



**Matematički fakultet
Univerzitet u Beogradu**

**Tehnologije poslovne inteligencije i
Cognos alati: modul ljudskih resursa
poslovnog informacionog sistema**

Master rad

Mentor:
Prof. dr Gordana Pavlović-Lažetić

Autor:
Nenad Pribićević

Beograd, 2010.

Sadržaj

I Tehnologije skladištenja podataka.....	4
1. Ciljevi skladištenja podataka.....	4
2. Osnovne komponente i funkcionisanje skladišta podataka.....	4
3. <i>IBM Cognos 8 BI</i> platforma.....	6
4. Dimenziono modelovanje.....	8
4.1 Osnovne komponente dimenzionog modela.....	8
- tabele fakata.....	8
- tabele dimenzija.....	9
- datumska dimenzija.....	10
- zvezdasta shema.....	11
- besfaktne tabele fakata.....	11
- degenerisane dimenzije.....	13
- proširivost dimenzionog modela.....	13
- dimenziona normalizacija i denormalizacija tabela fakata.....	14
- nedeskriptivni ključevi.....	14
- poluaditivna fakta.....	15
- sporo promenljive dimenzije.....	15
- integrisanje lanca vrednosti.....	16
4.2 Proces dimnezionog projektovanja.....	17
5. <i>ETL</i> procesi.....	18
5.1 Uvod.....	18
5.2 Ekstrakcija podataka.....	19
5.3 Transformacija podataka.....	22
5.4 Učitavanje podataka.....	24
5.5 Izbor arhitekture <i>ETL</i> sistema.....	26
II <i>IBM Cognos 8</i>	27
1. Arhitektura <i>Cognos 8</i> platforme.....	27
- <i>IBM Cognos 8</i> korisnički interfejs.....	27
- Nivo I: Veb server.....	30
- Nivo 2: <i>IBM Cognos 8</i> server.....	30
- Nivo 3: Podaci.....	32

2. <i>Windows</i> orijentisane komponente <i>Cognos 8</i> platforme.....	32
2.1 <i>Data Manager</i> - alat za kreiranje skladišta podataka.....	33
2.2 <i>Framework Manager</i> – alat za modelovanje podataka.....	34
2.3 <i>Metric Designer</i> – alat za kreiranje karti rezultata.....	36
III Modul ljudskih resursa poslovnog informacionog sistema.....	38
Zaključak.....	50
Literatura.....	51

I Tehnologije skladištenja podataka

Poslovna inteligencija (engl. *Business Intelligence*, skraćeno BI) predstavlja skup tehnologija koje se koriste za upravljanje performansama i podršku odlučivanju u nekoj organizaciji. Ona omogućava donosiocima odluka na svim nivoima da budu blagovremeno i kvalitetno obavješteni kako o performansama poslovanja tako i o različitim poslovnim podacima koji im pomažu da kvalitetno i efikasno donose odluke o daljem poslovanju preduzeća.

Skladište podataka (engl. *Data Warehouse*) je osnov poslovne inteligencije. Predstavlja sistem koji ekstrahuje, prečišćava, standardizuje i učitava podatke iz postojećih produkcionih sistema preduzeća u dimenzione sheme i podržava, pa i implementira korisničke alate za postavljanje upita nad podacima kojima je cilj da omoguće krajnjim korisnicima (koji po struci ne moraju biti informatičari) lako i efikasno snalaženje i pretraživanje velikih količina podataka.

1. Ciljevi skladištenja podataka

Tradicionalne relacije baze podataka, kao i različiti sistemi za upravljanje poslovanjem preduzeća (engl. *Enterprise Resource Planning*, skraćeno ERP) predstavljaju neophodan alat za poslovanje jednog preduzeća, ali ne i dovoljan. Relacione baze su pogodne za masovno unošenje podataka, jer su zasnovane na normalizovanom modelu podataka u kome je otklonjena redundantnost. Međutim, za krajnjeg korisnika je važno da podacima na odgovarajući način pristupi, da ih pročita i da ih protumači. Takođe krajnji korisnik želi brzo i efikasno da dođe do odgovora na pitanja koja ga zanimaju, pa čak i da „u letu” postavlja upite na osnovu odgovora koje je dobio od prethodno postavljenih upita. Međutim, performanse tradicionalnih relacionih baza su veoma loše kada je izvršavanje složenih upita u pitanju. Takođe, korisnik nema mogućnosti da na lak i brz način dinamički postavlja upite već mora da poznaje složene upitne jezike što mu znatno otežava i usporava rad. Otuda se pojavila potreba za razvijanjem tehnologija koje će omogućiti krajnjem korisniku da na lak i brz način pretražuje podatke i dolazi do odgovora koji ga zanimaju. Na engleskom se ovakvo pretraživanje obeležava terminom „*slice and dice*” koji bukvalno znači sečenje na kriške i kockice, a odnosi se na rastavljanje informacije u manje delove da bi se bolje ili lakše razumela [17]. Ovo je ujedno i jedan od osnovnih ciljeva skladištenja podataka koji se smatra ostvarenim kada su podaci krajnjem korisniku razumljivi, čitljivi, smisleni; on podrazumeva da su informacije predstavljene kompletno i sistematično bez obzira na to iz kog su izvora potekle, npr. iz relacionih baza, excel tabela ili magnetnih traka *mainframe* računara. Pored dostupnosti informacija, od skladišta podataka se očekuje konzistentnost u predstavljanju informacija bez obzira na to iz kog su poslovnog procesa ili produkcionog sistema potekle.

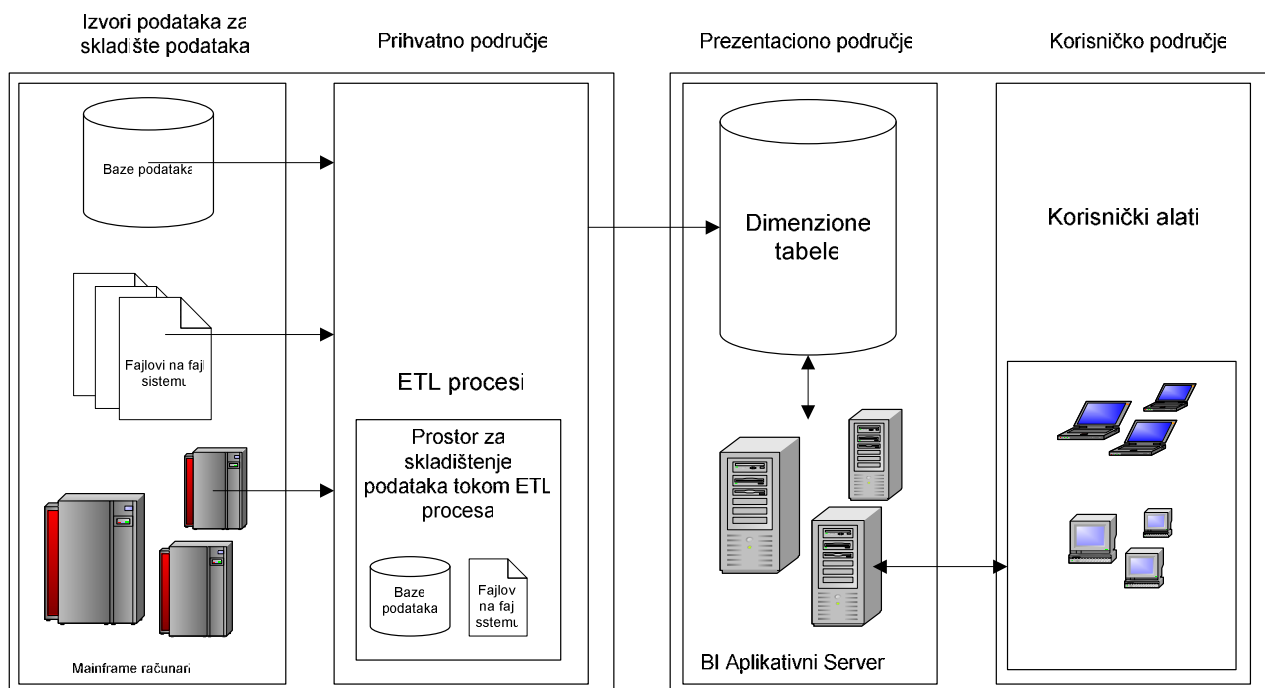
2. Osnovne komponente i funkcionisanje skladišta podataka

Osnovne komponente skladišta podataka [10] (slika 1) su:

1. produkcionni sistemi (engl. *Operational Source Systems*);
2. prihvatno područje (engl. *Data Staging Area*);
3. prezentaciono područje (engl. *Data Presentation Area*);
4. korisničko područje sa alatima za pristup podacima (engl. *Data Access Tools*).

Gore navedene komponente predstavljaju komponente skladišta podataka u širem smislu tj. predstavljaju sve komponente koje učestvuju u sistemu poslovne inteligencije. U užem smislu, skladište podataka obuhvata samo komponente 2 i 3. Ovde pod terminom „skladište podataka”

podrazumevamo sve četiri navedene komponente, tj. sve komponente koje učestvuju u procesu kreiranja i eksploatacije skladišta podataka.



Slika 1: Komponente skladišta podataka

Produkcioni sistemi su postojeći sistemi u kompaniji i predstavljaju izvore podataka za skladište podataka. Takođe ćemo ih nazivati i izvornim sistemima podataka.

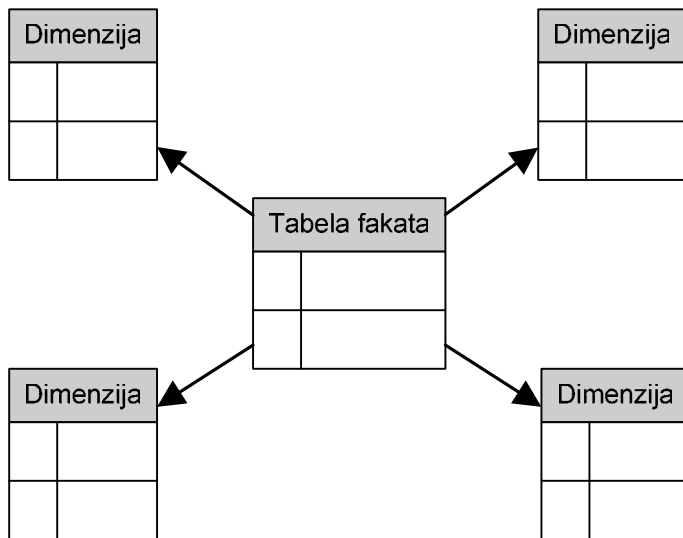
Prihvatno područje je prostor u koji se učitavaju podaci iz operacionih sistema i nad kojima se dalje vrše određene transformacije. Tako transformisani i adekvatno sređeni podaci se učitavaju u prezentaciono područje. Podaci iz prihvatnog područja su nedostupni krajnjim korisnicima. Važno je zabraniti pristup krajnjim korisnicima ovom području iz više razloga: pre svega zato što podaci u ovom području nisu obezbeđeni odgovarajućim sigurnosnim pravilima, nisu prilagođeni i optimizovani za korisničke upite, a takođe se za njih ne može garantovati konzistentnost. Zadovoljavanje ovih kriterijuma je obaveza prezentacionog područja. Transformacije koje se vrše na ovom nivou nazivaju se ETL procesi (ETL skraćeno od *Extract-Transform-Load*). Iako naziv aludira na to da se ETL sastoji od tri koraka, zapravo ETL se sastoji od četiri koraka i to:

1. **Ekstrakcija podataka** (engl. *Extracting*). Ekstrakcija je korak u kome se podaci iz izvornih sistema upisuju u odgovarajuće strukture pogodne za dalju obradu.
2. **Čišćenje podataka** (engl. *Cleaning*). U većini slučajeva kvalitet podataka koji je dovoljan za izvorne operacione sisteme ne mora biti dovoljan i za skladište podataka. Čišćenje podataka podrazumeva transformaciju podataka dobijenih ekstrakcijom u oblik pogodan za primenu u skladištu podataka. To može da podrazumeva proveru da li su vrednosti validne (u odgovarajućem opsegu), da li su podaci konzistentni, da li postoje duplirani podaci, da li su zadovoljena različita poslovna pravila, i sl. Obično se ekstrahovani podaci nakon čišćenja odbacuju, a prečišćeni podaci se čuvaju za dalju obradu.
3. **Ujednačavanje podataka** (engl. *Conforming*). Ako izvorni podaci pristižu iz više različitih izvora, zadatak ovog koraka je da podatke ujednači, tj. da se ne dozvoli da se

npr. isti podaci obeležavaju različitim nazivima. Ovaj korak predstavlja pre svega standardizaciju domena i mera.

4. **Učitavanje podataka** (engl. *Delivering*). Poslednji korak ETL procesa je učitavanje i strukturiranje prethodno pripremljenih podataka u pogodne šeme optimizovane za korišćenje od strane krajnjih korisnika sistema.

Prezentaciono područje predstavlja podatke složene u odgovarajuće strukture pogodne za korišćenje od strane krajnjih korisnika koji im pristupaju preko određenih namenskih alata. Struktura podataka je zasnovana na dimenzionom modelu. Kada se prezentaciono područje implementira u relacionim bazama podataka dobijena struktura dimenziono modelovanih tabela se naziva zvezdasta struktura podataka (engl. *Star Schema*). Jedna takva struktura prikazana je na slici 2 (deteljnije objašnjenje je dato u sledećem odeljku). Ako se prezentacioni sloj implementira u multidimenzionim bazama podataka ili OLAP (skraćeno od *On Line Analytical Processing*) tehnologiji, tada se podaci smeštaju u hiperkocke ili samo kocke (engl. *Cube*).



Slika 2: Zvezdasta shema

Korisničko područje predstavlja skup analitičkih alata koji koristi krajnji korisnik za pristup podacima sa prezentacionog područja.

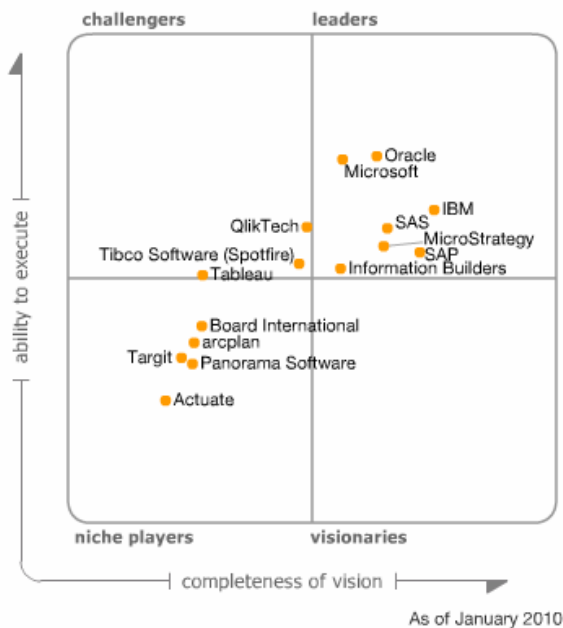
3. IBM Cognos 8 BI platforma

IBM Cognos 8 predstavlja platformu, tj. skup međusobno integrisanih alata koji pokrivaju većinu korisničkih aktivnosti iz domena poslovne inteligencije. Na slici 3, prikazan je Gartnerov¹ magični kvadrant za platforme poslovne inteligencije.

Magični kvadrant² je alat pomoću koga možemo steći uvid u pozicije proizvođača softvera na tržištu. Osnovne ose magičnog kvadranta su označene sa „*Ability to execute*” i „*Completness of vision*”. Prva označava sposobnost i uspeh proizvođača u implementiranju sopstvenih proizvoda na tržištu. Druga osa predstavlja meru shvatanja tržišta od strane proizvođača i njihovu sposobnost da ponude tržištu adekvatno i inovativno rešenje.

¹ Gartner, Inc. je vodeća kompanija za istraživanje IT tržišta i konsalting, <http://www.gartner.com>

² Videti: http://www.gartner.com/DisplayDocument?doc_cd=131166



Slika 3: Gartnerov magični kvadrant

Magični kvadrant je podeljen na četiri kvadrata koja opisuju proizvođače:

- *Challengers* – proizvođači koji su trenutno dobri ili imaju veliki udeo na tržištu, ali ne shvataju dobro tržišne trendove;
- *Niche Players* – proizvođači koji spadaju u ovaj kvadrat su uspešni u malom segmentu tržišta, ali nisu u stanju da ponude kompetitivna rešenja kojima bi mogli da prevaziđu konkurenciju;
- *Visionaries* – proizvođači koji dobro shvataju zahteve tržišta, ali još uvek nisu ponudili adekvatan odgovor;
- *Leaders* – proizvođači koji nude adekvatne odgovore na zahteve tržišta danas, i dobro shvataju njegovo kretanje i pripremljeni su da ponude adekvatne odgovore i u budućnosti.

Iz kvadranta se vidi da je IBM-ova platforma jedna od vodećih platformi u BI domenu. Gartner ističe, kao osnovnu prednost IBM platforme nad konkurencijom, odličnu integraciju među alatima. Takođe, Gartner navodi da je robusnost, kao i mogućnosti proširenja platforme i integracija sa drugim sistemima kod IBM odlična, što je i jedan od osnovnih pokazatelja uspešnosti ovakvih platformi.

Osnovne korisničke funkcionalnosti *Cognos* platforme grupisane su u komponente, a komponente su međusobno integrisane, tako da je moguće izveštaj generisan jednom komponentom otvoriti u drugoj i tome slično. Početna tačka pristupa funkcionalnostima *Cognos*-a je *Cognos Connection* – mrežni portal *Cognos*-a 8. Preko portala, uz odgovarajuća prava pristupa, korisnici mogu pristupiti alatima za pronalaženje, organizovanje, objavljivanje i pregled sadržaja poslovne inteligencije. *Cognos Connection* takođe koriste i administratori za pristup administratorskim alatima *Cognos*-a. Pregled korisničkih alata *Cognos*-a dat je u tabeli 1. Detaljnije objašnjenje ovih alata biće izloženo u drugom delu rada.

<i>Query Studio</i>	Ad hoc upiti
<i>Analysis Studio</i>	Analiziranje podataka
<i>Report Studio</i>	Organizovano izveštavanje
<i>Event Studio</i>	Upravljanje događajima i alarmima

Tabela 1: Pregled *Cognos* korisničkih alata

4. Dimenziono modelovanje

U ovom radu će biti izložena metodologija izgradnje skladišta podataka bazirana na paradigmi Ralfa Kimbala [6] (engl. *Ralph Kimball*). Paradigma se zasniva na modelu podataka koji se naziva dimenzioni model. Dimenzioni model koristimo da bismo složili podatke u odgovarajuće strukture pogodne za postavljanje složenih upita od strane krajnjih korisnika.

Normalizovani model, iako pogodan za masovno unošenje podataka, ne pogoduje ciljevima postavljenim za skladište podataka. Podaci koji su u ovom modelu smešteni u trećoj normalnoj formi čine kompleksnu mrežu koja je teška za shvatanje i ne pogoduje krajnjem korisniku. Takođe, složeni upiti nad ovakvim modelom se izvršavaju suviše sporo.

Dimenzioni model sadrži iste informacije kao i normalizovani model, međutim podaci su na pogodan način spakovani kako bi bili lako razumljivi i dostupni krajnjem korisniku, kao i da bi iskazali dobre performanse i bili otporni na promene. Pošto su upiti koje će postavljati krajnji korisnik nepredvidivi, podaci moraju biti na nivou atomične granularnosti, tj. moraju biti na najnižem nivou granularnosti. U ovom modelu tabele su obično u drugoj normalnoj formi tj. smatraju se denormalizovanim. Dimenzioni model se sastoji od centralne tabele fakata i tabela dimenzija koje je okružuju i zajedno čine zvezdastu shemu.

4.1. Osnovne komponente dimenzionog modela

Tabela fakata

Tabela fakata [8], primer strukture je dat na slici 4, predstavlja osnovnu tabelu u dimenzionom modelu. Fakt predstavlja jedno poslovno merenje. U tabeli 2 dat je primer tabele fakata – tabela dnevne prodaje u kojoj jedan red predstavlja proizvod koji je određenog dana prodat u određenoj količini i ukupnoj vrednosti.

Tabela fakata	
PK,FK3	<u>Šifra Datuma</u>
PK,FK2	<u>Šifra proizvoda</u>
PK,FK1	<u>Šifra prodavnice</u>
	Prodata Količina Ukupan iznos dinara

Slika 4: Tabela fakata

Šifra_Datuma (SK)	Šifra_Proizvoda (SK)	Šifra_Prodavnice (SK)	Prodata_Kolicina	Ukupan_iznos_dinara
10	124	12	145	10000
10	125	12	16	350
10	127	12	0	0

Tabela 2: Primer tabele fakata

Važno je da su merenja u tabeli fakata na istom nivou granularnosti [9, 15]. Isto tako, najkorisniji fakti u tabeli fakata su numerički ili aditivni. Aditivnost fakata je veoma važna, jer aplikacije nad skladištem podataka skoro nikada ne vraćaju po jedan red već milione redova odjednom, a upravo je sumiranje najvažnija funkcija koja se može primeniti nad tolikim redovima. U prethodnom primeru, jasno je da se prodate količine, kao i ukupan iznos uvek lako može sumirati u validan iznos.

Teoretski je moguće i da postoje tekstualna fakata, međutim ona se obično ili kvantifikuju ili postaju deo opisnih dimenzija ili se zanemaruju, jer ih je nemoguće analizirati.

Tabele fakata su obično „male” po širini, ali zato mogu imati milione redova. S obzirom na potencijalno veliki broj redova, u tabelu fakata se ne unose „nula fakata” (fakta kojima su vrednosti u nekom trenutku jednake nuli), jer nemaju nikakvog uticaja na sumiranja podataka, a zatrpavaju tabelu. Treći red iz tabele 2 je „nula fakt” i po pravilu se izostavlja.

Svaka tabela fakata se sastoji iz dva ili više stranih ključeva. Jedinствен прост ključ se ne uvodi. Primarni ključ tabele fakata se sastoji od podskupa skupa stranih ključeva i naziva se složen ključ. Svaka tabela fakata u dimenzionom modelu ima složen ključ i obrnuto, svaka tabela dimezionog modela koja ima složen ključ je tabela fakata. Isto tako, može se reći i da je tabela fakata svaka tabela koja predstavlja više-prema-više relacije između dimenzija relacionog modela.

Tabela dimenzija

Tabela dimenzija predstavlja tekstualni opis poslovanja. Karakteriše je veliki broj kolona (50, 100 i više), a mali broj redova u odnosu na broj redova u tabeli fakata. Zauzima okvirno oko 10% prostora baze skladišta podataka. Primer strukture tabele dimenzija je dat na slici 5.

Tabela Dimenzija Proizvoda	
PK	Šifra proizvoda
	Opis proizvoda Opis kategorije Težina Jedinica mere težine Način pakovanja Tip pakovanja Način skladištenja ...

Slika 5: Tabela koja predstavlja dimenziju koja opisuje proizvode

Šifra_proizvoda	Opis_proizvoda	Kategorija	Potkategorija	Brend	Težina	Jedinica_mere
1	Smrznuto mešano povrće	Smrznuta hrana	Smrznuto povrće	Frikom	400	g
2	Beli hleb Sava	Pekara	Hleb	Beogradska pekarska industrija	500	g
3	Mešani ražani hleb	Pekara	Hleb	Hleb & Kifle	400	g

Tabela 3: Primer dela tabele dimenzija

Svaka tabela dimenzija ima jedinstveni prost primarni ključ. Atributi dimenzija se koriste za definisanje ograničenja upita, grupisanje i za prikazivanje rezultata upita. Pošto će se po atributima dimenzija kasnije vršiti upiti, kvalitet skladišta podataka direktno je srazmeran njihovom broju i preciznom određenju. Najbolji atributi su tekstualni i predstavljaju npr. opis nekog proizvoda. I numerički podatak može biti atribut, ali samo ako je nepromenljiv.

Tabele dimenzija opisuju i hijerarhijske veze poslovanja. Na primer, proizvodi se grupišu u potkategorije, pa u kategorije, pa u brendove, itd. Za svaki red u dimenziji proizvoda čuvamo podatak i o potkategoriji, kao i o kategoriji i brendu kojem pripada. Primer je dat u tabeli 3. Iako je ovakav način čuvanja podataka redundantan, normalizacija ne bi značajnije doprinela smanjenju skladišta podataka, a znatno se dobija na performansama upita.

Datumska dimenzija

Posebno važnu dimenziju predstavlja datumska dimenzija. Standardna i najviše korišćena datumska dimenzija je sa nivoom granularnosti od jednog dana. Nalazi se u skoro svakoj zvezdastoj shemi (dobra praksa je svakoj tabeli fakata dodeliti i datumsku dimenziju, jer svaki fakt je vezan za određeni vremenski trenutak) [15] i omogućava poslovnim korisnicima lako baratanje podacima i po nestandardnim vremenskim funkcijama, kao što su vikend, praznik, fiskalni mesec, i sl. Pravi se unapred nezavisno od izvornog sistema za duži vremenski period, koji uključuje prethodno poslovanje koje se unosi u skladište i narednih nekoliko godina. Za 10 godina ima približno 3652 reda, u zavisnosti od broja prestupnih godina. Takođe, nazivi atributa su opisni, da bi izveštaji bili jasni. Ukoliko kompanija ima potreba za granularnošću većom od nivoa dana (npr. po smenama zaposlenih), uvodi se posebna dimenzija doba dana. Ako je granularnost na nivou minuta, dimenzija doba dana će imati 1440 redova, a 86400, ako je na nivou sekunda. Na slici 6 dat je primer datumske dimenzije.

Tabela Dimenzija Datuma	
PK	Šifra Datuma
	Datum
	Pun opis datuma
	Dan u nedelji
	Broj dana u kalendarskom mesecu
	Broj dana u kalendarskoj godini
	Broj dana u fiskalnom mesecu
	Broj dana u fiskalnoj godini
	Indikator da li je poslednji dan u nedelji
	Indikator da li je poslednji dan u mesecu
	Kalendarski završetak nedelje
	Kalendarski broj nedelje u godini
	Kalendarski naziv meseca
	Kalendarski broj meseca u godini
	Kalendarska godina-mesec (gggg-mm)
	Kalendarska godina
	...

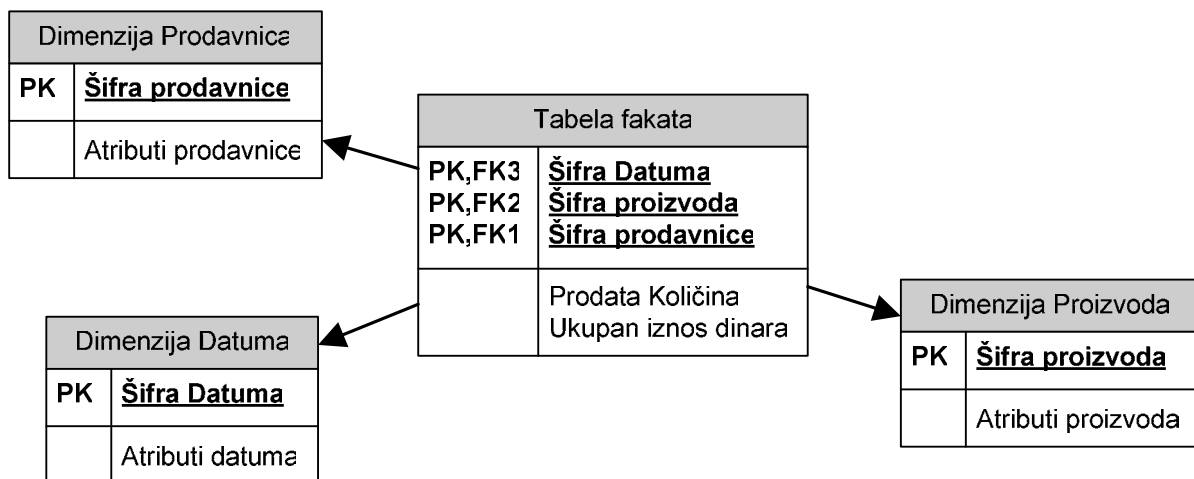
Slika 6: Tabela koja predstavlja Datumsku dimenziju

Često se dešava da jedan red tabele fakata ima više datuma, npr. datum narudžbine nekog proizvoda i datum njegove isporuke, odnosno da je u zvezdastoj shemi za jednu tabelu fakata vezano više datumskih dimenzija, a da su pri tom podaci tih tabela potpuno isti (sa obaveznom

razlikom u nazivu atributa tabela), npr. datum narudžbine i datum isporuke ili dan u nedelji narudžbine i dan u nedelji isporuke. Da bi se izbeglo dupliranje istih podataka, na fizičkom nivou se koristi ista tabela, a na logičkom nivou se definišu različiti pogledi na istu datumsku dimenziju. Na taj način se sa pozicije korisničkih alata i krajnjih korisnika dobijaju dve jedinstvene potpuno nezavisne tabele dimenzija. Ovakav način upotrebe tabele dimenzija se na engloskom označava terminom „*Dimension Role-Playing*”.

Zvezdasta shema

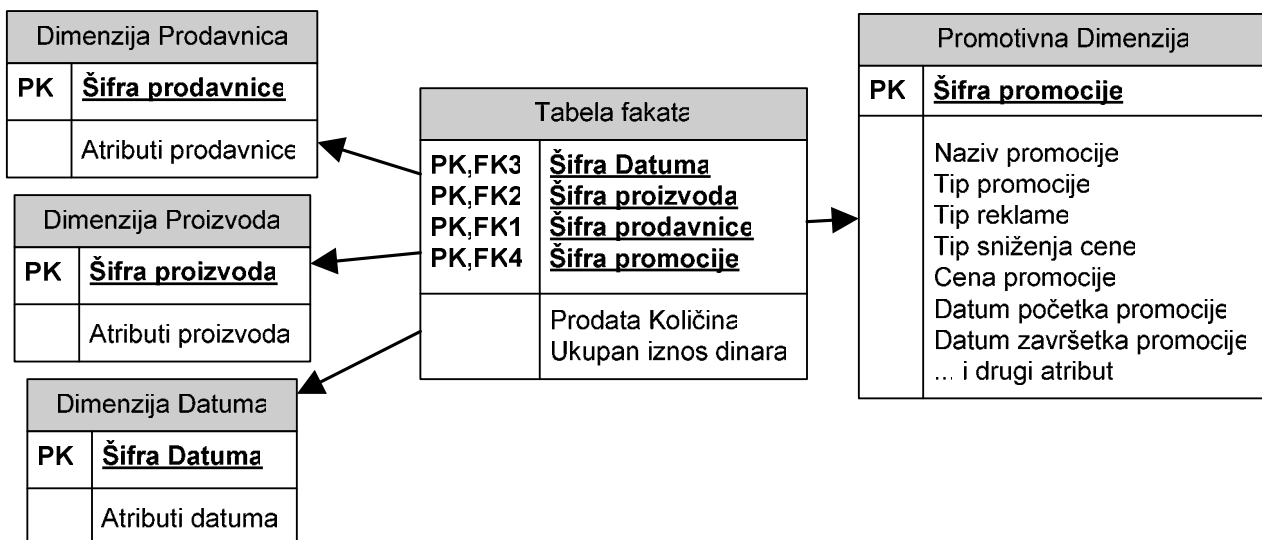
Kao što je već napomenuto, zvezdasta shema se sastoji od tabele fakata i njenih tabela dimenzija. Primer zvezdaste sheme dat je na slici 7. Zvezdasta shema može dodatno biti razgranata na taj način što će se dozvoliti tabelama dimenzija da imaju svoje poddimenzije, odnosno tako što će im se omogućiti hijerarhija atributa. Na primer, za atribut prodavnice se može uzeti šifra lokacije, a da se dimenziji – prodavnica, doda posebna poddimenzija – tabela lokacija, koja će da opisuje geografsku lokaciju na kojoj je prodavnica smeštena. Zapravo zvezdasta shema je specijalan slučaj pahuljaste sheme koja ima dubinu 1, odnosno njene dimenzije nemaju poddimenzije. Pahuljasta shema je shema čije tabele dimenzija imaju sopstvene poddimenzije. Pitanje oko koga se još vode polemike je da li nam je uopšte potrebna pahuljasta shema u modelu, s obzirom na to da usporava performanse upita čime se dovodi u pitanje jedan od osnova za izgradnju ovakvog modela. Međutim, u nekim slučajevima, logička podela podataka na poddimenzije je neophodna.



Slika 7: Primer zvezdaste sheme

Besfaktne tabele fakata

Postoje različiti poslovni procesi čije su dimenzije fakata slične onim koje smo već predstavljali, ali sa jednom razlikom, a to je da nemaju merenja. Takve tabele fakata se nazivaju besfaktne tabele fakata. Besfaktne tabele se obično koriste za opis događaja. Da bismo naveli jedan primer upotrebe besfaktne tabele fakata, pođimo od tabele fakata date na slici 7. Pridodajmo toj tabeli i promotivnu dimenziju datu na slici 8. Red promotivne dimenzije opisuje jednu moguću kombinaciju promotivnih aktivnosti.

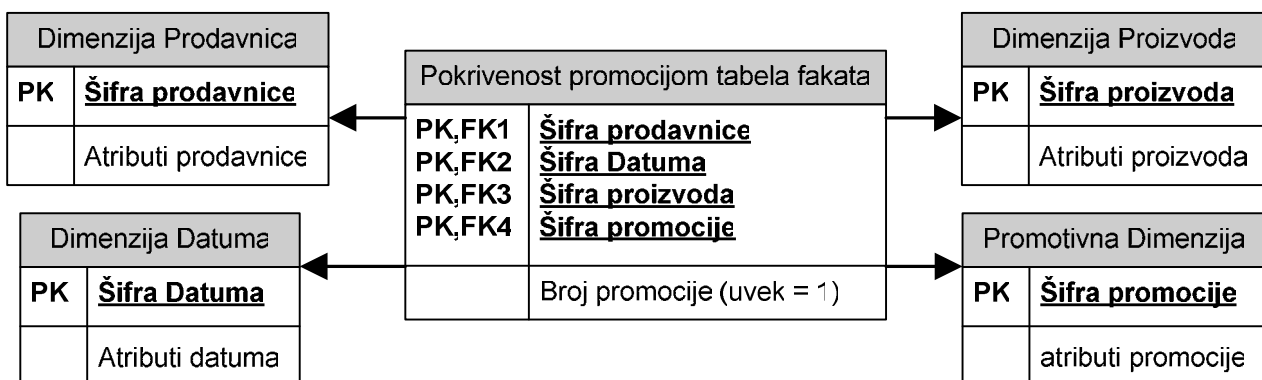


Slika 8: Primer tabele prodaje sa promotivnom dimenzijom

Promotivna dimenzija može biti važna za prodaju, jer opisuje uslove pod kojima je došlo do prodaje određenog proizvoda, samim tim predstavlja i primer dimenzije koja definiše uzroke nastanka nečega, odnosno predstavlja uzročnu dimenziju. Može dati odgovore na sledeća pitanja: da li je došlo do rasta prodaje tokom kampanje? Da li je pad prodaje pre i posle kampanje poništio efekte kampanje? Da li su ostali proizvodi u kategoriji doživeli pad prodaje?

Treba uočiti da na promocijama nisu uvek svi proizvodi koji se prodaju,, što može dovesti do toga da pojedina merenja u tabeli fakata imaju *NULL* vrednost za šifru promocije. Međutim, zbog referencijalnog integriteta, odnosno nemogućnosti upita koji imaju spajanja (*JOIN*) sa *NULL* ključevima, dodaje se poseban red „Bez ikakve promocije” u Promotivnu dimenziju.

Jedno od pitanja na koje prodajna tabela fakata ne može da dâ odgovor je: koji proizvodi su bili na promociji, ali se nisu prodavali? Tabela fakata prodaje beleži samo proizvode koji su se prodavali, a ne i one koji nisu. Da bismo dali odgovor na ovo pitanje prvo ćemo da kreiramo besfaktnu tabelu – tabelu pokrivenosti promocijom koja će beležiti sve proizvode koji su na promociji. Takva tabela data je na slici 9.

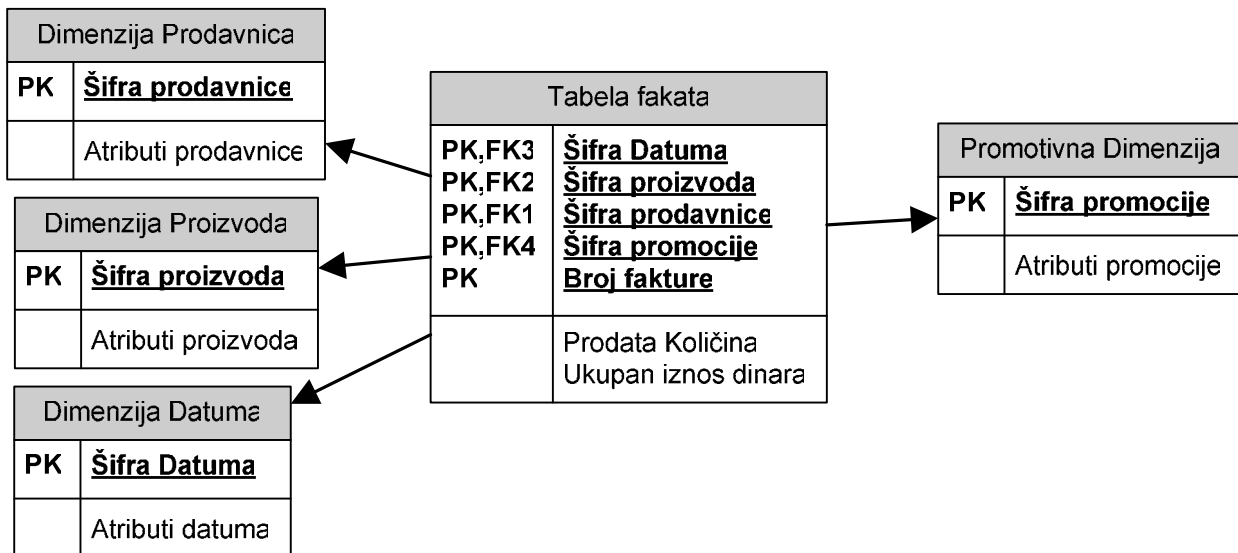


Slika 9: Pokrivenost promocijom – besfaktna tabela fakata

Neprodani, a reklamirani proizvodi koji predstavljaju odgovor na postavljeno pitanje, dobijaju se operacijom skupovne razlike skupa promovisanih i skupa prodatih proizvoda, ili automatski u OLAP okruženju.

Degenerisane dimenzije

Atributi koji figurišu u primarnom ključu tabele fakata predstavljaju podskup njenih stranih ključeva, jer nam nije uvek potreban svaki strani ključ u primarnom ključu da bismo garantovali jedinstvenost. Ukoliko neki takav atribut ne pokazuje ni na kakvu tabelu, on se naziva degenerisanom dimenzijom. Primer degenerisane dimenzije predstavlja atribut – broj fakture, dat na slici 10.



Slika10: Broj fakture kao degenerisana dimenzija

Nema potrebe za postojanjem dimenzije fakture pošto su svi potencijalni atributi, koji bi ušli u sastav te dimenzije, već raspoređeni po drugim dimenzijama. Atribut – broj fakture služi grupisanju po fakturi kao dopuna ključu tabele fakata radi jedinstvenosti. U praksi, obično brojevi kao što su broj fakture, broj transakcije, broj narudžbenice, i sl., predstavljaju degenerisane dimenzije.

Proširivost dimenzionog modela

U praksi obično postoji potreba za proširenjem i izmenom dimenzionog modela. Da bi proširenje bilo što bezbolnije po postojeće aplikacije, veoma je važno da su izmerena fakta na atomičnom nivou, a ne sumarni podaci. Preвременa sumarizacija ili agregacija podataka može ograničiti mogućnosti za proširenje modela, pre svega zato što bi pojedine dimenzije bile besmislene na višem nivou granularnosti. Navešćemo neke mogućnosti proširenja:

- Novi atributi dimenzija. Ukoliko otkrijemo nove attribute, dodaćemo ih kao nove kolone tabele dimenzija. Novi atributi neće smetati postojećim aplikacijama.
- Nove dimenzije. Postojećim tabelama fakata takođe možemo dodati nove tabele dimenzija, dodajući im nove strane ključeve i popunjavajući ih korektno vrednostima primarnih ključeva novih dimenzija.
- Nova merena fakta. Ukoliko su nova izmerena fakta na istom nivou granularnosti kao postojeća, jednostavno ćemo ih dodati u postojeća merenja prostim dodavanjem novih kolona u tabelu fakata. Ukoliko su izmerena fakta na različitom nivou granularnosti od postojećih, najbolje je napraviti novu tabelu fakata. Mešanje fakata različite granularnosti u istu tabelu fakata je gotovo uvek greška.
- Povećanje granularnosti dimenzija. Obično se atributi originalne dimenzije bez problema uklapaju u novu, granularniju odnosno usitnjeniju dimenziju, međutim granularnija

dimenzija može zahtevati granularniju tabelu fakata, što može prouzrokovati ponovno popunjavanje tabele fakata novim mernim podacima. Najvažnija činjanica je da time neće biti narušeno funkcionisanje postojećih aplikacija nad skladištem podataka.

- Dodavanje novih izvora podataka. Ukoliko ovakva promena logički bitno menja dimenzionalnost i granularnost, potrebno je napraviti novi model. Postojeće aplikacije neće biti ugrožene, jer će imati i dalje stari, nepromenjen model na raspolaganju.

Dimenziona normalizacija i denormalizacija tabela fakata

Već je napomenuo da je normalizacija dimenzija (eng. Snowflaking) još uvek diskutabilna, ali da je u pojedinim slučajevima neophodna. Međutim, praksa pokazuje da se normalizaciji dimenzija treba generalno odupreti i to iz više razloga:

- Komplikuje se korišćenje podataka i dovode u neprijatnu situaciju korisnici da se bore sa kompleksnošću, čime narušavamo jedan od osnovnih principa skladišta podataka, a to je jednostavnost.
- Usporava izvršenje upita. Veliki broj tabela i spajanja obično ima za posledicu sporo izvršavanje upita čime se dovodi u pitanje efikasnost skladišta podataka.
- Ušteda prostora koja bi se dobila normalizacijom tabela dimenzija je zanemarljiva u odnosu na veličine, npr. tabela fakata. Normalizacija tabela dimenzija zbog uštede prostora je gubljenje vremena.
- Krstarenje po dimenzijama i različita grupisanja zbog preglednosti veoma se teško i sporo izvode.
- Dimenziona normalizacija osujećuje *bitmap* indeksiranje [12]. *Bitmap* indeksiranje je prvenstveno namenjeno skladištima podataka i pogodno je za pretragu velike količine podataka i ad hoc upite i veoma je efikasno kod niskokardinalnih atributa. Vrednosti takvih atributa se lako kodiraju bitovima, a *AND* i *OR* uslovi u upitima se brzo mogu razrešiti korišćenjem odgovarajućih bulovih operacija. Standardni indeks obezbeđuje listu vrednosti sa identifikatorima redova iz tabele za svaku ključnu vrednost. *Bitmap* indeks obezbeđuje istu funkcionalnost, ali umesto identifikatora redova koristi *bitmapu*, odnosno niz bitova. Određena kombinacija bitova odgovara određenoj ključnoj vrednosti.

Tokom projektovanja dimenzionog modela može se doći na ideju da se izvrši denormalizacija tabela fakata. Tako bismo, na primer, mogli umesto jedinstvenog ključa proizvoda da uvedemo i ključ vrste proizvoda, linije proizvoda i sl., čime denormalizujemo tabelu prodajnih fakata. Ovakav model je opasan po performanse, jer je tabela fakata površinski najveća u dimenzionom modelu i ne bi trebalo da se dodatno povećava. Hijerarhijski uređeni atributi i statistički značajno korelirani atributi treba da su u okviru iste dimenzije. Na engleskom se denormalizovana tabela fakata naziva „*Centipede fact table*”, tj. stonogasta tabela fakata, zbog pojave velike količine dimenzija u modelu.

Nedeskriptivni ključevi

Važno je napomenuti da se kao šifre stranih ključeva tabele fakata, odnosno primarnih ključeva tabela dimenzija koriste nedeskriptivni ključevi i to obično prirodni brojevi. Dizajneri imaju tendenciju da iskoriste već postojeće deskriptivne ključeve iz produkcionih sistema. Međutim, to nije dobra praksa iz više razloga [15]. Pre svega, samim tim što su deskriptivni ključevi iz živih produkcionih sistema, može se, npr. desiti, da posle nekog vremena neaktivnosti budu reciklirani i ponovno upotrebljeni, čime bi izazvali konfuziju u izgrađenom skladištu podataka – imali bismo isti ključ za dve različite stvari. Nedeskriptivni ključevi čuvaju skladište podataka od izmena u

produkcioni sistemima, a takođe dopuštaju integraciju podataka iz više izvora. Mogu se takođe javiti problemi oko performansi sistema, jer deskriptivni ključevi produkcionih sistema često umeju da budu glomazne alfanumeričke šifre. Ako uzmemo u obzir činjenicu, da jedan bajt u redu faktabele u tabeli od milijardu slogova zauzima gigabajt prostora na disku, jasno je da su glomazne šifre potpuno besmislene. Četvorobajtni pozitivni brojevi pokrivaju oko dve milijarde pozitivnih vrednosti što je uglavnom dovoljno za bilo koju tabelu dimenzija.

Poluaditivna fakta

Posmatrajmo model zaliha prodajnog objekta. Dimenzije su identične dimenzijama iz modela prodaje date na slici 7, s tom razlikom što je podatak koji se meri u tabeli fakata umesto prodate količine i ukupnog iznosa, sada stanje na zalihama. Model je takođe prikazan i na slici 11. Merenje u tabeli fakata predstavlja periodičan snimak zaliha. Razlika u odnosu na prodaju je u tome što se prodaja beleži samo za prodate artikle, a zalihe se beleže svakodnevno za sve artikle. Da bismo izbegli preterano nagomilavanje podataka možemo merenja starija od npr. 90 dana prorediti i čuvati na nivou niže granularnosti, npr. nedeljnom, u odgovarajućim istorijskim tabelama.

Periodični snimak zaliha koji smo napravili je sabirljiv po objektima i po proizvodima, ali ne i po datumima. Poluaditivna su ona fakta koja je moguće sumirati po jednim, a nije moguće sumirati po drugim dimenzijama, npr. stanje bankovnog računa, merenja fizičkih veličina poput temperature, pritiska, i sl. U ovakvim situacijama, agregaciju je moguće uspešno izvršiti izvlačeći prosek po broju vremenskih jedinica, npr. ako hoćemo prosek stanja na zalihama za poslednjih deset dana, možemo da saberemo stanja i podelimo sa deset čime dobijamo odgovarajuću agregaciju.

Sporo promenljive dimenzije

Sporo promenljive dimenzije (engl. *Slowly Changing Dimensions* ili skraćeno SCDs) definišu se kao dimenzije koje podležu minornim promenama tokom vremena. Takođe se nazivaju i skoro konstantnim dimenzijama. Po pravilu, dimenzije nisu potpuno nezavisni entiteti i kako se poslovni proces menja i razvija, tako dolazi i do povremenih promena vrednosti atributa dimenzija, dimenzionog modela. Za svaki atribut tabele dimenzije potrebno je definisati strategiju – kako odgovoriti na promene. Promene obično nastaju u produkcionim sistemima odakle se reflektuju na skladište podataka.

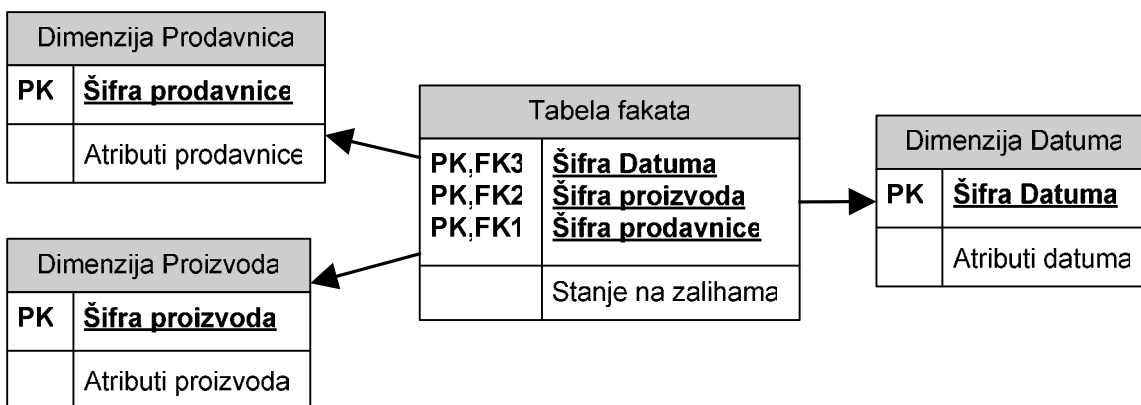
Postoje tri osnovne tehnike, odnosno strategije koje se primenjuju pri promeni atributa, a mogu se koristiti njihove kombinacije. To je tzv. hibridni pristup.

- Tip 1: Prepisivanje preko postojećih vrednosti. Jednostavno se vrednost starog atributa zameni vrednošću novog atributa. Tip 1 predstavlja jednostavan i brz način za rešavanje promena vrednosti atributa, a pri tome se struktura tabele fakata uopšte ne menja. Međutim, postoji jedan veliki problem sa promenama Tipa 1, jer ne čuvaju istoriju vrednosti atributa. Posle promene nemamo više načina da saznamo prethodnu vrednost. Ova strategija je prihvatljiva ako nam nije potrebno da pamtimo istoriju vrednosti atributa, odnosno ako to nije potrebno poslovnim korisnicima koji o tome odlučuju.
- Tip 2: Dodavanje reda u tabelu dimenzije. Promenu rešavamo na taj način što dodajemo novi red u tabelu dimenzije. Takođe ne vršimo nikakve promene u tabeli fakata. Ova tehnika je najčešće korišćena tehnika za rešavanje promena.
- Tip 3: Dodavanje kolone u tabelu dimenzija. Iako je Tip 2 promene veoma pogodan, on ipak ima određena ograničenja, npr. ne omogućava nam da povežemo novu vrednost atributa sa istorijom fakata i obrnuto, odnosno ne omogućava nam da vidimo podatke iz tabele fakata kao da promene nije ni bilo, a poslovni korisnici mogu iz nekog razloga

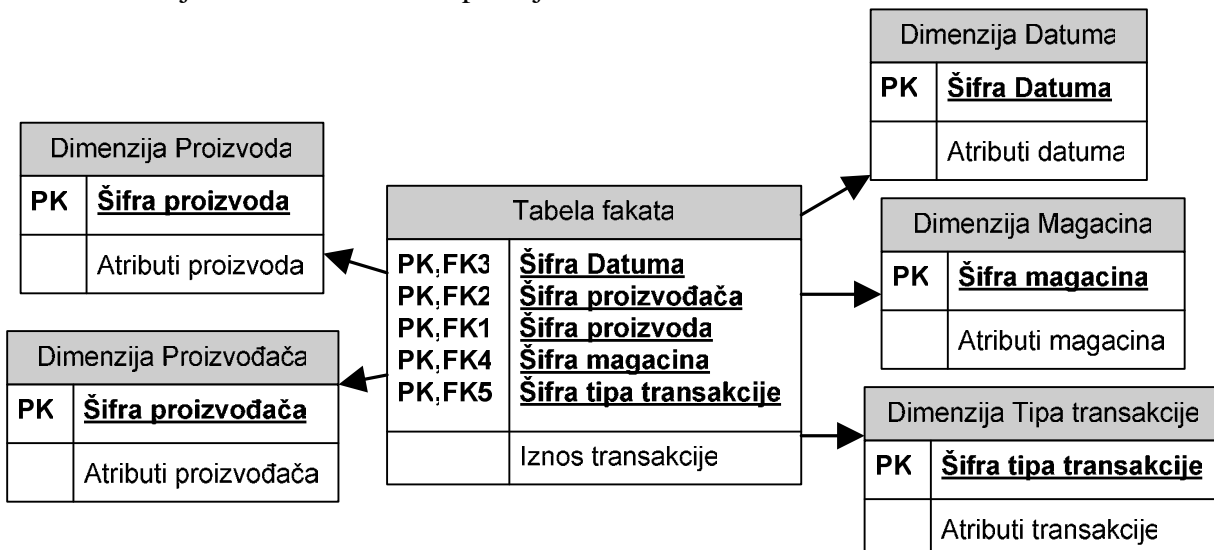
zahtevati upravo da vide podatke prikazane na stari način. Neki dizajneri ovakav prikaz nazivaju alternativna stvarnost (engl. *Alternate Reality*). Ovom tehnikom se uvodi nova kolona u dimenziju i u njoj se čuva stara vrednost atributa. Na taj način, iako se promena desila, moguće je logički posmatrati sistem kao da nije. Tip 3 se ne koristi često, ali postoje situacije u kojima je neophodan.

Integrisanje lanca vrednosti

Svaka organizacija sadrži lanac vrednosti koji se sastoji od njenih ključnih poslovnih procesa. Lanac vrednosti identifikuje prirodni tok organizacionih najvažnijih aktivnosti. Na primer, firma koja se bavi prodajom može da pošalje narudžbenicu proizvođaču nekog proizvoda. Proizvođač dostavlja naručene proizvode firmi u njen magacin. Odatle se proizvod transportuje u magacin lokalne prodavnice u kome se nalazi dok ga ne kupi neki potrošač. Ovim smo obuhvatili tri procesa odnosno tri zvezdaste sheme: model prodaje, model stanja na zalihama u maloprodaji i model transakcija nad zalihama u magacinu. Model prodaje prikazan je na slici 10. Model stanja na zalihama u maloprodaji dat je na slici 11, a model transakcija nad zalihama dat je na slici 12.



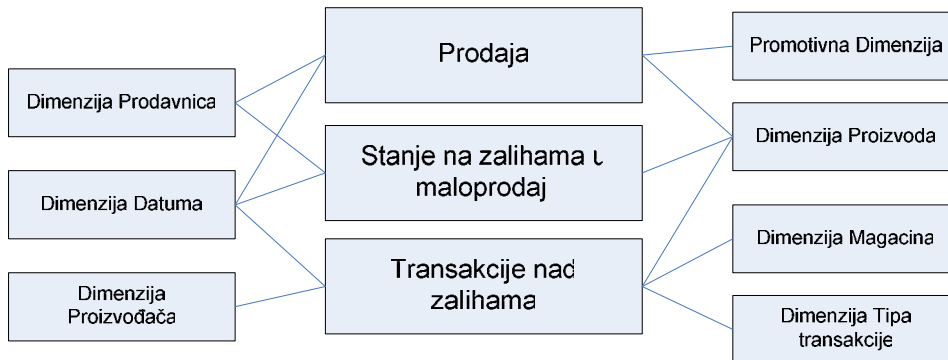
Slika 11: Stanje na zalihama u maloprodaji



Slika 12: Model transakcija nad zalihama

Iako su u pitanju različiti poslovni procesi i različite tabele fakata, one imaju nekoliko zajedničkih tabela dimenzija i to su dimenzija datuma, dimenzija proizvoda i dimenzija prodavnice. Logičko deljenje dimenzija prikazano je na slici 13. Da bi se ostvarila integracija među poslovnim procesima, odnosno integracija lanca vrednosti, jasno je da je potrebno modelirati takve dimenzije

na takav način da ih je moguće deliti među procesima. Ovakve dimenzije se nazivaju usklađene dimenzije (engl. *Conformed Dimensions*). Zajedničke dimenzije omogućavaju kombinovanje performansi iz različitih procesa u isti prikaz, čime omogućavaju konzistentan prikaz informacija kroz više procesa što je po pravilu posebno interesantno menadžmentu preduzeća. Na kraju, imaćemo skup zajedničkih dimenzija čijim ćemo pogodnim kombinovanjem implementirati različite poslovne procese iz lanca vrednosti neke organizacije.



Slika 13: Deljenje dimenzija među poslovnim procesima

Skup dimenzija iz kojih se inkrementalno grade zvezdaste sheme naziva se magistrala skladišta podataka (engl. *Data Warehouse Bus*). Definisane usklađene dimenzije, tokom faze arhitekture rešenja, pokazalo se dobrim u praksi, za izgradnju skladišta podataka. Takođe, na ovaj način različiti skupovi zvezdastih shema mogu biti građeni od strane različitih timova u različito vreme, a opet su u mogućnosti da koegzistiraju zahvaljujući zajedničkom interfejsu usklađenih dimenzija. Usklađene dimenzije su ili identične ili su podskup najusitnjenije i najdetaljnije dimenzije; moraju imati iste ključeve i iste nazive i skupove vrednosti atributa.

Još jedan pogodan način za prikaz magistrale skladišta podataka je matrica magistrale skladišta podataka (engl. *Data Warehouse Bus Matrix*). Matrica skladišta podataka za naš primer data je u tabeli 4. Red matrice predstavlja jednu poslovnu celinu skladišta podataka (engl. *data mart*). Ako matricu magistrale pogledamo po kolonama, vidimo da se dimenzije datuma i proizvoda pojavljuju u sva tri poslovna procesa koja želimo da implementiramo. Ovo se može uzeti kao reper, odnosno pokazatelj, koje su dimenzije najvažnije i od kojih prvo treba započeti modelovanje i na koje treba obratiti najviše pažnje.

Poslovni procesi	Usklađene dimenzije						
	Datum	Proizvod	Promocija	Magacin	Prodavnica	Proizvođač	Tip transakcije
Prodaja	x	x	x		x		
Stanje na zalihama	x	x			x		
Transakcije nad zalihama	x	x		x		x	x

Tabela 4: Matrica magistrale skladišta podataka

4.2 Proces dimenzionog projektovanja

Proces izgradnje dimenzionog modela se odvija u četiri koraka:

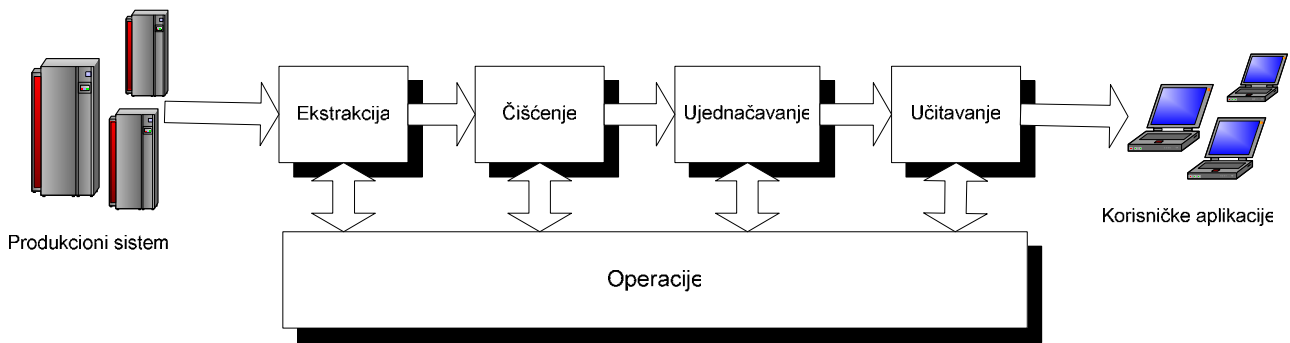
- 1. Izbor poslovnog procesa koji se modeluje.** Velika je greška definisati dimenzioni model po odeljenjima preduzeća (npr. marketing, finansije, prodaja), jer kad se dobro sagledaju potrebe ovih odeljenja dolazi se do zaključka da su im u velikoj meri potrebni isti podaci, kao npr. podaci o porudžbinama. Umesto da pravimo posebno shemu porudžbina za marketing, posebnu za prodaju i posebnu za finansije, bolje je napraviti jednu detaljnu shemu porudžbina, kojoj će pristupati različita odeljenja preduzeća. Izbor poslovnog procesa koji će prvi biti modelovan je obično onaj koji može dati odgovore na najvažnija pitanja, ali i koji poseduje podatke najpogodnije za ekstrakciju i punjenje u dimenzione modele. Pošto najvažniji resurs nekog preduzeća obično predstavljaju zaposleni u tom preduzeću, često se prvo implementira HR menadžment (engl. *Human Resources*) ili proces upravljanja ljudskim resursima.
- 2. Definisanje granularnosti.** Definisanjem granularnosti se direktno određuje šta će biti sadržano u tabelama fakata. Šta sve može predstavljati red u tabeli fakata? Na primer, dnevni snimak nivoa zaliha za svaki artikal na lageru ili mesečni snimak stanja bankovnog računa, i sl. Veoma je važno da granularnost bude na atomičnom nivou ili možemo doći u situaciju da korisnici žele da pristupe nivou detalja koji nismo predvideli i time ostanu uskraćeni za odgovore koje upravo skladište podataka i treba da im pruži.
- 3. Definisanje dimenzija.** Dimenzije se dobijaju iz opisa podataka ljudi koji učestvuju u samom poslovnom procesu koji se modeluje. Takođe, ako smo i jasno definisali granularnost, neće nam biti teško da definišemo atribut tabele dimenzija. To su obično dimenzije datuma, proizvoda, klijenata, i sl. Kada se implementira HR, važna dimenzija će biti dimenzija sa podacima o zaposlenima u preduzeću, odnosno njihov profil.
- 4. Definisanje fakata.** Definisanje fakata se obično vrši odgovorom na pitanje: „Šta merimo?“ Pre svega, to su brojčane vrednosti koje mogu da se saberu. Jasno je da fakti moraju biti u skladu sa definisanom granularnošću u koraku 2 procesa i fakti na različitim nivoima granularnosti moraju biti u zasebnim tabelama fakata. Takođe bi trebalo uključiti i izvedene vrednosti, jer postoji opasnost da ih korisnik sam ne bi dobro izračunao (npr. neto profit kao razlika maloprodajne cene i nabavne cene proizvoda). Preporuka je da se i ovakve izvedene vrednosti fizički čuvaju u bazi podataka, jer njihovo skladištenje nije skupo.

Kako je fokus dimenzionog modelovanja poslovni proces, skup zvezdatih shema koje opisuju jedan poslovni proces ćemo nazivati tržištem podataka (engl. *data mart*). Tržište podataka predstavlja podskup celokupnog skladišta podataka i ne zavisi od organizacionih jedinica ili odeljenja preduzeća. U skladu sa definisanim procesom dimenzionog modelovanja, tržište podataka je za sva odeljenja isto.

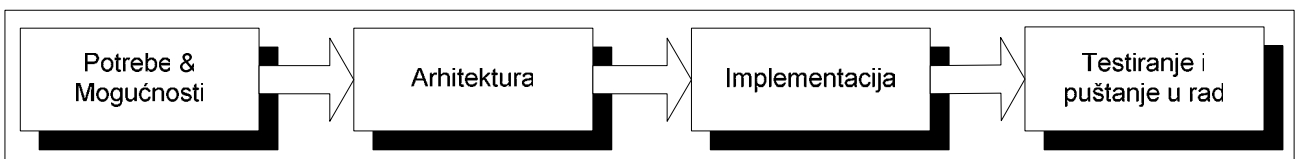
5. ETL

5.1 Uvod

Niz obrada koje podatke iz različitih produkcionih sistema integrišu u skladište podataka naziva se ETL [7], skraćeno od ekstrakcija, transformacija i punjenje. Prema istraživanjima, aktivnosti vezane za ETL zauzimaju čak 70% vremena jednog projekta izrade skladišta podataka. Pri izgradnji ETL sistema treba voditi računa o dva toka, prikazana na slikama 14 i 15, koja se odvijaju paralelno: to su tok planiranja i projektovanja ETL procesa i tok podataka u ETL procesu.



Slika 14: Tok podataka u ETL procesu



Slika 15: Tok planiranja i projektovanja ETL procesa

Pre same faze ekstrakcije podataka potrebno je pripremiti prihvatno područje, odnosno definisati određene strukture podataka u koje će biti smešteni ekstrahovani podaci. Ove strukture podataka mogu biti različite, npr. obični tekstualni fajlovi ako koristimo *Perl*, *VBScript* ili *JavaScript* za ETL, ili npr. *XML* strukture podataka. Takođe se često koriste relacione baze podataka za prihvatno područje, kao i kombinacije različitih struktura. Implementirati prihvatno područje u relacionoj bazi podataka može biti prilično komplikovano, jer takva struktura mora biti pogodna, odnosno da iskazuje dobre performanse kako za unošenje velikih količina podataka tako i za njihovo iznošenje.

5.2 Ekstrakcija podataka

Bez podataka, skladište podataka je beskorisno. Prvi korak ka integraciji različitih izvora podataka je ekstrakcija podataka iz produkcionih sistema. Osnovno pitanje je kako pristupiti izvornim produkcionim sistemima, s obzirom na činjenicu da se oni mogu sastojati iz različitih sistema za upravljanje bazama podataka, operativnih sistema, hardvera, da koriste različite komunikacione protokole, i sl. Pristup i ekstrahovanje podataka iz izvornih sistema u prihvatno područje obično vrši poseban podsistem za ekstrakciju podataka. Podaci jednom ekstrahovani u prihvatno područje postaju nezavisni od izvornih sistema i mogu se dalje transformisati prema potrebama krajnjih korisnika.

Pre početka konkretne implementacije ETL sistema, izrađuje se logička mapa podataka koja opisuje veze između samog početka i samog kraja ETL sistema. Logička mapa podataka se sastoji iz sledećih komponenti:

- Naziv ciljne tabele. Naziv tabele koja se nalazi u skladištu podataka, a u koju će biti smešten podatak koji se ekstrahuje.
- Naziv ciljne kolone. Naziv kolone u koju će biti smešten ekstrahovani podatak.
- SCD tip. Ova komponenta mape određuje način na koji će se implementirati promene i to je obično Tip 1, 2 ili 3.
- Izvorna baza podataka. Naziv instance baze podataka iz koje se ekstrahuju podaci. Ovo takođe može biti i string za povezivanje sa izvornom bazom podataka.
- Naziv izvorne tabele. Lista naziva izvornih tabela iz kojih se učitavaju podaci.
- Naziv izvorne kolone. Lista naziva izvornih kolona iz kojih se učitavaju podaci.

- Transformacija. Tačna manipulacija koja je potrebna da se izvrši nad izvornim podatkom da bi podatak odgovarao ciljnom formatu.

Da bi logička mapa podataka mogla uspešno da se kreira, potrebno je potpuno poznavanje produkcionih sistema organizacije (koje ćemo u kontekstu ETL-a nazivati i izvorni sistemi podataka) za koju se skladište podataka kreira. U tom cilju, vrši se analiza produkcionih sistema u kojoj se detaljno dokumentuju postojeći izvori podataka relevantni za skladište podataka, a čiji je rezultat detaljan izveštaj sa njihovim opisom, kao i njihovim korisnicima. Jedna od važnih komponenti ovog izveštaja je i ER shema ili shema entiteta i odnosa (engl. *Entity Relationship Diagram*) pod uslovom da su izvori podataka relacione baze podataka. Često je slučaj u praksi da ovakva shema nije dostupna i da se mora kreirati reverznim inženjeringom baze podataka. Postoje različiti alati koji obavljaju ovu funkciju i obično su uključeni u standardne pakete alata za ETL. Oni rade tako što od postojećih metapodataka baze kreiraju ER shemu.

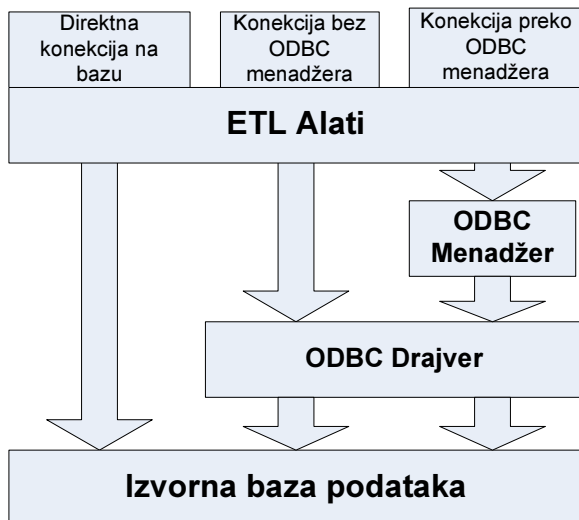
Osnovne karakteristike koje hoćemo da saznamo o produkcionoj bazi podatka su:

- Jedinstveni identifikatori i prirodni ključevi. Sa tačke gledišta poslovne primene, veoma je važno da znamo šta jedinstveno određuje neki red u tabeli, a da to nije veštačka šifra koja je obično primarni ključ tablele. Važno je uočiti i prirodni ključ.
- Tipovi podataka. Veoma je važno utvrditi tip podatka da ne bismo kasnije došli u problematičnu situaciju da npr. pokušavamo da podatak tipa CHAR učitamo u NUMBER, i sl.
- Odnosi među tabelama. Prilikom učitavanja podataka u ciljnu tabelu iz različitih izvora veoma je važno poznavati relacije među njima. Pokazano je u praksi da je dobro prethodno učitati izvorne tablele u alat za modelovanje i nacrtati relacije među njima.
- Kardinalnost odnosa. Takođe je važno znati kardinalnost odnosa među izvornim tabelama, odnosno na koji način su one spojene.

Pored metapodataka baze podataka, takođe nas zanimaju i konkretni podaci koji se u toj bazi nalaze. Naš cilj je da upravo te podatke ekstrahujemo, transformišemo i učitamo u skladište podataka. Razumevanje podataka zahteva određenu analizu sadržaja baze, kao i otklanjanje određenih anomalija, a to mogu biti:

- *NULL* vrednosti. *NULL* vrednosti mogu predstavljati veliki problem za ETL proces, pogotovu ako se nalaze u ulozi stranog ključa tablele. Prvenstveno treba proveriti i utvrditi da li postoje strani ključevi sa vrednostima *NULL*. Takođe, nije praksa da se učitavaju *NULL* vrednosti u skladište podataka. Umesto toga treba definisati neke podrazumevane vrednosti, npr. – string "Nepoznata vrednost", koji bismo upisali umesto *NULL*.
- Datumi u nedatumskim poljima. Datumi su specifični, jer se mogu koristiti različiti formati i zapisi za isti datum. Isto tako, može se desiti da datumi u starijim sistemima budu zapisani u tekstualnim poljima.

Prikupljanje podataka iz različitih baza podataka se najlakše može rešiti korišćenjem ODBC (engl. *Open Database Connectivity*) tehnologije. Svrha ODBC tehnologije je da učini aplikacije nezavisnim od baze podataka kojoj pristupaju, tj. da promena baze ne zahteva ponovnu kompilaciju aplikacije. Načini na koje se ETL povezuje sa izvorima podataka dat je na slici 16.



Slika 16: Povezivanje ETL alata sa bazom

Sama činjenica da ODBC omogućava konekciju na skoro svaku bazu na skoro svakoj mogućoj platformi, predstavlja istovremeno i njegovo veliko ograničenje, pre svega po pitanju performansi i po pitanju gubitka SQL komandi specifičnih za izvorne baze koje možda ODBC SQL ne podržava. Do pada performansi dolazi pre svega zbog korišćenja još dve dodatne komponente, a to su *ODBC Manager* i *ODBC Driver*. *ODBC Manager* prihvata SQL komande od ETL aplikacije i usmerava ih ka *ODBC Driver*-u. *ODBC Driver* prevodi ODBC SQL u izvorni SQL baze nad kojom radi.

Pored inicijalnog učitavanja podataka, obaveza ETL-a je da prati i detektuje promene podataka u izvornim sistemima. Da bi skladište podataka pratilo poslovanje preduzeća, potrebno je u njemu reflektovati tekuće stanje poslovanja. Takođe, treba voditi računa o „važnosti” podataka. Krajnji korisnici mogu zahtevati da im u svakom trenutku budu na raspolaganju ažurni podaci (engl. *real-time delivery*), međutim takav zahtev može drastično uticati na performanse celokupnog sistema i drastično povećati cenu i kompleksnost implementacije. Da bi se takav problem prevazišao, preporuka je podeliti podatke po „važnosti”. Podaci se, po tom kriterijumu, obično dele na četiri kategorije:

1. Potrebno ih je odmah ekstrahovati i učitati;
2. Ekstrahovanje se vrši nekoliko puta u toku dana;
3. Ekstrahovanje se vrši jednom dnevno;
4. Ekstrahovanje se vrši u vremenskom periodu dužem od jednog dana.

Postoji nekoliko strategija za praćenje promena:

- Korišćenje kolona za praćenje. U većini slučajeva izvorni sistemi podataka sadrže posebne kolone za praćenje koje se dodaju na kraj tabele, a u kojima se upisuje vreme i datum kada je neki slog dodat u tabelu, odnosno kada je modifikovan. Ove kolone mogu biti iskorišćene za detektovanje promena. Osnovni postupak je da se uporedi datum poslednje promene svakog sloga sa datumom poslednjeg učitavanja podataka u skladište podataka. Ukoliko je datum promene veći, onda se ti podaci učitavaju. Sami datumi smešteni u kolonama za praćenje se ne učitavaju u zvezdaste sheme već se čuvaju u ETL tabelama koje su tome namenjene. Ukoliko tabele fakata zahtevaju takve podatke zbog, na primer, statistike, onda bi trebalo kreirati posebnu dimenziju namenjenu praćenju podataka.

- Praćenje logova baze podataka. Praćenje logova baze podataka podrazumeva periodično očitavanje logova i pronalaženje transakcija koje su se izvršile nad tabelama relevantnim za ETL. Mana ove tehnike je što logovi mogu da se prepune i da sprečavaju odvijanje novih transakcija. Prirodna reakcija DB administratora na ovakvu situaciju je da obrišu logove, čime se i gube relevantni podaci za ETL. Da bi se ovakve situacije izbegle, poželjno je definisati poseban log koji će biti namenjen isključivo za potrebe ETL procesa.
- Proces eliminacije. Ovaj postupak učitava sve podatke ponovo iz svih izvornih tabela i upoređuje ih sa prethodno učitanim, tražeći promene. Iako neefikasan, ovaj postupak je veoma pouzdan. Takođe, lako će detektovati obrisane slogove iz baze koji bi možda ostalim tehnikama promakli.

Podatke ekstrahovane u prihvatno područje trebalo bi sačuvati do završetka celokupnog ETL procesa, odnosno do njihovog upisa u zvezdaste sheme iz bezbednosnih razloga, jer može doći do neočekivanog pucanja ETL procesa i potencijalnog gubljenja podataka. Samim tim, potrebno je imati sačuvane inacijalne podatke sa kojima je moguće ponoviti ceo postupak.

ERP sistemi kao izvori podataka

Veoma važni izvori podataka za skladište podataka su ERP (engl. *Enterprise Resource Planning* – softver za upravljanje poslovanjem preduzeća) sistemi, jer predstavljaju osnovni sistem za poslovanje jednog preduzeća. Međutim ERP sistemi kao što su SAP³ (skraćeno od: *Systems, Applications and Products in Data Processing*) ili Oracle EBS⁴ (skraćeno od: *Enterprise Business Suite*) mogu imati na desetine hiljada tabela u svojim bazama podataka koje sadrže ogromne količine podataka, koje skoro da je nemoguće pravilno ekstrahovati. Postavlja se pitanje, da li je uopšte potrebno da se iz ERP sistema ekstrahuju podaci u skladište podataka kad već sami ERP sistemi imaju svoje alate za izveštavanje. Međutim, pokazuje se da je to, ipak, neophodno, i da dati alati ne mogu dati zadovoljavajuće odgovore kad je u pitanju podrška odlučivanju. Zato mnogi proizvođači velikih ERP sistema proizvode i adaptere koji upravo služe povezivanju ERP sistema sa skladištem podataka. SAP je npr. kreirao svoju nadogradnju za podršku odlučivanju koja se naziva „*Business Information Warehouse*” ili skraćeno SAP BW. SAP BW skladište podataka sadrži sve komponente jednog skladišta podataka navedene u glavi 2. Takođe, dozvoljava i povezivanje alata za izveštavanje drugih proizvođača na svoje repozitorijume. Preporuka je da se ovakvi alati obavezno koriste kad su u pitanju veliki ERP sistemi.

5.3 Transformacija podatka

Nakon završene faze ekstrakcije podataka podrazumevamo da su podaci iz različitih izvornih sistema smešteni u određene strukture podataka samog ETL procesa. Pre svega, potrebno je definisati kvalitet podataka, a nakon prolaska podataka kroz podsistem za transformaciju i proveriti da li je definisani kvalitet zaista postignut. Od podataka očekujemo da budu tačni, nedvosmisleni, konzistentni i kompletni. Pod kompletnošću podrazumevamo da su sve vrednosti definisane (nisu *NULL*) i da se u samom procesu nisu neki podaci izgubili.

Određenom politikom kvaliteta koju kreiraju krajnji korisnici, a koja zavisi od konkretne implementacije, definišemo kvalitet podataka. Ako podatak nije kvalitetan znači da ga podsistem za

³ <http://www.sap.com>

⁴ <http://www.oracle.com/us/products/applications/ebusiness/index.html>

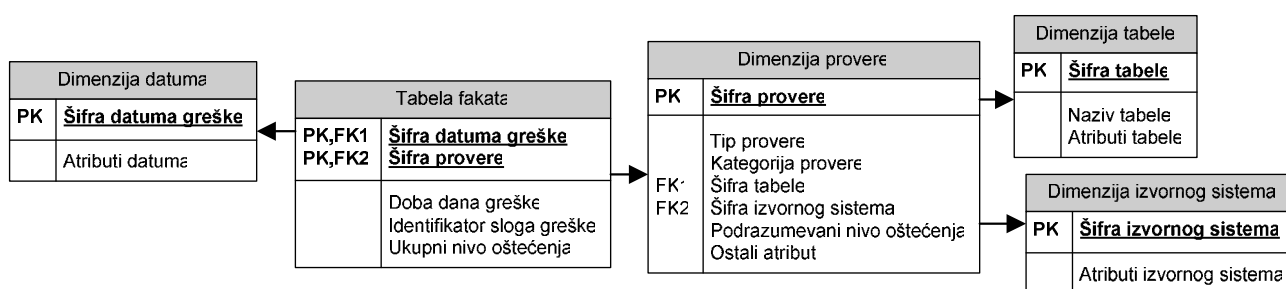
transformaciju, ukoliko je to moguće, mora popraviti. Da bi što efikasnije sproveli popravke nad nekvalitetnim podacima, podelićemo ih u nekoliko kategorija na sledeći način:

- Kategorija A. Podaci koji su u ovoj kategoriji ne mogu se tehnološki generisati u podsistemu za transformaciju, npr. zato što ne postoje u izvornom sistemu i moraju se popraviti na nivou izvornog sistema. Podsystem za transformaciju podataka treba podatak iz ove kategorije da prepozna kao nedostatak izvornog sistema i da ga jasno obeleži kao takvog, kako bi skrenuo pažnju, npr. rukovodstvu preduzeća, na nedostatke u izvornim sistemima.
- Kategorija B. U ovu kategoriju spadaju podaci koje bi bilo najbolje popraviti na nivou izvornog sistema, čak i ako postoji neki način da se oni generišu na nivou ETL-a.
- Kategorija C. U ovu kategoriju spadaju podaci koje treba popraviti na nivou ETL-a pre nego na samom izvornom sistemu.
- Kategorija D. Podaci iz ove kategorije se mogu popraviti isključivo na nivou ETL sistema. U praksi nema mnogo podataka koji potpadaju pod ovu kategoriju. Sve promene koje se vrše nad podacima na ovaj način treba da su opisane i vidljive krajnjim korisnicima kroz metapodatke.

Pred sam sistem za transformaciju podataka, postavljaju se velika očekivanja; pre svega očekivanje da bude temeljan u pronalaženju grešaka, njihovom ispravljanju i dokumentovanju kvaliteta informacija radi otkrivanja izvora nekvalitetnih podataka. Takođe se, pored navedenog, očekuje da sistem bude brz u obavljanju svog posla.

Čišćenje podataka

Da bi se realizovao sistem za praćenje kvaliteta podataka i lociranje izvora nekvalitetnih podataka tokom rada ETL sistema, kreira se tabela fakata za praćenje grešaka (engl. *Error Event Table*) i skup odgovarajućih dimenzija. Svaki podatak koji nije kvalitetan, podsistem za transformaciju podataka beleži kao red u tabeli fakata za praćenje grešaka. Granularnost ove tabele fakata je na nivou pojedinačne greške. U ovom kontekstu, pod greškom podrazumevamo nekvalitetan podatak. Zvezdasta shema za praćenje grešaka, data je na slici 17. Da bismo utvrdili da li je neki podatak kvalitetan ili ne, vršimo nad njim različite provere.



Slika 17: Zvezdasta shema za praćenje grešaka

U dimenziji provere definisani su svi tipovi provera koje vršimo nad podacima. U dimenziji tabele opisane su tabele podataka koje predstavljaju domen za datu proveru. U dimenziji izvornog sistema opisan je izvorni sistem odakle su podaci ekstrahovani. Poslednje dve dimenzije služe za lociranje izvora nekvalitetnih podataka. Tabela fakata ove sheme predstavlja ključni skup metapodataka za otkrivanje, analizu i kontrolu kvaliteta podataka, koja ima za cilj povećanje kvaliteta podataka tokom vremena.

Provere koje vršimo nad podacima možemo podeliti u tri kategorije:

- Provera svojstava atributa. Ove provere koristimo da bismo utvrdili da li se podaci uklapaju u postavljena svojstva. Provere mogu biti sledeće:
 - da li postoje NULL vrednosti u atributima čije su vrednosti zahtevane;
 - da li su numeričke vrednosti u definisanom opsegu;
 - da li su vrednosti atributa u definisanom diskretnom skupu podataka;
 - da li vrednosti nisu u skupu pogrešnih vrednosti (u slučaju da je skup ispravnih vrednosti veliki);
 - da li su vrednosti ispravno napisane (*spell-check*);
 - da li se vrednost uklapa u određenu mustru (zadatu nekim regularnim izrazom), i sl.
- Provera strukture podataka. Ove provere koristimo da utvrdimo ispravnost strukturnih podataka, kao što su primarni i strani ključevi podataka i da li poštuju referencijalni integritet. Takođe se proverava hijerarhijska ispravnost podataka, ako su podaci u hijerarhiji; na primer, ako je u pitanju dete – roditelj hijerarhija (engl. *parent-child relationship*), da li svako dete ima svog roditelja, i sl.
- Provera poslovne logike. Ove provere testiraju da li podaci zadovoljavaju određeno poslovno pravilo (engl. *Business Rules*). Na primer, provera statusa neke kompanije koja je partnerska firma, ne može biti istovremeno i „*silver partner*” i „*gold partner*”, ili je jedno ili je drugo.

Niz definisanih provera se izvršava nad podacima i ukoliko se detektuje greška, ona se beleži u tabeli fakata za praćanje grešaka. Takođe se beleži i nivo oštećenja podataka. Ukoliko je nivo oštećenja podatka veoma veliki i iznad određene vrednosti, greške se smatraju fatalnim i mogu zaustaviti ceo ETL proces. Na primer, nedostaju kompletni podaci iz prodaje u nekoliko prodavnica tokom nekoliko dana ili su otkrivene nemoguće vrednosti u određenim važnim kolonama i sl., i ne postoje definisana pravila transformacije koja u ovim slučajevima možemo primeniti. Ukoliko nije došlo do fatalnih grešaka, ETL proces se normalno nastavlja. Da bi se postigle optimalne performanse, provere se mogu paralelizovati. Na primer, može se definisati za svaku proveru određeni redni broj provere; ukoliko neke provere imaju isti redni broj, izvršavaju se paralelno nad podatkom koji se trenutno proverava.

Ujednačavanje podataka

Nakon provere kvaliteta podataka, eventualno ćemo nad njim izvršiti određene transformacije kako bi podatak bio ujednačen i u skladu sa standardima definisanim za tržišta podataka. Na primer, ako smo za kolonu *pol* iz izvornog sistema učitali podatak M ili Ž, on predstavlja validnu vrednost, međutim nije u skladu sa standardom prema kome se očekuje da vrednost za *pol* bude Muški ili Ženski. Tada ćemo izvršiti konverziju iz M u Muški odnosno Ž u Ženski. Pod transformacijama se može podrazumevati i dodavanje određene vrednosti atributu, ukoliko on nije u odgovarajućem skupu definisanih vrednosti, ispravljanje zapisa adresa i imena, i sl.

Važna karakteristika sistema za transformaciju podataka je i da otkriva duplikate. Pod duplikatima podrazumevamo podatke koji se odnose na isti objekat, ali su dobijeni iz različitih izvornih sistema. Duplikati se mogu otkriti proverom vrednosti po određenim ključnim kolonama kao što su matični broj, i sl. Međutim, u nekim situacijama nije tako jednostavno utvrditi duplikate i postoje specijalizovani alati koji se bave otkrivanjem duplikata i njihovim uklanjanjem. Nakon detekcije, duplikati se moraju objediniti u jedan, najkvalitetniji podatak koji će biti u skladu sa definisanom politikom kvaliteta podataka i koji će dalje biti prosleđen korisničkom području.

5.4 Učitavanje podataka

Učitavanje podataka u tabele dimenzija

Retke su dimenzije koje su kreirane isključivo samim ETL procesom bez spoljnog izvora podataka. Uglavnom se tabele dimenzije popunjavaju podacima iz izvornih odnosno produkcionih sistema. Velike i kompleksne dimenzije, kao što su npr. dimenzija porizvođača, dobavljača ili proizvoda mogu imati svoje izvore podataka i u više različitih produkcionih sistema. Odgovarajući podaci dimenzija iz različitih izvora moraju biti korektno prepoznati, transformisani i učitani. Transformacija ovakvih podataka se odvija po principu koji je opisan u delu 5.3. Učitavanje se vrši na način opisan u delu o sporo promenljivim dimenzijama.

Učitavanje podataka u tabele fakata

Pre učitavanja podataka u tabele fakata potrebno je obaviti nekoliko pripremnih radnji kako bi učitavanje bilo što lakše i efikasnije, pogotovu, ako je u pitanju inicijalno punjenje tabela. Veliki uticaj na performanse učitavanja podataka vrše indeksi. Iako ubrzavaju izvršavanje upita, pri učitavanju drastično smanjuju performanse sistema. Preporaka je, da se pre učitavanja podataka, obavezno uklone svi indeksi i da se nakon učitavanja ponovo izgrade. Ako proces učitavanja podataka uključuje i ažuriranje podataka, te dve operacije bi trebalo razdvojiti i obavljati u sledećem redosledu:

1. Razdvojiti podatke za učitavanje (tj. one nad kojima će biti vršena insert naredba) od onih koji su potrebni za ažuriranje;
2. Odbaciti sve indekse osim onih koji su potrebni za ažuriranje;
3. Izvršiti ažuriranje;
4. Odbaciti sve preostale indekse;
5. Učitati podatke (koristeći neki alat za masovno učitavanje);
6. Izgraditi indekse.

Još jedna od strategija za poboljšanje performansi je da se velike tabele fakata fizički podele na više manjih tabela. Particionisanje tabela će biti skriveno od krajnjih korisnika. Najlogičniji način da se tabele fakata particionišu je po datumu, npr. na mesečnom nivou, i to po šifri datuma, ključu datumske dimenzije. Ako je tabela fakata particionisana po mesecima, a učitavanje podataka se vrši na mesečnom nivou, jasno je da će na taj način punjenje biti efikasno.

Ispravljanje grešaka u tabelama fakata

U praksi se dešava da se pogrešna merenja zabeleže u fakt tabele skladišta podataka. Postoje tri načina koja se primenjuju kako bi se ispravile greške merenja:

- Negiranje fakata. Pogrešno merenje se duplira, množi sa -1 i dodaje u tabelu fakata čime se u agregacijama poništava pogrešno merenje.
- Ažuriranje fakata. Operacija ažuriranja korišćenjem *UPDATE* komande može biti veoma skupa operacija u smislu zauzeća resursa zbog logovanja koje izvršava baza podataka radi oporavka od pada transakcije. U cilju poboljšanja performansi, treba koristiti alate za masovno učitavanje podataka i odbacivanje indeksa, kao što je već napomenuto.
- Brisanje fakata. U većini slučajeva, ovo je najbolji način za ispravljanje grešaka u tabelama fakata. Iako je teoretski brisanje fakata iz tabele fakata zabranjeno, brisanje fakata je u praksi uobičajena pojava. Brisanje se može obaviti na dva načina:

- Fizičko brisanje. Podaci se potpuno brišu iz sistema i više nisu dostupni.
- Logičko brisanje. Tabeli fakata se dodaje još jedan atribut pod nazivom Obrisano koji je obično tipa *Boolean* i služi za identifikaciju obrisanih fakata. Mana logičkog brisanja je ta što svaki upit mora da proverava ovaj podatak pri svom izvršavanju.

Učitavanje podataka u OLAP kocke

OLAP (engl. *Online Analytic Processing*) je sve popularnija komponenta skladišta podataka. Dve osnovne funkcije OLAP-a su:

- Performanse upita. OLAP serveri automatski upravljaju agregacijama i indeksima i takođe koriste specijalizovane strukture podataka.
- Analitička specijalizovanost. OLAP serveri koriste upitne jezike specijalizovane za kompleksne analize podataka. OLAP serveri takođe imaju mehanizme za skladištenje kompleksnih kalkulacija i mehanizme za zaštitu podataka na samom serveru.

Najbolji izvori podataka za OLAP kocke su, u svakom slučaju, dimenzione strukture podataka implementirane u relacionim bazama podataka. OLAP je pre svega dizajniran i predviđen za rad nad dimenzionim strukturama i podatke u OLAP kocke treba učitavati iz njih. Mapiranje vrednosti, odnosno merenja iz dimenzionog modela u OLAP kocke, skoro je 1–1, pri čemu dimenzije postaju ivice kocke. Učitavanje podataka u OLAP strukture podataka treba postaviti kao poslednju fazu ETL procesa [11].

5.5 Izbor arhitekture ETL sistema

Na kraju, pitanje koje se postavlja oko arhitekture ETL sistema je, da li koristiti gotove komercijalne alate za ETL ili razvijati sopstveni ETL sistem [1].

Prednosti komercijalnih ETL sistema su:

- Jednostavnost rukovanja i brži i jeftiniji razvoj. Inicijalna sredstva koja su potrošena za kupovinu alata biće vrlo brzo nadoknađena, pogotovu, na velikim projektima.
- Mogu ih koristiti i ljudi koji nisu profesionalni programeri.
- Većina komercijalnih alata automatski generiše većinu potrebnih metapodataka ETL procesa.
- Imaju ugrađene konektore za mnoge izvorne sisteme.
- Pružaju veoma dobre performanse.
- Dobro upravljaju raspoređivanjem podataka u multiserverskim okruženjima (engl. *Load-balancing*)
- Fleksibilnost alata koji su napravljeni modularno i takođe dopuštaju i mogućnost programiranja i dodavanja novih modula i funkcionalnosti.
- Razvijen sistem za dokumentovanje ETL procesa.

Prednosti samostalno razvijanih ETL sistema su:

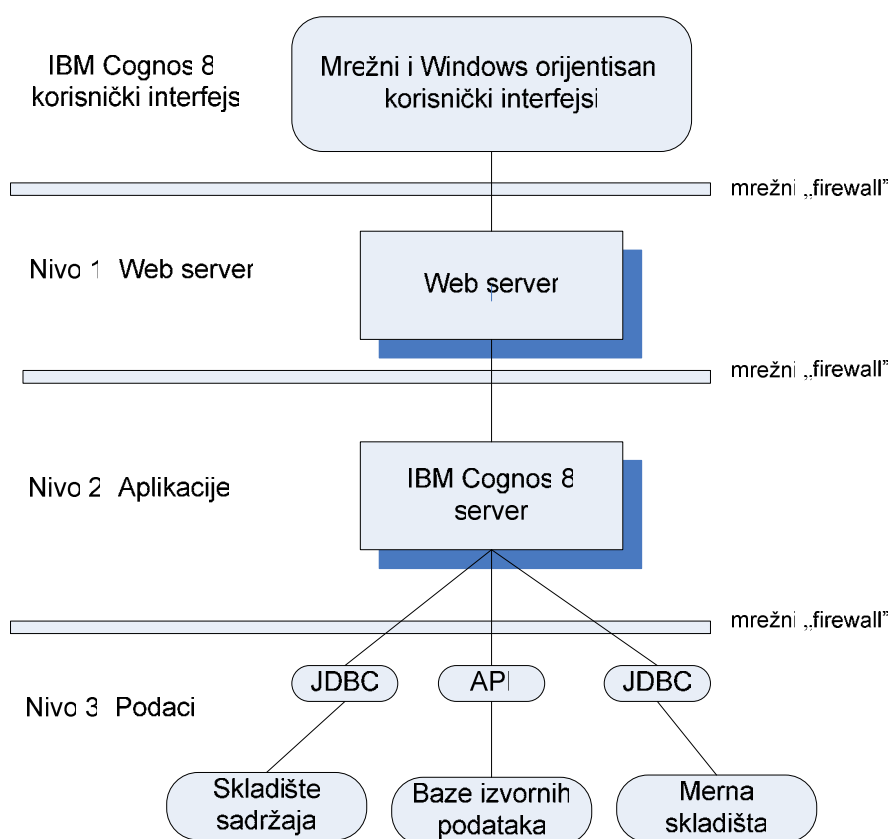
- Upotreba alata za automatsko testiranje izgrađenog ETL sistema.
- Fleksibilnost je potpuna i ne postoji zavisnost od proizvođača softvera.
- Slodoba u izboru struktura podataka potrebnih za ETL proces.
- Inicijalna ulaganja u razvoj ETL sistema su znatno manja.

Najvažniji parametar koji figuriše u izboru ETL sistema i o kome treba voditi računa jesu potrebe organizacije. Treba izabrati takav ETL sistem koji najbolje može da odgovori na potrebe poslovnih korisnika i da najefektnije pruži informacije koji su im potrebne.

II *IBM Cognos 8*

1. Arhitektura *Cognos 8* platforme

U prvom delu rada u tabeli 1, dat je pregled osnovnih korisničkih alata *Cognos*-a 8. Sada ćemo detaljnije opisati svaku od navedenih komponenti i predstaviti arhitekturu same platforme. Platforma je sastavljena iz tri nivoa prikazanih na slici 17: nivo Veb servera, nivo aplikacija i nivo podataka. Korisnički interfejs se nalazi iznad ova tri nivoa i obuhvata alate iz tabele 1.



Slika 17: Arhitektura *Cognos 8* platforme

IBM Cognos 8 korisnički interfejs

Metric Studio omogućava kreiranje okruženja, koje je korisniku prilagođeno, za nadgledanje i analize poslovnih merenja širom organizacije. Pomaže u prevođenju strategije organizacije u relevantne merljive ciljeve, čime se akcije i odgovornost zaposlenih usaglašavaju sa strateškim planovima. *Metric Studio* beleži performanse u odnosu na dogovorene ciljane vrednosti i pokazuje tekuće stanje – gde je organizacija uspešna, a gde treba da se poboljša, tako da donosioci odluke na svakom nivou organizacije mogu reagovati i isplanirati korake koje dalje treba preduzeti u poboljšanju poslovanja.

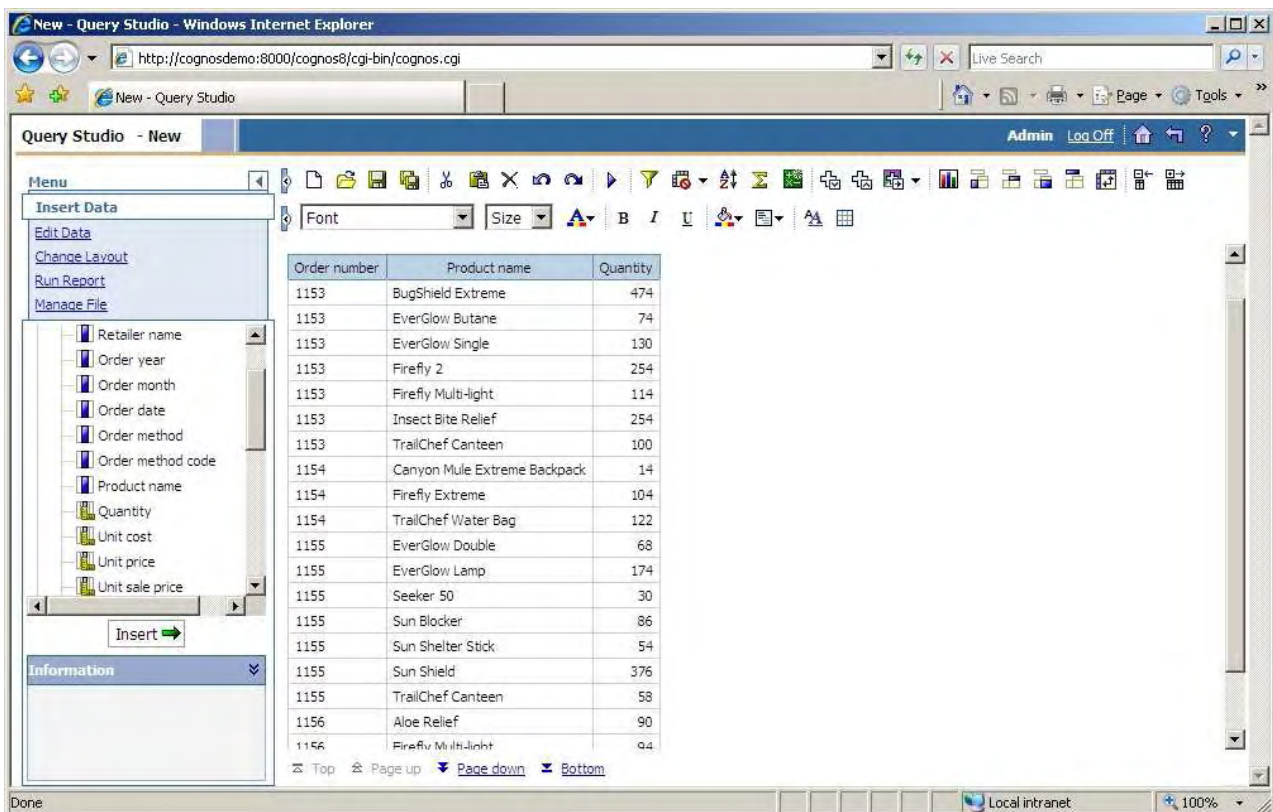
Kada se odrede ciljevi koji će biti mereni, *Metric Studio* se koristi radi organizovanja merenja u karte rezultata (engl. *Scorecards*) koje korisnicima omogućavaju da vide željene pokazatelje performansi u odgovarajućem kontekstu. Pošto se različite organizacione celine preduzeća mogu fokusirati na različite mere, u alatu *Metric Studio* se mogu implementirati različite karte rezultata za različite poslovne korisnike. Takođe, karte rezultata možemo urediti hijerarhijski, npr. tako da rukovodstvo preduzeća vidi samo jednu kartu rezultata sa pokazateljima performansi poslovanja celokupnog preduzeća, dok pojedinačne organizacione jedinice mogu imati nekoliko karata rezultata sa merenjima za njihove konkretne ciljeve.

Kada se završi sa planiranjem i organizovanjem mera, kreira se *Metric Studio* aplikacija. Ova aplikacija je skup karata rezultata, mera i projekata koji se koriste za praćenje performansi preduzeća. Aplikacija se kreira u dva koraka. Prvo se kreira merno skladište (engl. *Metric Store*), a zatim se kreira odgovarajući merni paket (engl. *Metric Package*). Merno skladište sadrži podatke za merenja i karte rezultata, kao i odgovarajuća podešavanja samog alata. Merno skladište koristi relacionu bazu podataka za smeštanje sadržaja, a podržane baze su *Oracle*, *SQL Server* i *DB2* baza podataka. Kreiranje mernog paketa će detaljnije biti izloženo u delu 2.3 rada koji opisuje alat *Metric Designer*.

Nakon podešavanja paketa, sledeći korak je učitavanje podataka u *Metric Studio*. *Metric Studio* sadrži prihvatno područje u koje se učitavaju podaci pre nego što se prebace u merno skladište. Podaci se u prihvatno područje mogu prebaciti pomoću alata *Metric Designer* ili nekog drugog ETL alata. Iz prihvatnog područja podaci se prebacuju u skladište pomoću *Cognos Connection* portala. Da bismo završili proces podešavanja alata *Metric Studio*, potrebno je još podesiti odgovarajuća prava pristupa korisnicima.

Query Studio se koristi za jednostavno i intuitivno izvršavanje različitih upita nad podacima. Rezultati upita su izveštaji u okviru kojih mogu postojati filtrirani i sortirani podaci, formatirana polja, grafici, itd. Na slici 18 prikazan je *Query Studio*. Upiti se kreiraju prostim prevlačenjem stavki upita iz predmeta upita i mera koje se nalaze sa leve strane u prostor za izveštavanje sa desne strane. Pojmovi – stavka upita i predmet upita su objašnjeni u poglavlju 2.2 *Framework Manager* – alat za modelovanje podataka.

Da bi bilo moguće kreirati izveštaje, prethodno je potrebno kreirati odgovarajuće pakete korišćenjem alata *Framework Manager*, o kome će više biti rečeno u poglavlju 2.2 *Framework Manager*. Paket je takođe potrebno publikovati na *Cognos Connection* portalu, kao i dodeliti korisnicima odgovarajuća prava pristupa kako paketu, tako i samim funkcionalnostima koje pruža *Query Studio* alat.



Slika 18: Postavljanje upita kroz Query Studio

Analysis Studio je komponenta *Cognos*-a 8 namenjena multidimenzionoj analizi, nazvanoj još i OLAP analiza ili analitička obrada podataka u realnom vremenu. Karakteriše je velika moć obrade velike količine podataka pri interaktivnom korišćenju. Analize su namenjene korisnicima koji ne zanju tačno šta konkretno traže. Oni obično postavljaju niz pitanja odnosno upita sistemu, i svaki novi upit zavisi od odgovora dobijenog na prethodni. Niz odgovora koji može da dâ analiza se ne može dobiti pomoću izveštaja. Namena izveštaja je da dâ odgovor na konkretno pitanje i u tome se razlikuje analiza od izveštaja. Alat *Analysis Studio* je upravo namenjen analizi.

Report Studio predstavlja veoma razvijeno okruženje u kome profesionalni autori izveštaja i programeri projektuju sofisticirane izveštaje. U njemu se prave šabloni izveštaja, takozvani predefinisani izveštaji, na osnovu kojih se nad konkretnim podacima generišu korisnički izveštaji. Važno je naglasiti da *Report Studio* podržava i analitičke, tj. izveštaje poslovne inteligencije, poput raznih grafičkih prikaza, sumiranja rezultata, prikaza trendova i slično, ali i radne, tekuće ili transakcione izveštaje, poput faktura, spiskova plata itd.

Event Studio čine alati za obaveštavanje donosilaca odluka u organizaciji o događajima u organizaciji. Pomoću njega se prave agenti koji nadgledaju status promena, akcije ključnih kupaca, ili bilo kakav drugi za organizaciju bitan događaj. Kada detektuje događaj, agent izvrši odgovarajući zadatak, kao što je slanje elektronske pošte, dodavanje informacija na portal ili pokretanje izveštaja.

Opisani alati spadaju u mrežno-orijentisane alate, jer im se pristupa preko internet pretraživača i *Cognos Connection* mrežnog portala. Pored mrežnih, postoje i *Windows* orijentisani alati i to su: *Data Manager*, *Framework Manager*, *Metric Designer*, *Transformer* i *Map Manager*.

Framework Manager je alat za integrisanje organizacionih podataka. U ovom alatu se definiše jedinstven pristup organizacionim podacima koji se mogu nalaziti u različitim produkcionim

bazama, skladištima podataka, hiperkockama, tabelama, datotekama i drugim izvorima. Pristup se definiše modelom podataka i to u nekoliko slojeva, kao što su sloj pogleda na izvorne podatke, sloj normalizovane šeme podataka, sloj denormalizovane, multidimenzione šeme podataka, sloj namenskih pogleda, sloj korisničkih celina. U okviru svakog sloja nalaze se nazivi podataka kojima se pristupa, putanje do izvornih podataka, zatim način formatiranja podataka, prava pristupa i slični metapodaci. Takođe, definisani su odnosi podataka kojima se pristupa, poput relacione kardinalnosti ili, što je posebno važno kod multidimenzione analize, hijerarhijskog uređenja. Za krajnje korisnike metapodaci se pomoću *Framework Manager*-a grupišu u korisničke celine koje se nazivaju paketi. Paketi se obično formiraju po funkcijama organizacije, poput proizvodnje, finansija ili marketinga. *Framework Manager* će još detaljnije biti opisan u delu 2.2.

Polazeći od ovako definisanog pristupa podacima, u alatu *Metric Designer* se mogu odrediti podaci koji će se nadgledati i način na koji će se računati izvedeni merni pokazatelji. Paket koji je spreman za upotrebu se objavljuje, to jest, omogućava se pristup izvornim podacima iz portala *Cognos Connection* i komponentama tipa *Viewer* i *Studio*.

Nivo 1: Veb server

Nivo veb servera [2] sadrži bar jedan program koji se naziva *IBM Cognos 8* kapija (engl. *IBM Cognos 8 gateway*). Kapija je ekstenzija veb servera i svrha joj je da analizira, obrađuje, šifrjuje i prosleđuje informacije sa veb servera ka drugim serverima, odnosno sa *Cognos Connection* portala ka aplikativnom nivou tj. *IBM Cognos 8* serveru. Mrežna komunikacija se može ostvariti i direktno sa dispečerom, koji pripada aplikativnom nivou i koristi se za pokretanje *Cognos* servisa, zaobilazeći kapiju, ali to nije uobičajeno.

Cognos 8 podržava nekoliko tipova kapija i to su:

- *CGI*⁵ (skraćeno od Common Gateway Interface) – može se koristiti za sve podržane veb servere i predstavlja standard, premda nije dovoljno brz;
- *ISAPI*⁶ (skraćeno od Internet Server Application Programming Interface) – predstavlja *Microsoft*-ovu tehnologiju i koristi se za *Microsoft IIS* (skraćeno od Internet Information Services) veb server;
- *Apache_mod*, koristi se za *Apache Web Server*;
- *Servlet* – ako instalirani veb server podržava servlete, može se koristiti ova kapija ili ako se umesto veb servera koristi aplikativni server.

Tipičan posao koji obavlja kapija kada primi zahtev je da:

- Enkriptuje lozinke da bi osigurala bezbednost;
- Pripremi informacije potrebne *Cognos 8* serveru kome prosleđuje zahtev;
- Pridruži vrednosti promenljivih okruženja samog veb servera;
- Prosledi zahtev *IBM Cognos 8* dispečeru na dalju obradu.

Nivo 2: IBM Cognos 8 server

Nivo 2 ili aplikativni nivo se sastoji iz jednog ili više fizičkih servera na kojima je instaliran softver *IBM Cognos 8* server. Operativni sistemi koje server podržava su: IBM AIX, HP-UX, Windows Server 2008, Windows Server 2003, Windows XP, SUSE Enterprise Linux, Red Hat

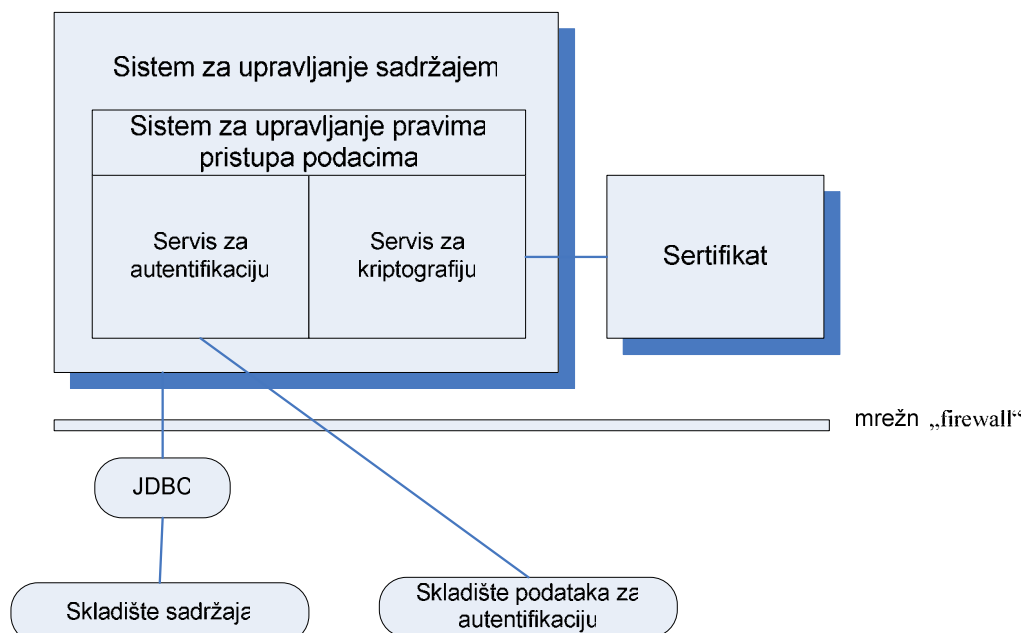
⁵ <http://www.w3.org/CGI/>

⁶ http://en.wikipedia.org/wiki/Internet_Server_Application_Programming_Interface

Enterprise Linux, Sun Solaris⁷. Ovaj nivo prima zahteve preko dispečera, koje prosleđuje kapija, i izvršava ih. Zahtevi mogu biti, na primer, pokretanje upita, izveštaja, analiza i sl. Pojedinačni *Cognos 8* server se sastoji od dispečera, sistema za upravljanje sadržajem (engl. *Content Manager*) i odgovarajućih servisa.

Dispečer, koji je prihvatio zahtev sa kapije, dalje, preko određenih protokola, aktivira instance drugih servisa, poput servisa za proveru prava pristupa, kreiranje izveštaja, prikazivanje u određenom formatu, pokretanje drugih dispečera i sl. Dispečer podržava i višenitni rad, tj. može da koristi jednu ili više niti da izvrši zahtev. Kada se dispečer startuje, on se registruje na sistem za upravljanje sadržajem čime se obezbeđuje da i ostali dispečeri u sistemu imaju informaciju da je pokrenut. Ukoliko postoji više dispečera na sistemu oni među sobom raspodeljuju poslove koristeći „*round-robin*” algoritam.

Sistem za upravljanje sadržajem, prikazan na slici 19, jeste program koji upravlja skladištenjem korisničkih podataka kao što su: konfiguracioni podaci servera, korisnički podaci, paketi metrika, izveštaji i specifikacije izveštaja, rezultati upita i sl. Sistem za upravljanje sadržajem skladišti informacije u skladište sadržaja koje se nalazi na nivou 3 arhitekture sistema.



Slika 19: Sistem za upravljanje sadržajem

Instalacija *Cognos*-a može da sadrži više sistema za upravljanje sadržajem, svaki na različitom računaru. U ovakvoj konfiguraciji samo jedan sistem za upravljanje sadržajem može biti aktivan, ostali instalirani sistemi su pasivni i služe kao rezerva za aktivni sistem. Ukoliko aktivni sistem za upravljanje sadržajem iz nekog razloga nije dostupan, aktivira se pasivni sistem, koji sada postaje aktivan i zahtevi se preusmeravaju na njega. Kada aktivni sistem za upravljanje sadržajem padne, može doći do gubljenja nesačuvanih korisničkih podataka i do traženja od korisnika da se ponovo uloguje.

Sistem za upravljanje sadržajem sadrži sistem za upravljanje pravima pristupa podacima (engl. *Access Manager*), u daljem tekstu AM. AM predstavlja osnovnu bezbednosnu komponentu *Cognos*

⁷ Detaljan spisak podržanih operativnih sistema se može videti na adresi: <http://www-01.ibm.com/support/docview.wss?uid=swg27014110>

platforme. AM pruža određene bezbednosne funkcije, kao što su autorizacija, autentifikacija i enkripcija. Autentifikacija korisnika se obavlja preko eksternih sistema za autentifikaciju, odnosno AM predstavlja interfejs između *Cognos* platforme i spoljnih aplikacija za autentifikaciju, kao što je npr. *Microsoft Active Directory*. AM takođe poseduje i sopstveni prostor imena (engl. *Namespace*) za definisanje sopstvenih grupa i korisničkih uloga nezavisno od spoljnih sistema za autentifikaciju. Kombinacijom ovih mogućnosti se dobija veoma fleksibilan sistem za upravljanje bezbednošću i pravima pristupa.

Nivo 3: Podaci

Nivo podataka se sastoji od:

- Skladišta sadržaja;
- Baza izvornih podataka i
- Mernih skladišta.

Skladište sadržaja je relacionala baza podataka koja sadrži podatke neophodne za funkcionisanje *Cognos 8* platforme. Sama *Cognos* platforma nudi podrazumevanu bazu za skladište sadržaja koja se naziva *IBM Cognos Content Database*. Ali umesto nje, može se upotrebiti i *Microsoft SQL Server*, *Oracle*, *IBM DB2* ili *Sybase*. Kao što je već rečeno u tekstu koji opisuje nivo 2, servis koji koristi skladište sadržaja je sistem za upravljanje sadržajem. Ovaj servis koristi *JDBC*⁸ (skraćeno od *JAVA Database Connectivity*) interfejs da bi pristupio skladištu sadržaja i mernom skladištu. *Cognos* platforma sadrži *JDBC Driver* za podrazumevanu *IBM Cognos Content Database* bazu podataka, kao i za *MS SQL Server*. Za baze podataka ostalih proizvođača se mora nabaviti odgovarajući *JDBC Driver*.

Merno skladište je relacionala baza podataka koja sadrži merne pakete, kao i korisnička podešavanja vezana za alat *Metric Studio*. Može postojati i više od jednog mernog skladišta. Na primer, jedno merno skladište može sadržati podatke vezane za prodaju, a drugo može sadržati podatke vezane za finansije.

Komponente *Cognos* platforme komuniciraju međusobno koristeći magistralu koja se naziva BI magistrala (engl. *BI Bus*). Ovu magistralu takođe moraju koristiti i aplikacije razvijane van *Cognos* platforme, ali čiji je cilj integracija sa *Cognos*-om. Magistrala je zapravo *SOAP*⁹ (engl. *Simple Object Access Protocol*) API (engl. *Application Programming Interface*) i podržava *WSDL*¹⁰ (engl. *Web Services Definition Language*). BI magistrala je skup pravila odnosno protokola koji definišu komunikaciju među *Cognos 8* servisima. Takođe definišu kako se industrijski standardi i protokoli upotrebljavaju sa *Cognos 8* platformom.

2. Windows orijentisane komponente *Cognos 8* platforme

U ovom delu ćemo prezentovati *Windows* orijentisane komponente *Data Manager* za kreiranje skladišta podataka, *Framework Manager* za upravljanje i distribuiranje podataka krajnjim korisnicima, kao i *Metric Designer*, alat za kreiranje karti rezultata. Ovi alati spadaju u administratorske alate i krajnjim korisnicima ne bi trebalo da budu dostupni.

⁸ <http://www.oracle.com/technetwork/java/overview-141217.html>

⁹ <http://www.w3.org/TR/soap>

¹⁰ <http://www.w3.org/TR/wsdl>

2.1 Data Manager – alat za kreiranje skladišta podataka

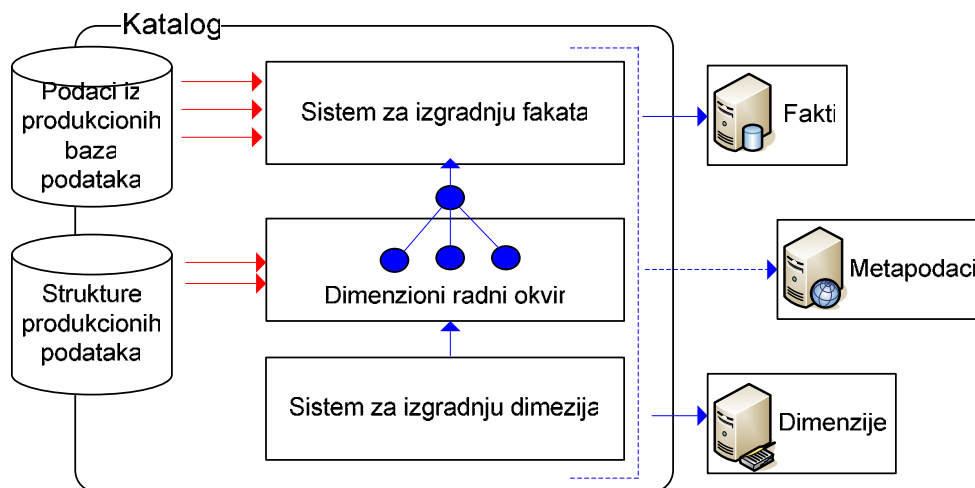
Data Manager [3] je alat kome je namena da kreira skladišta podataka i repozitorijume potrebne za izveštavanje, analize i upravljanje performansama. *Data Manager* obavlja sledeće funkcije, takođe prikazane na slici 20.

- Ekstrahuje podatke iz različitih produkcionih izvora podataka;
- Vršiti transformacije nad podacima da bi se zadovoljile poslovne potrebe za analizom podataka i izveštavanjem;
- Isporučuje transformisane podatke u usklađene dimenzije.

Sama izgradnja skladišta podataka počinje izgradnjom i definisanjem dimenzija kroz Sistem za izgradnju dimenzija i Dimenzioni radni okvir. Kad se završi izgradnja dimenzija prelazi se na izgradnju tabela fakata pomoću Sistema za izgradnju fakata.

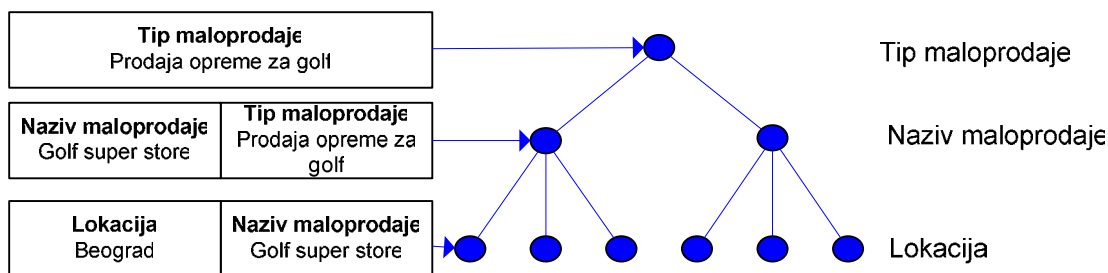
Data Manager može rezultate svog rada isporučiti u bilo koju podržanu bazu podataka (npr. *Microsoft SQL Server*, *IBM DB2* ili *Oracle Database*). Međutim, preporuka je da se podaci isporučuju u dimenziono skladište podataka, kao što je i namena alata, sa usklađenim dimenzijama. *Data Manager* je fleksibilan alat koji omogućuje da se i završeni projekti, nad kojim je sa *Framework Manager*-om izgrađen model, mogu ponovo upotrebiti, dopuniti i proširiti po potrebi.

Data Manager katalog je objekat koji predstavlja repozitorijum za podatke koji definišu način na koji *Data Manager* ekstrahuje, transformiše i isporučuje produkcione podatke. Katalog može biti smešten u bilo koju bazu podataka određenu za potrebe kataloga.



Slika 20: Tok podataka u *Data Manager*

Dimenzioni radni okvir podataka definiše hijerarhijsku strukturu poslovnih elemenata, odnosno hijerarhijske veze poslovanja. Predstavlja način na koji organizacija shvata podatke, nezavisno za izvor podataka. Svaka dimenzija predstavlja skup povezanih podataka o određenom vidu poslovanja i omogućuje kontekst za analizu, dok hijerarhija predstavlja određeni pogled na dimenziju. Hijerarhija može imati više nivoa, npr. tip maloprodaje, naziv maloprodaje i lokacija prodavnice (videti sliku 21). Ukoliko je neophodno, svaki nivo hijerarhije se može izdvojiti u zasebnu poddimenziju, čime se kreira pahuljasta shema.



Slika 21: Primer hijerarhijske organizacije

Sistem za izgradnju dimenzija definiše usklađene dimenzije u odnosu na dimenzioni radni okvir, učitava podatke iz produkcionih struktura koje smo odredili i upisuje ih u definisane tabele dimenzija. Ovaj sistem je takođe neophodan za praćenje promena nad atributima. Da bi mogao da prati promene atributa, sistem mora da ima uvid u tekuće i istorijske podatke svakog atributa, svih dimenzija čije promene se prate. Međutim, mala je verovatnoća da će transakcioni sistem iz koga se učitavaju podaci sadržati sve potrebne istorijske podatke. Takođe, transakcioni sistem nema podataka o nedeskriptivnim ključevima koji se koriste u skladištu podataka. *Data Manager* može ove podatke dobiti samo iz dimenzionih tabela i tek nakon izgradnje dimenzija se gradi struktura koja će omogućiti *Data Manager* alatu da prati promene nad atributima i nedeskriptivnim ključevima skladišta podataka. Shodno tome, celokupan proces izgradnje dimenzija se odvija u dve faze.

Sistem za izgradnju fakata definiše kako će se podaci ekstrahovati, transformisati i isporučivati. Pošto podaci mogu poticati iz više različitih izvora, zadatak ovog sistema je da se sa njima poveže, da prikupi i pročisti podatke, standardizuje ih u odnosu na dimenzioni radni okvir i isporuči u odgovarajuće tabele.

2.2 *Framework Manager* – alat za modelovanje podataka

Framework Manager [5] je alat kome je namena da kreira model podataka koji će na adekvatan način da poveže izvorne podatke (engl. *Source Data*) sa korisnicima *Cognos*-a. Model podataka je kolekcija metapodataka koja uključuje fizičke i poslovne informacije iz jednog ili više izvora podataka koje tim informacijama daju određeno značenje. Izvore podataka mogu predstavljati normalizovani i denormalizovani podaci (npr. ako je u relacionoj bazi implementiran dimenzioni model) iz relacionih baza podataka, kao i razne OLAP strukture. Takođe, izvor podataka za *Framework Manager* može biti skladište podataka kreirano alatom *Data Manager*. Kada se model podataka kreira, on se distribuira krajnjim korisnicima kao paket. Ove distribucije nazivamo paketima zato što možemo napraviti više paketa od istog modela. Takođe, paketi mogu sadržati samo pojedine delove modela, a ne i ceo model.

Model podataka omogućava odgovarajuće poslovne poglede (engl. *Business View*) na podatke radi što lakšeg kreiranja izveštaja, analiza i upita. Ovim pogledom mogu da se:

- Podaci smeštaju u foldere koji predstavljaju poslovne oblasti za izveštavanje;
- Formatiraju podaci koristeći različite numeričke formate, kao i formate za datum, vreme i ostale formate;
- Podaci predstave korisnicima na različitim jezicima;
- Automatizuju SQL upiti, ukoliko je potrebno dobiti određene podatke iz relacione baze podataka.

Da bi se kreirao model nad relacionom bazom podataka, potrebno je izvršiti sledeće korake [16]:

- Kreirati konekciju ka relacionoj bazi podataka (ili iskoristiti već postojeću);
- Kreirati predmete upita (engl. *Query Subject*) nad izvorima podataka. Predmet upita je skup stavki upita (engl. *Query Item*) koje su u određenoj vezi. U većini slučajeva ponašaju se kao tabele. Stavka upita je najmanji deo modela koji se može postaviti u izveštaj i predstavlja jedinstvenu karakteristiku nečega, npr. datum rođenja. Za svaku tabelu i pogled koji su uvezeni u model iz relacione baze podataka, *Framework Manager* automatski kreira predmete upita;
- Kreirati predmete upita modela (engl. *Model Query Subject*). Predmeti upita modela se ne kreiraju direktno iz izvora podataka, već su zasnovani na stavkama upita iz drugih predmeta upita, uključujući i druge predmete upita modela;
- Postaviti svojstva stavki upita. Stavke upita su veoma važne za krajnjeg korisnika, jer će ih on koristiti da kreira izveštaje. Oni takođe koriste svojstva stavki upita da bi izgradili izveštaje;
- Definisati spajanja među predmetima upita;
- Kreirati pakete koji će biti prezentovani krajnjim korisnicima;
- Objaviti pakete.

Pored navedenih osnovnih koraka, može se još definisati bezbednost modela, uraditi implementacija na više jezika i sl.

Nakon pokretanja *Framework Manager* aplikacije, pokreće se čarobnjak metapodataka (engl. *Metadata Wizard*). Ovaj čarobnjak omogućava da se na lak i efikasan način kreira novi projekat. Takođe, ovaj čarobnjak se koristi i da bi se odredili izvori podataka koji će biti korišćeni u modelu koji se dizajnira, kao i način na koji će *Cognos* pristupati podacima i sa kojim pravima. Najbolje je za pristup koristiti nalog baze podataka koji ima isključivo pravo da čita podatke, kako korisnici *Cognos*-a ne bi na bilo koji način promenili ili obrisali izvorne podatke.

Pošto smo, koristeći čarobnjaka, definisali konekcije ka relacionim bazama podataka, ostaje nam da izaberemo koje objekte iz baze ćemo uključiti u model. Objekti mogu biti tabele, pogledi, sinonimi, procedure, funkcije i sl. Pored izbora samih objekata koji će učestvovati u modelu, potrebno je da definišemo i odnose među njima. Sam čarobnjak nudi pomoć dizajneru tako što će odrediti neke odnose umesto njega, na osnovu stranih i primarnih ključeva, koji će kasnije biti korišćeni u modelu za definisanje spajanja (engl. *Join*) među tabelama. Posle koraka definisanja odnosa, čarobnjak prelazi na njihovo kreiranje i završava sa radom.

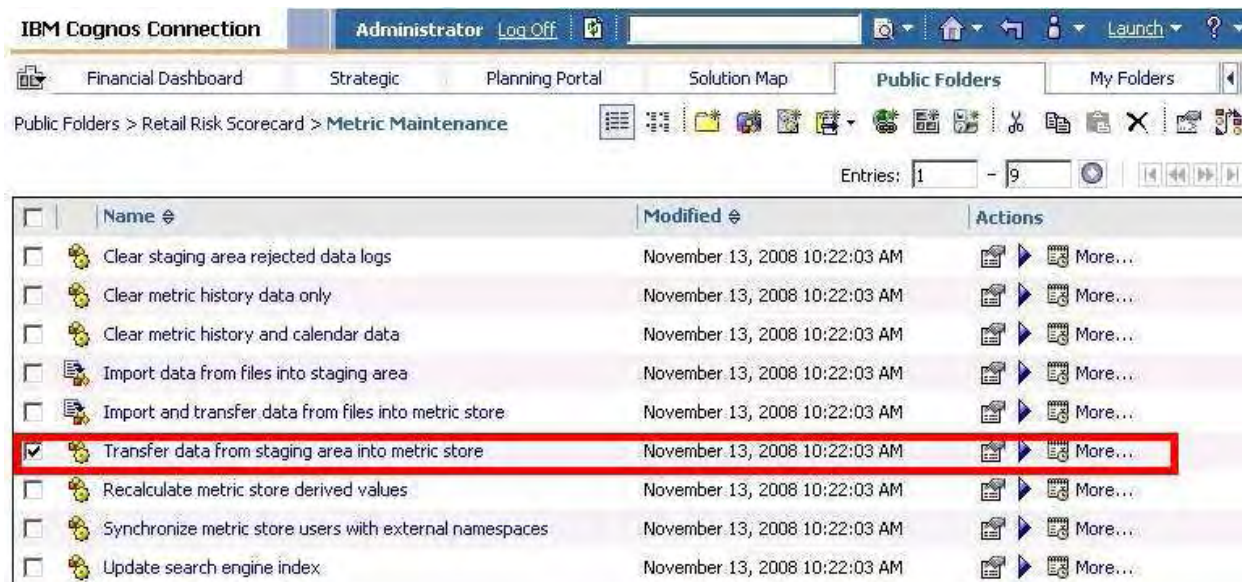
Framework Manager korisnicima omogućava veoma fleksibilnu radnu površinu pogodnu za istraživanje izvora podataka, uočavanje odnosa među podacima, koja se prilagođava samom izvoru podataka. Postojeće razlike u prikazu, ako se kao izvor podataka za projekat koristi relaciona baza podataka u odnosu na skladište podatka ili OLAP izvor podataka. Na slici 22 prikazan je deo dijagrama dimenzionog izvora podatka. U sredini su tabele fakata, okružene tabelama dimenzija.

- Jedinica mere – može biti valuta, procenat ili različite korisnički definisane jedinice;
- Težina mere – definiše koliko mera ima uticaja na izvedene mere. Izvedene mere su mere zasnovane na prethodno definisanim merama;
- Kalkulacija – određuje da li će se vrednosti mere učitavati sa neke lokacije, ili će se dobijati računskim operacijama ili određenom kombinacijom i jednog i drugog.

Takođe, osobine za tip mere koje moramo definisati pored konkretnih mera su i ciljna vrednost i dozvoljeno odstupanje, jer ciljna vrednost će nam biti reper u odnosu na koju ćemo utvrđivati da li smo postigli dobre rezultate ili ne.

Nakon pokretanja, čarobnjak od nas očekuje da kreiramo novi projekat ili otvorimo postojeći. U sledećem koraku, traži se da unesemo referencu na merni paket ili da kreiramo novi. Da bi ekstrahovali merenja i koristili ih u alatu *Metric Studio*, moramo imati prethodno kreiran bar jedan merni paket. Mernim paketima pristupamo kroz alat *Metric Studio* i u njima se nalaze definisani tipovi mera i karte rezultata. Posle definisanja mernog paketa koji ćemo koristiti, u nekoliko narednih koraka, čarobnjak nas vodi kroz proces identifikovanja mera koje će biti ekstrahovane i učitane u merno skladište, kao i kroz proces dizajniranja i povezivanja karti rezultata sa odgovarajućim merama. Kroz ove korake takođe mapiramo vremenske periode, određujemo podrazumevane jedinice mera i opciono filtriramo podatke. Po završetku čarobnjaka, možemo pokrenuti kompletan proces ekstrahovanja, povezivanja i učitavanja koji smo upravo definisali.

Nakon ekstrahovanja podataka, sledeći korak je transfer podataka u *Metric Studio* i ažuriranje karti rezultata. Ove korake obavlja komponenta alata *Metric Studio* koja se naziva alat za održavanje mera (engl. *Metric Maintenance*). Odatle ćemo pokrenuti sistem za transfer podataka, kao što je prikazano na slici 23.



Slika 23: Opcije koje nudi alat za održavanje merenja

Nakon transfera, imamo spremne karte rezultata koje možemo pregledati kroz *Metric Studio*.

U ovom delu su bile izložene osnovne korisničke funkcionalnosti *IBM Cognos 8* platforme, osnove arhitekture sistema, a takođe su prikazane i osnove naprednijih administratorskih alata. Pored navedenih osnovnih funkcionalnosti, platforma nudi i različite druge mogućnosti i pakete, kao što su integracija sa paketom *Microsoft Office*, *IBM Cognos 8 Go! Mobile* paket za integraciju

Cognos-a i mobilnih uređaja tipa *BlackBerry*, *Cognos TMI for Planning & Analysis* paket za poslovno planiranje i planiranje budžeta i mnogi drugi paketi. Na nivou arhitekture, platforma je izuzetno fleksibilna i skalabilna. Kompletnu instalaciju platforme je moguće uraditi na samo jednom računaru, a isto tako i izvršiti instalaciju na čitavim farmama i klasterima servera sa implementiranom visokom dostupnošću (engl. *High-availability*) i opravkom od havarije (engl. *Disaster Recovery*). Na nivou podataka, *Cognos* omogućava korisnicima kompletan i integrisan uvid u podatke koje imaju u svim svojim produkcionim sistemima radi praćenja poslovanja, analiza i podrške u odlučivanju i vođenju preduzeća.

III Modul ljudskih resursa poslovnog informacionog sistema

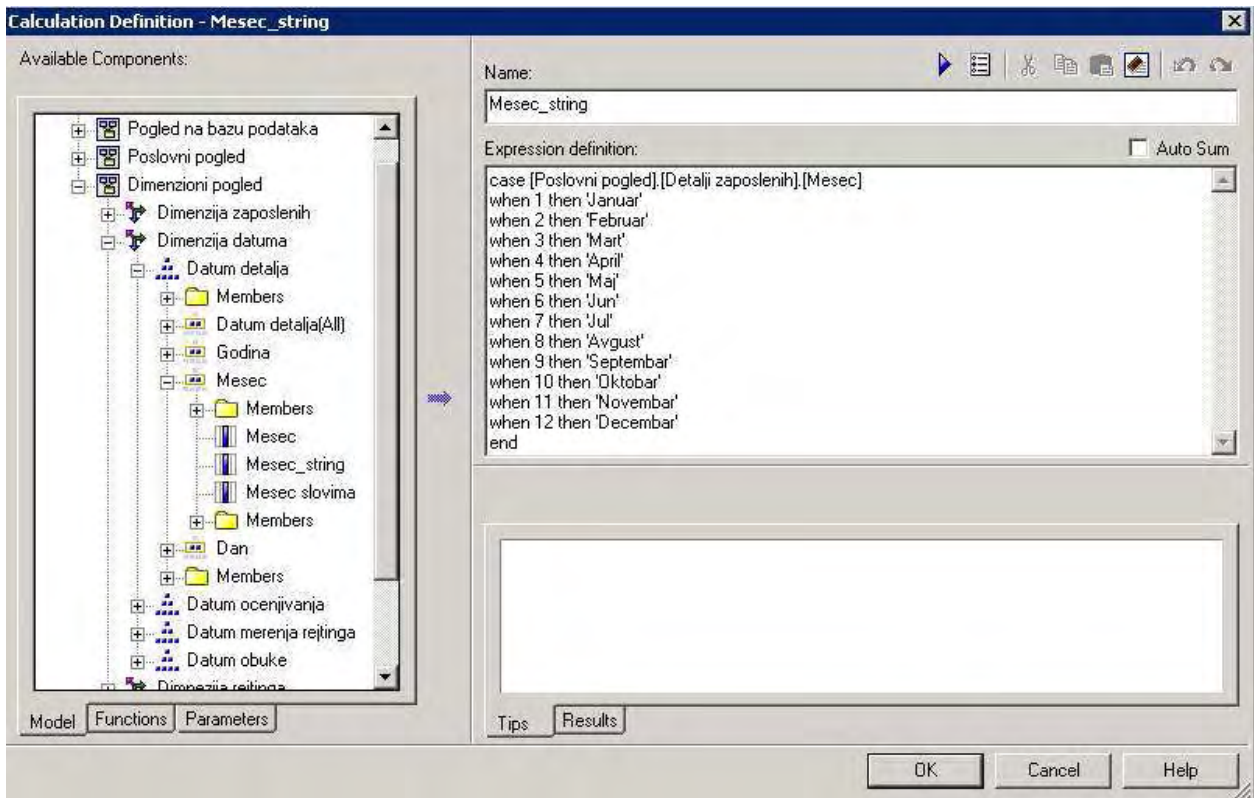
Za odeljenje ljudskih resursa (engl. *Human Resources*, skraćeno HR) je najvažnije da zaposli i zadrži kvalitetno osoblje u preduzeću, kao i da posreduje u rešavanju konflikata između zaposlenih ili zaposlenih i menadžmenta. HR odeljenje takođe mora da zna da odredi broj ljudi potreban firmi, kao i njihov profesionalni profil, da ih zaposli, ali i da nađe adekvatne zamene za ljude koji su otišli iz preduzeća. HR organizuje obuke i vežbe za zaposlene radi povećanja produktivnosti i efikasnosti zaposlenih u cilju unapređenja celokupnog poslovanja preduzeća.

Osnovni pokazatelji koje HR odeljenje prati i koji ujedno pokazuju uspešnost samog odeljenja su:

- koliko dugo je zaposleni ostao u firmi, odnosno koliko je zaposleni zadovoljan firmom;
- porast broja zaposlenih;
- veštine i iskustvo zaposlenih;
- rejting zaposlenih, odnosno kako se zaposleni rangira u firmi.

Poslovni korisnici iz HR odeljenja će na osnovu izgrađenog modela moći na lak način da dođu do različitih odgovora na pitanja koja ih zanimaju i da utvrde, na primer, da li je su zaposleni zadovoljni ili nezadovoljni, kao i čime su konkretno nezadovoljni; da li se njihovo zadovoljstvo menja sa porastom plate ili ne; da li popularnost ili rejting zaposlenog ima veze sa platom ili njegovom zadovoljnošću, obukama, radnim mestom ili položajem, i sl.

Model za potrebe HR odeljenja u ovom primeru je izgrađen u alatu *Framework Manager* i sastoji se iz tri pogleda i to su: Pogled na bazu podataka, Poslovni pogled i Dimenzioni pogled. Pogled na bazu podataka predstavlja podskup baze podataka i predstavlja nam osnov za izgradnju Poslovnog pogleda. U Pogledu na bazu podataka smo izdvojili samo one subjekte upita i stavke upita koje su nam relevantne za dalju izgradnju modela. Poslovni pogled modula HR je prikazan na slici 25. Poslovni pogled izgrađujemo kreiranjem novih subjekata upita i stavki upita iz postojećih iz Pogleda na bazu podataka. Takođe, stavkama upita dodajemo, ako je potrebno, odgovarajuća formatiranja, kao što su format datuma ili format zapisa broja, i sl. Kreiranje nove stavke upita prikazano je na slici 24.



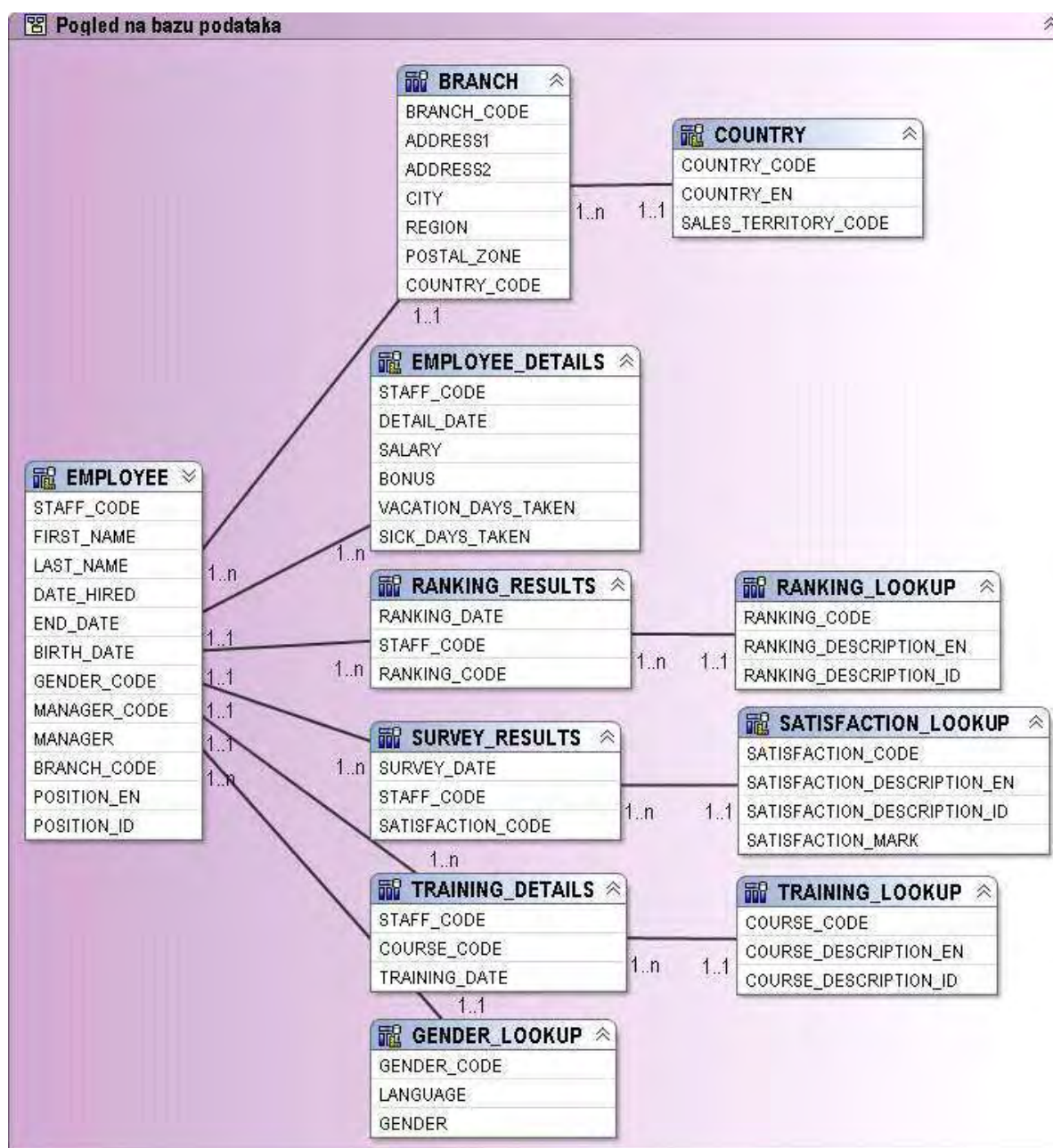
Slika 24: Kreiranje nove stavke upita



Slika 25: Poslovni pogled na bazu ljudskih resursa

U tabeli Informacije o zaposlenima date su osnovne lične informacije o svakom zaposlenom u firmi, kao i podaci o lokaciji na kojoj se nalazi njegovo radno mesto. U tabeli Detalji zaposlenih prikazane su informacije o zaposlenima koje su zapravo merni fakti, kao što su plata, bonus i sl. U dimenzionom pogledu, tabela Informacije o zaposlenima bi bila osnov za izgradnju tabele dimenzije, dok bi tabela Detalji zaposlenih bila osnov za izgradnju tabele fakata. Tabela Rejting zaposlenih se odnosi na popularnost zaposlenih u firmi. Tabela Ocene zaposlenih se odnosi na zadovoljstvo zaposlenih firmom i rukovodstvom. Tabela Obuka zaposlenih vodi evidenciju o održanim obukama i vežbama zaposlenih. Sve četiri poslednje navedene tabele sadrže i datum kad je izvršeno merenje, kao i attribute, odnosno stavke upita koje predstavljaju komponente datuma. Ove komponente ćemo dalje koristiti pri kreiranju dimenzionog pogleda kao osnov za kreiranje datumske dimenzije i odgovarajuće hijerarhije.

Sama baza podataka koja se nalazi ispod poslovnog modela data je na slici 26.

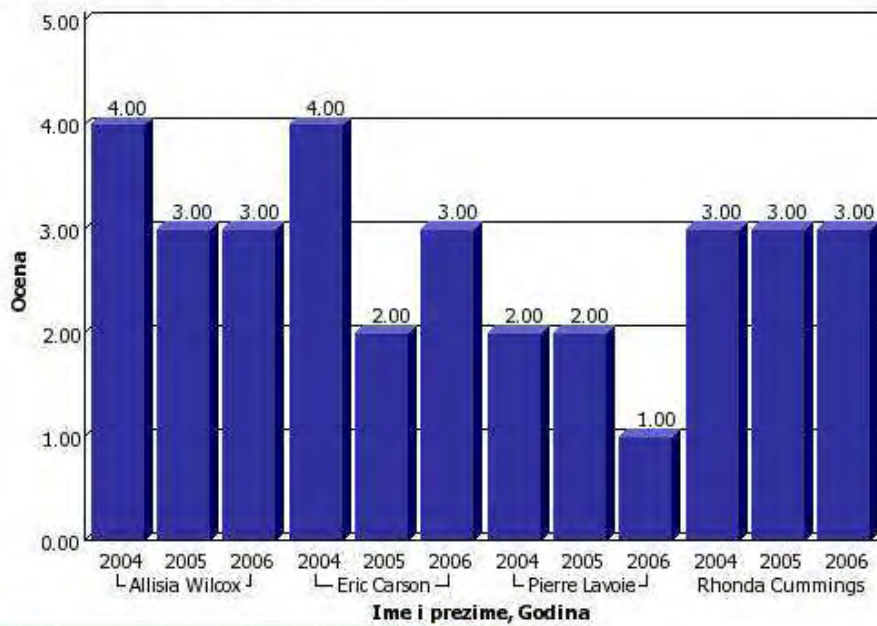


Slika 26: Baza podataka odeljenja ljudskih resursa

Na slici 26, jasno se vide sve tabele koje postoje u bazi, kao i veze koje postoje među njima. Baza podataka je implementirana u *Microsoft SQL Server 2005* alatu. Odmah se jasno uočava razlika između poslovnog pogleda i pogleda na bazu podataka. Za poslovnog korisnika koji nije stručni informatičar i ne zna npr. SQL upitni jezik, mnogo je lakše da se snađe u poslovnom pogledu na podatke nego da direktno radi nad pogledom na bazu podataka. U poslovnom pogledu, nazivi stavki upita su prikazani razumljivo za korisnika, a postoji i nekoliko izvedenih stavki upita koje pomažu korisniku da lakše napravi potrebne izveštaje ili dođe do podataka, a kojih nema u pogledu na bazu podataka. Jedna takva stavka je Ime i prezime iz tabele Informacije o zaposlenima koja spaja ime i prezime zaposlenog u jednu stavku. Iako na prvi pogled može izgledati redundantno, ovakva stavka može znatno olakšati postavljanje upita. Takođe, na određene stavke upita su dodata pogodna formatiranja da bi podaci bili prikazani korisnicima na adekvatan i odgovarajući način. *Framework Manager* nudi bogat skup opcija za izgradnju novih stavki upita kombinovanjem postojećih uz pomoć različitih matematičkih i logičkih operatora, konstanti i funkcija.

U modelu je izgrađeno nekoliko paketa koji sadrže određene komponente poslovnog pogleda. Poslovnom korisniku, koji prati kretanje zadovoljnosti zaposlenih u firmi, biće omogućeno da pristupi paketu koji se naziva *Zadovoljstvo zaposlenih* i sadrži dve tabele Informacije o zaposlenima i ocene zaposlenih. Koristeći alat *Query Studio*, ovaj poslovni korisnik će lako moći da otvori navedeni paket i da dođe do informacija koje su mu potrebne. Takođe će moći, koristeći *Report Studio*, da napravi potrebne izveštaje. Na slici 27, dat je primer izveštaja koji je moguće napraviti kroz *Query Studio* koristeći samo ovaj paket, koji pokušava da da odgovor na pitanje koliko su ljudi u firmi zadovoljni menadžmentom. Ceo poslovni pogled sadrži paket koji se naziva HR Paket.

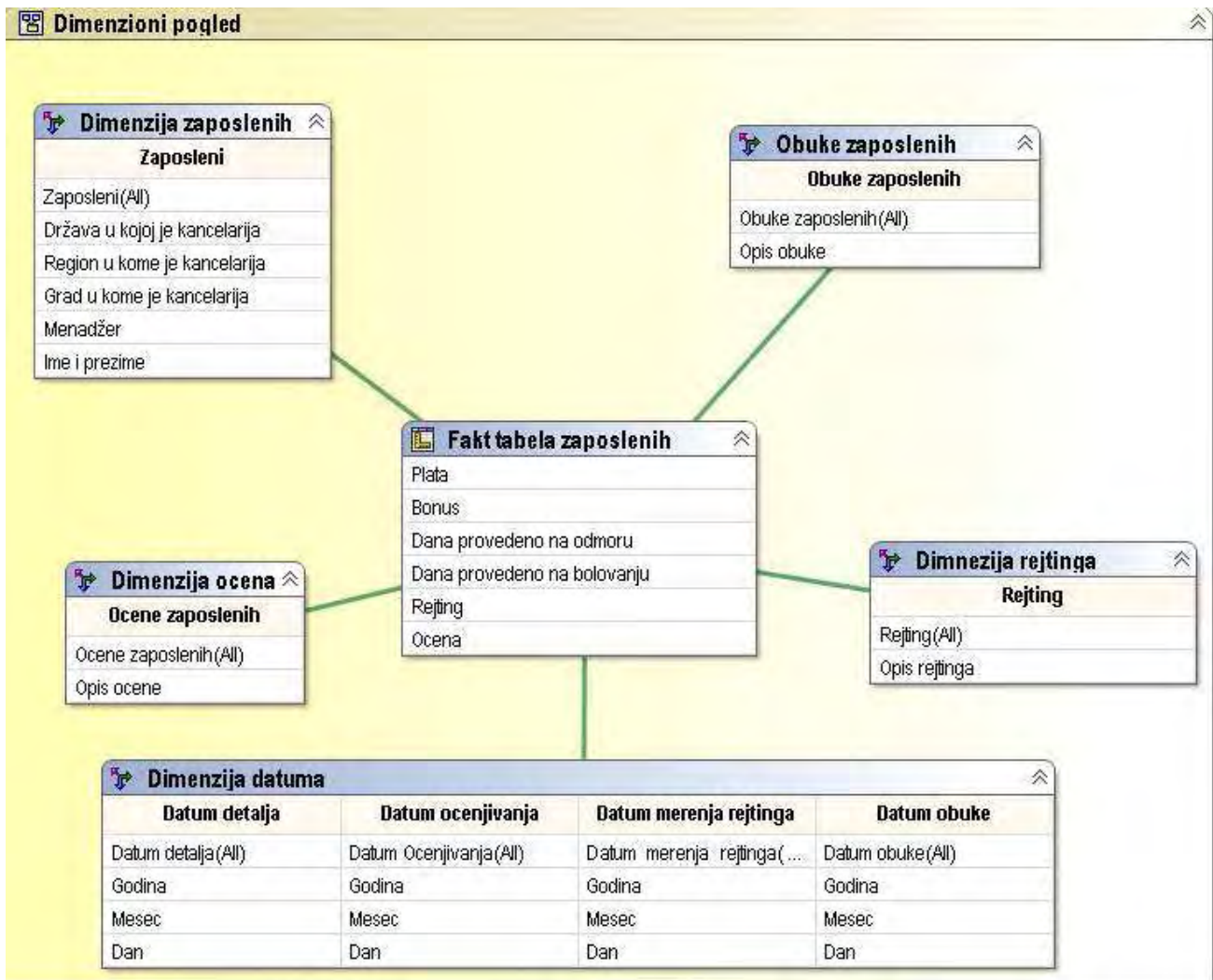
Menadžer: Alex Rodriguez



Ime i prezime	Godina	Opis ocene	Ocena
Allisia Wilcox	2004	Very Satisfied	4.00
	2005	Satisfied	3.00
	2006	Satisfied	3.00
Allisia Wilcox			3.33
Eric Carson	2004	Very Satisfied	4.00
	2005	Less than satisfied	2.00
	2006	Satisfied	3.00
Eric Carson			3.00
Pierre Lavoie	2004	Less than satisfied	2.00
	2005	Less than satisfied	2.00
	2006	Not satisfied	1.00
Pierre Lavoie			1.67
Rhonda Cummings	2004	Satisfied	3.00
	2005	Satisfied	3.00
	2006	Satisfied	3.00
Rhonda Cummings			3.00
Alex Rodriguez			2.75

Slika 27: Primer korišćenja paketa Zadovoljstvo zaposlenih kroz *Query Studio*

Na osnovu Poslovnog pogleda na podatke, napravljen je Dimenzioni pogled. Dimenzioni pogled je prikazan na slici 28.

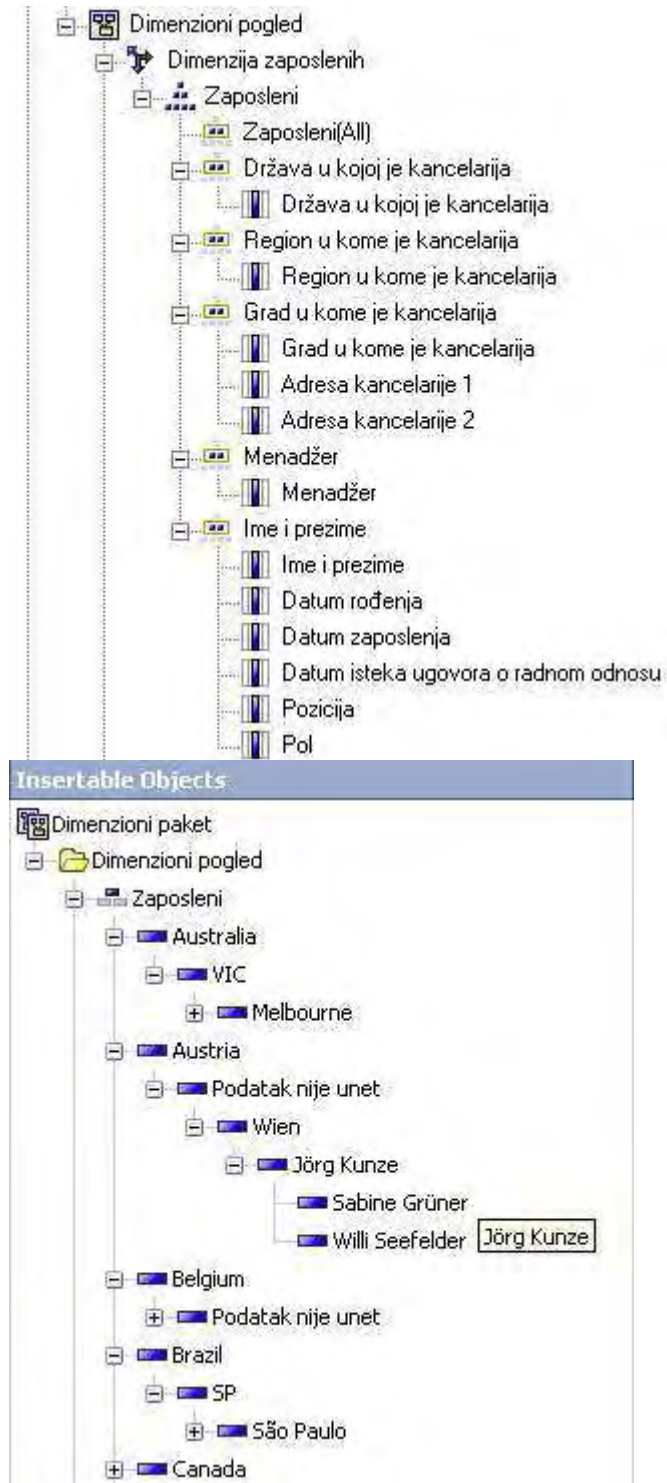


Slika 28: Dimenzioni pogled na podatke

Stavke upita su u dimenzijama raspoređene hijerarhijski. Na primer, u dimenziji zaposlenih, najviši nivo zauzima stavka upita Zaposleni(All), nakon nje Država u kojoj je kancelarija, Region u kome je kancelarija, Grad u kome je kancelarija, Menadžer i Ime i prezime. Nivo sa oznakom (All) je veštački nivo koji podrazumevano kreira *Framework Manager* i biće upotrebljen u analizama za grupisanje podataka. Ove stavke upita pored toga što su standardni atributi dimenzije, definišu i nivo hijerarhije. Pored njih na istom nivou mogu postojati i druge stavke upita. Kompletan skup stavki upita po nivoima za dimenziju zaposlenih dat je na slici 29 sa leve strane. Hijerarhija predstavlja osnov za OLAP analizu kroz alat *Analysis Studio*, koji ćemo koristiti za hijerarhijsko kretanje kroz podatke (engl. *Drill Up* i *Drill Down*) i sumarno prikazivanje podataka na različitim nivoima detalja. Prikaz hijerarhije kroz alat *Analysis Studio* dat je na slici 34 sa desne strane. Na primer, na nivou države, koji smo definisali u *Framework Manager*-u, sada se u alatu *Analysis Studio*, vide konkretni podaci na tom nivou, kao što su npr. Australija, Austrija, Belgija, itd.

Na slici 30 prikazan je *Analysis Studio*. Crvenim kvadratom je uokviren levi deo slike. U tom delu biramo dimenzije i mere koje ćemo prevući u prostor za analizu (na slici uokviren žutim pravougaonikom). Prostor za analizu je tabela. U prvoj koloni i u prvom redu su prevučene dimenzije, a u sredinu je postavljena određena mera. Na primer, na slici je u prvoj koloni postavljena dimenzija zaposlenih i to na nivou države, a u prvom redu dimenzija datuma na nivou godine. Mera koja je postavljena u sredini je plata zaposlenih. Cela analiza prikazuje sumarno plate zaposlenih i njihovo kretanje u periodu od 3 godine. Na slici je takođe označeno polje koje

prikazuje ukupan iznos isplaćen zaposlenima u SAD u 2004. godini. Prikazane su opcije koje su nam na raspolaganju, uokvirene zelenim pravougaonikom. Biranjem opcije „Drill Down” možemo ići dublje u analizu i spustiti se niz hijerarhiju i videti iznose na nižem nivou detalja. Rezultati su prikazani na slici 31. Možemo se i dalje spuštati u dubinu dok ne dođemo do najnižeg nivoa u hijerarhiji.



Slika 29: Hijerarhija atributa dimenzije zaposlenih u *Framework Manager*-u i alatu *Analysis Studio*

Insertable Objects

- Dimenzioni paket
 - Dimenzioni pogled
 - Zaposleni
 - Dimenzija datuma
 - Datum detalja
 - 2004
 - 2005
 - 2006
 - Datum ocenivanja
 - Datum merjenja rejtinga
 - Datum obuke
 - Rejting
 - Ocene zaposlenih
 - Obuke zaposlenih
 - Fakt tabela zaposlenih
 - Plata
 - Bonus
 - Dana provedeno na odmoru
 - Dana provedeno na bolovanju
 - Rejting
 - Ocena

Rows: Zaposleni(All) | **Columns:** Datum detalja(A...)

Plata	2004	2005	2006	Datum detalja(All)
Australia	€118 154.00	€176 988.00	€212 682.00	€507 824.00
Austria	€101 245.00	€121 858.00	€155 129.00	€378 232.00
Belgium	€118 011.00	€179 072.00	€225 492.00	€522 575.00
Brazil	€119 181.00	€201 346.00	€174 867.00	€495 394.00
Canada	€638 082.00	€884 662.00	€926 674.00	€2 449 418.00
China	€18 132.00	€20 182.00	€24 899.00	€63 213.00
Finland	€62 660.00	€64 168.00	€87 535.00	€214 363.00
France	€303 482.00	€386 358.00	€407 290.00	€1 097 130.00
Germany	€529 943.00	€797 670.00	€773 454.00	€2 101 067.00
Italy	€147 389.00	€193 000.00	€228 687.00	€569 076.00
Japan	€145 489.00	€164 819.00	€181 372.00	€491 680.00
Korea	€42 389.00	€51 861.00	€57 861.00	€152 111.00
Mexico	€141 272.00	€203 686.00	€223 474.00	€568 432.00
Netherlands	€200 587.00	€234 732.00	€288 611.00	€723 930.00
Spain	€74 164.00	€75 889.00	€88 686.00	€238 739.00
Sweden	€177 438.00	€230 025.00	€280 420.00	€687 883.00
Switzerland	€56 681.00	€159 249.00	€195 915.00	€411 845.00
Taiwan	€33 305.00	€37 874.00	€40 382.00	€111 561.00
United Kingdom	€617 361.00	€872 828.00	€869 839.00	€2 360 028.00
United States	€1 628 920.00	€2 128 924.00	€2 162 041.00	€5 919 885.00
Zaposleni(All)	€5 273 811.00	€5 191.00	€7 605 310.00	€20 064 386.00

Slika 30: Analysis Studio

Rows: United States | **Columns:** Datum detalja(A...)

Plata	2004	2005	2006	Datum detalja(All)
California	€380 160.00	€478 250.00	€446 744.00	€1 305 154.00
Florida	€334 359.00	€410 066.00	€406 811.00	€1 151 236.00
Massachusetts	€456 320.00	€593 443.00	€624 270.00	€1 674 033.00
Washington	€458 081.00	€647 165.00	€684 216.00	€1 789 462.00
United States	€1 628 920.00	€2 128 924.00	€2 162 041.00	€5 919 885.00

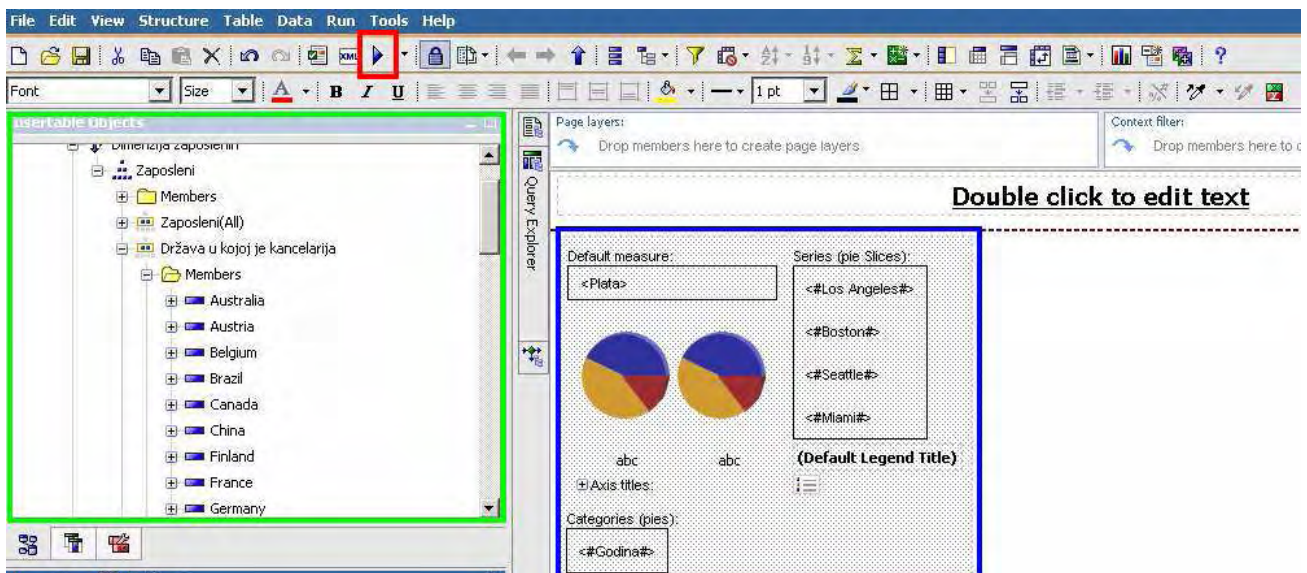
Plata	2004	2005	2006	Datum detalja(All)
Los Angeles	€380 160.00	€478 250.00	€446 744.00	€1 305 154.00
California	€380 160.00	€478 250.00	€446 744.00	€1 305 154.00

Plata	2004	2005	2006	Datum detalja(All)
Corey Wright	€293 028.00	€348 552.00	€297 540.00	€939 120.00
Frank Bretton	€87 132.00	€129 698.00	€149 204.00	€366 034.00
Los Angeles	€380 160.00	€478 250.00	€446 744.00	€1 305 154.00

Plata	2004	2005	2006	Datum detalja(All)
Dale Fowler	€60 623.00	€78 843.00		€139 466.00
Janice Thomas	Dimenzija zaposlenih.Zaposleni.Zaposleni(All).United States.California.Los Angeles.Corey Wright.Dale Fowler		€71 011.00	€188 199.00
Lowell Johnson	€60 623.00	€67 623.00	€78 843.00	€207 089.00
Margaret Lewiston	€60 623.00	€78 843.00	€78 843.00	€218 309.00
Valerie Cohen	€59 623.00	€57 591.00	€68 843.00	€186 057.00
Corey Wright	€293 028.00	€348 552.00	€297 540.00	€939 120.00

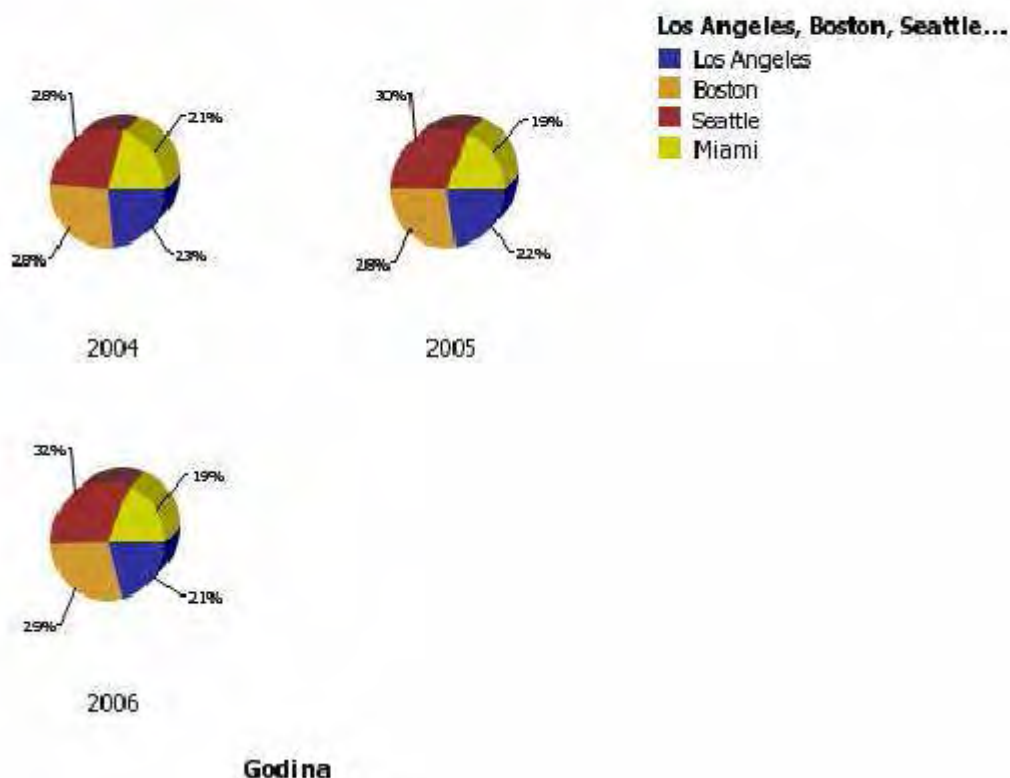
Slika 31: Spuštanje na niže nivoe hijerarhije. Prokazani su nivo regiona, nivo grada, nivo koji grupiše zaposlene po njihovom menadžeru i na kraju najniži nivo – nivo pojedinačnog zaposlenog.

Na slici 32 prikazan je *Report Studio*. Zelenim pravougaonikom je uokviren levi deo slike. Ovo je područje sa koga možemo prevući predmete upita na izveštaj koji je na slici uokviren plavom bojom. Elementi koji su nam ponudeni za prevlačenje u izveštaj su i nivoi hijerarhije koje smo definisani u alatu *Framework Manager*, a i konkretni podaci koji pripadaju odgovarajućem nivou. U ovom konkretnom primeru, izveštaj koji pravimo je grafikon. Crvenom bojom je uokvirena komanda za generisanje izveštaja.



Slika 32: *Report Studio*

Izveštaj dat na slici 33, prikazuje sumirane plate po gradovima u procentima isplaćene tokom 2004, 2005. i 2006. godine.

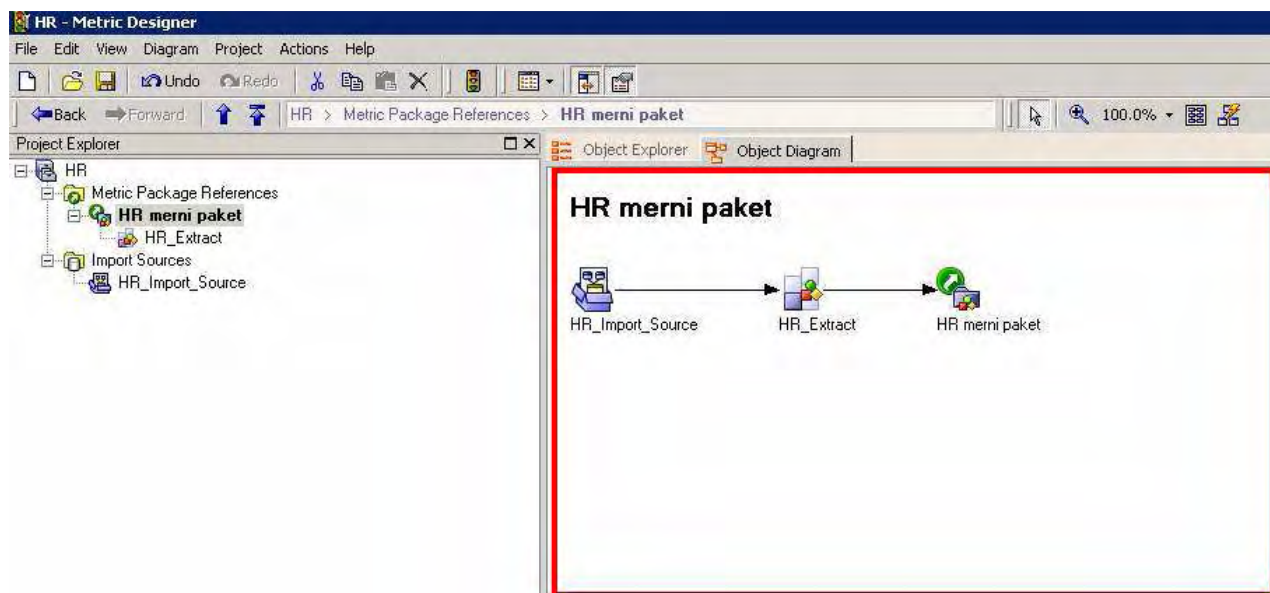


Slika 33: Izveštaj generisan u alatu *Report Studio*

Na slici 34 prikazan je deo prozora *Metric Designer* alata. U crveno uokvirenom delu nalazi se dijagram toka koji smo napravili čarobnjakom *Metric Extract Wizard*. Pri kreiranja ovog toka, napravljen je jedan merni paket i definisan tip mere koji se zove Dani – detalji. Unutar ovog tipa mere, definisana je kroz čarobnjak mera – dani provedeni na bolovanju. Ova mera prati koliko dana su zaposleni proveli na bolovanju. Ukoliko zaposleni provedu više od 10 radnih dana godišnje na bolovanju signalizira se problem – odnosno HR odeljenje bi trebalo da ispita npr. zašto je zaposleni toliko bolestan.

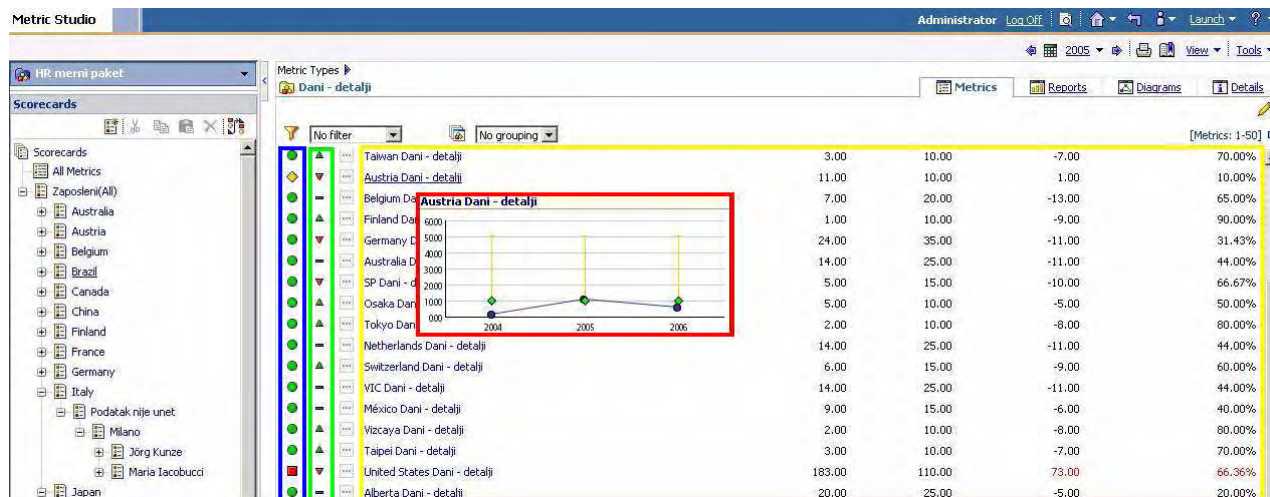
Naslov u dijagramu „HR Merni Paket” predstavlja naziv paketa u koji će biti smeštene karte rezultata. Da bismo videli rezultate mera koje smo kreirali i karte rezultata, potrebno je da pristupimo ovom paketu kroz *Metric Studio*. Sam tok ima tri koraka. Prvi korak predstavlja izvore podataka ka kojima smo napravili konekciju i iz kojih uzimamo podatke. Konkretno u našem slučaju, izvor podataka se naziva *HR_Import_Source*, a paket iz koga uzima podatke je Dimenzioni paket koji smo napravili alatom *Framework Manager* nad Dimenzionim pogledom, prikazanim na slici 28. Drugi korak nazvan *HR_Extract* čini skup definicija koje opisuju način ekstrahovanja podataka, kreiranja mera i njihovo smeštanje u tipove mera, kreiranje karti rezultata, a treći korak predstavlja paket u koji će rezultati biti smešteni. Sada možemo naredbom *Execute*, izvršiti ekstrahovanje podataka u merno skladište. Merno skladište je u ovom konkretnom slučaju napravljeno takođe u bazi *Microsoft SQL Server 2005*. Da bi napravili merno skladište posebno je da kreiramo bazu podataka kroz *SQL Server Management Studio*. Pri definisanju ekstrahovanja podataka moramo navesti i šta će nam biti merno skladište i definisati konekciju ka odgovarajućoj bazi podataka. Čarobnjak sam prepoznaje da li je baza inicijalizovana ili ne, i ukoliko nije, izvršice inicijalizaciju, pri čemu ćemo mi definisati nivo poslovnog kalendara. U našem konkretnom slučaju,

kalendar je na nivou godine, jer su merenja u bazi data na godišnjem nivou. Inicijalizacija znači kreiranje potrebnih tabela za smeštanje podataka.



Slika 34: Dijagram toka podataka napravljen čarobnjakom *Metric Extract Wizard*

Izgled *Metric Studio* dat je na slici 35. Statusi mera su na slici uokvireni plavim pravougaonikom. Ako je status prikazan crvenim kvadratom znači da je vrednost mere više od dva puta ispod dozvoljenog odstupanja od ciljne vrednosti i govori nam da su postignuti rezultati veoma loši i ispod očekivanja. Ukoliko je status prikazan zelenim krugom, znači da je vrednost mere iznad dozvoljenog odstupanja od ciljne vrednosti i govori nam da su postignuti rezultati veoma dobri. Ako je status prikazan žutim deltoidom, znači da je postignuta vrednost u granici dozvoljenog odstupanja bilo iznad, bilo ispod ciljne vrednosti.



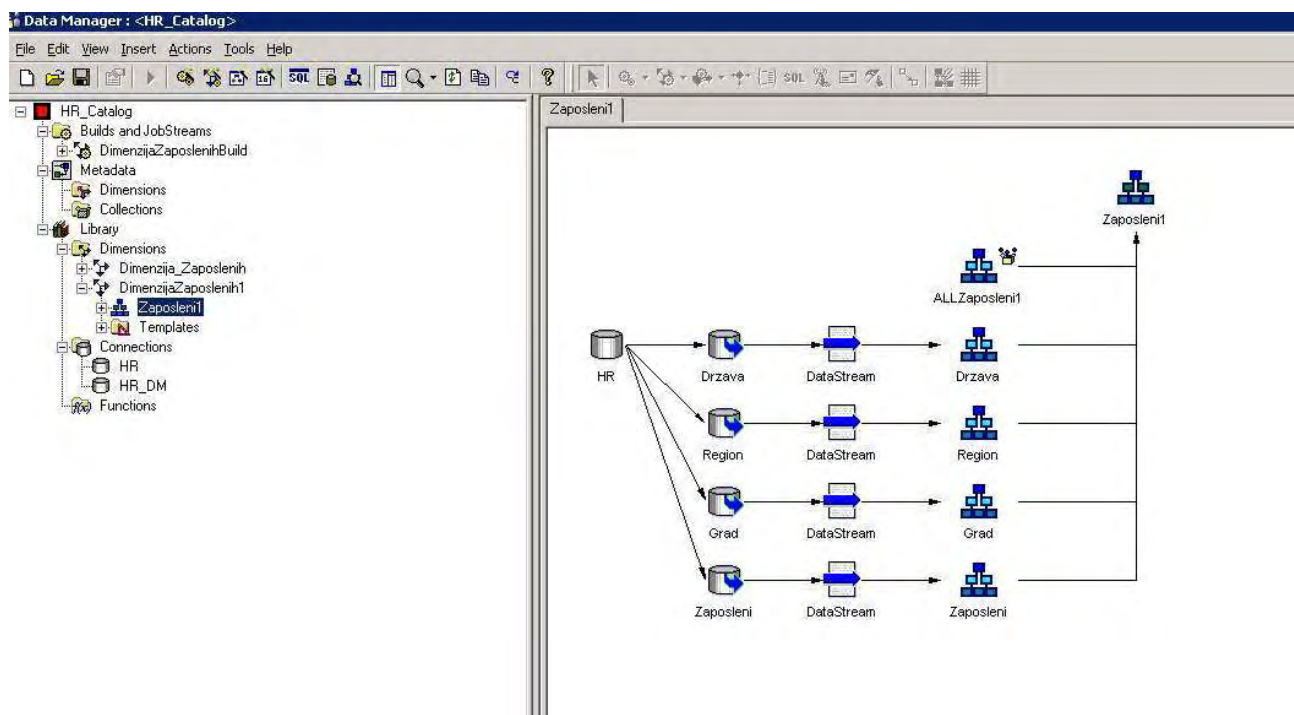
Slika 35: *Metric Studio*

Zelenim pravougaonikom su na slici uokvireni indikatori trendova. Indikator trendova pokazuje smer promene mere, tj. da li se poboljšava, pogoršava ili se ne menja. Poboljšanje je označeno zelenim trouglom, pogoršanje crvenim trouglom, a crticom ukoliko nema promena. Crvenim kvadratom je uokviren dijagram koji se dobija ukoliko postavimo pokazivač miša preko naziva mere. Dijagram nam daje detaljnije informacije o izvršenim merenjima, postignutim vrednostima i postavljenim ciljevima tokom određenog vremenskog perioda. Žutim pravougaonikom je uokvirena

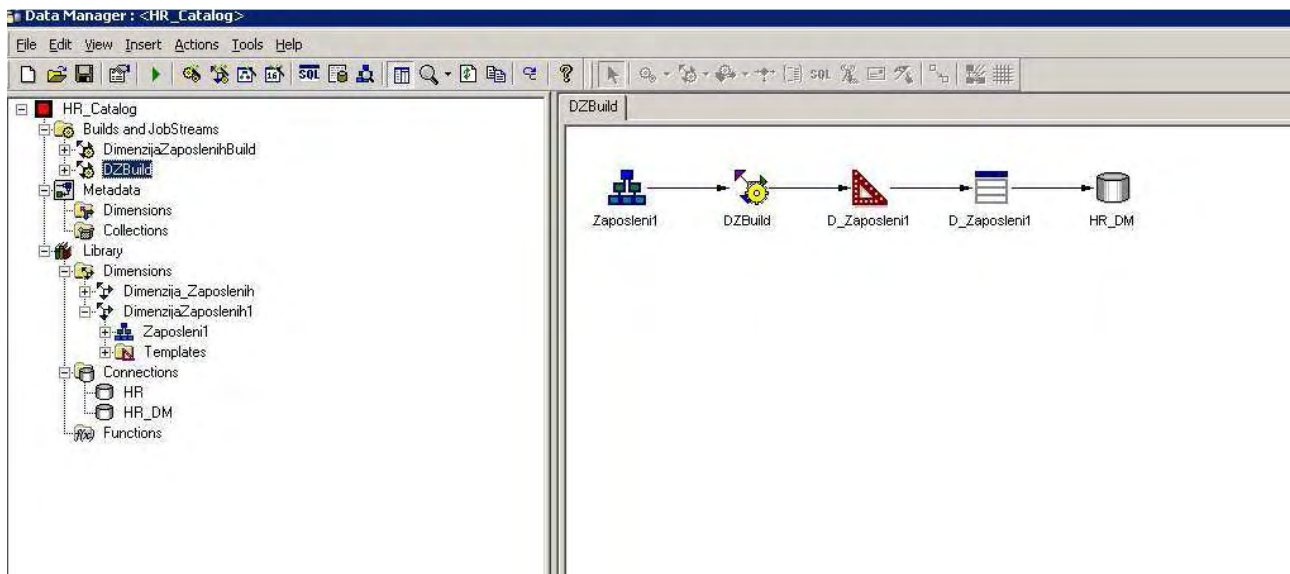
tabela u kojoj su prikazani nazivi mera koje pripadaju posmatranoj karti rezultata, kao i njihove trenutne vrednosti, ciljne vrednosti i odstupanje od ciljne vrednosti dato u brojevima i u procentima za određeni vremenski period. U levom delu slike, vidi se hijerarhijski prikaz karti rezultata.

Kreirani model nad relacionom bazom podataka nam ne omogućava uvid u istorijske podatke već samo u tekuće produkcijske podatke. Na primer, ukoliko promenimo podatak o poziciji zaposlenog u bazi, nećemo više imati uvid u njegovu prethodnu poziciju. Na ovaj način, gubimo istoriju pozicija zaposlenog, a zaposleni mogu napredovati u firmi i veoma je važno za HR da poseduje i istorijske podatke. Da bi prevazišli ovakve probleme moramo izgraditi odgovarajuće skladište podataka. Pošto već imamo izgrađen funkcionalan model, upotrebićemo alat *Data Manager* da izgradimo adekvatnu – sporo promenljivu dimenziju tipa 2 koja će pamti i istorijske podatke, a koju ćemo pridružiti našem modelu. Na ovaj način ćemo praktično implementirati ETL proces. Pre kreiranja same dimenzije potrebno je napraviti dve pomoćne baze podataka. Jednu će koristiti *Data Manager* za svoj katalog. Druga će biti ciljna baza u koju ćemo učitati izgrađenu i napunjenu tabelu dimenzija.

Dimenziju gradimo iz dva koraka. Prvo definišemo hijerarhiju i pridružimo svakom nivou hijerarhije odgovarajuće atribute. Kreirana hijerarhija za dimenziju zaposlenih je prikazana na slici 36. Pri definisanju ove hijerarhije napravili smo vezu ka našoj relacionoj bazi podataka HR odeljenja i svakom kreiranom nivou pridružili odgovarajuće atribute iz tabela ove baze. Nakon kreiranja hijerarhije pokrenućemo čarobnjaka za kreiranje dimenzija koji se naziva *Dimension Build Wizard*. Na slici 37 prikazan je niz koraka kreiranih ovim čarobnjakom kojim se od definisane hijerarhije kreira tabela u ciljnoj bazi sa popunjenim podacima. Klikom na komandu *Execute* pokreće se definisani proces.



Slika 36: Kreiranje hijerarhije zaposlenih u *Data Manager*-u



Slika 37: ETL proces *Data Manager*-a

Novoizgrađena dimenzija, pored toga što sadrži sve standardne produkcione attribute, sadrži i attribute tipa *datetime* koji beleže kada je određeni podatak kreiran odnosno ažuriran, na osnovu čega znamo u kom vremenskom periodu je taj podatak bio validan. Takođe dimenzija sadrži indikator koji nam govori da li je vrednost tekuća ili istorijska. Na ovaj način smo postigli da naš model prati i istorijske podatke zaposlenih.

Navedenim primerima nisu iscrpljene mogućnosti *Cognos* alata i njihove dalje primene u HR odeljenju. Model za HR bi mogao biti proširen dodavanjem i pogleda na druge baze podataka u sistemu, kao što je npr. baza prodaje, i ispitivati da li postoji neka povezanost između npr. količine prodatih proizvoda ili ostvarene zarade sa zadovoljnošću zaposlenih i njihovim kretanjem plata, i sl. Takođe, dalji koraci u izgradnji HR modula bi uključivali izgradnju hiperkocki nad datim modelom čime bi se znatno ubrzale performanse upita kojima se na *Cognos* platformi pristupa kroz alat *PowerPlay Studio*.

Zaključak

Tehnologije poslovne inteligencije omogućavaju poslovnim korisnicima i menadžmentu preduzeća da relativno lako iskoriste ogromne količine podataka pohranjene u relacionim bazama podataka produkcionih sistema i dođu do konkretnih odgovora koji će im omogućiti da donesu prave odluke vezane za poslovanje preduzeća. *IBM Cognos 8* je jedna od najboljih platformi trenutno na tržištu koja praktično implementira sve teoretske koncepte dimezionog modela i izgradnje skladišta podataka i to na više nivoa. Može ponuditi logički izgrađen dimenzioni model nad relacionom bazom podataka, a može da ponudi i kompletno izgrađeno skladište podataka sa ETL procesom nad kojim ćemo dalje izgraditi model za poslovne korisnike. Takođe, platforma nudi i bogat skup alata za poslovne korisnike pomoću kojih oni na jednostavan način upotrebljavaju izgrađene modele. Sa svim prednostima koje ove tehnologije donose, možemo sa optimizmom očekivati i njihovu sve veću primenu i na domaćem tržištu.

Literatura

1. Becker, B. (2009), „Six Key Decisions for ETL Achitectures”,
<http://www.ralphkimball.com/html/articles.html>
2. IBM Cognos 8 Business Inteligence – Architecture and Deployment Guide, 2008.
3. IBM Cognos 8 Business Inteligence – Data Manager Getting Started, 2008.
4. IBM Cognos 8 Business Inteligence – Metric Designer User Guide, 2008.
5. IBM Cognos 8 – Framework Manager User Guide, 2008.
6. Kimball, R. (2002), The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling, John Wiley & Sons, New York.
7. Kimball, R. & J. Caserta (2004), The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning, Conforming and Delivering Data, Wiley Publishing, Indianapolis.
8. Kimball, R. (2008), „Fact Tables”,
<http://www.ralphkimball.com/html/articles.html>
9. Kimball, R. (2003), „Declaring the Grain”,
<http://www.ralphkimball.com/html/articles.html>
10. Kimball, R. (2002), „Two Powerful Ideas”,
<http://www.ralphkimball.com/html/articles.html>
11. Kimball, R. (2007), „Dimensional Relational vs. OLAP: The Final Deployment Conundrum”,
<http://www.ralphkimball.com/html/articles.html>
12. Lane, P. (2002), „Oracle 9i Data Warehousing Guide”, Oracle Corporation, Redwood City.
13. Levene, M. & G. Loizou, „Why is the Snowflake Schema a Good Data Warehouse Design?”, School of Computer Science and Information Systems, University of London.
14. Mundy, J. (2008), „Should You Use An ETL Tool?”,
<http://www.ralphkimball.com/html/articles>
15. Ross, M. (2009), „The 10 Essential Rules of Dimensional Modeling”,
<http://www.ralphkimball.com/html/articles.html>
16. Volitich, D. (2008), IBM Cognos 8 Business Inteligence: The Official Guide, McGraw-Hill, New York.
17. Vujošević, D. (2006), Primena poslovne inteligencije u podizanju performansi organizacije – magistarski rad, Fakultet organizacionih nauka, Beograd.