

UNIVERZA V LJUBLJANI
VISOKA UPRAVNA ŠOLA

Študijsko gradivo

Zapiski s predavanj

Predmet: Informacijski sistemi
Profesor: dr. Mirko VINTAR
Študentka

Ljubljana, februar 2003

KAZALO

1 POSLOVNI VIDIKI RAZVOJA IS.....	5
1.1 OPREDELITEV INFORMACIJSKEGA SISTEMA.....	5
1.2 KLJUČNE TOČKE INFORMATIZACIJE UPRAVE.....	6
1.3 VLOGA IS V MODERNIH ORGANIZACIJAH.....	6
1.4 GLAVNE RAZVOJNE SMERI E-UPRAVE.....	6
1.4.1 <i>Nova paradigma pri nadaljnjem razvoju IS v upravi.....</i>	<i>7</i>
1.4.2 <i>Izhodišča za nadaljnji razvoj IS v upravi.....</i>	<i>8</i>
1.5 KLJUČNI DEJAVNIKI USPEŠNEGA RAZVOJA IS.....	8
1.5.1 <i>Potrebna znanja informatikov in uporabnikov za uspešen razvoj IS.....</i>	<i>9</i>
1.6 INFORMATIZACIJA KOT PRILOŽNOST ZA PRENOVO POSLOVANJA UPRAVE.....	10
1.7 TEMELJNA IZHODIŠČA PRENOVE POSLOVNIH PROCESOV.....	11
2 METODOLOGIJE NAČRTOVANJA IN GRADNJE IS.....	12
2.1 OPREDELITEV OSNOVNIH POJMOV	12
2.2 PRISTOPI K NAČRTOVANJU IN GRADNJI IS.....	13
2.2.1 <i>Linearni pristop.....</i>	<i>13</i>
2.2.2 <i>Prototipni pristop.....</i>	<i>14</i>
2.2.3 <i>Objektni pristop (še v razvoju).....</i>	<i>14</i>
2.2.4 <i>Značilne razvojne faze is v okviru sodobnejših postopkov.....</i>	<i>15</i>
3 MODELIRANJE INFORMACIJSKIH SISTEMOV.....	18
4 MODELIRANJE PODATKOV.....	20
4.1 GLAVNI KONCEPTI ABSTRAKCIJE.....	20
4.2 RAZVOJ PODATKOVNIH MODELOV.....	21
4.2.1 <i>KLASIČNI MODELI (Izvedbeni podatkovni modeli).....</i>	<i>22</i>
5 MODEL ENTITETA – POVEZAVA (E-R MODEL).....	23
5.1 ENTITETA.....	23
5.2 VRSTE ATRIBUTOV.....	25
5.3 LASTNOSTI POVEZAV.....	27
5.4 GRAFIČNA NOTACIJA E-R MODELA.....	30
5.5 RAZVOJ E-R MODELA.....	31
5.6 PODATKOVNI SLOVAR.....	31
5.6.1 <i>Slovar entitet.....</i>	<i>31</i>
5.6.2 <i>Slovar atributov.....</i>	<i>32</i>
5.6.3 <i>Slovar povezav.....</i>	<i>33</i>
5.7 IZVEDBENA FAZA IN M:N KARDINALNOST.....	34
5.8 MOČNI IN ŠIBKI TIPI ENTITET.....	34
6 MODELIRANJE POSTOPKOV.....	36
6.1 OPREDELITEV OSNOVNIH KONCEPTOV MODELIRANJA POSTOPKOV.....	37
6.2 METODE IN TEHNIKE MODELIRANJA POSTOPKOV.....	38
7 PROBLEMI PREHODA IZ LOGIČNE NA FIZIČNO RAVEN ZASNOVE IS.....	41
7.1 PRETVORBA E-R MODELA V RELACIJSKI MODEL.....	42
8 TEHNOLOŠKA, ORGANIZACIJSKA IN KADROVSKA IZHODIŠČA.....	43
8.1 UPORABA INFORMACIJSKIH ORODIJ.....	43

8.2 ORGANIZACIJSKI, TEHNOLOŠKI IN KADROVSKI VIDIKI RAZVOJA INFORMACIJSKE INFRASTRUKTURE.....	45
8.2.1 Tehnološki vidiki.....	46
8.2.2 Organizacijski vidiki.....	47
8.2.3 Kadrovski vidiki.....	48
9 INFORMATIKA IN STANDARDI.....	49
9.1 ISO 9000-3 (PROCESNI STANDARD).....	49
9.2 DIN 66285.....	50
10 OSNOVNE ZNAČILNOSTI SODOBNE METODOLOGIJE.....	51
10.1 GENERACISKI RAZVOJ METODOLOGIJ NAČRTOVANJA IN GRADNJE IS.....	51
10.1.1 1. GENERACIJA.....	51
10.1.2 2. GENERACIJA.....	51
10.1.3 3. GENERACIJA.....	52
10.1.4 4. GENERACIJA.....	52
11 PRILOGE.....	53

1 POSLOVNI VIDIKI RAZVOJA IS

1.1 OPREDELITEV INFORMACIJSKEGA SISTEMA

IS¹ je skupek **ljudi**, **postopkov** in **naprav**, zasnovan za zbiranje, obdelavo, shranjevanje in distribucijo podatkov oziroma informacij.

IS predstavlja **organizacijsko tehnološko okolje** za upravljanje z informacijami in informacijskimi tokovi obravnavanega poslovnega sistema.

Sestavine IS:

- organizacijske rešitve,
- ljudje,
- metodološke rešitve,
 - predstavitev poslovnih struktur,
 - predstavitev postopkov,
- programska oprema,
- strojna oprema.

Primeri IS: IS šole, IS univerze, IS bolnice.

Glavne naloge IS:

- dostop do informacij (*tistih, ki jih potrebujemo za delo*),
- obdelava informacij,
- izmenjava informacij (*krajevne in časovne razdalje*).

Glavne aktivnosti:

- zajem podatkov,
- procesiranje,
- arhiviranje,
- distribucija,
- izmenjava.

¹ Informacijski sistemi

1.2 KLJUČNE TOČKE INFORMATIZACIJE UPRAVE

1. Uvajanje **informacijske tehnologije** v vse faze zbiranja, obdelave, shranjevanja in posredovanja informacij,
2. prenova **poslovnih procesov** na osnovi inovativne uporabe informacijske tehnologije,
3. preureditev **informacijskih tokov** ter njihova prilagoditev (*včasih telefaks, danes internet*) možnosti IT,
4. prilagoditev ali sprememba **organizacijske strukture**, v katero se uvaja sodobna tehnologija (*npr. bančništvo*),
5. prilagoditev **metod managementa** uporabi sodobnih informacijskih virov.

1.3 VLOGA IS V MODERNIH ORGANIZACIJAH

- Strateška **vloga** informatike,
- IS in **konkurenčna** sposobnost organizacij,
- strateško **načrtovanje** IS.

Organizacija, ki ima dobro razvit IS, ima dobre osnove za učinkovito upravljanje z informacijskimi viri in tokovi, kar je pomembna prednost pri konkurenci.

IS lahko najbolj vpliva na konkurenčno sposobnost. Z uporabo elektronskih rešitev poenostavimo izvajanje npr. zadev in postopke - cilj je bistveno izboljšanje storitev.

Vse to pa zahteva drugačen pristop k načrtovanju tega področja - pristopiti je potrebno dolgoročno in sistematično, ciljno. To načrtovanje je zelo pomembno in od tega so odvisni rezultati organizacije.

1.4 GLAVNE RAZVOJNE SMERI E-UPRAVE

Glavne razvojne smeri e-uprave so predvsem omogočati razvoj storitev, ki jih prej uprava ni mogla ponuditi:

- **smer:** uprava ↔ občani (*razvoj novih storitev – kakovost storitev, krajši čas pri opravljanju storitev, ...*),
- **smer:** uprava ↔ organizacija (*razvoj novih storitev – pomoč gospodarskim organizacijam, ...*),
- **smer:** uprava ↔ uprava (*razvoj novega načina poslovanja – internetno poslovanje, hitrejše in kakovostnejše notranje poslovanje uprave, ...*).

Danes smo na to tehnologijo življenjsko vezani. V preteklosti smo videli to tehnologijo samo kot sredstvo za doseganje ciljev, danes pa moramo z razvojem e-uprave redefinirati tudi cilje.

Pri tem moramo upoštevati **štiri temeljna načela pri poslovanju z občani**:

- načelo enkratnega obveščanja (*znotraj celotnega javnega sektorja*),
- načelo obveznega pridobivanja mnenj in soglasij po uradni dolžnosti,
- načelo poslovanja brez osebnega stika s stranko,
- načelo ponudbe po sistemu »vse na enem mestu«.

NAČELO ENKRATNEGA OBVEŠČANJA

Danes so naši naslovi (*osebni podatki*) zbrani v številnih podatkovnih skladiščih (*banka, energetika, ptt,...*). Ključni problem: imamo množico IS, ki med seboj niso povezani. S pomočjo povezanih IS, naslov spremenimo samo enkrat.

NAČELO OBVEZNEGA PRIDOBIVANJA MNENJ IN SOGLASIJ PO URADNI DOLŽNOSTI

PRIMER: radi bi gradili hišo, torej potrebujemo gradbeno dovoljenje; žal potrebujemo 8 različnih soglasij, za kar porabimo ogromno časa. Vsi imajo svoje IS, ki med seboj niso povezani - uprava sama pridobi vsa potrebna mnenja in soglasja.

NAČELO POSLOVANJA BREZ OSEBNEGA STIKA S STRANKO

Poslovanje na daljavo in s pomočjo elektronskega podpisa.

NAČELO PONUDBE PO SISTEMU »vse na enem mestu«

Vstopna točka (*državni vstopni portal*), s katero se ponudi vse storitve javnega sektorja na enem mestu.

1.4.1 Nova paradigma pri nadaljnjem razvoju IS v upravi

Kriteriji	Staro načelo	Novo načelo
ORGANIZIRANOST UPRAVE	Formalna, hierarhična struktura, birokratska organizacija	Dinamična, mrežna struktura bolj avtonomnih organizacij
NAČINI IZVAJANJA NADZORA IN VODENJA SPREJEMANJA ODLOČITEV	Od vrha navzdol	Decentralizacija – menedžerski principi upravljanja in vodenja
ODNOS DO JAVNOSTI	Zaprta struktura, delovanje pod oznako »interno« in »zaupno«	Transparentnost, odprtost
DELITEV PRISTOJNOSTI	Resorno in načelo krajevne pristojnosti	Načelo maksimiziranja razpoložljivih virov
NAČIN PONUDBE STORITEV	Parcialen (<i>nepopoln</i>), resoren	»Vse na enem mestu«
DOSTOPNOST UPRAVNIH STORITEV	Dekonzracija služb	Načelo socialne pravičnosti (<i>Digital divide</i>)

POSLANSTVO	Oblikovanje in izvajanje zakonov	Zagotavljanje potreb občanov in organizacij
UGOTAVLJANJE UČINKOVITOSTI IN KAKOVOSTI DELA	Ugotavlja uprava sama	Ugotavljajo njeni uporabniki (<i>občani, organi</i>), sistemi kakovosti

1.4.2 Izhodišča za nadaljnji razvoj IS v upravi

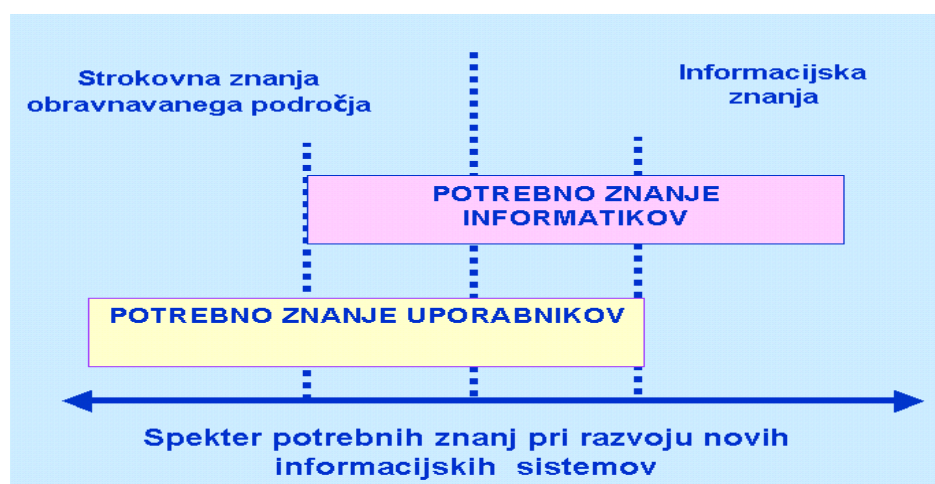
Kriteriji	Staro načelo	Novo načelo
GLAVNI FUNKCIJI IS	Informacijske potrebe uprave	Informacijske potrebe uprave in zunanjih uporabnikov
ZASNOVA	Sektorska, parcialna	Integralna
VLOGA OBČANOV	Vir informacij	Vir in prejemnik informacij
NAČIN POSREDOVANJA INFORMACIJ	Parcialen, po resorjih	Vse na enem mestu
PREVERJANJE PODATKOV	Osebna navzočnost (<i>lastnoročni podpis</i>)	Na daljavo, digitalen podpis
DOKAZOVANJE URADNIH DEJSTEV	Stranka s pisnimi dokazili	Organ, ob uporabi javnih baz podatkov
SPREMLJANJE STANJ OBČANOV	Parcialno, resorno	Enkrat za vselej in vse primere

1.5 KLJUČNI DEJAVNIKI USPEŠNEGA RAZVOJA IS

- **STRATEŠKO NAČRTOVANJE INFORMATIKE:** ne moremo načrtovati dolgoročno. Oprema hitro zastara, zato je potrebno hitro prilagajati načrte in investirati.
- **OPREDELITEV KLJUČNIH DEJAVNIKOV USPEHA IN VLOGE IS:** vloga mora biti opredeljena znotraj strateškega načrta organizacije, da bo poslovanje učinkovito, ne zaradi izgleda. Ključni dejavniki uspeha organizacije – vloga IS – jih dviguje ali slabša.

- **SODELOVANJE IN PODPORA VODSTVA:** brez sodelovanja vodstva organizacije, le-ta ne more uspeti. Spremembe zahtevajo tudi nova znanja – usposabljanja.
- **SODELOVANJE UPORABNIKOV:** brez tega ni mogoče razviti dobrih sistemov. Uporabniki so zaposleni na področju, za katere razvijamo IS, zato morajo tesno sodelovati pri načrtovanju teh sistemov.
- **PROJEKTNI TIM:** mora biti sestavljen tako, da vključuje uporabnike in strokovnjake, vodstvo organizacije in načrtovalce. Uporabniki morajo imeti aktivno vlogo. Morajo imeti znanja, znanja informatikov in znanja uporabnikov pa se morajo prekrivati.
- **PROJEKTNI PRISTOP:** pomemben za uspešen razvoj. Načela projektnega vodenja:
 - **naloga** – jasno in natančno opredeljena,
 - **cilji** – prav tako,
 - določen mora biti **projektni tim**, ki je zadolžen za uresničitev projekta,
 - določen **vodja** projekta,
 - **roki** za izdelavo in
 - **vir** in **sredstva** za uresničevanje projekta.
- **UPORABA INFORMACIJSKIH ORODIJ:** orodja nam olajšajo delo, znižujejo stroške, skrajšajo čas, ...

1.5.1 Potrebna znanja informatikov in uporabnikov za uspešen razvoj IS



1.6 INFORMATIZACIJA KOT PRILOŽNOST ZA PRENOVO POSLOVANJA UPRAVE

- Od avtomatizacije k informatizaciji,
- vpliv novih tehnologij na poslovanje,
- temeljna izhodišča prenove poslovanja,
- prenova poslovanja na makro ravni,
- prenova poslovanja na mikro ravni,
- integracija procesov.

V začetku je bilo govora o avtomatizaciji, zdaj ga zamenjuje pojem informatizacija, ki je bistveno širši pojem. Najpomembnejše razlike med procesom avtomatizacije in procesom informatizacije organizacij:

KARAKTERISTIKE PRISTOPA	AVTOMATIZACIJA	INFORMATIZACIJA
NAČIN UVAJANJA	Od spodaj navzgor	Od zgoraj navzdol
VPLIV NA ORGANIZACIJO	Majhen, predvsem na operativno poslovanje	Velik, spremembe v organizacijski kulturi organizacije
POTREBNA TEHNOLOGIJA	Samostojno računalništvo, lokalne mreže	Lokalne in globalne mreže, internet
INICIATOR SPREMEMB	Nižji in srednji management	Vrhovni management
ODGOVORNOST ZA IZVEDBO	Nižji in srednji management	Vrhovni management
OBSEG SPREMEMB V POSLOVNIH PROCESIH	Majhna, predvsem v načinu izvajanja, ročna opravila se nadomešča z avtomatiziranim	Velike, možna je popolna prenova poslovnih procesov (BPR)
BAZE PODATKOV	Parcialne po poslovnih funkcijah	Integrirane za celotno organizacijo
UPRAVLJANJE INFORMACIJSKIH VIROV	Decentralizirano po organizacijskih enotah in poslovnih funkcijah	Decentralizirano ali centralizirano
VPLIV NA MANAGEMENT	Delen	Velik
VLOGA INFORMACIJSKE TEHNOLOGIJE V ORGANIZACIJI	Vpliv je čutiti predvsem na operativni in tehnični ravni	IT dobiva strateško vlogo, vse vitalne funkcije organizacije so odvisne od uporabe IT
SPREMEMBE V ORGANIZACIJSKI STRUKTURI	Običajno jih ni	Lahko tudi zelo velike, odvisno od narave organizacije in njenega vodenja
SPREMEMBE V NORMATIVNI UREDITVI	Niso nujne	Koristne, včasih celo pogoj za uspeh projektov informatizacije

Informatizacija je proces, ki do podrobnosti spreminja funkcioniranje neke organizacije, ki ga je potrebno razumeti kot priložnost za prenovu procesov.

1.7 TEMELJNA IZHODIŠČA PRENOVE POSLOVNIH PROCESOV

- a) Pristop po načelu »**nepopisanega lista papirja**«,
- b) preseganje obstoječih **organizacijskih struktur** in **procesna orientacija**,
- c) potreba po radikalni spremembi v pogledu **učinkovitosti poslovanja**,
- d) obravnava **informacijske tehnologije** kot vzvoda in sredstva za spremembe,
- e) sprememba **organizacije** in **organizacijske kulture** kot nujnega spremljevalca sprememb.

Ad a) PRISTOP PO NAČELU »nepopisanega lista papirja«

Pomeni, da pozabimo vse kar smo delali do sedaj in začnemo popolnoma na novo. Izkušnje: V praksi žal ni mogoče pozabiti vsa organizacijska znanja (*razlogi: npr. zakonodaja, obstoječe organizacijske rešitve – so zelo toge,...*).

Ad b) PROCESNA ORIENTACIJA

Danes je uprava zasnovana hierarhično, da lahko izvaja zakone. Sami procesi, ki so vezani na nudenje upravnih storitev pa potekajo horizontalno (*npr. povezava več upravnih organov na istem nivoju*). Procesna orientacija je danes eden temeljnih trendov v razvoju upravnih sistemov, pomeni pa bistveno reorganizacijo obstoječih organizacijskih struktur.

Ad e) SPREMEMBA ORGANIZACIJE IN ORGANIZACIJSKE KULTURE

Ni povezave poslovnih procesov brez sprememb v organizacijski kulturi. Ključni dejavniki tu, so ljudje (*spremembe poslovnega procesa in spremembe v organizacijski strukturi*).

2 METODOLOGIJE NAČRTOVANJA IN GRADNJE IS

1. Opredelitev osnovnih pojmov
2. Pregled dosedanjega razvoja
3. Ocena trenutnega stanja
4. Vloga modeliranja pri načrtovanju in gradnji IS
5. Opredelitev pomembnejših pristopov:
 - a. linearni pristop
 - b. prototipni pristop
 - c. objektni pristop
6. Osnovne značilnosti sodobne metodologije
7. Opredelitev značilnih razvojnih faz

2.1 OPREDELITEV OSNOVNIH POJMOV

Metodologija

Je po definiciji skupek postopkov, tehnik, metod, ki jih uporabljamo pri reševanju nekega problema.

Pod pojmom metodologija gradnje IS si vsaj v praksi predstavljamo **organizacijsko tehnično znanje**, ki ga uporabljamo pri zasnovi in izdelavi računalniških rešitev.

METODA je način, tehnika izvajanja posameznih faz v IS. Disciplina je zelo mlada, še vedno nimamo na voljo metodologij, ki bi nas natančno in zanesljivo vodila skozi celoten proces. IS sodijo v kategorijo izredno kompleksnih in abstraktnih sistemov.

GLAVNI ELEMENTI celovite metodologije:

- ključne razvojne faze jasno **opredeljene** in tudi njihovo **zaporedje**,
- vsebinski **opis** vsake faze z opredelitvijo ključnih aktivnosti,
- **navodilo** za izvedbo aktivnosti,
- prikaz **metod** in **tehnik** za izvedbo aktivnosti,
- opredelitev zahtevanih **rezultatov** posamezne faze,
- opredelitev **kriterijev**,
- navodila glede **kadrovskih** in **tehničnih** pogojev,
- opredelitev **področja uporabnosti**.

KLJUČNI PROBLEMI razvoja IS:

- **predolgi** razvojni cikli IS (*želja je, da je razvoj IS pripravljen kar čez noč, ti cikli pa lahko trajajo po nekaj mesecev ali celo nekaj let - metodologija naj bi čimbolj skrajšala razvoj*),
- nizka **produktivnost** zaposlenih,
- nepredvidljiva **kvaliteta končnega** programskega produkta (*v fazi razvoja težko natančno predvidimo kvaliteto končnega produkta*),
- **visoki** razvojni in vzdrževalni **stroški**.

Razvoj programske opreme zaostaja za razvojem strojne.

Ocena stanja

Ocena stanja je, da nismo in ne moremo biti zadovoljni – metodologije so dostopne in to povzroča precej težav.

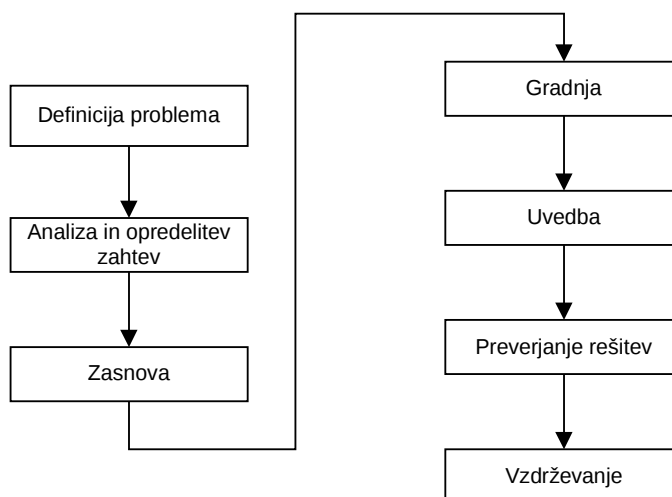
IS opisujejo realni svet

Skušamo izdelati najboljši model realnega sveta, uporaba tehnike temelji na modelih. Model je sredstvo, ki omogoča najboljšo predstavitev lastnosti bodočega izdelka (*imamo opravka s fizičnimi modeli in objekti*). Pri IS pa so modeli, ki so abstraktne narave in si ga je zato težje predstavljati in opredeliti, saj je v računalniku in je težko predstavljen.

2.2 PRISTOPI K NAČRTOVANJU IN GRADNJI IS

Življenjski cikel IS:

Definicija problema => analiza in opredelitev zahtev => zasnova => gradnja => uvedba => preverjanje rešitev => vzdrževanje.



Tega življenjskega cikla se (*linearni, prototipni, objektni*) pristopi lotevajo vsak po svoje.

2.2.1 Linearni pristop

- Temelji na »**kaskadnem principu**«: razvoj IS je razdeljen na določeno število razvojnih faz, ki si sledijo po vnaprej določenem zaporedju, so natančno definirane in dokler ni v celoti zaključena predhodna faza, ni mogoče nadaljevati z naslednjo fazo,
- faze so dobro **dokumentirane**,
- pristop je **neodvisen od vrste problemov** - lahko gre za velike ali majhne IS, (*univerzalen pristop, neodvisen od uporabljenih orodij*).

- **SLABOSTI:**
 - predolgi razvojni cikli,
 - visoki **razvojni stroški**,
 - napake in pomanjkljivosti se odkrijejo šele **na koncu**,
 - **sodelovanje uporabnikov** je otežkočeno ali onemogočeno.

2.2.2 Prototipni pristop

- nastal kot odgovor na ugotovljene slabosti linearnega,
- uvaja pojem **prototipa** v načrtovanje in gradnjo IS. Prototip se razvije v tesnem **sodelovanju z uporabniki**, ki že vsebuje njegove ključne karakteristike – vidi se ali gre razvoj v pravo smer, ugotovi se pomanjkljivosti, dokler se ne razvije do dovršenosti. Prototip ne zavržemo, to je evolutivni pristop, vloga uporabnikov je ključna.
- **SLABOSTI:**
 - faze **niso natančno opredeljene**, niti vsebina,
 - celoten razvojni proces **ni dobro evidentiran**,
 - sisteme, ki so bili na ta način razviti, je kasneje **težko vzdrževati**.

Uporabljamo ga kot dopolnilo linearnemu pristopu in za manj zahtevne rešitve.

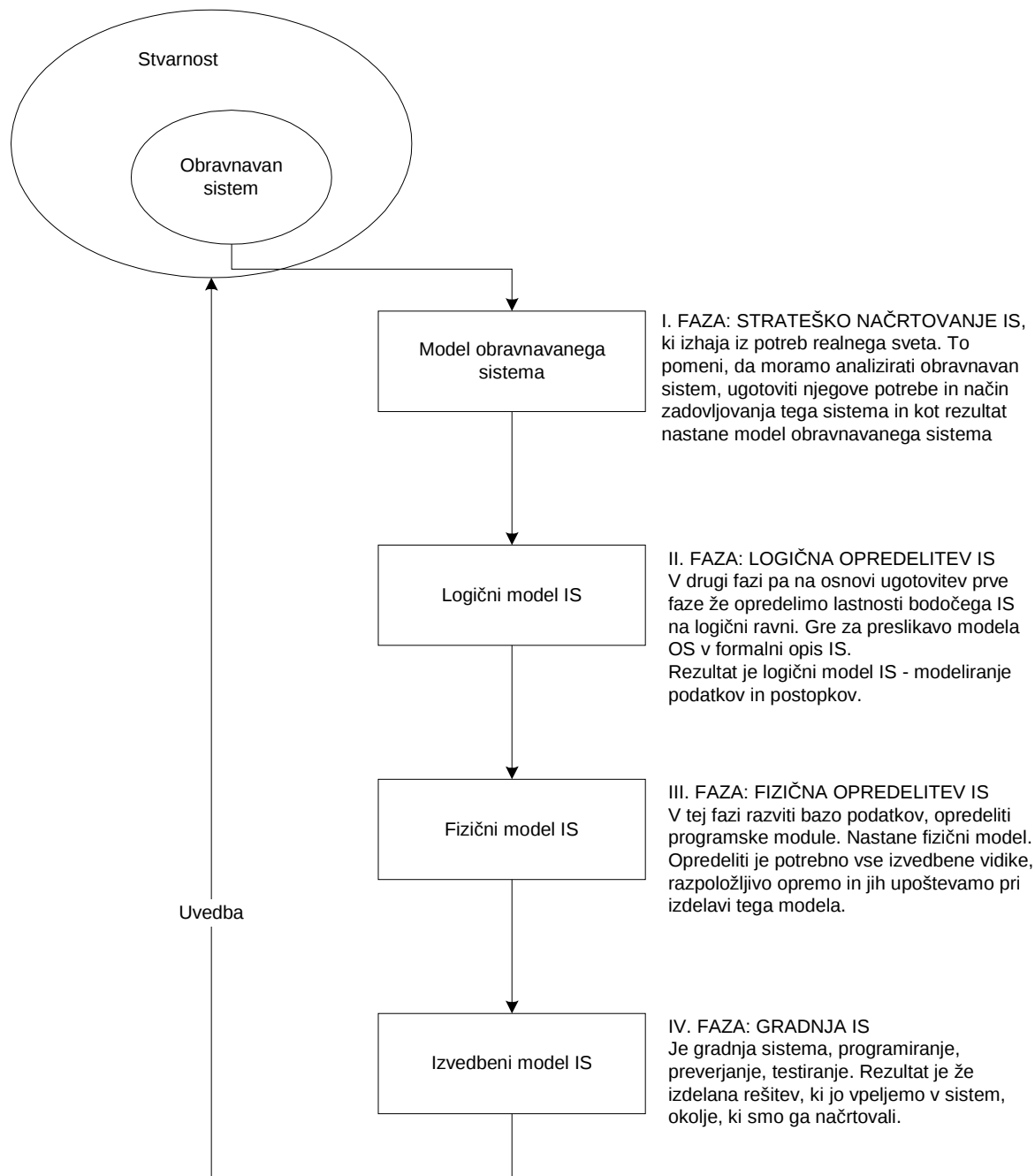
2.2.3 Objektni pristop (še v razvoju)

- je nastal iz želje odpraviti slabosti linearnega pristopa. To je bilo izhodišče, naprej pa nima primerjave s prejšnjima dvema,
- stoji na treh temeljnih izhodiščih:
 - **OBJEKT** (*osrednji koncept*) - izvzeti so iz realnega sveta. Uporabljamo čimbolj identične objekte, kot jih bomo naredili. Objekti vsebujejo **podatkovne strukture** in **pripadajoče postopke** na teh strukturah (*enovita obravnava podatkov in postopkov*). Vsak IS zajema dva vidika poslovnega sistema ali realnega sveta: en vidik se nanaša na **podatke**, ki se premikajo skozi poslovni sistem, drugi vidik pa so **postopki**, s katerimi se podatke obdeluje. Podatke in postopke obravnava ta pristop povezano. Objekt se sestoji vedno iz dveh delov – na eni strani podatki, na drugi postopki.
 - **SPOROČILO** – sredstvo, s pomočjo katerega objekti komunicirajo med seboj pri vzpostavljanju postopkov,
 - **TIPI OBJEKTOV** – omogočajo realizacijo konceptov abstrakcije,
- poizkus **standardizacije elementov** informacijskih rešitev - gre za uporabo ideje sistema lego kock (*IS gradimo iz bolj ali manj standardiziranih elementov, ki jih lahko večkrat uporabimo na različnih rešitvah*). Nove rešitve so sestavljene iz že obstoječih in preizkušenih objektov – spreminjanje individualnih objektov ter zmanjšanje stroškov.

2.2.4 Značilne razvojne faze is v okviru sodobnejših postopkov

IS vedno razvijamo za neko konkretno poslovno okolje, ki je del realnega sveta. To okolje je odvisno od naše odločitve, na kaj se bomo osredotočili - obravnavan sistem je en del realnega sveta.

Če želimo razviti IS za ta obravnavan sistem najprej pristopimo k



Razlog za ločitev na logični in fizični model je v tem, ker se tehnologija hitro spreminja. Logični model je bolj stabilen, fizični je odvisen od izvedbenih karakteristik. Fizični se bo bistveno pogosteje spreminjal.

Prehod iz II. faze na III. fazo pomeni prehod logične izvedbe na tehnološko. V fazi III se razvoj konča, v IV. fazi pa se izdelana rešitev vpelje v obravnavano okolje.

I. FAZA: STRATEŠKO NAČRTOVANJE IS (modeliranje obravnavanega sistema)

Vedno začnemo z analizo obravnavanega sistema:

1. **organiziranost** obravnavanega sistema (*temeljita analiza*),
2. **funkcija** obravnavanega sistema (*v okviru celotnega poslovnega sistema organizacije*).
3. analiza **informacijskih tokov** (*kako se podatki pretakajo; interni in eksterni tokovi*),
4. analiza **procesov in postopkov**,
5. opredelitev **informacijskih potreb** (*kar se da natančno*),
6. **strateški načrt informatizacije** (*proces je potrebno gledati kot celoten proces informatizacije v neki organizaciji*).

Osrednji pomen I. faze je ugotavljanje informacijskih potreb.

II. FAZA: LOGIČNO MODELIRANJE IS

1. **značilnosti** logičnega modela (*predstavlja vsebinske lastnosti obravnavanega sistema*),
2. **preslikava** obravnavanega sistema v model,
3. **tradicionalni pristop**:
 - a. podatkovni model (*opredeljuje vse lastnosti podatkov*),
 - b. postopkovni model (*opredeljuje vse lastnosti postopkov*),
4. **objektni pristop**:
 - a. objektni model.

Tukaj gre predvsem za vsebinske značilnosti. **Podatkovni model** – lastnosti, struktura podatkov poslovnega sistema, za katerega ta sistem gradimo, na drugi strani pa razvijamo **postopkovni model**, ki predstavlja neločljivo povezano celoto.

V tej fazi nastaneta dva modela – model podatkov in model postopkov.

III. FAZA: IZDELAVA FIZIČNEGA MODELA

Značilnosti:

- **lastnosti** fizičnega modela,
- **funkcija** fizičnega modela,
- **prehod** iz logičnega na fizični model (*normalizacija*).

Uporablja se mrežni, relacijski model.

3 MODELIRANJE INFORMACIJSKIH SISTEMOV

Splošne značilnosti:

1. IS in stvarnost,
2. predstavitev znanja o IS,
3. glavni koncepti modeliranja IS.

Ad 1) IS IN STVARNOST

Med IS in poslovnim sistemom so neke zelo tesne povezave. IS predstavlja neko preslikavo poslovnega sistema. V tej preslikavi se osredotočimo na vidike IS - pri vsaki preslikavi pa pride do popačenja.

PRESLIKAVA STVARNOSTI V MODEL:

Na eni strani imamo stvarnost – objekte, ki nastopajo, mi pa bi radi naredili model tega realnega sveta.

Pri izdelavi tega modela se srečamo z naslednjimi problemi:

- model delamo vedno na osnovi **predstave** o stvarnosti, ki si jo naredi posamezni opazovalec,
- na **osnovi te predstave v glavi** izdelamo model, zato pride do popačenja.



Drug problem pa je, da je kakovost tega modela odvisna še od:

- **pogleda** na svet,
- **inštrumentarija**, ki ga imamo na voljo.



Zato bo preslikava pri ljudeh različna. Tudi orodja, ki jih imamo na voljo, vplivajo na izdelavo modela. Pri modeliranju IS nam ta inštrumentarij predstavljajo različni abstraktni koncepti.

Ad 2) PREDSTAVITEV ZNANJA O IS

Kako zdaj predstaviti ta poslovni sistem, informacijske vidike, s kakšnimi sredstvi, glede na to, da gre za sisteme, ki niso fizične narave.

Ad 3) GLAVNI KONCEPTI MODELIRANJA IS

Seznam konceptov modeliranja IS:

- entiteta
- atribut
- povezava
- vrednost

Uporabljamo za modeliranje podatkov

- postopek
- zunanja entiteta
- zbirka podatkov
- informacijski tok
- dogodek
- sporočilo

Uporabljamo pri modeliranju postopkov

Modeliranje podatkov ponavadi teče vzporedno z modeliranjem postopkov.

4 MODELIRANJE PODATKOV

Podatkovni model – je zbirka konceptov, s katerimi skušamo izraziti statične in dinamične lastnosti podatkov v okviru IS.

Statične lastnosti – so tiste, ki se s časom ne spreminjajo (*struktura podatkov*),

dinamične lastnosti – razumemo vse spremembe, ki jih podatki doživljajo v okviru IS.

Zasnova podatkovnih modelov temelji na določenih konceptih abstrakcije, s katerimi si pomagamo pri predstavitvah posameznih lastnosti podatkov.

Abstrakcija je način razmišljanja oziroma reševanja problema, kjer zavestno **zanemarimo podrobnosti** in se osredotočimo na splošne (*skupne*) lastnosti pojavov, objektov, pojmov, odvisno od narave problema, ki ga rešujemo.

4.1 GLAVNI KONCEPTI ABSTRAKCIJE

1. Klasifikacija
2. Generalizacija
3. Agregacija
4. Asociacija

DEFINICIJE:

Entiteta – je neka reč (*objekt, subjekt, pojem,...*), ki obstaja v realnem svetu in je pomembna z vidika načrtovanega IS in o kateri nameravamo zbirati podatke. Entitete so lahko fizične ali abstraktne narave - so objekti, subjekti, pojmi, ki so pomembni za obravnavan sistem, v okviru katerega se zbirajo podatki o teh entitetah.

Atribut – je **opisna lastnost** nekega tipa entitete, ki jo lahko pripišemo celotni množici primerkov danega tipa (*primer: študent = tip entitete; ime, priimek, spol = atributi tipa entitete*).

KLASIFIKACIJA

Je koncept abstrakcije, pri katerem **množici primerkov entitet**, ki imajo neko skupno lastnost, **priredimo skupen tip entitete**. Tip entitete odraža neko skupno lastnost množice primerkov entitet (*tip entitete: študent; primerek entitete: Janez, Miha,...*).

GENERALIZACIJA

Gre za postopek abstrakcije, pri katerem **množici tipov entitet**, ki imajo neko skupno lastnost, priredimo nek **splošnejši tip entitete na višjem nivoju**.

Primer:

Občan – splošen tip entitete,

Študent, zaposlen, upokojenec – tip entitete (vsi spadajo pod splošen tip entitete OBČAN).

AGREGACIJA

Je oblika abstrakcije, pri kateri se objektom, ki predstavljajo **dele nekega sestavljenega objekta**, priredi **sestavljen objekt na višjem nivoju**.

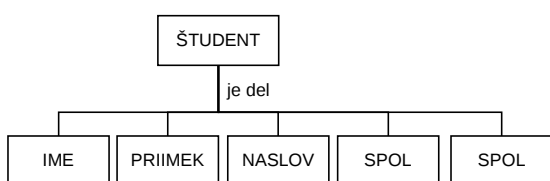
Uporablja se na dva načina:

- kartezična agregacija: posameznemu tipu entitete se priredijo njihovi atributi,
- na ravni objektov.

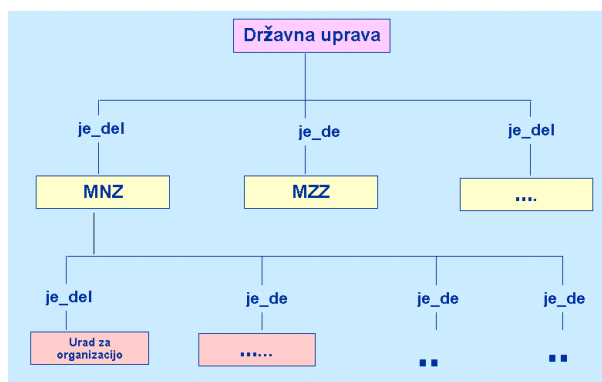
ASOCIACIJA

Je oblika abstrakcije, pri kateri se **podmnožici primerkov nekega tipa**, priredi **na višjem nivoju (asociirani) tip**, ki **predstavlja skupne lastnosti (asociiranih) primerkov**.

Slika kartezične agregacija na posameznem tipu entitete



Slika kartezične agregacija na ravni objektov



4.2 RAZVOJ PODATKOVNIH MODELOV

PRIMITIVNI MODEL

Razvile so se preproste podatkovne strukture na ravni datotek, preprosta struktura (*polje, zapis, datoteka – malo podatkov*).

KLASIČNI MODELI

Hierarhični, mrežni, relacijski model - za njih velja, da so dobri na **izvedbeni ravni**, niso pa prikladni za modeliranje podatkov na logični ravni - **manjka jim izrazne moči**. Zaradi te pomanjkljivosti so se razvili *semantični modeli*. Ti trije modeli so usmerjeni na izvedbo (na ravni fizičnega modeliranja podatkov in izvedbeni ravni), na logični ravni niso koristni.

SEMANTIČNI MODELI

So nastali iz potrebe, da bi z njimi čimbolje **izrazili pomen in vlogo**, ki jo igra podatek v okviru obravnavanega poslovnega sistema - omogočajo dobro predstavitev (*E-R model*).

SPECIALIZIRANI MODELI

So namenjeni specializiranim področjem – za potrebe dokumentacijskih IS.

4.2.1 KLASIČNI MODELI (Izvedbeni podatkovni modeli)

1. Hierarhični model
2. Mrežni model
3. Relacijski model

Vsi trije so naslov dobili okoli 70-ih let. Potem je bila njihova vloga različna. Najprej se je uveljavil hierarhični, potem za njim mrežni, relacijski pa je potreboval največ časa, da se je uveljavil v praksi - danes je prevladujoč model.

Najpomembnejša razlika med njimi je v tem, kako je **predstavljena struktura podatkov**, ki nam nastopajo v okviru obravnavanega IS.

Vsak od teh modelov v bistvu predstavlja drugačno strukturo podatkov - iz tega izvirajo tudi glavne razlike med njimi.

Ad 1) HIERARHIČNI MODEL – temelji na **hierarhični** ali **drevesni strukturi**. Za ta model veljata dve pravili:

- prvo pravilo, da je na najvišji ravni **samo en vozel**,
- drugo pravilo »**oče – sin**« - vsak vozel ima lahko poljubno število podrejenih zapisov ali vozlov in le enega samega nadrejenega.

Slaba stran je, da se v tako podatkovno strukturo vstopa samo **od vrha navzdol** in samo **po eni poti**. V vsakem zapisu imamo poleg podatkov še en delček, ki mu pravimo **kazalec**, ki vsebuje naslov naslednjega zapisa v verigi. **Povezave so vnaprej opredeljene, opredeljene pa so tudi poti, po kateri pridemo do podatka**. Ta model je učinkovit za situacije, ki so vnaprej predvidene in vgrajene v model - čim želimo dostop do zapisov po nekem drugem zaporedju, ki ni bilo vnaprej predvideno, se začno kazati slabosti tega modela.

Ad 2) MREŽNI MODEL – je razširitev hierarhičnega modela s tem, da sta pri mrežnem modelu **odpravljeni obe pravili**, ki veljata za hierarhični model - imamo lahko tudi na najvišji ravni več zapisov in več nadrejenih.

V tako strukturo lahko vstopamo **po več poteh** in tako pridemo tudi do podatkov po več poteh. Ta model izboljšuje lastnosti - **je zelo uporaben, dokler iščemo podatke na način, ki je bil vnaprej predviden v snovanju takega modela**. Osnovna slabost pa še vedno ostaja – če želimo priti do zapisov po nekem drugem zaporedju, ki ga pri snovanju nismo predvideli.

Ad 3) RELACIJSKI MODEL – odpravlja prej omenjeno slabost, ker temelji na čisto drugačni podatkovni strukturi - je edini od teh treh, ki je bil razvit na osnovi matematičnih pravil (*naslonjen je na matematično teorijo relacij*).

Pri tem modelu so podatki v bazi podatkov zbrani v **dvodimenzionalnih tabelah**, ki se imenujejo relacije in te relacije so oblikovane okrog tipov entitet. Pogosto gre kar za direktno preslikavo tipov entitet v relacije (*stolpci predstavljajo attribute, vrstice pa primerke entitet*).

Pri tem modelu povezave med podatki niso vnaprej določene, ampak se določajo v času obdelave na osnovi uporabniških zahtev (*povezave se vzpostavijo v času obdelave*).

5 MODEL ENTITETA – POVEZAVA (E-R MODEL)

(Entity – Relationship)

Nastal je v drugi polovici 70-ih let in se naglo širil na področju modeliranja IS na logični ravni – vsebinski, funkcionalni ravni. Danes je najbolj razširjen semantični model in najpomembnejši model za modeliranje na logični ravni.

Temelji na treh osnovnih konceptih:

1. Entiteta (*nekaj o čemer se podatki zbirajo*)
2. Atribut (*opisuje lastnosti tipov entitet*)
3. Povezava

PREDNOSTI E-R MODELA:

- usmerjen je v predstavitev podatkov **kot jih vidi uporabnik**,
- združuje lastnosti **mrežnega** in **relacijskega** modela,
- enostavnost:
 - **konceptov**,
 - **transformacije**,
- predstavlja **statične** in **strukturne** lastnosti podatkov.

SLABOSTI E-R MODELA:

- **ne omogoča**:
 - enotnega pogleda na podatke,
 - modeliranja postopka,
- **ni univerzalen** model – ne podpira vseh faz modeliranja podatkov,
- **ni standardiziran** v pogledu:
 - uporabljenih konceptov,
 - grafične notacije,
- **šibak** je v pogledu predstavitve **dinamičnih lastnosti** podatkov.

5.1 ENTITETA

Tipi entitet:

- so primerki z **enakimi lastnostmi**,
- **vsa imena** tipov entitet morajo biti **enolična**,
- **predstavljeni** so:
 - z imenom,
 - s seznamom atributov,

- entitetna množica vsebuje vse atributne vrednosti atributov tipa entitete.

Predstavitev tipa entitete:

Državljan – **tip entitete**

Ime, spol, poklic – **atributi**

Marko, M, Mehanik – **primerek entitete**

5.2 VRSTE ATRIBUTOV

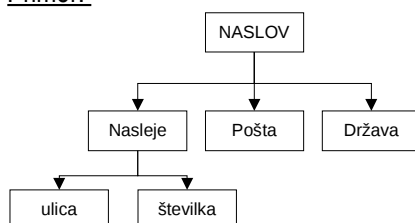
ELEMENTARNI ATRIBUT

Tisti, ki ga ni **mogoče razstaviti** na manjše dele (*spol, priimek, ...*).

SESTAVLJENI ATRIBUT

Te je **mogoče razstaviti** na elementarne attribute.

Primer:



VREDNOST ATRIBUTA

Je podatek, ki nastopa pri posameznih atributih.

Primer:

Atribut: IME SPOL POKLIC

Vrednost: Teja Ž Direktor

DOMENA ATRIBUTA

Je **množica vseh vrednosti**, ki jih nek atribut lahko zavzame.

Primer:

Atribut: IME – vsa imena

SPOL – samo M ali Ž

ENOVREDNOSTI ATRIBUTI

Pri posameznem primerku entitete zavzamajo vedno **eno samo vrednost** (*na primer: rojstni datum*).

VEČVREDNOSTNI ATRIBUTI

Pri posameznem primerku entitete zavzamejo **več vrednosti** (*npr. poklic*).

KLJUČNI ATRIBUTI

Se uporabljajo za **iskanje** in **vzpostavljanje povezav** med primerki različnih tipov entitet;

- primarni ključ,
- sekundarni ključ,
- tuji ključ,
- speti ključ.

PRIMARNI KLJUČ

Za attribute, ki igrajo vlogo primarnega ključa velja, da morajo zagotavljati enolično identifikacijo vseh primerkov entitete, ki nastopajo v okviru določenega tipa entitete. Vsak tip entitete ima en sam atribut, ki igra vlogo primarnega ključa, kandidatov za primarni ključ pa je lahko več (*npr. za študenta: EMŠO, davčnašt., vpisna številka*) - moramo se odločiti in izbrati samo en primarni ključ.

En primerek tipa entitete ima en sam primarni ključ, en primarni ključ pripada enemu samemu primerku tipa entitete.

SEKUNDARNI KLJUČ

So tisti atributi, po katerih se podatki **pogosto iščejo** (*npr. za študenta: priimek in ime, leto rojstva*). Sekundarnih ključev je lahko več in nam služijo kot iskalni pripomoček za iskanje podatkov. Za sekundarni ključ ni več nujno, da izpolnjuje pogoj enoličnosti. *Vlogo sekundarnega ključa igrajo vsi atributi, razen primarnega ključa.*

SPETI KLJUČI

So vezani predvsem na primarne ključe. Idealno je, če ima določen tip entitete nek naraven atribut, ki omogoča enolično identifikacijo. Največkrat pa moramo vpeljati nek umeten atribut. Včasih pa se lahko izognemo vpeljavi umetnega atributa pri določanju primarnega ključa, tako da do njega (*primarnega ključa*) pridemo s sestavljanjem več naravnih atributov in takemu ključu rečemo speti ključ. Speti ključ je vedno primarni ključ.

TUJI KLJUČI

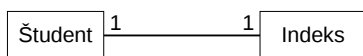
So primarni ključi neke druge entitete – primarni ključ tistega tipa entitete, s katerim je ta v povezavi. Uporabljamo ga za vzpostavljanje povezav med entitetami, ki so v nekem razmerju.

5.3 LASTNOSTI POVEZAV

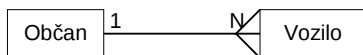
- **tip, primerek povezave** (*kot pri entitetah moramo tudi tu ločiti med tipi in primerki povezav*),
- **ime povezave** (*vsaka povezava ima svoje ime, ki opredeljuje vsebino razmerja*),
- **stopnja povezave** (*opredeljuje število tipov entitet, ki sodelujejo v neki povezavi*). Poznamo:
 - o binarne (*so najbolj pogoste*),
 - o ternarne,
 - o rekurzivne (*sama v sebe*),
- **kardinalnost povezave** – opredeljujemo **število primerkov enega tipa entitete, ki sodeluje v povezavi s primerki drugega tipa entitete.**

Ločimo tri vrste:

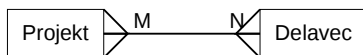
ena : ena (1 : 1)



ena : več (1 : N)

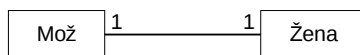
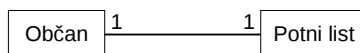
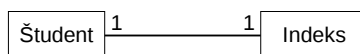
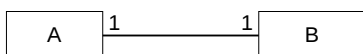


več : več (M : N)

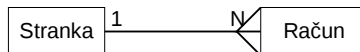
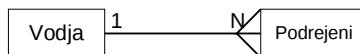
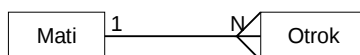
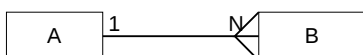


Primeri:

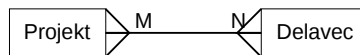
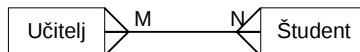
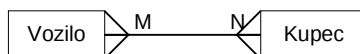
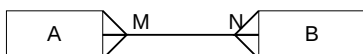
Primer za (1:1) – pomeni, da je en primerek tipa entitete A povezan z enim samim primerkom tipa entitete B.



Primer za (1:N) – en primer tipa entitete A povezan z enim ali več primerki tipa entitete B.

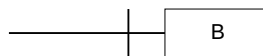


Primer za (M:N) – en primer tipa entitete A je povezan z enim ali več primerki tipa entitete B ali en primer tipa entitete B povezan z enim ali več primerki tipa entitete A.

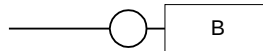


- **obveznost povezave/neobveznost povezave**

Če je povezava obvezna, označimo:

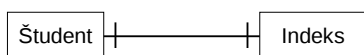


Če je povezava neobvezna, označimo:

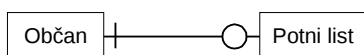


Primeri:

Povezava 1:1

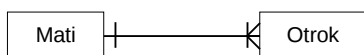


Študent mora imeti vedno indeks in to en sam.

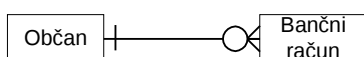


Občan ima lahko 0 ali 1 potni list.

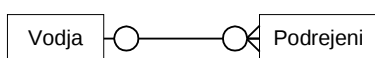
Povezava 1:N



Otrok mora imeti mater; če je mati mora imeti vsaj enega otroka.

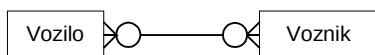


Občan ima lahko 0,1 ali več bančnih računov.

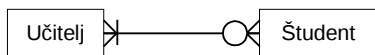


Vodja ima lahko 0, 1 ali več podrejenih; podrejeni ima lahko 0 ali enega vodjo.

Povezava M:N





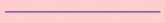


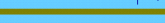


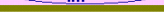


Vozilo ima 0, 1 ali več voznikov in obratno.



Učitelj ima lahko 0, 1 ali več študentov, študent ima lahko 1 ali več učiteljev.

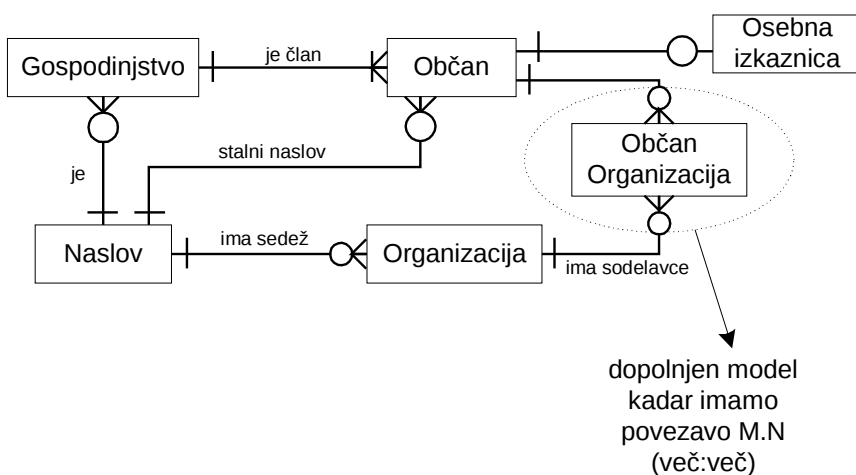
5.4 GRAFIČNA NOTACIJA E-R MODELA

Tip entitete		
Šibki tip entitete		
Tip povezave		
Tip povezave (druga možna)		
Kardinalnost tipa povezave	1 : 1	
	1 : N	
	M : N	
Obveznost povezave	obvezne	
	neobvezne	
Atributi	ključni	
	ostali	

5.5 RAZVOJ E-R MODELA

Kako razvijemo E-R model?

- Najprej **analiziramo poslovne procese** (to je osnova kajti skozi analizo ugotovimo vse informacijske potrebe),
- **identifikacija tipov entitet**,
- za te tipe entitet **ugotovimo njihove povezave** in narišemo model,
- **opredelimo lastnosti povezav**.



E-R model (E-R graf):
 Razberemo lahko tip entitet, povezave in lastnosti povezav, nič pa ne moremo razbrati o atributih.

5.6 PODATKOVNI SLOVAR

Njegov namen je, da nam podrobno **dokumentira** in **opredeli vse lastnosti podatkovnega modela** (tipe entitet, njihove attribute, vloge teh atributov, lastnosti atributov, tipe in lastnosti povezav), zato se sestoji iz:

1. SLOVARJA ENTITET
2. SLOVARJA ATRIBUTOV
3. SLOVARJA POVEZAV

5.6.1 Slovar entitet

Oznaka entitete	naziv entitete	atributi
E01	OBČAN	EMŠO#, priimek, ime, datum rojstva, spol, davčna številka občana, šifra naselja+šifra ulice+hišna številka, EMŠO nosilca gospodinjstvo
E02	GOSPODINJSTVO	EMŠO nosilca gospodinjstva#, število članov, datum vzpostavitve, datum ukinitve, šifra naselja+šifra ulice+hišna številka
E03	NASLOV	šifra naselja+šifra ulice+hišna številka#, številka parcele, geografski centroid, popisni okoliš, statični okoliš.
E04	ORGANIZACIJA	šifra organizacije#, naziv organizacije, dejavnost, številka ŽR, davčna številka organizacije, šifra naselja+šifra ulice+hišna številka
E05	OSEBNA IZKAZNICA	številka OI#, datum izdaje, datum veljavnosti, kraj izdaje, EMŠO
E06	OBČAN / ORGANIZACIJA	EMŠO+šifra organizacije#

Je dvodimenzionalna tabela, ki nam opredeljuje **vse tipe entitet**, ki sodelujejo, njihove **attribute** ter **vloge teh atributov**. Pod vlogo atributa razumemo predvsem to ali je nek atribut zgolj navaden atribut ali pa igra vlogo nekega ključa (*primarnega, sekundarnega, tujega, spetega*).

Na prvem mestu je običajno tisti ključ, ki igra vlogo primarnega ključa (*označeno z # in enkrat podčrtano, dvakrat podčrtani so tuji ključi, sekundarni ključ je enkrat podčrtan*). Kadar med navadnimi atributi tipa entitete ni nobenega takega, ki bi zagotavljal enolično identifikacijo njegovih primerkov, se za tvorbo primarnega ključa poslužimo spetega ključa (+ med atributi).

5.6.2 Slovar atributov

oznaka atributa	naziv atributa	standardno ime	tip	dolžina	standardna vrednost
OBČAN					
A0101	EMŠO	EMSO	N	13	
A0102	Priimek	Priim	A	20	
A0103	Ime	Ime	A	15	
A0104	Spol	Spol	A	1	M-moški Ž-ženski
A0105	Davčna številka občana	Dav-st-obc	N	8	
A0106	Datum rojstva	Dat-roj	D	8	
ORGANIZACIJA					
A0401	Šifra organizacije	Sif-org	N	10	
A0402	Naziv organizacije	Naz-org	AN	35	
A0403	Šifra dejavnosti	Sif-dej	N	7	
A0403	Davčna številka organizacije	Dav-st-org	N	8	
A0404	Številka ŽR	Stev-ZR	AN	15	

Opisuje **lastnosti** - karakteristike **atributov posameznega tipa entitete**. Za vsak tip entitete imamo svoj slovar atributov. Lastnosti atributov morajo biti enolično označene.

DOLOČIMO:

oznako atributa (A-0101), ime atributa (opisno ime), standardno ali skrajšano ime (uporablja se pri vseh opisih atributa v računalniških programih do 8 znakov, Vsako ime mora biti sestavljeno tako, da je napisano z neprekinjenim nizom znakov, brez šumnikov v imenu - npr. VP_STEV).

Tip atributa - ločimo tri tipe:

- **numerični** atributi (**N**): zanje velja, da le ti lahko zavzamejo zgolj numerične vrednosti, številke (npr. VP_STEV = vpisna številka študenta),
- **alfabetški** atributi (**A**): zanje velja, da le ti lahko zavzamejo vrednosti, ki so sestavljene samo iz črk (priimek in ime);
- **alfanumerični** atributi (**AN**): to so atributi, katerih vrednosti so sestavljene iz poljubnih števil in črk (naslov).

Ti opisi atributov nam koristijo pri pravilnosti vnesenih podatkov.

Naslednja lastnosti je dolžina - z njo razumemo **maksimalno število znakov**, ki jih **vrednost nekega atributa lahko zavzame**.

Standardne vrednosti določimo lahko **samo pri nekaterih atributih** (npr. spol = M in Ž).

OZNAKA ATRIBUTA	IME ATRIBUTA	STANDARDNO IME	TIP	DOLŽINA	STANDARDNA VREDNOST
A-01	Vpisna številka	VPIS_ST	N	5	
A-02	Priimek	PRIIMEK	A	24	
A-03	Ime	IME	A	18	
A-04	Spol	SPOL	A	1	M – moški Ž – ženski

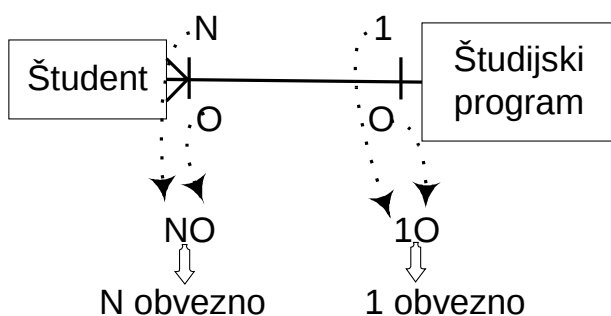
5.6.3 Slovar povezav

oznaka povezave	povezani entiteti	ime povezave	kardinalnost in obveznost
P01	OBČAN: GOSPODINJSTVO	je-član	Mo:1o
P02	GOSPODINJSTVO: NASLOV	je	Mn:1o
P03	OBČAN: NASLOV	stanuje	Mn:1o
P04	OBČAN: OSEBNA IZKAZNICA	IMA	1o:1n
P05	ORGANIZACIJA: NASLOV	ima-sedež	Mn:1o
P06	OBČAN: ORGANIZACIJA	sodeluje-z	1o:Mn
P07	ORGANIZACIJA: OBČAN/ORGANIZACIJA	ima sodelavce	1o:Mn

Vsebuje opis **lastnosti vseh povezav**, ki nam nastopajo v okviru obravnavanega IS.

Tu so prisotne, za vsako povezavo, oznaka povezave (P-01), povezani entiteti (STUDENT:ST. PROGRAM), ime povezave (je-opravlil), kardinalnost in obveznost oz. neobveznost (NO:1O).

Primer:



OZNAKA	POVEZANI ENTITETI	IME POVEZAVE	KARDINALNOST in OBVEZNOST
P-01	STUDENT: PROGRAM	je_vpisan	NO : 1O
P-02	STUDENT : OPR_OBVEZNOST	je_opravlil	1O : NN

Ko smo model predstavili v grafični obliki in ko smo ga dokumentirali s temi tremi slovarji, smo razvoj modela zaključili.

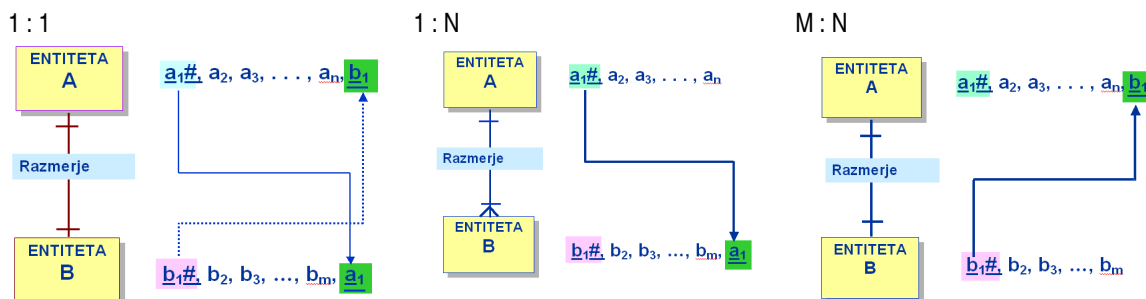
VGRADNJA POVEZAV S POMOČJO TUJIH KLJUČEV

Tuji ključi nam omogočajo vzpostavitev povezav med tipi entitet, ki so med seboj v nekem razmerju - računalniku omogočajo ugotavljati razmerja med tipi entitet.

TUJI KLJUČ je vedno primarni ključ tistega tipa entitete, s katerim je ta tip entitete v razmerju - omogočajo vzpostavljanje povezav.

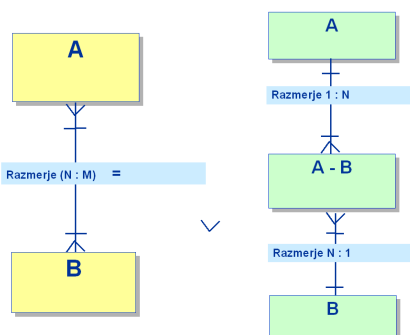
Na kakšen način se vgradijo pa je odvisno od kardinalnosti, imamo 3 možne primere:

Primeri »Tujega ključa«:



Napaka v sliki !! Obe povezavi sta M:N. Potrebno je uvesti presečno entiteto.

5.7 IZVEDBENA FAZA IN M:N KARDINALNOST



Ko gremo v izvedbeno fazo, se moramo kardinalnosti M:N vedno znebiti, ker jih ni mogoče izvesti. Ta problem rešimo tako, da **razmerje M:N razstavimo vedno v dve povezavi 1:N**. Od tu naprej je postopek enak kot pri kardinalnosti 1:N.

5.8 MOČNI IN ŠIBKI TIPI ENTITET

1. Močni tipi entitet
2. Šibki tipi entitet
3. Eksistenčna odvisnost
4. Identifikacijska odvisnost

Ad 1) MOČNI TIPI ENTITET

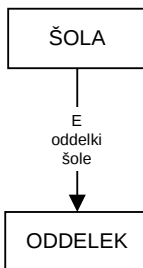
Za močne tipe entitet velja, da lahko njihovi primerki obstajajo neodvisno od primerkov nekaterih drugih tipov entitet.

Ad 2) ŠIBKI TIPI ENTITET

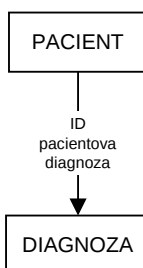
Obstajajo pa tudi tipi entitet, katerih obstoj primerkov je odvisen od obstoja primerkov nekaterih drugih tipov entitet – šibki tipi entitet (*odvisni od močnih*).

Ad 3 in 4) EKSISTENČNA ODVISNOST in DENTIFIKACIJSKA ODVISNOST

Govorimo o odvisnosti med tipi entitet:



Eksistenčna: oddelek je odvisen od šole - če šola ne obstaja, tudi oddelek ne more.



Identifikacijska: sam tip entitete nima nobenega takega atributa, ki bi zagotavljal enolično identifikacijo – uporabljamo primarni ključ tistega tipa entitete, na katerega je ta tip identifikacijsko navezan.

Tip entitete diagnoza je vezana na pacienta in je (*diagnoza*) identifikacijsko odvisna od tipa entitete pacient.

6 MODELIRANJE POSTOPKOV

1. Poslovni procesi in postopki
2. Identifikacija postopkov
3. Funkcijska dekompozicija
4. Sestavljeni in elementarni postopki
5. Opredelitev elementarnega postopka

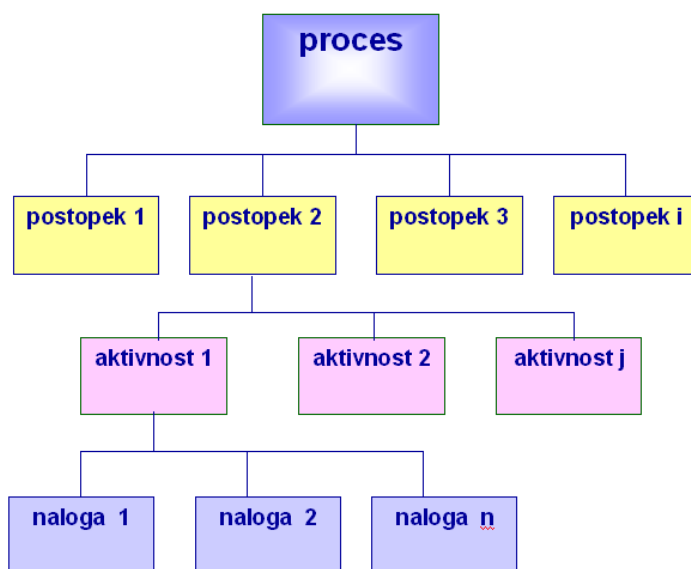
Ad 1) OPREDELITEV POSLOVNEGA PROCESA

Poslovni proces je **strukturirana množica aktivnosti**, katerih **rezultat** je nek **proizvod** ali **storitev s tržno vrednostjo**.

Zajema **vhode** in **izhode**, ki predstavljajo neko **dodano vrednost za uporabnike**. Praviloma se sestoji iz **več postopkov** in posega **na več funkcijskih področij** (T. Davenport).

Proces je precej širši pojem od postopka. V vsakem poslovnem sistemu se odvijajo neki poslovni procesi, ki se sestojijo iz večih poslovnih postopkov.

Struktura poslovnega procesa



Pridemo do štirih nivojev, odvisno od primera – pogosto se ustavimo na tretjem nivoju (v praksi največkrat dva nivoja).

Ad 2) IDENTIFIKACIJA POSTOPKOV

Pojavlja se vprašanje, kako identificirati postopke. Ponekod so postopki jasno opredeljeni (npr. na UE), ker so predpisani z zakonom. Na področju nekega ministrstva je vse bolj komplicirano in je potreben sistematičen pristop k analizi.

Ad 3) FUNKCIJSKA DEKOMPOZICIJA

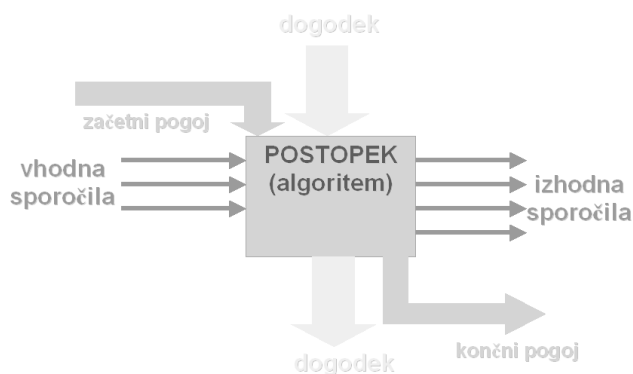
Je postopek sistematičnega razstavljanja postopka na manjše dele – področja, dokler ne pridemo do elementarnega postopka.

- Sistematična **analiza** in **razstavljanje** poslovnih procesov (*funkcij*) na njihove **elementarne sklope**,
- pristop od **zgoraj navzdol**;
- **identifikacija elementarnih postopkov**;
- **predstavitev strukture** (*strukturni graf*)

Elementarni postopek – razumemo skupino medsebojno povezanih aktivnosti, ki jih ni mogoče ali pa jih ni smiselno razstavljati naprej. Je najmanjša smiselna aktivnost, katere izvedba predstavlja zaključeno celoto in je opredeljena z **vhodnimi** in **izhodnimi podatki** ter **izvedbenimi pravili**. Elementarni postopek mora biti kar se da dobro opredeljen.

Cilj funkcijske dekompozicije je priti do elementarnih postopkov. Vmes, na poti do tega pa pridemo tudi do sestavljenih postopkov.

6.1 OPREDELITEV OSNOVNIH KONCEPTOV MODELIRANJA POSTOPKOV



1. **POSTOPEK**: predstavlja **smiselno zaključeno množico operacij** na podatkih, s katerimi se IS pripelje iz enega v drugo stanje. **Opredeljen je z algoritmom in vhodno izhodnimi podatki.**

2. **OPERACIJA**: je **poseg na primerku entitete**, s katerimi se primerki berejo, kreirajo, spreminjajo in brišejo ali na **vrednostih njegovih atributov**, s katerimi se te vrednosti

spreminjajo.

3. **DOGODEK**: je neke vrste impulz, do katerega pride bodisi zaradi sprememb ali posegov **iz okolice IS** ali zaradi **notranjega stanja IS**. Povzroči izvedbo enega ali več postopkov, s katerimi se spremeni stanje sistema.

4. **ZAČETNI POGOJ**: opredeljuje vse pogoje, ki morajo biti izpolnjeni, da je zagotovljen **pravilen potek izvajanja postopka** oziroma da se nek **postopek lahko začne izvajati**.

5. **KONČNI POGOJ**: opredeli vse pogoje, ki morajo biti **izpolnjeni po izvedbi postopka**, da se lahko šteje, da je bil postopek pravilno izvršen in normalno zaključen ter, da je zagotovljena integriteta IS.

6. **SPOROČILO**: je zaključena celota informacij, ki **vstopa** v postopek ali **izstopa** iz njega - deli se na **vhodna** (so vsi vhodni podatki) in **izhodna** (izstopajo in nastanejo kot rezultat postopka).

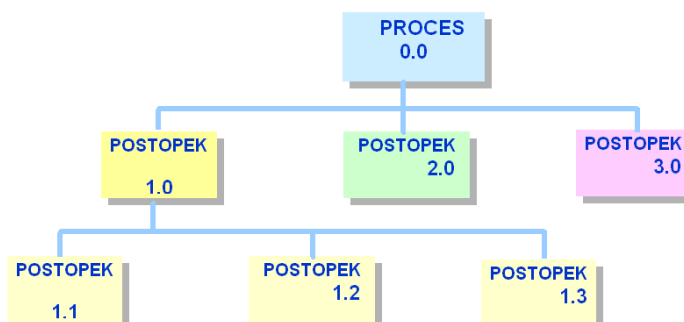
Sporočilo je **predmet obdelave** v postopku in je potrebno za njegovo izvedbo ali pa je **rezultat njegove izvedbe**.

6.2 METODE IN TEHNIKE MODELIRANJA POSTOPKOV

1. STRUKTURNI GRAF
2. DIAGRAM TOKA PODATKOV

Ad 1) STRUKTURNI GRAF

- Je **hierarhični graf**, ki predstavlja celoto in strukturo postopkov obravnavanega poslovnega procesa,
- sestavljajo ga **sestavljene in elementarne postopki**,
- **pravila** pri izdelavi grafa,
- **številčenje** postopkov.



Struktura ima vedno **obliko drevesne strukture** – na najvišjem nivoju je vedno en sam postopek, ki predstavlja celoto postopkov v celotnem grafu.

Vsak sestavljen postopek mora biti razstavljen na vsaj dva elementarna postopka (če je samo eden postopek, je elementarni). Vsi postopki morajo biti oštevilčeni – enolične oznake, kratke oznake postopkov (na vrhu 0.0, naprej pa 1.0, 2.0, ...).

Strukturalni graf je namenjen **modeliranju postopkovnega dela**, ne prikazuje pa dinamičnosti, temveč le hierarhično strukturo postopkov. Do grafa pridemo s **funkcijsko dekompozicijo**, ki nas pripelje do razmerja (med nadrejenim in podrejenimi vozlišči; zaključki vej so elementarni postopki).

Ad 2) DIAGRAM TOKA PODATKOV

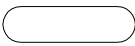
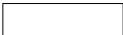


Je grafična tehnika, s katero predstavimo vse **podatkovne tokove** v okviru obravnavanega poslovnega procesa in **postopke**, ki znotraj tega procesa nastopajo.

V obliki mreže predstavimo postopke IS in pretok podatkov oziroma dokumentov med njimi.

Namenjen je **modeliranju postopkovnega dela IS**, prikazuje **dinamičnost IS** ter **zaporedje izvajanja postopka**.

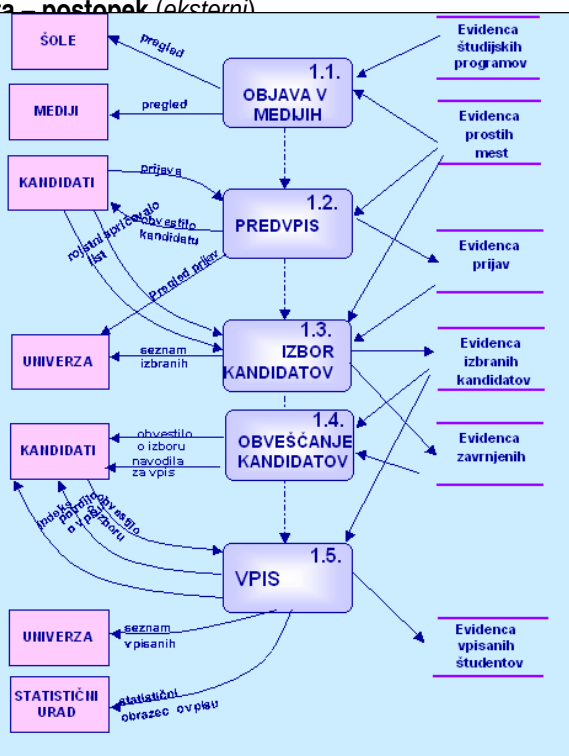
Omogoča predstavitev 4-ih **elementov sistema**:

- postopek (*del sistema, ki izvaja neko aktivnost, s katero transformira vhode in izhode*),
- zunanja entiteta,
- zbirka podatkov,
- tok podatkov.

1.	Postopek (proces, aktivnost)	Poimenovanje je odvisno od ravni obravnave, ki jo zavzamemo.
		
2.	Zunanja entiteta	Vsi tisti zunanji akterji (<i>entitete</i>), ki nastopajo v okolici obravnavanega sistema , s katerimi ta sistem ali proces kommunicira . Pomeni, da od njih (<i>fizične, pravne osebe ali objekti</i>) dobiva podatke ali pa jim podatke pošilja .
		
3.	Zbirka podatkov	S tem konceptom predstavimo (<i>opredelimo</i>) vse zbirke podatkov (<i>ne glede na njihovo vsebino, obliko, medij</i>), ki so potrebne za izvedbo obravnavanega procesa ali postopka ali, ki nastanejo kot rezultat izvedbe tega procesa ali postopka. Pri tem medij, na katerem je zbirka podatkov, ni pomemben.
		
4.	Tok podatkov	S tem ponazarjamo podatkovne oziroma informacijske tokove med procesi in njihovo okolico , kjer pa se lahko ti tokovi, podatki, odvijajo samo na dveh relacijah:
		<ul style="list-style-type: none"> - zunanja entiteta – postopek (akstorni) - postopek –

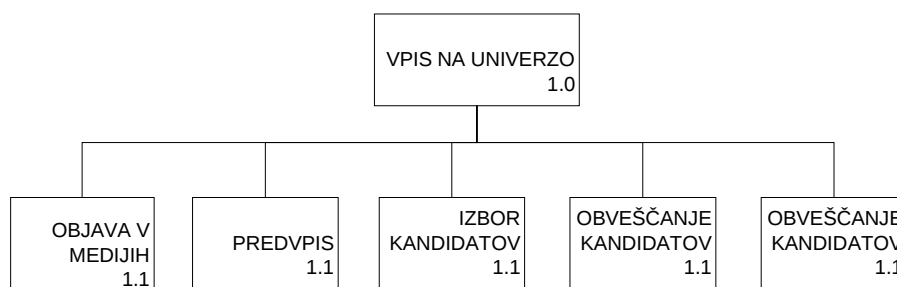
POSTOPEK / PROCES/AKTIVNOST	sprememba stalnega bivališča
ZUNANJA ENTITETA	občan
ZBIRKA PODATKOV	evidenca stalnih prebivalcev
TOK PODATKOV / DOKUMENTOV	vloga

Primer DIAGRAMA TOKA PODATKOV



Na levi strani mora biti vedno IME toka, ki pa ne sme biti glagol ampak ime podatka. Med samimi postopki ni povezave. Zunanja entiteta (*npr. univerza*) se lahko ponovi zaradi preglednosti.

Primer STRUKTURNEGA GRAFA:



7 PROBLEMI PREHODA IZ LOGIČNE NA FIZIČNO RAVEN ZASNOVE IS

OBRAVNAVANI SISTEM	MODELIRANJE PODATKOV	MODELIRANJE POSTOPKOV	REZULTATI
	Razpoložljive metode in tehnike		
MODEL OBRAVNAVANEGA SISTEMA	- tekstualni opis - E-R model	- tekstualni opis - organigram	- opis funkcije OS - organiziranost OS - opredeljene informacijske potrebe - identifikacijski postopki
LOGIČNI MODEL IS	- E-R model - podatkovni slovar (entitet, atributov, povezav)	- strukturni graf - diagram toka podatkov - prehodni diagram	Opredeljene vse lastnosti sistema na logični ravni.
FIZIČNI MODEL IS	- relacijski model - objektno orientirani model - hierarhični model - mrežni model	- podrobne specifikacije programskih modulov - strukturni diagram - diagram poteka - vhodno/izhodne maske	Opredeljene vse lastnosti sistema na fizični ravni.
IZVEDBENI MODEL IS IZDELANA REŠITEV	Generirana baza podatkov na osnovi izbranega modela	Programska koda v izbranem programskem jeziku	Izdelana in preverjena informacijska rešitev.

E-R model je uporaben na **logični ravni**, na izvedbeni ravni pa ni uporaben, zato ga je potrebno **pretvoriti v enega od treh izvedbenih modelov** (*relacijski, hierarhični, mrežni*).

Postopke, ki smo jih identificirali, grobo opisali, je potrebno zdaj **podrobno opisati, da bo možno izdelati že kar računalniške programe**. Tu si pomagamo s sodobnimi tehnikami – strukturnim diagramom, diagramom poteka.

7.1 PRETVORBA E-R MODELA V RELACIJSKI MODEL

Relacijski model je zasnovan na matematični teoriji, E-R model ni standardiziran, za seboj nima matematične teorije. Pri pretvorbi preslikamo tipe entitet (iz E-R modela) v več relacij (gre za preslikavi 1:N).

Postopek te preslikave imenujemo **NORMALIZACIJA** - ta postopek je formaliziran in ima točno določene korake:

1. V prvem koraku, ki nas pripelje do modela v prvi normalni formi, moramo odstraniti iz relacij t.i. **ponavljajoče se skupine atributov** ter z njimi tvoriti nove relacije.
2. V drugem koraku se osredotočimo na tiste relacije, katerih ključi so **sestavljene iz več atributov in iz njih izločimo tiste attribute, ki niso odvisni od celotnega sestavljenega ključa** relacije in z njim ponovno oblikujemo nove relacije in s tem pripeljemo model v drugo normalno formo.
3. V tretjem koraku odstranimo še t.i. **prehodne odvisnosti med atributi** in tudi tu po potrebi oblikujemo nove relacije in s tem pripeljemo model v tretjo normalno formo.

8 TEHNOLOŠKA, ORGANIZACIJSKA IN KADROVSKA IZHODIŠČA

8.1 UPORABA INFORMACIJSKIH ORODIJ

1. KAJ SO INFORMACIJSKA ORODJA?

Gre za vrsto različnih računalniško podprtih orodij, ki jih uporabljamo skozi različne faze razvoja informacijskih orodij. Teh orodij je danes na trgu izredno veliko in so zelo raznolika.

MED TA ORODJA SODIJO:

JEZIKI ČETRTE GENERACIJE: so programski jeziki, ki so se začeli razvijati po letu 1980. Osrednji cilj pri razvoju teh jezikov je bil kar se da **poenostaviti razvoj** programskih rešitev, **programiranje** in s tem **uporabo računalnika**.

Razvoj teh jezikov se je začel kot neke vrste odgovor na probleme, ki so jih sami s seboj prinesli JEZIKI 3. GENERACIJE - za te jezike velja, da je **delo z njimi zahtevno, potrebno je veliko računalniškega znanja, razvoj rešitev pa traja zelo dolgo** (*Cobol, Pascal, Basic*). Kljub vsemu pa še danes večina rešitev temelji na jezikih 3. generacije.

JEZIKI 4. GENERACIJE so se začeli intenzivneje uporabljati v 90-ih letih. Sodiijo med **neproceduralne jezike, skrajšajo razvojni čas, znižajo razvojne stroške, povečajo produktivnost programerjev, zahtevajo pa zmogljivejše računalnike**.

Kadar predvidevamo, da bodo razvojni stroški veliki, vzdrževalni pa nizki, bomo uporabljali jezike 4. generacije.

Če pa predvidevamo, da bodo veliki obratovalni stroški in nizki razvojni pa bomo uporabili jezike 3. generacije.

KRMILNI SISTEMI BAZ PODATKOV: pomeni prenavo starih rešitev z novo tehnologijo (*npr. napisan program s starim jezikom - III.generacije*), je pa vsebinsko še povsem spremenljiva.

Je v bistvu sistem programov, ki nam omogoča:

- **kreiranje baze podatkov** (*opredeli njeno zgradbo na osnovi modela*),
- njeno **vzdrževanje**
- njeno **uporabo** (*dostop do podatkov*).

BAZA PODATKOV – koncept organiziranja podatkov v okviru sodobnih IS, ki nam zagotavlja **splošno uporabnost podatkov**, omogoča, da iste podatke **uporablja poljubno število uporabnikov**, zagotavlja **minimalno podvajanje** ter visoko stopnjo **razvoja** ter **zaščito podatkov**.

Krmilnega sistema baze podatkov ne razvijamo sami (*redko in to le za manjše IS*) ampak ga kupimo na trgu, ker so za razvoj takega sistema potrebna izjemna specializirana znanja. V te sisteme so tudi že **vgrajena varovanje in zaščita podatkov** (*Paradox, Access na PC in Oracle, Informix na velikih sistemih in PC-jih*).

Krmilni sistem baz podatkov so ključnega pomena in danes v večini organizacij nepogrešljivo informacijsko orodje. Baza podatkov je zgrajena na nekem modelu (*hierarhični, mrežni, relacijski*).

ORACLE – največji neodvisni proizvajalec razvoja baz podatkov.

SQL – programski jezik, ratificiran kot standardni jezik za delo s krmilnimi sistemi relacijskih baz podatkov.

Standardizacija SQL, predvsem pa njegova uveljavitev v sodobnih informacijskih orodjih, mu daje možnost nadaljnje širše uporabe.

Večnivojskost SQL-a se kaže v stopnji kompleksnosti in zmožnosti nivoja uporabe jezika, je celovit in povezovalen programski jezik **uporabnika** in **razvijalca**, uporaben tako v **fazi razvoja** kot tudi v **fazi uporabe** informacijskega sistema v organizaciji.

V povezavi s krmilnim sistemom relacijskih baz podatkov je osnovno orodje za delo s tabelami.

Je visokonivojski nepostopkovni programski jezik, s katerim opredelimo KAJ naj se izvede in ne KAKO naj se izvede.

SLOVARJI PODATKOV: sestavljeni so iz treh podslovarjev (*entitet, atributov in povezav*). Izdelava in vzdrževanje je zamudno in občutljivo delo, saj v slovarju prihaja nenehno do sprememb. **Slovarji podatkov so posebna informacijska orodja, ki nam omogočajo vzpostavitev slovarja, njegovo vzdrževanje in uporabo.**

Obstajajo specialna informacijska orodja teh treh slovarjev, vendar pa se orodja za vzdrževanje podatkov slovarja redko uporabljajo kot samostojna orodja, saj se izkaže, da je vzdrževanje takega slovarja ločeno od drugih aktivnosti. Danes se podatkovni slovarji običajno pripravljajo znotraj CASE orodij.

CASE ORODJA: (*CASE computer aided system engineering*) - gre za sistem orodij za razvoj IS. Namenjena so podpori razvoja IS skozi **vse razvojne faze** takega projekta.

Ker gre za kompleksen proces z velikim številom faz, potrebujemo orodja z različnimi vrstami funkcij.

Razvoj: v začetku 80-ih let. Nekatera orodja so boljša v začetnem delu, druga v srednjem, tretja pa v zadnjem. Ni orodja, ki bi zadovoljivo pokrival vse razvojne faze. Proizvajalci so se različno osredotočili na posamezne razvojne faze.

Vrste CASE orodij:

1. Orodja za vzdrževanje dokumentacije – če želimo imeti dobro dokumentiran sistem, potrebujemo tudi do 50% delovnega časa. Vzdrževanje dokumentacije je eno od najbolj perečih področij na razvoju informacijskih rešitev, saj so le redke rešitve dobro dokumentirane. V tem smislu, da vsebujejo **celovito uporabniško informacijo** in tisto dokumentacijo, ki je potrebna za **enostavno vzdrževanje** te rešitve. Do dobre rešitve je možno priti samo tako, da nam ta dokumentacija nastane kot stranski produkt.
2. Orodja za reinženiring (prenovo) – ideja za ta orodja se je porodila v začetku 90-ih let, ko so se v večjih ustanovah zavedli, da imajo še zelo stare rešitve. Ta orodja so že zelo dolgo na trgu. Njihov zlato obdobje se je končalo z letom 2000. Vse starejše rešitve je bilo potrebno pregledati in ugotoviti ali so še primerna in jih prenoviti v skladu z današnjo tehnologijo.
3. Orodja za podporo celotnemu razvojnemu ciklu – predstavljajo osrednji del – vsebuje vse **grafične tehnike** in omogočajo **izdelavo različnih modelov** skozi vse razvojne faze.
4. Orodja za podporo vodenju projekta – so kompleksni zaradi narave dela, vplivajo na uspeh ali neuspeh.
5. Orodja za izboljšanje kakovosti – izboljšajo **spremljanje vseh aktivnosti**, ki se odvijajo na projektu in **poročanje o projektu** skozi posamezne razvojne faze.

8.2 ORGANIZACIJSKI, TEHNOLOŠKI IN KADROVSKI VIDIKI RAZVOJA INFORMACIJSKE INFRASTRUKTURE

Potrebno je zagotavljati dolgoročno načrtovanje razvoja informacijske infrastrukture:

1. integracija tehnologij,
2. spremljanje tehnoloških razvojnih trendov,
3. naložbeni vidik,
4. organizacijski vidik, kadrovski vidik.

Gre za sistematično načrtovanje razvoja informacijske infrastrukture. Nobeno podjetje si ne more več privoščiti improvizacije – potrebno je pristopiti sistematično in dolgoročno – za več let naprej, v katero smer bomo razvijali našo informacijsko infrastrukturo. Ker se informacijska tehnologija naglo razvija, se je težko odločiti, v katero smer razvijati, katero tehnologijo. Soočimo se z vrsto vprašanj, ki jih je potrebno upoštevati.

1. Prvo vprašanje: **INTEGRACIJA TEHNOLOGIJ** – v številnih organizacijah nastopa množica različnih rešitev, ki med seboj niso povezane. Realna situacija je, da imamo veliko nepovezanih rešitev, vendar nadaljnji razvojni trendi narekujejo integracijo tehnologij. Ta integracija se dogaja na celih vrstah orodij – računalniško povezani, priključeni na internet. To je tudi najbolj pereče vprašanje v vseh večjih organizacijah.
2. Drugo vprašanje: je spremljanje tehnoloških **RAZVOJNIH TRENDOV** – razvojne smeri, katere tehnologije so med seboj kompatibilne (*združljive*). To je zelo zahtevna naloga – potrebno je imeti strokovnjake, ki dobro spremljajo tehnološki razvoj.
3. Tretje vprašanje: je **NALOŽBENI VIDIK** – investicije v te naložbe so ogromne in izjemno pomembne. Ni vseeno, koliko časa bo uporabna ta oprema. Te naložbe ponekod predstavljajo polovico vseh naložb.
4. Četrto vprašanje: na vse to se navezuje tako **ORGANIZACIJSKI** kot tudi **KADROVSKI VIDIK**. Moramo biti tudi ustrezno organizirani in razvijati kader, da bodo vsa ta znanja pokrita.

8.2.1 Tehnološki vidiki

1. Decentralizacija procesne moči
2. Cenejši in zmogljivejši računalniki
3. Omreženje računalnikov
4. Integriranje lokalnih in oddaljenih omrežij
5. Internet (intranet)
6. Omrežno računalništvo

Ad 1) DECENTRALIZACIJA PROCESNE MOČI

V organizaciji so bili nekaj let nazaj (*10-20 let*) centralni računalniki. Sredi 80-ih let so se pojavili osebni računalniki. V začetku 90-ih je nastala decentralizacija procesne moči. S seboj je prinesla vrsto pozitivnih in negativnih učinkov. Na ta način je uporaba informacijske tehnologije postala prilagojena posameznim delovnim mestom in s tem pa zmanjšamo investicijske stroške.

Slaba stran decentralizacije pa je varovanje in zaščita podatkov (*nemogoče je zagotoviti 100% zaščito*), koordiniranje in standardizacija, računalnika ne izkoristimo toliko kot nam nudi, ...

Ad 2) CENEJŠI IN ZMOGLJIVEJŠI RAČUNALNIKI

Drugi vidik, ki je očiten, je dejstvo, da se računalniki cenijo - za isti denar imamo vsako leto več moči.

Ad 3) OMREŽENJE RAČUNALNIKOV

Tretji vidik – računalniki so vse bolj povezani med seboj. Razdalja med računalniki ni več pomembna, saj nam je dostopen katerikoli podatek.

Ad 4) INTEGRIRANJE LOKALNIH IN ODDALJENIH OMREŽIJ

Popolna integracija lokalnih in globalnih mrež je posledica omreženja računalnikov.

Ad 5) INTERNET, INTRANET

Vsakdo, ki ima osebni računalnik povezan v mrežo, ima dostop do **interneta**, ki je glavni komunikacijski medij preko svojih servisov od elektronske pošte do svetovnega spleta.

Prav tako poznamo tudi **intranet** – interni internet, znotraj poslovnega sistema.

Ad 6) OMREŽNO RAČUNALNIŠTVO

Omrežno računalništvo – naredimo korak nazaj k terminalu (*zmogljivejše računalnike zamenjamo z omrežnimi*). Centraliziramo tiste funkcije, ki jih na posameznih delovnih mestih ne potrebujemo in ta način lahko razbremenimo delovno postajo nepotrebne navlake ter tako postanejo računalniki zmogljivejši in cenejši.

8.2.2 Organizacijski vidiki

Spremenjena vloga informacijske poslovne funkcije:

- od **avtomatizacije** k **informatiki**,
- od **AOP centra** (*avtomatska obdelava podatkov*) k **štabni službi za informatiko**.

Informatizacije je celovit proces prenove poslovanja organizacije kot celote - to pomeni, da je tudi poslovna funkcija dobila drugo vlogo.

Štabna služba za informatiko ima v bistvu drugačne **naloge** in **zadolžitve**:

- **strateško načrtovanje** in **razvoj** celotne infrastrukture,
- vsi potrebujemo **izobraževanje**, ker gre razvoj hitro naprej, računalniki so zmogljivejši, ne znamo uporabiti vseh funkcij,
- uporabnikov je toliko kot je delovnih mest – nuditi jim je potrebno **strokovno pomoč, podporo**,
- **varovanje** in **zaščita podatkov** – da se podatki ne izgubijo, uničijo, poškodujejo. Zaščita se izvrši predvsem tehnično,
- **standardi** so **tehnološki, tehnični** in standardi **kakovosti**.

8.2.3 Kadrovski vidiki

Pridobivanje ključnih **informacijskih znanj**:

- strateško načrtovanje in razvoj informacijske infrastrukture (*tu gre za vprašanje ali izobraževati lastne kadre ali najemati strokovnjake od zunaj. Najprej je potrebno imeti ključna znanja in potem ugotavljati, katera znanja kam*),
- razvoj IS (*ena od permanentnih nalog – nenehno posodabljati*),
- uporaba informacijskih rešitev in informacijskih orodij (*uporaba že izdelanih rešitev in uporaba orodij*),
- vzdrževanje strojne, programske opreme in telekomunikacij (*vzdrževanje strojne in programske opreme je vse večje in potrebno, isto velja za telekomunikacije - nekdo mora skrbeti za vse to in imeti potrebna znanja*).

To so vsa znanja, ki jih je potrebno v neki organizaciji zagotoviti, tu pa se postavi vprašanje, kaj postoriti z lastnimi strokovnjaki in kaj s tujimi, zunanji.

9 INFORMATIKA IN STANDARDI

V tem poglavju bomo govorili izključno o standardih kakovosti na IS.

- Pomen kakovosti na področju IS
- Razvoj standardov kakovosti (ISO-9000)
- **ISO 9000-3**
 - o Značilnosti standarda
 - o Prednosti in slabosti
- **DIN 66285**
 - o Značilnosti standarda
 - o Prednosti in slabosti

Razvoj standardov kakovosti se je začel v 80-ih, ko so na nekaterih področjih ugotovili, da bi bilo dobro uvesti standarde in stvari poenotiti.

ISO – mednarodna organizacija za standardizacijo s sedežem v Ženevi, je v začetku 80-ih vpeljala serijo novih standardov, ki so dobili oznako ISO 9000. V okviru ISO 9000 se je kasneje razvila množica standardov za različna področja, tudi za storitve (*ISO 9001, 9002, 9003, ...*).

Ti splošni standardi niso najbolj prikladni za ugotavljanje kakovosti IS, zato so se začeli razvijati posebni standardi **ISO 9000-3** (*standard organizacije ISO*) in **DIN 66285** (*nemški standard*).

Po namenu, cilju in uporabnosti sta si standarda povsem različna.

9.1 ISO 9000-3 (Procesni standard)

Je nastal na podlagi tega, da je ugotavljanje kakovosti informacijskih rešitev in IS izredno kompleksna, zahtevna naloga in da je zato ugotavljanje kakovosti neke rešitve potem, ko je že narejena, izjemno zahtevno opravilo.

Zato se ISO 9000-3 usmerja na uvajanje in izgradnjo celovitega sistema **kakovosti skozi celotni razvojni cikel neke informacijske rešitve oziroma IS, ne ukvarja pa se z ugotavljanjem kakovosti končnega izdelka**. ISO 9000-3 se vpeljuje sistem kakovosti skozi vse razvojne faze, da bi zadostili standardom in dobili končen proizvod.

Kakovost izdelka ugotavljamo tako, da skrbno spremljamo **vse razvojne faze** (*ki so predhodno podrobno opredeljene*) in če so bile vse razvojne faze kakovostno izdelane, bo kakovosten tudi končen izdelek.

ISO 9000-3 polaga **odgovornost** za kakovost na obe strani, izvajalcu in naročniku (*to je eno od glavnih sporočil tega standarda*) - tudi naročnik se mora odgovorno obnašati, če hoče na koncu kakovosten izdelek. ISO 9000-3 je osredotočen na izgradnjo in izvedbo celovitega sistema kakovosti.

9.2 DIN 66285

Nemški standard izhaja **iz končnega proizvoda**, za katerega opredeljuje **elemente kvalitete** in **načine** njihovega **ugotavljanja**.

DIN 66285 navaja naslednje **elemente kvalitete** programskega proizvoda:

- k vsaki programski rešitvi sodi tudi **funkcionalen zapis** in **dokumentacija**,
- funkcionalni opis proizvoda mora **vsebovati vse njegove funkcije** in te mora proizvod tudi **izpolnjevati**,
- proizvod mora biti opremljen z **dokumentacijo**, ki mora vsebovati vse o **uporabi, namestitvi** in **vzdrževanju** proizvoda,
- **specifikacija vseh programov**, ki nek proizvod sestavljajo.

Ta standard se dejansko v Nemčiji že uporablja.

10 OSNOVNE ZNAČILNOSTI SODOBNE METODOLOGIJE

1. Zajemati mora **celoten življenjski cikel** IS in pripadajoče programske opreme,
2. omogočati mora **systematičen prehod** iz ene v naslednjo fazo,
3. omogočati mora **verifikacijo pravilnosti razvojnega procesa** skozi vse faze razvojnega cikla IS,
4. omogočiti mora **timsko delo** na projektu ter uporabo sodobnih metod organizacije in vodenja projektov,
5. biti mora uporabna za čim širši spekter računalniških projektov,
6. biti mora dovolj enostavna za **priučitev**. Sposobni naj bi jo bili razumeti povprečno sposobni ljudje,
7. omogoča naj uporabo **čim več avtomatiziranih orodij** za povečanje produktivnosti posameznikov in celotnega tima,
8. omogočiti mora **dokumentiranje spremljanja** razvoja IS skozi vso njegovo življenjsko dobo.

10.1 GENERACISKI RAZVOJ METODOLOGIJ NAČRTOVANJA IN GRADNJE IS

10.1.1 1. GENERACIJA

- **dokumentacijske tehnike:**
 - o tekstualni opis,
 - o diagram poteka,
 - o odločitvene tabele,
 - o diagrami toka podatkov.

10.1.2 2. GENERACIJA

- metodologije, utemeljene na **strukturnih tehnikah:**
 - o strukturna analiza in dizajn,
 - o *Jaksonova* programska metoda,
 - o razvoj sistemov po *Jaksonu*,
 - o razvoj IS po metodi ISAC.

10.1.3 3. GENERACIJA

- metodologije **višje stopnje formalizacije**:
 - sistemi višje stopnje,
 - informacijsko inženirstvo,
 - modeliranje aktivnih in pasivnih komponent sistema.

10.1.4 4. GENERACIJA

- **objektno orientirane** metodologije:
 - v razvoju.

11 PRILOGE

Možna izpitna vprašanja

1. Osnovne značilnosti sodobne metodologije!
2. Koncepti razvoja modeliranja podatkov!
3. Ključni problemi razvoja programskih rešitev!
4. Generacijski razvoj metodologije načrtovanja gradnje IS?
5. Razlike med informatizacijo in avtomatizacijo?
6. Kaj zajema življenjski cikel IS?
7. Primerjava med pristopi (načrtovanja gradnje IS)?
8. Značilnosti (prednosti) prototipnega pristopa?
9. Značilnosti objektnega in linearnega pristopa?
10. Razlika med logičnim in fizičnim nivojem modeliranja sistemov in razlogi za to razliko?
11. Kaj je modeliranje?
12. Kaj je povezava in kaj končni pogoj?
13. Kaj je entiteta?
14. Kaj je atribut?
15. Koncepti abstrakcije pri modeliranju podatkov?
16. Opiši koncepte abstrakcije?
17. Razlike med podatkovnim in postopkovnim modelom?
18. Opiši klasične podatkovne modele!
19. Hierarhični model!
20. Opiši hierarhični model in navedi slabosti?
21. Razlika med mrežnim in relacijskim modelom!
22. Prednosti relacijskega modela v primerjavi z linearnim in mrežnim!
23. Mrežni model!
24. Prednosti in slabosti relacijskega modela podatkov?
25. Opiši temeljne koncepte E-R modela!
26. Opiši zasnovo E-R modela!
27. Slabosti E-R modela!
28. Značilnosti E-R modela!
29. Slovar podatkov!
30. Temeljne tehnike modeliranja postopkovnega dela IS!
31. Koncepti modeliranja postopkovnega dela !
32. Koncepti modeliranja postopkov!
33. Koncepti abstrakcije pri modeliranju postopkov!
34. Predstavi osnovne koncepte modeliranja postopkov!
35. Postopek normalizacije!
36. Normalizacija!
37. Kaj je baza podatkov in na kakšen način oblikujemo podatke?
38. Krmilni sistemi podatkovnih baz!
39. Opiši CASE orodja!
40. Značilnosti CASE orodij!
41. CASE orodja!
42. Razlike med ISO9000-3 in DIN 66285?
43. ISO 9000-3!
44. ISO standardi!
45. Kaj so standardi?
46. Jeziki 4.generacije!
47. Povezave v E-R modelu?

48. Kaj je entiteta in kaj zunanja entiteta? Ali je možno, da zunanja entiteta nastopa kot entiteta?
49. Opredelitev IS!
50. Klasifikacija in generalizacija!
51. Opredeli poslovni proces in metodo funkcijske dekompozicije!
52. Povezave!