

# MODELIRANJE DELOVNIH PROCESOV PRI UVAJANJU SISTEMOV ZA UPRAVLJANJE DELOVNIH PROCESOV

Anamarija Leben

Univerza v Ljubljani, Visoka upravna šola, Kardeljeva pl. 5, 1000 Ljubljana  
E-pošta: Anamarija.Leben@uni-lj.si

## Povzetek

Modeliranje delovnih procesov je izredno pomembna faza v celotnem ciklu upravljanja delovnih procesov. Z razvojem sistemov za upravljanje delovnih procesov pa ta faza še pridobiva na pomenu. Članek tako podaja pregled nad različnimi metodami in tehnikami modeliranja procesov - od klasičnih do tistih, ki že upoštevajo modeliranje tudi tistih konceptov, ki so pomembni za samo upravljanje delovnih procesov.

## Abstract

*Business process modeling represents a very important planning phase in the entire workflow management cycle. Due to the development of workflow management systems, that phase has been becoming even more important. In the article, a review of different approaches in proces modelling are presented - from the classical ones to those, relevant for workflow management.*



## 1. Uvod

Upravljanje delovnih procesov lahko opredelimo kot ciklično izvajanje faz načrtovanja, izvedbe in analize delovnega procesa [7]. Faza načrtovanja, ki zajema tudi modeliranje procesov, je prav gotovo najbolj kritična od vseh, saj sta od kvalitete modelov v veliki meri odvisna uspeh oziroma neuspeh novega sistema ter njegova učinkovitost z ozirom na čas in stroške. Še posebej pa moramo biti pri modeliranju procesov pozorni v primeru, ko je končni cilj uvedba sistema za upravljanje delovnih procesov - SUDP (*Workflow Management System - WFMS*).

Modeliranje procesov zajema modeliranje dinamičnih lastnosti informacijskega sistema in organizacije same. Z njihovim modeliranjem posredno opredeljujemo tudi poslovna pravila oziroma pravila obnašanja obravnavanega sistema [5]. Modeliranje procesov je kot del načrtovanja in izgradnje informacijskih sistemov seveda odvisno od pristopa k informatizaciji [2,4], ki je lahko:

- *tehnološki*, kjer je glavni poudarek informatizacije na podatkovni in informacijski ravni - možnost za povečanje učinkovitosti obstoječih procesov se kaže v avtomatizaciji izvajanja delovnih opravil in zbiran-

ju ter oblikovanju podatkov v ustrezne informacije s pomočjo informacijske tehnologije, ne da bi pri tem karkoli razmišljali o preoblikovanju strukture procesa samega;

- *strateški*, kjer je učinkovita uporaba informacijske tehnologije pomemben faktor pri doseganju strateške in primerjalne prednosti podjetja s pomočjo preoblikovanja ali celo prenove poslovnih procesov, kar je mnogokrat povezano tudi z uvedbo sistemov za upravljanje delovnih procesov.

## 2. Modeliranje poslovnih procesov

V nadaljevanju bomo prikazali različne metode in tehnike modeliranja procesov, s poudarkom na metodah, ki so usmerjene v upravljanje delovnih procesov.

### 2.1 Klasične metode in tehnike modeliranja procesov

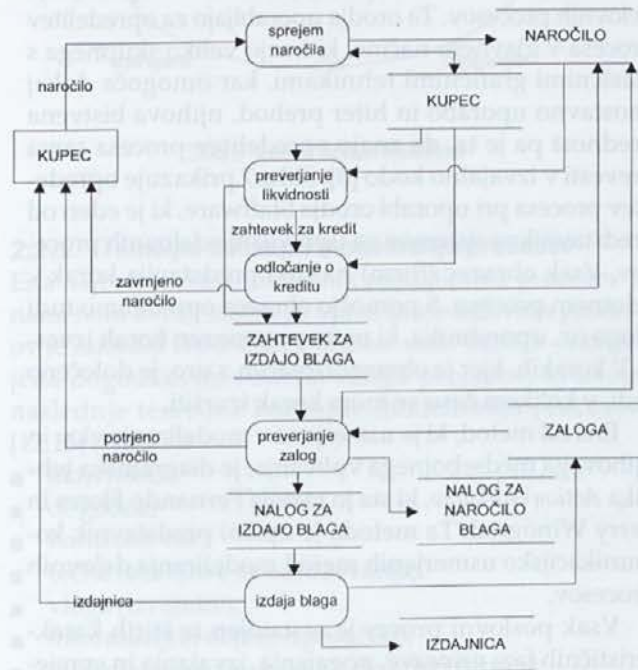
Klasične metode in tehnike modeliranja procesov so se v veliki meri nanašale na ustrezen podatkovni model, kar je v okviru tehnološkega pristopa popolnoma zadostovalo. Značilni predstavnik teh tehnik modeliranja je *diagram tokov podatkov - DTP (data flow diagram -*

Slika 1), kjer si funkcije oz. aktivnosti sledijo zaporedoma in so medsebojno povezane preko zbirke podatkov, ki jih funkcije polnijo ali pa iz njih črpajo podatke. Glavni namen takega modeliranja je prikazovanje vseh podatkovnih vhodov, ki so potrebni za izvedbo obravnavanega procesa, in vseh podatkovnih izhodov, ki nastanejo kot rezultat izvedbe procesa [5,16]. Vsa logika izvajanja procesov (pogojna razvejanja in združevanja, rekurzivne zanke) se obravnava izključno znotraj funkcije in se v tehniki *diagrama poteka* (*flowchart* - Slika 2) prikazuje ločeno kot notranji algoritem funkcije [3], za katerega pa lahko uporabimo tudi *odločitvene tabele* (*decision tables* - Slika 3) [13,16]. Iz povedanega je razvidna podatkovna usmerjenost tudi pri modeliranju procesov.

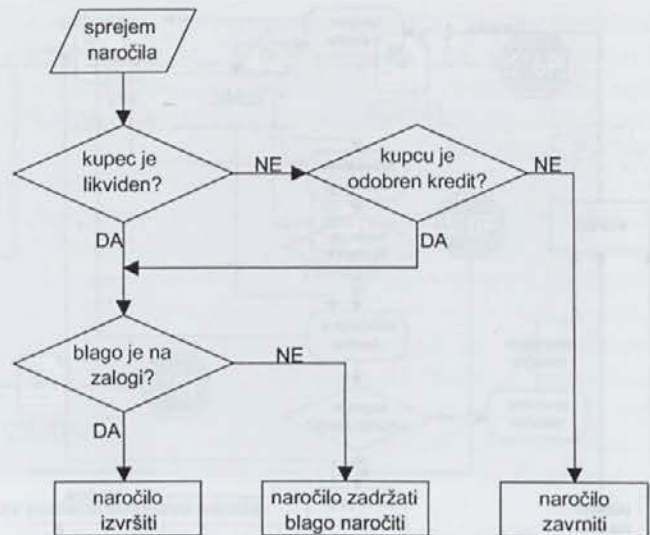
Opisane metodologije nič ne govorijo o dogodkih, vlogah, stanjih in spremembah stanj različnih objektov (npr. dokumentov) [9], kar pa so poleg logike izvajanja procesov bistveni elementi pri uvajanju sistema za upravljanje delovnih procesov.

## 2.2 K računalniško podprtem upravljanju delovnih procesov usmerjene metode in tehnike

Ena od možnosti, kako rešiti nakazane probleme, je razširitev standardnih tehnik z novimi gradniki, ki dodatno opredeljujejo prej opisane pojme. Pogosto se uporablja razširitev diagrama podatkovnih tokov, pri



Slika 1: Diagram tokov podatkov - obdelava naročil



Slika 2: Diagram poteka - obdelava naročil (povzeto po [13] - str. 101)

seznam pogojev	P1	P2	P3	P4	P5	P6	P7
kupec je likviden	DA	NE	DA	NE	NE	—	—
kupec ima odobren kredit	—	DA	—	DA	NE	DA	DA
blago je na zalogi	DA	DA	NE	NE	—	DA	NE

### možne akcije

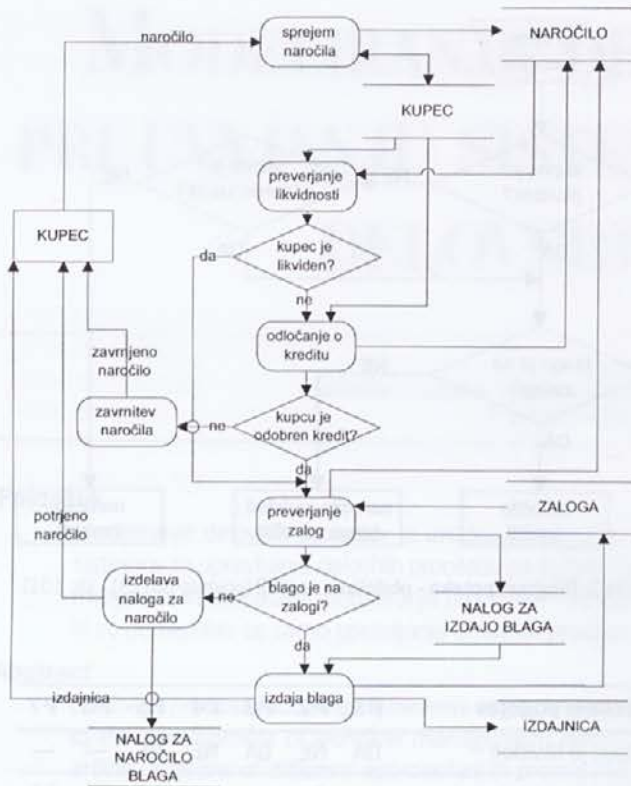
naročilo se lahko izvrši	x	x					x
naročilo se zavrne						x	
naročilo se zadrži, blago naroči			x	x			x

Slika 3: Odločitvena tabela - obdelava naročil (povzeto po [13] - str. 109)

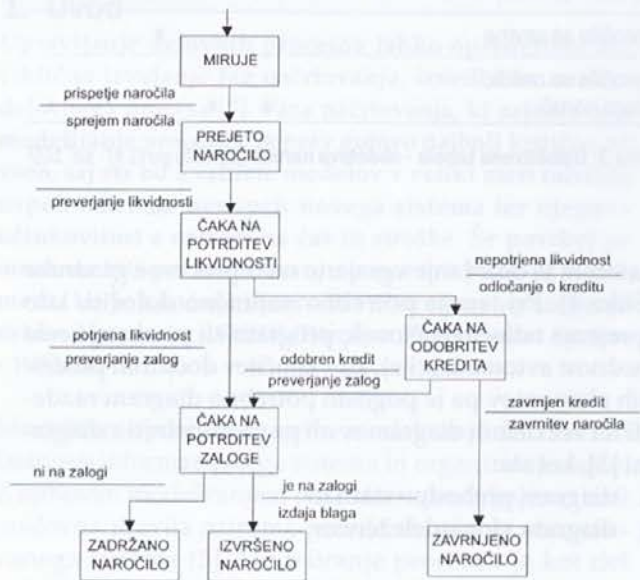
katerem je odločanje vgrajeno med procesne gradnike (Slika 4). Pri tem je potrebno natančno določiti, kdo sprejema odločitve (človek, program ali pa obstaja celo možnost avtomatizacije). Za vključitev dodatnih procesnih elementov pa je pogosto potrebno diagram razdeliti na več delnih diagramov ali pa ga dopolniti z diagrami [3], kot sta:

- diagram prehodov stanj in
- diagram vlog udeležencev.

Slika 5 prikazuje *diagram prehajanja stanj* (*State Transition Diagram* - STD) [16], pri katerem so različna stanja sistema prikazana s pravokotniki, prehodi med njimi pa s puščicami. Diagram prikazuje tudi pogoje za prehode med stanji in aktivnosti, ki se ob tem izvedejo. Pogoji napišemo nad črto poleg puščice, ki označuje prehod, aktivnost pa pod njo.

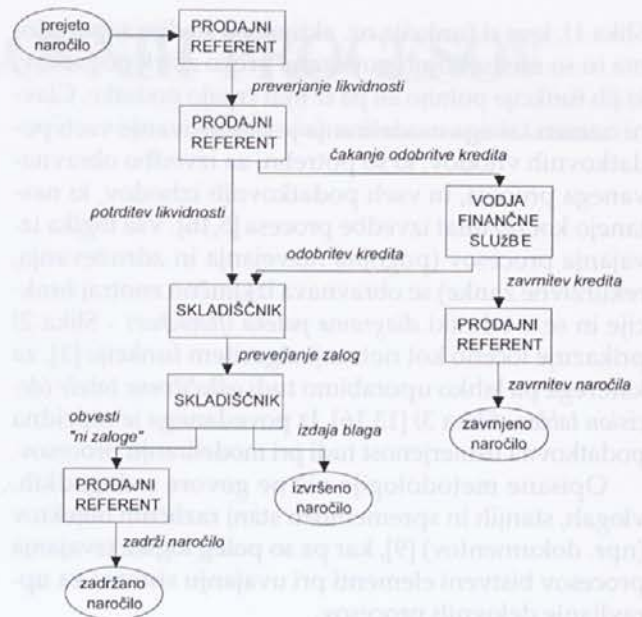


Slika 4: Kombinacija diagrama podatkovnih tokov in diagrama poteka - obdelava naročil



Slika 5: Diagram prehajanja stanj - obdelava naročil

V diagramu vlog udeležencev v procesu (Slika 6) prikazujemo s pravokotniki vloge, z elipsami stanja, s puščicami pa aktivnosti.



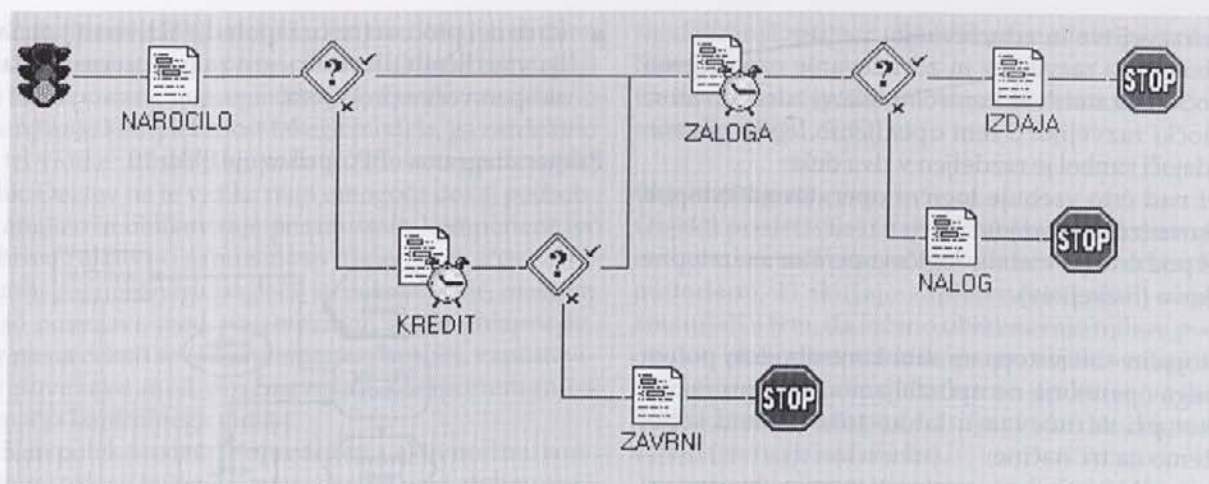
Slika 6: Diagram vlog udeležencev - obdelava naročil

Positivne strani teh tehnik so preglednost, jasnost in enostavnost, na tržišču pa obstaja cela paleta orodij, ki omogočajo tako načrtovanje, vendar je bistvena slabost teh tehnik ta, da orodja omogočajo le omejeno shranjevanje informacij v slovar podatkov (ali pa še to ne), kar seveda pomeni, da je avtomatski prenos rezultatov načrtovanja v izvajalno okolje izredno težaven, če ne že nemogoč.

Omenjene slabosti odpravljajo sistemi za upravljanje delovnih procesov. Ta orodja uporabljajo za opredelitev procesa v glavnem načine, ki imajo veliko skupnega s klasičnimi grafičnimi tehnikami, kar omogoča dokaj enostavno uporabo in hiter prehod, njihova bistvena prednost pa je ta, da znajo opredelitev procesa sama prevesti v izvajalno kodo [3]. Slika 7 prikazuje opredelitev procesa pri uporabi orodja Staffware, ki je eden od predstavnikov sistemov za upravljanje delovnih procesov. Vsak obrazec (form) na sliki predstavlja korak v celotnem procesu. S pomočjo obrazca opredelimo tudi vlogo oz. uporabnika, ki mora posamezen korak izvesti. V korakih, kjer je obrazec označen z uro, je določeno tudi, v kolikem času se mora korak izvršiti.

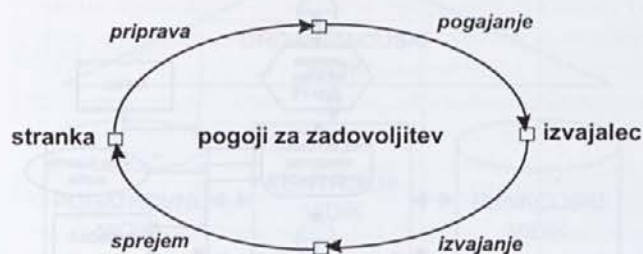
Ena od metod, ki je usmerjena v modeliranje vlog in njihovega medsebojnega vplivanja, je diagramska tehnika *ActionWorkflow*, ki sta jo razvila Fernando Flores in Terry Winograd. Ta metoda je tipični predstavnik komunikacijsko usmerjenih metod modeliranja delovnih procesov.

Vsak poslovni proces je sestavljen iz štirih karakterističnih faz: priprave, pogajanja, izvajanja in sprejema, ki povezujejo aktivnosti stranke in izvajalca v delovnem procesu. Omenjene faze so v tej tehniki prikazane v obliki eliptične zanke, ki jo imenujemo kar

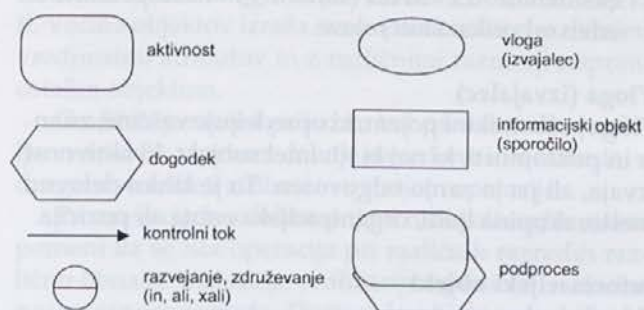


Slika 7: Staffware - opredelitev procesa obdelava naročil

zanka ActionWorkflow (Slika 8). Vloga stranke je prikazana na levi strani, vloga izvajalca pa na desni strani zanke. Vsaka od teh faz je lahko predstavljena kot poseben proces (zanka ActionWorkflow), ki je z glavnim procesom povezan s puščico [9].



Slika 8: Zanka ActionWorkflow



Slika 9: Osnovni koncepti modeliranja procesov

### 2.2.1 Temeljni koncepti modeliranja procesov

Ena najbolj razširjenih tehnik modeliranja procesov z namenom uvesti sistem za upravljanje delovnih procesov je *extended event-driven process chain (eEPC)* – razširjena dogodkovno vodena veriga procesov, ki uvaja naslednje temeljne koncepte modeliranja procesov [12,15] (Slika 9):

- aktivnost,
- dogodek,
- kontrolni tok,
- točke razvejitve in združevanja,
- vloga (izvajalec),
- informacijski objekt (sporočilo),
- podproces oz. povezava z drugimi procesi.

#### Aktivnost

Aktivnost je zaporedje korakov oziroma operacij, ki so potrebne, da opravimo neko delo. Ima časovno dimenzijo in predstavlja *aktivno komponento procesa*.

#### Dogodek

Dogodek je impulz, ki sproži izvajanje neke aktivnosti, ali pa je rezultat aktivnosti. Definiramo ga lahko kot pojav nekega objekta ali pa kot spremembo stanja objekta in je del *pasivne komponente procesa* brez časovne dimenzije.

Po nastanku je dogodek lahko:

- zunanji (npr. prispetje naročila),
- notranji (sprejetje odločitve zaposlenega - npr. zavrnitev naročila),
- časovni (ko se aktivnost sproži ob vnaprej določenem času - npr. pretečen rok za plačilo predračuna) [5,14].

#### Kontrolni tok

Nakazuje *potek procesa* od aktivnosti do aktivnosti.

Te osnovne koncepte bomo v nadaljevanju skušali podrobneje opredeliti.

### Točke razvejitev in združevanja

S simbolom za razvejitev in združevanje označujemo tiste točke, v katerih se kontrolni tokovi združijo ali se v tej točki razvejijo. S tem opisujemo *logiko procesa*. Pripadajoči simbol je razdeljen v dva dela:

- del nad črto vsebuje logični operator za vstopne tokove (združevanje),
- del pod črto pa vsebuje logični operator za izstopne tokove (razvejitev).

Če vstopa in - ali izstopa en sam kontrolni tok, potem logičnega operatorja ne uporabljamo. Tako pri razvejitvi kot pri združevanju lahko tokove med seboj povežemo na tri načine:

- logični IN (simbol  $\wedge$ ) - nastopiti morajo obvezno vsi tokovi,
- logični ALI (simbol  $\vee$ ) - nastopijo lahko posamezni prikazani tokovi ali njihova poljubna kombinacija,
- ekskluzivni ALI - XALI (simbol  $\underline{\vee}$ ) - nastopi lahko le eden od prikazanih tokov.

### Vloga (izvajalec)

Vloga je abstraktni pojem, ki opredeljuje veščine, znanja in pristojnosti, ki naj bi jih imel subjekt, ki aktivnost izvaja, ali pa je zanjo odgovoren. To je lahko delovno mesto, skupina ljudi, organizacijska enota ali pozicija.

### Informacijski objekt

Združuje dva pojma:

- sporočilo, ki lahko označuje posamezni dokument, skupek dokumentov oziroma kakršnokoli sporočilo; lahko je v elektronski ali papirni obliki, lahko je ustno sporočilo, ni pa nujno, da se o njem vodijo kakršnokoli evidence;
- zbirka podatkov, ki označuje kakršnokoli evidenco oz. skladišče podatkov ne glede na medij, na katerem so ti podatki shranjeni - to je lahko računalniška evidenca, ročna kartoteka, zvezek ali knjiga ipd.

Informacijski objekt lahko nastopa kot vhod v aktivnost ali kot izhod iz aktivnosti in podobno kot dogodek predstavlja *pasivno komponento procesa*, skupaj z vlogo pa podrobneje opredeljuje aktivnost.

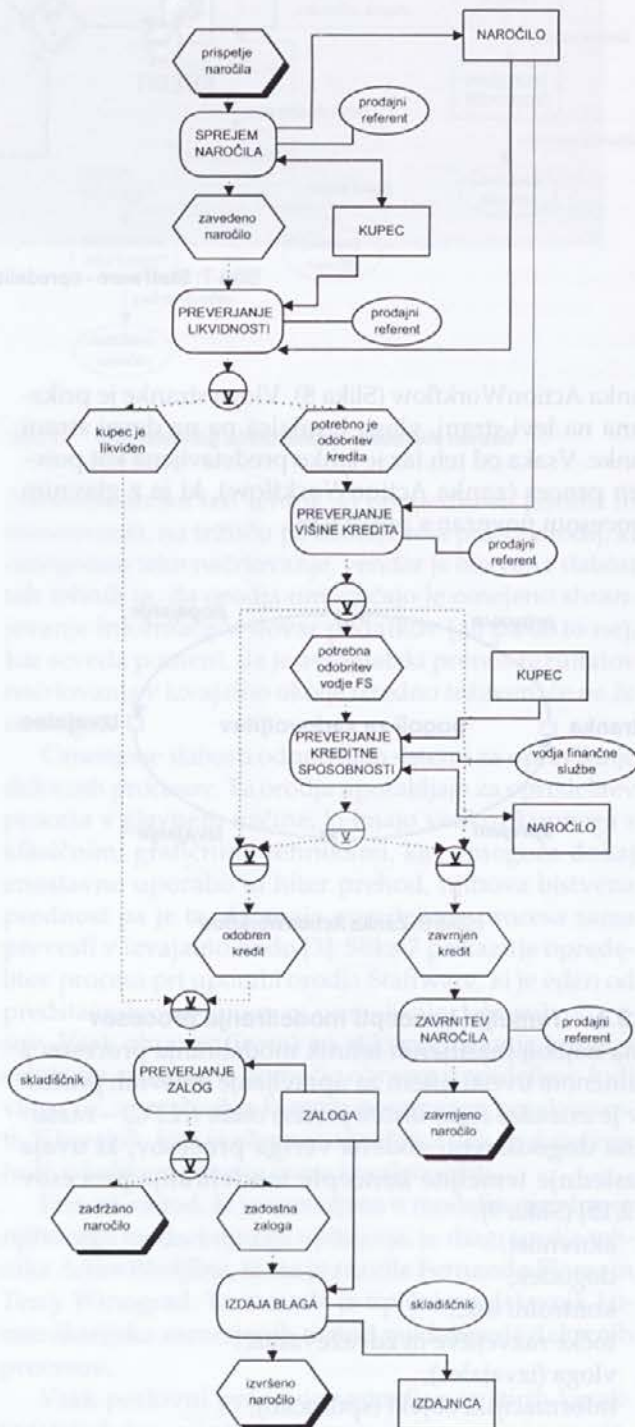
### Podproces

Ko z diagramom prikazujemo nek proces, se pogosto odločamo, da določeno zaporedje aktivnosti prikazemo kot samostojen proces. To se uporablja predvsem v dveh primerih:

- ko dela procesa zaradi preglednosti diagrama ne moremo opisati znotraj trenutnega diagrama - zaključeno zaporedje aktivnosti je sicer del nekega procesa, vendar pa bi prikazovanje znotraj tega procesa zameglilo njegov osnovni potek - gre v bistvu za podproces posameznega procesa;

- ko se del procesa (neko zaporedje aktivnosti) ponavlja v različnih procesih - gre za podproces, ki pa je skupen večim procesom.

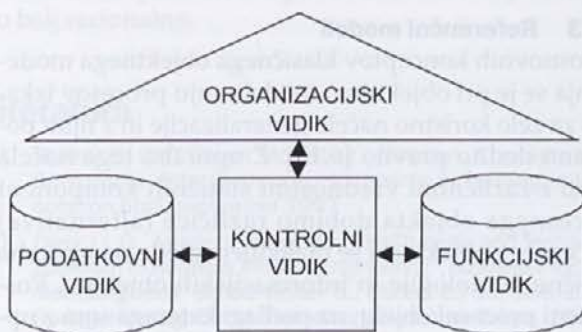
Primer diagrama eEPC prikazuje Slika 10.



Slika 10: Diagram eEPC - obdelava naročil (osenčeni so začetni in končni dogodki)

Z diagrami eEPC lahko izdelamo dokaj podroben model procesa, ki pa pri zapletenejših procesih lahko kaj hitro postane nepregleden [12]. Da vsaj navidezno zmanjšamo kompleksnost takega modela, ga razdelimo na tri vidike: funkcijski, podatkovni in organizacijski vidik. Delitev na te vidike nam omogoča dokaj podrobno analizo in oblikovanje posameznih komponent ter njihovo relativno samostojno obravnavo. Povezave znotraj posameznega pogleda so dokaj močne, medtem ko so povezave med posameznimi vidiki ohlapnejše. Ker pa na celoto seveda ne smemo pozabiti, vzpostavimo povezave med posameznimi komponentami s pomočjo kontrolnega vidika.

Sam potek procesa je opredeljen z aktivnostmi, kontrolnimi tokovi in logiko povezovanja med aktivnostmi (točke združevanja in razvejitve), kar predstavlja *funkcijski vidik*. Informacijski objekti in dogodki sestavljajo *podatkovni vidik* - pasivno komponento, izvajalci (vloge) v povezavi z organizacijsko strukturo pa tvorijo *organizacijski vidik*. Povezave osnovnih vidikov znotraj arhitekture integriranega informacijskega sistema (*Architecture of Integrated Information Systems - ARIS*) prikazuje Slika 11.



Slika 11: ARIS-ovi osnovni vidiki modela procesa

## 2.3 Objektno usmerjeno modeliranje procesov

Objektno usmerjeno modeliranje procesov sledi splošnim načelom objektno usmerjenega modeliranja pri razvoju informacijskih sistemov [5,6,12].

### 2.3.1 Osnovni koncepti objektne modeliranja

Ena najbolj razširjenih tehnik objektne modeliranja je *Object Modelling Technique (OMT)*, ki jo je v začetku 90-tih let razvil J. Rumbaugh. V nadaljevanju bomo na kratko opredelili osnovne koncepte objektne

modeliranja, primer objektne modela pa prikazuje Slika 12.

Osnovni koncepti klasičnega objektne usmerjenega modeliranja [Š1, 5, 12] so:

#### Objekt (Object)

Objekti predstavljajo največkrat objekte, ki nastopajo v realnem svetu. V nasprotju s klasičnimi strukturiranimi metodami, ki skušajo zapletenost realnih objektov zmanjšati s tem, da ločeno obravnavajo njihov podatkovni in postopkovni vidik, pa pri objektne pristopu objekt združuje tako podatkovne strukture (*atributi objekta*) kot tudi postopke, ki se nad temi strukturami izvajajo (*operacije nad objekti*).

#### Razredi objektov (Class)

Objekte lahko po načelu generalizacije združujemo v razrede. Razred je torej skupina objektov s podobnimi lastnostmi, obnašanjem in povezavami z ostalimi objekti. Večina objektov izraža svoje posebnosti z različnimi vrednostmi atributov in z različnimi razmerji napram ostalim objektom.

Grafično je razred prikazan kot pravokotnik, razdeljen v tri dele: v prvem delu je navedeno ime razreda, v drugem so naštetih atributi in v tretjem operacije.

Pri operacijah se lahko pojavi tudi *polimorfizem*, kar pomeni da se ista operacija pri različnih razredih različno obnaša. *Metoda* je realizacija operacije v okviru posameznega razreda. Dostop do objektov in njihova obdelava sta mogoča samo z uporabo ene od realiziranih metod.

#### Razmerja (Association)

Razmerja nastopajo med razredi in prikazujejo, na kakšne načine so posamezni objekti lahko med seboj povezani. Razmerja lahko podrobneje opredelimo z *atributi razmerij, kardinalnostjo in vlogami*, ki jih imajo posamezni razredi v razmerju.

Razmerja med razredi so prikazana s črto, ki povezuje dva razreda med seboj. Ime povezave napišemo



Slika 12: Objektne diagram (povzeto po [12] - str. 54)

nad črto. Kardinalnost povezave je prikazana s polnim krogcem, če v eni povezavi lahko nastopa nič ali več objektov istega razreda, prazni krogec pa nakazuje, da lahko v povezavi nastopa le nič ali pa en objekt razreda. Na vsak konec razmerja lahko vpišemo ime vloge, ki jo ima posamezen razred v razmerju. Attribute razmerja vpišemo v pravokotnik, ki ga z zanko pripnemo k razmerju.

### Dedno pravilo (Inheritance)

Po načelu generalizacije lahko več razredov, ki imajo podobne podatkovne strukture in operacije, združimo v en nadrazred. Tako dobljen razred se imenuje splošeni (abstraktni) razred in združuje skupne attribute in operacije elementarnih razredov. Vsi elementarni razredi dedujejo attribute in operacije od posplošenega razreda, poleg tega pa ima lahko vsak razred opredeljene še svoje attribute in operacije.

Simbol, ki nakazuje povezanost razredov po načelu generalizacije, je trikotnik, poleg katerega vpišemo ime povezave.

### Prednosti objektnega pristopa

Prednosti, ki jih objektni pristop [5,8,10] v splošnem prinaša, so naslednje:

- skupna obravnava podatkovnega in postopkovnega vidika sistema, kar pripomore k večji konsistentnosti modela in tudi končne informacijske rešitve;
- večkratna uporaba že razvitih razredov (reusability) znatno skrajša čas razvoja nove informacijske rešitve in s tem prispeva tudi k zmanjšanju stroškov;
- prilagodljivost (flexibility) - enak objektni model lahko uporabimo na različnih področjih;
- uporaba že preverjenih objektnih razredov prispeva k večji zanesljivosti in kakovosti nove rešitve.

### 2.3.2 Procesi kot objekti

Klasični objektno-orientirani pristopi obravnavajo objekte kot statične komponente sistema (entitete), zato je potrebno dinamično obnašanje sistema prikazati z drugimi tehnikami, kot je na primer diagram prehajanja stanj [9,11]. Poleg tega so ti pristopi v glavnem osredotočeni na posamezen razred objektov, ki običajno pokriva le majhen del poslovnega procesa, kar omejuje tudi prikaz in razlago medsebojnih povezav objektov v okviru celotnega poslovnega procesa.

Glavna prednost novejših objektno usmerjenih pristopov pa je, da kot objekte obravnavajo tudi procese (dinamične komponente sistema). Proces kot objekt (procesni objekt) lahko opredelimo z naslednjimi komponentami [6]:

- z začetnim in končnim dogodkom;
- s procesnimi objekti in povezavami med njimi (npr. v obliki dogodkovno vodene verige - EPC);

- z informacijsko tehnologijo, ki podpira izvajanje procesa;
- z informacijskimi objekti, ki so potrebni za izvajanje procesa ali pa v procesu nastajajo oziroma se spreminjajo.

Prvi dve komponenti predstavljata dinamični vidik procesa (pri klasičnih objektih so to operacije) in sta za opredelitev procesnega objekta obvezni, drugi dve komponenti pa predstavljata statični vidik.

Podobno kot pri klasičnih metodah modeliranja procesov lahko tudi na tem področju uporabimo *metodo dekompozicije*, ki vodi v hierarhijo procesnih objektov, kjer se procesi obravnavajo na različnih nivojih abstrakcije. Vendar pa se tu dekompozicija loči od klasičnega strukturnega grafa po tem, da pri procesnem objektu na višjem nivoju navedemo tudi, kako so procesni objekti na nižjem nivoju povezani med sabo, saj je to - kot smo že omenili - ena od komponent, s katerimi je opredeljen proces kot objekt. V tem pogledu ne ločimo med aktivnostmi, podprocesu in procesu, temveč vsakega od naštetih konceptov obravnavamo kot procesni objekt na različnem nivoju abstrakcije.

### 2.3.3 Referenčni modeli

Od osnovnih konceptov klasičnega objektnega modeliranja se je pri objektnem modeliranju procesov izkazalo za zelo koristno načelo generalizacije in z njim povezano dedno pravilo [6,12]. Z uporabo tega načela lahko z različnimi vrednostmi statičnih komponent procesnega objekta dobimo različice (alternative) procesnega objekta, ki se razlikujejo glede na uporabo različne tehnologije in informacijskih objektov. Splošeni procesni objekt, na podlagi katerega smo z uporabo dednega pravila razvili nove različice, lahko imenujemo tudi *referenčni model procesa*. Po njem se lahko v naši organizaciji ravna več različnih procesov, ki se med seboj razlikujejo le po tem, da v njih nastopajo različni informacijski objekti (dokumenti in podatkovne zbirke) ali pa da se nekatera zaporedja aktivnosti, ki jih referenčni model dovoljuje, v posameznem procesu, sploh ne izvajajo.

## 3. Končne ugotovitve

Pregled prikazanih metod in tehnik modeliranja procesov lahko strnemo v naslednjo tabelo, ki prikazuje modeliranje karakteristik, pomembnih za upravljanje delovnih procesov, ki jih posamezne metode omogočajo in katerih ne.

Ne glede na to, kakšne metode in tehnike bomo uporabili pri modeliranju procesov, se moramo zavedati, da modeliranje procesov kot tudi uvajanje sistemov za upravljanje delovnih procesov ne sme biti samo sebi

Tabela 1: Primerjava različnih tehnik in metod modeliranja procesov glede na karakteristike, pomembne za upravljanje delovnih postopkov

	diagram toka podatkov	kombiniran DTP	diagram prehodov stanj	diagram vlog udeležencev	eEPC	objektni model
aktivnosti	√	√	√	√	√	√
vrstni red	√	√	√	√	√	√
združevanje in razvejanje	-	?	?	?	√	√
pogoji	-	√	√	-	?	?
vloge	-	-	-	√	√	√
dogodki	-	-	?	-	√	√
stanja	-	-	√	?	?	?
informacijski objekti	√	√	-	-	√	√
tok inf. objektov	√	√	-	-	-	√
orodja za izvedbo aktivnosti	-	-	-	-	-	√

namen. Oboje moramo videti kot priložnost za prenovu ali vsaj optimizacijo obstoječih procesov, saj je z novo in nemalokrat drago informacijsko tehnologijo nesmiselno podpirati obstoječe procese, ki lahko zajemajo aktivnosti, ki niti niso potrebne ali pa ki bi se dale izvajati mnogo bolj racionalno.

## Literatura

- [1] Bakker G: OMT Object Modelling - Notation, Concepts and Constructs, <http://www.edu.cs.utwente.nl/centing./miso/notation.html>, september 1997
- [2] Dietz J.L.G., Mulder: "Integrating the Strategic and Technical Approach to Business Process Engineering", *Business Process Modelling* (eds. Scholz-Reiter B., Stickle E.) str. 188-204, Springer-Verlag, Berlin-Heidelberg-New York-Tokyo, 1996; ISBN 3-540-61707-8
- [3] Jensterle R.: "Načrtovanje 'workflow' aplikacij", *Zbornik referatov s posvetovanja Dnevi slovenske informatike '97*, str. 89-98, Slovensko društvo Informatika, 1997, ISBN
- [4] Kovačič A.: "Prenova in informatizacija poslovanja: pristopi in izkušnje", *Zbornik referatov s posvetovanja Dnevi slovenske informatike '97*, str. 252-260, Slovensko društvo Informatika, 1997, ISBN
- [5] Kovačič A., Vintar M.: *Načrtovanje in gradnja informacijskih sistemov*, DZS, Ljubljana 1993, ISBN 86-341-1179-2
- [6] Lang K., Taumann W., Bodendorf F.: "Business process Reengineering with Reusable Reference Process Building Blocks", *Business Process Modelling* (eds. Scholz-Reiter B., Stickle E.) str. 265-290, Springer-Verlag, Berlin-Heidelberg-New York-Tokyo, 1996; ISBN 3-540-61707-8
- [7] Leben A., Vintar M.: "Od prenove poslovanja k upravljanju delovnih procesov", *Uporabna informatika*, letnik V št. 3, str. 18-25, 1997; ISSN 1318-1882
- [8] Mehler-Bicher A.: "An Object-Oriented and Business Process-Based Meta Model of an Architecture for Management Support System", *Business Process Modelling* (eds. Scholz-Reiter B., Stickle E.) str. 291-332, Springer-Verlag, Berlin-Heidelberg-New York-Tokyo, 1996; ISBN 3-540-61707-8
- [9] Miers D.: "Use of Tools and Technology Within a BPR Initiative", *Business Process Re-engineering: myth & reality* (ed. Coulson-Thomas C.), str. 142-165, Kogan Page Ltd., London, 1994; ISBN 0-7494-1442-1
- [10] Mihelič L.: "Primerjava tradicionalnega in objektno orientiranega pristopa pri razvoju informacijskih sistemov ob uporabi orodja CASE", *Uporabna informatika*, letnik V št. 3, str. 26-32, 1997; ISSN 1318-1882
- [11] Rohloff M.: "An Object Oriented Approach to Business Process Modelling", *Business Process Modelling* (eds. Scholz-Reiter B., Stickle E.) str. 251-264, Springer-Verlag, Berlin-Heidelberg-New York-Tokyo, 1996; ISBN 3-540-61707-8
- [12] Scheer A.W.: *Business Process Engineering*, Springer-Verlag, Berlin-Heidelberg-New York-Tokyo, 1996; ISBN 3-540-58234-7
- [13] Vintar M.: *Informatika*, PACO, Ljubljana 1996, ISBN 961-90327-0-5
- [14] Wiczerzycki W.: "Process Modelling and Execution in Workflow Management Systems by Event-Driven Versioning", *Business Process Modelling* (eds. Scholz-Reiter B., Stickle E.) str. 43-66, Springer-Verlag, Berlin-Heidelberg-New York-Tokyo, 1996; ISBN 3-540-61707-8
- [15] Zukunft O., Rump F.: "From Business Process Modelling to Workflow Management: An Integrated Approach", *Business Process Modelling* (eds. Scholz-Reiter B., Stickle E.) str. 3-22, Springer-Verlag, Berlin-Heidelberg-New York-Tokyo, 1996; ISBN 3-540-61707-8
- [16] Yourdon E.: *Modern Structured Analysis*, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1989, ISBN 0-13-598632-X

*Anamarija Leben je diplomirala na Fakulteti za organizacijske vede v Kranju s področja oblikovanja podatkovnih modelov. Svojo poklicno pot je začela kot programerka in kasneje nadaljevala kot sistemski analitik na področju oblikovanja in izgradnje celovitih informacijskih rešitev. Od leta 1995 je redno zaposlena kot asistentka pri predmetih Informatika ter Informacijski sistemi na Visoki upravni šoli v Ljubljani.*