

Univerzitet u Tuzli
Mašinski Fakultet Tuzla

Ahmed Kovačević

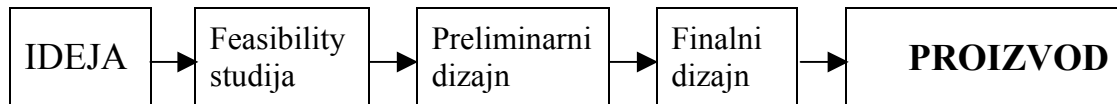
Osnovi Projektovanja

skripta za studente IV i V godine
Energetskog i Proizvodnog smjera

Tuzla, Oktobar 2002

1 DIZAJN PROCES - Proces projektovanja

Cilj procesa projektovanja je da se na efikasan i pogodan način nađe put od IDEJE do PROIZVODA. Da bi se od kvalitetne ideje došlo i do kvalitetnog proizvoda potrebno je pri projektovanju držati se određenog redoslijeda aktivnosti. Taj redoslijed aktivnosti je prikazan slijedećim blok dijagramom:



slika 1: Blok – dijagram procesa projektovanja (design) od Ideje do Proizvoda

1.1.1.1 Faza 1: IDEJA ili 'Prepoznavanje potreba'

Ideja o razvoju novog proizvoda ili unapređenje postojećeg se javlja kao posljedica potrebe industrije, vlade, privatnog sektora ili neke druge instance, a u cilju smanjenja troškova bilo investicionih ili eksploatacionih i povećanja efikasnosti, komfora i slično. Ova faza je izuzetno važna i zahtijeva dobro poznavanje tržišta, potreba privrede, propisa kao i osnovnih operacionih principa i načela. Ideja je obično posljedica 'individualnog' rada ali može nastati i kao posljedica aktivnosti grupe ljudi - tima.

1.1.1.2 Faza 2: Feasibility study ili 'Kreiranje i prezentacija mogućih rješenja'

Ova faza se sastoji od dva nivoa. Prvi nivo je *Kreiranja mogućih načina realizacije ideje* nastale u prethodnoj fazi. Prijevod na engleski je 'Feasibility study', termin koji je i kod nas uobičajen. Prema teoriji projektovanja, koja se danas sve češće svrstava u posebnu naučnu oblast dizajna (Design science), ova faza se može uspješno realizovati samo grupnom aktivnošću (timski rad) i karakteriše se kao 'Brainstorming' faza ili faza razmišljanja. U toj fazi se sve ideje koje se mogu javiti stavljaju u razmatranje te se ocjenjuje da li ih je moguće realizovati ili ne. Ova faza zahtijeva multidisciplinarnost u timu koji se tim bavi. To znači da stručnjaci iz različitih oblasti moraju učestvovati u procesu da bi on bio uspješan. Ideje koje se pojave u toku tog procesa se također moraju i prezentirati kako bi se proces odlučivanja i daljeg projektovanja mogao adekvatno obaviti.

1.1.1.3 Faza 3: 'Preliminarni dizajn'

Na osnovu prezentacija iz prethodne faze, u toku ove faze se vrši izbor rješenja za koje će se raditi razrada i kasnije provesti kompletan proces i realizacija. Izvodi se preliminarna konstrukcija praćena modelom, sklopnim crtežom i analizom mogućnosti proizvodnje.

1.1.1.4 Faza 4: Konačni 'Finalni dizajn'

Finalna konstrukcija i razrada detalja. Kao rezultat ove faze dobije se radionička dokumentacija praćena ostalom dokumentacijom za proizvodnju, marketing, ekonomsku analizu i slično.

1.1.1.5 Faza 5: 'Proizvodnja'

Konstruktivski tim treba da prati proizvodnju, testiranje proizvoda i njegov marketing u cilju unapređenja procesa konstruisanja i samog proizvoda.

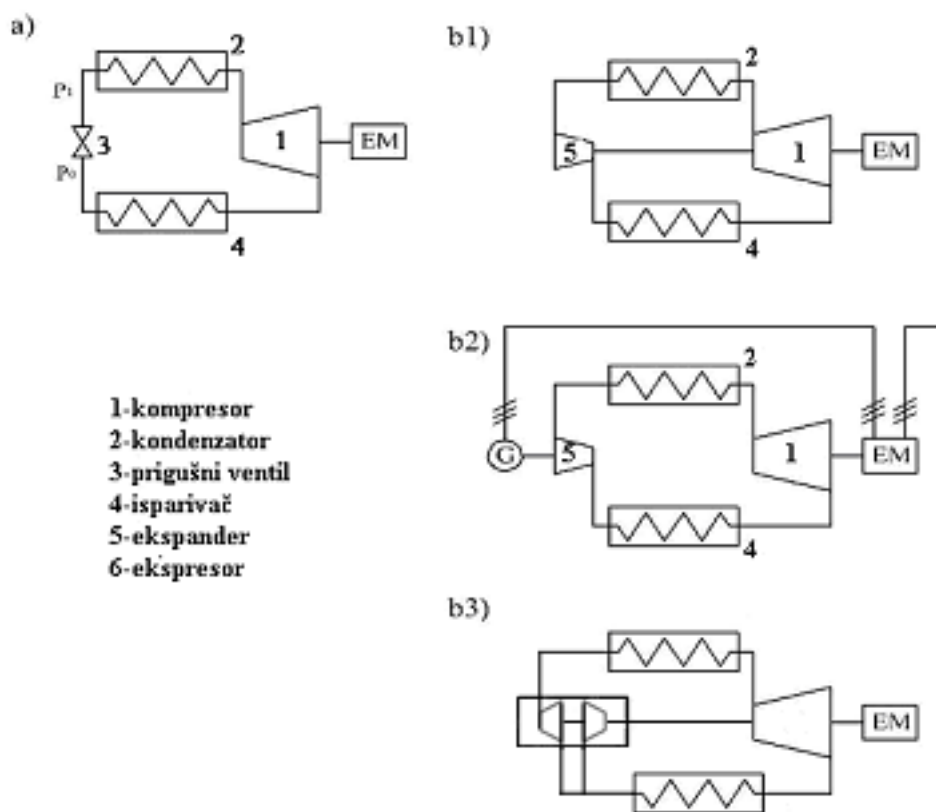
O svim ovim faza projektovanja biće više riječi u nastavku teksta. Objašnjenje ovih faza procesa konstruisanja biće praćeno primjerima koji slijede.

1.2 Faza 1: Ideja

Razrada ideje će biti prikazana kroz primjer.

Primjer 1: Zamjena prigušnog ventila u rashladnom postrojenju ekspanzionim uređajem u cilju povećanja stepena iskorištenja postrojenja.

Prigušnim ventilom se u rashladnom procesu vrši smanjenje pritiska radnog medija, nakon čega se kondenzuje u kondenzatoru. Kako se proces u prigušnom ventilu vrši pri konstantnoj entalpiji ($di=0$), to je očito da se pomenuto smanjenje pritiska vrši uz određene gubitke energije. Iz tog razloga se dosta truda ulaže kako bi se prigušni ventil u rashladnom postrojenju zamijenio ekspanderom. Ekspander bi imao ulogu da dio, inače izgubljene energije vrati nazad u proces i time smanji gubitke usljed prigušivanja. Upravo način vraćanja sačuvane energije u proces predstavlja najveći problem. U ovom primjeru su predložene tri ideje:



slika 2: Razrada varijanti rješenja za poboljšanje rashladnog procesa

- b1) prijenos energije od ekspandera do kompresora mehanički (vratilom);
 - b2) uvođenje generatora, kojim bi se energija ekspandera pretvorila u električnu i njome smanjila potrošnja energije iz distribucijske mreže i
 - b3) uvođenje uređaja, koji bi vršio funkciju i kompresora i ekspandera (tzv.ekspresor).
- U ovom uređaju se kondenzat ekspankira na pritisak u isparivaču čime se dobije rad na vratilu. Taj rad se prenosi direktno u kompresor koji komprimira dio gasa i time smanjuje rad glavnog kompresora.

Kod prve varijante problem se javlja u sinhronizaciji rada ekspandera i kompresora, pa težnja za iskorištenjem može uzrokovati komplikacije u izvođenju cjelokupnog procesora. Druga izvedba bi predstavljala veliku financijsku investiciju, zbog većeg broja elemenata uključenih u izvedbu. Sa druge strane dolazi do višestrukog mijenja vida energije čime se količinama sačuvane energije značajno smanjuje. Iz ovih razloga kao najbolje rješenje, sa aspekta energetske bilansa postrojenja i inicijalnih troškova, se nameće treća izvedba.

Prikazano na prethodnoj slici (b1, b2, b3) predstavlja načelnu razradu mogućih načina za praktičnu primjenu ideje. Poslije načelne razrade mogućih verzija nastupa takozvana feasibility study (studija podobnosti) svake od ponuđenih verzija.

1.3 Faza 2: Feasibility study - 'Kreiranje i prezentacija mogućih rješenja'

Ova faza predstavlja nastavak prethodne faze. Sastoji se od dva dijela: kreiranje i razrada. Ova faza je skupa sa prethodnom 'brainstorming' faza.

Feasibility study se svodi na analizu:

- 1) Ekonomskih efekata proizvoda, u koju je uključeno ispitivanje tržišta, utvrđivanje preliminarne cijene proizvoda i njeno upoređivanje sa dosadašnjim cijenama, ispitivanja dostupnosti potrebnih repromaterijala itd.
- 2) Razrada podobnosti različitih varijanti prema raznim kriterijima pomoću 'decision matrix' (tabela odlučivanja):

weighting factor (faktor važnosti)

| Kriteriji Varijanta | TD iskorištenje | | Cijena | | Težina, tehnologija proizvodnje, nabavka sirovine..... | Σ |
|------------------------|--------------------|--------|--------|---------|--|-----|
| | 6 | | 10 | | | |
| b1 | 5 | 5·6=30 | 3 | 3·10=30 | | 60 |
| b2 | 5 | 30 | 5 | 50 | | 60 |
| b3 | 8 | 48 | 9 | 90 | | 148 |

slika 3: Tabela odlučivanja 'Decision matrix'

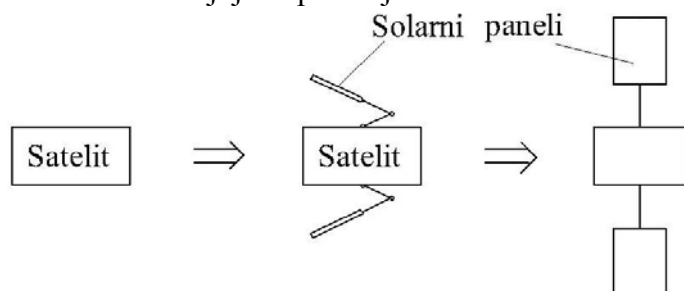
U tabelu odlučivanja upisujemo određenu brojčanu vrijednost, od 0 do 10, kojom ocjenjujemo pogodnost svake varijante proizvoda u odnosu na unaprijed određene kriterije (pri čemu 0 – znači veoma nepogodno, a 10 – veoma pogodno). Varijantu koju izaberemo odlučit će suma proizvoda ovih brojeva sa faktorima važnosti, za svaki od predviđenih kriterija (posljednja kolona) za pojedine varijante. Naravno, treba izabrati onu verziju, za koju je ova suma najveća. Pomoću faktora važnosti se vrši diferencijacija između izabranih kriterija, prema njihovoj važnosti za tržišnu podobnost proizvoda.

Važno je napomenuti da su u tabeli odlučivanja sve vrijednosti date kao pretpostavka, a ne kao rezultat nekih testiranja ili slično.

Za takvo nešto potrebno je veliko iskustvo!!!

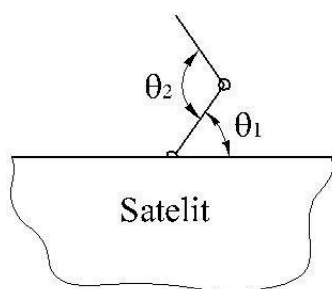
Primjer 2: Projektovanje mehanizma zgloba za otvaranje solarnih panela na satelitu.

Energetsko napajanje komunikacionog satelita u orbiti se vrši pomoću solarnih panela. Solarni paneli su pri lansiranju zbog svoje veličine sklopljeni, pa ih je po smještanju satelita u orbitu potrebno otvoriti, odnosno postaviti u radni položaj. Glavni problemi kod ovog primjera su: kako osigurati energiju za pokretanje mehanizma, kako osigurati potrebnu pouzdanost i sigurnost; i kako osigurati zaustavljanje mehanizma u tačno određenom krajnjem položaju?



slika 4: Otvaranje krila sa solarnim panelima

Da bi se ovaj problem detaljno razmotrio potrebno je postaviti uslove kretanja poluga mehanizma za otvaranje krila:



$$0 \leq \theta_1 \leq \frac{\pi}{2}$$

$$0 \leq \theta_2 \leq \pi$$

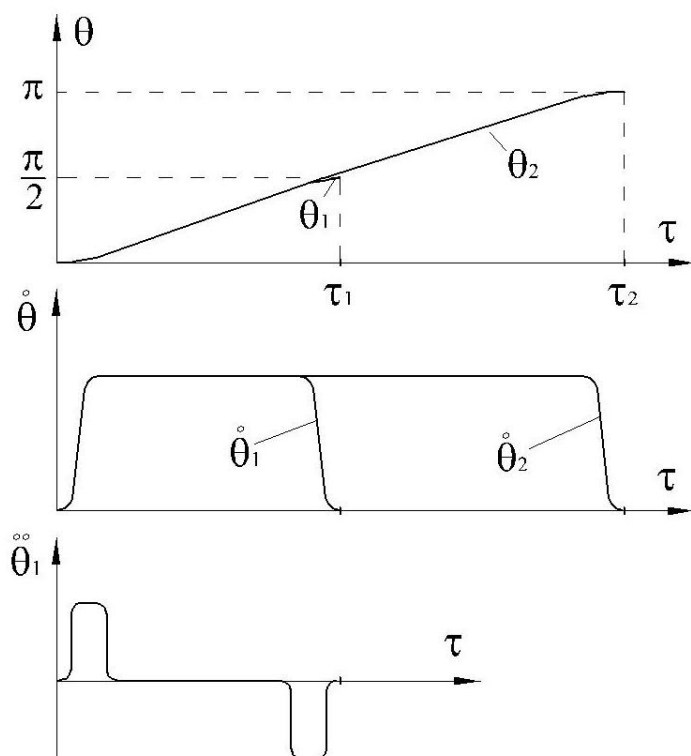
$$0 \leq \left| \omega_i = \dot{\theta}_i \right| \leq \dot{\theta}_{\max}$$

$$\dot{\theta}_2 = \dot{\theta}_1$$

$$0 \leq \left| \varepsilon_i = \ddot{\theta}_i \right| \leq \ddot{\theta}_{\max}$$

slika 5: Kretanje poluga

kao i postaviti karakteristične dijagrame kretanja poluga mehanizma za otvaranje krila:



slika 6: Dijagrami kretanja poluga

Moguće varijante izvedbe mehanizma su:

- 1) Mehanički – sa satnim mehanizmom;
- 2) Mehanički – sa sistemom opruga;
- 3) Mehanizam sa elektromotorom i baterijom za njegovo napajanje;
- 4) Pneumatski – sa spremnikom komprimiranog zraka;
- 5) Pneumatsko – hemijski, kod kojeg se gas potreban za pokretanje pneumatskog klipa dobiva hemijskom reakcijom između dvije određene čvrste supstance.

Ponovno se na osnovu načelne razrade varijanti rješenja kreira tabela odlučivanja:

| Varijanta | | 1 | | 2 | | 3 | | 4 | | 5 | |
|------------|----|-------------|-----|-----|----|-----|----|-----|----|-----|----|
| Kriterij | | | | | | | | | | | |
| Cijena | 4 | 10 4·10= | 40 | 10 | 40 | 7 | 28 | 5 | 20 | 4 | 16 |
| Pouzdanost | 10 | 10 | 100 | 9 | 90 | 7 | 70 | 4 | 40 | 6 | 60 |
| Sigurnost | 9 | 9 | 81 | 9 | 81 | 6 | 54 | 2 | 18 | 4 | 36 |
| Težina | 8 | 9 | 72 | 10 | 80 | 4 | 32 | 2 | 16 | 4 | 32 |
| Kvaliteta | 6 | 6 | 36 | 6 | 36 | 10 | 60 | 6 | 36 | 6 | 36 |
| Σ | | 329 | | 327 | | 244 | | 130 | | 180 | |

Prema ovoj matrici odlučivanja se može zaključiti da su za našu primjenu najpodesnije varijante 1) i 2).

1.4 Faza 3: Preliminarni dizajn

Kada smo izabrali koju ćemo varijantu upotrijebiti pristupa se izradi preliminarnog dizajna proizvoda. Izrada preliminarnog dizajna se sastoji iz sljedećih faza i podfaza:

1) Ekonomski pogled na predloženi dizajn

- Istraživanje tržišta:
 - cijena standardnih dijelova;
 - cijena izrade nestandardnih dijelova kod kooperanata;
 - cijena izrade nestandardnih dijelova u sklopu preduzeća;
- Izbor materijala, s obzirom na funkcionalnost proizvoda;
- Izbor proizvodnih postupaka, s obzirom na raspoložive tehnologije u preduzeću, mogućnost uvođenja novih tehnologija i mogućnost izrade dijelova kod kooperanata;
- Analiza raspoloživih resursa, s obzirom na mogućnost proširenja kapaciteta i mogućnost izrade dijelova kod kooperanata;
- Utvrđivanje ukupne cijene proizvoda sa osvrtom na:
 - porebe tržišta
 - raspoloživi obim proizvodnje
 - vrijeme otplate investicije i zaradu;

U cijenu proizvoda ulazi (cijena dizajna, cijena standardnih dijelova, materijala, proizvodnih procesa, troškovi upotrebe kooperanata, marketinga, distribucije i prodaje, kao i ZARADA $\approx 20\%$).

2) Utvrđivanje funkcionalnosti proizvoda, uz moguće promjene oblika proizvoda i pojedinih materijala.

3) Izrada uputa za upotrebu i standardizacija proizvoda (usklađivanje funkcije, energetske potrebe, upotrijebljenih materijala, otpadnih materijala ... sa važećim standardima).

4) Dizajn matrica 'What - How Matrix' u prevodu 'Šta - Kako Matrica'

| | | ŠTA (osobina koju treba postići) | Čvrstina | Sigurnost | Težina | Cijena | Funkcionalnost |
|---------------------|----------|-------------------------------------|----------|-----------|--------|--------|----------------|
| | | KAKO (način postizanja) | | | | | |
| Materijal | Čelik | | | | | | |
| | Al | | | | | | |
| | Polimeri | | | | | | |
| | Titan | | | | | | |
| Konstrukcija | I | | | | | | |
| | II | | | | | | |
| | III | | | | | | |
| | ... | | | | | | |
| Veličina | | | | | | | |
| Proizvodni postupci | | | | | | | |

slika 7: Dizajn matrica

Dizajn matrica je vrlo slična tabeli odlučivanja. Razlika je u tome da, uz pomoć tabele odlučivanja, odabiremo najpogodniju varijantu proizvoda, a pomoću dizajn matrice odlučujemo o karakteristikama već izabrane varijante. Unutar dizajn matrice se upisuju kružići, koji mogu biti prazni, jednu polovinu osjenčeni ili potpuno osjenčeni. Pomoću ovih kružića ukazujemo koliko određeni materijal, konstrukcija doprinosi ili ne doprinosi određenoj osobini koju proizvod treba da posjeduje. Potpuno osjenčeni kružić bi na primjer značio da proizvod od čelika ima najveću čvrstoću; dok se na presjeku reda sa čelikom i kolone sa težinom treba nalaziti potpuno neosjenčen krug, koji znači da izvedba od čelika ima najnepovoljniju (najveću) težinu.

1.5 Faza 4: Finalni dizajn

Posljednji korak do krajnjeg proizvoda je finalni dizajn. U sklopu finalnog dizajna se u potpunosti definišu svi relevantni podaci o proizvodu, potrebni za njegovu izradu. U sklopu finalnog, se često vrši izrada jednog ili čak više prototipa (ili čak nulte serije). Prototip se izrađuje kako bi se na njemu izvršila ispitivanja ponašanja proizvoda u toku eksploatacije. Na osnovu ponašanja prototipa zaključujemo da li je konstrukcija proizvoda predimenzionisana ili poddimenzionisana, da li su pravilno izabrani materijali, tolerancije, proizvodni procesi itd. Na taj način, na osnovu povratnih informacija od prototipa se donosi odluka, da li početi sa proizvodnjom, da li napraviti neke promjene na proizvodu ili napraviti novi prototip (princip pokušaja i pogreške).

Mora se navesti da dobra ideja ne mora dovesti do pojave proizvoda na tržištu. Od odlučujuće važnosti za dobivanje kvalitetnog proizvoda (onog koji posjeduje kvalitetu i prihvatljivu cijenu) i njegovu pojavu na tržištu je projektovanje. Težnja da se projektovanje proizvoda obavi na što lakši i što efikasniji način dovela je do uvođenja CAD sistema u proces projektovanja.

Postoje tri razloga za uvođenje CAD sistema u proces projektovanja:

- Izbjegavanje grešaka izazvanih projektovanjem na osnovu empirijskih izraza;
- Bolji ekonomski efekti i
- Multidisciplinarnost u projektovanju.

1.6 Faza 5: Proizvodnja – izrada prototipa

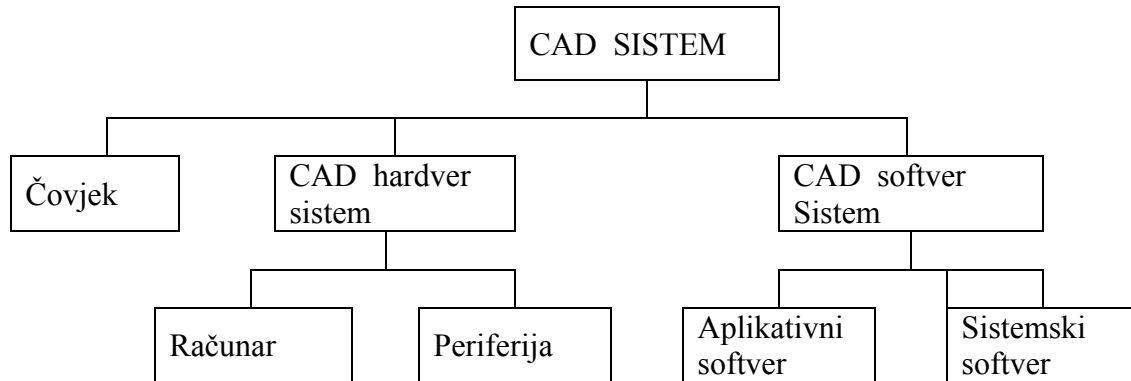
Izrada prototipa je sastavni dio procesa projektovanja, jer često dovodi do promjene u konstrukciji. Povezane naučne discipline koje se koriste u procesu projektovanja su:

- CFD
- FEM
- CCM
- Prototyping Rapid

2 CAD Sistemi

2.1 Struktura CAD sistema

CAD sistem se sastoji od hardware i software, u daljnjem tekstu hardvera i softvera:



slika 8: Podjela CAD sistema

Hardver se sastoji od: računara i periferne jedinice.

Pod računarom podrazumijevamo CPU (centralnu procesnu jedinicu), fiksnu memorijsku jedinicu (čvrsti disk "hard disc") i radnu memoriju (RAM).

Periferne jedinice služe za unošenje podataka u računar i iznošenje podataka iz njega. Na osnovu navedenog periferne jedinice se mogu podijeliti na ulazne i izlazne jedinice. U ulazne jedinice ubrajamo: tastaturu, miš, skener, vanjske memorijske jedinice (bušene trake, magnetne trake, floppy diskovi i CD-ovi) i jedinice za unošenje podataka sa vanjskih memorijskih jedinica (CD-ROM, floppy jedinica itd.).

Izlazne jedinice su: monitor, štampač, ploter itd.

Ipak sam hardver nije upotrebljiv bez softvera. Pod softverom podrazumijevamo programe, koji su napisani u obliku koji računar može prepoznati, pomoću kojih se upravlja radom računara. Razlikujemo systemske i aplikativne (korisničke) softvere. Zadatak sistemskih softvera je da osiguraju optimalan pravilan rad računara, dok putem aplikativnog softvera operater pomoću određenih instrukcija (naredbi, predviđenih aplikativnim programom) usmjerava rad računara, kako bi ga iskoristio za izvršenje željenih operacija.

Aplikativni CAD softveri rade sa sljedećim grupama instrukcija:

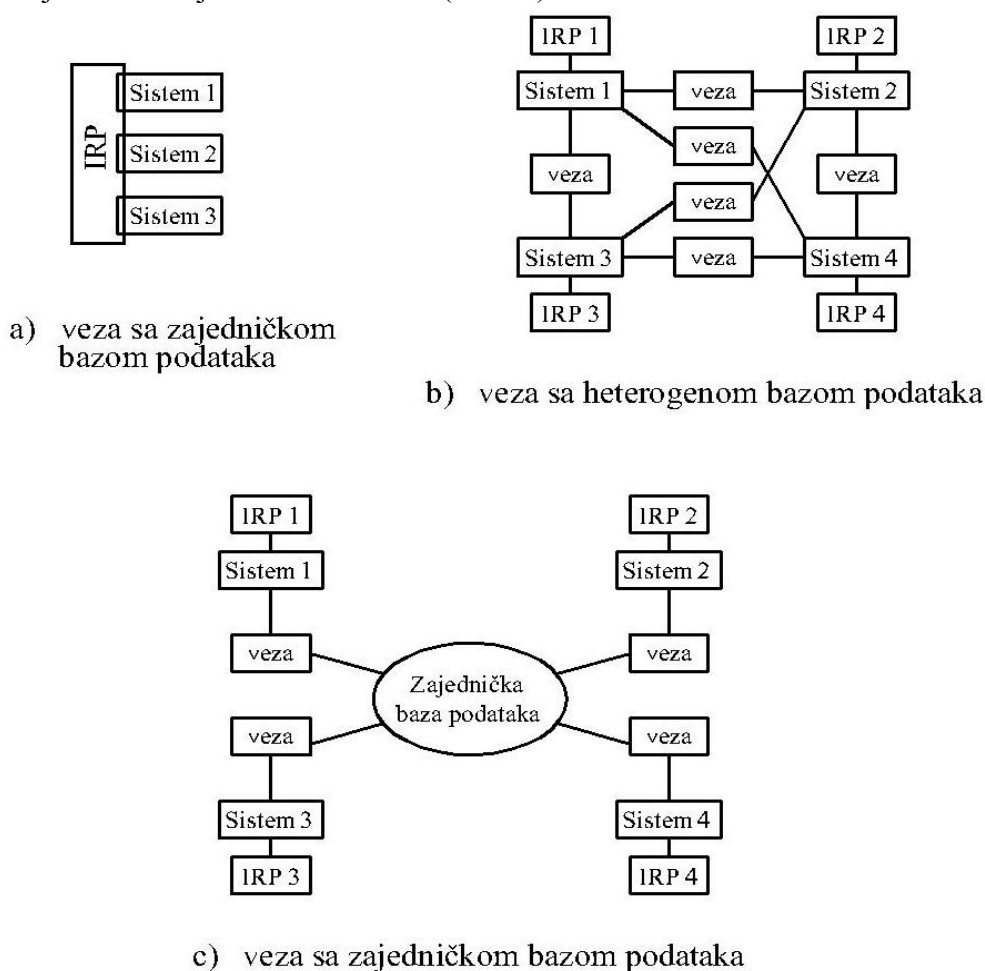
- Naredbe za crtanje (line, polyline, multiline, circle, arc...)
- Naredbe za izmjene (erase, copy, move, array...)
- Naredbe za projiciranje (view, zoom, pan...)
- Naredbe za kotiranje
- Naredbe za šrafiranje
- Naredbe za proračun (za izračunavanje težine, težišta, površina, momenata inercije...)
- Naredbe za informisanje (izrada sastavnica, studija, specifikacija, izvještaja...)
- Naredbe za layer-tehniku
- Naredbe za prikaz atributa i
- Naredbe za simulaciju (kinematska i numerička simulacija)

2.2 Način čuvanja, prikazivanja i prijenosa podataka

U današnjoj industriji pored protoka materijala i energije, velika pažnja se posveđ protoku informacija (podataka iz i između: konstrukcionog biroa, proizvodnih pogona, službe za nabavku itd.). Da bi se osigurao nesmetan i efikasan protok informacija neophodno je adekvatno organizovati način njihovog čuvanja. Način "skladištenja" informacija zavisi od njihovih korisnika (CAD sistema) i veze između njih.

Veza između CAD softverskih sistema je otežana zato što se razlikuju načini njihovog internog računskog predstavljanja podataka za geometrijske, topološke i tehnološke karakteristike.

Generalno obzirom na softverske mogućnosti, veza između CAD sistema ostvaruje se na jedan od sljedeća tri načina (slika 9):



slika 9: Veze CAD sistema

- veza pomoću homogene baze podataka. Kod ovako postavljene veze između korisnika svi oni su povezani sa takozvanim server – računarnom na kojega su instalirani svi korisnički softveri i memorija. Na taj način svi korisnici S_1, S_2, \dots, S_n , imaju identično interno računsko predstavljanje (IRP), a svi podaci se pohranjuju na jednu bazu podataka, koja se nalazi pri serveru.

- veza pomoću heterogene baze podataka, koja se koristi kod CAD sa zasebnim bazama podataka, a različitim internim računskim predstavljanjem. U ovom slučaju je potrebno pri prenošenju podataka iz jednog sistema u drugi prilagoditi način na koji su podaci zapisani novom korisniku (pred- i post-procesiranje).
- veza pomoću zajedničke baze podataka, kod koje se podaci smještaju u zajedničku bazu podataka (za razliku od prethodnog slučaja, gdje se podaci izmjenjuju direktno između CAD sistema), iz koje su potom dostupni svakom od korisnika. Korisnici ne moraju imati interno računsko predstavljanje, zato što je predviđena mogućnost pred- i post-procesiranja podataka.

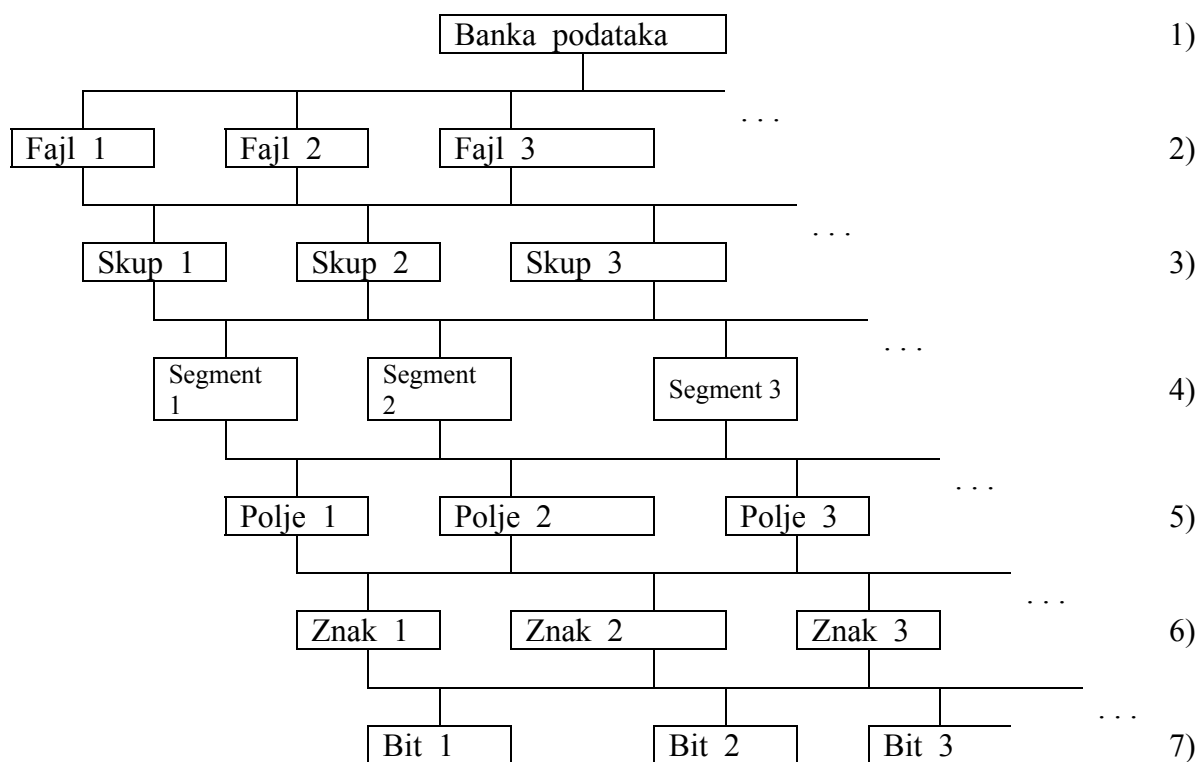
Neki od ciljeva tehničke obrade podataka i njihovog uređenog čuvanja su:

- sprečavanje pojave redundantnih podataka i
- decentralizovana obrada podataka.

Redundantnost podataka predstavlja višestruko ponavljanje istih podataka unutar baze podataka. To je neželjena pojava ne samo iz razloga što se na taj način nepotrebno zauzima prostor u memoriji, već i zbog toga što se postojanjem redundantnih podataka smanjuje preglednost banke podataka, a može doći i do zabuna uslijed njihove eventualne nekonzistentnosti.

2.3 Struktura baza podataka:

Prema sveopšte usvojenom principu današnje baze podataka se sastoje iz sedam nivoa, kao što je prikazano na "slici 10":



slika 10: Struktura baze podataka

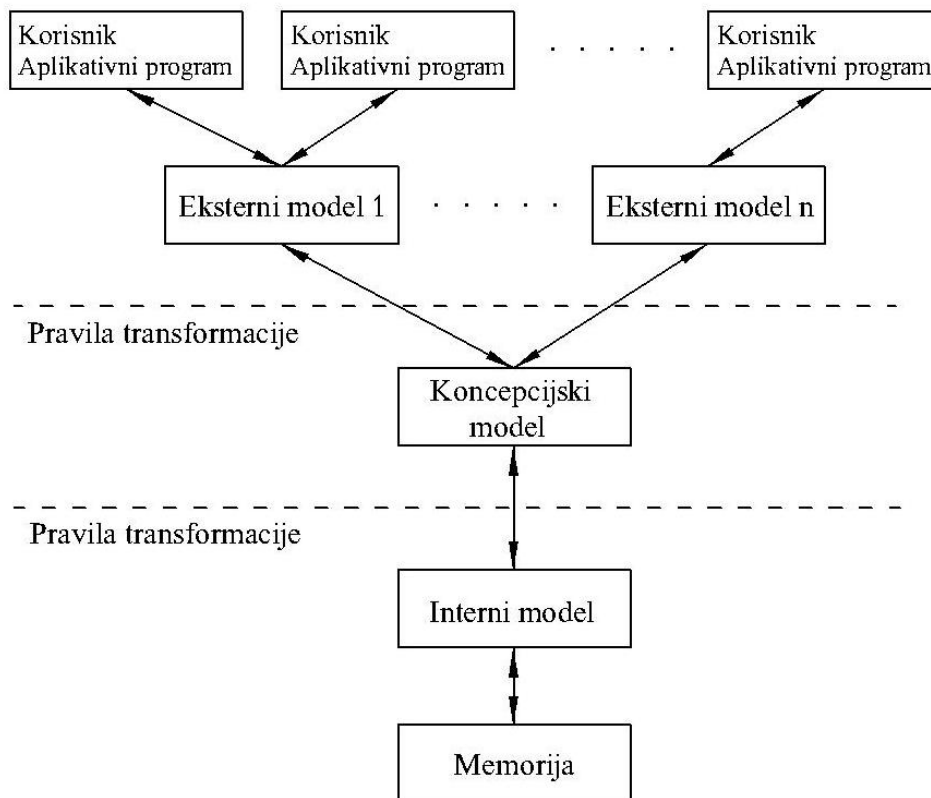
Nivo bita "bit" predstavlja osnovnu jedinicu "sporazumijevanja razuma" računara. Na ovom nivou se podaci pohranjuju u digitalnom – binarnom obliku (0, 1).

Nivo znaka "byte" je nivo u kojem se spajanjem osnovnih informacija – bita dobivaju znakovi, koji su osnovne jedinice sporazumijevanja ljudi. U zavisnosti od broja bita koji čine jedan "byte" imamo: 8-bitni sistem (npr. ASCII – kod), 16-bitni sistem, 32, 64 i 128-bitni sistem [$8=2^3$, $16=2^4$, $32=2^5$...].

Za dalje pojašnjenje strukture baza podataka uzet ćemo primjer tekst – procesora (npr. Microsoft Word), gdje je: polje ekvivalentno sa riječi, segment sa rečenicom, skup sa Word dokumentom itd.

2.4 Arhitektura baze podataka

Arhitektura baze podataka je prikazana na sljedećoj slici:



slika 11: Arhitektura baze podataka

Pod arhitekturom baze podataka podrazumijevamo način na koji se gradi jedna baza podataka. To je proces u kojem neku za nas logičnu "konstrukciju" (geometriju tijela, tekst, sliku ...) preoblikujemo u "konstrukciju" logičnu računaru – binarni kod.

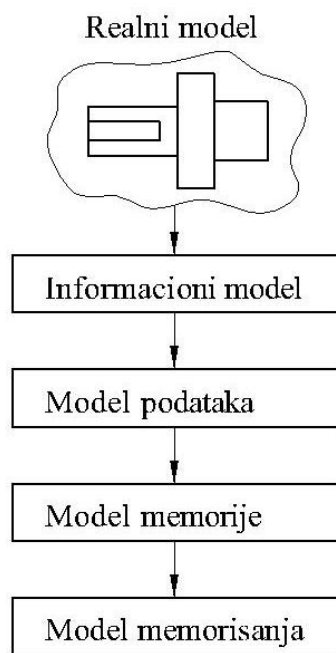
Tokom ovog procesa razlikujemo tri nivoa:

- Korisnički nivo (iznad linije prve transformacije),
- Prelazni nivo (između linija transformacija) i
- Mašinski nivo.

U korisničkom nivou operater koristi naredbe korisničko – aplikativnog programa da računaru saopšti na primjer geometriju nekog radnog komada. Na osnovu tih instrukcija aplikativni program kreira "eksterni model" geometrije komada, koji se sastoji iz skupa: linija, kružnica, kotnih crta itd. Ovaj model u biti predstavlja raščlanjenu geometriju komada i mi ga još uvijek možemo razumjeti. Nakon kreiranja eksternog modela nastupa njegova transformacija u tzv. "konceptijski" model (prelazni nivo). Ova transformacija se vrši pomoću unutrašnjih instrukcija aplikativnog modela. Ove instrukcije nisu poznate korisniku programa, nego samo kreatoru. Konceptijski model je oblik predstavljanja geometrije alata sa kojega će se najlakše sljedećom transformacijom do skupa informacija pogodnog za interno računarsko predstavljanje računara - "interni model" (mašinski nivo). Zadnja faza je kreiranje "mašinskog modela", pisanog u binarnom kodu, kakav možemo pohraniti na jednu od memorijskih jedinica.

2.5 IRP (interno računsko predstavljanje)

Pojednostavljeno, IRP predstavlja oblik četiri koraka kojima računar pohranjuje realni model u svoju memoriju.



slika 12: Interno računsko predstavljanje

Zadatak informacionog modela je da preko tačaka i elemenata kontura pretvori realni model (geometrija i dimenzije tijela) u model podataka.

Model podataka predstavlja apstraktno – logičko predstavljanje podataka. U ovoj fazi računar, na osnovu podataka, koje je dobio i njihove strukture, formira tabele elemenata, relacija i relacija između elemenata (npr. linija se predstavlja memorisanjem dva elementa – tačke i jedne relacije između njih).

Model memorije je apstraktno-parametarsko predstavljanje realnog modela. Ovaj model se sastoji iz gore navedenih primitiva (tačke, linije, ...) predstavljenih u parametarskom obliku. Ovim se generišu polja sa njihovim tokovima. Model memorije se kreira potpuno automatski (pomoću programa napisanih u nekom od programskih jezika)

Model memorisanja je fizički model zapisivanja podataka na memorijski medij (piše se u mašinskom programskom jeziku – binarno).

2.5.1 Informacioni model

Na nivou informacionog modela zadatak je apstrakcija i formalizacija realnog modela (crtež), odnosno pretvaranje vizualnog u adekvatne primitive, pomoću naredbi programa.

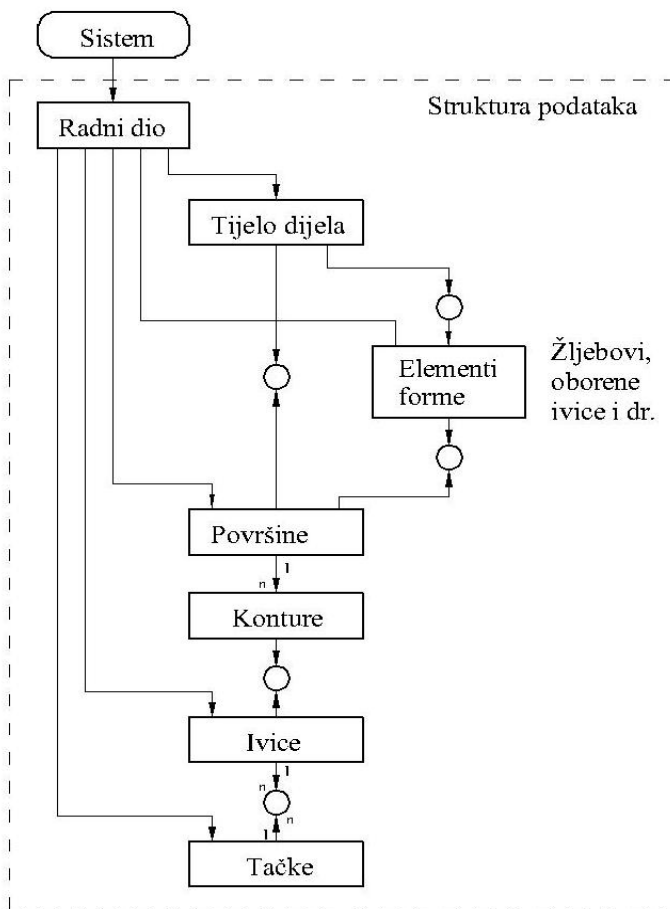
Javlja se pet različitih modela predstavljanja realnog modela:

- 1) 3D – zapreminski model
- 2) 3D – površinski model,
- 3) 3D – žičani (ivični) model,
- 4) 2L / 2D model i
- 5) 2D model.

Trodimenzionalni – zapreminski model obuhvata izgled, oblik i karakteristika proizvodnog radnog komada. Kreiranje ovakvog modela počinje uvođenjem jednostavnih prostornih elemenata, koji se spajaju Bool-ovim operacijama (unija, presjek i razlika).

Trodimenzionalni – površinski model također polazi od prostorno orijentisanih elemenata, koji su ovom slučaju površina, elementi predstavljeni konturama (ivice) i tačke, ali su mogućnosti korištenja ovih modela u odnosu na 3D – zapreminske modele smanjene (npr. radi nemogućnosti vršenja analize napona u materijalu).

Trodimenzionalno – žičani modela polazi od prostornog oblika radnog komada koji se gradi pomoću graničnih ivica. Problem ovog modela ponekad je u lošoj preglednosti, koja je uzrokovana ne razlikovanjem vidljivih i nevidljivih ivica.



Kod 2L / 2D modela se kao osnova, crta 2D tijelo u x-y ravni, a treća dimenzija se dobiva izvlačenjem tog 2D tijela po nekoj pravoj.

Najjednostavniji od svih modela je dvodimenzionalni model, ali se pomoću njega mogu jednoznačno opisati samo ravanski i rotacioni modeli. Utrošak vremena potrebnog korisniku za kreiranje 2D modela nije značajno manji od vremena potrebnog za kreiranje 3D modela, ali je značajna prednost ovog modela to, što je dobivanje crteža sa aspekta računara mnogo jednostavnije; kraće je računsko vrijeme, a i razvoj softvera je mnogo jednostavniji.

slika 13: Informacioni model

2.5.2 Model podataka

Model podataka predstavlja način prikazivanja primitiva. Cilj je osigurati potpuno pohranjivanje podataka, sa težnjom da se zauzme što manje prostora u memoriji, a da se pri tome i osigura što jednostavniji i brži pristup svakom od memorisanih podataka. Također je vrlo važan način na koji je moguće vršiti pojedine izmjene.

Postoji više modela (podataka) za prikazivanje primitiva, a to su:

- 1) Formalni dvolinijski sistem baze podataka i
- 2) Relacioni sistem baze podataka.

Za pojašnjenje dvolinijskog sistema baze podataka uzet ćemo primjer memorisanja dvaju linija sa različitim debljinama i bojama.:

| | | |
|----------------|---|----------------|
| T ₂ | Crvena | T ₃ |
| zelena | <div style="display: flex; align-items: center;"> <div style="margin-right: 5px;">l₂</div> <div style="flex-grow: 1; border-bottom: 1px solid black; position: relative;"> T₃ </div> </div> <div style="margin-top: 5px;"> <div style="display: flex; align-items: center;"> <div style="margin-right: 5px;">l₁</div> <div style="flex-grow: 1; border-bottom: 1px solid black; position: relative;"> T₂ </div> </div> <div style="margin-top: 5px;"> <div style="display: flex; align-items: center;"> <div style="margin-right: 5px;">T₁</div> <div style="flex-grow: 1; border-bottom: 1px solid black; position: relative;"> T₂ </div> </div> </div> </div> | |

| LINIJA | | | TAČKA | |
|----------------|------|----------|----------------|------------|
| Ime | Boja | Debljina | Ime | Vrsta |
| L ₁ | Z | 1 | T ₁ | K(krajnja) |
| L ₁ | Z | 1 | T ₂ | K |
| L ₂ | C | 2 | T ₂ | K |
| L ₂ | C | 2 | T ₃ | K |

slika 14: Formalna dvolinijska baza podataka

U datoj tabeli se u svakom redu jednom elementu pridružuje po jedna tačka. Također se unose i osobine linija i tačaka. Kako jedna tačka pripada dvjema linijama (T₂), ona će se u tabeli pojaviti dva puta, što znači da imamo redundantnost podataka. Da bi se pri memorisanju većih CAD datoteka (fajlova) "files" što je moguće više smanjilo ponavljanje podataka pribjegava se relacionom modelu podataka (slika 14). Kod ovog modela se tvore tabele u kojima su zasebno memorisani različiti primitivi sa svojim osobinama, kao što su tačke, zasebno linije, kružnice, lukovi itd. Pored ovih tvore se tabele u kojima se prikazuju relacije između pojedinih primitiva.

| LINIJA | | |
|----------------|------|----------|
| Ime | Boja | Debljina |
| L ₁ | crv. | 2 |
| L ₂ | zel. | 1 |

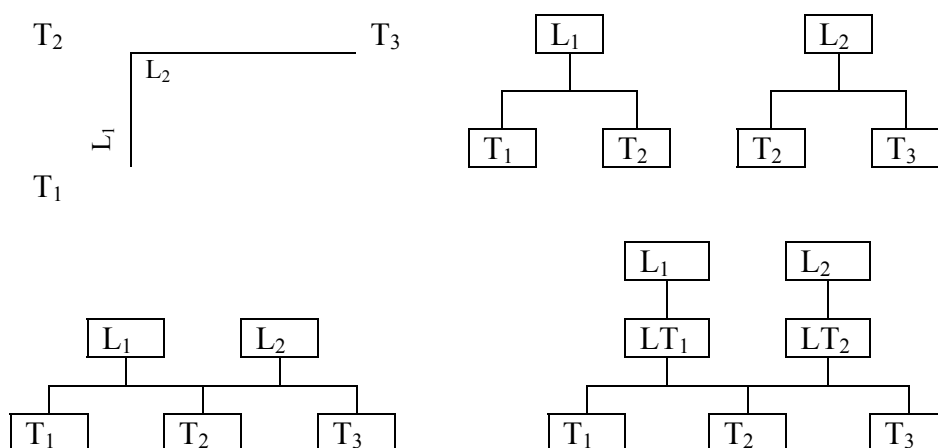
| TAČKA | |
|----------------|-------|
| Ime | Vrsta |
| T ₁ | K |
| T ₂ | K |
| T ₃ | K |

| TAČKA-LINIJA | |
|----------------|----------------|
| Ime tačke | Ime linije |
| T ₁ | L ₁ |
| T ₂ | L ₁ |
| T ₂ | L ₂ |
| T ₃ | L ₂ |

Slika 15: Relaciona baza podataka

Posljednja od ove tri tabele je tabela sa relacijama tačka – linija. U njoj se naznačuje koje tačke pripadaju određenoj liniji. Ovaj model se u našem primjeru čini glomaznijim, ali kada se poveća broj elemenata, kojima se opisuje (crta) realni model, tada ovako organizovana baza podataka zauzima mnogo manje prostora na memorijskom mediju nego dvolinijska.

Poznajemo hijerarhijski i mrežni model relacione baze podataka:



slika 16: Hijerarhijski i mrežni model relacione baze podataka

Hijerarhijski model je zbog ograničenja na jedan ulaz i dugi pristup prema osnovnom stablu vrlo redundantan i zbog toga je nepogodan za primjenu u CAD sistemima. Mrežni model pruža više ulaza i izlaza po elementu, zato što, u opštem slučaju, ima proizvoljan broj relacija – asocijacija nagore i na dole. Na taj način je pristup do svakog elementa moguć sa nekoliko strana. Poseban mrežni model je CODASYL sistem (Conference on Data Systems Languages), koji predviđa dodatne vezne elemente (link records) za gradnju željenog minimuma asocijacija između elemenata. Mrežni model je zastupljen kod većine CAD sistema (od toga je najviše CODASYL modela).

2.5.3 Model memorije

U prethodnom modelu smo iz primitiva dobili tabele, ali da bismo ih memorisali moramo ih "organizovati". Postavlja se pitanje: "Gdje i kako ih memorisati?"

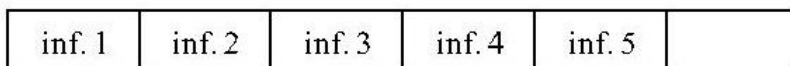
To je zadatak modela memorisanja. Poznajemo tri vrste organizovanja podataka:

- 1) Sekvencijalna organizacija podataka,
- 2) Organizacija u obliku lista i
- 3) Način direktnog pristupa.

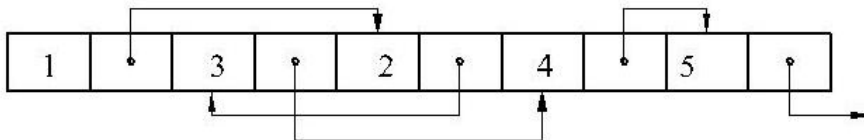
Kod sekvencijalne organizacije podataka kreće se od jedne tabele, koja se u potpunosti upisuje u memoriju, zatim se nastavlja sa drugom tabelom itd. Iako se na ovaj način podaci pohrane korištenjem najmanje prostora u memoriji, on nije praktičan, jer da bi pročitali neki podatak moramo svaki puta krenuti od početka. Problem je također vršenje izmjena u memorisanom skupu podataka.

Ukoliko naše podatke organizujemo u obliku lista utrošit ćemo više prostora, ali će se poboljšati preglednost podataka. Kod ovog modela se poslije svakog memorisanog podatka upisuje "adresa" podatka, kojeg je potrebno sljedećeg pročitati. Podaci se i dalje upisuju sekvencijalno – u obliku trake.

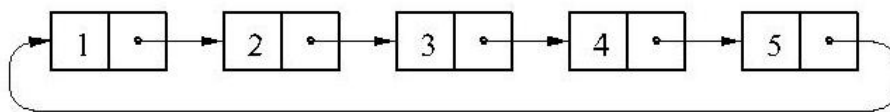
Način direktnog pristupa se ostvaruje tako što se uz svaki podatak upisuje informacija o položaju sljedećeg, ali i prethodnog podatka. Na taj način se osigurava brz pristup podacima, ali i to da se podaci ne moraju pohranjivati sekvencijalno, tako da se pojedini dijelovi mogu pohraniti tamo gdje je to trenutno najpogodnije.



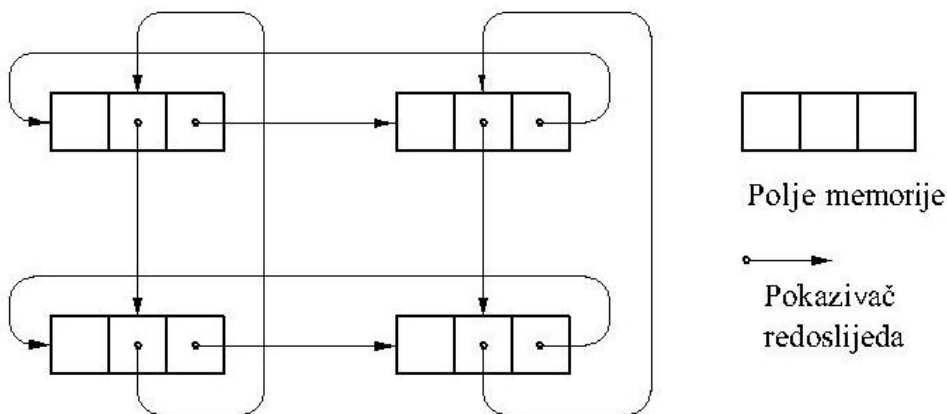
- Linearna lista (sekvencionalna organizacija)



- Lančana lista



- Prstenasta lista



- Prstenasta struktura

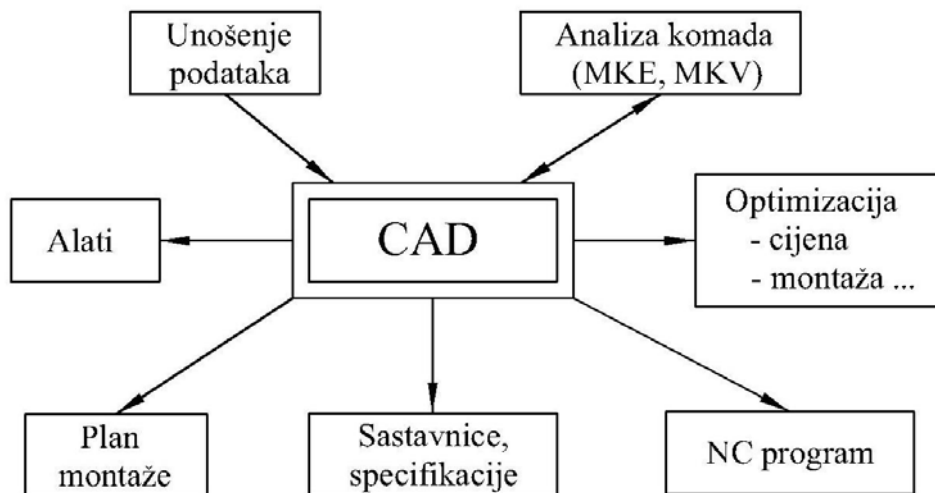
slika 17: Struktura memorisanja

2.5.4 Model memorisanja

Model memorisanja definiše način na koji se podaci koji su obrađeni u prethodnim modelima zapisuju u memoriju računara i na taj način mogu kasnije koristiti na svim nivoima. Ovaj model omogućava direktnu razmjenu podataka između sistema.

2.6 Pred- i post – procesori (“interface” u daljem tekstu interfejs)

Hardver jednog CAD sistema se sastoji iz računara i raznih perifernih uređaja.

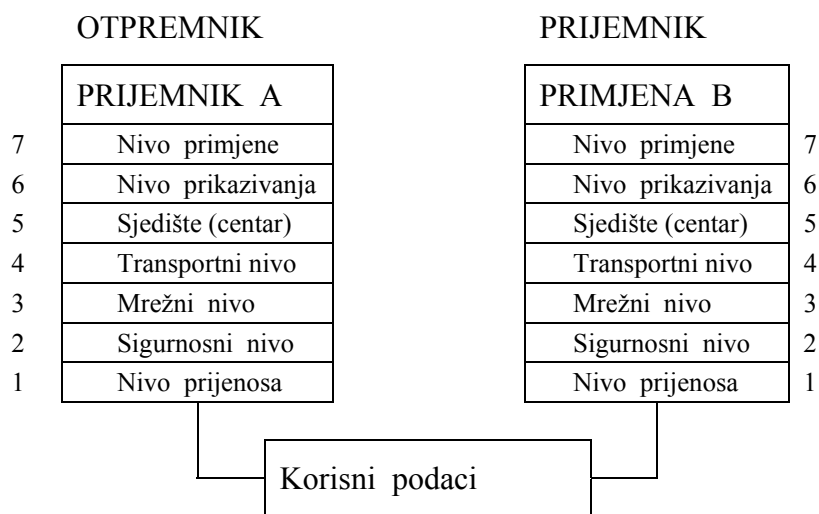


Slika 18: Struktura CAD sistema

Svi ovi uređaji imaju različita interna računaska predstavljanja. Da bi se mogla vršiti razmjena podataka između računara i ovih uređaja, kako i između njih međusobno koriste se tzv. interfejsi odnosno pred- i post-procesori. Pred- i post – procesori su uređaji za vezu i prevođenje (između raznih jezika), između raznih dijelova CAD sistema i njihove okoline. Predprocesor je interfejs za unos podataka, dok je postprocesor interfejs za izvoz podataka.

2.6.1 Osnovna struktura interfejsa

Struktura inetrfejsa se sastoji od sedam nivoa (analogno bankama podataka):



Slika 19: ISO – sistem prijena podataka

Prva četiri nivoa (od nivoa prijenosa do nivoa transporta) predstavljaju transportni sistem interfejsa, dok preostala tri nivoa tvore područje (sistem) primjene.

1. Nivo prijenosa bita interfejsa osigurava direktan prijenos podataka između otpremnika i primaoca i definiše način prijenosa. Sastoji se od hardvera i softvera.
2. Nivo sigurnosti (osiguranja) prijenosa definiše način provjere kvaliteta prijenosa podataka, zatim prepoznaje i uklanja eventualne greške nastale tokom istog. Zadatak ovog nivoa je također da kontroliše adrese prijemnika i predajnika, kao i da sinhronizuje prijenos podataka.
3. Mrežni nivo je taj koji služi za istraživanje i iznalaženje optimalnog puta za podatke kroz transportnu (komunikacionu) mrežu.
4. Zadatak transportnog nivoa je spajanje i razdvajanje različitih paketa podataka, koji se prenose, kako bi osigurao da u određenom trenutku optimalna količina podataka putuje kroz mrežu. Ponovno spajanje paketa podataka, na način na koji će oni poprimiti prvobitni izgled po prijemu kod prijemnika je također zadatak transportnog nivoa prijenosa.
5. Nivo centra reguliše izgradnju komunikacije (veza) između sistema primjene i transporta (npr. sprečava situacije u kojima bi interfejs pošiljaoca poslao sve podatke koji su mu proslijeđeni, ali bi prijemnik bio u nemogućnosti da otvori datoteku zato što mu nedostaje jedan paket podataka).
6. Nivo prikazivanja ima, prije svega, zadatak da različite formate podataka (npr. ASCII, EBCDIC, ...) međusobno prilagodi.
7. Nivo primjene ima zadatak da zapiše podatke u obliku, u kojem će ga prijemnik moći koristiti

Postoje razne izvedbe interfejsa. Napomenimo, za nas, samo najvažnije:

- IGES interfejs, koji se u CAD sistemima koristi za prijenos 3D podataka i
- DXF interfejs, koji je od AutoCAD-ovog interfejsa postao standardan interfejs za prijenos 2D podataka kod CAD softvera, ali je od nedavno prilagođen i za prijenos 3D podataka.