ngb/superstore-data. Prilikom analize podataka, došli smo do sljedećih rezultata:

- Količina podataka – postoji 51290 redova podataka
- Tipovi podataka – postoje stringovi, datumi i brojevi
- Raznolikost podataka – postoje 3 složene dimenzije, 2 degenerirane dimenzije i vremenska dimenzija
- Potpunost podataka – svi bitni podaci su potpuni

2.2. Opis završnog rada

U ovom završnom radu ću objasniti proces izrade područnog skladišta podataka za upravljanje globalnom maloprodajom koje će služiti kao potporni sustav kod procesa donošenja važnih poslovnih odluka.

Tvrtka "Superstore" d.o.o. bavi se globalnom maloprodajom. Proizvodi se prodaju u 147 država diljem svijeta. Postoji više kategorija i pod-kategorija proizvoda. Narudžbe se provode putem Interneta te se proizvodi šalju odabranim načinom otpreme. Svaki kupac pripada određenom segmentu. Uz kupca, proizvod i lokaciju, svaka narudžba ima svoj datum, prioritet narudžbe, vrstu otpreme, cijenu otpreme, količinu, popust, prihod i profit. Vremenski period u kojem se promatra poslovanje tvrtke od 2011. do 2014. godine.

Glavni cilj završnog rada je definirati broj prodanih proizvoda i ukupan profit u definiranom vremenskom periodu pomoću implementiranog skladišta podataka, implementirati inkrementalno punjenje područnog skladišta podataka kako bi omogućili

kontinuirano osvježavanje istog, i prikazati kako upravljati sa podacima u stvarnom vremenu koristeći Apache Flink.

3. Transakcijski sustav

3.1. Uvod u transakcijske sustave

Transakcijski (ili operativni) sustav direktno podržava izvršavanje poslovnih procesa. Bilježi detalje o događajima i transakcijama unutar tvrtke kako bi evidentirao aktivnosti poduzeća. Takvi se sustavi također zovu i OLTP (Online Transaction Processing) sustavi. Takvi sustavi mogu biti CRM sustavi, ERP sustavi i ostale aplikacije koje evidentiraju transakcije i aktivnosti unutar tvrtke. Transakcijski sustav koji je implementiran u relacijskog bazi podataka u pravilu treba biti u trećoj normalnoj formi kako bi ga smatrali optimalnim. Takav shematski dizajn nazivamo ER (Entity-Relationship) model. On osigurava visoko optimizirano izvođenje operacija brisanja, ažuriranja i stvaranja transakcija (Anon., 2012.). Tablica 1. prikazuje razlike između transakcijskih (OLTP) sustava i skladišta podataka, odnosno analitičkih (OLAP) sustava.

Skladište podataka (OLAP)	Transakcijski sustav (OLTP)
Sadrži povijesne podatke	Sadrži trenutne podatke
Vrlo je fleksibilan	Vrlo je optimiziran
Pruža sažete i konsolidirane podatke	Pruža primitivne i vrlo detaljne podatke
Poslužuje mali broj korisnika	Poslužuje veliki broj korisnika
Pristupa velikim količinama podataka	Pristupa malim količinama podataka

Tablica 1. Prikaz razlika skladišta podataka i transakcijskog sustava (Mekterović & Brkić, 2017.)

Transakcijski sustav, za razliku od analitičkog sustava, stvoren je da podržava poslovne aktivnosti i procese, a ne da analizira podatke. Stoga, bilo koji upit može usporiti performanse transakcijske baze podataka (Anon., 2012.).

Najčešći načini organizacije i pohrane transakcijskih podataka su:

- Datotečni sustav – koristi se kada se upravlja manjim količinama podataka koji su često nepovezani. Optimalno je izvoditi jednostavne operacije kao što su pisanje i čitanje te se zna često javiti redundancija, zavisnost i niska produktivnost (Coronel & Morris, 2016.)

- Relacijska baza podataka – neophodna je za vođenje transakcijskog sustava današnjih organizacija. Potrebno ju je dizajnirati da efikasno podržava poslovanje organizacije (Coronel & Morris, 2016.)

Postoje tri pristupa dizajna baze podataka:

- Konceptualni dizajn – napraviti model podataka neovisno o DBMS-u
- Logički dizajn – napraviti bazu podataka u danom DBMS-u
- Fizički dizajn – kako je baza spremljena na hardveru

Rezultat dizajna su modeli, odnosno apstraktna reprezentacija stvarnosti koja isključuje mnoštvo detalja iz stvarnog svijeta. Time se smanjuje kompleksnost i obraća pažnja samo na bitne detalje.

3.2. Entity/Relationship modeliranje

Entity/Relationship model je podatkovni model koji opisuje veze među entitetima na konceptualnoj razini uz pomoć Entity/Relationship dijagrama. Entity/Relationship dijagram prikazuje odnos entiteta, atributa i veza modela. Skup entiteta predstavlja objekt interesa za krajnjeg korisnika te kod Entity/Relationship modeliranja predstavljaju jednu tablicu, a ne jedan redak u tablici. Najčešće su prikazani kao pravokutnici koji imaju smisleno ime koje jednoznačno opisuje taj entitet. Svaki entitet ima svoja svojstva, odnosno atribute. Najčešće su prikazani kao ovalni oblici koji imaju ime te predstavljaju činjenice koji nam daju detaljniji uvid u entitet kojem su pridruženi neprekidnom ravnom linijom. Kako bi se definirao odnos između dva i više entiteta oni moraju biti povezani (Coronel & Morris, 2016.). Kardinalnost veze predstavlja svojstvo funkcionalnosti i obaveznosti. Tri glavna kardinalna odnosa su:

- Jedan naprema jedan (One-To-One)
- Jedan naprema više (One-To-Many)
- Više naprema više (Many-To-Many)

Prilikom stvaranja konceptualnog modela potrebno je identificirati takve entitete, atribute, veze i kardinalnosti veza kako bi se stvorio Entity/Relationship model. Identifikacija navedenih komponenata dijagrama se vrši iz opisa problema.

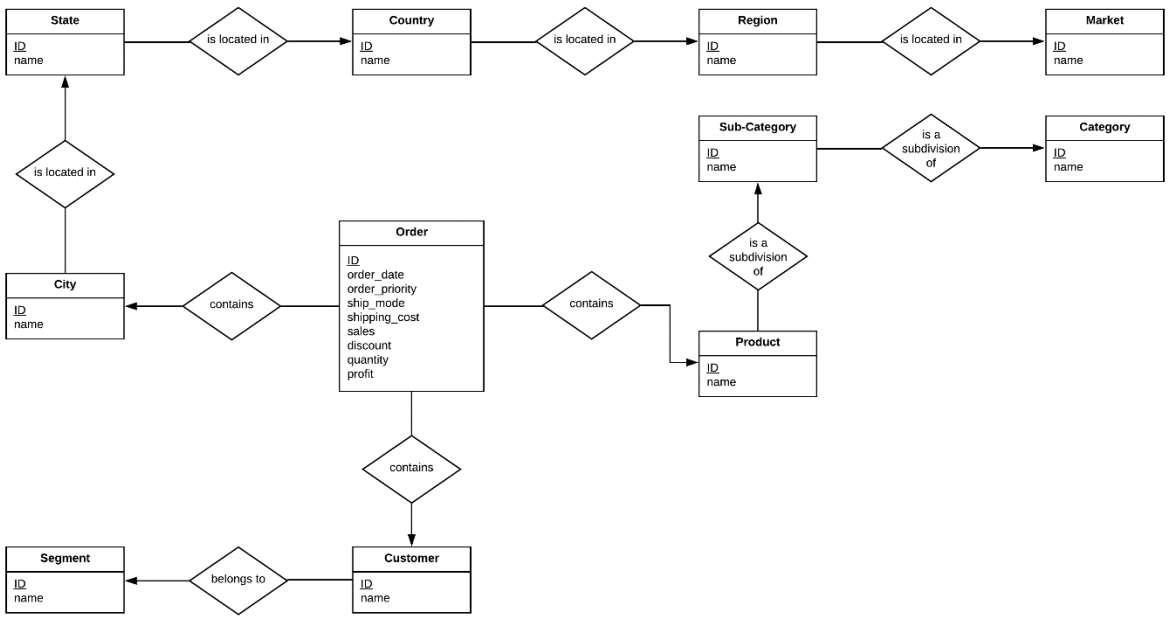
Iz opisa završnog rada možemo identificirati 13 tablica koji tvore relacijsku bazu podataka. Spomenuti entiteti su market, regija, zemlja, država, grad, kategorija proizvoda, pod-kategorija proizvoda, proizvod, segment, kupac, prioritet narudžbe, vrste otpreme i narudžba. Entitet koji povezuje sve ostale entitete je “Narudžba“. Ona prati sve bitne podatke koji omogućuju detaljnu evidenciju transakcija, kao što su količina i vrsta

prodane robe, vrsta i cijena otpreme, profite i popuste te lokacije kupaca. Svaki proizvod ima svoju pod-kategoriju koja pripada određenoj nad-kategoriji.

Također je sadržana velika hijerarhija relacijskih tablica koje predstavljaju lokaciju, od grada – najniže razine – do marketa – najviše razine (globalne). Uz to je definiran i segment koji omogućuje praćenje određene skupine kojima kupci pripadaju. Iz opisa problema možemo vidjeti da svaki entitet ima svoj identifikacijski broj (primarni ključ) te odgovarajuće ime.

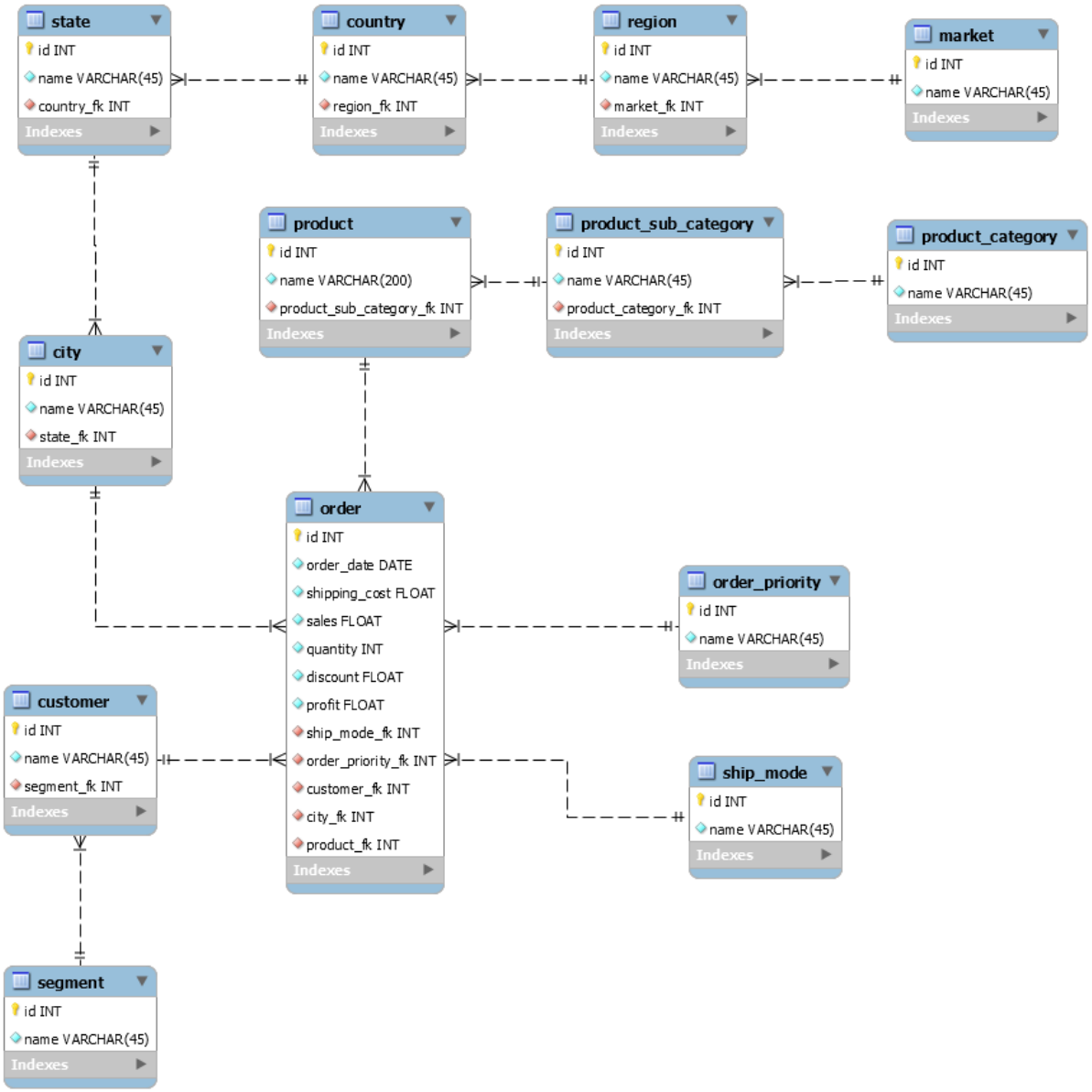
Nakon identifikacije atributa potrebno je pronaći veze entiteta te njihovu kardinalnost. Veze su predstavljene glagolom koji se na dijagramu prikazuje u obliku romba, te svi entiteti koriste kardinalnost jedan naprema više, gdje je viša razina uvijek na lijevoj strani veze (jedan), a niža na desnoj (više).

Izgrađeni konceptualni model koristi se kao polazište za izgradnju logičkog modela (slika 1.). Takav model visoke razine opisuje sve entitete, atribute, veze i kardinalnosti navedene u prethodnom dijagramu na korisnicima razumljiviji način. U logičkom modelu, atributi se nalaze unutar entiteta dok su veze entiteta i kardinalnosti veza prikazane kao linije sa odgovarajućom oznakom na svakom kraju.



Slika 1. Prikaz konceptualnog modela

Iz prikazanog logičkog modela možemo vidjeti da su nazivi tablica i atributa ostali isti, te je svakom entitetu niže razine pridružen strani ključ koji predstavlja poveznicu na odgovarajući entitet više razine (slika 2.).



Slika 2. Prikaz logičkog modela baze podataka

3.3. Punjenje baze podataka sa podacima

Nakon izrade Entity/Relationship modela slijedi punjenje modela sa podacima. Kao bazu podataka u završnom radu sam koristio MySQL. MySQL je besplatan sustav otvorenog koda za upravljanje bazom podataka te se prikazao kao savršen kandidat budući da je visoko kompatibilan sa Pentaho sustavom integracije koji će se koristiti prilikom punjenja dimenzijskog modela podataka. Za stvaranje tablica i punjenje baze podataka podacima koristio sam programski jezik Python. Prije samog punjenja baze podataka, bilo je potrebno očistiti neke podatke iz CSV datoteke kako bi bili dovedeni do formata koji je prihvatljiv MySQL bazi podataka. Sam proces izrade tablica i učitavanja podataka bio je jednostavan zbog modula koje nam Python pruža koji ubrzavaju takve procese. Slika 3. prikazuje dio podataka iz izvorne CSV datoteke.

	A	B	C	D	E	F	G	H	I	J	K
1	Row ID	Order ID	Order Date	Ship Date	Ship Mode	Customer	Customer	Segment	City	State	Country
2	42433	AG-2011-2	1/1/2011	6/1/2011	Standard C	TB-11280	Toby Braur	Consumer	Constantir	Constantir	Algeria
3	22253	IN-2011-4	1/1/2011	8/1/2011	Standard C	JH-15985	Joseph Ho	Consumer	Wagga Wa	New Soutl	Australia
4	48883	HU-2011-1	1/1/2011	5/1/2011	Second Cla	AT-735	Annie Thur	Consumer	Budapest	Budapest	Hungary
5	11731	IT-2011-3E	1/1/2011	5/1/2011	Second Cla	EM-14140	Eugene Mo	Home Offi	Stockholm	Stockholm	Sweden
6	22255	IN-2011-4	1/1/2011	8/1/2011	Standard C	JH-15985	Joseph Ho	Consumer	Wagga Wa	New Soutl	Australia
7	22254	IN-2011-4	1/1/2011	8/1/2011	Standard C	JH-15985	Joseph Ho	Consumer	Wagga Wa	New Soutl	Australia
8	21613	IN-2011-3(	1/2/2011	3/2/2011	Second Cla	PO-18865	Patrick O'I	Consumer	Dhaka	Dhaka	Bangladesl
9	34662	CA-2011-1	1/2/2011	3/2/2011	First Class	LC-17050	Liz Carlisle	Consumer	Mission Vi	California	United Sta
10	44508	AO-2011-1	1/2/2011	4/2/2011	Second Cla	DK-3150	David Kenc	Corporate	Luanda	Luanda	Angola
11	23688	ID-2011-5(	1/2/2011	3/2/2011	Second Cla	SP-20650	Stephanie	Corporate	Yingcheng	Hubei	China
12	25293	IN-2011-3(	1/2/2011	5/2/2011	Second Cla	DK-13150	David Kenc	Corporate	Chongqing	Chongqing	China
13	8483	US-2011-1	1/2/2011	6/2/2011	Standard C	DH-13075	Dave Halls	Corporate	San Migue	Panama	Panama
14	41445	IR-2011-65	1/2/2011	6/2/2011	Standard C	PO-8850	Patrick O'I	Consumer	Mashhad	Razavi Kho	Iran
15	16727	ES-2011-5	1/2/2011	3/2/2011	Second Cla	GH-14485	Gene Hale	Corporate	La Rochell	Poitou-Cha	France
16	21615	IN-2011-3(	1/2/2011	3/2/2011	Second Cla	PO-18865	Patrick O'I	Consumer	Dhaka	Dhaka	Bangladesl
17	8484	US-2011-1	1/2/2011	6/2/2011	Standard C	DH-13075	Dave Halls	Corporate	San Migue	Panama	Panama
18	19796	ES-2011-5	1/2/2011	5/2/2011	Standard C	RR-19315	Ralph Ritte	Consumer	Parma	Emilia-Ron	Italy
19	21614	IN-2011-3(	1/2/2011	3/2/2011	Second Cla	PO-18865	Patrick O'I	Consumer	Dhaka	Dhaka	Bangladesl
20	21616	IN-2011-3(	1/2/2011	3/2/2011	Second Cla	PO-18865	Patrick O'I	Consumer	Dhaka	Dhaka	Bangladesl

Slika 3. Prikaz podataka iz CSV datoteke

Najveća hijerarhija relacijskih tablica odnosi se na lokacijske tablice. Ona je predstavljena tablicama "Market", "Regija", "Država", "Zemlja" i "Grad" gdje svaki redak u tablici sadrži jedinstveni identifikator, ime, i strani ključ kao poveznica na odgovarajuću roditeljsku tablicu (slike 4. i 5.). Tablica "Proizvod" osim jedinstvenog identifikatora i imena sadrži poveznicu na tablicu "Pod-kategorija", koja je podređena tablici "Kategorija" (slike 6. i 7.). Najmanja hijerarhija relacijskih tablica sadržana je u tablicama koje se odnose na kupca. Tablica "Kupac" podređena je tablici "Segment" (slika 8.). Tablice "Način otpreme" i "Prioritet narudžbe" najjednostavnije su relacijske tablice koje će prilikom

izrade dimenzijskog modela podataka služiti kao degenerirane dimenzije (slika 9.). Sve navedene tablice se vežu na tablicu "Narudžba" koja sadrži i određene mjere kao što su popust, profit, kvantiteta proizvoda i slično (slika 10).

id	name
1	Africa
2	APAC
7	Canada
3	EMEA
4	EU
6	LATAM
5	US

id	name	market_fk
1	Africa	1
2	Oceania	2
3	EMEA	3
4	North	4
5	Central Asia	2
6	West	5
7	North Asia	2
8	Central	6
9	South	4
10	Canada	7

id	name	region_fk
1	Algeria	1
2	Australia	2
3	Hungary	3
4	Sweden	4
5	Bangladesh	5
6	United States	6
7	Angola	1
8	China	7
9	Panama	8
10	Iran	3

Slika 4. Prikaz relacijskih tablica "Market", "Regija" i "Zemlja"

id	name	country_fk
1	Constantine	1
2	New South Wales	2
3	Budapest	3
4	Stockholm	4
5	Dhaka	5
6	California	6
7	Luanda	7
8	Hubei	8
9	Chongqing	8
10	Panama	9

id	name	state_fk
1	Constantine	1
2	Wagga Wagga	2
3	Budapest	3
4	Stockholm	4
5	Dhaka	5
6	Mission Viejo	6
7	Luanda	7
8	Yingcheng	8
9	Chongqing	9
10	San Miguelito	10

Slika 5. Prikaz relacijskih tablica "Država" i "Grad"

id	name
2	Furniture
1	Office Supplies
3	Technology

id	name	product_category_fk
1	Storage	1
2	Supplies	1
3	Paper	1
4	Furnishings	2
5	Copiers	3
6	Bookcases	2
7	Appliances	1
8	Art	1
9	Accessories	3
10	Binders	1

Slika 6. Prikaz relacijskih tablica "Kategorija" i "Pod-kategorija"

	id	name	product_sub_category_fk
▶	1	Tenex Lockers, Blue	1
	2	Acme Trimmer, High Speed	2
	3	Tenex Box, Single Width	1
	4	Enermax Note Cards, Premium	3
	5	Eldon Light Bulb, Duo Pack	4
	6	Eaton Computer Printout Paper, 8.5 x 11	3
	7	Brother Personal Copier, Laser	5
	8	Sauder Facets Collection Library, Sky Alder Finish	6
	9	Fellowes Lockers, Wire Frame	1
	10	Tenex Trays, Single Width	1

Slika 7. Prikaz relacijske tablice "Proizvod"

	id	name
▶	1	Consumer
	3	Corporate
	2	Home Office

	id	name	segment_fk
▶	1	Toby Braunhardt	1
	2	Joseph Holt	1
	3	Annie Thurman	1
	4	Eugene Moren	2
	5	Patrick O'Donnell	1
	6	Liz Carlisle	1
	7	David Kendrick	3
	8	Stephanie Phelps	3
	9	Dave Hallsten	3
	10	Patrick O'Brill	1

Slika 8. Prikaz relacijskih tablica "Segment" i "Kupac"

	id	name
▶	3	First Class
	4	Same Day
	2	Second Class
	1	Standard Class

	id	name
▶	3	Critical
	2	High
	4	Low
	1	Medium

Slika 9. Prikaz relacijskih tablica "Način otpreme" i "Prioritet narudžbe"

	id	order_date	shipping_cost	sales	quantity	discount	profit	ship_mode_fk	order_priority_fk	customer_fk	city_fk	product_fk
▶	1	2011-01-01	35.46	408.3	2	0	106.14	1	1	1	1	1
	2	2011-01-01	9.72	120.366	3	0.1	36.036	1	1	2	2	2
	3	2011-01-01	8.17	66.12	4	0	29.64	2	2	3	3	3
	4	2011-01-01	4.82	44.865	3	0.5	-26.055	2	2	4	4	4
	5	2011-01-01	4.7	113.67	5	0.1	37.77	1	1	2	2	5
	6	2011-01-01	1.8	55.242	2	0.1	15.342	1	1	2	2	6
	7	2011-02-01	57.3	285.78	2	0	71.4	2	3	5	5	7
	8	2011-02-01	54.64	290.666	2	0.15	3.4196	3	2	6	6	8
	9	2011-02-01	53.08	206.4	1	0	92.88	2	3	7	7	9
	10	2011-02-01	44.36	162.72	3	0	68.31	2	3	8	8	10

Slika 10. Prikaz relacijske tablice "Narudžba"

4. Skladište podataka

4.1. Uvod u skladišta podataka

Zbog današnjih uvjeta poslovanja potrebno je zadovoljiti sve faktore koji utječu na uspješnost tvrtke. Neki od tih faktora su stavljanje naglaska na kupce i njihove potrebe, povišenje kvalitete usluga i proizvoda, reduciranje vremena isporuke, povećanje profita same tvrtke i smanjenje poslovnih troškova. Učinkovito donošenje odluka i upravljanje tvrtke je otežano zbog neočekivanih i brzih promjena koje se događaju unutar tvrtke i njene okoline (Ćurko, 2001.).

Najvažniji faktor kod ostvarenja konkurentne prednosti je pravovremeno stjecanje točnih i kvalitetnih informacija. Menadžer tvrtke mora imati pristup informacijama čim ih zatraži u pravilnom formatu. Moderna informacijska tehnologija pomoću skladišta podataka osigurava stvaranje modernog sustava za potporu odlučivanju te ima utjecaj na sam razvoj i unaprjeđenje informacijskog sustava tvrtke koji osigurava pravovremene i kvalitetne informacije koje pomažu kod upravljanja poslovanjem. Sustavu potpore odlučivanju je glavna svrha osigurati kvalitetne i točne informacije koje menadžeru pomažu kod donošenja što točnijih odluka. Informacije koje pruža sustav potpore odlučivanju moraju biti točni i u odgovarajućem formatu kako bi se budući događaji mogli prognozirati (Inmon, 2005.).

4.2. Osnovni pojmovi skladišta podataka

Skladište podataka spada pod novu generaciju softverskih sustava za potporu odlučivanja. U najjednostavnijem smislu, ideja iza skladišta podataka je da je potrebno podatke iz baze podataka prikupiti i skladištiti u skladišta podataka kako bi se vršile analize za rudarenje podataka i pronašli smisleni obrasci koji pomažu kod donošenja odluka. Skladišta podataka prikupljaju podatke koji se odnose na subjektna područja tvrtke bez kojih se ne mogu donositi smislene odluke u dotičnim subjektnim područjima. Sadrže podatke iz ranijih aplikacija koje su logički integrirane u tim aplikacijama te se podaci ne mogu mijenjati tijekom obrade. Skladišta podataka sadrže podatke koji služe za opisivanje događaja u dužim vremenskim intervalima kako bi se moglo uspoređivati i predviđati događaje, stoga, skladišta podataka su vremenski usmjerena (Kimball & Ross, 2013.). Tablica 2. prikazuje razlike skladišta podataka i Data Mart-a.

Skladište podataka je subjektno orijentiran, integriran, postojan i vremenski različit skup podataka koji služi kao potpora odlučivanju (Inmon, 2005.).

- Subjektno orijentiran – podaci su organizirani po poslovnim temama

- Integriran – podaci se prikupljaju iz više različitih izvora
- Postojan – podaci se ne mijenjaju u skladištu podataka, samo se dodaju novi
- Vremenski različit – skladište sadrži vremensku dimenziju koja omogućuje pregled podataka kroz vrijeme

Neki od osnovnih zahtjeva za uspostavu skladišta podataka opisuju svrhu i ciljeve skladišta podataka su (Kimball & Ross, 2013.):

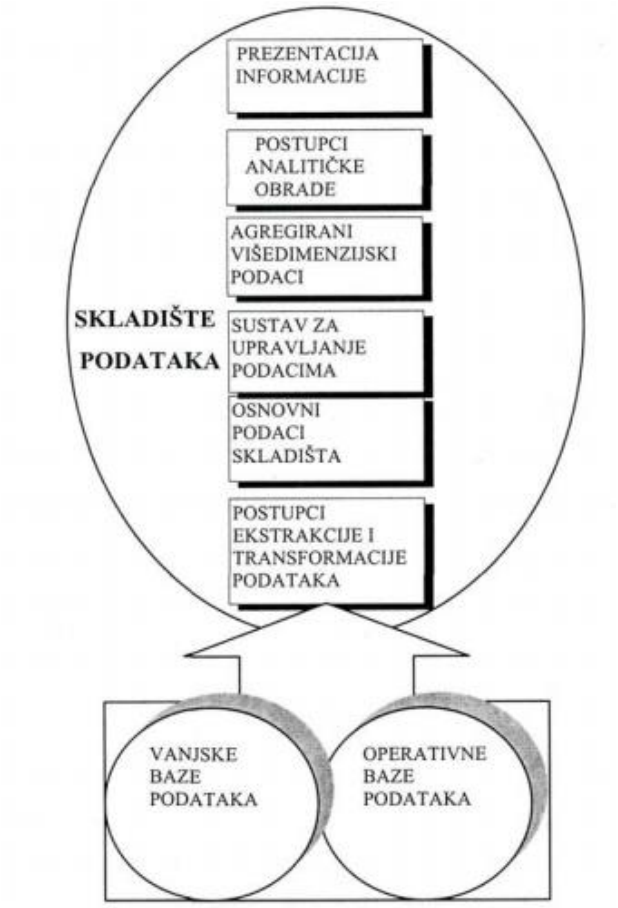
- Skladišta podataka moraju osigurati pristup podacima tvrtki. Menadžer tvrtke pomoću svojeg računala treba moći pristupiti skladištu podataka na brz, jednostavan i efikasan način
- Podaci unutar skladišta podataka moraju biti konzistentni, što znači da kada više korisnika postavljaju identične upite sa različitih lokacija, ti rezultati moraju biti identični
- Podaci unutar skladišta podataka se moraju moći međusobno kombinirati kako bi se dobili svi pokazatelji i mjere poslovanja u tvrtki
- Skladište podataka mora moći dati odgovor na upite, analizirati i prikazivati informacije
- Skladište podataka mora služiti kao mjesto za objavljivanje podataka. Podaci su u skladištu pažljivo sakupljeni iz različitih izvora, transformirani i očišćeni te su samo tako kvalitetni dostupni korisnicima
- Kvaliteta podataka unutar skladišta ukazuje je li potrebno redizajnirati sustav poslovanja. Loši ulazni podaci ne mogu nikada proizvesti kvalitetne i točne izlazne podatke. U slučaju postojanja loših podataka u skladištu, potrebno je redizajnirati sustav jer se samo tako mogu ispraviti loši podaci

Data mart	Skladište podataka
Često sadrži samo jedno predmetno područje	Sadrži više predmetnih područja
Sadrži sažetije podatke	Sadrži vrlo detaljne podatke
Koncentrira se na integriranje svih informacija iz određenog područja	Koncentrira se na integriranje svih izvora podataka
Fokusira se na izgradnju dimenzijskog modela pomoću zvjezdaste sheme	Ne koristi nužno dimenzijske modele

Tablica 2. Prikaz razlika Data Mart-a i skladišta podataka (Mekterović & Brkić, 2017.)

4.3. Struktura skladišta podataka

Podaci i alati za manipulaciju podataka su dva osnovna djela koja tvore skladište podataka (slika 11.). Dio skladišta podataka s podacima tvore agregirani višedimenzionalni podaci i osnovni podaci, a pod mehanizme manipulacije spadaju procesi ekstrakcije i transformacija, sustav za upravljanje podacima, prezentacija i procesi analize podataka (Ćurko, 2001.).



Slika 11. Prikaz strukture skladišta podataka (Ćurko, 2001.)

Vanjski i unutarnji izvori podataka služe kako bi se napunilo skladište podataka. Količina vanjskih podataka je veća kada je nivo odlučivanja viši. Osnovni podaci skladišta podataka se dobivaju procesima transformacije i ekstrakcije koji služe kao veza skladišta podataka sa njenim okruženjem. Sustav za upravljanje podataka osigurava višedimenzionalne agregatne podatke (Ćurko, 2001.). Procesi analize podataka osiguravaju obrade kao što su izdvajanje, selekcija i spajanje dimenzija, grafičko prikazivanje informacije, prognoziranje, itd. Proces analize podataka pronalazi različite obrasce koji pomažu kod donošenja odluka. Najviši nivo skladišta je prezentacija samih informacija. To je korisničko sučelje koje određuje način na koji se upiti postavljaju i format u kojem se prikazuju rezultati tog upita.

S obzirom da je skladište podataka namijenjeno menadžerima tvrtke, važno je da korisničko sučelje osigurava jednostavan i optimiziran rad prilikom postavljanja upita i smisleno i intuitivno prikazivanje rezultata tih upita (Mekterović & Brkić, 2017.).

4.4. Pristupi implementacije skladišta podataka

Prilikom dizajna skladišta podataka važno je uzeti u obzir kakvo skladište podataka razvijamo, za koga ga razvijamo te koliko vremena i budžeta imamo na raspolaganju. Postoje dva pristupa implementacija skladišta podataka, a to su:

- **Top-Down pristup** – najprije se razvija potpuno skladište podataka, a potom se stvaraju područna skladišta podataka. Zagovaratelj ovakvog pristupa je Bill Inmon
- **Bottom-Up pristup** – najprije se razvijaju područna skladišta podataka, a potom se spajaju u veliko skladište podataka ukupne organizacije. Zagovaratelj ovakvog pristupa je Ralph Kimball

Svaki pristup ima svojih prednosti i nedostataka. Kod Top-Down pristupa, inicijalni troškovi su visoki, potrebno je duže vrijeme za početak, olakšano je održavanje te je sama izrada skladišta podatka vremenski iscrpna. Za razliku od Top-Down pristupa, Bottom-Up pristup ima niske inicijalne troškove, potrebno je kraće vrijeme za početak, izrada skladišta podataka nije toliko vremenski iscrpna, ali je održavanje otežano. Izbor optimalnog pristupa ovisi od slučaja.

4.5. Uvod u dimenzijski model podataka

Dimenzijski model podataka specijalizirana je pretvorba relacijskog modela koji se koriti za prikaz podataka u skladištima podataka kako bi se podaci mogli jednostavno agregirati uporabom OLAP alata. U dimenzijskom modelu podataka baza podataka sastoji se od tablice činjenica koja je opisana uporabom mjera i dimenzija. Dimenzija predstavlja kontekst tablice činjenice, te se koristi kod upita di je potrebno grupirati bliske činjenice. Dimenzije su često hijerarhijske te nastoje biti zasebne. Mjera predstavlja količinu koja služi za opisivanje činjenice, kao što je primjerice, profit ili količina prodanih proizvoda (Inmon, 2005.).

Tablica činjenica predstavlja središnju točku interesa kod procesa donošenja poslovnih odluka. To je polazište kod procesa poslovne analize. Mjere su atributi s kontinuiranim (neprekidnim) skupom vrijednosti koji opisuju činjenicu. Budući da skladište podataka sadrži ogroman broj zapisa, upravo će se zbrajanjem numeričkih podataka dobiti vrijednosti zanimljive za analizu (Kimball & Ross, 2013.).

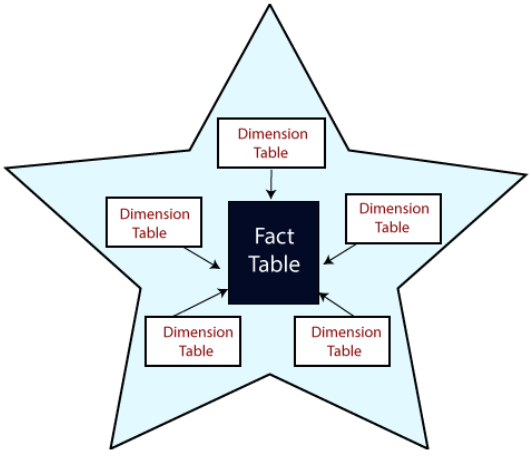
Dimenzijske tablice čine skup međusobno nezavisnih parametara koje opisuju činjenicu. Svaka dimenzija sastoji se od jednog ili više atributa. Najbolji opisni atributi su tekstualni. Dimenzije određuju razinu zrnatosti u skladištu podataka (Kimball & Ross, 2013.).

Za razliku od relacijskog modela podataka, dimenzijski model podataka se oblikuje u denormaliziranom obliku. Postoje dva pristupa oblikovanja dimenzijskog modela podataka, a to su zvjezdasta shema (slika 13.) i pahuljasta shema. Slika 12. prikazuje strukturu dimenzijske tablice i tablice činjenica.

Tablica činjenica	Dimenzijska tablica
SK	SK
SK_Kupac	ID
SK_Proizvod	Grad
SK_Lokacija	Zemlja
SK_Datum	Država
Profit	Regija
Količina	Market

Slika 12. Prikaz strukture dimenzijskih tablica i tablice činjenica

Zvjezdasta shema dobiva svoje ime po svojem obliku koji liči na zvijezdu. Središnja komponenta takve sheme je tablica činjenica, dok vrhovi koji izlaze iz nje predstavljaju dimenzijske tablice. Dimenzijske tablice su najčešće denormalizirane. Zvjezdasta shema je u današnje vrijeme najprihvaćeniji oblik dimenzijskog modela, unatoč njenoj jednostavnoj strukturi (Kimball & Ross, 2013.).



Slika 13. Prikaz zvjezdaste sheme (Anon., n.d.)

Pahuljasta shema predstavlja detaljniji prikaz zvjezdaste sheme. Dimenzijske tablice su povezano sa drugim dimenzijskim tablicama, odnosno, normalizirane su za razliku od zvjezdaste sheme. Složen oblik pahuljastog izgleda pojavljuje se kada su dimenzije pahuljaste sheme razrađene na više razina odnosa (Kimball & Ross, 2013.).

4.6. Koraci u procesu dizajna skladišta podataka

Potrebno je definirati osnovne korake pri izradi skladišta podataka kako bi se izgradilo što kvalitetnije skladište podataka. Osnovni koraci koje treba proći prilikom dizajna skladišta jesu:

- Odabrati poslovni proces – obuhvaća dnevne aktivnosti koje se odvijaju u tvrtki i koje su podržane transakcijskim sustavima. Potrebno je definirati potrebe korisnika skladišta podataka što zahtijeva kvalitetne podatke
- Definirati granularnost – opisuje razinu detalja te je potrebno identificirati najnižu razinu informacija za svaku tablicu u modelu
- Definirati dimenzije – pružaju kontekst poslovnom procesu, odnosno, opisuju promatranu mjeru. Najčešće daju odgovore na pitanja kao što su tko?, što?, gdje?, kada?
- Definirati mjere – mjere ili činjenice poslovnog procesa predstavljaju predmet interesa procesa kod donošenja odluka. Mjere se pohranjuju u tablicu činjenica

4.7. Napredna svojstva dimenzijskih tablica i surogat ključevi

Jedno od osnovnih načela dimenzijskog modeliranja jest da uvodimo surogat ključeve u sve tablice dimenzijskog modela. Surogat ključevi su predstavljeni cijelim brojevima. Originalni primarni ključevi se zadržavaju te se koriste kao veza na izvorne podatke. Surogat ključevi ne smiju biti povezani sa izvornim primarnim ključevima te se koriste kao veza prema tablici činjenica (Mekterović & Brkić, 2017.).

Uz uvođenje surogat ključeva, potrebno je identificirati i definirati dodatna svojstva dimenzijskih tablica. Neka od najkorištenijih dodatnih svojstva jesu:

- **Usklađene dimenzije** se javljaju ukoliko su neke dimenzijske tablice iste za više od jedne tablice činjenice. Takve dimenzijske tablice, koje su identične ili su podskup jedna druge, a pojavljuju se u više zvjezdastih shema se nazivaju usklađene dimenzije (Mekterović & Brkić, 2017.)
- **Sporo mijenjajuće dimenzije** su dimenzijske tablice čiji se atributi mijenjaju sporo odnosno niskom frekvencijom. Kada se javi potreba praćenja promjena u dimenzijskog tablici potrebno je za svaki atributi specificirati strategiju

upravljanja promjenama (Kimball & Ross, 2013.). Ralph Kimball je izvorno predložio tri strategije (Kimball, et al., 2008.):

- **Tip 1** – ovom se strategijom prepisuje stara vrijednost atributa novom. Najčešće se koristi za ispravljanje pogrešaka u podacima te za održavanje aktualne vrijednosti atributa uz uvjet da nije potrebno pamtiti prethodne vrijednosti (ispravne ili pogrešne). Očiti nedostatak pristupa je nemogućnost rekonstruiranja povijesnih podataka jer se prati samo najnovija trenutna vrijednosti atributa
- **Tip 2** – ovom se strategijom povijesni podaci prate dodavanjem nove n-torke u dimenzijsku tablicu svaki put kada se promijeni vrijednosti bilo kojeg atributa. Ova strategija podrazumijeva da je u shemi dimenzijske tablice uključen primarni ključ izvorne relacije kako bi se omogućilo direktno povezivanje n-torke u dimenzijskom s izvornom n-torkom u relacijskom modelu. Prednosti ovakvog pristupa je što se dodaju novi atributi za praćenje vremenskog intervala aktivnosti zapisa te se također može i dodati indikator "aktivan"
- **Tip 3** – ovom se strategijom prati povijest podataka korištenjem zasebnih atributa, jednih za pohranu inicijalne vrijednosti i drugih za pohranu aktualne vrijednosti. Ovakva strategija se koristi u situacijama u kojima je potrebno činjenične podatke promatrati u kontekstu starih vrijednosti dimenzijskih podataka i obratno

- **Degenerirane dimenzije** su atributi u tablici činjenica koja imaju svojstva dimenzijskog ključa ali ne postoji odgovarajuća dimenzijska tablica koju bi referencirao. Mogu biti korisne za grupiranje zapisa, kao i za povratnu veza na izvorišne podatke (Mekterović & Brkić, 2017.)
- **Mini-dimenzije** – kod dimenzija s velikim brojem atributa određene (najčešće logički povezane) skupine atributa se izdvajaju u posebnu dimenziju koju nazivamo mini-dimenzija. Mini-dimenzije se direktno spajaju s tablicom činjenica, odnosno ključ mini-dimenzije se dodaje u tablicu činjenica (Mekterović & Brkić, 2017.)
- **Kompozitne dimenzije** su dimenzije koje nastaju kada se atributi niske kardinalnost, zastavice, indikatori i sl. iz tablice činjenica grupiraju i izmjeste iz tablice činjenica u posebnu dimenziju (Kimball, et al., 2015.)

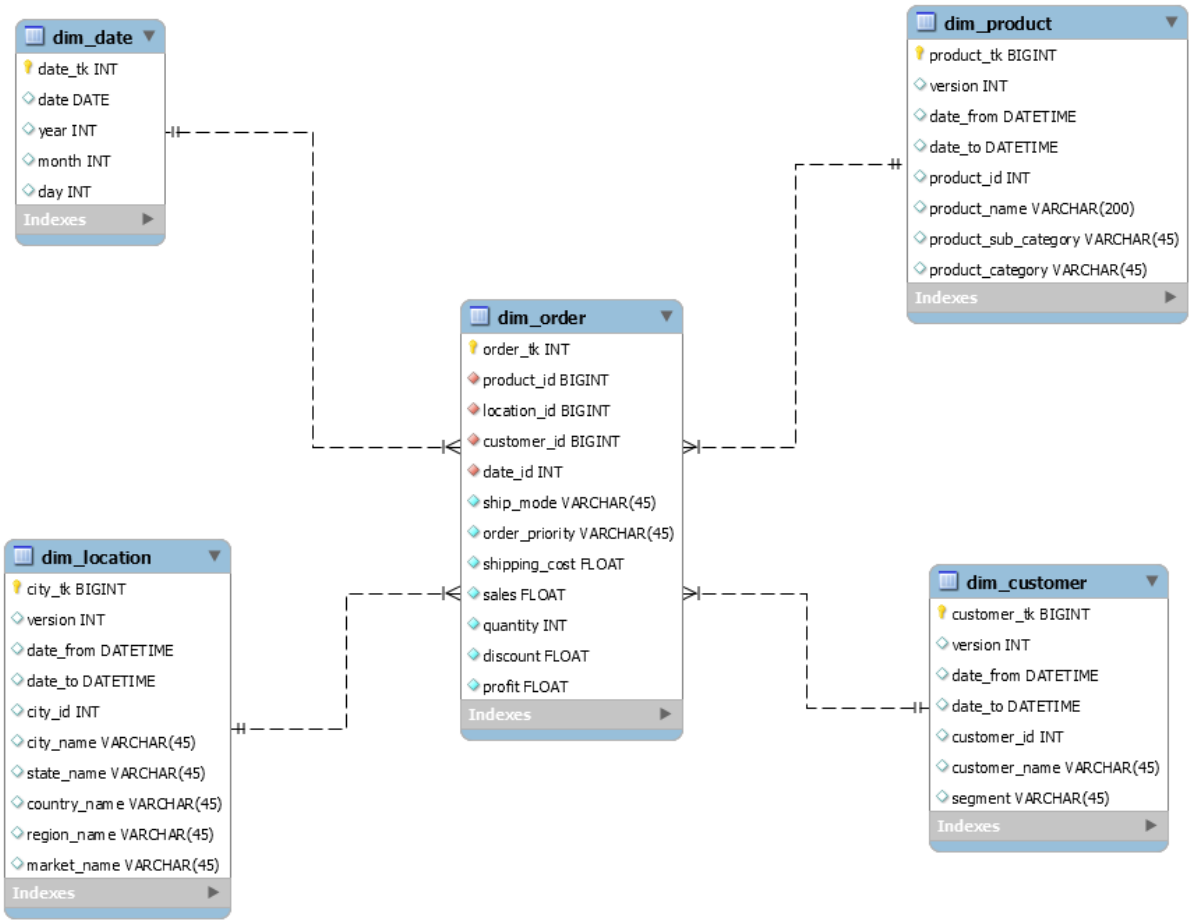
- **Heterogene dimenzije** se javljaju kad neka poslovanja prate prodaju proizvoda koji nemaju mnogo toga zajedničko. Takvi proizvodi nazivaju se heterogeni proizvodi. Potrebno je napraviti jednu temeljnu činjeničnu i dimenzijsku tablicu koja sadrži samo zajedničke atribute i niz različitih činjeničnih i dimenzijskih tablica koje sadrže detaljne opise različitih proizvoda. Time se ostvaruje detaljno pregledavanje podataka po dodatnim atributima svakog proizvoda (Kimball & Ross, 2013.)

4.8. Izrada dimenzijskog modela podataka

Sljedeći korak pri izradi skladišta podataka je izrada dimenzijskog modela podataka. Dimenzijski model podataka je izrađen na temelju izrađenog Entity/Relationship modela. Dimenzijski model podataka je temeljen na zvjezdastoj shemi, iako je bio dobar kandidat za korištenje pahuljaste sheme. Razlog odabira zvjezdaste sheme je to što se postigao isti rezultat korištenjem jednostavnije strukture. Konačni izrađeni dimenzijski model se sastoji od jedne tablice činjenica i četiri dimenzijske tablice. Središnja tablica u modelu, odnosno tablica činjenica povezana je sa svim ostalim dimenzijskim tablicama surogat ključevima. Tablica činjenica je "Narudžba", a dimenzijske tablice su "Proizvod", "Lokacija", "Kupac" i "Vrijeme prodaje" gdje se unutar tablice činjenica osim mjera i surogat ključeva kao veza na dimenzijske tablice nalaze i dvije degenerirane dimenzije "Način otpreme" i "Prioritet narudžbe" (slika 14.). Također, bilo je potrebno stvoriti vremensku dimenziju što je postignuto izdvajanjem datuma prodaje iz tablice činjenica i premještanjem ju u zasebnu dimenziju. Mjere koje se promatraju u tablici činjenica su ostale iste, te su dimenzijske tablice stvorene denormalizacijom izvornog Entity/Relationship modela. Struktura dimenzijskih tablica je sljedeća:

- Grad **–>** Država **–>** Zemlja **–>** Regija **–>** Market
- Proizvod **–>** Pod-kategorija **–>** Kategorija
- Kupac **–>** Segment

Dodatna svojstva dimenzije koja su korištena u završnom radu su sporo mijenjajuće dimenzije tipa 2, kako bi se omogućilo praćenje povijesnih podataka.

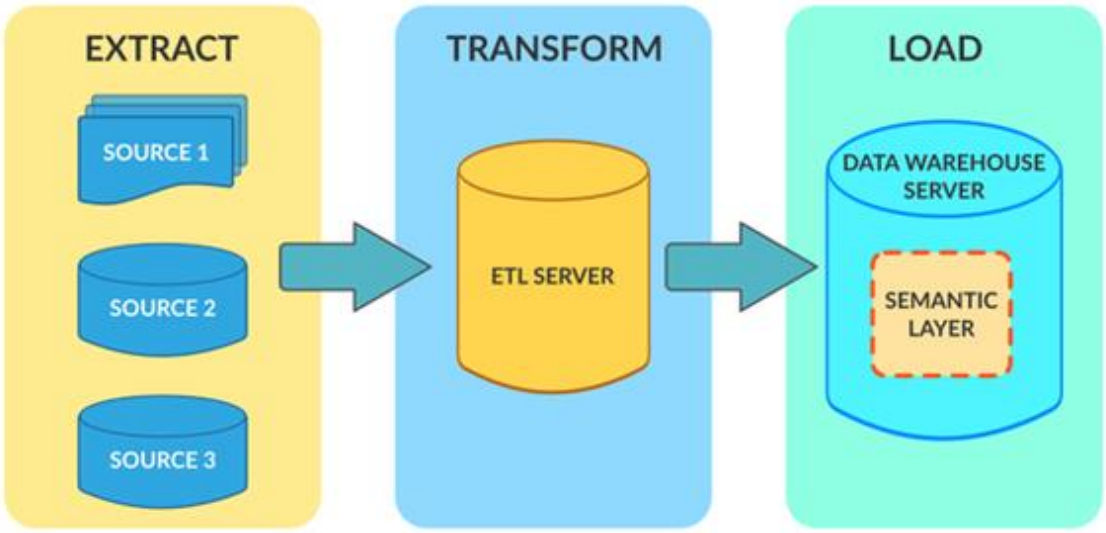


Slika 14. Prikaz dimenzijskog modela podataka

5. ETL proces

5.1. Uvod u ETL proces

Sljedeći korak u izradi skladišta podataka je punjenje dimenzijskog modela sa podacima. Proces kojim se to postiže naziva se ETL proces (engl. Extract-Transform-Load) (slika 15.). ETL je složen proces s mnogo parametara, a osnovne operacije u procesu punjenja su izdvajanje, transformiranje i učitavanje podataka (Kimball, et al., 2015.).



Slika 15. Prikaz ETL procesa (Shimko, n.d.)

Prvi korak ETL je izdvajanje podataka iz izvornog sustava u pripremno područje. U mnogim slučajevima to predstavlja najvažniji aspekt ETL procesa, jer ispravo vađenje podataka postavlja osnovu za uspjeh sljedećih procesa (Kimball, et al., 2015.). Postoje dva načina izdvajanja podataka, a to su sljedeća:

- **Potpuno izdvajanje** – neki sustavi ne mogu prepoznati koji podaci su promijenjeni, pa je ponovno učitavanje svih podataka jedini način da se podaci izdvoje iz sustava. Kako ovaj pristup uključuje veliku količinu prijenosa podataka, preporučuje ga se koristiti u krajnjem slučaju i to samo za male tablice
- **Parcijalno izdvajanje** – neki sustavi nas ne mogu obavijestiti da je došlo do ažuriranja, ali mogu prepoznati koji zapisi su izmijenjeni i te podatke izdvojiti. Jedan od nedostataka parcijalnog, odnosno inkrementalnog izdvajanja je taj što možda neće biti moguće otkriti izbrisane zapise u izvornim podacima

Kako bismo mogli prepoznati takve promjene podataka na izvorišnom sustavu, moramo koristiti inovativni mehanizam koji se naziva Change Data Capture (CDC). Najpoznatije CDC tehnike jesu:

- **Dodavanje vremenske oznake** – dimenzijske tablice čije promjene moraju biti zabilježene mogu imati atribut koji predstavlja vrijeme zadnje promjene. Imena kao što su "LAST_UPDATE" su česta. Svaki se redak u bilo kojoj dimenzijskoj tablici koji ima vremensku oznaku u tom stupcu čija je vrijednost novija u odnosu na posljednji put kada su podaci bili snimljeni, smatra se da se promijenio (Kimball, et al., 2015.)
- **Razlika snimki stanja** – uspoređuje se trenutno stanje podataka s prethodnim stanjem podataka kako bi se identificirale promjene. Izazovi ovog pristupa uključuju:
    - Da bi se izvršile razlike stanja potrebno je mnogo resursa za izračunavanje razlika između podataka, a potrošnja resursa raste barem linearno s rastom volumena podataka
    - CDC se ne može izvoditi u stvarnom vremenu jer tehnika zahtijeva previše resursa da bi se izvodila cijelo vrijeme
- **Aplikacijsko bilježenje promjena** – promjene podataka se bilježe na posebno mjesto previđeno za tu namjenu. Kako bi sustav funkcionirao, svi aplikacijski programi moraju identično evidentirati promjene. Ovakav pristup je pogodan za korištenje kada se koriste baze podataka koje ne podržavaju mehanizme za evidentiranje promjena (Kimball, et al., 2015.)
- **Korištenjem okidača (engl. Trigger) baze podataka** – okidači baza podataka mogu se koristiti za izvođenje CDC-a u "Shadow" tablicama. Takve tablice mogu pohraniti cijeli red radi praćenja svake promjene svakog stupca ili se pohranjuje samo primarni ključ kao i tip operacije (umetanje, ažuriranje, brisanje) (Kimball, et al., 2015.)
- **Korištenje dnevnika transakcija** – transakcijske baze podataka pohranjuju sve promjene u dnevniku transakcija kako bi se obnovilo počinjeno stanje baze podataka u slučaju da se baza podataka sruši iz bilo kojeg razloga. CDC temeljen na dnevniku transakcija koristi ovaj aspekt transakcije baze podataka za čitanje promjena iz zapisnika (Kimball, et al., 2015.)

Bez obzira na korištenu metodu, izdvajanje podataka ne bi trebalo utjecati na performanse i vrijeme odaziva izvornih sustava. Izvorni sustavi su najčešće proizvodne baze podataka. Svako usporavanje ili blokiranje može utjecati na poslovanje tvrtke.

Podaci izdvojeni iz izvornog sustava su "sirovi" i nisu iskoristivi u izvornog obliku.

Stoga, potrebno ih je očistiti, preslikati i transformirati. U stvari, to je ključni korak u kojem ETL proces dodaje vrijednost i mijenja podatke tako da se mogu stvoriti korisni izvještaji. U ovom se koraku na izdvojene podatke primjenjuje skup pravila ili funkcija kako bi se pretvorili u jedinstveni standardni format (Kimball & Ross, 2013.) . Može uključivati:

- Filtriranje – učitavanje samo određenih atributa u skladište podataka
- Čišćenje – popunjavanje NULL vrijednosti s nekim zadanim vrijednostima
- Spajanje – spajanje više atributa u jedan
- Razdvajanje – razdvajanje jednog atributa na više atributa
- Sortiranje – sortiranje redova na temelju nekog atributa

Provjera kvalitete podataka se obavlja pomoću testova kvalitete (engl. Quality Screens). Testovi kvalitete se aktiviraju pojavom greške te bilježe istu u shemu grešaka (engl. Error Event Schema). Shema grešaka evidentira sve greške koje su se javile tijekom ETL procesa. Zadnji korak transformacije je generiranje surogat ključeva i uklanjanje duplih zapisa što će transformirati podatke u odgovarajući format prikladan za dimenzijski model podataka (Kimball, et al., 2015.).

Treći i posljednji korak ETL procesa je učitavanje podataka. U ovom koraku transformirani podaci konačno se učitavaju u skladište podataka. Ovisno o zahtjevima organizacije, ovaj postupak uvelike varira. U tipičnom skladištu podataka, potrebno je učitati ogromnu količinu podataka u relativno kratkom vremenu. Dakle, proces učitavanja treba biti optimiziran za izvođenje. Nakon uspješnog učitavanja podataka u skladište, slijedi izrada analitičkih izvještaja nad tim podacima.

5.2. Primjena ETL procesa nad dimenzijskim tablicama

Sljedeći korak je punjenje skladišta podataka sa podacima iz transakcijskog sustava. Softver korišten za integraciju podataka je Pentaho. Inicijalni korak u punjenju skladišta podataka je stvaranje konekcije na bazu podataka. Zatim je bilo potrebno stvoriti model koji će dohvaćati podatke, transformirati ih i učitati u skladište podataka. Izvori podataka su baza podataka i CSV datoteka, koji su raspodijeljeni tako da baza podataka ima 80% podataka, a CSV datoteka preostalih 20%.

5.2.1. Učitavanje podataka u dimenzijske tablice

Naredba korištena za dohvaćanje podataka iz baze je “Table Input“. Unutar “Table Input“ naredbe je bilo potrebno odabrati željene atribute iz dotične relacije te ju povezati sa odgovarajućim roditeljskim tablicama po potrebi.

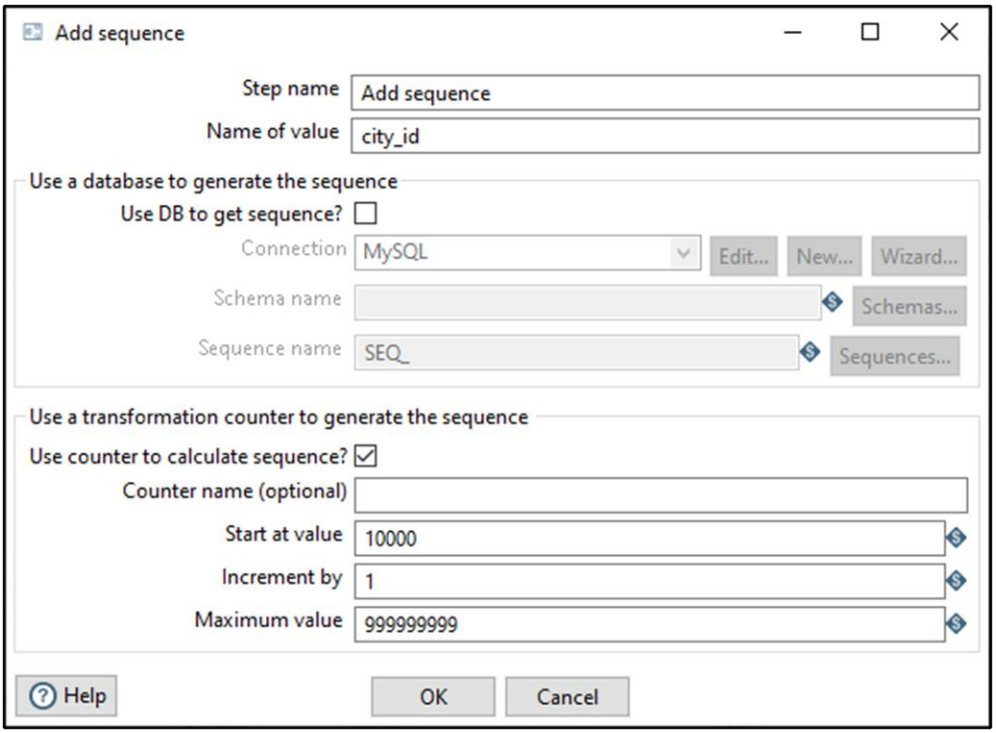
Zatim slijedi “Sort Rows“ naredba koja povezanu relaciju sortira po određenom

atributu. Preostali podaci iz CSV datoteke dohvaćeni su naredbom "CSV File Input" koju također sortiramo po određenom atributu. Kako bi se riješili duplikata iz CSV datoteke bilo je potrebno koristiti naredbu "Unique Rows" (slike 18., 24. i 31.) koja prima željeni atribut kao argument po kojem će očistiti podatke. Naredbom "Add Sequence" (slike 16., 22. i 29.) smo simulirali dodavanje identifikacijskog ključa podacima iz CSV datoteke započevši od broja 10000. Razlog tome je mogućnost razlikovanja izvora podataka, odnosno, podatke dohvaćene iz CSV datoteke moći ćemo prepoznat po identifikacijskom ključu koji je veći ili jednak 10000.

Sljedeći korak je bio korištenje naredbe "Select Values" kako bi odabrali samo one atribute koji su nam od interesa te po potrebi promijenili ostale atribute podataka kako bi ih doveli u željeni oblik. Redoslijed kojim su podaci odabrani je bitan, odnosno mora biti jednak redoslijedu kojim su dohvaćeni podaci iz baze podataka. Podatke dohvaćene iz navedenih izvora spojili smo korištenjem naredbe "Sorted Merge" (slike 17., 23. i 30.). Spojene podatke je bilo potrebno ponovno sortirati te ukloniti duplikate. Proces sortiranja i uklanjanja duplikata je identičan kao i u prijašnjim koracima.

Posljednji korak je učitavanje podataka u dimenzijsku tablicu. Taj proces se može jednostavno izvršiti uz pomoć naredbe "Dimension Lookup/Update". Unutar naredbe "Dimension Lookup/Update" bilo je potrebno odabrati konekciju na bazu podataka, shemu u kojoj se nalazi željena dimenzijska tablica te ciljanu dimenzijsku tablicu (slike 19., 25. i 32.). Nakon odabira željenih atributa i ključeva koje dimenzijska tablica očekuje, pokrenuli smo transformaciju i uspješno učitali podatke u dimenzijsku tablicu (slike 21., 27., 28. i 34.). Transformacijski model je identičan kod svih dimenzijskih tablica (slike 20., 26. i 33.). U nastavku poglavlja je opisani proces dokumentiran slikama implementacije iz Pentaho alata.

5.2.1.1. Transformacija dimenzijske tablice "Lokacija"



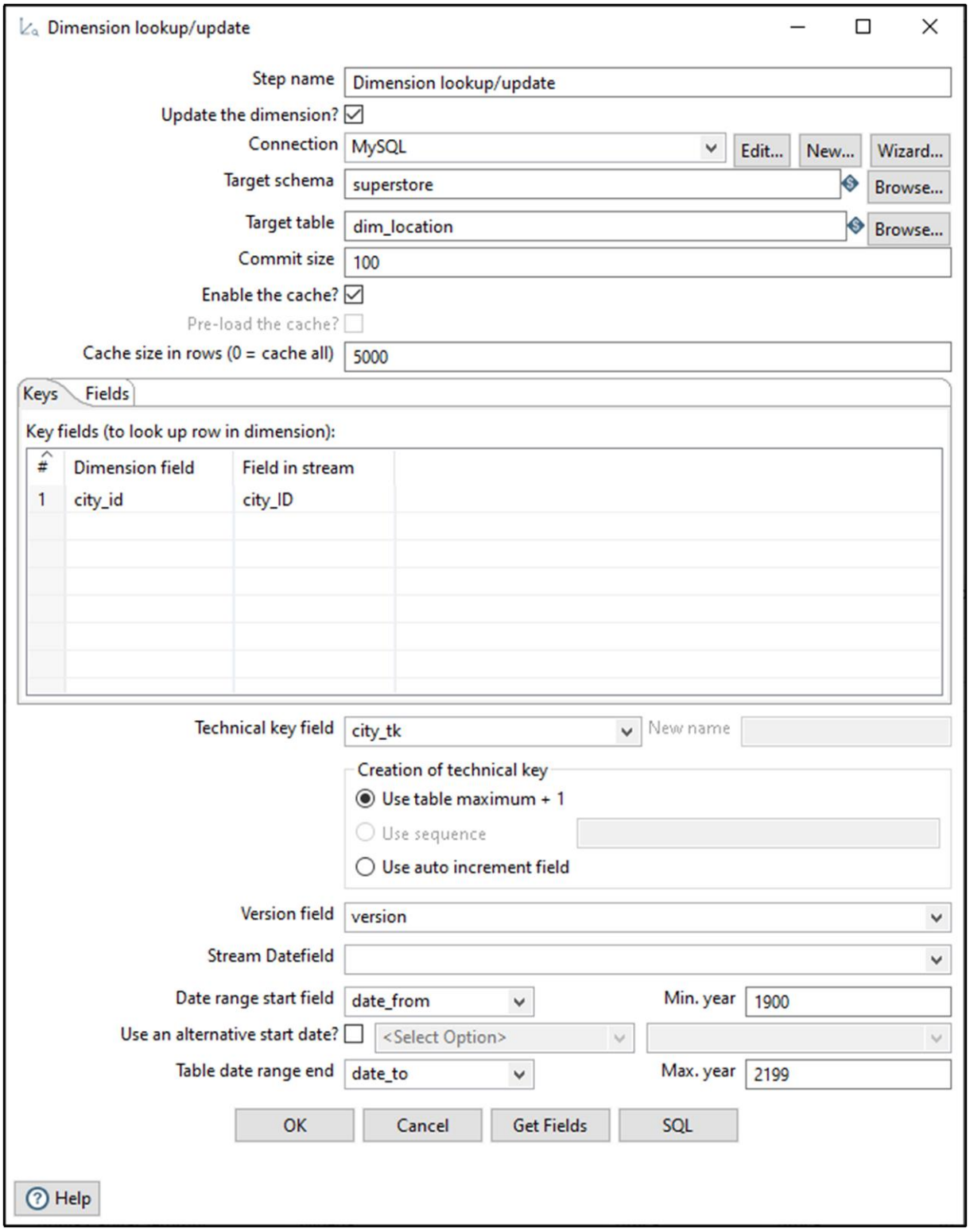
Slika 16. Prikaz naredbe "Add Sequence"

#	city_ID	city	state	country	region	market
1	1290	Aachen	North Rhine-Westphalia	Germany	Central	LATAM
2	10000	Aachen	North Rhine-Westphalia	Germany	Central	EU
3	2709	Aalen	Baden-Wurttemberg	Germany	Central	LATAM
4	10001	Aalst	East Flanders	Belgium	Central	EU
5	1183	Aba	Abia	Nigeria	Africa	Africa
6	10002	Aba	Abia	Nigeria	Africa	Africa
7	433	Abadan	Khuzestan	Iran	EMEA	EMEA
8	10003	Abadan	Khuzestan	Iran	EMEA	EMEA
9	10004	Abakaliki	Ebonyi	Nigeria	Africa	Africa
10	3159	Abbeville	Picardy	France	Central	LATAM

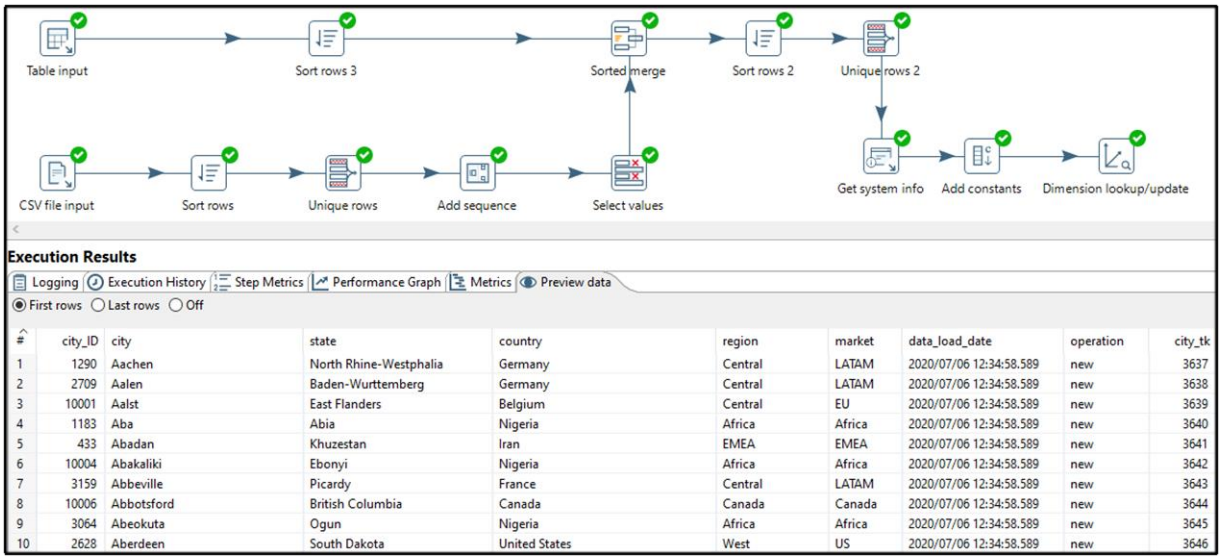
Slika 17. Prikaz rezultata naredbe "Sorted Merge"

#	city_ID	city	state	country	region	market
1	1290	Aachen	North Rhine-Westphalia	Germany	Central	LATAM
2	2709	Aalen	Baden-Wurttemberg	Germany	Central	LATAM
3	10001	Aalst	East Flanders	Belgium	Central	EU
4	1183	Aba	Abia	Nigeria	Africa	Africa
5	433	Abadan	Khuzestan	Iran	EMEA	EMEA
6	10004	Abakaliki	Ebonyi	Nigeria	Africa	Africa
7	3159	Abbeville	Picardy	France	Central	LATAM
8	10006	Abbotsford	British Columbia	Canada	Canada	Canada
9	3064	Abeokuta	Ogun	Nigeria	Africa	Africa
10	2628	Aberdeen	South Dakota	United States	West	US

Slika 18. Prikaz rezultata naredbe "Unique Rows"



Slika 19. Prikaz naredbe “Dimension Lookup/Update“



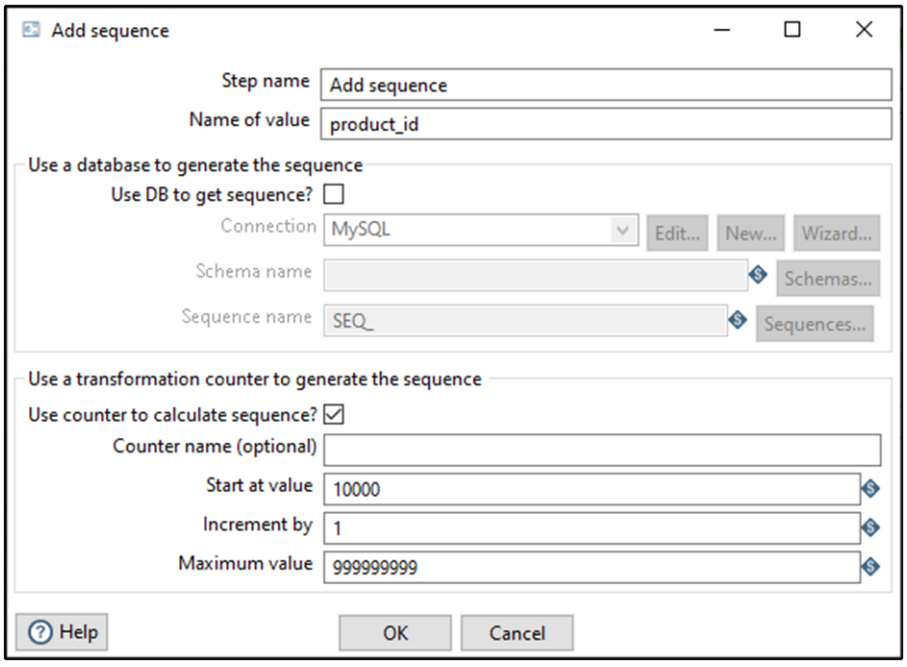
#	city_ID	city	state	country	region	market	data_load_date	operation	city_tk
1	1290	Aachen	North Rhine-Westphalia	Germany	Central	LATAM	2020/07/06 12:34:58.589	new	3637
2	2709	Aalen	Baden-Wurttemberg	Germany	Central	LATAM	2020/07/06 12:34:58.589	new	3638
3	10001	Aalst	East Flanders	Belgium	Central	EU	2020/07/06 12:34:58.589	new	3639
4	1183	Aba	Abia	Nigeria	Africa	Africa	2020/07/06 12:34:58.589	new	3640
5	433	Abadan	Khuzestan	Iran	EMEA	EMEA	2020/07/06 12:34:58.589	new	3641
6	10004	Abakaliki	Ebonyi	Nigeria	Africa	Africa	2020/07/06 12:34:58.589	new	3642
7	3159	Abbeville	Picardy	France	Central	LATAM	2020/07/06 12:34:58.589	new	3643
8	10006	Abbotsford	British Columbia	Canada	Canada	Canada	2020/07/06 12:34:58.589	new	3644
9	3064	Abeokuta	Ogun	Nigeria	Africa	Africa	2020/07/06 12:34:58.589	new	3645
10	2628	Aberdeen	South Dakota	United States	West	US	2020/07/06 12:34:58.589	new	3646

Slika 20. Prikaz transformacijskog modela dimenzijske tablice "Lokacija"

	city_tk	version	data_load_date	operation	date_from	date_to	city_id	city_name	state_name	country_name	region_name	market_name
▶	0	1	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
	1	1	2020-06-02 16:39:23	new	1900-01-01 00:00...	2200-01-01 00:00...	1290	Aachen	North Rhi...	Germany	Central	LATAM
	2	1	2020-06-02 16:39:23	new	1900-01-01 00:00...	2200-01-01 00:00...	2709	Aalen	Baden-W...	Germany	Central	LATAM
	3	1	2020-05-31 17:25:42	new	1900-01-01 00:00...	2200-01-01 00:00...	10001	Aalst	East Flan...	Belgium	Central	EU
	4	1	2020-06-02 16:39:23	new	1900-01-01 00:00...	2200-01-01 00:00...	1183	Aba	Abia	Nigeria	Africa	Africa
	5	1	2020-06-02 16:39:23	new	1900-01-01 00:00...	2200-01-01 00:00...	433	Abadan	Khuzestan	Iran	EMEA	EMEA
	6	1	2020-05-31 17:25:42	new	1900-01-01 00:00...	2200-01-01 00:00...	10004	Abakaliki	Ebonyi	Nigeria	Africa	Africa
	7	1	2020-06-02 16:39:23	new	1900-01-01 00:00...	2200-01-01 00:00...	3159	Abbeville	Picardy	France	Central	LATAM
	8	1	2020-05-31 17:25:42	new	1900-01-01 00:00...	2200-01-01 00:00...	10006	Abbotsford	British Col...	Canada	Canada	Canada
	9	1	2020-06-02 16:39:23	new	1900-01-01 00:00...	2200-01-01 00:00...	3064	Abeokuta	Ogun	Nigeria	Africa	Africa
	10	1	2020-06-02 16:39:23	new	1900-01-01 00:00...	2200-01-01 00:00...	2628	Aberdeen	South Da...	United States	West	US

Slika 21. Prikaz dimenzijske tablice "Lokacija"

5.2.1.2. Transformacija dimenzijske tablice "Proizvod"



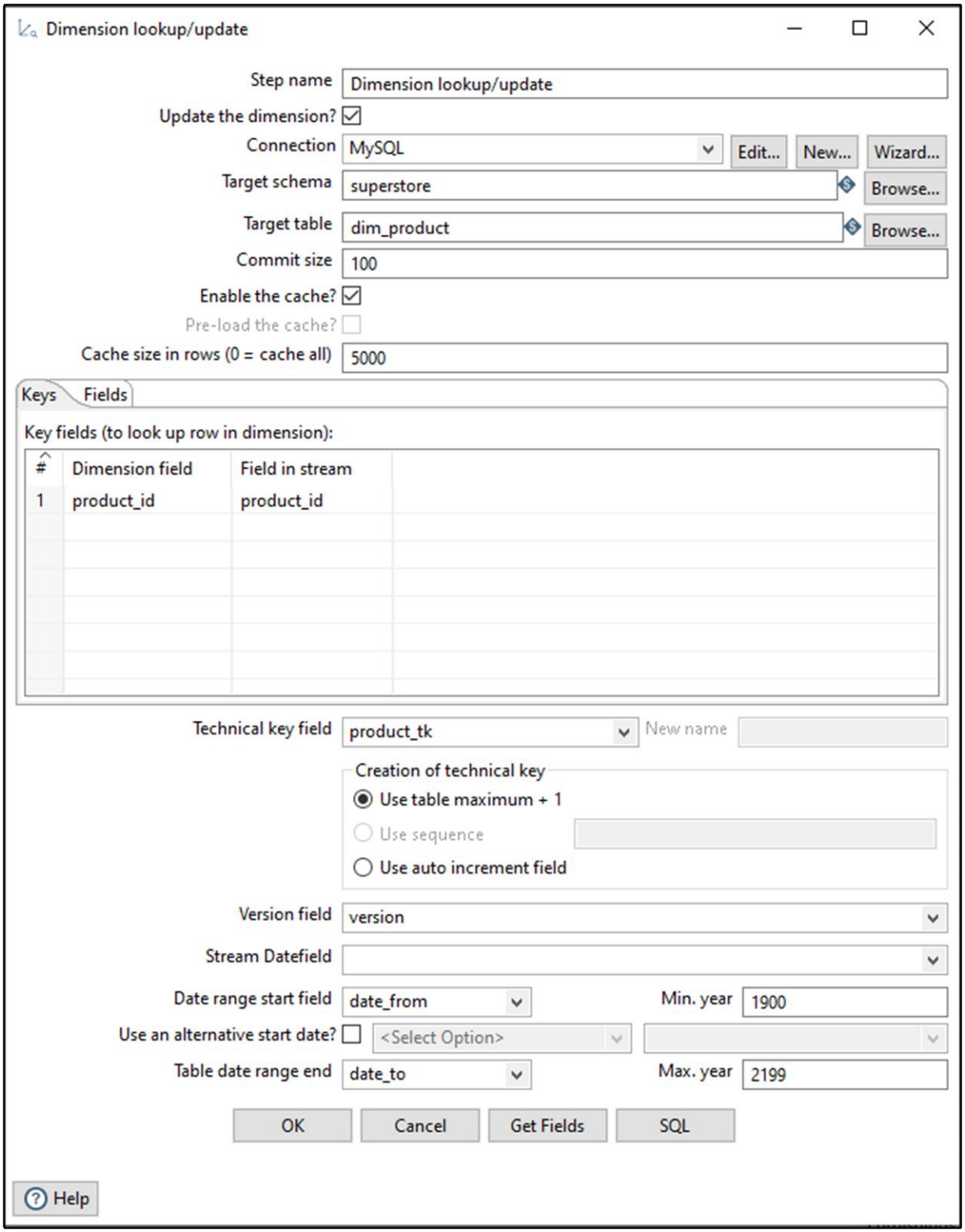
Slika 22. Prikaz naredbe "Add Sequence"

#	product_id	product	sub_category	category
1	3344	"While you Were Out" Message Book, One Form per Page	Paper	Office Supplies
2	10000	"While you Were Out" Message Book, One Form per Page	Paper	Office Supplies
3	2257	#10 Gummed Flap White Envelopes, 100/Box	Envelopes	Office Supplies
4	3130	#10 Self-Seal White Envelopes	Envelopes	Office Supplies
5	3062	#10 White Business Envelopes,4 1/8 x 9 1/2	Envelopes	Office Supplies
6	10001	#10 White Business Envelopes,4 1/8 x 9 1/2	Envelopes	Office Supplies
7	1183	#10- 4 1/8" x 9 1/2" Recycled Envelopes	Envelopes	Office Supplies
8	10002	#10- 4 1/8" x 9 1/2" Recycled Envelopes	Envelopes	Office Supplies
9	1770	#10- 4 1/8" x 9 1/2" Security-Tint Envelopes	Envelopes	Office Supplies
10	3245	#10-4 1/8" x 9 1/2" Premium Diagonal Seam Envelopes	Envelopes	Office Supplies

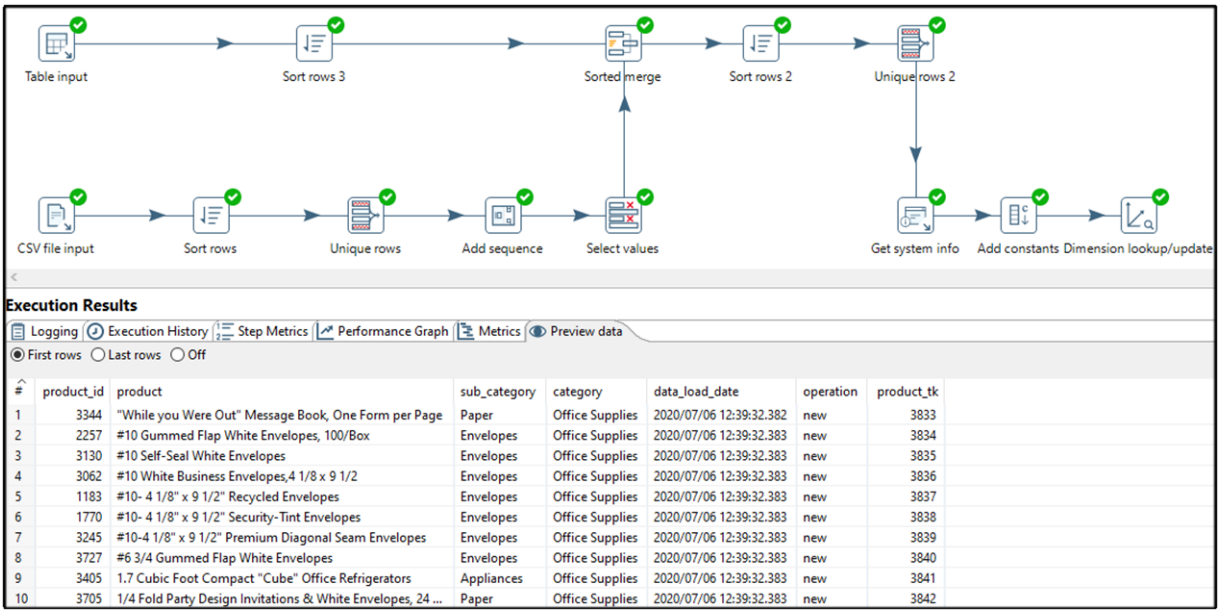
Slika 23. Prikaz rezultata naredbe “Sorted Merge“

#	product_id	product	sub_category	category
1	3344	"While you Were Out" Message Book, One Form per Page	Paper	Office Supplies
2	2257	#10 Gummed Flap White Envelopes, 100/Box	Envelopes	Office Supplies
3	3130	#10 Self-Seal White Envelopes	Envelopes	Office Supplies
4	3062	#10 White Business Envelopes,4 1/8 x 9 1/2	Envelopes	Office Supplies
5	1183	#10- 4 1/8" x 9 1/2" Recycled Envelopes	Envelopes	Office Supplies
6	1770	#10- 4 1/8" x 9 1/2" Security-Tint Envelopes	Envelopes	Office Supplies
7	3245	#10-4 1/8" x 9 1/2" Premium Diagonal Seam Envelopes	Envelopes	Office Supplies
8	3727	#6 3/4 Gummed Flap White Envelopes	Envelopes	Office Supplies
9	3405	1.7 Cubic Foot Compact "Cube" Office Refrigerators	Appliances	Office Supplies
10	3705	1/4 Fold Party Design Invitations & White Envelopes, 24 ...	Paper	Office Supplies

Slika 24. Prikaz rezultata naredbe “Unique Rows“



Slika 25. Prikaz naredbe “Dimension Lookup/Update“



#	product_id	product	sub_category	category	data_load_date	operation	product_tk
1	3344	"While you Were Out" Message Book, One Form per Page	Paper	Office Supplies	2020/07/06 12:39:32.382	new	3833
2	2257	#10 Gummed Flap White Envelopes, 100/Box	Envelopes	Office Supplies	2020/07/06 12:39:32.383	new	3834
3	3130	#10 Self-Seal White Envelopes	Envelopes	Office Supplies	2020/07/06 12:39:32.383	new	3835
4	3062	#10 White Business Envelopes,4 1/8 x 9 1/2	Envelopes	Office Supplies	2020/07/06 12:39:32.383	new	3836
5	1183	#10- 4 1/8" x 9 1/2" Recycled Envelopes	Envelopes	Office Supplies	2020/07/06 12:39:32.383	new	3837
6	1770	#10- 4 1/8" x 9 1/2" Security-Tint Envelopes	Envelopes	Office Supplies	2020/07/06 12:39:32.383	new	3838
7	3245	#10-4 1/8" x 9 1/2" Premium Diagonal Seam Envelopes	Envelopes	Office Supplies	2020/07/06 12:39:32.383	new	3839
8	3727	#6 3/4 Gummed Flap White Envelopes	Envelopes	Office Supplies	2020/07/06 12:39:32.383	new	3840
9	3405	1.7 Cubic Foot Compact "Cube" Office Refrigerators	Appliances	Office Supplies	2020/07/06 12:39:32.383	new	3841
10	3705	1/4 Fold Party Design Invitations & White Envelopes, 24 ...	Paper	Office Supplies	2020/07/06 12:39:32.383	new	3842

Slika 26. Prikaz transformacijskog modela dimenzijske tablice "Proizvod"

product_tk	version	data_load_date	operation	date_from	date_to	product_id	product_name	product_sub_category	product_category
0	1	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
1	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	3344	"While you ...	Paper	Office Supplies
2	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	2257	#10 Gumme...	Envelopes	Office Supplies
3	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	3130	#10 Self-Se...	Envelopes	Office Supplies
4	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	3062	#10 White B...	Envelopes	Office Supplies
5	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	1183	#10- 4 1/8" ...	Envelopes	Office Supplies
6	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	1770	#10- 4 1/8" ...	Envelopes	Office Supplies
7	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	3245	#10-4 1/8" x...	Envelopes	Office Supplies
8	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	3727	#6 3/4 Gum...	Envelopes	Office Supplies
9	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	3405	1.7 Cubic Fo...	Appliances	Office Supplies
10	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	3705	1/4 Fold Part...	Paper	Office Supplies

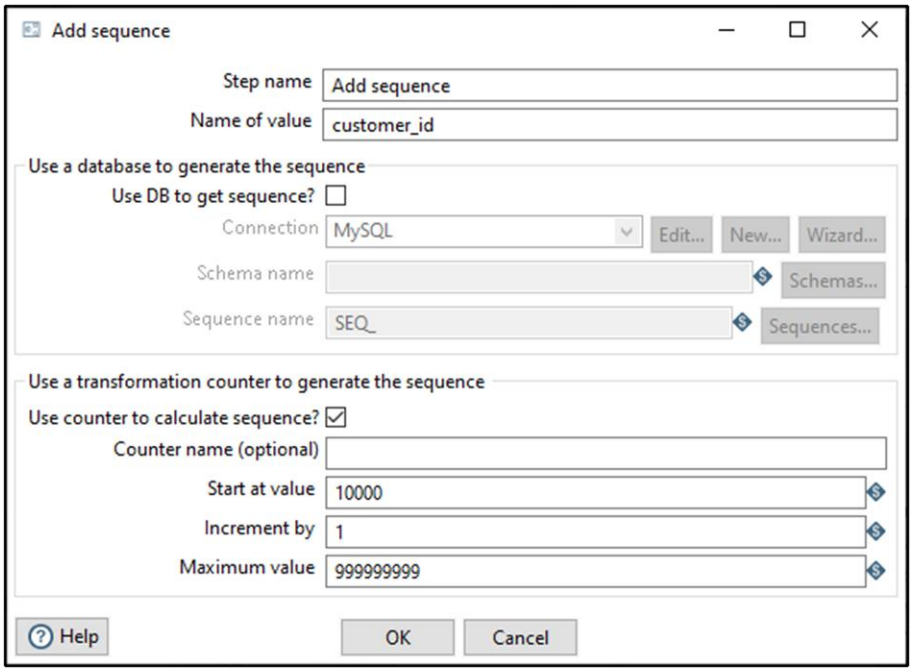
Slika 27. Prikaz dimenzijske tablice "Proizvod"

5.2.1.3. Transformacija dimenzijske tablice "Datum Prodaje"

date_tk	date	year	month	day
1	2011-01-01	2011	1	1
2	2011-02-01	2011	2	1
3	2011-03-01	2011	3	1
4	2011-04-01	2011	4	1
5	2011-06-01	2011	6	1
6	2011-07-01	2011	7	1
7	2011-08-01	2011	8	1
8	2011-09-01	2011	9	1
9	2011-10-01	2011	10	1
10	2011-11-01	2011	11	1

Slika 28. Prikaz dimenzijske tablice "Datum Prodaje"

5.2.1.4. Transformacija dimenzijske tablice "Kupac"



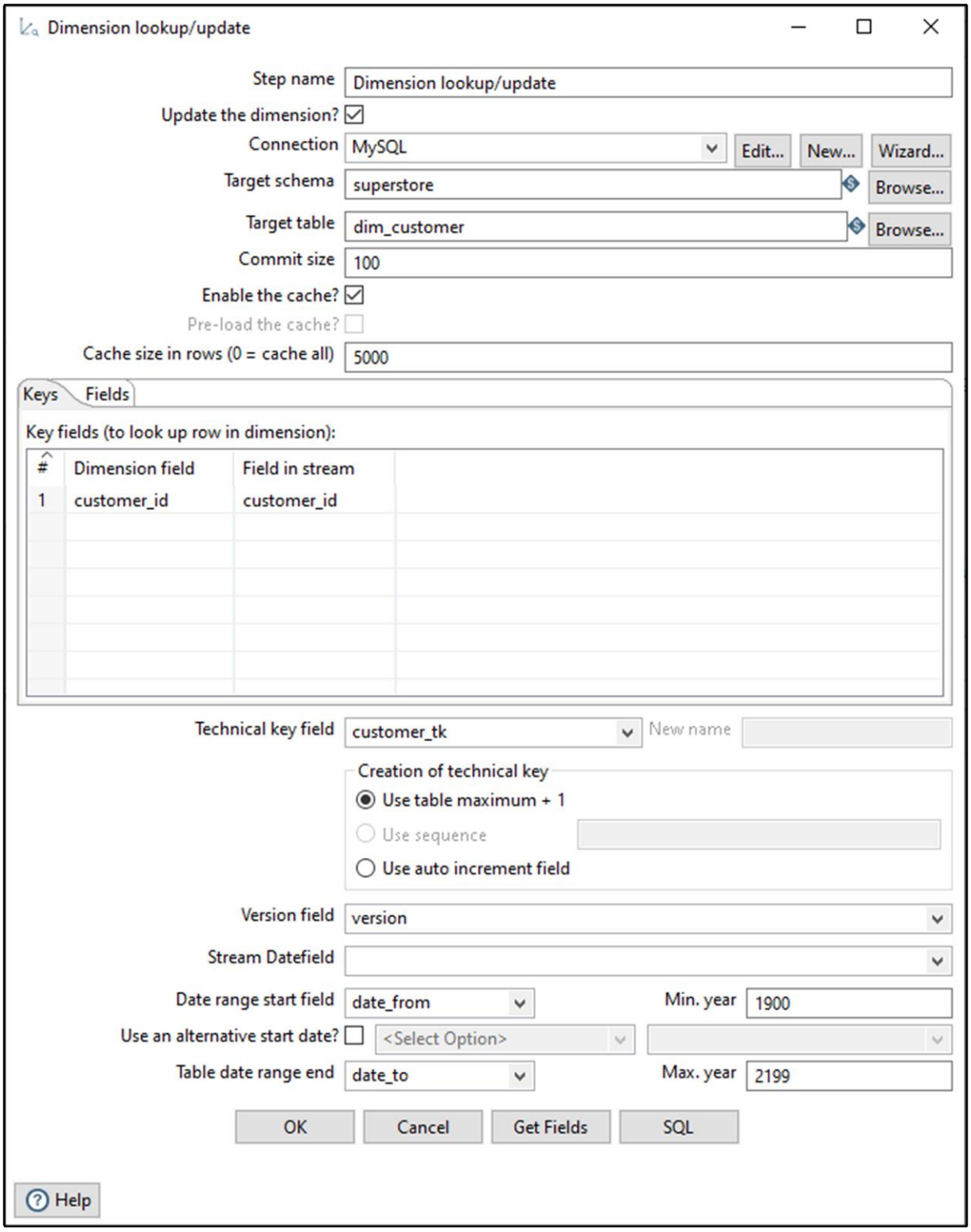
Slika 29. Prikaz naredbe "Add Sequence"

#	customer_id	customer	segment
1	335	Aaron Bergman	Consumer
2	10000	Aaron Bergman	Consumer
3	640	Aaron Hawkins	Corporate
4	10001	Aaron Hawkins	Corporate
5	33	Aaron Smayling	Corporate
6	10002	Aaron Smayling	Corporate
7	42	Adam Bellavance	Home Office
8	10003	Adam Bellavance	Home Office
9	127	Adam Hart	Corporate
10	10004	Adam Hart	Corporate

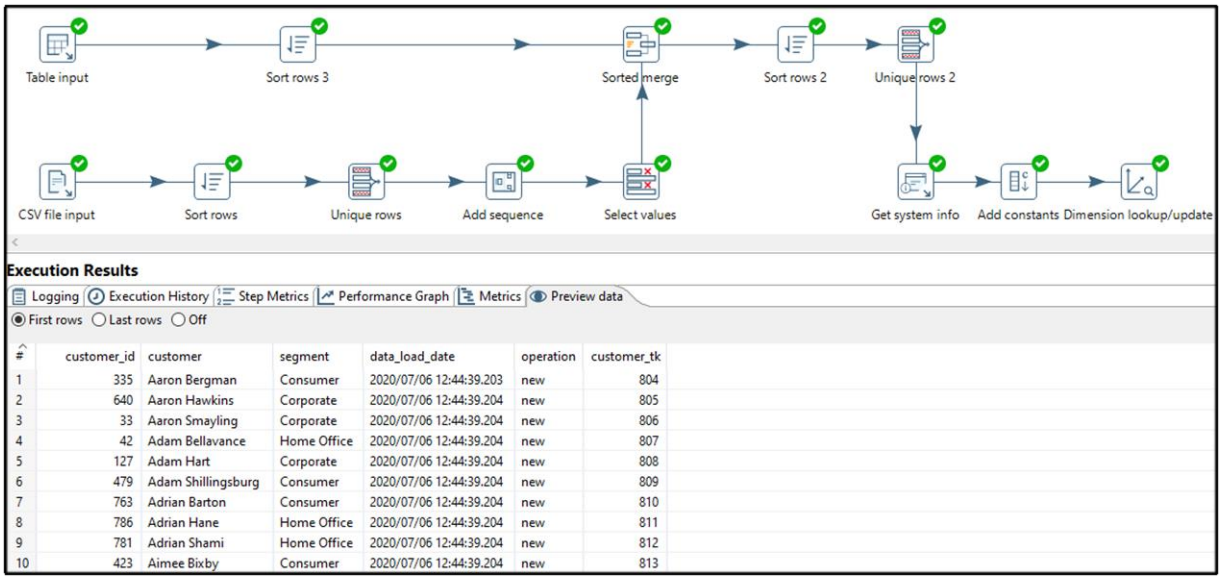
Slika 30. Prikaz rezultata naredbe "Sorted Merge"

#	customer_id	customer	segment
1	335	Aaron Bergman	Consumer
2	640	Aaron Hawkins	Corporate
3	33	Aaron Smayling	Corporate
4	42	Adam Bellavance	Home Office
5	127	Adam Hart	Corporate
6	479	Adam Shillingsburg	Consumer
7	763	Adrian Barton	Consumer
8	786	Adrian Hane	Home Office
9	781	Adrian Shami	Home Office
10	423	Aimee Bixby	Consumer

Slika 31. Prikaz rezultata naredbe "Unique Rows"



Slika 32. Prikaz naredbe “Dimension Lookup/Update“



#	customer_id	customer	segment	data_load_date	operation	customer_tk
1	335	Aaron Bergman	Consumer	2020/07/06 12:44:39.203	new	804
2	640	Aaron Hawkins	Corporate	2020/07/06 12:44:39.204	new	805
3	33	Aaron Smayling	Corporate	2020/07/06 12:44:39.204	new	806
4	42	Adam Bellavance	Home Office	2020/07/06 12:44:39.204	new	807
5	127	Adam Hart	Corporate	2020/07/06 12:44:39.204	new	808
6	479	Adam Shillingsburg	Consumer	2020/07/06 12:44:39.204	new	809
7	763	Adrian Barton	Consumer	2020/07/06 12:44:39.204	new	810
8	786	Adrian Hane	Home Office	2020/07/06 12:44:39.204	new	811
9	781	Adrian Shami	Home Office	2020/07/06 12:44:39.204	new	812
10	423	Aimee Bixby	Consumer	2020/07/06 12:44:39.204	new	813

Slika 33. Prikaz transformacijskog modela dimenzijske tablice "Kupac"

customer_tk	version	data_load_date	operation	date_from	date_to	customer_id	customer_name	segment
0	1	NULL	NULL	NULL	NULL	NULL	NULL	NULL
1	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	335	Aaron Bergman	Consumer
2	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	640	Aaron Hawkins	Corporate
3	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	33	Aaron Smayling	Corporate
4	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	42	Adam Bellava...	Home Office
5	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	127	Adam Hart	Corporate
6	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	479	Adam Shillings...	Consumer
7	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	763	Adrian Barton	Consumer
8	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	786	Adrian Hane	Home Office
9	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	781	Adrian Shami	Home Office
10	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	423	Aimee Bixby	Consumer

Slika 34. Prikaz dimenzijske tablice "Kupac"

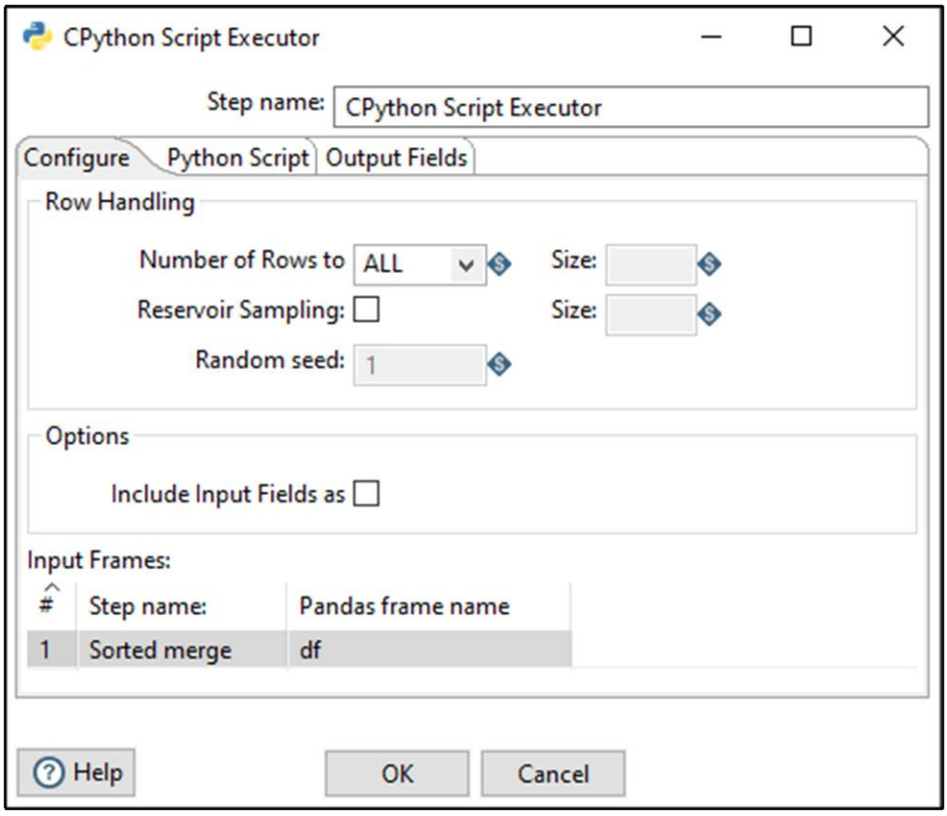
5.2.2. Učitavanje podataka u tablicu činjenica

Nakon uspješnih transformacija svih dimenzijskih tablica, potrebno je učitati podatke u tablicu činjenica. Prvi korak je identičan kao i kod transformacija dimenzijskih tablica, učitavamo podatke iz baze podataka pomoću naredbe "Table Input", te podatke iz CSV datoteke pomoću naredbe "CSV File Input". Ukoliko su podaci uspješno spojeni pomoću naredbe "Sorted Merge" (slika 35.), pozivamo naredbu "CPython Script Executor" kako bi ubrzali i pojednostavili proces učitavanja podataka u tablicu činjenica. Prvi korak unutar "CPython Script Executor" naredbe je dohvaćanje podataka iz prethodnog koraka (slika 36.). Sljedeći korak je stvaranje konekcije na bazu kako bi napravili upite koji dohvaćaju podatke iz dimenzijskih tablica vezane za kupce, proizvode i lokacije. Dohvaćamo njihove surogat ključeve, te zamjenjujemo originalne vrijednosti unutar dohvaćene kolekcije sa dohvaćenim surogat ključevima (slika 37.). Posljednji korak unutar navedene naredbe je postavljanje izlaznih vrijednosti transformirane kolekcije (slika 38.). Za učitavanje surogat ključeva vremenske dimenzije korištena je naredba

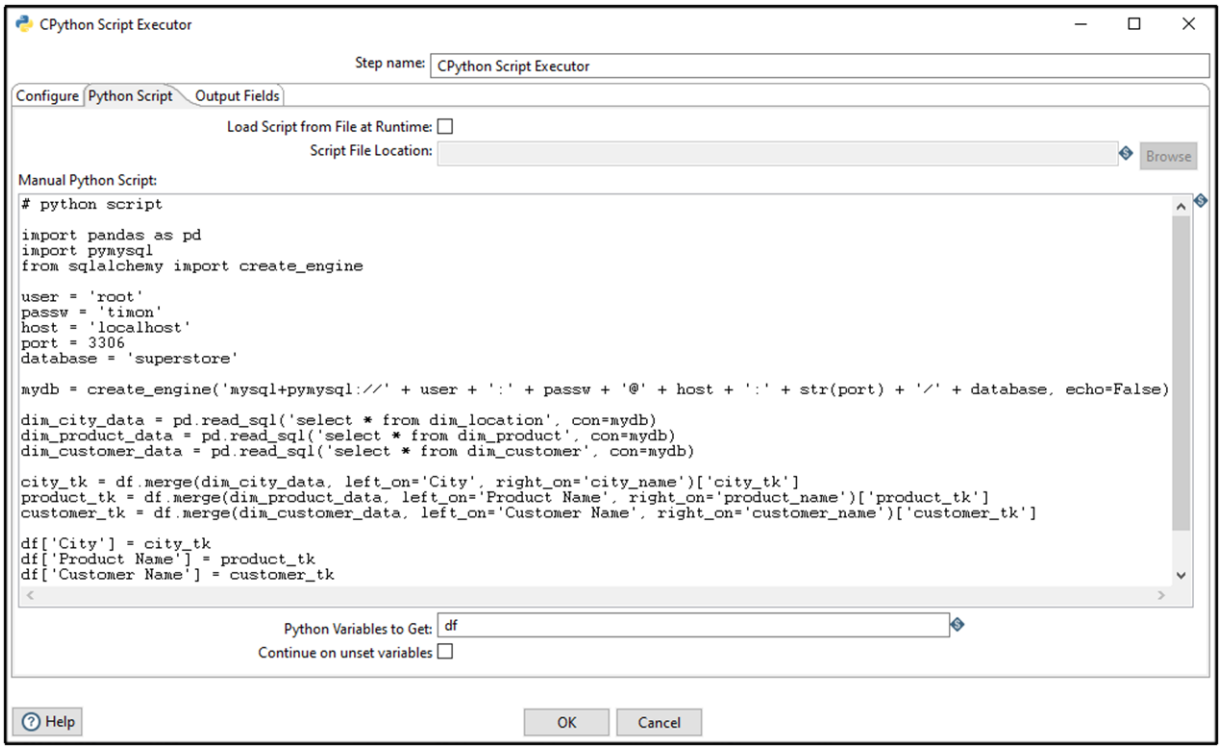
"Database Lookup". Unutar nje navodimo one atribute koje želimo usporediti uz odgovarajući komparator, te željenu izlaznu vrijednost koja je u ovom slučaju surogat ključ (slika 39.). Pomoću naredbe "Select Values" odabiremo željene atribute te ih transformiramo u format koji tablica činjenica očekuje (slika 40.). Upotrebom naredbe "Add Sequence" dodajemo surogat ključeve (slika 41.), te učitavamo podatke u tablicu činjenica pomoću naredbe "Table Output" (slike 42. i 43.). U nastavku poglavlja je opisani proces dokumentiran slikama implementacije iz Pentaho alata.

#	Shipping Cost	Sales	Quantity	Discount	Profit	Order Date	City	State	Country	Region	Market	Order Priority	Ship Mode	Product Name	Sub-Category	Category	Customer Name	Segment	Order ID
1	35.46	408.3	2	0.0	106.14	2011/01/01 00:00:00.000	Constantine	Constantine	Algeria	Africa	Africa	Medium	Standard Class	Tenex Locke...	Storage	Office ...	Toby Braunhar...	Consu...	1
2	9.72	120.366	3	0.1	36.036	2011/01/01 00:00:00.000	Wagga Wagga	New South Wales	Australia	Oceania	APAC	Medium	Standard Class	Acme Trim...	Supplies	Office ...	Joseph Holt	Consu...	2
3	8.17	66.12	4	0.0	29.64	2011/01/01 00:00:00.000	Budapest	Budapest	Hungary	EMEA	EMEA	High	Second Class	Tenex Box, S...	Storage	Office ...	Annie Thurman	Consu...	3
4	4.82	44.865	3	0.5	-26.055	2011/01/01 00:00:00.000	Stockholm	Stockholm	Sweden	North	EU	High	Second Class	Enermax No...	Paper	Office ...	Eugene Moren	Home ...	4
5	4.7	113.67	5	0.1	37.77	2011/01/01 00:00:00.000	Wagga Wagga	New South Wales	Australia	Oceania	APAC	Medium	Standard Class	Eldon Light ...	Furnishings	Furnit...	Joseph Holt	Consu...	5
6	1.8	55.242	2	0.1	15.342	2011/01/01 00:00:00.000	Wagga Wagga	New South Wales	Australia	Oceania	APAC	Medium	Standard Class	Eaton Comp...	Paper	Office ...	Joseph Holt	Consu...	6
7	57.3	285.78	2	0.0	71.4	2011/02/01 00:00:00.000	Dhaka	Dhaka	Bangladesh	Central Asia	APAC	Critical	Second Class	Brother Pers...	Copiers	Techn...	Patrick O'Don...	Consu...	7
8	54.64	290.666	2	0.15	3.4196	2011/02/01 00:00:00.000	Mission Viejo	California	United States	West	US	High	First Class	Sauder Face...	Bookcases	Furnit...	Liz Carlisle	Consu...	8
9	53.08	206.4	1	0.0	92.88	2011/02/01 00:00:00.000	Luanda	Luanda	Angola	Africa	Africa	Critical	Second Class	Fellowes Lo...	Storage	Office ...	David Kendrick	Corpo...	9
10	44.36	162.72	3	0.0	68.31	2011/02/01 00:00:00.000	Yingcheng	Hubei	China	North Asia	APAC	Critical	Second Class	Tenex Trays, ...	Storage	Office ...	Stephanie Phe...	Corpo...	10

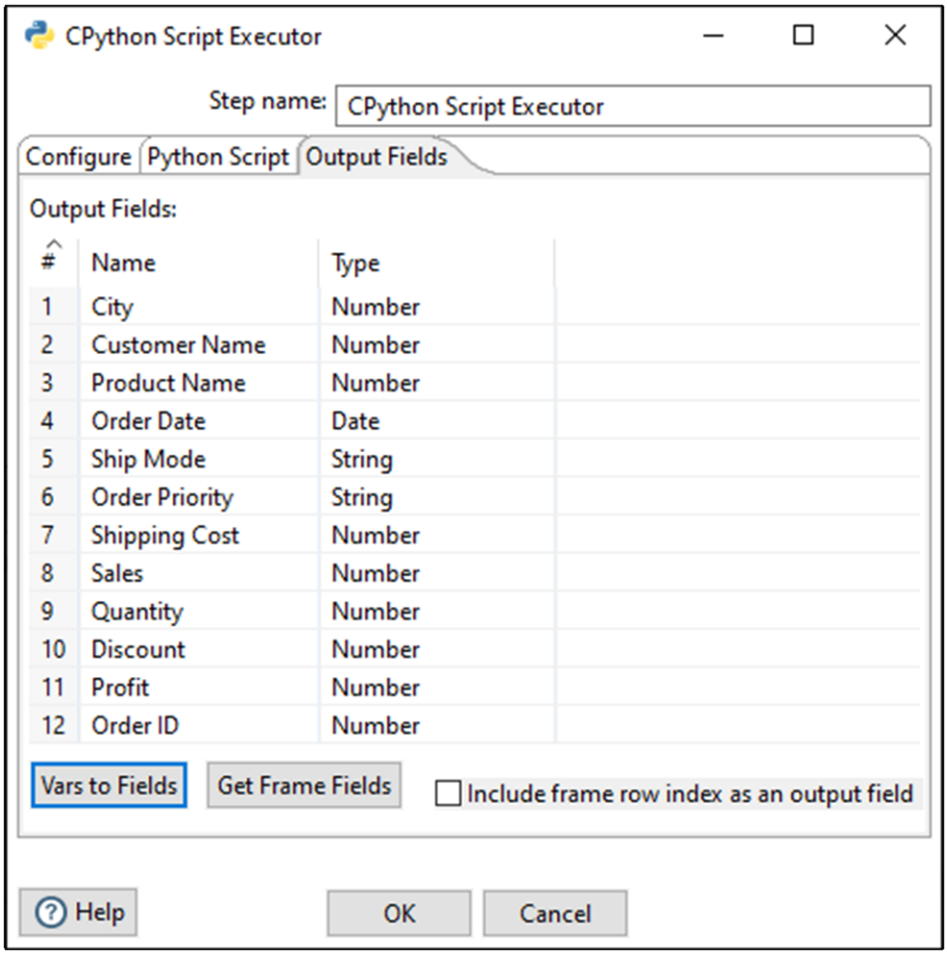
Slika 35. Prikaz rezultata naredbe "Sorted Merge"



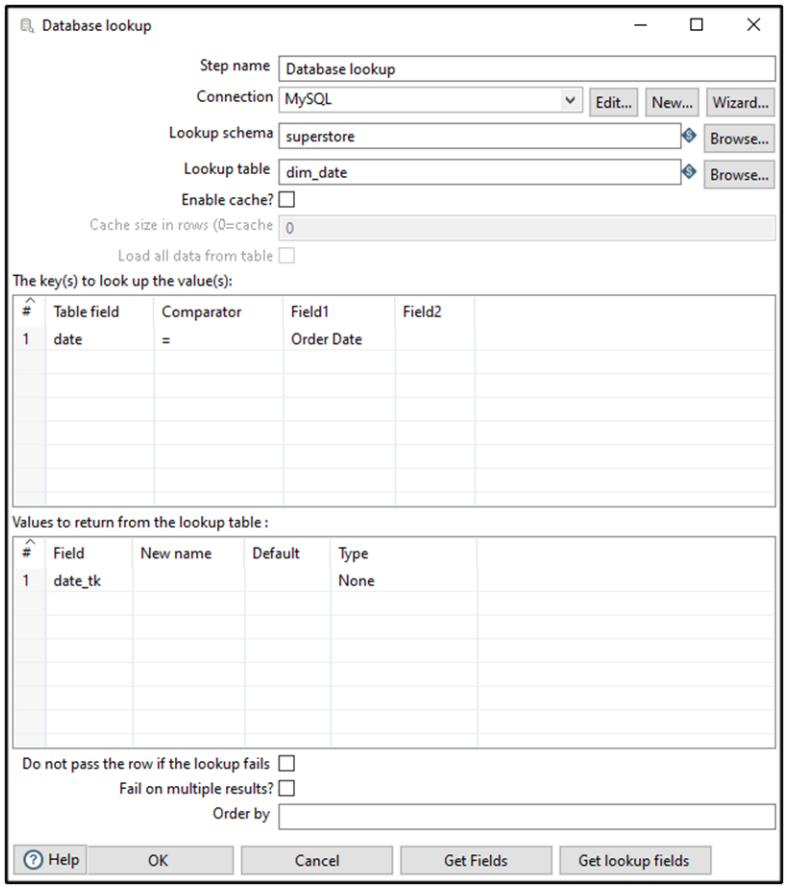
Slika 36. Prikaz naredbe "CPython Script Executor – Configuration"



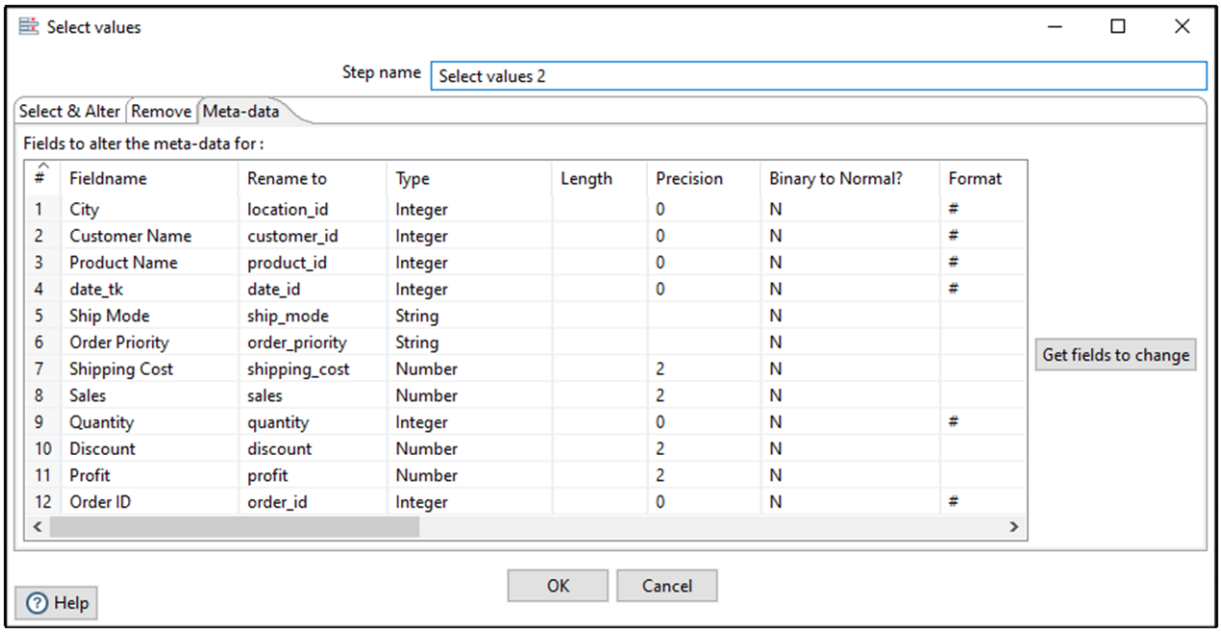
Slika 37. Prikaz naredbe "CPython Script Executor – Python Script"



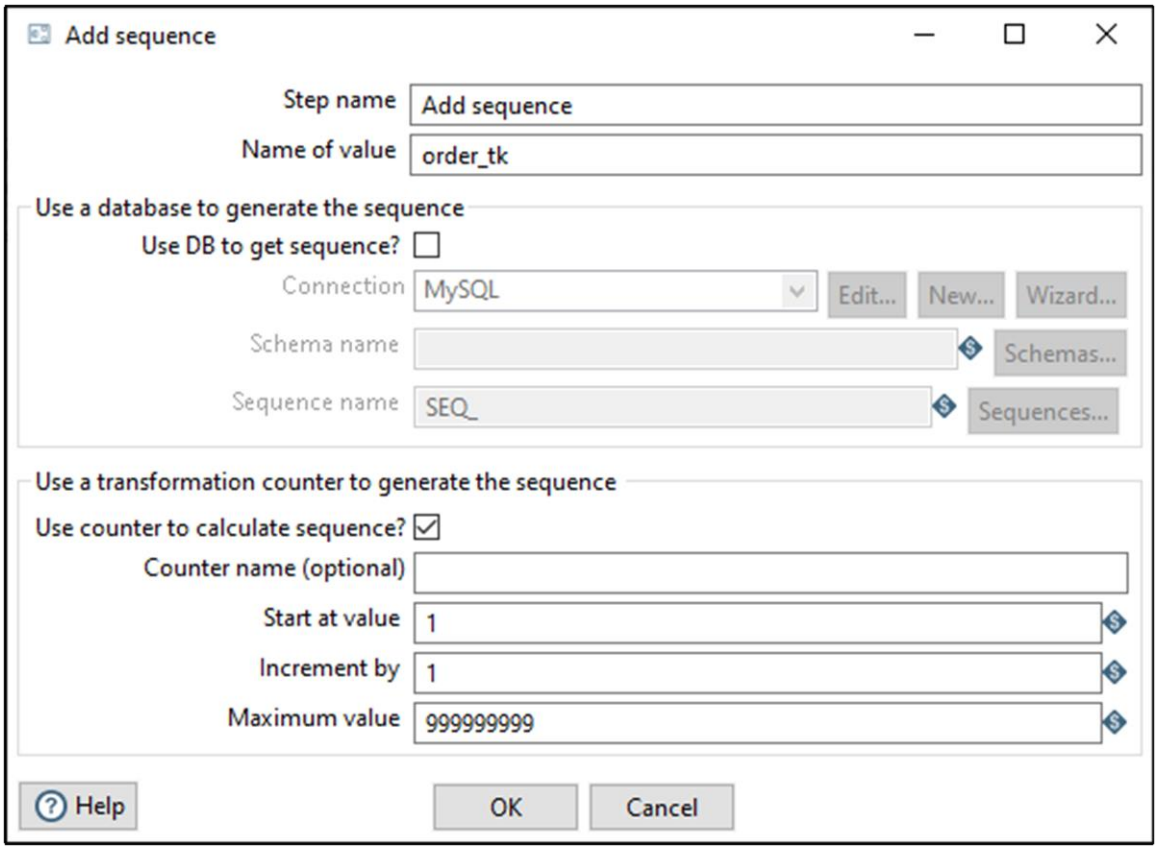
Slika 38. Prikaz naredbe "CPython Script Executor – Output Fields"



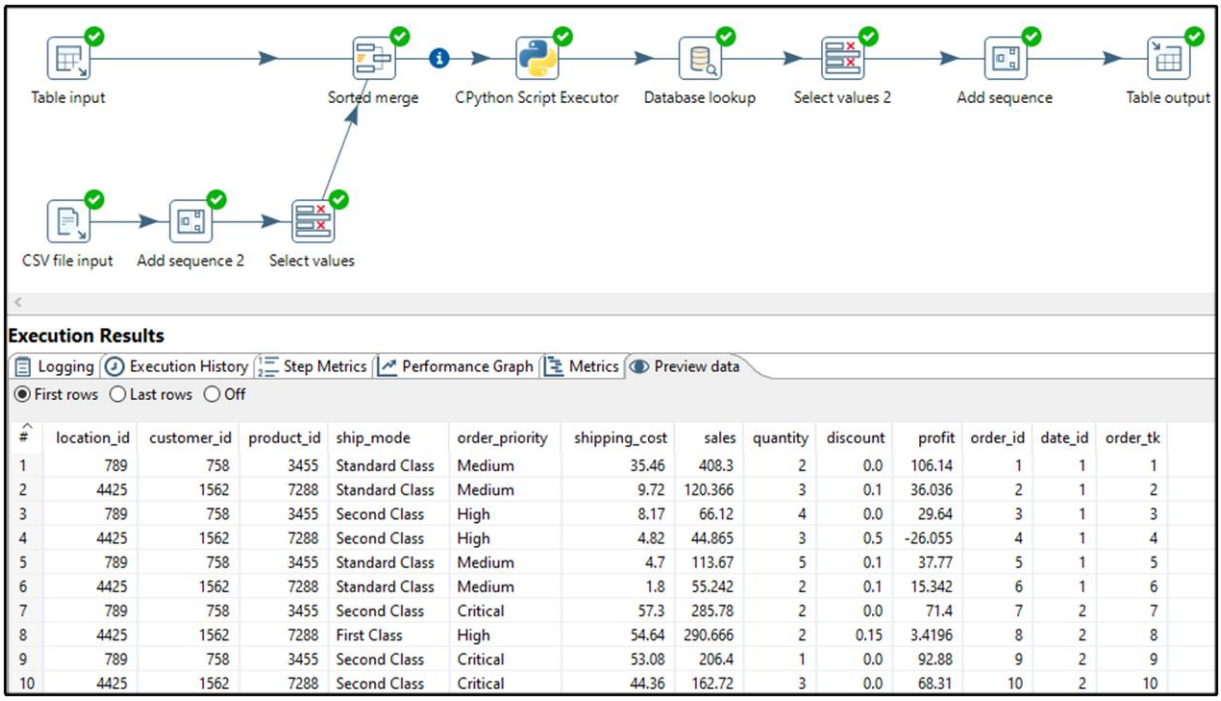
Slika 39. Prikaz naredbe “Database Lookup“



Slika 40. Prikaz naredbe “Select Values“



Slika 41. Prikaz naredbe "Add Sequence"



#	location_id	customer_id	product_id	ship_mode	order_priority	shipping_cost	sales	quantity	discount	profit	order_id	date_id	order_tk
1	789	758	3455	Standard Class	Medium	35.46	408.3	2	0.0	106.14	1	1	1
2	4425	1562	7288	Standard Class	Medium	9.72	120.366	3	0.1	36.036	2	1	2
3	789	758	3455	Second Class	High	8.17	66.12	4	0.0	29.64	3	1	3
4	4425	1562	7288	Second Class	High	4.82	44.865	3	0.5	-26.055	4	1	4
5	789	758	3455	Standard Class	Medium	4.7	113.67	5	0.1	37.77	5	1	5
6	4425	1562	7288	Standard Class	Medium	1.8	55.242	2	0.1	15.342	6	1	6
7	789	758	3455	Second Class	Critical	57.3	285.78	2	0.0	71.4	7	2	7
8	4425	1562	7288	First Class	High	54.64	290.666	2	0.15	3.4196	8	2	8
9	789	758	3455	Second Class	Critical	53.08	206.4	1	0.0	92.88	9	2	9
10	4425	1562	7288	Second Class	Critical	44.36	162.72	3	0.0	68.31	10	2	10

Slika 42. Prikaz transformacijskog modela tablice činjenica

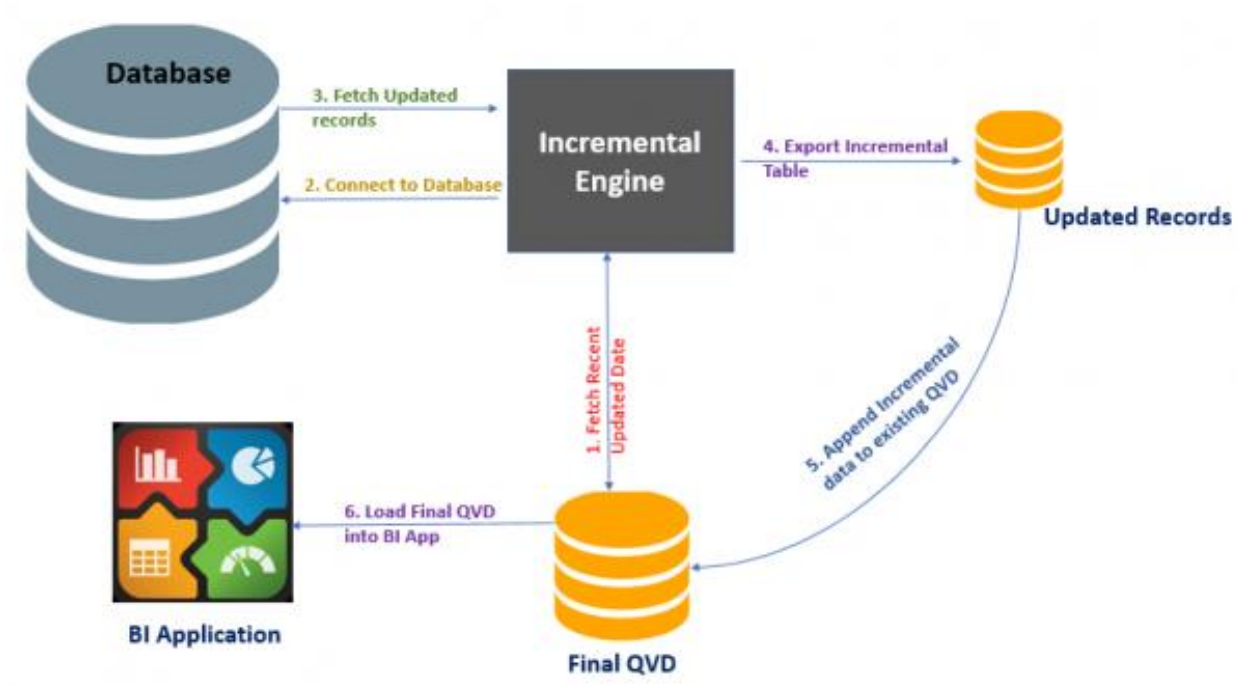
order_tk	order_id	product_id	location_id	customer_id	date_id	ship_mode	order_priority	shipping_cost	sales	quantity	discount	profit
1	1	3455	789	758	1	Standard Class	Medium	35.46	408.3	2	0	106.14
2	2	3455	789	758	1	Standard Class	Medium	9.72	120.366	3	0.1	36.036
3	3	3455	789	758	1	Second Class	High	8.17	66.12	4	0	29.64
4	4	3455	789	758	1	Second Class	High	4.82	44.865	3	0.5	-26.055
5	5	3455	789	758	1	Standard Class	Medium	4.7	113.67	5	0.1	37.77
6	6	3455	789	758	1	Standard Class	Medium	1.8	55.242	2	0.1	15.342
7	7	3455	789	758	2	Second Class	Critical	57.3	285.78	2	0	71.4
8	8	3455	789	758	2	First Class	High	54.64	290.666	2	0.15	3.4196
9	9	3455	3453	758	2	Second Class	Critical	53.08	206.4	1	0	92.88
10	10	3455	3453	758	2	Second Class	Critical	44.36	162.72	3	0	68.31

Slika 43. Prikaz tablice činjenica

6. Inkrementalni ETL proces

6.1. Uvod u inkrementalni ETL proces

Alati za izvođenje ETL procesa primarno su dizajnirani za učitavanje podataka u skladište podataka, tj. za fizičku integraciju podataka. Kada dođe do promjene operativnih izvora podataka, skladište podataka postaje "nepomično". Da bi se osigurala pravovremenost podataka, skladište podataka povremeno se osvježava. Naivni pristup ponovnog učitavanja skladišta podataka očito je neučinkovit. Tipično se samo mali dio izvornih podataka mijenja tijekom ciklusa učitavanja. Stoga je poželjno da se te promjene zabilježe u operativnim izvorima podataka i da se skladište podataka ažurira inkrementalno (slika 44.). Takav pristup je poznat kao inkrementalno punjenje ili inkrementalan ETL proces (Mekterović & Brkić, 2017.).



Slika 44. Prikaz koraka tradicionalnog inkrementalnog ETL procesa (Dusa, 2017.)

6.2. CDC (engl. Change Data Capture)

Kao što ime sugerira, CDC tehnike služe za prepoznavanje promjena u podacima. CDC može biti osnova za sinkronizaciju drugog sustava s istim inkrementalnim promjenama ili za pohranu revizijskog traga promjena. Revizijski trag može se naknadno koristiti za druge svrhe, npr. ažurirati skladište podataka, analizirati promjene ili prepoznati obrasce promjena. U ovom završnom radu je korištena razlika snimki stanja kao CDC tehnika za prepoznavanje promjena (Mekterović & Brkić, 2017.).

6.2.1. Razlika snimki stanja

Razlika snimki stanja uspoređuje trenutačno stanje podataka s prethodnim stanjem podataka kako bi se identificirale promjene (Mekterović & Brkić, 2017.). Izazovi takvog pristupa uključuju:

- Za izvršavanje razlike snimki stanja potrebno je mnogo resursa za izračunavanje razlika između podataka, a potrošnja resursa raste bar linearno s rastom volumena podataka (Mekterović & Brkić, 2017.)
- CDC se ne može izvoditi u stvarnom vremenu, jer razlika snimki stanja realno zahtijeva previše resursa da bi se izvodila cijelo vrijeme (Mekterović & Brkić, 2017.)

U usporedbi sa CDC tehnikom dodavanja vremenske oznake, razlika snimki stanja nema izazova s izbrisanim redovima te je efikasna za male količine podataka.

6.3. Primjena inkrementalnog ETL procesa nad dimenzijskim tablicama

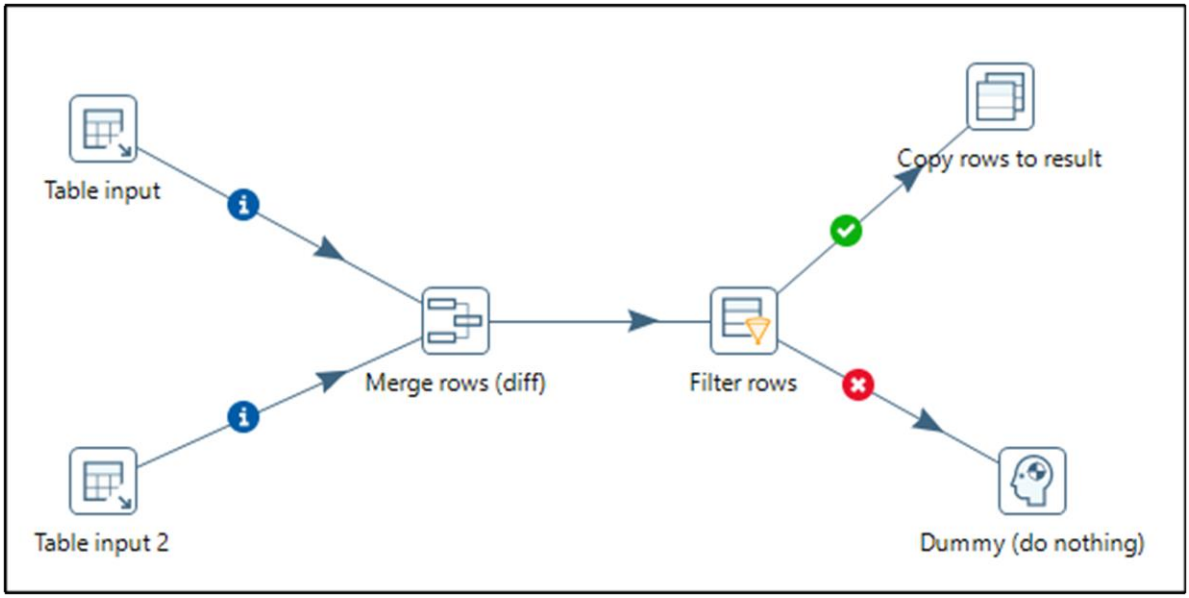
Prvi korak inkrementalnog punjenja skladišta podataka je priprema podataka za učitavanje istih u pripremno područje. Korištenjem naredba “Table Input“ učitavamo podatke iz izvorišne baze i podatke iz pripremne tablice (slika 45.). Naredbi “Merge Rows (diff)“ predajemo ključ po kojem će uspoređivati promjene vrijednosti na temelju danih izvora i svakoj postaviti dodatni atribut “flagfield“ koji označava dali je pojedini redak promijenjen, isti ili izbrisan (slika 46.). Koristeći naredbu “Filter Rows“ filtriramo nepromijenjene vrijednosti te ukoliko su sve vrijednosti nepromijenjene, ne radimo ništa, a u suprotnom spremamo retke od interesa u rezultat (slika 47.).

U sljedećem koraku dohvaćamo spremljene vrijednosti (slika 48.) te pomoću naredbe “Insert/Update“ (slika 49.), spremamo, odnosno ažuriramo podatke u pripremnoj tablici.

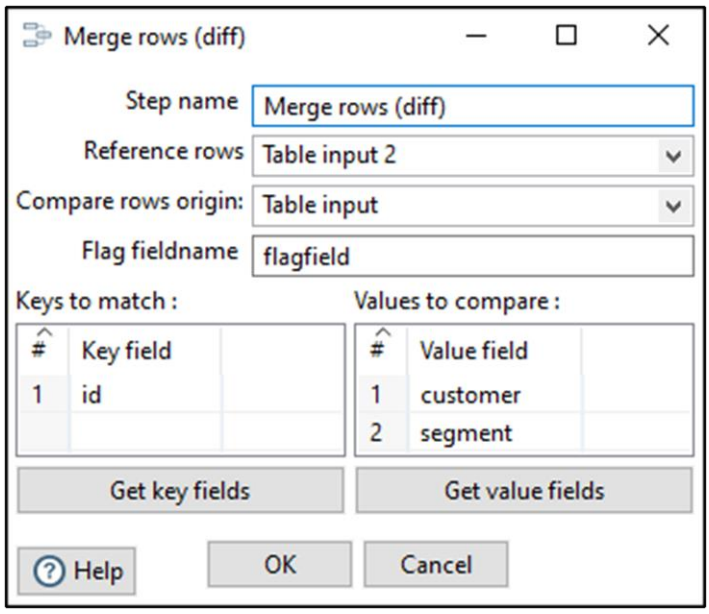
Posljednjim korakom dohvaćamo sve podatke iz pripremne tablice, dodajemo poseban atribut pomoću naredbe “Get System Info“ (slika 50.) koji označava trenutno

vrijeme izvršenja inkrementalnog ETL procesa (slika 51.) te pomoću naredbe "Dimension lookup/update" (slika 52.) ažuriramo ili dodajemo nove podatke.

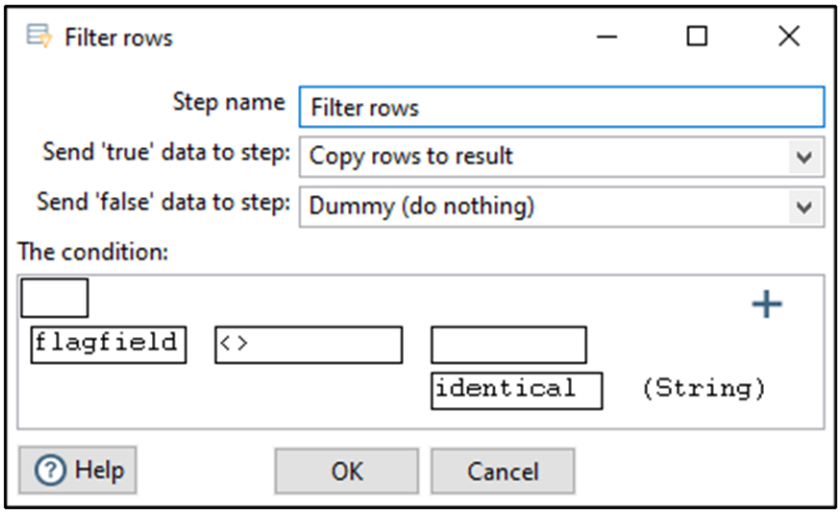
Sve tri navedene transformacije potrebne za izvršenje inkrementalnog ETL procesa izvode se pomoću ETL poslova (engl. Jobs) koji koordiniraju aktivnosti unutar ETL procesa, u ovom slučaju sekvencijalno (slika 53.). U nastavku poglavlja je opisani proces dokumentiran slikama implementacije iz Pentaho alata.



Slika 45. Prikaz transformacije za identifikaciju novih i promijenjenih podataka

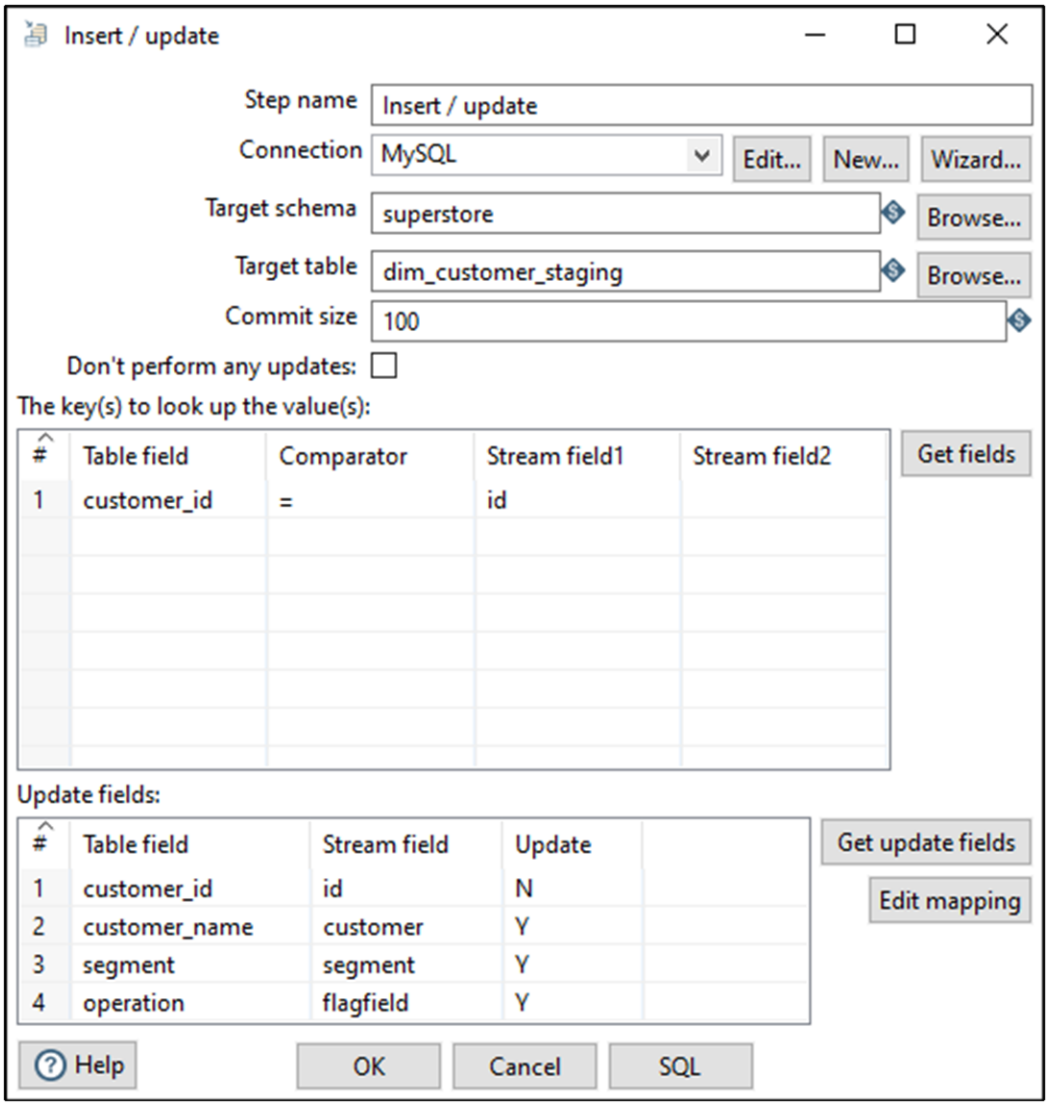


Slika 46. Prikaz naredbe "Merge Rows (diff)"

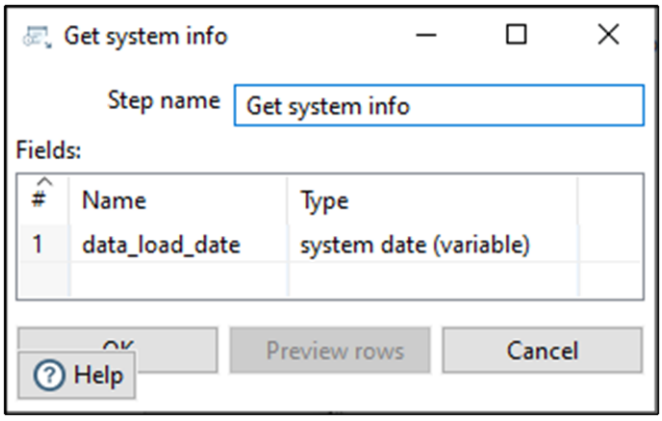


Slika 47. Prikaz naredbe “Filter Rows“

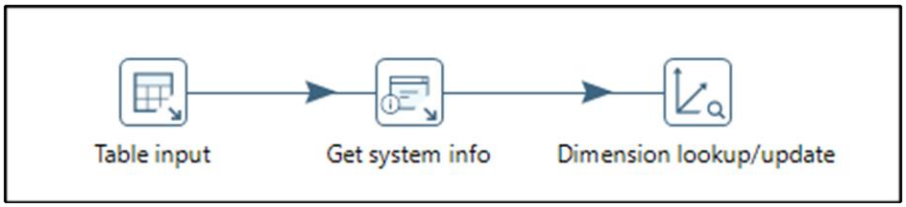
Slika 48. Prikaz transformacije za punjenje i/ili ažuriranje pripremne tablice



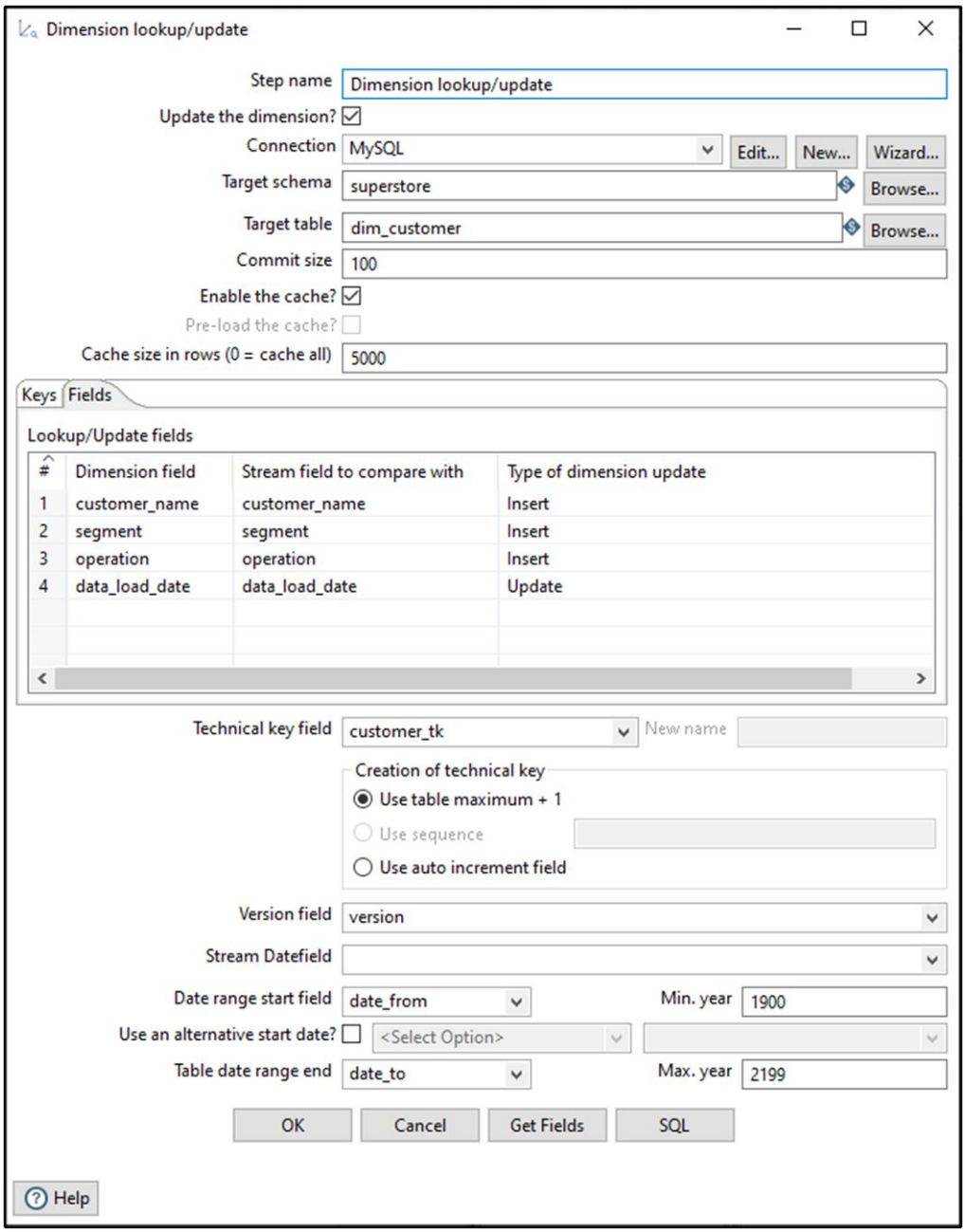
Slika 49. Prikaz naredbe “Insert/Update“



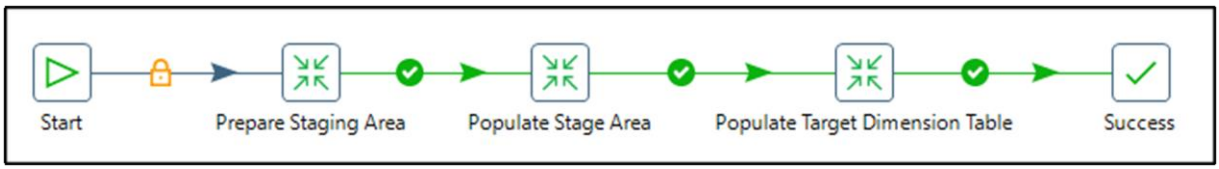
Slika 50. Prikaz naredbe “Get System Info“



Slika 51. Prikaz transformacije za punjenje i/ili ažuriranje dimenzijske tablice



Slika 52. Prikaz naredbe "Dimension lookup/update"



Slika 53. Prikaz sekvencijalnog ETL posla

6.3.1. Inkrementalno punjenje dimenzijske tablice “Kupac“

Nakon promjene imena kupca u bazi podataka, pokretanjem ETL posla se isto promijenilo i u pripremnom području (slika 54.), te i u dimenzijskog tablici “Kupac“ što možemo vidjeti po indikatoru “version“ (slika 55.).

customer_id	customer_name	segment	operation
796	Timon Dudaković	Consumer	changed
795	Harold Ryan	Corporate	new
794	Carlos Meador	Consumer	new
793	Ed Jacobs	Consumer	new
792	Stefanie Holloman	Corporate	new
791	Robert Barroso	Corporate	new
790	Carol Darley	Consumer	new
789	Liz MacKendrick	Consumer	new
788	Anemone Ratner	Consumer	new
787	Jane Waco	Corporate	new

Slika 54. Prikaz pripremne tablice “Kupac“ nakon promjene imena kupca

customer_tk	version	data_load_date	operation	date_from	date_to	customer_id	customer_name	segment
803	2	2020-06-02 15:49:14	changed	2020-06-02 15:49:14	2200-01-01 00:00:00	796	Timon Dudaković	Consumer
802	1	2020-06-02 15:48:27	new	1900-01-01 00:00:00	2020-06-02 15:49:14	796	Timon	Consumer
801	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	469	Zuschuss Don...	Consumer
800	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	751	Zuschuss Carroll	Consumer
799	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	138	Yoseph Carroll	Corporate
798	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	47	Yana Sorensen	Corporate
797	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	242	Xylona Preis	Consumer
796	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	51	William Brown	Consumer
795	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	397	Vivian Mathis	Consumer
794	1	2020-06-02 15:49:14	new	1900-01-01 00:00:00	2200-01-01 00:00:00	565	Vivek Sundare...	Consumer

Slika 55. Prikaz dimenzijske tablice “Kupac“ nakon promjene imena kupca

6.3.2. Inkrementalno punjenje dimenzijske tablice “Proizvod“

Dodavanjem novog proizvoda u bazu podataka pod jedinstvenim identifikatorom “3732“, te pokretanjem ETL posla za ažuriranje dimenzijske tablice proizvod, možemo vidjeti da se ažuriralo i pripremno područje (slika 56.) i dimenzijska tablica “Proizvod“ (slika 57.).

product_id	product_name	product_sub_category	product_category	operation
3732	Huawei P30 Lite	Phones	Technology	changed
3731	Griffin GC17055 Auxiliary Audio Cable	Phones	Technology	new
3730	Epson Perfection V600 Photo Scanner	Machines	Technology	new
3729	Bevis Round Table, Fully Assembled	Tables	Furniture	new
3728	ACCOHIDE Binder by Acco	Binders	Office Supplies	new
3727	#6 3/4 Gummed Flap White Envelopes	Envelopes	Office Supplies	new
3726	SAFCO Commercial Wire Shelving, 72h	Storage	Office Supplies	new
3725	Xerox 1901	Paper	Office Supplies	new
3724	Avery Hi-Liter GlideStik Fluorescent ...	Art	Office Supplies	new
3723	Disposable Triple-Filter Dust Bags	Appliances	Office Supplies	new

Slika 56. Prikaz pripremne tablice “Proizvod“ nakon promjena imena proizvoda

product_tk	version	data_load_date	operation	date_from	date_to	product_id	product_name	product_sub_category	product_category
3832	2	2020-06-02 15:59:05	changed	2020-06-02 15:59:05	2200-01-01 00:00:00	3732	Huawei P30 ...	Phones	Technology
3831	1	2020-06-02 15:57:58	new	1900-01-01 00:00:00	2020-06-02 15:59:05	3732	Huawei P30	Phones	Technology
3830	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	96	Zipper Ring ...	Binders	Office Supplies
3829	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	2596	Zebra ZM40...	Machines	Technology
3828	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	620	Zebra Zazzle...	Art	Office Supplies
3827	1	2020-05-31 17:23:28	new	1900-01-01 00:00:00	2200-01-01 00:00:00	13364	Zebra GX420...	Machines	Technology
3826	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	3086	Zebra GK420...	Machines	Technology
3825	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	2905	XtraLife Clea...	Binders	Office Supplies
3824	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	2874	XtraLife Clea...	Binders	Office Supplies
3823	1	2020-06-02 15:59:05	new	1900-01-01 00:00:00	2200-01-01 00:00:00	3587	Xiaomi Mi3	Phones	Technology

Slika 57. Prikaz dimenzijske tablice “Proizvod“ nakon promjena imena proizvoda

6.3.3. Inkrementalno punjenje dimenzijske tablice “Lokacija“

Kao i kod dimenzijske tablice “Proizvod“, nakon dodavanja novog grada i pokretanjem ETL posla možemo vidjeti iste promjene u pripremnom području (slika 58.) i u dimenzijskog tablici “Lokacija“ (slika 59.).

city_id	city_name	state_name	country_name	region_name	market_name	operation
3247	Pula	Istarska	Croatia	EMEA	EMEA	changed
3246	Taixing	Jiangsu	China	North Asia	APAC	new
3245	Teramo	Abruzzi	Italy	South	EU	new
3244	Jizan	Jizan	Saudi Arabia	EMEA	EMEA	new
3243	Lincoln Park	Michigan	United States	West	US	new
3242	Tokat	Tokat	Turkey	EMEA	EMEA	new
3241	Ragusa	Sicily	Italy	South	EU	new
3240	Caraguatatuba	Sao Paulo	Brazil	South	EU	new
3239	Guntur	Andhra Pradesh	India	Central Asia	APAC	new
3238	Bagnolet	Ile-de-France	France	Central	LATAM	new

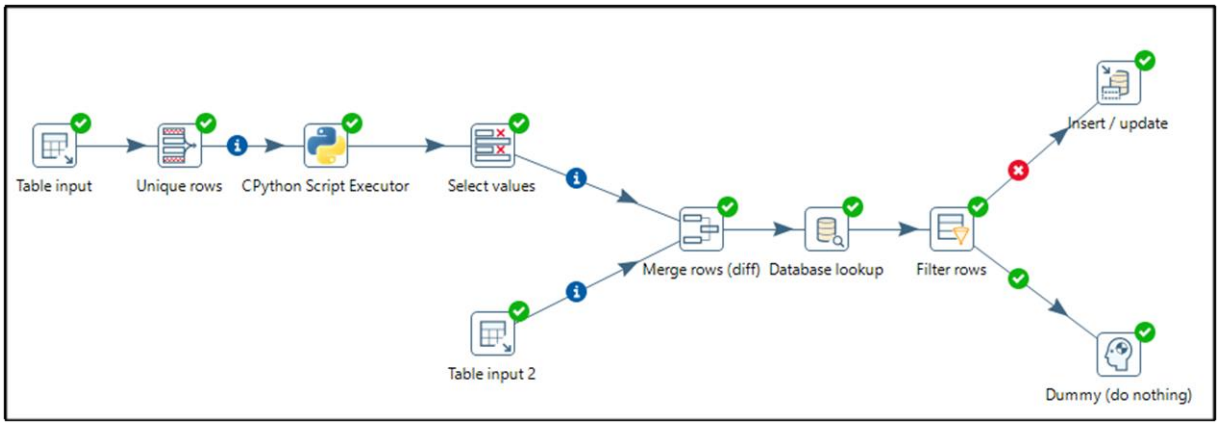
Slika 58. Prikaz pripremne tablice “Lokacija“ nakon promjene imena grada

city_tk	version	data_load_date	operation	date_from	date_to	city_id	city_name	state_name	country_name	region_name	market_name
3636	3	2020-06-02 16:39:23	changed	2020-06-02 16:39...	2200-01-01 00:00...	3247	Pula	Istarska	Croatia	EMEA	EMEA
3635	2	2020-06-02 16:38:44	changed	2020-06-02 16:38...	2020-06-02 16:39...	3247	Pula	Istarska	Croatia	EMEA	EMEA
3634	1	2020-06-02 16:35:55	new	1900-01-01 00:00...	2020-06-02 16:38...	3247	Pula	Istarska	Croatia	EMEA	EMEA
3633	1	2020-06-02 16:39:23	new	1900-01-01 00:00...	2200-01-01 00:00...	2278	Zwolle	Overijssel	Netherlands	Central	LATAM
3632	1	2020-06-02 16:39:23	new	1900-01-01 00:00...	2200-01-01 00:00...	2563	Zwickau	Saxony	Germany	Central	LATAM
3631	1	2020-06-02 16:39:23	new	1900-01-01 00:00...	2200-01-01 00:00...	412	Zwedru	Grand Ge...	Liberia	Africa	Africa
3630	1	2020-06-02 16:39:23	new	1900-01-01 00:00...	2200-01-01 00:00...	1137	Zurich	Zurich	Switzerland	Central	LATAM
3629	1	2020-06-02 16:39:23	new	1900-01-01 00:00...	2200-01-01 00:00...	263	Zunyi	Guizhou	China	North Asia	APAC
3628	1	2020-06-02 16:39:23	new	1900-01-01 00:00...	2200-01-01 00:00...	1890	Zlatoust	Chelyabinsk	Russia	EMEA	EMEA
3627	1	2020-06-02 16:39:23	new	1900-01-01 00:00...	2200-01-01 00:00...	1953	Zinder	Zinder	Niger	Africa	Africa

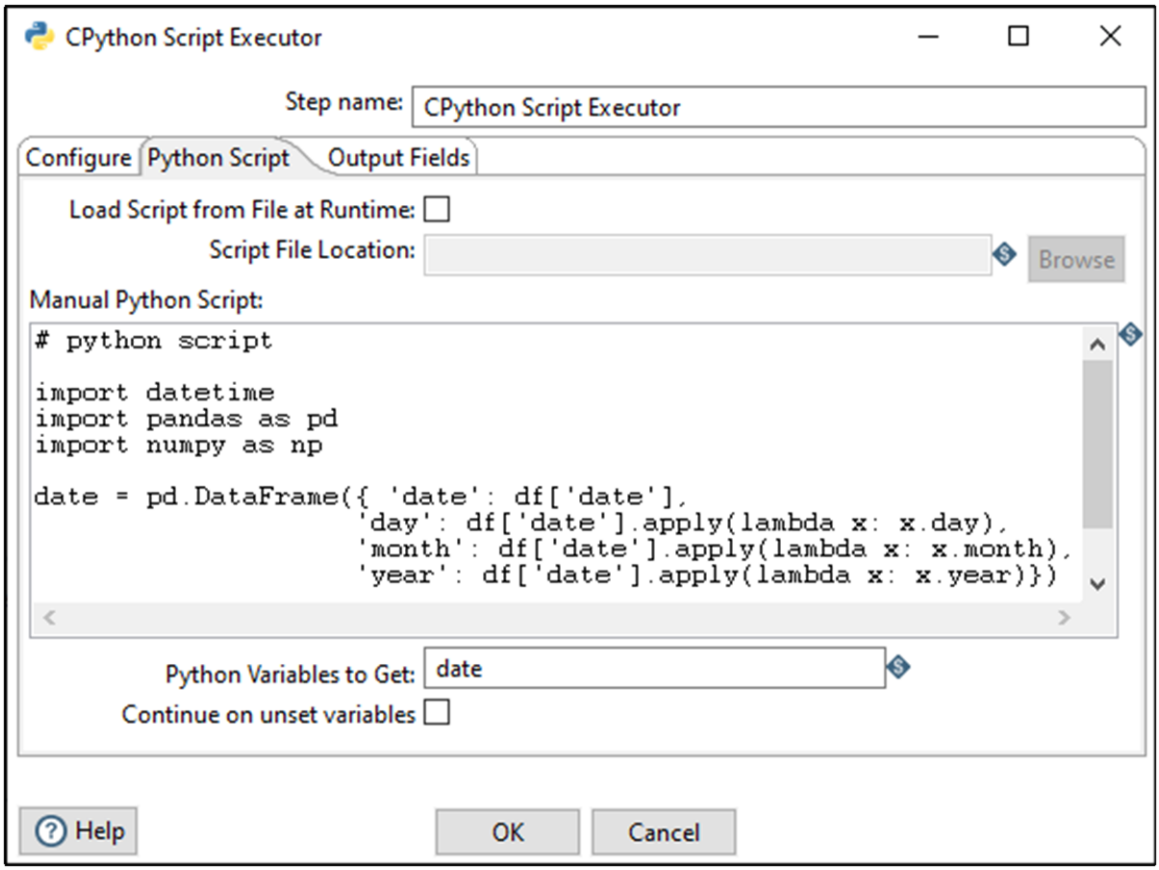
Slika 59. Prikaz dimenzijske tablice “Lokacija“ nakon promjene imena grada

6.3.4. Inkrementalno punjenje dimenzijske tablice “Datum prodaje“

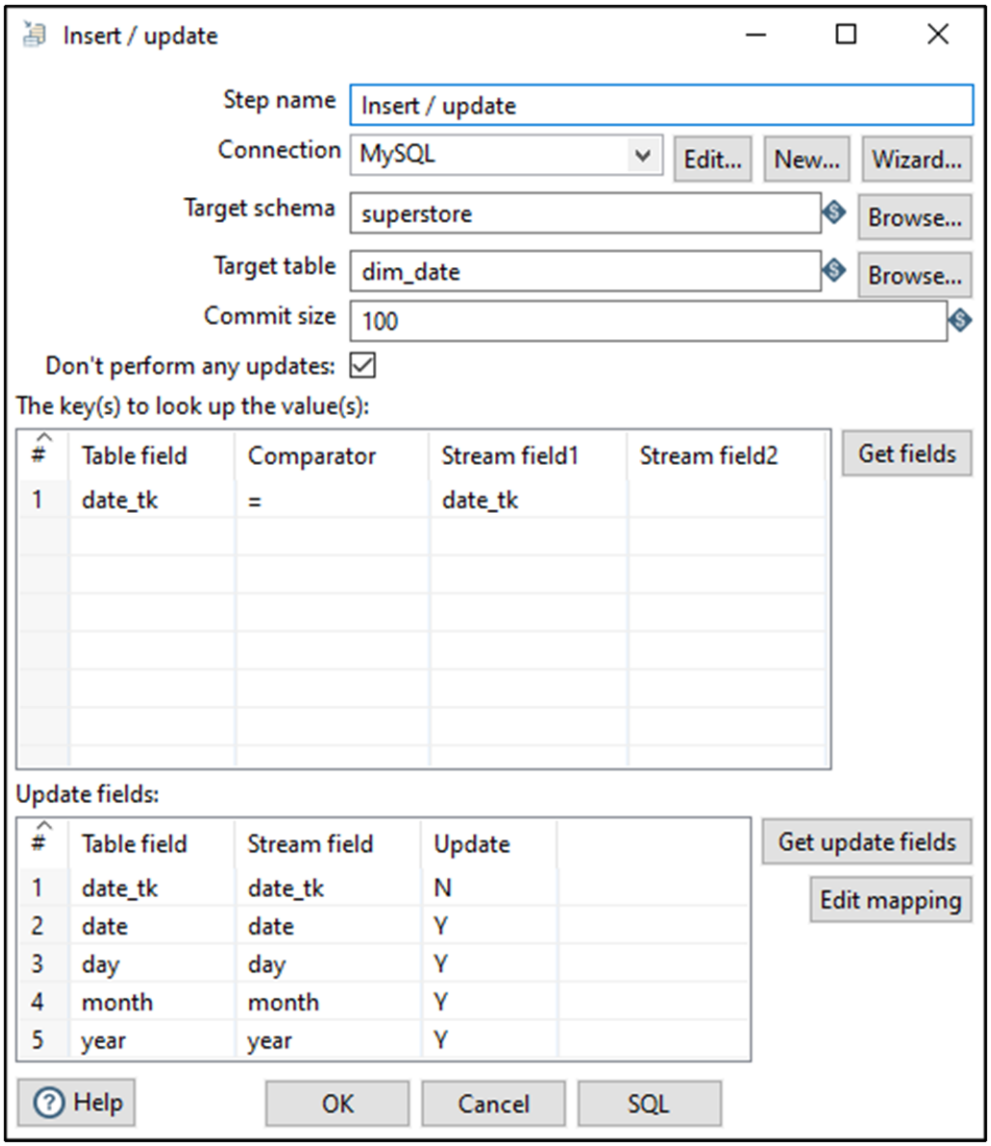
Za inkrementalno punjenje dimenzijske tablice “Datum prodaje“ bilo je potrebno izdvojiti datum iz relacijske tablice “Narudžba“, transformirati podatke u ispravan format (slika 61.), te ih spojiti sa podacima iz dimenzijske tablice “Datum prodaje“ pomoću naredbe “Merge Rows (diff)“. Naredbom “Database lookup“ dohvaćamo surogat ključeve iz dimenzijske tablice te filtriramo datume koji imaju vrijednost “identical“ ili “deleted“ u “flagfield“ atributu kako bi pomoću naredbe “Insert/Update“ (slika 62.) unijeli samo željene vrijednosti. Slika 60. prikazuje transformaciju za inkrementalno punjenje vremenske dimenzije.



Slika 60. Prikaz transformacije za inkrementalno punjenje dimenzijske tablice



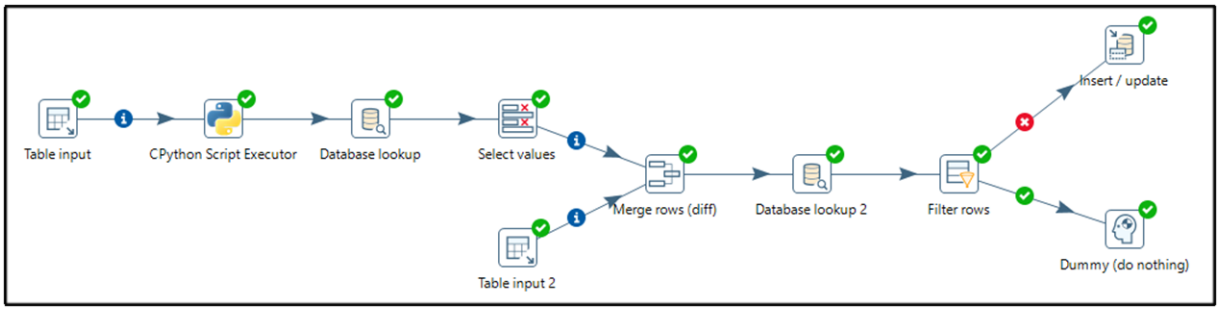
Slika 61. Prikaz naredbe "CPython Script Executor"



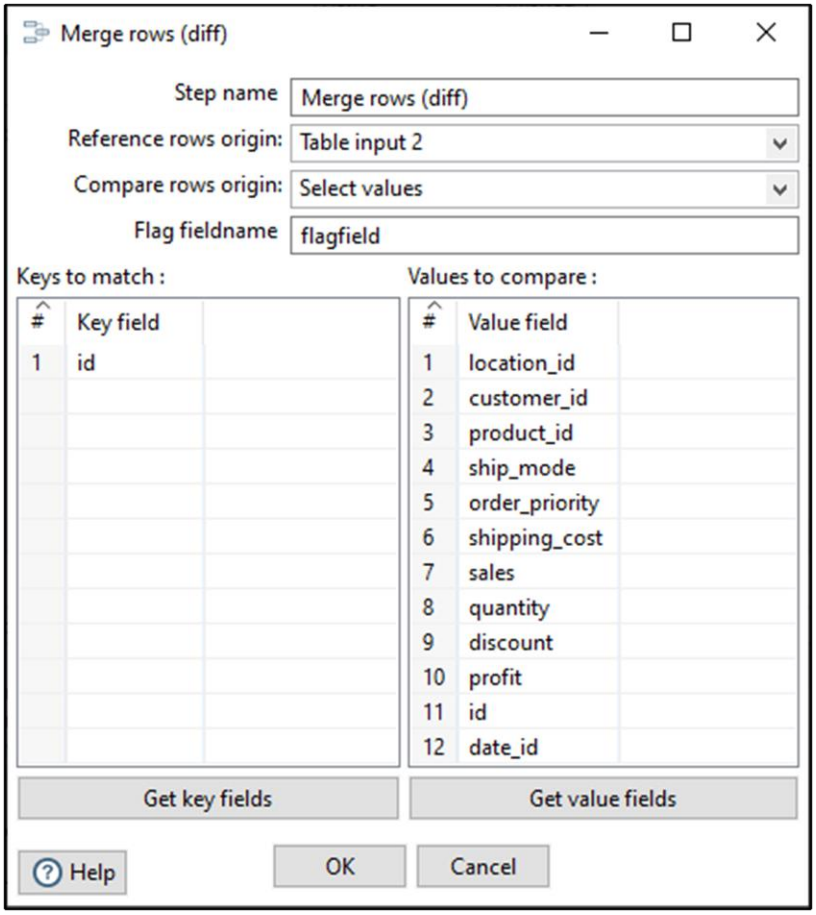
Slika 62. Prikaz naredbe “Insert/Update“

6.4. Primjena inkrementalnog ETL procesa nad tablicom činjenica

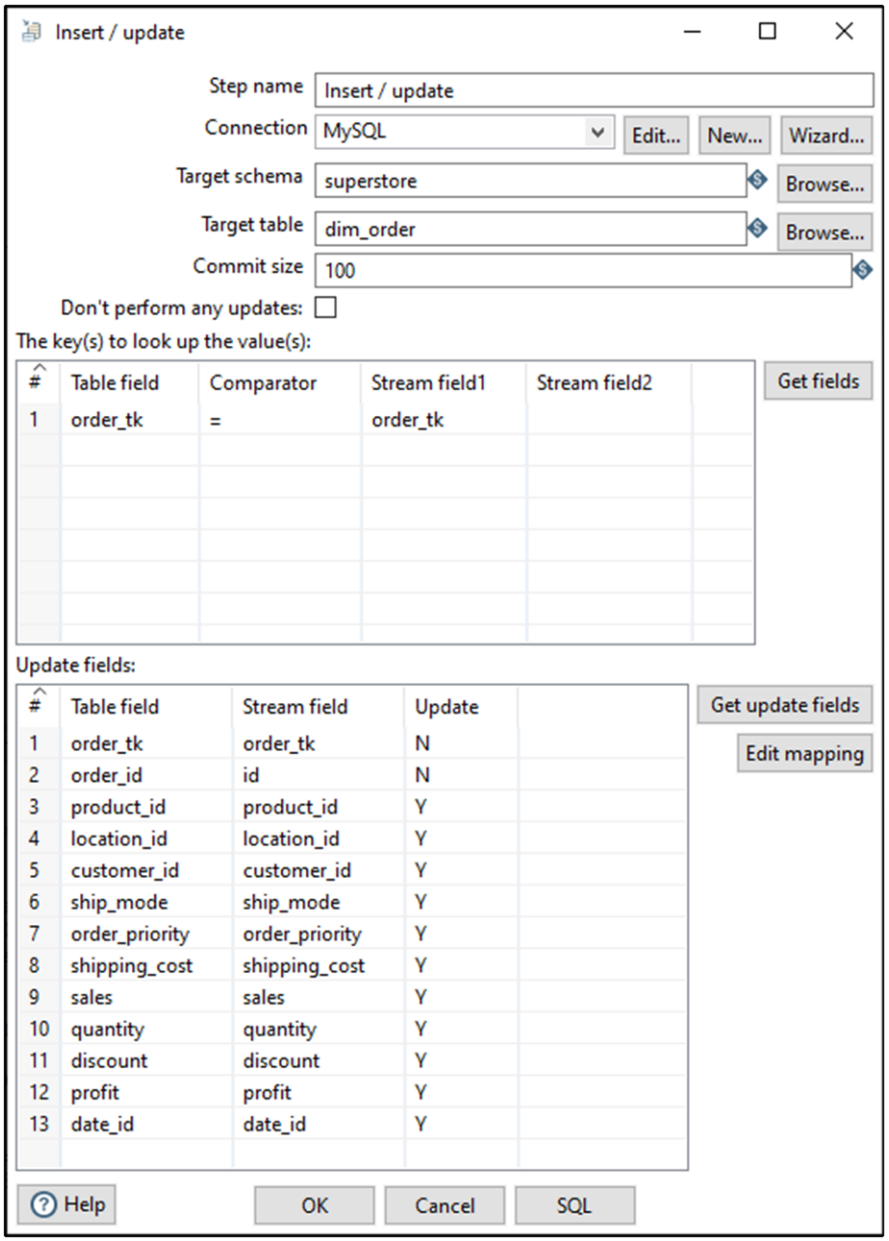
Inkrementalno punjenje tablice činjenica je jednostavnije nego kod složenih dimenzija s obzirom da tablica činjenica ne sadrži sporo mijenjajuće dimenzije. Kao i kod složenih dimenzijskih tablica, učitavamo podatke iz izvorne baze podataka i tablice činjenica kako bi detektirali promjene ili nove podatke (slika 63.). Učitane podatke spajamo naredbom “Merge Rows (diff)“ (slika 64.), dohvaćamo sve surogat ključeve pomoću naredbe “Database Lookup“ te filtriramo podatke koji nam nisu od interesa. Ako su detektirane promjene ili nove transakcije, naredbom “Insert/Update“ (slika 65.) ih ažuriramo ili unosimo nove transakcije u tablicu činjenica (slika 66.).



Slika 63. Prikaz transformacije za inkrementalno punjenje tablice činjenica



Slika 64. Prikaz naredbe “Merge Rows (diff)“



Slika 65. Prikaz naredbe "Insert/Update"

order_tk	order_id	product_id	location_id	customer_id	date_id	ship_mode	order_priority	shipping_cost	sales	quantity	discount	profit
51291	35001	3832	3634	803	978	Standard Class	Medium	40	150.32	1	0	57.98
51290	1016289	2465	886	47	1430	Standard Class	Medium	0.17	3.02	3	0.2	-0.6
51289	1016288	64	1781	47	1430	Standard Class	Medium	0.2	7.12	1	0	0.56
51288	1016287	2769	1781	47	1430	Second Class	Medium	0.35	26.4	3	0	12.36
51287	1016286	2262	2138	47	1430	Standard Class	Medium	0.49	3.99	1	0	0.42
51286	1016285	2262	2065	47	1430	Standard Class	Medium	0.89	13.9	2	0.2	4.52
51285	1016284	3324	2858	47	1430	Standard Class	Medium	1.32	16.74	3	0	0.66
51284	1016283	3544	501	47	1430	Standard Class	Medium	1.41	79.47	3	0	25.38
51283	1016282	2049	1558	47	1430	Standard Class	Medium	1.7	27.84	4	0	6.12
51282	1016281	2052	1558	47	1430	Standard Class	Medium	2.06	20.72	2	0.2	6.48

Slika 66. Prikaz nove transakcije u tablici činjenica

6.4.1. Primjena inkrementalnog ETL procesa nad dimenzijskim modelom

Kako bi inkrementalan ETL proces nad tablicom činjenica bio uspješan, to zahtijeva da su novi ili izmijenjeni podaci iz relacijskih tablica već učitani u odgovarajuće dimenzijske tablice. Primjenom ETL posla (engl. Job), sekvencijalno pozivamo sve transformacije potrebne kako bi se uspješno napunile sve dimenzijske tablice, a i samim time i tablica činjenica (slika 67.).

Slika 67. Prikaz sekvencijalnog ETL posla nad dimenzijskim modelom

7. OLAP (engl. Online Analytical Processing)

7.1. Uvod u OLAP alate

OLAP (engl. Online Analytical Processing) je pristup koji daje odgovore na višedimenzionalne analitičke upite u računarstvu. OLAP je dio šire kategorije poslovne inteligencije, koja također obuhvaća relacijske baze podataka, pisanje izvještaja i rudarenje podataka. Tipične primjene OLAP-a uključuju poslovno izvještavanje za prodaju, marketing, upravljačko izvješćivanje, upravljanje poslovnim procesima, budžetiranje i predviđanje, financijsko izvješćivanje i slična područja. OLAP alati omogućuju korisnicima interaktivnu analizu višedimenzionalnih podataka iz više perspektiva (Rovčanin, et al., 2012.). OLAP se sastoji od četiri osnovne analitičke operacije:

- **Selekcija i projekcija (engl. Slice and Dice)** – selekcija je čin odabira pravokutne podskupine kocke odabirom jedne vrijednosti za jednu od njenih dimenzija, što rezultira stvaranjem nove kocke s jednom manje dimenzijom. Projekcija stvara podskup kocke tako što dozvoljava analitičaru da odabere određene vrijednosti iz više dimenzija (Kimball & Ross, 2013.)
- **Roll-Up i Drill-Down** – Roll-Up operacija vrši agregaciju na kocki podataka bilo usponom uz hijerarhiju dimenzije ili smanjenjem dimenzija. Drill-Down operacija je obrnuta od Roll-Up. Navigira od manje detaljnih podataka do detaljnijih podataka. To se može ostvarili ili spuštanjem niz hijerarhiju dimenzije ili uvođenjem dodatne dimenzije (Kimball & Ross, 2013.)
- **Drill-Across** – pristupa više tablica činjenica koje su povezane zajedničkim dimenzijama, odnosno kombinira kocke koje dijele jednu ili više dimenzija (Kimball, et al., 2015.)

- **Pivot** – mijenja prostornu orijentaciju kocke, tj. rotira podatkovne osi u svrhu pregleda podataka iz različitih perspektiva (Kimball & Ross, 2013.)

7.2. Arhitektura OLAP sustava

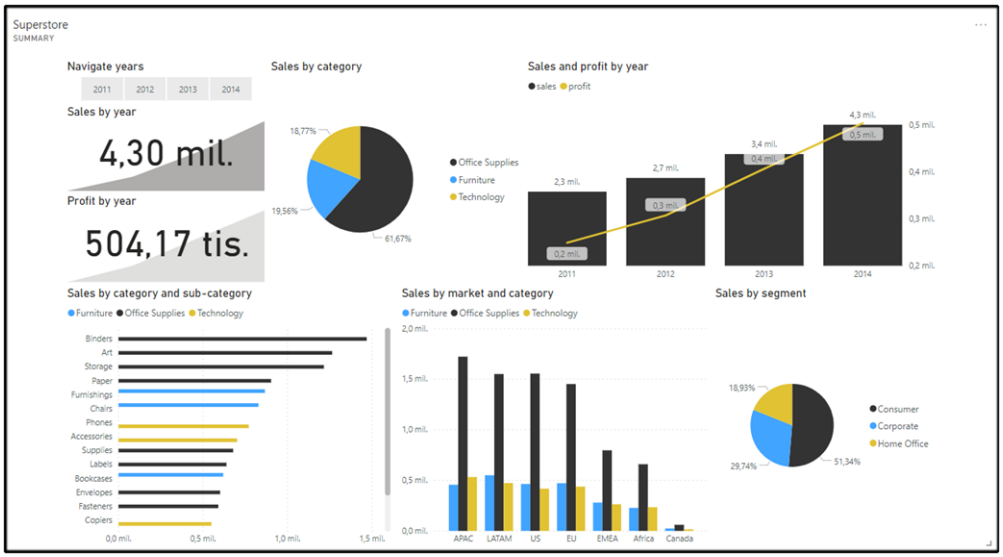
OLAP sustavi tradicionalno su kategorizirani korištenjem sljedeće taksonomije:

- **MOLAP (engl. Multidimensional OLAP)** – MOLAP je klasičan oblik OLAP-a i ponekad se naziva samo OLAP. MOLAP pohranjuje podatke u optimizirano višedimenzionalne polje, umjesto u relacijsku bazu podataka. Neke od prednosti MOLAP-a su što podržava brze performanse upita zbog optimiziranog pohranjivanja, višedimenzionalnog indeksiranja i predmemoriranja. Nedostatak MOLAP-a su što neke njegove metodologije uvode redundantnost podataka (Kimball & Ross, 2013.)
- **ROLAP (engl. Relational OLAP)** – ROLAP djeluje izravno s relacijskim bazama podataka i ne zahtijeva prethodno računanje. Osnovni podaci i dimenzijske tablice pohranjuju se kao relacijske tablice, a nove tablice se stvaraju za čuvanje agregiranih podataka. Smatra se da je ROLAP skalabilniji u rukovanju s velikim količinama podataka, posebno s modelima čije dimenzije sadrže vrlo visoku kardinalnost. Budući da se ROLAP alati oslanjaju na SQL za sve proračune, neki od nedostataka su da nisu prikladni za korištenje kada model koristi puno proračuna koji se ne prevode dobro u SQL. Primjeri takvih modela uključuju budžetiranje, financijsko izvještavanje i ostale scenarije (Kimball, et al., 2015.)
- **HOLAP (engl. Hybrid OLAP)** – HOLAP je kombinacija ROLAP i MOLAP, odnosno hibridni pristup koji omogućuje dizajnerima modela da odluče koji će se dio podataka pohraniti u ROLAP, a koji u MOLAP. Takav pristup pokušava riješiti nedostatke MOLAP-a i ROLAP-a kombinirajući mogućnosti oba pristupa (Kimball, et al., 2015.)

7.3. Vizualizacija podataka

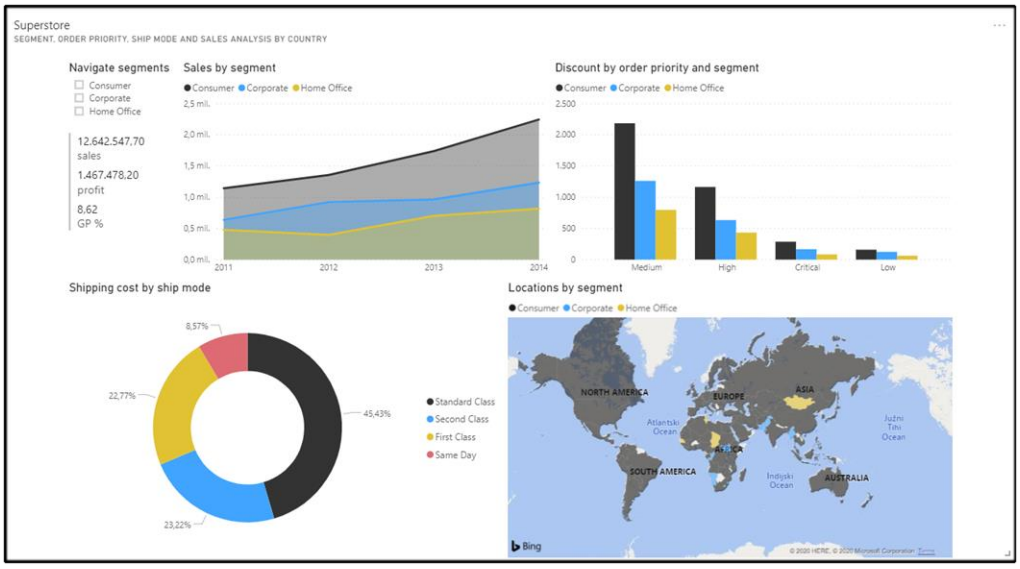
Posljednja faza izrade sustava potpore odlučivanju odnosi se na grafičko prikazivanje podataka. Korišteni alat za vizualizaciju podataka je Microsoft-ov Power BI. Prvi korak prilikom otvaranja Power BI desktop aplikacije bio je stvaranje konekcije na bazu podataka. Nakon uspješne konekcije, odabiremo željene dimenzijske tablice koje Power BI alat automatski povezuje sa odgovarajućim ključevima. Nakon provjere točnosti učitanih tablica, mogao sam započeti sa izradom kontrolne ploče. Slike veće rezolucije mogu se pronaći na poveznicama pored opisa slika.

Na prvoj stranici je prikazan sažetak poslovanja tvrtke. Prikazana je ukupna dobit i prihod kroz vremenski period u kojem je tvrtka poslovala, grafovi koji prikazuju koje kategorije i pod-kategorije proizvoda se najbolje prodaju, na kojim lokacijama se najbolje prodaju navedene kategorije i odnos prihoda i dobiti kroz vrijeme (slika 68.).



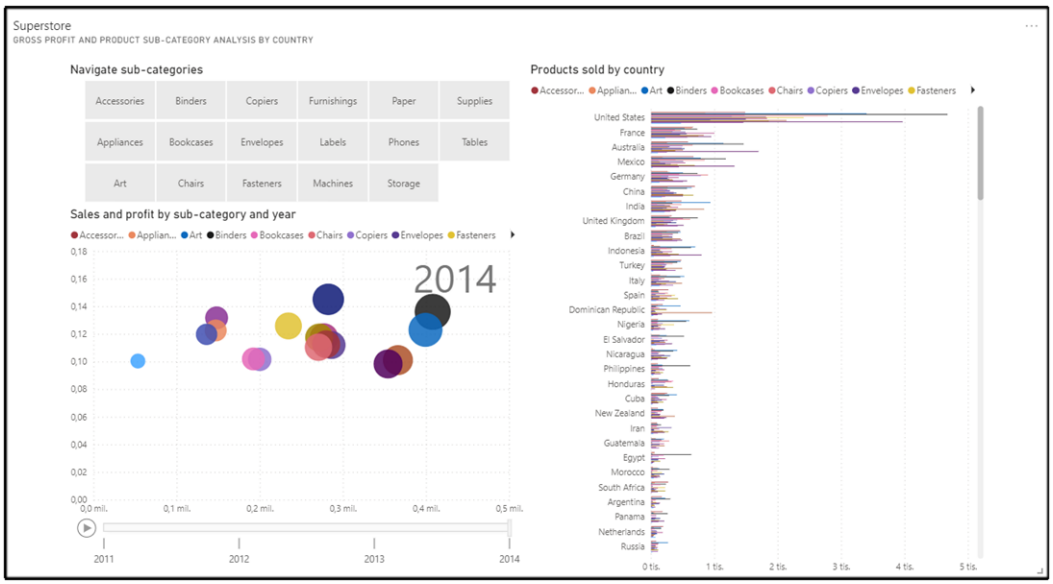
Slika 68. Prikaz sažetka poslovanja (veća rezolucija)

Na sljedećoj stranici prikazujemo koji segment je najzastupljeniji na našem tržištu kroz vrijeme (slika 69.). Također, analiziramo koji segment dobiva najveće popuste u odnosu na njihov prioritet narudžbe. Lokacije kupaca prikazane su ugrađenom mapom koja dolazi sa Power BI alatom te analiziramo koji način otpreme je najzastupljeniji u odnosu na cijenu otpreme.



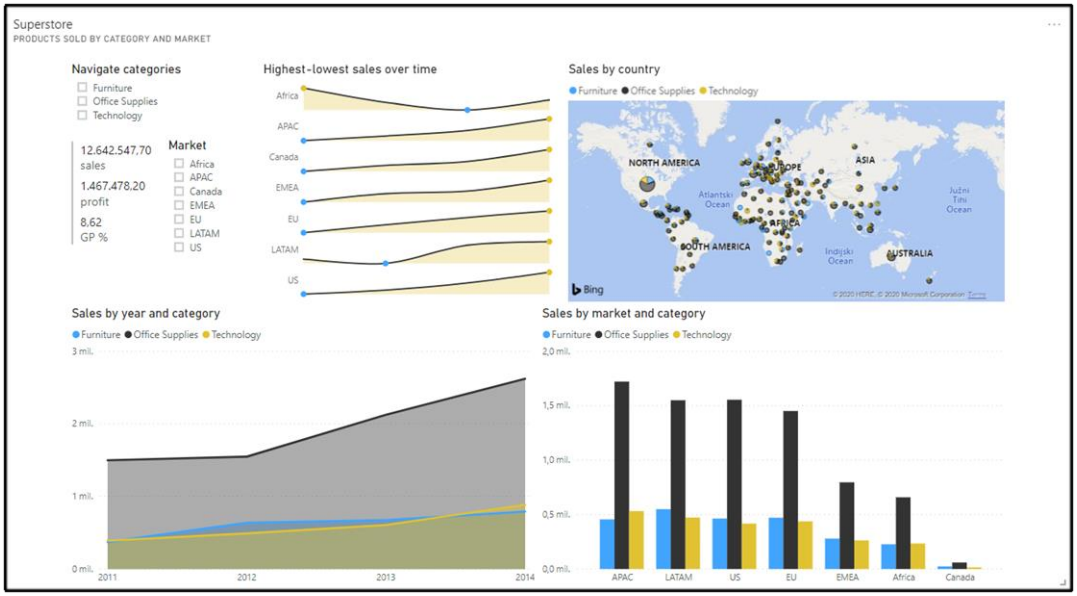
Slika 69. Prikaz analize segmenta, načina otpreme i prodaje po zemljama (veća rezolucija)

Na trećoj stranici analiziramo koje države su kupile najveću količinu proizvoda te prikazujemo pod-kategorije proizvoda kroz animaciju di horizontalna os predstavlja dobit tvrtke a usporedna os postotak profita (slika 70.). Veličina kugle je određena prihodom koju je ta pod-kategorija donijela te promatramo ponašanja zastupljenosti pod-kategorija kroz vremenski period u kojem je tvrtka poslovala.



Slika 70. Prikaz analize profita i pod-kategorija proizvoda po zemljama (veća rezolucija)

Posljednja stranica prikazuje kako se mijenja zastupljenost kategorija proizvoda kroz vrijeme (slika 71.). Uz to, postavljen je geografski pregled zastupljenosti proizvoda te graf koji prikazuje kada se dogodila najniža i najviša dobit za svaki pojedini market kroz cjelokupno vrijeme poslovanja tvrtke.



Slika 71. Prikaz analize prodanih proizvoda po marketu (veća rezolucija)

8. Apache Flink

Skladištenje podataka se sve više prebacuje na obrađivanje podataka u stvarnom vremenu, što zahtijeva sposobnost obrađivanja tokova podataka s malim kašnjenjem kako bi se podaci mogli analizirati u stvarnom (ili skoro stvarnom) vremenu. Korisnici postaju sve manje tolerantni prema kašnjenjima između stvaranja podataka do trenutka kada dođu u njihove ruke. Korisnici očekuju minute, ili čak sekunde kašnjenja podataka u skladištu podataka kako bi dobili što brže uvide, i samim time mogli brže i učinkovitije donositi poslovne odluke (Bowen, 2020.).

8.1. Uvod u Apache Flink

Apache Flink je aplikacijski okvir otvorenog koda za procesiranje streamova podataka. Jezgra Apache Flink-a je distribuirani uređaj za stream-anje tokova podataka napisan u programskim jezicima Java i Scala. Flink izvršava proizvoljne programe tokova podataka na paralelni i cjevovodni način. Flink-ov cjevovodni runtime sustav omogućuje izvršavanje programa za bulk/batch i stream obradu podataka. Nadalje, Flink-ov runtime sustav podržava izvorno izvršavanje iterativnih algoritama. Flink pruža streaming uređaj s visokom propusnošću i niskim kašnjenjem, kao i podršku za obradu događaja i upravljanje stanjima. Flink aplikacije su otporne na greške u slučaju kvara stroja i podržavaju "exactly-once" semantiku. Programi se mogu pisati u Javi, Scali, Python-u i SQL-u te se automatski kompiliraju i optimiziraju u programe tokova podataka koji se izvode u klasteru ili "cloud" okruženju (Foundation, 2020.).

8.2. Programski model i distribuirano vrijeme izvršavanja

Po izvršenju, Flink programi se preslikavaju u stream-ove tokova podataka. Svaki Flink-ov tok podataka započinje s jednim ili više izvora (unos podataka, npr. red poruka ili datotečni sustav) i završava s jednim ili više izlaza (izlaz podataka, npr. red poruka, datotečni sustav ili baza podataka). Na stream-u se može izvršiti proizvoljan broj transformacija. Ti se stream-ovi mogu organizirati kao usmjereni, aciklički graf toka podataka, što aplikaciji omogućuje grananje i spajanje tokova podataka (Xingcan, 2019).

8.3. Stanje – checkpoints, savepoints i tolerancija na pogreške

Apache Flink uključuje lagani mehanizam tolerancije na pogreške koji se temelji na distribuiranim checkpoint-ovima. Checkpoint je automatski, asinkroni snimak stanja aplikacije i položaja u izvornom stream-u. U slučaju neuspjeha, Flink program s omogućenim checkpoint-om će nakon oporavka nastaviti s obradom s posljednjeg dovršenog checkpoint-a, osiguravajući da Flink održava "exactly-once" semantiku stanja

unutar aplikacije. Checkpoint mehanizam otkriva kuke (engl. Hooks) aplikacijskom kodu za uključivanje vanjskih (eksternih) sustava u isti (npr. otvaranje i izvršavanje transakcija sa transakcijskim sustavom). Flink također uključuje savepoint mehanizam, koji zapravo predstavljaju ručno aktivirane checkpoint-ove. Korisnik može generirati savepoint, zaustaviti izvršavanje Flink programa, a zatim nastaviti program iz istog stanja aplikacije i položaja u stream-u. Savepoint-ovi omogućavaju ažuriranja Flink programa ili Flink klastera bez gubitka stanja aplikacije (Foundation, 2020.).

8.4. DataStream API

DataStream programi u Flink-u su programi koji implementiraju i vrše transformacije na tokove podataka. Neke od tih transformaciju su:

- Filtriranje
- Ažuriranje stanja
- Definiranje prozora (engl. Windows)
- Agregiranje

Tokovi podataka su inicijalno stvoreni iz različitih izvora podataka (npr. redovi poruka, datoteke, …). Rezultati se vraćaju preko Sink-a, koji mogu, na primjer, zapisati podatke u datoteke, ili na standardni izlaz (npr. terminal naredbenog retka). Flink programi se izvode u različitim kontekstima, samostalnim ili ugrađenim u druge programe. Izvršenje se može dogoditi u lokalnom JVM-u ili na nekakvim klasterima. Flink-ov DataStream API ubiti omogućuje transformacije na ograničenim ili neograničenim tokovima podataka (Xingcan, 2019).

8.4.1. Izvršno okruženje programa

Izvršno okruženje programa (engl. Execution Environment) je kontekst u kojem se izvršava program za tokove podataka. U ovom završnom radu se fokusiramo na tokove podataka, stoga moramo u svakom programu definirati “StreamExecutionEnvironment“. Izvršno okruženje pruža metode za kontrolu izvršenja programa (npr. postavljanje paralelizma, postavljanje parametara tolerancije na pogreške ili kontrolne točke) i za interakciju s vanjskim svijetom (npr. pristupanje podacima) (Foundation, 2020.).

8.5. Izvori podataka

Izvori su mjesta iz kojih programi čitaju ulazne podatke. Izvore možemo dodati u program pomoću funkcije “StreamExecutionEnvironment.addSource(sourceFunction)“. Flink dolazi s brojnim unaprijed implementiranim izvornim funkcijama, ali uvijek možemo pisati vlastite prilagođene izvore implementacijom “SourceFunction“ za

neparalelne izvore ili implementacijom “ParallelSourceFunction“ sučelja ili proširenjem “RichParallelSourceFunction“ za paralelne izvore (Foundation, 2020.).

Postoji nekoliko unaprijed definiranih izvora podataka dostupnih iz “StreamExecutionEnvironment“ funkcije:

- Bazirani na datotekama
  - readTextFile(path) – čita tekstualne datoteke i vraća ih kao String-ove
  - readFile(fileInputFormat, path) – čita (jednom) datoteke kako je diktirano određenim formatom unosa datoteke
- Socket-based
  - socketTextStream – čita podatke iz Socket-a. Elementi mogu biti razdvojeni delimiterom
- Bazirani na kolekcijama
  - fromCollection(Collection) – stvara DataStream objekt gdje svi elementi unutar kolekcije moraju biti istoga tipa podataka
  - fromElements(T …) – stvara DataStream objekt od dane sekvence objekata gdje svi elementi unutar sekvence moraju biti istoga tipa podataka
  - fromParallelCollection(SplittableIterator, Class) – stvara DataStream objekt iz iteratora, paralelno. Klasa specificira tip podatka elemenata koje iterator vraća
  - generateSequence(from, to) – generira sekvencu brojeva u danom intervalu, paralelno
- Korisnički definirani
  - addSource – prilaže novu funkcija izvora

8.6. Sink-ovi podataka

Sink-ovi podataka troše tokove podataka i prosljeđuju ih u datoteke, Socket-e, vanjske sustave ili ih ispisuju (Foundation, 2020.). Flink dolazi s raznim ugrađenim izlaznim formatima koji su enkapsulirani u pozadini operacija tokova podataka:

- writeAsText() – elemente sprema linijski kao String-ove. String-ovi su stvoreni pozivanjem “toString()“ metode na svakom elementu
- writeAsCsv – elemente sprema u CSV datoteku kao String-ove. . String-ovi su stvoreni pozivanjem “toString()“ metode na svakom elementu
- print() – poziva “toString()“ metodu na svakom elementu na standardnom

izlaznom toku

- writeUsingOutputFormat – metoda i bazna klasa za korisnički definirane datotečne izlaze
- writeToSocket – sprema elemente u Socket
- addSink – poziva korisnički definiranu Sink funkciju. Flink dolazi u paketu s konektorima na druge sustave koji su implementiranu kao Sink funkcije

8.7. Primjer programa

Sljedeći program je cjelovit, radni primjer aplikacije za brojanje riječi koja broji riječi koje dolaze iz web Socket-a u intervalu od pet sekundi. Web Socket pokrećemo naredbom "nc -lk 9999" pomoću Netcat-a (slika 73.). Netcat je uslužni program za umrežavanje računala za čitanje i pisanje na mrežne veze pomoću TCP ili UDP protokola. Program se sastoji od "StreamExecutionEnvironment" objekta koji priprema izvršno okruženje programa i toka podataka koji predstavlja n-torku String-a i cijelog broja, gdje String predstavlja unesenu riječ, a cijeli broj količinu puta koji se pojavila ta riječ. Unutar objekta toka podataka spremamo izvršno okruženje programa na koji primjenjujemo niz metoda kako bi postigli željeno ponašanje. Metoda "socketTextStream" se povezuje na tok u koji unosimo riječi, "flatMap" razdvaja svaku liniju u riječi, "keyBy" grupira po unesenom ključu, "timeWindow" određuje vremenski interval u kojem možemo unijeti dvije ili više istih riječi kako bi se povećala količina puta koji se ta riječ pojavila i "sum" povećava brojač za svaku unesenu riječ. Na objektu toka podataka pozivamo metodu "print" za ispis navedenih n-torki te pozivamo metodu "execute" na izvršnog okruženju programa kako bi se pokrenuo program.

```
1.  package wc;
2.  import org.apache.flink.api.common.functions.FlatMapFunction;
3.  import org.apache.flink.api.java.tuple.Tuple2;
4.  import org.apache.flink.streaming.api.datastream.DataStream;
5.  import org.apache.flink.streaming.api.environment.StreamExecutionEnvironment;
6.  import org.apache.flink.streaming.api.windowing.time.Time;
7.  import org.apache.flink.util.Collector;
8.  public class WordCount {
9.      public static void main(String[] args) throws Exception {
10.         StreamExecutionEnvironment env = StreamExecutionEnvironment.getExecutionEnvi
    ronment();
11.         DataStream<Tuple2<String, Integer>> dataStream = env
12.                 .socketTextStream("localhost", 9999)
13.                 .flatMap(new Splitter())
14.                 .keyBy(0)
15.                 .timeWindow(Time.seconds(5))
16.                 .sum(1);
17.         dataStream.print();
18.         env.execute("WordCount");
19.     }
```

```
20.     public static class Splitter implements FlatMapFunction<String, Tuple2<String, I
    nteger>> {
21.         public void flatMap(String sentence, Collector<Tuple2<String, Integer>> out)
     throws Exception {
22.             for (String word: sentence.split(" ")) {
23.                 out.collect(new Tuple2<String, Integer>(word, 1));
24.             }
25.         }
26.     }
27. }
```

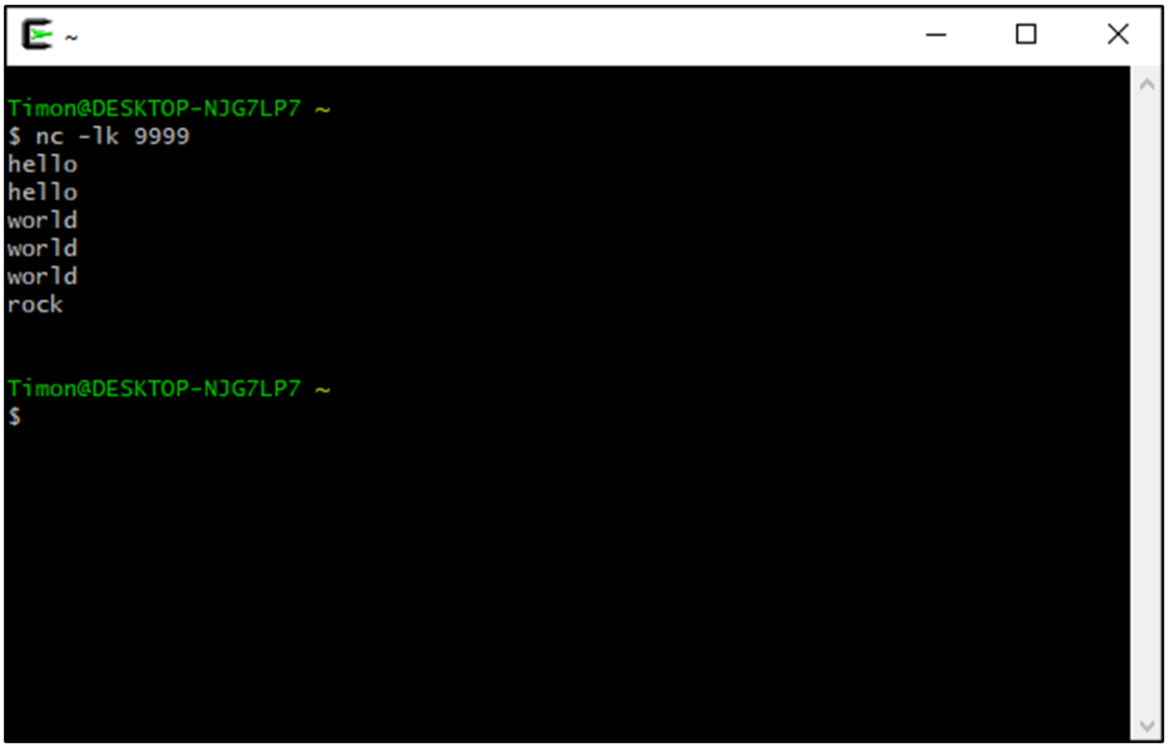
Slika 72. prikazuje pokretanje klastera naredbom "./flink-1.4.2/bin/start-cluster.sh" te pokretanje Apache Flink posla pozivanjem "wc.jar" aplikacije.

```
/cygdrive/c
Timon@DESKTOP-NJG7LP7 /cygdrive/c
$ ./flink-1.4.2/bin/start-cluster.sh
Starting cluster.
Starting jobmanager daemon on host DESKTOP-NJG7LP7.
Starting taskmanager daemon on host DESKTOP-NJG7LP7.

Timon@DESKTOP-NJG7LP7 /cygdrive/c
$ ./flink-1.4.2/bin/flink run ./Users/Timon/eclipse-workspace/wc.jar
Cluster configuration: Standalone cluster with JobManager at localhost/127.0.0.1:6123
Using address localhost:6123 to connect to JobManager.
JobManager web interface address http://localhost:8081
Starting execution of program
Submitting job with JobID: af27a3a3d948fdb354c1db06658b740c. Waiting for job completion.
Connected to JobManager at Actor[akka.tcp://flink@localhost:6123/user/jobmanager#-1670963015] with leade
r session id 00000000-0000-0000-0000-000000000000.
06/27/2020 14:53:16     Job execution switched to status RUNNING.
06/27/2020 14:53:16     Source: Socket Stream -> Flat Map(1/1) switched to SCHEDULED
06/27/2020 14:53:16     TriggerWindow(TumblingProcessingTimeWindows(5000), ReducingStateDescriptor{seria
lizer=org.apache.flink.api.java.typeutils.runtime.TupleSerializer@7295e0f7, reduceFunction=org.apache.fl
ink.streaming.api.functions.aggregation.SumAggregator@34b7ac2f}, ProcessingTimeTrigger(), WindowedStream
.reduce(WindowedStream.java:241)) -> Sink: Unnamed(1/1) switched to SCHEDULED
06/27/2020 14:53:16     Source: Socket Stream -> Flat Map(1/1) switched to DEPLOYING
06/27/2020 14:53:16     TriggerWindow(TumblingProcessingTimeWindows(5000), ReducingStateDescriptor{seria
lizer=org.apache.flink.api.java.typeutils.runtime.TupleSerializer@7295e0f7, reduceFunction=org.apache.fl
ink.streaming.api.functions.aggregation.SumAggregator@34b7ac2f}, ProcessingTimeTrigger(), WindowedStream
.reduce(WindowedStream.java:241)) -> Sink: Unnamed(1/1) switched to DEPLOYING
06/27/2020 14:53:16     TriggerWindow(TumblingProcessingTimeWindows(5000), ReducingStateDescriptor{seria
lizer=org.apache.flink.api.java.typeutils.runtime.TupleSerializer@7295e0f7, reduceFunction=org.apache.fl
ink.streaming.api.functions.aggregation.SumAggregator@34b7ac2f}, ProcessingTimeTrigger(), WindowedStream
.reduce(WindowedStream.java:241)) -> Sink: Unnamed(1/1) switched to RUNNING
06/27/2020 14:53:16     Source: Socket Stream -> Flat Map(1/1) switched to RUNNING
06/27/2020 14:53:26     Source: Socket Stream -> Flat Map(1/1) switched to FINISHED
06/27/2020 14:53:26     TriggerWindow(TumblingProcessingTimeWindows(5000), ReducingStateDescriptor{seria
lizer=org.apache.flink.api.java.typeutils.runtime.TupleSerializer@7295e0f7, reduceFunction=org.apache.fl
ink.streaming.api.functions.aggregation.SumAggregator@34b7ac2f}, ProcessingTimeTrigger(), WindowedStream
.reduce(WindowedStream.java:241)) -> Sink: Unnamed(1/1) switched to FINISHED
06/27/2020 14:53:26     Job execution switched to status FINISHED.
Program execution finished
Job with JobID af27a3a3d948fdb354c1db06658b740c has finished.
Job Runtime: 9914 ms

Timon@DESKTOP-NJG7LP7 /cygdrive/c
$
```

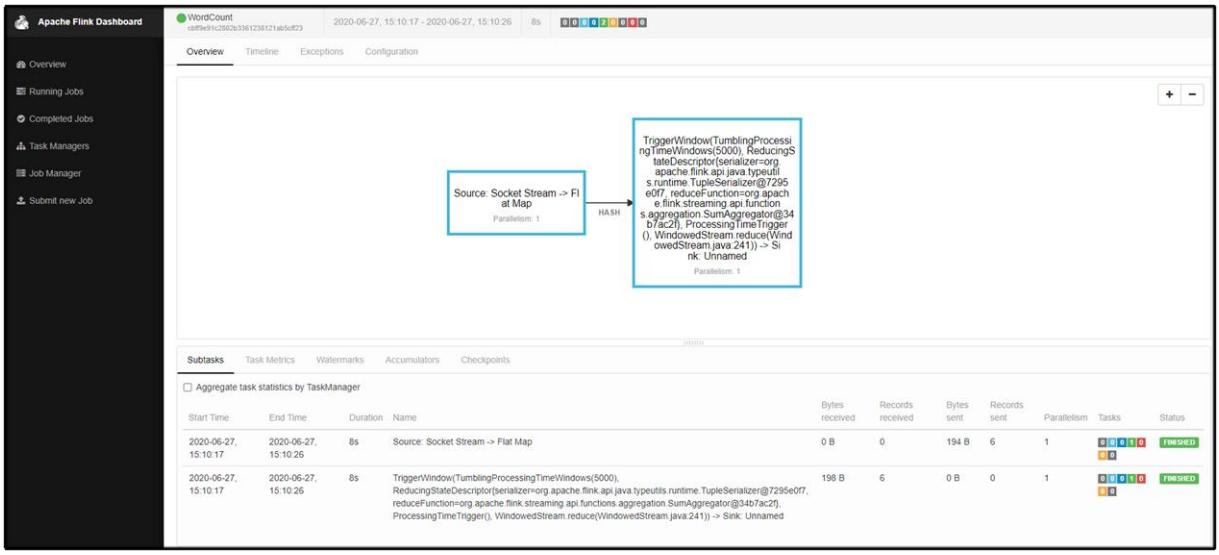
Slika 72. Prikaz pokretanja klastera i JAR aplikacije



Slika 73. Prikaz pokretanja web Socket-a i unosa riječi (Xingcan, 2019)

Vremenska crta izvršenog Apache Flink posla prikazuje vremenski koliko je dugo trebalo da se posao izvrši te kako su se funkcije pozivale kroz vrijeme (slika 74.), dok na primarnoj stranici izvršenog posla možemo vidjeti sažetak izvršenja posla koji prikazuje utroške resursa, grub prikaz pozvanih funkcija i početno, završno i vrijeme trajanja izvršenja posla (slika 75.). Slika 76. prikazuje rezultate izvršenog Apache Flink posla.

Slika 74. Prikaz vremenske crte izvršenog posla



Slika 75. Prikaz procesa izvršenog posla

Slika 76. Prikaz rezultata izvršenog posla

9. Zaključak

Sa ogromnom količinom podataka koje tvrtke danas posjeduju, samo one sa alatima za pravilan pristup i manipuliranje tim podacima moći će u potpunosti iskoristiti i vidjeti prednosti svog poslovanja. Tvrtke koje posjeduju dobro razvijene sustave poslovne inteligencije, moći će povećati svoju produktivnost, ostati konkurentni na tržištu i omogućiti cijeloj tvrtki donošenje bržih i pametnijih poslovnih odluka na temelju prikupljenih podataka.

Kako se poslovna inteligencija kontinuirano razvija u skladu s poslovnim potrebama i tehnologijom, potrebno je redovito identificirati trenutne trendove kako bi korisnici bili u toku s inovacijama.

S obzirom da tvrtke teže k tome da budu vođene podacima, povećavat će se napori za suradnju i razmjenu podataka. Vizualizacija podataka bit će još važnija za zajednički rad timova i odjela. Stoga, analizom ključnih podataka pomoću poslovne inteligencije uklanjaju se nagađanja iz poslovanja kvantificiranjem rješenja složenih problema, a ne oslanjanjem na nejasne dojmove ili instinkte.

Stvoreno skladište podataka predstavlja središnji repozitorij integriranih povijesnih podataka tvrtke koji se koriste za izradu analitičkih izvještaja kako bi pomogli menadžerima kod donošenja važnih poslovnih odluka, i samim time cijeloj tvrtki. Specifično, implementirano skladište podataka omogućuje svojim korisnicima jednostavan pregled prihoda i količine prodanih proizvoda, analizu kategorija proizvoda koji se najbolje prodaju, analizu geografske dimenzije, te daje uvid u segmentirano tržište, odnosno prikazuje tko su najčešći korisnici tvrtke i kako se njihovo ponašanje mijenja kroz vrijeme. Također, podržava inkrementalno punjenje skladišta podataka što omogućuje kontinuirano evidentiranje novonastalih promjena u poslovanju kako bi donositelji odluka mogli što brže i efikasnije donositi točnije i bolje poslovne odluke.

Apache Flink je izvrstan izbor za razvoj i pokretanje mnogih različitih vrsta aplikacija zbog svog opsežnog skupa značajki. Flink-ove značajke uključuju podršku za stream i batch procesiranje podataka, sofisticirano upravljanje stanjima i “exactly-once“ semantiku. Štoviše, Flink se može implementirati na različite davatelje resursa poput YARN-a, Apache Mesos-a i Kubernetes-a, ali i kao samostalni klaster na hardveru. Konfiguriran za visoku dostupnost, Flink nema središnju točku kvara. Dokazano je da Flink doseže tisuće jezgara i terabajta stanja aplikacije, omogućuje visoku propusnost i nisko kašnjenje te pokretanje nekih od najzahtjevnijih aplikacija za obradu podataka.

10. Literatura

1. Anon., 2012.. *sqlbicro.* [Mrežno]
   Dostupno na: https://sqlbicro.wordpress.com/2012/10/15/razlika-izmedu-operacijskih-i-analtickih-sustava/
   [Pokušaj pristupa 22 Svibanj 2020.].
2. Anon., n.d. *javatpoint.* [Mrežno]
   Dostupno na: https://www.javatpoint.com/data-warehouse-what-is-star-schema
   [Pokušaj pristupa 8 Svibanj 2020.].
3. Coronel, C. & Morris, S., 2016.. *Database systems: design, implementation, & management.* s.l.:an.
4. Ćurko, K., 2001.. *Hrčak.* [Mrežno]
   Dostupno na: https://hrcak.srce.hr/
   [Pokušaj pristupa 26 Svibanj 2020.].
5. Dusa, S., 2017.. *ezdatamunch.* [Mrežno]
   Dostupno na: https://ezdatamunch.com/implementation-incremental-load-qlikview-benefits/
   [Pokušaj pristupa 2 Lipanj 2020.].
6. Foundation, A. S., 2020.. *Apache Flink.* [Mrežno]
   Dostupno na: https://ci.apache.org/projects/flink/flink-docs-stable/dev/datastream_api.html
   [Pokušaj pristupa 14 Lipanj 2020.].
7. Inmon, W. H., 2005.. *Building the data warehouse.* 3. ur. s.l.:John Wiley & Sons, Inc..
8. Kimball, R. & Ross, M., 2013.. *The data warehouse toolkit: The definitive guide to dimensional modeling.* 3. ed. Indianopolis(Indiana): John Wiley & Sonc, Inc..
9. Kimball, R. i dr., 2015.. *Kimball Group.* [Mrežno]
   Dostupno na: https://www.kimballgroup.com/
   [Pokušaj pristupa 28 Svibanj 2020.].
10. Kimball, R. i dr., 2008.. *The data warehouse lifecycle toolkit.* 2. ur. s.l.:John Wiley & Sons, Inc..
11. Mekterović, I. & Brkić, L., 2017.. *FER.* [Mrežno]
    Dostupno na: https://www.fer.unizg.hr/_download/repository/SKRIPTA_-_Skladista_podataka_i_poslovna_inteligencija.pdf
    [Pokušaj pristupa 28 Svibanj 2020.].

12. Rovčanin, A., Mataradžija¸, A. & Mataradžija, A., 2012.. *Upravljanje znanjem kroz primjenu alata poslovne inteligencije.* Mladenovac, an.
13. Shimko, J., n.d. *medium.* [Mrežno]
    Available at: https://medium.com/@jennifershimko14/a-beginners-introduction-to-the-etl-process-7f5beb5c24fe
    [Pokušaj pristupa 14 Svibanj 2020.].
14. Xingcan, C., 2019. *alibabacloud.* [Mrežno]
    Available at: https://www.alibabacloud.com/blog/basic-apache-flink-tutorial-datastream-api-programming_595685
    [Pokušaj pristupa 14 Lipanj 2020].
15. Bowen, L., 2020.. *Apache Flink.* [Mrežno]
    Available at: https://flink.apache.org/features/2020/03/27/flink-for-data-warehouse.html
    [Pokušaj pristupa 7 Srpanj 2020.].