



## Koncepcija baze podataka

---

*Istorijat razvoja postupaka za  
upravljanje podacima*

# Sadržaj

- Motivacija
- OM i klasična organizacija datoteka
- Baze podataka i SUBP
- Šema baze podataka
- Podšema / Eksterna šema
- Pogled
- Sistemi baza podataka

# Motivacija

---

- Vrednost svakog sistema, pa i sistema baza podataka, najbolje se shvata
  - ne samo na osnovu poznavanja samog sistema
  - već na osnovu činjenice da taj sistem predstavlja korak u evoluciji rešavanja onih problema, koje prethodni sistemi nisu mogli da reše
- Da bi se stekla precizna slika o bazama podataka
  - nije dovoljno samo definisati pojam baze podataka
  - potrebno je prvo baze podataka sagledati u kontekstu njihovog istorijskog razvoja

# Sadržaj

---

- Motivacija
- OM i klasična organizacija datoteka
- Baze podataka i SUBP
- Šema baze podataka
- Podšema / Eksterna šema
- Pogled
- Sistemi baza podataka

# Operativna memorija

---

- Podaci se mogu skladištiti u OM
- Brza i mogućnost random pristupa
- Mogućnost koju nudi svaki programski jezik
- Nedostaci:
  - mali kapacitet
  - privremena

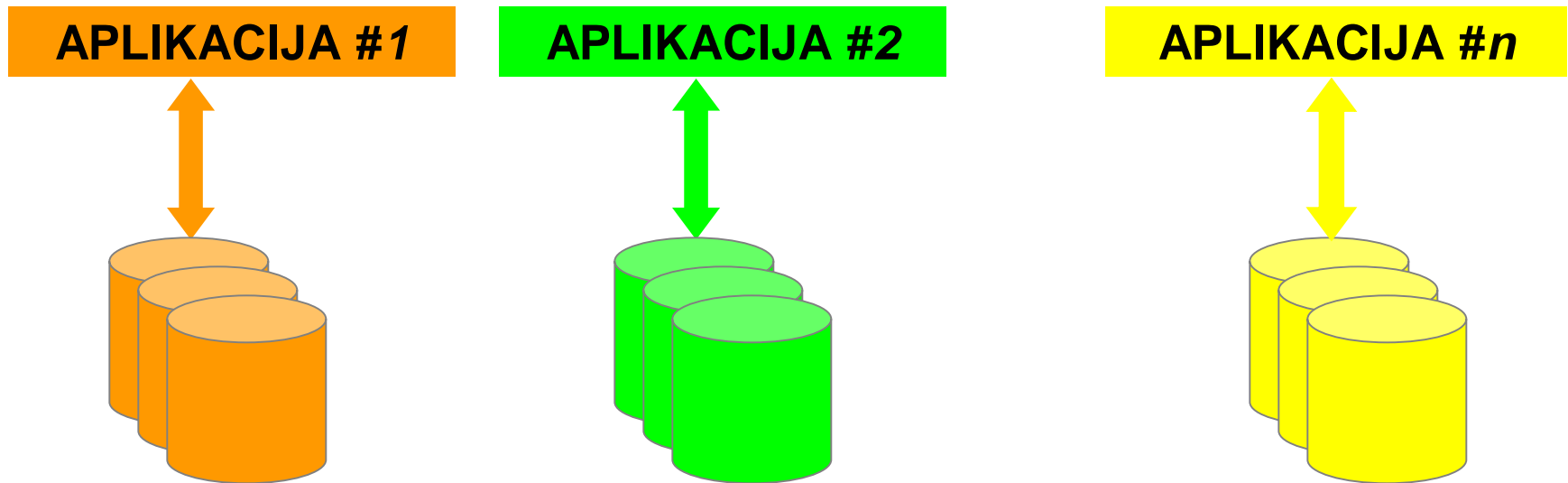
# Klasična organizacija datoteka

---

- Najstarije rešenje
  - IS "na karticama", bez podrške računara
    - medijum za trajno memorisanje struktura podataka - papir
- Naprednije rešenje
  - IS organizovani nad sistemima datoteka
    - medijum za trajno memorisanje struktura podataka – sistem diskova
  - IS je sačinjavao skup nezavisnih aplikacija
    - svaka aplikacija - sopstvene datoteke
    - “skladište podataka” - skup datoteka
    - podaci o istom entitetu u različitim datotekama
    - vremenom, takav IS dolazi u kontradikciju sa samim sobom

# Klasična organizacija datoteka

---



# Klasična organizacija datoteka

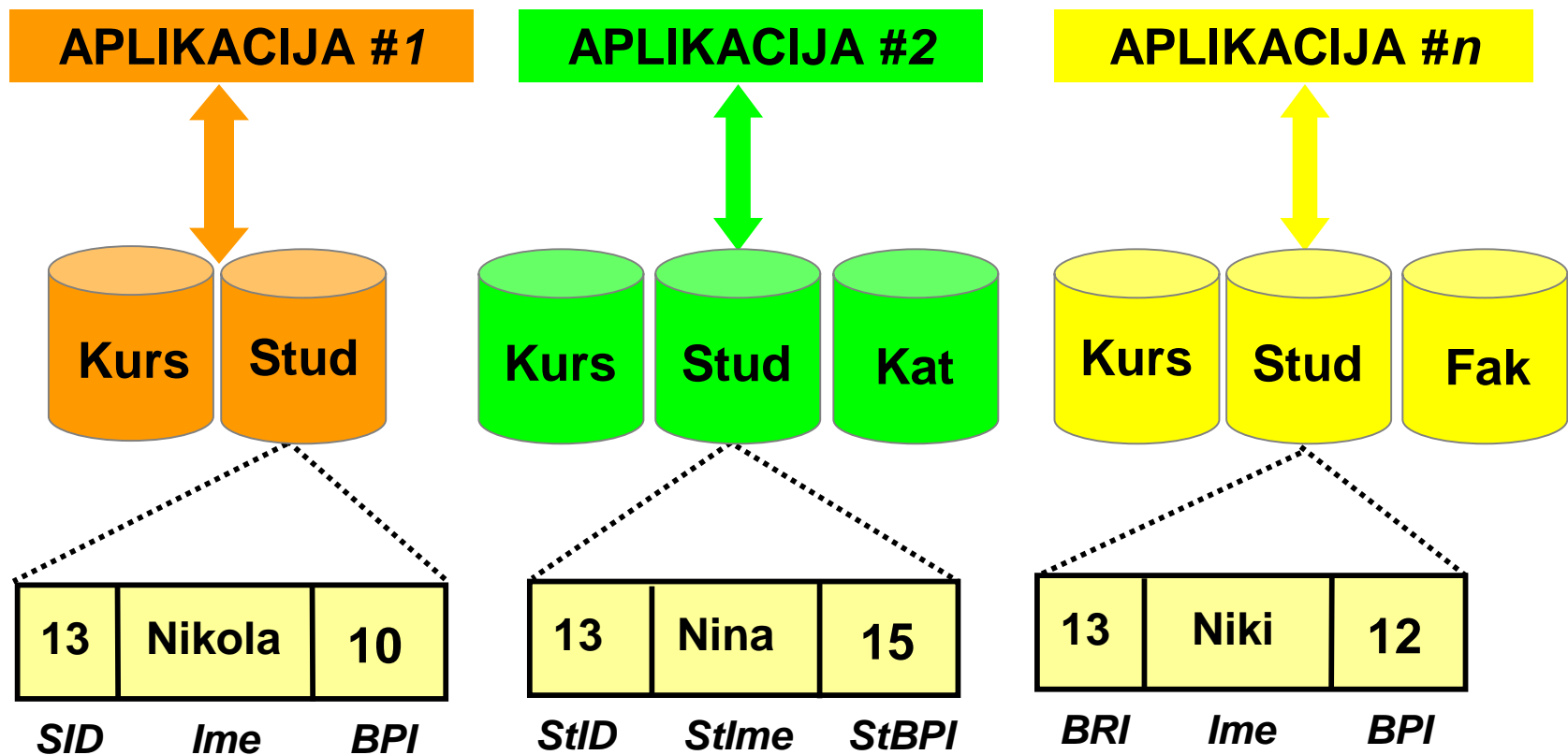
---

- Osnovni nedostaci
  - nepovezanost aplikacija
    - potreba ručnog prepisivanja istih ili sličnih podataka
  - redundantnost podataka
    - potreba višestrukog memorisanja istih ili sličnih podataka
  - čvrsta povezanost programa i podataka
    - program vodi računa o FSP datoteke, kako u opisu, tako i u proceduri
  - konkurentni pristup više korisnika
- Posledice
  - otežano održavanje IS-a
  - otežan dalji razvoj IS-a



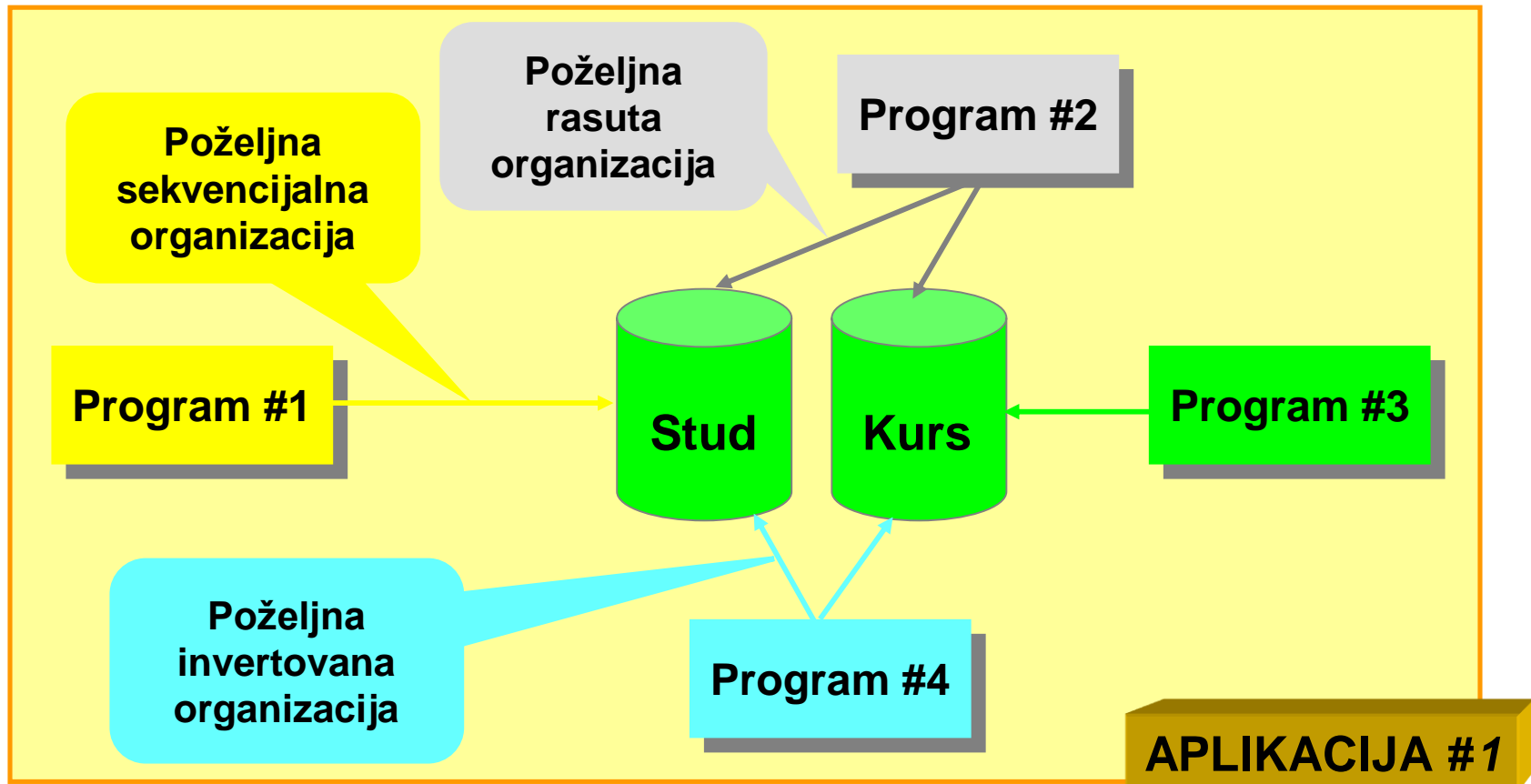
# Klasična organizacija datoteka

- Primer - nepovezanost i redundantnost



# Klasična organizacija datoteka

- Čvrsta povezanost programa i podataka



# Klasična organizacija datoteka

- Čvrsta povezanost – primer COBOL

**IDENTIFICATION DIVISION.**  
**PROGRAM-ID. Seq2Rel.**  
**AUTHOR. MC**

```
IDENTIFICATION DIVISION.
PROGRAM-ID. Seq2Rel.
AUTHOR. MC

ENVIRONMENT DIVISION.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT SupplierFile ASSIGN TO "RELSUPP.DAT"
        ORGANIZATION IS RELATIVE
        ACCESS MODE IS RANDOM
        RELATIVE KEY IS SupplierKey
        FILE STATUS IS Supplierstatus.

    SELECT SupplierFileSeq ASSIGN TO "SEQSUPP.DAT"
        ORGANIZATION IS LINE SEQUENTIAL.

DATA DIVISION.
FILE SECTION.

FD SupplierFile.
01 SupplierRecord.
    02 SupplierCode          PIC 99.
    02 SupplierName          PIC X(20).
    02 SupplierAddress       PIC X(50).

FD SupplierFileSeq.
01 SupplierRecordSeq.
    88 EndOfFile             VALUE HIGH-VALUES.
    02 SupplierCodeSeq       PIC 99.
    02 SupplierNameSeq       PIC X(20).
    02 SupplierAddressSeq    PIC X(50).

WORKING-STORAGE SECTION.
01 SupplierStatus           PIC X(2).

01 SupplierKey              PIC 99.

PROCEDURE DIVISION.
Begin.
    OPEN OUTPUT SupplierFile.
    OPEN INPUT SupplierFileSeq.

    READ SupplierFileSeq
        AT END SET EndOfFile TO TRUE
    END-READ
    PERFORM UNTIL EndOfFile
        MOVE SupplierCodeSeq TO SupplierKey
        MOVE SupplierRecordSeq TO SupplierRecord
        WRITE SupplierRecord
            INVALID KEY DISPLAY "Supplier status = " SupplierStatus
        END-WRITE
        READ SupplierFileSeq
            AT END SET EndOfFile TO TRUE
        END-READ
    END-PERFORM.

    CLOSE SupplierFile, SupplierFileSeq.
    STOP RUN.
```

# Klasična organizacija datoteka

- Čvrsta povezanost – primer COBOL

**ENVIRONMENT DIVISION.**

**INPUT-OUTPUT SECTION.**

**FILE-CONTROL.**

**SELECT SupplierFile ASSIGN TO "RELSUPP.DAT"**

**ORGANIZATION IS RELATIVE**

**ACCESS MODE IS RANDOM**

**RELATIVE KEY IS SupplierKey**

**FILE STATUS IS Supplierstatus.**

**SELECT SupplierFileSeq ASSIGN TO  
"SEQSUPP.DAT"**

**ORGANIZATION IS LINE SEQUENTIAL.**

```
IDENTIFICATION DIVISION.
PROGRAM-ID. Seq2Rel.
AUTHOR. MC

ENVIRONMENT DIVISION.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT SupplierFile ASSIGN TO "RELSUPP.DAT"
    ORGANIZATION IS RELATIVE
    ACCESS MODE IS RANDOM
    RELATIVE KEY IS SupplierKey
    FILE STATUS IS Supplierstatus.

    SELECT SupplierFileSeq ASSIGN TO "SEQSUPP.DAT"
    ORGANIZATION IS LINE SEQUENTIAL.

DATA DIVISION.
FILE SECTION.

FD SupplierFile.
01 SupplierRecord.
02 SupplierCode
02 SupplierName
02 SupplierAddress
PIC 99.
PIC X(20).
PIC X(50).

FD SupplierFileSeq.
01 SupplierRecordSeq.
88 EndOfFile
02 SupplierCodeSeq
02 SupplierNameSeq
02 SupplierAddressSeq
VALUE HIGH-VALUES.
PIC 99.
PIC X(20).
PIC X(50).

WORKING-STORAGE SECTION.
01 SupplierStatus
01 SupplierKey
PIC X(2).
PIC 99.

PROCEDURE DIVISION.
Begin.
    OPEN OUTPUT SupplierFile.
    OPEN INPUT SupplierFileSeq.

    READ SupplierFileSeq
    AT END SET EndOfFile TO TRUE
END-READ
PERFORM UNTIL EndOfFile
    MOVE SupplierCodeSeq TO SupplierKey
    MOVE SupplierRecordSeq TO SupplierRecord
    WRITE SupplierRecord
    INVALID KEY DISPLAY "Supplier status = " SupplierStatus
END-WRITE
    READ SupplierFileSeq
    AT END SET EndOfFile TO TRUE
END-READ
END-PERFORM.

CLOSE SupplierFile, SupplierFileSeq.
STOP RUN.
```

# Klasična organizacija datoteka

- Čvrsta povezanost – primer COBOL

**DATA DIVISION.**

**FILE SECTION.**

**FD SupplierFile.**

**01 SupplierRecord.**

02 SupplierCode	PIC 99.
02 SupplierName	PIC X(20).
02 SupplierAddress	PIC X(50).

**FD SupplierFileSeq.**

**01 SupplierRecordSeq.**

88 EndOfFile	VALUE HIGH-VALUES.
02 SupplierCodeSeq	PIC 99.
02 SupplierNameSeq	PIC X(20).
02 SupplierAddressSeq	PIC X(50).

```
IDENTIFICATION DIVISION.
PROGRAM-ID. Seq2Rel.
AUTHOR. MC

ENVIRONMENT DIVISION.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT SupplierFile ASSIGN TO "RELSUPP.DAT"
        ORGANIZATION IS RELATIVE
        ACCESS MODE IS RANDOM
        RELATIVE KEY IS SupplierKey
        FILE STATUS IS Supplierstatus.

    SELECT SupplierFileSeq ASSIGN TO "SEQSUPP.DAT"
        ORGANIZATION IS LINE SEQUENTIAL.
```

```
DATA DIVISION.
FILE SECTION.
```

```
FD SupplierFile.
01 SupplierRecord.
    02 SupplierCode      PIC 99.
    02 SupplierName      PIC X(20).
    02 SupplierAddress   PIC X(50).
```

```
FD SupplierFileSeq.
01 SupplierRecordSeq.
    88 EndOfFile         VALUE HIGH-VALUES.
    02 SupplierCodeSeq   PIC 99.
    02 SupplierNameSeq   PIC X(20).
    02 SupplierAddressSeq PIC X(50).
```

```
WORKING-STORAGE SECTION.
01 SupplierStatus       PIC X(2).
```

```
01 SupplierKey          PIC 99.
```

```
PROCEDURE DIVISION.
```

```
Begin.
```

```
    OPEN OUTPUT SupplierFile.
    OPEN INPUT SupplierFileSeq.
```

```
    READ SupplierFileSeq
        AT END SET EndOfFile TO TRUE
    END-READ
    PERFORM UNTIL EndOfFile
        MOVE SupplierCodeSeq TO SupplierKey
        MOVE SupplierRecordSeq TO SupplierRecord
        WRITE SupplierRecord
            INVALID KEY DISPLAY "Supplier status = " SupplierStatus
        END-WRITE
        READ SupplierFileSeq
            AT END SET EndOfFile TO TRUE
        END-READ
    END-PERFORM.
```

```
    CLOSE SupplierFile, SupplierFileSeq.
    STOP RUN.
```

# Klasična organizacija datoteka

- Čvrsta povezanost – primer COBOL

## WORKING-STORAGE SECTION.

01 SupplierStatus PIC X(2).

01 SupplierKey PIC 99.

```
IDENTIFICATION DIVISION.
PROGRAM-ID. Seq2Rel.
AUTHOR. MC

ENVIRONMENT DIVISION.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT SupplierFile ASSIGN TO "RELSUPP.DAT"
        ORGANIZATION IS RELATIVE
        ACCESS MODE IS RANDOM
        RELATIVE KEY IS SupplierKey
        FILE STATUS IS Supplierstatus.

    SELECT SupplierFileSeq ASSIGN TO "SEQSUPP.DAT"
        ORGANIZATION IS LINE SEQUENTIAL.

DATA DIVISION.
FILE SECTION.

FD SupplierFile.
01 SupplierRecord.
    02 SupplierCode PIC 99.
    02 SupplierName PIC X(20).
    02 SupplierAddress PIC X(50).

FD SupplierFileSeq.
01 SupplierRecordSeq.
    88 EndOfFile VALUE HIGH-VALUES.
    02 SupplierCodeSeq PIC 99.
    02 SupplierNameSeq PIC X(20).
    02 SupplierAddressSeq PIC X(50).

WORKING-STORAGE SECTION.
01 SupplierStatus PIC X(2).

01 SupplierKey PIC 99.

PROCEDURE DIVISION.
Begin.
    OPEN OUTPUT SupplierFile.
    OPEN INPUT SupplierFileSeq.

    READ SupplierFileSeq
        AT END SET EndOfFile TO TRUE
    END-READ
    PERFORM UNTIL EndOfFile
        MOVE SupplierCodeSeq TO SupplierKey
        MOVE SupplierRecordSeq TO SupplierRecord
        WRITE SupplierRecord
            INVALID KEY DISPLAY "Supplier status = " SupplierStatus
        END-WRITE
        READ SupplierFileSeq
            AT END SET EndOfFile TO TRUE
        END-READ
    END-PERFORM.

CLOSE SupplierFile, SupplierFileSeq.
STOP RUN.
```

# Klasična organizacija datoteka

## • Čvrsta povezanost – primer COBOL

**PROCEDURE DIVISION.**

**BEGIN.**

**OPEN OUTPUT SupplierFile.**

**OPEN INPUT SupplierFileSeq.**

**READ SupplierFileSeq**

**AT END SET EndOfFile TO TRUE**

**END-READ**

**PERFORM UNTIL EndOfFile**

**MOVE SupplierCodeSeq TO SupplierKey**

**MOVE SupplierRecordSeq TO SupplierRecord**

**WRITE SupplierRecord**

**INVALID KEY DISPLAY "Supplier status = " SupplierStatus**

**END-WRITE**

**READ SupplierFileSeq**

**AT END SET EndOfFile TO TRUE**

**END-READ**

**END-PERFORM.**

**CLOSE SupplierFile, SupplierFileSeq.**

**STOP RUN.**

**IDENTIFICATION DIVISION.**

**PROGRAM-ID. Seq2Rel.**

**AUTHOR. MC**

**ENVIRONMENT DIVISION.**

**INPUT-OUTPUT SECTION.**

**FILE-CONTROL.**

**SELECT SupplierFile ASSIGN TO "RELSUPP.DAT"**

**ORGANIZATION IS RELATIVE**

**ACCESS MODE IS RANDOM**

**RELATIVE KEY IS SupplierKey**

**FILE STATUS IS Supplierstatus.**

**SELECT SupplierFileSeq ASSIGN TO "SEQSUPP.DAT"**

**ORGANIZATION IS LINE SEQUENTIAL.**

**DATA DIVISION.**

**FILE SECTION.**

**FD SupplierFile.**

**01 SupplierRecord.**

**02 SupplierCode PIC 99.**

**02 SupplierName PIC X(20).**

**02 SupplierAddress PIC X(50).**

**FD SupplierFileSeq.**

**01 SupplierRecordSeq.**

**88 EndOfFile VALUE HIGH-VALUES.**

**02 SupplierCodeSeq PIC 99.**

**02 SupplierNameSeq PIC X(20).**

**02 SupplierAddressSeq PIC X(50).**

**WORKING-STORAGE SECTION.**

**01 SupplierStatus PIC X(2).**

**01 SupplierKey PIC 99.**

**PROCEDURE DIVISION.**

**Begin.**

**OPEN OUTPUT SupplierFile.**

**OPEN INPUT SupplierFileSeq.**

**READ SupplierFileSeq**

**AT END SET EndOfFile TO TRUE**

**END-READ**

**PERFORM UNTIL EndOfFile**

**MOVE SupplierCodeSeq TO SupplierKey**

**MOVE SupplierRecordSeq TO SupplierRecord**

**WRITE SupplierRecord**

**INVALID KEY DISPLAY "Supplier status = " SupplierStatus**

**END-WRITE**

**READ SupplierFileSeq**

**AT END SET EndOfFile TO TRUE**

**END-READ**

**END-PERFORM.**

**CLOSE SupplierFile, SupplierFileSeq.**

**STOP RUN.**

# Klasična organizacija datoteka

---

- Čvrsta povezanost – primer COBOL

**SELECT STUDENT ASSIGN TO “STUD.DAT”  
ORGANIZATION IS INDEXED  
ACCESS MODE IS RANDOM  
RECORD KEY IS BRI  
ALTERNATE RECORD KEY IS PREZIME  
WITH DUPLICATES  
FILE STATUS IS StudStatus.**

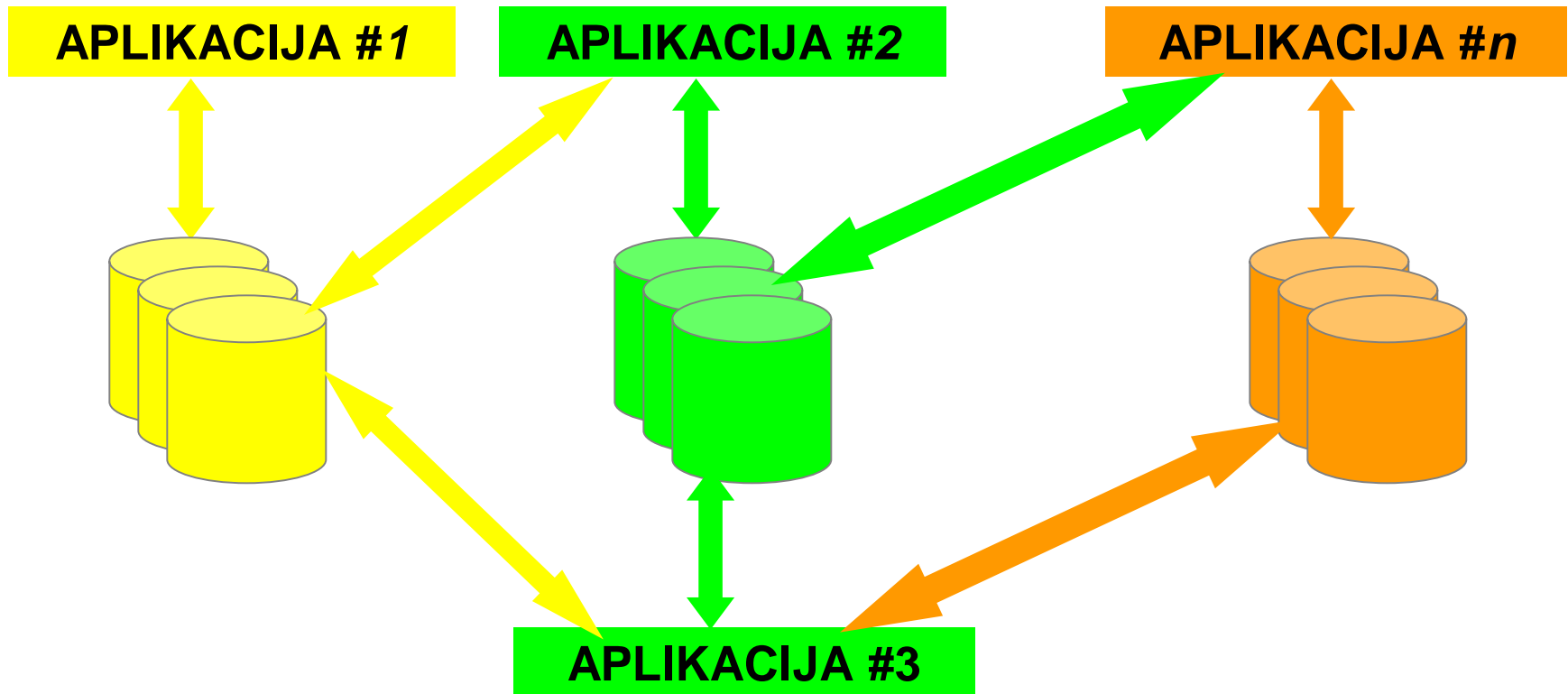


# Klasična organizacija datoteka

---

- Problemi koji se mogu ublažiti, ili čak razrešiti u klasičnoj organizaciji
  - nepovezanost aplikacija
  - redundantnost
- Problem koji je gotovo nemoguće ublažiti ili razrešiti u klasičnoj organizaciji
  - čvrsta povezanost programa i podataka

# Klasična organizacija datoteka



# Sadržaj

---

- Motivacija
- OM i klasična organizacija datoteka
- Baze podataka i SUBP
- Šema baze podataka
- Podšema / Eksterna šema
- Pogled
- Sistemi baza podataka

# Baze podataka i SUBP

---

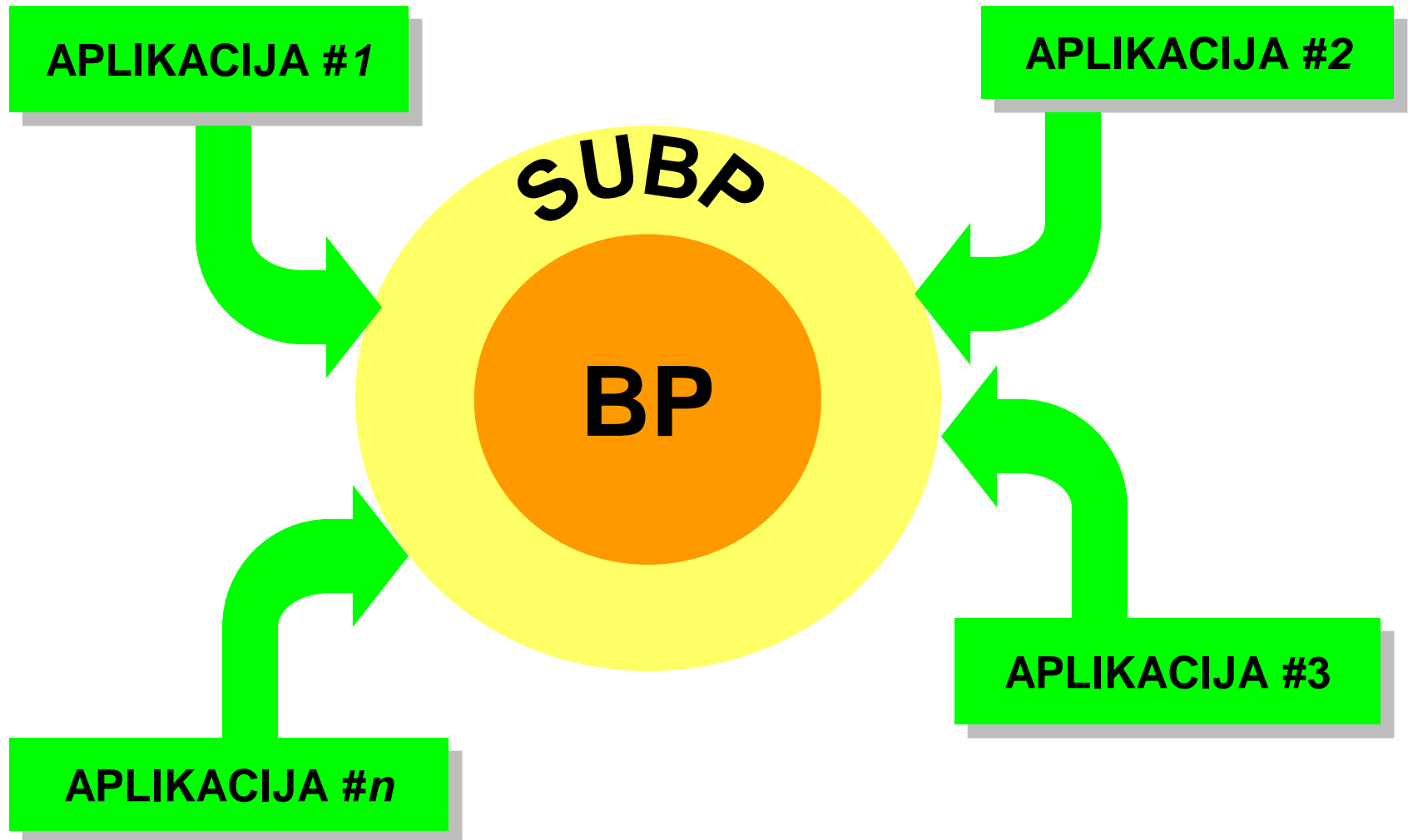
- Osnovne ideje:
  - da se svi podaci jednog IS integrišu u jednu veliku “datoteku”
    - nastanak pojma baze podataka
  - nereduntantno memorisanje podataka
    - izbegavanje nepotrebnog višestrukog memorisanja istih ili sličnih podataka
  - da se uvede poseban softverski proizvod za podršku kreiranja i korišćenja baze podataka
    - sistem za upravljanje bazama podataka (SUBP)
    - da svi programi koriste podatke iz baze podataka, ili je ažuriraju koristeći isključivo usluge SUBP

# Baze podataka i SUBP

---

- Osnovne ideje:
  - transakcijska obrada
    - teška implementacija nad sistemom datoteka
    - SUBP ima ugrađenu podršku
  - višekorisnički konkurentni pristup
  - autorizacija korisnika

# Baze podataka i SUBP



# Definicija BP

---

- *James Martin (IBM)*: Skup međusobno povezanih podataka, pohranjenih zajedno, bez bespotrebne redundanse, koji mogu zadovoljiti različite primene. Podaci koji su pohranjeni na način nezavisan od programa koji ih koriste. Prilikom dodavanja novih podataka, menjanja i pretraživanja postojećih podataka, primenjuje se zajednički i kontrolisani pristup. Podaci su strukturirani tako da služe kao osnova za razvoj budućih primena.

# Baze podataka i SUBP

---

- **Sistem za upravljanje bazama podataka (SUBP)**
- **Database Management System (DBMS)**
  - softverski proizvod, namenjen da omogući izgradnju i korišćenje baza podataka
  - prilagodljiv i lak za korišćenje
  - Sadrži:
    - jezik za opis podataka
      - Data Definition Language - DDL
    - jezik za manipulisanje podacima
      - Data Manipulation Language - DML
    - upitni jezik
      - Query Language - QL



# Baze podataka i SUBP

---

- Jezgro SUBP
  - obezbeđenje fizičke organizacije podataka
  - rutine za upravljanje podacima
  - zaštita od neovlašćenog pristupa i od uništenja
  - obezbeđenje višekorisničkog režima rada
  - obezbeđenje distribuirane organizacije BP
  - obezbeđenje zadavanja šeme baze podataka
    - nad skupom obeležja ranijih datoteka formira se struktura šeme BP
    - nad šemom BP se kreira, koristi i ažurira baza podataka

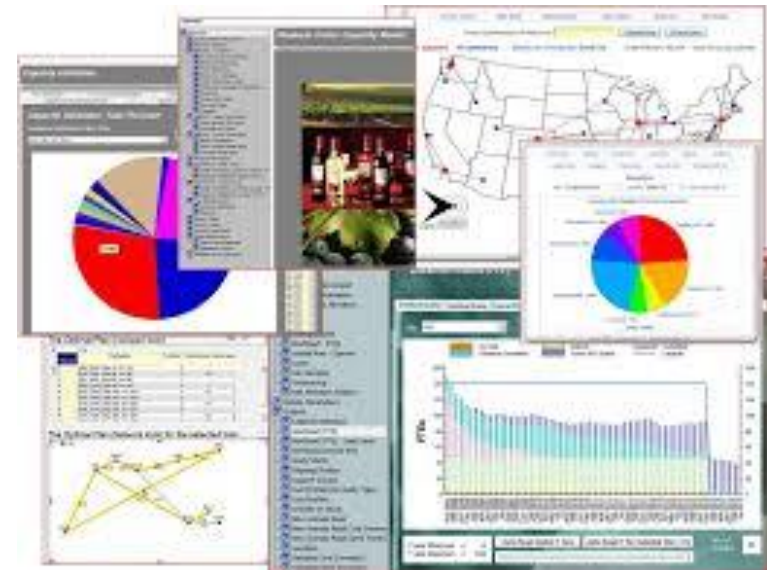
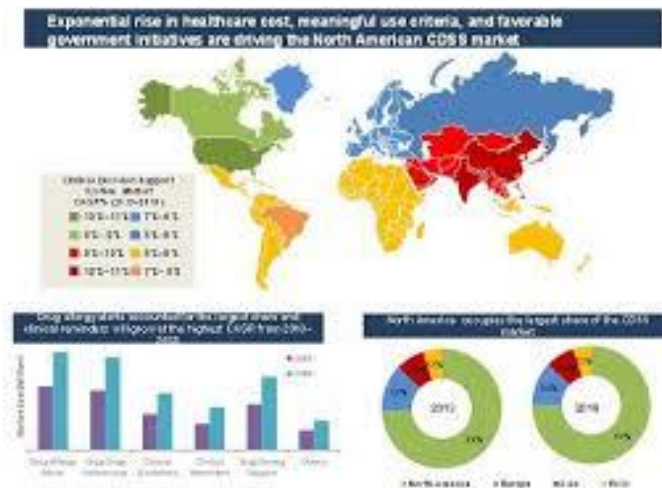
# Primena BP

- Bankovne aplikacije
  - tekući računi
  - transakcije
  - štednja



# Primena BP

- Sistemi za pomoć pri odlučivanju
  - podaci o poslovanju
  - izveštaji
  - višedimenzionalni pogledi
  - data mining



# Primena BP

- Transport
  - red vožnje/letenja
  - rezervacija i kupovina karata
  - odabir mesta za sedenje

THAI AIRWAYS FLIGHT SCHEDULE							
ROUTE: BANGKOK - PHUKET				ROUTE: PHUKET - BANGKOK			
FLIGHT NUMBER	DEPARTURE	ARRIVAL	FREQUENCY	FLIGHT NUMBER	DEPARTURE	ARRIVAL	FREQUENCY
TG 201	7.35 AM	9.00 AM	DAILY	TG 226	7.25 AM	8.50 AM	DAILY
TG 203	8.30 AM	9.55 AM	DAILY	TG 202	10.00 AM	11.30 AM	DAILY
TG 205	10.00 AM	11.30 AM	DAILY	TG 206	10.55 AM	12.25 AM	DAILY
TG 207	10.55 AM	12.15 AM	DAILY	TG 208	1.05 PM	2.30 PM	DAILY
TG 211	12.25 AM	1.45 PM	DAILY	TG 212	2.40 PM	4.05 PM	DAILY
TG 215	2.15 PM	3.35 PM	DAILY	TG 214	3.10 PM	4.35 PM	DAILY
TG 217	4.00 PM	5.20 PM	DAILY	TG 216	4.25 PM	5.50 PM	DAILY
TG 221	6.20 PM	7.40 PM	DAILY	TG 218	6.25 PM	7.50 PM	DAILY
TG 223	7.20 PM	8.40 PM	DAILY	TG 222	8.50 PM	10.15 PM	DAILY
TG 225	10.40 PM	11.59 PM	DAILY	TG 224	9.30 PM	10.55 PM	DAILY

中国南方航空

CHINA SOUTHERN

[Home](#)
[Bookings](#)
[Holiday](#)
[Special Offers](#)
[Travel Guide](#)
[Sky Pearl Club](#)
[Customer Service](#)

Online Check-in

Current position : Home > Bookings > Online Check-in > Select A Seat

[网上值机中文版]

Select A Seat

Please select your favorite seat, click on Up and Down arrows on the seat right for more seats.

ZHANG/WEIBING

Flight No. CZ368

Depart GUANG ZHOU

Arrive WU HAN

Class Y

Seat No. 42B

	A	B	C		H	J	K	
35								
36								
37								
38								
39								
40								
41								
42								

Front row

Back row

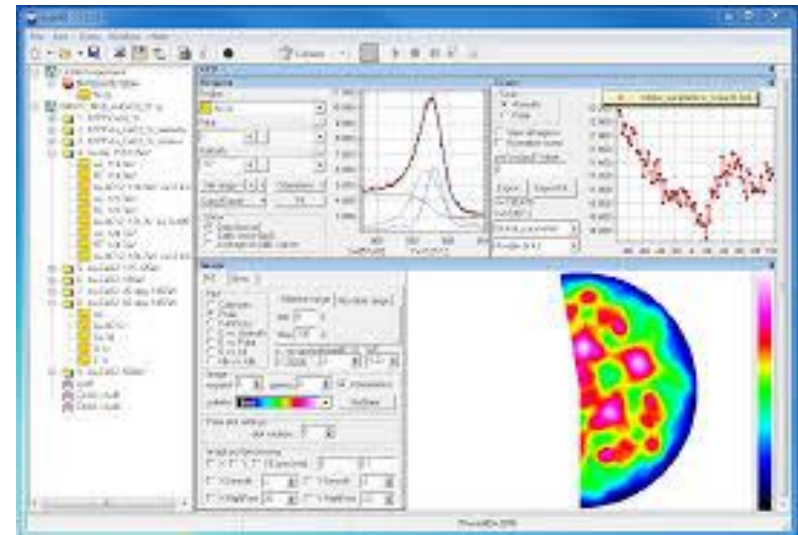
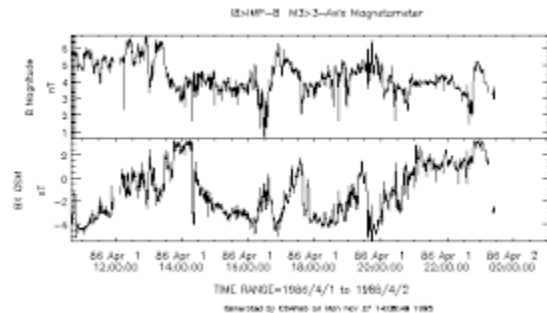
Available
  Selected
  Occupied
  Blocked
  Wing
  Emergency Exit

Confirm

Back

# Primena BP

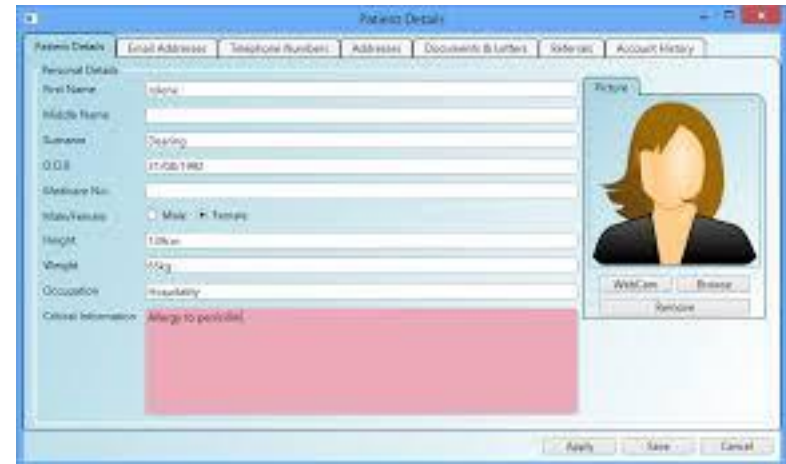
- Nauka
  - prikupljanje i obrada podataka





# Primena BP

- Medicina
  - podaci o pacijentima
  - istorija bolesti
  - pomoć pri dijagnostikovanju

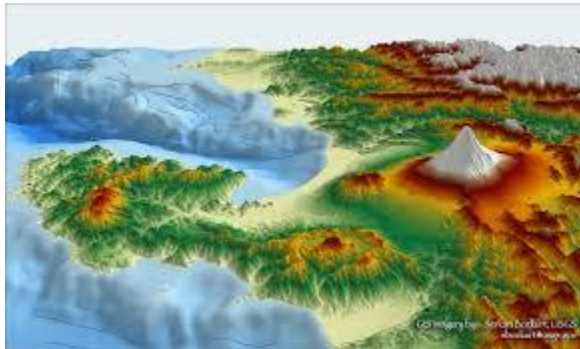


The screenshot shows a 'Patient Details' window with several tabs: Personal Details, Email Address, Telephone Numbers, Address, Documents & Letters, Radiology, and Account History. The 'Personal Details' tab is active, displaying fields for First Name (John), Middle Name, Surname (Doe), Date of Birth (17-Feb-1982), Sex (Male), Height (1.80m), Weight (75kg), and Occupation (Insurance). A 'Picture' section on the right shows a placeholder image of a woman with a 'WebCam' button and a 'Remove' button. At the bottom, there is a 'Critical Information' section with a red background and the text 'Allergic to penicillin'. Buttons for 'Apply', 'Save', and 'Cancel' are at the bottom right.



# Primena BP

- Geoinformacioni sistemi
  - zaštita životne sredine
  - upravljanje resursima



# Primena BP

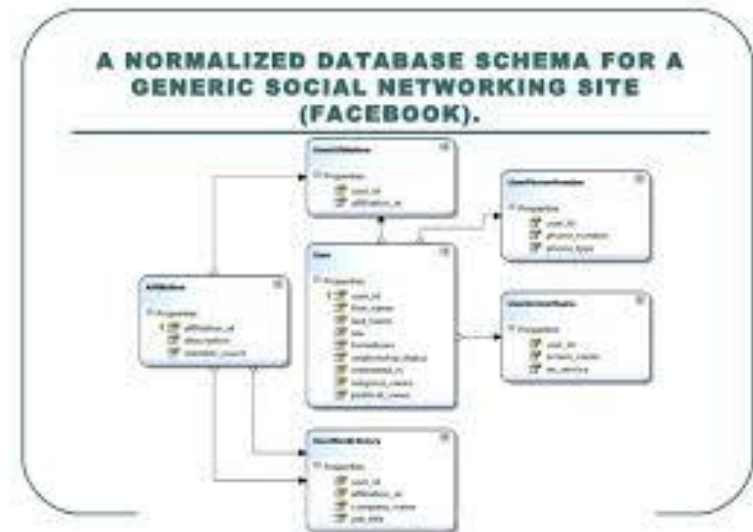
- Telekomunikacije
  - podaci o pozivima
  - telefonski računi
  - praćenje kvarova
  - model mreže





# Primena BP

- Društvene mreže
  - korisnički nalozi
  - poruke, objave
  - video klipovi, slike



# Mane SUBP

---

- Da li je SUBP rešenje bez mane?
  - Visoka cena
  - Komplikovana instalacija i administracija (ali mnogostruko manja nego u slučaju sistema baziranim na datotekama)
  - Proizvod opšte namene. Npr. nije namenjen za skladištenje vremenski serija ili pretragu po tekstu

# Sadržaj

---

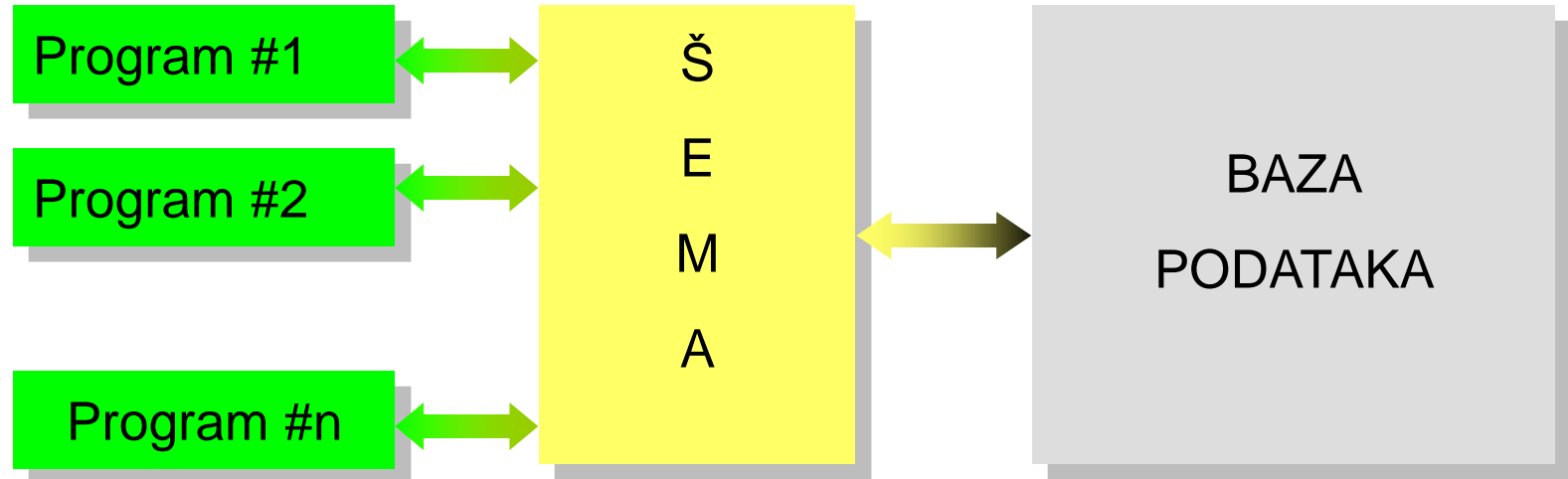
- Motivacija
- OM i klasična organizacija datoteka
- Baze podataka i SUBP
- Šema baze podataka
- Podšema / Eksterna šema
- Pogled
- Sistemi baza podataka

# Šema baze podataka

---

- Program koji koristi usluge SUBP
  - poznaje samo šemu BP, kao logičku kategoriju
  - nad šemom BP, koristi logičku strukturu podataka (LSP), saglasno konkretnom zadatku
  - ne sme da vodi računa o fizičkoj strukturi podataka (FSP)
    - koja, po pravilu, može biti veoma kompleksna
- Preslikavanje  $LSP \leftrightarrow FSP$  - zadatak SUBP
  - primeri potreba različitih programa za istim podacima
    - pristup saglasno rastućim vrednostima primarnog ključa
    - pristup saglasno vrednostima sekundarnog ključa
    - direktni pristup, saglasno zadatoj vrednosti primarnog ključa

# Šema baze podataka



# Šema baze podataka

---

- Efekti uvođenja koncepta šeme baze podataka
  - smanjenje zavisnosti programa i šeme BP od promena FSP
  - smanjenje redundantnosti
    - povećanje konzistentnosti podataka
  - uvođenje uloga
    - projektant baze podataka (šeme i FSP)
    - administrator baze podataka (DBA)

# Projektant baze podataka

---

- Definiše šemu BP
- Ima potpuno razumevanje činilaca i poslovnih procesa projektovanog IS
- Zaduženja:
  - Analiza zahteva
  - Identifikacija tipove entiteta i poveznika
  - Definisanje pomoćnih struktura

# Administrator baze podataka

---

- Upravlja aktivnostima SUBP
- Ima dobro razumevanje resursa i potreba organizacije
- Zaduženja:
  - Implementacija i održavanje šeme
  - Podešavanje parametara servera
  - Instalacija i održavanje SUBP
  - Dodela prava korisnicima SUBP i baza podataka
  - Praćenje i unapređivanje performansi servera



# Sadržaj

---

- Motivacija
- OM i klasična organizacija datoteka
- Baze podataka i SUBP
- Šema baze podataka
- Podšema / Eksterna šema
- Pogled
- Sistemi baza podataka

# Podšema / Eksterna šema

---

- Šema BP je, po pravilu,
  - kompleksna
  - podložna češćim modifikacijama
- Motivacija
  - promene šeme BP mogu izazivati česte i nepotrebne promene postojećih programa
    - kada bi program direktno koristio koncepte šeme BP
  - posledica: otežano održavanje softverske podrške IS
- Rešenje
  - uvođenje novog koncepta i novog sloja - podšema

# Podšema / Eksterna šema

---

- **Podšema ili eksterna šema**

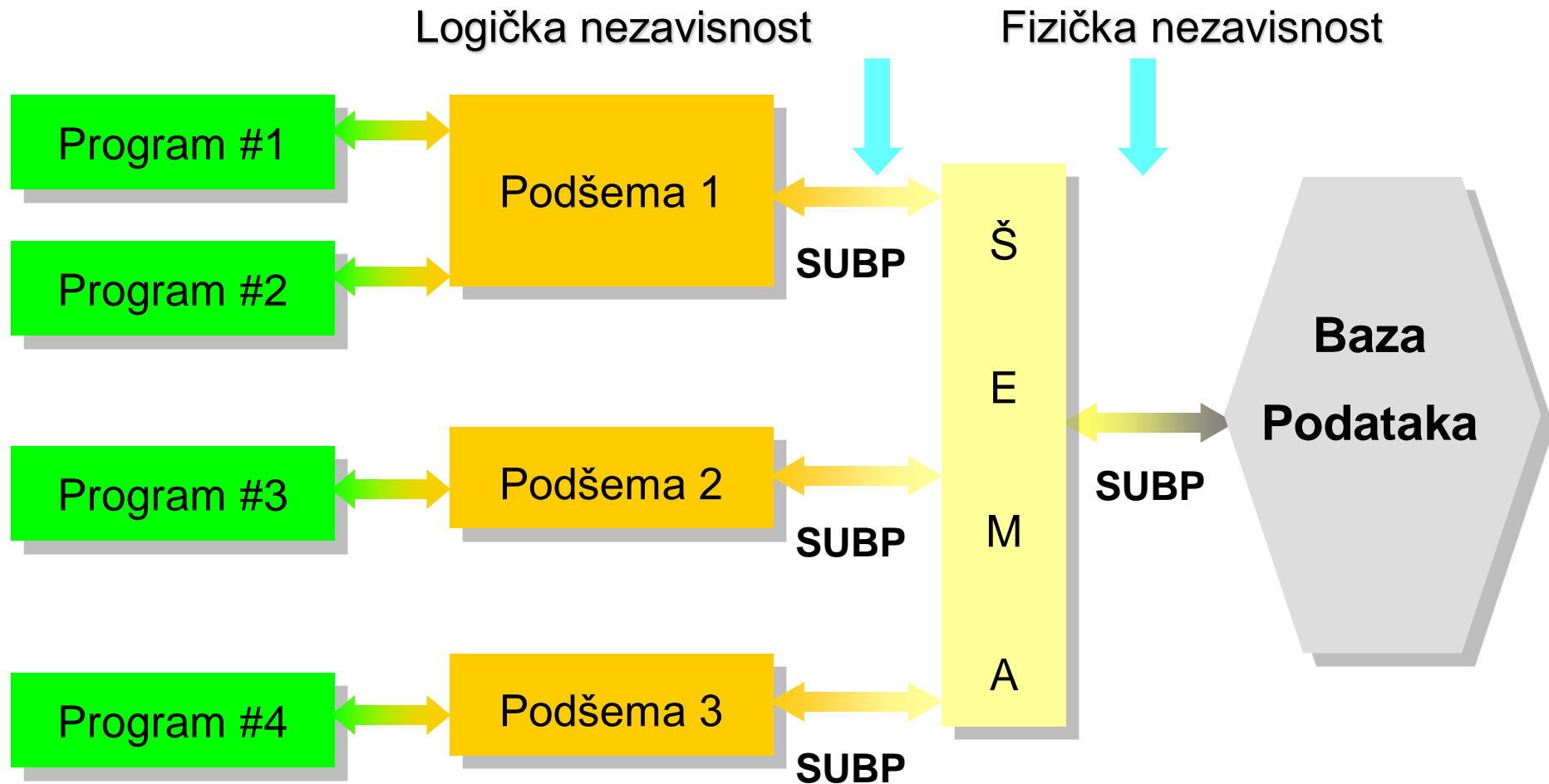
- logička struktura obeležja (LSO), dobijena na osnovu dela šeme BP
  - potrebna i dovoljna za realizaciju zadatka jednog, ili grupe sličnih transakcionih programa
    - sličnih sa stanovišta modelovanih procesa poslovanja i korisničkih zahteva
- model dela BP realnog sistema
  - za razliku od šeme BP koja predstavlja model cele BP realnog sistema
- projektuje se, kao i šema BP, u procesu razvoja IS
  - treba da egzistira kao projektantska specifikacija
- poželjna takva organizacija transakcionih programa
  - da koriste BP isključivo putem podšema

# Podšema / Eksterna šema

---

- Preslikavanje Podšema  $\leftrightarrow$  Šema BP
  - moguće i poželjno da bude zadatak SUBP
  - SUBP prevodi
    - zahtev programa, definisan s obzirom na koncepte podšeme, u zahtev definisan s obzirom na koncepte šeme BP
    - podatke strukturirane **s obzirom na koncepte šeme BP** u podatke strukturirane **s obzirom na koncepte podšeme** i obratno
  - alternativno, to može biti delimično ili u celosti zadatak samog transakcionog programa
    - danas, često, onog dela transakcionog programa koji upravlja logičkim strukturama podataka
      - nalazi se u višenivoovskim arhitekturama na tzv. “donjem” sloju
      - sloju modela podataka

# Podšema / Eksterna šema



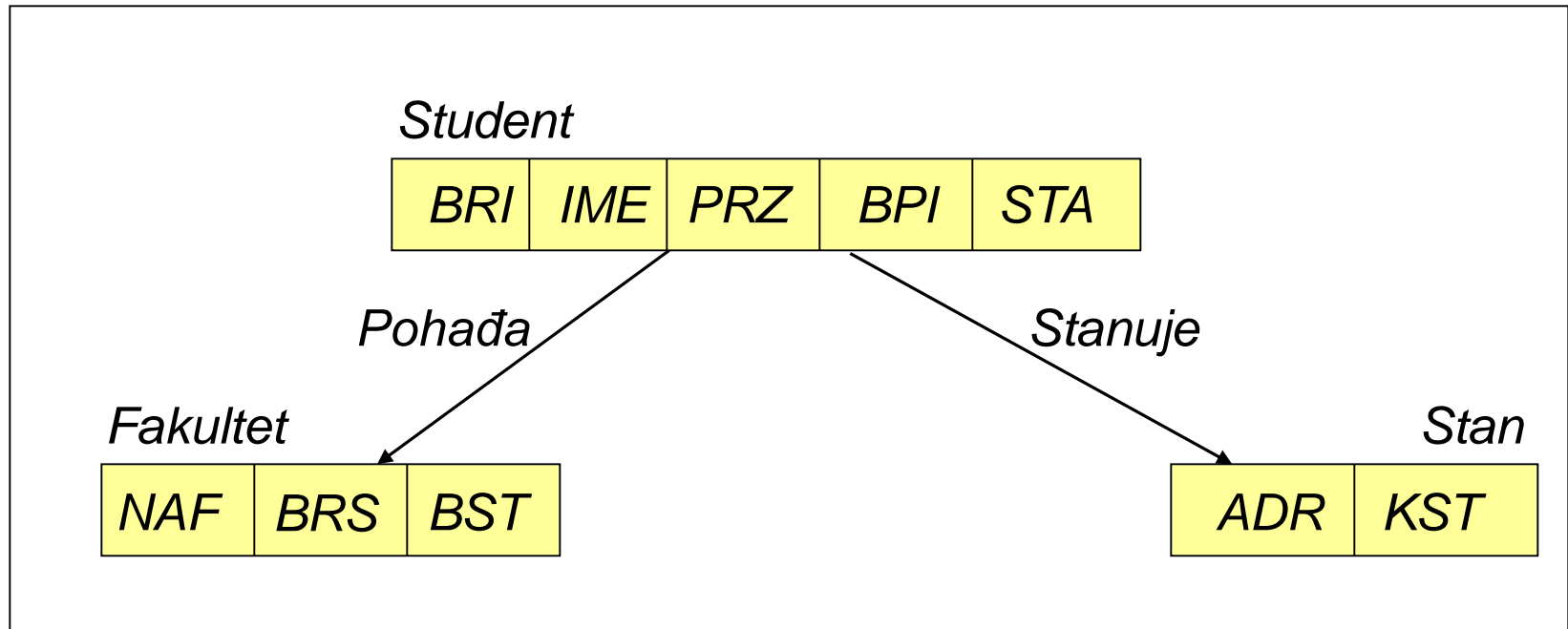
# Podšema / Eksterna šema

---

- Efekat uvođenja koncepta podšeme
  - **logička nezavisnost** programa od podataka
    - promene šeme ne izazivaju promene podšeme i programa
- Efekat uvođenja koncepta šeme BP
  - **fizička nezavisnost** programa od podataka
    - promene FSP ne izazivaju promene šeme, podšeme i programa
- Fizička i logička nezavisnost su uslovne, a ne apsolutne kategorije!

# Podšema / Eksterna šema

- Primer male šeme BP u mrežnom modelu podataka



# Podšema / Eksterna šema

## PODŠEMA #1

*Student\_Fakultet*

<i>BRI</i>	<i>IME</i>	<i>PRZ</i>	<i>BPI</i>	<i>NAF</i>
------------	------------	------------	------------	------------

## PODEŠEMA #2

*Student*

<i>BRI</i>	<i>IME</i>	<i>PRZ</i>	<i>STA</i>
------------	------------	------------	------------

*Stanuje*

*Stan*

<i>ADR</i>	<i>KST</i>
------------	------------



# Sadržaj

---

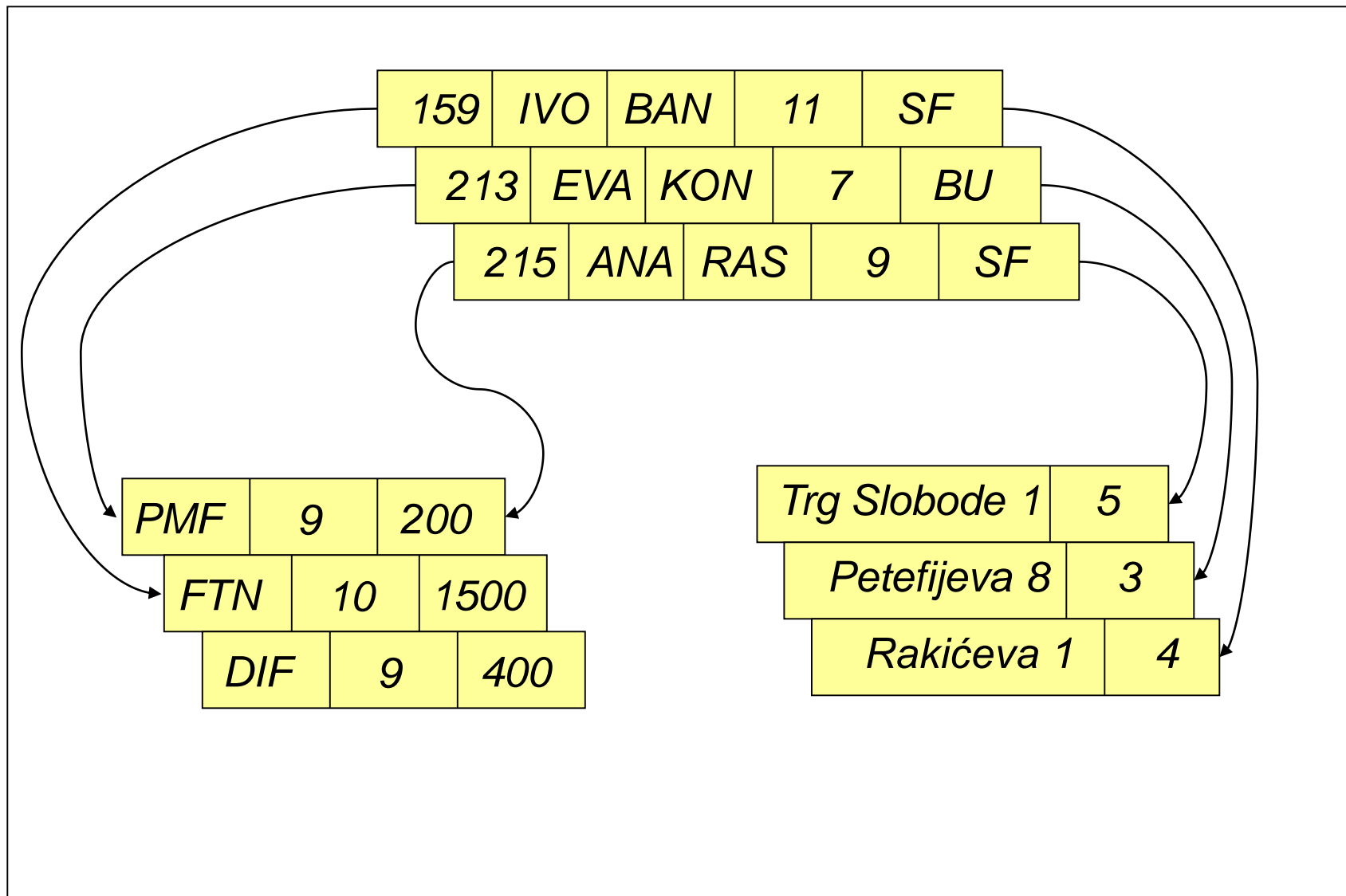
- Motivacija
- OM i klasična organizacija datoteka
- Baze podataka i SUBP
- Šema baze podataka
- Podšema / Eksterna šema
- Pogled
- Sistemi baza podataka

# Pogled

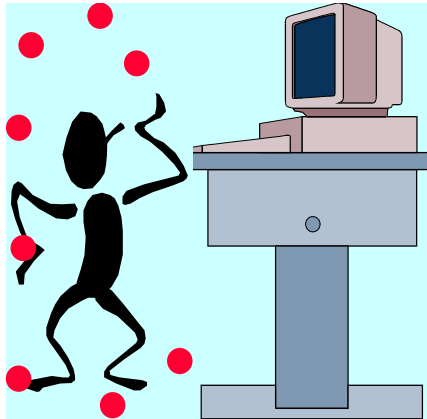
---

- Šema i podšema
  - modeli na nivou apstrakcije obeležja
- Globalni pogled i pogled
  - modeli na nivou apstrakcije podataka
- **Pogled**
  - pojava (LSP) nad podšemom
  - slika dela BP kako je vidi programer, ili korisnik
- Globalni pogled
  - pojava (LSP) nad šemom BP – baza podataka
  - slika stanja modelovanog dela sistema

# Globalni pogled

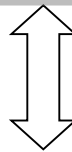


# Pogled



Program #n

PODŠEMA #1

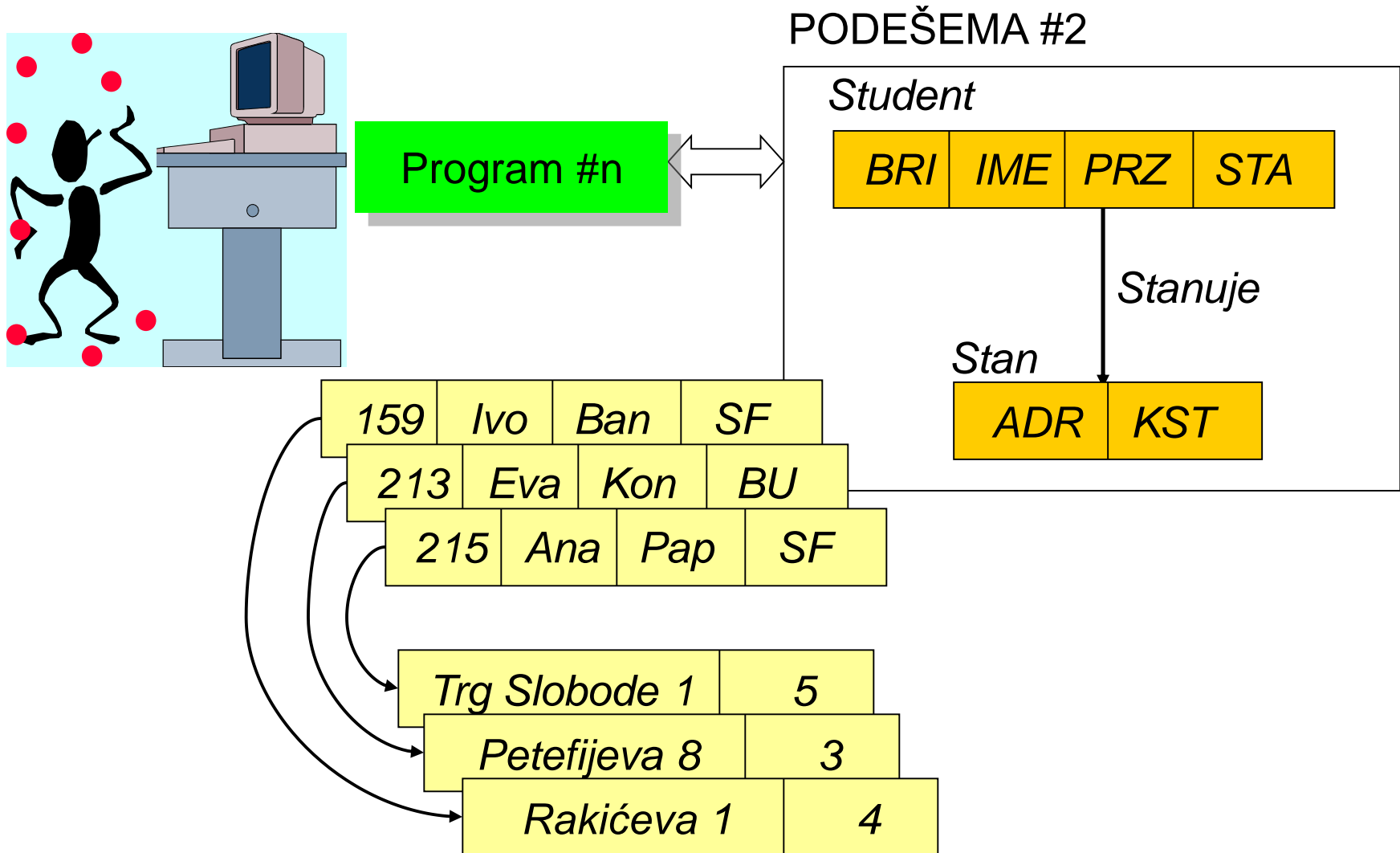


*Student\_Fakultet*

<i>BRI</i>	<i>IME</i>	<i>PRZ</i>	<i>BPI</i>	<i>NAF</i>
------------	------------	------------	------------	------------

159	IVO	BAN	11	FTN
213	EVA	KON	7	PMF
215	ANA	RAS	9	PMF

# Pogled



# Sadržaj

---

- Motivacija
- OM i klasična organizacija datoteka
- Baze podataka i SUBP
- Šema baze podataka
- Podšema / Eksterna šema
- Pogled
- Sistemi baza podataka

# Sistemi baza podataka

---

- Sistem baze podataka
  - obuhvata
    - bazu podataka
    - SUBP, sistemski softver i računare (servere) na kojima je BP kreirana
    - šemu baze podataka, implementiranu na SUBP
    - jezike i operacije za kreiranje, ažuriranje i korišćenje BP

# Sistemi baza podataka

## • Istorijat tehnološkog razvoja

<ul style="list-style-type: none"> <li>- Maš. jezici</li> <li>- HW – I gen.</li> </ul>	<ul style="list-style-type: none"> <li>- Asemb. jezici</li> <li>- HW – II gen.</li> <li>- OS – I gen.</li> <li>- Ekst.mem.(sek.)</li> <li>- U/I rutine – fiz.UI</li> </ul>	<ul style="list-style-type: none"> <li>- 3GL</li> <li>- HW – III gen.</li> <li>- OS – II gen.</li> <li>- Ekst.mem.(dir.)</li> <li>- U/I podsistem</li> <li>- Metode pristupa</li> </ul>	<ul style="list-style-type: none"> <li>- 3GL</li> <li>- HW – IV gen.</li> <li>- OS – III gen.</li> <li>- Sist. za upravlj. datotekama (usl. niskog i vis. nivoa)</li> </ul>	<ul style="list-style-type: none"> <li>- 4GL</li> <li>- HW - <math>\geq V</math></li> <li>- OS - <math>\geq IV</math></li> <li>-SUBP</li> <li>-CASE</li> </ul>
--	--	---	---	--

Napor potreban za pisanje dela programa posvećenog upravljanju podacima

**Mogućnost  
trajnog arhiviranja  
podataka**

**Operativna upotreba  
perzist. podataka  
Nezavis. od uređaja**

**Nezavisnost od  
fizičke strukture  
i organizac. datoteka**



# Sistemi baza podataka

---

- Rezime
  - opisane karakteristike sistema BP predstavljaju ciljeve kojima treba težiti
  - u kojoj meri će ti ciljevi biti ostvareni, zavisi od
    - projektanta BP
      - kvalitet / znanje raspoloživih ljudskih resursa
    - izabranih koncepata, metoda i tehnika projektovanja
      - kvalitet procesa razvoja sistema
    - karakteristika SUBP
      - kvalitet primenjenih informacionih tehnologija
  - razvoj postupaka za organizovanje i upravljanje podacima vodi ka:
    - povećanju produktivnosti razvojnog tima
    - izgradnji integrisanih IS

# Sadržaj

- Motivacija
- OM i klasična organizacija datoteka
- Baze podataka i SUBP
- Šema baze podataka
- Podšema / Eksterna šema
- Pogled
- Sistemi baza podataka

# Literatura

---

- Pavle Mogin, Ivan Luković: Principi baza podataka
  - Poglavlja: 3.1, 3.2, 3.3 i 3.4

# Pitanja i komentari

---



## Koncepcija baze podataka

---

*Istorijat razvoja postupaka za  
upravljanje podacima*