FAKULTET ZA INFORMATIKU I MENADŽMENT

Branislav Ničić

ETL PROCES U RAZVOJU SISTEMA POSLOVNE INTELIGENCIJE

- Diplomski rad -

Univerzitet Singidunum
Beograd, 2009.

FAKULTET ZA INFORMATIKU I MENADŽMENT

ETL PROCES U RAZVOJU SISTEMA POSLOVNE INTELIGENCIJE

- Diplomski rad -

Mentor:
Prof. dr Angelina Njeguš

Student:
Branislav Ničić

Br. indeksa:
III-6/2007

Univerzitet Singidunum
Beograd, 2009.

FAKULTET ZA INFORMATIKU I MENADŽMENT

UNIVERZITET SINGIDUNUM
FAKULTET ZA INFORMATIKU I MENADŽMENT
Beograd, Danijelova 32

Broj: _____/2005

Kandidat: Branislav Ničić

Broj indeksa: III-6/2007

Smer: Projektovanje i programiranje

Tema: ETL PROCES U RAZVOJU SISTEMA POSLOVNE INTELIGENCIJE

Zadatak: Cilj ovog rada je da definiše pojam i prikaže osnovne karakteristike poslovne inteligencije. Rad se bavi i prikazom osnovnih tehnika i alata poslovne inteligencije i metodologijom njihove implementacije sa fokusom na ETL proces. Cilj ovog rada je da se prikaže prednost koju donosi koncept i praktična primena alata poslovne inteligencije. Urađen je i konkretan primer primene ETL alata.

MENTOR

_____
Prof. dr Angelina Njeguš

Datum odobrenja teme:

Beograd

DEKAN

_____
Prof. dr Milan Milosavljević

Sažetak

U prvom delu rad se bavi opštim pojmom poslovne inteligencije. Date su definicije pojma poslovne inteligencije, kratak prikaz istorijskog razvoja, opisane su tehnike za integraciju poslovne inteligencije u procese poslovanja. Zatim je pored definicije *data warehouse-a* dat i prikaz *arhitekture data warehouse sistema*, uz opis svih komponenti sa posebnim naglaskom na ETL proces. Četvrto poglavlje rada se odnosi na *ETL proces*, jednu od najkompleksnijih i najčešće upotrebljavanih tehnika poslovne inteligencije. Ključni aspekt je način na koji se podaci učitavaju u osnovnu i agregiranu šemu. Na kraju je urađen praktičan primer primene ETL alata.

Abstract

The first part is about the general term of business intelligence. The definition of business intelligence, a short review of historic development and techniques for integration of business intelligence are given. Besides the definition of data warehouse, the architecture of data warehouse system is given, with the description of all the components, with an accent on the ETL process. The fourth chapter is about ETL process, one of the most complex and most used techniques of business intelligence. The key aspect is the way that data is uploaded into the basic and aggregated schema. A practical example of the way ETL tools are used is at the end.

Sadržaj

1 UVOD ..... - 1 -
2 POSLOVNA INTELIGENCIJA ..... - 2 -
2.1 Definisanje pojma poslovne inteligencije ..... - 2 -
2.2 Istorijski razvoj ..... - 4 -
2.3 Razlozi i potrebe za uvođenjem koncepta poslovne inteligencije ..... - 4 -
2.4 Mogućnosti integracije poslovne inteligencije ..... - 5 -
2.4.1 BI komponente integrisane u enterprise portale ..... *- 6 -*
2.4.2 Metode poslovne inteligencije kao delovi operativnih aplikacija ..... *- 7 -*
2.4.3 Web servisi ..... *- 8 -*
2.5 Klasifikacija aplikacija poslovne inteligencije ..... - 9 -
2.6 Elementi BI rešenja ..... - 9 -
3 DATA WAREHOUSE ..... - 11 -
3.1 Definisanje pojma data warehouse-a ..... - 11 -
3.2 Uloga Data Warehouse-a ..... - 12 -
3.3 OLAP i OLTP (analitičke i transakcione baze podataka) ..... - 13 -
3.4 Arhitektura Data warehouse-a ..... - 14 -
3.4.1 Izvori podataka ..... *- 16 -*
3.4.2 Ekstrakcija, Transformacija, Punjenje (ETL) ..... *- 16 -*
3.4.2.1 Ekstrakcija (Extraction) ..... - 17 -
3.4.2.2 Transformacija (Transformation) ..... - 17 -
3.4.2.3 Punjenje (Loading) ..... - 18 -
3.4.3 Model baze podataka (Database model) ..... *- 18 -*
3.4.3.1 Fizički model skladišta podataka ..... - 18 -
3.4.3.2 Logički model baze podataka ..... - 19 -
3.4.4 Meta podaci (Metadata) ..... *- 19 -*
3.4.5 Skladište operativnih podataka (Operational Data Storage) ..... *- 20 -*
3.4.6 Data martovi (data marts) ..... *- 21 -*
3.4.7 Alati za izveštavanje i analizu (Reporting and Analitycal tools) ..... *- 22 -*
4 ETL (Extraction, Transformation, Loading) ..... - 23 -
4.1 ETL alati ..... - 23 -
4.2 Inkrementalni ETL proces ..... - 24 -
4.3 Proces učitavanja podataka u skladište podataka ..... - 24 -
4.3.1 Učitavanje osnovne šeme ..... *- 25 -*
4.3.1.1 Učitavanje podataka u tabelu dimenzija ..... - 25 -
4.3.1.2 Učitavanje podataka u tabelu činjenica ..... - 29 -
4.3.2 Učitavanje agregirane šeme ..... *- 30 -*
4.3.2.1 Učitavanje podataka u agregiranu tabelu dimenzija ..... - 33 -
4.3.2.2 Učitavanje podataka u agregiranu tabelu činjenica ..... - 34 -
5 PRAKTIČAN PRIMER PRIMENE ETL-a ..... - 35 -
5.1 Definisanje problema ..... - 35 -
5.2 Istraživanje podataka ..... - 35 -
6 ZAKLJUČAK ..... - 48 -
7 LITERATURA ..... - 49 -

1 UVOD

Danas živimo u društvu čiji je razvoj i uspešno funkcionisanje ponajviše determinisano permanentnim promenama i inovacijama u različitim oblastima nauke, tehnike i komunikacija. Sa sigurnošću se može tvrditi da je posedovanje pravovremenih i relevantnih informacija, kao i njihovo racionalno korišćenje, ključ uspeha u svim sferama ljudskog delovanja. Globalni trendovi, dinamičnost, turbulentnost primoravaju poslovne organizacije na visok stepen efikasnosti, adaptilnosti, integracije i koordinacije aktivnosti. U tom kontekstu može se govoriti o poslovnoj inteligenciji kao realizaciji potrebe za dodatnim, neotkrivenim, neopaženim znanjem i mogućnostima.

Poslovna inteligencija (Business Intelligence – BI) je deo informacionog sistema prevashodno razvijen da omogući upravljanje performansama organizacije i stavi ukupne informacione potencijale preduzeća u funkciju donošenja najkvalitetnijih odluka, kako bi se ostvarili utvrđeni strateški ciljevi preduzeća. Poslovna inteligencija kao disciplina obuhvata različite vrste tehnologija, alata i metoda i agregira znanja iz najvećeg broja naučnih oblasti.

Razlozi uvođenja BI u organizaciju su brojni. BI daje pogled na celokupnu organizaciju, pri čemu svi zaposleni mogu dobiti informaciju koja im je u tom trenutku neophodna radi donošenja boljih, bržih i relevantnijih odluka. Poslovna inteligencija omogućava proaktivan način vođenja preduzeća, što znači da se može predvideti budućnost, izraditi nekoliko scenarija i biti pripremljen za svaku situaciju. Problem je kako pretvoriti informaciju u znanje. Danas se preduzeća vode na osnovu znanja o konkurenciji, kupcima, dobavljačima,procesima i dr. Poslovna inteligencija proizvodi znanje koje je osnova za donošenje poslovnih odluka.[5]

2 POSLOVNA INTELIGENCIJA

2.1 Definisanje pojma poslovne inteligencije

Postavlja se pitanje šta u stvari predstavlja pojam business intelligence, odnosno poslovne inteligencije. Na početku, treba istaći da je ovo dosta kompleksan pojam i s toga je veoma teško pronaći, ili govoriti o univerzalnoj definiciji.

Definicija poslovne inteligencije je danas proširena, tako da obuhvata različite vrste aplikacija koje doprinose formiranju sveobuhvatnih pregleda organizacionih performansi i kreiranja analiza, predviđanja, dajući im tako veću mogućnost razumevanja ključnih faktora poslovanja i planiranja budućnosti.

Business intelligence predstavlja široku oblast aplikacija, alata i tehnologija namenjenih za skupljanje, skladištenje, omogućavanje pristupa i analiziranje podataka u cilju pomaganja prilikom donošenja različitih odluka. Prosto je neophodno poznavati sve poslovne faktore, kao što su kupci, konkurenti, poslovni partneri, ekonomsko okruženje, interne aktivnosti, da bi odluke koje se donose bile dobre i efikasne.

Poslovna inteligencija kao disciplina predstavlja skup metodologija i platformi za skladištenje podataka (engl. Data Warehousing), on-line analitičku obradu (engl. On-Line Analytical Processing), izveštavanje, statističke analize, predviđanja i otkrivanje znanja u bazama podataka (engl. Data Mining). Ti činioci omogućavaju kompanijama kreiranje korisnih upravljačkih informacija iz podataka o poslovanju. Informacije se nalaze razbacane na različitim transakcijskim sistemima i dolaze iz različitih internih i eksternih izvora.[8]

Poslovnu inteligenciju možemo posmatrati sa dva aspekta. Sa makroaspekta ona označava složenu agregiranu kategoriju koja se stvara sistematskim, ali ne ciljanim prikupljanjem podataka o makroekonomskim kretanjima u određenoj geopolitičkoj sredini. Ona, takođe, podrazumeva njihovo organizovanje i strukturirano beleženje, kao i logičko-računsku obradu radi otkrivanja trendova.

Navešćemo neke od definicija poslovne inteligencije:

- ”Poslovna inteligencija nije ni proizvod, ni sistem. To je arhitektura i kolekcija integrisanih operativnih aplikacija i aplikacija za potporu odlučivanju, te baza podataka koje poslovnim korisnicima omogućuju lak pristup poslovnim podacima”.[8]

- Poslovna inteligencija predstavlja korišćenje svih potencijala podataka i informacija u preduzeću radi donošenja boljih poslovnih odluka i u skladu s tim identifikaciju novih poslovnih mogućnosti.[1]

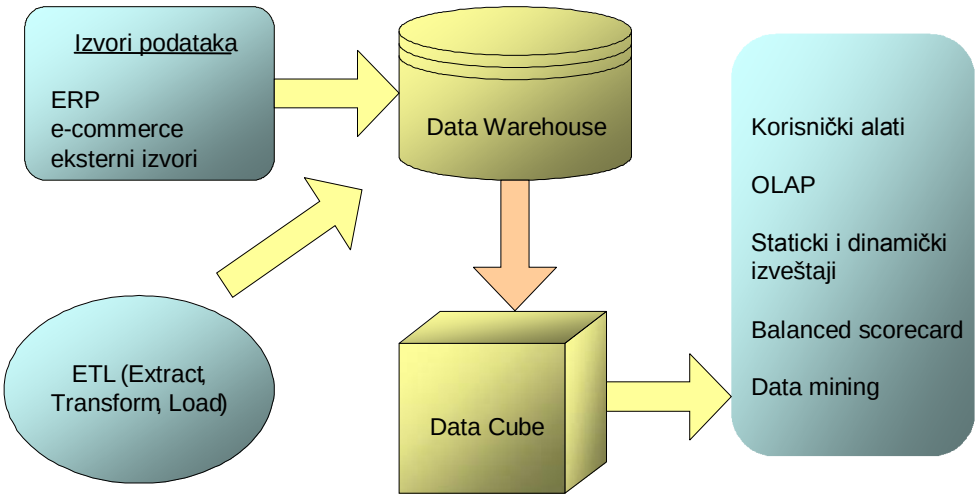
- Poslovna inteligencija je arhitektura koja predstavlja zbirni naziv za kolekciju integrisanih alata, aplikacija i baza podataka koje obezbeđuju organizaciji efikasan i lak pristup poslovnim podacima, analizu i međusobno deljenje informacija u cilju donošenja kvalitetnijih,bržih i relevantnijih odluka i poboljšanja sveukupne poslovne efektivnosti.[5]

- ”Poslovna inteligencija je proces prikupljanja raspoloživih internih i relevantnih eksternih podataka, te njihove konverzije u korisne informacije, koje mogu pomoći poslovnim korisnicima pri donošenju odluka”.[1]

- ”Poslovna inteligencija je okvir raznovrsnih organizacionih disciplina i poslovnih arhitektura za konstruisanje i upravljanje integrisanim aplikacijama i bazama za podršku odlučivanju koji poslovnoj zajednici donosi lakši pristup i omogućava im stvaranje preciznijih odluka”.[8]

- U stvarnosti, poslovna inteligencija je s jedne strane način poslovnog ponašanja, koji omogućava da se poslovne odluke na svim nivoima odlučivanja donose temeljeno na relevantnim i ažurnim poslovnim informacijama, a ne na predosećaju i subjektivnom utisku. Sa informatičke strane, poslovna inteligencija je složeni informacioni sistem koji automatizovanim procedurama prikuplja podatke iz različitih izvora, obrađuje ih, transformiše i integriše, a korisnicima omogućava pristup do kvalitetne informacije na intuitivan i lako razumljiv način.

I alati su povezani sa detaljnom, temeljnom analizom istorijskih transakcionih podataka, skladištenih u DataWarehouse-u, ili online serveru povezanom sa bazom. BI poseduje širok spektar poslovnih i nekomercijalnih aplikacija sa zajedničkim ciljem da pomognu u procesu donošenja odluka. Iz tog razloga se ne retko svrstava u klasu softvera za podršku odlučivanju. Na Slici 2.1 prikazana je keonceptualna arhitektura poslovne inteligencije.



Slika 2.1. Konceptualna arhitektura poslovne inteligencije [1]

2.2 Istorijski razvoj

Termin poslovna inteligencija je prvi put upotrebljen 1989.god, u jednom izveštaju Howarda Dresnera, analitičara kompanije Gartner Group:

”Do 2000.god., informaciona demokratija će se pojavititi u naprednim, fleksibilnim preduzećima, sa poslovno inteligentnim informacijama i aplikacijama koje će biti na raspolaganju zaposlenima, konsultantima, kupcima, dobavljačima i javnosti. Za donošenje odgovarajućih odluka zasnovanih na preciznim i ažurnim informacijam treba mnogo više od intuicije. Alati namenjeni analizi podataka, kreiranju izveštaja i upita, pomažu korisnicima da u moru podataka sakupe vredne informacije - danas ovi alati pripadaju kategoriji sa zajedničkim nazivom ”Business Intelligence” (BI).

Iz ove perspektive se čini da je sve teklo prirodnim tokom, ali sve većom brzinom. U nastavku je dat kratak pregled bitnih momenata (*milestones*) u razvoju poslovne inteligencije:

- Prvi jezici - korisnički usmereni, pojavili su se kao most između krajnjih korisnika i odbojnog IT okruženja, uspostavljajući koncept analize od strane korisnika.
- Informacioni centri kompetencije su kreirani sa ciljem unapređivanja sredstava namenjenih poboljšanju efikasnosti krajnjih korisnika. Potreba za uspostavljanjem korporacijskih standarda za analitičke alate je najznačajnija korist dobijena od informatičkih centara.
- Sa pojavom klijent/server sistema dolazi se do zaključka da držanje podataka *u mestu* nije pogodno za analizu. Zato je reinžinjering podataka u pomoćne BI forme i formate označen kao cilj kome se teži. Relacione baze su bile najprihvatljiviji oblik koje podržava SQL.
- *Information Warehouse* su dokazale da sakupljanje *meta podataka* o postojećim informacijama ima savršenog smisla. Pre nego što izvršimo transformaciju neophodno je saznati što više o njihovom sadržaju i oblicima.
- *Data warehouse* donosi revolucionarni napredak i pruža nove mogućnosti. Dizajniran je da neprekidno zadovoljava potrebe poslovnih korisnika, a ne samo dnevne operativne potrebe. Informacija postaje čista, jasna i konzistentna, bez potrebe da je tehničko osoblje intepretira u obliku koji korisnik može da razume. Sadrži i istorijske informacije, a ne samo informacije o tekućem poslovanju.
- Ulazimo u period kada se zahtevaju ugrađena, ”real-time” BI rešenja. Ovde pokretačku silu predstavljaja potreba za obezbeđivanjem sofisticiranih metrika i analiza za top menadžment.

2.3 Razlozi i potrebe za uvođenjem koncepta poslovne inteligencije

Ključni cilj poslovne inteligencije je poboljšanje vremenske tačnosti, kvaliteta i efektivnosti informacija. Tako se istovremeno otkriva ponašanje eksternih i internih činilaca i definišu pravci promena na tržištu. Iako se iz koncepta i definicije poslovne inteligencije uočavaju osnovni razlozi za njenu primenu u savremenom poslovanju, korisno je nabrojati neke od primarnih potreba za uvođenje BI:

Postojanje potrebe da se povećaju prihodi, smanje troškovi i posluje konkurentnije:

Prošlo je vreme kada su krajnji korisnici mogli da planiraju i upravljaju poslovnim operacijama pomoću mesečnih *batch* izveštaja, a IT organizacije imale mesece da završe potrebne aplikacije. Danas kompanije moraju ubrzano da projektuju i implementiraju potrebne aplikacije i omoguće korisnicima lak i brz pristup potrebnim informacijama, u skladu sa promenljivim poslovnim okruženjem, koje vlada. Sistemi poslovne inteligencije su, kada je u pitanju pristup i raspodela informacija, orijentisani ka krajnjim korisnicima.

Postojanje potrebe da se modeluje i upravlja kompleksnošću savremenog poslovnog okruženja:

Kompanije danas nude širok spektar proizvoda i usluga, sve većem i raznovrsnijem broju konzumenata. Shvatanje i upravljanje poslovanjem, kao i maksimiziranje investicija, postaje sve teže u tako kompleksnom okruženju. Sistemi poslovne inteligencije pružaju više od klasičnih mehanizama koji se zasnivaju na prostim upitima i izveštajima, pored toga nude i savremene alate za otkrivanje i analizu informacija, koji su kreirani za obradu i upravljanje složenim poslovnim informacijama.

Postojanje potrebe da se smanje IT troškovi:

Investicije u IT sisteme predstavljaju značajan procenat u odnosu na ukupne investicije u preduzeću. Imajući u vidu taj podatak, nije dovoljno samo smanjiti troškove poslovanja IT odeljka, već izvući maksimalan poslovni učinak iz poslovnih informacija koje IT sistemu generišu. Nove informacione tehnologije kao što je korporativni intranet i mobilne tehnologije, pomažu u smanjivanju troškova pristupa većeg broja korisnika sistemima poslovne inteligencije. Ovi sistemi su takođe proširili opseg informacija koje se mogu obraditi tako da pored operativnih, obuhvataju i podatke sa korporativnih Web servera.

Postojanje potrebe za proaktivnim ponašanjem, poboljšanjem komunikacije:

U uslovima nestabilnog tržišta na svim nivoima, predviđanje trendova, i planiranje brze reakcije na promene finansijskih faktora postaju zahtevi na koje kompanije moraju odgovoriti. Sistemi poslovne inteligencije olakšavaju kontrolu, međusobnu saradnju i koordinaciju svih procesa i aktivnosti.

Postojanje potrebe za permanentnim praćenjem performansi preduzeća:

Od kompanija se zahteva da poseduju jasno definisanu viziju, ciljeve, planove i s tim u vezi se analiziraju ključni pokazatelji poslovanja (Key Performance Indicators). Postavljaju se ulazni parametri za izradu scorecard projekta, čiji se rezultati opet koriste kao ulazni podaci za analitičke izveštaje.

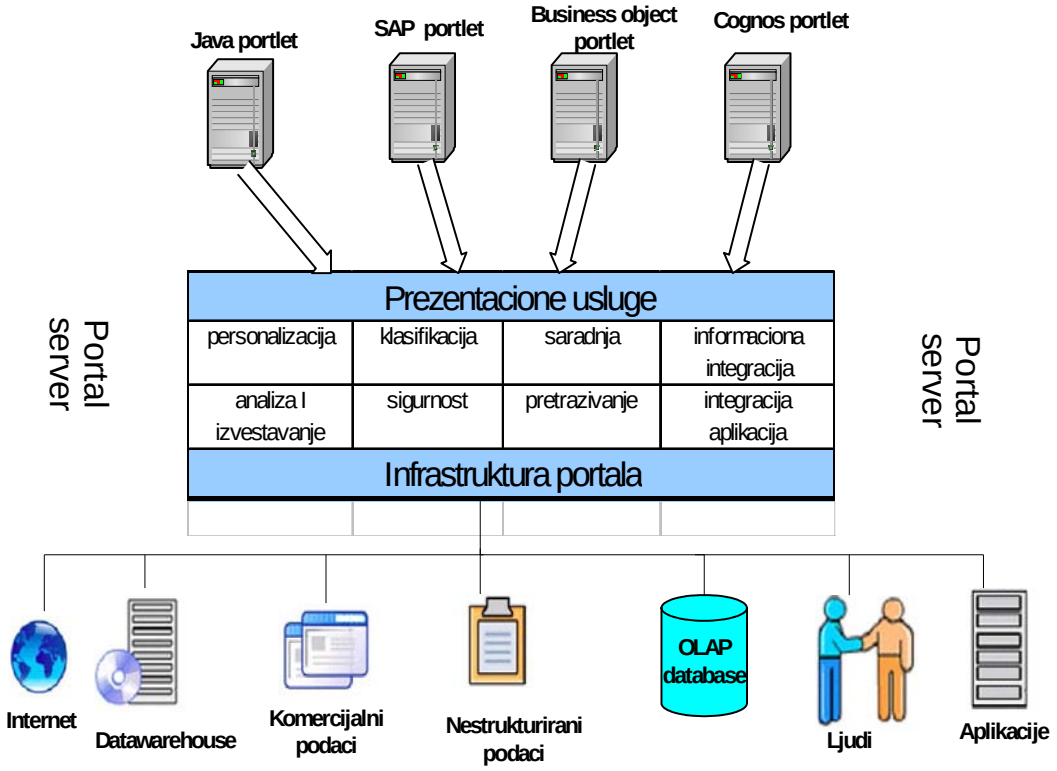
2.4 Mogućnosti integracije poslovne inteligencije

Poslovna inteligencija se može integrisati u poslovne procese preduzeća na nekoliko načina:

- Integracijom analitičkih aplikacija sa operativnim aplikacijama korišćenjem *enterprise* portala, da bi podacima mogli pristupiti i interni i eksterni korisnici.
- Ugrađivanjem analitičkih metoda u operativne aplikacije u procesu razvoja aplikacija.
- Uvođenjem Web servisa koji će dinamički integrisati analitičke metode internim, ili partnerskim operativnim aplikacijama radi podrške zajedničkom poslovanju.

2.4.1 BI komponente integrisane u enterprise portale

Korporativni informacioni portal (Enterprise Information Portal - EIP) predstavlja koncept Web sajta koji služi kao jedinstvena kapija (gateway) za kompanijske informacije, kao i baza znanja i procesa za zaposlene, poslovne partnere, klijente i (delom) za širu zainteresovanu javnost. Obezbeđuje internim i eksternim korisnicima jedinstven, standardizovan, Web baziran, korisnički interfejs za personalizaciju integrisanih sadržaja, gde se pod sadržajem podrazumevaju informacije i integrisane aplikacije i alati, kao i komunikacioni alati. Aplikacije mogu biti interne i eksterne, analitičke ili operativne, dok informacije obuhvataju strukturirane podake, poslovnu inteligenciju (izveštaje, grafikone, kocke, dijagrame i dr.), i interni i eksterni nestrukturiran sadržaj. Kao komunikacioni alati pojavljuju se Web-chat, net meeting, e-mail, instant messaging i drugi. Integracija BI sistema u korporativne portale trebala bi omogućiti da menadžer kada dođe na posao, pokrene svoj internet browser i vidi sve što se juče desilo (ili što se dešava ako se radi o real-time sistemima) u preduzeću i njegovoj okolini.



Slika 2.2. Integracija BI u enterprise portale

Treba napomenuti da postoji razlika između *enterprise* portala i korporativnog intraneta. Glavna razlika je u tome što *enterprise* portal za razliku od intraneta dozvoljava personalizaciju, odnosno omogućava da svaki korisnik ima sopstveni prikaz prilagođen njegovim potrebama. Sam Web portal se ne bavi integracijom poslovnih procesa i aplikacija, već integracijom korisničkih interfejsa raznih sistema (portleta) tako da

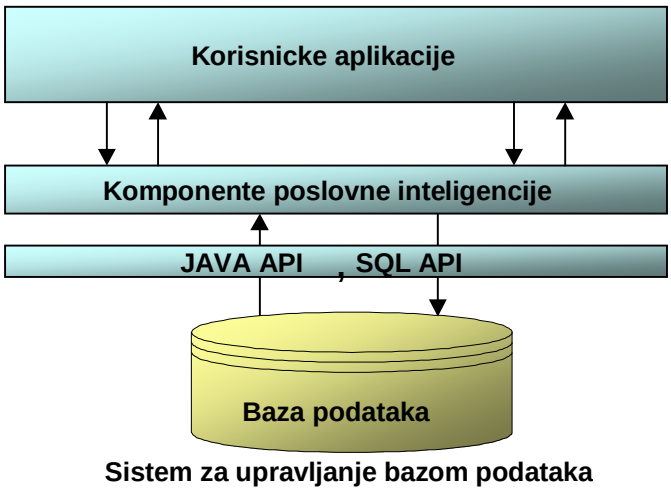
korisnici imaju utisak da se radi o delovima jednog jedinstvenog sistema. Na portalu se može nalaziti više stranica portala od kojih svaka ima skup portleta koji prikazuju sadržaje iz različitih sistema. Na taj način vrši se personalizacija ne samo sa aspekta korisnika već i sa aspekta različitih aktivnosti koje se obavljaju. Takođe treba napomenuti da su portali uglavnom nezavisni od vrste uređaja i platforme sa koje im se pristupa. Najpoznatiji Web portali su: SAP Enterprise portal, IBM WebSphere Portal, Oracle 10g AS Portal, Microsoft Office Sharepoint.[1]

2.4.2 Metode poslovne inteligencije kao delovi operativnih aplikacija

Ugradnja analitičkih metoda u operativne aplikacije vrši se u toku razvoja aplikacija i može se realizovati na aplikativnom nivou, ili na nivou samih podataka. Ukoliko se ugradnja radi na aplikativnom nivou, može se izvesti pomoću *aplikativnog programskog interfejsa (API – Application programming interface).* Primer API industrijskog standarda je *Java OLAP interfejs – JOLAP*. Operativne aplikacije mogu da pozivaju neke od alata poslovne inteligencije i na taj način da šalju zahteve za automatsku obradu i da kao rezultat te obrade dobijaju izveštaje u obliku kocki podataka, grafikona i slično. Softver dizajneri mogu da prilagode pozive tih alata sopstvenim potrebama, ili mogu da koriste gotove komponente iz biblioteke komponenata poslovne inteligencije.

Što se tiče ugrađivanja analitičkih metoda u operativne aplikacije na nivou podataka, u poslednje vreme kod proizvođača sistema za upravljanje bazama podataka (DBMS - Database Management Systems), postoji trend povećanja funkcija poslovne inteligencije koje oni ugrađuju u svoje proizvode. Microsoft je pokrenuo ovu praksu ugrađujući MS SQL Server 2000 analitički servis (Analysis Server) u sistem za upravljanje relacionom bazom podataka i nadopunjujući ga izveštajnim servisom (Reporting Server). Proizvođači se trude da povećaju funkcionalnost svojih proizvoda što se ogleda u:

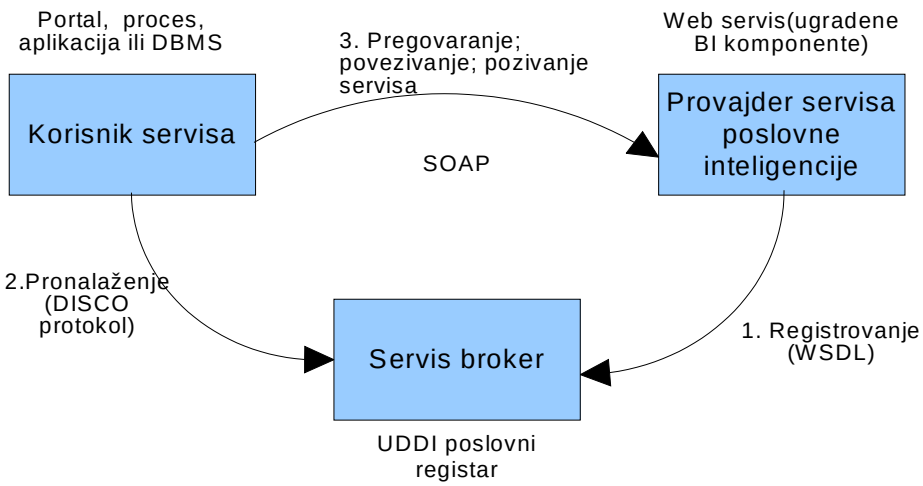
- *Proširenju standardnog SQL-a novim agregatnim funkcijama*
  U oblasti SQL-a mnogi proizvođači su uveli dodatnu podršku za agregatne funkcije, tako da sad SQL sadrži statičke, analitičke, regresione i OLAP funkcije.
- *Uvođenju novih tehnologija kao što su materijalizovani pogledi*
  Materijalizovani pogledi omogućuju sistemima za upravljanje relacionim bazama podataka podršku za sumarne podatke u formi pogleda na relacione podatke. Razlika u odnosu na kreiranje statičkih sumarnih tabela je u tome što materijalizovani pogledi imaju automatsko osvežavanje i ne zahtevaju periodično izvršavanje posebnih batch procedura za tu svrhu.
- *OLAP-u u bazi podataka*
  U zavisnosti od proizvođača DBMS-a postoje različite ekstenzije SQL-a, koje omogućuju primenu OLAP-a nad relacionom bazom podataka (ROLAP, Relational OLAP). Tako naprimer, Oracle RDBMS uvodi multidimenzionalne mehanizme u obliku funkcija koje se koriste za pristup kockama podataka (Data cubes).
- *Data mining*
  Proizvođači DBMS-a uključuju i podršku za data mining proces u svoje proizvode, korišćenjem PMML - Predictive Modelling Markup Language standarda i korisnički definisanih funkcija (UDF - User Defined Functions) koje se uvoze u DBMS, nakon čega operativna aplikacija može da koristi data mining uključivanjem UDF-a u SQL upite.



Slika 2.3. Integracija BI komponenti u operativne aplikacije [1]

2.4.3 Web servisi

Web servisi omogućuju dinamičku integraciju aplikacija korišćenjem Web tehnologija. Svaki Web servis poseduje interfejs koji opisuje operacije dostupne posredstvom standardizovanih XML poruka. Web servisi se opisuju pomoću industrijskog standarda koji se naziva *Web Service Description Language (WDSL)*, koji predstavlja skup XML tagova. Web servisi moraju da se objave i registruju u posebnim katalozima koji se nazivaju *UDDI (Unified Description Discovery and Invocation) registri*, a nezavisni su od platforme. Funkcionisanje Web servisa se odvija na sledeći način: aplikacija šalje upit UDDI registru, pronalazi potreban Web servis i dinamički se priključuje na njega šaljući mu XML poruku poznatiju kao *SOAP (Simple Object Access Protocol)* poruka. Ona inicira rad Web servisa i vraćanje rezultata u XML formatu.



Slika 2.4. Funkcionisanje Web servisa [1]

2.5 Klasifikacija aplikacija poslovne inteligencije

Obzirom na širok spektar zahteva korisnika poslovne inteligencije, kompanije danas koriste različite vrste aplikacija kako bi zadovoljili te potrebe. Aplikacije poslovne inteligencije se mogu klasifikovati na sledeći način:

1) *Izveštajne aplikacije*
   Obezbeđuju statističke ili parametrizovane izveštaje. Aplikacije za izveštavanje sa minimalnim analitičkim zahtevima bazirane su na relacionim bazama podataka i koriste SQL, dok upravljački informacioni sistemi imaju mnogo složenije analitičke zahteve i koriste specijalizovane baze podataka.

2) *Ad hoc upiti i izveštavanje*
   Pružaju korisniku najviši mogući stepen interakcije sa podacima uz mogućnost korišćenja velikog broja tehnika za odabir podataka i navigaciju. Baziraju se na relacionim bazama podataka i nude ograničene, ali veoma korisne analitičke mogućnosti. Ovi upiti koriste relacione baze podataka i SQL za jednodimenzionalne upite.

3) *Analitičke aplikacije*
   Takođe podržavaju ad hoc pretraživanje podataka, ali imaju mogućnost realizacije mnogo složenijih upita. Takvi upiti su multidimenzionalni, jer npr. u posmatranju prodaje imaju dimenziju vremena, dimenziju proizvoda i dimenziju kupaca. Istraživanja u oblasti tržišta i finansiranja obično zahtevaju izvršavanje znatno složenijih upita, pa su shodno tome, nosioci tih funkcija u preduzeću najčešći korisnici ovog tipa aplikacija.

4) *Statističke analize i data mining*
   Analize predviđanja uz korišćenje naprednih tehnika predstavljaju značajan komplement ad hoc analizama, jer predviđanje događanja na tržištu može u velikoj meri da utiče na ostvarivanje strateških ciljeva. U tu svrhu koriste se složene analitičke i statističke tehnike.

5) *Aplikacije za planiranje*
   Omogućavaju korisnicima da predvide rezultate poslovanja i daju odgovore na pitanja kao što su:
   - Koliko će preduzeće biti profitabilno u kvartalu?
   - Kako će promena cene proizvoda uticati na njegovu prodaju?
   - Koliki treba da bude obim proizvodnje da bi se zadovoljile potrebe tržišta? [1]

2.6 Elementi BI rešenja

Da bi se komplementirala priča o poslovnoj inteligenciji i njenoj integraciji u poslovne procese korisno je navesti ključne komponente poslovno inteligentne solucije:

- *Ekstrakcioni sloj* pomaže dopremanju svih istorijskih, ali i trenutnih podataka u zajedničko skladište (data warehouse). Na taj način se formira temelj za dalju analizu.
- *OLAP mašina* smešta podatke u specijalizovane forme (tzv. kocke) za bržu analizu i pretragu.
- *Komponente za izveštavanje i upite* omogućavaju predstavljanje odgovora i rezultata u vidu dijagrama, tabela, formatiranih izveštaja.
- *Data mining* pronalazi paterne u podacima, predviđa buduće trendove u poslovanju.
- *Informacioni portal* prati poslovne rezultate preko ključnih indikatora performansi, merenja vremena, odbrambenih mehanizama, itd.
- *Poslovna analitika,* poseduje aplikacije namenjene pretraživanju i analizi u terminima mera i dimenzija.

Složeni uslovi poslovanja generišu svakim danom sve veći broj poslovnih događaja u okviru poduzeća i izvan njega, a dobijeni podaci najčešće su skladišteni u operativnim bazama podataka. Dok se za potrebe operativnog vođenja poslovanja koriste klasične baze podataka, zasnovane na relacijskom modelu, koje odražavaju ažurno, stvarno stanje poslovnog sistema, a određenim podacima se nakon ažuriranja gubi trag, za donošenje pravilnih poslovnih odluka potrebno je imati uvid i u vremenski redosled zbivanja poslovnih događaja, pa takve baze podataka ne predstavljaju zadovoljavajuće rešenje. Upravo je to iniciralo kreiranje novih oblika organizovanja podataka u računskim memorijama informacioniih sistema. Razvijena je nova generacija računarskih sistema koja se temelji na konceptu skladištenja podataka.

3 DATA WAREHOUSE

3.1 Definisanje pojma data warehouse-a

Skladište podataka sadrži podatke prikupljene iz različitih izvora, istorijske o poslovanju poduzeća kao i podatke iz spoljnog okruženja. Dizajniranje je tako da omogućava pretraživanje podataka, on-line analitičku obradu, izveštavanje i podržavanje procesa donošenja odluka. Skladište podataka je po tehničkim zahtevima i po sadržaju sasvim drugačije od transakcijskih sistema. Mada je operativna baza njegova pretpostavka, skladište podataka se u svom dizajnu oslanja na višedimenzionalni koncept. Dakle, nova generacija računarskih sistema sada se sastoji od dva dela, operativnog (transakcionog) i analitičkog skladišta podataka. Na taj način se postiže izdvajanje procesa za generisanje informacija (ekstrakcije, agregacije, izveštaji, analize) koji se po svojoj prirodi razlikuju od operativnih procesa. Ekspanzija koncepta Data Warehouse-a omogućena je razvojem tehnologija za paralelno procesiranje relacionih baza podataka. Pojava 64-bitnih RISC procesora i velikih memorijskih kapaciteta dovodi do drastičnog poboljšanja performansi skladišta podataka. [9]

Pojam ”skladište podataka” (engl.Data Warehouse) podrazumeva zbirku podataka dobijenih iz operativnih baza, spremljenih u posebne baze odnosno skladišta podataka. Opšte prihvaćena definicija skladišta podataka koju je razvio Bill Inmon krajem 80-ih glasi: Data warehouse predstavlja subjektno usmeren (subject-oriented), integrisan, vezan za vreme (time-variant) i sadržajno nepromenjiv (non-volatile) skup podataka, a krajnji cilj mu je potpora menadžmentu pri donošenju odluka.

Subjektna usmerenost – podaci se organizuju oko predmeta, na način da daju informacije o tačno određenim predmetima u okviru funkcionalnih područja (npr. u okviru prodaje, nabavke...) umesto o tekućim operacijama preduzeća. Suprotno tome operativne baze podataka organizovane oko poslovnih aplikacija, dakle, usmerene su na tekuće operacije (npr. obrade narudžbina, isporuka i sl.).

Integrisanost - podaci se skupljaju u bazu podataka iz različitih izvora i pohranjuju uvek u istom formatu, konzistentni i prikazuju se na dosledan način.

Vezanost uz vreme - svi podaci u skladištu podataka vezani su i identifikuju se uz određeni vremenski period, što znači da imaju istorijski karakter. Za razliku od njih, u operativnim bazama podataka skladišteni su samo aktuelni, najsveži podaci. Međutim, s gledišta koncepta poslovne inteligencije, sveobuhvatno predviđanje budućih događaja nije moguće izvesti bez poznavanja prošlosti istih ili nekih drugih događaja. S toga sledi da, premda podaci u skladištu podataka odražavaju prošlost, njihova usmerenost okrenuta je ka budućnosti.

Sadržajna nepromenjivost - podaci u skladištu su stabilni i kad se jednom učitaju u skladište po pravilu se ne menjaju. Time se omogućuje da menadžment, ili neko drugi ko koristi skladište podataka može biti siguran da će dobiti jednak odgovor, nezavisno od vremena, ili učestalosti postavljanja upita.

Ralph Kimball definiše skladište kao ”kopiju transakcijskih podataka specijalno strukturiranu za upite i analize”. U tom svetlu, Data Warehousing je skup metodologija i tehnologija koje služe za dizajn, kreiranje, punjenje i održavanje skladišta podataka. Ovaj koncept obezbeđuje fleksibilan i efikasan način raspolaganja podataka u formatu pogodnom za savremene poslovne aplikacije

Postupak skladištenja podataka predstavlja kontinuirani proces planiranja, građenja i prikupljanja podataka iz različitih izvora, njegovog korišćenja, održavanja upravljanja i stalnog unapređenja. Među mnogim koracima u tom kompleksnom procesu bitno je naglasiti važnost posedovanja vizije o tome šta se želi da postići kreiranjem skladišta podataka.

Pre ulaska u dublja razmatranja vezana za koncept data warehouse-a treba istaći i to da se vrlo često ovaj pojam pogrešno izjednačava sa poslovnom inteligencijom. Međutim, skladištenje podataka je osnova, odnosno temelj nad kojim se primenom različitih alata ostvaruje poslovna inteligencija. Treba ga shvatiti kao najbitniji segment, koji u stvari poslovna inteligencija enkapsulira u jedinstven sistem. Odnosno, ako bi BI posmatrali kao neki građevinu, onda bi data warehouse predstavljao noseće stubove. [7]

3.2 Uloga Data Warehouse-a

Iz prethodno navedenih definicija, običnim jezikom rečeno, glavni cilj skladišta podataka je osloboditi informacije koje su ”zaključane” u operativnim bazama podataka i ”pomešati” ih sa informacijama iz ostalih, po pravilu eksternih izvora podataka. Da bi skladište podataka moglo ispuniti cilj i svrhu svog postojanja, moraju pre svega biti ispunjeni i sledeći preduslovi:

- skladište mora sadržati veliku količinu detaljnih podataka, što znači da sve poslovne transakcije relevantne za donošenje poslovnih odluka, koje su nastale u procesima preduzeća moraju biti evidentirane u skladištu podataka. Uneseni podaci moraju biti konzistentni. Na primer, ako je sa dva različita mesta, u različito vreme postavljen jednak upit i rezultat tih upita mora biti isti.
- ažuriranje novim podacima mora biti kontinuirani proces, po mogućnosti treba da se odvija u stvarnom vremenu, praktično odmah nakon što se neki poslovni događaj odigrao ili odmah po završetku nekog procesa.
- mora biti uvek raspoloživo i oblikovano na način da može poslužiti svakoj svrsi koju nije uvek moguće unapred predvideti.
- treba predvideti mogućnost izdvajanja i međusobnog povezivanja podataka u smislu dobijanja svih mera i pokazatelja poslovanja u preduzeću.
- podaci u skladištu koji se skupljaju iz različitih izvora, čiste se uz osiguranje kvaliteta i samo takvi su dostupni korisnicima. Loši ulazni podaci ne mogu dati dobre izlazne podatke.
- mora biti proširivo da bi moglo slediti strategiju proširenja poslovanja kompanije. mora zadovoljiti odgovarajuće mere zaštite tajnosti osetljivih podataka, što se postiže sprovođenjem rigoroznih mera čuvanja tajnosti.

3.3 OLAP i OLTP (analitičke i transakcione baze podataka)

Za razumevanje suštine koncepta data warehouse-a, od velikog značaja je uočavanje osnovnih karakteristika, prednosti i nedostataka, OLTP transakcione i OLAP analitičke obrade podataka.

OLTP (On Line Transaction Processing) opisuje način na koji računarski sistemi i krajnji korisnici obrađuju podatke. Usmeren je na detalje, sa čestim ažuriranjem od strane krajnjih korisnika. Često se opisuje kao način korišćenja računara za vođenje neprekidnih (trajnih) poslovnih procesa.

Budući da je proces donošenja poslovne odluke analitički proces, stvorena je druga sintagma - stalno dostupne analitičke obrade.

OLAP (On Line Analitical Processing) je vrsta tehnologije koja omogućava analitičarima i menadžerima uvid u podatke kroz brz, konzistentan i interaktivan pristup velikom broju raznovrsnih izveštaja, sačinjenih na osnovu informacija dobijenih transformacijom sirovih podataka. Na ovaj način se oslikavaju sve dimenzije jednog poslovnog poduhvata. Data warehouse podrazumeva ovaj pristup.

OLTP sistemi se zasnivaju na relacionim bazama podataka, a izgrađeni su u skladu sa Kod-ovim pravilima normalizacije, da bi se obezbedio integritet i konzistentnost podataka. Koristeći pravila, kompleksne informacije su podeljene u najednostavnije strukture-tabele. Pogodne su za brza ažuriranja podataka, utvrđivanje relacija, dok odabiranje i prebacivanje podataka za kompleksne izveštaje može da uzme mnogo procesorskog vremena i da naruši performanse baze. Dodatni problem je što implementatori IS-a nikad ne mogu da predvide sve upite koji će biti potrebni, tako da korisnicima uvek nedostaju kvalitetni izveštaji. Osim toga, izvršenje upita je sporo zato što u upitima učestvuje mnogo tabela, a njihovo spajanje (join) zahteva određeno procesorsko vreme, a naročito je zahtevno kada je reč o tabelama sa mnogo slogova.

S druge strane, podaci u data warehouse-u su denormalizovani (relaksirani), izvršavanje upita nad tako organizovanim podacima je znatno brže. Istovremeno, pojednostavljuje se šema baze podataka. Tako je olakšano pretraživanje i osoblju koje nije tehnički osposobljeno za izvršavanje složenih upita nad relacionim bazama podataka. Model se zasniva na metodologiji multidimenzione analize, što znači da podaci mogu istovremeno da se “gledaju” kroz veći broj filtara, koji se u stručnoj terminologiji nazivaju dimenzijama. U OLAP bazama podataka akcenat se stavlja na čitanje i njihovo pregrupisavanje, da bi se iz podataka napravile informacije. Konzistentnost podataka je i ovde veoma bitna karakteristika. Jer, ako podaci nisu konzistentni bilo koja informacija proistekla iz njih je pogrešna. Osim toga ovi sistemi se ne bave obradom podataka, nego tumačenjem i analizom. Pravi OLAP sistemi sadrže kopije podataka iz OLTP sistema, tako da ne ugrožavaju konkurentnost obrade, jer se nad istim podacima ne vrše operacije ažuriranja i kreriranja izveštaja. OLAP sistem može da preuzme podatke iz bilo kog relacionog ili nerelacionog izvora podataka, jer on podatke skladišti na ¨svoj¨ način i nije preterano bitna vrsta izvora podataka. Ova osobina omogućena je zahvaljujući i činjenici da je moguće uzimanje podataka iz razlčitih DBMS i objedinjavanje u zajednički OLAP sistem. Način skladištenja podataka je optimizovan u cilju kreiranja veoma kompleksnih izveštaja. OLAP alati u sebi uključuju vrlo robusne sposobnosti računanja, potrebne za zadovoljavanje specifičnih proračunskih zahteva koje postavlja višedimenzijska struktura. Teorijski, broj dimenzija je neograničen, a u praksi nisu retki slučajevi da broj u sofisticiranim analizama prelazi cifru od 10. Očigledno da ovo prevazilazi i kognitivne mogućnosti prosečnog čoveka, pa se može tvrditi da kvalitetni OLAP alati na neki način

predstavljaju proširenje, ili nadogradnju čovekove inteligencije. Na taj način, mogu "pokriti" čitav hijerarhijski lanac, što čini okosnicu donošenja poslovnih odluka – lanac koji započinje podacima, nastavlja se informacijama, a završava poslovnom inteligencijom.[8]

OLTP	OLAP
✓ Obrada podataka	✓ Analiza podataka
✓ Velik broj transakcija	✓ Kompleksni upiti
✓ Mb-Gb podataka	✓ Gb-Tb podataka
✓ Sirovi podaci	✓ Sumarni i prerađeni podaci.
✓ Mnoštvo različitih korisnika	✓ Donosioci odluka i analitičari
✓ Ažurirani podaci	✓ Istorijski podaci

Tabela 3.1. Sažet prikaz osnovnih razlika OLTP i OLAP obrade podataka.

Funkcija OLAP servera je izvršavanje ad-hoc analiza koje zahtevaju brz odgovor. Tipične operacije su: drill-down (npr. razdvajanje ukupnih prodajnih rezultata na oblasti), roll-up (sumiranje podataka po nekoj dimenziji), slice (npr. pregled podataka o samo jednom objektu), dice (npr. iz pregleda prodaje po proizvodima u pregled prodaje po godinama), itd. OLAP je veoma pogodna pomoć za postavljanje brzih, ad-hoc upita.

Data warehouse čuva i upravlja podacima, dok OLAP transformiše ove podatke u strateške informacije. OLAP obuhvata opseg operacija od jednostavnog pregleda informacija, do ozbiljnih analiza kao što su vremenske serije i kompleksno modeliranje. Na taj način korišćenjem OLAP sistema donosioci odluka prolaze put od prostog pristupa podacima, do dobijanja potrebnih informacija i u krajnjoj liniji znanja.[11]

3.4 Arhitektura Data warehouse-a

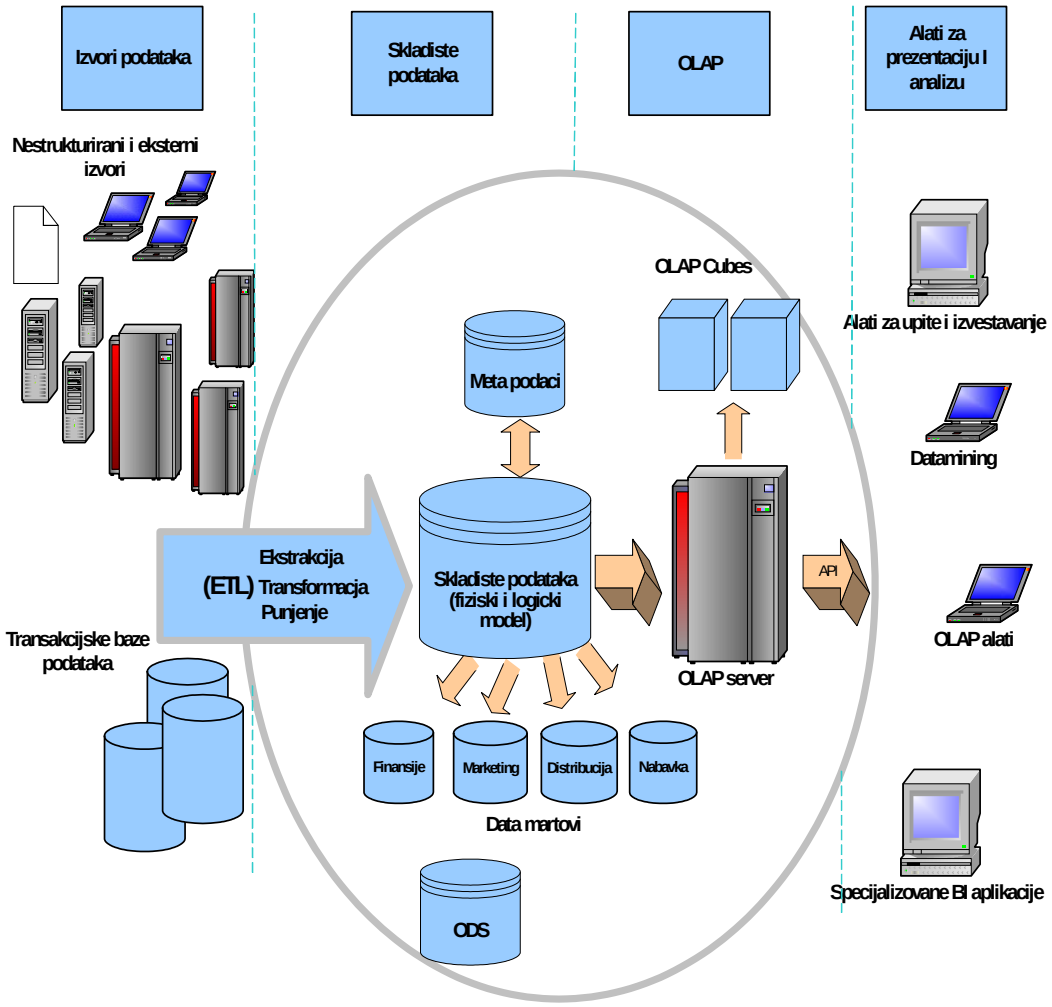
Arhitektura data warehouse-a opisuje elemente i usluge koje skladištenje pruža, sa detaljnim prikazom integracije i optimizacije komponenata, kao i potencijalnog rasta razvoja. Tokom vremena razvijene su mnogobrojne teorije i stanovišta, vezana za pitanje dizajniranja i implementacije najpogodnije arhitekture skladišta. Zbog fokusiranja na različite segmente i elemente javljaju se alternativne metode i rešenja.

Postoje dva vodeća pristupa organizovanju podataka u skladištima: dimenzionalni (uveden i predstavljen od strane Ralfa Kimbala) i normalizovan (zastupan od strane Bila Inmona).

U dimenzionalnom pristupu transakcioni podaci su podeljeni, ili u izmerene činjenice, tj. numeričke podatke koji predstavljaju specifičnu vrednost, ili dimenzije koje sadrže referentne informacije, dajući svakoj transakciji njen kontekst. Naprimer, prodajna transakcija se može razbiti na činjenice: broj naručenih proizvoda, cena i dimenzije kao štu su datum, kupac, proizvod, lokacija. Osnovna prednost ovog pristupa je lakoća upravljanja i brzina, usled prespajanja u dimenzije. Međutim, istovremeno, u slučaju promene ili dodavanja nastaju velike poteškoće.

Normalizovane tabele se grupišu po predmetnim oblastima koje oslikavaju definiciju podataka (Kupac, Proizvod, Finansije). U toj situaciji, pri aktivnosti dodavanja, promene su jednostavne, ali se gubi na performansama.

Sledeća slika na najbolji način opisuje arhitekturu data warehouse sistema. Naravno, slojevi odnosno delovi se ne mogu tretirati kao izolovani, već samo kao skladna i dobro ukomponovana celina. [3]



Slika 3.1. Arhitektura data warehouse sistema

Delovi arhitekture su:

- Izvori podataka (Data sources)
- ETL procesi (Extraction Transformation Loading)
- Model baze podataka (logički i fizički)
- OLAP server, kocka
- Meta podaci (Metadata)
- Skladište operativnih podataka(Operational Data Storage)
- Data martovi (Data Marts)
- Alati za izveštavanje i analizu (Reporting and Analitycal tools)

U nastavku teksta ukratko je obrađen svaki od navedenih delova.

3.4.1 Izvori podataka

U principu, postoje dve vrste izvora podataka za skladište podataka: spoljašnji i unutrašnji podaci.

Unutrašnji podaci pripadaju kompaniji i generisani su putem transakcijskog sistema. Opisuju aktivnosti koje su se dogodile u preduzeću. U tipičnom preduzeću se može prepoznati 30-40 transakcijskih aplikacija, odnosno podsistema, kao što su finansijski podsistem, logistički, podsistemi prodaje, proizvodnje, praćenja resursa, naplate, informacioni podsistemi.

Eksterni podaci se pribavljaju izvan kompanije, najčešće posredstvom specijalizovanih funkcija koje se bave prikupljanjem i distribucijom informacija. Od kritične važnosti su za strateške odluke, jer pomoću njih organizacija uočava povoljne mogućnosti kao i pretnje. Različite vrste eksternih podataka moguće je npr. uvrstiti u podatke o konkurentnosti (proizvodi, usluge, promene konkurentnih preduzeća...), ekonomske podatke (fluktuacija valuta, politički indikatori, kretanja kamata, berzanski podaci...), strukovne (tehnološki trendovi, marketinški trendovi...), ekonometrijske (prihodi pojedinih grupa, ponašanje kupaca...), psihometrijske (profilisanje kupaca...), demografske i marketinške podatke. Mogu se nalaziti na raznim platformama koje sadrže strukturirane podatke, kao što su tabele, ili spreadsheet-ovi, ili nestrukturirane podatke, kao što su tekstualni fajlovi, fotografije i multimedijalni sadržaji.[11]

3.4.2 Ekstrakcija, Transformacija, Punjenje (ETL)

Kao što je prethodno rečeno, podaci ulaze u skladište podataka iz različitih izvora. Najopsežniji posao u aktivnostima skladištenja podataka predstavljaju procesi integrisanja podataka i organizovanje sadržaja. Pri tom glavnu ulogu predstavlja skup procesa kojima je cilj vađenje, preoblikovanje i punjenje, ili unošenje podataka iz jednog, ili više transakcijskih sistema u skladište podataka. Pre samog početka ETL procesa potrebno je izvršiti pripremne aktivnosti: reformatiranje, usklađivanje i čišćenje podataka. Podaci u stvarnom svetu su nekompletni (nedostaju vrednosti atributa, tačne vrednosti važnih atributa, ili postoje samo agrgeirani podaci), neprecizni (greske, ili autlejeri), nekonzistentni (neslaganje u imenima ili siframa). Izvorne podatke, dopremljene iz različitih datoteka i baza podataka je potrebno unificirati, odnosno prikazati u jedinstvenom formatu. U takvom formatu podaci će se koristiti u svim daljim fazama obrade. Usklađivanje podataka se sprovodi da bi se izbegla redundansa podataka. Čišćenje kao pripremna aktivnost ETL procesa ima zadatak da ukloni one podatke koji se pojavljuju kao posledica ranijih grešaka u radu informacionih sistema. Skalabilnost ETL sistema je vrlo bitna karakteristika, i odnosi se na razumevanje obima podataka koji će biti procesiran. Povećanje količine podataka može prouzrokovati neophodne promena u dizajnu sistema. U poslednjih nekoliko godina se razvijaju alati zasnovani na paralelnom procesiranju u cilju poboljšanja ukupnih performansi.

Postoje tri osnovna tipa paralelizma koji su implementirani u ETL aplikacijama:

Data: deljenjem posebnog fajla u manje fajlove podataka štiti se paralelni pristup;

Pipeline: dopušten je simultani rad nekoliko komponenata na istom data stream-u.

Component: simultani rad višestrukih procesa na različitom data stream-u u okviru istog posla. Sortiranje jednog ulaznog fajla dok se obavlja rad na drugom fajlu može biti

primer component paralelizma.Sva tri tipa paralelizma se obično kombinuju u okviru istog posla.

Dodatna teškoća je obezbediti konzistentnost podataka. Svi višestruki izvori podataka imaju drugačiji ciklus ažuriranja (nekima je ciklus ažuriranja svakih nekoliko minuta, dok je kod drugih nekoliko dana ili nedelja). Zbog toga ETL proces mora obezbediti držanje pouzdanih podataka dok se svi izvori ne sinhronizuju. Za izgradnju ETL procesa kompanije obično kupuju ETL alate. ETL alati moraju biti takvi da omoguće komunikaciju sa različitim relacionim bazama podataka i čitanje fajlova različitih formata.

Karakteristika standardnih ETL alata je da, ako imaju bolje performanse u procesu čišćenja, tada im je slabiji kapacitet vezan za procese transformacije, ili obrnuto. S toga je pre odabira potrebno znati karakteristike podataka koji će se slivati u skladište podataka.

U prilog konstataciji o kompleksnosti ETL procesa zanimljivo je spomenuti mišljenje Kimbala da se može izvršiti podela na 38 podsistema, a pri tom je svaki podjednako bitan. On smatra da više od 70% vremena u projektovanju skladišta ide na ove procese.

3.4.2.1 Ekstrakcija (Extraction)

Ekstrakcija podataka je proces prikupljanja podataka iz različitih izvora i platformi i smeštanja tih podataka u data warehouse. Ekstrakcija podataka u data warehouse okruženju, predstavlja selektivni proces unošenja informacija bitnih za odlučivanje u data warehouse. Ekstrakcija podataka je mnogo više od prostog kopiranja podatak sa jednog sistema na drugi. Proces ekstrakcije je potrebno sprovoditi sinhronizovano tako da pri tom redovni operativni poslovi što manje trpe. S toga su programi i alati za ekstrakciju oblikovani tako da ETL procese mogu obavljati što efikasnije, uz nastojanje da potrebne podatke iz operativnih procesa zahvataju što je moguće brže. Pri tom se kao problem može pojaviti potencijalno visok stepen redundanse podataka u transakcijskim sistemima. Zato treba odabrati takav pristup ekstrakciji kojim se vrši uzimanje samo onih podataka koji će se koristiti u aplikacijama poslovne inteligencije.

3.4.2.2 Transformacija (Transformation)

U okviru ETL procesa najviše se vremena troši na postupak transformacije podataka, prema stručnim procenama i preko 80 procenata od ukupnog ETL procesa. U postupku transformacije mogu se pojaviti različiti problemi koji usporavaju proces, a kao najčešći izdvajaju se:

- *Nekonzistentne vrednosti podataka* - pojavljuju se prilikom kopiranja podataka, a kopija pri tom nije potpuno odgovarajuća originalu.
- *Nepodudarnost primarnih ključeva* - korišćenih u izvornim datotekama i bazama podataka s primarnim ključevima koje pretpostavljaju aplikacije poslovne inteligencije.
- *Netačnost podataka* - potrebno je definisati logiku čišćenja za ispravljanje vrednosti podataka. Čišćenje je potrebno sprovoditi stalno, odnosno sa svakim novim ciklusom punjenja podataka.
- *Različiti formati podataka* - format podataka, recimo datum kupovine evidentiran u operativnoj bazi mora biti u formatu kako ga očekuje alat poslovne inteligencije, pa ga treba prilagođavati.

- *Problem sinonima i homonima* - ponekad se isti podaci pojavljuju pod različitim nazivima. Isto tako događa se da različiti podaci nose isti naziv tj.govori se o homonimima. Oba slučaja su nepoželjna i treba ih uklanjati.

Neke od metoda transformacije su:

- Selektovati samo odgovarajuće tabele za unošenje (izbaciti null kolone),
- Prevođenje kodiranih podataka (ako izvorni sistem predstavlja ženski pol sa F, a muški M, a u skladištu podataka se obeležabvaju sa 1 i 2),
- Šifrovanje slobodnih formi podataka (mapiranje «Male», ili Mr. Kao 1),
- Stvaranje nove vrednosti (iznos kupovine = količina x prodajna cena),
- Spajanje podataka iz raznih izvora (merdž, pretraga),
- Sumiranje više redova podataka (ukupna prodaja po regionu),
- Kreiranje vrednosti surogat ključeva.

3.4.2.3 Punjenje (Loading)

Za procese punjenja skladišta podataka koristi više vrsta ETL alata kao što su alati za inicijalno punjenje, punjenje istorijskih podataka i programi za (inkrementalno) punjenje.

Karakteristika programa za inicijalno punjenje skladišta podataka je da sadrže rutine za čišćenje i usklađivanje podataka, da bi se iz podataka eliminisale greške. Kod istorijskih podataka ponekad nije moguće primeniti postupke čišćenja koji se primenjuju za ”žive”podatke, jer je od vremena nastanka tih podataka do danas možda došlo do različitih promena u slogovima i formatima podataka. Za razliku od ažuriranih, istorijski podaci su statičkog karaktera i čine samo sadržaj arhivskih datoteka. Treću vrstu predstavljaju programi za inkrementalno punjenje podataka, a aktiviraju se vremenski nakon prethodna dva. Pokreću se periodično i imaju ulogu stalno aktivnog mehanizma punjenja skladišta odgovarajućim sadržajima.

3.4.3 Model baze podataka (Database model)

3.4.3.1 Fizički model skladišta podataka

Postoje dve vrste modela vezanih za data warehouse: relacioni i višedimenzionalni.

Relacioni model je vezan za dvodimenzionalni prostor i grafički se može predstaviti tabelom u kojoj horizontalni redovi mogi imati ćelije sa detaljnijim podelama. U osnovi to je lista informacija koje su u relaciji sa glavnim poljima.

Multidimenzionalni model je u osnovi n-dimenzionalni prostor, gde svaki subjekat postaje dimenzija (npr. region, broj kupovina, vreme). Subjekti se predstavljaju na koordinatnim osama i za svaki novi subjekat dijagram dobija novu koordinatu. Dobijanje informacija vezanih za više dimenzija je ustvari isecanje odgovarajućih delova. Podaci su organizovani u ”grafovima”, gde se veze momentalno identifikuju, a upiti i izveštaji prave mnogo brže. Na primer, odmah se može pronaći odgovor na pitanje tipa: Koliko je proizvoda prodato protekle godine u sektoru A. Pogledi tipa ”spreadsheet” su uobičajeni izlaz iz višedimenzionalnih modela. Ovo eliminiše potrebu za korišćenjem SQL-a, za razliku od relacionog modela. Međutim, multidimenzionalne baze zahtevaju specijalizovane API-e (Application programming interface) za pristup ovakvim

arhitekturama. Istovremeno, zbog toga što su informacije skladištene onako kako se i gledaju, dolazi do redukcije podataka i vremena održavanja u poređenju sa relacionim modelom, koji zahteva indeksiranje i druge metode sortiranja. Multidimenzionalni model je optimalan izbor kada postoje zajedničke veze između elemenata. Poređenja radi, u spisku radnika, gde ni jedno ime nije isto, niti su njihovi brojevi isti pogodnije je koristiti relacioni model. U slučaju velike količine podataka i kompleksnih veza pogodnije je koristiti multidimenzionalni model.

3.4.3.2 Logički model baze podataka

Dosadašnja istraživanja vodećih stručnjaka pokazuju da je logička organizacija ključna u procesu data warehouse-a. U prilog ovom može se navesti Kimbalova koncepcija izgradnje skladišta, koja razmatra devet pitanja. Pitanja se zovu i tačke odlučivanja (decision points):

1. Koje poslovne procese treba modelovati, odnosno koje su tablice činjenica?
2. Šta je srž svake tabele činjenica?
3. Koje su dimenzije svake tabele činjenica?
4. Koje su činjenice bitne?
5. Koji su atributi dimenzija?
6. Kako pratiti dimenzije koje se sporo mijenjaju?
7. Koje agregacije koristiti, koje tablice su heterogene, mini dimenzije, načini upita i ostali problemi fizičkog skladištenja podataka?
8. Koliko dugo čuvati podatke?
9. S kojom učestanošću se podaci izvlače i učitavaju u skladište?

Odluke bi trebalo donositi u navedenom redosledu. Ova metodologija je od vrha prema dnu (top - down), zato što na vrhu počinje identifikovanjem ključnih poslovnih procesa. Dakle, u centar razmatranja se stavljaju osnovne karakteristike dimenzionalnog modelovanja. Podaci se dele na mere i kontekst. Mere su uglavnom numeričke i uzimaju se više puta. Numeričke mere se nazivaju činjenice (facts). One su okružene tekstualnim kontekstom koji opisuje situaciju u momentu kada su nastale činjenice. Mada nije neizvodljivo rasporediti kontekste u dugačke, logičke zapise, ipak je pogodnije podeliti ih u logičke grupe. Kada se zabeleže podaci o kupovini proizvoda npr. u nekoj e-prodavnici, kontekst se deli na: Proizvod, Vreme, kupac, itd. Upravo ove grupe se nazivaju dimenzije. Najzastupljeniji oblici šema baza su zvezdasta (*star join*) i pahuljičasta (*snowflake*).

Kriterijumi za izbor odgovarajuće šeme su:

- složenost problema,
- veličina tabela,
- performanse i
- memorijski prostor.

3.4.4 Meta podaci (Metadata)

Meta podaci ili ”podaci o podacima”, opisuju informacije i podatke unutar skladišta, integrišu dolazeće podatke, predstavljaju alat za redefinisanje i ažuriranje određenog modela data warehouse-a. Uključuju opise elemenata podataka, kao što su opisi tipova podataka, opisi atributa, opisi domena, zatim nazive, veličinu i dozvoljene vrednosti elemenata podataka. Meta podaci strukturiraju informacije u data warehouse-u u

kategorije, teme, grupe, hijerarhije itd. Oni služe da pruže informacije o podacima koji su smešteni u data warehouse i imaju sledeće karakteristike:

- Predmetno su orijentisani - zasnivaju se na apstrakciji realnih entiteta (npr. projekat, kupac, organizacija...)
- Definišu način na koji će se transformisani podaci interpretirati (npr. 5/10/09 kao 5. oktobar 2009.)
- Pružaju informacije o srodnim podacima u data warehouse-u
- Predviđaju vreme odziva, prikazujući broj slogova koji treba da se obradi u upitu.
- Čuva izračunate vrednosti i prekalkulisane formule kako bi se izbegla loša interpretacija.

Imena tabela i kolona, njihovi detaljni opisi, veze sa poslovnim značenjem, nedavna punjenja, broj trenutno ulogovanih korisnika, način indeksiranja, specifikacija raznim komponenata, sve su to meta podaci.

Sa stanovišta administratora data warehouse-a, meta-podaci predstavljaju skladište podataka i dokumentaciju o sadržaju i procesima u data warehouse-u. Sa druge strane, sa stanovišta korisnika meta podaci predstavljaju mapu za kretanje kroz informacije. Meta podaci su doslovno vezivno tkivo skladišta podataka i BI sistema, sadrže sve što je bitno za ceo proces data warehouse-a.

3.4.5 Skladište operativnih podataka (Operational Data Storage)

ODS je integrisana baza operativnih podataka. Sadrži ”žive” podatke, ne pregled stanja i vrlo malo informacija o prošlim stanjima izvora. Tradicionalna data warehouse arhitektura nije u skladu sa potrebama menadžera za “up-to-the-minute” podacima, neophodnim u odlučivanju u stvarnom vremenu. U takvoj situaciji kad performanse postaju kritične, javlja se ideja o kreiranju živih, operativnih skladišta podataka, odnosno tzv. ”report server” ili ”ogledalo baze”. Neke karakteristike izvora operativnih podataka su:

- *ODS je predmetno orijentisan:* projektovan je i organizovan oko glavnih predmeta interesovanja kompanije, kao što su kupac i proizvod. Znači, na jednom mestu su svi podaci o pojedinom kupcu (kao npr. jedna datoteka sa svim podacima o XY klijentu banke), lični podaci, podaci o transakcijama koje je obavio taj klijent, koje usluge je koristio itd.
- *ODS je integrisan:* on predstavlja integrisanu sliku predmetno orijentisanih podataka izvučenih iz bilo kog operativnog sistema.
- *ODS je orijentisan na trenutnu vrednost*: on odslikava trenutni sadržaj njegovih izvornih sistema, pri čemu se trenutna vrednost može definisati na različite načine, za različite izvore, u zavisnosti od zahteva implementacije. Pri tome, ovaj izvor ne treba da sadrži više prikaza trenutnog stanja, ma kako trenutno stanje bilo definisano. Ako je “trenutno” jedan računovodstveni period, onda ODS ne sadrži podatke za više od jednog računovodstvenog perioda.
- *ODS je promenljiv:* kako je ODS orijentisan na predmet, on je podložan promenama onoliko često koliko je to potrebno za oslikavanje trenutnog stanja. To znači da se podaci menjaju u stilu OLTP sistema, te će jedan isti upit dati različite vrednosti u različim trenutcima vremena, jer su se podaci u međuvremanu promenili.

- *ODS je detaljan:* definicija "detaljnog" takođe zavisi od problema koji se rešava. Granularnost ODS-a može ali ne mora da se razlikuje od granularnosti svog izvornog operativnog sistema.
- *Brzina ažuriranja* - obavlja se u mnogo kraćim vremenskim intervalima, nego kod skladištenja podataka u DW. Tada govorimo o «instant» update-u, tj. u trenutku nastanka podataka.

Na prvi pogled možda izgleda da postoji mnogo suvišnih podataka između skladišta podataka i ODS-a. ODS se napaja od skladišta podataka. Međutim, postoji jako mali broj podataka koji se prepliću između skladišta podataka i ODS-a. Skladište podataka sadrži pojedinačnu transakciju podataka, dok ODS sadrži ono što se podrazumeva pod terminom "profil" podatka.

3.4.6 Data martovi (data marts)

Data Martovi su podskupovi podataka skladišta podataka i mesto gde se odvija najviše analitičkih aktivnosti u BI okruženju. Podaci u svakom data martu su uobičajeno kreirani za određenu mogućnost ili funkciju, segment. To takođe mogu biti oblast poslovanja: marketing, prodaja, analiza profitabilnosti proizvoda, demografske analize kupaca, itd. Svaki specifični data mart je optimizovan za unapred definisano područje i ne mora biti odgovarajući za druge upotrebe. Najčešći oblik data marta je multi-dimenzionalan, što omogućava lak pristup, brzu i kvalitetnu analizu podataka. Problem koji se može pojaviti u organizaciji koja je implementirala nekoliko data martova pre implementacije centralnog skladišta podataka je integracija postojećih martova u celovit sistem. Verovatno najveći zagovornik implementacije data martova i decentraliziranih sistema je Ralph Kimball. Potrebno je neprestano praviti balans između težnje da se oni kreiraju kao odvojeni silosi ili odeljena, i potrebe za uspešnim funkcionisanjem skladišta na globalnom nivou. Međusobno usklađeni i koordinirani data martovi se nazivaju super martovi. Svaki data mart se sastoji iz niza tabela činjenica, čiji je ključ sastavljen od više spoljnih ključeva koji dolaze iz tabela dimenzija. Konformisana dimenzija (conformed dimension) je ona koja ima potpuno isto značenje u svakoj tabeli činjenica sa kojom je povezana. Zato je ta dimenzija identična u svakom data martu. Upravo to dovodi do integracije data martova, a međusobne veze se uspostavljaju preko deljenih dimenzija (Kupac, Proizvod). Data warehouse je kolekcija zasebno implementiranih supermartova povezanih zajedno sa moćnom arhitekturom, zasnovanom na konformiranim dimenzijama i standardizovanim činjenicama. U Data Warehouse Bus arhitekturi razlikuju se dve vrste data martova:

Atomski (Atomic Data Marts) - drže multidimenzionalne podatke na najnižem nivou.

Agregirani (Aggregated Data Marts) – skladište podatke u skladu sa suštinom poslovnih procesa.

Dimenzionalni modeli su izgrađeni po poslovnim procesima (koji odgovaraju poslovnim merama ili događajima), a ne poslovnim odeljenjima. Tako, na primer, podaci vezani za narudžbine su na raspolaganju na korporativnom nivou, a ne razdvojeni u tri departmenta za finansije, marketing i prodaju. Bus arhitektura identifikuje i održava veze između metrike poslovnih procesa (činjenica) i opisnih atributa (dimenzija).[3]

3.4.7 Alati za izveštavanje i analizu (Reporting and Analitycal tools)

Da bi se ispunio primarni cilj postojanja data warehouse-a podaci moraju biti na raspolaganju za analizu, izveštavanje, postavljanje upita i slične procese. Postoji mnoštvo aplikacija koje mogu obavljati ove funkcije. Neke od njih su:

- Business intelligence alati
- Izvršni informacioni sistemi (poznati kao Dashboards).
- OLAP alati
- Analitičke aplikacije
- Data Mining

Sa aspekta krajnjeg korisnika ovaj sloj je najbitniji sloj u data warehouse arhitekturi. Kako bi se pronašli odgovarajući prezentacioni alati za informacione zahteve krajnjih korisnika, pretpostavka je da postoje četiri kategorije korisnika, kao i njihove kombinacije:

"moćni korisnici" – korisnici koji su spremni i sposobni da koriste kompleksnije alate za kreiranje sopstvenih izveštaja i analiza. Oni poznaju strukturu data warehouse-a i međuzavisnosti koje vladaju u njegovoj organizacionoj formi.

"povremeni korisnik" – ova grupa korisnika sastoji se od ljudi koje direktno nisu zainteresovani za detalje o data warehouse-u, ali im je povremeno potreban pristup informacijama. Ovi korisnici su obično zauzeti svakodnevnim poslovima i nemaju vremena niti potrebu za opširnim radom sa informacijama i njihove sposobnosti u korišćenju analitičkih alata su ograničene.

korisnici koji imaju potrebu za statičkim informacijama – ovi korisnici imaju potrebu za precizno definisanim podacima u određenom vremenskom intervalu.

korisnici koji zahtevaju dinamičke ili ad hoc upite i analitičke mogućnosti alata – to su uglavnom analitičari. Ovim korisnicima svaka informacija u data warehouse-u može biti značajna u nekom trenutku. Zahtevaju visoke performanse i drill-down mogućnosti.

Različite vrste korisnika zahtevaju različite prezentacione alate, ali svi mogu da pristupaju zajedničkom data warehouse-u. Takođe, različite sposobnosti korisnika određuju i razne načine prezentacije rezultata obrade, od grafikona do tabelarnih prikaza za složene analize. Sa napretkom tehnologije i povećanjem dinamike poslovanja data warehouse sistemi postaju sve zahtevniji, baze podataka sve veće, a takođe se povećava i potreba za multimedijalnim podacima. [3]

4 ETL (Extraction, Transformation, Loading)

Kao što je predhodno rečeno, 70 % ukupnog vremena potrebnog za projektovanje skladišta podataka ide na ETL proces. Nedovoljno pripremnjeni podaci neće biti validni u daljem procesu projektovanja kladišta. Zvog toga posebnu pažnju treba posvetiti ETL-u koji je tema ovog poglavlja.

Razvoj ETL procesa je ključna komponenta svake implementacije. Konstrukcija ETL procesa zahteva upotrebu različitog seta alata koji se jednim imenom nazivaju ETL alati. Bez obzira na upotrebu različitih alata svaki glavni proces mora biti definisan, dok neki procesi mogu biti funkcionalno dekomponovani u manje procese.

Kompleksnost tranformacije podataka iz tradicionalnog modela u analitički model je jedan od problema. Pored toga, podaci mogu biti uneti zajedno iz višestrukih izvora. ETL proces mora da grupiše slične entitete unutar različitih sistema kako bi proizveo standardnu sliku za data warehouse.

4.1 ETL alati

ETL proces se može razvijati korišćenjem ETL alata ili može biti kodiran korišćenjem programskih alata. ETL proces koji je kodiran bez korišćenja ETL alata se naziva hand-coded ETL. Ova vrsta alata se obično koristi za ekstrakciju i transformaciju podataka. Ovi programi su pisani korišćenjem alata dostupnih na izvoru i/ili na odredišnom sistemu.

Hand-coded ETL je pogodan za korišćenje iz mnogo razloga. Prvo, potrebni alati se već nalaze u enterprise-u i koriste se za izgradnju i očuvanje operativnog sistema. Drugo, trening alata nije potreban. U poređenju sa ETL procesima koji su izgrađeni korišćenjem ETL alata, hand-coded su mnogo jednostavniji za izmenu. Pored toga, hand-coded ETL ne obezbeđuje metapodatke o transakcionim izvorima podataka i o warehouse šemama. Ovi metapodaci omogućavaju krajnjim korisnicima da razumeju poreklo informacija u data warehouse-u i korišćeni su od strane I.T.-a da procene mogućnosti promene transakcije i/ili data warehouse sistema. [7]

Iz ovih razloga specijalizovan ETL softver je obično implementiran. Paket ETL softvera obično podržava skladište informacija o izvornim sistemima I data warehouse šemama. Grafički alati dopuštaju razvoj procesa bez pisanja koda, sve više razumevajući i održavajući ETL process. ETL alati generalno se mogu podeliti u dve grupe: alati generisani kodom (code generators) i alati zasnovani na serveru (server based tools). ETL alati generisani kodom mogu biti mainframe-based kod (kao što je COBOL), RDBMS-based kod (kao sto su storage procedure) i kodovi koji je izvršavaju na serveru (kao što je Java). Server- based alati su danas uobičajeni. Oni uključuju mašine koje sprovode ETL process, I imaju direktnu interakciju sa izvorom i odredištem. Dopunom ETL softvera, ETL proces može da uključi mnoštvo ETL alata. Neki od njih su: integracioni softveri i softveri poruka, specijalizovani alati za sortiranje, aplikacije za čišćenje podataka itd. Svaki od ovih alata učestvuje na određeni način u ETL procesu, obavljajući specifične zadatke sa zadovoljavajućom efikasnošću.

4.2 Inkrementalni ETL proces

Data warehouse punjenje se često opisuje pomoću dva procesa: inicijalno punjenje i inkrementalno punjenje. Inicijalno punjenje je teoretski "one-time-only" proces u kome se svi postojeći izvori podataka nalaze u data warehouse-u. Za razliku od inicijalnog punjenja inkrementalno punjenje podrazumeva smeštanje izmenjenih i novih podataka u data warehouse-u. Inkrementalno punjenje upravlja manjom količinom podataka ali je mnogo složenije. Zbog toga je nephodna primena određenih pravila. Ova pravila određuju način na koji se obrađuju podaci koji se menjaju u izvornom sistemu.

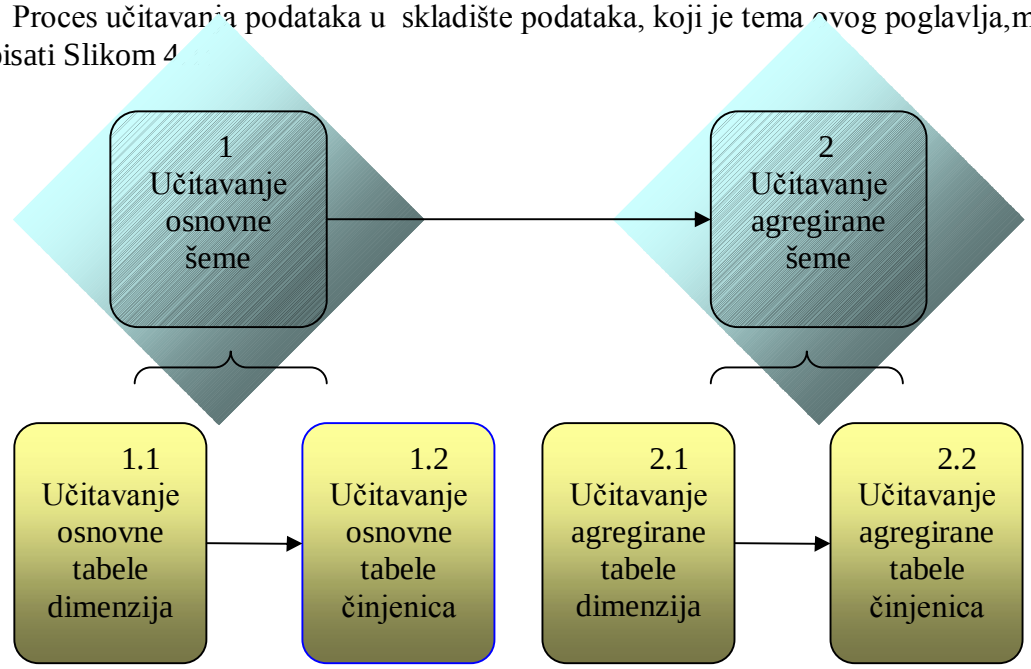
Svako data warehouse punjenje je u suštini inkrementalno punjenje. Ideja inicijalnog punjenja, da se ignorišu promene podataka u izvornom sistemu, je obično neostvariva. Skladište podataka koje se puni svakodnevno može se sukobiti sa višestrukim promenama u okviru jednog dana. Čak i kada ovo nije slučaj, inicijalno punjenje je prikladno za uključivanje istorijskih podataka toliko dobro kao i za tekuće podatke. Ovi istorijski podaci će biti punjeni u vremenskim isečcima, počev od najstarijeg isečka. Jedino prvi isečak će biti pravo inicijalno punjenje. Obrada svakog sledećeg isečka mora biti realizovana uz uključivanje malih promena.

Uzimajući proces inkrementalnog punjenja kao potreban, odvojen proces inicijalnog punjenja je suvišan. Proces inkrementalnog punjenja će biti dovoljan. Inicijalno punjenje se može koristiti kao prototip, gde se brzo punjenje uzoraka koristi u procesu provere validnosti šeme projekta.

Jedan od ključeva za izradu procesa inkrementalnog punjenja je razvoj mehanizma koji dopušta razmatranje jedino izvornih podataka koji su izmenjeni ili su novi. Ovo značajno redukuje obradu zahteva data warehousa. Informacije koje su nepromenjene ne mogu biti analizirane i obrađene.

4.3 Proces učitavanja podataka u skladište podataka

Proces učitavanja podataka u skladište podataka, koji je tema ovog poglavlja,može se opisati Slikom 4.



Slika 4.1. Proces učitavanja podataka u data warehouse

Osnovna šema sadrži podatke koji su direktno učitani iz izvornog sistema i ona predstavlja originalnu šemu. Agregirana šema je izvedena iz osnovne šeme i ona sadrži redukovan broj redova koji se dobijaju izvršavanjem nekog upita.

Tabela činjenica sadrži podatke koji su povezane sa procesom i spoljašnji ključ pomoću koga se određuje sa kojom tabelom dimenzija je povezana. Činjenice su pokazatelji pomoću kojih se proces ocenjuje. One sadrže nepotpune informacije i bez dimenzija one su beskorisne. Dimenzije daju smisao činjenicama. Skup atributa određuje jednu tabelu u kojoj jedna kolona predstavlja surogat ključ. Surogat ključ je jedinstven identifikator jednog sloga u data warehouse-u.

Slična terminilogija se koristi i u agregiranoj šemi. Agregirana tabela činjenica sadrži sažete informacije dobijene iz osnovne table činjenica. Slično, agregirana tabela dimenzija sadrži sažete informacije, ili može biti identična osnovnoj tabeli dimenzija.

Dok je proces ilustrovan određenim zavisnostima (strelice na dijagramu), on u stvari predstavlja celinu sastavljenu od nezavisnih procesa. Na primer, učitavanje podataka u osnovnu šemu se deli na dva procesa: učitavanje osnovne table dimenzija i učitavanje osnovne table činjenica. Veza između ova dva procesa ukazuje na to da tabela dimenzija mora biti napunjena pre table činjenica, zato što tabela činjenica ima njen primarni ključ kao spoljašnji.

4.3.1 Učitavanje osnovne šeme

Proces učitavanja osnovne šeme zahteva učitavanje tabele činjenica i tabele dimenzija. Tabela činjenica nosi spoljni ključ tabele dimenzija i predstavlja zavistan entitet. To znači da se tabela dimenzija učitava prva.

Kada postoji jedinstven izvor podataka, obično se koristi jedna ekstrakcija ili upit i podaci se obrađuju u jednom programu za punjenje. Program će detaljno obraditi svaki slog, ubaciti ili izmeniti svaku dimenziju na zahtev i registovati činjenicu. Jedan od glavnih razloga za implementiranje odvojenih procesa punjenja za svaku tabelu je održavanje. Pojedina punjenja koja menjaju višestruke tabele mogu biti veoma teška za održavanje. Pošto se delokrug warehouse povećava, neke od tabela mogu biti referencirane sa nekoliko dodatih tabela činjenica. Ove dodatne tabele moraju da sačekaju da predhodno sve prvobitne tabele činjenica budu napunjene.

4.3.1.1 Učitavanje podataka u tabelu dimenzija

Učitavanje tabele dimenzija zahteva obradu ulaznih podataka prema pravilima koja se uspostavljaju za vreme projektovanja osnovne šeme. Sam proces punjenja mora uzeti ulazne podatke, sakupiti vrednosti dimenzija, obraditi nove zapise dimenzija, izvršiti promene tipa 1 i promene tipa 2.

Atributi tabele dimenzija

Pre nego što pređemo na proces učitavanja tabele dimenzija moramo razmotriti karakteristike ključeva koji se javljaju u tabeli dimenzija. Pravilno razumevanje ovih karakteristika je od suštinskog značaja za sam proces učitavanja podataka. Na sledećoj slici data je tabela Proizvod sa osnovnim atributima:

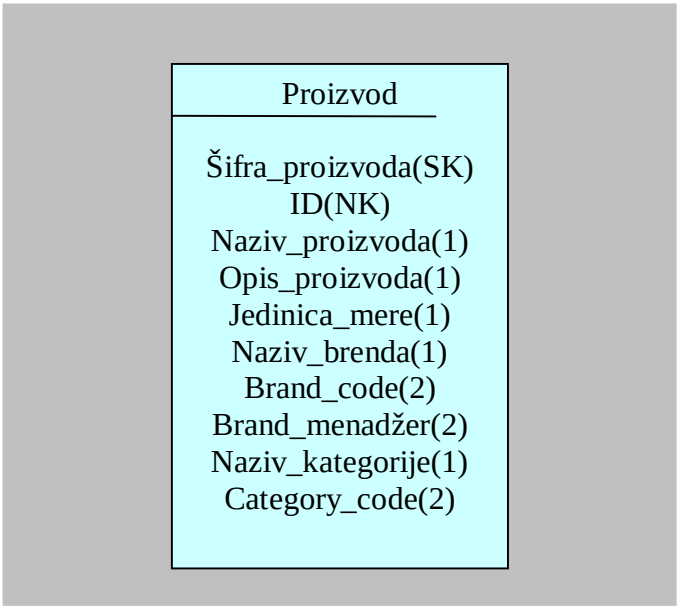


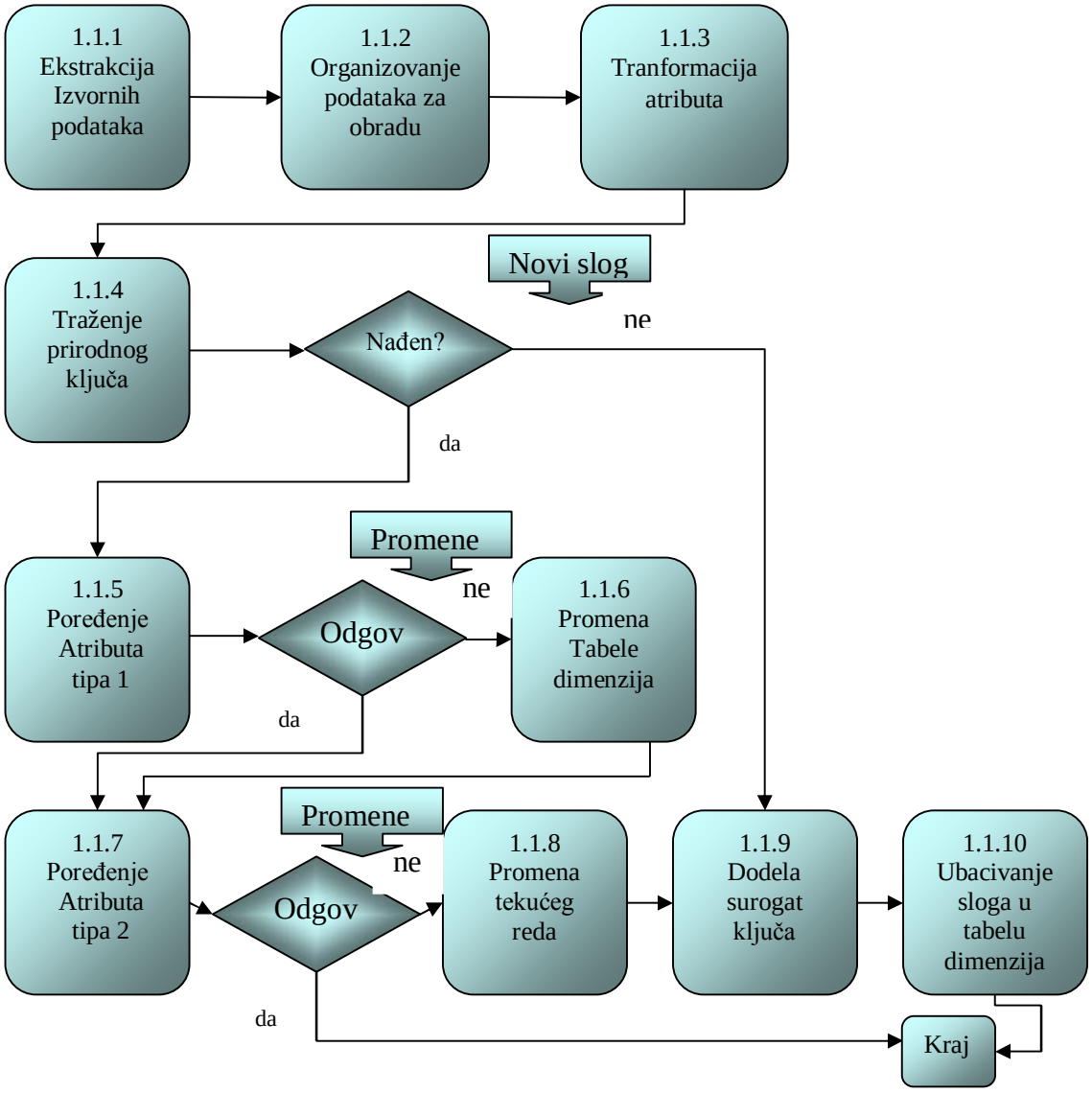
Tabela 4.1. Tipovi kolona u dimenzionoj tebeli

Prirodni ključ (NK) je jedinstven identifikator sloga u izvornom sistemu, u ovom slučaju to je ID. U tabeli Proizvod, prirodni ključ mogu imati više odgovarajućih slogova, zato što data warehouse upravlja promenama na drugačiji način nego što to izvorni sistem radi. Surogat ključ (SK) se posebno generiše za data warehouse. U ovom slučaju to je Šifra_proizvoda. To je jedan atribut koji jedinstveno identifikuje red u tabeli. Surogat ključ se koristi u data warehouse zato što je prirodni ključ nedovoljan za jedinstvenu identifikaciju sloga.

U tabeli postoje i atributi označeni kao atributi tipa 1 i atributi tipa 2. Ova klasifikacija određuje šta se dešava kada dolazi do promene istih u izvornom sistemu. Kada je atribut tipa 2 znači da promena ovog atributa dovodi do kreiranja novog sloga u data warehouse-u. Kada se menja atribut tipa 1 znači da dolazi do promene odgovarajućeg sloga u data warehouse.

Osnovni koraci u procesu učitavanja podataka u tabelu dimenzija

Na sledećoj slici dat je sažet prikaz osnovnih koraka u procesu učitavanja podataka u tabelu dimenzija:



Slika 4.2. Učitavanje osnovne tabele dimenzija

Prvi korak je ekstrakcija podataka. Podaci se mogu dobiti na više načina. Na primer, relacioni podaci mogu biti ekstrahovani korišćenjem SQL servera, eksterni podaci se mogu nabaviti on-line, itd. Nakon ekstrakcije podataka svi značajni atributi koji se odnose na jedan slog moraju se organizovati u jedan red. Ovaj proces je označen kao korak 1.1.2.

U sledećem koraku svaki red izvornih podataka se prevodi u odgovarajući atribut tabele dimenzija. Za datu transformaciju koriste se određena pravila. Na primer, polja sa višestrukim atributima mogu biti podeljena, tako da se dobije niz atributa. Polja sa null vrednostima mogu biti zamenjena sa razumljivijim tekstom kao što je “nedostupan” ili “nepozat”. Stanje i format atributa može biti podešen, itd.

Nakon što su ova tri koraka završena, set potencijalnih slogova je spreman za obradu. Sledećih nekoliko koraka određuju kako se svakim od ovih slogova upravlja. Korak 1.1.4 je označen kao proces traženja prirodnog ključa. Ako prirodni ključ nije pronađen u postojećoj tabeli slog je nov. U koraku 1.1.9 novom slogu se dodeljuje surogat ključ, dok se koraku 1.1.10 novi slog ubacuje u tabelu. Primer je prikazan na Slici 4.3

Postojeća tabela Proizvoda:

SK	NK	Tip 1	Tip 2	Tip 1	Tip2
Šifra_proizvoda	ID	Naziv_proizvoda	Brand_code	Naziv_brenda	Brand_menadžer
1011	13X22	9x12Envelope	67A	Bubble Pak	Riley, Wes
1022	13X22	9x12Envelope	67A	Bubble Pak	Jonhon, Rena
1302	A0322	6x6x6Box	80B	FlatBox	Jones, Del
1499	B1205	Sur Stick	255	Seal It	Hale,Kay

Slogovi koji se ubacuju inkrementalnim punjenjem:

ID	Naziv_proizvoda	Brand_code	Naziv_brenda	Brand_menadžer
12B88	9x12Envelope	67A	Bubble Pak	Jonhon, Rena
13X22	9x12Bubble Envelope	67A	Bubble Pak	Jonhon, Rena
A0322	6x6x6Box	80B	FlatBox	Wang, Jay
B1205	Sure Stick Tape	255	Seal It	Davis, Jon

Slika 4.3. Obrada tabele Proizvod

Prvi slog koji treba da se ubaci je slog čiji je ID 12B88. U tabeli Proizvod se proverava da li već postoji red sa datom ID vrednošću. Pošto vrednost nije pronađena slog je nov. Dodeljuje mu se surogat ključ i slog se ubacuje u tabelu.

Svaki slog koji nije nov može da sadrži promene tipa 1 ili tipa 2. Kada se atribut tipa 1 menja u izvornom sistemu potrebno je izvršiti odgovarajuće promene u data warehouse-u. Konkretno, slog koji treba sledeći da se ubaci već postoji u tabeli Proizvod. Identifikator ID se koristi u cilju pronalaženja odgovarajućih sloga u skladištu. Atributi tipa 1, u ovom slučaju naziv proizvoda, se porede i pošto se ne slažu potrebno je promeniti tabelu Proizvod.

Promene se moraju odnositi na sve slogove koje imaju isti prirodni ključ. U ovom slučaju došlo je do promena u koloni Brand_menadžer, tako da se naziv proizvoda za oba sloga mora promeniti.

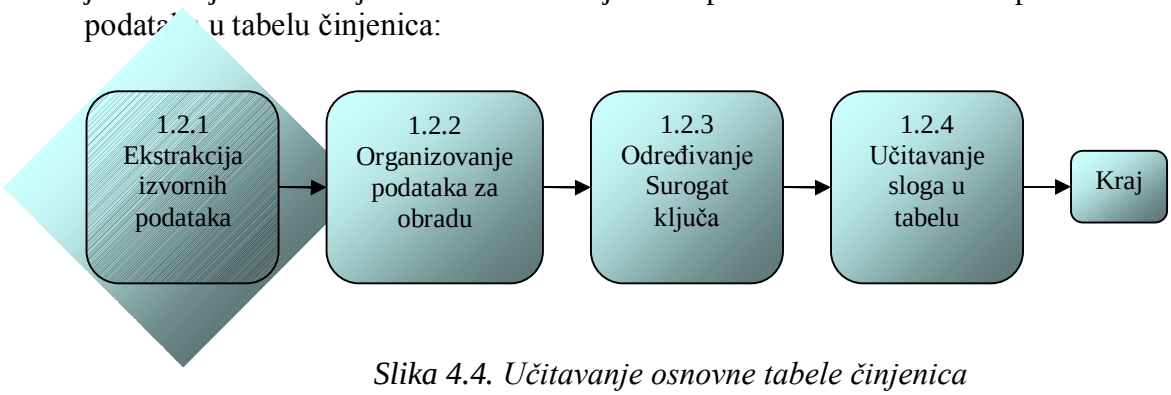
Nakon što su promene atributa tipa 1 identifikovane i primenjene, potrebno je ispitati atribute tipa 2 izvornog sloga. Na Slici 8 to je korak 1.1.7. Odgovarajući atributi se porede i ukoliko postoji slaganje znači da se slog već nalazi u tabeli i isti se odbacuje. Ako se atributi razlikuju slogu se dodeljuje surogat ključ (korak 1.1.9) i isti se ubacuje u tabelu (1.1.10).

Na primer, treći slog koji je potrebno ubaciti je slog čiji je ID A0322. Slog se već nalazi u tabeli ali je atribut Brand_menadžer promenjen, tako da je potrebno ubaciti novi red u tabeli Proizvod.

Kod nekih slogova mogu biti promenjenu i atributi tipa 1 i tipa 2. To je slučaj sa sledećim slogom koji treba ubaciti. Njegov ID je B1205. Promene se odnose na atribut Naziv_proizvoda, koji je tipa1, i na atribut Brand_menadžer, koji je tipa 2. Novi naziv proizvoda se mora dodeliti svim slogovima sa istim prirodnim ključem. Novi brend menadžer zahteva ubacivanje novog sloga u tabelu Proizvod.

4.3.1.2 Učitavanje podataka u tabelu činjenica

U poređenju sa učitavanjem podataka u tabelu dimenzija punjenje tabele činjenica je znatno jednostavnije. Na Slici 4.4 dat je sažet prikaz osnovnih koraka procesa učitavanja podataka u tabelu činjenica:



Slika 4.4. Učitavanje osnovne tabele činjenica

Kada postoji višestruki izvor podataka, navedeni koraci se moraju ponoviti za svaki od njih. Prvi korak je ekstrakcija izvornih podataka. Izvorni podaci se mogu dobiti na više načina, u zavisnosti od sistema i od raspoloživih alata. Dobijene podatke ponekada je potrebno rekonstruisati tako da može biti učitan jedan red odvojeno od drigih (korak 1.2.2).

Za svaku dimenziju povezanu sa tabelom činjenica, neophodno je odrediti odgovarajući surogat ključ. Za podatke koje dobijamo iz izvornog sistema, jedino je prirodni ključ dostupan. Potrebno je da se korišćenjem prirodnog ključa odredi odgovarajući surogat ključ. Ovaj korak mora biti ponovljen za svaki slog. Prirodni ključ, dobijen od izvornog sistema, se traži u tabeli dimenzija. Kada je pogodan slog pronađen, odgovarajući surogat ključ se dodeljuje slogu kada se isti ubacuje u tabelu činjenica.

Primer je prikazan na slici 4.5. :

Slogovii koje je potrebno učitati u tabelu činjenica:

Datum	Id_osobe	Vrednost ($)	ID	Sifra_proizvoda
1/07/2007	1197	1700000	A1011	344
1/07/2007	1197	1500000	B3691	

Postoje tabela Proizvod:

SK	NK	HK	Type 1	Type 2	Type 2
Sifra_proizvoda	ID	Tekuci_rekord	Naziv_proizvoda	Brand_code	Brand_menadzer
22	A33	Tekuci	Flat Box12	110	Michele
344	A1011	Tekuci	Packing Tape	221	John
1001	B3691	Nije tekuci	Twine Spool	501	Jones
1201	B3691	Nije tekuci	Twine Spool	501	John
2700	B3691	Tekuci	Twine Spool	621	Pit

Slika 4.5. Određivanje surogat ključa

Slog koji je potrebno učitati je slog čiji je ID A1011. Ova vrednost se traži u tabeli dimenzija. Proizvod sa traženim prirodnim ključem postoji i njegova šifra je 344. Ova vrednost se dodeljuje slogu koji se učitava.

U procesu učitavanja novog sloga promene atributa tipa 2 takođe treba ispitati. U cilju olakšavanja ovog procesa uvodi se nova kolona,Tekuci_record, u tabeli Proizvod. Kada se menja atribut, predhodni "Tekuci" record prelazi u "Nije tekuci".

Na primer, u tabeli Proizvod Brand_code i Brand_menadzer su atributi tipa 2. Ako se neki od ovih atributa promeni, za istu vrednost prirodnog ključa, biće učitan novi slog u tabelu Proizvod.

U gornjoj tabeli Slike 4.5 vidimo da je drugi slog koji je potrebno učitati slog čiji je ID B3691. Program mora naći proizvod sa traženim prirodnim ključem i za koji važi da je Tekuci_record= "Tekuci". Slog koji zadovoljava date uslove je proizvod čija je šifra 2700. Ova vrednost se dodeljuje slogu koji se učitava.

4.3.2 Učitavanje agregirane šeme

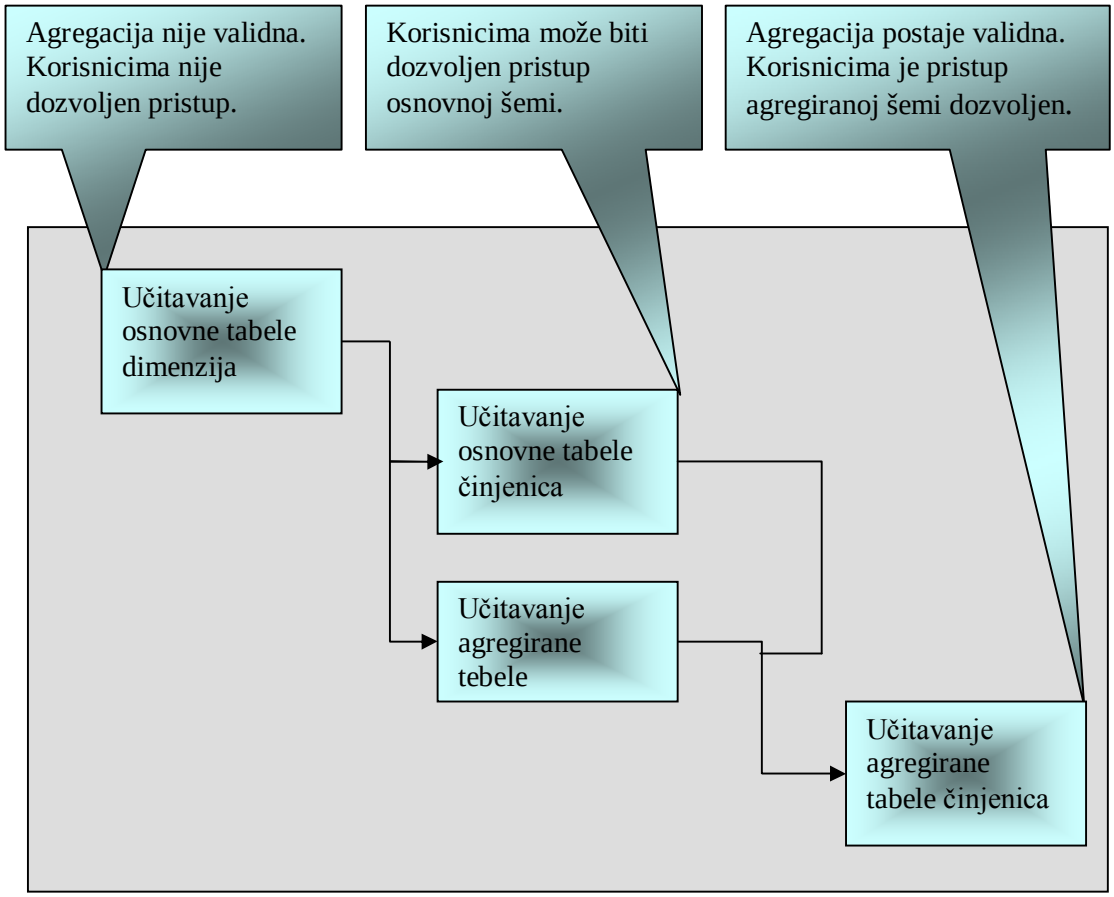
Proces učitavanja šeme agregacije je sličan procesu učitavanja osnovne šeme. Kompleksnost koja se javlja prilikom obrade osnovne šeme je eliminisana u procesu obrade šeme agregacije.

U procesu punjenja osnovne šeme, projektovani su odvojeni procesi za svaku tabelu. Odvojeni procesi se koriste i u procesu učitavanja podataka u agregiranoj tabeli činjenica i agregiranoj tabeli dimenzija. Ovakva organizacija je lakša za održavanje i nezavisni procesi obezbeđuju veću efikasnost u automatizaciji procesa učitavanja podataka. Ovo je posebno bitno za agregaciju, zato što je agregacija podložnija promenama od osnovne šeme. Agregirana tabela činjenica može biti odbačena, ali agregirana tabela dimenzija se ne sme odbaciti zato što može biti uključena u novu agregaciju.

Pošto su procesi slični, često se dešava da se simultano učitavaju podaci u osnovnu šemu i u agregiranu šemu. Ali programi koji se pri tom koriste su mnogo teži za održavanje od skupa odvojenih programa, tako da raste neefikasnost u procesu učitavanja podataka. Ako se osnovna i agregirana tabela činjenica pune odvojeno krajnji korisnik će moći da pristupi osnovnoj tabeli činjenica odmah nakon njenog učitavanja. Ali ako se učitavanje podataka vrši istovremeno, proces će biti znatno duži i krajnji korisnik neće imati pristup sve dok se obe tabele ne učitaju.

Da agregacija ne bi postala nekorisna i nevažeća potrebno je da se učitavanje vrši istom frekvencijom kao i učitavanje osnovne šeme. Na primer, ako se osnovna šema učitava svakodnevno i agregirana šema mora biti učitana istom frekvencijom. Inače, agregirana šema neće biti validna ako dođe do promena u osnovnoj šemi.

Kao što je predhodno rečeno, ako se osnovne tabele, iz kojih su izvedene agregirane table, menjaju agregacija će dati različite rezultate upita. Ovo predstavlja narušavanje vodećih principa agregacije. Ova situacija se mora izbeći. Na Slici 4.6 je prikazan redosled učitavanja podataka.



Slika 4.6. Redosled učitavanja tabela

Kao što je već rečeno agregacija u početku nije validna. Proces počinje učitavanjem podataka u osnovnu tabelu. Nakon toga podaci se istovremeno učitavaju u tabelu činjenica i agregiranu tabelu dimenzija. Učitavanje agregirane tabele činjenica počinje nakon završetka svih ostalih učitavanja.

U ovoj situaciji, krajnjim korisnicima u početku nije dozvoljen pristup osnovnoj i agregiranoj šemi. Kada se završi učitavanje osnovne tabele činjenica krajnji korisnici mogu pristupiti osnovnoj šemi. Nakon učitavanja agregirane tabele činjenica korisnici imaju pristup celokupnom skladištu podataka.

Upravljanje promenama

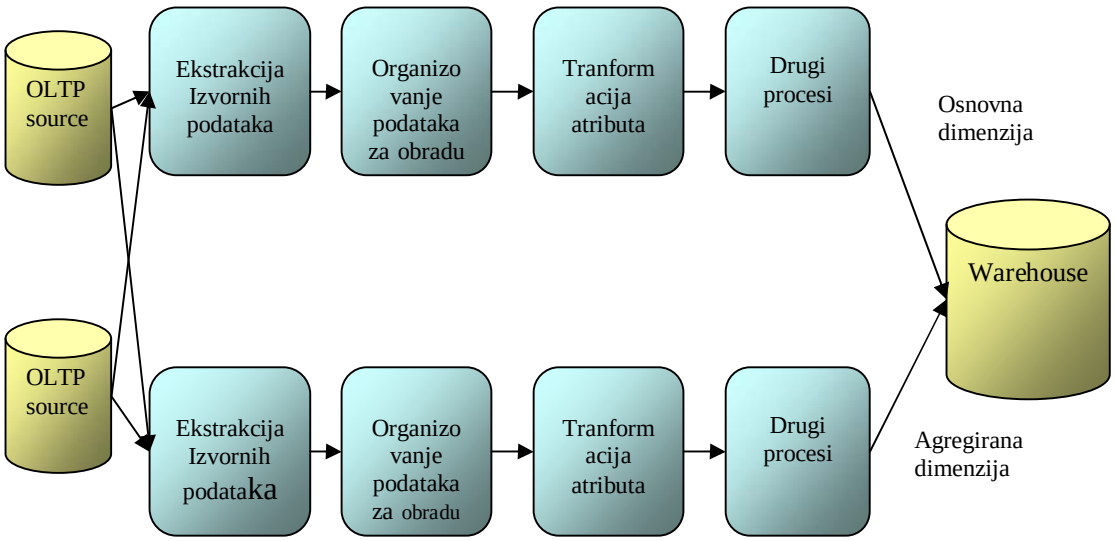
Efikasnost ETL procesa je uvek u centru intresovanja. Jedan od načina da se obezbeti veća efikasnost je da se redukuje količina podataka kojom je potrebno upravljati. Na primer, ako su novi ili izmenjeni podaci identifikovani u izvornom sistemu, neophodno je samo ove podatke razmotriti. Međutim, nije lako ustanoviti kada su se desile promene u izvornom sistemu. U cilju rešavanja ovog problema uvodi se nova kolona u tabeli dimenzija, koja pruža informacije o datumu poslednje promene. Ovaj datum se menja svaki put kada se menja ili dodaje novi slog.

Ovaj jednostavan korak obezbeđuje mehanizam pomoću koga se mnogo lakše upravlja promenama. Kada se učitavaju podaci u agregiranu tabelu, nephodno je da se razmotre samo oni podaci kod kojih je datum promene kasniji od datuma kada je poslednji slog učitan. Ovo značajno umanjuje količinu podataka koje je inače potrebno obraditi.

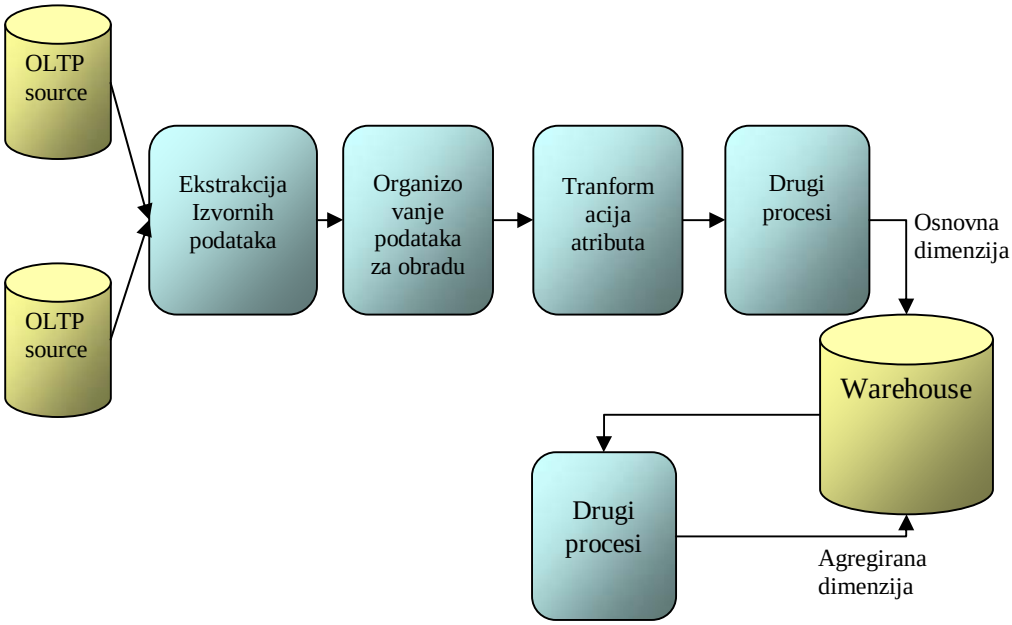
Eliminisanje redudantnog procesa

Druga karakteristika procesa učitavanja podataka u agregiranu šemu iz osnovne šeme je eliminaisanje redudantnog procesa. Na Slici 4.7 su prikazani redudantan proces i proces u kome ne dolazi do redudantnosti. Prvo ćemo razmotriti šta se dešava kada se agregirana šema učitava iz istog izvora kao i osnovna šema.

Redudantan proces



Neredudantan proces



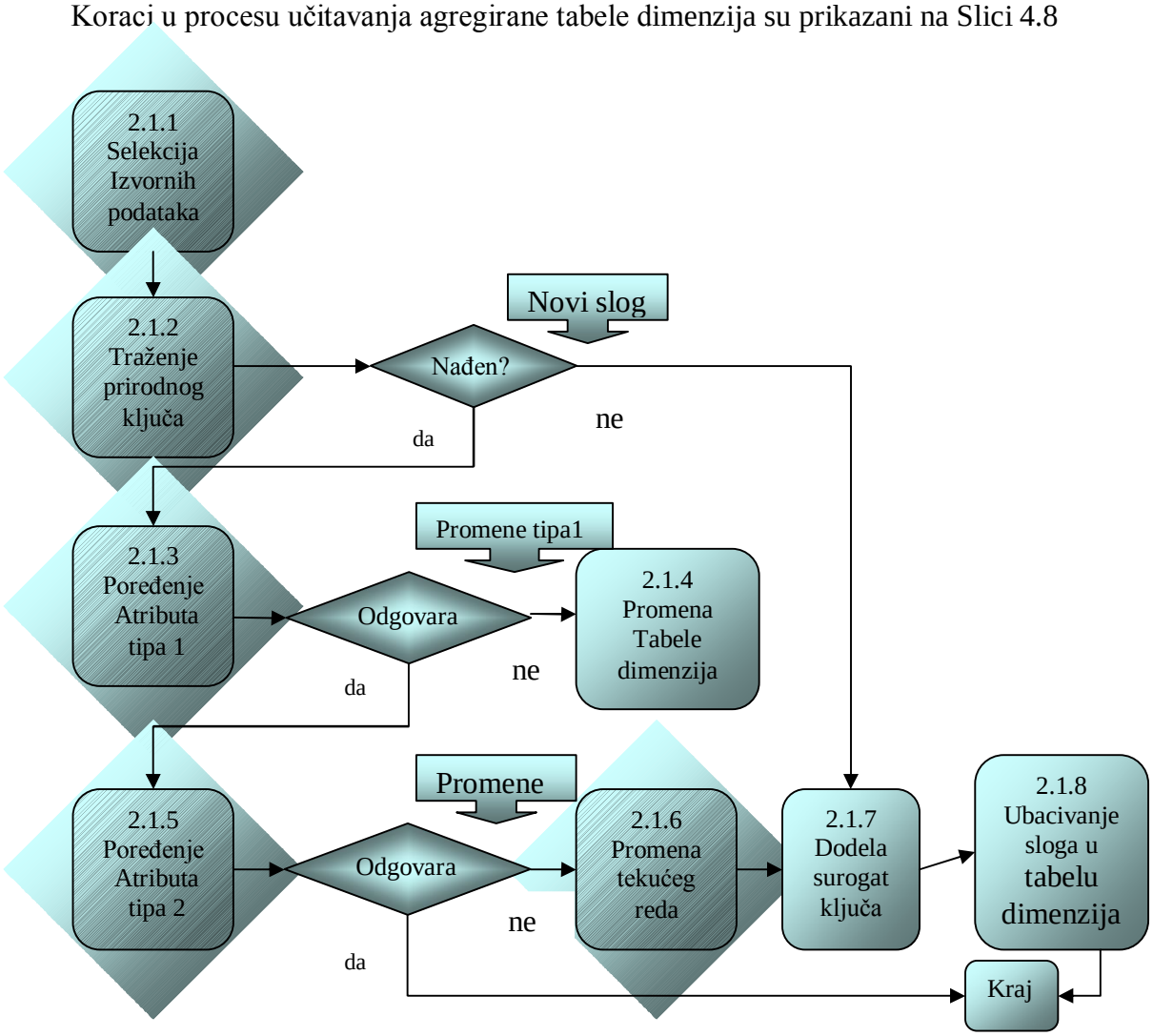
Slika 4.7. Načini na koji se učitavaju podaci u agregiranu tabelu

Svaki atribut koji se nalazi u obe šeme biće dva puta ekstrahovan. Ukoliko postoje višestruki izvori podataka, tada će svaki zahtev ili svaki integrisan proces biti dva puta izvršen. Ako su podaci u izvornom sistemu hijerarhijski organizovani ili nisu u normalnoj formi, tada će svaka zahtevana transformacija ili normalizacija biti ponovljena.

Učitavanjem agregirane šeme iz osnovne šeme eliminišu se navedene redudantnosti. Ekstrakcija i priprema podataka iz OLTP izvora se izvršava jednom, tranformacija atributa se obavlja jednom.

4.3.2.1 Učitavanje podataka u agregiranu tabelu dimenzija

Proces učitavanja podataka u agregiranu tabelu dimenzija je veoma sličan procesu učitavanja osnovne tabele dimenzija. Najpre se moraju ekstrahovati i pripremiti podaci, identifikovati nove slogove i promene tipa 1 i tipa 2. Takođe je potrebno odrediti surogat ključ. Mnogi aspekti procesa učitavanja podataka su olakšani zato što se agregacija puni iz osnovne šeme. Međutim, postoje i određene teškoće. Na primer, obrada atributa tipa 1 koji ne zavisi od nekog drugog atributa u tabeli dimenzija donosi mnoge teškoće za agregaciju. Koraci u procesu učitavanja agregirane tabele dimenzija su prikazani na Slici 4.8



Slika 4.8. Učitavanje agregirane tabele dimenzija

Prvi korak je ekstrakcija podataka. Pošto se agregacija bazira na tabelama koje se već nalaze u data warehouse-u, ovaj korak će se realizovati korišćenjem posebnog upita

nad osnovnom tabelom dimenzija. Kod učitavanja podataka u agregiranu tabelu nije potrebno organizovati podatke zato što je ovaj proces transformacije već obavljen prilikom učitavanja osnovne šeme.

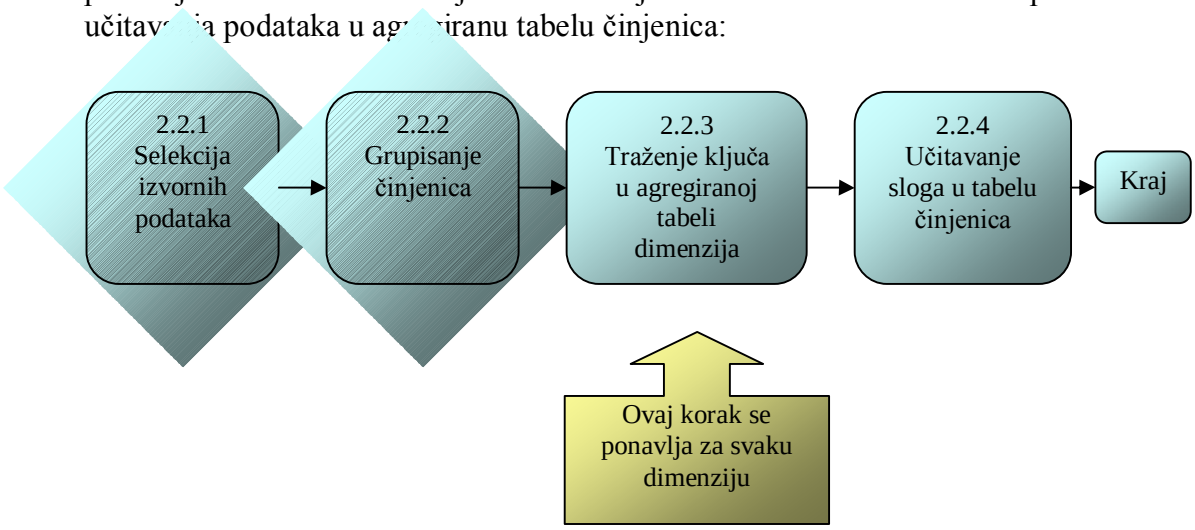
Prilikom učitavanja novog sloga potrebno je ispitati da li on već postoji u agregiranoj tabeli dimenzija. To se postiže ispitivanjem prirodnih ključeva. Ako prilikom traženja ne dođe do poklapanja prirodnih ključeva potrebno je učitatii slog u tabelu. Novom slogu se dodeljuje surogat ključ (korak 2.1.7) i isti se ubacuje u tabelu (korak 2.1.8).

Promene tipa 1 se utvrđuju poređenjem atributa tipa 1 sloga koji se ubacuje sa atributima svih sloga koji imaju isti prirodni ključ kao novi slog (korak 2.1.3). Ako se vrednosti ne slažu, promene tipa 1 su identifikovane i dolazi do promena u tabeli dimenzija (korak 2.1.4).

Svaki slog koji nije označen kao nov može da sadrži i promene tipa 2. Kao i kod promena tipa 1, dolazi do poređenja atributa tipa 2 svih slogova koji imaju isti prirodni ključ kao i slog koji se učitava (korak 2.1.5). Ako se vrednosti ne slažu, slogu se dodeljuje surogat ključ i ubacuje se u tabelu.

4.3.2.2 Učitavanje podataka u agregiranu tabelu činjenica

Učitavanje agregirane tabele činjenica je analogan procesu učitavanja osnovne tabele činjenica. Pomoću određenog upita vrši se selekcija neophodnih podataka. Pretraživanje je neophodno samo kod traženja surogat ključa kojim se tabela činjenica povezuje sa tabelom dimenzija. Na sledećoj slici dati su osnovni koraci u procesu učitavanja podataka u agregiranu tabelu činjenica:



Slika 4.9. Učitavanje agregirane tabele činjenica

Prvi koraci su selekcija izvornih podataka i grupisanje činjenica. Na Slici 15 to su koraci 2.2.1 i 2.2.2, respektivno. Ova dva koraka se najčešće obavljaju odvojeno, ali mogu biti izvršena istovremeno, postavljanjem jednog upita koji će obaviti oba zadatka. Navedenim upitom se vrši selekcija samo novih slogova.

Sledeći korak je određivanje surogat ključa korišćenjem prirodnog ključa. Ovaj proces je analogan procesu traženja ključa za osnovnu tabelu činjenica. Na osnovu prirodnog ključa sloga koji se učitava traži se odgovaraju slog u agregiranoj tabeli dimenzija. Kada je pogodan slog pronađen, odgovarajući surogat ključ se dodeljuje slogu prilikom učitavanja u agregiranu tabelu činjenica.

5 PRAKTIČAN PRIMER PRIMENE ETL-a

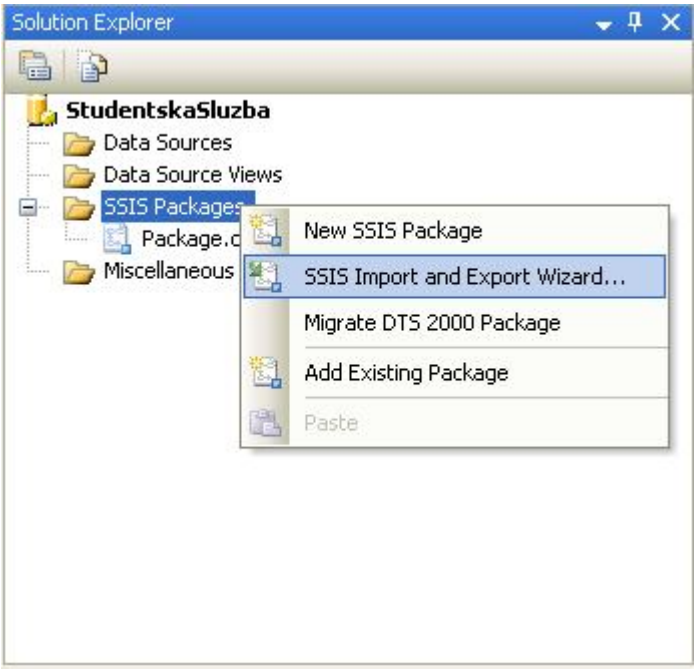
U ovom poglavlju su prikazane neke od mogućnosti primene tehnika i alata za analizu podataka iz baze podataka studentske službe. Primer je realizovan korišćenjem MS SQL Server-a 2005 i njegovih alata za poslovnu inteligenciju, koji se nalaze u modulu SQL Server Business Intelligence Development Studio.

5.1 Definisanje problema

Potrebno je obaviti operaciju uvoženja podataka u naš sistem iz spoljnog sistema pomoću alatke Integration Services. Ova alatka omogućuje ekstrahovanje podataka (*Extract*), transformaciju (*Transform*) i učitavanje podataka (*Load*) u željenu bazu podataka. Osnovni zadatak je utvrđivanje korelacije između podataka, analiza međuzavisnosti po različitim kriterijumima i donošenje zaključaka na osnovu njih.

5.2 Istraživanje podataka

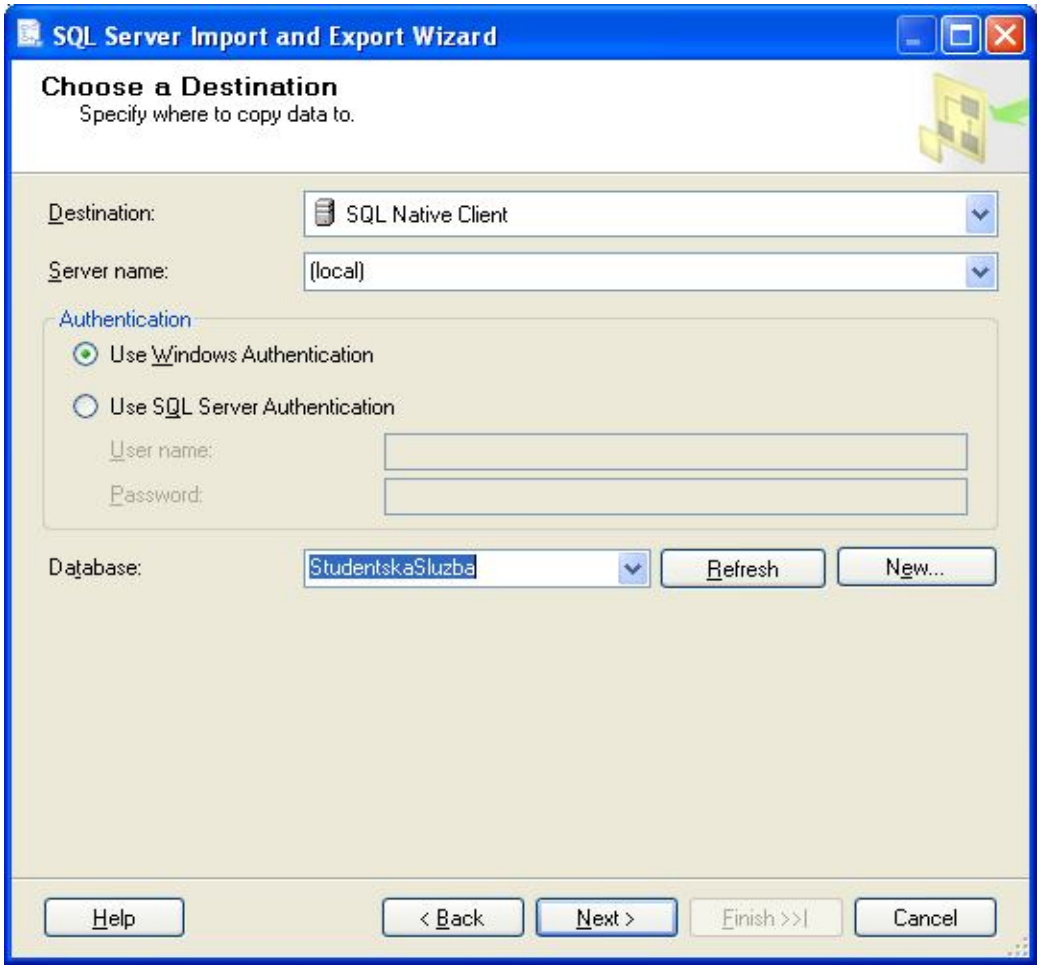
Nakon pokretanja novog projekta otvara se prozor u kome se nalazi čvor u Solution Exploreru pod nazivom "SSIS Packages". Pritiskom desnog tastera miša na ovaj čvor otvara se padajući meni u kome biramo čvor SSIS Import and Export Wizard.



Slika 5.1. Formiranje novog toka korišćenjem čvora SSIS Import and Export Wizard

Ovo nam donosi uvodni dijalog koji je prilično beskoristan, ali pritiskom na polje Next otvara se dijalog koji nam omogućava uspostavljanje veze sa Data Source. Data Source je konekcija sa podacima, koja je snimljena i sa kojom se upravlja u okviru projekta. Format baze podataka odgovara Microsoft Access-u.

U padajućem okviru za aktuelni izvor podataka biramo Microsoft Access, dok u polju File name unosimo putanju do naše baze. Pritiskom na Next otvara se dijalog u kome biramo bazu u koju želimo da kopiramo podatke iz spoljnog sistema.

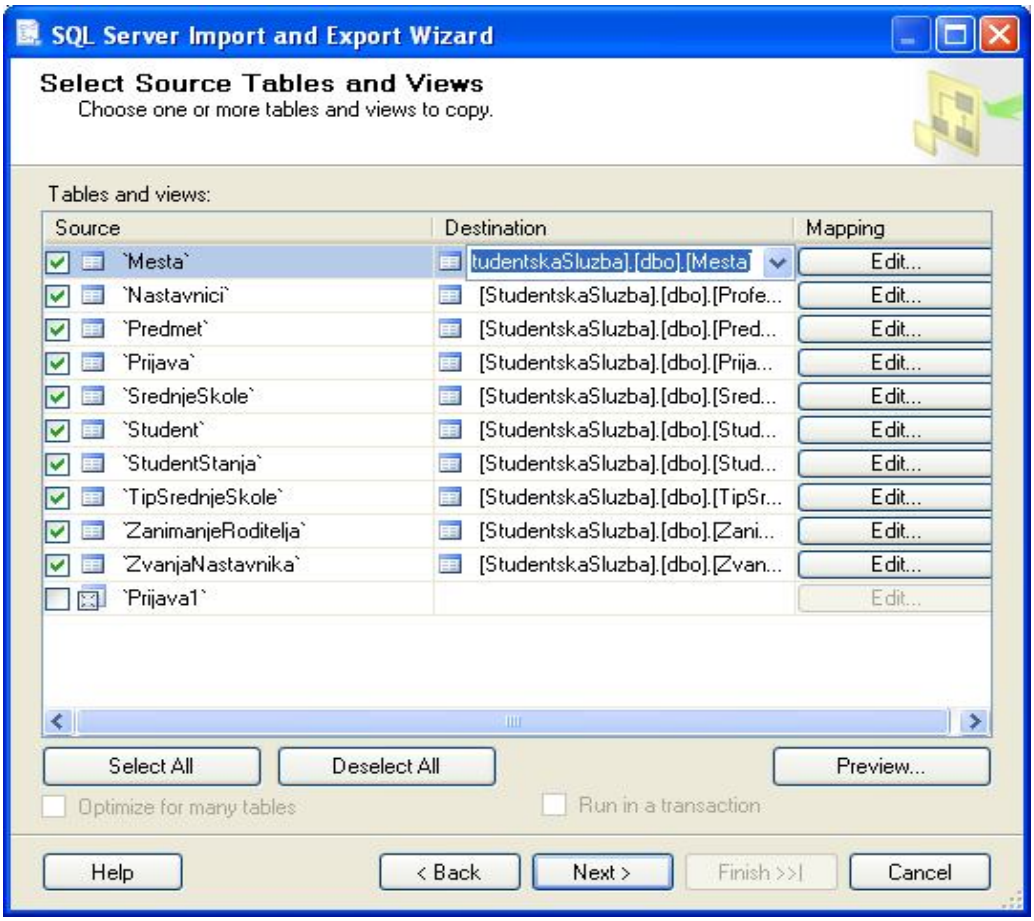


Slika 5.2. Izbor baze u koju se kopiraju podaci

Destinacija je SQL Native Client, dok u polju Database biramo bazu u koju želimo da kopiramo podatke. Baza je predhodno napravljena u SQL Server Menagment Studiju i ima istu strukturu kao i baza u Access-u. Ovde je moguće postaviti informacije za proveru autentičnosti (u ovom slučaju korisi se Windows Authentication). Pritiskom na Next otvara se prozor u kome možemo da izaberemo koje podatke želimo da kopiramo. Na raspolaganju su dve opcije:

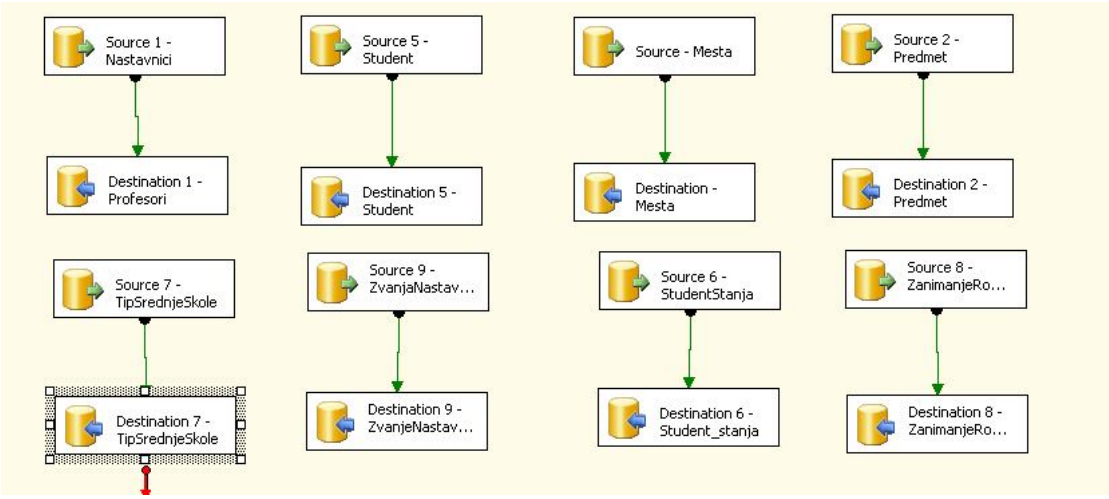
- **Copy**: Ovom opcijom direktno se kopiraju svi podaci iz tabele.
- **Paste**: Ova opcija omogućuje izbor nekih podataka koje dobijamo postavljanjem upit.

U ovom primeru izabrana opcija je Copy. Pritiskom na Next, dobija se dijalog u kome biramo koje tabele želimo da kopiramo. Veze između tabela, u pogledu ključeva i ograničenja, nisu dobro definisane.



Slika 5.3. Selekcija željenih tabela

U ovom primeru izabrane su sve tabele. Pritiskom na Next paket je napravljen, ali on neće biti izvršen odmah. Potrebno je izvršiti ga ručno pritiskom čvora Execute Package. Na sledećoj slici prikazan je deo tokova podataka za izabrane tabele.



Slika 5.4. Data Flow

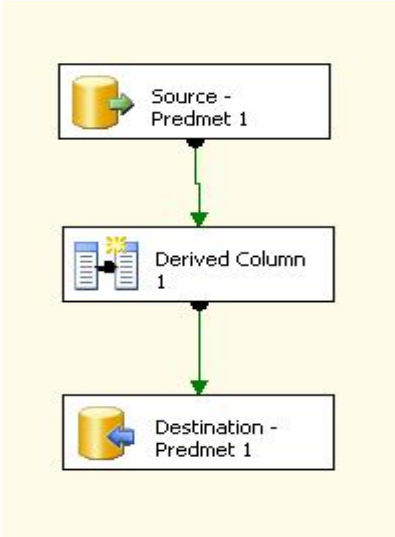
Podaci smešteni u navedenim tabelama se detaljno razmatraju. Potrebno ih je pripremiti, prečistiti i konsolidovati, kako bi dobijeni modeli i rezultati analiza bili tačni i upotrebljivi. Neophodno je najpre odrediti primarne ključeve i referencijalni integritet između datih tabela, zato što on nije definisan u polaznoj bazi.

Prvo će biti razmotrena tabela Predmet. Atributi tabele dati su na sledećoj slici.

Column Name	Data Type	Allow Nulls
IDPredmet	nvarchar(50)	☐
Sifra_obaveze	nvarchar(50)	☑
Statut	nvarchar(50)	☑
Grupa	nvarchar(50)	☑
Opis	nvarchar(50)	☑
Tip_obaveze	nvarchar(50)	☑
Broj_semestra	float	☑
Pun_naziv	nvarchar(120)	☑

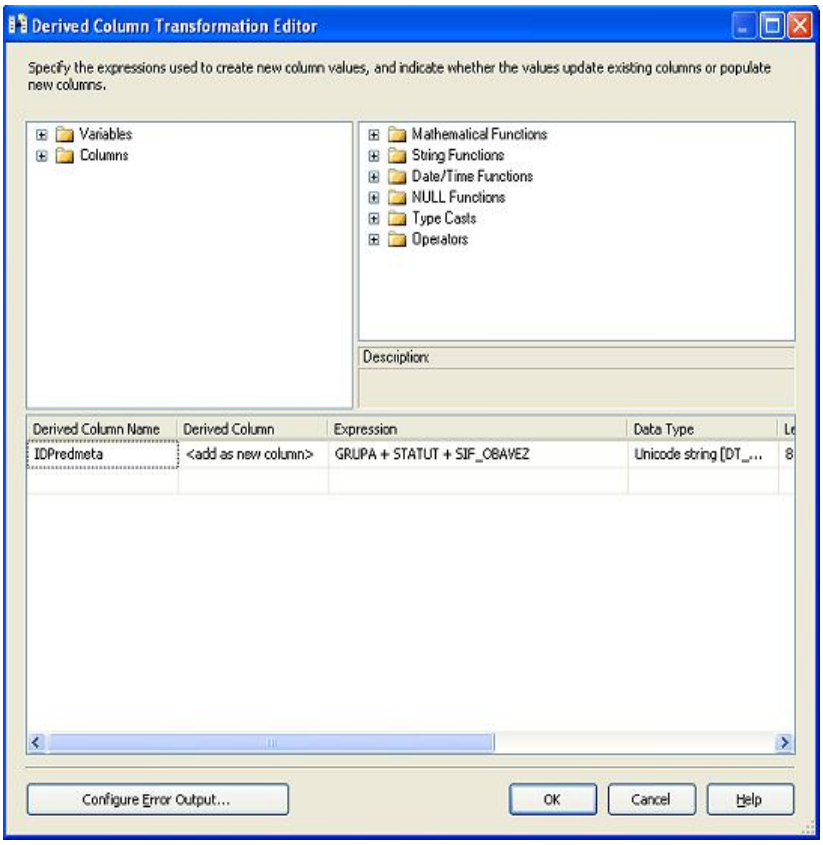
Slika 5.5 Atributi tabele Predmet

Kao što se vidi primarni ključ je IDPredmet, međutim ova kolona nije postojala u polaznoj bazi. Njeno definisanje bilo je neophodno zbog određivanja primarnog ključa. Na slici je prikazan tok podataka (Data Flow) za navedenu transformaciju.



Slika 5.6. Data Flow

U čvoru Derived Column definišemo novu kolonu IDPredmeta koja će predstavljati primarni ključ.



Slika 5.7. Definisanje nove kolone pomoću čvora Derived Column

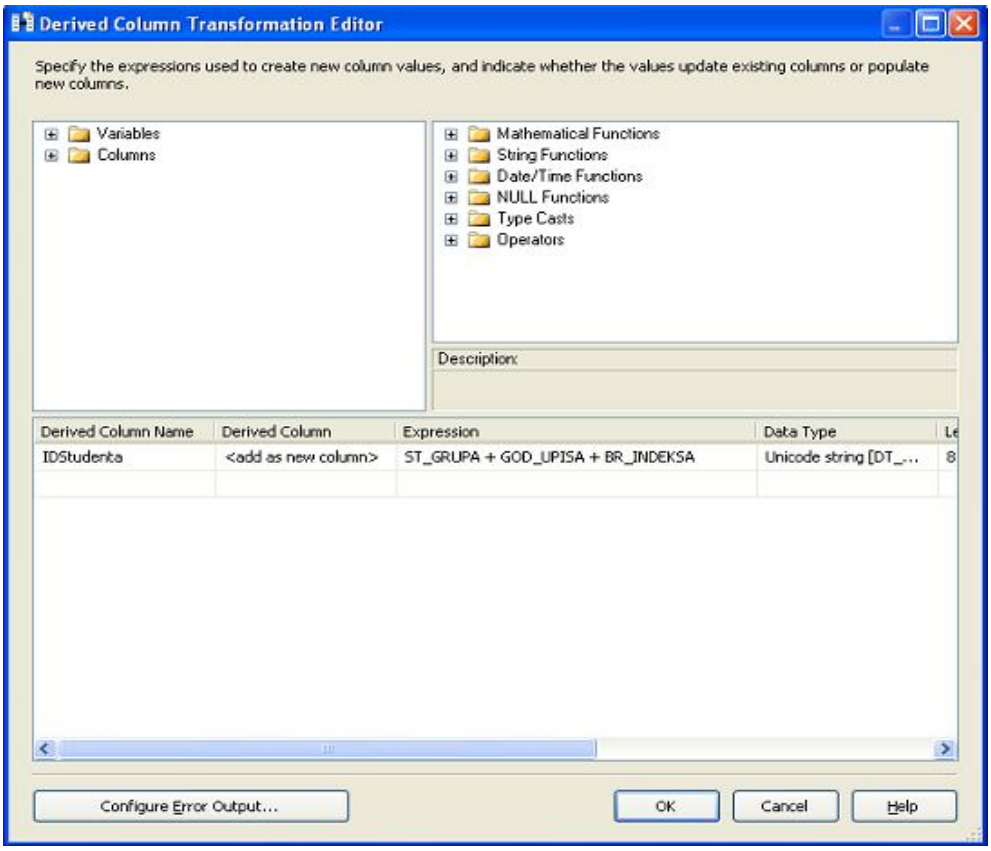
IDPredmeta predstavlja kombinaciju atributa *GRUPA,STATUT i SIF_OBAVEZE.* Tek sada atribut IDPredmet može biti setovan kao primarni ključ.

Isti problem postoji i u tabeli Student. Atributi tabele su dati na Slici 5.8.

Column Name	Data Type	Allow Nulls
Sifra_studenta	nvarchar(50)	☐
Broj_indexa	nvarchar(50)	☑
Godina_upisa	nvarchar(50)	☑
St_grupa	nvarchar(50)	☑
Statut	nvarchar(50)	☑
Ime	nvarchar(50)	☑
Prezime	nvarchar(50)	☑
Ime_roditelja	nvarchar(50)	☑
Zanimanje_roditelja	nvarchar(50)	☑
Licni_broj	nvarchar(50)	☑
Datum_rodjenja	datetime	☑
Post_broj	nvarchar(50)	☑
Drzavljani	nvarchar(50)	☑
Pol	nvarchar(50)	☑
Sifra_stanja	nvarchar(50)	☑
Datum_stanja	datetime	☑

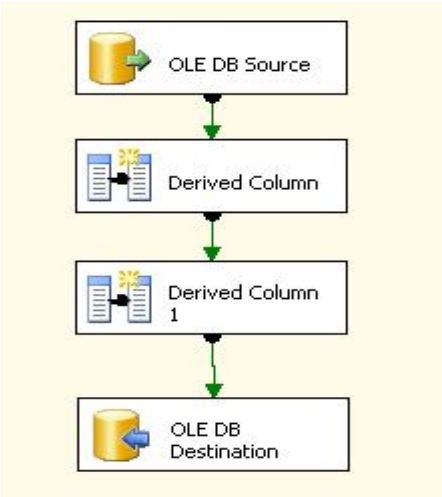
Slika 5.8. Atributi tabele Student

Sifra_studenta je nova kolona definisana pomoću čvora Derived Column na isti način kao kolona u predhodnoj tabeli.



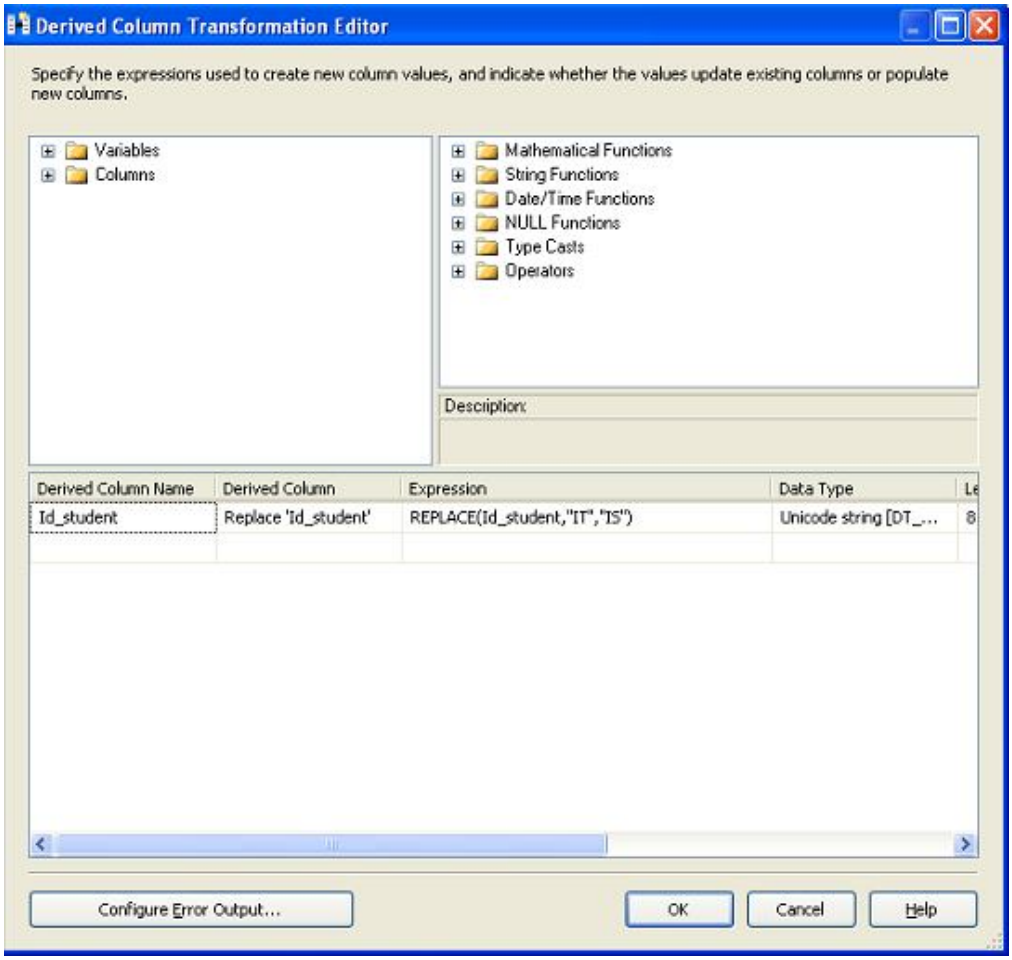
Slika 5.9. Definisanje nove kolone pomoću čvora Derived Column

Atribut Sifra_studenta, koji je primarni ključ, u ovom slučaju predstavlja kombinaciju atributa *St_grupa, Godina_upisa i Broj_indexa* . Kao što vidimo u sastav ključa ulazi sifra smera koji je student upisao. Pošto su se nazivi smerova menjali, tako da postoji više oznaka za isti smer, dolazi do neefikasnosti prilikom pretraživanja baze. U cilju poboljšanja performansi urađene su određene tranformacije. Na primer, smer Informacioni sistemi javlja se pod oznakom *IT* i *IS*. Da bi se olakšalo pretraživanje baze oznaka *IT* je transformisana u oznaku *IS*. Ovo je urađeno pomoću čvora Derived Column. Tok podataka (Data Flow) je prikazan na Slici 5.10.



Slika 5.10. Data Flow

U čvoru Derived Column1 menjamo kolonu Id_studend pomoću string funkcije *REPLACE()*, dok se u prvom čvoru Derived Column formira nova kolona Id_student.



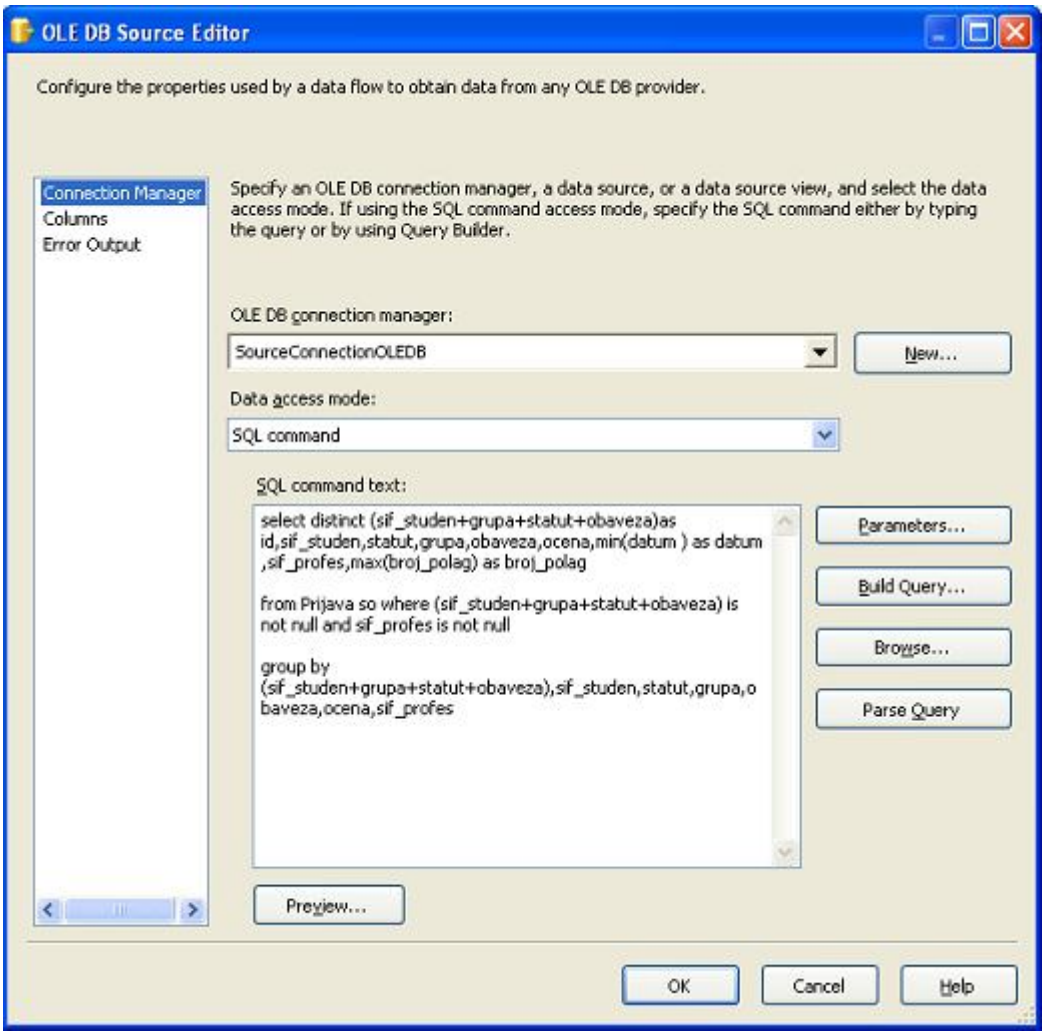
Slika 5.11. Transformacija pomoću čvora Derived Column

Sledeća tabela je Prijava. Atributi tabele su dati na Slici 5.12. Kao što se vidi, primarni ključ je složen zato što tabela Prijava predstavlja agregaciju tabela Predmet i Student.

Column Name	Data Type	Allow Nulls
ID_student	nvarchar(50)	☐
ID_Predmet	nvarchar(50)	☐
Statut	nvarchar(50)	☑
Grupa	nvarchar(50)	☑
Obaveza	nvarchar(50)	☑
Ocena	float	☑
Datum	datetime	☑
Sifra_profesora	nvarchar(50)	☑
Broj_polaganja	float	☑

Slika 5.12. Atributi tabele Prijava

Međutim, ovde se javljaju još neke poteškoće. Naime, podaci u bazi nisu dobro unošeni tako da se javljaju duplikati. Pojedini studenti su evidentirani dva puta za isti ispit, što je nemoguće. Ovaj problem je rešen postavljnjem upita u OLE DB Source-u. Upit je dat na Slici 5.13

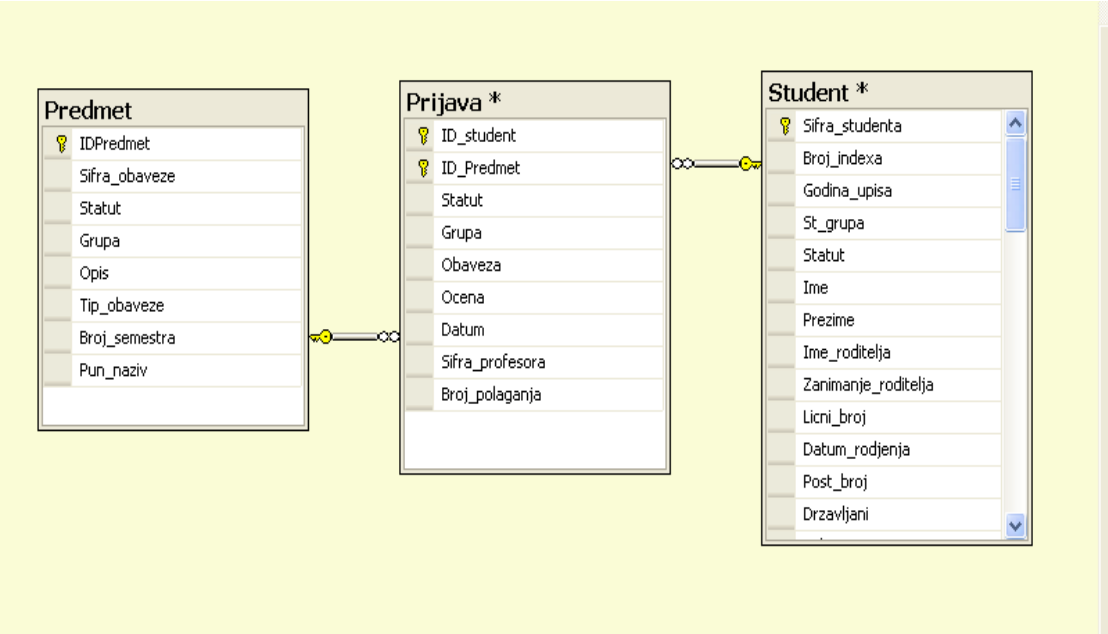


Slika 5.13. OLE DB upit

Ovim upitom selektovani su studenti koji su jednom položili ispit, a oni studenti koji su evidentirani dva puta sa istim ispitom, selektovani su tako da im se datum prvog polaganja prizna kao datum zvanično položenog ispita. Tek nakon realizacije ovog upita primarni ključevi u tabeli Prijava mogu biti setovani.

Kod ostalih tabela kolona koja treba da predstavlja primarni ključ već postoji, tako da nisu potrebne dodatne transformacije. Sledeći problem koji se javlja su ograničenja između tabela. Ograničenja obezbeđuju da podaci ispunjavaju određena pravila koja se odnose na integritet podataka. Za obezbeđivanje integriteta podataka nisu odgovorni programi koji koriste bazu podataka, već sama baza podataka. Mnogi problemi koji se odnose na to šta programi dozvoljavaju unutar baze podataka, sada se uočavaju mnogo ranije u procesu razvoja zato što baza podataka zna da odbaci podatke, iako je korisnički program dozvolio njegov pristup.

Najpre ćemo posmatrati veze između entiteta *Predmet, Student i Prijava*. Kao što je predhodno rečeno entitet *Prijava* predstavlja agregaciju druga dva entiteta. Veze između ovih entiteta prikazuju se pomoću *Dijagrama*. Dijagrami predstavljaju važan alat za efikasno projektovanje baze podataka.



Slika 5.14. Dijagrami skladišta podataka

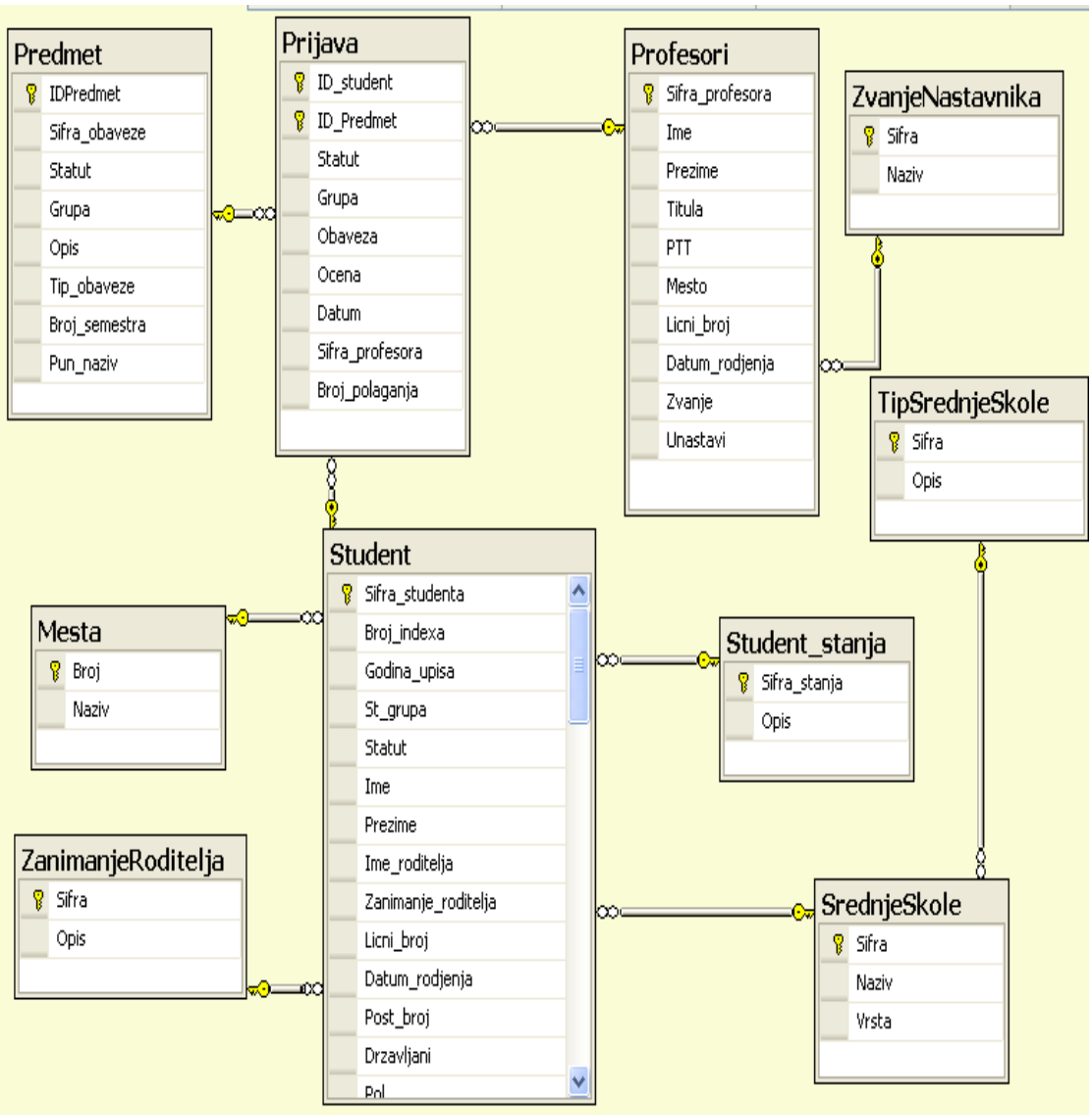
Nakon definisanja veza između navedenih entiteta javljaju se neki problemi. U *Prijavi* postoje studenti čija šifra ne postoji u tabeli *Student*. Neophodno je pronaći te studente pomoću sledećeg upita:

```
select distinct ID_student
from Prijava p
left join Student s on p.ID_student=s.Sifra_studenta
     where s.Sifra_studenta is null
```

Nakon realizacije ovog upita dobijamo sifru studenta koji ne postoji u tabeli *Student*. Student sa datom šifrom se izvršavanjem sledećeg upita učitava u tabelu *Student*.

```
INSERT INTO
Student(sifra_studenta,broj_indexa,godina_upisa,st_grupa,statut,im
e,prezime,ime_roditelja,zanimanje_roditelja,licni_broj,datum_rodje
nja,post_broj,drzavljani,pol,srednja_skola,sifra_skole,god_zav_sre
dnje,poeni_prij,rodno_mesto)
VALUES ('100000','III-6-07', '07', 'PR', '94', 'Branislav',
'Ničić', 'Miodrag', '1', '2008984910000', '1984-20-08', '11000',
'SRB', 'M', '36', 'ETSRK', '03',34.9,'BEOGRAD');
```

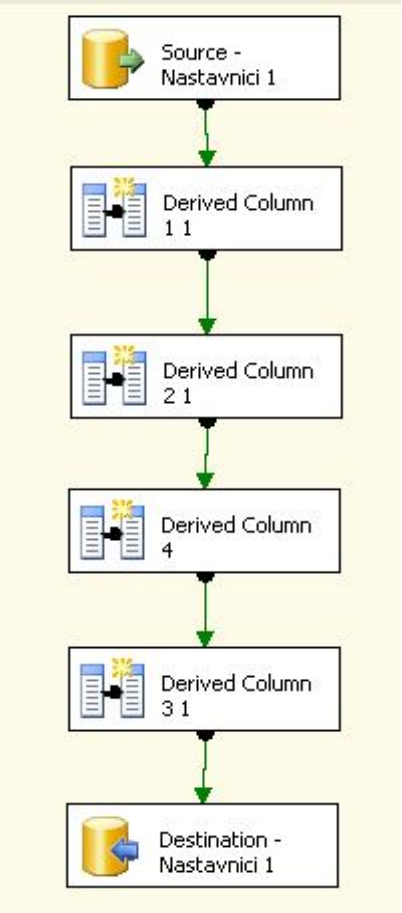
Nakon realizacije upita integritet između datih entiteta je dobro definisan. Sada možemo dodati druge tabele i uspostaviti veze između njih kao što je prikazano na Slici 5.15.



Slika 5.15. Dijagrami skladišta podataka

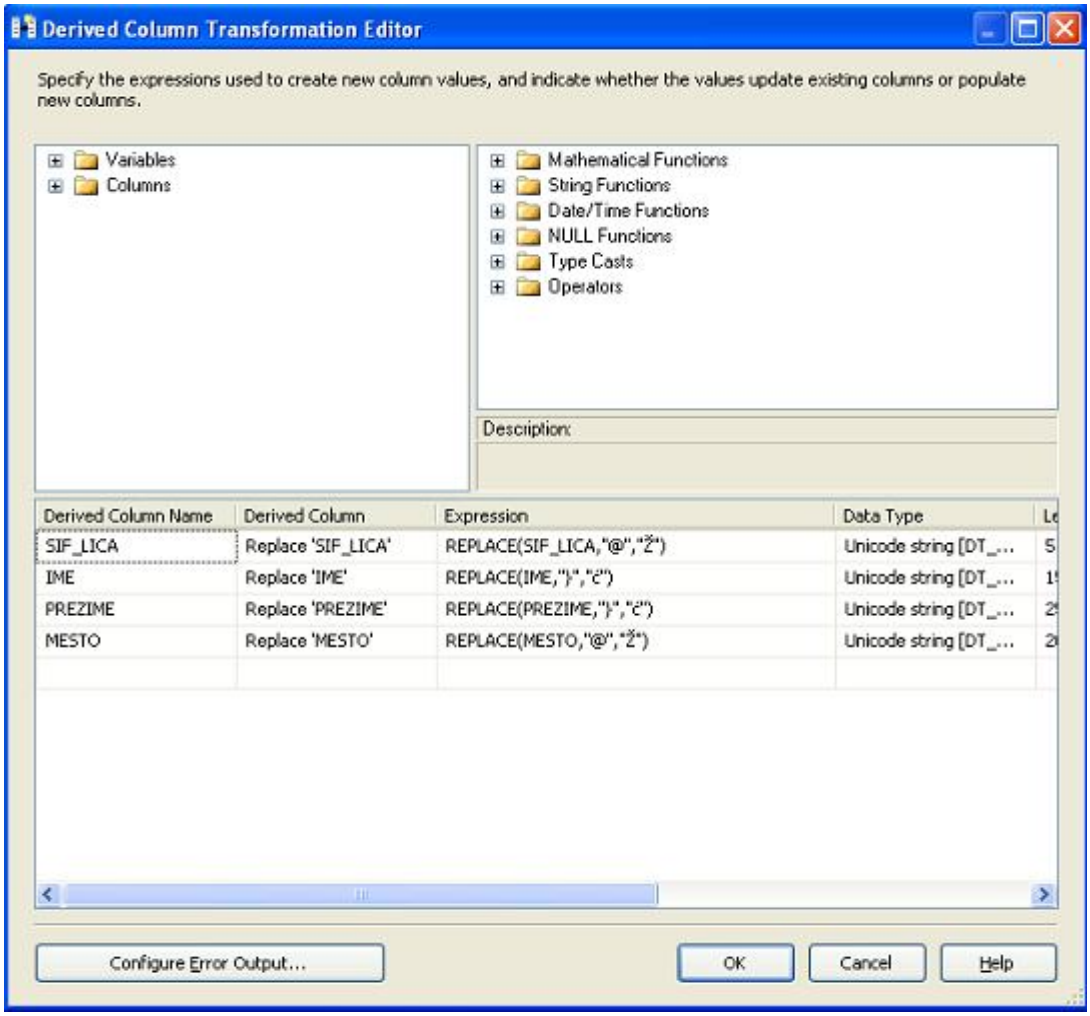
Nakon definisanja ovih veza i ograničenja podaci mogu biti učitani u bazu. Međutim, u polaznoj bazi podaci nisu kompletni, tako da će sva mesta koja su prazna biti popunjena NULL vrednostima.

U polaznoj bazi postoje i dodatni problemi. Na primer, pojedini opisi i nazivi sadrže određene znakove kao što su slova č,ć,š itd. U cilju eliminisanja ovih nedostataka napravljieni su dodatni tokovi za sve tabele. Kao primer uzećemo tabelu Nastavnici. Tok je prikazan na sledećoj slici:



Slika 5.16. Data Flow

U čvoru Derived Column pomoću string funkcije pronađene su odgovarajuće oznake i iste zamenjene slovima. Kolone u tabeli *Nastavnik* koje je potrebno transformisati su *Sif_lica, Ime, Prezime, Mesto*. Na primer, umesto znaka '@' je veliko slovo Ž, umesto znaka '{' je slovo š itd.



Slika 5.17. Transformacija pomoću čvora Derived Column

Navedena transformacija je urađena za sve tabele u kojima postoji dati problem na isti način.

6 ZAKLJUČAK

U predviđanju daljeg toka razvoja poslovne inteligencije ističu se dva bitna pravca: poslovno-inteligentna rešenja i odgovori u realnom vremenu, kao i veća demokratizacija sistema BI.

Permanentne promene u svetu potrošača primoravaju kompanije na pružanje mnogo bržih i efikasnijih usluga u poslovanju. Da bi ostale konkurentne i otkrile trendove i buduće događaje, moraju u potpunosti zadovoljiti očekivanja i želje komintenata i mnogo snažnije se oslanjati na sisteme poslovne inteligencije. Korisnici poslovne inteligencije sada zahtevaju rešenja u realnom, ili skoro stvarnom vremenu, sa analizom poslovnih dešavavanja. Uskoro će se očekivanja odnositi na trenutne i sveže informacije, a mesečni i nedeljni izveštaji neće odgovarati.

”BI 2.0” je nedavno kreiran termin kao deo kontinualnog razvoja poslovne inteligencije i nagoveštaj sledećeg koraka. Opisuje akviziciju, snabdevanje i analizu trenutnih podataka i to na sofisticiraniji način od već postojećeg. Potrebno je obratiti pažnju na dva problema. Prvi se odnosi na uspostavljanje uniformnosti i standardizovanja poslovnih pravila i struktura, a drugi na povećanje adaptivnosti kada su u pitanju neprestane promene u strukturi samih kompanija.

U narednih nekoliko godina poslovne informacije će postati više ”demokratske”, a krajnji korisnici širom kompanije će relativno jednostavno moći da prave upite i poglede nad specifičnim segmentima poslovanja i da imaju uvid u za njih ključne performanse. Mogućnosti i kvalitet rešenja aplikacija poslovne inteligencije će rasti istom brzinom kao i zahtevi potrošača. Termini poslovna inteligencija, data mining i sl. će polako izlaziti iz upotrebe, a zameniće ih poslovne akcije koje će automatski pokretati sistemi sa ”korporativnim predviđanjem”. Umesto ograničenog broja tehničkih analitičara, ove tehnologije će biti primenjene na svim nivoima. [2]

Danas se javljaju sve kompleksniji zahtevi za projektovanjem i implementacijom adekvatnih informacionih sistema i tehnologija kao infrastrukturne osnove svih procesa u poslovnom sistemu. Globalni trendovi, dinamičnost, turbulentnost primoravaju poslovne organizacije na visok stepen efikasnosti, adaptilnosti, integracije i koordinacije aktivnosti. U tom kontekstu može se govoriti o poslovnoj inteligenciji kao realizaciji potrebe za dodatnim, neotkrivenim, neopaženim znanjem i mogućnostima.

7 LITERATURA

[1] Ćirić B., *Poslovna inteligencija*, Data status, 2006.

[2] Raisinghani M., *Business Intelligence in the Digital Economy: Opportunities, Limitations and Risks*, Idea Group Publishing, 2004.

[3] Reinschmidt J., Francoise A, *Business Intelligence Certification Guide, e-book*, IBM Redbooks, 2001.

[4] Grupa autora, *Data Modeling Techniques for Data Warehousing*, e-book, IBM Redbooks, 2001.

[5] Njeguš A., *Poslovbi informacioni sistemi*, Univerzitet Singidunum, 2008.

[6] Books On Line, *help uz Microsoft SQL Server 2005*, Microsoft Corporation, 2005.

[7] Kimball R., *Data Warehouse Toolkit* , John Wiley & Sons, 2000;

[8] http://www.skladistenje.com

[9] http://www.dwinfocenter.org

[10] http://en.wikipedia.org/wiki/

[11] http://www.rkimbal.com/html/articles.html