

UVOD U PROIZVODNE INFORMACIONE SISTEME

Razvitak naučne misli kroz istoriju može se podeliti u tri etape:

- Etapa primene metoda zapažanja.

(Metoda zapažanja je karakteristična za početne korake svakog naučnog pristupa nekom uočenom skupu problema, predmeta ili pojava. Sastoji se od zapažanja i zatim detaljnog opisivanja onoga što je zapaženo, bez pokušaja da se ustanove razlozi i zakonitosti koji stoje iza uočenih pojava.)

- Etapa primene analitičkih metoda.

(Analitički pristup karakterišu: 1. Svi emirijski fenomeni i predmeti se posmatraju skup nezavisnih elemenata koje treba analizirati da bi se pojava razumela u celini. 2. Odnos između elemenata tumači se po principu uzročno-posledične povezanosti. 3. Opisivanje predmeta i pojava vrši se krutim definicijama, pravilima ili strukturama uz eventualno korišćenje statističkih metoda, gde god nije moguće pojave razumeti bez uvođenja određene sučajnosti i verovatnoće nastajanja pojedinačnih događaja)

- Etapa primene sistemskih metoda (sistemski pristup).

(Sistemski pristup je baziran na posmatranju svih predmeta i pojava u njihovoj dinamičnosti i celovitosti u odnosu na okruženje (posmatraju se kao sistem). Praktična primena sistemskog pristupa je vezana za primenu računara. Sistemski pristup je postao podloga za razvoj novih naučnih disciplina kao što su: kibernetika, opšta teorije sistema, informatika itd.)

OPŠTA TEORIJA SISTEM

Opšta teorija sistema je naučna disciplina koja se bavi izučavanjem sistema i zakonitosti koje u njima vladaju.

Cilj opšte teorije sistema je da:

- služi kao jedinstveni metodološki i pojmovno-kategorijski okvir za sporazumevanje ljudi različitih specijalnosti, i
- da obuhvati i objedini osnovne pojmove koji važe u svim specifičnim sistemima i teorijama koje se njima bave.

Opšta teorija sistema nudi više definicija sistema.

Za naše potrebe na narednom slajdu se daje jedna od ponuđenih definicija sistema i njegove karakteristike.

DEFINICIJA SISTEMA

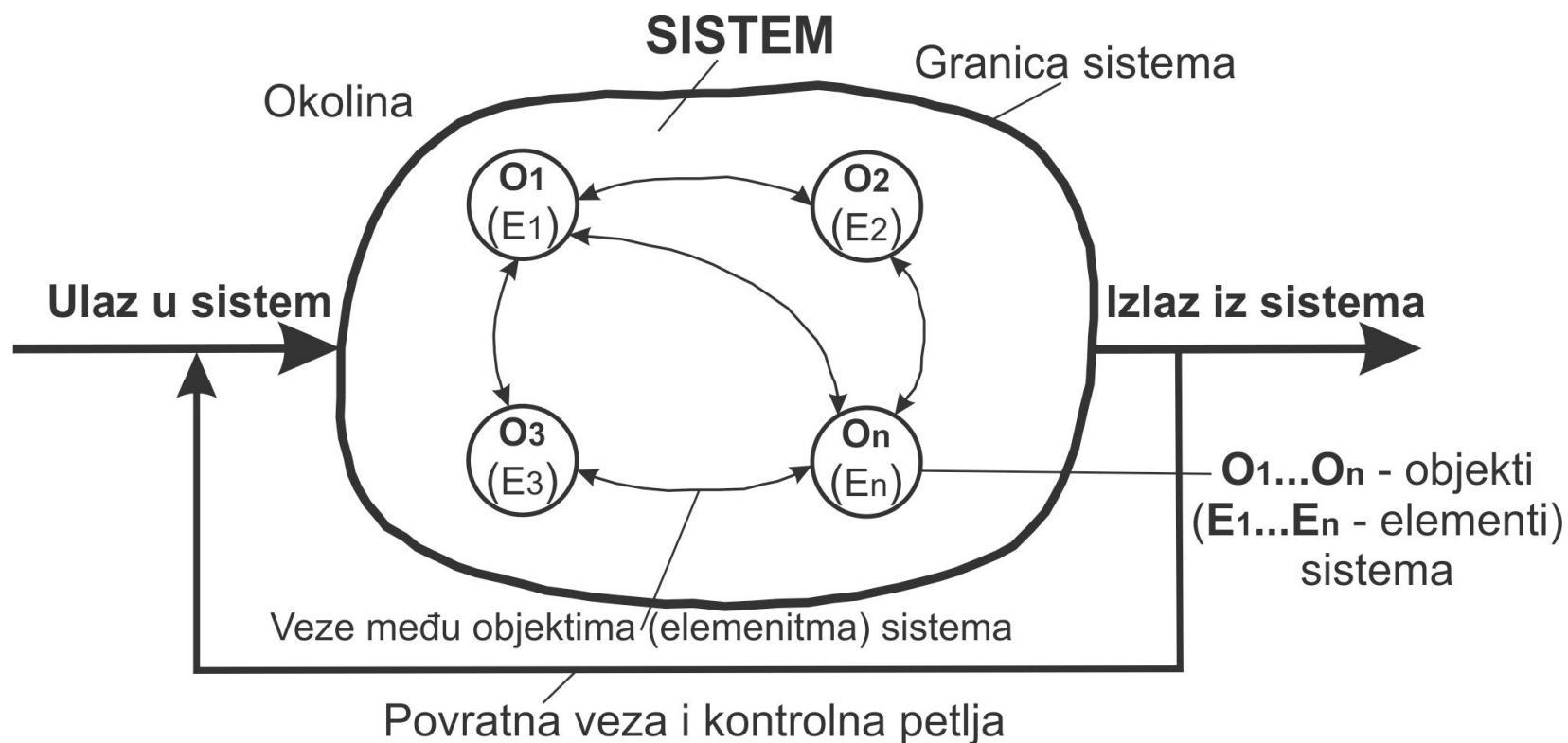
Sistem je funkcionalni skup objekata (elemenata) i njihovih veza. Svaki od objekata u sistemu poseduje niz svojstava (**atributa**).

Takođe i veze između pojedinih objekata (elemenata) sistema se mogu opisati nizom atributa. Sistem je u interakciji sa okolinom.

Uticaj okoline na sistem definiše se **ulazom** dok se uticaj sistema na okolinu definiše **izlazom**.

Veze sistema sa okolinom se može ostvariti razmenom materije, energije i informacija.

OPŠTA STRUKTURA SISTEMA

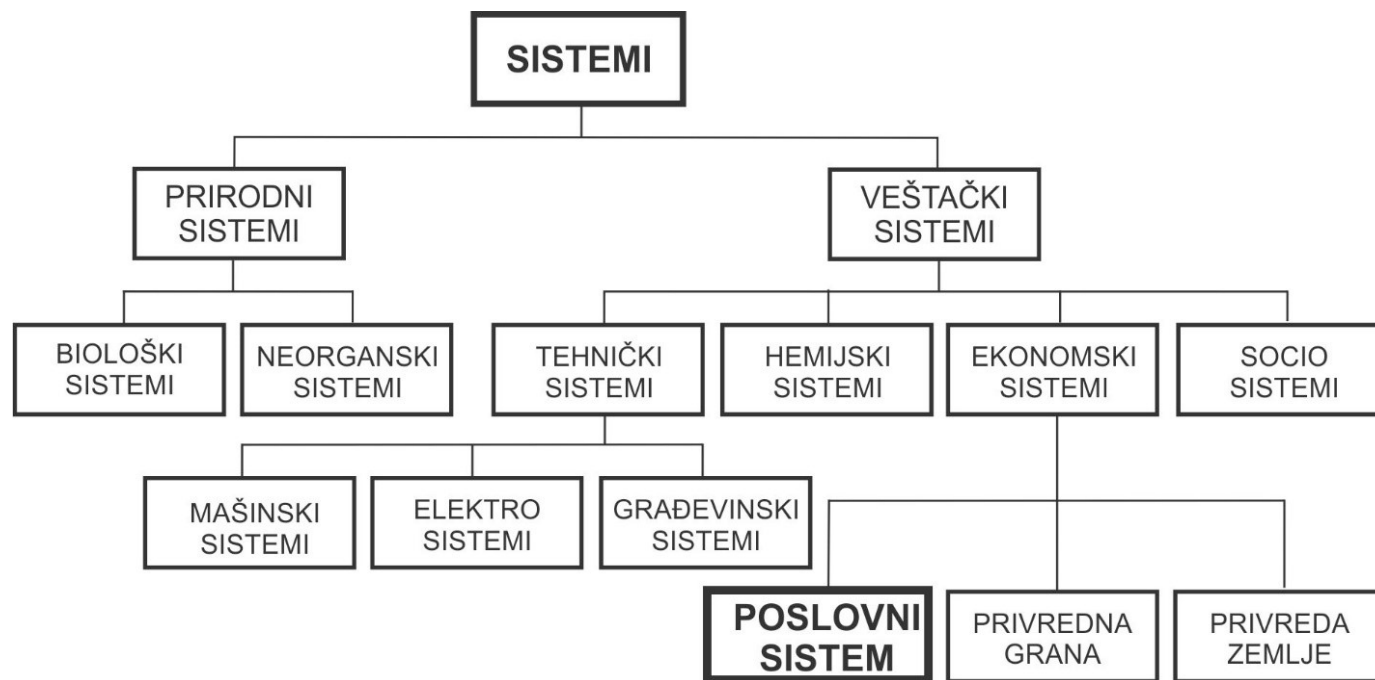


Svaki sistem može se dekomponovati na podsisteme. Istovremeno svaki sistem je deo nekog šireg sistema.

KLASIFIKACIJA SISTEMA

- Po načinu kreiranja:
 - prirodni
 - veštački
- Po stepenu apstrakcije:
 - apstraktni
 - realni
- Po promjenljivosti u vremenu:
 - statički
 - dinamički
- Po određenosti ponašanja:
 - deterministički
 - stohastički
- Po složenosti:
 - prosti
 - složeni
- Po povezanosti sa okolinom:
 - otvoreni
 - zatvoreni

HIJERARHIJA SISTEMA PREMA PRIRODI, FUNKCIJI I OBLASTI PRIMENE



Poslovni sistem (proizvodno poslovno preduzeće ili fabrika ili multinacionalna kompanija) su **kompleksni sistemi** sa velikim brojem komponenti koje, komplementarnom interakcijom obezbeđuju uspešno poslovanje i prestižan nivo na svetskom tržištu proizvoda, usluga, znanja i kapitala. Za njihovo funkcionisanje potrebna je primena visokih informaciono-komunikacionih tehnologija (digitalni opis elemenata poslovanja). Više detalja o ovim sistemima biće dato u nastava predavanja.

PODATAK I INFORMACIJA

- **Podatak** je kodirana predstava neke osobine, nekog koncepta, objekta, činjenice ili događaja u realnom sistemu.
- **Informacija** je iskaz odnosno znanje koje se može "izvući" iz datog podatka ili datih podataka. **Pojam "vest" se najčešće koristi za sinonim pojma "informacija"**.
- Podatak je međutim objektivna kategorija dok je informacija subjektivna kategorija odnosno zavisi od načina na koji neki pojedinac tumači ili koristi predstavljene podatke.
- *Ovde ćemo podrazumevati da se podaci mogu predstaviti na računaru na prihvatljiv način, odnosno da se mogu kvantifikovati brojevima, karakterima, datumima itd.*
- **Informacioni sistem (IS)** se može definisati kao sistem u kome se veze između objekata i veze sistema sa okolinom obavljaju razmenom informacija. *Informacioni sistem nemora biti zasnovan na računaru, ali će biti ovde razmatrani u nastavku samo IS realizovani na računarima.*

TIPOVI INFORMACIONIH SISTEMA

- **Upravljački informacijski sistem - UIS** (Management Information Systems - MIS) je sistem koji prenosi, čuva i obrađuje podatke potrebne za upravljanje.
- **Sistem za podršku odlučivanju** (Decision Support Systems) je informacijski sistem koji potpomaže donošenje odluka.

TIPOVI INFORMACIONIH SISTEMA

Po načinu primene IS možemo grubo podeliti na one koji vrše *samo skladištenje i pronalaženje podataka* one koji vrše *obradu*. U nastavku će se razmatrati sistem koji sa svojim podsistemima može da realizuje *obe operacije*.

Prema sposobnosti povezivanja različitih podsistema razlikujemo:

- *integrisane i*
- *neintegrisane informacione sisteme.*

Prema stepenu primene informacione tehnologije razlikujemo:

- *centralizovane,*
- *autonomne i*
- *distribuirane informacione sisteme.*

AUTOMATIZACIJA KANCELARIJSKOG POSLOVANJA

- Komunikacija unutar organizacije (Electronic Mail);
- Komunikacija između organizacija (Electronic Data Interchange - EDI);
- Planiranje radnog vremena (Time management&Planning);
- Upravljanje projektima (Project management);
- Obrada dokumenata (Graphics Editor);
- Arhiviranje i pretraživanje dokumenata (Document Library);
- Priprema upravljačkih odluka;
- Memorisanje i pretraživanje poslovnih podataka;
- Priprema i realizacija poslovnih sastanaka.

RAZVOJ INFORMACIONOG SISTEMA

Projekat informacioni sistemi po pravilu realizuje tim stručnjaka koji nisu sastavni deo organizacije u kojoj se implementira. Očekuje se visok stepen saradnje organizacije i projektanata. Da bi se obavio uspešan razvoj informacionog sistema neophodno je:

- Obezbediti podršku rukovodstva firme;
- Uključiti korisnike u sve faze razvoja;
- Koristiti dokazane metodologije razvoja informacionog sistema;
- Jasno definisanje ciljeva i zadataka sistema;
- Usredsrediti se na najvažnije probleme i eventualn povoljnosti;
- Jednostavan i adekvatan dizajn;
- Dobar program obuke uključenih osoba;
- Adekvatan plan uvođenja nakon završetka projekta;
- Dobro definisan i organizovan program održavanja.

FAZE U RAZVOJU IS

- Planiranje razvoja;
- Analiza i dizajn;
- Implementacija;
- Funkcionisanje i održavanje;
- Vrednovanje i kontrola.

Napomena:

Faze se mogu deliti na podfaze ili aktivnosti.

PLANIRANJE RAZVOJA

1. Identifikacija i definisanje problema

- Problemi sa postojećim informacionim sistemom;
- Želja za iskorišćavanjem novih šansi i pogodnosti;
- Rastuća konkurencija;
- Potreba za efikasnijim korišćenjem informacija;
- Reorganizacija u organizaciji (spajanje, razdvajanje, i slično);
- Razvoj same organizacije;
- Promene na tržištu i okruženju.

PLANIRANJE RAZVOJA

- 2. Analiza postojećeg informacionog sistema.**
- 3. Treba odgovoriti na pitanje šta se promjenama u informacionom sistemu želi postići.**
- 4. Treba se odlučiti na uvođenje potpuno novog ili modifikovanog informacionog Sistema.**

PLANIRANJE RAZVOJA IS

5. **Projektovanje logičke strukture baze podataka.** *Ovaj deo se realizuje na nivou entiteta (objekata) i veza među njima (ERWIN).*
6. **Definisanje koncepcije tehničke podrške.** (Na osnovu ciljeva koji se žele postići IS, veličine organizacije itd.)
7. **Definisanje modela kadrovske podrške.**
8. **Specifikacija potrebnih ulaganja.**

PLANIRANJE RAZVOJA IS

9. **Analiza izvodljivosti projekta.**
10. **Utvrdjivanje prioriteta i izbor projektanta.**
11. **Planiranje realizacije projekta.**
12. **Prihvatanje ili odbijanje plana razvoja.**

ANALIZA I DIZAJN IS

Kod dizajna sistema postoje dve važne faze:

- Logičko projektovanje i
- Fizičko projektovanje.

Logičko projektovanje obuhvata modeliranje podataka i modeliranje procesa. Rezultat logičkog projektovanja mora biti razumljiv za korisnika i projektanta i treba da bude nezavisan od implementacije.

Fizičko projektovanje je prevođenje logičkog projekta na konkretnu opremu, konkretne softvere koji uključuju konkretne baze podataka.

IMPLEMENTACIJA IS

Faze u implementaciji IS su:

- Najava (cilj ove faze je uklanjanje straha zaposlenih i savlađivanje otpora koji postoje);
- Organizacija kadrova informacionog sistema;
- Izbor i instalacija opreme;
- Izrada ili nabavka softvera;
- Formiranje baze podataka (jedna od najvažnijih faza u razvoju informacionog sistema);
- Obuka korisnika
- Zamena starog sistema novim

ODRŽAVANJE IS

- Hardver:
 - redovne aktivnosti, provere, čišćenje, podešavanje i zamena potrošnog materijala;
 - zamena delova i uređaja;
 - instalacija novih uređaja.
- Softver:
 - prepravke koje ispunjavaju nove zahteve korisnika;
 - otklanjanje grešaka u ranijim verzijama;
 - prepravke softvera uzrokovane promenama hardvera ili sistemskog softvera.

VREDNOVANJE I KONTROLA IS

- Uticaj informacionog sistema na poslovanje u smislu ubrzanja pojedinih aktivnosti;
- Koliko sam informacioni sistem može da odradi brzo pojedine operacije i da li može brže;
- Da li je pogodan za korišćenje;
- Da li je održavanje skupo i na koji je način rešeno;
- itd.

ALATI ZA PROJEKTOVANJE IS

Celokupna današnja informaciona tehnologija je usmerena ka podršci onima koji se bave razvojem informacionih sistema.

Ta tehnologija se naziva sistemski inženjering pomoću računara (Computer Aided Software Design (CASE)).

CASE alati su programi (softveri) koji automatizuju i podržavaju jednu ili više faza životnog ciklusa razvoja sistema.

CASE (Computer Aided Software Design) alati omogućavaju:

- Povećanje kvaliteta dobijenog projekta
- Povećanje produktivnosti projektanata
- Skraćivanje vremena izrade projekta

Podela CASE alata

- Alati za modeliranje struktura podataka;
- Alati za izradu dijagrama toka podataka i hijerarhije modula;
- Alati za izradu prototipa korisničkog interfejsa; i
- Alati za generisanje koda.

Reprezentativni primeri CASE alata

Reprezentativni primeri CASE alata su:

- BPWin,
- ERWin,
- *System Architect*,
- *Rational ROSE UML*,
- *DataArchitect*,
- *Oracle Designer*,
- *SmartDraw*,
- *Power Designer* i mnogi drugi.

KLASIČNI METODI PROJEKTOVANJA IS

CASE tehnologija automatizuje celokupnu metodologiju razvoja sistema. Najpoznatiji klasični metodi projektovanja IS su:

- BSP (Business Systems Planning) metod;
- SDM (System Development Method) firme “IBM”;
- SADT (System Analysis Design Technics) metod;
- SSA (Structured System Analysis) metod;
- itd.

Business systems planning (BSP)

IBM je razvio metodu BSP kojom se planiranje i analiza obavljaju od vrha ka dnu dok se projektovanje i uvođenje obavljaju od dna ka vrhu.

BSP metodologijom se:

- Definiše opšta arhitektura informacionog sistema na osnovu poslovnih procesa kao relativno najstabilnija komponenta realnog sistema (u odnosu na organizacionu strukturu, način upravljanja i odlučivanja koji su brzo promenljivi).
- Modeliraju se podaci kao osnova informacionog sistema, koji tretira podatke kao posebne resurse u sistemu.

Ciljevi BSP metode

1. Uključiti najviše rukovodstvo u izradu plana razvoja informacionog sistema.
2. Postići da ciljevi razvoja informacionog sistema podržavaju ciljeve poslovanja.
3. Razumevanje poslovanja sa stanovišta najvišeg rukovodstva.
4. Pristup odozgo na dole u planiranju IS, implementacija odozdo na gore.
5. Kreiranje plana izgradnje integrisane arhitekture.
6. Aktivno rukovođenje resursima IS.
7. Podaci su osnovni resurs poslovnog sistema.

SADT (System Analysis Design Technics) metod

Prihvatajući SADT tehniku, avijacija SAD razvila je SADT kao deo ICAM (Integrated Computer Aided Manufacturing) programa tokom kasnih 70-ih, koji je dobio naziv IDEF tehnika (Integration DEFINition).

Učesnici u izgradnji ICAM programa uvideli su sve prednosti korišćenja IDEF tehnike, jer tekstualni opis ne predstavlja efikasan način za dokumentovanje procesa i podataka.

U ranim 90-im, IDEF Users Group, u kooperaciji sa National Institutes for Standards and Technology (NIST), preduzela je određene postupke za stvaranje standarda *IDEF0* za funkcionalno modeliranje i *IDEF1X (eXtend)*, kao tehniku za informaciono modeliranje (semantičko modeliranje podataka) koje je prihvatila i International Organization of Standards (ISO).

PROIZVODNI INFORMACIONI SISTEMI

- Šta je poslovni sistem
- Šta je informacioni sistem (IS)
- Delovi informacionog sistema
- Funkcionalni podsistemi IS-a
- Integralni informacioni sistem

POSLOVNI SISTEM

Svaki poslovni sistem se karakteriše

- ulaznim tokovima (materijalnim, finansijskim, informacionim, ..)
- unutar kojeg se odvijaju poslovni procesi u kojima su angažovani raspoloživi resursi
- izlaznim tokovima u kojima su nove vrednosti (proizvodi, usluge)
- preko svojih ulaznih i izlaznih tokova povezan je sa drugim sistemima iz okruženja

PODACI U POSLOVANJU

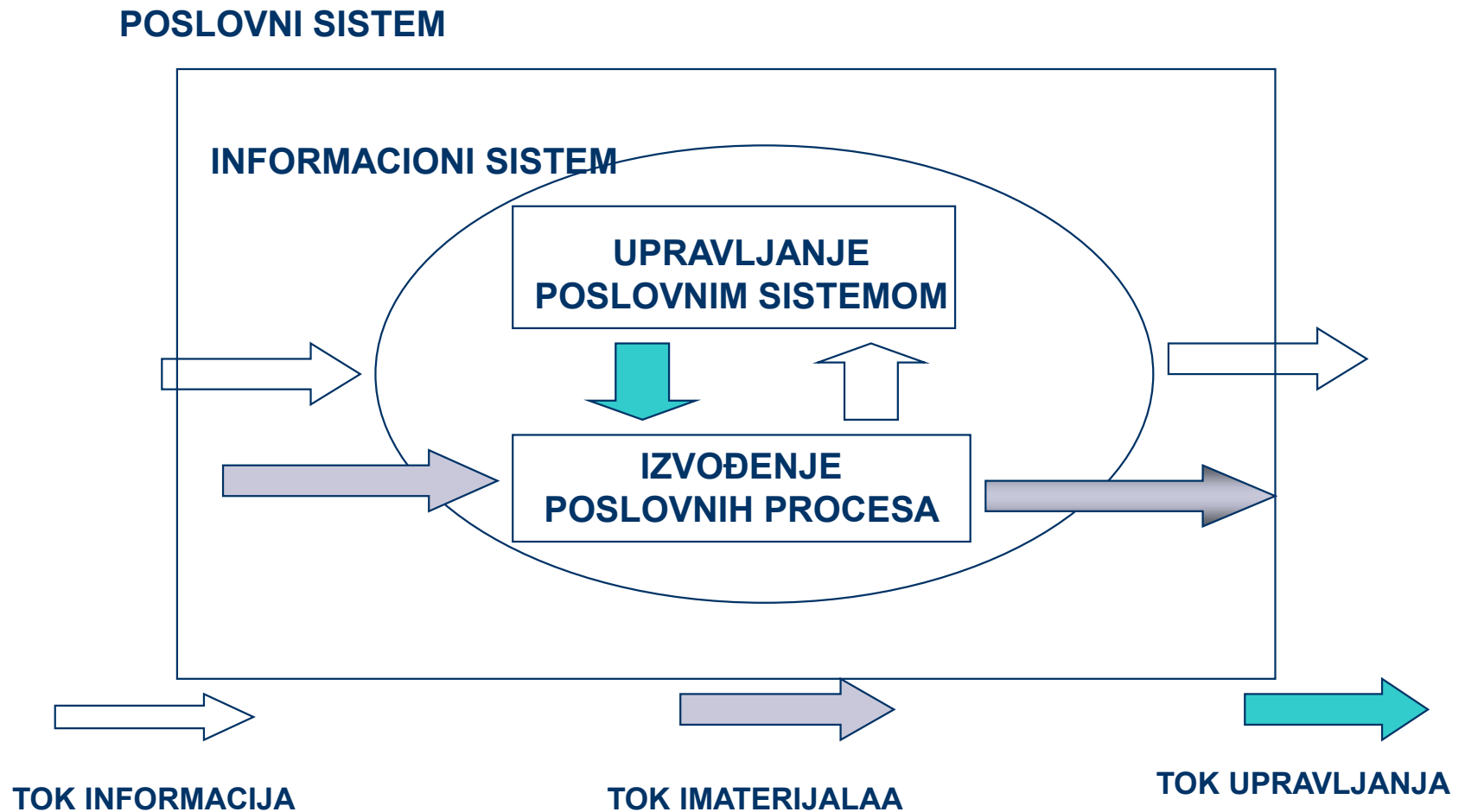
- Bez podataka ne bi bilo moguće poslovati
 - podaci prate svaku poslovnu aktivnost
- Bez informacija bilo bi teško donositi odluke
 - informacije se baziraju na podacima
(npr. izvještaj o zatvoreni radnim nalogima, termin plan proizvodnje i td.)

STRUKTURA POSLOVNOG SISTEMA

Poslovni sistem sadrži:

- izvršni podsistem koji izvršava poslove (izvršava poslovne procese),
- upravljački podsistem koji upravlja poslovnim sistemom i
- **informacioni podsistem** koji manipuliše podacima i informacijama.

POSLOVNI SISTEM - INFORMACIONI SISTEM



ERP sistemi

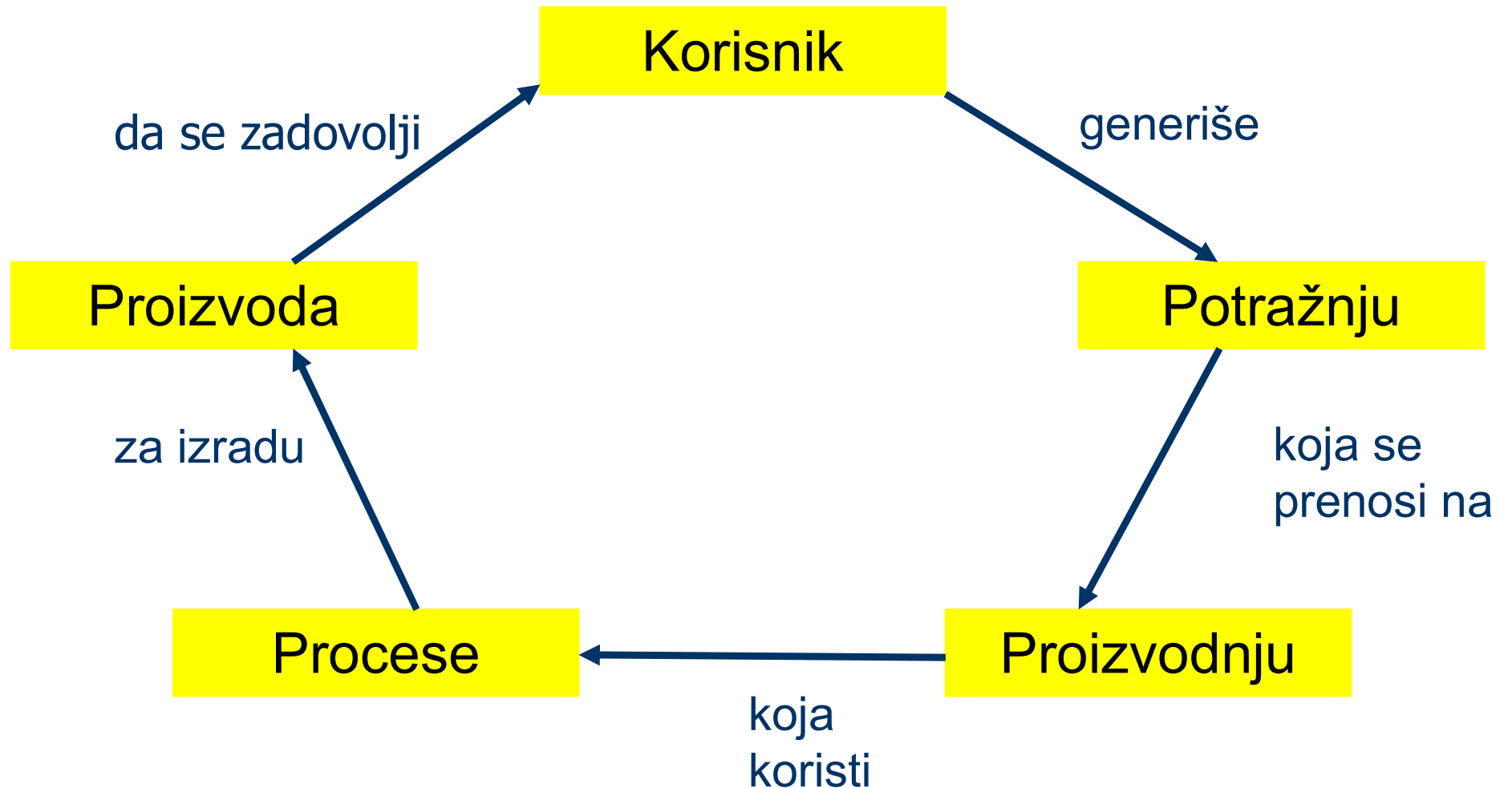
Poslovni IS imaju za cilj da skladište poslovne informacije samo jedanput, u formi u kojoj se omogućava pristup od strane više različitih korisnika, a u cilju donošenja različitih tipova odluka.

Za većinu ljudi termin poslovni informacijski sistem podrazumeva ERP softver.

ERP (*Enterprise Resource Planning*) sistem je poslovni softver koji omogućava organizacijama da automatizuju i integrišu glavne poslovne procese, dele opšte podatke i pristupaju informacijama u realnom okruženju.

Na domaćem tržištu, najzastupljeniji ERP sistemi su: **SAP R/3, Oracle Applications, Microsoft Dynamics NAV, HansaWorld, Datalab PantheonTM** i mnogi drugi.

PROIZVODNI SISTEM – LANAC SNABDEVANJA



INFORMACIONI SISTEM - POSLOVNI SISTEM

- U poslovni sistem ulaze i izlaze različiti tokovi (materijal, kapital, informacije ...).
- Informacioni sistem obuhvata informacije, obrađuje ih i prezentira poslovnom sistemu ili njegovoj okolini.
- Omogućava poslovnom sistemu da komunicira unutar sebe i sa svojom okolinom.

ŠTA JE INFORMACIONI SISTEM

- Informacioni sistem je sistem koji:
 - prikuplja,
 - memoriše,
 - obrađuje i
 - isporučuje informacije važne za poslovni sistem i društvo, tako da budu dostupne i upotrebljive svakome kome su potrebne

CILJ INFORMACIONOG SISTEMA

Da osigura poslovnom sistemu podatke i informacije potrebne:

- Izvršnom podsistemu za izvođenje poslovnog procesa
- Upravljačkom podsistemu za upravljanje poslovnim sistemom
- Podsistemu za koordinaciju i komunikaciju unutar poslovnog sistema i komunikaciju sa okruženjem.

Komponente IS-a

- Materijalno tehničke - Hardware
- Nematerijalne – Software
 - Sistemski
 - Aplikativni
- Podaci
- Ljudski resursi - Lifeware
- Komunikaciona tehnologija - Netware
- Organizacija - Orgware
 - Usklađuje komponente

UPRAVLJANJE I POSLOVNI SISTEM

- Zadatak informacionog sistema je da obezbedi informacije za upravljanje poslovnim sistemom.
- Upravljanje znači donositi odluke, odlučivati.
- Uspešno odlučivanje zahteva kvalitetne informacije (tačne, potpune, adekvatne, pravovremene).

Upravljanje poslovnim sistemom

- **STRATEGIJA** usuglašava zahteve i mogućnosti (definše dugoročne ciljeve organizacije i stvara dugoročne planove - priprema za budućnost)
- **TAKTIKA** upoređuje ciljeve s aktualnim stanjem i stvara planove za ostvarenje tih ciljeva (srednjeročno planiranje)
- **OPERATIVA** realizuje taktičke planove kroz poslovne aktivnosti

IZVORI PODATAKA ZA ODLUČIVANJE

- Podaci nastali u poslovnom procesu (interni podaci, npr izvestaj o radnim nalogima).
- Podaci nastali van poslovnog sistema (eksterni podaci, npr. stanje na tržištu).
- Podaci nastali u postupku odlučivanja (npr. doneti planovi proizvodnje).

UPRAVLJANJE POSLOVNIM SISTEMOM

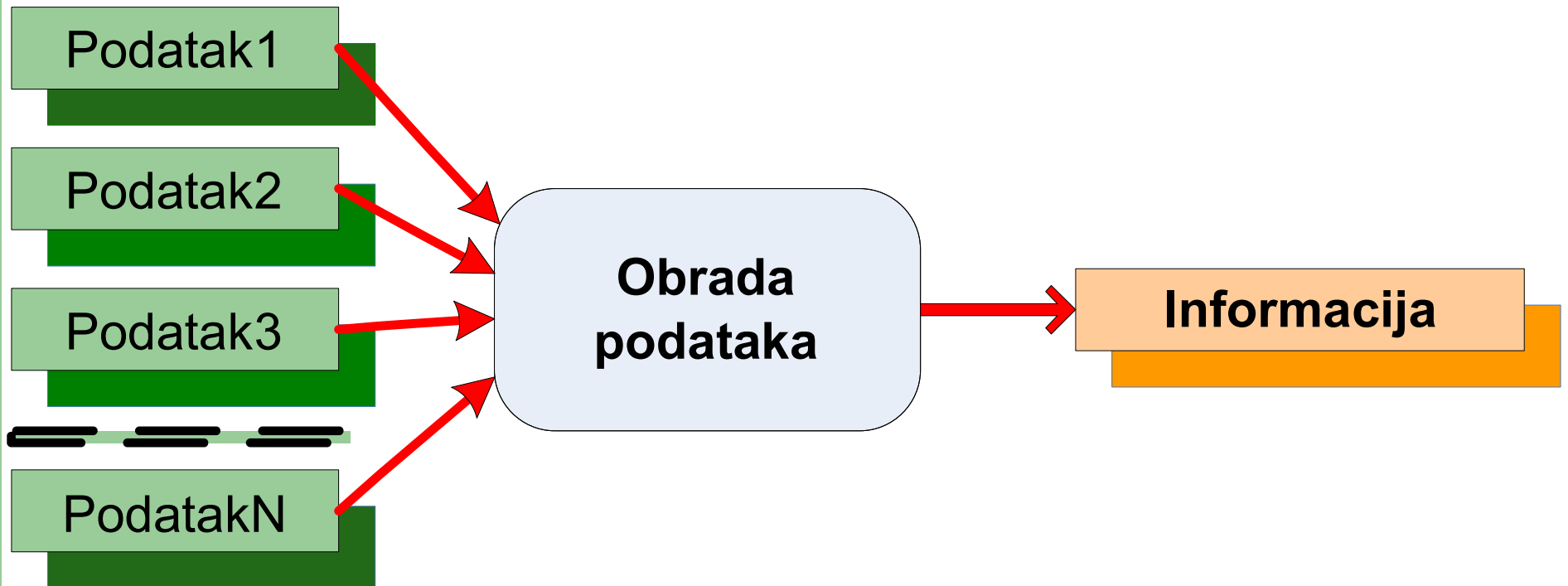
Nivoi upravljanja:

- Operativno upravljanje - dnevne aktivnosti, osnovne informacije.
- Taktičko upravljanje – delimično sintetizovane informacije
- Strateško upravljanje – visoko sintetizovane informacije

INTEGRALNI INFORMACIONI SISTEM

- Sistem za obradu transakcija.
- Sistem za podršku odlučivanju.
- Sistem za komunikaciju.

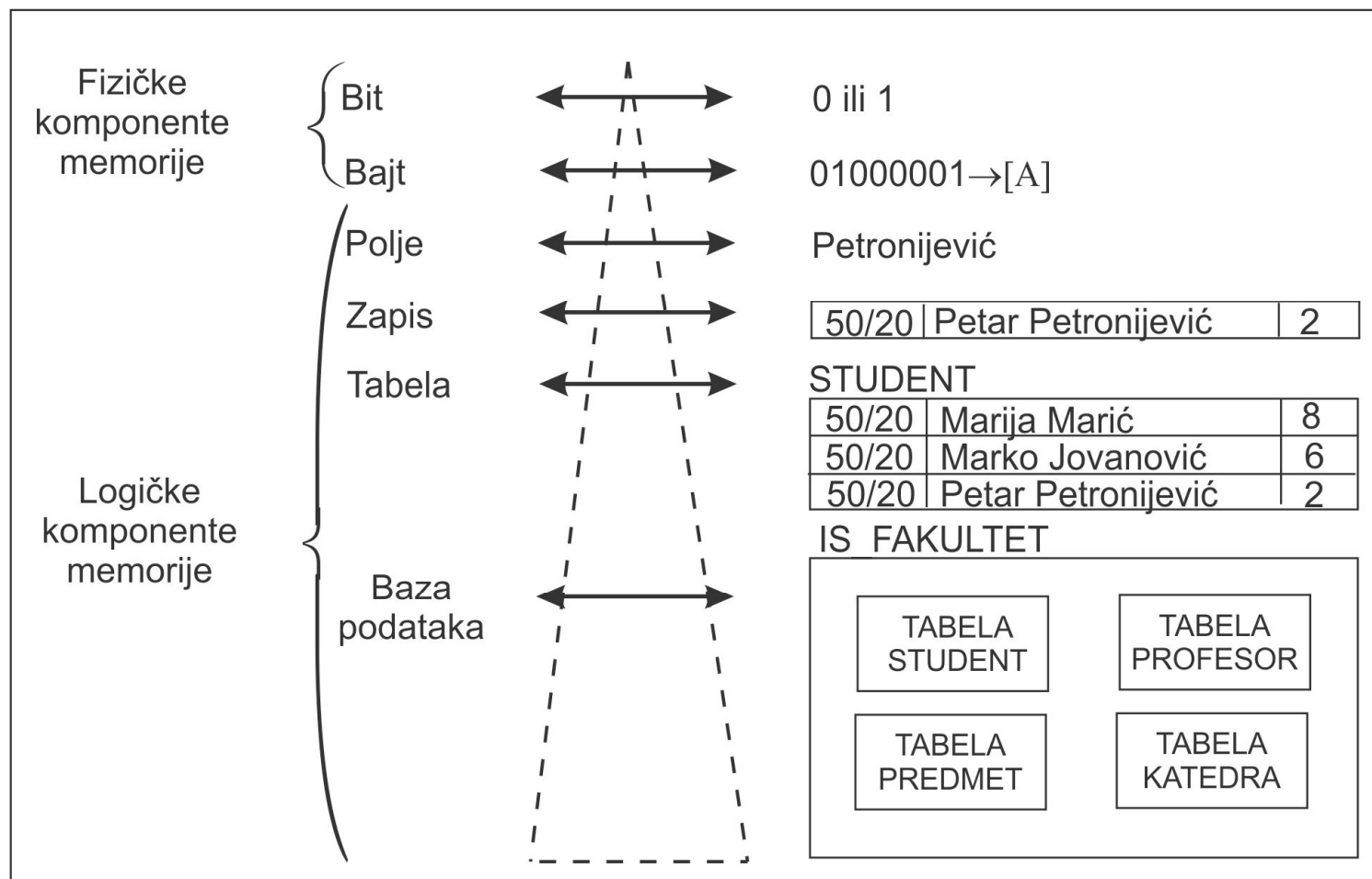
Informacija



Hijerarhijska struktura podataka

- Baza podataka
- Datoteka
- Slog
- Polje
- Bajt
- Bit

Primer: Hijerarhijska struktura podataka



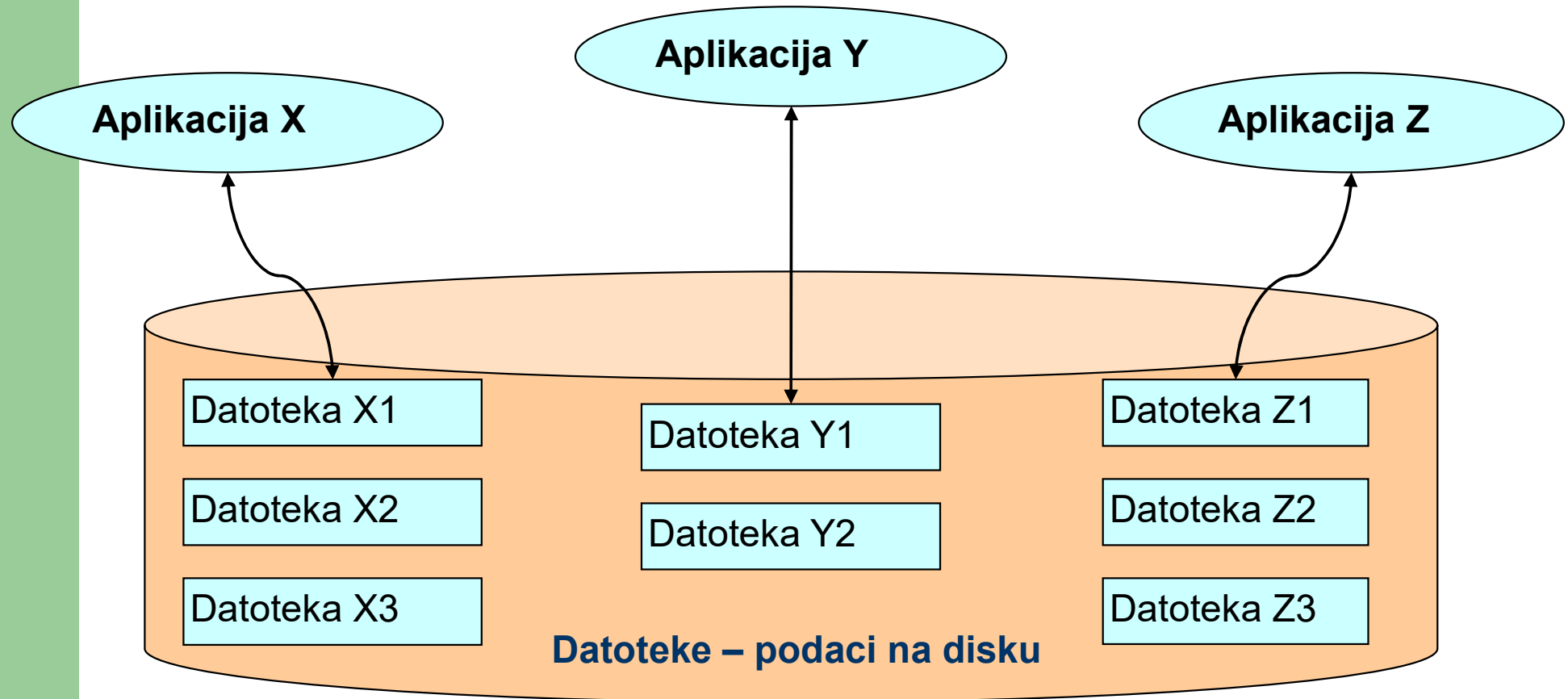
Organizacija podataka

- način prikupljanja i prihvatanja podataka
- obrada (ažuriranje) podataka
- način memorisanja podataka
- način prezentacije podataka preko izlaznih jedinica
 - sekvencijalna organizacija
 - indeks-sekvencijalna organizacija
 - direktna organizacija

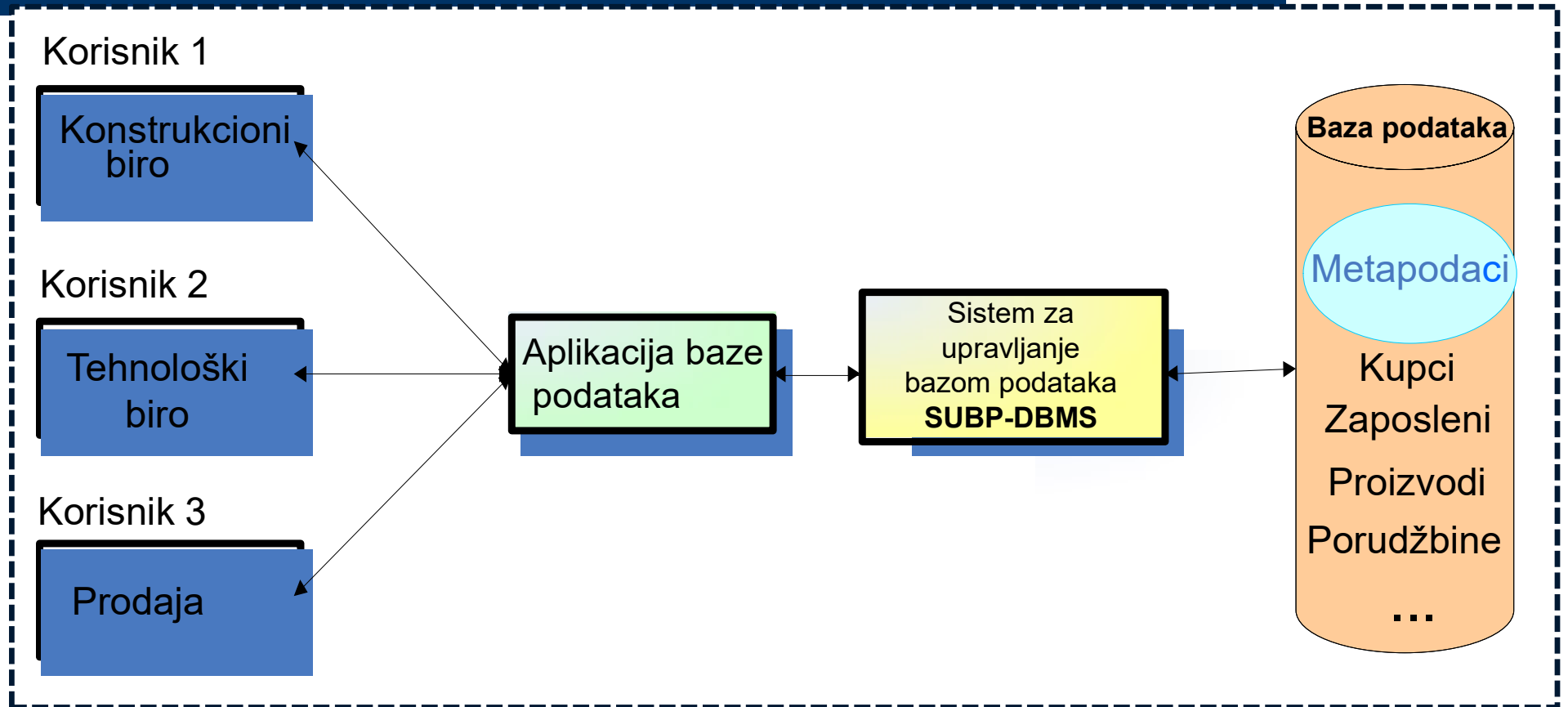
Datoteke se razlikuju prema:

- vrsti (tipu) datoteke
- veličini datoteke
- brzini pristupa slogovima
- uređenosti datoteke
- mogućnostima povezivanja sa drugim datotekama

Klasični sistemi (pomoću aplikacija)



SISTEM ZA UPRAVLJANJE BAZOM PODATAKA



Osnovna uprošćena struktura sistema baze podataka (eng. *database system*).

SISTEM ZA UPRAVLJANJE BAZOM PODATAKA

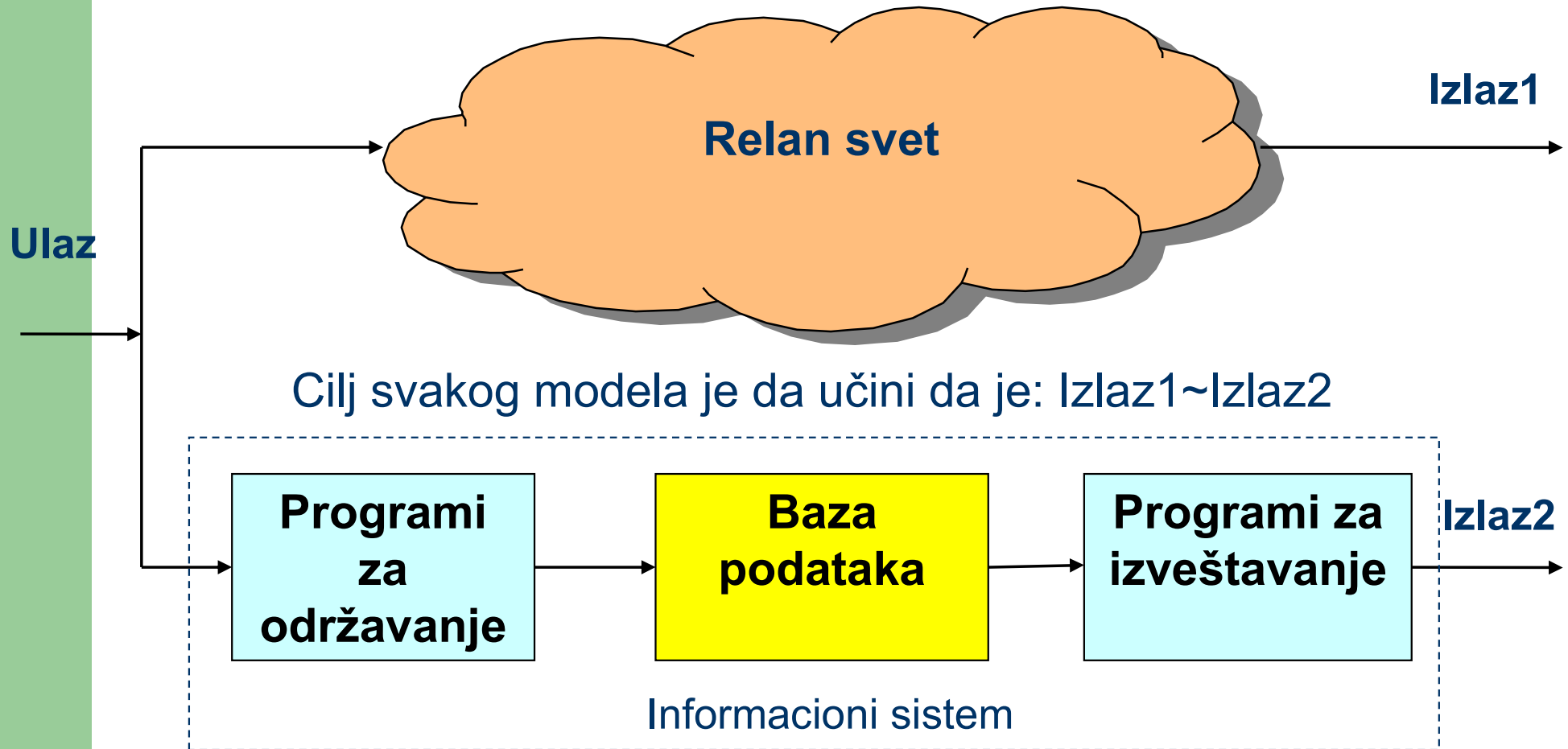
Sistemi za upravljanje bazom podataka SUBP (Database Management System – DBMS) predstavljaju složene softverske alate za: kreiranje baza podataka, efikasan pristup i manipulaciju podacima, rešavanje problema obezbeđenja integriteta baze podataka pri istovremenom radu više korisnika nad istim podacima kao i u slučajevima sistemskih (hardverskih i softverskih) otkaza sistema, sprečavanje neovlašćenog korišćenja podataka itd.

Za sistem sa bazom podataka **podataka** (eng. *database system*). često koristi i termin **informacioni sistem**, ovde ćemo navesti razliku između ova dva termina. Pod sistemom sa bazom podataka podrazumevamo tehnički aspekt ovakvog sistema, dok problematika informacionih sistema obuhvata i razmatra i druge aspekte, kao što su poslovni, organizacioni, ekonomski, psihološki, etički, zakonodavni, itd, što informacione sisteme čini složenim sociotehničkim sistemima.

AKTUELNI SOFTVERSKI PAKETI ZA RAD S BAZOM PODATAKA

Proizvođač	Proizvod	Operativni sistem	Jezici
IBM Corporation	DB2	Linux, UNIX (razni), MS Windows, VMS, MVS, VM, OS/400	SQL, COBOL, Java, ...
Oracle Corporation	Oracle	MS Windows (razni), Mac OS, UNIX (razni), Linux i drugi	SQL, Java i drugi
IBM Corporation (pre: Informix)	Informix	UNIX (razni), Linux, MS Windows	SQL, Java i drugi
Microsoft	MS SQL Server	MS Windows NT/2000/XP	SQL, C++
MySQL AB	MySQL	Linux, UNIX (razni), MS Windows (razni), Mac OS	SQL, C, PHP. ...
Sybase Inc.	Sybase SQL Server	MS Windows, OS/2, Mac, UNIX (razni), UNIXWare	SQL, COBOL, ...
Hewlett Packard Co.	Allbase/SQL	UNIX (HP-UX)	SQL, 4GL, C, ...
Cincom Systems Inc.	Supra	UNIX, Linux, OpenVMS	SQL, COBOL, ...
Microsoft	MS Access	MS Windows (razni)	Access, Basic, SQL

MODEL - REALNI SVET



MODEL - REALNI SVET

Realni svet je izvor podataka za formiranje modela. U praksi se realni sistem prvo mora modelirati (postoji više različitih sredstava za modeliranje sistema).

Model treba da što vernije predstavi stvarnost, ali istovremeno da ima izabrane **samo elemente i karakteristike sistema koje su značajne za realizaciju cilja modela.**

Samu suštinu informacionog sistema predstavlja **baza podataka** (ili sistem baza podataka) koja sadrži međusobno povezane podatke kojima se modeliraju objekti, veze između objekata (entiteta) kao i odgovarajući atributi objekata i veza.

Pored podataka informacioni sistem treba da sadrži i podatke o procesima (model procesa). Pod procesima se podrazumevaju aktivnosti koje generišu promenu stanja u informacionom sistemu i koje generišu izlaze iz sistema.

Baza podataka ...

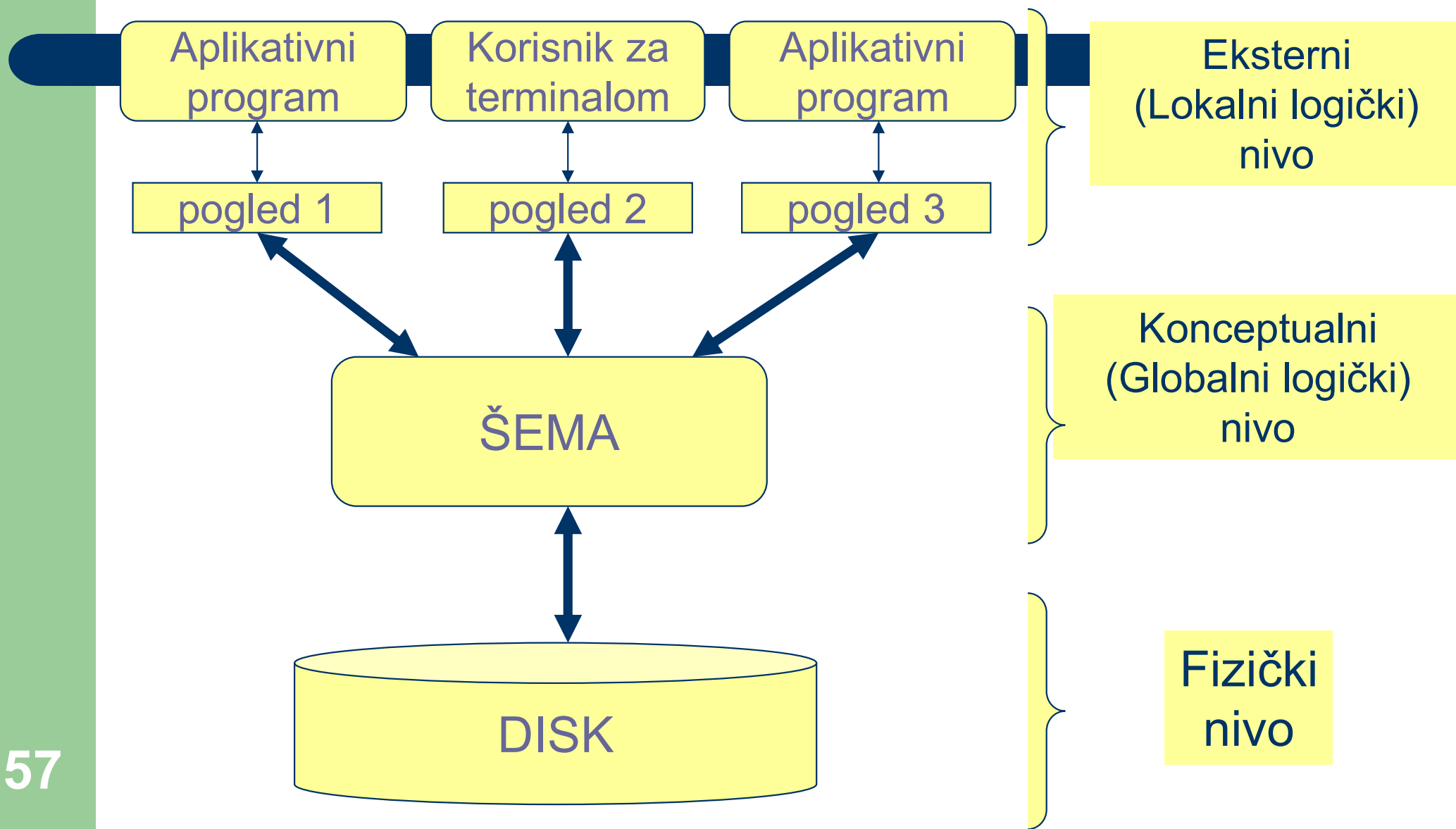
- ... predstavlja kolekciju povezanih podataka sa višestrukom namenom.
- ... je kolekcija međusobno povezanih podataka uskladištenih sa minimumom redundanse, koje koriste svi procesi obrade u sistemu.

Baza podataka

Integracijom informacionih podsistema u jedan poslovni sistem stvorila je potrebu da se skupovi datoteka sistemski integrišu u baze podataka.

Baza podataka nije samo skup međusobno povezanih datoteka, nego svaka od tih datoteka (odnosno skup podataka), mora zadovoljavati određena pravila.

Arhitektura BP



Modeli podataka

- **Konceptualni modeli** – kako korisnici poimaju podatke iz IS
- **Logički modeli podataka** - stvaraju ga informatičari kao podlogu za projektovanje i razvoj IS
- **Fizički modeli podataka** – kako su podaci predstavljeni u kompjuteru

Model podataka – logička struktura baze podataka

Model podataka je intelektualno sredstvo za prikazivanje objekata sistema, njihovih atributa i njihovih međusobnih veza.

Mogu se klasifikovati u sledeće “generacije”:

- prva generacija modela podataka – **konvencionalni programski jezici** (jezici III generacije).
- druga generacija modela podataka – tri klasična modela baze podataka (**hijerarhijski, mrežni i relacioni model**) (jezici IV generacije).
Relacioni model je i danas najviše u primeni i u nastavku će biti detaljnije razmatran.
- treća generacija modela podataka – semantički bogati modeli podataka i objektni modeli podataka (**model objekti-veze**) (izrađuju se pomoću C++, JAWA, Basic)

Razvoj programskih jezika

III generacija jezika:

- **FORTRAN** (1957.) FORmula TRANslation
- **COBOL** (COmmon Business Oriented Language),
- **ALGOL** (ALGOrithmic Language)
- **BASIC** (Beginners All-purpose Symbolic Instruction Code)
- **PL/I** (Program Language I)
- **PASCAL** – 1968. uvodi struktorno programiranje

IV generacija jezika:

- **QBE** (1978. Query-By-Example);
- Obrada teksta (Word);
- Obrada tablica podataka (Excel);
- Grafička obrada podataka (AutoCAD);
- Rad s bazama podataka (Access); i
- Generatori izveštaja (DELPHI).

Danas:

Izrađuju se pomoću C++, JAWA, Basic

Što je neposredna budućnost?

Skladištenje

podataka (Data Warehousing)-

pohranjivanje podataka
neke organizacije
neophodnih za
poslovno odlučivanje

Iskopavanje

podataka (Data Digging)-

traženje određenih
podatak u cilju
stvaranja potrebnih
informacija

Rudarenje podataka (Data mining)-

izvođenje znanja iz pohranjenih informacija

Logičke i fizičke strukture podataka

- Najveća prednost IS zasnovanih na BP u odnosu na IS zasnovane na sistemu datoteka je u tome što sistemi zasnovani na BP odvajaju logičke od fizičkih struktura podataka.
- Šta je LSP?
 - To je način na koji korisnik ili projektant IS i BP konceptualno organizuje i razume podatke.

Logičke i fizičke strukture podataka

- Šta je FSP?
 - Ona definiše kako i gde su podaci fizički uređeni i memorisani na HD, CD-ROM, ili drugim magnetnim i/ili optičkim medijumima.
- SUBP kontroliše bazu podataka tako da korisnici mogu pristupati podacima, realizovati upite, ili ažurirati BP, bez referenciranja kako i gde su podaci fizički memorisani i organizvani.

Logičke i fizičke strukture podataka

Logički pogled korisnika A

Logički pogled korisnika B

Radni nalozi u toku

<u>RN</u>	<u>Lansirano</u>	<u>Datum</u>
34567	3000	02/02/08
87654	1600	05/10/08

Pregled nezavršenih radnih naloga



SUBP

Operativni sistem

Baza podataka

Konceptualni model podataka Model objekti (entiteti)-veze

Model objekti-veze koji je predložio Čen (Chen) 1976. godine je verovatno najrasprostranjenija tehnika analize podataka.

U **modelu objekti-veze** sistem (npr: preduzeće) se posmatra kao skup podataka koji predstavljaju **objekte** (entitete) od interesa za sistem i **veze** između ovih objekata (entiteta).

Objekti i veze su opisani **atributima**.

Iz **modela objekti-veze** je moguće jednostavnim pravilima generisati **relacioni model podataka**, na kojem su bazirani danas najaktuelniji **DBMS**.

Entiteti

- **Entitet** je objekat, pojam, stanje i slično o kome se prikupljaju, memorišu, obrađuju i prezentuju informacije.
- **Entitet** po svojoj prirodi može biti različit:
 - Deo okruženja (član kolektiva, aparat, zgrada, artikal, vozilo ...);
 - Apstraktni pojam (neka mera, nečije zvanje, boja, ...);
 - Događaj (kontrola, postupak upisa studenata,...);
 - Asocijacija (student-predmet, predmet-profesor, ..., fakultet-profesor).
- Napomena: Prilikom kreiranja modela neophodno je utvrditi sve entitete pomoću kojih je moguće opisati sistem (npr: preduzeće).

Atribut

- Atribut je zajednička osobina koju poseduju svi entiteti jedne klase (istog tipa entiteta). Za svaki atribut se uglavnom definišu: tip podatka, domen i defaultna vrednost.
- Svojstva objekta se opisuju preko atributa.

Primer:

- Atributi entiteta STUDENT:
 - BrInd, Ime, Prezime, Fakultet, Smer, Adresa
- Broj atributa nije fiksna.
- Atributi svih entiteta poprimaju određene vrednosti. Skup dozvoljenih vrednosti nekog atributa predstavlja njegov **domen**.

Primer: Vrednosti atributa entiteta STUDENT (jedan entitet iz klase studenata):

- Br.Ind: **123/03**
- Ime: **Marko**
- Prezime: **Marković**
- Broj tel.: 065/1234-567
- Adresa: Krfaska 156, 11000 Beograd

Tipovi (klase) entiteta, skupovi entiteta, ključevi i skupovi vrednosti

- **Tip (klasa) entiteta** predstavlja kolekciju ili skup entiteta koji imaju iste attribute. Svaki tip entiteta u bazi podataka opisan je svojim imenom (ime se piše velikim slovima) i atributima.
- **Skup entiteta** je skup svih entitete određenog tipa u bazi podataka u bilo kom trenutku vremena. Skup entiteta se obično označava istim imenom kao i tip entiteta.
- Entitet se jedinstveno određuje pomoću jednog ili skupa atributa koji predstavljaju **ključ**. Tip entiteta obično ima atribut ili skup atributa čije su vrednosti različite za svaki pojedinačni entitet iz skupa entiteta i koriste se za jedinstvenu identifikaciju svakog entiteta (zove se **ključ**).
- Svakom atributu dodeljen je **skup vrednosti (domen)** koji predstavlja skup vrednosti koje mogu biti dodeljene tom atributu za svaki pojedinačni entitet.

Veza (poveznik)

- Između različitih tipova entiteta u modelu postoje **veze**. U modeliranju, **veze** predstavljaju interakcije koje postoje između entiteta.
- Pošto su definisani tipovi entiteta i njihovi atributi neophodno je tipove entiteta međusobno povezati **vezama**. I same **veze** mogu imati attribute.
- **Tip veze** R između n tipova entiteta E_1, E_2, \dots, E_n predstavlja **skup veza** između entiteta iz tih tipova entiteta. Kao i u slučaju tipa entiteta i skupa entiteta, **tip veze** i njemu odgovarajući **skup veza** se obično označavaju istim imenom R . Tip veze mora da ispunjava određene uslove.

Veza (poveznik)

- Matematički, skup veza R je skup **pojedinačnih veza** r_i gde svaka veza r_i povezuje n entiteta (e_1, e_2, \dots, e_n) i svaki entitet e_j u r_i je član tipa entiteta E_j , $1 \leq j \leq n$.
- Svaka pojedinačna veza r_i u R predstavlja asocijaciju između entiteta, pri čemu asocijacija uključuje tačno jedan entitet iz svakog od tipova entiteta između kojih je posmatrana veza.
- Svaka takva pojedinačna veza r_i ukazuje na činjenicu da su entiteti koji učestvuju u r_i na neki način povezani u minisvetu.

Veza (poveznik)

- **Stepen** tipa veze odgovara broju tipova entiteta koji učestvuju u vezi.
- **Tip veze** stepena dva zove se **binarna veza**, a ako je stepen tri onda je to **ternarna veza**, a ako učestvuje više entiteta za vezu se kaže da je **n-arna**.
- Moguće je da veza bude i **unarna** odnosno da je u vezu uključen samo jedan entitet. Unarne veze se najčešće označavaju kao **rekurzivne** veze pošto one povezuju entitete istog tipa.
- U praksi se najčešće koristi binarna veza. Svaka n-arna veza ($n > 2$) se može zameniti modelom koji sadrži samo binarne veze.

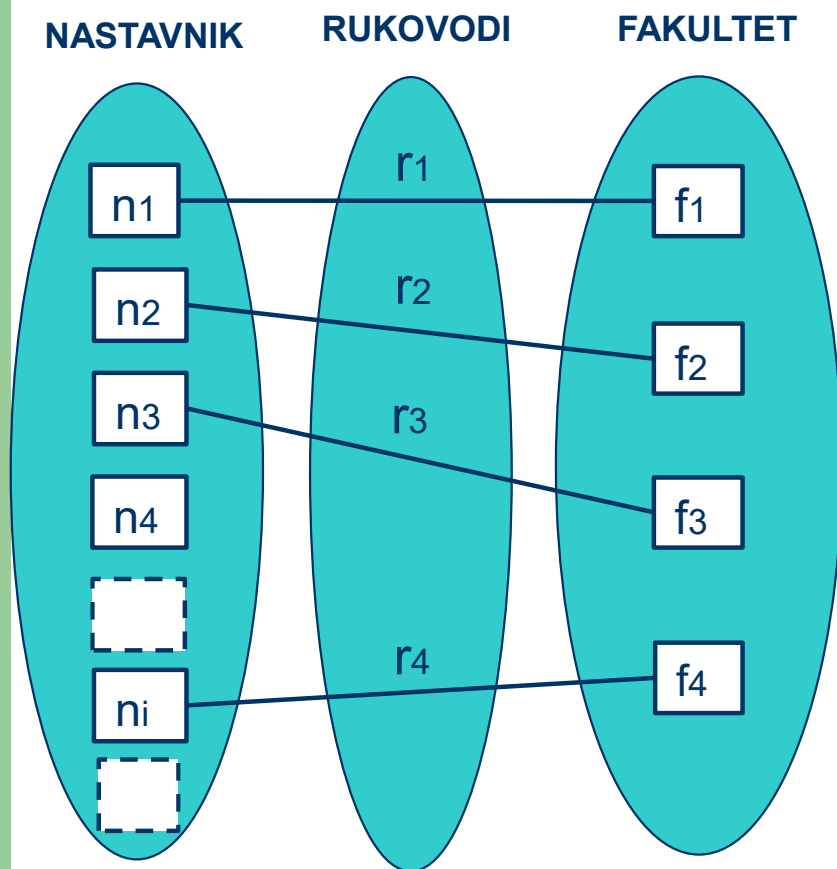
Kardinalnost veza (poveznik)

- U modelu realnog sistema informaciju o prirodi odnosa između entiteta povezanih klasa (tipova) daje tkozvana **kardinalnost tipa veze (poveznika)** R odnosno kardinalitet odgovarajuće relacije R.
- Kardinalitet tipa veze (poveznika) R između tipova entiteta E_1 i E_2 se označava sa:
 - $R(E_1(a_1, b_1): E_2(a_2, b_2))$, gde parametar **a** uzima vrednost iz skupa $\{0, 1\}$, a **b** iz skupa $\{1, N\}$.
- Kardinalnost tipa poveznika govori u koliko pojava tipa veze (poveznika) može učestovati minimalno (**a**) i maksimalno (**b**) jedna pojava tipa entiteta.
- S obzirom na maksimalne vrednosti kardinalnost binarne veze može biti: 1:1, 1:N ili N:1 i M:N (više prema više).

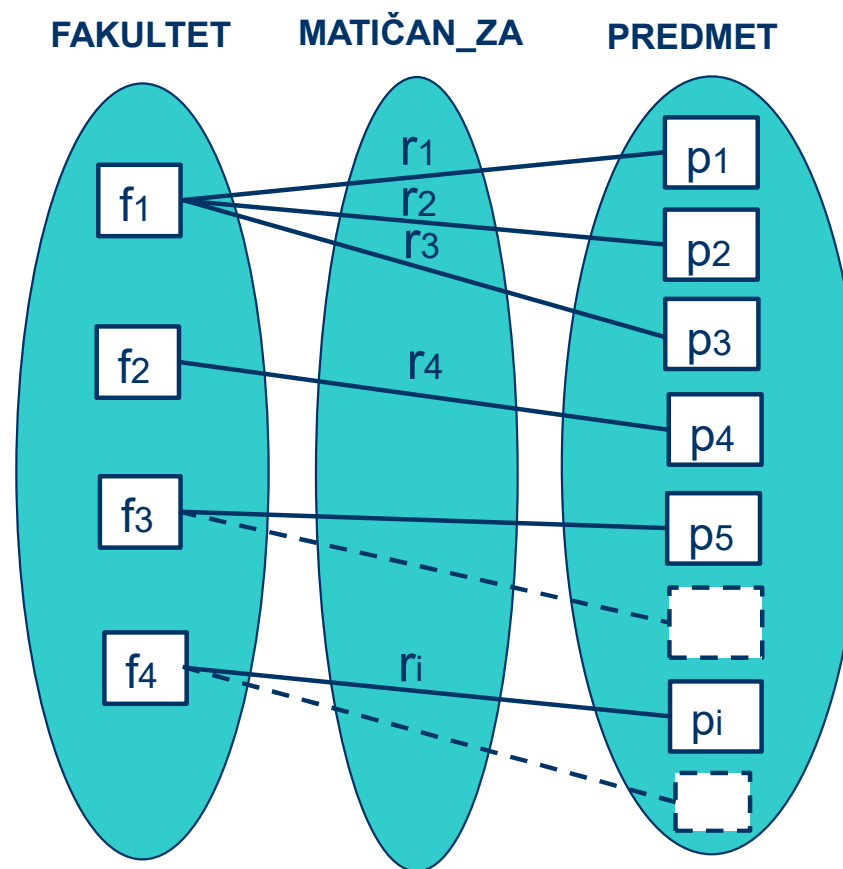
Kardinalnost binarne veze (poveznika)

- Odnosi između entiteta:
- Veza 1:1 (jedan prema jedan) između tipova entiteta E_1 i E_2 ukazuje da za svaki entitet u jednom tipu entiteta postoji najviše jedan entitet u drugom tipu entiteta s kojim je on u vezi.
- Veza 1:M (jedan prema više) između tipova entiteta E_1 i E_2 ukazuje da entitet E_1 može biti u vezi ni sa jednim, jednim ili više entiteta iz tipa entiteta E_2 , dok entitet iz tipa entiteta E_2 može biti u vezi samo sa jednim entitetom iz tipa entiteta E_1 .
- Veza M:N (više prema više) između tipova entiteta E_1 i E_2 ukazuje da ne postoji restrikcija u pogledu broja entitet jednog tipa entiteta koji su u vezi sa entitetom drugog tipa entiteta.

Veza: 1:1 i 1:M



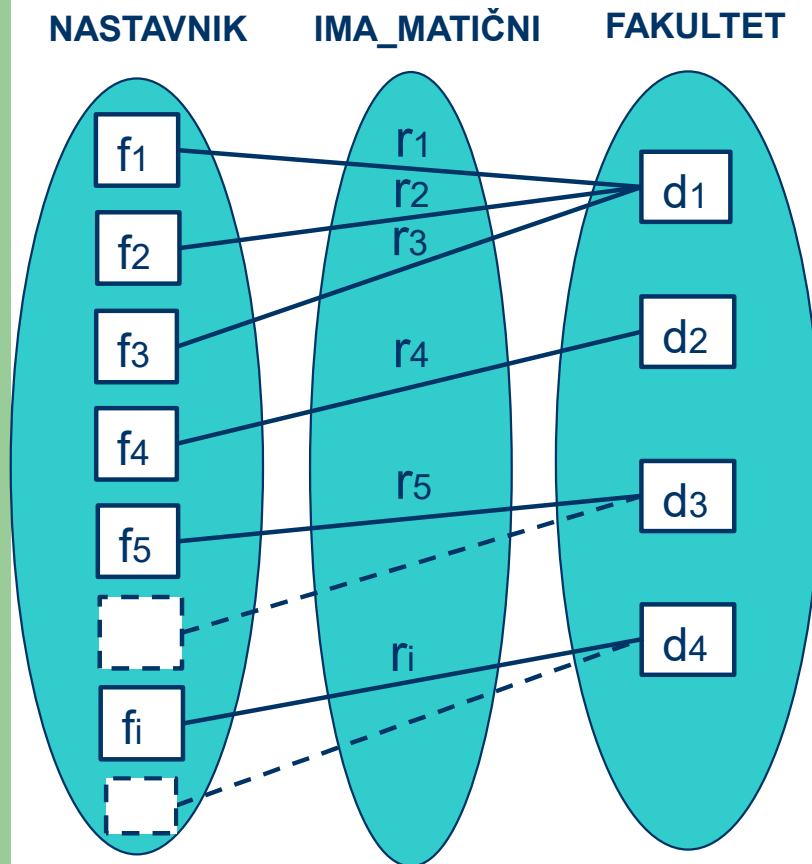
KADINALITET VEZE: **JEDAN PREMA JEDAN (1:1)**



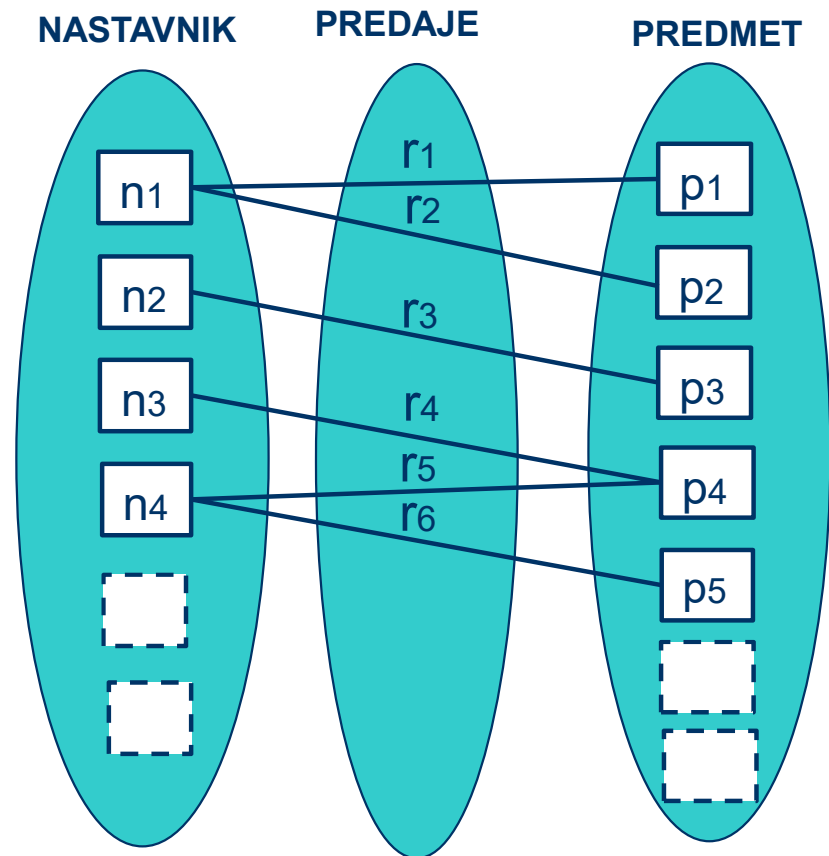
KADINALITET VEZE: **JEDAN PREMA VIŠE (1:M)**

Veza: M:1 i M:N

R



KADINALITET VEZE: VIŠE PREMA JEDAN (M:1)



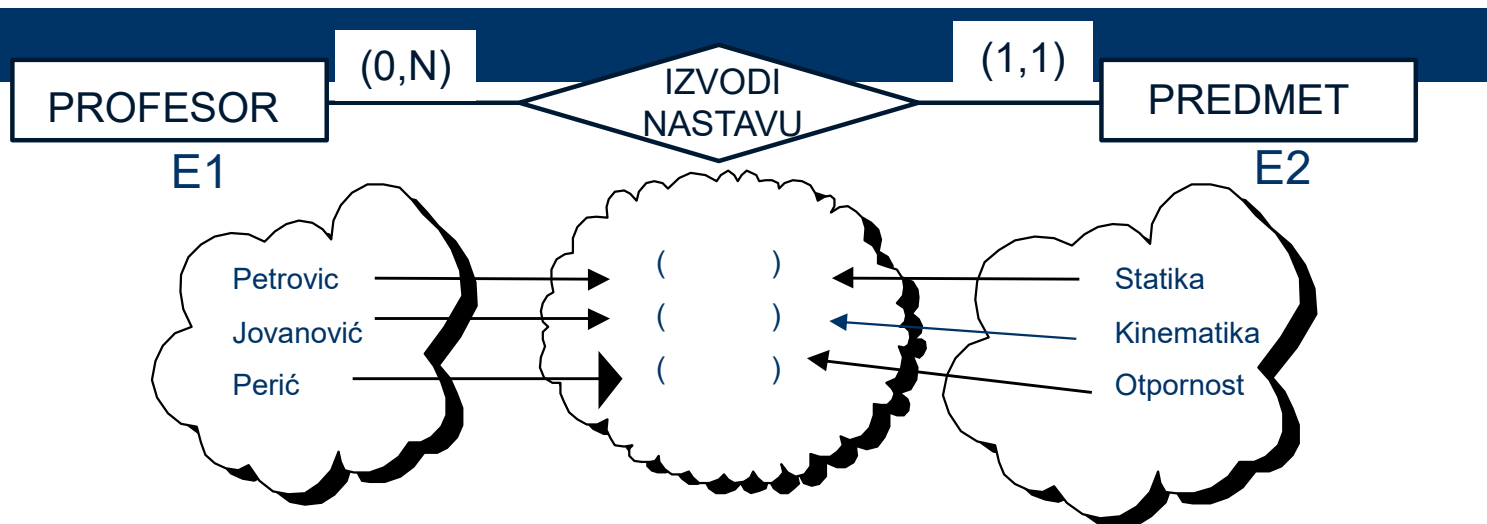
KADINALITET VEZE: VIŠE PREMA VIŠE (M:N)

Kardinalnost tipa veze (poveznika)

- Ugovor se sklapa sa tačno jednim poznatim kupcem;
- Jedan kupac može sklopiti više ugovora, ali mogu postojati kupci koji nisu sklopili ni jedan ugovor.



Strukture sa kardinalitetima grupe 1:1



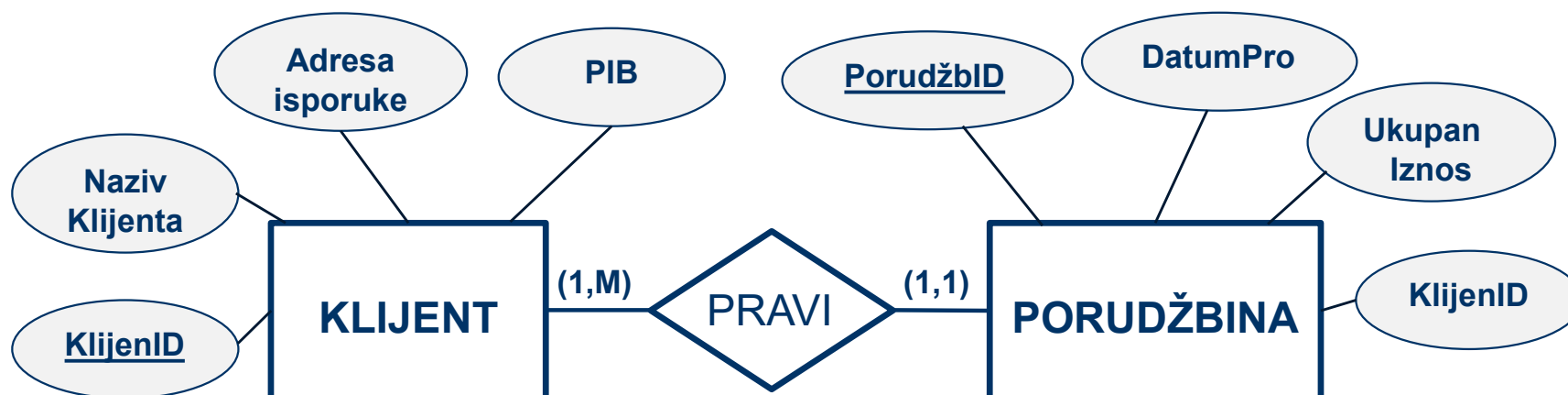
- Slučaj kada je $a_1=0$ i $a_2=0$ implicira da svaki profesor može izvoditi nastavu na najviše jednom predmetu, a ne mora ni na jednom, dok predmet može realizovati najviše jedan profesor, a mogu postojati predmeti koje niko ne drži.
 - **Izvodi nastavu (Profesor(0,1):Predmet(0,1)).**
- Generalno posmatrano, slučaj kada su oba minimalna kardinaliteta jednaki nuli, ukazuje na odnos između klasa entiteta E_1 i E_2 gde svaki entitet jedne klase može biti povezan sa najviše jednim entitetom druge klase.

Grafičko predstavljanje Modela objekti (entiteti)-veze

Za grafičko predstavljanje entiteta i veza između njih koristi se dijagram (E/R) entiteti veze prema Čenovoj notaciji:

- Tip entiteta predstavlja se **pravougaonikom** u koji se upisuje tip entiteta.
- Imena atributa se upisuju u **elipse** koje su sa tipom entiteta povezane pravim linijama. Atributi koji čine ključ su podvučeni.
- Veza se prikazuje **rombom** koji je **pravim linijama** povezan sa pravougaonicima koji predstavljaju tipove entiteta koji učestvuju u vezi.
- Kardinalnost veze (poveznika) je napisana iznad linije koja spaja entitete u vezi (mini/max notacija).

Primer E/R dijagrama



E/R dijagram “Klijent – Porudžbina”

Konceptualni model podataka- zahtevi

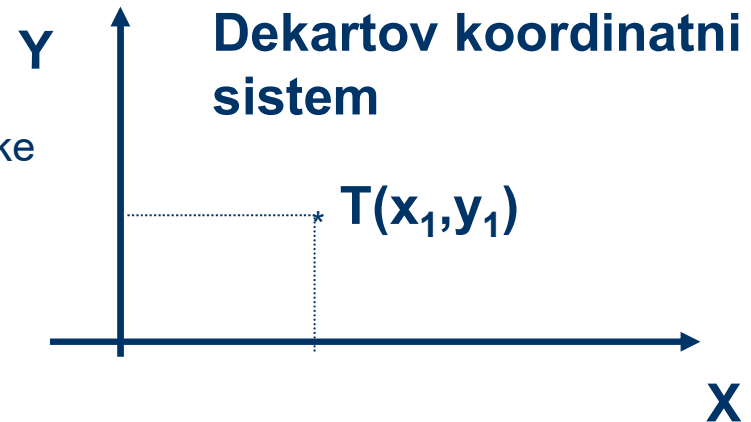
- **Celovit** - obuhvata sve relevantne podatke
- **Konzistentan** - dosledno preslikani podaci o događajima i interpretacija podatka nezavisna od načina pristupa.
- **Nereduntantan** - ne sme biti ponavljanja istih podataka
- **Nezavisan** - od načina implementacije tj od programa

RELACIONE BAZE PODATAKA

Otkud naziv?

Osnivač relacionog modela Edgar Kod je osnovne pojmove relacionog modela preuzeo iz matematičke Teorije.

Teorija skupova; $A=\{a,b,c,d\}$
 $B=\{1,2,3\}$



Dekartov proizvod $A*B = \{(a,1); (a,2); (a,3); (a,4); (b,1); (b,2); (b,3); (b,4);$
 $(c,1); (c,2); (c,3); (c,4); (d,1); (d,2); (d,3); (d,4)\}$

Svaki podskup S (na primer $S = \{(a,2); (b,3); (b,1)\}$) **Dekartovog proizvoda** dvaju skupova naziva se **relacija**.

Termin relacija zasniva se na matematičkoj oblasti koja se zove teorija skupova, tako da je relacioni model podataka matematički zasnovan model.

RELACIJE

S obzirom da Dekartov proizvod može biti:

$A*B$ ili $A*B*C$ itd.

Njegovi su elementi parovi, trojke ili uopšteno rečeno n-torke elemenata u kojima je svaki element iz jednog od skupova koji čine taj proizvod.

Dekartov koordinatni sistem u ravni je u stvari proizvod $R*R$ a prostorni $R*R*R$ itd.

RELACIJE

Kako je relacija matematički concept, to se za praktične primene u relacionom modelu relacija izjednačava sa tabelom. Svaki objekat (entitet) u relacionoj bazi predstavljen tabelom. Svaka n-torka predstavlja red a svaki atribut odgovara imenu kolone koje ukazuje na ulogu ili značenje vrednosti datih u toj koloni. Stepen relacije odgovara broju kolona u tabeli

ID kupca	Naziv kupca	Broj porudžbine	Šifra artikla	Količina
1123	Metcut	4312	5423	16,00
1123	Metcut	4312	5411	18,00
1124	Meteor	3333	5532	15,00
2231	Astra	4576	5423	10,00
2231	Astra	4576	6711	15,00

Relacija

Student	Godina
Marko	85
Ivan	84
Perica	85
Tomislav	84

Student	Broj indeksa
Marko	11801
Ivan	11701
Perica	11803
Tomislav	11601

Broj indeksa	Knjiga
11801	22333
11701	22121
11803	21332
11601	21555

**RELACIJSKA BAZA PODATAKA
SASTAVLJENA JE OD TABELA, KOJE
PREDSTAVLJAJU RELACIJE**

Veze kod relacionog modela baze

Kod relacionih modela veze se uspostavljaju između atributa relacija. *Pojednostavljeno:* Teminom veza (Relationship) ukazuje se na međusobne odnose između primarnih i stranih ključeva tabela.

Strani ključ (*Foreign key*) tabele je atribut (skup atributa) koji ukazuje na zavisnost od neke druge tabele.

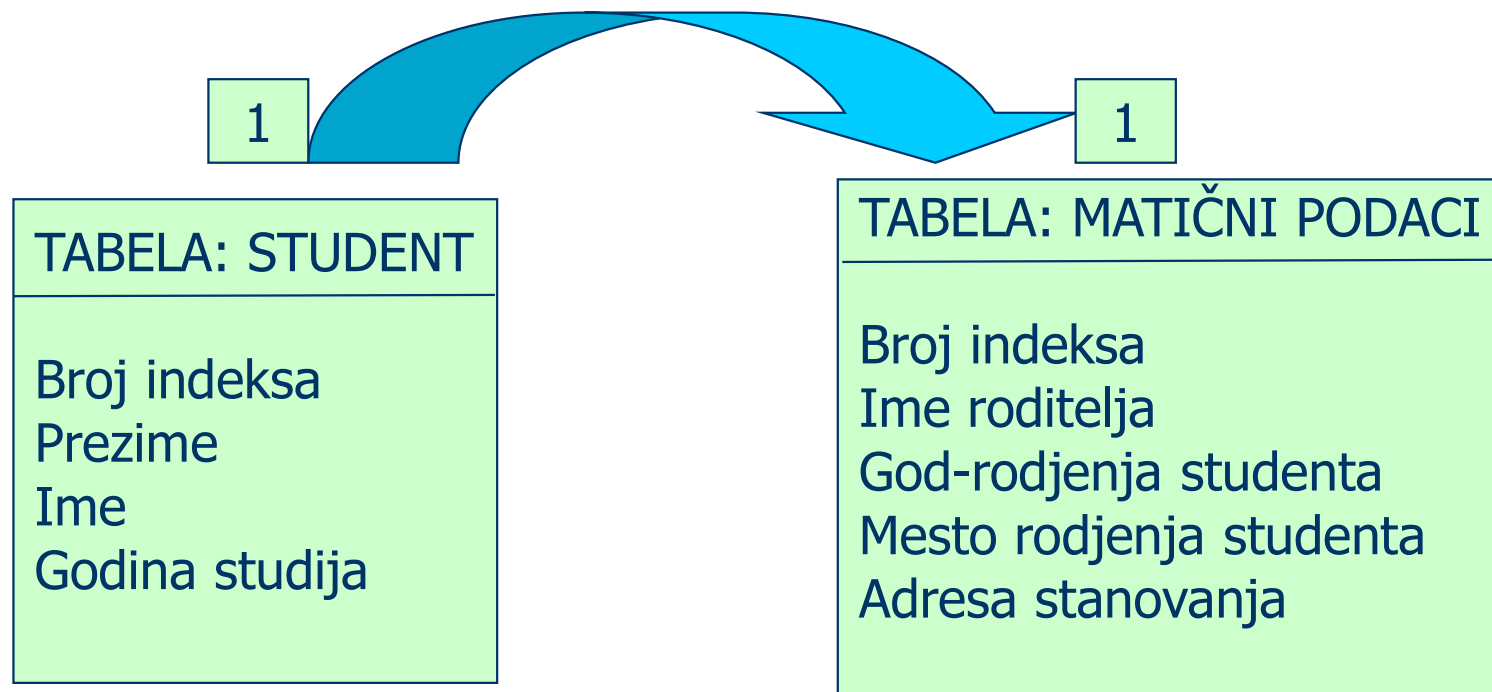
Strani ključ mora da zadovolji uslov da je skup njegovih atributa primarni ključ druge tabele.

ID kupca	Broj porudžbine	Šifra artikla	Količina	JM
1123	4312	5423	12,00	Kg
1123	4312	5411	6,00	m ²
1123	4433	5532	10,00	L
2231	4576	5423	7,00	Kg
2231	4576	6711	3,00	T

Strani ključ

Primarni ključ- jednoznačno identifikuje vrstu u tabeli

Tipovi zavisnosti: jedan prema jedan (1:1)



Koncept RBP:

Na slici su prikazane 3 tabele sa ukupno 12 atribuda (različitih kolona)

- Tab1 i Tab2 “dele” atribut 3
- Tab2 i Tab3 “dele” atribut 7

Tab1

Attrib. 1	Attrib. 2	Attrib. 3	Attrib. 4	Attrib. 5

Tab2

Attrib. 3	Attrib. 6	Attrib. 7

Tab3

Attrib. 8	Attrib. 9	Attrib. 10	Attrib. 11	Attrib. 7	Attrib. 12

Normalizacija

... rekonstrukcija fajlova podataka čiji je cilj smanjenje podataka u bazi na najjednostavniju strukturu i minimizacija redudantnosti tj. normalizacija je takva organizacija podataka koja će omogućiti najefikasniji način čuvanja i pretraživanja i povećati integritet baze podataka.

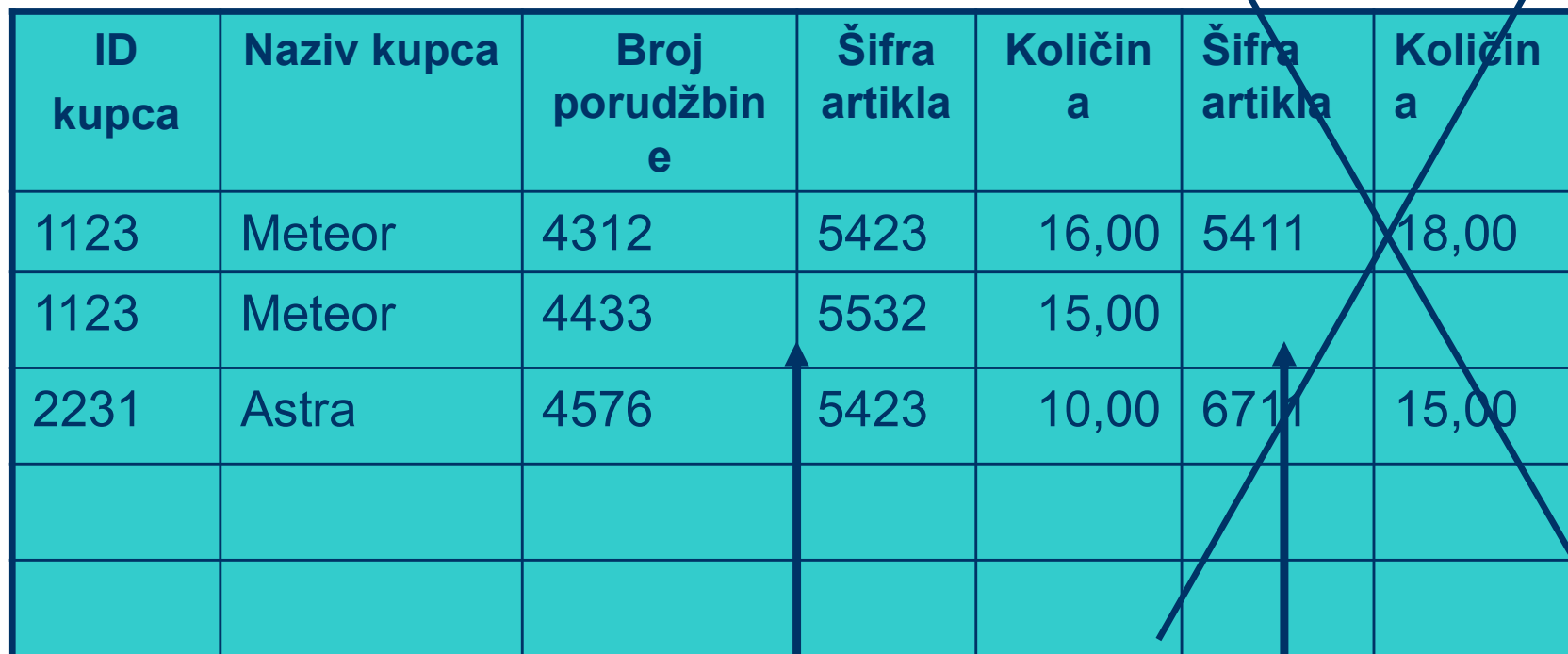
- Pravilo 1: eliminacija grupa podataka koje se ponavljaju (1NF)
- Pravilo 2: eliminacija redudantnih podataka (2NF)
- Pravilo 3: eliminacija kolona koje ne zavise od primarnog ključa (3NF)
- Pravilo 4: Izdvajanje nezavisnih višestrukih zavisnosti (4NF)
- Pravilo 5: Izdvajanje povezanih višestrukih zavisnosti (5NF)

1NF – opis svojstava

1. Nema duplih kolona
2. Sve vrijednosti u koloni su istovetne vrednosti
3. Svaka vrsta je jedinstvena
4. Redosled kolona nije bitan
5. Redosled vrsta nije bitan
6. Svaka kolona ima jednoznačno ime

Normalizacija -1NF

Ova tabela nije u 1NF



ID kupca	Naziv kupca	Broj porudžbine	Šifra artikla	Količina	Šifra artikla	Količina
1123	Meteor	4312	5423	16,00	5411	18,00
1123	Meteor	4433	5532	15,00		
2231	Astra	4576	5423	10,00	6711	15,00

Prva normalna forma (1NF) – svaki atribut mora se navesti samo jednom

Normalizacija -1NF

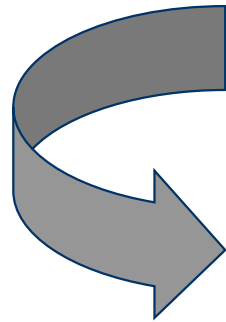
ID kupca	Naziv kupca	Broj porudžbine	Šifra artikla	Količina
1123	Metcut	4312	5423	16,00
1123	Metcut	4312	5411	18,00
1123	Meteor	4433	5532	10,00
2231	Vektra	4576	5423	10,00
2231	Vektra	4576	6711	15,00

Prva normalna forma (1NF) – svaki atribut se javlja samo jednom.

Gornja tabela zadovoljava ovaj uslov

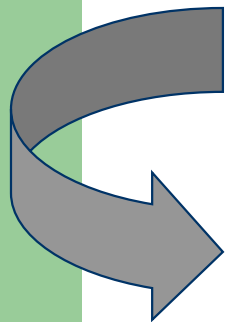
Pravilo 1: Eliminacija grupa podataka koje se ponavljaju (Prva normalna forma)

Tabela: student



- broj indeksa
- prezime
- ime
- godina studija
- strani jezik I
- nivo znanja I
- strani jezik II
- nivo znanja II
- strani jezik III
- nivo znanja III

Pravilo 1: Eliminacija grupa podataka koje se ponavljaju (Prva normalna forma)



Normalizovana baza podataka (Prva normalna forma)

Tabela : matični podaci

broj indeksa
prezime
ime
godina studija

Tabela : znanje stranih jezika

broj indeksa
šifra jezika
naziv jezika
nivo znanja

Pravilo 2: Eliminacija redundantnih podataka (Druga normalna forma)

Normalizovana baza podataka (**Druga normalna forma**)

Tabela : matični podaci

broj indeksa
prezime
ime
godina studija

Tabela : znanje stranih jezika

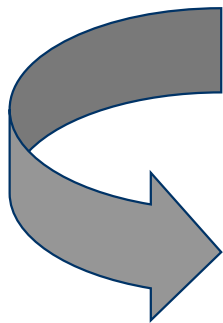
broj indeksa
šifra jezika
nivo znanja

Tabela: šifarnik stranih jezika

šifra jezika
naziv jezika

Pravilo 3: Eliminacija kolona koje ne zavise od primarnog ključa (Treća normalna forma)

Tabela: matični podaci



- broj indeksa
- prezime
- ime
- godina studija
- odeljenje
- mesto
- adresa
- telefon
- faks
- koordinator

Pravilo 3: Eliminacija kolona koje ne zavise od primarnog ključa (Treća normalna forma)

Normalizovana baza podataka (Treća normalna forma)

Tabela : matični podaci

broj indeksa
prezime
ime
godina studija
odelenje

Tabela : fakultetska odelenja

odelenje
mesto
adresa
telefon

Rečnik podataka- Data dictionary

To je datoteka (baza) u kojoj je definisana baza podataka što uključuje popis svih baza podataka, tabela, slogova, naziv i tip svih polja u slogovima.

Dakle u rečniku podataka su smešteni metapodaci (podaci o podacima).

Rečnik podataka omogućava SUBP održavanje logičkog integriteta baze podataka.

Programski jezik SUBP

Svaki SUBP mora omogućiti izvršenje tri bazične funkcije:

- 1 Kreiranje BP
- 2 Ažuriranje BP
- 3 Pretraživanje BP

Skup komandi koje izvršavaju ove funkcije se definišu kao:

1. Jezik za definisanje podataka (data definition language DDL).
2. Jezik za manipulaciju podacima (data manipulation language DML)
3. Upitni jezik (data query languages DQL – SQL).

Napomena: Kod relacionih baza podataka postoji tendencija da se sva tri jezika integrišu u jedan jezik.

Jezik za definisanje podataka - DDL

DDL se koristi za:

- Izgradnju rečnika podataka.
- Inicijalizaciju ili kreiranje BP.
- Opis logičkog pogleda za svakog pojedinačnog korisnika, projektanta ili programera.
- Specifikaciju mase ograničenja iz domena sigurnosti ili pristupu tabelama, slogovima ili obeležjima BP.

JEZIK ZA MANIPULACIJU PODACIMA DML

DML se koristi za održavanje podataka u BP.

Šta to uključuje?

- Ažuriranje dela BP - insert
- Unos u odabrane delove BP - update
- Brisanje odabranih delova BP - delete

DQL - SQL

- DQL se koristi za preglede podataka u BP.
- DQL pretražuje, sortira, postavlja u zadatom redosledu i prezentuje ma koji podskup podataka u BP u odnosu na korisnički definisan zahtev.

SQL (*Structured Query Language*)

SQL je u početku projektovan kao jezik za kreiranje upita zasnovan na relacionom računu, s tim da je matematička notacija zamenjena ključnim rečima nalik na govorni engleski jezik. Pored toga realizuju i sve operacije iz relacione algebre. Osnovni softverski paketi koji koriste SQL su: *Oracle, Microsoft Access, DB2, Microsoft SQL server, SQL Windows, itd.*

Danas je SQL mnogo više od običnog jezika za postavljanje upita, već se može smatrati **standardnim jezikom za rad sa relacionim bazama podataka.**

Pomoću SQL-a korisnik samo definiše informaciju koju želi da dobije (definiše rezultat kako treba da izgleda). Računar zatim određuje i izvršava najbolju proceduru za dobijanje tih informacija. Pošto je fokus usmeren na rezultat ovaj programski jezik je *deklarativan (neproceduralan)*.

SQL (Structured Query Language)

Neproceduralni jezik 4. generacije - međunarodni standard

Šifra Kupca	Broj naloga	Šifra artikla	Količina
1123	4312	6432	12,00
1123	4312	5411	6,00
1123	4433	5532	10,00
2231	4576	6432	7,00
2231	4576	6711	3,00

```
select * from Broj naloga  
where sifrant=6432
```

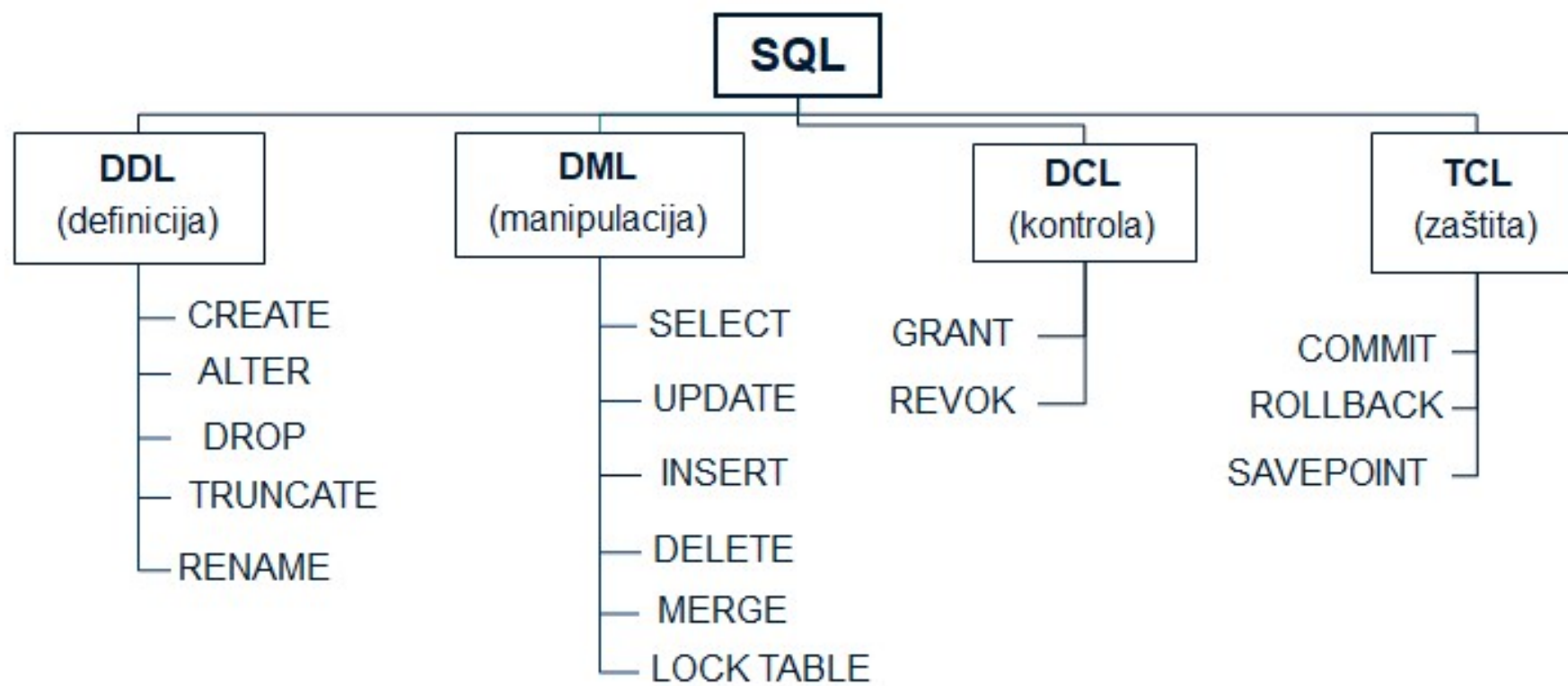
Operacije:

- Selekcije (selection)
- Projekcije (projection)
- Spajanja (join)

Projekcija

Operacija spajanja je obrnut postupak od razdvajanja tablica kod normalizacije, ali to nije denormalizacija, jer je to samo prikaz, a ne utiče na organizaciju podataka.

SQL (Structured Query Language)



Primeri: SQL naredbi

Postavljanje upita

Upit se postavlja fleksibilnom naredbom **SELECT** (nije isto što i operacija selekcije u relacionoj algebri). Rezultat upita se shvata kao nova privremena relacija, izvedena iz stalnih (po tome je SQL sličan relacionoj algebri).

Pronađi brojeve i imena svih studenata na stepenu 1.

```
SELECT SNO, SNAME  
FROM STUDENT  
WHERE LEVEL = 1;
```

Pronađi brojeve i imena studenata koji su upisali bar jedan ispit kojeg predaje Quinn.

```
SELECT SNO, SNAME  
FROM STUDENT  
WHERE SNO IN  
(SELECT SNO  
FROM REPORT  
WHERE CNO IN  
(SELECT CNO  
FROM COURSE  
WHERE LNAME = 'Quinn'));
```

SISTEM OZNAČAVANJA

- KONVENCIONALNI
- KOMPAKTNI
- PARALELNI

OZNAKA

- IDENT BROJ
- NAZIV
- KLASIFIKACIONI BROJ

IDENT BROJ

- JEDNOZNAČAN (KLJUČ SLOGA)
- NEPONOVLJIV (JEDAN ENTITET - JEDAN ID)
- BAR KOD
- RF-ID (radio frekventna identifikacija)

KLASIFIKACIJA

- DECIMALNI SESTEM

$X_1 X_2 X_3 X_4 X_5 X_6 \dots$



Oznaka materijala

Jedna oznaka ↔ jedan materijal

- **Zamenjivost** – materijali koji nisu potpuno zamjenljivi zahtevaju različitu oznaku.
- **Faza proizvodnje** – za svaku fazu proizvodnje (ili nabavke materijala) koja mora biti skladištena zahteva se posebna šifra.
- **Jedinica mjere** - višestruke jedinice mere mogu stvarati probleme.
- **Razni atributi** – posebno mnemoničko ime za svaki atribut

Kontrolni broj po modulu 11

1. svaka pozicija 12-cifarskog broja se množi sa težinskim faktorom i sukcesivno se sabira:

X X X X X X X X X X X X

7 6 5 4 3 2 7 6 5 4 3 2

2. dobijena suma se deli sa 11 i celobrojni ostatak (ako je 0 ili 10 uzima se 0) se oduzima od 11. To je kontrolni broj !!

PRIMER IZRAČUNAVANJA KONTROLNOG BROJA

2802947363006

765432765432

$$7*2=14$$

$$6*8=48$$

$$5*0=0$$

$$4*2=8$$

$$3*9=27$$

$$2*4=8$$

$$7*7=49$$

$$6*3=18$$

$$5*6=30$$

$$4*3=12$$

$$3*0=0$$

$$2*0=0$$

$$11-5=6$$

11

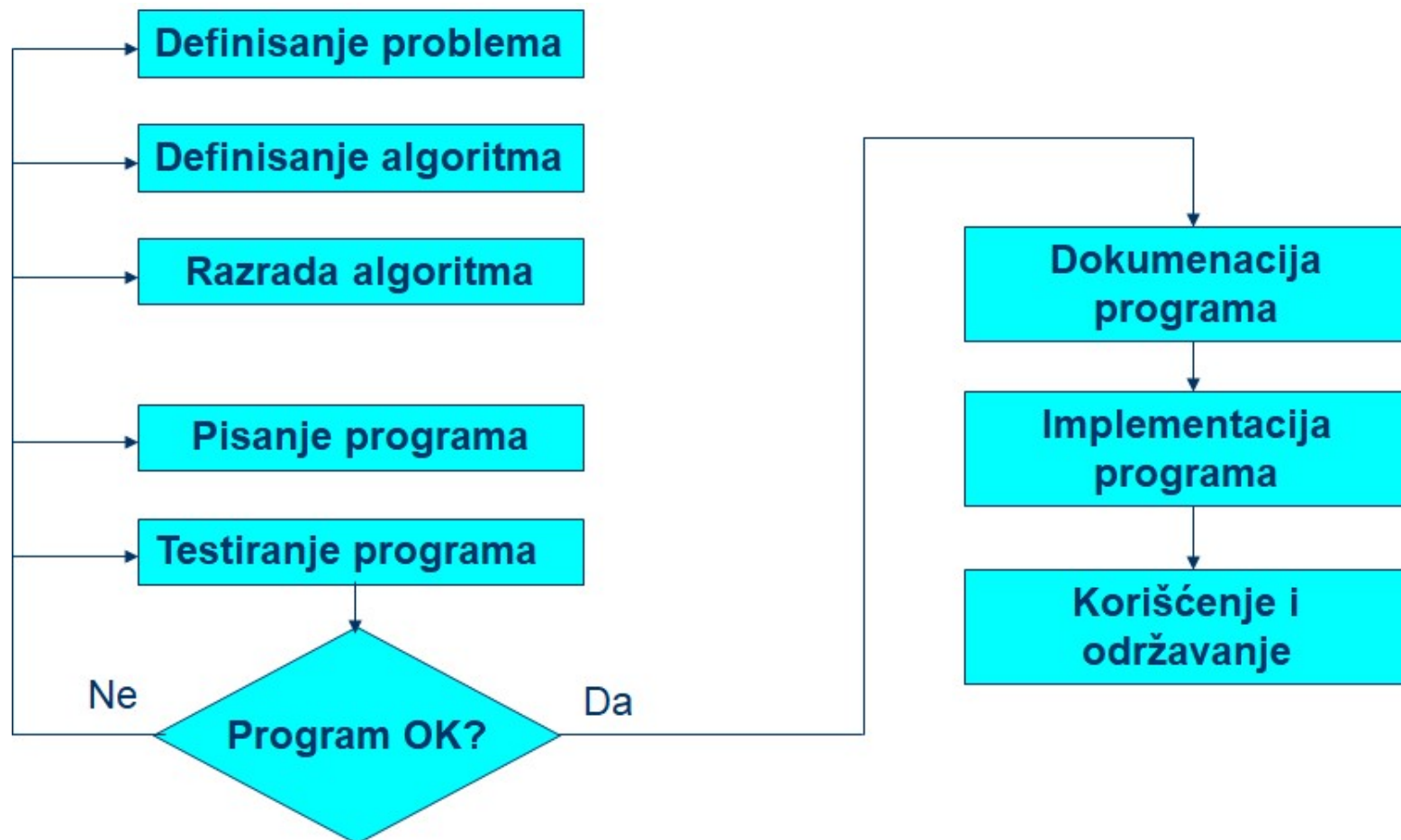
2

$$\longrightarrow \Sigma = 214 / 11 = 19 \rightarrow 11 * 19 = 209 \rightarrow 214 - 209 = 5$$

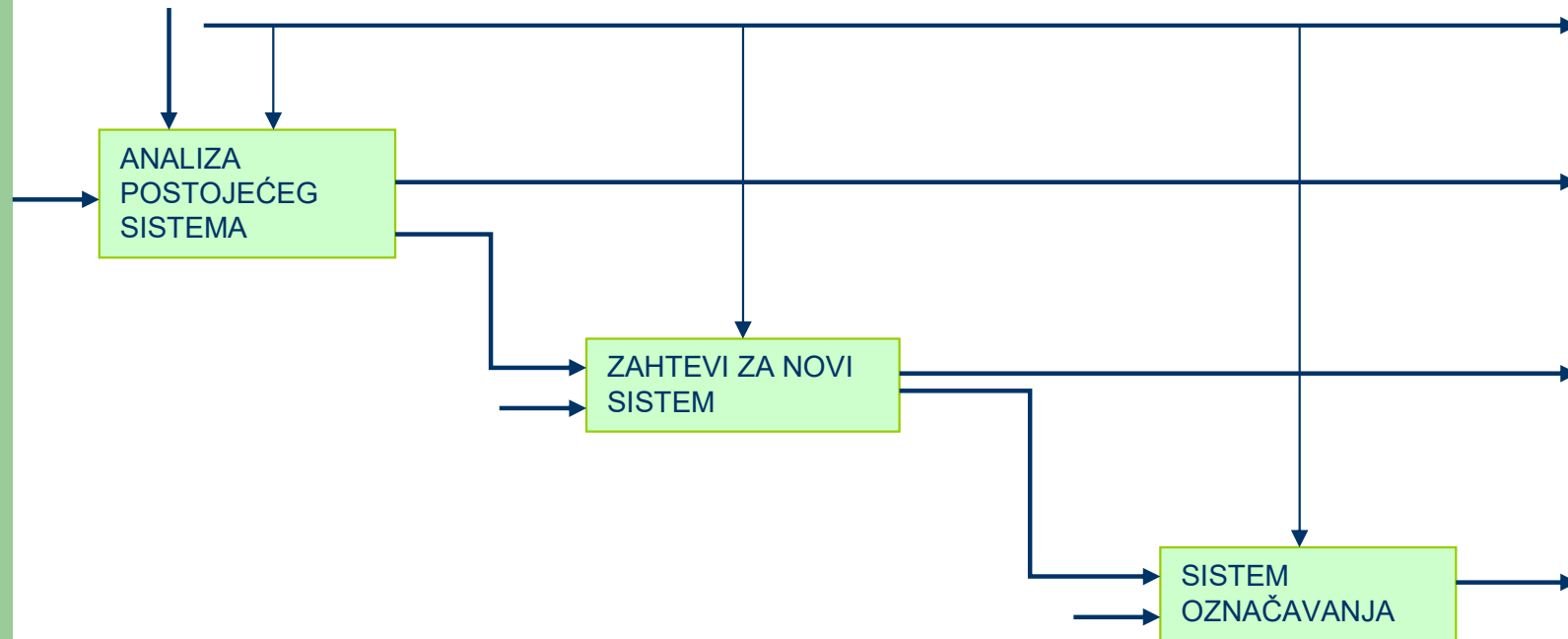
Dizajn i implementacija sistema baze podataka?

- Sest bazičnih koraka u dizajnu i implementaciji sistema baze podataka:
 1. Inicijalno planiranje za određivanje potrebe i izvodljivosti razvoja novog sistema (faza planiranja).
 2. Identifikacija korisničkih zahteva (faza analize zahteva).
 3. Razvoj konceptualne, eksterne i interne šeme BP (faza dizajna).
 4. Prevođenje interne šeme BP u strukture aktuelne baze podataka koja će biti implementirana u novom sistemu (faza kodiranja).
 5. Transfer svih podataka iz postojećeg sistema u novu bazu podataka (faza implementacije).
 6. Korišćenje i održavanje novog sistema (faza korišćenja i održavanja).

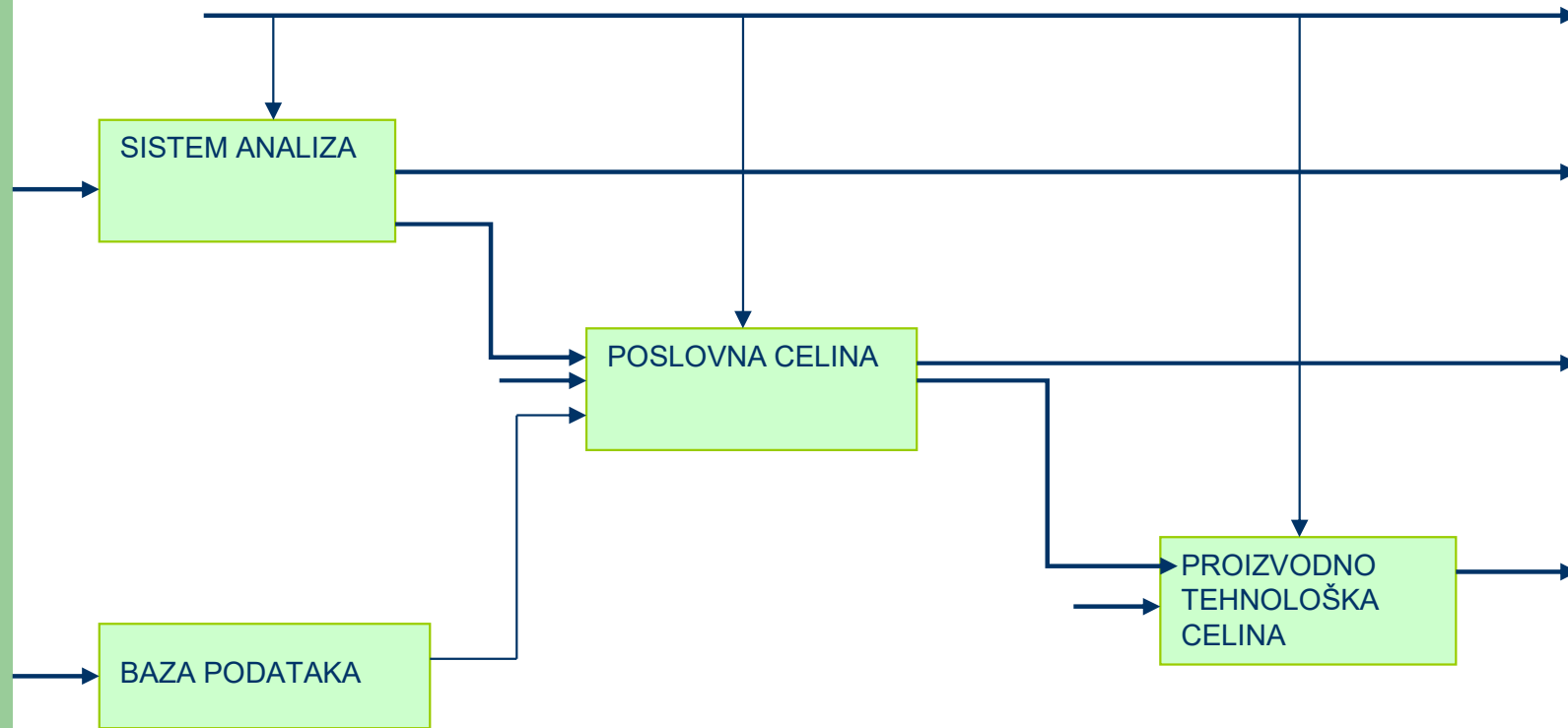
FAZE U RAZVOJU I EKSPLOATACIJI APLIKACIJE



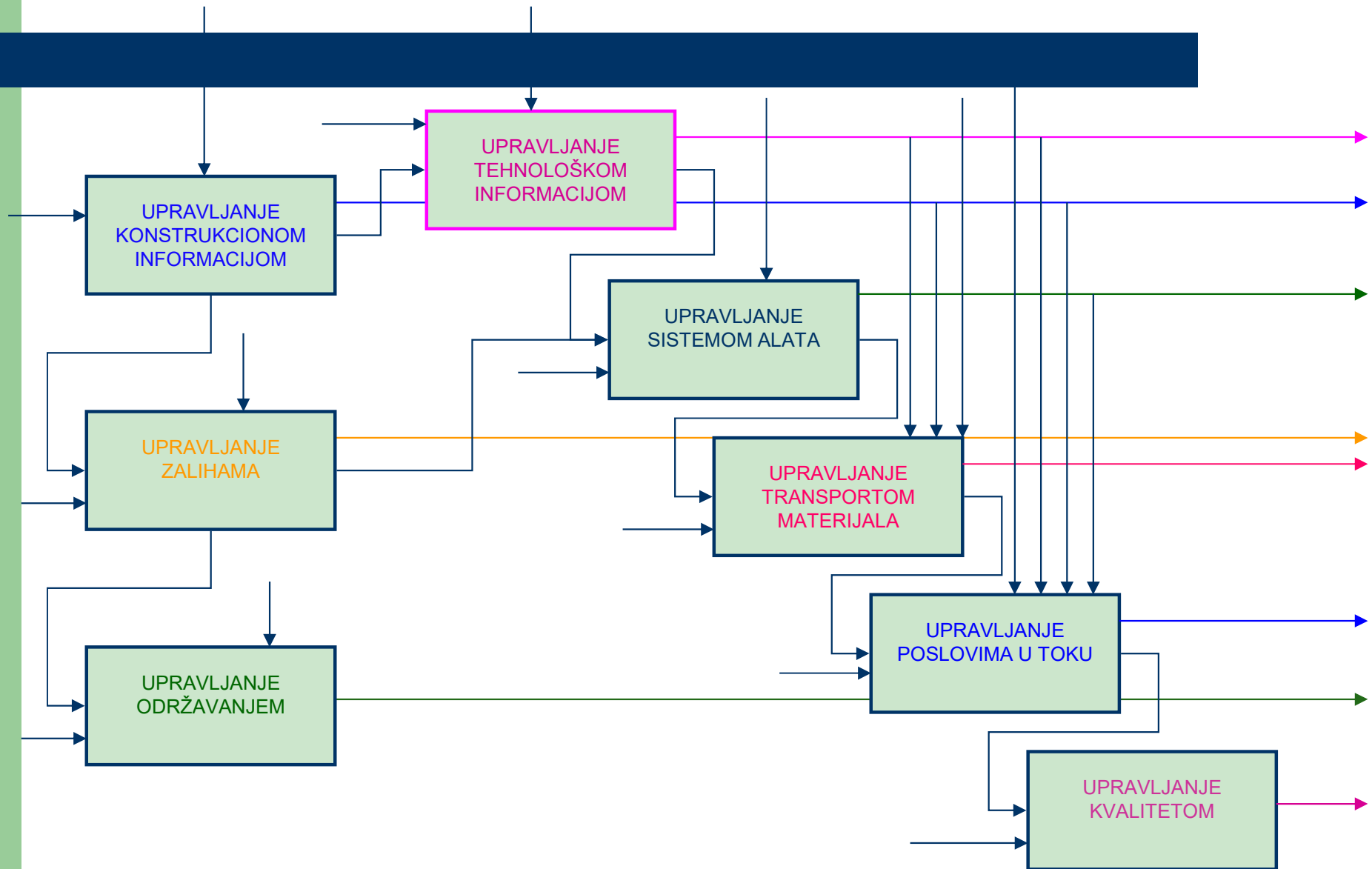
SISTEM ANALIZA



STRUKTURA PROIZVODNOG INFORMACIONOG SISTEMA



PROIZVODNO TEHNOLOŠKA CELINA

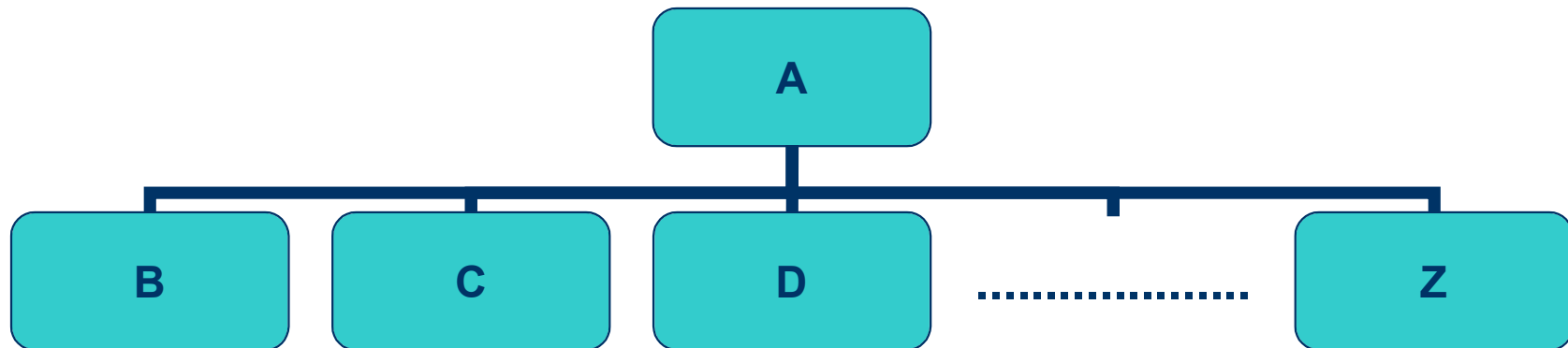


PODSISTEM ZA UPRAVLJANJE KONSTRUKCIONOM DOKUMENTACIJOM

- Bruto potrebe
- Deo konstrukcione dokumentacije
- Unifikacija i standardizacija
- Sistem izmena

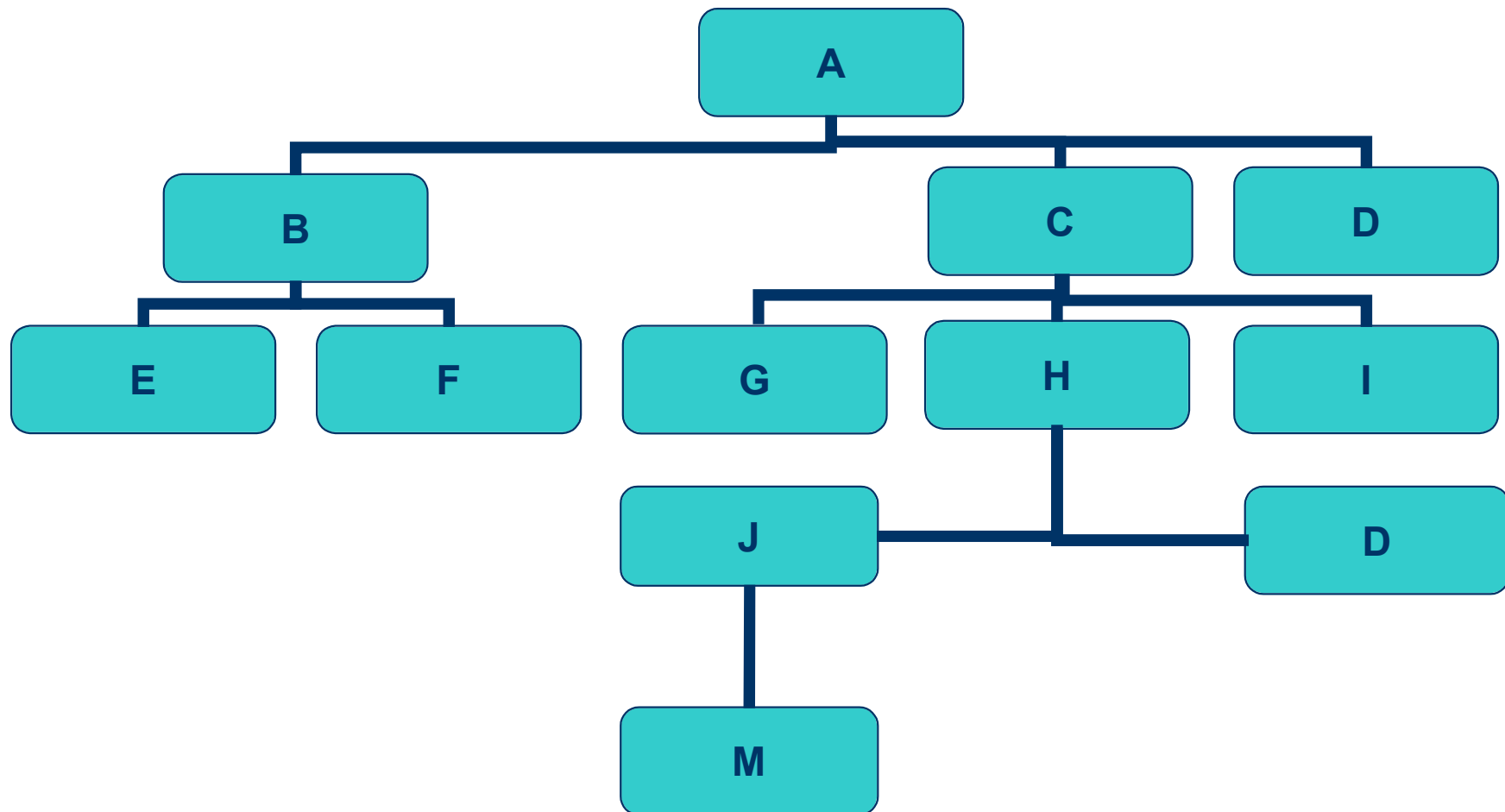
STRUKTURISANJE PROIZVODA

-jednonivojska struktura-



STRUKTURISANJE PROIZVODA

-višenovojska struktura-



MODULARNA SASTAVNICA

KOMPONENTA **A**

Komponenta JM Količina

B

C

D

KOMPONENTA **C**

Komponenta JM Količina

G

H

I

STRUKTURNA SASTAVNICA

KOMPONENTA **A**

Nivo Komponenta JM Jed. količina Ukupna količina

1	B
2	E
2	F
1	C
2	G
2	H	..		
3	J
4	M
3	D
2	I
1	D

Količinska sastavnica

KOMPONENTA **A**

Komponenta JM Količina

B
C
D
F
G
H
I
J
M

KOMPONENTA **C**

Komponenta JM Količina

D
G
H
I
J
M

SASTAVNICE

- MODULARNE
- STRUKTURNE
- KOLIČINSKE

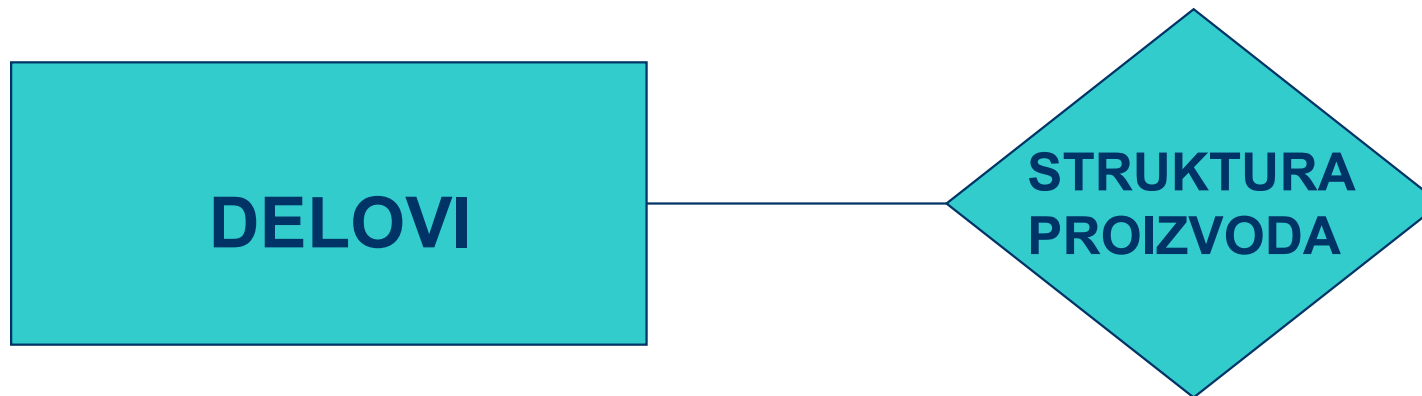
PREGLEDI UPOTREBE

- MODULARNI
- STRUKTURNI
- KOLIĆINSKI

PODSISTEM ZA UPRAVLJANJE KONSTRUKCIONOM DOKUMENTACIJOM

- Bruto potrebe
- Deo konstrukcione dokumentacije
- Unifikacija i standardizacija
- Sistem izmena

PODSISTEM ZA UPRAVLJANJE KONSTRUKCIONOM DOKUMENTACIJOM



LOGIČKA STRUKTURA BAZE PODATAKA

SADRŽAJ TABELA

DELOVI

**STRUKTURA
PROIZVODA**

C
B
D
A
F
E
G
H
I
J
M

C G ...
A D ...
H J ..
A B...
A C ...
C H ...
C I...
H D ...
J M ...