



UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Matic Standeker

Obravnava in modeliranje ad-hoc poslovnih procesov

MAGISTRSKO DELO

Mentor: prof. dr. Marko Bajec

Ljubljana, 2010

Št.: 108-MAG-ISO/2010

Datum: 4.3.2010



MATIC STANDEKER, univ. dipl. inž. rač. in inf.

Ljubljana

Fakulteta za računalništvo in informatiko Univerze v Ljubljani izdaja naslednjo magistrsko nalogo

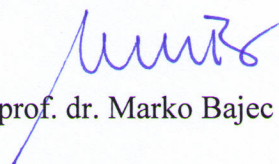
Naslov: Obravnava in modeliranje ad-hoc poslovnih procesov

Introducing and modelling of ad-hoc business processes

Tematika naloge:

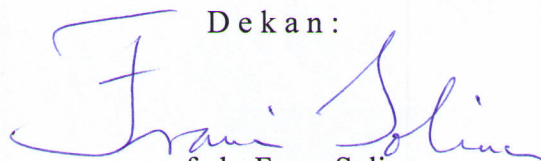
Tema magistrske naloge obravnava modeliranje dinamike v poslovnih procesih. Poslovni procesi v sodobnih združbah so vse bolj dinamični in nepredvidljivi. Zaradi te dinamike jih je vse težje popisati in definirati aktivnosti ter vrstni red izvajanja. Danes v običajnih pristopih manjkajo koncepti, ki bi uspešno in učinkovito opisali dinamičen vidik poslovnih procesov. Kandidat naj pregleda uveljavljene pristope k modeliranju poslovnih procesov. Osredotoči naj se na to, kako se v teh pristopih zajema dinamiko poslovnih procesov. Na osnovi študije primerov naj razišče, v kolikšni meri in kako učinkovito se lahko z obstoječimi pristopi modeliranja zajame dinamiko poslovnih procesov. Predlaga naj novo, prilagojeno modelirno tehniko, s katero se odpravi pomanjkljivosti obravnave dinamike poslovnih procesov v običajnih modelirnih tehnikah. Nov pristop naj preizkusi na realnem primeru dinamičnega poslovnega procesa in tako preveri uspešnost novega pristopa.

Mentor:


prof. dr. Marko Bajec



Dekan:


prof. dr. Franc Solina

IZJAVA O AVTORSTVU

magistrskega dela

Spodaj podpisani **Matic Standeker**,

z vpisno številko **63010137**,

sem avtor magistrskega dela z naslovom:

Obravnava in modeliranje ad-hoc poslovnih procesov.

S svojim podpisom zagotavljam, da:

- sem magistrsko delo izdelal samostojno pod vodstvom mentorja **prof. dr. Marka Bajca**;
- so elektronska oblika magistrskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko magistrskega dela;
- soglašam z javno objavo elektronske oblike magistrskega dela v zbirki "Dela FRI".

V Ljubljani, dne 12.3.2010

Podpis avtorja:

Zahvala

Na tem mestu bi se rad zahvalil prof. dr. Marku Bajcu za mentorstvo pri magistrski nalogi.

Prav tako bi se rad zahvalil Štefanu Furlanu in Lovru Šublju za koristne ideje in nasvete.

Kazalo

Povzetek	1
1 Uvod	5
1.1 Opis problema	5
1.2 Opis ciljev	6
1.2.1 Izhajanje iz realnega problema	6
1.2.2 Percepcija in definicija poslovnih procesov	7
1.2.3 Predlog modeliranja poslovnih procesov	7
1.2.4 Premostitev prepada med IT in poslovnimi uporabniki	7
2 Opis problematike	9
2.1 Funkcijska delitev združb	9
2.2 Poslovni procesi	10
2.3 Procesna usmerjenost	11
2.4 Delovni postopek	14
2.5 Storitveno usmerjena arhitektura	15
2.6 Dinamika poslovnih procesov	16
2.6.1 Zunanje spremembe poslovnih procesov	16
2.6.2 Notranja dinamika poslovnih procesov	17
2.6.3 Medorganizacijski procesi	19
2.7 Motivacija za obvladovanje dinamike v poslovnih procesih	20
3 Raziskovalni pristop	21
3.1 Metode dela	21
3.1.1 Predlog procesa	21
3.1.2 Prototip	21
3.1.3 Definicija diagramske tehnike	22
3.2 Pregled temeljnih virov	22
4 Ad-hoc poslovni procesi	24
4.1 Definicija	24
4.2 Obravnava ad-hoc procesov	25

5	Pregled modeliranja poslovnih procesov	26
5.1	Modeliranje poslovnih procesov	26
5.2	Uporaba Petrijevih mrež kot formalnega jezika za modeliranje	29
5.2.1	Osnovni gradniki Petrijeve mreže	29
5.2.2	Primer modeliranja ad-hoc procesa s Petrijevimi mrežami	31
5.2.3	Uporabnost in primernost Petrijevih mrež za modeliranje ad-hoc procesov	33
5.3	BPMN diagramska tehnika	34
5.3.1	Osnovni gradniki BPMN	35
5.3.2	Primer modeliranja ad-hoc procesa z BPMN notacijo	37
5.3.3	Uporabnost in primernost BPMN za modeliranje ad-hoc procesov	39
5.4	Diagram EPC	40
5.4.1	Osnovni gradniki EPC jezika	40
5.4.2	Primer modeliranja ad-hoc procesa z EPC diagramom	42
5.4.3	Uporabnost in primernost BPMN za modeliranje ad-hoc procesov	43
5.5	Jezik UML	43
5.5.1	Osnovni gradniki diagrama aktivnosti	44
5.5.2	Primer modeliranja ad-hoc procesa z diagramom aktivnosti	45
5.5.3	Uporabnost in primernost diagrama aktivnosti za modeliranje ad-hoc procesov	46
5.6	Ostali jeziki	47
6	Alternativni pristopi k modeliranju poslovnih procesov	48
6.1	Obravnava poslovnih procesov in njihovega okolja kot sistemov	48
6.2	Uporaba ECA pravil	50
6.3	Kognitiven pristop	52
6.4	Agentni sistemi	54
7	Predlagan koncept modeliranja ad-hoc procesov	57
7.1	Konceptualni model ad-hoc procesa	57
7.2	Osnovni elementi modeliranja ad-hoc procesa	59
7.2.1	Okolje ad-hoc procesa	59
7.2.2	Končni in vmesni cilji ad-hoc procesa	63
7.2.3	Procesna entiteta in procesni agent	67
7.3	Izvajanje ad-hoc procesa	69
8	Diagramska tehnika	72
8.1	Pregled	72
8.2	Gradniki diagrama	73
8.2.1	Vmesni cilj	73
8.2.2	Množica aktivnosti	74
8.2.3	Potrebni viri	74
8.2.4	Dogodki	75

8.2.5	Artefakti	76
8.2.6	Ključni kazalci uspeha	76
9	Študija primera	78
9.1	Opis primera	78
9.1.1	Goljufije v zavarovalništvu	78
9.1.2	Proces razreševanja sumljivega zahtevka	79
9.1.3	Ad-hoc proces zmodeliran z UML diagramom aktivnosti	80
9.2	Procesni diagram z novo diagramsko tehniko	81
9.2.1	Kombinacije doseganja končnega cilja ad-hoc procesa	81
9.2.2	Primer: vmesni cilj <i>potrditev nepravilnosti na postavkah računa</i>	84
9.2.3	Primer: vmesni cilj <i>spremljanje in podpora postopka civilne tožbe</i>	86
10	Glavne ugotovitve	89
10.1	Današnji poslovni procesi in združbe	89
10.2	Ciljna usmerjenost	90
10.3	Procesne entitete in procesni agenti	90
10.4	Možnosti implementacije	91
11	Zaključek	92

Seznam uporabljenih kratic in simbolov

B2B	Business to business - opisuje zvezo med dvema poslovnima sistemoma
B2C	Business to customer - opisuje zvezo med poslovnim sistemom in končno stranko
BAM	Business activity monitoring - nadzor nad izvajanjem poslovnih procesov
BPD	Business process diagram - diagram poslovnega procesa, narejen v BPMN
BPEL	Business process execution language - jezik za izvajanje poslovnih procesov
BPI	Business process improvement - izboljšava in optimizacija poslovnih procesov
BPM	Business process management - upravljanje poslovnih procesov
BPMI	Business process management initiative - neprofitna organizacija, ki skrbi za razvoj in standardizacijo BPM področja
BPMN	Business process modelling notation - grafični jezik za modeliranje poslovnih procesov
BPR	Business process redesign - postopek uvedbe in prenove poslovnih procesov
BRMS	Business rules management system - sistem za upravljanje poslovnih pravil
CSF	Critical success factor - faktor doseganja ciljev
DFD	Diagram funkcionalne dekompozicije
EPC	Event-driven process chain - Dogodkovno voden proces
ERP	Enterprise resource planning - integriran informacijski sistem
IKT	Informacijsko komunikacijska tehnologija
IT	Information technology - mišljeno je celotno področje, ki pokriva IKT rešitve

IZS	Izvajalec zdravstvenih storitev
KPI	Key performance index - ključni kazalci uspešnosti
MDA	Model driven development - modelno voden razvoj
OCL	Object constraint language - Objektni jezik za izražanje omejitev
OMG	Object management group - konzorcij z namenom standardizacije in razvoja standardov na področju objektno orientiranih sistemov
PAIS	Process aware information system - informacijski sistemi, ki so osnovani na poslovnih procesih
SOA	Service oriented approach - storitveno usmerjen pristop
UML	Unified modelling language - grafična notacija za modeliranje informacijskih sistemov
WfMC	Workflow Management Coalition - organizacija, ki skrbi za razvoj tehnologij za podporo delovnih postopkov in BPM
XPDL	XML Process definition language - jezik za definiranje poslovnega procesa v XML notaciji, ki ga predlaga WfMC
YAWL	Yet another workflow language - jezik za definiranje delovnih postopkov

Povzetek

V magistrski nalogi je predstavljeno področje modeliranja dinamičnih poslovnih procesov. Definirali smo pojem "ad-hoc" procesa in raziskali različne dimenzije dinamike takšnih procesov. Z namenom, da razvijemo dopolnjeno modelirno tehniko, ki bi znala takšne procese dobro opisati, smo pregledali, kako se v obstoječih in uveljavljenih pristopih modeliranja obravnava dinamične procese. Ugotovili smo, da obstoječe modelirne tehnike ne ponujajo dovolj dobrih možnosti za obravnavo in modeliranje dinamičnih poslovnih procesov. Zato smo predlagali nov pristop k modeliranju poslovnih procesov, s katerim lahko lažje in na učinkovit način opišemo dinamiko v poslovnih procesih.

Predlagan koncept dopolnjuje obstoječe modelirne tehnike in ga lahko uporabimo v kombinaciji z uveljavljenimi pristopi modeliranja. Uporabimo ga lahko kot osnovo za procesni model ad-hoc procesov ali pa kot osnovo za podroben implementacijski načrt IT podpore ad-hoc procesov.

Uporabnost takšnega pristopa smo prikazali na realnem primeru iz gospodarstva. Uporabili smo poslovni proces iz zavarovalništva, za katerega je značilna visoka notranja in zunanja dinamika ter nepredvidljivost zaporedja izvajanja aktivnosti v procesu. Gre za proces razreševanja sumljivih škodnih primerov, katere je treba pred likvidacijo ustrezno obravnavati in v primeru neupravičenosti zahtevka izpeljati proces povrnitve stroškov. Z novo modelirno tehniko smo proces uspešno zmodelirali in prikazali uporabnost dopoljnega načina modeliranja poslovnih procesov.

Ključne besede:

poslovni procesi, ad-hoc procesi, modeliranje poslovnih procesov, modeliranje ad-hoc procesov

Abstract

This dissertation introduces the modelling of dynamic business processes. The term "ad-hoc" process is being defined as well as research of different dimensions of dynamics in this type of processes. With the intention of introducing a new, supplemented version of modelling technique - one that could describe this type of "ad-hoc" processes - we examined how already existing and recognised modelling approaches deal with dynamic processes. Our research determined that the existing modelling techniques do not offer sufficient options for treating and modelling of dynamic business processes. Therefore we introduced a new approach of modelling business processes, one that offers a more efficient and elegant description of dynamic business processes.

The proposed concept completes existing modelling techniques and can be used in combination with already established approaches of modelling. It can be used as a basis for process model of ad-hoc business processes as well as a basis for a detailed plan for implementing IT support of ad-hoc processes.

We described the applicability of this approach with a key-study from the insurance field. The business process from the insurance field, which is significant for its high internal and external dynamics as well as unpredictable succession of execution along the process is being applied. The key study deals a business process with suspicious cases which require proceedings prior to the act of liquidation, as well as carrying out the process of refund in unjustifiable cases. With the new modelling technique we were able to model the process successfully. At the same time we were able to present employment of supplemented method of modelling of ad-hoc business processes.

Key words: business processes, ad-hoc processes, business process modelling, ad-hoc process modelling

Poglavje 1

Uvod

V uvodnem poglavju je na kratko podana problematika področja. Definirali bomo raziskovalno vprašanje in napovedali cilje, ki smo si jih zastavili pred izdelavo magistrske naloge.

1.1 Opis problema

Današnja IT podpora ponuja dobre rešitve za dobro definirane, statične in predvidljive procese. Na voljo imamo različne metodologije, s katerimi lahko na preizkušen in učinkovit način pristopimo k uvajanju in podpori poslovnih procesov.

Skozi čas je v organizacijski znanosti in predvsem v praksi prišlo do spoznanj, da današnja IT podpora poslovnim procesom ne nudi ustreznega zajema dinamične komponente procesov. Izkušnje namreč kažejo, da je velik delež poslovnih procesov dinamičen (glej Poglavje 2.6) in da trenuten pristop IT področja dinamičnega vidika procesov ne obravnava dovolj učinkovito.

Problem se pojavi na dveh stopnjah. Zaradi togosti IT podpore v združbah le stežka sledijo spremembam poslovnih procesov, do katerih pride zaradi zahtev od zunaj. S togostjo IT podpore je mišljena takšna implementacija, ki za vnos sprememb zahteva veliko časa in energije. Posledično so stroški sledenja spremembam, ki so vsiljene od zunaj, visoki in se zanje vodstveni kader pogosto ne odloči. IT lahko tako postane zaviralni element pri hitremu odzivu na spremembe na trgu, kar pa danes zaradi ostrih razmer na trgih vsak poslovni sistem hitro pripelje do propada.

Po drugi strani se izkaže, da je velik delež poslovnih procesov notranje dinamičen. Sodobni pristopi sicer omogočajo podporo takšnim poslovnim procesom na račun povečevanja kompleksnosti procesnih modelov in delovnih postopkov. Običajni BPM pristop se osredotoča na pretok podatkov in kontrolnega toka skozi poslovni proces in v primeru visoke dinamičnosti in nedoločenosti procesa je pretok modeliran z veliko vejitvami. Vejitve lahko dodajamo v nedogled, vendar se v praksi to izkaže za zelo slabo rešitev. Kompleksni procesni modeli so neberljivi in praktično neuporabni, zelo težko pa je v takšen model vnesti tudi naknadne spremembe.

Zaradi razvoja IKT se danes IT podpora uporablja v praktično vsakem segmentu delovanja sodobnih združb. V preteklosti je bilo zaradi tehnoloških omejitev smotrno podpreti samo najbolj kritičen del ustvarjanja poslovnih učinkov. Običajno govorimo o podpori proizvodnih poslovnih procesov, ki spadajo med ključne poslovne procese združb. Zaradi večje dostopnosti IKT, razvoja IT področja in potrebe po vse boljši in celovitejši podpori delovanja združb, se je danes IT podpora razširila in je vse bližje končnemu uporabniku. Odvisnost med IT razvijalci in poslovnimi uporabniki, katerim je IT podpora namenjena, je vse večja. Nastane velik problem prepada med IT in poslovnim svetom, saj v IT področju uporabljamo tehnične koncepte, ki so poslovnim uporabnikom težko razumljivi. Ta prepad je očiten tudi pri uvedbi IT podpore poslovnih procesov. Ljudje iz IT popišejo in analizirajo poslovno domeno, da lahko identificirajo poslovne procese, ki jih je treba. Vendar je rezultat takšne analize zaradi tehnične narave pogosto nerazumljiv poslovnim uporabnikom. Podobno je ob spremembi poslovnega procesa, ki jo mora poslovni uporabnik sporočiti IT podpori. Za premostitev prepada med IT področjem in poslovnimi uporabniki se izvaja vse več raziskav in poizkusov, kjer lahko poslovni uporabnik aktivno sodeluje pri tvorbi in spreminjanju IT podpore. Danes je miselnost, da je IT samostojna in samozadostna panoga, preživeta. Zato je treba razmišljati o takšni podpori poslovnih procesov, ki je poslovnemu uporabniku razumljiva in je vanjo tudi sam vključen aktivno. To smo imeli tudi v mislih pri razvoju našega pristopa modeliranja ad-hoc procesov.

1.2 Opis ciljev

Opisani problemi se pojavljajo v gospodarstvu in so popolnoma praktične narave. Z njimi se v IT področju srečamo vsak dan in jih poizkušamo reševati na različne načine. Namen pričujoče magistrske naloge je zajem in opis te problematike in poizkus definiranja novega pristopa, ki bo omogočal odpravo teh problemov. Poslovne procese bi radi opisali na takšen način, pri katerem bi lahko učinkovito podpirali spremembe, ki nenehno nastajajo v izvajanju poslovanja. Še posebej veliko pozornosti smo namenili zajemu in podpori notranje dinamike poslovnih procesov. Obenem pa naj bi bil takšen pristop uporaben tako na poslovnem kot tehničnem nivoju.

1.2.1 Izhajanje iz realnega problema

Problematika podpore dinamike poslovnih procesov ne izhaja iz teorije, temveč iz realnega problema, na katerega smo naleteli pri izdelavi podpore poslovnih procesov v zavarovalništvu. Gre za izdelavo specifikacije za podporo procesa razreševanja škodnih primerov, kar spada sicer v širši kontekst detekcije in obravnave goljufij, ki se dogajajo na področju zavarovalništva. Sistem, ki ga implementiramo, v prvi fazi ponuja dobro rešitev za odkrivanje nepravilnosti in goljufij pri škodnih zahtevkih, ki jih mora plačati zavarovalnica. V drugi fazi pa mora sistem podpirati proces razreševanja posameznih primerov, ki se nanašajo na potencialne goljufije. Izkazalo se je, da ima proces razreševanja visoko stopnjo nedoločenosti in da je izpostavljen nenehnim spremembam iz okolja. Zaradi

teh karakteristik nismo mogli uporabiti običajnih BPM metodologij, temveč smo morali ubrati drugače pristop.

1.2.2 Percepcija in definicija poslovnih procesov

Običajne definicije poslovnih procesov ne opisujejo dinamike poslovnih procesov. V Poglavju 2.2 navedene definicije poslovnega procesa bodo služile kot izhodišče za izpeljavo nove definicije dinamičnih ali t.i. ad-hoc poslovnih procesov. Takšna definicija bo služila kot osnova za izpeljavo prilagojene percepcije poslovanja združbe.

Za uspešno podporo ad-hoc poslovnih procesov je treba prilagoditi percepcijo delovanja združbe. Kot enega izmed ciljev magistrske naloge bomo predlagali dojetanje poslovnih procesov kot množico vmesnih ciljev, ki skupaj tvorijo končni rezultat procesa. Poizkušali bomo dokazati, da je takšna percepcija primerna za učinkovito modeliranje poslovnih procesov, zajem dinamike in hitrih sprememb iz okolja, ter da lahko ob takšni percepciji uspešno vpeljemo razumljivo, skalabilno in odzivno IT podporo.

Glavni cilj uvedbe nove percepcije poslovnih procesov je učinkovita, enostavna in razumljiva obravnava ter podpora za spremljanje sprememb in notranje dinamike poslovnih procesov. Ob izhajanju iz realnega problema, ki obsega zelo dinamične in nepredvidljive poslovne procese, bi radi predlagali nov pristop, ki bo odpravil pomanjkljivosti običajnega BPM pristopa. Nov pristop mora omogočati učinkovito obravnavo ad-hoc poslovnih procesov. Omogočati mora dobre temelje za implementacijo IT podpore z omejitvami, ki jih prinaša današnja stopnja razvoja IKT.

1.2.3 Predlog modeliranja poslovnih procesov

Hkrati s predlogom prilagojene percepcije poslovnih procesov je treba uvesti tudi ustrezno tehniko modeliranja poslovnih procesov. Predlagana tehnika se bo osredotočala predvsem na zajem dinamičnega vidika ad-hoc procesov, hkrati pa ne bo zapostavljala ostalih lastnosti poslovnih procesov. V magistrski nalogi bi radi predstavili novo, prilagojeno modelirno tehniko, ki bo temeljila na predlagani percepciji, izhajala pa bo iz obstoječih in uveljavljenih notacij za zapis poslovnih procesov. Radi bi uporabili čim več znanja iz BPM področja in predlagali takšno notacijo ad-hoc procesov, s katero bo možno zajeti dovolj tehničnih detajlov in bo primerna za uporabo na tehnični IT ravni, hkrati pa bo dovolj razumljiva in povedna tudi za poslovne uporabnike. Nova modelirna tehnika ne sme izključevati uporabe uveljavljenih BPM modelirnih tehnik in mora služiti kot dopolnilo za celovito predstavitev in zapis poslovnih procesov.

1.2.4 Premostitev prepada med IT in poslovnimi uporabniki

Eno izmed vodil, katerih smo se držali, je izdelava takšnega pristopa, ki bo čim bolj zmanjšal prepad med tehničnim IT področjem in poslovnimi uporabniki. Kot smo opisali že v Poglavju 1.1, je to velik problem in z uvedbo poslovnim uporabnikom nerazumljivega pristopa za podporo ad-hoc procesov lahko povzročimo še več težav. Zavedati se moramo

namreč dejstva, da je dobra IT podpora zelo odvisna od poslovnih uporabnikov. Če poslovnim uporabnikom ponudimo takšno podporo ad-hoc procesov, ki bo njim razumljiva in pregledna, bodo lahko do precejšnje mere sami prevzeli skrb za ustrezno prilagajanje in razvoj podpore dejanskemu stanju v združbi. Poslovni uporabnik je tisti, ki spremembo prvi občuti ali jo celo povzroči. Iz tega razloga bi v našem pristopu radi uporabili takšne gradnike in postopke, ki bodo dovolj blizu tudi poslovnim uporabnikom in jih bodo lahko zato uporabljali sami, brez pretiranega vmešavanja osebja iz področja IT.

Poglavje 2

Opis problematike

V naslednjem poglavju sledi opis področja in problematike dinamičnih poslovnih procesov. Pogledali bomo, kako je organizirano delo v združbah z vidika organizacijske znanosti in kako je informacijska znanost poizkušala podpreti delovanje združb. Podali bomo motivacijo in razloge za uvedbo poslovnih procesov. Pregledali bomo kakšne tipe poslovnih procesov poznamo in kako definiramo dinamiko poslovnih procesov, kar je bistveno za nadaljnji potek magistrske naloge.

2.1 Funkcijska delitev združb

Za doseganje čim večjega učinka pri ustvarjanju dobrin je moral človek že od nekdaj ustrezno organizirati svoje dejavnosti. Ljudje so kmalu ugotovili, da morajo za uspešno nastopanje na trgu svoje delo ustrezno organizirati. Skozi zgodovino so se zaradi vse večjega povpraševanja po dobrinah, širitve trga in zaostrovanja konkurence rojevale različne organizacijske šole, ki so raziskovale in gojile načrtno organiziranje dela [14]. Eden izmed prvih ukrepov za doseganje boljših delovnih rezultatov je specializacija, ki smo ji priča že zelo zgodaj v zgodovini. Dejstvo je, da lahko delavec, ki se specializira za opravila z določenega področja, to delo opravlja boljše, hitrejše in kakovostneje, kot če bi opravljal širši asortiment opravil. Pri prehodu v industrijsko dobo so uvedli dodatno specializacijo, kjer je bil posamezen delavec osredotočen na točno določen tip opravil. V združba se ustvari *funkcijska delitev*.

Funkcijska delitev v združbo uvede funkcionalna področja. Vsako funkcionalno področje združuje množico aktivnosti, ki uresničuje posamezno posebno nalogo združbe [15]. S funkcijsko delitvijo združb se doseže manjšo kompleksnost in specializiranost opravkov po funkciji [17].

V funkcionalnih področjih se stalno in ponavljajoče odvijajo *poslovne funkcije*. V običajnem proizvodnem podjetju govorimo o kadrovske, naložbeni, nabavni, proizvodni, prodajni in finančni poslovni funkciji. Glede na posebne naloge posamezne združbe se seveda uvede dodatne poslovne funkcije.

2.2 Poslovni procesi

Do končnega izdelka pride samo s sodelovanjem več funkcionalnih področij. Aktivnosti, ki so potrebne za dokončan izdelek ali storitev, potekajo v določenem vrstnem redu v različnih funkcionalnih področjih. Tako pridemo do definicije poslovnega procesa.

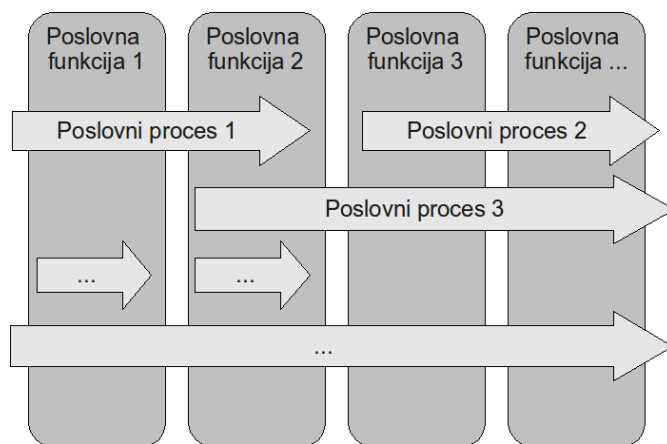
Definicija 2.2.1. Poslovni proces je logično in časovno povezana množica aktivnosti, ki na vhod dobi zahtevo uporabnika, ustvari dodano vrednost in na izhodu poda nov izdelek [7].

Definicija 2.2.2. Poslovni proces je množica koordiniranih aktivnosti, ki jih izvedejo ljudje ali orodja s ciljem doseganja poslovnega rezultata [9].

Najbrž najbolj preprosta, intuitivna in za naše potrebe primerna je naslednja definicija poslovnega procesa:

Definicija 2.2.3. Poslovni proces je zaporedje aktivnosti, ki pripeljejo do nekega rezultata [2].

Poslovni proces povezuje aktivnosti, ki se praviloma izvajajo v več različnih funkcionalnih področjih. Za uspešno delovanje združbe je potrebno sodelovanje med funkcionalnimi področji, to pa je zaradi običajno stroge funkcionalne ureditve združb oteženo. Zavedati se moramo, da je funkcijska delitev t.i. vertikalna razdelitev, saj združuje posamezna področja sorodnih opravil oz. aktivnosti. Da pridemo do končnega izdelka ali storitve, kar je cilj združbe oz. poslovnega sistema, morajo biti izvedena opravila iz več poslovnih funkcij. Gre torej za sosledje aktivnosti, ki se nahajajo v različnih poslovnih funkcijah. Poslovni procesi v vertikalni funkcijski razdelitvi potekajo horizontalno.



Slika 2.1: Poslovne funkcije in poslovni procesi

Poslovni proces ima točno definiran začetek z definiranim vhodom, prav tako pa ima definiran konec z definiranim izhodom.

Temeljna delitev poslovnih procesov v združbi glede na njihovo vlogo pri ustvarjanju dodane vrednosti je [1, 9]:

Ključni ali temeljni poslovni procesi V teh procesih se neposredno vpliva na dodano vrednost pri ustvarjanju izdelkov oz. storitev.

Podporni poslovni procesi V teh poslovnih procesih se posredno vpliva na dodano vrednost pri ustvarjanju izdelkov oz. storitev. Takšni poslovni procesi zagotavljajo ustrezne vire in pogoje za izvajanje ključnih poslovnih procesov.

Za dobro razumevanje tematike in snov magistrske naloge je potrebno omeniti tudi glavne tri skupine lastnosti poslovnega procesa po [2]:

Strukturne lastnosti poslovnega procesa Vsak poslovni proces se da opazovati na različnih stopnjah podrobnosti. V osnovi je poslovni proces sestavljen iz zaporedja aktivnosti, glede na stopnjo opazovanja pa se lahko posamezno aktivnost podrobneje opredeli kot dodaten pod - poslovni proces.

Izvajalne lastnosti poslovnega procesa Za vsak poslovni proces je treba določiti, kakšen je vhod v poslovni proces in kakšen je cilj oz. izhod izvajanja poslovnega procesa. Treba je definirati, kdo je lastnik poslovnega procesa, kdo so akterji in uporabljeni viri med izvajanjem poslovnega procesa in kakšne morajo biti izvajalne sposobnosti poslovnega procesa.

Omejitve poslovnega procesa Stopnja določenosti poslovnega procesa je ena izmed glavnih lastnosti, ki nižajo entropijo celotnega sistema oz. poslovanja združbe. Omejitve se nanašajo tako na konsistentnost rezultatov izvajanja poslovnega procesa kot na notranjo strukturo in zaporedje izvajanja aktivnosti v poslovnem procesu. Prav tako je treba pri določanju poslovnega procesa upoštevati omejitve okolja, v katerem se procesi izvajajo.

Lahko rečemo, da je bilo v preteklosti na področju IT podpore največ pozornosti usmerjene na lastnosti poslovnih procesov iz prvih dveh naštetih skupin. Trenutno stanje IT podpore kaže na dejstvo, da ne obstaja veliko rešitev v produkciji, ki bi celovito podpirale poslovne procese z lastnostmi iz omenjene tretje skupine.

2.3 Procesna usmerjenost

Z razvojem IKT je skozi čas prišlo do uvedbe IT podpore v praktično vseh poslovnih funkcijah. Ker so bile poslovne funkcije v združbah praviloma precej samostojne enote, je bila IT podpora posledično razdeljena in heterogena. Avtomatična obdelava podatkov in posamezne aplikacije znotraj poslovnih funkcij so sicer predstavljale dobro podporo za opravila posamezne poslovne funkcije, vendar je bila bistvena pomanjkljivost takšne porazdeljene in heterogene podpore, da ne podpira prehoda informacij iz ene poslovne funkcije v drugo. Zaradi ločenosti in neenotne podpore poslovanja je prihajalo do problemov, kot so redundanca podatkov, neusklajenost in nekonsistentnost podatkovnih zbirk, časovna neusklajenost, in nenavsezadnje dvojno delo za zaposlene. Zaradi vseh naštetih

problemov in zaradi vse bolj prevladujočega mnenja organizacijske znanosti o pomembnosti dobro definiranih poslovnih procesov se je v IT pozornost usmerila na PAIS informacijske sisteme.

Zavedati se moramo, da je bila za implementacijo PAIS potrebna evolucija strojne in mrežne opreme [4]. Pred 90' prejšnjega stoletja namreč tehnologija še ni omogočala ekonomične implementacije podpore za poslovne procese in je bila bolj smotrna implementacija posameznih aplikacij za podporo poslovnih funkcij. Poudarek je bil podan na pretoku podatkov in informacij znotraj poslovnih funkcij brez posebne podpore poslovnih procesov. Pogosto se je dogajalo, da je bil za posamezno poslovno funkcijo implementiran manjši integriran informacijski sistem, v katerem so uporabljali samostojno podatkovno infrastrukturo. Pri prehodu izvajanja poslovnih procesov med poslovnimi funkcijami s takšno podporo je običajno prišlo do velikega števila problemov zaradi neenotnosti podatkovnih modelov. Za načrtovanje in implementacijo IT podpore so bile potrebne različne metode podatkovnega modeliranja. Glavno vlogo v IT podpori je imel torej podatek in lahko rečemo, da je bila IT podpora *podatkovno usmerjena* [4].

Uvedba PAIS sistemov predstavlja preobrat v mišljenju v IT področju. PAIS sistemi povzročijo premik od usmerjenosti na podatke v usmerjenost na poslovne procese. Takšni sistemi so *procesno usmerjeni* in omogočajo bistveno boljšo podporo poslovnih procesov.

Zaradi gospodarskih kriz in zaostrene konkurence se v 90' letih še dodatno okrepi miselnost, da je treba poslovanje združb za doseganje čim boljših poslovnih učinkov dobro podpreti in uvajanje PAIS sistemov je bila najboljša izbira. Organizacijska znanost je namreč uprave združb prepričala, da je treba za uspešno poslovanje dobro definirati poslovne procese v združbi. Dobro definirani poslovni procesi namreč omogočajo:

Formalizacijo znanja Dejstvo je, da je v velikih združbah zaradi velikega števila udeležencev delovanje posameznikov tako razpršeno, da je težko ohraniti dober pregled nad celotnim dogajanjem v združbi. Znanje o postopkih je običajno implicitno shranjeno pri posameznikih. Ta problem je zelo pereč v poslovnih sistemih, kjer morajo vodstveni kadri za dobro vodenje poslovnih procesov le te dobro poznati. Ker se znanje o poslovnih procesih brez formalizacije prenaša iz generacije v generacijo implicitno, je takšno znanje običajno skrito in ni dostopno vsem, ki ga za uspešno izvajanje poslovnih procesov potrebujejo.

Odstranitev odvečnih aktivnosti v poslovnem procesu V praksi se izkaže, da pri nesistematičnem uvajanju poslovnih procesov pride do odvečnih in nepotrebnih aktivnosti v poslovnih procesih. Zaradi hudih konkurenčnih pritiskov morajo sodobni poslovni sistemi iz poslovnih procesov odstraniti vse odvečne aktivnosti, ki v celotni sliki ne pripomorejo pri ustvarjanju dodane vrednosti. Poslovni procesi so zreducirani na tiste aktivnosti, ki so nujno potrebne za ustvarjanje končnega poslovnega učinka. Šele z uvedbo poslovnih procesov, ko se vzpostavi dober pregled nad vsemi poslovnimi procesi v poslovnem sistemu, lahko s t.i. *celostno analizo* (angl. big picture analysis) [7] vidimo, katere so nepotrebne aktivnosti v procesih in na kakšen način jih lahko odstranimo.

Optimizacija Optimizacija poslovnih procesov se začne že pri odstranjevanju odvečnih aktivnosti iz poslovnih procesih, obsega pa še več možnosti za doseganje bolj optimalnega poteka poslovnih procesov. Z dobro definiranimi poslovnimi procesi lahko določimo optimalen vrstni red izvajanja aktivnosti, ki bo pripeljal do čim manjših zastojev in zasedanj virov. Na ta način se skrajša čas izvajanja poslovnih procesov, kar vodi v nižje stroške in višjo konkurenčnost. Optimizacija se dogaja tako znotraj poslovnih procesov na stopnji aktivnosti kot na nivoju poslovnih procesov z usklajevanjem vseh poslovnih procesov v poslovnem sistemu. Dobro definirani poslovni procesi nam predvsem omogočijo, da se odpravijo težave zaradi (običajne) izoliranosti poslovnih funkcij, saj poslovni proces poteka horizontalno (glej Slika 2.1) in tako združuje aktivnosti iz različnih poslovnih funkcij.

Učenje Kot smo že omenili, uvedba poslovnih procesov v združbi omogoča formalizacijo znanja, ki ga ima združba. Z učenjem pa lahko to znanje izboljšamo in skozi postopke optimizacije in analize delovanja poslovnih procesov le te še izboljšamo. Pri funkcijski delitvi združb je bilo takšno učenje zaradi izoliranosti posamezni poslovnih funkcij oteženo. Običajno je bilo znanje skoncentrirano znotraj posamezne poslovne funkcije, ni pa prehajalo med vsemi poslovnimi funkcijami v združbi. Poslovni procesi zaradi prehoda med poslovnimi funkcijami to pomanjkljivost odpravljajo. Pri uvajanju poslovnih procesov se torej združuje znanje iz celotne združbe, kar omogoča nadaljnje učenje in ustrezne ukrepe za izboljšavo poslovnih procesov.

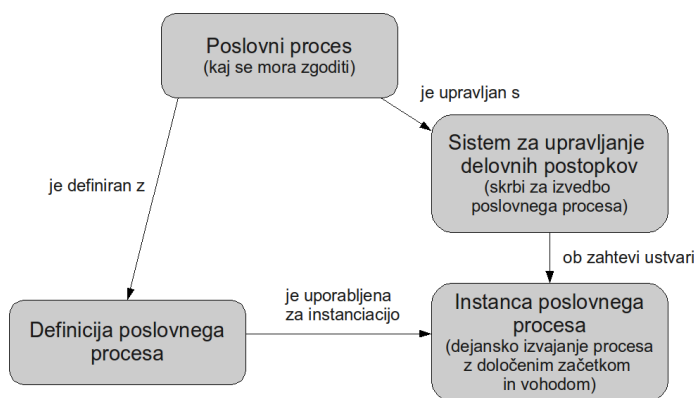
Boljšo podporo poslovanju Jedro poslovanja so poslovni procesi. Poslovni procesi so tisti, ki pripeljejo do poslovnega učinka, zato je treba za uspešno poslovanje podpreti celoten potek delovanja poslovnih procesov. Podpora posameznih poslovnih funkcij ne omogoča celovitega pregleda nad izvajanjem poslovnih procesov. Zato je sodobna miselnost na IT področju prišla do tega, da če hočemo podpreti čim večji del poslovanja, je treba podpreti poslovne procese in ne samo posameznih poslovnih funkcij.

Zaradi vseh prednosti, ki jih prinaša procesna usmerjenost lahko rečemo, da je danes glavni poudarek pri implementaciji IT podpore podan na poslovnih procesih. V kombinaciji z organizacijskimi znanostmi se je uveljavil pristop upravljanja poslovnih procesov oz. BPM [8, 12]. V poslovnih sistemih se je začel uvajati procesno usmerjen pristop, ta miselnost pa se je uporabila tudi v IT področju, kar je pripeljalo do PAIS informacijskih sistemov. Dejstvo namreč je, da samo informatizacija slabih poslovnih procesov še ne pripelje do boljših poslovnih učinkov. Še preden se gre v informatizacijo je treba poslovne procese dobro definirati in jih ustrezno spremeniti [11]. V zadnjih dveh desetletjih, ko je področje BPM in miselnost o procesni usmerjenosti dozorela, so se začeli izvajati BPR projekti z namenom boljših poslovnih procesov in dobre IT podpore. Področju izboljšav, optimizacije in avtomatizacije poslovnih procesov rečemo BPI področje.

2.4 Delovni postopek

S poslovnim procesom se definira zaporedje, v katerem morajo biti izvedene posamezne aktivnosti za doseg poslovnega učinka. Za podporo izvajanja takšnega zaporedja se uporablja t.i. *delovne postopke* (angl. workflow) in *sisteme za upravljanje delovnih postopkov* (angl. Workflow management system). Delovni postopek je odgovoren za usmerjanje odgovornosti in toka dela k pravih ljudem oz. virom [3, 16]. V delovnem postopku so določene aktivnosti, poti, vloge in pravila, kot je definirano v poslovnem procesu. Gre torej za izvajalni tok poslovnega procesa. Po [4] lahko delovni postopek označimo kot avtomatizirano oz. računalniško podprto izvajanje poslovnega procesa.

Za standardizacijo in usklajenost na področju delovnih postopkov skrbi organizacija WfMC [39], ki je poenotila pojme in ponudila določene standarde.



Slika 2.2: Glavne entitete WfMC metamodela [4, 39]

Odgovornost za uvedbo in definicijo poslovnega procesa leži na poslovnih uporabnikih. Običajno poslovni analitiki uporabljajo različne metodologije analize in popisa poslovnih procesov v poslovnem sistemu, rezultat pa je zapisan v obliki kakšne od diagramskih tehnik ali metodologij za definiranje poslovnih procesov¹. Takšne metodologije so poslovno usmerjene in so po eni strani razumljive poslovnim uporabnikom, po drugi strani pa so dovolj formalizirane, da jih lahko uporabijo tudi v področju IT za implementacijo podpore. Vloga IT podpore je učinkovita pretvorba poslovnega procesa v ustrezen delovni postopek, ki ga bo izvajal sistem za upravljanje delovnih postopkov. Rezultat analize poslovnih procesov morajo pretvoriti v formalen jezik za opis poslovnih procesov² in ustrezen model delovnih postopkov. Delovni postopek je torej najbolj rigorozno določena oblika izvajanja poslovnega procesa, ki je primerna za izvajanje z IKT. V računalniško podprtem sistemu za podporo poslovnih procesov je treba zelo podrobno opredeliti vse aktivnosti,

¹Primeri pristopov analiziranja poslovnih procesov in dogajanja v združbi so različne strukturne in objektne tehnike.

²Eden izmed takšnih jezikov je XPDL, ki je ga je standardizirala in predlagala organizacija WfMC [39].

vloge, možne izjeme ter poteke izvajanja in modelirne tehnike, ki si jih bomo ogledali v nadaljevanju, to omogočajo.

Uvedba delovnih postopkov je bil eden izmed prvih poizkusov avtomatizacije poslovnih procesov. V uporabi so bili še preden je organizacijska znanost predlagala prehod na procesno usmerjene sisteme. Takšen pristop avtomatizacije poslovnih procesov pa je bil zaradi premalo zmogljive IKT omejen samo na podporo tistih poslovnih procesov, ki so bili zelo strukturirani in običajno niso prestopali meja organizacije. Običajno je šlo za avtomatizacijo preprostih poslovnih procesov, ki so se pogosto ponavljali in jih je bilo možno avtomatizirati. Avtomatizacija je sicer prinesla velike prihranke pri stroških in odzivnem času poslovanja, vendar je bila običajno omejena na notranje poslovne procese. Danes, ko smo priča razmahu razvoja IKT, nam ta omogoča implementacijo dobre podpore delovnih postopkov za poslovne procese, ki potekajo ne samo preko več poslovnih funkcij oz. funkcionalnih enot združb, ampak tudi med združbami. Ne govorimo več zgolj o avtomatizaciji poslovnih procesov, temveč o celostni podpori poslovnih procesov, kar pripelje v znižanje entropije v združbi in je danes nujno za uspešno upravljanje in vodenje.

2.5 Storitveno usmerjena arhitektura

SOA je danes eden izmed najbolj uveljavljenih načinov gradnje informacijskih sistemov. SOA lahko opredelimo kot množico postopkov in usmeritev za gradnjo informacijskih sistemov, ki temelji na orkestraciji posameznih čimbolj neodvisnih storitev [9].

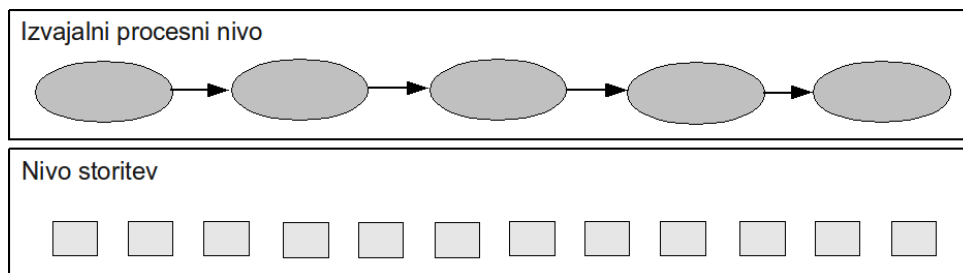
Če vzamemo definicijo poslovnega procesa kot zaporedje aktivnosti, nam pregled SOA arhitekturnega stila hitro pokaže, da SOA k podpori poslovnih procesov pristopa na zelo "naraven" način. Z ustrezno orkestracijo storitev med seboj poveže tiste, ki za konkreten poslovni proces opravijo ustrezne aktivnosti. Orkestracija je torej izvedba kontrolnega toka med posameznimi storitvami, ki za poslovni proces predstavljajo aktivnosti.

Na izvajalnem nivoju so definirane aktivnosti poslovnega procesa, ki jih izvedemo z ustrezno orkestracijo storitev. Na Sliki 2.3 je prikazan samo en nivo abstrakcije storitev, v praksi pa se običajno storitve abstrahira od poslovnega do tehničnega nivoja storitev.

Glavni jezik za modeliranje poslovnih procesov s končnim ciljem implementacije SOA je BPMN jezik, ki ga bomo podrobneje predstavili v Poglavju 5.3. Z BPMN poslovne procese zmodeliramo, nato pa dobljene procesne modele preslikamo v izvajalni jezik BPEL, ki predstavlja navodila za izvajanje na procesnih strežnikih.

Eden izmed ciljev SOA je hitro in učinkovito obvladovanje sprememb v poslovnih procesih. Ker je poslovni proces v SOA podprt z orkestracijo ustreznih storitev, lahko v primeru spremembe poslovnega procesa spremenimo samo izvajalni procesni nivo in prilagodimo orkestracijo. Storitve, ki naj bi bile čimbolj atomarne, samostojne in neodvisne, ostanejo enake. Na ta način se da dobro obravnavati dinamiko, ki nastane zaradi zahtev od zunaj.

Še vedno pa je potrebno tudi v SOA v fazi orkestracije storitev vnaprej določiti vrstni red klicanja storitev. V primeru podpore dinamičnih poslovnih procesov bi bila orkestracija storitev zelo zapletena in podobno kot pri izdelavi delovnih postopkov za dinamične



Slika 2.3: SOA izvajalni nivo in storitve

poslovne procese, neučinkovita.

2.6 Dinamika poslovnih procesov

2.6.1 Zunanje spremembe poslovnih procesov

Zaradi hitrega razvoja IKT, prevladujočega elektronskega poslovanja in nasploh globalizacije trgov, se morajo današnje združbe hitro odzivati na spremembe. Močna konkurenca na praktično katerem koli trgu sili podjetja v hiter odziv na zahtevo stranke. Če je bilo v preteklosti tako, da se je morala zaradi omejene ponudbe prilagajati stranka, je dandanes popolnoma obratno. Ker ima stranka različne potrebe in želje, prihaja do pogostih sprememb zahtevanega asortimenta poslovnih učinkov združb. Če se na spremembe poslovni sistem ne odzove dovolj hitro, bo stranka na globalnem in zasičenem trgu hitro našla drugega ponudnika. Spremembe za poslovni sistem pa pomenijo spremembo poslovnih procesov.

Ko je prevladala miselnost o procesno usmerjenih poslovnih sistemih in je prišlo do implementacije podpore poslovnih procesov je bila večina pozornosti IT področja usmerjena zgolj na implementacijo podpore. Črpali so znanje in navodila iz organizacijske znanosti in počasi gradili različne sisteme za upravljanje delovnih postopkov ali integrirane informacijske sisteme. Običajno so bili to kompleksni produkti, ki so za vzpostavitev terjali veliko časa, denarja in energije. Ker se je implementirala podpora za jasno definirane poslovne procese, ni bilo veliko pozornosti usmerjene na možnost naknadnega spreminjanja poslovnih procesov in kako se bo to odražalo v podpori. Danes je torej realno stanje takšno, da so produkti, ki ponujajo podporo poslovnim procesom, sicer tehnično zelo dovršeni in omogočajo velike pohitritve in prihranke pri izvajanju poslovnih procesov, vendar ne omogočajo hitrega prilagajanja na spremembe poslovnih sistemov.

Današnja podpora poslovnih procesov večinoma temelji na formalnih in zapletenih jezikih, ki opisujejo dogajanje v poslovnih procesih, infrastruktura pa z IKT tehnologijo te jezike pretvori v izvedbo poslovnih procesov. Ko pride do potrebe po spremembi poslovnega procesa, se vodstvo pogosto preprosto ne odloči za takšen projekt, ker bi trajalo preveč časa in energije, da bi z obstoječo implementacijo takšno prilagoditev izvedli.

Običajno prilagoditev in sprememba poslovnih procesov sovpadata z večjimi preno-

vami in preureditvami IT podpore celotne združbe. Ker smo priča hudim gospodarskim razmeram je jasno, da so takšni večji posegi v IT združbe zaradi velikih stroškov redki. S tega vidika vidimo, da lahko postane IT podpora zaviralni faktor pri delovanju združbe in da trenutno še ne obstajajo dobre rešitve, ki bi omogočale hitre prilagoditve podpore poslovnih procesov.

2.6.2 Notranja dinamika poslovnih procesov

Za potrebe magistrske naloge bomo izraz "dinamika poslovnega procesa" uporabljali v smislu stopnje določenosti poslovnega procesa. Definicija poslovnega procesa (glej Poglavje 2.2) namreč opredeljuje poslovni proces kot zaporedje aktivnosti. Vprašanje je, kako lahko opredelimo poslovni proces, kjer se lahko spreminja vrstni red aktivnosti ali celo množica aktivnosti. Z dinamiko poslovnega procesa bi radi označili stopnjo določenosti vseh elementov poslovnega procesa. Gre torej za tri vidike notranje dinamike procesa:

- Stopnja določenosti zaporedja aktivnosti;
- Stopnja določenost vrste aktivnosti;
- Stopnja določenosti vlog in akterjev poslovnega procesa.

Zavedati se moramo, da v praksi poslovni procesi niso vedno določeni vnaprej. Realna slika kaže, da je dovršen del poslovanja dinamičen v smislu nedeterminističnosti zaporedja izvajanja aktivnosti. Z vsako instanco poslovnega procesa se lahko vidiki notranje dinamike poslovnega procesa spremenijo. Bolj kot je proces dinamičen, težje ga je učinkovito podpreti, saj mora sistem za upravljanje delovnih postopkov, ki izvaja poslovni proces (glej Poglavje 2.4), upoštevati veliko možnih poti in izjem v poslovnem procesu. Najlažje je implementirati podporo za statičen poslovni proces, kjer je točno definirano, kako se bo proces odvijal.

Prva delitev po [4, 33], ki razvrsti delovne procese po stopnji njihove dinamike, je naslednja:

Produksijski poslovni proces Produkcijski poslovni proces bi po delitvi iz Poglavja 2.2 spadal med ključne poslovne procese. Takšni poslovni procesi so običajno dobro definirani, zaporedje aktivnosti je določeno in nedvoumno. To izhaja predvsem iz tehnične narave izdelave nekega končnega izdelka ali storitve, kjer je pomemben vrstni red aktivnosti, kako to dosežemo. V tipičnem izdelovalnem procesu³ ne moremo npr. sestaviti dveh komponent, če jih nismo predhodno izdelali ali dobavili. V tem smislu je vrstni red aktivnosti vnaprej določen in takšen poslovni proces ne vsebuje veliko notranje dinamike. Takšno vrsto poslovnih procesov je najlažje podpreti, saj ne vsebujejo veliko kompleksnosti, ki izvira iz dinamike procesa. Podpora za takšne procese je v sodobnem IT že raziskana in dosežena. V ta namen obstaja veliko

³Z izrazom *izdelovalni proces* mislimo na proces izdelave končnega izdelka, npr. izdelava avtomobila, tehničnega aparata, itn.

programskih izdelkov, ki dobro podpirajo takšne poslovne procese in so uporabljeni in preizkušeni v produkciji.

Administrativen poslovni proces Ti procesi bi po delitvi iz Poglavja 2.2 spadali med podporne poslovne procese. Podporni poslovni procesi so običajno že bolj kompleksni, kar se tiče možnosti izvajanja. Vendar lahko rečemo, da so manjše spremembe, na katere naletimo v administrativnih procesih, precej omejene in predvidljive. Skozi čas lahko ali prilagodimo delovni postopek spremenjenemu administrativnemu procesu, ali pa v delovni postopek vnesemo dodatno vejitev, ki takšno spremembo podpre. Sicer gre za povečevanje kompleksnosti delovnega postopka, vendar do obvladljive mere, ki omogoča dovolj preprosto in standardno implementacijo.

Ad-hoc poslovni proces Ad-hoc poslovni procesi so poslovni procesi, v katerih ni vnaprej definirane vzorca ali zaporedja aktivnosti, ki se morajo izvesti za doseganje cilja poslovnega procesa. Težko je napovedati tok informacij. Povezave med udeleženci v poslovnem procesu⁴ se ustvarjajo v vsakem trenutku posebej na navidez stohastičen način brez vnaprej definirane vzorca. Z običajnim delovnim postopkom bi takšen poslovni proces težko podprli. Delovni postopki sicer omogočajo uporabo veliko različnih poti skozi poslovni proces, ki se v točkah vejitve izbirajo glede na trenutne pogoje, vendar bi bila pri podpori ad-hoc poslovnega procesa takšna podpora zaradi kombinatorične zahtevnosti praktično neuporabna.

Podana razdelitev deli procese po dveh lastnostih - po *kompleksnosti* in *predvidljivosti*. Ker se hočemo pri obravnavi dinamike poslovnih procesov omejiti samo na predvidljivost poslovnih procesov, ki je neposredno odvisna od dinamike procesa, je naslednja delitev mogoče še bolj primerna [4]:

Neokvirjen poslovni proces Če poslovni proces ne temelji na nobenem modelu, lahko rečemo, da nima nobenega okvira in je popolnoma nedoločen. Takšne poslovne procese imenujemo tudi *popolnoma dinamični poslovni procesi*. Nimajo nobene notranje strukture, aktivnosti se izvajajo v vsakič posebej določenem vrstnem redu, vloge pa so določene šele ob izvedbi aktivnosti, ki določeno vlogo potrebuje. Edina podpora za takšne poslovne procese so različne implementacije informacijske podpore za skupinsko delo (angl. groupware), ki pa poslovnega procesa ne izvajajo, temveč omogočajo zgolj beleženje poteka procesa in olajšajo komunikacijo med udeleženci procesa.

Ad-hoc okvirjen poslovni proces Ad-hoc okvirjen poslovni proces je proces, ki je sicer apriori definiran in ima določeno šibko strukturo, ki ni povsem statična in se lahko tekom izvajanja spreminja. Po pričujoči delitvi poslovnih procesov so takšni procesi izvedeni malokrat, pogosto samo enkrat, nato pa je njihov model zavržen. Vsakič ali skoraj vsakič, ko se instancira nov poslovni proces, se izdelava nov procesni model. Podpora za takšne poslovne procese mora upoštevati kratkoročnost in spremenljivost poslovnih procesov. S to tematiko se ukvarja magistrska naloga.

⁴Kot udeleženci so mišljeni nosilci vlog v poslovnem procesu - ljudje in različni viri.

Šibko okvirjen poslovni proces Šibko okvirjeni poslovni procesi temeljijo na procesnem modelu, ki mora upoštevati dinamiko poslovnega procesa. To pomeni, da vsebuje veliko alternativnih poti procesa, kar povečuje dinamično kompleksnost zapisa procesa. Opis poslovnega procesa mora vsebovati dovolj natančno opredeljene omejitve, ki določajo potek aktivnosti poslovnega procesa. Ob instanciaciji šibko okvirjenega procesa je vrstni red izvajanja aktivnosti določen, vendar se lahko spremeni v okviru alternativnih poti skozi poslovni proces oz. v tolikšni meri, da sovпада z vsemi podanimi omejitvami procesa. Podpore za takšen tip poslovnih procesov je mogoče implementirati z običajnimi pristopi podpore poslovnih procesov. Odločiti pa se je treba, do kakšne mere se podpre dinamiko procesa in kako se najbolj učinkovito zapiše omejitve, ki jih mora proces med izvajanjem izpolnjevati.

Krepko okvirjen poslovni proces Krepko okvirjen poslovni proces ima natančno določen potek. Ima notranjo strukturo, določene akterje, aktivnosti in vrstni red izvajanja aktivnosti. Procesni model se izdelava predhodno s poljubno metodologijo procesnega razvoja. Primer krepko okvirjenih procesov so ključni poslovni procesi v različnih proizvodnih poslovnih sistemih, kjer je proces izdelave končnih izdelkov natančno načrtovan, projektiran in izveden. Za podporo izvajanja takšnih procesov se uporabi sistem za upravljanje delovnih postopkov (primer Staffware [36, 4]).

Iz napisanega sledi, da lahko notranjo dinamiko poslovnih procesov opredelimo na različnih nivojih podrobnosti in z različnimi robnimi pogoji. Privzemamo pa, da neokvirjenih poslovnih procesov ne moremo obravnavati, ker že v osnovi ne izpolnjujejo definicije ad-hoc procesa.

V nadaljevanju magistrske naloge se bomo ukvarjali večinoma s kompleksnimi administrativnimi in ad-hoc poslovnimi procesi iz prve delitve ter z ad-hoc in šibko okvirjenimi poslovnimi procesi iz druge delitve.

2.6.3 Medorganizacijski procesi

Poslovne procese glede na področje njihovega delovanja delimo na:

Notranje poslovne procese Ti poslovni procesi delujejo znotraj združbe. To pomeni, da se vse aktivnosti izvedejo znotraj združbe in imamo nad njimi pregled in kontrolo. Vloge v poslovnem procesu zavzamejo akterji znotraj združbe, prav tako se ljudje in drugi viri, ki jih uporablja poslovni proces, nahajajo znotraj združbe.

Medorganizacijske poslovne procese Medorganizacijski procesi se odvijajo tako znotraj združbe kot v zunanjih združbah. Neka aktivnost znotraj združbe lahko aktivira aktivnost, ki je izvedena v drugi združbi. Poslovni proces se tako razteza čez več združb.

Z razvojem IKT je v uporabi vse več medorganizacijskih poslovnih procesov. Ti so lahko v osnovi tipa B2B ali B2C, z razvojem vse bolj kompleksnih in raznovrstnih komunikacijskih kanalov pa se tipi medorganizacijskih poslovnih procesov rojevajo iz dneva v dan.

Takšni poslovni procesi so dinamični že v svoji osnovi, saj aktivnosti, ki se izvajajo izven združbe, ne moremo kontrolirati in so zaradi tega za nas neznanka. Običajen pristop k reševanju tega problema je ta, da se definira točno določene vmesnike in točke, kjer se izvajanje poslovnega procesa preseli v drugo združbo. S točno definiranimi vhodi in izhodi v teh točkah se deloma odpravi negotovost zaradi delovanja aktivnosti izven naše pristojnosti. Medorganizacijski procesi niso del problematike dinamike poslovnih procesov s stališča določenosti in predvidljivosti izvajanja poslovnih procesov. Zavedati se moramo, da izvajanje izven združbe prinese s seboj določeno stopnjo negotovosti, saj ne moremo natančno nadzorovati poteka izvajanja takšnih procesov.

2.7 Motivacija za obvladovanje dinamike v poslovnih procesih

Skozi poglavje smo pokazali, da današnja podpora poslovnih procesov ne obravnava problema dinamike poslovnih procesov v dovoljšnji meri. Glede na to, da je dinamika prisotna v velikem deležu poslovnih procesov, mora IT področje prilagoditi uveljavljen "statičen" pristop podpore poslovnih procesov, ki predvideva statično in točno določeno obnašanje le teh. Ker je softverski inženiring in področje IT že zrela panoga in ker je IKT dovolj razvita, da lahko z njo podpremo bistveno bolj kompleksne poslovne procese, kot so to lahko mogoče storili v preteklosti, je čas, da začnemo v IT poslovne procese jemati takšne, kot so. Poslovne procese moramo obravnavati brez pretiranega poenostavljanja in uporabe posplošitev glede njihove dinamike. Pogosto se namreč dinamične poslovne procese poenostavi do te mere, da postanejo dovolj "statični" in jih lahko nato podpremo z obstoječimi BPM pristopi, ob tem pa se vidik dinamike poslovnih procesov in njihovo pravo obnašanje zanemari.

V magistrski nalogi bi radi pokazali enega izmed poizkusov uvedbe takšnega pristopa, kjer poizkušamo obravnavati tudi dinamičen vidik poslovnih procesov.

Poglavje 3

Raziskovalni pristop

Sledi kratek pregled pristopa pri izdelavi magistrske naloge. Na kratko je predstavljen način, kako smo prišli do rezultatov in kakšne metode dela smo pri tem uporabili. Podan je tudi pregled knjig in člankov, na katere smo se opirali pri izdelavi magistrske naloge.

3.1 Metode dela

Izdelava magistrskega dela je temeljila na reševanju realnega problema iz prakse. V prvi fazi je bil izdelan predlog poslovnega procesa. Ko je bil predlog predstavljen naročniku in se je strinjal s konceptualnimi rešitvami, smo predlog podrobno analizirali in izdelali prototip, s katerim smo prikazali možnosti delovanja predlagane rešitve. Nadaljnje delo je vključevalo formalizacijo problema, pregled področja in izdelavo predloga novega pristopa k opisu dinamike v poslovnih procesih. V ta namen smo izdelali prilagojeno diagramsko tehniko in z njo opisali poslovni proces naročnika.

3.1.1 Predlog procesa

Kot je opisano že v Poglavju 1.2.1, smo morali v širšem kontekstu izdelave celovite rešitve za detekcijo in obravnavo goljufij v zavarovalništvu izdelati predlog procesa razreševanja sumljivih primerov. Proces smo predlagali na podlagi analize naročnikovih zahtev in potreb ter pri tem upoštevali širši kontekst celotnega pristopa obravnave goljufij v zavarovalništvu. Pri izdelavi predloga procesa smo upoštevali tudi strokovno mnenje eksperta s področja prava, ki dobro pozna postopke razreševanja škodnih primerov v pravnem sistemu. Po nekaj iteracijah usklajevanja s stranko smo prišli do končnega predloga procesa razreševanja, na katerem temelji naše nadaljnje delo.

3.1.2 Prototip

Izdelava prototipa je vključevala zaslonske maske, ki so uporabnika vodile skozi proces razreševanja. Prototip je služil za lažjo komunikacijo z naročnikom in praktičen prikaz

delovanja predlagane rešitve. Z delom na prototipu smo morali že sami rešiti precej konceptualnih vprašanj, predvsem pa smo od naročnika dobili povratno informacijo. Prikazali smo konceptualen potek procesa razreševanja sumljivih primerov z možnimi koraki in vlogami. Naročnik je lahko videl, na kakšen način bo proces deloval, kakšne so možne aktivnosti in ali proces ustreza njegovim predstavam in potrebam. Tehnika prototipiziranja se je izkazala za zelo učinkovito kar se tiče uspešne komunikacije z uporabnikom, saj smo na ta način dobili veliko informacij in smernic za dodelavo procesa.

3.1.3 Definicija diagramске tehnike

Po izdelanem predlogu procesa razreševanja smo se lotili analize. Zaradi pomanjkanja uveljavljenih in splošnih definicij za opis dinamike v poslovnih procesih je bilo treba definirati novo definicijo za dinamične poslovne procese. Prav tako je bilo za uspešno modeliranje takšnih procesov treba uporabiti drugačno, manj konvencionalno percepcijo poslovnih procesov in njihovega odvijanja v združbah. Po temeljitem pregledu teorije BPM področja smo iterativno razvijali koncept modeliranja dinamičnih poslovnih procesov. Izhajali smo torej iz praktičnega primera, ki je služil kot izhodišče za izdelavo teoretičnega pristopa. Ko smo definirali vse potrebne gradnike nove diagramске tehnike, smo do konca zmodelirali proces razreševanja in tako preverili pravilnost in smotrnost vseh gradnikov.

3.2 Pregled temeljnih virov

Pri izdelavi magistrske naloge smo izhajali iz realnega problema v gospodarstvu. Za ustrezno rešitev smo morali zato najprej podrobno pregledati obstoječe pristope in metodologije za podporo poslovnih procesov. Nato smo pregledali številne članke, ki obravnavajo ad-hoc procese in nove načine modeliranja ter diagramске tehnike.

Podpora in upravljanje poslovnih procesov BPM Področje BPM je razvito in uveljavljeno področje, na katerem dela veliko število raziskovalcev. Zato ni bilo težav dobiti literature na to temo. Opirali smo se predvsem na knjige in članke [9, 4, 8, 12, 22, 25, 32].

Pristopi za izboljšavo poslovnih procesov BPI Zaradi trenutne popularnosti projektov prenove in izboljšave poslovnih procesov v združbah je tudi s tega področja na voljo veliko literature. Osredotočili smo se na [2, 17, 7, 11, 3].

Pregled obstoječih diagramskih tehnik Diagramske tehnike so najboljše opisane na spletnih straneh avtorjev. Glavne informacije za jezik UML smo črpali iz [37, 38]. Za širši kontekst pa smo uporabili knjige [16, 5, 6].

Pregled pristopov za podporo ad-hoc procesov Obravnava dinamike poslovnih procesov je sistematično zajeta v številnih člankih. Članki na temo formalnih jezikov

za modeliranje poslovnih procesov so [31, 35, 21], sledijo članki na temo agentnih sistemov za samostojno planiranje dinamičnih procesov [19, 23, 24, 27]. Nekaj splošnih napotkov za ustvarjanje novega pristopa in diagramске tehnike smo prebrali v [20, 34, 18, 29].

Navedeni temeljni viri so podrobneje razčlenjeni in opisani v nadaljevanju magistrske naloge.

Poglavje 4

Ad-hoc poslovni procesi

Za potrebe obravnave dinamike poslovnih procesov (dinamiko poslovnih procesov smo definirali v Poglavju 2.6) je treba natančno definirati pojem ad-hoc poslovnih procesov. V poglavju je podan pregled poizkusov definicij v obstoječi literaturi in naša definicija, ki združuje in povzema obstoječe definicije. V nadaljnjem besedilu bomo besedno zvezo *ad-hoc poslovni proces* zaradi določenega konteksta magistrske naloge skrajšali v besedno zvezo *ad-hoc proces*.

4.1 Definicija

Pregled področja upravljanja in modeliranja poslovnih procesov je hitro pokazal pomanjkanje pozornosti obravnavi dinamike poslovnih procesov. Zato tudi le stežka najdemo kakšno uveljavljeno definicijo dinamičnosti poslovnih procesov in ad-hoc procesov. Dinamiko poslovnih procesov smo že opisali v Poglavju 2.6. Z definicijo ad-hoc procesa bi radi označili poslovne procese takšne vrste, ki ustrezajo določeni stopnji določenosti, predvidljivosti in strukturiranosti.

Ena izmed definicij ad-hoc procesov po [4] je:

Definicija 4.1.1. Ad-hoc proces je proces, kjer ne obstaja vnaprej (apriori) definiran vzorec premikov informacij in opravil med ljudmi.

Definicija takšnih poslovnih procesov sledi delitvama poslovnih procesov, ki smo ju podali v Poglavju 2.6.2. Gre torej za poslovne procese, ki se po stopnji dinamike uvrščajo med ad-hoc okvirjene poslovne procese. Vprašanje tukaj je, do katere mere takšne nedoločenosti poslovnega procesa poslovni proces sploh še ustreza osnovni definiciji poslovnega procesa (glej Poglavje 2.2). Če zaporedje aktivnosti sploh ni določeno in je popolnoma sporadično, ali je to sploh še poslovni proces ali je to skupek naključno opravljenih aktivnosti? Slednje poslovne procese običajno imenujemo *popolnoma dinamični poslovni procesi* in jih v nadaljnjem besedilu ne bomo obravnavali, ker praktično ne spadajo med klasične poslovne procese.

Splošna definicija za ad-hoc procese, ki združuje definicijo dinamičnosti, stopnjo določenosti in strukturiranosti poslovnega procesa, ter še vedno ustreza splošni definiciji

poslovnega procesa, bi bila naslednja:

Definicija 4.1.2. Ad-hoc proces je v nekem trenutku instanciran poslovni proces, ko vrstni red vsaj šibko določenega nabora aktivnosti še ni določen. Vrstni red aktivnosti se določa sproti v odvisnosti od časa in izpolnjevanja pogojev v določenem trenutku.

Poslovni procesi, ki ustrezajo Definiciji 4.1.2, ustrezajo tudi Definiciji 2.2.1, Definiciji 2.2.2 in Definiciji 2.2.3, še vedno pa smo s takšno definicijo zajeli bistvo notranje dinamike ad-hoc procesov.

Pri običajnih poslovnih procesih se osredotočamo na zaporedje aktivnosti v procesu. Pri ad-hoc poslovnih procesih tega zaporedja nimamo definiranega vnaprej, temveč se določa skozi čas. Ob izhodu iz vsake aktivnosti je izbor naslednje aktivnosti poljuben oz. definiran s pogoji okolja. Zaporedje aktivnosti se torej tvori tekom samega izvajanja poslovnega procesa, zato rečemo, da je notranja dinamika takšnega procesa visoka (glej Poglavje 2.6.2).

4.2 Obravnava ad-hoc procesov

Ad-hoc procesi omogočajo opis dinamike poslovnih procesov, ki je z običajnimi pristopi obravnave poslovnih procesov ne moremo zadovoljivo obravnavati. Običajno se ad-hoc procesov v združbah ni temeljito obravnavalo zaradi prevladujočega mnenja, da se takšnih procesov niti ne da dobro podpreti in da spadajo v kategorijo popolnoma dinamičnih procesov. Zaradi razvoja IKT in vse več dinamičnih procesov v poslovanju združb, je danes uvedba obravnave ad-hoc procesov nujna.

Menimo, da obravnava ad-hoc procesov kljub njihovi nedoločenosti zniža entropijo v dogajanju združb in olajša upravljanje in vodenje dela. Z ad-hoc procesi uokvirimo tiste poslovne procese, ki jih prej nismo obravnavali in so zaradi nepoznavanja njihovega stanja bistveno povečali tveganja v poslovanju. Ad-hoc poslovni procesi uokvirijo slabo predvidljive poslovne procese v določene okvirje, pri katerih poznamo omejitve in vmesne delne cilje. Z ustrezno obravnavo ad-hoc procesov je možno uvesti tudi učinkovito podporo takšnih procesov.

Obravnava poslovnih procesov omogoča njihovo izboljšavo in optimizacijo. Na ta način odpravimo slabosti v poslovnih procesih in izboljšamo poslovanje. Poslovni procesi, katerih se ne obravnava na ustrezen način, predstavljajo šibko točko poslovanja združbe in znižujejo možnosti za njeno preživetje. Naše mnenje je, da to še posebej velja za ad-hoc procese. Ti so že sami po sebi vir negotovosti in posledično predstavljajo visoko tveganje v poslovanju. Lahko rečemo, da je obravnava ad-hoc procesov najšibkejši člen na področju obravnave vseh poslovnih procesov in zato je treba temu področju nameniti še posebej veliko pozornosti.

Poglavje 5

Pregled modeliranja poslovnih procesov

Ker ob pregledu literature nismo našli modelirne tehnike, ki bi bila namenjena posebej modeliranju ad-hoc procesov, smo z namenom razvoja prilagojene modelirne tehnike za ad-hoc procese pregledali obstoječe tehnike dokumentiranja poslovnih procesov. V poglavju so predstavljeni osnovni koncepti modeliranja poslovnih procesov s poudarkom na tistih lastnostih, ki omogočajo zajem dinamike poslovnih procesov. Predstavljene so najbolj razširjene diagramске tehnike, iz katerih smo črpali ideje za diagramsko tehniko za ad-hoc procese.

5.1 Modeliranje poslovnih procesov

Modeliranje poslovnih procesov (kratica BPM) lahko definiramo kot proces formaliziranja in predstavitve poslovnih procesov v neki združbi oz. poslovnemu sistemu. Rezultat takšnega modeliranja so različni modeli, ki so uporabljeni za analizo dogajanja v poslovnem sistemu.

Modeliranje poslovnih procesov je v prvi vrsti aktivnost, ki se dogaja na strateškem nivoju upravljanja združbe [7]. Rezultat modeliranja je namenjen vodstvenemu kadru z namenom boljšega upravljanja in organiziranja dela. Gre torej za poslovne uporabnike, ki dogajanje v združbi predstavijo v obliki procesnih modelov.

Po drugi strani pa je modeliranje poslovnih procesov namenjeno tudi IT podpori. Uspešna implementacija IT podpore poslovnih procesov je neposredno odvisna od dobrih procesnih modelov, ki predstavljajo osnovo za uspešno analizo in izdelavo specifikacij informacijskega sistema.

Za modeliranje poslovnih procesov uporabljamo različne dokumentacijske tehnike [7]. Namen dokumentacijskih tehnik je zajem in podajanje opisa poslovnih procesov. Za potrebe opisa poslovnih procesov se običajno uporablja različne vizualne jezike oz. diagramске tehnike. Izbor diagramске tehnike je odvisen od t.i. dokumentacijskih faktorjev [7] poslovnih procesov, ki jih želimo dokumentirati. To so lastnosti procesov, ki bi jih radi

poudarili in prikazali v procesnem modelu. Najbolj pomembni faktorji so:

Narava procesov Pri izboru diagramске tehnike je treba vedeti, ali bomo modelirali ključne ali podporne poslovne procese (glej delitev poslovnih procesov v Poglavju 2.2). Ključni poslovni procesi so običajno proizvodni poslovni procesi, ki so zaradi proizvodne narave dobro strukturirani in v katerih je zelo pomemben vrstni red in optimizacija aktivnosti. Po drugi strani so podporni poslovni procesi, običajno so to administrativni procesi, slabše definirani in vsebujejo več dinamike. Pri takšnih procesih ni bistven prikaz zaporedja aktivnosti, temveč prikaz vlog in pretoka informacij.

Časovna komponenta Za določene poslovne procese je zelo pomembna časovna dimenzija. Pri časovno kritičnih procesih je treba poudariti začetek in konec izvajanja posameznih aktivnosti procesa. Diagramska tehnika mora dobro prikazati trajanje in časovno sosledje aktivnosti. Faktor za izbor je tudi kako dobro diagramska tehnika prikaže možnosti paralelizacije in kritičnih poti skozi proces.

Pretok dokumentov in informacij Skozi poslovni proces se ustvarjajo, prenašajo in uporabljajo različni dokumenti. Tukaj je treba poudariti, da ne gre nujno za elektronske dokumente, temveč za dokumente kakršnekoli oblike. Še bolj pomemben je pretok informacij, ki so nujno potrebne za ustrezen potek poslovnega procesa. Diagramske tehnike, ki so odvisne od teh faktorjev, bodo osredotočene na podatkovni in dokumentni tok skozi poslovni proces. Poudarjeni bodo različni nosilci informacij.

Geografski faktorji Geografski faktorji izpostavljajo lokacijsko postavitev združbe in prikaz poti, ki jih se jih v poslovnem procesu uporablja. V velikih združbah, ki so sestavljene iz geografsko razpršenih enot, je treba prikazati, kateri poslovni procesi ali kateri deli poslovnih procesov se odvijajo v kateri geografski enoti.

Dekompozicijski faktorji Pri modeliranju poslovnih procesov moramo označiti, katere poslovne funkcije so odgovorne za izvajanje aktivnosti. Diagramske tehnike morajo prikazati odgovornosti za izvajanje posameznih aktivnosti tako s stališča funkcionalne dekompozicije združbe, lahko pa tudi s stališča posameznikov.

Naštevaje dokumentacijskih faktorjev bi lahko še nadaljevali, vendar je najbolj pomembno zavedanje dejstva, da moramo za prikaz posameznih lastnosti poslovnih procesov uporabiti ustrezno modelirno oz. diagramsko tehniko. Bolj kot je diagramska tehnika specifična in se osredotoča na določen vidik poslovnega procesa, manj so prikazani ostali vidiki procesa. Da se sodobni pristopi modeliranja poslovnih procesov ne osredotočajo v veliki meri na prikaz dinamike poslovnega procesa nakazuje dejstvo, da v splošni literaturi, ki opisuje modeliranje poslovnih procesov, skoraj nikjer niso eksplicitno omenjeni dokumentacijski faktorji, ki se tičejo dinamike poslovnih procesov.

Dokumentacijske tehnike so običajno vizualne oz. grafično predstavljene diagramske tehnike. Uporaba vizualnih tehnik se je skozi čas uveljavila zaradi večje izraznosti ter razumljivosti diagramov v primerjavi z običajnimi tekstovnimi dokumentu. Grafično

predstavitev poslovnih procesov si lažje predstavljamo, nosi več informacij in je hkrati uporabna tako za poslovne kot tehnične uporabnike. S podporo sodobnih orodij, ki omogočajo enostavno in hitro risanje diagramov, so diagramске tehnike praktično izpodrinile modele drugih oblik.

Pred uporabo vsake diagramске tehnike se moramo odločiti za nivo podrobnosti poslovnih procesov, ki bi ga radi predstavili. Ko z diagramsko tehniko zajamemo konceptualni nivo poslovnih procesov, torej predstavimo celotne procese brez prikaza podrobnosti, običajno govorimo o konceptualnih procesnih modelih. Ti so namenjeni splošnemu razumevanju poslovnih procesov in omogočajo hiter pregled dogajanja v združbi. Takšne modele pa lahko poljubno razčlenimo in tako dobimo zelo podrobno sliko posameznih procesov ali aktivnosti. Vprašanje je, do katere mere lahko poslovne procese razčlenimo. Običajno velja pravilo, da se modeliranje poslovnih procesov konča pred tisto stopnjo podrobnosti, ki je že primerna za tehnično implementacijo podpore samega procesa.

Modeliranje poslovnih procesov običajno vključuje uporabo več dokumentacijskih tehnik, ki se med seboj dopolnjujejo. Med seboj kombiniramo dokumentacijske tehnike, ki so si čimbolj komplementarne, da lahko na takšen način prikažemo celotno sliko poslovnih procesov. Nekatere diagramске tehnike so bolj primerne za analizo s strani poslovnih uporabnikov, nekatere so bolj primerne za tehnične kadre. Na voljo so različna orodja, ki omogočajo preslikave med modeli¹.

Zelo pomembni aktivnosti modeliranja sta validacija in verifikacija procesnih modelov [11, 26], s katerima preverimo pravilnost in natančnost izdelanih modelov. Z aktivnostjo validacije preverimo, ali procesni model ustreza dejanskemu stanju poslovnih procesov. Treba je zagotoviti popolno konsistenco med procesnimi modeli in dejanskim stanjem v združbi. Skozi verifikacijo pa preverimo, ali so procesni modeli ustrezno grajeni in ali vsebujejo vse potrebne informacije. Posebnih pristopov za validacijo in verifikacijo ni definiranih in so odvisni od izvajalca modeliranja poslovnih procesov. V literaturi [11] sta sicer predlagana naslednja pristopa:

Skupinski pregled modelov (angl. walkthrough) Gre za sestanke, ki se jih udeležijo avtorji izdelanih modelov in neodvisni opazovalci, ki sicer razumejo modelirne tehnike, vendar ne poznajo poslovnih procesov. Avtorji razložijo in razkažejo procesne modele, opazovalci pa ocenjujejo, ali so modeli dovolj konsistentni, razumljivi in če podajo dovolj informacij o poslovnih procesih.

Sestanki za skupinsko načrtovanje (angl. Joint requirement planning session) Pri JRP gre za skupinsko vrednotenje procesnih modelov, ki jih običajno pripravijo poslovni uporabniki. Na sestanku morajo biti tako implementatorji IT podpore kot poslovni uporabniki, ki so nosilci izdelave procesnih modelov. Bistvo takšnega pristopa je učinkovito širjenje informacij med vsemi akterji procesa modeliranja poslovnih procesov.

¹Danes je že zelo razširjen MDD pristop, kjer se izdelane modele preslika v stanje, ki je čimbolj blizu končni implementaciji podpore poslovnih procesov.

V magistrski nalogi smo opravili validacijo in verifikacijo predlagane tehnike modeliranja ad-hoc poslovnih procesov s pomočjo študija primera (glej Poglavlje 9), kar je opisano v Poglavlju 9.

5.2 Uporaba Petrijevih mrež kot formalnega jezika za modeliranje

Potreba po čimbolj rigoroznem in natančnem opisu poslovnih procesov zahteva formalen in natančen jezik. Eden prvih pristopov k takšnemu opisu procesov so bili različni matematični jeziki. Dejstvo je, da če lahko poslovni proces matematično formaliziramo, lahko to formalizacijo preslikamo v poljubno obliko. Primer takšnega jezika je notacija Petrijevih mrež [4, 31, 26, 30].

Petrijeve mreže so formalen matematični jezik, ki ga lahko uporabljamo za opis diskretnih sistemov. Omogočajo nazoren in natančen opis pretoka podatkov v sistemu, prikaz toka in vzorcev izvajanja, hkrati pa nam ponujajo dobre možnosti za izvajanje simulacij in testov. V nadaljnjem besedilu se bomo osredotočali predvsem na Petrijeve mreže z vidika grafične notacije.

5.2.1 Osnovni gradniki Petrijeve mreže

Petrijeve mreže so matematično formalno četvorka:

$$P(S, T, W, M_0)$$

S je množica *mest*: $S = (s_1, s_2, s_3, \dots)$, T je množica *prehodov*: $T = (t_1, t_2, t_3, \dots)$, W je množica *povezav*: $W = (S \times T) \cup (T \times S)$, M_0 pa je začetna označitev mreže.

Mesta so med seboj vedno povezana s povezavami, ki tečejo od mesta do prehoda in od prehoda do mesta. Gre za vhodne in izhodne funkcije mest. Množica povezav določa možnost prehoda iz enega mesta v ostala mesta grafa. Povezava nikoli ne povezuje dveh mest ali dveh prehodov.

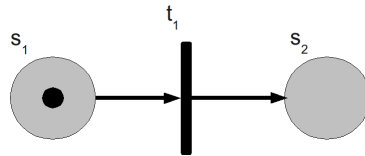
Pomemben koncept Petrijevih mrež je *označevanje*. Z označitvijo mestom dodelimo *žetone*. Vsakemu mestu lahko dodelimo poljubno število žetonov, kar običajno zapišemo v vektorski obliki:

$$\mu = (\mu_1, \mu_2, \mu_3, \dots)$$

Ta zapis nam pove, da je mestu s_1 dodeljeno μ_1 žetonov, mestu s_2 je dodeljeno μ_2 žetonov, itn. Število in položaj žetonov določa možnosti izvajanja prehodov po mreži.

Izvajanju prehodov po mreži rečemo *izvajanje Petrijeve mreže*. Vsako izvajanje je določeno z dano začetno porazdelitvijo M_0 žetonov. Izvajanje mreže poteka tako, da se vršijo *vžigi prehodov*, med katerimi se žetoni prenesejo iz vhodnega na izhodno mesto prehoda. Pogoji za prehod je seveda ustrezna razpoložljivost žetonov na vhodu prehoda kar pomeni, da mora biti v vhodnih mestih vsaj toliko žetonov, kot je povezav med

vhodnim mestom in prehodom. Temu pravimo, da je prehod *omogočen*. Ob prehodu se žetoni odstranijo iz vhodnih mest prehoda, na vsako od izhodnih mest prehoda pa se doda toliko žetonov, kolikor povezav vodi od prehoda do izhodnega mesta. Izvajanje Petrijeve mreže se vrši toliko časa, dokler je omogočen vsaj en prehod v mreži.



Slika 5.1: Slika preproste Petrijeve mreže

Na Sliki 5.1 vidimo, da je v začetni porazdelitvi žetonov mestu s_1 dodeljen 1 žeton, kar pomeni, da je prehod t_1 omogočen. Po vžigu prehoda se žeton odstrani iz mesta s_1 in se ga doda v mesto s_2 .

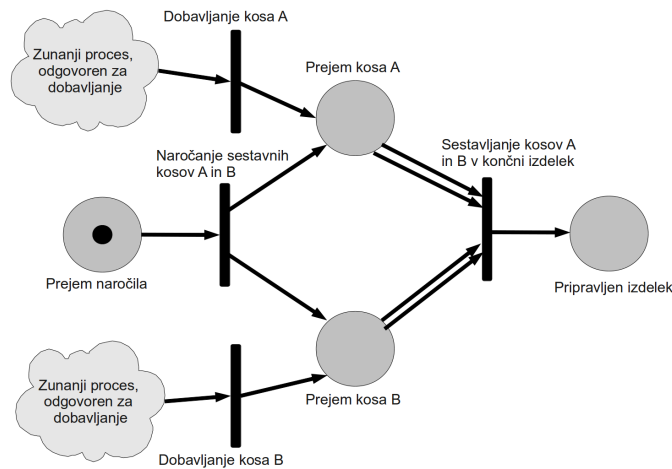
S stališča vizualnega jezika in za potrebe modeliranja poslovnih procesov lahko Petrijevo mrežo definiramo kot usmerjen graf z dvema tipoma vozlišč [4]:

Prehodi predstavljajo aktivnosti Aktivnosti, ki se odvijajo v poslovnih procesih, se v notaciji Petrijevih mrež prikaže v obliki prehodov. Aktivnost se izvede, ko se izvrši vžig prehoda. Uporaba takšne notacije omogoča preprost prikaz različnih vzorcev odvijanja poslovnih procesov².

Mesta predstavljajo pogoje Mesta v Petrijevih mrežah predstavljajo pogoje za izvedbo aktivnosti. Pogoji so običajno razpoložljivost virov v poslovnem procesu, lahko pa pomen pogoja razširimo na poljuben koncept, ki predstavlja pogoj za izvajanje aktivnosti.

Na Sliki 5.2 je prikazan primer preprostega poslovnega procesa, ki se začne s prejemom naročila. Po prejemu naročila je treba naročiti sestavna kosa končnega izdelka. Ker se ta proces ne osredotoča na samo dobavo sestavnih kosov, je proces dobave abstrahiran z zunanjim procesom, ki ob dobavi v naš proces pošlje žeton. Po dobavi obeh kosov se lahko izvede sestavljanje kosov, ki pripelje do konca procesa. Na sliki vidimo, da so pogoji za izvajanje posameznih poslovnih aktivnosti prikazani z mesti oz. razpoložljivostjo žetonov vhodnih mest za posamezen prehod, ki predstavlja aktivnost. Prejem naročila je predstavljeno z mestom *prejem naročila*. Šele ko je v tem mestu žeton, se izvede aktivnost *naročanja sestavnih kosov A in B*. Prejem naročila je potemtakem predpogoj za to aktivnost, kar je na Petrijevi mreži predstavljeno z omogočanjem prehoda oz. razpoložljivostjo žetona na vhodnem mestu. Pri aktivnosti *sestavljanje kosov A in B v končni izdelek* vidimo, da ima iz posameznega vhodnega mesta po dve povezavi. Po definiciji velja, da se lahko ta aktivnost sproži šele takrat, ko je v vhodnem mestu vsaj toliko žetonov, kot je povezav med vhodnim mestom in prehodom. Zato morata biti v mestih *prejem kosa A in*

²Z vzorci potekov poslovnih procesov mislimo na vzorce kot so npr. zaporedno in paralelno izvajanje, izvajanje zank, vejitev, itn.



Slika 5.2: Vzorec vzporednega poteka poslovnega procesa s sodelovanjem zunanjih procesov, prikazan s Petrijevo mrežo

prejem kosa B na voljo po dva žetona, kar pomeni, da mora po en žeton v vsako od teh dveh mest priti od zunaj oz. iz prehodov *dobavljanje kosa A* in *dobavljanje kosa B*. Ker se naš model poslovnega procesa ne osredotoča na dobavo, smo ta del abstrahirali in ga naredili za predpogoj za izvajanje aktivnosti *sestavljanje kosov A in B v končni izdelek*. Slika lepo prikazuje, kako lahko s preprostimi in osnovnimi gradniki zmodeliramo poslovni proces in prikažemo vzorce kot so vzporedno izvajanje aktivnosti in čakanje na pogoje za začetek aktivnosti.

5.2.2 Primer modeliranja ad-hoc procesa s Petrijevim mrežami

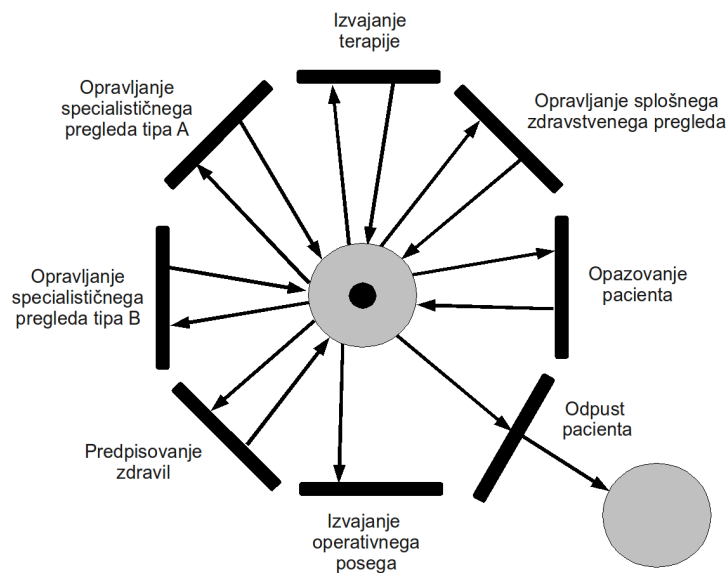
Petrijeve mreže so sintaktično in semantično močan in poln jezik[26, 30]. Zato je s Petrijevo mrežo v matematično formalnem smislu možno zmodelirati praktično vsak poslovni proces, ne glede na njegovo dinamiko. Da bi prikazali, kako se modeliranje ad-hoc procesov obnese v praksi, smo poizkusili zmodelirati preprost primer procesa zdravstvene obravnave pacienta.

Z ad-hoc procesi se zaradi narave dela srečujejo pri zdravstveni obravnavi pacientov. Ko pride pacient z zdravstvenimi težavami v bolnico, mora opraviti različne preglede in preiskave. Preglede izvajajo specialisti v različnih ambulantah in podajajo svoje diagnoze, ter pacienta pošiljajo na dodatne preiskave, zdravljenja, terapije, posege, itn. Dejstvo je, da se aktivnosti v takšnem procesu ne more napovedati vnaprej. Za pacienta se ne ve, katere vse preiskave bo moral opraviti in v kakšnem vrstnem redu. Glede na rezultat posameznih preiskav in posegov se sproti odreja nadaljnje aktivnosti.

Za zgledni primer uporabe različnih modelirnih tehnik na primeru ad-hoc procesov bomo predpostavili primer, ko pride pacient po zdravstveno oskrbo v bolnišnico. V bolnišnici sta poleg splošnega oddelka na voljo dve specialistični ambulanti. Splošni in

specialistični zdravniki lahko odredijo opazovanje pacienta, operativni poseg, izvajanje terapije, ali pa predpišejo določena zdravila. Vsak izmed specialistov lahko pacienta v primeru odprave zdravstvenih težav odpusti iz bolnice.

Da poudarimo sestavo ad-hoc procesa, naštejmo možne aktivnosti. Ad-hoc proces sestavljajo aktivnosti opravljanja treh možnih pregledov (pri splošnem zdravniku, kjer se obravnava pacienta običajno začne, in pri specialističnih zdravnikih A in B), aktivnosti predpisovanja zdravil, aktivnost opazovanja pacienta, aktivnost izvajanje terapije in aktivnost izvajanja operativnega posega. Na začetku, ko pacient pride v bolnišnico, ne vemo, katere aktivnosti se bodo tekom njegove obravnave izvajale. Prav tako se lahko po zaključku posamezne aktivnosti glede na rezultate tekoče obravnave začne katerakoli druga aktivnost. Vrstni red aktivnosti se torej določa sproti in glede na potek zdravljenja.



Slika 5.3: Preprost ad-hoc proces, zmodeliran s Petrijevo mrežo

Na Sliki 5.3 je prikazan primer zglednega modela, narejenega s Petrijevo mrežo. V sredini je mesto, v katerem je v začetni označitvi en žeton. To sredinsko mesto zagotavlja možnost prehajanja med vsemi aktivnostmi. En žeton zagotavlja, da se lahko za pacienta izvaja samo ena aktivnost naenkrat. Če bi hoteli dodati možnost vzporednega izvajanja dveh aktivnosti, bi v sredinsko mesto dodali še en žeton.

Izbor naslednje aktivnosti v takšnem modelu ni eksplicitno določen. Vsakič, ko se ena od aktivnosti izvede in se žeton vrne v sredinsko mesto, se lahko vžge katerikoli prehod. Če bi hoteli zagotoviti pogojnost izvajanja posameznih aktivnosti, bi v vsakega od prehodov speljali povezave iz dodatnih mest, ki predstavljajo pogoje za izvajanje posameznih prehodov.

Poslovni proces se konča, ko se izvede prehod *odpust pacienta*. Ko pride enkrat žeton v desno spodnje stanje, se ne more več vžgati noben prehod in izvajanje ad-hoc procesa je zaključeno.

Slabost takšnega prikaza je ta, da se vse vrtili okoli enega mesta. Gre torej za mesto sinhronizacije in je za logiko izvajanja poslovnega procesa brez pomena, vendar ga moramo uporabiti zaradi same notacije. Sredinsko mesto zagotavlja stohastičnost oz. nedoločeno izvajanje ad-hoc procesa. Vedno, ko se žeton vrne nazaj v sredinsko mesto, je možna izvedba katerekoli aktivnosti.

Takšen diagram postane težko berljiv, če hočemo zmodelirati kompleksnejši proces. Formalno gledano je dodajanje možno in ga izvedemo z novimi aktivnostmi okoli sredinskega mesta. Berljivost bi povečali tako, da bi nove dodane aktivnosti generalizirali z novim podgrafom, prvotno sliko pa bi uporabljali kot kontekstni graf vseh aktivnosti ad-hoc procesa. Podobno smo naredili na Sliki 5.2, kjer smo abstrahirali pogoje za izvajanje aktivnosti.

Semantika Petrijeve mreže omogoča prikaz različnih vzorcev poteka poslovnih procesov, kot so npr. zaporedni in vzporedni tok (glej Slika 5.2), zanke, vejitve, itn. Če bi hoteli vse to izraziti tudi v ad-hoc poslovnem procesu, bi zopet prišlo do prevelike kompleksnosti diagramov. S poizkusi modeliranja več preprostih ad-hoc procesov, kjer smo postopoma povečevali kompleksnost in število pojavitev kompleksnejših vzorcev poteka procesa, smo prišli do ugotovitve, da je izdelava takšnih diagramov zapletena in zahteva veliko energije tako za izdelavo kot za branje. Deloma se da to poenostaviti in zaobiti z generalizacijo in postopnim dodajanjem podrobnosti glede na globino prikazanega procesa, vendar se v tem primeru že za preproste ad-hoc procese zelo hitro poveča število nivojev prikaza procesov, kar pa spet vodi v slabšo berljivost.

Slika 5.3 s stališča logike ad-hoc procesa ne poda veliko informacij. Diagram s tako malo aktivnostmi je sicer še pregleden, vendar bi bila berljivost diagrama v primeru več aktivnosti zelo hitro veliko slabša. Diagram praktično ne poda nič več informacij, kot bi jih dobili v navadnem seznamu aktivnosti.

5.2.3 Uporabnost in primernost Petrijevih mrež za modeliranje ad-hoc procesov

Petrijeve mreže so dober pripomoček za formalno modeliranje poslovnih procesov. Polna semantika in sintaksa omogočata izdelavo praktično vsakega vzorca poteka procesa, zaradi matematične in formalne narave pa je relativno lahko na takšnih modelih izvajati simulacije in testiranja.

Vendar se moramo zavedati dejstva, da je modeliranje s Petrijevim mrežami zelo formalen pristop in se zelo oddaljuje od uporabnosti, ki jo zahteva poslovni uporabnik. Petrijeve mreže so lahko dobra osnova za izdelavo podrobnih načrtov za implementacijo IT podpore za poslovne procese, vendar jih morajo narediti tehnični uporabniki. Kljub temu, da imajo Petrijeve mreže polno semantiko in sintakso, je izraznost za modeliranje poslovnih procesov majhna. Pri uporabi Petrijevih mrež se je treba za dosego prikaza lastnosti poslovnega procesa preveč osredotočati na same semantične konstrukte. Pri modeliranju ad-hoc poslovnih procesov smo ugotovili, da semantičen konstrukt za prikaz nedoločeno vrstnega reda izvajanja aktivnosti že sam po sebi ni najbolj intuitiven,

hkrati pa poslabša splošno razumljivost celotnega diagrama.

Uporaba Petrijevih mrež za formalizacijo ad-hoc procesov je sicer mogoča, vendar se ne izkaže za najboljšo izbiro. Poizkusi uporabe Petrijevih mrež in pregled tega področja pa nam da dobro osnovo za razumevanje, kakšna mora biti diagramska tehnika za modeliranje ad-hoc procesov. Iz Petrijevih mrež se lahko naučimo, kaj moramo v diagramski tehniki ponuditi tako poslovnemu kot tehničnemu uporabniku. Prvim mora biti na voljo visoko izrazen vizualni jezik, ki bo omogočal zajem in pregled ad-hoc procesa, slednjim pa možnost preslikave takšnega modela v čim bolj končen načrt za implementacijo [29].

5.3 BPMN diagramska tehnika

Notacija BPMN [9, 40, 22, 25] je grafična notacija za modeliranje poslovnih procesov. Strokovnjaki na področju modeliranja poslovnih procesov ocenjujejo, da je BPMN med najbolj primernimi modelirnimi tehnikami na voljo [9]. Skrb za razvoj notacije trenutno nosi konzorcij OMG [38].

BPMN naj bi izpolnjevala dva pomembna cilja [9]:

Razumljivost notacije za poslovne in tehnične uporabnike vseh nivojev Kot smo že večkrat poudarili, je popis in zasnova poslovnih procesov domena poslovnih uporabnikov. Zato mora biti notacija oz. diagramska tehnika, v kateri poslovni uporabniki modelirajo procese, dovolj razumljiva in uporabna za ljudi brez tehničnega predznanja. Prav tako naj bi uporaba BPMN omogočala sodelovanje uporabnikov z različnih odločitvenih ravni, od taktične do strateške. Tako naj bi bili isti diagrami uporabni za upravnike kot za izvajalce poslovnih procesov. Hkrati pa naj bi BPMN diagrami omogočali zajem dovolj tehničnih podrobnosti, ki so pomembne za implementatorje IT podpore.

Možnost preslikave BPMN diagrama v delujočo kodo Uporaba večine tehnik modeliranja poslovnih procesov nam ne prihrani dela preslikave procesnih modelov v končno implementacijo podpore procesov. Poslovni uporabniki v okviru BPR projektov izdelajo podrobne procesne modele, ki pa jih morajo nato v IT področju pogosto izdelati še enkrat za namen tehnične implementacije. Cilj uporabe BPMN je izognitev takšnemu dvojnemu delu. To ne samo da omogoča hitrejši razvoj podpore, temveč se spremembe, ki se zgodijo v poslovnih procesih in jih popišejo poslovni uporabniki, hitreje in z manj dela prenesejo v podporo procesov. Običajno se da BPMN diagrame preslikati neposredno v BPEL, ki omogoča izvajanje poslovnih procesov.

BPMN je na področje modeliranja poslovnih procesov uvedla standardizacijo. Ker na področju modeliranja poslovnih procesov ni poenotenih pogledov na to, katere tehnike modeliranja so najbolj primerne, se za modeliranje uporablja veliko število različnih tehnik. To predstavlja težavo tako pri izobraževanju uporabnikov teh tehnik kot pri prenosljivosti procesnih modelov v različna izvajalna okolja. Pogosto se zgodi, da v neki

združbi začnejo uporabljati določeno notacijo in naredijo celoten popis poslovnih procesov. Vendar se zaradi dolžine trajanja takšnih projektov lahko uporaba in podpora takšne modelirne tehnike vmes opusti. Tako združbi ne preostane nič drugega, kot da stare procesne modele naredijo še enkrat v kakšni bolj aktualni diagramski tehniki. Ker se število novih diagramskih tehnik zelo hitro povečuje, je verjetnost da bo diagramaska tehnika procesnih modelov združbe zastarela, precej velika.

Pomanjkanje standardizacije vpliva tudi na izbor različnih tehnik modeliranja za različne tipe poslovnih procesov. Dokumentacijski faktorji za temeljne poslovne procese narekujejo izbor strukturiranih, v aktivnosti usmerjenih diagramskih tehnik (glej Poglavlje 5.1), dokumentacijski faktorji za podporne procese pa se običajno osredotočajo na tok informacij in dokumentov. Pogosto je popis vseh poslovnih procesov v združbi zaradi tega narejen v različnih diagramskih tehnikah, kar otežuje avtomatično preslikavo v izvajalno okolje oz. implementacijo IT podpore. BPMN naj bi omogočala opis procesov različnih tipov in tako odpravila potrebo po uporabi različnih diagramskih tehnik.

V zadnjem obdobju je postala BPMN zelo razširjena. Na voljo obstaja veliko orodij, s katerimi olajšamo izdelavo procesnih modelov v BPMN, poleg tega pa je na trgu kar nekaj proizvodov, ki na podlagi BPMN modelov neposredno ali posredno s prevedbo v BPEL poslovne procese tudi izvajajo. Eden izmed razlogov za takšno razširjenost BPMN je tudi razširjen SOA pristop, v katerem se poslovne procese zmodelira v BPMN in nato prevede v izvajalni BPEL.

Zaradi uporabnosti in razširjenosti BPMN se je pri definiranju standardov ISO sprejelo sklep, da lahko diagrami, narejeni v BPMN, služijo kot del poslovnika kakovosti po standardih ISO 9001:2000. Hkrati so takšni diagrami dobra osnova za navodila za udeležence v poslovnih procesih, saj so razumljivi tako tehničnemu kot poslovnemu osebju.

5.3.1 Osnovni gradniki BPMN

Z BPMN notacijo izdelamo t.i. BPD diagram. Na voljo imamo precej omejeno množico osnovnih gradnikov diagrama, vendar so dovolj izrazno močni, da lahko z njimi opišemo praktično vsak poslovni proces. Gradniki se delijo v 4 kategorije:

Procesni elementi Procesni elementi so lahko aktivnosti, poslovni dogodki ali kretnice.

So temeljni gradniki vsakega BPMN diagrama in opisujejo osnovne značilnosti zmodeliranega poslovnega procesa. Aktivnosti so enote dela, ki se opravljajo tekom poslovnega procesa, lahko pa jih podrobneje predstavimo z novimi pod-procesi. Posebna vrsta pod-procesov so tudi transakcije, ki jih moramo obravnavati kot atomarna enota dela. Poslovni dogodki so dogodki, ki vplivajo na potek poslovnega procesa. To so lahko impulzi iz zunanosti procesa, lahko je to prehod v določeno stanje procesa, začetek ali konec procesa, itn. Kretnice so gradniki, ki omogočajo vejitev toka poslovnega procesa. Gre za odločitvena mesta z določenimi pogoji, katerih izpolnjevanje tekom izvajanja poslovnega procesa določa tok procesa.

Povezovalni elementi S povezovalnimi elementi med seboj povezujemo procesne elemente. Možni tipi povezovalnih elementov so sekvenčni, sporočilni in asociacijski

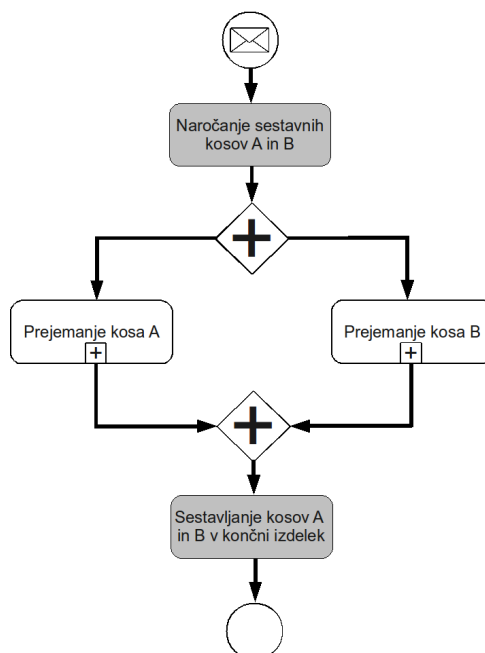
elementi. Sekvenčni element je najbolj preprost in prikazuje vrstni red izvajanja procesnih elementov. To je usmerjena povezava med dvema procesnima elementoma, katere smer določa vrstni red izvajanja teh dveh procesnih elementov. Sporočilni povezovalni elementi prikazujejo tok sporočil med različnimi vlogami v poslovnem procesu. Običajno tukaj govorimo o toku sporočil med področji odgovornosti, ki jih predstavimo s stezami, ali pa gre za tok sporočil med različnimi organizacijami. Zadnji tip povezovalnih elementov so asociacijski povezovalni elementi, ki povezujejo različne informacijske artefakte s procesnimi elementi. Asociacijska povezava poda informacijo o tem, kaj se dogaja z informacijskimi artefakti tekom poslovnega procesa.

Bazeni in steze Bazene in steze uporabljamo za prikaz odgovornosti v poslovnem procesu. Običajno uporabljamo bazene za predstavitev odgovornosti po združbah, steze pa za predstavitev odgovornosti po poslovnih funkcijah znotraj združb. Takšna uporaba ni nujna in lahko abstrakcijo te hierarhije premaknemo na nižji nivo, tj. da bazene uporabljamo za predstavitev odgovornosti po poslovnih funkcijah znotraj združbe, steze pa za odgovornosti tipov posameznih udeležencev v poslovnem procesu.

Informacijski artefakti Z informacijskimi artefakti opišemo dodatne informacije o poslovnem procesu, ki jih sicer ne moremo prikazati z do sedaj naštetimi gradniki. Informacijski artefakte so lahko treh tipov in sicer dokumenti, anotacije in skupine. Z dokumenti opišemo specifično uporabo informacij, ki nastajajo, se uporabljajo ali pa so potrebne za izvajanje neke aktivnosti. Anotacije so dodatno besedilo, ki ga lahko dodamo v povezavi s katerikoli procesnim elementom in tako dodatno opišemo njegove lastnosti. Tako dokumente kot anotacije povežemo s procesnimi elementi z asociacijskimi povezavami. Zadnji tip informacijskih artefaktov so skupine, s katerimi med seboj združujemo procesne elemente v neko logično povezano skupino. Skupine so namenjene zgolj dodatnemu pojasnjevanju in boljši dokumentaciji, nimajo pa neposredne povezave z izvajanjem procesov.

Modeliranje procesa poteka postopno in hierarhično. V BPD diagramu najprej izdelamo kontekstno sliko procesa, kjer procesni elementi za prikaz aktivnosti predstavljajo pod-procese. Na naslednji stopnji prikažemo posamezne od teh aktivnosti na bolj podrobni ravni in modeliramo dejanske aktivnosti poslovnega procesa. Seveda je lahko število stopenj takšne generalizacije poljubno, vendar se moramo zavedati, da preveč postopna generalizacija vpliva tudi na zmanjšano berljivost celotne slike diagramov.

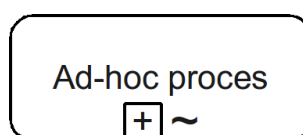
Na Sliki 5.4 je prikazan isti poslovni proces, kot smo ga prikazali pri modeliranju s Petrijevo mrežo na Sliki 5.2. Vidimo, da lahko z gradniki BPMN zelo preprosto prikažemo vzporeden potek poslovnega procesa. Aktivnosti *prejemanje kosa A* in *prejemanje kosa B* sta predstavljeni s posebnim gradnikom, ki predstavlja pod-proces in ga lahko podrobneje predstavimo na novem BPD diagramu.



Slika 5.4: Vzorec vzporednega poteka poslovnega procesa, prikazan v BPMN

5.3.2 Primer modeliranja ad-hoc procesa z BPMN notacijo

Za modeliranje ad-hoc procesov vsebuje BPMN poseben gradnik, s katerim lahko prikažemo ad-hoc pod-proces.

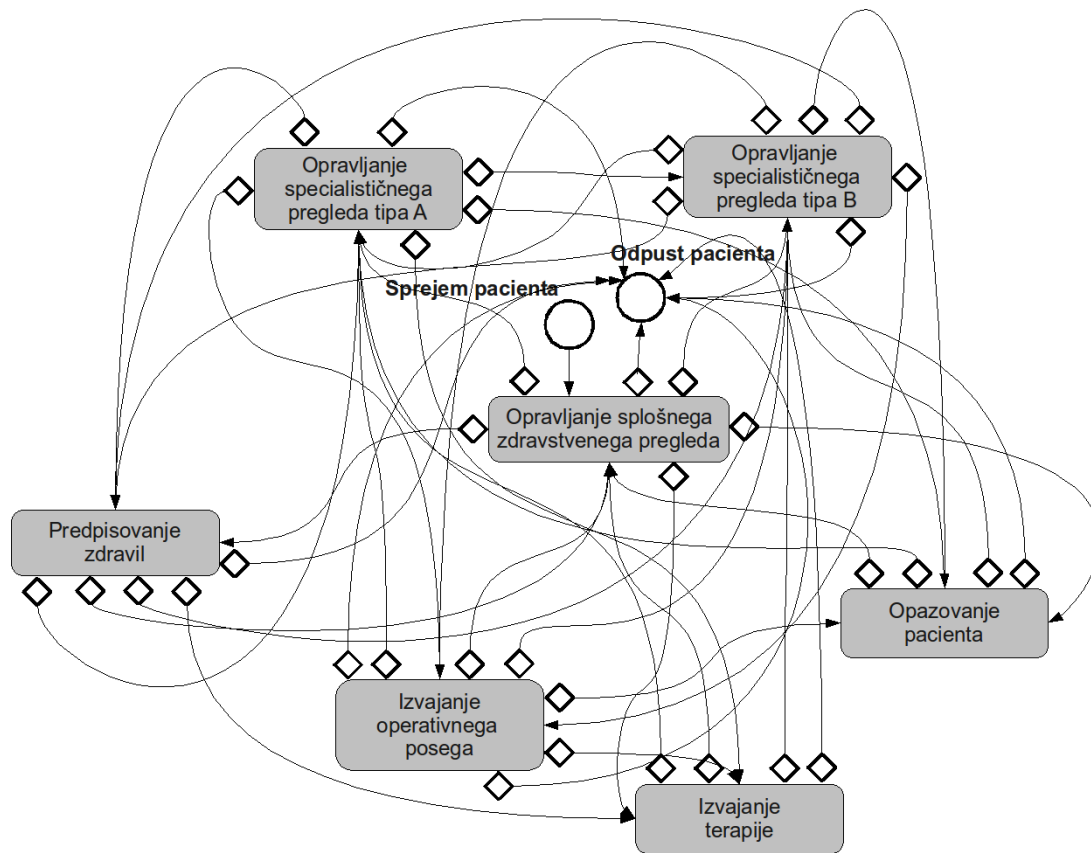


Slika 5.5: Primer, kako v BPMN prikažemo ad-hoc pod-proces

Gradnik, prikazan na Sliki 5.5 je uporaben na kontekstnem nivoju, ko hočemo med množico pod-procesov poudariti ad-hoc pod-procese. Po definiciji je v BPMN ad-hoc pod-proces definiran kot množica aktivnosti brez določenega zaporedja [9]. Takšen gradnik nam ne omogoča prikaza dejanskih aktivnosti oz. poteka ad-hoc procesa, temveč lahko z njim označimo samo naravo procesa.

Za namen prikaza možnosti modeliranja ad-hoc procesa na nivoju aktivnosti smo zopet zmodelirali preprost ad-hoc poslovni proces iz zgledega primera, ki je opisan v Poglavlju 5.2.2.

Ena izmed možnosti izražanj pogojnega izvajanja je uporaba pogojnega sekvenčnega povezovalnega elementa, ki je označen s puščico, ki se začne z rombo. Romb predstavlja poljuben pogoj, ki mora biti izpolnjen za prehod v naslednjo aktivnost.



Slika 5.6: Preprost ad-hoc proces, zmodeliran v BPMN diagramu

Začetek procesa je v krogu *sprejem pacienta*, iz katerega lahko preidemo v aktivnost opravljanja splošnega zdravstvenega pregleda, s katerim se običajno začne proces obravnave pacienta. Iz ostalih aktivnosti so možni različni medsebojni prehodi. Konec procesa je označen v krogu *odpust pacienta*.

V primerjavi z modeliranjem ad-hoc procesa s Petrijevo mrežo (glej Slika 5.3) na BPD diagramu ne uporabljamo nobenega posebnega sredinskega gradnika, s katerim bi označevali nedoločenost. Pri Petrijevi mreži smo morali uporabiti sredinsko mesto, s katerim smo simulirali možnost poljubnega prehajanja med stanji. V BPD takšnega mesta ni potrebno uporabiti, ker lahko uporabimo pogojne povezave med elementi. Pogoji na povezavi so izvidi in stanja zdravljenja pacienta. Iz vsake aktivnosti so možni prehodi v ostale aktivnosti glede na pogoje, ki so izpolnjeni v trenutku izhoda iz aktivnosti. Praktično si to lahko predstavljamo kot določen izvid ali diagnozo, ki jo postavi specialist tekom izvajanja aktivnosti specialističnega pregleda, in ki napoti pacienta na dodatne preiskave, operativni poseg, itn. Preverjanje pogojev se izvaja v času izvajanja ad-hoc procesa ob koncu posamezne aktivnosti, ko se izbira naslednjo aktivnost. Vnaprej ne moremo predvideti, katere aktivnosti se bodo izvedle in zato ad-hoc procesa ne moremo prikazati v običajnem sekvenčnem načinu.

Ekspliciten zapis pogojev običajno postavimo neposredno na pogojno povezavo. V primeru, ko je takšnih pogojnih povezav relativno malo, zapis pogojev ne pokvari berljivosti diagrama. V našem primeru, ko je praktično vsaka povezava pogojna, pa bi ekspliciten zapis pogojev še dodatno poslabšal berljivost diagrama. Že brez teh zapisov je diagram zaradi velikega števila povezav precej nepregleden in za uporabnika neintuitiven, ob dodatnih zapisih pogojev pa bi bil praktično neberljiv.

Dejstvo je, da je na Sliki 5.6 zmodeliran zelo preprost primer ad-hoc procesa. V primeru dodajanja novih aktivnosti bi morali na vsaki obstoječi aktivnosti dodati novo povezavo na aktivnosti, ki jih dodajamo. Ko smo poizkušali risati takšne diagrame smo prišli do ugotovitve, da je že pri 5 aktivnostih diagram nepregleden zaradi velikega števila povezav. Kljub temu, da smo uporabili računalniško podprto orodje, je zaradi velikega števila možnih kombinacij medsebojnih povezav takšen diagram praktično neuporaben.

BPD ne prikaže, kateri viri procesa so potrebni za izvajanje posamezne aktivnosti. Če bi se v zglednem primeru odvijala dva ad-hoc procesa hkrati, iz diagrama ni možno razbrati, katere aktivnosti zasedajo katere vire in tako onemogočajo izvajanje vzporednega ad-hoc procesa.

Na Sliki 5.6 vidimo, da lahko iz katerekoli aktivnosti preidemo v katerokoli naslednjo aktivnost. Z uporabo pogojnih povezovalnih elementov smo podali tudi pogoje prehodov med stanji. Takšna predstavitev je precej neintuitivna. Pogoje se sicer lahko navede v dodatnih informacijskih artefaktih, vendar to še dodatno zmanjša berljivost in preglednost diagrama. Po našem mnenju diagram sicer lepo prikazuje možnosti pogojnih prehodov med aktivnostmi, vendar nam ne pove veliko o teh pogojih. Zato je model procesa nepregleden in težko razberemo, kateri prehodi so sploh možni.

5.3.3 Uporabnost in primernost BPMN za modeliranje ad-hoc procesov

BPMN ponuja zelo dobro diagramsko tehniko, v kateri lahko zmodeliramo praktično vsak poslovni proces. Primeren je tako za uporabo poslovnih kot tehničnih uporabnikov. Vendar lahko po temeljitem pregledu zaključimo, da ne omogoča učinkovitega modeliranja ad-hoc procesov. Da so pri ustvarjanju notacije že mislili na obstoj ad-hoc procesov potrjuje dejstvo, da v notaciji obstaja gradnik za predstavitev ad-hoc procesa. Najbrž zaradi splošnosti diagramske tehnike ta gradnik ni bil dodatno razdelan.

To kaže na dejstvo, da je danes še vedno premalo pozornosti usmerjene v obravnavno dinamike poslovnih procesov. Za BPMN notacijo lahko rečemo, da je zelo ažurna in da sproti izdajajo nove, dopolnjene verzije. Zaključimo lahko, da BPMN *ni* namenjena modeliranju ad-hoc procesom. Mogoče bi modeliranje ad-hoc procesov preveč obremenilo poslovne uporabnike s podrobnostmi in to ne bi predstavljalo dodane vrednosti v celotni obravnavi poslovnih procesov.

Ena izmed bistvenih prednosti uporabe BPMN je možnost, da BPD diagrame neposredno prevedemo v kakšnega od izvajalnih jezikov za poslovne procese, ki jih nato uporabimo kot vhod za procesne strežnike. Tako iz procesnega modela pridobimo možnost

neposrednega prehoda na implementacijo podpore poslovnih procesov. Vendar zaradi pomanjkanja možnosti učinkovite predstavitve ad-hoc procesov to ne velja za zelo dinamične procese. Zaključimo lahko, da je BPMN nadvse uporabna za modeliranje dobro definiranih in statičnih poslovnih procesov, za modeliranje ad-hoc procesov pa ji manjkajo gradniki in dodatni koncepti.

5.4 Diagram EPC

Diagramaska tehnika EPC je bila razvita leta 1992 in je bila na začetku ključni modelirni jezik za ERP sisteme podjetja SAP [10, 4]. Razvita je bila v sklopu pristopa ARIS [17]. Malo po zaslugi podjetja SAP, največ pa po zaslugi dobre zasnovanosti in uporabnosti, se je njena uporaba hitro razširila. Danes obstaja veliko orodij in metodologij, ki uporabljajo EPC kot modelirni jezik. Diagramaska tehnika je zelo razširjena med podjetji, ki ponujajo celovite informacijske rešitve in izvajajo BPR projekte.

Zaradi razširjene uporabe se je EPC diagramaska tehnika hitro razvijala. Ker je bila uporabljena predvsem za namene v gospodarstvu in ne toliko v raziskovalne namene, se je dokaj hitro izkazalo, kateri gradniki manjkajo v praksi. Zato so kmalu izdali novo različico diagramske tehnike, imenovane eEPC (angl. extended Event-driven process chain). V nadaljnjem besedilu bomo sicer uporabljali kratico EPC, vendar je s tem mišljen celotna množica gradnikov, ki je na voljo pod imenom eEPC.

EPC temelji na konceptu stohastičnih mrež in Petrijevih grafov. EPC diagram je usmerjen graf, katerega vozlišča predstavljajo dogodki in aktivnosti poslovnih procesov. Močna sintaksa in semantika ter formalna definicija jezika diagramski tehniki omogočajo, da je zelo intuitiva in uporabna tudi s stališča uporabnikov, ki ne zahtevajo matematično formalnega zapisa, temveč pregleden in berljiv diagram poslovnih procesov.

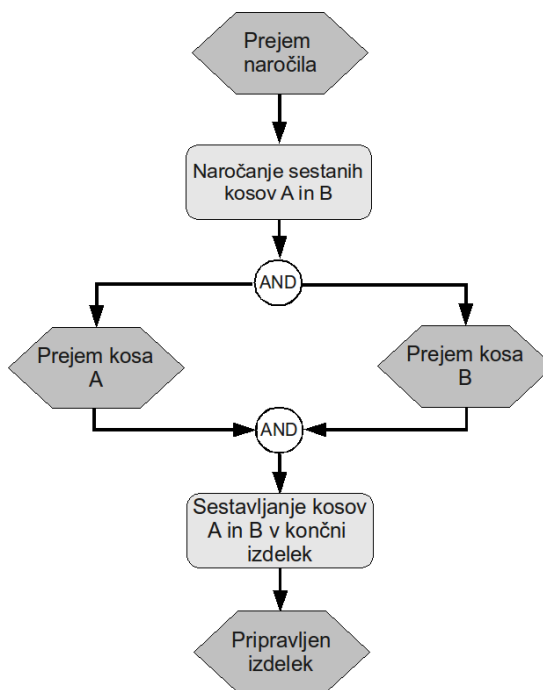
5.4.1 Osnovni gradniki EPC jezika

Glavna dva gradnika jezika EPC sta:

Dogodek Dogodki so pasivni elementi EPC diagrama. Predstavljajo neko stanje ali dogodek v poslovnem procesu. Določajo, pod kakšnimi pogoji se izvajajo aktivnosti v smislu določitve njihovega vhoda ali rezultata. Dogodek je lahko tudi stanje poslovnega procesa oz. trenutek, ko so določene lastnosti poslovnega procesa v določenem stanju. V splošnem se mora EPC diagram začeti in končati z dogodkom.

Aktivnost Aktivnosti so aktivni elementi EPC diagrama in predstavljajo aktivnosti poslovnega procesa. V skladu z definicijo aktivnosti poslovnega procesa, ki na vhode prejete objekte transformira (ustvari neko dodano vrednost) v izhodne objekte, tudi v diagramu EPC z aktivnostjo modeliramo takšno spremembo stanja procesa. Aktivnost lahko ob potrebi po višji stopnji podrobnosti modeliramo z novim EPC pod-diagramom.

Aktivnosti in dogodke med seboj povežemo s kontrolnim tokom. Kontrolni tok omogoča uporabo operatorjev *AND*, *OR* in *XOR*, s čimer lahko modeliramo različne vzorce poteka poslovnih procesov.



Slika 5.7: Vzorec vzporednega poteka poslovnega procesa, prikazan v EPC diagramu

Poleg osnovnih gradnikov lahko v EPC diagramu uporabimo še dodatne artefakte, s katerimi opišemo lastnosti izvajanja poslovnih procesov. Izpostavili bi naslednje:

Vloga Če hočemo eksplicitno predstaviti udeležence poslovnega procesa, uporabimo element *vloga*. S tem za aktivnosti določimo odgovornost ali potrebne udeležence.

Informacijski objekt Kot vhod ali izhod iz aktivnosti se uporabljajo različni podatkovni nosilci. Z dodatnimi elementi EPC diagrama lahko pokažemo, da gre za različne dokumente, slike ali druge izdelke.

Aplikacija Element *aplikacija* ni neposredno vezan na modeliranje poslovnih procesov, temveč se že rahlo dotika implementacije podpore poslovnega procesa. Z njim povelimo, katera aplikacija (ali sistem) je potrebna za izvajanje posamezne aktivnosti.

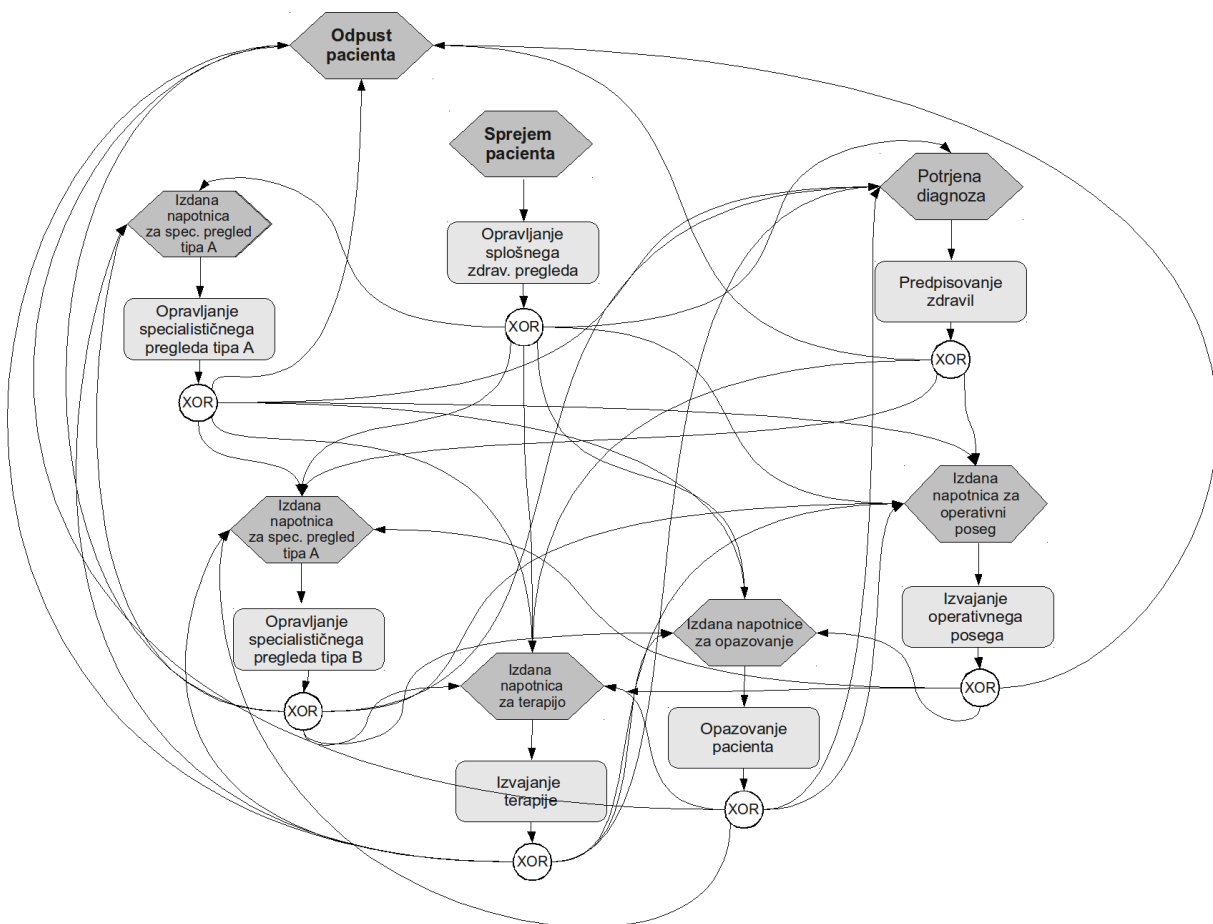
Pri izdelavi EPC diagrama moramo upoštevati 3 EPC pravila:

1. Kontrolni tok povezuje dogodek in aktivnost.
2. Operatorji povezujejo aktivnosti in dogodke preko kontrolnega toka.
3. Vsi dodatni elementi diagrama EPC se vežejo na aktivnost in ne na dogodek.

Osnovni primer uporabe EPC diagrama je prikazan na Sliki 5.7. Dogodki so predstavljeni z romboidi. Romboida *prejem naročila* in *pripravljen izdelek* sta vhodni in izhodni dogodek za ta preprost poslovni proces. Romboida *prejem kosa A* in *prejem kosa B* sta vmesna dogodka. V pravokotnikih so predstavljene aktivnosti poslovnega procesa. Kontrolni tok povezuje med seboj dogodke in aktivnosti. Opazimo, da je za prikaz vzporednega poteka poslovnega procesa potrebna uporaba operatorja *AND*, ki tok procesa najprej veji na dva pod-toka, nato pa predstavlja pogoj za izvajanje zadnje aktivnosti.

5.4.2 Primer modeliranja ad-hoc procesa z EPC diagramom

Primer modeliranja ad-hoc procesa z EPC diagramom smo prikazali na zglednem primeru, opisanem v Poglavju 5.2.2.



Slika 5.8: Primer, kako v EPC diagramu prikažemo ad-hoc proces

Ad-hoc proces se začne izvajati z *vhodnim dogodkom* procesa, ki predstavlja prihod pacienta v bolnišnico. Podobno kot pri diagramu na Sliki 5.6, kjer se najprej opravi splošni zdravstveni pregled, iz vhodnega dogodka sledi izvajanje aktivnosti *opravljanje splošnega*

zdravstvenega pregleda. Ob izhodu iz aktivnosti je uporabljen gradnik *XOR*, ki glede na rezultate opravljenega zdravstvenega pregleda omogoča prehod v naslednje aktivnosti ad-hoc procesa. Vhodne aktivnosti v posamezne aktivnosti ad-hoc procesa so rezultat prejšnje aktivnosti. Tako je npr. v splošnem pregledu izdana napotnica za specialističen pregled tipa A vhodni dogodek v aktivnost opravljanja specialističnega pregleda tipa A. Ad-hoc proces se zaključi, ko pride do izhodnega dogodka *odpust pacienta*.

Ob zaključku vsake aktivnosti je uporabljen gradnik *XOR*, ki dovoljuje nadaljevanje procesa v samo eni naslednji aktivnosti. Če bi hoteli omogočiti vzporedno izvajanje aktivnosti, bi lahko uporabili manj zavezujoč disjunktne gradnik.

Zaradi velikega števila povezav med aktivnostmi smo podobno kot v primeru modeliranja z BPMN tudi pri uporabi EPC diagramске tehnike dobili nepregleden diagram. Veliko število povezav onemogoča dober pregled prehodov med aktivnostmi. Prav tako v diagramu ni razvidno, kateri viri so potrebni za izvajanje posameznih aktivnosti. V primeru, da bi dodali dodatne aktivnosti, bi morali zopet dodati veliko število možnih povezav in v praksi se izkaže, da že v primeru petih aktivnosti takšen diagram postane nepregleden.

5.4.3 Uporabnost in primernost BPMN za modeliranje ad-hoc procesov

Uporaba diagramске tehnike EPC za modeliranje ad-hoc procesov je podobno kot BPMN precej nerodna. Zaradi velikega števila povezav med aktivnostmi postane EPC diagram hitro neberljiv. Že pri izdelavi diagrama na Sliki 5.8, kjer smo zmodelirali ad-hoc proces s samo sedmimi aktivnostmi, je procesni diagram precej nepregleden.

Pri EPC diagramu smo pogoje za prehode med aktivnostmi predstavili s kontrolnim tokom z gradnikom *XOR* in vhodnimi dogodki za posamezne aktivnosti. Takšen zapis pogojev je po našem mnenju bolj intuitiven v primerjavi s pogojnim tokom v BPMN (glej Poglavlje 5.3.2), vendar kljub temu uporabniku ne nudi dovolj dobrega pregleda nad pogoji za izvajanje aktivnosti.

Uporaba EPC diagramске tehnike je za modeliranje ad-hoc procesov dobra samo pri zelo osnovnih, majhnih procesih, kjer je število aktivnosti zelo omejeno. Uporabnost EPC diagramov je precej podobna BPMN diagramom in ne omogoča izdelave preglednih modelov ad-hoc procesov.

5.5 Jezik UML

Zadnji jezik oz. grafična notacija, ki smo jo podrobneje pregledali ob iskanju primernih rešitev za modeliranje ad-hoc procesov, je UML jezik [6, 37, 38, 4]. UML je družina grafičnih jezikov, s katerimi naredimo vizualne modele sistemov. UML notacije so izrazito objektno orientirane in se jih kombinira z objektnim razvojem.

UML je nastal kot poizkus poenotenja in standardizacije velikega števila modelirnih grafičnih jezikov v 90' letih prejšnjega stoletja. Trenutno UML standard vzdržuje OMG

skupina [38].

Pod imenom UML se skriva več diagramskih tehnik, s katerimi opišemo sistem, ki ga analiziramo. Diagrami se osredotočajo na dva vidika:

Statičen vidik Statičen vidik se osredotoča na strukturo sistema. Diagrami opisujejo akterje, objekte, njihove attribute, razmerja med objekti in operacije nad objekti. V to skupino diagramov spadajo razredni diagrami in diagrami komponent.

Dinamičen vidik Dinamičen vidik izpostavlja obnašanje sistema. Diagrami se osredotočajo na to, kaj objekti delajo, kakšno funkcijo opravljajo, katere aktivnosti se izvajajo v sistemu in kakšne spremembe se ob izvajanju teh aktivnosti dogajajo nad objekti. V to skupino diagramov spadajo sekvenčni diagrami, diagrami aktivnosti in diagrami stanj.

Pri modeliranju poslovnega procesa se prikazuje dinamičen vidik sistema. Zato se bomo v nadaljevanju osredotočili na diagram aktivnosti.

5.5.1 Osnovni gradniki diagrama aktivnosti

Diagram aktivnosti je podobno kot EPC ali BPMN zelo bogata notacija s številnimi elementi, kar nam omogoča standardiziran zapis različnih tipov procesov. Za namen ilustracije modeliranja preprostega poslovnega procesa bomo predstavili naslednje elemente:

Aktivnost Element aktivnost predstavlja aktivnosti poslovnega procesa. Podobno kot pri ostalih diagramskih tehnikah lahko aktivnost dodatno razčlenimo v novem, podrobnejšem diagramu aktivnosti.

Akcija Akcije so manjše, bolj podrobno razdelane enote dela, ki sestavljajo aktivnost. Vsako aktivnost lahko prikažemo kot podrobnejši diagram med seboj povezanih akcij.

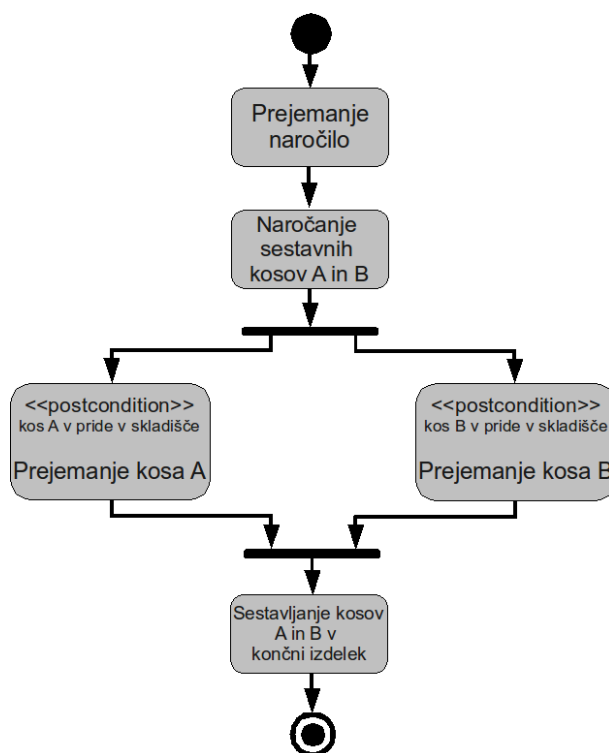
Tok Tok med seboj povezuje kontrolna vozlišča in aktivnosti. Predstavlja smer izvajanja poslovnega procesa.

Kontrolna vozlišča S kontrolnimi vozlišči koordiniramo tok poslovnega procesa. Najbolj osnoven tok procesa je *zaporedje*, kjer sta dve aktivnosti med seboj povezani s tokom. S kontrolnimi vozlišči pa lahko prikažemo vejitve, vzporedna izvajanja, sinhronizacijo, itn.

Za izražanje pogojev pri vejitvah toka med aktivnostmi lahko uporabljamo OCL jezik. OCL jezik je poenoten jezik za izražanje pogojev in je podoben psevdokodi običajnih programskih jezikov.

Poleg OCL lahko v diagramih aktivnosti za opisovanje pogojnosti uporabljamo posebne gradnike, ki nad izvajanje posameznih aktivnosti in akcij postavijo predpogoje in pogoje za izstop iz aktivnosti. Ko na aktivnost postavimo globalne vstopne pogoje, morajo

biti vsi izpolnjeni, da se lahko začnejo izvajati akcije znotraj aktivnosti. Če na aktivnost postavimo globalne izstopne pogoje, se lahko tok izvajanja izven aktivnosti nadaljuje šele, ko so vsi ti pogoji izpolnjeni. Podobno je z lokalnimi pogoji, vendar se ti nanašajo na akcije.



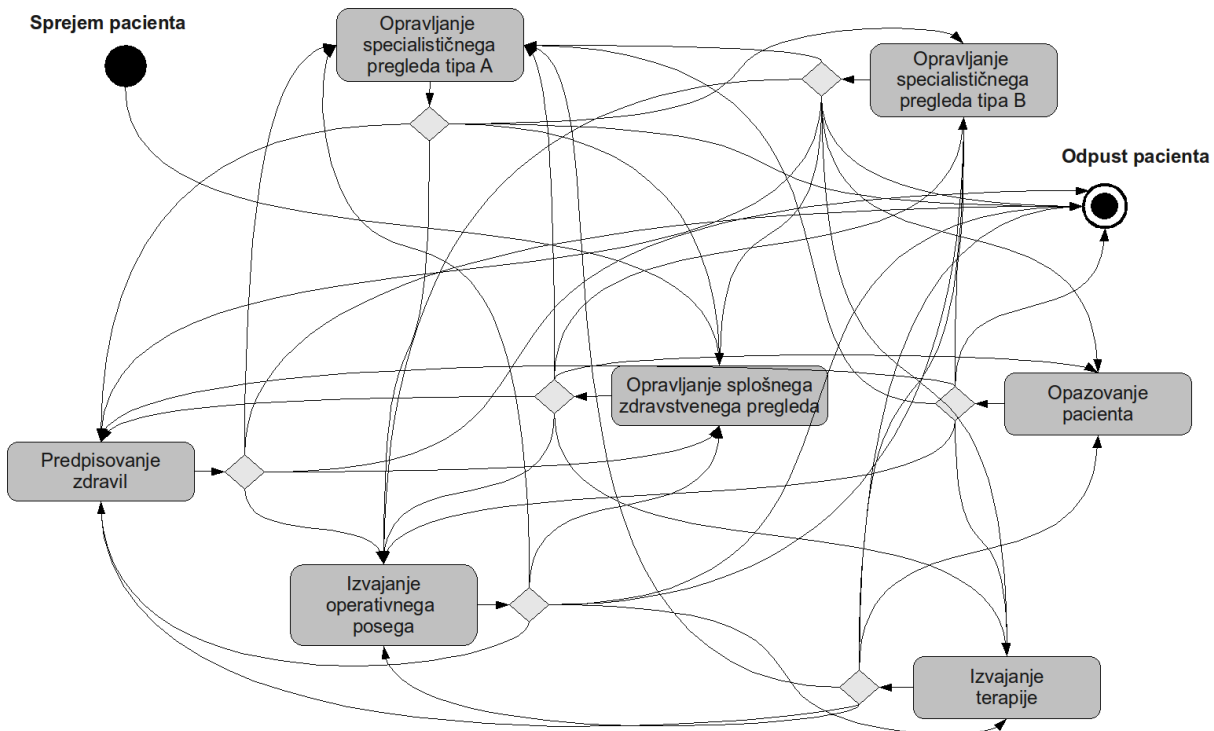
Slika 5.9: Vzorec vzporednega poteka poslovnega procesa, prikazan v UML diagramu aktivnosti

Na Sliki 5.9 vidimo, da se proces začne s črnim krogom, ki predstavlja sprožitveni dogodek, nadaljuje pa se v aktivnostih. Po izvedbi naročila se tok procesa razdeli v dva vzporedna pod-toka. Za aktivnosti *prejemanje kosa A* in *prejemanje kosa B* smo nastavili globalne izstopne pogoje, ki se jih označi s stereotipom `<<precondition>>` in napiše v OCL jeziku. Aktivnosti sta zaključeni šele ob skladiščenju naročenih kosov. Zadnja aktivnost *sestavljanje kosov A in B v končni izdelek* se zaradi predhodnega kontrolnega vozlišča, ki sinhronizira oba pod-toka, lahko izvede šele kot so izpolnjeni obojni izhodni pogoji predhodnih aktivnosti.

5.5.2 Primer modeliranja ad-hoc procesa z diagramom aktivnosti

Pri diagramu aktivnosti smo za prikaz ad-hoc procesa postopali podobno kot pri uporabi BPMN na Sliki 5.6. Ker je vrstni red aktivnosti nedoločen, moramo s povezavami med vsemi aktivnostmi omogočiti poljubno nadaljevanje toka procesa. Vejitve so izražene z

rombom, ki predstavlja pogojne prehode. Proces se začne v črnem krogu, od koder sledi aktivnost *opravljanje splošnega zdravstvenega pregleda*. Iz te aktivnosti pa je izbor prehoda odvisen od izpolnjevanja globalnih vstopnih pogojev posamezne aktivnosti. Proces se konča, ko so v neki aktivnosti doseženi globalni pogoji za izstop iz procesa.



Slika 5.10: Primer, kako v UML diagramu aktivnosti prikažemo ad-hoc proces

Podobno kot pri EPC in BPMN diagramih tudi UML diagram pri prikazu velikega števila povezav postane hitro nepregleden. Prikaz globalnih vhodnih in izhodnih pogojev za posamezno aktivnost lahko izvedemo z uporabo stereotipa `<<precondition>>`. Jasno je, da če bi na diagramu na Sliki 5.10 uporabili še ta dodaten OCL zapis, bi postal diagram popolnoma nepregleden.

5.5.3 Uporabnost in primernost diagrama aktivnosti za modeliranje ad-hoc procesov

Vidimo, da je diagram aktivnosti za prikaz ad-hoc procesov neroden na podoben način kot BPMN ali EPC diagramska tehnika. Razlike med diagramskimi tehnikami poleg same grafične notacije obstajajo večinoma pri podajanju vstopnih in izstopnih pogojev za posamezne aktivnosti. Vse diagramske tehnike ob večjem številu aktivnosti, zaradi potrebe po prikazu velikega števila povezav, postanejo hitro neberljive. Takšni procesni modeli ne predstavljajo dodane vrednosti in so tako za poslovne kot tehnične uporabnike neprimerni za nadaljnjo uporabo.

5.6 Ostali jeziki

Tekom izdelave magistrske naloge smo pregledali še nekaj dodatnih modelirnih jezikov, kjer smo iskali možnosti za modeliranje ad-hoc procesov:

XPDL XPDL [39] jezik se uporablja predvsem pri uporabi različnih sistemov za upravljanje delovnih postopkov. Ker na tržišču obstaja kar nekaj implementacij sistemov za upravljanje delovnih postopkov, je sčasoma prišlo do problema zaradi nestandardiziranih formatov opisov delovnih postopkov. Tako je lahko definicija procesa v določeni notaciji postala popolnoma neuporabna v primeru menjave izvajalnega okolja delovnih postopkov. Uporabniki so bili tako odvisni od implementacije in proizvajalcev teh sistemov, slednji pa so imeli težave s pomanjkanjem standardov in posledično ponovno uporabljivostjo komponent. S XPDL jezikom so poizkusili vzpostaviti standard za enoten zapis delovnih postopkov. XPDL jezik je definirala koalicija WfMC, ki skrbi za standardizacijo in razvoj tega jezika. Ker gre za jezik, ki je namenjen modeliranju delovnih postopkov, ki so sami po sebi zelo rigorozno definirani, tudi XPDL jezik ne omogoča zajema dinamike, razen če dinamiko modeliramo kot veliko število možnih povezav med aktivnostmi.

YAWL YAWL [41] jezik je alternativa BPELu. Temelji na teoriji Petrijevih mrež in različnih vzorcih najbolj razširjenih delovnih postopkov. Jezik lahko zaradi odprtokodne narave uporablja kdorkoli. V paketu je na voljo močno računalniško podrto orodje, s katerim lahko modeliramo procese, poleg tega pa lahko uporabimo odprtokodno izvajalno okolje. Kljub temu, da je to odprtokodna rešitev, je bila že uporabljena v velikih združbah, kjer se je izkazala za dovolj stabilno in učinkovito. Podobno kot XPDL jezik ne vsebuje posebnih gradnikov, s katerimi bi olajšali opis dinamike ad-hoc procesov.

UML diagrami stanj UML diagrami stanj [6] se osredotočajo na možna stanja sistema. Prikazujejo prehode in pogoje za prehode med stanji. Spadajo v skupino UML diagramov, ki prikazujejo dinamičen vidik sistema. Predvsem so primerni za prikaz vstopnih in izstopnih pogojev v različne aktivnosti, hkrati pa je z njimi možno na preglednejši način kot pri ostalih diagramih zajeti veliko število možnih prehodov, kar je značilno za ad-hoc procese.

Študij posameznih modelirnih tehnik na primeru ad-hoc procesa nam je omogočil dober pregled, kaj uveljavljenim modelirnim tehnikam manjka. Po našem mnenju se za modeliranje ad-hoc procesov najbolje obnese EPC diagram. Pri definiranju prilagojene diagramske tehnike, ki bi bila še posebej primerna za modeliranje ad-hoc procesov, smo se tako opirali na izsledke primerjave opisanih modelirnih tehnik in uporabe teh modelirnih tehnik na preprostem ad-hoc procesu ter tako poizkušali obdržati najboljše, kar se že uporablja, dodali pa smo možnost učinkovite obravnave notranje dinamike procesov. Prilagojena diagramska tehnika ne sme izključevati uporabe ostalih diagramskih tehnik in je predstavljena kot dodatek uveljavljenim tehnikam.

Poglavje 6

Alternativni pristopi k modeliranju poslovnih procesov

Ad-hoc procese je zaradi njihove nedeterminističnosti težko modelirati v običajnih modelirnih tehnikah, ki so sicer primerne za modeliranje poslovnih procesov. V naslednjem poglavju sledi pregled nekaj idej, ki predstavljajo odmik od običajnega in uveljavljenega razmišljanja o poslovnih procesov, hkrati pa omogočajo boljše možnosti za obravnavo dinamike ad-hoc procesov. Na teh idejah smo tudi gradili novo modelirno tehniko, razvito v magistrski nalogi.

6.1 Obravnava poslovnih procesov in njihovega okolja kot sistemov

Ideja, da se nek problem modelira tako, da problemsko domeno razstavimo na možna stanja in definiramo prehode med njimi, je že stara in predstavlja uveljavljen način reševanja problemov predvsem na področju strojnega učenja. Tudi različne metodologije gradnje informacijskih sistemov ponujajo diagramске tehnike, s katerimi opišemo možna stanja sistemov. Ko definiramo še dogodke, ki sprožajo prehode med stanji, imamo praktično na voljo sliko in obnašanje celotnega sistema.

Sistem sestavlja množica komponent, vhod v sistem, izhod iz sistema, ter množica možnih transformacij med stanji sistema. Sistem je torej skupina komponent, ki so v medsebojni interakciji. Njihov cilj je, da vhod v sistem skozi zaporedje transformacij spremenijo v izhod oz. rezultat.

Iz opisanega sledi, da takšna definicija sistema ustreza definiciji poslovnega procesa (glej Poglavje 2.2). Komponente sistema si lahko predstavljamo kot udeležence v sistemu. To so različni viri in ljudje. Vhod in izhod sistema ustrezata vhodu in izhodu iz poslovnega procesa, zaporedje transformacij med stanji pa je množica aktivnosti, ki se odvijajo tekom izvajanja poslovnega procesa.

Bistvena prednost predstavitve poslovnega procesa kot sistema je, da lahko na stohastičen način opišemo vrstni red izvajanja aktivnosti. V ad-hoc procesu, kjer je vrstni red

izvajanja aktivnosti nedoločen, lahko z ustrezno postavitvijo pogojev na izvajanje transformacij lepo obvladujemo kombinatorično zahtevnost izbora naslednje možne aktivnosti, hkrati pa še vedno prikazujemo zaporedje izvajanja aktivnosti.

Za modeliranje poslovnih procesov lahko sistemski pristop uporabimo na dva načina:

Poslovni proces kot sistem Kot smo opisali že zgoraj, je takšna definicija poslovnega procesa s sistemskega vidika zelo intuitivna. Poslovni proces kot sistem ima definiran vhod, ki se skozi izvajanje sistema spremeni v izhod. Izvajanje sistema je definirano kot zavzemanje različnih stanj skozi transformacije, ki predstavljajo aktivnosti poslovnega procesa. Transformacije so sprožene takrat, ko se zgodi določen notranji ali zunanji dogodek.

Poslovni proces se odvija v sistemu Po drugi strani si lahko poslovni proces predstavljamo kot entiteto, ki s svojim delovanjem spreminja stanje sistema. Sistem je torej okolje, v katerem se izvaja poslovni proces. Komponente sistema so viri in ljudje, ki so na voljo entitetam, ki predstavljajo poslovne procese. Ko entiteta spremeni stanje kakšne od teh komponent, npr. ko zasede nek vir, s tem spremeni stanje svojega okolja. Takšna predstavitev delovanja poslovnih procesov je še posebej primerna, če bi radi prikazali možnosti sočasnega izvajanja več poslovnih procesov. Ker si entitete pri tej predstavitvi druga drugi zasedajo vire, s tem lepo prikažemo realno dogajanje v poslovnem okolju, kjer je treba paziti na razpoložljivost virov. Vhod in izhod poslovnega procesa lahko pri takšni predstavitvi predstavimo ali s posebnimi stanji celotnega sistema ali pa kot vhode in izhode enolično označenih entitet, ki predstavljajo posamezne instance poslovnega procesa.

Izvajanje poslovnega procesa lahko s sistemskega vidika predstavimo kot avtomat končnih stanj [27]. Avtomat končnih stanj je model sistema, ki lahko zavzame omejeno število stanj. Poznamo *deterministične* in *nedeterministične* avtomate končnih stanj. Pri determinističnih avtomatih je točno določeno, v katero naslednje stanje gre lahko avtomat pod določenimi pogoji. V nasprotju je pri nedeterminističnih avtomatih možen katerikoli prehod. Za potrebe modeliranja poslovnih procesov je sicer po našem mnenju bolj primerna uporaba determinističnega avtomata.

Obravnava sistemov kot poslovnih procesov omogoča tudi uporabo razvitih metod za optimizacijo poslovnih procesov. Če si izvajanje poslovnega procesa predstavljamo kot spreminjanje možnih stanj sistema, lahko za iskanje optimalne poti uporabimo različne algoritme, razvite na področju umetne inteligence. Prehajanje med stanji sistema lahko razvijemo v graf ali mrežo stanj, ki jih med seboj povezujejo transformacije. V mreži sta dve stanji sosednji, če je med njima možna transformacija. Takšen graf mora predstavljati vse možne prehode med stanji in je kombinatorično zelo kompleksna struktura. Vendar je na tem področju opravljenega veliko raziskovalnega dela in algoritmi kot so *iskanje v globino*, *iskanje v širino* in A^* so samo osnovni predstavniki algoritmov, ki rešujejo problem iskanja poti skozi graf. Poleg tega se lahko poslužimo različnih pristopov *planiranja* [27]. Metode planiranja so podobno kot metode iskanja poti skozi prostor možnih stanj razvite na področju umetne inteligence. Planiranje je metoda, kjer se na podlagi

trenutnega stanja sistema odločamo za naslednji možni korak. Planer mora kot vhodni podatek dobiti podatek o trenutnem stanju sistema. Na podlagi strategije, zadanih ciljev in izpolnjevanja pogojev za prehode v naslednja možna stanja, planer določi naslednji prehod. Planer si lahko predstavljamo kot hevristično funkcijo pri iskanju poti skozi graf možnih stanj.

Omeniti je treba tudi podporo na področju modeliranja z različnimi diagrami stanj. Že standard UML ponuja *diagram stanj*, ki ga je mogoče preslikati v npr. *diagram aktivnosti* ali obratno. Obstaja kar nekaj računalniško podprtih orodij za risanje diagramov stanj, kot so npr. Microsoft Visio¹ in Eclipse².

Z obravnavo poslovnih procesov kot sistemov je mogoče poslovne procese modelirati precej učinkovito, hkrati pa je zaradi možnosti prevedbe takšne oblike predstavitve poslovnega procesa na znane probleme iz različnih raziskovalnih področij, možno uporabiti dobro definirane in razvite načine optimizacije. Vprašanje je, kako intuitiven je takšen način modeliranja. Najprej se moramo spomniti, da je procesna usmerjenost osredotočena na *tok* informacij in aktivnosti. Z uporabo systemskega pristopa zato odstopamo od splošno uveljavljenega dojemanja poslovnih procesov. Vendar pa se izkaže, da je modeliranje toka v ad-hoc procesu zaradi visoke dinamike zelo neprikladno, saj ta tok ni definiran vnaprej. Zato ocenjujemo, da je systemski pristop k obravnavi ad-hoc procesov dobra alternativna rešitev.

6.2 Uporaba ECA pravil

Z ECA pravili [35, 19, 24] označujemo pravila, ki se za krmiljenje in izvajanje uporabljajo v dogodkovno usmerjeni arhitekturi. Dogodkovno usmerjena arhitektura [13] je splošno uporabljen arhitekturni vzorec za sisteme, pri katerih je glavni poudarek na proženju, detekciji in obravnavi dogodkov. Dogodek je običajno definiran kot neka sprememba v stanju sistema, ki vpliva na izvajanje pravil in aktivnosti. Poenostavljeno rečeno je dogodek vsaka opazna sprememba ali dogajanje znotraj in zunaj sistema. V dogodkovno usmerjeni arhitekturi je dogodek glavna entiteta, od katere je odvisno nadaljnje dogajanje v sistemu.

ECA pravila so bila najprej predmet raziskav v aktivnih podatkovnih bazah. Aktivne podatkovne baze so glede na spremembe, vnos in izbris podatkov, sposobne reagirati z ustreznimi baznimi procedurami. Skrbnik podatkovne baze vnaprej določi, kakšni so pogoji za sprožitev procedur, aktivna baza pa sama prevzame nalogo, da takšne pogoje zazna in procedure proži. Koncept ECA pravil se je kmalu uveljavil tudi na drugih področjih. Uporabna se izkažejo predvsem v sistemih, ki so sestavljeni iz šibko sklopljenih in neodvisnih komponent. Vsaka takšna neodvisna komponenta je sama zadolžena, da spremlja dogodke in na njih ustrezno reagira [19]. ECA pravila so primerna tudi za

¹Microsoft Visio je del programskega paketa Microsoft Office in podpira risanje različnih standardnih diagramskih tehnik.

²Eclipse je odprtokodno orodje za razvoj programske opreme. V sklopu razširitev podpira risanje diagramov UML.

asinhrono izvajanje, kjer se izvajanje sproži ob izpolnjevanju pogojev, tok izvajanja pa je neodvisen od sprožitelja. Takšen primer je asinhrona obdelava izjem, ki se zgodijo med izvajanjem programov.

Za generiranje, prenos in obdelavo dogodkov, je v dogodkovno usmerjeni arhitekturi definirana infrastruktura iz štirih logičnih nivojev:

Izvor dogodkov Vsak dogodek ima svoj izvor, tj. mesto nastanka. To so lahko različne aplikacije, senzorji, klici iz drugih poslovnih procesov, ročno sproženi dogodki, itn. Dogodki imajo določeno vsebino, ki opisujejo njihov pomen. Če oblika vsebine ni standardizirana, moramo dogodke pred vstopom v dogodkovno infrastrukturo translirati v standardizirano obliko.

Dogodkovno vodilo Dogodkovno vodilo je komunikacijski kanal med ostalimi tremi nivoji dogodkovne infrastrukture. Po dogodkovnem vodilu potujejo dogodki od svojega izvora do svojega ponora. Dogodkovno vodilo lahko služi tudi kot translator različnih oblik dogodkov v standardizirano obliko, vendar se običajno uporablja zunanji translator, da ne bi mešali odgovornosti med nivoji infrastrukture.

Obdelava dogodkov Ko pride dogodek po dogodkovnem vodilu do obdelave dogodkov, se na njem uporabijo ECA pravila. Glede na dogodek se izberejo ustrezna pravila, ki se na ta dogodek nanašajo. Za takšna pravila rečemo da se *prožijo*. Posledica proženja pravil je izvedba neke reakcije na dogodek. Ta je lahko sprožitev novega dogodka, sprožitev novega procesa, sprememba stanja sistema, itn.

Dogodkovno vodene aktivnosti Dogodkovno vodene aktivnosti so aktivnosti, ki so posledica proženja pravil zaradi določenega dogodka. Gre za različne aktivnosti, ki se izvajajo asinhrono in brez časovne vezave na dogodke. Dogodki te aktivnosti samo *sprožijo* in ne sodelujejo pri nadzoru njihovega nadaljnjega izvajanja.

Če se osredotočimo na ECA pravila, lahko njihovo delovanje razčlenimo na:

Pojav dogodka Dogodek, ki nosi informacijo, se pojavi na dogodkovnem vodilu. V primeru uporabe centralizirane obdelave dogodkov se tja stekajo vsi dogodki, pri šibko sklopljeni arhitekturi pa je bolj pogosta uporaba lokalne obdelave dogodkov, kjer je vsaka komponenta sistema sama zadolžena za spremljanje dogodkovnega vodila in proženje lastnih pravil.

Logično preverjanje ustreznosti dogodka Med obdelavo dogodkov se preveri logična ustreznost dogodka za vsako pravilo. Pravila so običajno sestavljena iz *pogoja* in *akcije*. S pogojem preverimo, ali dogodek ustreza pravilu. Če je pogoj izpolnjen, se pravilo sproži in izvede se akcija.

Izvedba akcij Akcija je odgovor na dogodek. Akcija lahko obsega poljuben koncept. Lahko vrne nov dogodek na dogodkovno vodilo, lahko sproži nove aktivnosti, spremeni stanje sistema, itn.

Razlog za študij ECA pravil izvira iz dejstva, da lahko z njimi učinkovito dosežemo šibko sklopljenost in neodvisnost med komponentami sistema. V Poglavju 6.1 smo z obravnavo sistemov nakazali možnost, da poslovne procese opisujemo kot sisteme. Ker so aktivnosti ad-hoc procesa med seboj zaradi nedoločenosti vrstnega reda njihovega izvajanja neodvisne, se mora to pri abstrakciji s sistemom odražati kot neodvisnost komponent sistema. Takšne neodvisne komponente med seboj komunicirajo preko dogodkovnega vodila s pošiljanjem notranjih dogodkov, prav tako pa se lahko odzovejo na dogodke, ki pridejo iz zunanosti sistema. Z ECA pravili se torej preverja, katera komponenta sistema bo izvedla katero akcijo.

Podoben pristop lahko uporabimo pri modeliranju poslovnega procesa z diagramom stanj. Vsakemu stanju določimo samostojna ECA pravila, ki so zadolžena za upravljanje tega stanja. Če pride do določenega dogodka, ki ustreza ECA pravilom, se to odrazi kot sprožitev akcij v določenem stanju in prehod iz enega stanja v drugega. V primeru, da diagram stanj razvijemo v graf možnih stanj, so ECA pravila vhodni in izhodni pogoji na posameznih vozliščih.

V ad-hoc procesu se izbor naslednje aktivnosti izvaja sproti v realnem času, torej je ad-hoc proces odvisen od časovne dimenzije. Ker so aktivnosti med seboj neodvisne, bi radi dosegli asinhrono izvajanje. Asinhronost in neodvisnost dosežemo z uporabo ECA pravil, ki skrbijo za izvajanje posameznih neodvisnih aktivnosti. ECA pravila lahko določajo, katera aktivnost je v določenem trenutku primerna za izvajanje in to neodvisno od ostalih aktivnosti.

Kot smo si zadali že med cilji magistrske naloge v Poglavju 1.2, mora biti predlagana modelirna tehnika za ad-hoc procese v praksi izvedljiva z današnjo IKT. ECA pravila so za implementacijo zelo primerna, saj je danes na voljo kar nekaj sistemov za upravljanje poslovnih pravil, kar v osnovi ECA pravila tudi so.

ECA pravila se izkažejo za dober koncept pri modeliranju ad-hoc procesov. Z njimi lahko v praksi dosežemo to, da se vrstni red aktivnosti določa sproti tekom izvajanja ad-hoc procesa v odvisnosti od stanja sistema. Takšen način modeliranja pa že odraža realno stanje dinamičnih poslovnih procesov, kjer je izvajanje aktivnosti odvisno od razpoložljivosti virov in notranjih ter zunanjih dogodkov.

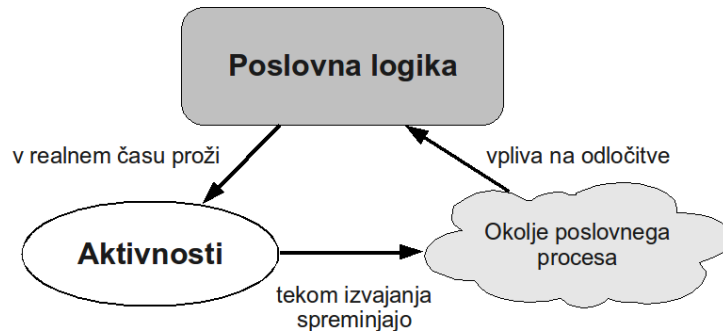
Naš zaključek je, da je koncept ECA pravil, v kombinaciji z ustrežno predstavitvijo aktivnosti poslovnega procesa in poslovnega procesa samega, pravi korak v smeri obravnave dinamike poslovnih procesov.

6.3 Kognitiven pristop

Temeljni poslovni procesi so običajno produktno usmerjeni in so zato s tehničnega vidika lažje določljivi. Če v nekem poslovnem procesu izdelujemo izdelek, lahko proces (učinkovitega) sestavljanja izdelka zapišemo na relativno malo možnih načinov. Okolje takšnega poslovnega procesa je dokaj kontrolirano in statično. V nasprotju s tem se podporni poslovni procesi odvijajo v bolj kaotičnem in dinamičnem okolju. Eden izmed namenov podpornih procesov je ustvarjanje takšnega okolja za temeljne poslovne procese,

v katerih bodo ti najlažje in najbolj učinkovito izdelovali dodano vrednost. Gre torej za urejanje in ustvarjanje ustreznih pogojev *okolja poslovnih procesov*. In ravno kognitiven pristop [35] uvaja percepcijo okolja, na podlagi katere se v realnem času vršijo odločitve o izvajanju aktivnosti poslovnih procesov.

Kognitiven pristop se zgleduje po sistemskemu pristopu k modeliranju poslovnih procesov in uporabi ECA pravil. V primerjavi z običajnim, procesno usmerjenim pristopom modeliranja poslovnih procesov, razširja procesno logiko v poslovno logiko. Pri običajnem modeliranju in izdelavi delovnih postopkov se običajno osredotočamo na procesno logiko, ki je zadolžena za tvorjenje zaporedja aktivnosti. Zaporedje aktivnosti je v tem primeru odvisno od zaključka in rezultatov trenutno končane aktivnosti in začetka ter vhodov naslednje možne aktivnosti. Lahko rečemo, da procesna logika deluje lokalno na nivoju aktivnosti in njihovih lastnosti. V nasprotju s tem pa naj bi se pri kognitivnem pristopu osredotočali na izvajanje aktivnosti glede na stanje okolja. Okolje je določeno s prostorom, kjer se poslovni proces izvaja. Podobno kot pri sistemskem pristopu, so komponente okolja viri in ljudje. Bistvo kognitivnega pristopa je poznavanje stanja okolja v vsakem trenutku. Ko se izvajanje neke aktivnosti zaključi, se na podlagi stanja okolja v realnem času odloči za naslednjo aktivnost. Procesna logika iz običajnega procesno usmerjenega pristopa je tako razširjena na poslovno logiko.



Slika 6.1: Razmerja in vloge poslovne logike, aktivnosti in okolja pri kognitivnem pristopu [35]

Na Sliki 6.1 vidimo, kako potekajo odločitve za določanje vrstnega reda aktivnosti v realnem času. V nasprotju z definicijo poslovnega procesa, kjer je zaporedje aktivnosti definirano, pri kognitivnem pristopu temu ni tako. Poslovna logika spremlja stanje okolja in se na podlagi tega odloča, katero aktivnost bo sprožila v naslednjem koraku. V poslovni logiki je zbrano znanje o okolju in omejitvah okolja. Ko se v določenem trenutku ugotovi stanje o okolju, lahko s poslovno logiko ugotovimo, katere aktivnosti se lahko trenutno izvedejo. Poleg tega mora poslovna logika vsebovati znanje o omejitvah, medsebojnih odvisnostih in potrebnih vhodih in izhodih oz. rezultatih aktivnosti. S poslovno logiko zagotovimo, da je zaporedje izvajanja aktivnosti smiselno. Bistveno je, da se odločitve o izvajanju aktivnosti sprejemajo na podlagi spremljanja stanja okolja.

Takšen pristop zelo ustreza obravnavi ad-hoc procesov. Kognitiven pristop ne določa zaporedja aktivnosti vnaprej, temveč izbira aktivnosti v realnem času. Takšna rešitev

popolnoma ustreza ad-hoc procesom, kjer zaradi visoke notranje dinamike ne moremo vnaprej določiti zaporedja izvajanja. Hkrati pa takšen pristop zelo dobro obravnava tudi spremembe, ki prihajajo iz zunanosti sistema. S tem mislimo na spremembe, kot so npr. spremenjen način poslovanja ali novi razpoložljivi viri. Takšne spremembe lahko zajamemo na dva načina. Lahko spremenimo poslovno logiko, ki bo od trenutka potrditve spremembe izbirala aktivnosti glede na novo znanje. Lahko pa spremembe obravnavamo kot spremembo komponent okolja. Takoj, ko dodamo ali odstranimo komponento iz okolja, bo poslovna logika še vedno delovala na obstoječem okolju in bo tako sprememba upoštevana.

Običajen pristop k implementaciji poslovne logike pri kognitivnem pristopu je uporaba repozitorijev poslovnih pravil. Ti so lahko razpršeni in vsebujejo pravila po kategorijah uporabe ali pa uporabljamo samo en centralni repozitorij, v katerem so zbrana vsa poslovna pravila. Pri običajnih pristopih modeliranja poslovnih procesov sta poslovna in procesna logika običajno ločeni kategorij, pri kognitivnem pristopu pa poslovna logika razširja procesno in zato potrebe po eksplicitni procesni logiki ni več.

Kognitiven pristop po našem mnenju združuje ugotovitve iz systemskega pristopa, uporabe ECA pravil, in še iz drugih področij, ki probleme obravnavajo na celostni način. Kognitiven pristop se ne osredotoča na posamezne aktivnosti poslovnih procesov, temveč na celotno sliko okolja, v katerem potekajo procesi. Danes, ko so združbe že same po sebi kompleksni sistemi, hkrati pa delujejo v hitro spreminjajočem se okolju, je takšen pristop najbrž edini možen, da zagotovimo ustrezno obravnavo dinamike in sprememb v poslovnih procesih.

6.4 Agentni sistemi

Sistemske in kognitivne pristope modeliranja poslovnih procesov temeljita na konceptu, da poslovni procesi potekajo v nekem okolju ali sistemu. Poslovni procesi so entitete, ki spremljajo dogajanje v sistemu in glede na stanje sistema oz. okolja izvajajo aktivnosti. Gre torej za neko vrsto avtonomnega izvajanja posameznega poslovnega procesa. Za vodenje takšnih avtonomnih procesov so zelo primerni agentni sistemi.

Agentni sistem je sistem ali okolje, v katerem delujejo agenti. Agenti [27, 35, 19] lahko glede na stanje sistema izvajajo akcije, ki so lahko interakcije z okoljem ali drugimi agenti. Agent je samostojna entiteta, ki ima določeno stopnjo avtonomije in jih ni potrebno eksplicitno klicati oz. aktivirati. Podobno kot govorimo pri objektu usmerjenem razvoju o objektih, ki enkapsulirajo podatke in metode, pri agentih govorimo o entitetah, pri katerih nas bolj kot njihova struktura zanima njihovo življenje, delovanje in obnašanje v sistemu.

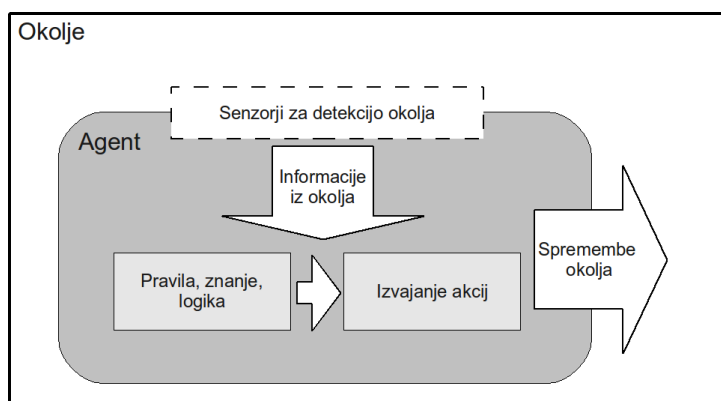
Poglavitne lastnosti agentov naj bi bile naslednje:

Avtonomija Agenti so za razliko od programov ali aplikacij, ki jih moramo upravljati, avtonomni. To pomeni, da ko se jih enkrat ustvari, jih lahko obravnavamo kot samostojne procese ali entitete, ki glede na njihov namen opravljajo naloge.

Obstojnost Agenti v času svojega obstoja v sistemu več ali manj samostojno opravljajo svoje naloge. Ko agenta ustvarimo, bo v sistemu obstajal in ne bo potreboval dodatnega nadzora. Agentu ni treba eksplicitno klicati in uporabljati njegovih metod, temveč v času njegovega obstoja sam poskrbi za svoje delovanje.

Interoperabilnost Ker so agenti samostojne entitete, morajo dojemati okolje v katerem delujejo. Poleg dojetja okolja, pa morajo biti sposobni tudi medsebojne komunikacije. Zato lahko na agentni sistem gledamo kot na platformo, ki agentom ponuja ustrezno podlago za komuniciranje in sodelovanje.

Reaktivnost Ker je agent samostojna entiteta, katere delovanje naj ne bi bilo odvisno od našega neposrednega nadzora, morajo biti sposobni reagirati na dogodke v njihovem okolju. Agenti lahko reagirajo na različne tipe klicev, na dogodke v sistemu, ali spremembe stanj sistema.



Slika 6.2: Shema preprostega agentnega sistema z enim agentom

Na Sliki 6.2 vidimo preprosto shemo agentnega sistema z enim agentom. Agent kot samostojna entiteta spremlja dogajanje v njegovem okolju. Glede na to dogajanje in na znanje, ki ga vgradimo v agenta, agent samostojno izvaja akcije. Posledica akcij so spremembe v okolju, te pa zopet vplivajo na aktivacijo agentov.

Glede na poudarjene lastnosti poznamo več vrst agentov, od katerih bi za nadaljnjo obravnavo izpostavili naslednje:

Inteligentni agenti Z razvojem inteligentnih agentov se ukvarja predvsem področje umetne inteligence. Bistvo inteligentnih agentov je njihova sposobnost, da se lahko prilagajajo spremembam v okolju in da se znajo iz njihovih preteklih akcij učiti in tako sproti ter samostojno spreminjati svoje obnašanje.

Mobilni agenti Mobilni agenti so postali aktualni predvsem z razvojem svetovnega spleta in zmogljive mrežne opreme. V takšnem okolju se lahko agentni sistem razteza preko več fizičnih okolij, ki so med seboj povezana z omrežjem. Tako lahko

mobilni agenti menjajo okolje svojega delovanja in se selijo med različnimi fizičnimi okolji. Mobilne agente si lahko predstavljamo tudi kot agente, ki za svoje akcije uporabljajo spletne storitve, ki se nahajajo na različnih in geografsko razpršenih strežnikih.

Več-agentni sistemi Več agentni sistemi sicer niso vrsta agentov, ampak agentni sistem, v katerem lahko naenkrat deluje več agentov hkrati. Agenti so samostojni in morajo komunicirati z okoljem, poleg tega pa znajo sodelovati tudi med sabo.

Pri sistemskem in kognitivnem pristopu lahko poslovni proces modeliramo z agentom. Agent se instancira ob začetku procesa, njegova življenjska pot pa se zaključi, ko se proces konča. Tekom svojega obstoja komunicira z okoljem in glede na stanja okolja izvaja akcije, ki predstavljajo aktivnosti poslovnega procesa. Podoben pristop je predlagan v [27], kjer agenti samostojno vodijo poslovne procese v okolju, ki je sestavljeno iz virov in omejitev.

Takšen pristop je zelo primeren pri modeliranju ad-hoc poslovnih procesov, saj določanje vrstnega reda aktivnosti prepustimo agentu. Agent kot ad-hoc proces se izvaja in ob koncu vsake aktivnosti procesa preveri, katere naslednje aktivnosti glede na trenutno stanje procesa naj izvede, da bo čim prej dosegel cilj. Agent lahko za takšno planiranje uporabi različne hevristične funkcije ali tehnike planiranja. Za iskanje najbolj optimalnega zaporedja lahko pred začetkom vsake aktivnosti izvede simulacijo oz. iskanje po prostoru možnih potekov procesa (glej Poglavje 6.1). Okolje za agenta predstavlja omejitve kar se tiče razpoložljivosti virov. V primeru da so viri, potrebni za neko aktivnost v trenutku izbiranja zasedeni, se lahko agent odloči za izbor druge aktivnosti, ki jo je v tistem trenutku možno izvajati.

Takšen koncept se izkaže za koristnega pri modeliranju sočasnih ad-hoc procesov. Če se v sistemu izvaja več ad-hoc procesov hkrati, ti zasedajo vire, ki so običajno tudi skupni. Pri navadnih poslovnih procesih bi v takšnem primeru predhodno sestavili optimizacijsko nalogo in poizkusili določiti takšen vrstni red izvajanja aktivnosti v sočasnih procesih, da si bodo ti med sabo kar najmanj zasedali vire. Pri ad-hoc procesih pa takšne optimizacije predhodno ne moremo izvesti, saj ne poznamo zaporedja izvajanja aktivnosti. Zato morajo biti agenti dovolj inteligentni, da znajo sami izbirati takšne vrstne rede aktivnosti, ki bodo čim manj v konfliktih z drugimi agenti.

Dobra stran uporabe agentov je tudi ta, da v primeru spremembe okolja ni treba ponovno zagnati procesov, ampak bodo agenti sami zaznali spremenjeno stanje okolja in bodo temu primerno delovali naprej. Tako lahko spremembe, ki pridejo iz zunanosti, zelo hitro vnesemo v poslovanje združbe.

Vsi štirje alternativni pristopi k modeliranju poslovnih procesov pomembno prispevajo k spoznanjem, kako se lahko odmaknemo od striktno procesnega razmišljanja o poslovnih procesih. Vsi štirje pristopi omogočajo takšno modeliranje poslovnih procesov, ki upošteva dinamiko in interkacije tako znotraj kot zunaj okolja poslovnih procesov. Na podlagi teh razmišljanj in izdelavi konceptualne analize realnega primera ad-hoc procesa smo predlagali nov način modeliranja, ki se zgleduje po opisanih alternativnih pristopih, hkrati pa se izogne nekaterim pomanjkljivostim opisanih konceptov.

Poglavje 7

Predlagan koncept modeliranja ad-hoc procesov

Po zgledu obstoječih modelirnih konceptov in konkretnih potreb po izdelavi dobrega procesnega modela ad-hoc procesa smo razvili prilagojen pristop za modeliranje ad-hoc procesov. V poglavju je opisan pristop h konceptualnemu dojetanju ad-hoc procesov, kako se opiše njihovo okolje in kako smo pristopili k modeliranju tako notranje kot zunanje dinamike procesov. Predlagan koncept modeliranja služi kot dopolnilo ali razširitev k obstoječim modelirnim tehnikam.

7.1 Konceptualni model ad-hoc procesa

V našem pristopu k modeliranju ad-hoc procesov smo te procese opisali kot *poslovne procese*, ki se odvijajo v njihovem *okolju*. V nadaljevanju besedila bomo uporabljali izraz ad-hoc proces, čeprav lahko opisano modeliranje uporabimo za splošne poslovne procese. Okolje je definirano kot celoten prostor, v katerem potekajo ad-hoc procesi. To vključuje celotno združbo, v kateri se ad-hoc proces izvaja, vse vire in ljudi. Če ad-hoc proces prestopi meje združbe, lahko takšno izvajanje prikažemo kot izvajanje izven okolja ali kot izvajanje v razširjenem okolju ad-hoc procesov.

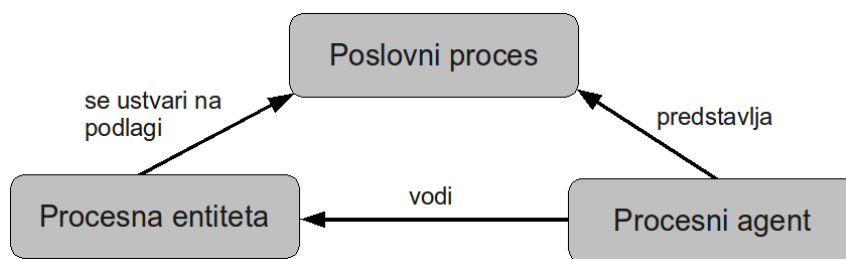
Kot smo opisali že v Poglavju 6.1, obravnava ad-hoc procesov kot sistemov omogoča opis kompleksnih procesov s celostnega vidika. Zato smo ta pristop uporabili pri modeliranju ad-hoc procesov. Sistemski pristop smo uporabili na dveh nivojih:

Model okolja kot sistem Celotno okolje je predstavljeno kot sistem. Sistem je neka zaključena celota komponent in transformacij, ki povzročajo prehajanje sistema med različnimi stanji. Sistem lahko zavzame končno mnogo stanj, ki smo jih za potrebe magistrske naloge definirali v diskretnem prostoru.

Model ad-hoc procesa kot sistem Ad-hoc proces je prav tako nek sistem ali entiteta, ki tekom izvajanja procesa zavzema različna notranja stanja. Stanje ad-hoc procesa je definirano kot obdobje v izvajanju poslovnega procesa, v katerem se izvaja omejen

nabor aktivnosti, ki vodijo k doseganju določenega vmesnega cilja. Ad-hoc proces med stanji prehaja tako, da si zada nov vmesni cilj, za katerega mora izvesti množico aktivnosti v vnaprej nedefiniranem zaporedju.

Ker je izvajanje poslovnih procesov v splošnem asinhrono¹ in vzporedno, jih bomo opisali kot samostojne *procesne entitete*. Vsaka procesna entiteta predstavlja en poslovni proces v izvajanju. Lahko rečemo, da je procesna entiteta *instanca ad-hoc procesa*. Ker so procesne entitete samostojne, jih moramo voditi vzporedno in medsebojno neodvisno. Takšen opis popolnoma ustreza delovanju agentov. Zato smo v našem pristopu predlagali, da ad-hoc procese kot procesne entitete vodijo *procesni agenti*.



Slika 7.1: Shema razmerja med ad-hoc procesom, procesno entiteto in procesnim agentom

Ko v okolju pride do zahteve za izvajanje ad-hoc procesa, se ustvari procesna entiteta, katere vodenje prevzame procesni agent. Vsak procesni agent vodi eno procesno entiteto oz. en ad-hoc proces. Skozi izvajanje ad-hoc procesa mora procesni agent zaznavati stanje okolja procesne entitete in glede na to ustrezno usmerjati njeno delovanje oz. izbor vmesnih ciljev.

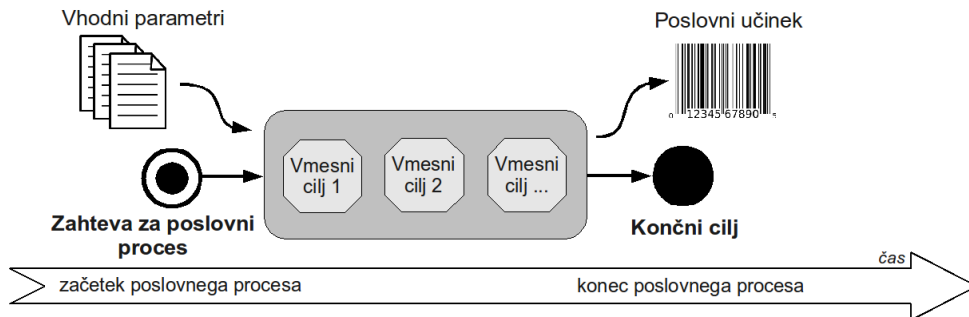
Obravnava ad-hoc procesov kot sistemov tako na nivoju celotnega okolja kot na nivoju ad-hoc procesa omogoča zajem notranje in zunanje dinamike poslovnega procesa (glej Poglavje 2.6). V primeru, da pride do sprememb v združbi, lahko to ustrezno modeliramo s spremembami v definiciji komponent okolja. Če pride do sprememb v načinu izvajanja poslovnega procesa, lahko to opišemo s spremembami v procesnih entitetah in v delovanju procesnih agentov.

Vsak poslovni proces ima po definiciji definiran začetek z določenim vhomom in konec z določenim izhodom (glej Poglavje 2.2). Začetek poslovnega procesa bomo opisali kot *zahtevo za izvajanje procesa*. Zahtevo za izvajanje procesa mora spremljati ustrezno definiran vhod, ki predstavlja *vhodne parametre* poslovnega procesa. Konec izvajanja poslovnega procesa bomo opisali kot *končni cilj* poslovnega procesa. Ob doseganju končnega cilja mora biti seveda kot rezultat procesa izdelan *učinek poslovnega procesa*. Učinek predstavlja neko dodano vrednost - to je lahko izdelek ali storitev.

Od zunaj je ad-hoc proces definiran še vedno tako kot običajen poslovni proces - z začetkom in koncem in pripadajočim vhomom in izhodom. Ker ne moremo vnaprej

¹Z izrazom *asinhrono* mislimo na to, da so poslovni procesi med seboj neodvisni, če si med seboj ne izključujejo virov, in se jim tako za izvajanje ni treba posebej časovno usklajevati.

definirati zaporedja aktivnosti v ad-hoc procesu, pa lahko z namenom znižanja entropije določimo *vmesne*, delne cilje, ki jih mora procesna entiteta za uresničitev končnega cilja izpolniti. Procesni agent se mora tekom izvajanja poslovnega procesa odločiti, h katerim vmesnim ciljem se bo usmeril.



Slika 7.2: Preprost prikaz začetka ad-hoc procesa, poteka procesa preko vmesnih ciljev in konec procesa z dosegom končnega cilja procesa

Vmesni cilji naj bi bili čimbolj neodvisni drug od drugega. Obenem je možnih več kombinacij doseganja vmesnih ciljev, ki pripeljejo do končnega cilja. Procesni agent se lahko v vsakem trenutku odloči, kateri vmesni cilj bo poizkusil uresničiti. Seveda mora ob tem upoštevati pogoje iz okolja, procesno in poslovno logiko.

S tem, ko se procesni agent odloči izpolniti določen vmesni cilj, postavi procesno entiteto v *stanje*, ki ustreza temu vmesnemu cilju. Stanje procesne entitete določa, katere pogoje v določenem trenutku izpolnjuje procesna entiteta in katere aktivnosti se lahko izvajajo.

Opisan način modeliranja ad-hoc procesov omogoča šibko sklopljenost komponent. Ad-hoc procesi delujejo vsak zase in pri tem uporabljajo vire in ljudi iz okolja. Na preprost način se da modelirati naključni vrstni red izvajanja aktivnosti z omejenimi nabori aktivnosti, ki se lahko izvajajo za doseg določenega vmesnega cilja. Takšen model lahko poljubno razčlenimo do takšnega nivoja, do kolikor se spleča oz. dokler ne pridemo do modeliranja psevdokode na stopnji posameznih aktivnosti.

7.2 Osnovni elementi modeliranja ad-hoc procesa

7.2.1 Okolje ad-hoc procesa

Z *okoljem* mislimo na prostor in čas, v katerem se izvajajo poslovni procesi. Običajno je okolje poslovnih procesov omejeno na izvajanje znotraj meja združbe oz. poslovnega sistema, vendar to ni vedno nujno. Zaradi odprtosti sodobnega poslovanja vse več poslovnih procesov prestopa meje med poslovnimi sistemi.

Že brez poudarjanja notranje dinamike poslovnih procesov je slika izvajanja procesov znotraj združbe zelo zapletena. Ker se v združbi običajno izvaja več poslovnih procesov, ki

uporabljajo in zasedajo različne vire, je okolje teh procesov treba dojemati kot kompleksno strukturo, ki daje poslovnim procesom na razpolago za njihovo izvajanje ustrezne vire. Viri so v vsakem okolju seveda omejeni tako s časom kot količino.

Za ustrezno modeliranje takšnega okolja je primeren sistemski pristop, ki omogoča modeliranje samostojnih komponent. Komponente so med seboj različno odvisne in različno vplivajo druga na drugo. Procesni agenti se za izvajanje ad-hoc procesov odločajo predvsem na podlagi stanja komponent, torej virov okolja. Hkrati pa mora okolje zagotavljati tudi omejitve za delovanje procesnih agentov. Omejitve lahko predstavljajo realno situacijo glede pravil in predpisov poslovanja.

Okolje ima vedno definirano *stanje*, ki pove, v kakšnem stanju so trenutno viri okolja. Stanje okolja se spreminja s transformacijami, ki so lahko notranje ali zunanje:

Notranje transformacije Notranje transformacije se dogajajo, ko si procesni agenti zadajo nov vmesni cilj. Pred vsako takšno izbiro mora procesni agent preveriti, če so za doseganje posameznega vmesnega cilja na voljo ustrezni viri. Procesni agent torej preveri stanje okolja. Ko si izbere naslednji vmesni cilj in tako spremeni stanje procesne entitete, s tem zasede določene vire. Na ta način se stanje okolja transformira v novo stanje, ki ga morajo upoštevati ostali procesni agenti ob njihovem odločanju za vmesne cilje.

Zunanje transformacije Zunanje transformacije se zgodijo, ko se okolje spremeni glede na zahtevo, ki ne pride od procesnih agentov, ampak kot poseg od zunaj. Takšna sprememba se npr. zgodi ob dodajanju ali ukinitvi kakšnega vira. Zunanje transformacije so lahko posledica spremembe pravil v načinu poslovanja in doseganja poslovnih učinkov. Lahko se zgodi, da se v asortiment poslovnih učinkov doda novo vrsto izdelkov ali storitev in je treba temu ustrezno prilagoditi poslovanje. Zunanje transformacije so lahko tudi posledica naknadne optimizacije v obnašanju procesnih agentov.

Meja okolja je predstavljena z mejo združbe in z mejo nadzora nad izvajanjem ad-hoc procesov. Ko ad-hoc proces prestopi mejo okolja, ne moremo več vplivati na njegovo izvajanje. To pomeni, da se ob prestopu meje nižanje entropije ustavi in da nadaljnega izvajanja ne moremo podpirati. Podpiramo lahko samo vrnitev izvajanja procesa nazaj znotraj meja okolja.

Okolje ad-hoc procesov definiramo kot sistem z naslednjimi komponentami:

$$E = (V, L, O, D)$$

Pri tem je V množica virov, ki so na voljo pri izvajanju ad-hoc procesov, L so človeški viri oz. množica ljudi, ki lahko sodeluje v ad-hoc procesih, O je množica izvajalnih omejitev, D pa je množica možnih dogodkov v celotnem okolju.

Množica virov

Kot viri so mišljeni vsi viri (razen človeških, ki so obravnavani posebej), ki so v okolju na voljo posameznim procesom tekom njihovega izvajanja. Naštejmo nekaj možnih vrst virov:

Storitve Ena izmed najbolj razširjenih vrst virov v sodobnih združbah so storitve. S storitvami mislimo na lokalne ali oddaljene storitve, ki ponujajo različne funkcionalnosti. Lokalne storitve izvirajo iz našega okolja in so lahko tudi novi poslovni procesi, zunanje storitve pa se nahajajo izven našega okolja. Takšno obliko virov dobro podpira danes razširjen in uveljavljen SOA pristop. Klicanje oddaljenih storitev je eden izmed načinov, kako ad-hoc proces prestopi mejo okolja.

Aplikacije Z aplikacijami mislimo na različne programske pakete in orodja, ki morajo biti na voljo za izvajanje ad-hoc procesov. Lahko bi poenostavili in aplikacije označili kot specializirane lokalne storitve, ki so zelo omejene kar se tiče razpoložljivosti. Aplikacije so lahko tudi del ERP sistema združbe.

Izvajalne platforme Za izvajanje aktivnosti je potrebna določena izvajalna platforma. Pri ad-hoc procesih, katerih vmesni cilji zahtevajo računsko zelo intenzivne operacije, je potrebno zagotoviti ustrezen procesorski čas in dovoljšnjo kvoto delovnega spomina. Podobno je pri obsežnih podatkovnih zbirkah in operacijah. Danes je za zagotavljanje izvajalnih platform zelo uporaben koncept virtualiziranih platform, nameščenih v sistemih, ki temeljijo na konceptu računalništva v oblaku. Virtualizirane izvajalne platforme lahko naročamo in uporabljamo po naročilu oz. glede na trenutno potrebo.

Strojni viri Pri ad-hoc procesih, ki so izdelčne narave², je treba tekom izvajanja izdelati končni izdelek. Za izdelavo izdelka se uporablja določene strojne vire. Vzemimo za primer ad-hoc proces, kjer je treba v končni fazi natisniti obvestila za uporabnike na papir. Za to je potreben strojni vir v obliki tiskarskega stroja in ustrezna ostala oprema, ki je potrebna za pakiranje in razpošiljanje obvestil.

Človeški viri

Človeški viri so običajno ključni vir za poslovne procese. V kolikor se le da, se sicer zaradi možnosti avtomatizacije uporabi človeških virov izogibamo. Vendar je dejstvo, da pri podpornih poslovnih procesih človek vedno predstavlja ključni del izvajanja procesov.

Pri človeških virih, ki sodelujejo v ad-hoc procesih, moramo še posebej paziti na njihovo razpoložljivost in usposobljenost. Pri ostalih, nečloveških virih, je običajno lažje poskrbeti za skalabilnost in večjo razpoložljivost, saj lahko zaradi materialne narave obseg takšnih virov lažje povečamo. Pri človeških virih smo običajno bolj omejeni in to morajo procesni agenti pri izvajanju ad-hoc procesov upoštevati.

²S tem mislimo na ad-hoc procese, katerih cilj je izdelava nekega izdelka.

Pri običajnih pristopih modeliranja poslovnih procesov obstaja t.i. gradnik ali aktivnost *človeško opravilo*³[6, 9]. Človeška opravila predstavljajo točke v poslovnem procesu, kjer se strojno vodeno, običajno avtomatizirano izvajanje poslovnega procesa prekine in čaka na posredovanje uporabnika. Bistveno pri ad-hoc procesih je, da z ustreznim modeliranjem omogočamo dobro komunikacijo med uporabniki in čim hitrejšo opravljanje takšnih človeških opravil.

Množica omejitev

Omejitve se nanašajo na delovanje procesnih agentov in na možna stanja virov. Procesni agenti tekom izvajanja ad-hoc procesov z opravljanjem nalog za doseganje vmesnih ciljev zasedajo in sproščajo vire okolja. S tem povzročijo transformacije med stanji okolja. Z omejitvami povemo, kakšne transformacije so dovoljene. Praktičen primer omejitve je, v kolikšni meri lahko procesni agent izrabi nek vir. Zaradi boljše razpoložljivosti virov lahko omejimo porabo virov po posameznih procesnih agentih. Omejimo lahko tudi npr. čas uporabe virov. Predvsem pa se z omejitvami poskrbi, da procesni agent ne bi sprožil takšne transformacije, ki bi okolje privedla v nedovoljeno stanje. Nedovoljeno stanje je npr. uničenje virov, prevelika izraba virov, itn.

Množica omejitev je določena tudi s strani pravil poslovanja. Pravila poslovanja so določena s poslovníkom in uradnimi predpisi, ki se jih mora združba držati iz pravno formalnih razlogov. Pravila poslovanja določajo dovoljene in nedovoljene aktivnosti v ad-hoc procesih ter način izvajanja teh aktivnosti. Z množico dovoljenih transformacij okolja lahko določimo poslovna pravila, ki preprečujejo izvajanje nedovoljenih aktivnosti.

Z množico omejitev na nivoju okolja določimo, kaj lahko procesni agenti počnejo in kaj ne. Sicer obnašanje procesnih agentov kontroliramo tudi s poslovno logiko, ki jo imajo vgrajeno, vendar je treba dodatno zagotoviti pravilnost izvajanja ad-hoc procesov z omejitvami okolja, ki jih morajo procesni agenti upoštevati.

Dogodki

Okolje mora za učinkovito komuniciranje med procesnimi agenti zagotoviti ustrezen komunikacijski kanal, ki ga predstavlja dogodkovno vodilo okolja. Po dogodkovnem vodilu se prenašajo dogodki, ki enkapsulirajo komunikacijo procesnih agentov. Procesni agenti po vzoru ECA pravil na te dogodke reagirajo.

Poznamo *notranje* in *zunanje* dogodke. Notranje dogodke prožijo procesni agenti in so namenjeni medsebojnemu obveščanju. Procesni agent lahko npr. ob zasedbi nekega vira ostalim procesnim agentom to eksplicitno sporoči. Prav tako lahko notranje dogodke prožijo viri. Ko se nek vir sprosti, se lahko na dogodkovno vodilo sproži dogodek, ki nosi informacijo o sprostitvi vira. Procesni agenti, ki so čakali na ta vir, tako dobijo obvestilo in lahko nadaljujejo z izvajanjem ad-hoc procesa.

Komunikacija preko dogodkovnega vodila je asinhrona. To pomeni, da kdorkoli na dogodkovno vodilo sproži dogodek, lahko opravlja svoje delo naprej. Procesni agenti,

³Z besedno zvezo *človeško opravilo* mislimo na izraz, ki se v angleščini glasi *human task*.

ki se zanimajo za določen tip dogodkov, se na ta tip dogodkov *registrirajo* in ves čas prisluškujejo dogodkovnemu vodilu, če se je željen dogodek sprožil. Izvor dogodka torej po sprožitvi dogodka opravlja svoje aktivnosti naprej in ga ne zanima, komu je ta dogodek namenjen. Vsak procesni agent mora sam poskrbeti za registracijo na njemu pomembne vrste dogodkov.

Vsi naštetih viri imajo določeno stopnjo razpoložljivosti. Procesni agenti za doseganje vmesnih ciljev potrebujejo vire, da lahko izvajajo aktivnosti. Pri izvajanju aktivnosti vire *zasedejo*. Zasedanje virov lahko izvedejo na dva načina:

Polna zasedenost Takšen vir je lahko samo ali zaseden ali prost. Gre torej za binarno lastnost. Če je takšen vir zaseden s strani neke aktivnosti, morajo ostali procesni agenti, ki ta vir potrebujejo, čakati na njegovo sprostitev. Če takšen vir predstavlja ozko grlo v skupnem izvajanju ad-hoc procesov, je treba dodati več instanc tega vira.

Delna zasedenost Takšen vir lahko uporablja več aktivnosti naenkrat. To pomeni, da vsak procesni agent zasede samo del zmogljivosti vira. V primeru, da je takšen vir vseeno ozko grlo pri skupnem izvajanju ad-hoc procesov, je treba povečati zmogljivost takšnega vira.

7.2.2 Končni in vmesni cilji ad-hoc procesa

Ker je po osnovni definiciji (glej Poglavje 2.2 poslovni proces razdeljen na aktivnosti, se običajni pristopi modeliranja osredotočajo na elemente, ki predstavljajo aktivnosti. Pri modeliranju ad-hoc procesov pa se ne moremo osredotočati na aktivnosti, ker dejansko ne vemo, v kakšnem zaporedju se bodo aktivnosti tekom izvajanja ad-hoc procesa izvedle.

Da vseeno opišemo potek ad-hoc procesov, jih razčlenimo na *vmesne*, delne cilje, ki morajo biti izpolnjeni, da ad-hoc proces v končni fazi doseže *končni* cilj, ki predstavlja konec procesa (glej Sliko 7.2):

Končni cilj ad-hoc procesa Vsak poslovni proces ima definiran začetek in konec. S koncem mislimo na trenutek, ko se izvedejo vse aktivnosti poslovnega procesa in pride do rezultata poslovnega procesa. Pri ad-hoc procesih ne vemo vnaprej, katere aktivnosti vse se bodo izvedle in potemtakem ne moremo kot konec procesa označiti trenutek, ko se izvedejo vse aktivnosti. Zato definiramo *končni cilj* ad-hoc procesa, ki predstavlja namen in konec procesa. Za ad-hoc proces rečemo da je končan, ko je dosežen končni cilj procesa.

Vmesni cilji ad-hoc procesa Pri običajnem pristopu modeliranja poslovnih procesov notranje izvajanje razdelimo na aktivnosti, ki se izvajajo v določenem zaporedju. Ker pri ad-hoc procesih ne moremo predvideti ne zaporedja na končne množice aktivnosti, lahko proces za obvladovanje entropije tekom izvajanja ad-hoc procesa razdelimo na *vmesne cilje*. Z vmesnimi cilji omejimo posamezne množice aktivnosti, ki se bodo izvajale v paketu za dosego vmesnih ciljev. Na ta način vemo, kaj se v ad-hoc procesu trenutno dogaja in lahko nudimo ustrezno podporo.

Končni cilj ad-hoc procesa je dosežen, ko je izpolnjen namen procesa. Namen procesa je nek poslovni učinek oz. dodana vrednost. V primeru, da je poslovni učinek ad-hoc procesa nek končni izdelek, lahko rečemo, da je končni cilj dosežen takrat, ko je izdelava izdelka končana. Končni cilj torej vedno spremlja poslovni učinek, ki je lahko vhod v naslednji poslovni proces ali pa že končna vrednost za uporabnika.

Vmesni cilj je abstrakcija notranje dinamike ad-hoc procesov. Ker ne vemo vnaprej, katere aktivnosti se bodo tekom izvajanja ad-hoc procesa izvedle, jih "zapakiramo" v omejene podmnožice, s katerimi se doseže vmesne cilje. Za modeliranje ad-hoc procesov mora biti omogočena vsaj identifikacija vmesnih ciljev. Če ne moremo določiti niti teh, govorimo o popolnoma dinamičnem procesu, ki ga lahko modeliramo samo na nivoju kontekstnega procesnega modela, kjer so navedeni procesi z njihovimi vhodi in izhodi. Vsaka omejena množica aktivnosti, ki jih je za uresničitev enega vmesnega cilja treba izvesti, običajno predstavlja neko logično celoto opravil. Bistveno je to, da v tej množici vrstni red aktivnosti ni določen. S stališča modeliranja ad-hoc procesa nas samo zanima, kateri vmesni cilji morajo biti doseženi. Nedoločenost vrstnega reda izvajanja aktivnosti torej abstrahiramo s tem, da jih samo pripišemo vmesnim ciljem, ne zanima pa nas točno, kako se bodo te aktivnosti izvedle. Vmesni in končni cilji so odgovor na vprašanje, *kaj je treba narediti*, da bo dosežen namen ad-hoc procesa. Poleg tega lahko že tekom modeliranja ugotovimo, *kateri viri in pogoji* so potrebni za doseganje posameznega vmesnega cilja in tako omogočimo proces optimizacije ad-hoc procesov. Usmerjenost v aktivnosti smo zamenjali z usmerjenostjo v vmesne cilje in tako odpravili potrebo po prikazu zaporedja izvajanja aktivnosti, ki v ad-hoc procesih ni določeno vnaprej.

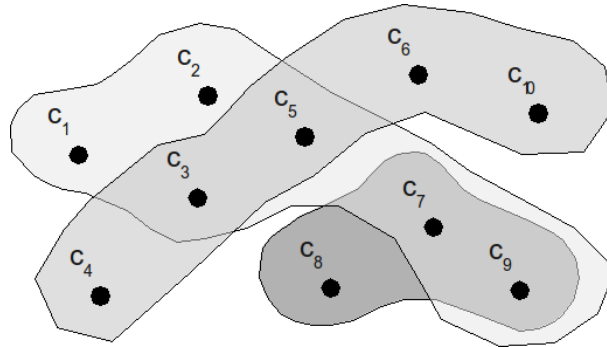
Stopnja razgradnje vmesnih ciljev

Vmesni cilji so oporne točke, s katerimi ad-hoc proces razdelimo na možne poteke izvajanja. Ko določamo vmesne cilje v ad-hoc procesu se moramo vprašati, do katerega nivoja lahko še predvidevamo strukturo procesa. Podobno kot pri običajnem pristopu modeliranja poslovnih procesov te razgradimo do poljubnega nivoja podrobnosti, lahko tudi pri ad-hoc procesih določimo različno podrobne vmesne cilje. Pri običajnem modeliranju poslovnih procesov se razgradnja procesa na posamezne aktivnosti konča na tisti stopnji, kjer moramo za opis aktivnosti uporabiti psevdokodo.

Podobno vodilo uporabljamo tudi pri razgradnji vmesnih ciljev. Vmesne cilje razdelujemo do nivoja, na katerem vrstni red aktivnosti, ki so potrebne za doseg vmesnega cilja, še ni pomemben. Takoj ko pridemo do stopnje, kjer bi morali za doseg vmesnega cilja definirati množico aktivnosti z *določenim* vrstnim redom, se modeliranje ad-hoc procesa konča. V slednjem primeru bi morali namreč uporabiti običajen pristop za modeliranje poslovnih procesov, vendar bi bil izdelan diagram zaradi upoštevanja velikega števila različnih možnih vrstnih redov izvajanja aktivnosti neberljiv.

Vrstni red izvajanja vmesnih ciljev

Ena izmed glavnih lastnosti ad-hoc procesov je nedoločenost vrstnega reda izvajanja aktivnosti. S tem, ko smo uvedli vmesne cilje, smo določili kontrolne točke, ki se morajo izvesti za doseg končnega cilja ad-hoc procesa. Ne zanima nas konkreten vrstni red izvajanja aktivnosti znotraj doseganja vmesnega cilja, temveč samo, ali je nek vmesni cilj dosežen.



Slika 7.3: Različne možnosti doseganja končnega cilja, prikazane s hiper-grafom [21]

Končni cilj se da doseči z uresničitvijo različnih kombinacij vmesnih ciljev. Ker je okolje kompleksno in ponuja različne vire, lahko ad-hoc proces za doseg končnega cilja ubere različne poti.

Na Sliki 7.3 so vmesni cilji prikazani z vozlišči nepovezanega grafa H , ki so povezani s hiper-povezavami iz množice vseh hiper-povezav E_H . Vsaka hiper-povezava označuje eno možno množico vmesnih ciljev, katerih uresničitve pripelje do uresničitve končnega cilja. Načinov, kako doseči končni cilj ad-hoc procesa, je običajno končno mnogo in jih lahko definiramo vnaprej. S takšnim pristopom odgovarjamo na vprašanje, *kaj vse je treba narediti, da se doseže končni cilj*, ne osredotočamo pa se na način in vrstni red doseganja vmesnih ciljev.

Formalno gledano lahko uresničitev končnega cilja ad-hoc procesa zapišemo kot disjunkcijo izjavnih računov:

$$C_k = \bigvee_{e \in E_H} F(e)$$

Pri tem je C_k končni cilj, E_H množica hiper-povezav oz. množica različnih možnih kombinacij vmesnih ciljev, ki vodijo k uresničitvi končnega cilja, $F(e)$ pa je izjavni račun nad posamezno kombinacijo vmesnih ciljev, ki vodijo k uresničitvi končnega cilja. Končni cilj je dosežen, ko je vsaj en izjavni račun $F(e)$ resničen.

Vmesni cilji v posamezni kombinaciji naj bi bili med seboj čim bolj neodvisni. Neodvisni naj bi bili na dva načina:

Časovno Dobro je, da lahko vsak vmesni cilj dosežemo ne glede na to, ali je bil prej dosežen kakšen drug vmesni cilj. Takšna sestava vmesnih ciljev omogoča, da se

lahko v kateremkoli trenutku procesni agent odloči za doseganje tistega cilja, ki je v tistem trenutku glede na stanje okolja sploh dosegljiv. Če so cilji med seboj časovno odvisni, potem je procesni agent pri izbiri naslednjega vmesnega cilja omejen tudi z množico že doseženih ciljev.

Glede zasedanja virov Dobro je, da vmesni cilji zahtevajo izvajanje tistih aktivnosti, ki si med seboj čim manj zasedajo vire. S tem se doseže višjo stopnjo vzporedno izvajanih aktivnosti in posledično možnost izvajanja več ad-hoc procesov hkrati.

V praksi se izkaže, da so izjavni računi $F(e)$ večinoma osnovne konjunkcije⁴. To pomeni, da je za vsako kombinacijo vmesnih ciljev, ki prepelje do končnega cilja, pomembno samo, da so ti vmesni cilji izpolnjeni. Pri tem predpostavljamo časovno neodvisnost med posameznimi cilji. Modeliranje ad-hoc procesov se z uporabo osnovnih konjunkcij bistveno poenostavi in omogoča boljše razumevanje procesa.

Če lahko vse izjavne račune $F(e)$ zapišemo kot konjunkcije, je celoten izjavni račun C_k v disjunktne normalni obliki. Za disjunktne normalne oblike lahko zapišemo resničnostno tabelo:

	c_1	c_2	c_{\dots}	c_{n-1}	c_n
e_1	0	1	\dots	1	1
e_2	1	1	\dots	0	0
e_3	0	0	\dots	1	0
e_i	\dots	\dots	\dots	\dots	\dots

V resničnostni tabeli vsaka vrstica predstavlja eno izmed kombinacij vmesnih ciljev, ki se jo mora tekom izvajanja ad-hoc procesa uresničiti. Pri disjunktne normalni obliki moramo vedeti, da opisuje *minimalne* nabore ciljev, ki jih mora ad-hoc proces uresničiti. To pomeni, da je lahko v vrsticah tam, kjer so zapisane ničle, tudi enica, saj lahko procesni agent zaradi trenutnega stanja okolja v kateremkoli trenutku začne uresničevati drugo kombinacijo vmesnih ciljev. V tem primeru je mogoče pri izpolnjevanju prejšnje kombinacije vmesnih ciljev kakšnega od vmesnih ciljev že izpolnil, čeprav v novi kombinaciji tega cilja ni potrebno doseči. Pri tem zopet predpostavljamo, da se cilji med seboj ne izključujejo, torej da izvedba enega cilja ne onemogoči doseganje drugega cilja. V tem primeru disjunktne normalne oblike ne more dovolj natančno opisati odvisnosti med cilji in je treba C_k izjavni račun zapisati z drugimi veznimi operatorji. Za potrebe magistrske naloge smo obravnavali samo primere, kjer je bila mogoča uporaba disjunktne normalne oblike.

Pogoji za doseganje ciljev

Pri predlaganem načinu modeliranja ad-hoc procesov so okolje in procesi ločeni. Izvajanje ad-hoc procesa je pogojeno s stanjem okolja. Ko se ad-hoc proces začne izvajati, mora

⁴Osnovna konjunkcija je izjavni račun, kjer med spremenljivkami nastopa samo operator konjunkcija.

procesni agent pri izbiri naslednjega vmesnega cilja preveriti, če je okolje v takšnem stanju, da je doseganje tega cilja sploh možna.

Za doseganje posameznega vmesnega cilja morajo biti na razpolago določeni viri. Razpoložljivost teh virov predstavlja glavni pogoj za doseganje vmesnih ciljev. Procesni agent mora ob izbiri naslednjega možnega cilja preveriti, ali je na voljo dovolj virov. Pri tem je treba upoštevati, da so lahko viri zasedeni v celoti ali delno. Če je potreben vir zaseden v celoti, potem ga procesni agent ne more uporabiti. Če je vir zaseden le deloma in prost delež vira zadostuje za doseganje vmesnega cilja, potem procesni agent ta vir ustrezno zasede.

Če ne predpostavimo časovne neodvisnosti vmesnih ciljev, potem se lahko zgodi, da uresničitev enega vmesnega cilja onemogoči uresničitev drugega vmesnega cilja. V tem primeru je treba to ustrezno zapisati v obliki logike izjavnega računa, ki opisuje posamezno kombinacijo vmesnih ciljev e .

7.2.3 Procesna entiteta in procesni agent

Že na Sliki 7.1 smo opisali osnovno relacijo med procesno entiteto, procesnim agentom in poslovnim procesom. Ko pride v okolju do zahteve za ad-hoc proces, se ta instancira v obliki procesne entitete. Vodenje procesne entitete prevzame procesni agent.

Procesna entiteta je entiteta, ki je enolično določena z začetkom izvajanja in tipom ad-hoc procesa. V okolju lahko v enem trenutku obstaja več procesnih entitet, kar pomeni, da se naenkrat izvaja več ad-hoc procesov. Procesna entiteta predstavlja strukturo ad-hoc procesa. Ker vsako procesno entiteto vodi en agent, lahko zaradi obstoja več procesnih agentov govorimo o multi-agentnem sistemu (glej Poglavlje 6.4).

Logiko ad-hoc procesa prevzame procesni agent, ki vodi procesno entiteto. Procesni agent vsebuje dva nivoja logike:

Procesna logika Pri procesu modeliranja ad-hoc procesov se ne osredotočamo toliko na procesno logiko, kot na poslovno. Procesna logika pomaga razumeti, kaj se dogaja tekom izvajanja ad-hoc procesa. Po drugi strani procesna logika daje procesnemu agentu na voljo funkcije, s katerimi preverja stanje okolja. Poleg tega mora imeti procesni agent na voljo logiko, ki omogoča ustrezno označevanje zasedenosti virov okolja. Procesna logika je že stvar implementacije podpore izvajanja ad-hoc procesov. S stališča poslovne vsebine procesna logika ni pomembna in je zato ne modelirajo poslovni uporabniki.

Poslovna logika Poslovna logika določa, kako bo procesni agent izbral vmesne cilje. Pri izbiri naslednjega vmesnega cilja, ki ga bo procesna entiteta izpolnjevala, se procesni agent opira na resničnostno tabelo, v kateri so zapisane možne kombinacije vmesnih ciljev (glej Poglavlje 7.2.2) ali izjavne račune, ki opisujejo odvisnosti med vmesnimi cilji. Hkrati mora imeti na voljo tudi dodatne hevristične funkcije (glej Poglavlje 7.3), ki določajo čimbolj optimalen vrstni red doseganja vmesnih ciljev. Poslovna logika določa obnašanje procesnih agentov, kar posledično pomeni način izvajanja

ad-hoc procesov in poslovanja v združbi. Poslovno logiko določajo in modelirajo poslovni uporabniki in jo zapišemo v obliki poslovnih pravil.

Ko se procesni agent na podlagi procesne in poslovne logike odloči za uresničevanje naslednjega vmesnega cilja c_i , postavi procesno entiteto v stanje i . Ko je procesna entiteta v stanju i , se izvajajo aktivnosti, ki so potrebne za doseg cilja c_i . S stališča nižanja entropije poslovanja tako vemo, kaj se trenutno dogaja v ad-hoc procesu in lahko predvidevamo, katero nadaljnje zaporedje vmesnih ciljev bi bilo najbolj primerno. Poleg tega lahko poslovnemu uporabniku za vsako stanje procesne entitete ponudimo:

Priloge Dejstvo je, da ne vemo, v kakšnem vrstnem redu se bodo v določenem stanju procesne entitete izvajale. Vendar pa z uvedbo vmesnih ciljev in stanj procesne entitete vsaj vemo, kateri vmesni cilj se takrat dosega. Z namenom podpore, pospešitve in predvsem lažjega izvajanja aktivnosti, lahko ponudimo *priloge*. Priloge so različni dokumenti, informacije ali drugi artefakti, ki olajšajo uresničitev vmesnega cilja. Priloge niso vhod ali potrebni viri za določeno stanje, ampak samo *pomoč* za izvajanje aktivnosti.

Ključne kazalce uspešnosti Ključni kazalci uspešnosti se uporabljajo že na kontekstnem nivoju vseh procesnih entitet z namenom pregleda dogajanja v poslovanju. Z uvedbo vmesnih ciljev smo omogočili pregled, v kolikšni meri so posamezni ad-hoc procesi izvedeni in kdaj lahko pričakujemo, da se bodo končali. V običajnih poslovnih procesih je eden izmed ključnih kazalcev uspešnosti število opravljenih aktivnosti poslovnega procesa, v primerjavi s preostalimi, ki še morajo biti opravljene. V ad-hoc procesu pa je ključni kazalec uspešnosti število vmesnih ciljev, ki so bili doseženi, v primerjavi s preostalimi vmesnimi cilji, ki morajo biti za doseg končnega cilja še izpolnjeni.

Procesni agenti naj bi bili čim bolj avtonomni agenti. To pomeni, da samostojno vodijo procesne entitete. Pri tem so omejeni z delovanjem ostalih procesnih agentov in stanjem okolja. Procesni agenti morajo imeti sposobnost komunikacije z okoljem in drugimi procesnimi agenti (glej Slika 6.2). Komunikacija poteka preko dogodkovnega vodila, ki ga zagotavlja okolje, v obliki dogodkov, ki jih prožijo agenti ali viri okolja. Procesni agenti imajo lastnost *reaktivnosti*, saj se morajo aktivno odzivati na takšne dogodke in ustrezno voditi procesne entitete.

Procesni agent je lahko *točka optimizacije*. Če bi radi čim bolj optimizirali potek ad-hoc procesa, je treba dati procesnim agentom na voljo ustrezno poslovno logiko, ki bo znala najti najbolj ugoden potek ad-hoc procesov. Procesne agente je možno dodelati do te mere, da znajo sami v določenih okvirih spreminjati poslovno logiko. V tem primeru govorimo že o *inteligentnih* agentih, kar pa je bolj stvar tehnične implementacije podpore kot pa del modeliranja ad-hoc procesov s strani poslovnih uporabnikov.

Prav tako je procesni agent *točka učenja* v ad-hoc procesih. Z namenom izboljšave in optimizacije ad-hoc procesov, lahko v procesne agente vgradimo dodatno logiko in znanje, ki glede na izvedene ad-hoc procese omogoča učenje in izboljšavo hevrističnih funkcij (glej Poglavje 7.3) za izbor vmesnih ciljev.

Če hočemo v izvajanju ad-hoc procesov zagotoviti *sledljivost in beleženje*, lahko to logiko vgradimo v procesne agente. Vsaka aktivnost, naj bo to izbira naslednjega vmesnega cilja ali pa dejansko izvajanje aktivnosti za doseganje vmesnega cilja, se izvede pod nadzorom procesnega agenta. Zato lahko v procesni logiki definiramo tudi proces beleženja, ki tekom izvajanja ad-hoc procesov piše dnevnik. Takšen dnevnik je naknadno koristen vir informacij za učenje in optimizacijo ad-hoc procesov.

7.3 Izvajanje ad-hoc procesa

Izvajanje ad-hoc procesa se začne, ko pride do zahteve za poslovni proces. Zahtevi za izvajanje sledijo naslednji koraki:

1. *Instanciacija nove procesne entitete* - Najprej se ustvari nova instanca procesne entitete. Procesna entiteta je določena s tipom ad-hoc procesa, ki ga predstavlja, enoličnim identifikatorjem in časom začetka izvajanja procesa. Enolični identifikator zagotavlja ločene identitete posameznih procesnih entitet in je uporabljen za ločevanje in identifikacijo vseh instanc procesov v okolju.
2. *Inicializacija procesnega agenta* - Inicializacija procesnega agenta sledi instanciaciji procesne entitete. Procesna entiteta brez procesnega agenta predstavlja le statičen del ad-hoc procesa, zato mora imeti vsaka procesna entiteta takoj ob instanciaciji na voljo svojega procesnega agenta. Procesni agent inicializira procesno in poslovno logiko ter kreira referenco na procesno entiteto.
3. *Prvo preverjanje stanja okolja* - Procesni agent takoj ob inicializaciji preveri stanje okolja. Tako dobi informacijo o razpoložljivosti virov in obstoju drugih procesnih entitet, ki so bile ustvarjene predhodno.

Sledi prva izbira kombinacije vmesnih ciljev, ki pripelje do uresničitve končnega cilja. Za lažje razumevanje bomo predpostavili, da lahko vse možne kombinacije vmesnih ciljev zapišemo v disjunktni normalni obliki.

Procesni agent mora pred izbiro vmesnega cilja opraviti naslednja opravila:

1. *Uporaba cenilne funkcije na nivoju osnovnih konjunkcij* - Procesni agent mora ugotoviti, katera kombinacija vmesnih ciljev v določenem trenutku najugodnejše pripelje do končnega cilja. Z izrazom "najugodnejše" opisujemo takšen potek ad-hoc procesa, kjer se porabi čim manj virov in časa za ustvarjanje čim večje dodane vrednosti. V ta namen uporabi funkcijo

$$f_c(F(e), t)$$

Ta funkcija preveri, koliko "stane" vsaka od osnovnih konjunkcij v času t . Z izrazom "stane" mislimo na potreben čas, vire in vpliv na izvajanje ostalih ad-hoc

procesov. Funkcija f_c upošteva posamezne cene vseh vmesnih ciljev, ki so v konjunkciji. V primeru, da so kakšni od teh vmesnih ciljev ob času t že doseženi (npr. v kasnejših iteracijah preverjanja), se cena posamezne osnovne konjunkcije zniža. Procesni agent izbere tisto osnovno konjunkcijo, kjer je vrednost f_c minimalna. Išče torej minimum funkcije

$$\min_{e \in E_H} f_c(F(e), t),$$

kjer je e ena izmed vseh možnih kombinacij vmesnih ciljev.

2. *Uporaba hevristične funkcije za izbiro vmesnega cilja znotraj konjunkcije* - Ko procesni agent izbere najbolj ugodno konjunkcijo, se mora odločiti, kateri vmesni cilj znotraj konjunkcije bo poizkusil uresničiti. Za to odločitev najprej potrebuje informacijo o tem, katere vmesne cilje je glede na trenutno stanje okolja sploh možno uresničiti. Uresničiti je možno tiste cilje, za katere so na voljo ustrezni viri v okolju. Procesni agent uporablja funkcijo

$$f_a(c_i, t),$$

kjer je c_i vmesni cilj iz izbrane konjunkcije, t pa čas, ob katerem funkcija preverja razpoložljivost virov, ki so potrebni za dosegov vmesnega cilja. Procesni agent se lahko ob času t odloči samo za uresničevanje tistih vmesnih ciljev, pri katerih je vrednost funkcije f_a v času t različna od nič⁵. Na vmesnih ciljeh, ki so uresničljivi, mora procesni agent uporabiti še hevristično funkcijo, s katero razvrsti, kateri vmesni cilj je najboljše najprej uresničiti.

$$h(c_i, t),$$

kjer je c_i vmesni cilj, ki je uresničljiv v času t . Bistvo hevristične funkcije je, da procesnemu agentu omogoča najbolj optimalno izvajanje ad-hoc procesa. Znotraj konjunkcije ne obstaja preprosto pravilo, kateri vmesni cilj se splača najprej uresničiti. Po intuiciji bi sicer lahko rekli, da se splača najprej doseči najcenejši cilj s stališča porabe virov, vendar temu ni vedno tako. Hevristična funkcija upošteva naravo celotnega okolja. Poizkuša izbrati takšen vmesni cilj, ki je čim manj odvisen od virov, ki predstavljajo ozko grlo procesa. Upošteva tudi stanje drugih ad-hoc procesov in predlaga tisti vmesni cilj, ki je v najmanj kombinacijah vmesnih ciljev ostalih procesov. V splošnem hevristična funkcija zagotavlja takšen izbor vmesnih ciljev, ki bo ad-hoc proces pripeljal do končnega cilja z najmanj porabljenimi viri v najkrajšem možnem času. Zaradi dinamike sprememb okolja ad-hoc procesa ne moremo vnaprej napovedati takšno najbolj ugodno kombinacijo, temveč se hevristična funkcija uporablja sproti in je odvisna od časa. Na ta način zagotovimo učinkovitejše izvajanje in hkrati nižjo entropijo ad-hoc procesov.

⁵Običajno se za izražanje izpolnjenosti pogoja uporablja neničelna vrednost. Seveda se lahko uporabi tudi drugo domenjeno vrednost, ki jo funkcija f_a vrne v primeru izpolnjenega pogoja.

Izvedba hevristične funkcije sodi tako v poslovno domeno kot v implementacijo podpore ad-hoc procesov. Lahko je del poslovne ali procesne logike. Predstavlja "inteligenco" procesnih agentov. V magistrski nalogi se na izvedbo hevristične funkcije nismo osredotočali, temveč smo jo samo definirali in jo umestili v celoten koncept modeliranja ad-hoc procesov.

3. *Sprememba stanja procesne entitete* - Ko se procesni agent odloči za doseganje določenega vmesnega cilja c_i , spremeni stanje procesne entitete v stanje i . Tako vemo, da se v ad-hoc procesu, ki ga procesna entiteta predstavlja, odvijajo aktivnosti, s katerimi se uresniči vmesni cilj c_i .
4. *Izstop iz stanja in izbira novega vmesnega cilja* - Ko se v procesni entiteti doseže vmesni cilj c_j , mora procesni agent znova izbrati naslednji možni vmesni cilj. Ponovi se celoten postopek od prvega koraka naprej. Razlog, da procesni agent preprosto ne uresniči vseh vmesnih ciljev iz trenutne konjunkcije je, da se je lahko vmes spremenilo stanje okolja, dodala nova (ugodnejša) kombinacija vmesnih ciljev, ali pa je kakšna druga procesna entiteta zasedla vse vire, ki so potrebni za doseg vmesnih ciljev iz trenutne konjunkcije. Procesni agent torej ob vsaki izbiri začne preverjanje na nivoju osnovnih konjunkcij in izvede vse opisane korake.

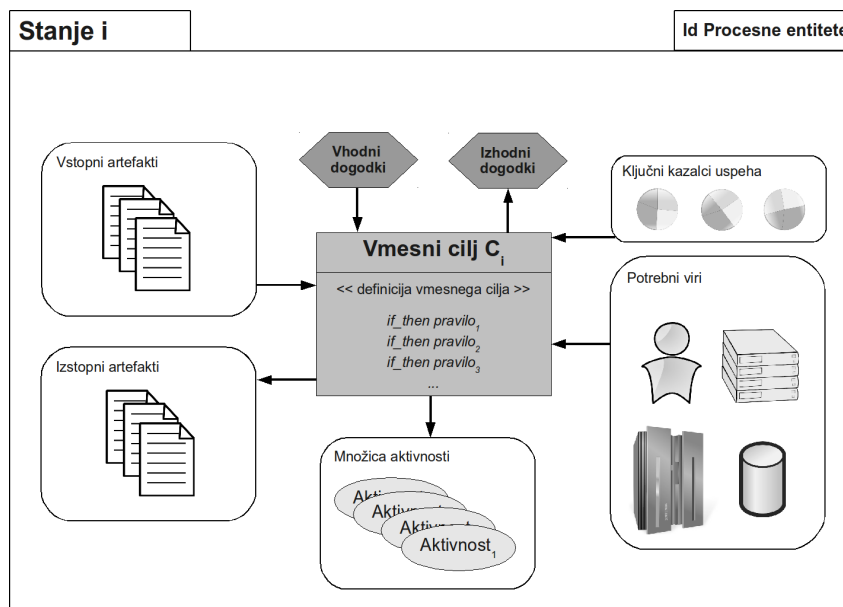
Poglavje 8

Diagramaska tehnika

V Poglavlju 7 smo opisali prilagojen koncept za modeliranje ad-hoc procesov, ki predstavlja dopolnitev oz. razširitev obstoječih diagramskih tehnik. V nadaljevanju sledi opis diagramske tehnike, ki podpira opisani pristop modeliranja.

8.1 Pregled

Za pregled celotne diagramske tehnike je na Sliki 8.1 prikazan glavni gradnik diagramske tehnike.



Slika 8.1: Osnoven diagram, ki prikazuje stanje procesne entitete

Glavna abstrakcija ad-hoc procesa je stanje procesne entitete. Kot smo opisali v Poglavlju 7, procesna entiteta predstavlja instanco ad-hoc procesa in tekom izvajanja

zavzema različna stanja. Vodi jo procesni agent, ki na diagramski tehniki ni eksplicitno prikazan. Procesnega agenta poslovni uporabniki ne modelirajo, saj spada v področje implementacije podpore ad-hoc procesa.

Vsako stanje je pogojeno z vmesnim ciljem, katerega procesna entiteta uresničuje v določenem trenutku. Za doseganje vmesnega cilja se izvajajo aktivnosti, vendar ne poznamo njihovega vrstnega reda. Zato na diagramu samo okvirno označimo, katere so lahko te aktivnosti.

Za doseganje vsakega vmesnega cilja je pomembno, da poznamo vire, ki so za to potrebni. Na diagramu so potrebni viri označeni na desni strani.

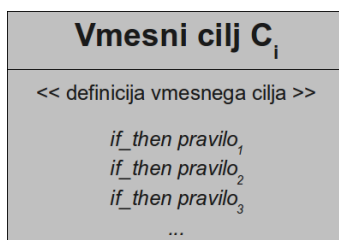
Za delovanje procesnega agenta je pomembno, kateri dogodki so potrebni za komunikacijo, ko je procesna entiteta v določenem stanju. Dogodki so odvisni npr. od virov, ki se uporabljajo v določenem stanju. Na vrhu diagrama so zato označeni vhodni in izhodni dogodki.

Za ustrezno podporo in nižanje entropije ima vsako stanje pripadajoče informacijske artefakte. To so različni dokumenti in izdelki, ki so potrebni za izvajanje aktivnosti v pripadajočem stanju procesne entitete. Na levi strani sta zato označeni dve množici artefaktov - vhodni in izhodni.

8.2 Gradniki diagrama

8.2.1 Vmesni cilj

Vmesni cilj je na diagramu predstavljen kot pravokotnik. Bistvena informacija o vmesnem cilju je, kakšni so pogoji za uresničitev tega cilja. Pogoji so zapisani v obliki *if...then*¹. Takšna notacija je primerna in razumljiva za poslovne uporabnike. Definicija pogojev izpolnjevanja vmesnih ciljev spada med poslovno logiko in je zato v domeni poslovnega uporabnika.



Slika 8.2: Grafična predstavitev vmesnega cilja

Eden izmed razlogov za izbiro takšne notacije je tudi možnost, da *if...then* pravila pri implementaciji neposredno pretvorimo v poslovna pravila. Z vse večjo uveljavljenostjo različnih BRMS sistemov je takšen diagram lažje pretvoriti v poslovno logiko v obliki

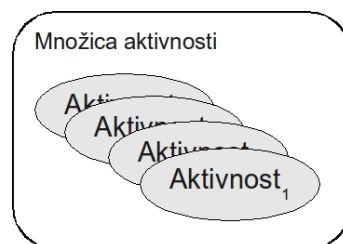
¹*If...then pravila so navadna pravila oblike pogoj...akcija, vendar smo zaradi razširjene uporabe izraza if...then pravila obdržali angleško različico izraza.*

poslovnih pravil. Hkrati smo mnenja, da bi bila grafična predstavitev pogojev manj intuitivna in bi zanjo potrebovali več znanja iz različnih grafičnih notacij, ki to omogočajo. Predvsem bi dodatni grafični elementi pokvarili splošno berljivost diagrama.

Seznam kombinacij vmesnih ciljev, ki pripeljejo do uresničitve končnega cilja ad-hoc procesa, je zapisan v obliki resničnostne tabele (glej Poglavje 7.2.2). Resničnostno tabelo si lahko predstavljamo kot globalni procesni diagram ad-hoc procesa, kjer lahko hitro in na pregleden način vidimo, katere cilje je v ad-hoc procesu treba uresničiti.

8.2.2 Množica aktivnosti

V ad-hoc procesu ne moremo točno določiti, katere aktivnosti se bodo izvajale in v kakšnem vrstnem redu. Namen uvedbe vmesnih ciljev je, da to negotovost abstrahiramo tako, da v ad-hoc procesu poznamo odgovor na vprašanje *kaj* se dogaja, ni nam pa treba vedeti *na kakšen način*. Ko določimo vmesne cilje, lahko vsaj okvirno določimo množice aktivnosti, ki naj bi se izvedle z uresničitvijo posameznih vmesnih ciljev.



Slika 8.3: Grafična predstavitev množice aktivnosti, ki se lahko izvajajo za uresničitve vmesnega cilja

Namen takšnega popisa aktivnosti je, da poslovni uporabnik okvirno določi aktivnosti za posamezen vmesni cilj v ad-hoc procesu. Tekom izvajanja procesa se bodo v procesni entiteti v stanju i izvajale aktivnosti iz te množice. S takšnim gradnikom ne določimo vrstnega reda izvajanja aktivnosti. Prav tako ni vnaprej določeno, ali se morajo za uresničitve vmesnega cilja izvesti vse aktivnosti iz te množice.

8.2.3 Potrebni viri

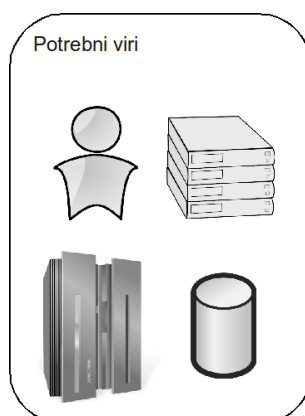
Za doseganje vsakega vmesnega cilja so potrebni določeni razpoložljivi viri. Procesni agent se na podlagi tega podatka sploh odloča, ali bo vmesni cilj poizkusil uresničiti. Vire potrebujejo aktivnosti, ki se v stanju procesne entitete izvajajo.

Viri so lahko različni, odvisno od področja ad-hoc procesov. Podrobno smo možne tipe virov opisali v Poglavju 7.2.1.

Poslovni uporabnik mora tekom modeliranja ad-hoc procesov pri definiranju vmesnih ciljev vedeti, katere vire se bo pri doseganju teh ciljev uporabljalo. Ko imamo na voljo celoten diagram z vsemi možnimi vmesnimi cilji oz. stanji procesnih entitet, lahko že s primerjavo razdelkov *potrebni viri* vseh stanj ugotovimo, katere vire se največ uporablja

in doseganje katerih ciljev se med seboj izključuje. Takšna grafična notacija, ki prikazuje porabo virov, je tako v pomoč pri statični optimizaciji ad-hoc procesov, ki jo lahko opravimo že v postopku popisa in modeliranja ad-hoc procesa.

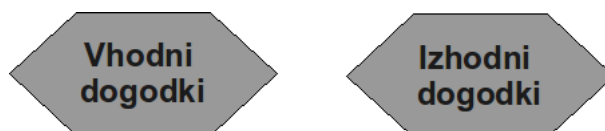
Prav tako je seznam potrebnih virov pomemben v času izvajanja ad-hoc procesa, ko mora procesni agent v realnem času ugotavljati, če so potrebni viri za posamezen vmesni cilj razpoložljivi. Hevristična funkcija $h(c_i, t)$, ki pomaga izbrati naslednji vmesni cilj, lahko pri izdelavi ocene posameznega stanja poda nižjo oceno za tista stanja, ki zaklenejo veliko virov in tako onemogočajo izvajanje ostalih ad-hoc procesov. Seznam potrebnih virov je tako pomemben tudi za procesno logiko procesnih agentov.



Slika 8.4: Grafična predstavitev virov, ki so potrebni za doseganje vmesnega cilja

8.2.4 Dogodki

Dogodki so pomembni za medsebojno komuniciranje med procesnimi entitetami in procesnimi entitetami ter okoljem. Ko se procesni agent odloči za določen vmesni cilj in spremeni stanje procesne entitete, se hkrati še registrira na dogodke, ki so za to stanje navedeni v diagramu. To so dogodki, ki so pomembni za aktivnosti, ki se izvajajo za uresničitev vmesnega cilja v tem stanju.



Slika 8.5: Grafična predstavitev vhodnih in izhodnih dogodkov

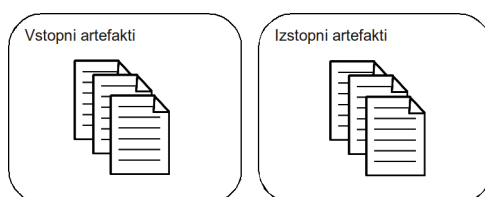
Poslovni uporabnik mora imeti možnost da določi, kateri dogodki so pomembni za posamezna stanja procesnih entitet. S tem določa, kakšna mora biti notranja komunikacija med procesnimi entitetami. Hkrati določa tudi dogodke, ki prihajajo do procesne entitete od posameznih virov. Na ta način se definira, kako naj procesna entiteta reagira na spremembe v okolju.

8.2.5 Artefakti

Za vsako stanje procesne entitete so pomembni artefakti. Ti so lahko dveh vrst:

Vhodni artefakti Vhodni artefakti so tisti artefakti, ki so potrebni pred vstopom v novo stanje procesne entitete. S temi artefakti se podpira aktivnosti, ki se opravljajo pri doseganju vmesnega cilja.

Izhodni artefakti Izhodni artefakti se izdelajo tekom doseganja vmesnega cilja in se lahko naknadno uporabijo pri naslednjih stanjih procesne entitete.

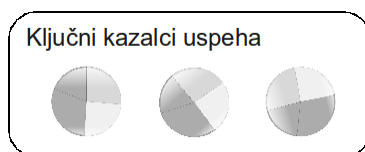


Slika 8.6: Grafična predstavitev vhodnih in izhodnih artefaktov za posamezno stanje

Poslovni uporabnik mora imeti glede na definicijo vmesnega cilja možnost definirati artefakte. Gre za dodatno pomoč in podporo pri izvajanju ad-hoc procesa. Artefakte lahko gledamo tudi kot dodaten vir informacij o izvajanju ad-hoc procesa, ki ga potrebujejo poslovni uporabniki.

8.2.6 Ključni kazalci uspeha

Za znižanje entropije izvajanja ad-hoc procesov morajo imeti poslovni uporabniki pregled nad tem, kaj se v procesih dogaja.



Slika 8.7: Grafična predstavitev ključnih kazalcev uspeha

V ta namen moramo že v postopku definiranja vmesnih ciljev določiti, kaj so ključni kazalci uspeha delovanja aktivnosti pri doseganju posameznega vmesnega cilja. Izpeljava ključnih kazalcev se namreč izkaže za precej pomembno opravilo, njihov obstoj pa med potekom ad-hoc procesov prinaša veliko koristi za vodenje in upravljanje ad-hoc procesov.

Z uvedbo grafičnega gradnika za prikaz ključnih kazalcev uspeha smo hoteli doseči, da se na to misli že pri modeliranju ad-hoc procesov. Doseganje vmesnih ciljev je v ad-hoc procesih zaradi nedoločenosti aktivnosti težko določiti. Že v splošnem je težko definirati dobre metrike, s katerimi bi lahko spremljali izvajanje poslovnih procesov. Zato

je koristno, da poslovni uporabnik ob določanju vmesnih ciljev določi tudi metriko ali način, kako se vrednoti izvajanje aktivnosti v posameznih stanjih procesnih entitet.

Poglavje 9

Študija primera

Uporabo predlaganega koncepta modeliranja ad-hoc procesov bomo prikazali na realnem primeru. V poglavju sledi opis ad-hoc procesa iz obravnave goljufij v zdravstvenem zavarovalništvu. Prikazali bomo, kako lahko po predlaganem konceptu modeliranja takšen proces razčlenimo in zapišemo v obliki vmesnih ciljev.

9.1 Opis primera

9.1.1 Goljufije v zavarovalništvu

Splošno znano dejstvo je, da zavarovalnice letno porabijo veliko denarja za poplačilo neupravičenih zahtevkov [28]. Zaradi velikega števila zahtevkov, ki jih morajo zavarovalnice izplačevati, je težko nadzirati in kontrolirati dejansko stanje škodnih primerov. V večini primerov zavarovalnice raje izplačajo zahtevke in tako obdržijo komitente in status dobre zavarovalnice. S tem po ocenah različnih organizacij izplačajo vsaj 2% do celo 18% zavarovancev, ki so svoje škodne primere priredili in tako prišli do neupravičene premije oz. izplačila [42, 43, 44, 45]. Takšnim primerom lahko upravičeno rečemo zavarovalniške goljufije.

V praksi se izkaže, da likvidatorji škodnih primerov¹ preprosto nimajo časa in virov, da bi preverjali upravičenost zahtevkov. Zato so posamezne zavarovalnice ustanovile dodatne oddelke, ki se ukvarjajo s preiskavo neupravičenih škodnih zahtevkov. Na trgu so na voljo že različne informacijske rešitve, ki avtomatizirajo detekcijo takšnih goljufij in olajšajo delo preiskovalcem.

Vendar je treba na obravnavo zavarovalniških goljufij gledati s celostnega vidika. Samo odkrivanje goljufij ni dovolj, da zavarovalnice prihranijo neupravičena izplačila. Prav tako zavarovalnice ne morejo preprosto ustaviti izplačil zavarovancem, ki zahtevajo neupravičene škodne zahtevke. Zgodi se lahko, da so se v oddelku za preiskavo goljufij zmotili ali pa da je prišlo do kakšnih drugih napak, ki zahtevke prikažejo kot goljufijo. Takšne

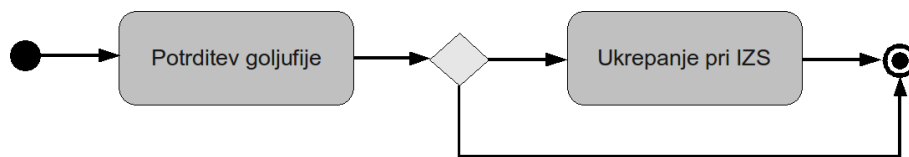
¹Likvidatorji škodnih primerov likvidirajo oz. izvedejo proces izplačila škodnih primerov.

primere je treba zato obravnavati vsakega posebej in glede na preverjene podatke in dodatno raziskovanje ukrepati. Gre torej za novo vrsto poslovnih procesov, ki jih izvajajo zavarovalnice.

Odkrivanje neupravičenih zahtevkov in obravnava goljufij v zavarovalništvu je še posebej pereč problem za zdravstvene zavarovalnice, saj je na tem področju število škodnih zahtevkov zaradi velikega števila zdravstvenih zavarovancev veliko. Delež izplačil v primerjavi s številom zavarovancev je precej večje kot v drugih zavarovalniških panogah. Poleg tega (govorimo za področje Slovenije) se škodni primeri ne obravnavajo na nivoju zavarovancev, temveč na nivoju *izvajalcev zdravstvenih storitev*, ki zavarovalnicam izstavljajo račune za opravljene storitve. Neupravičene zahtevke oz. goljufijo povzročijo IZS in ne dejanski zavarovanci, zato je obravnava posameznega primera goljufije toliko težja.

9.1.2 Proces razreševanja sumljivega zahtevka

Opisan primer je vzet iz realne situacije zdravstvene zavarovalnice, ki se proaktivno ukvarja z detekcijo in obravnavo goljufij.



Slika 9.1: Konceptualni diagram obravnave sumljivega zahtevka

Na Sliki 9.1 je prikazan proces *razreševanja sumljivega zahtevka*. Ta ad-hoc proces se začne, ko sistem za detekcijo goljufij sproži zahtevo za ad-hoc proces. Kot vhodni parameter v proces služi identifikator sumljivega škodnega zahtevka. Identifikator vključuje račun, na katerem so sumljive postavke, identifikacija IZS, ki je izstavil račun in razlaga sistema za odkrivanje goljufij, zakaj je ta zahtevek neupravičen.

Ad-hoc proces je razdeljen na dva večja dela:

Potrditev goljufije Najprej je treba neupravičen zahtevek potrditi kot goljufijo. V ta namen je v ad-hoc procesu na voljo kar nekaj različnih načinov oz. aktivnosti, katerih zaporedje ne moremo predvideti vnaprej. Preiskovalci, ki so na voljo, so različno zasedeni in različno primerni za raziskovanje posameznega primera. Obstaja tudi več različnih načinov, kako priti do podatkov, ki potrjujejo ali zavržejo goljufijo. Že sama narava teh aktivnosti je precej nedoločena in zato ne moremo natančno načrtovati njihovega izvajanja vnaprej. Gre torej za dinamičen proces, ki ustreza definiciji ad-hoc procesa.

Ukrepanje pri IZS Ko se goljufijo potrdi, je treba ukrepati pri IZS, ki je odgovoren za ta primer. Če zavarovalnica še ni likvidirala škodnega primera, lahko zaustavi izplačilo in o tem obvesti IZS. Če pa je zahtevek že izplačala, mora doseči povrnitev

neupravičenih stroškov. V obeh primerih gre za dinamičen proces, kjer so aktivnosti nepredvidljive, saj vnaprej ne poznamo obnašanja IZS. Običajno je postopek takšen, da se IZS pošlje ugovor z neformalno zahtevo po vrnitvi stroškov. Takšen ugovor mora biti ustrezno opremljen z vsem dokaznim gradivom. Če se IZS strinja in prizna svojo napako, se proces zaključi. V primeru da se IZS ne strinja, se lahko sproži kazenski postopek ali civilno tožbo. Vsi pravni postopki vnašajo v ad-hoc proces dodatno dimenzijo dinamike, saj so zelo nepredvidljivi in jih je težko modelirati. Običajno se jim poizkušajo zavarovalnice izogniti in doseči vsaj izvensodno poravnavo.

Opisan ad-hoc proces se lahko konča na dva načina. Če se škodni zahtevk izkaže za upravičenega, ni potrebe po ukrepanju pri IZS in proces se konča. Če pa se v ad-hoc procesu potrdi, da je škodni zahtevk res goljufija, se mora proces končati z izterjavo stroškov pri IZS.

9.1.3 Ad-hoc proces zmodeliran z UML diagramom aktivnosti

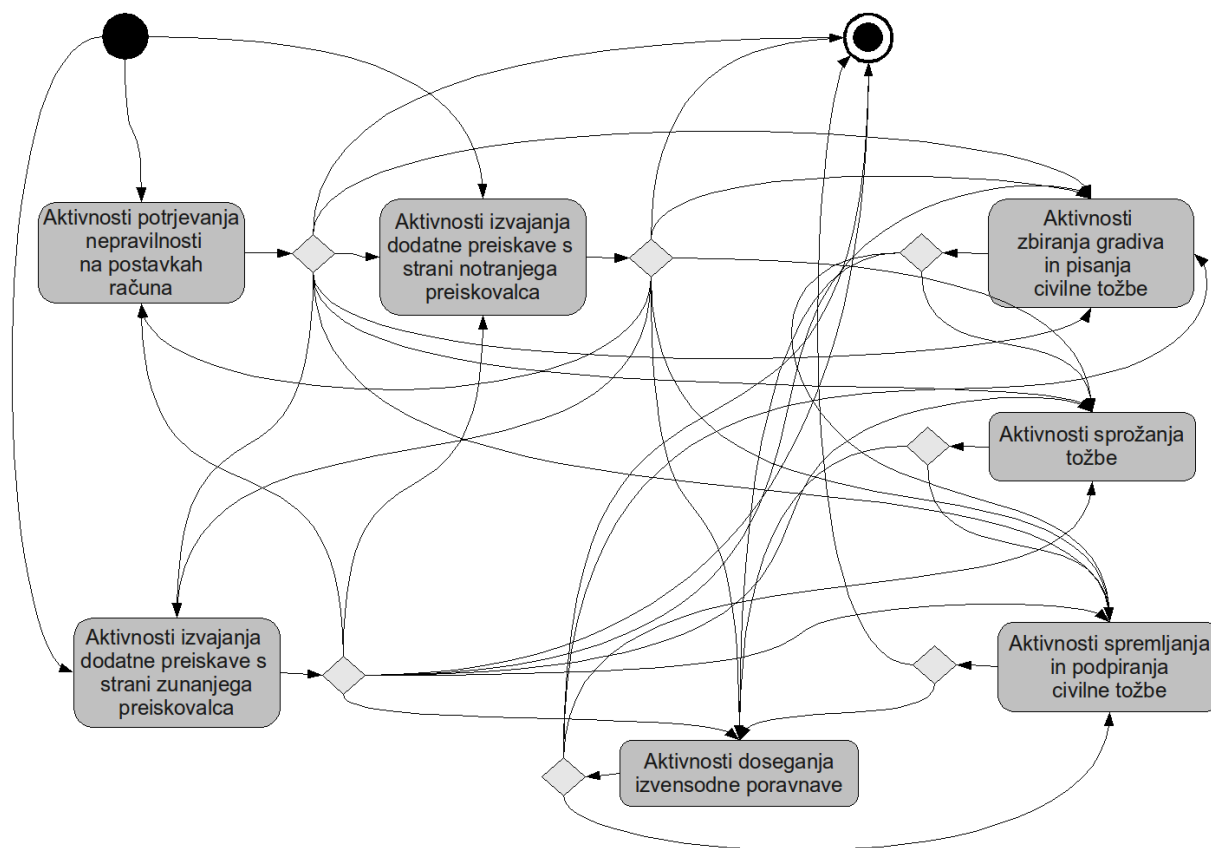
Da prikažemo prednosti nove diagramske tehnike, si bomo najprej ogledali, kakšne so možnosti za modeliranje v Poglavju 9.1.2 opisanega ad-hoc procesa.

Možnosti poteka procesa razreševanja lahko razdelimo na dva večja sklopa in sedem množic aktivnosti. V prvem sklopu so aktivnosti potrjevanja sumljivega zahtevka. Aktivnosti so razdeljene na tri podprocesse in sicer *potrjevanje nepravilnosti na postavkah računa*, *izvajanje dodatne preiskave s strani notranjega preiskovalca* in *izvajanje dodatne preiskave s strani zunanjega preiskovalca*. V drugem sklopu so aktivnosti, ki so povezane s civilno tožbo. Gre za štiri podprocesse in sicer *zbiranje gradiva in pisanje civilne tožbe*, *sprožanje civilnega postopka*, *spremljanje in podpora postopka civilne tožbe* ter *doseganje izvensodne poravnave*.

Našteti podprocesji opisujejo samo okvirno dogajanje v ad-hoc procesu. Dejstvo je, da teh podprocesov *ne moremo* dodatno razgraditi in jih opisati na nivoju aktivnosti, saj ne poznamo vrstnega reda posameznih aktivnosti znotraj podprocesov. Zunanji in notranji preiskovalci se za način dela odločajo na podlagi vsakega posameznega primera posebej in nimajo vnaprej določenega zaporedja aktivnosti izvajanje preiskave. Prav tako ne moremo vnaprej zmodelirati spremljanja in podpore postopka civilne tožbe, saj se ta izvaja izven našega poslovnega sistema. Tudi doseganje izvensodne poravnave je zaradi narave pravnih postopkov odvisen od primera do primera in ga je težko vnaprej opredeliti kot določeno zaporedje aktivnosti. Če bi se kljub dinamični naravi naštetih podprocesov odločili za modeliranje zaporedij njihovih aktivnosti, bi dobili zelo nepregleden diagram, poln alternativnih tokov, izjem in posebnih obravnav.

Že na globalnem procesnem nivoju na Sliki 9.2 vidimo, da je zaradi veliko možnih povezav med podprocesji diagram nepregleden. Ko smo nekatere od teh podprocesov razgradili do nivoja aktivnosti, je postal tako dobljen procesni diagram neuporaben zaradi prevelikega števila možnih povezav med aktivnostmi. Informacija o poteku ad-hoc procesa ni bila razvidna in diagram ni bil primeren ne za uporabo s strani poslovnih uporabnikov

ne s strani implementatorjev IT podpore.



Slika 9.2: Študija primera zmodelirana z UML diagramom aktivnosti

9.2 Procesni diagram z novo diagramsko tehniko

9.2.1 Kombinacije doseganja končnega cilja ad-hoc procesa

Za prikaz realnega primera modeliranja ad-hoc procesa se bomo osredotočili na del procesa, kjer se preverja upravičenost zahtevka in na del procesa, kjer se gre v civilno tožbo.

Končna cilja procesa sta lahko dva:

1. *Sumljiv škodni zahtevak ni goljufija* - V tem primeru se že v prvem delu procesa ugotovi, da je sumljiv zahtevak upravičen in da ne predstavlja goljufije. Ad-hoc proces se konča, rezultat procesa pa je poročilo o upravičenosti škodnega zahtevka.
2. *Sumljiv škodni zahtevak se spozna za goljufijo, izterja se stroške* - v tem primeru se v procesu potrdi neupravičenost škodnega zahtevka in se ga označi za goljufijo. Sproži se enega izmed načinov izterjave stroškov.

Proces razreševanja sumljivega zahtevka bomo prikazali kot kombinacije *osnovnih* vmesnih ciljev. Ad-hoc proces razreševanja je sicer bolj kompleksen in lahko v njem definiramo več vmesnih ciljev, vendar s tem ne bi pridobili na večji preglednosti prikaza uporabe modeliranja ad-hoc procesov.

Podobno kot smo pri modeliranju z UML diagramom aktivnosti v Poglavju 9.1.3 definirali podprocese, definirajmo za novo diagramsko tehniko vmesne možne cilje, ki služijo potrditvi ali ovržbi goljufije:

Vmesni cilj c_1 - potrditev nepravilnosti na postavkah računa Najbolj preprosto se nepravilnost kot goljufijo potrdi s pregledom cenikov storitev, ki jih imamo na voljo, in postavk na računu škodnega zahtevka. Na podoben način sicer deluje že predhodna detekcija goljufij, vendar lahko za potrditev neupravičenosti zahtevka "ročno" še enkrat preverimo, če se računi in ceniki storitev ne ujema in tako potrdimo obstoj goljufije.

Vmesni cilj c_2 - izvedena dodatna preiskava s strani notranjega preiskovalca Za potrditev goljufije lahko angažiramo posebnega preiskovalca, ki preveri vse podatke o škodnem zahtevku. Podobno vlogo imajo že definirane zavarovalnice v oddelkih za likvidacijo, kjer običajno pred likvidacijo več vrednih primerov preverijo vsa dejstva o primeru.

Vmesni cilj c_3 - izvedena dodatna preiskava s strani zunanjega preiskovalca Podobno kot v prejšnji točki lahko uporabimo zunanjega preiskovalca, ki izdela revizijo škodnega zahtevka. Takšna revizija je včasih potrebna zaradi objektivnosti in transparentnosti postopka potrjevanja goljufij.

Našteli bi lahko še veliko vmesnih ciljev, s katerimi se ugotovi, ali je škodni zahtevek goljufija ali ne, vendar se vsi držijo podobnega koncepta. Bistveno je to, da so aktivnosti, ki jih je treba izvesti za posamezen omenjeni vmesni cilj, več ali manj nedoločene. Ne moremo vedeti vnaprej, kako bo postopal kakšen od raziskovalcev ali kako bo izvedena dodatna raziskava. Če bi hoteli to modelirati na nivoju aktivnosti, bi morali uporabiti zelo kompleksne diagrame potekov aktivnosti, da bi zajeli notranjo dinamiko. Poleg tega se način izvajanja dodatnih preiskav skozi čas spreminja, prav tako pa imajo raziskovalci vsak svoj pristop. Zato nas s stališča celotnega ad-hoc procesa ne zanima, *točno kako se bo izvedla potrditev goljufije*, temveč samo ali je goljufija potrjena ali ne.

Našteti vmesni cilji so med seboj neodvisni in se jih lahko izvaja v poljubnem vrstnem redu. Če se pri izvajanju enega izmed teh ciljev ugotovi, da škodni zahtevek ni goljufija, se lahko v posebnem primeru vseeno odločimo za izvedbo dodatnega preverjanja z uresničevanjem drugega vmesnega cilja. Če pa se skozi uresničevanje teh ciljev nabere dovolj dokazov, da ne gre za goljufijo, se ad-hoc proces konča in zahtevek se izplača.

Če se goljufija potrdi, je na voljo več načinov izterjave (glej Poglavje 9.1.2). Prikazali bomo možnost izterjave preko civilne tožbe.

V splošnem se zavarovalnice izogibajo pravnim postopkom, saj so dragi, trajajo dolgo časa, poleg tega pa je izid običajno negotov. S pomočjo pravnikov in ustaljenih praks smo

definirali preprosto generalizacijo postopka, kjer gre zavarovalnica v civilno tožbo proti IZS. Zopet poudarjamo, da je za potrebe prikaza delovanja koncepta modeliranja ad-hoc procesa uporabljena poenostavljena oblika dejanskega procesa civilne tožbe. Vmesni cilji, ki so možni pri takšnem postopku, so:

Vmesni cilj c_4 - napisana civilna tožba Pravna služba mora napisati tožbo. Ker je treba pri tem upoštevati mnogo faktorjev, ki so odvisni od konkretnega primera, je praktično nemogoče vnaprej napovedati, katere vse aktivnosti in v kakšnem vrstnem redu so za doseg tega vmesnega cilja potrebne.

Vmesni cilj c_5 - dosežena izvensodna poravnava V vsakem primeru se skozi proces civilne tožbe poizkuša doseči izvensodno poravnavo, ki je za zavarovalnico najhitrejši način izterjave. V primeru, da se ta vmesni cilj uresniči, se ad-hoc proces konča, v nasprotnem primeru pa se izvajanje vrne nazaj v proces tožbe.

Vmesni cilj c_6 - sprožen proces tožbe Predpogoj za uresničevanje tega vmesnega cilja je izdelana civilna tožba. Ko se proces tožbe sproži na sodišču, ad-hoc proces prestopi meje združbe in izvajanje se prepusti zunanji združbi, na kar mi nimamo bistvenega vpliva. Predvsem lahko z različnimi artefakti podpremo pravno službo pri opravljanju njihovega dela.

Vmesni cilj c_7 - spremljanje in podpora postopka civilne tožbe Če pride do procesa civilne tožbe, na potek te ne moremo bistveno vplivati, lahko pa z beleženjem in zbiranjem podatkov proces spremljamo. Gre za izrazito dinamičen proces, kjer so aktivnosti nepredvidljive in je možnih poti skozi ad-hoc proces preveč, da bi jih lahko modelirali.

Za uresničitev končnega cilja izterjave neupravičenih stroškov so možne kombinacije doseganja vmesnih ciljev, napisane v obliki resničnostne tabele, naslednje:

	c_1	c_2	c_3	c_4	c_5	c_6	c_7
e_1	1	0	0	1	1	0	0
e_2	1	0	0	1	0	1	1
e_3	0	1	0	1	1	0	0
e_4	0	1	0	1	0	1	1
e_5	0	0	1	1	1	0	0
e_6	0	0	1	1	0	1	1

Vsaka vrstica ustreza eni kombinaciji vmesnih ciljev, ki vodi do končnega cilja ad-hoc procesa, indeks stolpca pa pove, za kateri vmesni cilj gre. Vidimo, da mora biti vedno izpolnjen vsaj eden izmed prvih treh vmesnih ciljev. To nam pove, da moramo nujno v procesu razreševanja najprej potrditi, ali gre za goljufijo ali ne. Zadnji štirje cilji nudijo dve možni kombinaciji. V obeh primerih se začne s civilno tožbo proti IZS, vendar v enem primeru pride do uresnitve vmesnega cilja, kjer se doseže izvensodna poravnava, v drugem primeru pa se proces nadaljuje v pravnem postopku na sodišču.

Pri implementaciji podpore procesa razreševanja bi upoštevali naravo samega procesa. Zaradi velikih stroškov, dolgotrajnosti in nepredvidljivega razpleta pravnih postopkov, se zavarovalnice običajno izognejo tožbam. Zato so najbolj ugodne vrstice za dosego končnega cilja ad-hoc procesa neparne vrstice, kjer se izognemo pravnemu postopku. Takšno razvrščanje bi dosegli z ustrezno izvedbo cenilne funkcije $f_c(F(e_i), t)$ in hevristične funkcije $h(c_i, t)$. Slednja bi uporabljala hevristiko, ki bi vse vmesne cilje, ki proces nadaljujejo na sodišču, ocenila kot slabe. Posledično bi tudi cenilna funkcija vsako konjunkcijo, ki vsebuje takšne cilje, ocenila za "drago". Tako bi se procesni agent odločal za tiste kombinacije vmesnih ciljev, ki se izognejo sodišču.

V zgornji resničnostni tabeli so navedene samo kombinacije vmesnih ciljev, kjer se proces razreševanja nadaljuje s civilno tožbo. Če bi hoteli v procesni model vključiti še druge načine izterjave, npr. neformalno poravnavo ali kazensko ovadbo, bi v seznam vmesnih ciljev dodali nove vmesne cilje, resničnostno tabelo pa razširili z ustreznimi kombinacijami.

Z navedenimi vmesnimi cilji smo skrili podrobnosti izvajanja aktivnosti. Vsak od vmesnih ciljev predstavlja množico aktivnosti, ki jih v času modeliranja ne moremo predvideti. Predvidimo pa lahko kaj je treba narediti, da se obravnava sumljivega zahtevka konča. Kadarkoli bi pogledali stanja procesnih entitet, ki predstavljajo doseganje vmesnih ciljev, bi vedeli, kaj lahko pričakujemo vnaprej in koliko so posamezni ad-hoc procesi že napredovali.

Za potrebe prikaza koncepta modeliranja ad-hoc procesov smo pri razgradnji ciljev ostali na dokaj visokem, konceptualnem nivoju, vendar bi lahko vsakega od teh vmesnih ciljev dodatno razgradili na podrobnejše vmesne cilje. Stopnja razgradnje je poljubna oz. se konča na nivoju, ko je treba definirati konkretna zaporedja aktivnosti in bi v primeru ad-hoc procesa to predstavljalo zapleten procesni model.

S poslovnega vidika modeliranja je diagram celotnega procesa razreševanja sestavljen iz diagramov posameznih vmesnih ciljev. Kombinacije vmesnih ciljev so predstavljene v obliki resničnostne tabele. Z implementacijskega vidika pa je treba v hevristično funkcijo zajeti takšne hevristike, ki ocenijo vmesne cilje, povezane s pravnimi postopki, nizko in dajo prednost izvensodni poravnavi.

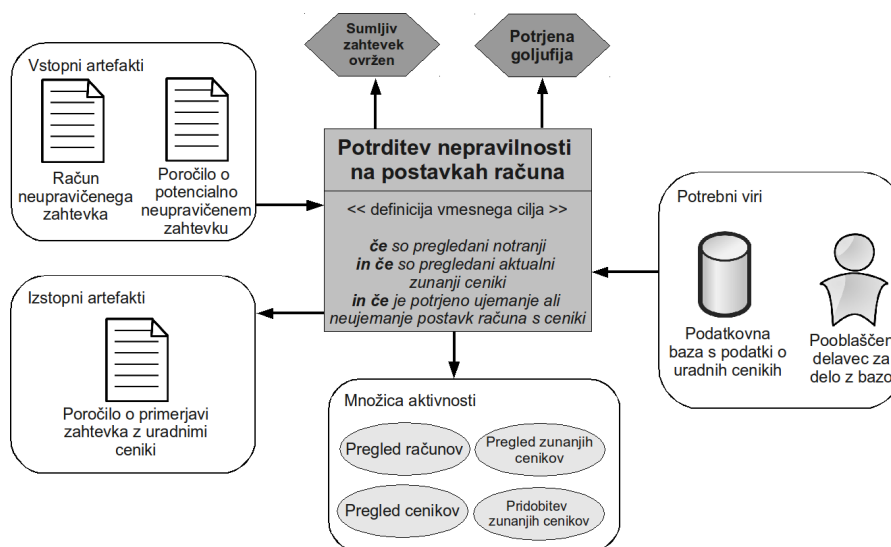
V nadaljevanju sledi opis dveh zgledeh vmesnih ciljev.

9.2.2 Primer: vmesni cilj *potrditev nepravilnosti na postavkah računa*

Prvi vmesni cilj, ki ga bomo prikazali, enkapsulira množico aktivnosti, ki so potrebne za preverjanje obstoja goljufije v določenem sumljivem zahtevku.

Jedro vmesnega cilja je definicija, kdaj je vmesni cilj izpolnjen. Z uporabo if...then pravil smo definirali, da je vmesni cilj dosežen, ko so preverjeni vsi možni vidiki podatkov na računu. Treba je torej potrditi ali ovržti možnost, da se je sistem za detekcijo goljufij pri iskanju sumljivih primerov zmotil in podatke napačno interpretiral.

Za doseganje tega cilja moramo imeti na voljo dva vira. Prvi vir so ustrezne podatkovne baze z uradnimi ceniki. Drugi vir je delavec, ki ima dostop do teh baz in je



Slika 9.3: Primer vmesnega cilja *potrditev nepravilnosti na postavkah računa*

usposobljen za iskanje ustreznih podatkov.

Vstopni artefakti za vmesni cilj so dokumenti, povezani z dokazovanjem sumljivosti zahtevka. Na diagramu smo navedli dokument, ki ga izdelata sistem za detekcijo goljufij. Prav tako je treba predložiti račun in ostalo dokumentacijo, ki spremlja zahtevek.

Med izstopne artefakte sodi poročilo o primerjavi zahtevka z uradnimi ceniki z ustrezno razlago. To poročilo je rezultat aktivnosti, ki so izvedene tekom doseganja vmesnega cilja.

Množica aktivnosti je določena s štirimi glavnimi skupinami aktivnosti, ki se odvijajo za uresničitev vmesnega cilja. Gre za pregled računov in dokumentacije ter pregled cenikov. Cenike lahko pridobimo iz lokalne podatkovne baze, lahko pa jih zaradi objektivnosti in transparentnosti pridobimo iz zunanjih virov. V razdelek *množica aktivnosti* bi lahko dodali še marsikatero aktivnost, ki se jih izvaja pri preverjanju cenikov in računov, vendar v času modeliranja ne moremo predvideti vseh. V trenutku modeliranja npr. ne vemo, ali bo potrebno mogoče kontaktirati računovodstvo preiskovanega IZS za dodatna pojasnila, ali pa mogoče preveriti, da se ni s plačilom zataknilo na banki. Prav tako ne moremo določiti vrstnega reda njihovega izvajanja. Vemo pa, da vse te aktivnosti vodijo k uresnitvi vmesnega cilja, katerega uresničitveni pogoji so opisani v središčnem gradniku diagrama.

Zadnja dva razdelka opisujeta dogodke, ki so pomembni za doseganje vmesnega cilja. Za vmesni cilj *potrditev nepravilnosti na postavkah računa* so pomembni predvsem izhodni dogodki. V primeru, da se goljufija potrdi, se na dogodkovno vodilo okolja pošlje dogodek z informacijo o potrditvi goljufije. Ad-hoc proces se v tem primeru nadaljuje. Če pa se ugotovi, da je plačilo upravičeno, se sproži dogodek ki pove, da je treba izvesti plačilo in da je zahtevek upravičen.

9.2.3 Primer: vmesni cilj *spremljanje in podpora postopka civilne tožbe*

Na Sliki 9.4 je prikazan primer vmesnega cilja *spremljanje in podpora postopka civilne tožbe*.

Modeliranje pravnega postopka na sodišču je zelo kompleksen problem, ker so pravni procesi zelo zapleteni. Impliciten dokaz za to je bogata pravniška literatura, ki opisuje pravne postopke. Vsak pravni postopek se lahko izvede na veliko različnih načinov. Stranke se lahko tekom procesa dogovorijo na različne načine in zato ne obstaja vnaprej definiranega vzorca. Za vsak pravni postopek bi lahko rekli, da je unikaten in ga je težko prikazati z nekimi splošnimi rigorozno določenimi modeli.

Ko se v ad-hoc procesu procesni agent odloči za pravni postopek civilne tožbe, se potek ad-hoc procesa prenese izven združbe. Na potek procesa tako ne moremo več bistveno vplivati. Lahko pa ponudimo ustrezno podporo pravni službi, ki je odgovorna za izvajanje procesa. Poleg tega lahko znižamo entropijo ad-hoc procesa tako, da beležimo, kaj se dogaja v pravnem postopku in na ta način vsaj ocenimo, kakšne možnosti imamo za izterjavo.

Vmesni cilj je dosežen takrat, ko je pravni postopek civilne tožbe izpeljan do konca in na sodišču razglasijo pravnomočno sodbo. Če je ta ugodna za zavarovalnico, potem pride do izterjave in povrnitve stroškov neupravičenega škodnega zahtevka. V nasprotnem primeru pa lahko z dodatnimi vmesnimi cilji v ad-hoc procesu poskrbimo za nadaljevanje procesa v obliki pritožb na sodiščih. Seveda mora procesni agent glede na poslovno logiko preračunati, ali se nadaljevanje procesa zaradi visokih stroškov sploh izplača.

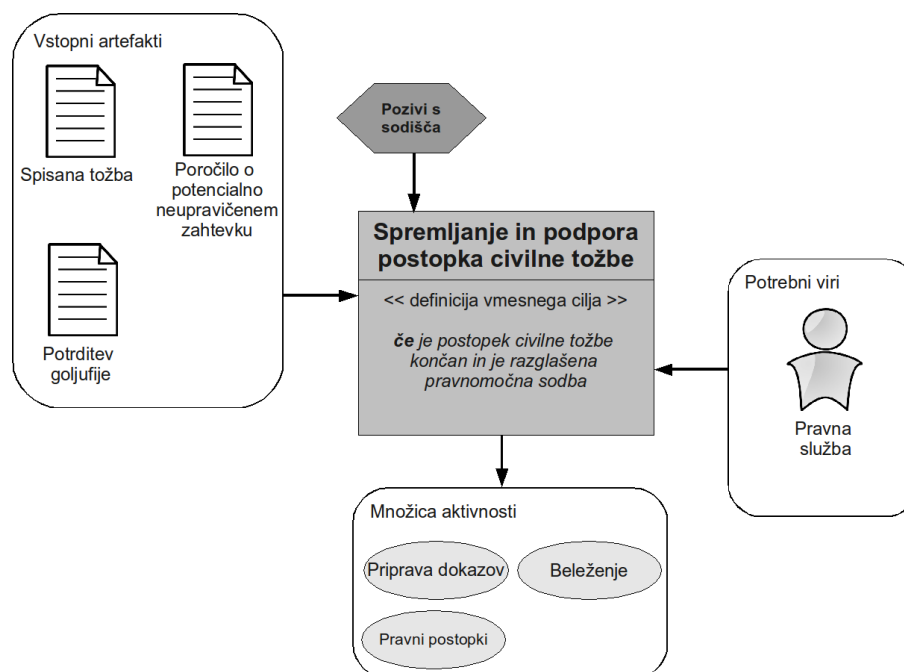
Praktično edini potreben vir za doseganje vmesnega cilja je pravna služba. Seveda pravni službi za učinkovito delovanje nudimo ustrezno podporo. Ta je v obliki vstopnih artefaktov, med katerimi je celotno možno gradivo, ki dokazuje neupravičenost škodnega zahtevka oz. da gre za goljufijo.

Ker je namen vmesnega cilja spremljanje postopka civilne tožbe, so za komunikacijo potrebni zunanji dogodki, ki nosijo informacijo o dogajanju na sodišču. Zunanji dogodki so zato, ker dejansko prihajajo izven meja združbe.

Vmesni cilj spremljanja in podpore postopka civilne tožbe je lep primer, ko ne moremo vnaprej definirati zaporedja množice aktivnosti, ki se bodo izvajale. Kot prvo je sama narava pravnih postopkov takšna, da je težko napovedati zaporedje izvajanja aktivnosti, kot drugo pa se te aktivnosti večinoma ne izvajajo znotraj združbe. Edine aktivnosti, ki jih lahko napovemo in zapišemo v razdelek *množica aktivnosti* so tiste, s katerimi nudimo podporo pravni službi.

Omeniti je treba, da smo se v magistrski nalogi v prvi vrsti osredotočali na modelirno tehniko kot način modeliranja in spoznavanja ad-hoc procesa. Hkrati smo nakzali, kako se da v primeru uporabe predlaganega koncepta izvesti implementacijo podpore, vendar se na to področje nismo osredotočali.

Oba opisana primera vmesnih ciljev predstavljata množice aktivnosti, ki jih ne moremo podrobno določiti vnaprej. Če bi hoteli te aktivnosti modelirati v kontekstu običajnih modelirnih pristopov, bi zaradi veliko možnih načinov izvajanja dobili neberljive in komple-



Slika 9.4: Primer vmesnega cilja *spremljanje in podpora postopka civilne tožbe*

ksne diagrame, ki ne bi imeli prave vrednosti ne za poslovnega uporabnika ne za kasnejše implementatorje podpore. Z obema primeroma smo pokazali, kako lahko z vmesnimi cilji, ki jih v posameznih stanjih procesne entitete dosega, abstrahiramo notranjo dinamiko aktivnosti.

Hkrati smo z opisom celotnega ad-hoc procesa prikazali, kako je v podpornih procesih težko vnaprej definirati celoten potek procesa. Z razdelitvijo ad-hoc procesa na vmesne cilje se osredotočimo na to, *kaj* vse je treba narediti in tako izpustimo vse podrobnosti samega izvajanja aktivnosti. Na ta način se izognemo kompleksnim diagramom poteka aktivnosti, ki bi sicer nastali, če bi hoteli zajeti vso dinamiko procesa. Bistvo modeliranja ad-hoc procesa je, da vso kompleksnost in dinamiko izvajanja aktivnosti enkapsuliramo z vmesnimi cilji. S tem ko vemo, kateri vmesni cilj trenutno v ad-hoc procesu uresničujemo, pa znižamo entropijo dogajanja in lahko nudimo ustrezno podporo oz. informacije za vodenje poslovanja.

Iz opisanega sledi, da smo z novo modelirno tehniko izvedli premik v ciljno usmerjenost in se nismo več osredotočali na aktivnosti in njihova zaporedja. Ker se zaporedje aktivnosti določa sproti tekom izvajanja ad-hoc procesa, je modeliranje z obstoječimi modelirnimi tehnikami zelo nepregledno. To smo pokazali v Poglavlju 9.1.3, kjer je že konceptualni diagram podprocesov zaradi velikega števila povezav precej nepregleden. Opazimo pa, da je takšen konceptualen diagram podprocesov, ki združujejo aktivnosti, ki so potrebne za nek delni cilj, dobra osnova za izvedbo diagrama vmesnih ciljev. Iz izsledkov iz Poglavlja 5 in študije primera lahko zaključimo, da je nova diagramska tehnika za modeliranje ad-hoc procesov bolj primerna kot ostale, uveljavljene modelirne tehnike. Na tem mestu

pa še enkrat poudarjamo, da je nova diagramska tehnika samo razširitev ali dopnilo k obstoječim diagramskim tehnikam in se jo uporablja skupaj z ostalimi uveljavljenimi modelirnimi tehnikami.

Poglavje 10

Glavne ugotovitve

V poglavju sledijo glavne ugotovitve, do katerih smo prišli tekom izdelave magistrske naloge.

10.1 Današnji poslovni procesi in združbe

Pregled poslovnih procesov v sodobnih združbah pokaže, da so ti procesi vse bolj kompleksni in dinamični. Čas, ko je bilo za poslovanje združbe treba definirati nekaj ključnih poslovnih procesov, vsa ostala podpora pa je bila razpršena in šibko definirana v poslovnih funkcijah, je minil. Podpora poslovnih procesov je danes strateškega pomena in zato se je treba ukvarjati tudi s podpornimi procesi, ki so manj strukturirani in jih je posledično težje modelirati in podpreti.

Dinamika v poslovnih procesih vnaša v delovanje združbe entropijo. Danes je zaradi zahtevnih trgov v vsakem trenutku nujno potrebno poznati vsak del poslovanja združbe. Upravniki in vodstveni kader mora imeti na razpolago kvalitetne informacije o poteku poslovnih procesov, da lahko na podlagi tega sprejemajo odločitve o poslovanju. Dejstvo je, da dinamika v poslovnih procesih to otežuje in jo je treba zato ustrezno obravnavati.

Ugotovili smo, da je dovršen del poslovnih procesov dinamičen do te mere, da ustreza definiciji ad-hoc procesov. Danes si ne moremo več privoščiti, da bi takšne procese ponostavili in jih jemali kot običajne poslovne procese. Dinamika je ena izmed glavnih lastnosti takšnih procesov in zato jo je treba ustrezno modelirati in obravnavati.

Današnje združbe so zelo kompleksni sistemi. Zaradi razvoja IKT ni več omejitev glede prostora in razdalj. Vse več opravil se opravlja na računalniški infrastrukturi. Viri poslovnih procesov so običajno storitve, razpršene po svetovnem spletu, do njih pa dostopamo preko mrežnih povezav. Poslovanje združbe se torej preseli iz omejenega, kontroliranega prostora v globalni prostor, ki je veliko bolj dinamičen in nepredvidljiv. Upoštevanje dinamike ad-hoc procesov mora nujno spremljati tudi ustrezna percepcija dinamike izvajalnega okolja ad-hoc procesov. Združbe niso več statične strukture, ki zagotavljajo točno določeno okolje za izvajanje poslovnih procesov, temveč so dinamične in kompleksne in se jim mora delovanje ad-hoc procesov stalno prilagajati.

10.2 Ciljna usmerjenost

Ena izmed glavnih ugotovitev magistrske naloge je dejstvo, da je treba za ustrezno obravnavo dinamike poslovnih procesov izvesti premik od procesne usmerjenosti k ciljni usmerjenosti.

Poslovne procese se pri običajnih pristopih dojema kot določena zaporedja aktivnosti, ki pripeljejo do končnega rezultata procesa. Glavni poudarek je na tem, *kako* se poslovni proces izvaja. Procesni diagrami, ki tako nastanejo, prikažejo točna zaporedja in odvisnosti med aktivnostmi poslovnih procesov.

V nasprotju pa s ciljno usmerjenostjo ne odgovarjamo na vprašanje *kako*, temveč *kaj je treba v poslovnem procesu narediti*. Če so procesi dinamični, ne moremo vnaprej določiti, kako se bodo izvajali. Če bi hoteli z običajnimi pristopi točno zmodelirati dinamične procese, bi dobili nepregledne in kompleksne diagrame. Poleg tega nam že sama definicija dinamičnosti da vedeti, da se je ne da na takšen način vnaprej statično zmodelirati.

Da bi zajeli dinamiko smo ad-hoc procese namesto na aktivnosti razdelili na vmesne cilje. Vsak vmesni cilj je enkapsulacija dinamične skupine aktivnosti, ki je ne moremo določiti in zmodelirati vnaprej. Postavljeni so samo jasni pogoji, kdaj je določen vmesni cilj dosežen. S tem zagotovimo, da so se izvedle aktivnosti iz dinamične množice, ki je enkapsulirana z vmesnim ciljem. Ko hočemo odgovoriti na vprašanje, kaj je še treba v ad-hoc procesu narediti, da bo dosežen njegov cilj, lahko pogledamo, kateri vmesni cilji so že doseženi in katere je še treba.

Vmesni cilji se lahko izvajajo v različnih kombinacijah. Te kombinacije smo zapisali v obliki izjavnega računa. Med sabo naj bi bili čimbolj neodvisni. To pomeni, da jih lahko izvajamo v poljubnem vrstnem redu in da so viri, ki jih potrebujemo za doseganje različnih vmesnih ciljev, čimbolj disjunktni oz. različni.

10.3 Procesne entitete in procesni agenti

Vsak ad-hoc proces, ki se začne izvajati, ima v okolju svojo predstavitev v obliki procesne entitete. Procesna entiteta je instanca ad-hoc procesa. Lahko si jo predstavljamo kot strukturo ad-hoc procesa. Med doseganjem vmesnih ciljev procesna entiteta spreminja svoje stanje in zaseda ter sprošča vire iz okolja.

Izvajalni vidik ad-hoc procesa predstavlja procesni agent. Po vzoru s področja multi-agentnih sistemov smo za vsako procesno entiteto uvedli procesnega agenta, ki na samostojen in avtonomen način vodi procesno entiteto. Tehnične naloge procesnega agenta vključujejo izvajanje in uporabo procesne logike. To so navodila in pravila, kako lahko glede na stanje okolja procesni agent zaseda in sprošča vire in procesno entiteto vodi med vmesnimi cilji. Poslovna logika pa je zbirka tistih navodil in pravil, ki procesnemu agentu povedo kako naj to naredi. To vključuje različne cenilne in hevristične funkcije, ki izbirajo naslednji vmesni cilj v izvajanju ad-hoc procesa.

Z uvedbo koncepta procesne entitete, ki obstaja v okolju, in procesnega agenta, ki mora glede na stanje okolja voditi entiteto, smo razdružili pojma proces in okolje procesa na

dva samostojna koncepta. Ravno zaradi takšne samostojnosti lahko v poslovanje hitro in učinkovito vnašamo spremembe. Če se spremenijo pogoji poslovanja, lahko to spremenimo s pravili, ki jih uporablja procesni agent. Če se spremenijo viri v združbi, lahko to neodvisno od ad-hoc procesov spremenimo z definicijami komponent okolja. Dodajanje in odstranjevanje virov je za procesne agente transparentno, saj so avtonomni in morajo sami poskrbeti za komunikacijo in lociranje ustreznih virov.

10.4 Možnosti implementacije

Že pri določanju ciljev magistrske naloge smo si zadali, da bo nov način modeliranja ad-hoc procesov možno integrirati in implementirati z obstoječimi tehnologijami.

Koncept modeliranja ad-hoc procesov se lepo vklaplja v SOA arhitekturo. Prav tako so različni procesni strežniki in izvajalna okolja za delovne postopke dobra platforma za implementacijo podpore. Večino procesne in poslovne logike lahko izvajajo različni BRMS sistemi. Poleg tega bi lahko procesne agente implementirali s pomočjo različnih agentnih platform, ki so danes na voljo.

Procesna in poslovna logika procesnih agentov bi se lahko zapisala v obliki poslovnih pravil, za katere skrbijo BRMS sistemi. BRMS sistemi omogočajo uporabo formalnega poslovnega jezika, ki je namenjen poslovnim uporabnikom. Tako bi lahko že v procesu modeliranja poslovni uporabniki navodila za delovanje procesnih agentov zapisali v končni obliki in ne bi bilo treba ob implementaciji modelov prevajati v tehničen jezik.

Kar je še bolj pomembno je to, da bi lahko poslovni uporabniki v primeru uporabe BRMS sistemov za hranjenje procesne in poslovne logike procesnih agentov sami nadzorovali celotno logiko ad-hoc procesov in tako veliko hitreje vnašali spremembe, ki nastajajo v poslovanju združbe. Ko bi prišlo do spremembe virov, spremembe v načinu poslovanja, do novih predpisov ali nove zakonodaje, ne bi bilo več potrebno posredovanje IT osebja za prilagoditev izvajanja podpore ad-hoc procesov. BRMS sistemi ponujajo dobre vmesnike in postopke, s katerimi lahko poslovni uporabniki spreminjajo poslovna pravila brez potrebe po zaustavljanju celotne IKT infrastrukture in ponovnem prevajanju programske kode.

Poglavje 11

Zaključek

Podpora poslovnih procesov je danes za združbe strateška nuja in ne več samo konkurenčna prednost. Vodenje dela v obliki poslovnih procesov zagotavlja največjo dodano vrednost za stranko. S tem, ko popišemo, podpremo in izvajamo poslovne procese, jih lahko optimiziramo, skrajšamo in prihranimo na vloženi sredstvih in energiji. Vse to zagotavlja večje zadovoljstvo stranke, kar združbi omogoča preživetje na trgu.

Za učinkovito izvajanje poslovnih procesov je nujna dobra podpora. Z dobro podporo lahko dosežemo avtomatizacijo in hitrejšo izvajanje poslovnih procesov, poleg tega pa nam podpora nudi dober vpogled v dogajanje poslovanja. S tem znižamo entropijo poslovanja in vodilnim kadrom nudimo informacije, ki so potrebne za sprejemanje odločitev v poslovanju.

Ugotovili smo, da je današnja dobro razvita IKT povzročila širjenje podpore tudi na manj strukturirane poslovne procese. Če je na začetku IT področje zaradi omejenih tehničnih sposobnosti zmoglo podpreti zgolj najbolj strukturirane in predvidljive proizvodne procese, se mora danes podpora nuditi tudi podpornim (administrativnim) poslovnim procesom. Ti procesi so bolj dinamični in jih je zato težje podpreti z informacijsko tehnologijo.

Zaradi vse bolj razvitega terciarnega sektorja in zaradi velikih globalnih trgov, obstaja vse več podpornih procesov, ki jih je treba podpreti. Običajni pristopi modeliranja poslovnih procesov njihove dinamike ne opisujejo v dovoljšnji meri. Še vedno je treba pri takšnih procesnih modelih definirati zaporedja aktivnosti, kar pa v dinamičnih procesih ni možno. Običajni pristopi modeliranja poslovnih procesov so primerni za različne proizvodne poslovne procese, kjer so aktivnosti v izdelovalnem procesu določene vnaprej, za modeliranje dinamičnih procesov pa je treba ubrati drugačen pristop.

Definicija ad-hoc procesa, ki smo jo predlagali, poizkuša zajeti tako notranji kot zunanji vidik dinamike poslovnih procesov. Skladno z definicijo smo predlagali novo usmerjenost pri dojemljanju ad-hoc procesov. Procesno usmerjenost, ki trenutno prevladuje v običajnih pristopih modeliranja, smo zamenjali s ciljno usmerjenostjo. Zaradi nedoločljivosti zaporedja aktivnosti v ad-hoc procesih ne moremo več modelirati samega toka procesa, lahko pa modeliramo vmesne cilje, ki povedo, *kaj* je treba v ad-hoc procesu narediti. S takšnim pristopom nam ni treba več natančno modelirati zaporedij aktiv-

nosti, ampak jih enkapsuliramo in skrijemo v vmesne cilje, kar omogoča boljši prikaz in razumevanje celotnih ad-hoc procesov.

Procesno logiko smo razširili s poslovno logiko, s katero določimo vmesne cilje. Poslovni uporabnik se lahko tako tekom modeliranja ad-hoc procesov osredotoči na poslovni del procesov, medtem ko je podrobno procesno logiko treba razviti šele kasneje v fazi implementacije podpore. Bistvo procesnega modela je tako zajeta s poslovno logiko, kar omogoča dobro razumevanje delovanja združbe in se ne osredotoča na tehnične podrobnosti.

Pristop modeliranja ad-hoc procesov smo dobro analizirali in preizkusili na realnem primeru iz gospodarstva. Zmodelirali smo ad-hoc proces razreševanja sumljivih škodnih primerov v zavarovalništvu in prikazali uporabno vrednost novega pristopa. V prihodnje načrtujemo tudi implementacijo podpore, ki bo uporabljala nov pristop. Zato smo imeli že pri izdelavi modeliranja v mislih možnosti tehnične implementacije. Menimo, da je pristop z uporabo procesnih entitet in procesnih agentov tehnično možno izvesti z današnjimi obstoječimi procesnimi strežniki v kombinaciji z BRMS sistemi. Prav uporaba BRMS sistemov bo omogočila hitro in transparentno prevedbo zmodelirane poslovne logike, hkrati pa bo poslovnim uporabnikom naknadno omogočala hitro vnašanje sprememb v poslovanju združbe.

Literatura

- [1] S. Alter, *Information Systems: a Management Perspective*, ZDA: Addison-Wesley, 1999
- [2] C. Armistead in P. Rowland, *Managing Business Processes: BPR and Beyond*, New York, ZDA: John Wiley & Sons, 1996
- [3] C. Coulson-Thomas, *Business Process Re-engineering: Myth & Reality*, Velika Britanija: Kogan Page Limited, 1994
- [4] M. Dumas, W. M. P. van der Aalst in A. H. M. ter Hofstede, *Process-Aware Information Systems*, ZDA: John Wiley & Sons, 2005
- [5] H. E. Eriksson in M. Penker, *Business modelling with UML: Business Patterns at Work*, ZDA: John Wiley & Sons, 2000
- [6] M. Fowler, *UML Distilled*, ZDA: Addison-Wesley, 2004
- [7] H. J. Harrington, E. K. C. Esseling in H. van Nimwegen, *Business Process Improvement Workbook*, ZDA: McGraw-Hill, 1997
- [8] J. Jeston in J. Nelis, *Business Process Management*, Velika Britanija: Elsevier, 2006
- [9] M. B. Juric in K. Pant, *Business Process Driven SOA Using BPMN and BPEL*, Velika Britanija: Packt Publishing, 2008
- [10] G. Keller in T. Teufel, *SAP R/3 Prozessorientiert Anwenden: Iteratives Prozessprototyping zur Bildung von Wertschoepfungsketten*, Nemčija: Addison-Wesley, 1998
- [11] L. C. Kubeck, *Techniques for Business Process Redesign: Tying It All Together*, ZDA: John Wiley & Sons, 1995
- [12] M. May, *Business Process Management*, ZDA: Pearson education limited, 2003
- [13] B. M. Michelson, *Event-driven Architecture Overview*, BPM Center Report, 2006. Dostopno na: <http://BPMcenter.org>
- [14] M. Mihelčič, *Organizacija in Ravnateljstvo*, Ljubljana: Založba FE in FRI, 2008

- [15] M. Mihelčič, *Poslovne Funkcije*, Ljubljana: Založba FE in FRI, 2008
- [16] H. A. Reijers, *Design and Control of Workflow Processes*, Nemčija: Springer-verlag, 2003
- [17] A. W. Scheer, *Business Process Engineering, Reference Models for Industrial Enterprises*, Heidelberg, Nemčija: Springer-Verlag, 1998
- [18] C. M. Belinda, Y. C. J. Lin in M. Orłowska, "Customizing internal activity behaviour for flexible process enforcement", v zborniku *ADC '04: Proceedings of the 15th Australasian database conference*, Dunedin, Nova Zelandija, 2004, str. 189-196
- [19] J. Meng, A. Helal in S. Su, "An Ad-Hoc Workflow System Architecture Based on Mobile Agents and Rule-Based Processing", v zborniku *Proceedings of The International Conference on Artificial Intelligence*, Nevada, ZDA, 2000, str. 245-251
- [20] M. Z. Muehlen, J. Recker in M. Indulska, "Sometimes less is more: Are process modelling languages overly complex", v zborniku *EDOC Conference Workshop 2007, Eleventh International IEEE*, ZDA, okt. 2007, str. 197-204.
- [21] A. Polyvyanyy in M. Weske, "Hypergraph-Based modelling of Ad-Hoc Business Processes", v zborniku *Business Process Management Workshops, BPM 2008 International Workshops*, Milano, Italy, sep. 2008, str. 278-289
- [22] M. Owen in J. Raj, *BPMN and business process management*. Dostopno na: www.bpmn.org/Documents/6AD5D16960.BPMN_and_BPM.pdf
- [23] M. Reichert, P. Dadam, M. Jurisch, U. Kreher in K. Göser, "Architectural Design of Flexible Process Management Technology", v zborniku *Proceedings of the PRIMIUM Subconference at the Multikonferenz Wirtschaftsinformatik (MKWI)*, Garching, Nemčija, feb. 2008, dostopno na: <http://ceur-ws.org/Vol-328/paper6.pdf>
- [24] D. ShuiGuang, Y. Zhen, W. ZhaoHui in H. LiCan, "A workflow model supporting flexible process using activities combination at run time", v zborniku *Computer Supported Cooperative Work in Design*, maj 2004, str. 409-417
- [25] P. Wohed, W. M. P. van der Aalst, M. Dumas, A. H. M. ter Hofstede in N. Russel, "Pattern based analysis of BPMN", *BPM Center Report*, 2006. Dostopno na: <http://BPMcenter.org>
- [26] M. D. Backer, M. Snoeck in K. U. Leuven, "Business Process Verification: A Petri Net Approach", *SSRN eLibrary*, 2007, dostopno na <http://ssrn.com/paper=1094617>.
- [27] W. Binder, I. Constantinescu, B. Faltings, K. Haller, in C. Türker, "A Multiagent System for the Reliable Execution of Automatically Composed Ad-hoc Processes", *Autonomous Agents and Multi-Agent Systems*, št. 2, zv. 12, str. 219-237, 2006.

- [28] Š. Furlan in M. Bajec, "Holistic approach to fraud management in health insurance", *Journal of Information and Organizational Sciences*, št. 2, zv. 32, str. 99-114
- [29] V. Gruhn in R. Laue, "What business process modelers can learn from programmers", *Science of Computer Programming*, št. 1, zv. 65, str. 4-13, 2007.
- [30] S. Guo-Ling, L. Yi-jun in M. A. Wei-zhong, "Study on business process modelling based on object-oriented Petri net", *Chinese business review*, št. 1, zv. 7, 2008.
- [31] S. Ha in H.-W. Suh, "A timed colored Petri nets modelling for dynamic workflow in product development process", *Computers in Industry*, št. 2-3, zv. 59, str. 193-209, 2008.
- [32] M. Koubarakis in D. Plexousakis, "A formal framework for business process modelling and design", *Information Systems*, št. 5, zv. 27, str. 299-319, 2002.
- [33] S. McCready, "There is more than one kind of workflow software", *Computerworld*, zv. 2, str. 86-90, 1992.
- [34] A. W. Scheer in M. Nuttgens, "ARIS architecture and reference models for business process management", *Business Process Management - Models, Techniques and Empirical Studies*, ZDA, Springer, str. 301-317, sep. 2008.
- [35] M. Wang in H. Wang, "From process logic to business logic - A cognitive approach to business process management", *Information and Management*, št. 2, zv. 43, str. 179-193, 2006.
- [36] (Januar 2010) Staffware. Dostopno na: <http://www.staffware.com>
- [37] (Januar 2010) Unified modelling language. Dostopno na: <http://www.uml.org>
- [38] (Januar 2010) Object management group. Dostopno na: <http://www.omg.org>
- [39] (Januar 2010) Workflow management coalition. Dostopno na: <http://www.wfmc.org>
- [40] (Januar 2010) Business process management notation. Dostopno na: <http://bpmn.org>
- [41] (Januar 2010) Yet another workflow language. Dostopno na: <http://www.yawlfoundation.org>
- [42] (Januar 2010) National health care anti-fraud association. Dostopno na: <http://64.211.220.122/eweb/StartPage.aspx>
- [43] (Januar 2010) European healthcare fraud & corruption network. Dostopno na: <http://www.ehfcn.org>

- [44] (Januar 2010) National insurance crime bureau. Dostopno na: <https://www.nicb.org>
- [45] (Januar 2010) Insurance fraud bureau. Dostopno na: <http://www.insurancefraudbureau.org>