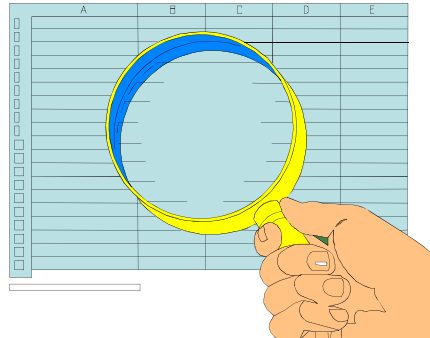


Baze podataka

Osnovni pojmovi



- Različite vrste softvera (aplikacija) imaju analogije sa realnim svetom
 - Windows, Word, Excel, ...
- Baze podataka – potpuno veštačka tvorevina
- Baze podataka nisu tabele
 - Tabele postoje u realnom svetu (telefonski imenik, rečnik i sl.), ali ne i relacione baze podataka
- Baze podataka - Okruženje koje je istovremeno:
 - pogodno i
 - efikasno za upotrebu

Šta su baze podataka?

- Najčešće se pod ovim podrazumeva **set podataka smeštenih u računaru**
- BP postoje u različitim formama:
 - Elektronski podaci: tekst, video i audio zapisi, brojevi
 - Knjige: kolekcija tekstova i slika, biblioteke
 - Datoteke organizovane u direktorijume
 - Uzorci sa naftnih bušotina
 - Medicinski uzorci
 - DNK uzorci u forenzičkim laboratorijama
 - Demografski podaci
 - Klijenti i njihovi računi u bankama
 - Podaci o turističkim kapacitetima
 - Podaci o studentima, predmetima, ocenama
 -



Šta su baze podataka?

- Baze podataka su povezane sa svim aspektima svakodnevnog života
- Aplikacije za rad sa bazama podataka:
 - Bankarstvo: transakcije, ...
 - Avio-prevoz: rezervacije, red letenja, ...
 - Fakulteti: upis, ocene, ...
 - Prodaja: kupci, prodavci, naručivanje
 - Upravljanje ljudskim resursima: podaci zaposlenih, kvalifikacije
 - Proizvodnja: roba, zalihe, snabdevanje, ...
 - E-trgovina: naručivanje, reklama, prilagodjavanje kupcu, ...
 -

Primeri aplikacija za rad sa BP

- Informacioni sistem fakulteta:
 - Dodavanje novog studenta (upis), instruktora (profesora na predmetu), kursa, ...
 - Registracija studenata za neki od kurseva, generisanje spiska studenata
 - Dodela zvanja studentima, izračunavanje proseka, uverenje o položenim predmetima sa ECTS poenima itd.
 - Prijava ispita
- Klasični sistemi
 - Nekada su se aplikacije baza podataka realizovale direktno u programskim jezicima i uz pomoć datoteka

Slabosti klasičnih sistema

- Redudansa i nekonzistentnost podataka
 - Različiti tipovi datoteka, dupliranje podataka u različitim datotekama
- Problemi sa pristupom podacima
 - Potreban je novi program za pristup podacima
- Izolacija podataka – više datoteka i formata
- Problem integriteta
 - Komplikovano je postavljati ograničenja
 - Teško je dodavati novo ograničenje i menjati postojeće

Slabosti klasičnih sistema - nastavak

- Atomnost pri promenama (*update*)
 - Greške mogu da ostave bazu podataka u nekonzistentnom stanju
- Konkurentan pristup za više korisnika
 - Konkurentnost je potrebna zbog performansi
 - Nekontrolisan konkurentan pristup može da dovede do nekonzistentnosti baze
- Problemi sa zaštitom podataka
 - Teško je obezbediti pristup korisnicima samo nekim podacima u bazi

Računarske baze podataka

- Danas se podaci čuvaju u digitalnom obliku
- Baze se sastoje od:
 - Bita, organizovanih u bajtove, koji mogu da čine stringove itd.
 - Polja (*fields*) – niz bajtova koji reprezentuje informaciju (broj, tekst, slika, audio/video zapis, ...)
 - Zapisa (*records*) - niz polja koji zajedno opisuju jedan “entitet”
 - Datoteka (*files*) – niz zapisa koji opisuju različite entitete
- Baze podataka tipično sadrže više relevantnih datoteka

- Baza podataka
 - organizovani skup logički povezanih podataka
 - integrisani skup podataka o nekom sistemu i skup postupaka za njihovo održavanje i korišćenje, organizovan prema potrebama korisnika
 - dobro strukturirana kolekcija podataka, koja postoji jedno određeno vreme, koja se održava i koju koristi više korisnika ili programa

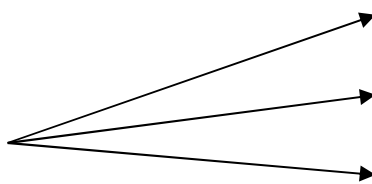
Podaci

- Podatak
 - Činjenica o nekom predmetu i/ili događaju koja se može zabeležiti i sačuvati na računaru
 - Strukturirani podaci: brojevi, karakteri, ...
 - Nestruktuirani podaci: slika, zvuk, video, ...
 - Podatak sam po sebi nema značenje, tek kada se interpretira nekom vrstom sistema za obradu podataka poprima značenje i postaje informacija.

- **Informacija**

- Podatak koji je obrađen na takav način da se znanje osobe koja koristi podatak povećalo

Podaci koji nisu korisni



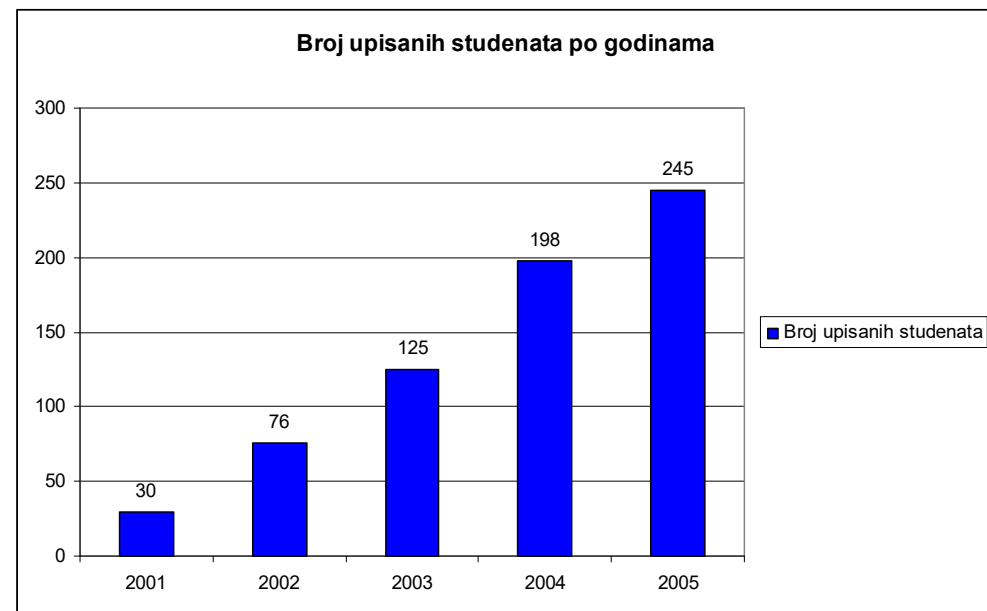
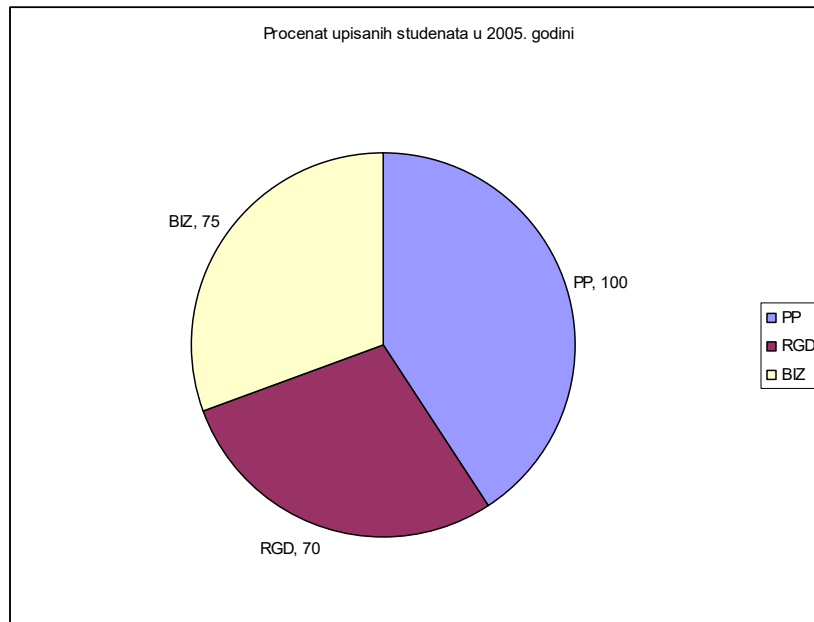
Petar Petrović	1506983710325
Marko Marković	0211979850123
Janko Janković	1112985830456
-----	-----

Ime i prezime	JMBG	Smer	Godina upisa
Petar Petrović	1506983710325	PFB	2012
Marko Marković	0211979850123	MT	2009
Janko Janković	1112985830456	IR	2011
-----	-----	ER	2012

Informacija o upisu – dopunjeni relevantni podaci

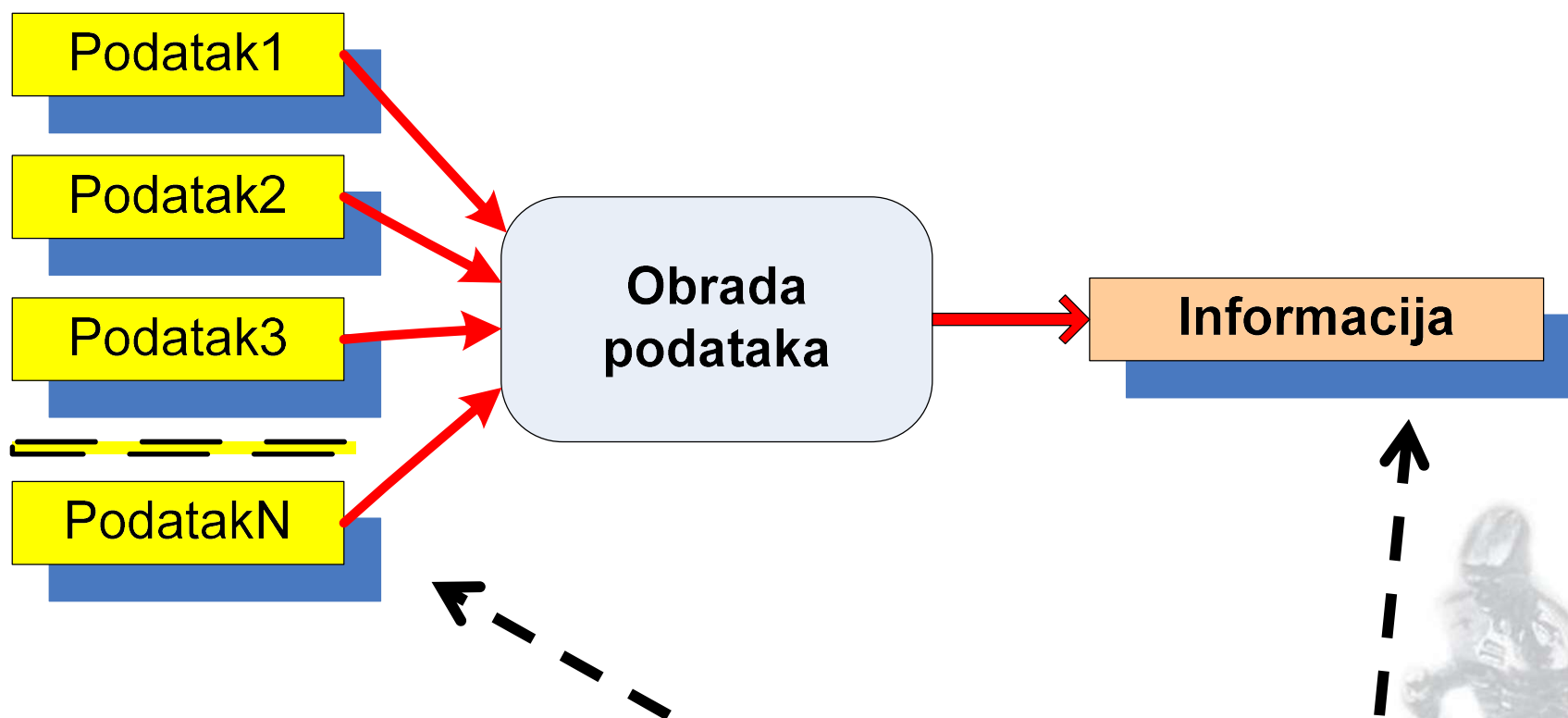
- **Informacije**

- podaci koji se sumiraju ili na neki drugi način obrađuju i prezentuju



Grafički prikaz podataka iz BP - informacija o upisu

Informacija

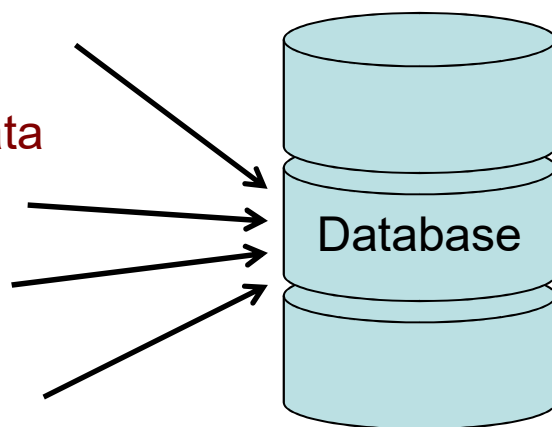


Važno je razumeti šta je smešteno (*stored*) u BP, a šta se može dobiti (*retrieved*) iz BP

Podaci/Informacije

PODACI

Ocene studenata
iz pojedinih
predmeta



INFORMACIJE

Prosečna ocena studenta

Prosečna ocena na
godini studija

Prosečna ocena na fakultetu

Spisak studenata koji neki
ispit polažu 7 i više puta

Metapodaci

- Metapodaci
 - Rečnik podataka
 - Podaci o podacima
 - Opisuju svojstva ili karakteristike podataka krajnjih korisnika i kontekst tih podataka
 - Tipična svojstva podataka su naziv (ime) podatka, definicija, dužina (veličina), i dozvoljene vrednosti (ograničenja)
 - Metapodaci omogućavaju dizajnerima i korisnicima baza podataka da razumeju koji podaci postoje u bazi i šta oni znače

Metapodaci

Field	Type	Null	Key	Default	Extra
Sifra	int(11)	NO	PRI	NULL	auto_increment
Naziv	varchar(45)	YES		NULL	
Cena	float	NO		NULL	
JedinicaMere	varchar(10)	NO		KG	
ZemljaPorekla	varchar(45)	YES		NULL	
Kategorija_Rb	int(11)	NO	MUL	NULL	



Registar korisnika

- Registar korisnika
 - Podaci o korisnicima i njihovim pravima pristupa
 - Ko može da pristupa
 - *Username, password*
 - Kojim podacima
 - Baza podataka, tabele, kolone, vrste (redovi)
 - Šta može da radi sa njima
 - INSERT, DELETE, UPDATE, SELECT



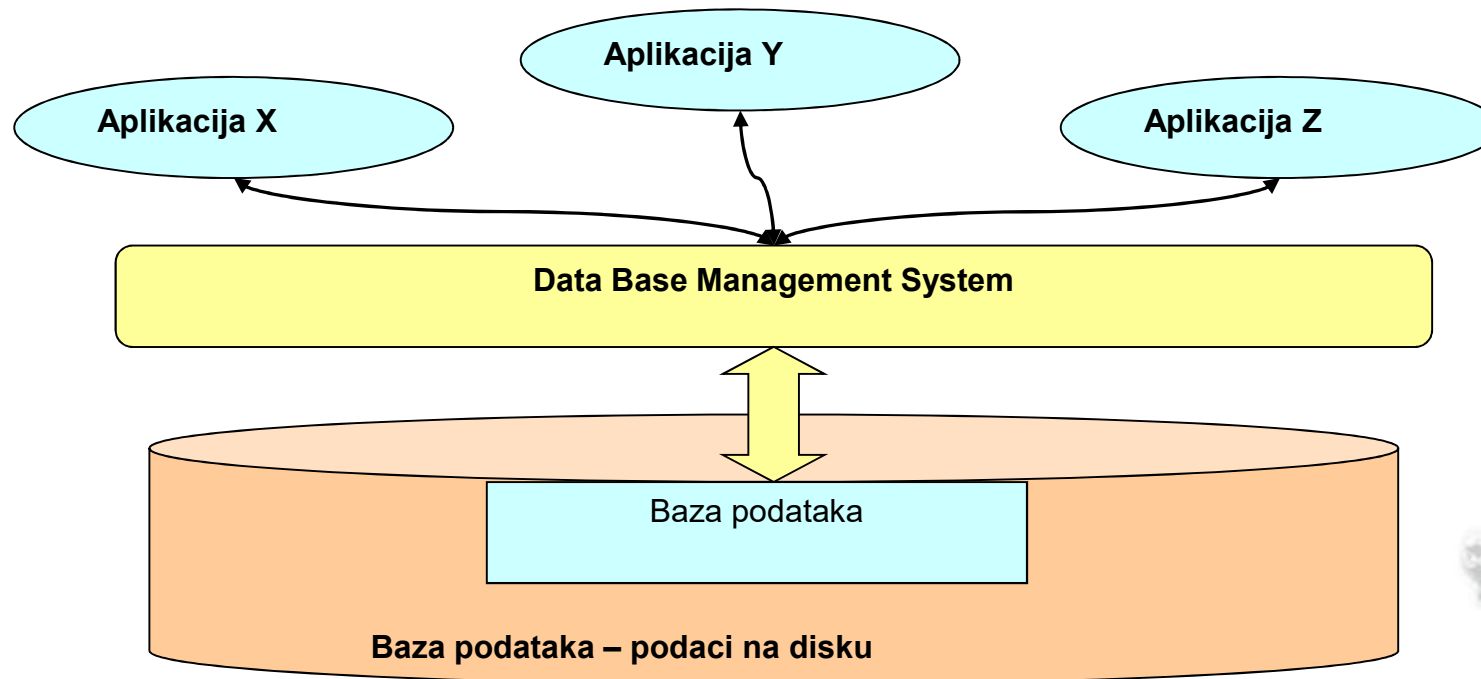
DBMS (SUBP)

DBMS - *Data Base Management System*

- Sistem za upravljanje bazama podataka
 - Računarski program
- Softverski sistem koji se koristi za:
 - kreiranje,
 - održavanje i manipulisanje podacima,
 - kontrolu prava pristupa bazi podataka



- DBMS - Interfejs između korisnika (korisničkih programa, aplikacija) i zapisa baze podataka na disku



- Od posebnog interesa su:
 - Relacione baze podataka - RBP
 - Programi za upravljanje sa RBP – Sistemi za upravljanje bazama podataka (DBMS)
Relational Database Management System (RDBMS)
- Reč “relaciona” danas se često izostavlja, zato što većina poznatih SUBP (kao što su: MS Access, MySQL, MS SQL Server, Oracle, IBM DB2, Ingres, PostgreSQL, Informix, Sybase) jesu relacione BP

- Karakteristike relacionih BP
 - Podaci su organizovani u redove i kolone, a skup takvih podataka je relacija
 - Sve vrednosti su skalarne – u jednom polju se može naći samo jedna vrednost
 - Ne mogu da postoje dupli zapisi (redovi u tabeli)
 - Sve operacije se obavljaju nad relacijama a rezultat operacija je opet relacija



- **Koncept RBP:**
 - Na slici su prikazane 3 tabele sa ukupno 10 atributa (različitih kolona)
 - Tab1 i Tab2 “dele” atribut 3
 - Tab2 i Tab3 “dele” atribut 7

Tab1

Atrib. 1	Atrib. 2	Atrib. 3	Atrib. 4	Atrib. 5

Tab2

Atrib. 3	Atrib. 6	Atrib. 7

Tab3

Atrib. 8	Atrib. 7	Atrib. 9	Atrib. 10



- **Koncept RBP:**
 - Na slici su prikazane 3 tabele sa ukupno 10 atributa (različitih kolona)
 - Tab1 i Tab2 “dele” atribut **BrInd**
 - Tab2 i Tab3 “dele” atribut **Šif.Predmeta**

Student

Ime	Prezime	BrInd	Mesto	Telefon
Marko	Marković	123456/2009	Beograd	063-123123
Petar	Petrović	222333/2009	Valjevo	065-232323
Janko	Janković	111222/2009	Niš	062-121212

Ispit

BrInd	Ocena	Šif.Predmeta	Datum
123456/2009	7	BP03	21.01.2010.
123456/2009	8	AM03	27.01.2010.
222333/2009	10	BP03	04.02.2010.

Predmet

Naziv	Šif.Predmeta	Oblast	Šif.Oblasti
Baze podataka	BP03	Informatika	INF
Aktuarska mat.	AM03	Finansije	FIN
Upravljanje fin.	UF03	Finansije	FIN



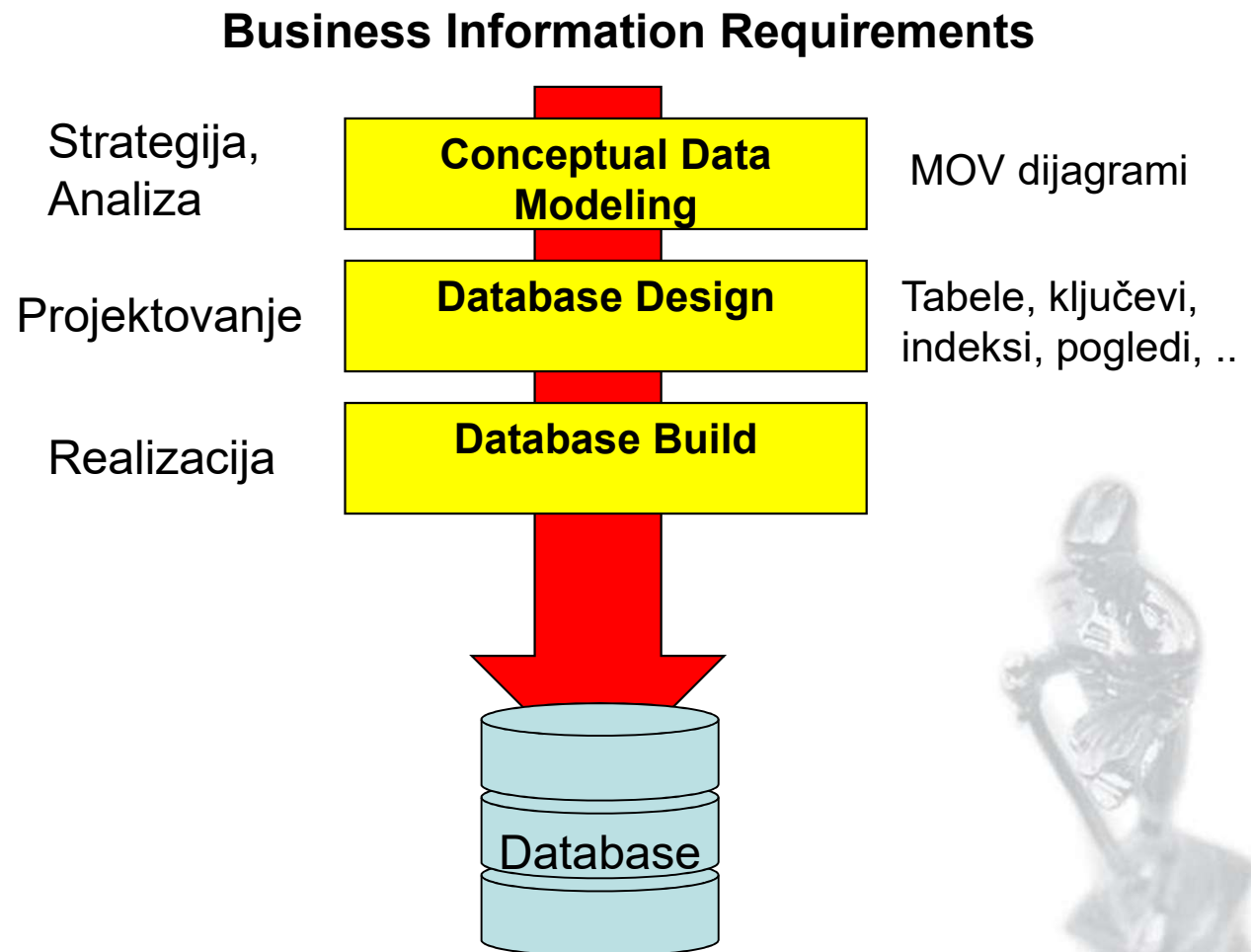
Korisnici baze podataka:

- Krajnji korisnici (*End users*)
 - Koriste bazu podataka za ostvarenje nekog cilja
- Programeri aplikacija (*Application developers*)
 - Programiraju softver koji krajnjim korisnicima omogućavaju pristup bazi podataka
- Administratori (*Database administrator*)
- Sistemski programeri (*Database systems programmer*)



Modelovanje - Razvoj BP-

- Modelovanje podataka je prvi korak u razvoju baza podataka (*database development process*)



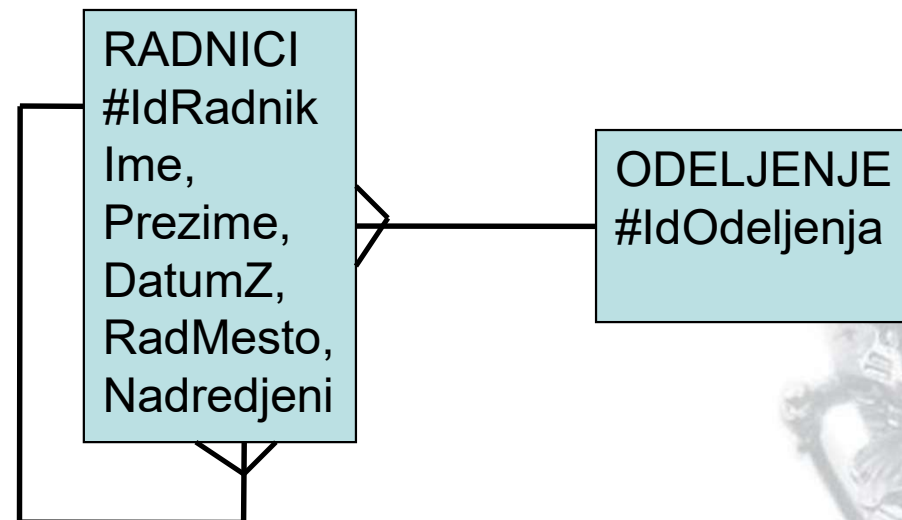
Modelovanje - Razvoj BP-

- Razvoj baza podataka počinje sa analizom zahteva krajnjih korisnika
- Primer - Problem upravljanja kadrovima (zaposleni):
 - Potrebno je voditi evidenciju za svakog zaposlenog (ime, prezime, radno mesto, datum zapošljavanja, zarada)
 - Svaki zaposleni ima jedinstven identifikacioni broj
 - Preduzeće ima više odeljenja (proizvodnja, računovodstvo, prodaja itd). Odeljenja imaju jedinstven ID
 - Svaki zaposleni radi u jednom odeljenju
 - Neki od zaposlenih su direktori odeljenja
 - Potrebno je da se zna direktor za svakog zaposlenog i svi zaposleni kojima upravlja određeni direktor

Modelovanje - Razvoj BP-

- ER model pokazuje odnose u datoj organizaciji i podržava poslovne funkcije

Primer ER modelovanja



Modelovanje - Razvoj BP-

- *Database design*: ER model se transformiše u Relacioni model:
 - Nazivi tabela
 - Nazivi kolona u tabelama
 - Primarni (PK) i spoljašnji ključevi (FK)
 - *Nulls*: indikacija da kolona mora da ima vrednost
 - *Unique*: vrednost u datoj koloni ne sme da se ponavlja
 - *Datatype*: format i definicija svake kolone
- Npr. tabela Radnik

ID radnika	Ime	Prezime	Radno mesto	Datum zaposlenja	Nadređeni	IdOdeljenja
------------	-----	---------	-------------	------------------	-----------	-------------

Modelovanje - Razvoj BP-

- Realizacija (*database build*)
 - Primena SQL (*Structured Query Language*) jezika za fizičku realizaciju

```
SQL>CREATE TABLE ODELJENJE  
      (IdOdeljenje NUMBER(5) NOT NULL PRIMARY KEY,  
       Naziv VARCHAR(25) NOT NULL);
```

```
SQL>CREATE TABLE RADNICI  
      (IdRadnik NUMBER(9) NOT NULL PRIMARY KEY,  
       Ime VARCHAR2(15) NOT NULL,  
       Prezime VARCHAR2(20) NOT NULL,  
       RadnoM VARCHAR2(15),  
       Zaposlen DATE NOT NULL,  
       Nadredjeni NUMBER(9) REFERENCES RADNICI  
       Odeljenje NUMBER(5) REFERENCES ODELJENJE);
```

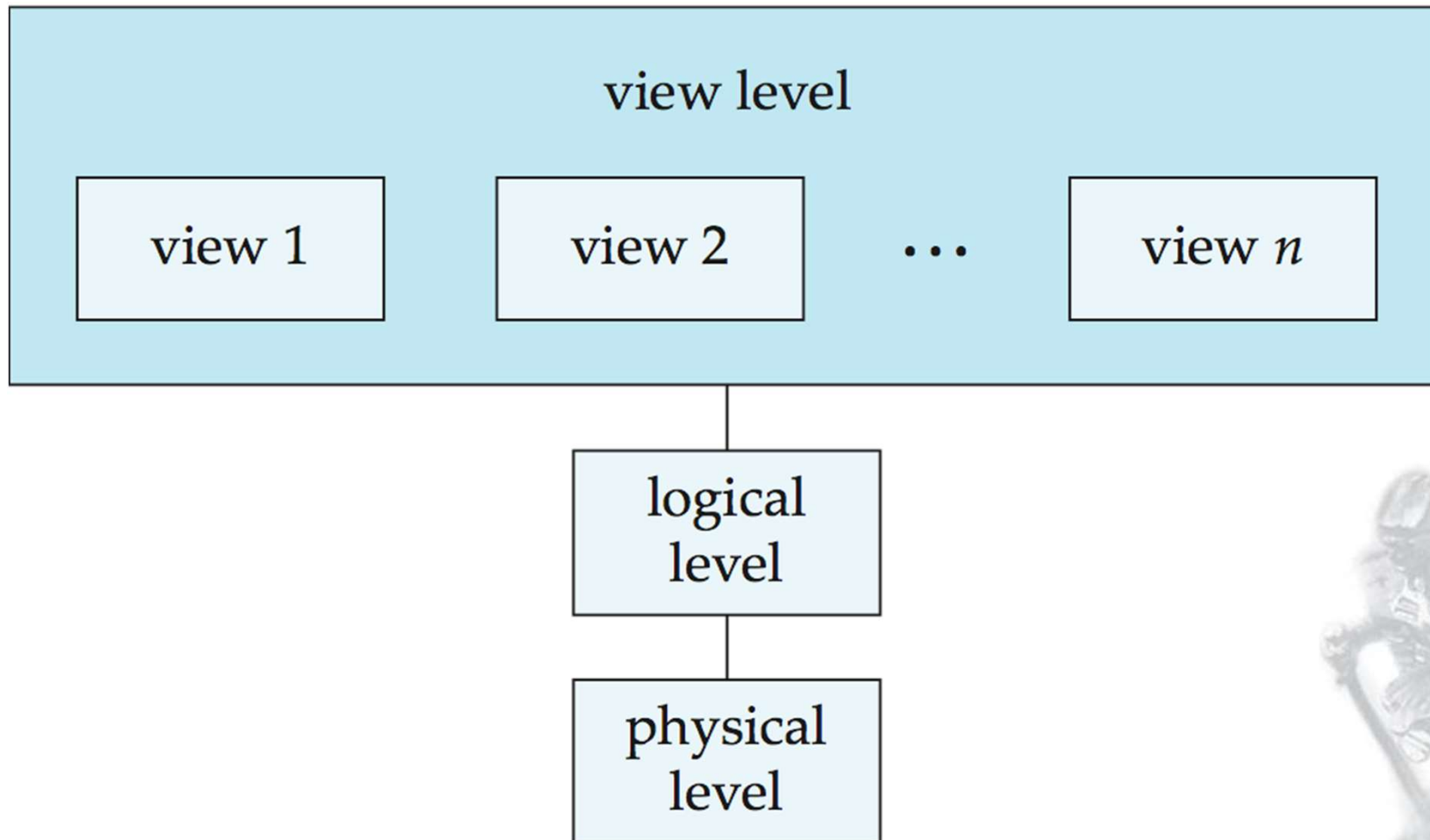


Standardizacija

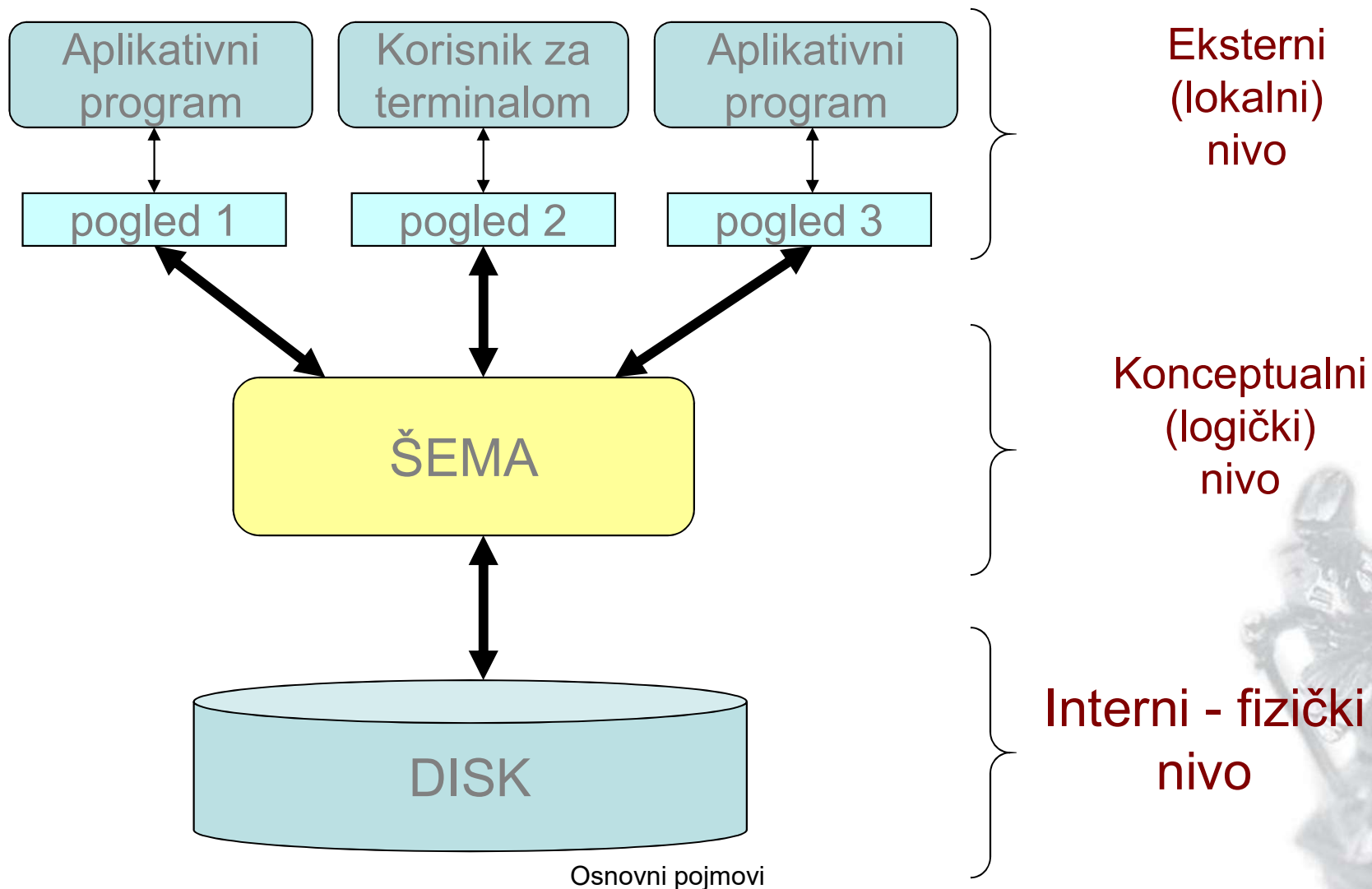
- Slojevi (*layers*) idealne BP
 - Slojevit rad – uvođenje standarda
 - Podela na 3 nivoa
 - Eksterni (Korisnik) View[1], ..., View[n]
 - Konceptualni (logički) Opis preduzeća
 - Interni (fizički) Datoteke, metod pristupa, indeksi
 - Različiti slojevi imaju različite zadatke
 - Nezavisnost slojeva



Višeslojna arhitektura BP



Višeslojna arhitektura BP



- Eksterni sloj
 - Odnosi se na korisnikov pogled na BP
 - Svaki pogled (podšema) liči na mali konceptualni sloj
 - Vremenom se menja
 - Promena pogleda,
 - Brisanje pogleda,
 - Dodavanje novog itd.



- **Konceptualni sloj**
 - Fundamentalan jer opisuje kompletne informacije i njihovu strukturu
 - Koristi se:
 - *Data Definition Language* (DDL)
 - *Data Manipulation Language* (DML)
 - Tabele (relacije) su primer upotrebe DDL instrukcija
 - Opis šema i njihova struktura
 - Ovaj sloj se može menjati kako se vrše promene npr. u preduzeću

- Fizički sloj

- Odnosi se na organizaciju datoteka i njihov fizički zapis na disku
- Vremenom se menja:
 - Uvode se novi memorijski uređaji za zapis podataka
 - Koriste se novi indeksi za pristup podacima (npr. za često korišćene upite)
 - Različita distribucija podataka na memorijskim modulima
 - Baferi za često korišćene podatke

- Postojanje relacije potpuno je nezavisno od fizičkog oblika podataka !!!
 - Relacija može da bude fizička tabela na disku
 - Relacija može da bude i izvedena iz više table, pojedine vrednosti mogu biti dodati (izračunati) itd.
- Nezavisnost slojeva – omogućava jednostavniji razvoj softvera, promenu platforme itd.

Primer RBP

1.	Ime	Id#	DatumRođenja	Položaj	Zarada
	A	121	12.2.1965.	2	80
	A	132	17.6.1973.	3	70
	B	101	28.2.1987.	4	70
	C	106	10.5.1991.	2	80

2.	Id#	Knjiga	Datum	Adresa
	132	xx	13.7.13.	ddd
	121	yy	26.8.14.	bbb

3.	Id#	Bolest	Datum
	101	hhhh	20.1.15.
	121	oooo	16.12.14.

4.	Id#	Adresa	DatumRođenja
	101	aaa	28.2.1987.
	121	bbb	12.2.1965.
	106	ccc	10.5.1991.
	132	ddd	17.6.1973.



Tipični zahtevi

- Zahtev \Rightarrow upit (*query*)
- Primer:
 - Prikazati spisak zaposlenih rođenih pre 01.01.1980.
 - Prikazati spisak zaposlenih koji su uzeli knjige pre 01.08.2013. godine
 - Poslati poruku zaposlenima koji imaju bolest 'oooo' da izvrše lekarski pregled
- Neki upiti se odnose na jednu tabelu, a neki na više
- Neophodno je da postoji jezik (što bliži čovekovom), da bi se izrazili prethodni upiti (jezik za rad sa BP)
 - SQL – *Structured query language*



Projektovanje BP

- Neophodnost dobrog projektovanja BP:
 - Važno je pažljivo razmišljati o tabelama i njihovoj strukturi
 - Poželjno je da se podaci pojavljuju samo jedanput
 - U datom primeru se **DatumRođenja** pojavljuje dva puta
 - Opasnost kod višestrukog pojavljivanja **Adrese** zaposlenih
 - Ako se zaposleni preseli, promena se mora evidentirati na više mesta
 - Izuzetno, zbog efikasnosti – brzog pristupa podacima, nekada je potrebno imati redudansu

Ograničenja

- Neka je politika preduzeća takva da se zarade određuju položajem zaposlenog
 - U datom primeru $POLOŽAJ=2$ implicira $ZARADA=80$, i dva puta je zapisano
 - Ovo je drugi tip redundanse, manje očigledan, ali koji znatno više smeta (zavisnost)
- Sledeći problem
 - Nemoguće je uneti zaradu, osim one koja već postoji za zaposlene
 - Npr. Nemoguće je upisati da $POLOŽAJ=2$ implicira $ZARADA=90$ (upropastili bi logiku za druge zaposlene)

Rešavanje problema

- | Ime | Id# | DatumRođenja | Položaj | Zarada |
|-----|-----|--------------|---------|--------|
| A | 121 | 12.2.1965. | 2 | 80 |
| A | 132 | 17.6.1973. | 3 | 70 |
| B | 101 | 28.2.1987. | 4 | 70 |
| C | 106 | 10.5.1991. | 2 | 80 |

Zameniti sa sledeće dve tabela

- | Ime | Id# | DatumRođenja | Položaj |
|-----|-----|--------------|---------|
| A | 121 | 12.2.1965. | 2 |
| A | 132 | 17.6.1973. | 3 |
| B | 101 | 28.2.1987. | 4 |
| C | 106 | 10.5.1991. | 2 |

- | Položaj | Zarada |
|---------|--------|
| 1 | 90 |
| 2 | 80 |
| 3 | 70 |
| 4 | 70 |



Rešavanje problema

- Različiti korisnici imaju potrebu za različitim podacima
 - Sa stanovišta prava/sigurnosti, različiti korisnici mogu imati različite privilegije
 - Odeljenje za plate može da vidi ZARADE, a ne i BOLESTI
 - Zdravstveno odeljenje može da vidi BOLESTI, a ne i ZARADE
 - Različiti korisnici imaju potrebu za različitim aspektima informacije
 - Zdravstveno odeljenje možda ima potrebu da vidi **GodineStarosti** umesto **DatumRođenja**
 - Podaci se mogu organizovati prema potrebama korisnika

Pogledi - Views

- Moguće rešenje: dati svakom korisniku privilegiju da može da vidi samo pogled na BP
- Pogled: izvedeni podaci iz BP
- Npr. Zdravstveno odeljenje može imati privid da postoji sledeća tabela:
 - | Id# | Ime | Adresa | Starost | Bolest |
|-----|-----|--------|---------|--------|
| 101 | B | aaa | 28 | hhhh |
| 121 | A | bbb | 42 | oooo |
- Ovo je jedan od načina ograničavanja prava korisnika
- Idealno: korisnik manipuliše (radi) samo nad pogledima

Fizički dizajn

- BP treba da bude organizovana tako da se upiti efikasno izvršavaju
 - Datoteke treba da budu organizovane na odgovarajući način
- Npr. Ako je često potreban **Položaj** zaposlenih po identifikacionim brojevima **Id#**, potrebno je imati datoteku sortiranu po **Id#**.
- Npr. Ako je potrebno štampati zarade zaposlenih rođenih 1977. godine, bolje je da je datoteka sortirana po **GodineRođenja**
- Fizički dizajn – bitan za optimizaciju upita



Oporavak BP

- BP mora da bude otporna na eventualne greške (npr. nestanak napajanja)
- Npr. Pretpostavimo da postoji sledeća tabela sa stanjem na određenom štednom računu

Id#	Stanje	Račun
101	40	123-1
106	40	111-2
121	0	222-1
132	10	321-6

- Svakom zaposlenom treba povećati stanje na računu za 10
 - Da bi se to uradilo transakcija (izvršenje programa) sekvencijalno (u FOR petlji) ažurira stanje svakog računa

Oporavak BP

- Datoteka se nalazi na disku. Podaci se učitavaju u RAM memoriju, obrađuju i ponovo zapisuju u istu datoteku:

- `i=100;`

- `X:=Stanje[i+1];`

- `X:=X+10;`

- `Stanje[i+1]:=X;`

- `.`

- `.`

- `.`

- Šta ako u toku obrade nestane el. napajanja?
- Ne znamo koja stanja su promenjena.
- Različite tehnike za upravljanje izvršenjem transakcija

Čitanje (*Read*)

Obrada (*Processing*)

Upisivanje (*Write*)



Konkurentan rad

- Paralelno se izvršavaju dve transakcije
- Transakcija 1 podiže, a transakcija 2 ulaže novac (10 dinara) na isti račun Id#=101

T1	T2
X:=Stanje[101];	-----
X:=X-10;	-----
-----	X:=Stanje[101];
-----	X:=X+10;
Stanje[101]:=X;	-----
-----	Stanje[101]:=X;
-----	-----
·	·
·	·

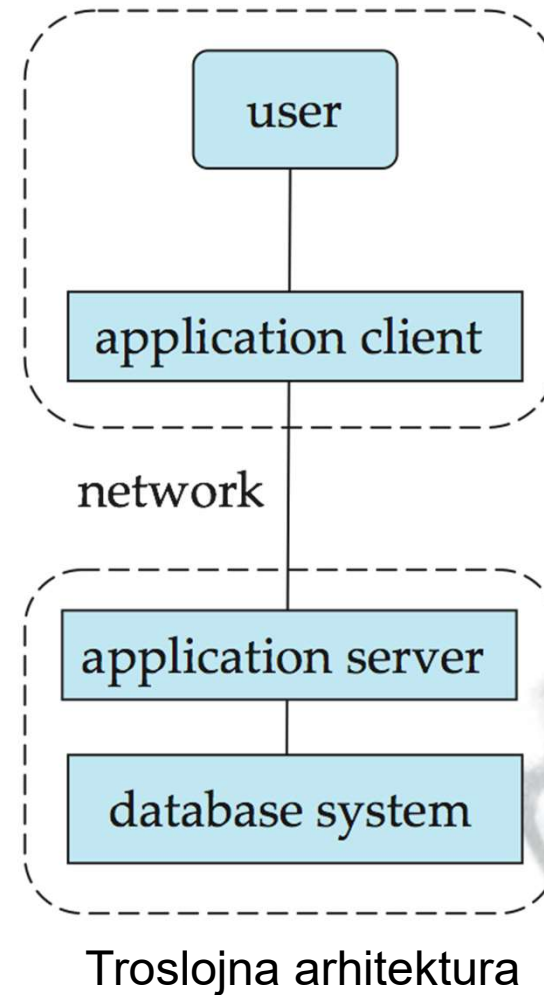
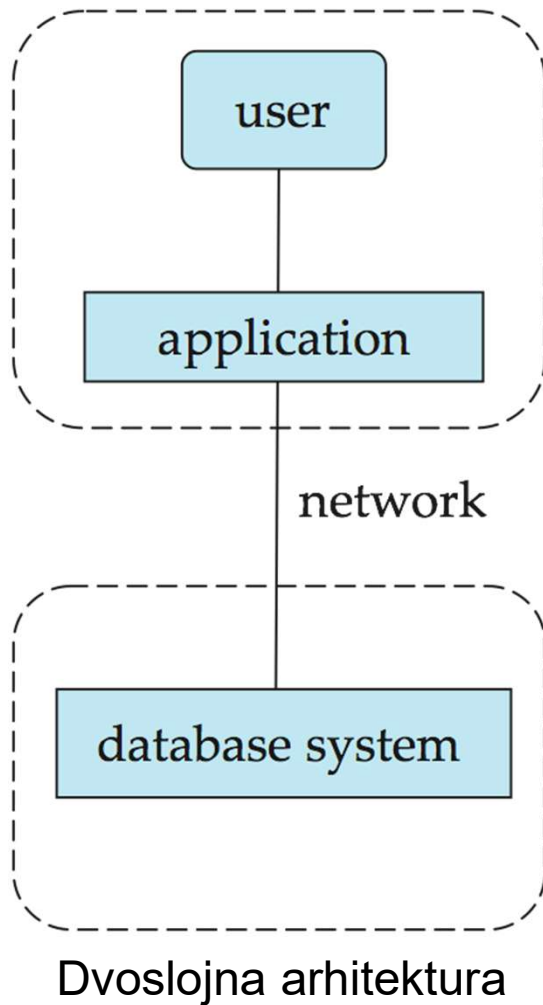
- Ažuriranje koje je izvršila transakcija T1, prebriše se ažuriranjem koje izvrši transakcija T2



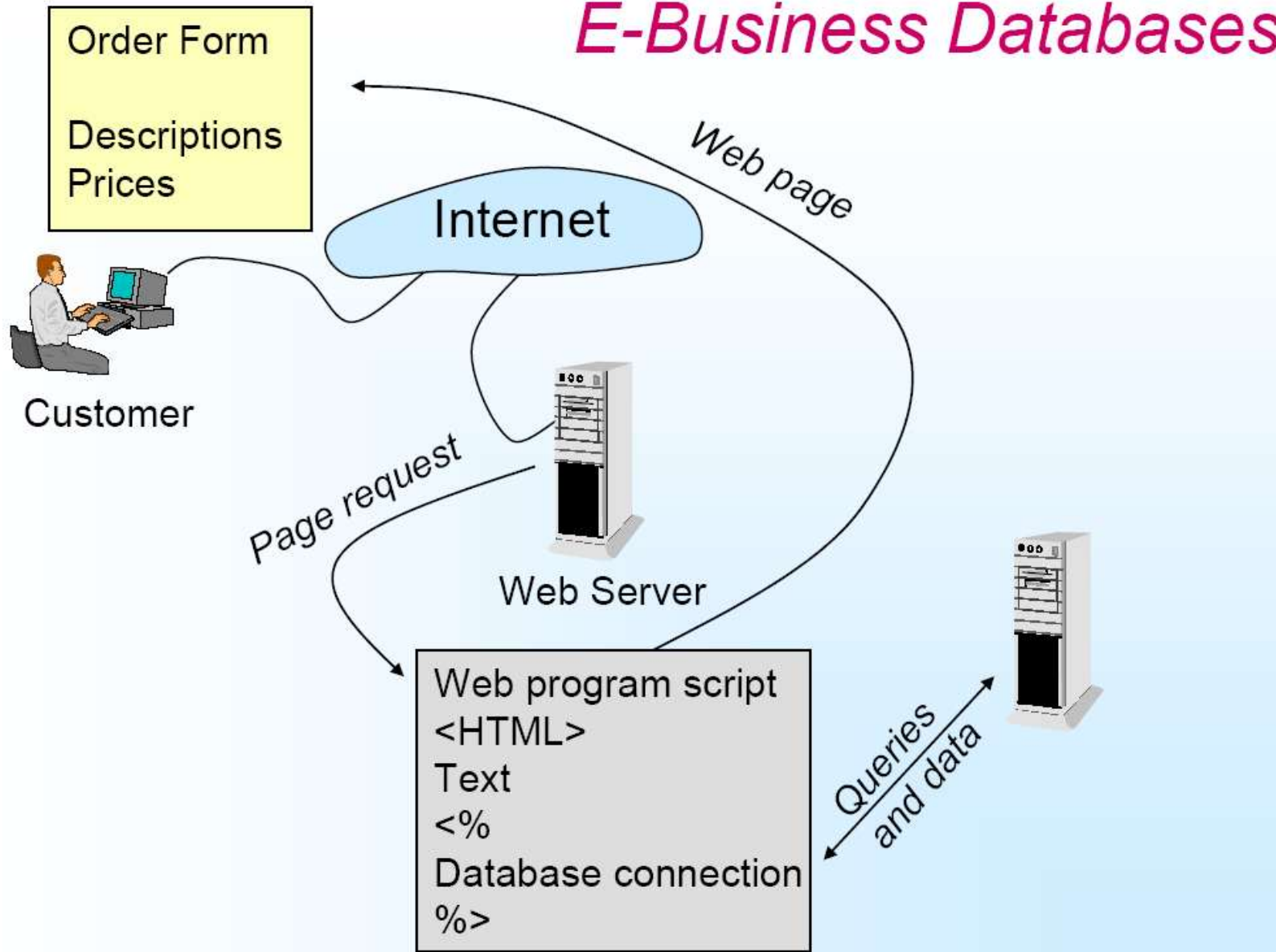
Konkurentan rad

- Konkurentni rad:
 - Standardni operativni sistemi ne pomažu kod prethodnih problema, ali se to rešava mehanizmima za konkurentno izvršavanje transakcija
 - Npr. Podizanje novca sa jednog računa i uplaćivanje na drugi račun iste sume novca – Konzistentnost pre i posle promena
 - Medjutim, ako se stanje na nekom računu povećava 2 puta, konzistentnost baze ne pomaže

Aplikacije i BP



E-Business Databases



Važno!

Ova prezentacija je nekomercijalna.

Slajdovi mogu da sadrže materijale preuzete sa Interneta, stručne i naučne građe, koji su zaštićeni Zakonom o autorskim i srodnim pravima. Ova prezentacija se može koristiti samo privremeno tokom usmenog izlaganja nastavnika u cilju informisanja i upućivanja studenata na dalji stručni, istraživački i naučni rad i u druge svrhe se ne sme koristiti –

Član 44 - Dozvoljeno je bez dozvole autora i bez plaćanja autorske naknade za nekomercijalne svrhe nastave:

(1) javno izvođenje ili predstavljanje objavljenih dela u obliku neposrednog poučavanja na nastavi;

- ZAKON O AUTORSKOM I SRODNIM PRAVIMA ("Sl. glasnik RS", br. 104/2009 i 99/2011)

.....
Mladen Veinović

mveinovic@singidunum.ac.rs



Pitanja ?

