SKLADIŠTE PODATAKA - UVOD

Definicije

Skladište podataka (data warehouse): Domenski orijentisan, integrisan, vremensko promenljiv i neuništiv skup podataka namenjen podršci odlučivanju kod upravljanja nekim sistemom.

Podskaldište podataka (data mart): Poseban izdvojeni deo skladišra podataka namenjen piotrebama nekog dela sistema.

Web skladište podataka (data webhouse): Distribuirano skladište podataka implementirano preko web-a (za koje ne postoji centralizovano čuvanje podataka).

U sva tri slučaja, reč je o strateškom IS, za razliku od operacionog IS.

Pogodnosti

Visok stepen povraćaja ulaganja
Povećanje konkurentnosti
Povećanje produktivnosti odlučivanja
Povećanje kvaliteta odlučivanja

Poređenje

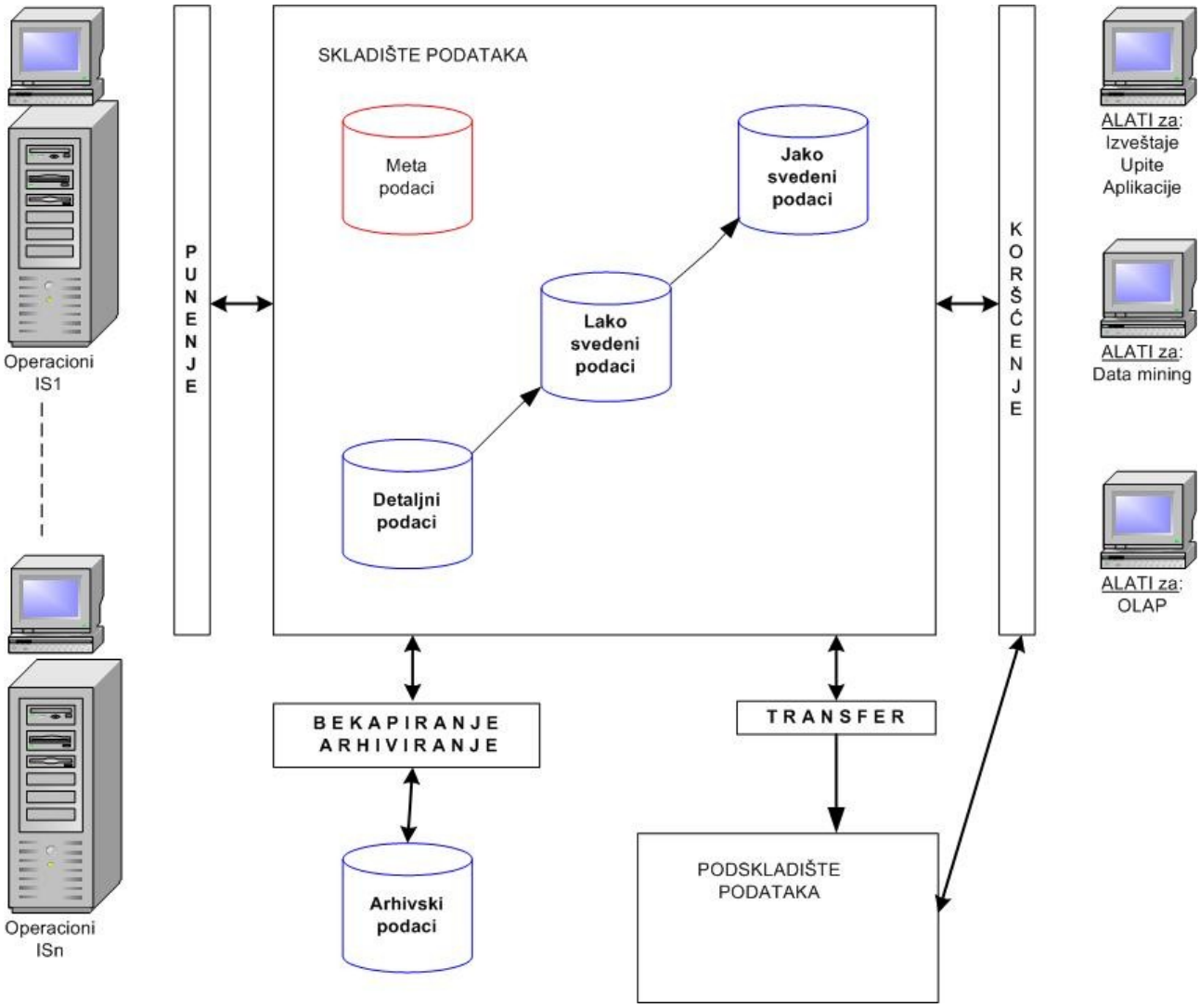
Operacioni (OLTP) IS	Strateški (OLAP) IS
Trenutni podaci stanja i prometa	Istorijski podaci stanja i prometa
Detaljne podatke	Detaljni i srednje i visoko svodni podaci
Dinamički podaci	Ugalvnom statički podaci
Ponavljajuća predefinisana obrada	Od hoc obrada po zahtevu
Visok nivo transakcione aktivnosti	Nizak / srednji nivo transakcione aktivnosti
Predvidiv način korišćenja	Nepredvidiv način korišćenja
Transakciono orijentisan	Analitički orijentisan
Aplikativno orijentisan	Domenski orijentisan
Podrška svakodnevnom odlučivanju	Podrška strateškom odlučivanju
Opslužuje veliki broj korisnika	Opslužuje manji broj korisnika

Problemi koji se javljaju u vezi skladišta podataka

Podcenjivanje resursa potrebnih za punjenje podacima
Skriveni problemi unutar izvornih IS
Neobuhvatanje neophodnih podataka unutar izvornih IS
Semantika i homogenizacija podataka
Visoki zahtevi za resursima
Vlasnisštvo/pristup podacima
Obimno naknadno održavanje
Dugoročnost projekta (>= 3 godine)
Kompleksnost integracije sistema

SKLADIŠTE PODATAKA – USTROJSTVO I RAD

Struktura skladišta podataka



DATA WAREHOUSE - skladište podataka (globalna namena)
DATA MART - podskladište podataka (specifična namena)

Implementacija skladišta podataka

Uobičajeno je da se vremenom iz jednog skladišta podataka formiraju podskladišta podataka, pri čemu isti podaci i dalje ostaju u skladištu podataka.

U praksi se dešava obrnuto - prvo se realizuju pojedina kritična podskladišta podataka koja se direktno pune iz operacionih IS, a zatim se naknadno formira skladište podataka. Od tog trenutka, tok podataka iz operacionih IS je ka skladištu podataka, a podskladišta podataka se pune transferom iz skladišta podataka.

Implementacija skladišta podataka je preko baze podataka (uglavnom relacione) sa visokim stepenom redudanse, a mehanizam svođenja je zasnovan na okidačima.

Po drugoj implementacionoj klasifikaciji, imamo:

- ✔ Realno skladište podataka: Podaci skladišta realno postoje, odvojeno od informacionog sistema operativne namene. Operacije nad skladištem podataka ne opterećuju informacioni sistem operativne namene.

- ✔ Virtuelno skladište podataka: Podaci skladišta realno ne postoje, nego sve svaki put izvode kao pogledi nad podacima informacionog sistema operativne namene. Operacije nad skladištem podataka opterećuju informacioni sistem operativne namene, pa je ova varijanta prihvatljiva samo mod manjih sistema.

Punjenje skladišta podataka

U principu postoje dva načina punjenja podacima iz operacionih IS:

- ✔ Totalno punjenje: U određenim vremenskim trenucima, skladište se isprazni a zatim ponovo napuni podacima iz operacionih IS.

- ✔ Inkrementalno punjenje: Prilikom punjenja, u skladište se prenose samo izmene nastale u operacionim IS nakon prethodnog punjenja.

Postoje dve varijante inkrementalnog punjenja:

- ✔ Paketno inkrementalno punjenje: Vrši se u određenim vremenskim trenucima. Zahteva izmene u operacionom IS (bazi podataka) koje će implementirati mehanizam prepoznavanja nastalih izmena.

- ✔ Neprekidno inkrementalno punjenje: Vrši se neprekidno. Nakon svake promene u operacionim IS mehanizmom okidača vrši se prenos podataka ka skladištu podataka.

Konkretne tehnike inkrementalnog punjenja skladišta podataka su:

- ✔ Eksport promena u log-fajlu baze podataka (paketno).

- ✔ Eksport efekata transakcija (paketno).

- ✔ Eksport promena u bazi podataka preko medijatora (međusloj, paketno).

- ✔ Eksport promena u bazi podataka preko servisa replikacije (direktno).

Mogući problemi kod punjenja skladišta podataka, naročito izraženi kod heterogenih operacionih IS, su:

- ✔ Netačnost podataka iz operacionih IS: Pri punjenju je neophodno filtriranje podataka, odnosno odbacivanje netačnih podataka.

- ✔ Neusaglašenost podataka po tipu/preciznosti: Pri punjenju je neophodno usaglašavanje po oba osnova.

SKLADIŠTE PODATAKA – DIMENZIONI MODELI

Dimenziono modeliranje

Polazi se od toga da se atributi entiteta (tabela) mogu podeliti na dve grupe:

- ✔ atributi mere: atributi čija vrednost odražava meru (veličinu) nečega; bitna osobina je mogućnost svođenja;
- ✔ atributi dimenzije: atributi čija vrednost služi kao osnov klasifikacije i svođenja; mogu biti bez i sa varijabilnom granulacijom (nivoima svođenja).

Dva specijalna atributa dimenzije sa varijabilnom granulacijom: Prostorni i Vremenski

```
Vreme                    Prostor
  Dan                      Opstina
    Nedelja                  Grad
      Mesec                    Oblast
        Kvartal                  Drzava
          Godina                   Regija
  Dan-u-nedelji
  Dan-u-mesecu
  Deo-dana
  Sat-u-danu
```

Prostorni dimenzioni atribut ima jednu šemu hijerarhije. Najniži nivo rezolucije je GPS geografska pozicija iz koje se dalje izvode viši nivoi rezolucije. U tom pogledu očekuje se uvođenje novog standardnog tipa podatka (POSITION?) i odgovarajućih standardnih funkcija izvođenja.

Vremenski dimenzioni atribut ima 5 paralelnih šema hijerarhije. Najniži nivo rezolucije je vreme (sat-minut-sekunda-...) i za to postoji standardni tip podatka TIMESTAMP. Postoji i čitav niz standardnih funkcija koje vrše izvođenje po raznim šemama hijerarhija (TIME, DATE, DAY_OF_WEEK, MONTH itd.

Kod dimenzionog modeliranja može istovremeno da postoji više vremenskih dimenzionih atributa (na primer, Dan-u-nedelji i Sat), pod uslovom da su one nezavisne, odnosno da nizu izvodive jedna iz druge (kombinacija Dan i Dan-u-mesecu nema smisla).

Dve vrste tabela u dimenzionom modelu:

- ✔ tabela fakata (FT, jedna): sastoji se iz složenog primarnog ključa u koji ulaze ID-ovi svih dimenzija i jednog ili više atributa mere;
- ✔ tabele dimenzija (DTi, više): sastoje se iz prostog prmarnog ključa koji odgovara jednoj komonenti primarnog ključa tabele fakata i jednog ili više atributa dimenzije.

Unutar jednog skladišta ili podskladišta podataka može se nalaziti više dimenzionalnih modela.

Varijante dimenzionog modeliranja

Šema “zvezda” (star): jedan dimenzioni model

Sadrži jednu FT i za svaku dimenziju tačno jednu DT. DT mogu da sadrže denormalizovane podatke (neključne funkcijske zavisnosti koje nisu u 3. NF). Maksimalna dubina referisanja je 1.

FT --> DTi

Šema “zvezda-pahuljica” (starflake): jedan dimenzioni model

Sadrži jednu FT i niz DT koje ne mogu da sadrže denormalizovane podatke, nego umesto toga sadrže reference na dodatne informacione tabele IT. Pri tome IT mogu da sadrže denormalizovane podatke. Maksimalna dubina referisanja je 2, pri čemu pojedine dimenzione tabele ne moraju imati reference na IT.

FT --> DTi --> ITi
DTj

Šema “pahuljica (snowflake): jedan dimenzioni model

Uz prethodni uslov ima i ograničenje da ni jedna IT ne sme da sadrži denormalizovane podatke, što dovodi do proizvoljne dubine referisanja

FT --> DTi --> ITia --> ITib ...
DTj

Sve prethodno odnosilo se na pojedinačni dimenzioni model.

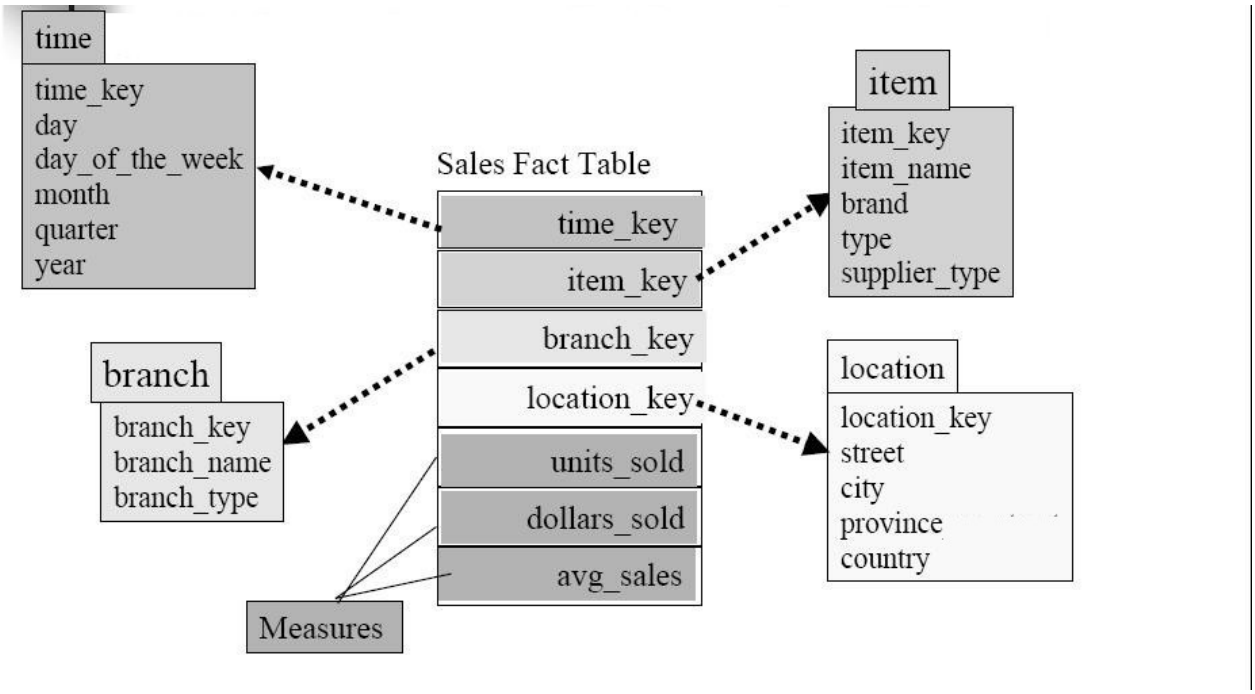
Šema sazvežđe (constellation): bar dva dimenziona modela

Situacija kada postoje bar dva dimenziona modela a njihove FT referišu bar jednu zajedničku DT. Pri tome dimenzioni modeli mogu biti bilo kog od prethodna tri tipa.

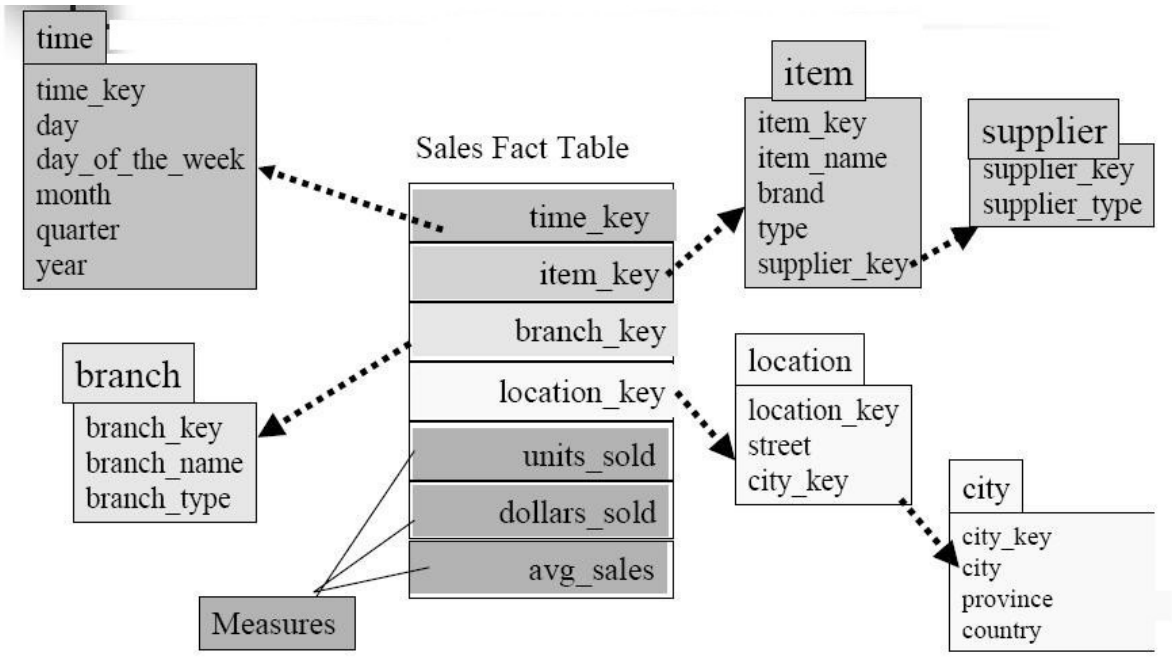
Prednost šeme "Zvezda" je smanjenje broja operacija spajanja, a mana povećanje memorijskog prostora.

Prednost ostalih šema je smanjenje memorijskog prostora, a mana povećanje broja operacija spajanja.

Primer "Zvezda": Prodaja artikala



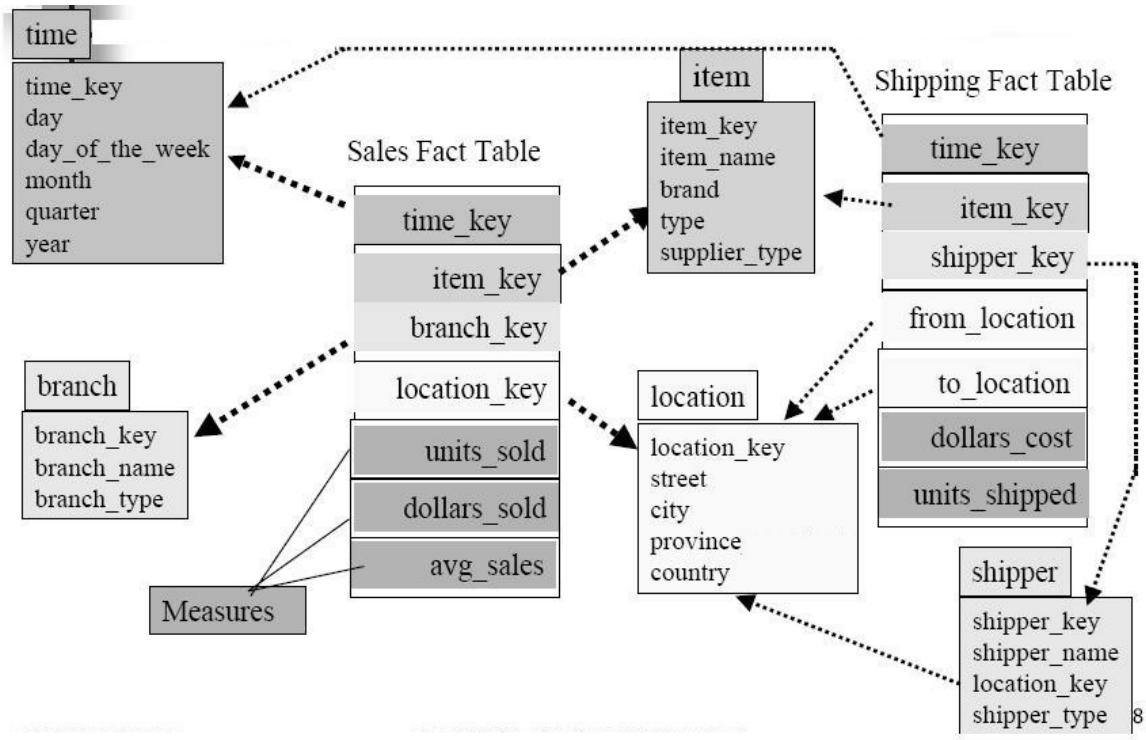
Primer "Zvezda-Pahuljica": Prodaja artikala



Napomene:

Poslednji model bi u varijanti "Pahuljica" imao dodatne informacione tabele province i country i lanac referenci location->city->province->country. Potpuna normalizacija po dimenziji time se izuzetno zbog kompaktnosti podataka ne primenjuje.

Primer "Sazvežđe":



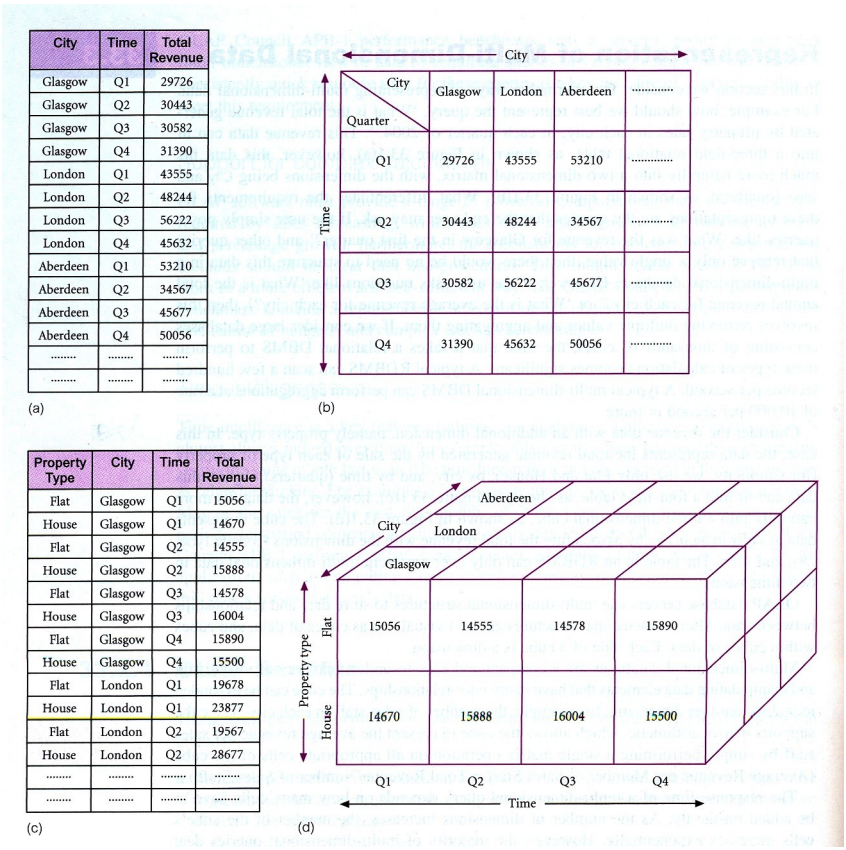
Napomene:

Prodaja (Sales) i otprema (Shipping) su dve odvojene tabele fakata zato što evidentiraju nezavisne i odvojene poslovne događaje.

U dimenzionom modelu otpreme dimenzija lokacija se javlja u dva svojstva (od i do) pa je referenca ka tabeli dimenzije lokacija.

SKLADIŠTE PODATAKA – DIMENZIONA ANALIZA (OLAP)

Osnovu OLAP dimenzione analize predstavlja OLAP kocka (CUBE) koja može biti sa 1 ili n dimezija (direktna vizualizacija je moguća do n=3).



City	Time	Total Revenue
Glasgow	Q1	29726
Glasgow	Q2	30443
Glasgow	Q3	30582
Glasgow	Q4	31390
London	Q1	43555
London	Q2	48244
London	Q3	56222
London	Q4	45632
Aberdeen	Q1	53210
Aberdeen	Q2	34567
Aberdeen	Q3	45677
Aberdeen	Q4	50056
.....	.....	.....
.....	.....	.....

(a)

Time \ City	Glasgow	London	Aberdeen	.....
Q1	29726	43555	53210	.....
Q2	30443	48244	34567	.....
Q3	30582	56222	45677	.....
Q4	31390	45632	50056	.....

(b)

Property Type	City	Time	Total Revenue
Flat	Glasgow	Q1	15056
House	Glasgow	Q1	14670
Flat	Glasgow	Q2	14555
House	Glasgow	Q2	15888
Flat	Glasgow	Q3	14578
House	Glasgow	Q3	16004
Flat	Glasgow	Q4	15890
House	Glasgow	Q4	15500
Flat	London	Q1	19678
House	London	Q1	23877
Flat	London	Q2	19567
House	London	Q2	28677
.....	.....	.....	.....
.....	.....	.....	.....

(c)

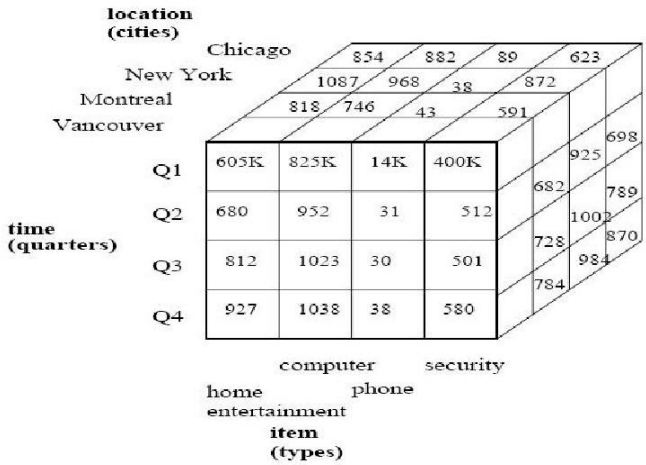
(d)

Primer: dvodimenziona (b) i trodimenziona (d) OLAP kocka za promet nekretnina. (a) i (c) su odgovarajuće tabelarne predstave.

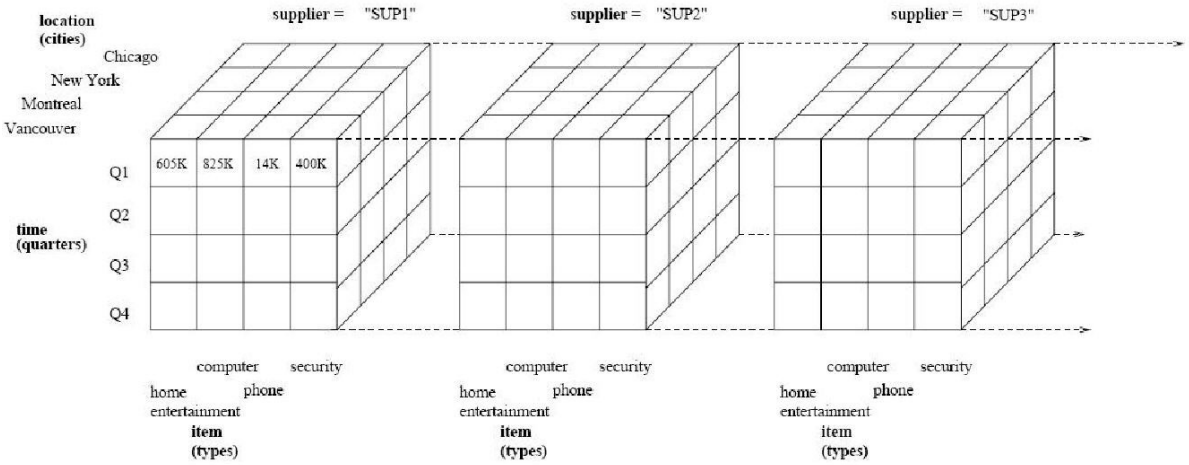
Primer: Prodaja artikala

	location = "Vancouver"				location = "Montreal"				location = "New York"				location = "Chicago"			
time	item				item				item				item			
	home ent.	comp.	phone	sec.	home ent.	comp.	phone	sec.	home ent.	comp.	phone	sec.	home ent.	comp.	phone	sec.
Q1	605K	825K	14K	400K	818K	746K	43K	591K	1087K	968K	38K	872K	854K	882K	89K	623K
Q2	680K	952K	31K	512K	894K	769K	52K	682K	1130K	1024K	41K	925K	943K	890K	64K	698K
Q3	812K	1023K	30K	501K	940K	795K	58K	728K	1034K	1048K	45K	1002K	1032K	924K	59K	789K
Q4	927K	1038K	38K	580K	978K	864K	59K	784K	1142K	1091K	54K	984K	1129K	992K	63K	870K

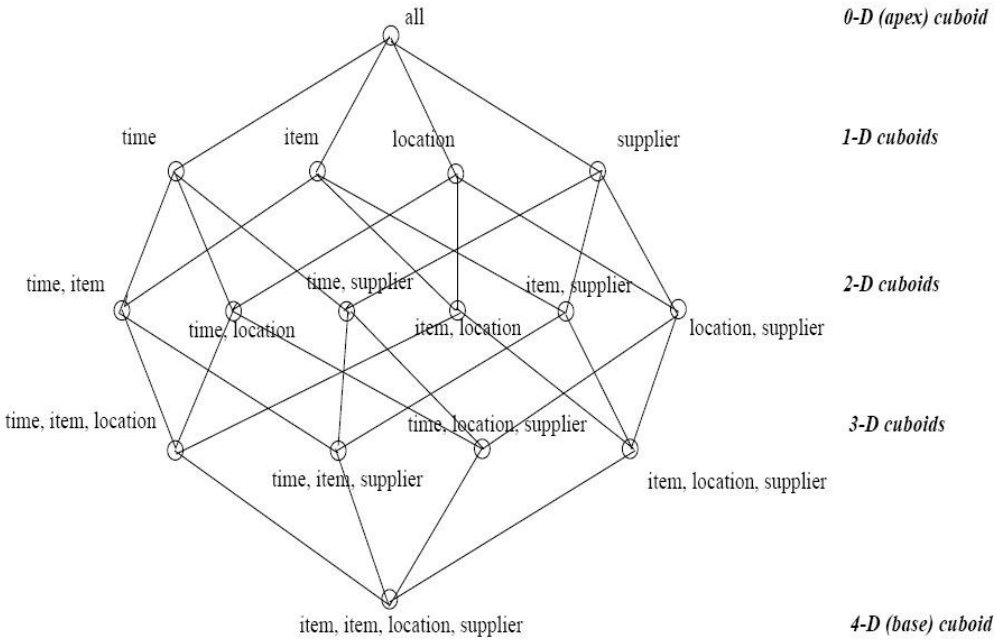
Tabelarni podaci za slučaj tri dimenzije (lokacija, vrsta artikla, vreme) i jedne mere (Iznos).



Trodimenziona OLAP kocka za prethodne tabelarne podatke.



Primer: Vizualizacija četvorodimenzione OLAP kocke preko trodimenzionih OLAP kocki od kojih svaka odgovara jednoj vrednosti četvrte dimenzije (isporučilac).



"Latisa kuboida" - sve moguće kocke za sve dimenzije (prethodni primer). U SQL jeziku običan GROUP BY generiše jedan kuboid, a GROUP BY CUBE celu latisu.

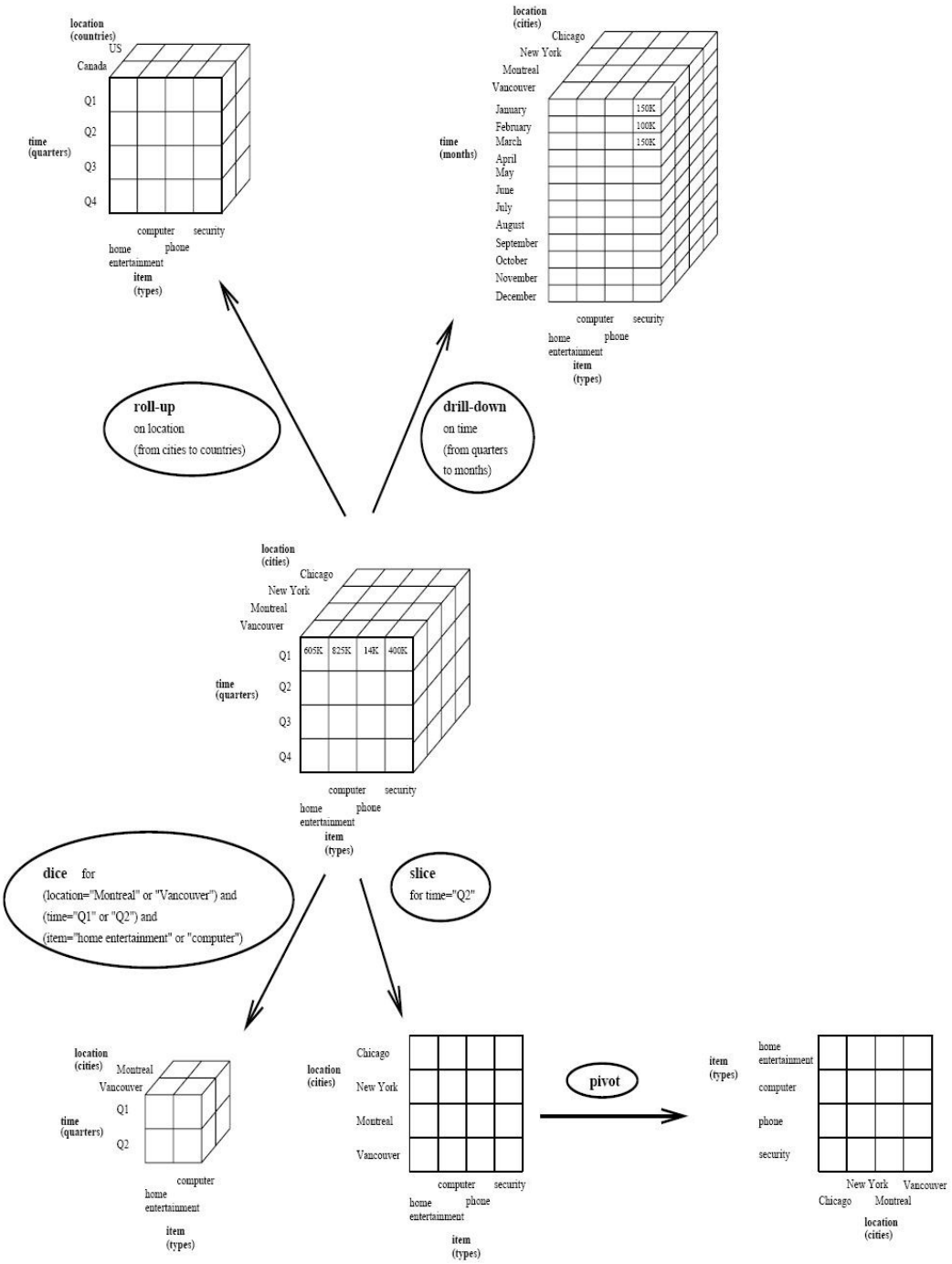
Operacije transformacije nad OLAP kockom (implementira ih OLAP Browser):

SLICE	Izdvajanje svodnih podataka za dati uslov po jednoj dimenziji. Rezultat je OLAP podkocka.
DICE	Izdvajanje svodnih podataka po datim uslovima dve ili više dimenzija. Rezultat je OLAP podkocka.
PIVOT (ROTATE)	Operacija vizualizacije koja obrće dimenzione ose radi alternativnog prikaza podataka.
ROLL UP (DRILL UP)	Svođenje OLAP kocke, bilo penjanjem po hijearhiji dimenzije bilo izostavljanjem jedne dimenzije.
DRILL DOWN	Detaljizacija OLAP kocke, bilo spuštanjem po hijerahiji dimenzije bilo uvođenjem jedne nove dimenzije. Obrnuto od ROLL UP.

U OLAP kocki se pri analizi uz izabrane dimenzije javlja samo jedna izabrana mera. Od dodatnih OLAP operacija najznačajnije su:

DRILL ACROSS	Operacija sa dve ili više tabela fakata, pri čemu između njih mora postojati mogućnost spajanja.
DRILL THROUGH	Detaljizacija OLAP kocke spuštanjem do nivoa izvornih tabela baze podataka nad kojima je definisana OLAP kocka.

Ilustracija OLAP operacija:



Definicija OLAP kocke

DMQL - Data Mining Querry Language. Ima i definicioni (DDL) i manipulativi (DML) deo. Osnove DDL dela (notacija sa zagradama):

```
DefincijaKocke ::=
   DEFINE CUBE Kocka [ Dimenzija,..]: Mera,..

DefinicijaDimenzije ::=
   DEFINE DIMENSION Dimenzija AS SpecifikacijaAtributa | { Dimenzija IN Kocka }

SpecifikacijaAtributa ::=
   ( { Atribut | { Atribut ( SpecifikacijaAtributa ) } | { AS IN Kocka },.. )

Mera ::=
   NazivMere = AgregatnaFunkcija ( KolonaTabeleFakata,.. )
```

Napomene:

Kod `DefinicijaDimenzije` druga opcija iza `AS` se koristi ako je u okviru šeme "Sazvežđe" dimenzija već definisana u okviru neke prethodno definisane kocke.

Podrazumeva se da za svaku kocku u skladištu podataka postoji tabela ili pogled fakata istog naziva kao i tabele dimenzija istih naziva.

Primer: Prethodna šema "Zvezda"

```
DEFINE CUBE Sales [time,item,branch,location] :
                  dollars_sold = SUM(sales_in_dollars),units_sold = COUNT(*)

DEFINE DIMENSION time AS (time_key,day,day_of_week,month,quarter,year)

DEFINE DIMENSION item AS (item_key,item_name,brand,type,supplier_type)

DEFINE DIMENSION branch AS (branch_key,branch_name,branch_type)

DEFINE DIMENSION location AS (location_key,street,city,province,country)
```

Primer: Prethodna šema "Pahuljica"

```
DEFINE CUBE Sales [time,item,branch,location] :
                  dollars_sold = SUM(sales_in_dollars),units_sold = COUNT(*)

DEFINE DIMENSION time AS (time_key,day,day_of_week,month,quarter,year)

DEFINE DIMENSION item AS (item_key,item_name,brand,type,
                          supplier(supplier_key,supplier_type))

DEFINE DIMENSION branch AS (branch_key,branch_name,branch_type)

DEFINE DIMENSION location AS (location_key,street,
                              city(city_key,province,country))
```

Primer: Prethodna šema "Sazvežđe"

```
DEFINE CUBE Sales [time,item,branch,location] :
                  dollars_sold = SUM(sales_in_dollars),units_sold = COUNT(*)

DEFINE DIMENSION time AS (time_key,day,day_of_week,month,quarter,year)

DEFINE DIMENSION item AS (item_key,item_name,brand,type,supplier_type)

DEFINE DIMENSION branch AS (branch_key,branch_name,branch_type)

DEFINE DIMENSION location AS (location_key,street,city,province,country)

DEFINE CUBE Shipping [time,item,shipper,from_location,to_location] :
                  dollars_cost = SUM(cost_in_dollars),units_shipped = COUNT(*)

DEFINE DIMENSION time AS IN CUBE Sales

DEFINE DIMENSION item AS IN CUBE Sales

DEFINE DIMENSION shipper AS (shipper_key,shipper_name,
                             location AS IN CUBE Sales,shipper_type)

DEFINE DIMENSION from_location AS IN CUBE Sales

DEFINE DIMENSION to_location AS IN CUBE Sales
```

NAPOMENE U VEZI PROJEKTOVANJA SKLADIŠTA PODATAKA

Koraci projektovanja skladišta podataka:

- ✔ Izbor poslovnih procesa koji se analiziraju. Na primer: porudžbine, prodaja, otprema, plaćanje.
- ✔ Utvrđivanje koje tabele fakata su potrebne.
- ✔ Za svaki izbrani poslovni proces, izbor granularnosti - nivoa detalja analize, odnosno "atomskog", najnižeg nivoa podataka u tabelama.
- ✔ Za svaku utvrđenu tabelu fakata, izbor dimenzija.
- ✔ Za svaku utvrđenu tabelu fakata, izbor mera.
- ✔ Za svaki dimenzioni model, izbor šeme.
- ✔ Sastavljanje definicija za odgovarajuće OLAP kocke (DMQL).

Po vrsti mere postoje dve vrste tabele fakata, odnosno :

- ✔ Tabele fakata prometa. Na primer, uplate, isplate, broj izdatih knjiga, itd.
- ✔ Tabele fakata stanja. Na primer, stanje računa, količina robe na zalihi, itd.