

Metode systemske analize

- Systemska analiza
- Faze systemske analize
- Model
- Modeliranje
- Systemski modeli

Funkcionalni model sistema

- predstavlja sistem kao "crnu kutiju"
- prikazuje se funkcionalnost sistema na na in kako je vide spoljni objekti
- predstavljaju se ulazi i izlazi iz sistema i funkcije koje transformišu ulaze (pobudu, stimulaciju) u izlaze
- pretstavlja model zahteva, jer treba da pokaže potpuno, precizno i nedvosmisleno kako e objekti van sistema (korisnici, akteri) koristiti posmatrani sistem.

Sistemska analiza

- Sistemska analiza je najkritičnija faza jednog projekta.
- Tokom sistemske analize treba da se shvate problemi konkretnog poslovnog sistema, definišu ciljevi za njegovo poboljšanje i definišu detaljni poslovni zahtevi, koje mora da ispuni bilo koje tehničko rešenje.
- **Sistemska analiza** je metodološki postupak dekompozicije nekog sistema na podsisteme (komponente) sa ciljem da se prouči njihov međusobni uticaj i rad.
- **U okviru sistemske analize** obavlja se (1) pregled sistema i planiranje projekta, (2) proučavanje i analiza postojećih poslovnih i informacionih sistema i (3) definisanje poslovnih zahteva i prioriteta za novi ili poboljšani sistem.

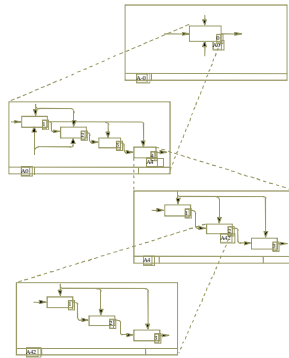
Faze sistemske analize

- **Sagledavanje projekta**
 - "Da li je projekat vredan pažnje?"
 - Definisanje cilja projekta, probleme, mogući nosioci i direktive koje aktiviraju projekat.
 - Određuje se projektni tim, učesnici, budžet projekta i raspored projekta.
 - Sistem analitičar zajedno sa vlasnicima i korisnicima sistema, menadžerom za IS i drugim osobljem IS (a) sagledavaju probleme, mogući nosioci i rešenja, (b) ugovoraju oblast projekta, (c) planiraju projekat i (d) prezentiraju projekat. Upravni odbor utvrđuje prioritete projekata IS.
- **Faza proučavanja**
 - Sistem analitičar i odgovarajući učesnici: (a) modelirati tekući sistem, (b) analizirati poslovne procese, (c) analizirati probleme i mogući nosioci, (d) utvrditi ciljeve i ograničenja poboljšanja sistema, (e) modifikovati oblast projekta i plana i (f) prezentirati zaključke i preporuke.
- **Faza definisanja**
 - Identifikuje šta novi sistem treba da radi ne uzimajući u obzir potrebnu tehnologiju, drugim rešenjima, cilj je da se definišu poslovni zahtevi novog sistema.
 - U okviru ove faze, sistem analitičar i odgovarajući učesnici: (a) grubo nacrtati poslovne zahteve, (b) modelirati zahteve poslovnog sistema, (c) izgraditi prototipove, (d) dati prioritete zahtevima poslovnog sistema i (e) modifikovati plan i oblast projekta.

Specifikacija SSA

- Izme u procesa na najvišem nivou i primitivnih procesa nalaze se dijagrami srednjeg nivoa. Broj takvih nivoa zavisi od složenosti sistema koji se analizira kao i od postupnosti u procesu dekompozicije.
- Preporu uje se da takvi nivoi sadrže 5-9 procesa.

(Ako procesa ima više, zna i da je presko en jedan apstakcioni nivo.)



Osnovni koncepti SSA

- Tok podataka (data flow)
- Proces (process)
- Skladište podataka (data store)
- Spoljni objekat (interface - external agent)
- Spojevi (junctions)

Model

- **Model** je prikaz stvarnosti. Kao što slika vredi hiljadu reči, tako i veštački i veštački sistemski modeli predstavljaju slikovite predstave stvarnosti.
- Modeli služe da bolje razumemo sisteme ili je to na in da dokumentujemo poslovne zahteve ili da postavimo tehnički dizajn.
- **Logički modeli** pokazuju *šta* je sistem i *šta* on radi. Oni opisuju sistem nezavisno od bilo koje tehničke implementacije.
- **Fizički modeli** pokazuju ne samo *šta* je sistem i *šta* on radi, već i *kako* je sistem fizički i tehnički implementiran.

Modeliranje

- **Modeliranje** je projektovanje softverskih aplikacija pre njihovog kodiranja odnosno programiranja.
- **Modeliranje podataka** je najpopularnija tehnika za izražavanje poslovnih zahteva za podacima koji će biti uskladišteni u bazu podataka sistema.
- **Modeliranje procesa** je tehnika koja se dosta praktikuje za izražavanje zahteva poslovnih procesa, tokova procesa, ulaza i izlaza.
- **Modeliranje mreže** je tehnika koja pomaže kompaniji da izrazi geografiju poslovanja koja će biti podržana od strane sistema.

Sistemska modela

- Kostur informacionih sistema identifikuje potrebu za četiri sistemska modela:
 - **PODACI** – Svi sistemi sakupljaju i skladište podatke. **Modeli podataka** (npr. dijagram objekti-veze) se koriste da modeliraju neophodne podatke za nove sisteme. Ovi modeli podataka su polazna tačka za projektovanje baze podataka.
 - **PROCESI** – **Modeli procesa** (npr. Dijagram toka podataka) se često koriste da modeliraju tokove procesa kroz poslovne sisteme. Ovi modeli procesa služe kao polazna tačka za projektovanje računarskih aplikacija i programa.
 - **INTERFEJSI** – Nijedan sistem ne egzistira izolovano od drugih ljudi, drugih sistema ili drugih kompanija. **Interfejsni modeli** se crtaju u fazi proučavanja, ali detaljnije se moraju prikazati u fazi definisanja. Interfejsni modeli opisuju ulaze (inpute) u sistem, njihove izvore, izlaze (outpute) iz sistema, njihove destinacije i deljene baze podataka. Ovi interfejsni modeli služe kao osnova za dizajniranje korisničkih i sistemskih interfejsa.
 - **GEOGRAFIJA** – Usled činjenice da današnji poslovni i informacioni sistemi imaju vešću u geografsku širinu, sistem analitičari pronalaze načine da modeliraju geografske lokacije. **Modeli mreže** služe kao polazna tačka za dizajniranje komunikacionih sistema za distribuiranje podataka, procesa i interfejsa do različitih geografskih lokacija.

Modeliranje procesa

- > Dijagram toka podataka
- > Osnovne komponente DTP-a
- > Hijerarhijska dekompozicija DTP-a
- > Sintaksa dijagrama toka podataka
- > Dijagram dekompozicije
- > Recept podataka

Proces (process)

- Proces obrade podataka je aktivna komponenta sistema, koja vrši transformaciju strukture i sadržaja ulaznog toka u izlazni tok.
- Naziv procesa treba precizno da označava funkciju koju on obavlja.

(ako analitičar ne zna da dodeli ime procesu, zna i da ne razume funkciju koju proces obavlja).



Tok podataka (data flow)

- Tok podataka je vod kroz koji protiču grupe podataka poznate strukture.
- Tok podataka ostvaruje vezu između ostalih komponenti sistema i na dijagramu toka podataka predstavlja se imenovanom, orijentisanom linijom.
- Svaki tok podataka u dijagramu toka podataka mora imati jedinstveno ime koje odražava značenje podataka koje on nosi. Izuzetak su tokovi koji idu prema, odnosno od skladišta podataka, koji ne moraju biti imenovani.



Skladište podataka (data store)

- Skladište podataka (kartoteka, fascikla, datoteka) služi za uvanje podataka
- Definiše se kao tok podataka u mirovanju
- Veza sa primitivnim funkcijama isključivo preko tokova podataka.
- Skladište podataka treba da omogući:
 - tok podataka KA skladištu označava operaciju održavanja, tj. ubacivanja, izbacivanja i promene sadržaja;
 - tok podataka OD skladišta označava korišćenje skladišta za izveštavanje.



Spoljni objekat (interface - external agent)

- Spoljni objekat (interfejs) je objekat van konteksta posmatranog sistema koji se javlja kao izvor ili ponor tokova podataka.



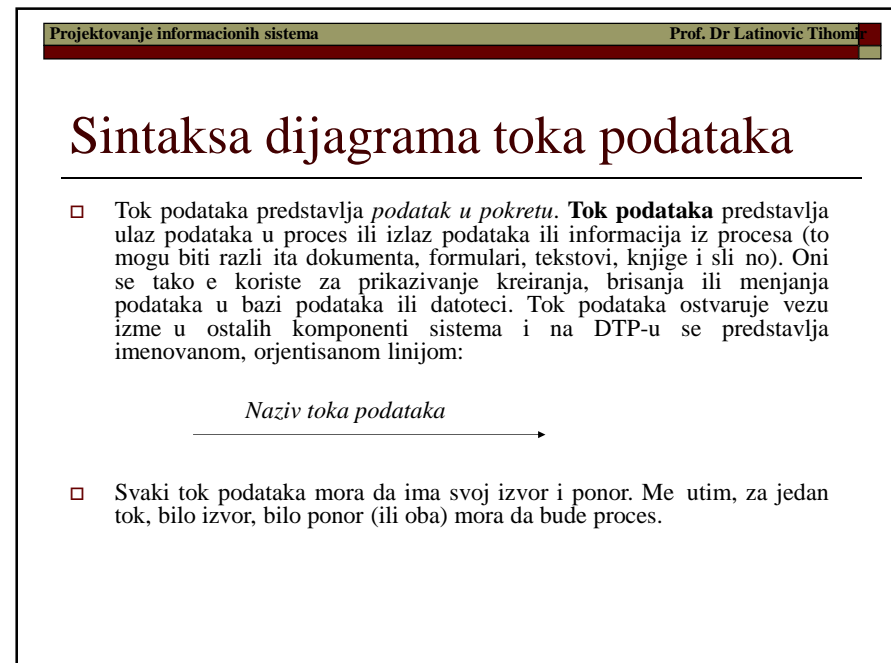
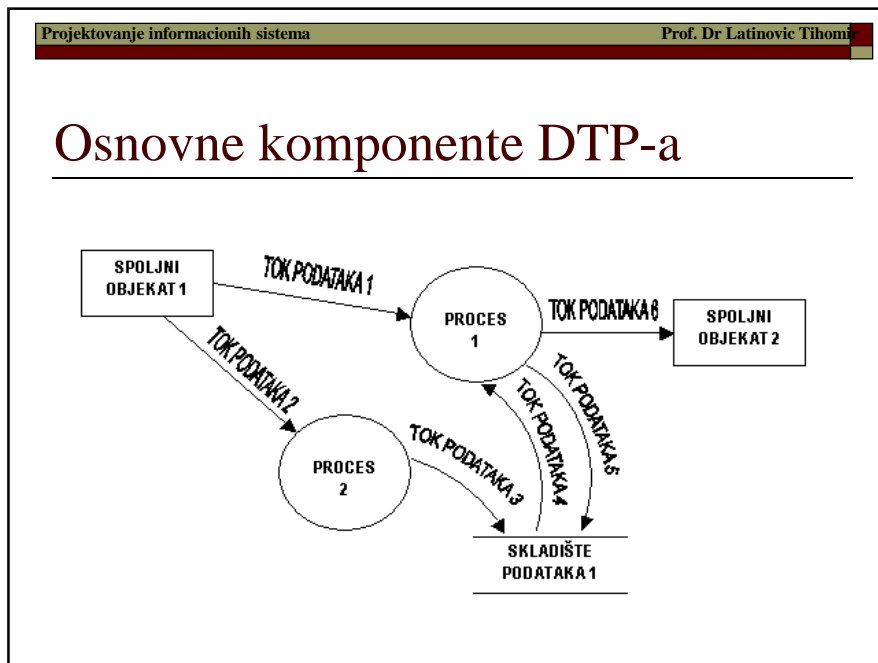
Spojevi (junctions)

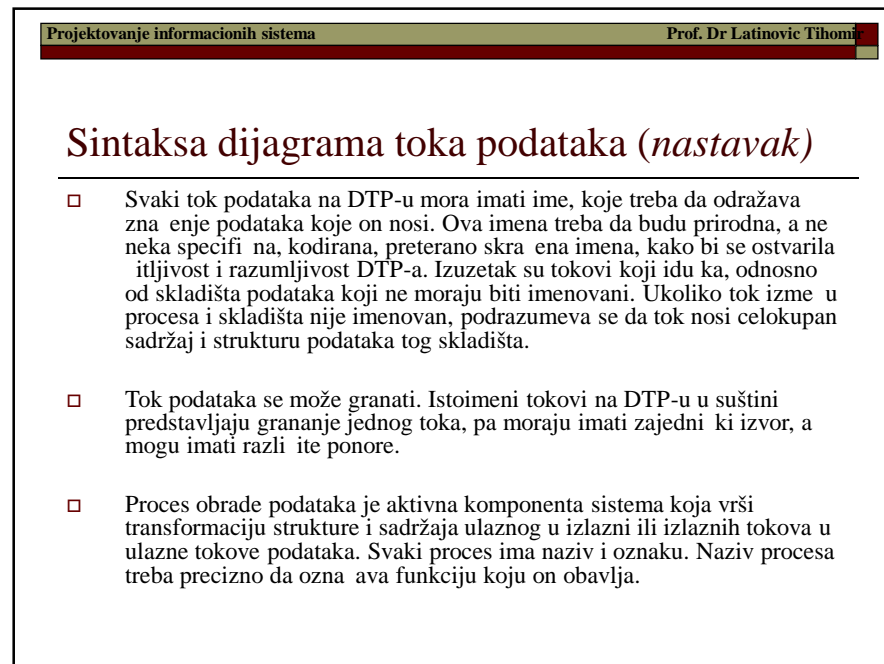
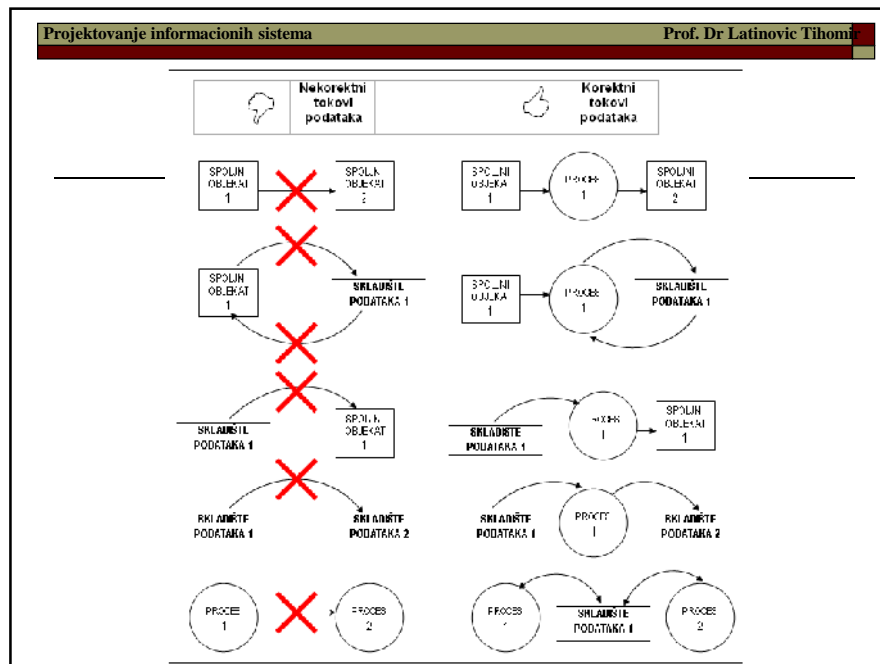
- Spojevi pomažu u organizovanju i pojašnjavanju dijagrama toka podataka.
- Postoje **divergentni** spojevi koji dele tok podataka na dva ili više izlaza, kao i **konvergentni** spojevi, koji spajaju dva ili više toka podataka u jedan.
- Spojevi sa jednim ulazom i jednim izlazom zovu se linearni spojevi.



Dijagram toka podataka

- **Modeliranje procesa** je tehnika koja organizuje i dokumentuje procese sistema i/ili implementira logiku, politike i procedure sistema.
- **Dijagram toka podataka (DTP)** je alat koji opisuje tokove podataka kroz sistem i procese koji se izvršavaju u sistemu. Sadrži četiri osnovne komponente:
 - **proces** (*processes*) obrade podataka, koji predstavljaju aktivne komponente sistema (grafi ki simbol: krug);
 - **spoljne objekte ili spoljne agente** (*external agents*) sa kojima sistem komunicira (grafi ki simbol: pravougaonik);
 - **skladišta podataka** (*data stores*) koje procesi koriste i/ili ažuriraju (grafi ki simbol: dve paralelne linije) i
 - **tokove podataka** (*data flows*) koji povezuju ostale komponente sistema u celinu (grafi ki simbol: usmerena linija).



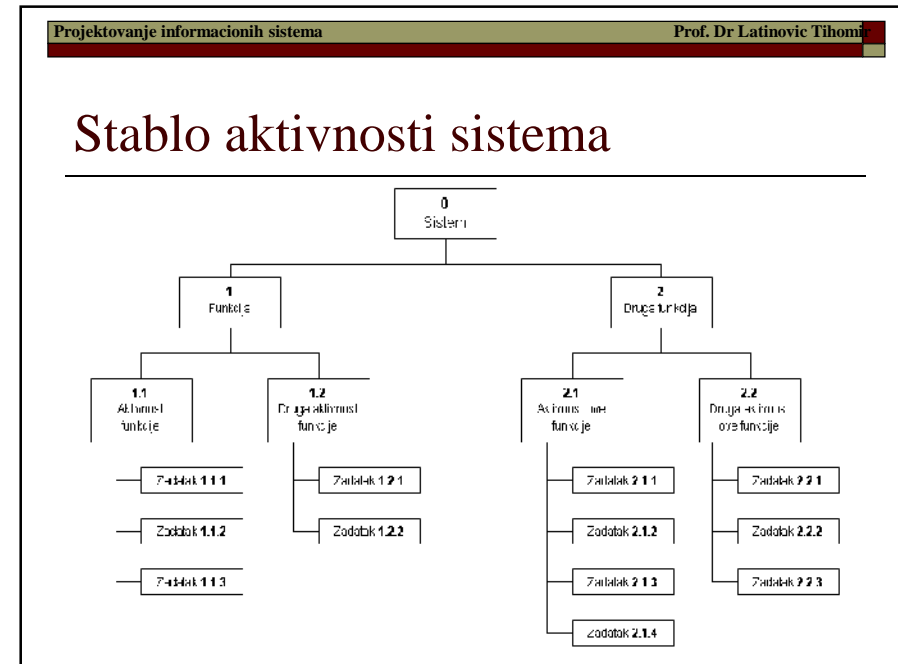
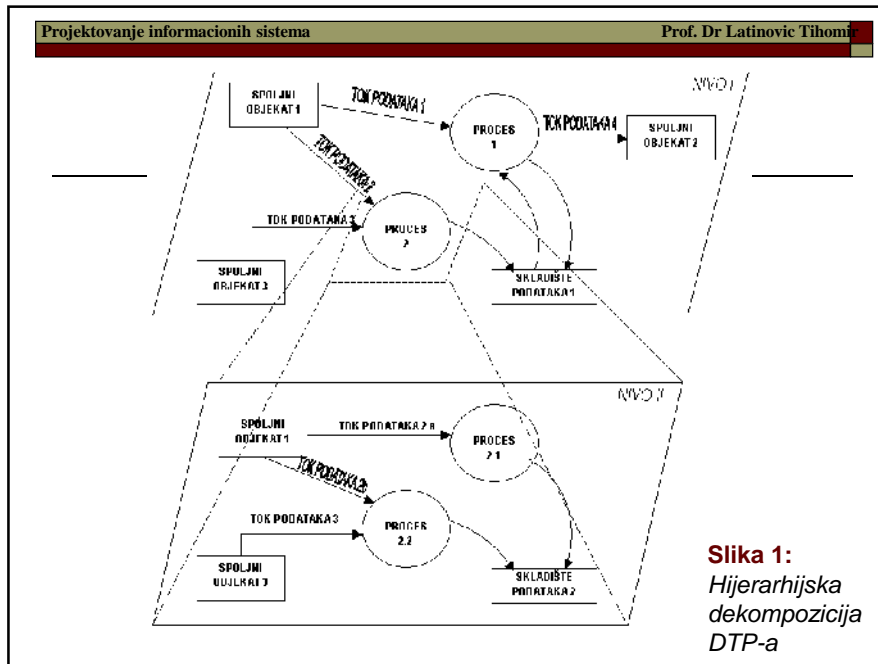


Sintaksa dijagrama toka podataka (*nastavak*)

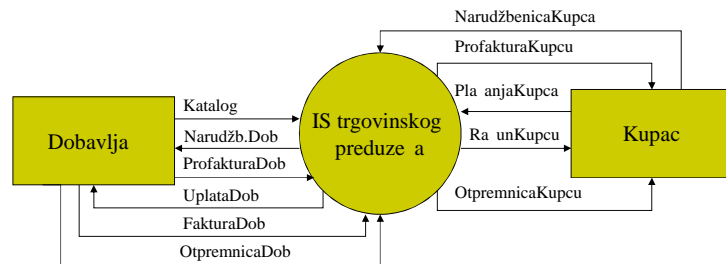
- Svaki proces mora da ima barem jedan ulazni i barem jedan izlazni tok podataka. Proces koji ima ulaze, a nema izlaze je tzv. crna rupa, jer izgleda kao da je podatak jednostavno nestao. Proces koji ima izlaze, a nema ulaze, je takođe nemoguć.
- Svako skladište podataka bi trebalo da ima barem jedan ulazni i barem jedan izlazni tok. Međutim, dozvoljava se da skladište nema ulazni tok, podrazumevajuć i da se formira i ažurira u nekom drugom sistemu, odnosno da nema izlazni tok, podrazumevajuć i da posmatrani sistem formira i ažurira skladište koje se koristi u nekom drugom sistemu.
- Svaki spoljni objekat (spoljni agent) mora da ima barem jedan, bilo ulazni, bilo izlazni tok podataka, inače bi bio izolovan od ostalog dela sistema.
- Da bi se izbeglo nepotrebno presecanje linija, dozvoljava se da se skladište ili spoljni objekat, na jednoj slici, višestruko ponovi.

Hijerarhijska dekompozicija DTP-a

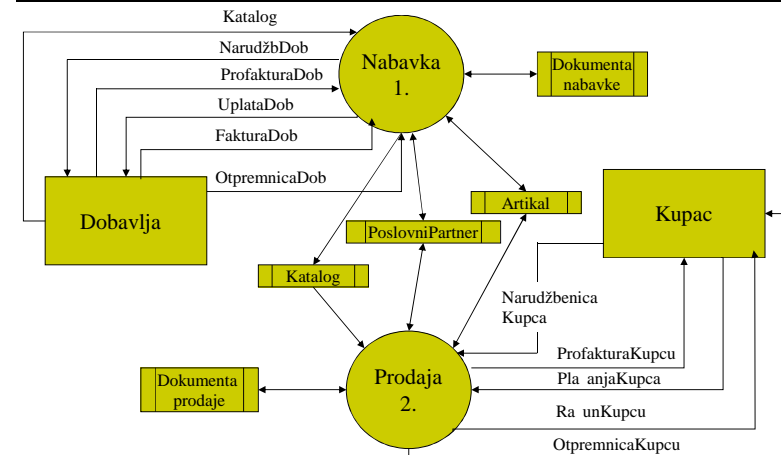
- **Dekompozicija** je način razlaganja sistema na njegove komponente podsysteme, procese i podprocesse.
- Pri dekompoziciji DTP-a moraju se poštovati sledeća pravila:
 - Dijagram najvišeg nivoa, koji po pravilu sadrži samo jedan proces koji predstavlja ceo IS, zatim spoljne objekte sa kojima IS komunicira i odgovarajuće tokove podataka naziva se **dijagram konteksta**.
 - Dijagram prvog nivoa predstavlja dekompoziciju dijagrama konteksta. Procesi se označavaju brojevima 1,2,3,
 - Svaki proces sa dijagrama prvog nivoa se dalje dekomponuje do nivoa primitivnih procesa. Procesi na dijagramima nižih nivoa, povlače sa sobom brojnu oznaku nadređenog procesa.
 - Uobičajeno je da se zatim celokupan sistem predstavi dijagramom dekompozicije odnosno stablom aktivnosti. **Dijagram dekompozicije** prikazuje *top-down* (sa vrha na dole) funkcionalnu dekompoziciju i strukturu sistema.
 - Pored procesa, mogu se dekomponovati i tokovi i skladišta. Njihov opis se detaljno daje u rešeniku podataka.
 - Najvažnije pravilo koje se mora poštovati pri dekompoziciji procesa je pravilo balansiranja tokova.

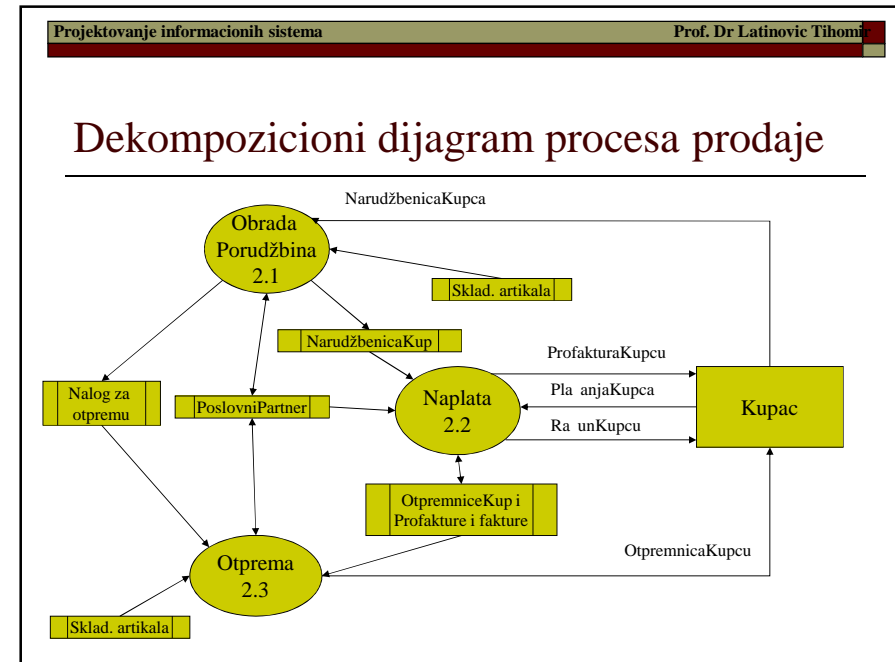
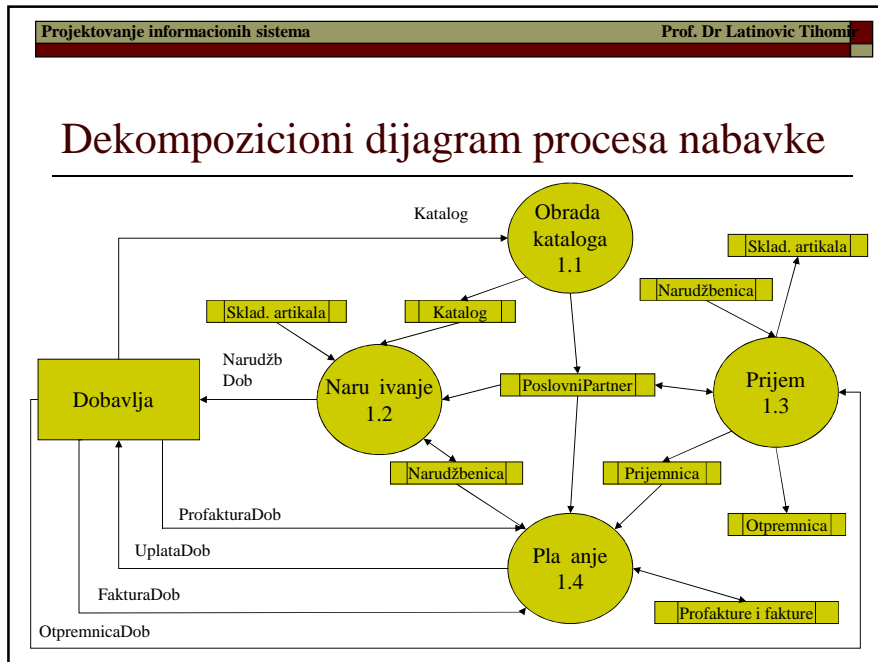


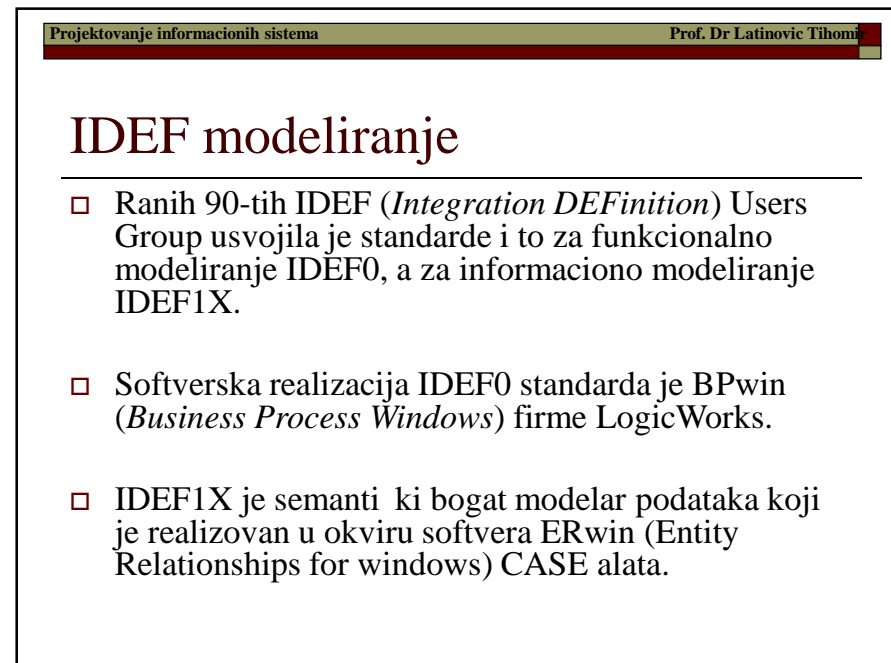
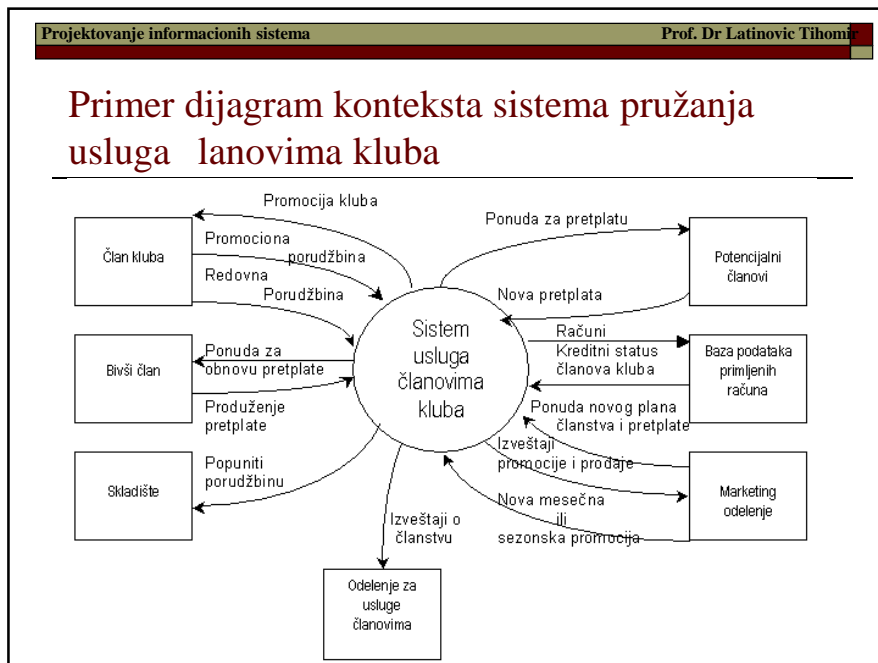
Primer dijagram konteksta manjeg trgovinskog preduzeća



Dijagram prvog nivoa







IDEF0

- Funkcionalno modeliranje omogućava dekomponovanje poslovnih funkcija i planiranje potrebnih resursa za realizaciju funkcija.
- Funkcionalno modeliranje je vezano za korišćenje IDEF0 tehnike.
- IDEF0 funkcionalni model se sastoji od hijerarhijskog niza dijagrama koji postepeno prikazuju sve više detalja o funkcijama i njihovoj povezanosti sa ostalim delovima sistema.


Struktura grafičkog jezika IDEF0

- **Pravougaonik** – predstavlja aktivnosti definisane kao funkcije, procese i transformacije. Svaki pravougaonik ima naziv i broj. Za naziv aktivnosti se koristi aktivan glagol ili glagolska fraza koja opisuje funkciju.



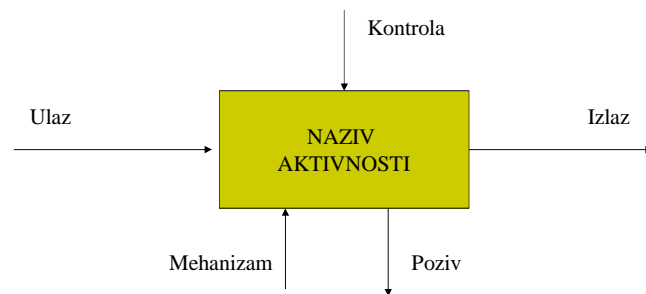
NAZIV AKTIVNOSTI

- **Strelice** – predstavljaju podatke ili objekte vezane za aktivnosti.



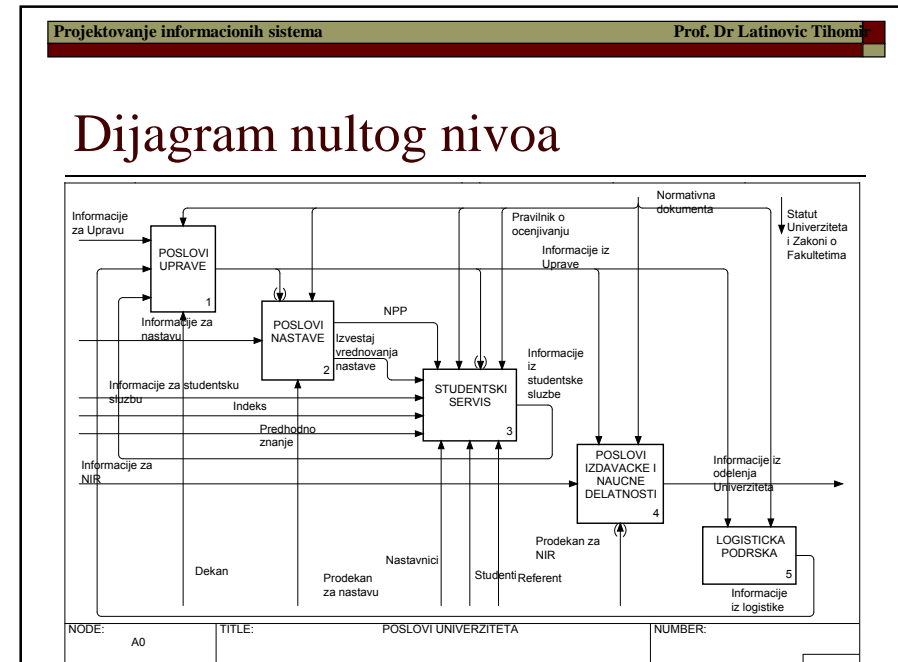
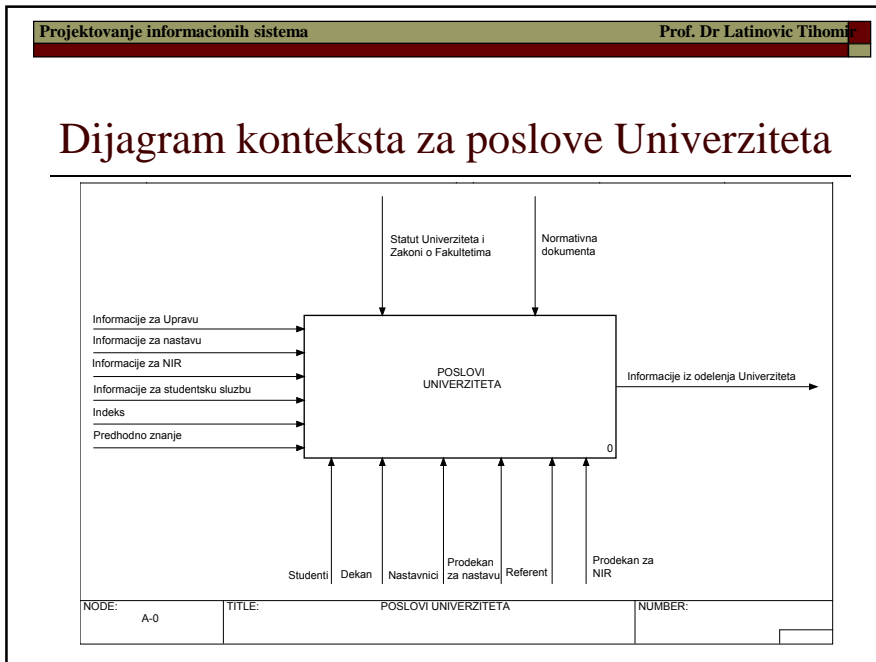
Naziv toka

Semantika grafičkog jezika IDEF0

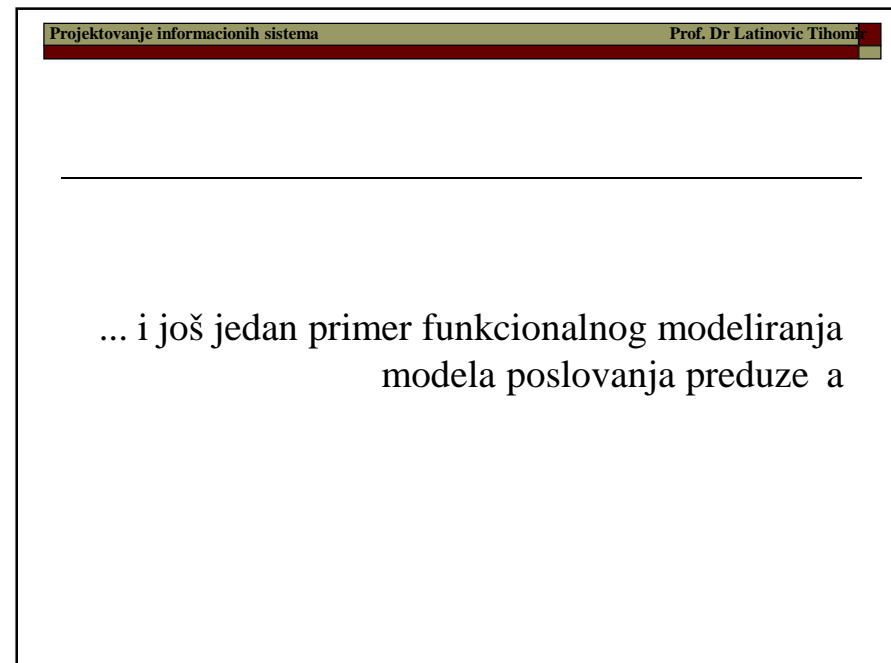


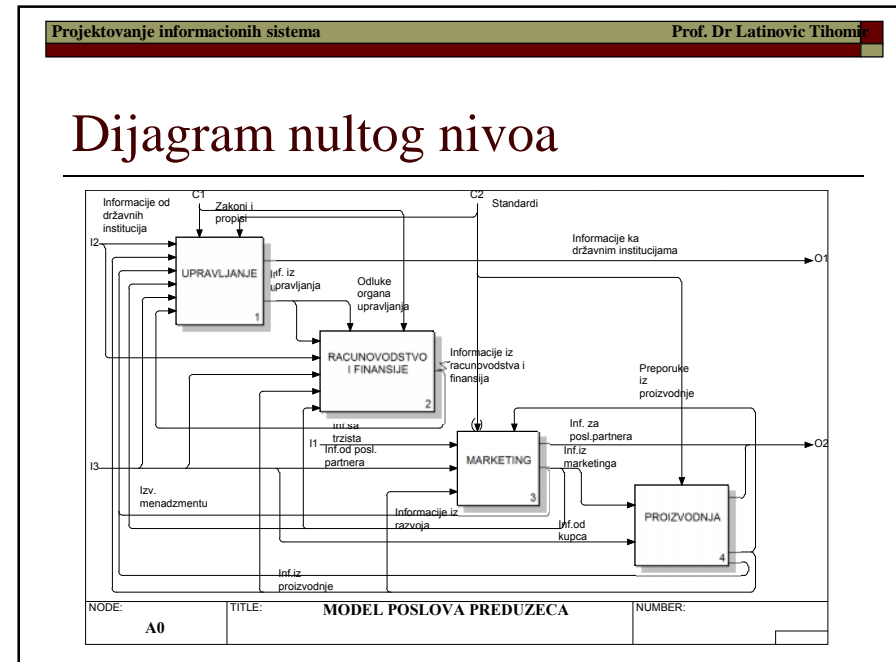
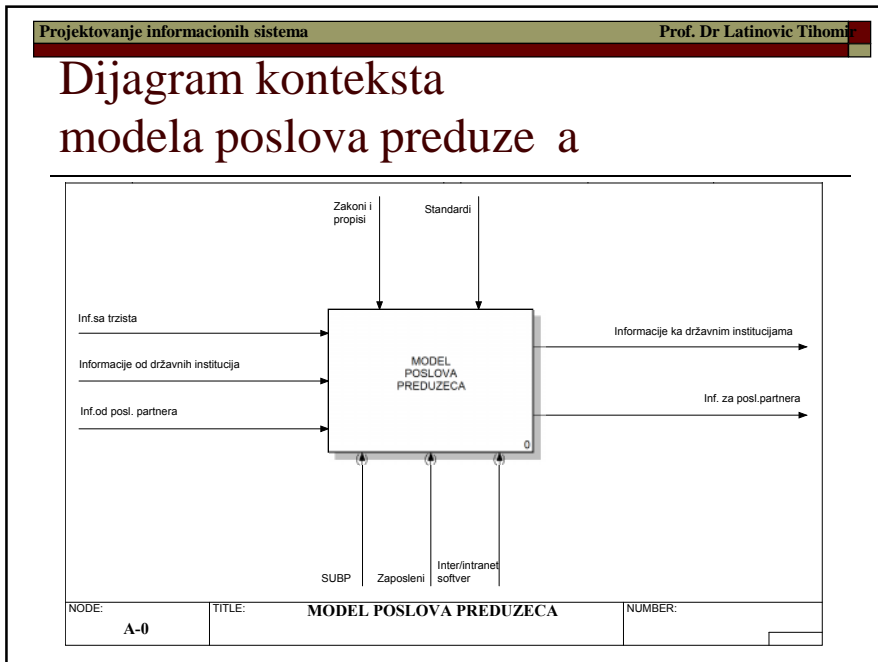
Semantika IDEF0

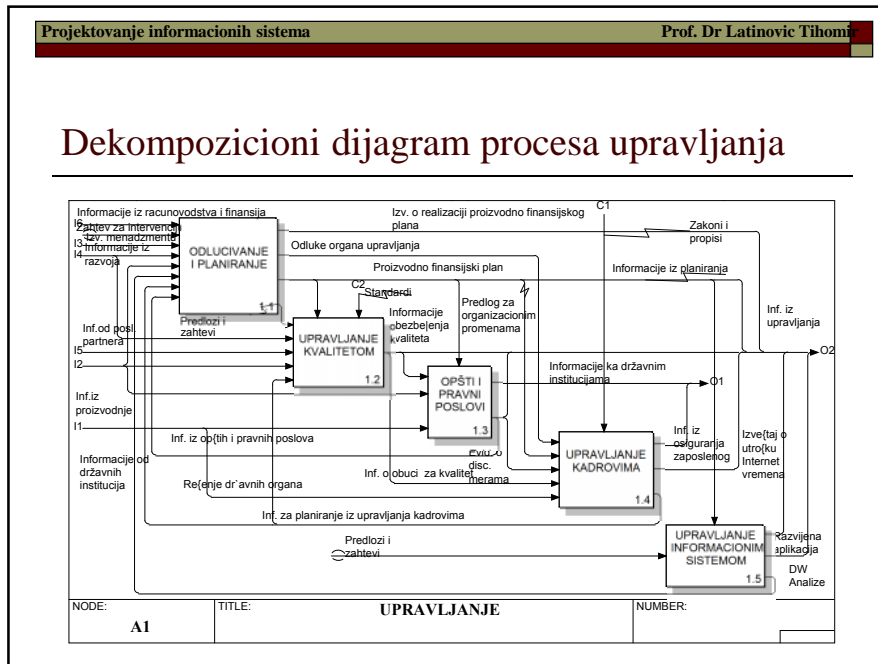
- **Ulazna strelica** – predstavlja materijal ili informaciju koja se koristi ili transformiše radi definisanja izlaza.
- **Kontrolne strelice** – regulišu, odnosno odgovorne su za to kako, kada i da li će se aktivnost izvesti. Uslovi izvođenja aktivnosti. Kontrole su često u obliku pravila, politika, procedura ili standarda. One utiču na aktivnost, ali ne mogu da budu transformisane ili upotrebljene.
- **Izlazne strelice** – predstavljaju materijal ili informacije stvorene aktivnošću.
- **Strelice mehanizama** - izvori koji izvode aktivnosti. Mehanizmi mogu biti ljudi, mašine, oprema, tehnologija i dr. sve ono što obezbeđuje energiju potrebnu za izvođenje aktivnosti.
- **Strelica poziv** - specifični slučaj strelice mehanizma i označava da pozivaju i pravougaonik nema vlastiti detaljniji dijagram, već daje detaljniji prikaz izveden na nekom drugom pravougaoniku u istom ili nekom drugom modelu.



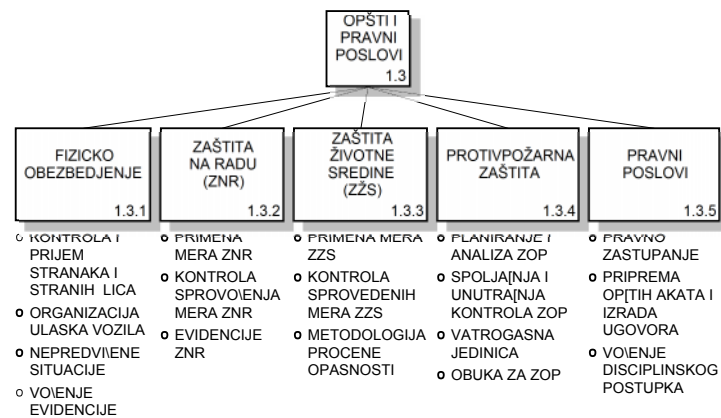




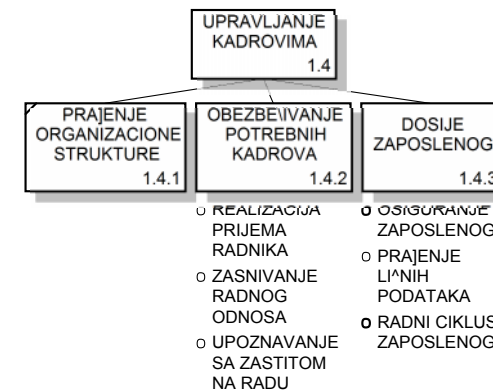


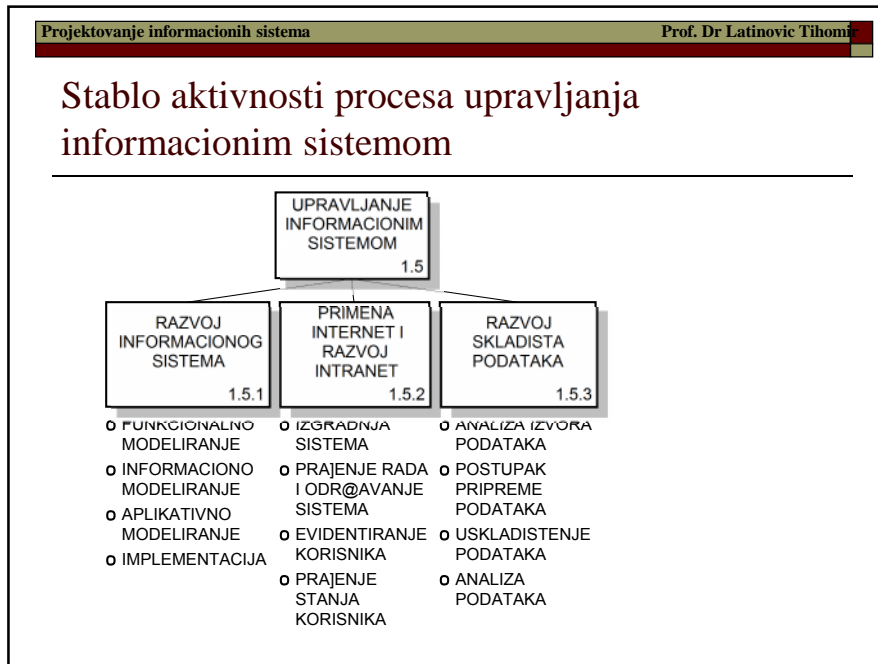


Stablo aktivnosti procesa opštih i pravnih poslova



Stablo aktivnosti procesa upravljanja kadrovima

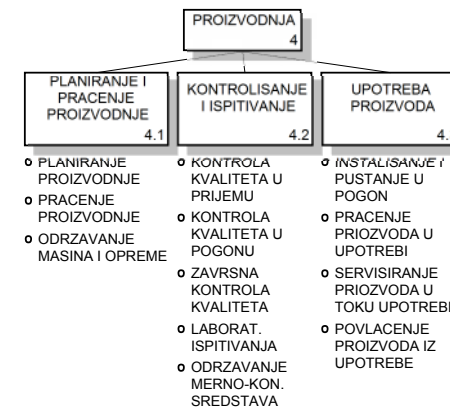




Stablo aktivnosti procesa marketinga



Stablo aktivnosti procesa proizvodnje



Rezik podataka SSA

- Rezik podataka opisuje sadržaj i strukturu svih tokova i skladišta podataka.
- **Polje** je elementarna struktura koja se dalje ne dekomponuje i koja ima svoju vrednost.
- Primeri polja u indeksu su BrojIndeksa, ImePrezime, Status, NazivPredmeta, Ocena i dr.
- Polja svoje vrednosti uzimaju iz skupova vrednosti koji se nazivaju domenima. Domeni mogu biti:
 - «Predefinisani», odnosno standardni programsko-jezički domeni, kao što su INTEGER (celobrojna vrednost), CHARACTER (karakter), REAL (decimalna vrednost), LOGICAL (logična vrednost: true ili false), DATE (datum) itd.
 - «Semantički», kada se definišu preko svog imena ili predefinisanih domena ili eventualno ograničavaju vrednosti predefinisanih domena. Na primer, domen SEMESTAR se definiše kao SEMESTRI DEFINED_AS INTEGER (1) BETWEEN 1,8.

Logički tipovi podataka

Logički tip podatka	Logičko značenje
NUMBER	Bilo koji decimalni ili celobrojni broj.
TEXT	String karaktera uključujući i brojeve sa kojima se ne može obavljati aritmetička operacija ili operacija poređenja.
MEMO	Kao i TEXT ali sa neodređenom veličinom.
DATE	Bilo koji oblik datuma.
TIME	Bilo koji oblik vremena, odnosno časova.
YES/NO	Atribut koji može da ima samo ove dve vrednosti.
VALUE SET	Skraćeni skup vrednosti. (Na primer, JR=junior, SR=senior).
IMAGE	Bilo koja slika.

Prosta i složena ograničenja

- Pored ograničenja na vrednosti polja, odnosno vrednosti domena, definišu se i druga, bilo prosta bilo složena ograničenja. Lista prostih ograničenja su:
 - “Operator konstanta”, gde je operator bilo koji operator poređenja (npr. <, >, =, ≥, ≤), a konstanta je neka definisana vrednost iz datog domena. Na primer, STAROST: INT(2) < 65.
 - “BETWEEN konstanta, konstanta”, gde su konstantne vrednosti iz datog domena. Na primer: SEMESTRI: INTEGER (2) BETWEEN 1,10.
 - “IN (lista vrednosti)”, gde se lista vrednosti formira od konstanti iz odgovarajućeg domena. Na primer: OCENA INT(2) IN (5,6,7,8,9,10).
 - “NOT NULL”, se specifikira za ona polja koja ne mogu da stoje bez neke vrednosti, znači moraju uvek da imaju neku vrednost. Na primer: BROJ_INDEKSA: CHARACTER (7) NOT NULL, što znači da polje broj indeksa ne može da stoji prazno, već se mora uneti neka vrednost.
- Složena ograničenja se formiraju od prostih ili drugih složenih ograničenja vezujućih ih logičkim operatorima AND, OR i NOT.

Agregacija komponenti

- **Struktura** tokova podataka i skladišta predstavlja neku kompoziciju polja, odnosno konstrukciju čije su komponente polja.
- Konstrukcije kojim se od komponenta gradi struktura mogu biti:
 - *Agregacija komponenti* koja predstavlja složenu strukturu liste komponenti. Označava se u “špicastim” zagradama - <a,b,c>. Na primer:


```

STUDENT:  <      BROJ_INDEKSA,
              NAZIV_STUDENTA,
              STAROST,
              SEMESTAR,
              PROSENA_OCENA
          >
```

Eksluzivna specijalizacija (unija) komponenti

- *Eksluzivna specijalizacija (unija) komponenti*, označava da se u strukturi pojavljuje eksluzivno jedna od navedenih komponenti, ili a ili b ili c. Ukoliko se tu nalazi samo jedna komponenta to znači da se ona može, a ne mora pojaviti. Označava se u uglastim zagradama – [a,b,c]. Na primer:

```

PROIZVOD: <      ŠIFRA_PROIZVODA,
              NAZIV_PROIZVODA,
              [KOLI  INA_PORUDŽBINE, STOPA_AMORT]
              >

```

- Ukoliko je proizvod osnovno sredstvo, onda će se pojaviti polje STOPA_AMORT, a ukoliko je proizvod materijal za proizvodnju, onda će se pojaviti polje KOLI INA_PORUDŽBINE.

Neeksluzivna specijalizacija (unija) komponenti

- *Neeksluzivna specijalizacija (unija) komponenti*, označava da se u odgovarajućoj strukturi pojavljuje bilo samo jedna, bilo dve, bilo sve komponente. Označava se u kosim zagradama - /a,b,c/. Na primer:

```

STUDENT_ZAHTEV:  / ZAHTEV_ZA_UVER_STATUS,
                  ZAHTEV_ZA_UVER_POL_ISPIT/

```

- Student može da podnese ili zahtev za uverenje o statusu (redovan ili vanredan) ili zahtev za uverenje o položenim ispitima ili oba.
- *Relevantni podatak* sadrži četiri osnovna segmenta za:
 - opis struktura skladišta i tokova.
 - opis polja,
 - opis semantičkih domena i
 - definiciju logičkih funkcija preko kojih se iskazuju složenija ograničenja.

Primer jednog rešenika podataka

FIELDS

NAZIV POLJA	DOMEN	OGRAĐENJE	AKCIJA
SIF_PROD	Šifra	not null	nekorektna šifra_prod
MATIČNI_BROJ_VLASNIKA	Matični_lični_broj	not null	nekorektan mib
IME_VLASNIKA	Ime		
ADRESA_VLASNIKA	Adresa		
TELEF_VLASNIKA	Telefon		
PTT_BR	Pošanski_broj	not null	nekorektan ptt_br
NAZIV_MESIA	Naziv		
VRSTA_KLIJENTA	char (1)	in ('K', 'P', 'O')	vrsta klijenta može uzeti vrednosti K-kupac, P-prodavaac, O-oba
SIF_TIPA_VRSTE	Šifra	not null	nekorektna šifra_vr
SIF_NEKRETNINE	Šifra	not null	nekorektna šif_nekretnine
LOK_NEK	Lokacija		
POV_NEK	Površina		
SPRATNOST	int (2)		
STAROST	int (3)		
POSED_TEL	char (1)	in ('D', 'N')	pcgrešan posed_tel

Primer jednog rešenika podataka - nastavak

DOMAINS

PREDEFINISANI DOMEN	NAZIV_DOMENA
Pošanski_broj	integer (5)
Naziv	char (30)
Ime	char (30)
Matični_lični_broj	char (13)
Šifra	integer (10)
Adresa	char (30)
Telefon	integer (15)
Lokacija	char (20)
Površina	integer (5)
Datum	date
Vreme	char (10)
Cena	real (10,2)