

*Baze podataka*

*Katedra za računarstvo  
Elektronski fakultet u Nišu*

# Uvod u Baze podataka

Prof.dr Leonid Stoimenov

# Uvod u baze podataka - Pregled

---

- ▶ Osnovni pojmovi
  - ▶ Šta je **Podatak, informacija**?
  - ▶ Šta je **Baza podataka**?
  - ▶ Šta je **Sistem za upravljanje bazama podataka**?
  - ▶ Šta je **Aplikacija nad bazom podataka**?
  - ▶ Šta je **Sistem baza podataka**?
- ▶ Konvencionalna obrada i obrada zasnovana na bazama podataka
- ▶ Sistem baza podataka
- ▶ Kategorizacija korisnika baza podataka
- ▶ Istorijat baza podataka
- ▶ Prednosti i nedostaci

# Podatak i informacija

---

- ▶ **Podatak** (lat. *datum* – deo informacije, eng. *Data*)
  - ▶ Jednostavna neobrađena izolovana činjenica koja ima neko značenje i koja se obrađuje i čuva na računaru
  - ▶ Podaci su znakovni prikaz činjenica i pojmova koji opisuju svojstva objekata
- ▶ **Obrada podataka** – proces prevođenja podataka u informacije
- ▶ **Informacija** (lat. *Informare*, eng. *Information*) - rezultat analize i organizacije podataka na način da daje/generiše novo znanje
- ▶ **Znanje** ...
  - ▶ (kurs Veštačka inteligencija, VII semestar)
- ▶ **Mudrost** ...

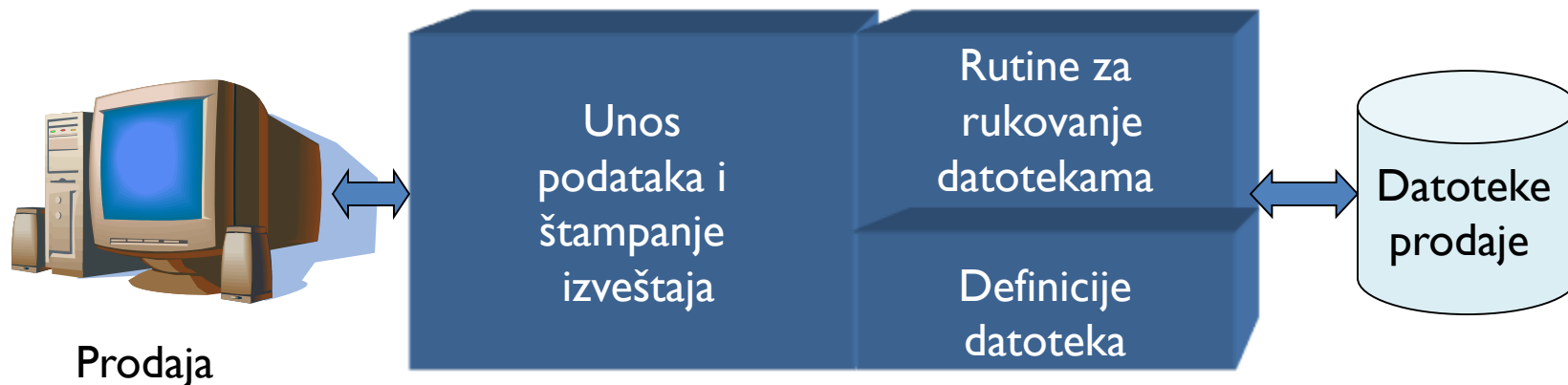
# Konvencionalni sistemi

---

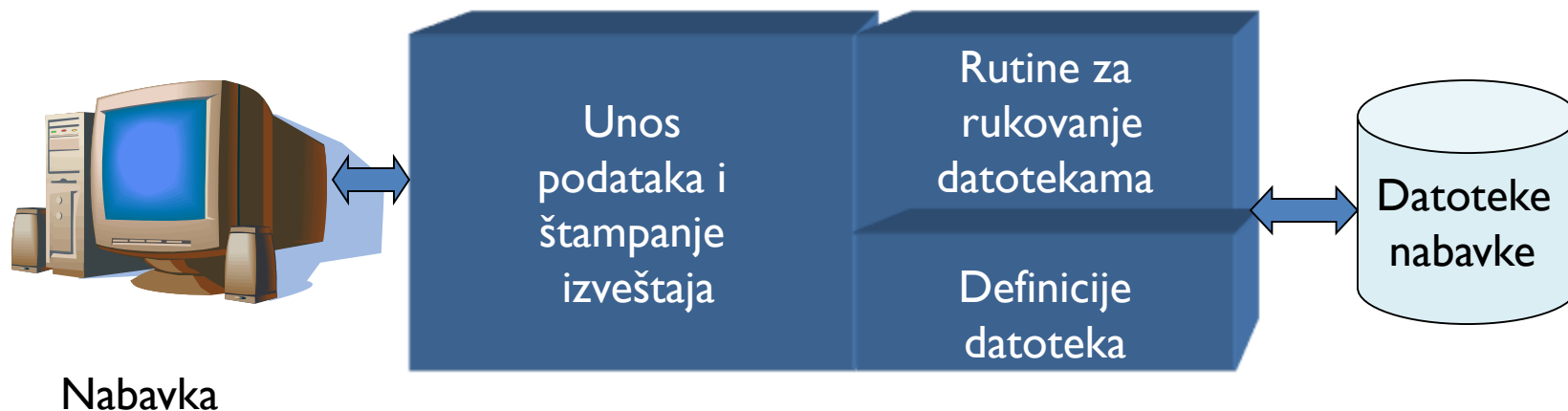
- ▶ Tradicionalni pristup (**sistem zasnovan na datotekama** engl. *File-based system*)
  - ▶ *Ad hoc* pristup: datoteka ili skup datoteka podataka potrebnih za definisanu aplikaciju
  - ▶ Razvijaju se programi koji obrađuju podatke iz datoteka
  - ▶ Svaki program definiše sopstvene podatke i upravlja tim podacima
- ▶ Obrada podataka (engl. *Data Processing*)
- ▶ Često se koristi pojam Automatska obrada podataka (AOP)
  - ▶ pojedine aplikacije jednog informacionog sistema (IS) međusobno nezavisne i kod koje se za svaku aplikaciju kreiraju i održavaju posebne datoteke sa svim potrebnim podacima

# Obrada zasnovana na datotekama

## Aplikacioni program prodaje



## Aplikacioni program nabavke



# Osnovne karakteristike sistema zasnovanog na datotekama

---

- ▶ Definicija podataka je ugrađena u aplikacioni program
  - ▶ poželjno je da bude odvojena i nezavisna
- ▶ Ne postoji kontrola pristupa podacima, osim one koju vrše aplikacioni programi
  - ▶ poželjna potpuna kontrola pristupa podacima i to nezavisno od konkretnog aplikacionog programa

# Prednosti i nedostaci konvencionalne obrade

---

## ► Prednosti

- Jednostavnost projektovanja i realizacije

## ► Nedostaci

- Nepovezanost aplikacija
- Ponavljanje podataka
- Neusaglašenost (nekonzistentnost) podataka
- Čvrsta povezanost programa i podataka
  - Programi za obradu podataka zavise od načina struktuiranja podataka
  - Promena strukture podataka zahteva promenu programa
- Ograničena mogućnost zajedničkog korišćenja podataka
- Ograničena raspoloživost podataka
- Neadekvatna realizacija oporavka od pada sistema

### **Posledice:**

- Obrada podataka je skupa

# Rešenje problema: baza podataka

---

- ▶ Baza podataka treba da obezbedi:
  - ▶ Nezavisnost struktura podataka od programa koji ih obrađuju,
  - ▶ Nezavisnost programa od struktura podataka;
  - ▶ Minimalnost ponavljanja podataka;
  - ▶ Da obrada podataka nije vezana za programski jezik opšte namene, već za viši “upitni” jezik;
  - ▶ Korišćenje baze podataka od strane većeg broja korisnika.



# Šta je baza podataka?

- ▶ **Baza podataka** (engl. *Data Base*, **Database**):
  - ▶ Integrisana kolekcija podataka o nekoj organizaciji
    - ▶ **Primer organizacija**: kompanija, banka, fakultet, grad, biblioteka, supermarket
- ▶ **Primer: Fakultetska baza podataka**
  - ▶ Obuhvata podatke o:
    - ▶ **Entitetima – objekti realnog sveta**
      - studijski programi,
      - studenti,
      - predmeti,
      - učionice
    - ▶ **Vezama** među ovim entitetima,
      - student sluša predmet,
      - predmet se izodi u učionici,
      - predmet pripada studijskom programu

# Baza podataka kao kolekcija podataka

---

- ▶ **Baza podataka** predstavlja **kolekciju povezanih** podataka organizovanih u logičke celine predstavljene **tabelama**.
- ▶ Skup podataka pripremljen tako da se mogu jednostavno **koristiti**, tj. pregledati, pretraživati, sortirati, upoređivati, itd., ali i menjati.
- ▶ Podaci u bazama podataka su organizovani u **dvodimenzionalne tabele ili relacije**
- ▶ **Relacione baze podataka**
  - ▶ Skup relacija/tabela
  - ▶ Tabela može da ima više **kolona**, gde svaka kolona predstavlja neku **osobinu** ili **atribut**.
  - ▶ **Vrste** tabele čine konkretni podaci, odnosno konkretne vrednosti osobina/atributa nekog objekta.

# Podaci u bazi podataka

- ▶ Dvodimenzionalna tabela (slika: tabela Student)
  - ▶ **Kolone** - osobine tj atributi objekta realnog sveta
  - ▶ **Vrste** – predstavlja jedan slog podataka o nekom objektu

Ime	Prezime	Indeks	MBR
Petar	Petrović	1111	123456
Milan	Milanović	2222	654321
Jovan	Jovanović	3333	345612

- ▶ Redosled podataka u tabeli nije relevantan
- ▶ Nema duplikata ili je njihov broj minimalan

# Relaciona baza podataka – primer veza između podataka

ADVISER : Table

	AdviserName	AdviserEmail
+	Baker	Baker@ourcampus.edu
+	Greene	Greene@ourcampus.edu
+	Taing	Taing@ourcampus.edu
+	Tran	Tran@ourcampus.edu
+	Valdez	Valdez@ourcampus.edu

Record: 5 of 5

STUDENT : Table

	StudentName	StudentEmail	Phone	Dorm	AdviserName
▶	Andrews, Matthew	MattA@ourcampus.edu	301.555.1234	McKinley	Baker
	Brisbon, Lisa	LisaB@ourcampus.edu	301.555.3335	Dorsett	Valdez
	Fischer, Douglas	DougF@ourcampus.edu	301.555.1688	McKinley	Baker
	Hwang, Terry	TerryH@ourcampus.edu	301.555.1837	McKinley	Taing
	Lai, Tzu	TzuL@ourcampus.edu	301.555.4139	Dorsett	Valdez
	Marino, Chip	ChipM@myserver.com	301.555.8665	Johnson	Tran
	Thompson, James	JamesT@myserver.com	301.555.3240	Johnson	Taing

Record: 1 of 7

Podaci o  
**ADVISERu** su  
povezani sa  
**STUDENTom**  
preko  
**AdviserData**

# Relacione baze podataka – primer rešavanja problema ažuriranja podataka

**ADVISER : Table**

	AdviserName	AdviserEmail
+	Baker	Baker@ourcampus.edu
+	Greene	Greene@ourcampus.edu
0	Taing	Taing2@ourcampus.edu
+	Tran	Tran@ourcampus.edu
+	Valdez	Valdez@ourcampus.edu

Record: 3 of 5

**STUDENT : Table**

	StudentName	StudentEmail	Phone	Dorm	AdviserName
▶	Andrews, Matthew	MattA@ourcampus.edu	301.555.1234	McKinley	Baker
	Brisbon, Lisa	LisaB@ourcampus.edu	301.555.3335	Dorsett	Valdez
	Fischer, Douglas	DougF@ourcampus.edu	301.555.1688	McKinley	Baker
	Hwang, Terry	TerryH@ourcampus.edu	301.555.1837	McKinley	Taing
	Lai, Tzu	TzuL@ourcampus.edu	301.555.4139	Dorsett	Valdez
	Marino, Chip	ChipM@myserver.com	301.555.8665	Johnson	Tran
	Thompson, James	JamesT@myserver.com	301.555.3240	Johnson	Taing

Record: 1 of 7

Dodata nova vrsta – ne zahtevaju se podaci o STUDENTu

Promenjen podatak – podaci ostaju konzistentni

Obrisana vrsta – ne gube se podaci o ADVISERu

# Baza podataka kao aspekt realnog sveta

---

- ▶ Slučajni skup podataka nije baza podataka!!
- ▶ Baza podataka **se projektuje i gradi za specifičnu namenu**
- ▶ Namenjena je konkretnim korisnicima za konkretne namene
- ▶ Baza podataka
  - ▶ predstavlja neki aspekt realnog sveta organizacije
  - ▶ tzv **minisvet** – deo realnog sveta za koji je neophodno čuvati i obrađivati podatke
    - ▶ Baza podataka za Fakultet – podaci o studentima, ispitima, nastavnicima
  - ▶ **Promene u minisvetu** utiču na bazu podataka
    - ▶ Promena u nastavnim planovima utiče na bazu podataka
      - ▶ Promene u podacima
      - ▶ Promene u tabelama

# Baze podataka: više posla?

---

- ▶ Relacione baze podataka su na prvi pogled komplikovanije od konvencionalnog pristupa
- ▶ Ipak, prednosti su značajne
  - ▶ pre svega minimizacija redundanse podataka, kao i
  - ▶ očuvanje kompleksnih veza između podataka
- ▶ Relaciona baza podataka predstavlja osnovu za korisničke forme i izveštaje
- ▶ Više drugačijih poslova – administracija, održavanje podataka...

# Šta je sistem za upravljanje bazama podataka?

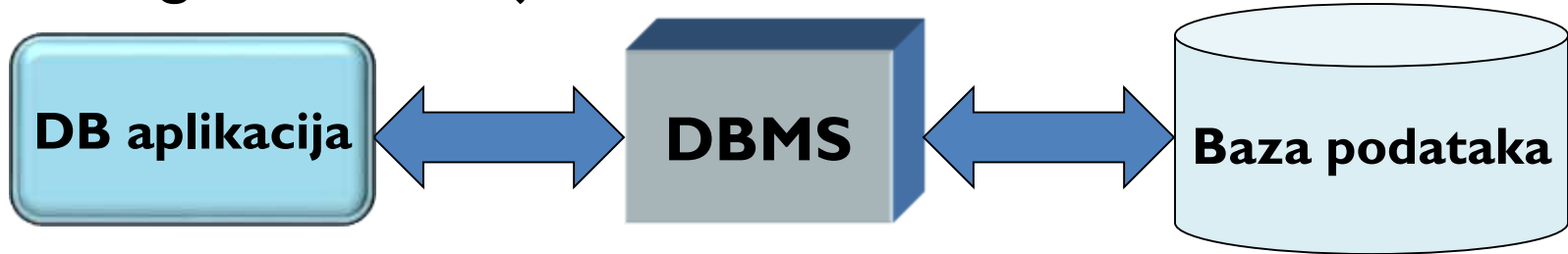
---

- ▶ **Sistem za upravljanje bazama podataka** (engl. *Data Base Management System, DBMS*)
- ▶ Softverski sistem koji omogućava definisanje, kreiranje i manipulisanje bazom podataka
  - ▶ **Definisanje:** Specificiranje tipova podataka, struktura i ograničenja za podatke koje treba memorisati u bazi podataka
  - ▶ **Kreiranje:** Proces memorisanja podataka na nekom medijumu koji kontroliše DBMS
  - ▶ **Manipulisanje:** Postavljanje upita bazi podataka da bi se našli specifični podaci, ažuriranje baze podataka da bi se unele promene nastale u mini svetu i generisanje izveštaja na osnovu podataka memorisanih u bazi podataka



# Šta je aplikacija baze podataka?

- ▶ **Aplikacija baze podataka** (engl. *Database application, DB application*)
  - ▶ Program koji interaguje sa bazom podataka u toku svog izvršenja
- ▶ Interakcija programa i baze podataka se obavlja preko DBMS-a
- ▶ Program šalje zahteve DBMS-u i DBMS vraća programu odgovor na ovaj zahtev



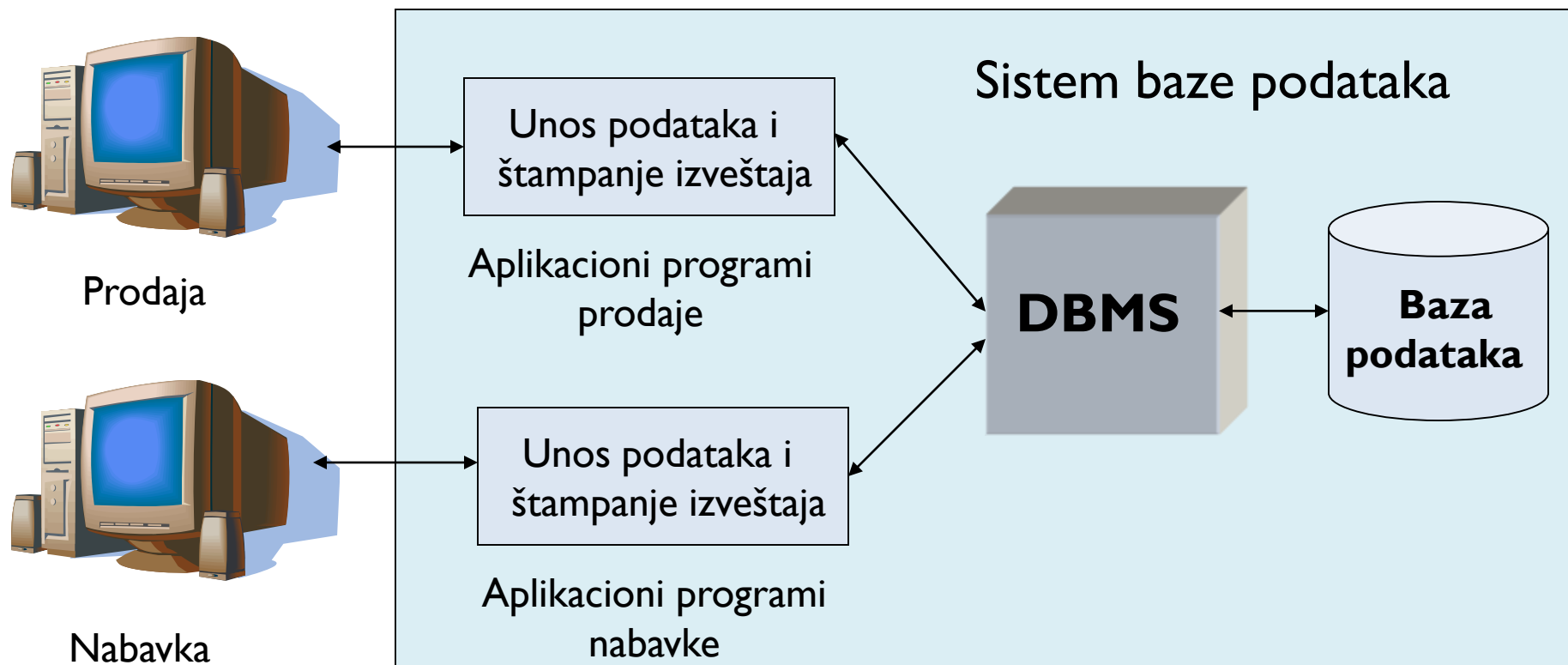
# Šta je sistem baze podataka?

- ▶ **Sistem baze podataka** (engl. *Database System*)
  - ▶ Kolekcija aplikacionih programa koji interaguju sa bazom podataka, DBMS i baza podataka

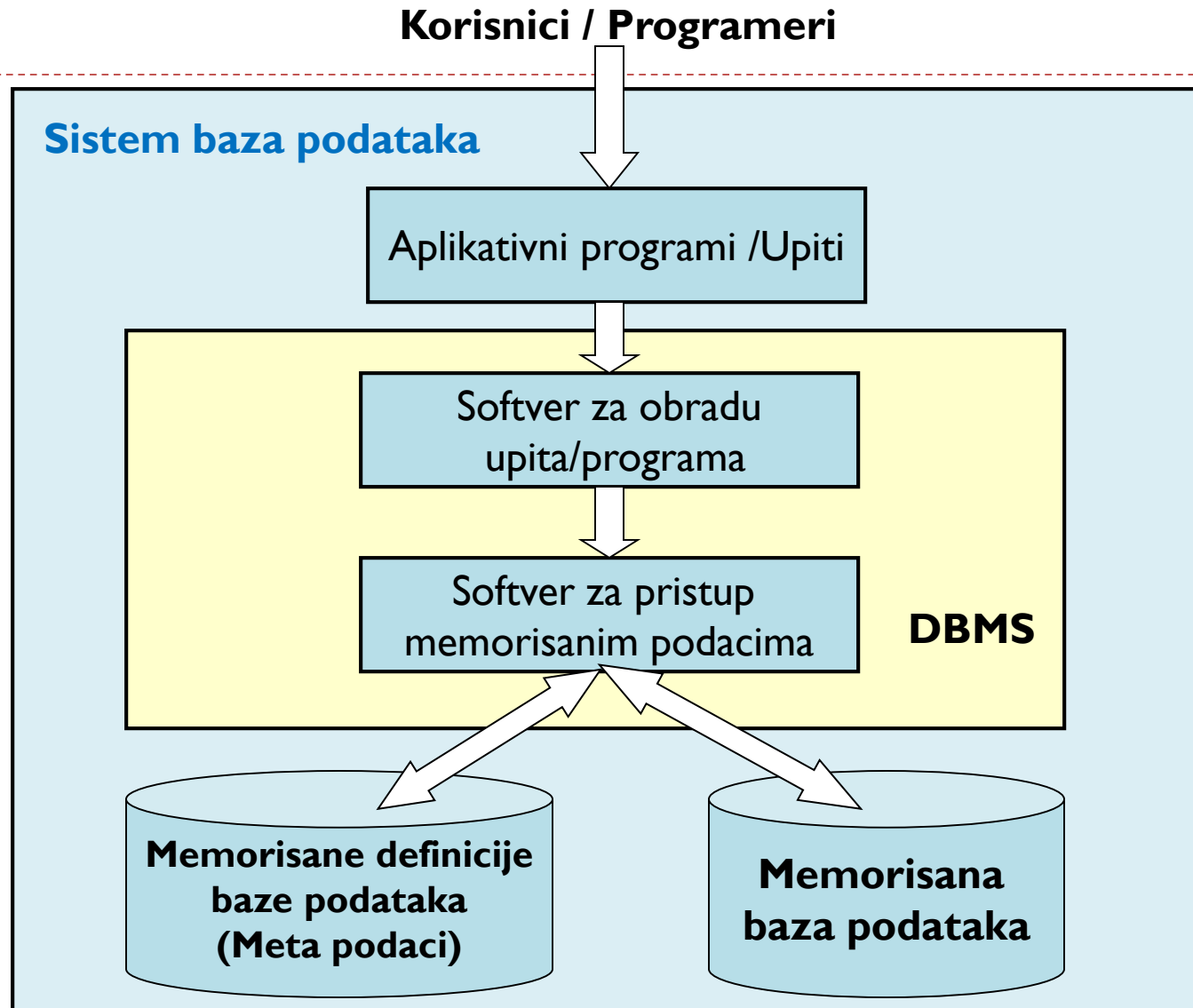
## *Sistem baze podataka*



# Obrada zasnovana na bazama podataka



# Uprošćena struktura sistema baze podataka



# Prilaz zasnovan na bazama podataka

---

## ▶ Baza podataka

- ▶ Deljiva kolekcija logički povezanih podataka i definicija tih podataka, projektovana sa ciljem da zadovolji informacione potrebe neke organizacije

## ▶ Važne odrednice

- ▶ Podaci su deljivi
- ▶ Baza podataka sadrži i podatke i njihove definicije (opisi, šeme)
- ▶ Definicije podataka se nalaze u **katalogu sistema (rečnik sistema)** i nazivaju se **meta podaci**
- ▶ Definicije podataka su odvojene od aplikacionih programa
- ▶ Podaci su logički povezani (entiteti, atributi, veze)

# Svojstva prilaza zasnovanog na BP

---

- ▶ **Opisi podataka i ograničenja nad podacima** (tzv. **meta podaci**) su sastavni deo sistema baze podataka
  - ▶ Opisi se nalaze u sistemskom **katalogu** koji je dostupan i DBMS-u i korisnicima kojima je potrebno poznavanje strukture baze podataka
- ▶ **Nezavisnost programa i podataka**
  - ▶ Opisi podataka su u DBMS katalogu i nezavisni su od programa za pristup podacima

# Uloge u okruženju baza podataka (osoblje)

---

- ▶ **Administrator podataka i administrator baze podataka**
  - ▶ Data Administrator (DA)
  - ▶ Database Administrator (DBA)
- ▶ **Projektant baze podataka (eng. Database Designer)**
  - ▶ Projektant logičke baze podataka
  - ▶ Projektant fizičke baze podataka
- ▶ **Inženjer razvoja aplikacija (eng. Application Developer)**
  - ▶ aplikacioni programer
- ▶ **Krajnji korisnici (eng. End-Users)**
  - ▶ Slučajni korisnici
  - ▶ Naivni ili parametarski korisnici
  - ▶ Napredni ili Sofistični korisnici

# Administrator podataka

---

- ▶ Odgovoran je za menadžment podataka kao resursa, uključujući:
  - ▶ Planiranje, razvoj i održavanje standarda, politike i procedura
  - ▶ Konceptualno i logičko projektovanje baze podataka
- ▶ Njegov zadatak je da obezbedi DB podršku za ostvarenje postavljenih ciljeva korporacije za koju radi



# Administrator baze podataka

---

- ▶ Odgovoran je za fizičku realizaciju sistema baze podataka, uključujući:
  - ▶ Fizičko projektovanje i implementaciju baze podataka
  - ▶ Sigurnost i integritet
  - ▶ Održavanje sistema u radnom stanju
  - ▶ Obezbeđenje odgovarajućih performansi aplikacija za potrebe korisnika (monitoring i reorganizacija, po potrebi)

# Glavni zadaci DB i DBA

## DA

- ▶ Učestvuje u strateškom planiranju IS-a
- ▶ Utvrđuje dugoročne ciljeve
- ▶ Nameće standarde, politiku i procedure
- ▶ Utvrđuje zahteve za podacima
- ▶ Razvija konceptualni i logički projekat baze podataka
- ▶ Razvija i održava zajednički model podataka
- ▶ Koordinira razvoj sistema
- ▶ Orijentisan je ka menadžmentu
- ▶ **DBMS nezavistan**

## DBA

- ▶ Evaluira nove DBMS-e
- ▶ Izvršava planove radi ostvarenja ciljeva
- ▶ Nameće standarde, politiku i procedure
- ▶ Implementira zahteve za podacima
- ▶ Razvija logički i fizički projekat baze podataka
- ▶ Implementira fizički projekat baze podataka
- ▶ Nadzire i kontroliše bazu podataka
- ▶ Orijentisan je ka tehnici
- ▶ **DBMS zavistan**

# Korisnici baze podataka

---

## ▶ Akteri na sceni

- ▶ Administrator baze podataka
- ▶ Projektanti baze podataka
- ▶ Aplikacioni programeri
- ▶ Sistem analitičari
- ▶ Krajnji korisnici
  - ▶ Slučajni
  - ▶ Naivni
  - ▶ Napredni

## ▶ Akteri iza scene

- ▶ Projektanti DBMS-a
- ▶ Inženjeri razvoja DBMS-a
- ▶ Operateri
- ▶ Osoblje za održavanje sistema baze podataka

# Krajnji korisnici

- ▶ **Krajnji korisnici** su lica čiji posao zahteva pristup bazi podataka radi pretraživanja, ažuriranja i generisanja izveštaja
- ▶ Vrste krajnjih korisnika
  - ▶ **Slučajni korisnici** povremeno pristupaju bazi podataka, njihovi zahtevi su vrlo promenjivi i nisu unapred definisani
    - ▶ Za njih se projektuju posebni upitni jezici i dodatna softverska pomagala
  - ▶ **Naivni (parametarski) korisnici** ažuriraju bazu podataka i koristeći standardne upite dobijaju potrebne podatke iz baze podataka
    - ▶ Primer ovih korisnika su službenici na šalterima
    - ▶ Ovi korisnici uvek obavljaju istu obradu nad bazom podataka i za njih se razvijaju posebne DB aplikacije
  - ▶ **Napredni (sofistički) korisnici** imaju znanje o DBMS-u tako da mogu formulisati vrlo složene zahteve
    - ▶ Primer takvih korisnika su inženjeri, naučnici, istraživači, analitičari

# Tipovi sistema baza podataka

---

- ▶ **Single-user:**
  - ▶ Podrška samo za jednog korisnika u isto vreme
- ▶ **Desktop:**
  - ▶ Single-user baza podataka na personalnom računaru
- ▶ **Multi-user:**
  - ▶ Podrška za više korisnika u isto vreme
- ▶ **Workgroup:**
  - ▶ Multi-user baza podataka za mali broj korisnika ili jedno odeljenje preduzeća
- ▶ **Enterprise:**
  - ▶ Multi-user baza podataka koja podržava veliki broj korisnika ili celo preduzeće/organizaciju

# Tipovi sistema baza podataka

---

## Klasifikacija po lokaciji:

- ▶ Centralizovane:
  - ▶ Podaci su na jednoj lokaciji
- ▶ Distribuirane:
  - ▶ Podaci mogu biti na više lokacija

## Po načinu korišćenja:

- ▶ Transakcije (ili produkcione):
  - ▶ Podrška za *day-to-day* operacije preduzeća
- ▶ Data warehouse:
  - ▶ Čuva podatke koji se koriste za generisanje informacija potrebnih za strateške odluke.
  - ▶ Često sadrže istorijske podatke i imaju drugačiju strukturu

# Prednosti DBMS-a

---

- ▶ **Nezavisnost podataka**

- ▶ Aplikacioni programi su nezavisni od reprezentacije i načina memorisanja podataka
- ▶ DBMS nudi aplikaciji apstraktni pogled na podatke

- ▶ **Efikasan pristup podacima**

- ▶ DBMS koristi posebne tehnike za efikasno memorisanje i pretraživanje podataka

- ▶ **Integritet i bezbednost podataka**

- ▶ DBMS kontroliše integritet podataka koristeći ograničenja koja je definisao projektant baze podataka
- ▶ DBMS nadgleda pristupne privilegije korisnika, tako da svaki korisnik može videti samo podatke za koje poseduje pristupne privilegije

- ▶ **Administracija podataka**

- ▶ Kada su podaci deljivi centralizovana administracija podataka donosi značajno poboljšanje
- ▶ Za to je zadužen administrator baze podataka

- ▶ **Konkurentni pristup i oporavak**

- ▶ DBMS upravlja konkurentnim pristupom podacima od strane više korisnika
- ▶ DBMS ima ugrađene mehanizme oporavka baze podataka od različitih neispravnosti

- ▶ **Kraće vreme razvoja aplikacija**

- ▶ DBMS ima ugrađene različite funkcije za rad sa podacima što aplikacionim programerima značajno olakšava zadatak razvoja aplikacije
- ▶ DBMS nudi interfejs iz jezika visokog nivoa ka bazi podataka

# Nedostaci DBMS-a

---

- ▶ Složenost
- ▶ Veličina
- ▶ Troškovi za DBMS
- ▶ Dodatni troškovi za HW
- ▶ Troškovi konverzije
- ▶ Performanse
- ▶ Veći uticaj neispravnosti



# Istorijat DBMS-a - 1.deo

---

- ▶ 60-tih godina 20-tog veka
  - ▶ projekat Apollo, na inicijativu predsednika Kenedija
  - ▶ Rezultat ovog projekta je GUAM (Generalized Update Access Method)
- ▶ Sredinom 60-tih
  - ▶ IBM je prihvatio GUAM ideju i razvio IMS (Information Management System) – to je **hijerarhijski DBMS**
- ▶ Sredinom 60-tih godina
  - ▶ General Electric je razvio IDS (Integrated Data Store) – to je **mrežni DBMS**
  - ▶ Prvi DBMS opšte namene koga je projektovao Charles Bachman, prvi dobitnik Turing-ove nagrade koju dodeljuje ACM (ekvivalent Nobelove nagrade za Računarske nauke)
  - ▶ Bachman je nagradu dobio za rad u oblasti baza podataka
- ▶ 1967. je formirana, u okviru CODASYL-a, Database Task Group (DBTG) koja je uradila **standard za DBMS**:
  - ▶ 1969 draft report
  - ▶ 1971 final report

# Istorijat DBMS-a - 2.deo

---

- ▶ 1970. E.F.Codd iz IBM Research Lab je predložio **relacioni model**
- ▶ Prvi komercijalni relacioni DBMS su proizvedeni krajem 70-tih i početkom 80-tih godina 20-tog veka:
  - ▶ Projekat System R (IBM) krajem 70-tih (SQL)
  - ▶ DB2 i SQL/DS (IBM) 80-tih
  - ▶ Oracle (Oracle Corporation) 80-tih
- ▶ Danas postoji na hiljade relacionih DBMS-ova za mainframe i PC okruženja:
  - ▶ INGRES (Computer Associates)
  - ▶ INFORMIX, DB2 (IBM)
  - ▶ Office Access, Visual FoxPro (Microsoft)
  - ▶ InterBase, JDataStore (Borland)
  - ▶ R:Base (R:Base Technologies)

# Istorijat DBMS-a - 3.deo

---

- ▶ 1976. Chen je predložio **ER model**
- ▶ 1979. Codd je predložio **proširenu verziju relacionog modela** RM/T i 1990. RM/V2 koji se ubraja u **semantičke modele podataka**
- ▶ 90-tih godina 20-tog veka **Objektno-orijentisani DBMS (OODBMS)** i **Objektno-relacioni DBMS (ORDBMS)**
- ▶ Početak ovog veka
  - ▶ **Postrelacione BP**
  - ▶ **XML native BP**
- ▶ ... **Relacione baze podataka + nove tehnologije**
  - ▶ **NoSQL baze podataka**
  - ▶ **Trendovi vraćanja na stare, dobre vrednosti RDBMS**

*Baze podataka*

*Prof. dr Leonid Stoimenov*

*Katedra za računarstvo*

*Elektronski fakultet u Nišu*

# Uvod u baze podataka

Kraj predavanja

Pitanja ???