

U P O R A B N A

I N F O R M A T I K A

2011

ŠTEVILKA 3

JUL/AVG/SEP

LETNIK XIX



Izpitni centri ECDL

ECDL (European Computer Driving License), ki ga v Sloveniji imenujemo evropsko računalniško spričevalo, je standardni program usposabljanja uporabnikov, ki da zaposlenim potrebno znanje za delo s standardnimi računalniškimi programi na informatiziranem delovnem mestu, delodajalcem pa pomeni dokazilo o usposobljenosti. V Evropi je za uvajanje, usposabljanje in nadzor izvajanja ECDL pooblaščen ustanova ECDL Foundation, v Sloveniji pa je kot član CEPIS (Council of European Professional Informatics) to pravico pridobilo Slovensko društvo INFORMATIKA. V državah Evropske unije so pri uvajanju ECDL močno angažirane srednje in visoke šole, aktivni pa so tudi različni vladni resorji. Posebno pomembno je, da velja spričevalo v 148 državah, ki so vključene v program ECDL. Doslej je bilo v svetu izdanih že več kot 11,6 milijona indeksov, v Sloveniji več kot 17.000 in podeljenih več kot 11.000 spričeval. Za izpitne centre v Sloveniji je usposobljenih 15 organizacij, katerih logotipe objavljamo.



U P O R A B N A I N F O R M A T I K A

2011 ŠTEVILKA 3 JUL/AVG/SEP LETNIK XIX ISSN 1318-1882

Uvodnik

Znanstveni prispevki

- Jaka Žgajnar, Emil Erjavec, Stane Kavčič, Lidija Zadnik Stirn:
Uporaba metod operacijskih raziskav pri načrtovanju kmetijske proizvodnje 125
- Robert Dukarič, Matjaž B. Jurič:
Migracija obstoječih aplikacij na platforme za računalništvo v oblaku 136
- Viktorija Sulčič:
Računalniška in internetna pismenost slovenske mladine 147
- Martin Potočnik, Matjaž B. Jurič:
Integracija procesiranja kompleksnih dogodkov in storitveno usmerjenih arhitektur 158

Strokovni prispevki

- Mirica Šafran, Mateja Škornik, Dejan Skok, Borut Jereb:
Uporaba odprtokodnih in prosto dostopnih programskih orodij – primer dobre prakse 167

Informacije

- Poročilo o konferenci DSI 2011** 176
- z Islovarja** 179
- Koledar prireditev** 181

Ustanovitelj in izdajatelj

Slovensko društvo INFORMATIKA
Revija Uporabna informatika
Vožarski pot 12, 1000 Ljubljana

Predstavniki

Niko Schlamberger

Odgovorni urednik

Jurij Jaklič

Uredniški odbor

Marko Bajec, Vesna Bosilj Vukšič, Gregor Hauc, Jurij Jaklič, Milton Jenkins, Andrej Kovačič, Katarina Puc, Vladislav Rajkovič, Heinrich Reinermann, Ivan Rozman, Rok Rupnik, Niko Schlamberger, John Taylor, Mirko Vintar, Tatjana Welzer Družovec

Recenzenti

Marko Bajec, Marko Bohanec, Vesna Bosilj Vukšič, Dušan Caf, Srečko Devjak, Tomaž Erjavec, Matjaž Gams, Izidor Golob, Tomaž Gornik, Janez Grad, Miro Gradišar, Jozsef Györkös, Marjan Heričko, Mojca Indihar Štemberger, Jurij Jaklič, Milton Jenkins, Andrej Kovačič, Jani Krašovec, Katarina Puc, Vladislav Rajkovič, Heinrich Reinermann, Ivan Rozman, Rok Rupnik, Niko Schlamberger, Tomaž Turk, Mirko Vintar, Tatjana Welzer Družovec, Lidija Zadnik Stirn, Alenka Žnidaršič

Tehnična urednica

Mira Turk Škraba

Oblikovanje

Bons
Ilustracija na ovitku: Luka Umek za BONS

Prelom in tisk

Boex DTP, d. o. o., Ljubljana

Naklada

600 izvodov

Naslov uredništva

Slovensko društvo INFORMATIKA
Uredništvo revije Uporabna informatika
Vožarski pot 12, 1000 Ljubljana
www.uporabna-informatika.si

Revija izhaja četrtletno. Cena posamezne številke je 20,00 EUR. Letna naročnina za podjetja 85,00 EUR, za vsak nadaljni izvod 60,00 EUR, za posameznike 35,00 EUR, za študente in seniorje 15,00 EUR. V ceno je vključen DDV.

Revijo sofinancira Javna agencija za knjigo RS.

Revija Uporabna informatika je od številke 4/VII vključena v mednarodno bazo INSPEC.

Revija Uporabna informatika je pod zaporedno številko 666 vpisana v razvid medijev, ki ga vodi Ministrstvo za kulturo RS.

Revija Uporabna informatika je vključena v Digitalno knjižnico Slovenije (dLib.si).

© Slovensko društvo INFORMATIKA

Vabilo avtorjem

V reviji Uporabna informatika objavljamo kakovostne izvirne članke domačih in tujih avtorjev z najširšega področja informatike v poslovanju podjetij, javni upravi in zasebnem življenju na znanstveni, strokovni in informativni ravni; še posebno spodbujamo objavo interdisciplinarnih člankov. Zato vabimo avtorje, da prispevke, ki ustrezajo omenjenim usmeritvam, pošljejo uredništvu revije po elektronski pošti na naslov ui@drustvo-informatika.si ali po pošti na naslov Slovensko društvo INFORMATIKA, Vožarski pot 12, 1000 Ljubljana.

Avtorje prosimo, da pri pripravi prispevka upoštevajo navodila, objavljena v nadaljevanju.

Za kakovost prispevkov skrbi mednarodni uredniški odbor. Članki so anonimno recenzirani, o objavi pa na podlagi recenzij samostojno odloča uredniški odbor. Recenzenti lahko zahtevajo, da avtorji besedilo spremenijo v skladu s priporočili in da popravljeni članek ponovno prejmejo v pregled. Uredništvo pa lahko še pred recenzijo zavrne objavo prispevka, če njegova vsebina ne ustreza vsebinski usmeritvi revije ali če članek ne ustreza kriterijem za objavo v reviji.

Pred objavo članka mora avtor podpisati izjavo o avtorstvu, s katero potrjuje originalnost članka in dovoljuje prenos materialnih avtorskih pravic. Nenaročenih prispevkov ne vračamo in ne honoriramo. Avtorji prejmejo enoletno naročnino na revijo Uporabna informatika, ki vključuje avtorski izvod revije in še nadaljnje tri zaporedne številke.

S svojim prispevkom v reviji Uporabna informatika boste prispevali k širjenju znanja na področju informatike. Želimo si čim več prispevkov z raznoliko in zanimivo tematiko in se jih že vnaprej veselimo.

Uredništvo revije

Navodila avtorjem člankov

Članke objavljamo praviloma v slovenščini, članke tujih avtorjev pa v angleščini. Besedilo naj bo jezikovno skrbno pripravljeno. Priporočamo zmernost pri uporabi tujk in – kjer je mogoče – njihovo zamenjavo s slovenskimi izrazi. V pomoč pri iskanju slovenskih ustreznih priporočamo uporabo spletnega terminološkega slovarja Slovenskega društva Informatika Islovar (www.islovar.org).

Znanstveni članek naj obsega največ 40.000 znakov, strokovni članki do 30.000 znakov, obvestila in poročila pa do 8.000 znakov.

Članek naj bo praviloma predložen v urejevalniku besedil Word (*.doc ali *.docx) v enojnem razmaku, brez posebnih znakov ali poudarjenih črk. Za ločilom na koncu stavka napravite samo en prazen prostor, pri odstavkih ne uporabljajte zamika.

Naslovu članka naj sledi za vsakega avtorja polno ime, ustanova, v kateri je zaposlen, naslov in elektronski naslov. Sledi naj povzetek v slovenščini v obsegu 8 do 10 vrstic in seznam od 5 do 8 ključnih besed, ki najbolje opredeljujejo vsebinski okvir članka. Pred povzetkom v angleščini naj bo še angleški prevod naslova, prav tako pa naj bodo dodane ključne besede v angleščini. Obratno pa velja v primeru predložitve članka v angleščini.

Razdelki naj bodo naslovljeni in oštevilčeni z arabskimi številkami.

Slike in tabele vključite v besedilo. Opremite jih z naslovom in oštevilčite z arabskimi številkami. Vsako sliko in tabelo razložite tudi v besedilu članka. Če v članku uporabljate slike ali tabele drugih avtorjev, navedite vir pod sliko oz. tabelo. Revijo tiskamo v črno-beli tehniki, zato barvne slike ali fotografije kot original niso primerne. Slik zaslonov ne objavljamo, razen če so nujno potrebne za razumevanje besedila. Slike, grafikoni, organizacijske sheme ipd. naj imajo belo podlago. Enačbe oštevilčite v oklepajih desno od enačbe.

V besedilu se sklicujte na navedeno literaturo skladno s pravili sistema APA navajanja bibliografskih referenc, najpogostejše torej v obliki: (Novak & Kovač, 2008, str. 235). Na koncu članka navedite samo v članku uporabljeno literaturo in vire v enotnem seznamu po abecednem redu avtorjev, prav tako v skladu s pravili APA. Več o APA sistemu, katerega uporabo omogoča tudi urejevalnik besedil Word 2007, najdete na strani <http://owl.english.purdue.edu/owl/resource/560/01/>.

Članku dodajte kratek življenjepis vsakega avtorja v obsegu do 8 vrstic, v katerem poudarite predvsem strokovne dosežke.

Spoštovani bralke in bralci,

v tej številki revije Uporabna informatika so objavljeni izbrani prispevki s konference Dnevi slovenske informatike 2011, ki so bili razširjeni za objavo v reviji in poglobljeno obravnavajo zanimive teme s področja operacijskih raziskav, računalništva v oblaku, internetne pismenosti, integracije s pomočjo storitveno usmerjenih arhitektur ter odprtokodnih sistemov.

Osemnajsta konferenca Dnevi slovenske informatike je potekala od 18. do 20. aprila 2011 v Portorožu pod geslom »Nove razmere in priložnosti v informatiki kot posledica družbenih sprememb«. Prispevke je predstavilo devet vabljenih vrhunskih tujih plenarnih predavateljev in več kot petdeset vrhunskih slovenskih vabljenih predavateljev v več kot stotih predavanjih v desetih tematskih sklopih, treh delavnicah in na okrogli mizi. Teme konference so pokrivalo področja, kot so poslovne aplikacije, poslovna inteligenca, menedžment poslovnih procesov, računalništvo v oblaku, informacijska varnost in upravljanje tveganj, nove priložnosti e-poslovanja, vodenje projektov in upravljanje odnosov z izvajalci, upravljanje informatike, informatika v javnem sektorju, podpora poslovnemu odločanju in operacijske raziskave.

Izbrani prispevki s konference, objavljeni v tej številki revije Uporabna informatika, pokrivajo področja uporabe metod operacijskih raziskav pri načrtovanju proizvodnje, migracije obstoječih aplikacij na platforme za računalništvo v oblaku, računalniško in internetno pismenost slovenske mladine, integracijo procesiranja kompleksnih dogodkov in storitveno usmerjenih arhitektur ter uporabo odprtokodnih in prosto dostopnih programskih orodij – primer dobre prakse.

Želim vam prijetno in zanimivo branje revije Uporabna informatika. Hkrati vas vabim k sodelovanju na devetnajsti konferenci Dnevi slovenske informatike 2012.

*Matjaž B. Jurič,
gostujoči urednik*

Najava

3. konference Informatika v javni upravi

Spoštovani,
obveščamo vas, da bo
28. in 29. novembra 2011 na Brdu pri Kranju
potekala 3. konferenca **Informatika v javni upravi**.

Konferenca bo potekala v organizaciji Ministrstva za javno
upravo in Ministrstva za visoko šolstvo, znanost in tehnologijo.
Prireditelj konference je Slovensko društvo Informatika.

Več informacij o konferenci na spletni strani www.iju2011.si.

Slovensko društvo Informatika



MANAGEMENT
POSLOVNIH
PROCESOV
2011

**Zakaj spremembe ne uspejo?
Umetnost managementa sprememb**

(Paul Levy)

MEDNARODNA POSLOVNA KONFERENCA

19. in 20. oktober 2011
kongresni center MONS v Ljubljani

Program in prijava za udeležence:
www.process-conference.org

Uporaba metod operacijskih raziskav pri načrtovanju kmetijske proizvodnje

Jaka Žgajnar, Emil Erjavec, Stane Kavčič, Lidija Zadnik Stirn
Univerza v Ljubljani, Biotehniška fakulteta, Jamnikarjeva 101, 1000 Ljubljana
jaka.zgajnar@bf.uni-lj.si, emil.erjavec@bf.uni-lj.si, stane.kavcic@bf.uni-lj.si, lidija.zadnik@bf.uni-lj.si

Izvleček

V prispevku izpostavljam kmetijstvo kot področje z izjemno kompleksnimi obratoslovnimi problemi. Podajamo pregled najpogosteje uporabljanih metod matematičnega programiranja za reševanje alokacijskih problemov upoštevajoč tveganje. Predstavljamo izvirni pristop povezovanja različnih metod, s katerim lahko podpremo kratkoročno operativno-taktično načrtovanje proizvodnje. Predlagani modelni pristop omogoča analizo dogajanja na hipotetičnem ali realnem kmetijskem gospodarstvu in modeliranje kompleksnih agrotehničnih, okoljskih in ekonomskih razmer. Z njegovo uporabo lahko v danih razmerah z različnih zornih kotov ovrednotimo razvojno perspektivnost kmetijskih gospodarstev in pokažemo na možnosti povečanja dodane vrednosti v kmetijstvu. Aplikacija modela je v prispevku prikazana na primeru kmetije, usmerjene v prirajo mleka.

Ključne besede: modeli za podporo odločanju, optimiranje, matematično programiranje, kmetijsko obratoslovje.

Abstract

The Application of Operation Research Methods in Agricultural Production Planning

In this paper agriculture as a sector with extremely complex management problems is presented. Most frequently applied mathematical programming methods for resource allocation considering risk are being presented. The paper illustrates innovative approach of linking different methods enabling support for short-term operative-tactical production planning. The proposed approach enables the analysis of hypothetical or real agricultural holding. It takes into account complex agro-technical, environmental and economic circumstances. The application of the developed model enables one to evaluate farmers' development perspectives from different points of view, indicating possibilities for increasing added value in farming. The model application has been illustrated in the case of a dairy farm.

Key words: decision support models, optimization, mathematical programming, farm management.

1 UVOD

Analizo procesov in sam postopek odločanja na mikro- (kmetijska gospodarstva), mezo- (raven regionalnega svetovanja) in tudi makroravni (nosilci odločanja v kmetijski politiki) je mogoče podpreti z razvojem primernih orodij. Ta morajo omogočiti modeliranje kompleksnih agrotehničnih, okoljskih in ekonomskih razmer ter zadostiti merilom formalne optimizacije, ki gospodarskim subjektom v danih razmerah z različnih vidikov ovrednoti njihovo razvojno perspektivnost. Za kmetijstvo je namreč tako kot za druge gospodarske sektorje značilno dinamično okolje, v katerem prihaja do prepletanja najrazličnejših odločitev na operativni, taktični in strateški ravni.

Pri načrtovanju gospodarjenja na posamezni kmetiji gre dejansko za temeljni alokacijski problem razvrščanja in razporejanja proizvodnih virov. Te želimo med izbrane aktivnosti razporediti tako, da dosežemo najvišjo (npr. dohodek) oz. najnižjo (npr. stroški) vrednost ciljne funkcije.

Deregulacija trgov, rastoče okoljske in druge družbene zahteve ter posledice podnebnih sprememb vodijo v vedno večja proizvodna, cenovna in dohodkovna nihanja, s tem tudi v tveganja, ki kmetijstvo v marsičem razlikujejo od drugih gospodarskih sektorjev (Aimin, 2010). Za kmetijska gospodarstva je tudi značilno, da so odločitve o izbiri aktivnosti sprejete veliko prej, kot so poznane tržne cene proizvodov, kar je pri liberaliziranih kmetijskih trgih še toliko bolj očitno in pomembno. Gre za izrazito izpostavljenost cenovno-tržnim silnicam, ki nastopijo pri nepredvidljivem obnašanju trgov (Huirne, Meuwissen in van Asseldonk, 2007). Nenehno spreminjanje cen pridelkov, razmer za pridelavo in drugih dejavnikov se odraža v nestabilnosti kmetijskih dohodkov in spreminjanju blaginje kmečkega prebivalstva (Harwood, Heifner, Coble, Perry in Somwaru, 1999),

zagotavljanje stabilnega dohodka pa ostaja vsakdanji problem nosilcev odločanja v kmetijski proizvodnji. Kmetijska gospodarstva so zato prisiljena v temeljitejšo obvladovanje tveganj in učinkovitejše ter večplastno načrtovanje gospodarjenja. To vključuje poleg iskanja alternativ in boljšega izkoriščanja obstoječih možnosti tudi povečevanja dodane vrednosti primarne pridelave in predelave.

Pri modeliranju v kmetijstvu je treba upoštevati dejstvo, da je praviloma težko pridobiti kakovostne primarne podatke, ki bi omogočali analizo konkretnega kmetijskega gospodarstva. Zato pri analizah navadno temeljimo na sekundarnem viru podatkov in standardiziranih izračunih, kot so modelne kalkulacije, ki jih v Sloveniji sistematično pripravlja Kmetijski inštitut Slovenije (Rednak, 1998). Vhodni podatki za analize so zato največkrat neke povprečne vrednosti, ki veljajo za določen tip kmetijskega gospodarstva ali tehnologije proizvodnje.

Pri večini raziskav s področja modeliranja kmetijskih obratoslovnih problemov pričakovane vrednosti pokritij ali dohodkov temeljijo na majhnem številu (deset ali celo manj) opazovanj, ki tvorijo matriko možnih stanj (Lien, Hardaker, van Asseldonk in Richardson, 2009). Potrebni podatki za daljše časovno obdobje največkrat niso dostopni. Če pa so, je njihova pomembnost manjša, saj se navadno v daljšem časovnem obdobju pogoji precej spremenijo (Lien in sod., 2009).

Zasledimo lahko različne pristope, s katerimi lahko izboljšamo in dopolnimo vrednost informacij kratkih časovnih serij podatkov. Najpogosteje je uporabljen postopek pretvorbe nominalnih cen v realne cene s pomočjo deflatorjev. Kadar gre npr. za spremembo tehnologije, genetski napredek in posledično spremembe pridelka, lahko uporabimo »postopek de-trend«, ki ga podrobno opisujejo Hardaker, Huirne, Anderson in Lien (2007). Ker pa obstajajo med kmetijskimi gospodarstvi pomembne razlike, lahko splošne statistične podatke preoblikujemo in obogatimo z informacijami za konkretno kmetijsko gospodarstvo (npr. ali dosega nadpovprečen ali podpovprečen pridelok). Hardaker in sod. (2007) opisujejo tudi metodo sintetiziranja podatkov, ki se najpogosteje uporablja pri analizah tveganja. Uspešno je bila uporabljena v številnih raziskavah (Lien in sod., 2009; Flaten in Lien, 2007; Lien in Hardaker, 2001). Njena značilnost je, da na podlagi mnenj strokovnjakov pridobimo ocene pričakovanih

vrednosti, ki naj bi jih pri posameznih aktivnostih dosegali v prihodnje.

Zaradi izpostavljene specifičnosti kmetijstva – zlasti z vidika dostopnosti kakovostnih primarnih podatkov – so se za primerne v tem sektorju izkazale metode s področja operacijskih raziskav, zlasti metode matematičnega programiranja. Pri tem pristopu potrebujemo ocene posameznih tehnoloških koeficientov, časovne serije pa uporabimo le za čim boljšo oceno povprečnih oz. pričakovanih vrednosti koeficientov.

Poleg optimizacijskih metod za podporo pri sprejemanju odločitev lahko uporabimo tudi pristop simulacijskega modeliranja. Pri tem je ključno povezovanje proizvodnih, finančnih in informacijskih tokov v integriran sistem (Pažek, Rozman, Bavec, Borec in Bavec, 2010). Nekateri avtorji so rezultate simulacijskih modelov povezali tudi z večkriterijsko analizo (npr. Zekri in Boughanmi, 2007; Pažek in sod., 2010) in tako prišli do teoretične rešitve. V prispevku se bomo omejili na uporabo metod matematičnega programiranja, ki temeljijo na optimizacijskem potencialu in so lahko neposredno uporabne za analizo odločanja na ravni kmetijskih gospodarstev.

V nadaljevanju prispevka na kratko predstavljamo ključne metode matematičnega programiranja s področja operacijskih raziskav, ki so uporabljene v predstavljenem orodju. Začnemo s klasičnim linearnim programiranjem in nadaljujemo z utemeljevanjem nujnosti večkriterijskega pristopa pri obravnavanju obratoslovnih problemov v kmetijstvu. Podrobneje predstavljamo ciljno programiranje kot eno izmed pogosteje uporabljenih metod. Nadaljujemo z modeliranjem tveganja v kontekstu pričakovanih vrednosti in varianc. Temu sledi kratka predstavitev razvitega elektronskega orodja in opis kmetijskega gospodarstva, na katerem je predstavljeno delovanje orodja. Kratki predstavitvi rezultatov sledi razprava o matematičnem modeliranju pri reševanju obratoslovnih problemov.

2 METODE MATEMATIČNEGA PROGRAMIRANJA ZA PODORO PRI ODLOČANJU

Operacijske raziskave in matematično programiranje so postali pomembno in široko uporabljeno orodje za analize v kmetijstvu in ekonomiki (Buysse, Huylenbroeck in Lauwers, 2007). Gre za pristop, ki zelo dobro združi neoklasično proizvodno teorijo z modeliranjem. Romero in Rehman (2003) poudarjata, da je vsem matematičnim metodam skupna teoretična

predpostavka, da mora dobljena optimalna rešitev zadostiti vsem omejitvam analiziranega problema ob hkratnem upoštevanju namenske funkcije, ki vključuje preference kmeta. Na podlagi klasične ekonomske teorije je to maksimiranje dobička, doseženega z maksimiranjem ene ciljne funkcije.

Najpreprostejša oblika matematičnega programiranja, za katero je nedvomno razvitih tudi največ reševalnih algoritmov, je klasično deterministično linearno programiranje (KDLP) (Hardaker in sod., 2007). Ključna predpostavka tega je, da so vsi odnosi linearne narave. Nedvomno lahko številne probleme povsem upravičeno poenostavimo prav s postopkom linearizacije. Hardaker in sod. (2007) ugotavljajo, da so dobljene rešitve poenostavljenih problemov lahko povsem zadovoljive, ob tem pa Ziolkowska (2009) izpostavlja predvsem enostavnost in preglednost modeliranja z linearnim programom. Ključni moment za presojo je seveda namen analize oz. orodja za podporo pri odločanju.

2.1 Enokriterijski ali večkriterijski pristop

Nekateri avtorji opozarjajo na potencialne nevarnosti enokriterijskega obravnavanja obratoslovnih problemov. Gomez-Limon, Arriaza in Riesgo (2003) opozarjajo, da v veliko primerih optimiranje samo na podlagi enega cilja lahko pomeni preveliko poenostavitev in s tem odstopanje od stvarnosti. Dejstvo je, da je odločanje in s tem tudi optimiranje kmetijske proizvodnje večkriterijski postopek, ki zasleduje celo vrsto ciljev različnih preferenc. V praksi najdemo številne potencialne cilje, ki se jim z rešitvijo želimo čim bolj približati. Nanašajo se lahko npr. na zagotavljanje stabilnega dohodka, zaposlovanje lastne delovne sile, minimiranje najetih strojnih storitev, doseganje konkurenčnosti, preprečevanje zaraščanja – izkoriščanje njivskih in travnih površin, maksimiranje donosnosti investicije,¹ minimiranje najetega kapitala, poslovanje pri čim manjšem tveganju, čim manjše obremenjevanje okolja idr. Iz tega izhaja tudi eden izmed glavnih očitkov klasičnemu determinističnemu linearnemu programiranju, ki pri iskanju optimalne rešitve upošteva samo en cilj (navadno maksimalen dohodek) kot edino merilo (Rehman in Romero, 1984; Rehman in Romero, 1987; Lara, 1993). V primeru maksimiranja dohodka ali pokritja to pomeni,

da se preveč zanašamo le na ozko ekonomski vidik, s tem pa spregledamo katerega izmed pomembnih dejavnikov (npr. okoljskih).

Kritiki klasičnega determinističnega linearnega programiranja izpostavljajo tudi problem togosti omejitev, pri katerih ni dovoljeno nikakršno odstopanje (Rehman in Romero, 1984). Nedvomno drži, da je dovolj dobra rešitev boljša od optimalne, a hkrati nerealne (Ignizio in Romero, 2003). Pogosto se tudi zgodi, da kompleksno linearno programiranje vsebuje omejitve, ki sočasno ne morejo biti izpolnjene in posledično sistem enačb nima možne rešitve.

Vse pomembnejša dejavnika pri načrtovanju in vodenju kmetijskih gospodarstev sta tudi okolje in varovanje okolja. Metode večkriterijskega programiranja so pogosto aplicirane prav na interdisciplinarnih področjih, ki povezujejo ekonomiko in varstvo okolja (Zekri in Boughanmi, 2007). Poleg klasičnega maksimiranja dohodka in drugih ekonomskih kazalnikov se pojavljajo tudi primeri minimiziranja erozije, uporabe fitofarmaceutskih sredstev, podtalnice, mineralnih hranil idr.

Romero in Rehman (2003) navajata, da sta bila Wheeler in Russell v drugi polovici sedemdesetih let prejšnjega stoletja prva, ki sta pri načrtovanju proizvodnje na ravni kmetijskega gospodarstva upoštevala večje število nasprotujočih si ciljev. V literaturi zasledimo veliko različnih metod, ki temeljijo na različnih tehnikah in iščejo kompromisne rešitve. V grobem jih lahko razdelimo na kvantitativne in kvalitativne metode. Za probleme načrtovanja proizvodnje, katere obravnavamo v tem prispevku, so primernejše kvantitativne metode, ki temeljijo na principu matematičnega programiranja z omejeno optimizacijo. Med njimi je ciljno programiranje (angl. goal programming – GP) nedvomno ključni pristop (Azmin in Tamiz, 2010). Gre za metodo, ki je bila in je še vedno najpogosteje uporabljena tehnika na področju večkriterijskega programiranja.

2.2 Ciljno programiranje

V letu 1955 sta ciljno programiranje razvila Charnes in Cooper in ga sprva poimenovala omejena regresija in šele šest let pozneje ciljni program (Jones in Tamiz, 2010). Metoda je svoj razmah doživela v sedemdesetih letih prejšnjega stoletja, kar Jones in Tamiz (2010) ter Martel in Aouni (1990) pripisujejo zlasti dejstvu, da gre za posebno, posplošeno obliko linearnega programiranja, katere osnovna filozofija je zelo jasna.

¹ Denimo prek maksimiranja kazalnikov, kot sta interna stopnja donosa (ISD) in neto sedanja vrednost (NSV) predlaganega projekta.

Tudi Steur in Na (2003), Aouni in Kettani (2001) ter Martel in Aouni (1998) popularnost pristopa pripisujejo dejstvu, da gre za izpeljanko linearnega programiranja in dostopnosti učinkovitih algoritmov za njegovo reševanje.

Ciljno programiranje velja za posebno obliko kompromisne metode reševanja večkriterijskih problemov, ki predpostavlja, da odločevalec (angl. decision maker – DM) pozna ciljne vrednosti in da lahko določi njihov relativen pomen (Liu, 2008). Martel in Aouni (1998) poudarjata, da ciljno programiranje omogoča upravljavcu ob hkratnem upoštevanju večjega števila ciljev iz množice možnih rešitev izbrati tisto, ki po vseh dosega najboljše vrednosti.

Odstopanje od ciljev omogočimo z dodatnimi spremenljivkami, ki jih opredelimo za vsak cilj posebej bodisi v pozitivno ali negativno smer. Dovoljena odstopanja prek deviacijskih spremenljivk naredijo model bolj prilagodljiv, s čimer razširimo polje možnih rešitev. Če je odstopanje določenega cilja kaznovano, pripišemo kazenskim koeficientom neničelne vrednosti, odstopanja pa so lahko dovoljena v obe strani ali pa samo v eno. Dobljeni rezultat pomeni kompromisno rešitev med običajno nasprotujočimi si cilji.

Doseganje ciljnih vrednosti tako pri reševanju ciljnega programiranja postane ključna strategija (Azmi in Tamiz, 2010). Ta predpostavlja, da odločevalec želi minimizirati razliko med doseženimi in definiranimi ciljnim vrednostmi. Pri teh modelih tako ne gre več za optimizacijo v osnovnem pomenu besede, saj ne obstoji nobena odločitev oz. rešitev, ki je najboljša po vsakem izmed vseh meril (Čančer, 2003). Pri reševanju problemov iz dejanskega sveta so namreč nasprotujoči si cilji dosegljivi le na račun »žrtvovanja« drugih ciljev – s tem pa je izpolnjen pogoj za doseg optimalne rešitve (Liu, 2008).

Za razliko od ciljne funkcije konvencionalnih metod (npr. linearno programiranje) namenska funkcija meri stopnjo »nedoseganja« zastavljenih ciljev in je ključni element ciljnega programiranja. Način merjenja odstopanj definira različico ciljnega programiranja. Pri vseh pa namenska funkcija teži k minimizirani stopnji neželenih odstopanj deviacijskih spremenljivk (Romero, 2004). Od analiziranega problema in filozofije večkriterijskega pristopa odločanja je odvisno, katera oblika je najbolj primerna. Dobljeni rezultati se namreč med različnimi namenskimi funkcijam pomembno razlikujejo (Romero, 2004).

Pri analizi obratoslovnih problemov, pri katerih vstopajo najrazličnejši kriteriji različnih prioritet, se *tehtano* ciljno programiranje (angl. Weighted Goal Programming – WGP) izkaže kot ustrezen pristop. Zasnovano je na principu Arhimedove ciljne funkcije, ki minimizira vsoto tehtanih odstopanj od posameznih ciljev. Konsenz je tako dosežen z minimiziranjem tehtane vsote nesoglasij (Gonzalez-Pachon in Romero, 1999).

2.3 Modeliranje tveganja

Poleg vse večje potrebe po večkriterijskem obravnavanju odločanja na ravni kmetijskih gospodarstev kot posledice povečane muldisciplinarnosti vpetosti kmetijstva v druga področja družbenega interesa in tudi same narave dela v kmetijstvu so pri odločanju na tem področju in posledično v raziskavah odločanja vse bolj prisotni tudi tveganje, negotovost in upravljanje s tveganji.

Hardaker in sod. (2007) poudarjajo, da tveganja pri analizah odločanja na kmetijskih gospodarstvih ne smemo zanemariti, saj nosilci odločanja v kmetijstvu niso naklonjeni tveganju in ga poskušajo zmanjševati s svojimi odločitvami. Na ravni kmetijskega gospodarstva se tveganje odraža na pričakovanem dohodku in posledično vpliva na kmetovo odločitev pri formuliranju proizvodnega načrta (Martins in Marques, 2007).

Matematično programiranje (angl. Mathematical Programming – MP), ki temelji na moderni portfeljski teoriji, je eno najbolj pogostih načinov analize odločanja v razmerah tveganja (Romero, 2000). Obravnavanje načrtovanja proizvodnje kot portfeljski problem namreč omogoča tehtanje med možnimi aktivnostmi, ki prispevajo zaščito in hkrati priložnost zaradi številnih nepredvidenih izdatkov (Kobzar, 2006). Gre za metodo, ki uporablja matematični koncept variance (kot merilo razpršenosti) za ovrednotenje tveganja pod pogojem, da so dohodki normalno porazdeljeni. Kobzar (2006) ugotavlja, da so mnenja poznavalcev o pogoju normalne porazdelitve še vedno deljena. Ob tem pa Hardeker in sod. (2007) poudarjajo, da na osnovi centralnega limitnega izreka lahko predpostavimo približno normalno porazdelitev. Skupni dohodek namreč predstavlja vsoto večjega števila naključnih spremenljivk (kombinacije različnih aktivnosti).

Če drži izhodišče normalne porazdelitve, lahko predpostavljamo, da se odločevalec odloča le na pod-

lagi pričakovanih vrednosti (povprečna vrednost) in variance (oz. standardne deviacije) kot mere tveganja. Pri svojih odločitvah tako odločevalec ne upošteva drugih značilnosti porazdelitve pričakovanih vrednosti (dohodkov), kot sta stopnja *asimetričnosti porazdelitve* (angl. skewness) in *stopnja sploščenosti* (angl. kurtosis). Modeliranje proizvodnega tveganja na ravni kmetijskega gospodarstva temelji na problemu maksimiranja pričakovane koristnosti. Namesto te pa lahko uporabimo splošnejšo obliko funkcije – model pričakovanih vrednosti in variance (angl. Expected value, Variance model – model E,V) (Escalante in Rejesus, 2008). Model temelji na stohastični proizvodni funkciji in oceni verjetnih porazdelitev pridelkov in cen (Rasmussen, 2004). Osnovno idejo je razvil Markowitz v petdesetih letih prejšnjega stoletja. Pionirsko delo na področju vključevanja tveganja v reševanje kmetijskih problemov pa je le štiri leta kasneje opravil Freund (Romero, 2000). Kljub temu da so agrarni ekonomisti model pričakovanih vrednosti in variance večkrat označili kot pomanjkljiv, pa Gomez-Limon in sod. (2003) navajajo nekaj del, ki opravičujejo njegovo uporabo. Osnovna ideja je neposredna povezava med pričakovanimi vrednostmi in tveganjem, kar pomeni, da moramo za boljši finančni rezultat prevzeti tudi večje tveganje in obratno.

2.4 Modeliranje in elektronska orodja

Matematično modeliranje je bilo v preteklosti prepuščeno predvsem informatikom in programerjem, zato so bili stroški razvoja programov v primerjavi z njihovo uporabnostjo zelo visoki, saj praviloma niso vključevali menedžerjev oz. upravljavcev, ki bolje poznajo problem in relacije znotraj določenega problema (Powell in Baker, 2010). Zaradi visokih stroškov programiranja številne rešitve, razvite v preteklosti v znanstvenoraziskovalnih krogih, tudi nikoli niso zaživele. Nedvomno pa tudi kmetijstvo zaradi svoje posebnosti poslovanja in večinoma konservativne narave gospodarjev ni bilo pretirano naklonjeno potencialno zanimivim rešitvam (kmetijstvo ni povpraševalo po njih). Poleg neugodne starostne strukture in nizke stopnje izobrazbe je ključni razlog v informacijski pismenosti in dejstvu, da dosežena ekonomika proizvodnje ni dovoljevala investiranja v podporna orodja.

V prispevku predstavljamo primer orodja, ki je zasnovano v obliki elektronskih preglednic. Ima številne prednosti, saj pri večini modelov lahko shajamo

s povsem osnovno matematično algebro (vključno s kvadratnimi, eksponentnimi in logaritemskimi funkcijami), osnovno logiko (izraženo s pogojnimi (angl. IF) stavki ali MAX funkcijo) in osnovnimi zakonitostmi verjetnosti. S tem zajamemo tudi modeliranje s predstavljenimi metodami matematičnega programiranja. Iz pregleda literature izhaja, da raziskovalci na kmetijskem področju razvijajo tudi »preproste« modele v obliki elektronskih preglednic, ki so jim v pomoč pri analizah (Winston, 2002; Buongiorno in Gilles, 2010; Powell in Baker, 2010; Žgajnar, 2011). Pri sprejemanju odločitev pa so lahko v pomoč tudi spretnejšim in računalniško pismenim kmetom.

3 MODEL IN NJEGOVA APLIKACIJA

V prispevku predstavljamo elektronsko orodje za večkriterijsko analizo odločanja na ravni kmetijskega gospodarstva, ki je podprto z različnimi modeli s področja operacijskih raziskav,² ki izkoriščajo optimizacijski potencial matematičnega programiranja. Vodilo pri pripravi orodja in izboru metod je bilo omogočiti analizo hipotetičnega ali konkretnega kmetijskega gospodarstva brez neposrednega vključevanja kmetijskega gospodarja v postopek reševanja. Takšen pristop omogoča potencialnim uporabnikom (nosilcem odločanja v kmetijski politiki, svetovalcem, raziskovalcem) proučevanje sprememb na določenih tipih kmetijskih gospodarstev brez njihovega podrobnejšega poznavanja in natančnih podatkov posameznega kmetijskega gospodarstva.

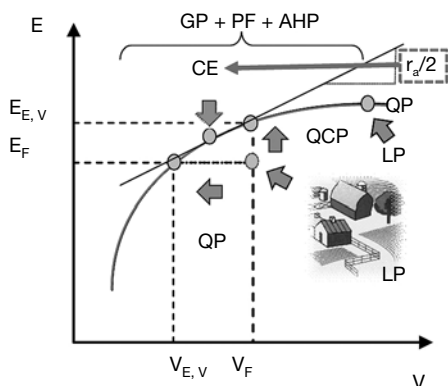
Namen predstavljenega orodja ni v tem, da bi uporabnik dobil v celoti eksaktno informacijo, ampak v osvetlitvi možnih alternativ na konkretnem gospodarstvu oz. tipu gospodarstva. Tako ima ambicijo nakazati smer njegovega razvoja oz. profesionalizacije.

Vsi modeli predpostavljajo optimizacijo na ravni celotnega kmetijskega gospodarstva v obdobju enega leta. Odločitve so sprejete na začetku posameznega leta. Predvidena ni nobena večja investicija, ampak le aktivnosti, povezane z operativnimi in taktičnimi odločitvami.

Z vsakim izmed modelov razrešujemo določeno problematiko, obravnavano pri analiziranju odločanja. Osnovni model predstavlja klasičen primer iskanja proizvodnega načrta, pri katerem se osredinimo le na doseženi ekonomski rezultat. Iščemo torej največji dohodek ob upoštevanju drugih proizvodnih

² Model je podrobneje predstavljen in opisan v Žgajnar (2011).

omejitev. Model temelji na klasičnem determinističnem pristopu linearnega programiranja. Zato je povsem pričakovano, da z njim pridemo do najvišjega dohodka, ki pa posledično vključuje tudi najvišjo stopnjo tveganja (slika 1 – zgoraj desno).



Legenda: LP – linearno programiranje (angl. Linear Programming); QP – kvadratno programiranje (angl. Quadratic Programming), QCP – omejeno kvadratno programiranje (angl. Quadratic Constrained Programming), r_A – absolutna mera tveganja (angl. Absolute Risk Aversion), CE – zanesljivi ekvivalent (angl. Certainty Equivalent), GP – ciljno programiranje (angl. Goal Programming), PF – kazenska funkcija (angl. Penalty Function), AHP – analitični hierarhični proces (angl. Analytic Hierarchy Process), E – pričakovana vrednost (angl. Expected Value), V – varianca.

Slika 1: **Položaj rešitev uporabljenih matematičnih modelov v prostoru pričakovanih vrednosti in varianc**

Za celovito analizo odločanja na ravni kmetijskega gospodarstva je pomembno poznavanje trenutnega stanja v smislu strukture proizvodnega načrta (katere aktivnosti so vključene) in doseženih ekonomskih rezultatov (dohodek idr.). Iz teh namreč izhajajo pomembni podatki, ki se nanašajo tako na upravljanje s tveganji kot tudi na večkriterijsko analizo. Trenutno proizvodnjo kmetijskega gospodarstva rekonstruiramo s pomočjo klasičnega linearnega programiranja s postopkom delne optimizacije (DO), pri čemer del (poznanih) aktivnosti fiksiramo, drugi del pa pustimo odprt. Z delno optimizacijo iščemo vrednosti tistih spremenljivk (aktivnosti), katerih vrednosti ne poznamo. To je lahko posledica agregiranih podatkovnih baz ali pa gre za transferne – netržne aktivnosti. Namen tega modela tako ni optimizacija proizvodnje, pač pa »ocena« trenutnega stanja na podlagi različnih tehnoloških, proizvodnih in logičnih omejitev.

Trenutno stanje potrebujemo kot izhodišče za izračun kmetovega odnosa do tveganja. Gre za pristop, ki ga podrobneje opisujejo Lien (2002) ter Žgaj-

nar in Kavčič (2011). Osnovna ideja je, da s pomočjo kvadratnega modela (angl. Quadratic Programming – QP), to je minimiziranja variance, izračunamo prvo točko ($E_F, V_{E,V}$), ki predstavlja proizvodni načrt z enakim finančnim rezultatom, kot ga dejansko dosega gospodarstvo (E_F), je pa bolj učinkovit z vidika tveganja. Enak finančni rezultat torej doseže z drugačnim proizvodnim načrtom z manjšim tveganjem. Druga točka je izračunana s pomočjo omejenega kvadratnega programiranja (angl. Quadratic Constrained Programming – QCP) in je primer maksimiranja pričakovanega dohodka. Z vidika doseženega finančnega rezultata gre za učinkovitejšo rešitev, saj je ta ob enakem tveganju (V_F), kot ga dosega kmetijsko gospodarstvo, boljši ($E_{E,V}$) (slika 1).

Po pristopu, ki ga podrobneje opisuje Lien (2002), na podlagi teh dveh točk ocenimo kmetov odnos do tveganja oz. absolutno mero tveganja (angl. Absolute Risk Aversion – r_A). Kot izhaja iz slike 1, gre za naklon premice med točkama na krivulji proizvodnih možnosti. To lahko izračunamo na več različnih načinov. Eden izmed pristopov je minimiziranje skupne variance in parametriziranje pričakovanega dohodka.

V celovit pristop je vključen tudi kvadratni model maksimiranja zanesljivega ekvivalenta (angl. Certainty Equivalent – CE). V tem primeru je ključna predpostavka, da poznamo kmetov odnos do tveganja, izražen kot absolutni koeficient odnosa do tveganja (r_A). Tega ocenimo v prejšnjem koraku na podlagi obeh točk na krivulji proizvodnih možnosti (slika 1). Kot izhaja iz literature, gre za proizvodni načrt, pri katerem je kmet ravnodušen med gotovo vrednostjo in zanesljivim ekvivalentom.

Vprašanje, ki se pri tem zastavi odločevalcu, je, katera od dobljenih rešitev v prostoru pričakovanih vrednosti in varianc je dejansko prava. Odgovor ni preprost in enoznačen. Odvisen je od namena in vidika analize. Da pa bi lahko poiskali rešitev, ki upošteva čim več vidikov, celovit pristop zajema tudi večkriterijsko optimizacijo. Temelji na filozofiji ciljnega programiranja, pri kateri rešitve posameznih že omenjenih modelov vstopajo kot ciljne vrednosti. Uporabili smo tehtano ciljno programiranje (angl. Weighted Goal Programming – WGP), katerega smo nadgradili s sistemom kazenskih funkcij (angl. Penalty Function – PF). Definiranje uteži in s tem pomena posameznega cilja je pomembna naloga tehtanega ciljnega programiranja, saj z njimi močno vplivamo na dobljeno rešitev. Njihov relativni pomen izraču-

namo s pomočjo metode analitičnega hierarhičnega procesa (angl. Analytic Hierarchy Process – AHP), ki na podlagi devetstopenjske lestvice omogoča primerjave po parih (Saaty, 1980). Pri metodi analitičnega hierarhičnega procesa je pomembno, da so izračunane relativne uteži za razvrščanje posameznega kriterija (cilja) dobljene ob predpostavki konsistentnosti.

Predstavljeni modeli kot tudi samo orodje so razviti v excelovem programskem okolju. To omogoča njihovo enostavno povezovanje in dopolnjevanje. Povezave in večino potrebnih operacij je avtomatiziranih s pomočjo makrov, zapisanih v obliki kode VBA (Visual Basic for Applications) v urejevalniku za Visual Basic.³ Zapis osnovnih matrik za reševanje matematičnih modelov kot tudi njihovo formiranje in množenje je tako povsem preprosto.

Modelno orodje je procesno odprt sistem. Po eni strani to pomeni, da je mogoče različno povezovanje modulov, njihovo dopolnjevanje in prilagajanje posameznim analiziranim primerom, po drugi strani pa odprt sistem pomeni, da lahko poljubno dodajamo nove aktivnosti in omejitve, obstoječim aktivnostim spreminjamo proizvodne parametre in omejitvam njihove lastnosti. Z orodjem lahko analiziramo različne tipe kmetijskih gospodarstev. V okviru prispevka predstavljamo tipično kmetijsko gospodarstvo, usmerjeno v prirejo mleka.

3.1 Primer – kmetijsko gospodarstvo, usmerjeno v prirejo mleka

Tipično kmetijsko gospodarstvo, na katerem predstavljamo delovanje elektronskega orodja, je bilo definirano na podlagi podatkov kmetijskega knjigovodstva (angl. Farm Accountancy Data Network – FADN). Ocena stroškov in dohodkov pa je bila izdelana s pomočjo modelnih kalkulacij (Rednak in sod., 2009).

Ključni namen analize je poiskati optimalen proizvodni načrt za obdobje enega leta. Podlaga za izračun so pričakovana pokritja (angl. Expected Gross Margin – EGM) po posameznih aktivnostih. Ključni vir ocene tveganja spremenljivk pa je varianca, izražena kot standardni odklon.

Pri pripravi podatkov smo izhajali iz časovne serije podatkov za obdobje desetih let (1999–2008). Za posamezna leta smo upoštevali povprečne letne

vrednosti cen, stroškov in pridelkov, izračunanih na Kmetijskem inštitutu Slovenije (KIS, 2009). Iz teh smo za posamezne aktivnosti izračunali pokritja po posameznih letih. Pričakovana pokritja smo izračunali v odvisnosti od predpostavljene verjetnosti za nastop posameznih stanj. Nihanja cen, stroškov in pridelkov po letih so rezultat različnih okoliščin v opazovanem desetletnem obdobju. Na podlagi ekspertne ocene smo s pomočjo parnih primerjav po metodi AHP izračunali verjetnosti za nastop posameznih stanj.

Analizirano kmetijsko gospodarstvo je primer visokospecializirane mlečne kmetije, ki poleg mleka prodaja tudi teleta in odrasle plemenske telice. Trenutni stalež predstavlja 58 krav molznic in 18 plemenskih telic. Laktacijska mlečnost krav črno-bele pasme znaša 7.800 litrov. Kmetija leži v ravninskem območju in gospodari s skupno 49 ha obdelovalnih površin, od tega je slaba polovica površin travinje (22 ha). Na večini površin kosijo štirikrat. Na njivskih površinah pridelujejo koruzo za silažo, koruzo za zrnje ter krmne koševine.

V postopku načrtovanja se v rešitev lahko vključijo tudi druge aktivnosti. Ključno vodilo pri izbiri aktivnosti in zlasti izbiri tehnologije je, da v proizvodni načrt lahko vstopajo le aktivnosti s podobnimi lastnostmi, kot jih imajo že vključene aktivnosti. V opisanem primeru lahko kot nove vstopajo aktivnosti za pridelavo krmnih oz. tržnih poljščin. Njihova intenzivnost pa se naravna glede na dosežen pridelek pri koruzi. Seveda je pri vključevanju poljedelskih aktivnosti treba upoštevati tudi kolobar, katerega omejitve so postavljene glede na zakonske predpise in uredbe.

Za kratko obdobje načrtovanja smo predpostavili, da stalni stroški (amortizacija) ostanejo nespremenjeni. Tako ne predvidevamo nobenih investicijskih aktivnosti in predpostavljamo, da so zmogljivosti infrastrukture za načrtovano časovno obdobje primerne. Stalni stroški zajemajo le stroške amortizacije strojev, zgradb in opreme in znašajo v analiziranem primeru na letni ravni 16.900 evrov.

3.2 Rezultati primera in razprava

Na kratko predstavljamo ključne rezultate pričakovanih vrednosti in varianc za analizirano hipotetično kmetijsko gospodarstvo. Izpostavljamo pričakovani dohodek in standardno deviacijo (kot parameter tveganja). Z vidika upravljalca kmetijskega gospodarstva so sicer zanimivi tudi posamezni proizvodni načrti, jih pa v tem prispevku ne prikazujemo, saj so

³ Zapisi kod ter osnovna načela programiranja so v večjem delu povzeti po Seref, Ahuja in Winston (2007).

zelo obsežni in bi za razumevanje potrebovali dodatno razlago.

Slika 2 prikazuje najpomembnejše rezultate analiziranega kmetijskega gospodarstva. Izhodišče krivulje proizvodnih možnosti predstavlja rešitev linearnega programa, ki zanemara tveganje. Povsem pričakovano je, da se rešitev nahaja v zgornjem desnem delu grafikona (LP max – maksimiranje dohodka), saj gre za rešitev, pri kateri bi kmetijsko gospodarstvo doseglo najvišji pričakovani dohodek (angl. Expected Income – EI). Zato je samoumevno, da gre za rešitev z najvišjo stopnjo tveganja, saj je za višji pričakovani dohodek treba prevzeti tudi večje tveganje (v našem primeru izraženo kot SD). Če primerjamo dobljeno rešitev s trenutnim stanjem kmetijskega gospodarstva (LP-DO), lahko ugotovimo, da bi se s takšnim proizvodnim načrtom pričakovani dohodek izboljšal za dobrih 30 odstotkov. Pri tem pa bi se tveganje povečalo kar za dobrih 43 odstotkov. Nedvomno gre za izrazite spremembe, ki na konkretnem tipu kmetije zaradi neugodnega razmerja med pričakovanimi vrednostmi

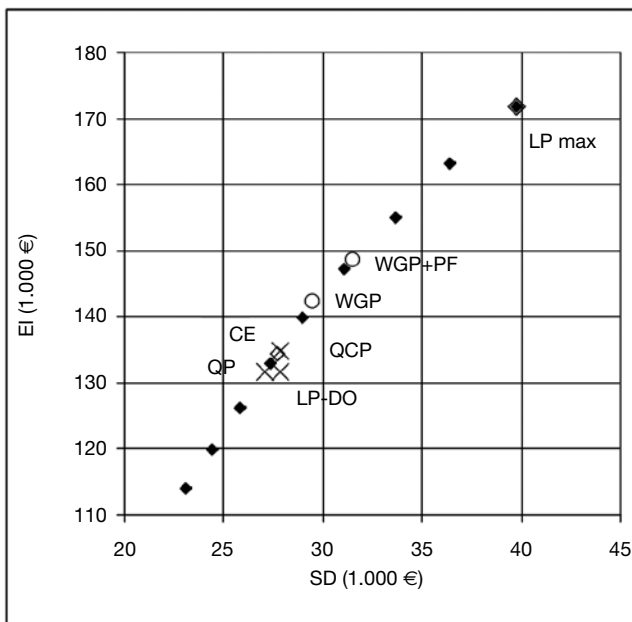
in povišano izpostavljenostjo tveganju niso sprejemljive. Razlogov za to je več, izpostavimo lahko predvsem vsesplošno konservativno naravo upravljavcev kmetijskih gospodarstev. Vsekakor pa je zanimiva tudi takšna »ekstremna« rešitev, saj predstavlja potencial in težišče možnih sprememb in izboljšav.

Eden izmed parametrov, ki ga podrobneje izpostavljamo, je tveganje in odnos do tveganja. Gre za pomembno komponento že izpostavljene konservativne narave kmetov. Da bi lahko posredno ocenili kmetov odnos do tveganja, smo uporabili neparametrični pristop (Lien, 2002). Kot je razvidno s slike 2, je trenutna proizvodnja analiziranega kmetijskega gospodarstva (LP-DO) neučinkovita tako z vidika pričakovanih vrednosti (EI) kot tudi z vidika stopnje tveganja. Točka se namreč nahaja pod krivuljo proizvodnih možnosti. Na sliki 2 sta prikazani točki QP in QCP, ki sta s teh vidikov učinkovitejši. Naklon med njima predstavlja odnos do tveganja, izražen kot absolutna mera tveganja (r_A).

Izkaže se, da je analizirano kmetijsko gospodarstvo relativno nenaklonjeno tveganju, saj r_A znaša 0,000156 oz. relativni koeficient (angl. Relative Risk Aversion – r_R) 15,64. Vse te točke tako ležijo na nasprotnem polu kot v primeru maksimiranja dohodka, dobljenega po linearnem programu, ki daje rešitve bližje izhodišču in s tem z nižjo stopnjo tveganja.

V celovit optimizacijski pristop smo kot parameter tveganja zajeli tudi maksimiranje zanesljivega ekvivalenta (angl. Certainty Equivalent – CE). Na analizirani kmetiji zaradi relativno visoke vrednosti izračunanega koeficienta nenaklonjenosti tveganju dosežemo sorazmerno nizko vrednost zanesljivega ekvivalenta (74.410 €). To je posledica visoke vrednosti premije tveganja (angl. Risk Premium – RP). Vendar primerjava rešitve z vidika pričakovanega dohodka pokaže, da na podlagi zanesljivega ekvivalenta (CE) dosežemo za 2 odstotka boljši rezultat (slika 2).

Glede na oba izpostavljena pola, ki vlečeta optimalno rešitev vsaksebi, je več kot očitno, da je za korektno obravnavo problema potreben večkriterijski pristop. Na primeru analiziranega kmetijskega gospodarstva prikazujemo rešitev, ki jo dobimo s pomočjo tehtanega ciljnega programiranja (WGP). Opozoriti je treba, da smo pri iskanju kompromisne rešitve upoštevali deset ciljev, med katerimi sta poleg ostalih tudi že omenjena skrajna pola. Prioritete posameznih ciljev so različne, določili so jih strokovnjaki na podlagi metode AHP. S slike 2 je razvidno, da se po kri-



Legenda: LP max – maksimiranje dohodka – rešitev linearnega programa; LP-DO – delna optimizacija, ki temelji na linearnem programiranju, QP – kvadratno programiranje (angl. Quadratic Programming), QCP – omejeno kvadratno programiranje (angl. Quadratic Constrained Programming), CE – zanesljivi ekvivalent (angl. Certainty Equivalent), WGP – tehtano ciljno programiranje (angl. Weighted Goal Programming), WGP+PF – tehtano ciljno programiranje, nadgrajeno s sistemom kazenskih funkcij (angl. Weighted Goal Programming supported by a system of Penalty Functions), EI – pričakovani dohodek (angl. Expected Income), SD – standardna deviacija.

Slika 2: Rešitve večkriterijske optimizacije za analizirano kmetijsko gospodarstvo v prostoru pričakovanih vrednosti in standardnega odklona

teriju pričakovanega dohodka dobljena rešitev v primerjavi s trenutnim stanjem izboljša za 4,2 odstotka (5.529 evrov). To je v primerjavi z zgoraj omenjenimi 30 odstotki v povsem razumljivih in dosegljivih mejah. Če odločevalcu omogočimo še nekoliko natančnejše nadziranje obnašanja modela – z nadgraditvijo modela s sistemom kazenskih funkcij (WGP+PF) –, pa se pričakovani dohodek izboljša za 4,6 odstotka. Spremeni se tudi standardni odklon (SD), saj se tveganje poveča – v kontekstu prostora pričakovanih vrednosti in standardnega odklona se premaknemo v desno (2.042 evrov). Seveda je to pričakovano, saj z dodatnimi kazenskimi funkcijami prisilimo model v bolj nadzorovano, s tem pa v manj racionalno (optimalno) vedenje. Tukaj ne prikazujemo podrobnejših rešitev. Povejmo le, da se zmanjša odstopanje po pričakovanem dohodku (EI) in da se poveča odstopanje po SD, kar vodi tudi do večjih odstopanj od ciljnih vrednosti zanesljivega ekvivalenta (CE).

Pri uporabi tehtanega ciljnega programiranja se je izkazal velik pomen kazenskih funkcij (PF), ki praviloma izboljšajo kakovost dobljene rešitve. Tudi za analizirano kmetijsko gospodarstvo bi jo priporočali kot najbolj *realno* rešitev. Ne gre za kakovost rešitve, ki bi jo ocenjevali le z vidika skupnega odstopanja, saj se le-to poveča. Sistem kazenskih funkcij omogoči, da model bolje ponazarja vedenje kmetijskega gospodarja, ki ni vedno racionalno. Razloge za to pa lahko iščemo v kompleksni naravi kmetijstva in tudi osebnostnih lastnostih gospodarja. Večjo stopnjo pozitivnosti dosežemo z natančnejšim kontroliranjem odstopanj od posameznih ciljev. Pri nadgradnji ciljev s sistemom kazenskih funkcij pa je treba biti racionalen, saj bistveno povečajo kompleksnost matematičnih modelov in s tem vplivajo tako na čas reševanja kot tudi na možnost, da z modelom ne najdemo rešitve.

4 SKLEP

Modeliranje, ki omogoča in zahteva podrobno poznavanje problema, vodi do boljših odločitev. Pri izbiri strukture modela je zelo pomembno, da preučimo, kakšen tip modela je najprimernejši, na katerih točkah je problem smiselno razdeliti na podprobleme in kako izbrati parametre in njihove povezave. Dejansko gre na eni strani za abstrakcijo problema, na drugi pa za poenostavitev realnega sveta na raven modelnega sveta. Ta korak je lahko problematičen, če ne pride do interdisciplinarnega sodelovanja strokovnjakov, saj se lahko zgodi, da bodisi uporabimo

napačen model (pomanjkanje sodelovanja matematikov), izpustimo ključne tehnološko-ekonomske zakonitosti (pomanjkanje sodelovanja strokovnjakov s področja problema) ali pa razvito modelno orodje ne zaživi (pomanjkanje sodelovanja informatikov).

Uporaba predstavljenih metod s področja operacijskih raziskav je ena izmed možnosti za podporo pri reševanju vsakodnevnih in tudi strateških odločitev. Njihova morebitna zanimivost je predvsem v njihovi relativni enostavnosti tako za modeliranje kot za razumevanje. V številnih raziskavah so se te metode izkazale kot dovolj zanesljive, zlasti če upoštevamo problem kvalitetnih podatkov in daljših časovnih serij podatkov. Velika prednost tovrstnih metod je tudi v tem, da jih lahko apliciramo v elektronskih preglednicah – po eni strani zaradi njihove dostopnosti širšemu krogu uporabnikov, po drugi strani pa zaradi uporabniku prijazne logike, na kateri temeljijo elektronske preglednice. Zato je z njihovim razvojem prišlo do pomembnega preobrata z vidika modeliranja, dostopnosti in uporabe modelnih orodij. Tako se je trend v zadnjih dvajsetih letih z razvojem elektronskih preglednic (npr. MS Excel) in dostopnostjo osebnih računalnikov povsem obrnil. Bistveno se je povečala tako uporaba modelov za analize kot tudi za podporo pri odločanju. Pričakujemo, da se bo v prihodnje njihova uporaba še povečala, in sicer tudi uporaba zahtevnejših elektronskih orodij.

Vsak model je neke vrste laboratorij, v katerem lahko eksperimentiramo in se učimo. Predstavljena problematika in elektronsko orodje sta le eden izmed primerov, kako lahko s pomočjo metod matematičnega programiranja podpremo odločanje in načrtovanje na ravni kmetijskih gospodarstev. Koristi takšnega modeliranja za podporo pri sprejemanju odločitev so številne: izognemo se številnim napakam, ne utrpimo posledic slabih idej, z njim lahko raziskujemo nemožne in tako pridemo do zanimivih alternativnih rešitev, intuicijo uporabimo za oblikovanje modela, ne pa za neposredno odločanje, končno pa z njimi lahko rešujemo povsem praktične probleme in pridemo do konkretnih rešitev. Te so uporabne predvsem pri vsakodnevem odločanju na kmetijskem gospodarstvu.

Za kmetijski prostor je v tem prispevku predstavljeno orodje pomembno, ker omogoča sistematično analizo dogajanj na določenih tipih kmetijskih gospodarstev. Prednost orodja je, da omogoča analizo v razmerah tveganja tudi pri slabši dostopnosti primarnih podatkov, s čimer se soočamo pri analizah

kmetijskih gospodarstev v Sloveniji. Kombiniranje metod matematičnega programiranja se izkaže za učinkovito, saj omogoča avtomatsko oceno manjka-jočih oz. agregiranih podatkov.

Med že razvitimi modelnimi orodji, ki temeljijo na v tem prispevku predstavljenih metodoloških izhodiščih, najdemo v slovenskem okolju tudi takšne, ki bi bili lahko uporabni za reševanje vsakodnevnih optimizacijskih problemov na ravni kmetijskih gospodarstev. Takšno je npr. orodje za načrtovanje in vodenje prehrane (Žgajnar, Juvančič in Kavčič, 2009; Žgajnar, Erjavec in Kavčič, 2010) ali pa orodje za načrtovanje gnojenja (Žgajnar in Kavčič, 2010). Hkrati pa lahko ugotovimo, da te modele redko uporabljajo končni uporabniki – kmetje ali kmetijske svetovalne službe. Razlogov je seveda več; izpostavimo lahko predvsem pomanjkljivo sodelovanje z informatiki, ki je ključno, da razviti modeli ne zaživijo. Če se osredinimo predvsem na aplikativne modele in orodja, pa lahko ugotovimo, da se poraja nova tržna niša e-storitev tudi pri orodjih za podporo odločanju v kmetijstvu. Tovrstne storitve so se na nekaterih drugih področjih že povsem uveljavile.

5 LITERATURA

- [1] Aimin, H. (2010). Uncertainty, Risk Aversion and Risk Management in Agriculture. *Agriculture and agricultural Science Procedia*, 1, 152–156.
- [2] Aouni, B., & Kettani, O. (2001). Goal programming model: A glorious history and a promising future. *European Journal of Operational Research*, 133, 225–231.
- [3] Azmi, R., & Tamiz, M. (2010). A review of Goal Programming for Portfolio Selection. In: D. Jones, J. Tamiz, & J. Ries (Eds.), *New Developments in Multiple Objective and Goal Programming*. Berlin: Springer.
- [4] Buongiorno, J., & Gilles, J. K. (2010). *Decision methods for forest resource management*, University of Wisconsin, Madison.
- [5] Buysse, J., Huylensbroeck, G. V., & Lauwers, L. (2007). Normative, positive and econometric mathematical programming as tools for incorporation of multifunctionality in agricultural policy modelling. *Agriculture, Ecosystems & Environment*, 120, 70–81.
- [6] Čančer, V. (2003). *Analiza odločanja: Izbrana poglavja*. Maribor: Ekonomsko-poslovna fakulteta Maribor.
- [7] Escalante, C. L., & Rejesus, R. M. (2008). Risk balancing decisions under constant absolute and relative risk aversion. *Review of Business Research*, 8, 50–61.
- [8] Flaten, O., & Lien, G. (2007). Stochastic Utility-Efficient Programming of Organic Dairy Farms. *European Journal of Operational Research*, 181, 1574–1583.
- [9] Gomez-Limon, J. A., Arriaza, M., & Riesgo, L. (2003). An MCDM analysis of agricultural risk aversion. *European Journal of Operational Research*, 151, 569–585.
- [10] Gonzalez-Pachon, J., & Romero, C. (1999). Distance-based consensus methods: a goal programming approach. *Omega*, 27, 341–347.
- [11] Hardaker, J. B., Huirne, R. B. M., Anderson, J. R., & Lien, G. (2007). *Coping with Risk in Agriculture* (2nd ed.). Oxfordshire: CABI Publishing.
- [12] Harwood, J., Heifner, R., Coble, K., Perry, J., & Somwaru, A. (1999). "Managing Risk in Farming: Concepts, Research, and Analysis". *Agricultural Economic Report No. 774*. Washington, DC: US Department of Agriculture, Economic Research Service Report.
- [13] Huirne, R., Meuwissen, M., & van Asseldonk, M. (2007). Importance of whole-farm risk management in agriculture. In: A. Weintraub, C. Romero, T. Bjordal, & R. Epstein (Eds.), *Handbook of Operations Research in Natural Resources* (pp. 3–15). New York: Springer Science & Business Media.
- [14] Ignizio, J. P., & Romero, C. (2003). Goal Programming. *Encyclopaedia of Information Systems*. Elsevier Science, 2, 489–500.
- [15] Jones, D., & Tamiz, M. (2010). *Practical Goal Programming*. International Series in Operations Research and Management Science. New York: Springer.
- [16] KIS. 2009. Modelne kalkulacije. (<http://www.kis.si/pls/kis/!kis.web?m=177&j=SI>).
- [17] Kobzar, O.A. (2006). *Whole-farm risk management in arable farming: portfolio methods for farm-specific business analysis and planning*. PhD thesis. Wageningen, Wageningen University.
- [18] Lara, P. (1993). Multiple objective fractional programming and livestock ration formulation: A case study for dairy cow diets in Spain. *Agricultural Systems*, 41, 321–334.
- [19] Lien, G. (2002). Non-parametric estimation of decision makers' risk aversion. *Agricultural Economics*, 27, 75–83.
- [20] Lien, G., Hardaker, J., van Asseldonk, M. A. P. M., & Richardson, J. W. (2009). Risk programming and sparse data: how to get more reliable results. *Agricultural Systems*, 101, 42–48.
- [21] Lien, G., & Hardaker, J. B. (2001). Whole-farm planning under uncertainty: impacts of subsidy scheme and utility function on portfolio choice in Norwegian agriculture. *European Review of Agricultural Economics*, 28, 17–36.
- [22] Liu, B. (2008). *Theory and Practice of Uncertain Programming* (3rd ed.). Beijing, China: Tsinghua University, Department of Mathematical Sciences.
- [23] Martel, J. M., & Aouni, B. (1990). Incorporating the Decision-maker's Preferences in the Goal-programming Model. *Journal of the Operational Research Society*, 41, 1121–1132.
- [24] Martel, J. M., & Aouni, B. (1998). Diverse Imprecise Goal Programming Model Formulations. *Journal of Global Optimization*, 12, 127–138.
- [25] Martins, M. B., & Marques, C. (2007). Methodological aspects of a mathematical programming model to evaluate soil tillage technologies in a risky environment. *European Journal of Operational Research*, 177, 556–571.
- [26] Pažek, K., Rozman, Č., Bavec, F., Borec, A., & Bavec, M. (2010). A Multi-Criteria Decision Analysis Framework Tool for the Selection of Farm Business Models on Organic Mountain Farms. *Journal of Sustainable Agriculture*, 34, 778–799.
- [27] Powell, S. G., & Baker, K. R. (2010). *Management science: the art of modeling with spreadsheets*. Hoboken (N. J.): John Wiley.
- [28] Rasmussen, S. (2004). *Optimizing Production under Uncertainty: Generalisation of the State-Contingent Approach and Comparison of Methods for Empirical Application*. Unit of Economics Working Papers 2004/2. Copenhagen, Denmark: The Royal Veterinary and Agricultural University.
- [29] Rednak, M. (1998). *Splošna izhodišča in metodologija izdelave modelnih kalkulacij za potrebe kmetijske politike*. Prikazi in informacije 189. Ljubljana: Kmetijski inštitut Slovenije.

- [30] Rednak, M., Erjavec, E., Volk, T., Zagorac, B., Mojk, B., Kavčič, S., Kožar, M., Turk, J., Rozman, Č., & Vučko, I. (2009). Analiza učinkov kmetijske politike z modelom tipičnih kmetijskih gospodarstev. Zaključno poročilo CRP projekta V4-0361. Ljubljana: Kmetijski inštitut Slovenije.
- [31] Rehman, T., & Romero, C. (1984). Multiple-criteria decision-making techniques and their role in livestock ration formulation. *Agricultural Systems*, 15, 23–49.
- [32] Rehman, T., & Romero, C. (1987). Goal Programming with penalty functions and livestock ration formulation. *Agricultural Systems*, 23, 117–132.
- [33] Romero, C. (2000). Risk programming for agricultural resource allocation. *Annals of Operations Research*, 94, 57–68.
- [34] Romero, C. (2004). A general structure of achievement function for a goal programming model. *European Journal of Operational Research*, 153, 675–686.
- [35] Romero, C., & Rehman, T. (2003). Multiple criteria analysis for agricultural decisions (2nd ed.). Amsterdam: Elsevier.
- [36] Saaty, T. L. (1980). *The analytic hierarchy process*. New York: Mc Graw.
- [37] Seref, M. M. H., Ahuja, R. K., & Winston, W. L. (2007). *Developing spreadsheet - Based Decision Support Systems: using Excel and VBA for Excel*. Belmont (Mass.): Dynamic Ideas.
- [38] Steuer, R. E., & Na, P. (2003). Multiple criteria decision making combined with finance: A categorized bibliographic study. *European Journal of Operational Research*, 150, 496–515.
- [39] Winston, W. L. (2002). *Operations Research: Applications and Algorithms*. Belmont: Duxbury Press.
- [40] Zekri, S., & Boughanmi, H. (2007). Modelling the interaction between agriculture and the environment. In: A. Weintraub, C. Romero, T. Bjorndal, & R. Epstein (Eds.), *Handbook of Operations Research in Natural Resources* (pp. 69–91). New York: Springer Science & Business Media.
- [41] Žgajnar, J., Erjavec, E., & Kavčič, S. (2010). Multi-step beef ration optimisation: application of linear and weighted goal programming with a penalty function. *Agricultural and Food Science*, 19, 193–206.
- [42] Žgajnar, J., Juvančič, L., & Kavčič, S. (2009). Combination of linear and weighted goal programming with penalty function in optimization of a daily dairy cow ration. *Agricultural Economics – Czech*, 55, 492–500.
- [43] Žgajnar, J., & Kavčič, S. (2011). Indirect estimation of farm's risk aversion; mathematical programming approach. *Bulgarian Journal of Agricultural Science*, 17, 218–231.
- [44] Žgajnar, J. (2011). Večkriterijsko optimiranje odločitev na kmetijskih gospodarstvih v razmerah tveganja. Doktorska disertacija. Ljubljana, Univerza v Ljubljani.
- [45] Ziolkowska, J. (2009). Hierarchical network modelling and multicriteria analysis for agri-environmental measures in Poland. In: F. Schaft, & A. Balmann (Eds.), 3rd MACE Conference. Multi-level Processes of Integration and Disintegration. Proceedings of the Third Green Week Scientific Conference, (pp. 168–178). Berlin: Leibniz Institute of Agricultural Development in Central and Eastern Europe – IAMO.

Jaka Žgajnar je asistent in raziskovalec za področje politike in ekonomike upravljanja z naravnimi viri na Biotehniški fakulteti Univerze v Ljubljani. Ukvarja se predvsem z obratoslovnimi vprašanji v kmetijstvu. Njegovo znanstvenoraziskovalno delo je usmerjeno v uporabo optimizacijskih metod s področja operacijskih raziskav za analizo in podporo poslovnim odločitvam na kmetijskih gospodarstvih. Sodeloval je v več ciljnih raziskovalnih projektih. Je avtor oz. soavtor več izvirnih znanstvenih člankov in objavljenih znanstvenih prispevkov na konferencah. Za diplomsko delo je leta 2006 prejel univerzitetno Prešernovo nagrado. Je aktivni član Društva agrarnih ekonomistov Slovenije (DAES), Evropskega društva agrarnih ekonomistov (EAAE) in Mednarodnega združenja za kmetijsko obratoslovje (IFMA).

Emil Erjavec je redni profesor za agrarno politiko in ekonomiko na Biotehniški fakulteti Univerze v Ljubljani. Je predavatelj različnih predmetov s področja ekonomike, agrarne politike, evropskih politik in metod raziskovalnega dela na vseh treh stopnjah visokošolskega izobraževanja na Biotehniški fakulteti in nekaterih drugih fakultetah Univerze v Ljubljani (EF, FDV, VF) ter od leta 2000 tudi stalni predavatelj na dunajski univerzi BOKU. Raziskovalno deluje na področju sektorskega modeliranja in politične ekonomije evropske kmetijske politike in evropskih integracijskih procesov. Je vodja programske skupine Konkurenčnost agroživilstva, ki združuje slovenske agrarne ekonomiste, in nosilec številnih domačih raziskovalnih projektov ter nacionalni koordinator petih okvirnih projektov EU. Predvsem na področju evropskih politik je vabljeni predavatelj številnih mednarodnih konferenc in svetovalnih projektov na evropski ravni. Od leta 2009 je predsednik Društva agrarnih ekonomistov Slovenije in član izvršnega odbora Evropskega združenja agrarnih ekonomistov (EAAE). Za svoje delo je dobil Jesenkovo priznanje in častni doktorat Univerze Cirila in Metoda v Skopju.

Stane Kavčič je izredni profesor za področje politike in ekonomike upravljanja z naravnimi viri na Biotehniški fakulteti Univerze v Ljubljani. Na raziskovalnem področju se ukvarja predvsem s sektorskim modeliranjem in z učinki kmetijske politike na obratoslovne vidike v kmetijstvu. Zanimajo ga predvsem praktični vidiki različnih ukrepov. Ureja konferenčni zbornik Društva agrarnih ekonomistov Slovenije, v zadnjih letih pa je bil vodja več ciljnih raziskovalnih projektov.

Lidija Zadnik Stirn se kot profesorica za operacijske raziskave na Biotehniški fakulteti Univerze v Ljubljani ukvarja z metodami optimiranja, ki jih uporablja za reševanje problemov v ekologiji, ekonomiki naravnih virov, gozdarstvu, lesarstvu, agronomiji in prehrani. Sodelovala je v raznih interdisciplinarnih raziskavah in bila vodja temeljnih in aplikativnih projektov, v letih 2001 do 2006 je bila koordinatorka EU projekta MEDMONT. Aktivno je sodelovala z referati ali vabljenimi predavanji na mednarodnih kongresih s področja operacijskih raziskav in bila organizatorka mednarodnih konferenc. Za te je leta 2003 prejela priznanje SDI. Je sourednica zbornikov, monografij in revij s področja operacijskih raziskav. Aktivno dela v mednarodnih društvih IFORS, EURO, IFIP TC7 in IUFRO 4.05. Od leta 1997 je predsednica slovenske sekcije OR, od leta 2002 pa podpredsednica SDI.

▣ Migracija obstoječih aplikacij na platforme za računalništvo v oblaku

¹Robert Dukarić, ²Matjaž B. Jurič

¹XLAB, d. o. o., robert.dukaric@cloud.si

²Fakulteta za računalništvo in informatiko Univerze v Ljubljani, matjaz.juric@fri.uni-lj.si

Izvleček

Danes je vedno več organizacij in podjetij, ki selijo svoje aplikacije v oblak z namenom, da bi modernizirali svojo informacijsko tehnologijo, znižali stroške ali da bi se pripravili na potrebe, ki bi jih organizacija morebiti imela v prihodnje. Migracija obstoječih aplikacij v oblak omogoča tako malim, srednjim kot tudi velikim podjetjem, da lansirajo nove produkte ali storitve na trg hitreje z minimiziranjem časa za namestitev konkretne informacijske tehnologije, kot so strežniki in druge omrežne komponente. Pri tem podjetja in organizacije eliminirajo investicijske stroške, zmanjšajo operativne stroške vzdrževanja in upravljanja infrastrukture ter si hkrati zagotovijo poslovno agilnost. V prispevku bomo predstavili osnovne tipe migracije v oblak, identificirali ključne prednosti in slabosti migracije ter jo analizirali na dveh različnih produktih računalništva v oblaku – Amazon AWS ter Windows Azure. Na koncu bomo predstavili nov, uniformen migracijski pristop, ki ga je mogoče aplicirati pri migraciji večine obstoječih informacijskih rešitev in je neodvisen od ponudnika računalništva v oblaku.

Ključne besede: računalništvo v oblaku, platforma kot storitev, infrastruktura kot storitev.

Abstract

Migrating Existing Applications to the Cloud

Several organizations and companies are nowadays moving their applications to the cloud in order to modernize their current IT assets, reduce their costs or to prepare for the requirements that an organization may have in the future. Migration of existing Word 2003.lnk applications to the cloud allows small, medium and large enterprises to bring new products or services to market faster by minimizing the time to install specific IT assets such as servers and other networking components, while substantially eliminating the investment costs and assuring business agility. In this article we present basic types of application migration, identify key advantages and disadvantages of migration and analyse it on two diverse cloud computing products – Amazon AWS and Windows Azure. Finally, we propose a novel, uniform migration approach that can be applied on several existing information systems migrations and is provider independent.

Keywords: Cloud computing, Platform as a Service, Infrastructure as a Service.

1 UVOD

Migracija aplikacije je proces ponovne postavitve aplikacije, tipično na novo platformo ali infrastrukturo. V primeru oblaka je lahko aplikacija prenesena iz obstoječega podatkovnega centra v poljuben ciljni oblak. Ciljno infrastrukturo lahko predstavlja javni, zasebni ali hibridni oblak. Zadnji transparentno kombinira zmožnosti prvih dveh. Aplikacije lahko ravno tako migriramo na različne tipe oblaka, kot sta infrastruktura kot storitev (IaaS) ter platforma kot storitev (PaaS). Pri identifikaciji aplikacij, ki jih želimo prenesti v oblak, je najprej treba preučiti in razumeti poslovne in tehnične faktorje migracije. Zmanjšanje stroškov in poslovna agilnost sta dva tipična poslovna faktorja za migracijo obstoječih aplikacij v oblak. Računalništvo v oblaku lahko zaradi povečane utilizacije ponudi bistvene prihranke. Povečana utilizacija je

posledica večnajemniškega modela (Multitenancy) in standardizacije ter avtomatizacije oblačnih storitev. Iz operativnega vidika so upravljanje, zmogljivost ter skalabilnost tipični vzroki, zakaj podjetja razmišljajo o prisvojitvi oblaka.

V nadaljevanju bomo opisali glavne tipe migracije v oblak, predstavili ključne prednosti in slabosti migracij ter analizirali dve možnosti prenosa obstoječih aplikacij v oblak. V prvem primeru bomo obravnavali migracijo klasične aplikacije na platformo Windows Azure, medtem ko bo drugi del opisoval prenos aplikacije v Amazonov oblak AWS. Obravnavana bosta dva različna migracijska pristopa: štirifazni migracijski pristop, ki ga predlaga Microsoft, ter Amazonov šestfazni migracijski pristop. V nadaljevanju bomo identificirali bistvene pomanjkljivosti

posameznega pristopa ter predstavili nov uniformen pristop, ki uspešno rešuje identificirane slabosti ter ga je mogoče aplicirati pri migraciji večine aplikacij in podatkov ter je neodvisen od ponudnika računalništva v oblaku.

2 MIGRACIJA APLIKACIJ V OBLAK

2.1 Tipi migracije v oblak

Računalništvo v oblaku lahko ponuja storitve infrastrukture, kot so npr. strežniki, omrežja ter shramba podatkov, kar z drugimi besedami imenujemo infrastruktura kot storitev (IaaS). Lahko ponuja aplikacijsko platformo z aplikacijskimi storitvami, kot je relacijska podatkovna baza, razvojno okolje itn. Takšno obliko imenujemo platforma kot storitev (PaaS). Nazadnje lahko ponuja aplikacije, ki so svojim uporabnikom zaračunane na mesečni ravni in so nameščene pri ponudniku oblaka, tj. programska oprema kot storitev (SaaS).

Danes številni ponudniki storitev omogočajo oskrbovanje, upravljanje in skaliranje storitev več končnim uporabnikom. Hkrati ponujajo zmožnosti, ki so zasnovane na ravni IaaS, pri čemer so številne organizacije za porabljene storitve zaračunane glede na poslovni model »plačaj toliko, kot uporabiš« (Pay-as-you-go). Na drugi strani številna podjetja in organizacije vzpostavljajo lastne zasebne oblake, pri katerih gre za infrastrukturne storitve informacijske tehnologije, ki jih upravlja organizacija sama in ravno tako podpira koncepte, kot so samopostrežba, poslovni model v obliki storitve, oskrbovanje na zahtevo ter občutek neskončnosti virov.

Ne glede na to, ali se organizacije odločijo za zasebni ali javni oblak kot njihov storitveni model, morajo najprej identificirati, katere aplikacije je smiselno prenesti v oblak in kako izvesti samo migracijo. Aplikacije so lahko torej prenesene iz obstoječih podatkovnih centrov v ciljni oblak, ki je lahko javni, zasebni ali hibridni. Ko je določena aplikacija identificirana kot kandidat za migracijo v oblak glede na poslovne in tehnične faktorje, je pomembno ugotoviti, za kateri tip računalništva v oblaku – PaaS ali IaaS – je aplikacija najbolj primerna (Cisco Systems Inc., 2010).

Platforma kot storitev

Platforma kot storitev lahko predstavlja možnost za migracijo poslovnih aplikacij, ki temeljijo na standard-

nih aplikacijskih strežnikov, kot so aplikacijski strežniki Java ali strežniki Microsoft SQL Server. Pri tem modelu ponudnik storitev upravlja aplikacijsko platformo in ponuja dostop do aplikacijskih storitev, kot je npr. podatkovna baza SQL. Aplikacijska platforma je lahko skupna več aplikacijam, pri čemer vsaka pripada drugi stranki. Kako je aplikacijska platforma povezana s fizično infrastrukturo, običajno nadzira ponudnik računalništva v oblaku. Odločitveni faktorji pri migraciji aplikacije so odvisni od tipa in verzije aplikacijskega strežnika, ki ga je naša aplikacija uporabljala pred tem. Nekatera okolja PaaS ne podpirajo vseh funkcionalnosti aplikacijskih strežnikov in lahko zahtevajo spremembe aplikacij. Merila, ki jih je treba upoštevati pri postavitvi PaaS, so (Cisco Systems Inc., 2010; Kaisler & Money, 2011):

- **dogovor o ravni storitve (SLA):** ponudnik storitev PaaS mora ponuditi SLA za razpoložljivost in zmožljivost aplikacijske platforme. Ponudnik mora ravno tako definirati jasno politiko in smernice za vzdrževanje in upravljanje verzij (Version Management) za platformo ter politike za kompatibilnost verzij API-jev med platformo in aplikacijo;
- **portabilnost podatkov:** podatki aplikacije so običajno shranjeni v podatkovni bazi, ki jo priskrbi ponudnik storitev oblaka. Uporabnik mora imeti možnost izvoziti podatke v format, ki bo omogočil migracijo k drugemu ponudniku. Enako velja za statične podatke, shranjene znotraj omrežnih datotečnih sistemov;
- **dolgoročni stroški:** finančni model ravni PaaS mora biti primerjan z modelom v internih namestitvah infrastrukture in aplikacijske platforme, ki uporablja model IaaS za postavitev aplikacijskih strežnikov na strežnike oblaka;
- **upravljanje uporabnikov:** aplikacija, postavljena na ravni PaaS, zahteva administrativne in uporabniške račune. Za oba tipa računov se mora uporabnik zavedati, kako je upravljanje z uporabniki usklajeno z obstoječimi imeniškimi storitvami (Directory Services) in procesi upravljanja z uporabniki (User Management Processes);
- **varnost:** v okolju PaaS lahko isti fizični strežnik izvaja aplikacije različnih uporabnikov. V takšnih večnajemniških okoljih (Multi-tenant Environments) so zahtevani dodatni varnostni ukrepi, ki zagotavljajo varno izolacijo posameznih aplikacij;
- **upravljanje s platformo:** aplikacijski strežniki ponujajo upravljaljske konzole (Management

Consoles) in orodja za spremljanje in upravljanje aplikacij, ki se izvajajo na njih. Okolje PaaS mora uporabnikom ponujati analogen nabor orodij za upravljanje in optimiziranje zmogljivosti aplikacij.

Infrastruktura kot storitev

Pri migraciji aplikacije na model IaaS je treba najprej ugotoviti, ali sta strežniška strojna oprema in operacijski sistem kompatibilna s trenutno strežniško strojno opremo in operacijskim sistemom. Če se aplikacija izvaja na strežnikih x86, mora ponudnik oblaka ravno tako implementirati inštrukcije x86. Če strojna oprema ni kompatibilna, je treba aplikacijo ponovno prevesti in postaviti na novo platformo. Če je operacijski sistem kompatibilen, bo pri migraciji aplikacije potrebnih le nekaj sprememb. Večino meril, ki smo jih upoštevali pri migraciji aplikacij v okolje PaaS, moramo upoštevati tudi na ravni IaaS (Cisco Systems Inc., 2010; Kaisler & Money, 2011):

- **dogovor na ravni storitve (SLA):** v okolju ravnih IaaS je treba zagotoviti SLA za razpoložljivost in zmogljivost delovanja strežnikov, omrežja in shrambe podatkov. Ravno tako morajo biti zagotovljeni dogovori o vzdrževanju in upravljanju infrastrukture in dogovori o času prekinitve (Downtime);
- **portabilnost podatkov:** aplikacija lahko uporablja strežnik podatkovne baze, ki se prav tako nahaja v oblaku. V tem primeru mora ponudnik storitev oblaka zagotoviti replikacijo in migracijo blokovne shrambe ali datotečnega sistema, ki se nahaja na strežnikih obstoječega podatkovnega centra;
- **dolgoročni stroški:** stroški aplikacije, postavljene na modelu IaaS, morajo biti primerjani s stroški postavitve v podatkovnih centrih organizacij. V nekaterih primerih ima javna namestitev modela IaaS očitne prednosti zaradi dinamičnega skaliranja in zaračunavanja glede na porabo. Za aplikacije v javnem oblaku so lahko dolgoročni stroški lastništva v nekaterih primerih tudi večji od stroškov lastništva v zasebnem oblaku;
- **upravljanje uporabnikov:** v okolju IaaS so zahtevane vsaj tri različne vloge uporabnikov – strežniški administrator, aplikacijski administrator in uporabnik aplikacije;
- **varnost:** virtualni stroji, ki pripadajo različnim strankam, so lahko implementirani na skupni fizični infrastrukturi. Preden aplikacijo prenesemo

v oblak, je treba naprej preveriti varnostne politike za virtualno in fizično izolacijo ponudnikove infrastrukture ter skladnost z zakonodajo;

- **skalabilnost:** aplikacije, ki so zasnovane za horizontalno skaliranje, so običajno večslojne (Multitiered) in imajo funkcijo za izenačevanje obremenitev, ki omogoča, da so lahko aplikacije brez stanj (Stateless Applications) skalirane navzgor ali navzdol. V primeru uporabe te funkcije mora ponudnik storitev oblaka ponuditi jasno politiko o tem, kako bo deloval ta tip skaliranja in kako bo poskrbljeno za dodeljevanje virov med več uporabniki in aplikacijami.

2.2 Prednosti in slabosti migracije v oblak

Računalništvo v oblaku ponuja organizacijam številne prednosti. Prva večja prednost migracije v oblak je, da organizacijam ni treba veliko investirati v fizično in programsko infrastrukturo, potrebna je le dobra internetna povezava. Ravno tako so odpravljeni operativni stroški za vzdrževanje in upravljanje infrastrukture. Zmožnost oblaka, da razširi svoje kapacitete samodejno glede na potrebe (elastičnost), je ključna prednost, ki diferencira računalniški oblak od drugih oblik gostovanja (Buyya et al., 2009). Tipično računalništvo v oblaku operira s poslovnim modelom »Pay-as-you-go«, kar nam omogoča, da dejansko najamemo toliko računalniških virov in storitev, kot jih potrebujemo. Naslednja prednost migracije je možnost dostopa do naših aplikacij in podatkov iz katerih koli naprav, ki imajo dostop do interneta. Oblak prav tako zagotavlja visoko stopnjo zanesljivosti in razpoložljivosti, saj omogoča replikacijo podatkov tudi med različnimi podatkovnimi centri in številne mehanizme za okrevanje po katastrofi (Disaster Recovery) (Kaisler & Money, 2011; Lee & Lee & Cheun, 2009). Ena izmed pomembnih prednosti prenašanja aplikacij in podatkov v oblak je agilnost, ki omogoča organizacijam večjo odzivnost na spremembe pri poslovanju in posledično hitrejšo lansiranje novih storitev na tržišče. Migracija aplikacij v oblak ravno tako zagotavlja neodvisnost od naprave in lokacije uporabnika ter s tem poveča dostopnost do aplikacij in podatkov (Smith, 2009).

Računalništvo v oblaku podpira izvajanje storitev in aplikacij, ki so že nameščene in dostopne v enem ali več podatkovnih centrih. Aplikacije, ki najbolj ustrezajo okolju računalništva v oblaku, so običajno tiste, ki strežejo veliki količini uporabnikov, kot so

e-poštne aplikacije, različni tipi splošnonamenskih spletnih aplikacij, socialna omrežja, sistemi ERP in CRM (Kaisler & Money, 2011). Ena izmed ključnih ovir pri migraciji aplikacij v oblak je zaklepanje znotraj ponudnika (Vendor Lock-In). To pomeni, da je podatke in aplikacije zelo težavno prenesti drugam, ko se enkrat odločimo za določenega ponudnika storitev računalništva v oblaku. Razlog je v tem, da različni ponudniki ponujajo lastne nestandardizirane formate podatkov in aplikacij (Borenstein & Blake, 2011). Naslednji oviri, ki močno vplivata na odločitev migracije informacijskih rešitev v oblak, sta varnost in zasebnost podatkov, ki povzročata podjetjem in organizacijam največ skrbi, še posebno ko je govor o prenosu poslovno kritičnih podatkov v roke zunanjemu ponudniku (Kandukuri, Paturi & Rakshit, 2011). Geografska lokacija je naslednja večja ovira. Če se podjetja odločijo gostovati svoje aplikacije v podatkovnih centrih na območju Združenih držav Amerike, bi v skladu s predpisi zakona U. S. Patriot Act imela vlada možnost vpogleda v njihove podatke. Ponudnik mora torej omogočiti svojim uporabnikom možnost izbire, v katerih podatkovnih centrih se bodo nahajale njegove aplikacije. Obstajajo še nekateri drugi pomisleki, ki so povezani z zagotavljanjem stabilne internetne povezave, visoke stopnje razpoložljivosti ter pomanjkanje robustnih dokumentov SLA (Smith, 2009; Kandukuri et al., 2011; Alhamed, 2010).

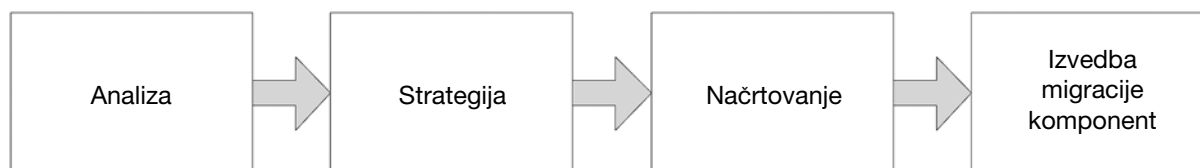
2.3 Migracija na Windows Azure

Zmogljivosti platforme Windows Azure, ki omogočajo razvijalcem uporabo različnih programerskih tehnik v razvojnih okoljih Visual Studio in Eclipse, so primarna prednost platforme Windows Azure, ki mora konkurirati drugim produktom računalništva v oblaku, kot so Amazonov AWS, Force.com ter Googlov App Engine. Upravljanje in vzdrževanje aplikacij lahko zahteva več časa in energije kot razvoj popolnoma nove programske opreme ali nadgradnja obstoječe. Ključne ovire migracije aplikaci-

je na Azure so premik spletne aplikacije, migracija podatkov v tabele Azure, blobe ali relacijsko bazo SQL Azure, premik projektov spletnih aplikacij v spletne in delovne vloge (Web/Worker roles), povezava s SQL Azure Database in prepričevanje vodstva informatike ter organizacijskega vodstva, da gostovanje aplikacij v Windows Azure ne pomeni bistvenih tveganj, poslovnih motenj oz. regularnih kršitev. Zadnja ovira je v začetni fazi migracije na platformo Windows Azure še največji problem, ki ga ni lahko premagati. Prvi korak pri razvoju aplikacij Azure je torej prepričati informatike ter organizacijsko vodstvo, da je oblak primerno okolje za gostovanje obstoječih ali novih projektov. Začetni ugovori in argumenti bodo verjetno povezani z zupanjanjem tretji organizaciji, ki bo ponujala razpoložljivost aplikacij, boljših od lokalno nameščenih v podjetjih oz. organizacijah. Vzdrževanje popolne zaupnosti dragocenih poslovnih informacij in njihova shramba sta največja skrb v vrhu vseh organizacij (Jennings, 2009).

Analiza, strategija ter načrtovanje migracije

Organizacije se sprašujejo, kako jim lahko zmožnosti in storitve računalništva v oblaku pomagajo oz. koristijo. Da bi to razumele, morajo najprej analizirati trenutno stanje svoje tehnološke infrastrukture in aplikacij ter oceniti, kje bi želele biti v prihodnje. Ko organizacije uspejo identificirati, katere oblačne prednosti – razširljivost, elastičnost, rapidna postavitve itn. – jim bodo najbolj ustrezale, želijo izbrati ocenjevalno ter migracijsko strategijo, ki bo najbolje ustrezala poslovnim potrebam. Ko so aplikacije identificirane kot del celotne strategije, je treba začeti razmišljati o njihovi arhitekturi. Aplikacije, ki ustrezajo določenim arhitekturnim konfiguracijam, bodo ustrezale tudi določenim migracijskim vzorcem. Iz tega razloga bo identifikacija aplikacij, ki ustrezajo določenemu migracijskemu vzorcu, olajšala evalvacijo ter načrtovanje uspešnih migracij (Cunningham, 2010). Ta postopek je prikazan na sliki 1.



Slika 1: **Postopek analize, izbire strategij ter načrtovanja migracije**

Da bi dosegli optimalne rezultate migracije, je treba najprej analizirati številne tehnične, poslovne ter finančne aspekte.

Izvedba migracije

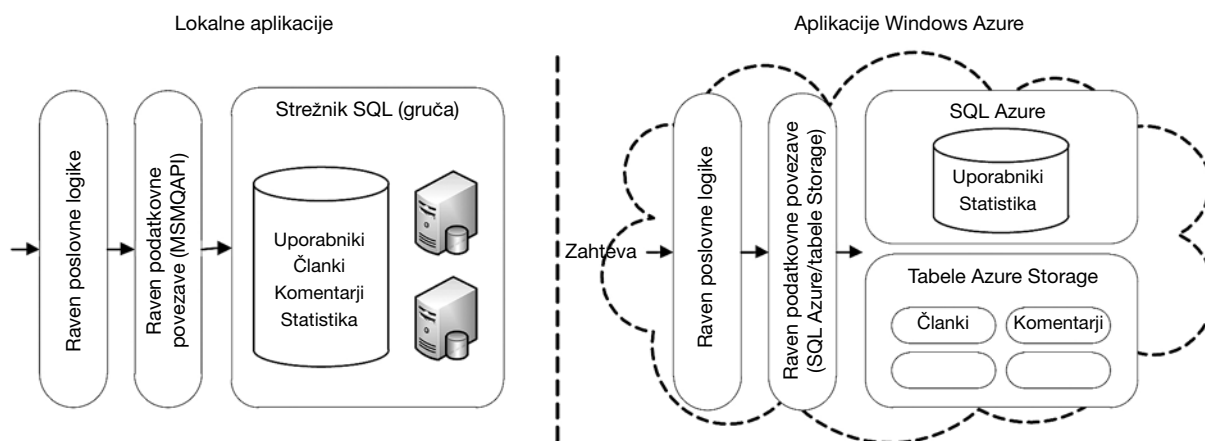
Medtem ko je vsak scenarij migracije Windows Azure različen, ta razdelek ponuja pristop ter tehnični pogled o tem, kako opraviti migracijo nekaterih podsistemov določene aplikacije. Za primer bomo vzeli poljubno lokalno aplikacijo z možnostjo rapidne razširljivosti, ki je gostovana v podatkovnih centrih s spletnimi strežniki v ospredju ter gručami podatkovne baze v ozadju. V nadaljevanju bomo prikazali sce-

narij migracije strežnika Microsoft SQL kot premik enega izmed podsistemov spletne aplikacije (slika 2).

Lokalna aplikacija, ki jo vzamemo za primer, uporablja centralizirano podatkovno bazo strežnika Microsoft SQL. V našem primeru je relacijska baza sestavljena iz dveh zelo obsežnih tabel. Za migracijo v oblak obstajata dve možnosti:

- premik velikih tabel v tabele Windows Azure,
- segmentacija podatkov teh dveh tabel prek več primerkov storitve SQL Azure.

Za nadaljnje delo smo se za shrambo podatkov odločili izbrati tabele Windows Azure, predvsem zaradi nekompleksne razširljivosti ter nižje cene uporabe.



Slika 2: Primer migracije podatkovne baze SQL Server v oblak

2.4 Migracija na Amazonov oblak AWS

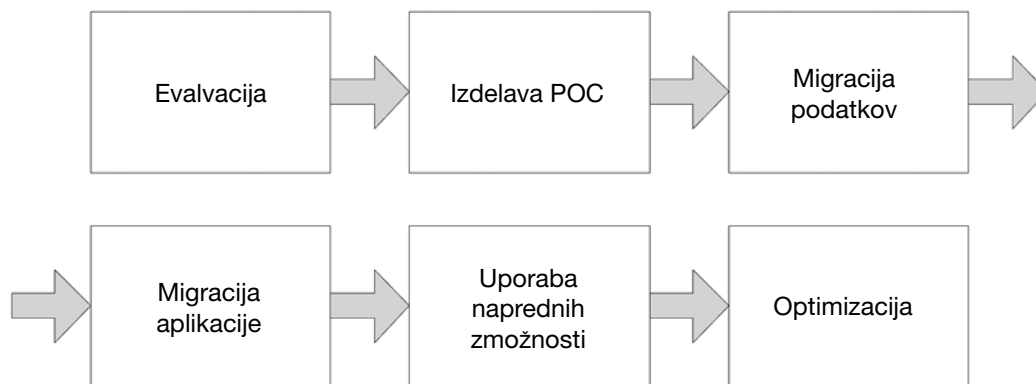
V tem poglavju bomo opisali korake in tehnike, ki so potrebni za migracijo obstoječih aplikacij v Amazonov oblak AWS, pri čemer bomo uporabili šestfazni pristop (slika 3). Ta pristop je sestavljen iz šestih faz:

- faze evalvacije,
- faze izdelave PoC (Proof of Concept),
- faze migracije podatkov,
- faze migracije aplikacije,
- faze uporabe naprednih zmožnosti ter
- faze optimizacije.

Razvijalci in arhitekti, ki imajo namen razviti novo aplikacijo za oblak, lahko preprosto oblikujejo komponente, procese in delovne tokove za njihovo aplikacijo, uporabijo API-je izbranega oblachnega ponudnika in sledijo dobrim praksam za načrtovanje, razvoj, testiranje ter postavitve aplikacij. Če se razvijalci odločijo postaviti svoje rešitve v oblachno infrastrukturo, kot je Amazon Web Services (AWS), lahko uporabi-

jo številne prednosti, kot so skalabilnost, elastičnost, izolacija procesov, zmanjšan operativni trud, oskrbovanje na zahtevo ter avtomatizacija (Varia, 2010).

Hkrati veliko podjetij išče načine, kako najbolje prenesti svoje obstoječe aplikacije v oblachno infrastrukturo, tako da lahko uporabijo prej naštete prednosti. Ključni diferenciator Amazonovih storitev pred platformo Windows Azure je njegova fleksibilnost. Podjetjem daje svobodo izbire programskih modelov in jezikov, operacijskih sistemov ter podatkovnih baz, s katerimi so že dodobra seznanjeni. Posledično danes veliko organizacij seli svoje aplikacije v oblak. Nekaterih aplikacij oz. informacijskih sredstev, postavljenih in nameščenih znotraj podatkovnih centrov organizacij, verjetno s poslovnega in tehničnega vidika ni smiselno premikati v oblak. Definitivno pa obstajajo informacijska sredstva oz. deli aplikacije znotraj organizacije, ki jih lahko brez večjega napora prenesemo v oblak in s tem veliko pridobimo.

Slika 3: **Večfazni pristop migracije obstoječih aplikacij v oblak**

Postavitev aplikacije v oblak AWS zniža infrastrukturne stroške, poveča poslovno agilnost in odstrani upravljanje z lastnim podatkovnim centrom. Uspešna migracija je odvisna od treh dejavnikov: kompleksnosti arhitekture aplikacije, stopnje šibke sklopljenosti aplikacije ter od truda, ki smo ga pripravljeno vložiti v samo migracijo (HyperStratus, 2009; Khajeh-Hosseini, Greenwood & Sommerville, 2010).

3 PRIMERJAVA MIGRACIJ NA AWS IN AZURE

Ko organizacije razmišljajo o migraciji aplikacij na Windows Azure ali v oblak AWS, je kritičnega pomena, da razumejo težave, ki jih želijo rešiti pri tem – tako s poslovnega kot tudi s tehnološkega vidika. Dobro načrtovana aplikacija z robustno arhitekturo lahko razširi svoje zmožnosti in bistveno zniža stroške, če jo prenesemo na eno izmed opisanih možnosti. Na drugi strani pa slabo načrtovane aplikacije niso nič bolj učinkovite v oblaku, kot če bi bile postavljene v klasičnem podatkovnem centru. Vsi že prej znani ključni elementi dobro načrtovane aplikacije veljajo tudi za postavitev v oblak, pri čemer pa so še toliko bolj pomembni.

Za razliko od Amazonovega šestfaznega migracijskega pristopa Microsoft predlaga selitev aplikacije v štirih fazah. Organizacije, ki želijo prenesti svoje obstoječe aplikacije na platformo Windows Azure, morajo najprej skozi fazo analize. V tej fazi je treba analizirati trenutno stanje tehnološke infrastrukture in aplikacij, da bi ocenili trenutno in hkrati želeno

prihodnje stanje. Prav tako je v tej fazi treba identificirati, katere oblačne prednosti – razširljivost, samopostrežba, agilnost, elastičnost, hitra postavitve, federacija itn. – bodo organizaciji najbolj ustrezale. V drugi fazi migracije bo organizacija morala izbrati ocenitev ter migracijsko strategijo, ki bo najbolj ustrezala njenim poslovnim potrebam. Ko so aplikacije identificirane kot del celotne strategije, je treba začeti razmišljati o njihovi arhitekturi. Identifikacijo aplikacij, ki ustrezajo določenemu migracijskemu vzorcu, izvedemo v fazi načrtovanja. S tem bistveno olajšamo ocenitev uspešno načrtovanih migracij. Ko smo enkrat izvedli vse tri faze predlaganega migracijskega pristopa, je treba izvesti ločeno migracijo vsakega izmed podsistemov aplikacije. Migracije, ki jih je treba realizirati, so migracija podatkovne baze, migracija omrežnega datotečnega sistema, migracija spletne aplikacije, migracije sporočilnih vrst, migracija procesnih vrst ter migracija distribuiranega predpomnilnika. Tabela 1 prikazuje primerjavo migracije posameznega podsistema aplikacije Amazon AWS in aplikacije, ki se izvaja na Windows Azure. Štirifazni pristop, ki ga predlaga Microsoft, sicer uspešno premakne aplikacijo v oblak, vendar ima nekatere pomanjkljivosti, in sicer daje premalo poudarka tehnični evalvaciji, evalvaciji stroškov, varnosti ter evalvaciji zakonodaje. Ravno tako ne predpisuje izgradnje pilotne aplikacije kot zelo pomembnega člena migracije ter ne vpeljuje koncepta optimizacije (tabela 2).

Tabela 1: Primerjava migracij logičnih komponent aplikacij

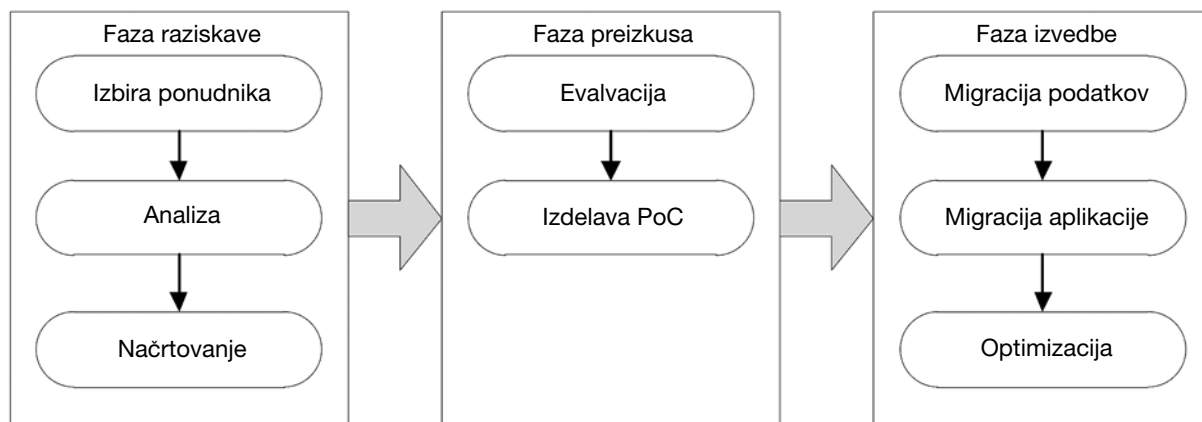
Komponente	Lokalno (On-premises)	Platforma Windows Azure	Amazon AWS
Spletna aplikacija	Internet Information Services (IIS), IBM Websphere, Apache HTTP Server, Tomcat itn.	Spletna ali delovna vloga (Web role/Worker role)	Elastic Compute Cloud (EC2)
Relacijska podatkovna baza	Oracle, DB2, SQL Server, MySQL ali kateri drugi SUPB	SQL Azure	Relational Database Service (RDS), EC2 + Elastic Block Store (EBS)
Nerelacijska podatkovna baza (NoSQL)	/	Storage Tables	SimpleDB
Sporočilne vrste	Java Message Service (JMS), Microsoft Message Queuing (MSMQ), JBoss Messaging itn.	Storage Queues	Simple Queue Service (SQS)
Distribuirani predpomnilnik	Memcached, Hazelcast, JBoss TreeCache itn.	AppFabric Caching	Elastic MapReduce
Statični podatki	Omrežni datotečni sistem (NFS)	Azure Blobs + Content Delivery Network (CDN)	Simple Storage Service (S3) + CloudFront

Večfazni pristop za migracijo obstoječih aplikacij v oblak AWS, ki ga razvijalcem priporoča Amazon, sestavlja šest soodvisnih faz. V prvi fazi, ki jo imenujemo faza evalvacije, je najprej treba izvesti ocenitev varnosti in tehnično evalvacijo. Pri ocenitvi varnosti je treba odgovoriti na vprašanja, kot so, kakšne so naše skrbi glede zaupnosti, integritete in dostopnosti podatkov, kakšne varnostna tveganja obstajajo ali kakšna je celotna toleranca tveganja. V primeru tehnične evalvacije nas zanima, katere aplikacije so bolj primerne za oblak z arhitekturne ter s strateške perspektive. Organizacije v tem primeru med drugim določijo, katere aplikacije premakniti v oblak najprej, katere kasneje in katere naj ostanejo v internem podatkovnem centru. Ko smo identificirali ustrezne kandidate za oblak in ocenili trud, ki je potreben za migracijo, je čas za izdelavo rešitve »Proof of Concept«. Cilj te faze je preučiti AWS in zagotoviti, da bodo točne naše predpostavke glede ustreznosti migracije v oblak. V tej fazi lahko razvijemo in postavimo testno aplikacijo v oblak in tako potipamo teren v oblaku AWS. To storimo tako, da razvijemo pilotno rešitev, ki pomeni fragment naše aplikacije in testira kritične funkcionalnosti aplikacije v okolju oblaka. V fazi migracije podatkov, ki je tretja faza, si morajo organizacije postaviti tale vprašanja: katere možnosti za shrambo podatkov obstajajo v oblaku, katere relacijske podatkovne baze (komercialne in odprtokodne) so na voljo v oblaku, kakšna je naša strategija, povezana s segmentacijo podatkov, ter koliko truda je treba vložiti za migracijo vseh podatkov v oblak. V fazi migracije aplikacije moramo odgovoriti na ključno vprašanje, kako premakniti ves poslovni sistem oz. njegov del v oblak, ne da bi pri tem oškodovali oz. onemogočili trenutno poslovanje. Po uspešnem prenosu aplikacije v oblak in opravljenih vseh po-

trebnih testih sledi faza uporabe oblaka. V tej fazi je treba ugotoviti, kakšne dodatne prednosti nam lahko prinese oblak, kaj je treba storiti za uspešno realizacijo elastičnosti in skalabilnosti ter kako lahko avtomatiziramo procese za lažje upravljanje in vzdrževanje aplikacij v oblaku. V zadnji, šesti fazi, ki jo imenujemo faza optimizacije, se je treba osrediniti na možnosti optimizacije oblačne aplikacije z razlogom, da bi še dodatno zmanjšali stroške. Ker smo zaračunani le za tiste storitve in vire, ki jih dejansko porabimo, je zelo zaželeno prizadevanje po optimizaciji. Amazonov šestfazni pristop sicer odpravlja vse pomanjkljivosti, ki smo jih v prejšnjem paragrafu očiteli pristopu Microsofta, vendar na drugi strani premalo poudarja analizo, strategijo in samo načrtovanje migracije (tabela 2). Slabost, ki je skupna obema pristopoma, je odsotnost pogajanj SLA in jasne definicije formalnih dogovorov o storitvenih pogojih med ponudnikom in uporabnikom računalništva v oblaku.

4 PREDLAGANI MIGRACIJSKI PRISTOP

Danes številne organizacije in podjetja želijo prenesti svojo informacijsko okolje oz. del tega okolja v oblak. Vsak izmed ponudnikov računalništva v oblaku predlaga svoje migracijske strategije, ki se razlikujejo od drugih. To pri vodilnih zaposlenih v organizacijah vzbuja zmedo in nezaupanje. Predlagani migracijski postopek (slika 3) je uniformen način, kako se lotiti migracije informacijskih rešitev in aplikacij v oblak, ter zagotavlja neodvisnost od ponudnika računalništva v oblaku. Ravno tako rešuje pomanjkljivosti komercialnih pristopov, ki smo jih identificirali v prejšnjem razdelku in prikazali v tabeli 2. Pristop, ki ga predlagamo, sestavljajo tri ključne faze: faza raziskave, faza preizkusa ter faza izvedbe. V nadaljevanju bomo vse tri predstavili bolj podrobno.

Slika 4: **Predlagani migracijski pristop**

Faza raziskave je sestavljena iz treh korakov:

- **izbira ponudnika:** v prvem koraku faze raziskave je najprej treba odločiti, na katero raven oblaka želimo seliti aplikacije (IaaS ali PaaS) ter kateri namestitveni model nam najbolj ustreza – javni, zasebni ali hibridni. Ko sta tip in namestitveni model določena, je treba izbrati ponudnika storitev računalništva v oblaku, ki bo ponudil informacijsko okolje za postavitve naše aplikacije. Pri izbiri je smiselno upoštevati dogovor SLA, da bi definirali zaupanje in kakovost storitev – QoS, ki jo različni ponudniki ponujajo svojim uporabnikom, kakor tudi dogovorjeno ogrodje (Agreed Framework) za stroške in ceno (Andrieux et al., 2004). SLA je zelo pomemben kot pogodba med ponudnikom in uporabnikom storitev. Ključna ideja SLA je dati jasno definicijo formalnih dogovorov o storitvenih pogojih (Service Terms), kot so zmogljivost, razpoložljivost in zaračunavanje. Zelo pomembno je, da SLA vključuje obveze in akcije, ki bodo izvršene v primerih kršitve z jasno definirano semantiko med posamezno entiteto, ki je vključena v elektronski pogodbi (Alhamed, Dillon & Chang, 2010);
- **analiza:** v drugem koraku je treba analizirati trenutno stanje tehnološke infrastrukture in aplikacij ter oceniti, kje bi želeli biti v prihodnje. Da bi dosegli optimalne rezultate migracije, je najprej treba analizirati številne tehnične, poslovne ter finančne aspekte. Identificirati je torej treba, katere prednosti oblaka – razširljivost, elastičnost, rapidna postavitve, agilnost, redukcija stroškov, visoka razpoložljivost, zanesljivost – bodo najbolj ustrezale organizacijam in podjetjem;
- **načrtovanje:** v zadnjem koraku faze raziskave je treba uskladiti poslovne potrebe z identificiranimi prednostmi oblaka in izbrati ocenjevalno ter migracijsko strategijo, ki bo najbolj ustrezala poslovnim potrebam. Ko so aplikacije identificirane kot del celotne strategije, je treba začeti razmišljati o njihovi arhitekturi. Aplikacije, ki ustrezajo določenim arhitekturnim konfiguracijam, bodo ustrezale tudi določenim migracijskim vzorcem. V tem koraku je torej ključnega pomena, da identificiramo aplikacije, ki ustrezajo določenemu migracijskemu vzorcu in s tem bistveno olajšamo ocenitev ter načrtovanje uspešnih migracij.

Druga faza (faza preizkusa) vsebuje dva koraka:

- **evalvacija:** v tem koraku je treba najprej izvesti oceno varnosti, stroškov ter zakonodaje in tehnično evalvacijo. Pri oceni varnosti je treba odgovoriti na vprašanja, kot so: kakšne so naše skrbi glede zaupnosti, integritete in dostopnosti podatkov, kakšna varnostna tveganja obstajajo ali kakšna je celotna toleranca tveganja. V primeru tehnične evalvacije nas zanima, katere aplikacije so bolj primerne za oblak z arhitekturne ter s strateške perspektive. Organizacije v tem primeru med drugim določijo, katere aplikacije premakniti v oblak najprej, katere kasneje in katere naj ostanejo v internem podatkovnem centru. Pri tem si lahko pomagamo z izdelavo drevesa odvisnosti (Dependency Tree) ter klasifikacijske tabele (Classification Chart) posameznih logičnih komponent aplikacije. Pri oceni stroškov je treba primerjati stroške, ki jih trenutno imamo v klasičnem podatkovnem centru, in stroške, ki bi jih imeli v primeru izvajanja aplikacije pri izbranem ponudniku. Pri oceni zakonodaje je tre-

ba identificirati politike posameznih ciljnih držav o shranjevanju podatkov ter lokalno zakonodajo, ki v večini primerov ne omogoča selitve zasebnih in poslovnih podatkov zunaj matične države;

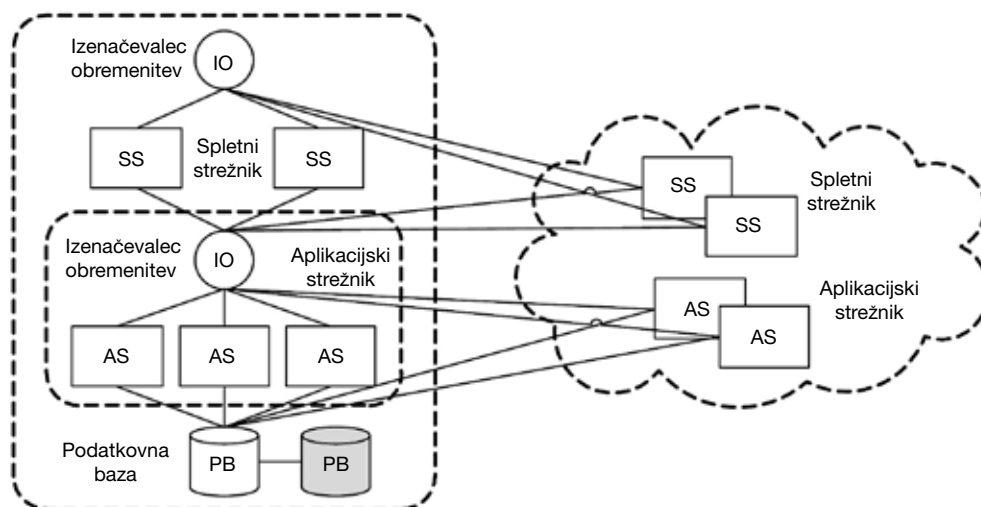
- **izdelava PoC:** v zadnjem koraku faze preizkusa je treba preučiti tehnologijo, ki jo ponuja izbrani ponudnik, in preveriti, ali so bile točne naše predpostavke glede ustreznosti migracije v oblak. V tej fazi razvijemo in postavimo testno aplikacijo in tako potipamo teren pri izbranem ponudniku storitev računalništva v oblaku. To storimo tako, da razvijemo pilotno rešitev, ki je jedro naše aplikacije in testira kritične funkcionalnosti v okolju oblaka. V tem koraku lahko izboljšamo podporo pri vodilnih členih organizacije, validiramo tehnologijo, testiramo »legacy« programsko opremo v oblaku ter definiramo glavna pričakovanja. V tem koraku je treba temeljito preučiti tehnologijo izbranega ponudnika ter ugotoviti, ali bomo s pomočjo pilota prepričali vodstvo organizacije in pridobili ustrezno podporo. Ravno tako je treba raziskati, koliko truda je treba vložiti v premik pilota v produkcijo, ter prepoznati, katere aplikacije so naslednji kandidati za premik v oblak.

Faza izvedbe, ki je zadolžena za dejansko migracijo, predpisuje tri korake:

- **migracija podatkov:** v prvem koraku faze izvedbe morajo organizacije najprej identificirati, katere možnosti za shrambo podatkov obstajajo v oblaku – od relacijske podatkovne baze in nerelacijske podatkovne baze vse do shranjevanja statičnih podatkov (tabela 1). V naslednjem koraku je treba

ugotoviti, katere podatkovne baze (komercialne in odprtokodne) so na voljo v oblaku ter katere so njihove prednosti in slabosti. V primeru relacijske podatkovne baze je treba uskladiti strategijo organizacije s procesom segmentacije podatkov ter oceniti, koliko truda je treba vložiti za migracijo vseh podatkov v oblak. V tem koraku je smiselno vključiti omrežje za dostavo vsebin (Content Delivery Network – CDN), da bi še dodatno izboljšali zmogljivost aplikacij predvsem z vidika končnega uporabnika;

- **migracija aplikacije:** v drugem koraku faze izvedbe moramo odgovoriti na ključno vprašanje, kako premakniti ves poslovni sistem v oblak, ne da bi pri tem oškodovali oz. onemogočili trenutno poslovanje. Pri tem sta na voljo dve možnosti: ali premaknemo celotno aplikacijo v oblak ali se lotimo migracije postopno. Glede na klasifikacijo tipov aplikacij iz prve faze se odločimo, katero možnost bomo aplicirali za kateri tip aplikacije. V primeru aplikacije brez stanja ali šibko sklopljene aplikacije je prva možnost ustrežnejša. V primeru kompleksnejših aplikacij, kot so npr. bančniški sistemi, spletne trgovine ter drugi poslovno orientirani informacijski sistemi, je primernejši postopen migracijski pristop, ki mu z drugimi besedami rečemo hibridni pristop (slika 5) in zagotavlja delovanja aplikacije v t. i. fazi koeksistence. Ko je aplikacija uspešno prenesena, je v tem koraku faze izvedbe treba prenesti še druge logične komponente aplikacije, kot so distribuirani predpomnilnik ter sporočilne vrste;



Slika 5: Hibridni pristop migracije aplikacij v oblak

- **optimizacija:** v tem koraku je smiselno identificirati dodatne prednosti oblaka, ki jih morebiti še nismo upoštevali. Ravno tako je treba narediti dodatne analize in ugotoviti, kako lahko še dodatno zmanjšamo stroške, da bomo zagotovili avtomatizirano elastičnost naših kapacitet v oblaku. Čeprav večina ponudnikov računalniškega oblaka poskrbi za varnost infrastrukture in platform z uporabo dobrih praks in tehnologij, je odgovor-

nost za varnost na aplikacijski ravni popolnoma prepuščena uporabnikom oblaka. V koraku optimizacije je tudi treba ugotoviti, kako lahko izboljšamo varnost ter zanesljivost naših aplikacij in podatkov. Določitev metrik za merjenje kritičnih zmogljivosti aplikacije ter zagotovitev uporabe administrativnih orodij za upravljanje in vzdrževanje aplikacij je prav tako smiselno obravnavati v tem koraku.

Tabela 2: Primerjava predlaganega pristopa s pristopoma komercialnih produktov

Predlagani pristop	Štirifazni migracijski pristop (Microsoft)	Šestfazni migracijski pristop (Amazon)
Faza raziskave		
• Izbira ponudnika	-	-
• Analiza	+	-
• Načrtovanje	+	-
Faza preizkusa		
• Evalvacija	-	+
• Izdelava PoC	-	+
Faza izvedbe		
• Migracija podatkov	+	+
• Migracija aplikacije	+	+
• Optimizacija	-	+

5 SKLEP

Migracija obstoječih aplikacij v oblak omogoča različnim tipom organizacij in podjetjem, da lansirajo nove produkte ali storitve na trg hitreje in z bistveno manj stroški. Prehod v oblak podjetjem omogoča tudi hitrejše vodenje analize tržišča ter omejevanje izgub v primeru hitrega padca, če npr. produkt ali storitev ne doseže zelenih pričakovanj. Danes je vse več organizacij, ki selijo svojo programsko opremo in podatke v oblak z razlogom, da bi modernizirali svoja trenutna informacijska sredstva, zmanjšali stroške, izrabili številne prednosti, ki jih prinaša oblak – elastičnost, zanesljivost, visoka razpoložljivost, ali da bi se pripravili na potrebe, ki bi jih morebiti imeli v prihodnje. Fazni pristop, ki smo ga predlagali v prispevku, omogoča identifikacijo idealnih projektov za migracijo ter povečuje zaupanje in podporo k migraciji znotraj posameznega podjetja.

V prispevku smo opisali osnovne tipe migracije v oblak, identificirali ključne prednosti in slabosti migracij ter predstavili dve možnosti prenosa obstoječih aplikacij v oblak. V prvem primeru smo obravnavali migracijo tradicionalne aplikacije na platformo Windows Azure, medtem ko je drugi del opisoval prenos

obstoječe aplikacije v Amazonov oblak AWS. Opisali smo dva različna migracijska pristopa: štirifazni migracijski pristop, ki ga predlaga Microsoft, ter Amazonov šestfazni migracijski pristop. V nadaljevanju smo identificirali bistvene pomanjkljivosti posameznega pristopa. Ugotovili smo, da štirifazni migracijski pristop sicer uspešno prenese aplikacijo v oblak, vendar ima nekatere pomanjkljivosti, kot so premajhen poudarek na tehnični evalvaciji, evalvaciji stroškov, evalvaciji varnosti ter zakonodaji. Ta pristop ravno tako ne predpisuje izgradnje pilotne aplikacije kot zelo pomembnega člena migracije ter ne vpeljuje koncepta optimizacije. Šestfazni pristop odpravlja vse pomanjkljivosti, ki smo jih očiteli pristopu Microsofta, vendar na drugi strani daje premalo poudarka na analizi, strategiji in načrtovanju migracije. Na koncu smo predstavili nov uniformni pristop, ki uspešno rešuje identificirane slabosti komercialnih pristopov in ga je mogoče aplicirati pri migraciji večine aplikacij in podatkov ter je neodvisen od ponudnika računalništva v oblaku. V prihodnjem delu bomo predlagani model še dodatno rafinirali z upoštevanjem novih pristopov ter ga z migracijami resničnih informacijskih sistemov validirali na številnih komercialnih in odprtokodnih produktih.

6 VIRI IN LITERATURA

- [1] Jennings, R. (2009). *Cloud Computing with the Windows Azure Platform*. Wiley Publishing, Inc., Indianapolis.
- [2] Cunningham, R. S. (2010). *Windows Azure – Application Profile Guidance: Custom Web (Rapid Scaling Focus) Application Migration Scenario*. Logic 20/20, Inc.
- [3] HyperStratus (2009). *Migrating Applications to the Cloud – An Amazon Web Services Case Study (A HyperStratus White Paper)*. Dosegljivo na www.hyperstratus.com.
- [4] Varia, J. (2010). *Migrating your Existing Applications to the AWS Cloud – A Phase-driven Approach to Cloud Migration (Amazon AWS White Paper)*.
- [5] Cisco Systems, Inc. (2010). *Planning the Migration of Enterprise Applications to the Cloud – A Guide to Your Migration Options: Private and Public Clouds, Application Evaluation Criteria, and Application Migration Best Practices*.
- [6] Armbrust, M., Fox, A. et al. (2010). *Above the Clouds: A Berkeley View of Cloud Computing*. Electrical Engineering and Computer Sciences, University of California at Berkeley.
- [7] Kaisler, S. & Money, W. H. (2011). *Service Migration in a Cloud Architecture*. System Sciences (HICSS), 2011 44th Hawaii International Conference on, pp. 1–10, 4–7.
- [8] Buyya, R. et al. (2009). *Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility*. Future Generation Computer Systems, Volume 25, Issue 6, Pages 599–616.
- [9] Smith, P. (2009). *Cloud Computing – A Strategy Guide for Board Level Executives*. Kynetix Technology Group.
- [10] Lee, J. Y., Lee, J. W., Cheun, D. W. & Kim, S. D. (2009). *A Quality Model for Evaluating Software-as-a-Service in Cloud Computing*, Software Engineering Research, Management and Applications, 2009. SERA '09. 7th ACIS International Conference on, pp. 261–266, 2–4.
- [11] Borenstein, N. & Blake, J. (2011). *Cloud Computing Standards: Where's the Beef?*. Internet Computing, IEEE, vol.15, no.3, pp.74–78.
- [12] Kandukuri, B. R., Paturi, V. R., & Rakshit, A.. (2009). *Cloud Security Issues*. Services Computing, 2009. SCC '09. IEEE International Conference on, pp. 517–520, 21–25.
- [13] Alhamed, M., Dillon, T. & Chang, E. (2010). *Conceptual SLA framework for cloud computing*, Digital Ecosystems and Technologies (DEST), 2010 4th IEEE International Conference on, pp. 606–610, 13–16.
- [14] Khajeh-Hosseini, A., Greenwood, D. & Sommerville, I. (2010). *Cloud Migration: A Case Study of Migrating an Enterprise IT System to IaaS*, Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on, pp. 450–457, 5–10.
- [15] Andrieux, A. et al (2004). *Web services agreement specification (WSAgreement)*.

■

Robert Dukarič je bil po končanem študiju računalništva in informatike na Fakulteti za elektrotehniko, računalništvo in informatiko Univerze v Mariboru zaposlen kot mladi raziskovalec iz gospodarstva v podjetju XLAB, d. o. o. Je tudi doktorski študent na Fakulteti za računalništvo in informatiko Univerze v Ljubljani. Raziskovalno se ukvarja s področjem računalništva v oblaku in sodeluje v številnih raziskovalnih in aplikativnih projektih za industrijo.

■

Matjaž B. Jurič je redni profesor na Fakulteti za računalništvo in informatiko Univerze v Ljubljani, kjer vodi laboratorij za integracijo informacijskih sistemov. Je avtor oz. soavtor štirinajstih knjig. Sodeloval je pri številnih projektih doma in v tujini, med drugim tudi pri razvoju RMI-IIOP, sestavnega dela platforme Java 2, in je član BPEL Advisory Boarda. Leta 2007 je od SDA World Journal dobil nagrado za najboljšo knjigo s področja SOA, leta 2010 pa nagrado za najodmevnejši znanstveni članek s področja storitev iz NMS. Ima naziva Java Champion in Oracle ACE Director.

■ Računalniška in internetna pismenost slovenske mladine

Viktorija Sulčič

Univerza na Primorskem, Fakulteta za management Koper, Cankarjeva 5, 6000 Koper
viktorija.sulcic@gmail.com

Izvleček

V prispevku s pomočjo statističnih podatkov primerjamo e-vključenost in e-pismenost slovenskih uporabnikov z državami EU 27 in EU 15. Na podlagi prikazanih podatkov lahko predpostavljamo, da se delovanje slovenske mladine vse bolj širi na internet, kar je pokazala tudi naša raziskava. Čeprav je mnenje slovenskih uporabnikov interneta glede e-spretnosti precej visoko, pa z raziskavo na vzorcu mladih med šestnajstim in štiriindvajsetim letom ugotavljamo, da informacijske in komunikacijske spretnosti, pridobljene v srednji šoli, niso zadostne za pridobitev spričevala ECDL. Po zbranih podatkih bi spričevalo ECDL iz posameznih modulov dobilo le 53,7 odstotka anketiranih dijakov in 63,3 odstotka anketiranih študentov. Ugotovitve slovensko šolstvo postavljajo pred velik izziv. Na področju komuniciranja smo ugotovili, da e-pošto in komunikacijo prek sistemov za neposredno sporočanje vedno bolj zamenjuje komuniciranje prek socialnega omrežja facebook.

Ključne besede: računalniška in internetna pismenost, e-pismenost, šolstvo.

Abstract

Computer and Internet Literacy of Slovenian Youth

In the article we compare e-inclusiveness and e-literacy of Slovenian users with EU 27 and EU 15 countries based on statistical data. The selected data pointed to the fact that users, especially young ones, are increasingly more present on the Internet, which was also confirmed by our research. Even though Slovenian Internet users highly value their e-skills, our research on a sample of youth between the ages of 16 and 24 showed that the ICT skills they acquire through high school are not sufficient for passing the ECDL certificate. According to the data we gathered, only 53.7% of high school pupils and 63.3% of higher education students would qualify for the ECDL certificate for specific modules. These findings represent a big challenge for the Slovenian educational system. When it comes to online communication, we also found out that e-mail and instant messaging communication is being increasingly replaced by communication over the social network Facebook.

Keywords: computer and internet literacy, e-literacy, education.

1 UVOD

Informacijska družba ali družba znanja zahteva ustrezno raven računalniške in internetne pismenosti posameznikov, saj v družbi znanja naj ne bi bilo ovir za prost dostop in prenos digitalnih informacij (Chaffey, 2007: 185). Čeprav o informacijski družbi govorimo že več kot desetletje (prav tam),¹ rezultati še vedno niso zadovoljivi. Izboljšanje digitalne pismenosti² je vključeno v evropsko digitalno agendo kot eno izmed sedmih področij strategije Evropa 2020 (EC, 2010), saj so pomanjkljivo digitalno pismeni posamezniki izključeni iz informacijske družbe in gospodarstva (prav tam). Digitalna kompetenca,³ ki zmanjšuje t. i. digitalni razkorak ali digitalno

ločnico, je ena izmed ključnih kompetenc za vseživljenjsko učenje (EC, 2007; e-inclusion, 2008: 4), saj se način dostopa do informacij in storitev naglo spreminja, zato se morajo posamezniki prilagoditi takšnim spremembam, če želijo ohraniti delovna mesta in enakopravno delovati v družbi. Čeprav se v praksi pogosto uporablja pojem digitalna pismenost, bomo v nadaljevanju ta pojem nadomestili s pojmom računalniška in internetna pismenost oz. e-spretnosti.

V digitalni agendi EU (EC, 2010) je do leta 2013 predvidena priprava enotnih kazalnikov za merjenje digitalne in medijske usposobljenosti. Eurostat že leta preverja doseganje ciljev informacijske družbe ter opravlja primerjave držav EU. Kazalniki »Information Society. Benchmarking Digital Europe 2011–2015« so razvrščeni v tri skupine (stebre) – ponudba informacijske in komunikacijske tehnologije, uporaba informacijske in komunikacijske tehnologi-

¹ Evropska komisija je prvo pobudo z namenom večjega zavedanja učinkov informacijske družbe in spodbujanja enakopravne vključitve v takšno družbo lansirala že leta 1998.

² Islovar omenja tudi pojma informacijska ali računalniška pismenost.

³ Islovar digitalno kompetenco opredeli kot »znanje, spretnosti, in motive, ki so potrebni za učinkovito izvršitev določene naloge ali rešitev problema z uporabo informacijske tehnologije«.

je in učinki uporabe informacijske in komunikacijske tehnologije. Ugotavljanje uporabe informacijske in komunikacijske tehnologije poteka na dveh skupinah uporabnikov – posamezniki oz. gospodinjstva ter podjetja. Med obema skupinama merijo način dostopa do interneta. Pri posameznikih oz. gospodinjstvih spremljajo pogostost uporabe informacijske in komunikacijske tehnologije, pri čemer za redno uporabo naprave oz. omrežja ali storitve upoštevajo vse, ki so le-te uporabili v zadnjih treh mesecih (Benchmarking Digital Europe, 2009: 7). Na vsaki dve leti spremljajo kazalnike o uporabi informacijske in komunikacijske tehnologije za osebno komunikacijo, zabavo, dostop do informacij, civilno in politično sodelovanje, ustvarjalnost, učenje, e-zdravje, urejanje osebnih financ, e-trgovanje, profesionalno delovanje, e-vključenosti in e-spretnosti (prav tam: 7–8). Med podjetji spremljajo integracijo poslovnih procesov, povezanost med dobavitelji in odjemalci, e-trgovanje, odnose s strankami in varnost transakcij (prav tam: 10–11). V testiranju je tudi kazalnik vezan na uporabo odprtokodnih rešitev.

E-spretnosti (angl. e-skills) so ena izmed najpomembnejših prioritet informacijske družbe (prav tam: 9). Največkrat e-spretnosti ugotavljajo z naborom nalog (opravil), ki jih je mogoče opraviti z računalnikom ali prek interneta. Primerjavo slovenskih uporabnikov s povprečjem EU prikazujemo v nadaljevanju (preglednica 3).

Pri delu s študenti na poslovni šoli ugotavljamo, da računalniška in internetna pismenost študentov ni skladna s pričakovanji, ki izhajajo iz prizadevanj na področju uvajanja informacijske in komunikacijske tehnologije v slovenskih šolah. Tem prizadevanjem je mogoče slediti že od leta 1994 s projektom RO (Računalniško opismenjevanje)⁴, sledita projekt E-gradiva,⁵ ki podpira razvoj e-vsebin z različnih predmetnih področij, in od leta 2008 še projekt E-šolstvo.⁶

Zaradi nezadovoljstva z izkazano računalniško in internetno pismenostjo študentov smo na začetku študijskega leta 2010/2011 naredili raziskavo med novinci poslovne šole. Rezultate ankete smo predstavili na Dnevih slovenske informatike (Sulčič, 2011). Zaradi zanimivih ugotovitev smo raziskavo razširili še na dijaško in študentsko populacijo. Za ugotavljanje sposobnosti uporabe računalniških rešitev (pro-

gramov) smo pripravili nabor opravil, ki jih je mogoče opraviti s posameznim programom, podobno kot to počne Eurostat z e-spretnostmi, z razliko, da je bil naš nabor opravil širši, predvsem pa povezan z vsebinami, ki naj bi jih poučevali v srednji šoli in se predvidevajo z nekaterimi moduli spričevala ECDL.⁷

Z raziskavo smo želeli ugotoviti e-spretnosti poučevane populacije ter morebitne razlike med njimi. Pri raziskavi smo postavili te trditve:

- H1: sposobnosti uporabe računalnika dijaške populacije se ne razlikujejo od sposobnosti študentske populacije, ugotovljenih z raziskavo, ki smo jo opravili pred tem med novinci poslovne šole (Sulčič, 2011);
- H2: v šoli pridobljene sposobnosti uporabe računalnika ne zadostujejo za pridobitev spričevala ECDL;
- H3: obstajajo statistično značilne razlike v računalniški in internetni pismenosti glede na spol;
- H4: obstajajo statistično značilne razlike v pismenosti glede na končano srednjo šolo in regijo;
- H5: komunikacija prek socialnega omrežja facebook nadomešča komunikacijo prek e-pošte ali sistemov za neposredno sporočanje.

Predpostavljamo, da mnenja anketirancev ustrezajo njihovim realnim sposobnostim, in lahko trditev, da znajo opraviti neko opravilo, razumemo kot njihovo e-spretnost.

Zbiranje podatkov je potekalo dvostopenjsko – oktobra smo zbrali podatke novincev visoke poslovne šole, maja 2011 pa smo k anketi prek učiteljev informatike v srednjih šolah v Kopru, Piranu, Kranju, Novem mestu in Mariboru povabili še dijake srednjih šol. K izpolnjevanju ankete smo prek informatikov povabili še študente samostojnega visokošolskega zavoda v Celju in študente treh drugih članic Univerze na Primorskem.

Podatke, zbrane z e-anketo, smo obdelali s programom SPSS. Poleg metod opisne statistike smo za ugotavljanje statistično značilne medsebojne povezanosti med spremenljivkami uporabili še korelacijsko in regresijsko analizo. Podatke prikazujemo razpredelnično in grafično. Za izdelavo grafičnih prikazov smo uporabili Calc iz zbirke NeoOffice.⁸

⁴ <http://ro.sio.si/programro.html>.

⁵ http://www.mss.gov.si/si/solstvo/ikt_v_solstvu/e_gradiva/.

⁶ http://www.sio.si/sio/projekti/e_solstvo.html.

⁷ Modul 3: Obdelava besedil, Modul 4: Preglednice, Modul 6: Predstavitev in del Modula 7 – Informacije in komunikacije (7.4 Elektronska pošta) – <http://www.ecdl.si/>.

⁸ OpenOffice.org za operacijski sistem Mac OS X.

2 E-VKLJUČENOST IN E-PISMENOST

2.1 E-vključenost

Eurostat (2011) spremlja podatke o *e-vključenosti* (digitalna vključenost) in jih primerja med posameznimi evropskimi državami. Primerjava temelji na deležu posameznikov, ki uporabljajo internet vsaj enkrat tedensko. Po teh podatkih smo bili Slovenci leta 2010 na ravni povprečja držav EU 27 (65 odstotkov posameznikov uporablja internet vsaj enkrat tedensko). Še leta 2009 smo zaostajali za povprečjem držav EU 27 za dve odstotni točki (58 odstotkov glede na 60 odstotkov). Seveda pa še vedno zaostajamo za povprečjem držav EU 15, v katerih je ta delež 69-odstoten.

Eurostat vodi statistiko o *e-spretnostih*⁹ posameznikov pri uporabi računalnika in interneta. Na vprašanje o e-spretnostih odgovarjajo anketiranci, ki so že

uporabljali računalnik in internet. Primerjavo pismenosti slovenskih uporabnikov glede na povprečje EU 15 in EU 27 na ravni uporabnikov interneta od šestnajstega do štiriinšestdesetega leta (navajamo pod »vsi«) in mladine med šestnajstim in štiriindvajsetim letom prikazujemo v preglednici 1. Kot je razvidno iz preglednice 1, ima mladina med šestnajstim in štiriindvajsetim letom veliko več e-spretnosti kot vsa populacija, ki uporablja računalnik in internet, tako na ravni povprečja EU 27 in EU 15 kot v Sloveniji. Če slovenski uporabniki računalnika in interneta zaostajajo za povprečjem evropskih držav, pa je slovenska mladina med šestnajstim in štiriindvajsetim letom pred vrstniki držav EU. Pri interpretaciji rezultatov moramo biti pazljivi na vsebino opravil, saj gre – razen pri programiranju in sestavi spletne strani – v bistvu za zelo enostavna opravila (gl. op. 9).

Preglednica 1: Primerjava e-spretnosti (%)

	EU 27		EU 15		SLO	
	Vsi	Stari 16–24 let	Vsi	Stari 16–24 let	Vsi	Stari 16–24 let
Na računalniku zna izvesti (2009) ^a	64	90	67	92	61	97
1–2 opravili	14	12	13	10	12	7
3–4 opravila	25	35	27	36	21	26
5–6 opravil	25	43	27	46	28	64
Na internetu zna izvesti (2010)	72	95	74	95	70	97
1–2 opravili	32	20	33	18	30	13
3–4 opravila	30	50	31	51	28	51
5–6 opravil	10	25	10	26	12	33

Opomba: a = zadnji dosegljivi podatki so le za leto 2009.

Eurostat (2011) v svojih anketah zbira tudi mnenje o tem, kaj uporabniki računalnika in interneta menijo o primernosti svojih e-spretnosti za zaposljivost – trenutno in v prihodnje. Kot je razvidno iz preglednice 2, je v Sloveniji večji delež uporabnikov računalnika in interneta, ki meni, da obstoječe e-spretnosti zagotavljajo zaposljivost (36 % v primerjavi s

33 % v državah EU 27 v starostni skupini 16–74 let oz. 63 % v primerjavi s 54 % v EU 27 med mladimi). Tudi delež tistih, ki menijo, da obstoječe e-spretnosti ne zagotavljajo zaposljivosti, je v Sloveniji v obeh skupinah anketirancev večji od povprečja EU 27 in EU 15 (preglednica 2).

Preglednica 2: E-spretnosti in zaposljivost (2007)

	E-spretnosti zagotavljajo zaposljivosti (%)		E-spretnosti ne zagotavljajo zaposljivosti (%)	
	Vsi	Stari 16–24 let	Vsi	Stari 16–24 let
EU 27	33	54	25	24
EU 15	34	56	24	23
Slovenija	36	63	28	27

Vir: Eurostat 2011.

⁹ V nabor *računalniških spretnosti* so vključena tale opravila: a) kopiranje ali premikanje datoteke ali mape, b) uporaba »kopiraj« in »prilepi« za podvajanje ali premikanje podatkov znotraj dokumenta, c) uporaba osnovnih aritmetičnih formul v programu za delo z razpredelnicami, č) stisnjenje datoteke (zip), d) namestitve novih naprav, e) izdelava računalniškega programa s poljubnim programskih jezikom. V nabor *internetnih spretnosti* pa so vključena opravila: a) uporaba iskalnikov za iskanje informacij, b) pošiljanje e-poštnih sporočil s priložnostjo (npr. dokumentom ali sliko), c) objava sporočil v klepetalnicah, forumih ali novinarskih skupinah, č) uporaba interneta za telefoniranje, d) uporaba P2P za izmenjavo datotek (glasbe filmov idr.), e) izdelava spletne strani. (e-inclusion, 2008: 5).

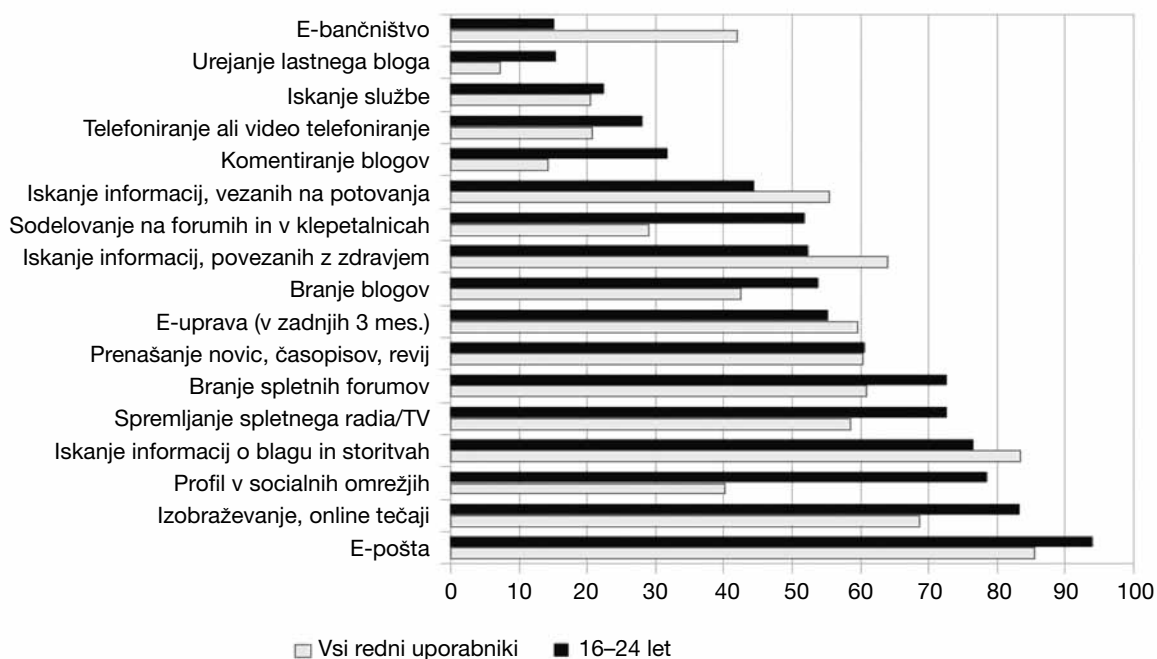
Če primerjamo podatke iz preglednic 1 in 2, lahko ugotovimo, da je npr. delež mladih, ki menijo, da njihove e-spretnosti zagotavljajo zaposljivost (63 %) večji od deleža tistih, ki obvladajo vsaj tri opravila na računalniku (47 %), oz. manjši od tistih, ki obvladajo vsaj tri opravila na internetu (70 %). Lahko bi sklepali, da so dejavnosti uporabnikov računalnika in interneta vse bolj usmerjene v delovanje na spletu.

2.2 Internet med slovensko mladino

V nadaljevanju predstavljamo podatke SURS (2011) o uporabi informacijske in komunikacijske tehnologije med mladino v starostni skupini 16 do 24 let. Medtem ko je v slovenski populaciji 70,0 odstotka rednih uporabnikov računalnika¹⁰ (67,9 odstotka uporabnikov interneta), jih je v starostni skupini 16 do 24 let 98,0 odstotka (97,5 odstotka uporabnikov interneta). Večina rednih uporabnikov (94,0 % v celotni populaciji, 98,1 % med mladimi) računalnik uporablja doma. Kot vidimo, skoraj vsi uporabniki računalnika uporabljajo tudi internet. Slovenski uporabniki so pogosti uporabniki interneta, saj 80,2 odstotka rednih uporabnikov celotne populacije in 91,6 odstotka v starostni skupini 16 do 24 let uporablja internet vsak dan ali skoraj vsak dan. In seveda internet uporabljajo v večini primerov tudi doma

(91,2 % v populaciji rednih uporabnikov in 95,2 % med mladino).

Redni uporabniki interneta uporabljajo internet za različne namene. E-pošta je najbolj razširjena storitev interneta, tako med vsemi slovenskimi uporabniki kot med proučevano skupino mladih med šestnajstim in štiriindvajsetim letom. Kot bomo videli v nadaljevanju, pa postaja komunikacija prek e-pošte, predvsem med mlajšimi, vse manj priljubljen način komuniciranja (preglednica 4). Iz spodnje slike je razvidno, da mlajša populacija od slovenskega povprečja odstopa v e-bančništvu (15,3 % v primerjavi s slovenskim povprečjem 42,0 %). Odstop v drugo smer pa je zaznati v sodelovanju na socialnih omrežjih, na katerih je mladina prisotna bolj (78,7 %) kot znaša slovensko povprečje (40,2 %). Mladina bolj intenzivno sodeluje v forumih in klepetalnicah (52,0 % na 29,0 %). Mladi tudi pogosteje komentirajo objave v spletnih dnevnikih (31,9 % glede na 14,3 %). Odstopanja med proučevanima skupinama je mogoče opaziti pri iskanju informacij za potrebe izobraževanja, vendar je tudi slovensko povprečje visoko – kar 68,7 odstotka rednih slovenskih uporabnikov interneta uporablja internet za izobraževanje. Zanimivo je, da je raven uporabe storitev e-uprave med mladino (55,4 %) skoraj enaka slovenskemu povprečju (59,5 %).



¹⁰ SURS za rednega uporabnika deklarira uporabnika, ki je napravo ali storitev uporabljal v zadnjih treh mesecih.

3 RAČUNALNIŠKA IN INTERNETNA PISMENOST DIJAKOV IN ŠTUDENTOV

Sposobnosti uporabe računalnika in interneta študentov zasledujemo že nekaj let, predvsem zaradi posodabljanja učnega načrta predmeta poslovna informatika, obveznega predmeta študijskega programa poslovne šole. Pred leti smo pri predmetu prenehali računalniško opismenjevati študente, saj smo na podlagi učnih načrtov srednjih šol predvidevali, da se dijaki računalniško opismenijo že v srednji šoli. Žal smo v praksi začeli opazovati, da imajo študenti težave pri uporabi računalnika za opravljanje vsakodnevnih študijskih obveznosti, kot sta npr. pisanje pisnih izdelkov ali iskanje za študij relevantnih informacij. Uporaba računalnika je vse prepogosto skrčena na socialna omrežja, predvsem na facebook.¹¹ Uporaba urejevalnika besedil in urejevalnika preglednic je zelo pomembna za študente poslovnih ved, pa najbrž tudi za študente drugih visokih šol. Zaradi tega smo pred tremi leti preverjanje spretnosti uporabe obeh programov vključili med študijske obveznosti študentov predmeta.

V študijskem letu 2010/2011 smo se odločili, da med novinci poslovne šole preverimo poznavanje uporabe standardnih računalniških rešitev. V ta namen smo pripravili seznam opravil, ki jih je mogoče izvesti v poljubnem urejevalniku besedil (28 opravil), urejevalniku preglednic (20 opravil), programu za pripravo elektronskih predstavitev (e-predstavitve) (10 opravil) in programu za odjem e-pošte (7 opravil). Pri vsakem opravilu smo predvideli tri mogoče odgovore – »Ne znam (1)«, »Sem že delal(a), vendar pozabil(a) (2)«, »Znam (3)«. E-vprašalnik¹² smo posredovali na 288 e-poštnih naslovov, ki so jih novinci ob vpisu navedli v vpisnem listu. Rezultate raziskave, ki smo jo izvedli oktobra 2010, smo predstavili na konferenci DSI 2011 (Sulčič, 2011). Maja smo anketo razširili še na dijake srednjih šol in študente nekaterih drugih visokih šol. Povabilo za izpolnjevanje ankete smo posredovali na srednje šole v Kopru, Piranu, Kranju, Novem mestu in v Mariboru ter na tri članice Univerze na Primorskem in na samostojni visokošolski zavod v Celju. Tako smo poleg podatkov 80 študentov poslovne šole v Kopru zbrali še 78 odgovorov drugih visokih šol in podatke 162 dijakov. V raziskavi je tako sodelovalo 320 anketirancev, ki so

bili k izpolnjevanju ankete povabljeni prek spletnih strani ali spletnih učilnic šole.

3.1 Predstavitev vzorca

V raziskavi je sodelovalo 158 študentov visokih šol (49,4 %) in 162 dijakov srednjih šol (50,6 %). Povprečna starost anketiranega dijaka je 16,6 leta. Povprečni študent je 6 let starejši (M = 22,6). V obeh skupinah anketirancev je sodelovalo več žensk – 58,0 % dijakin in 67,7 % študentk.

Med dijaki je na anketo odgovarjalo največ gimnazijcev – 58,0 % dijakov splošnih gimnazij in 12,4 % dijakov strokovnih gimnazij. Med študenti pa je delež študentov, ki so končali gimnazijo, precej manjši (31,6 %), saj je največ študentov svoje sekundarno izobraževanje končalo na štiriletni srednji strokovni šoli (45,0 %) ali na programu 3 + 2 (23,4 %) (preglednica 3).

Preglednica 3: Obiskovana srednja šola

	Dijaki (v %)	Študenti (v %)
Splošna gimnazija	58,0	24,7
Strokovna gimnazija	12,4	6,9
4-letna strokovna šola	12,3	45,0
Program 3 + 2	17,3	23,4

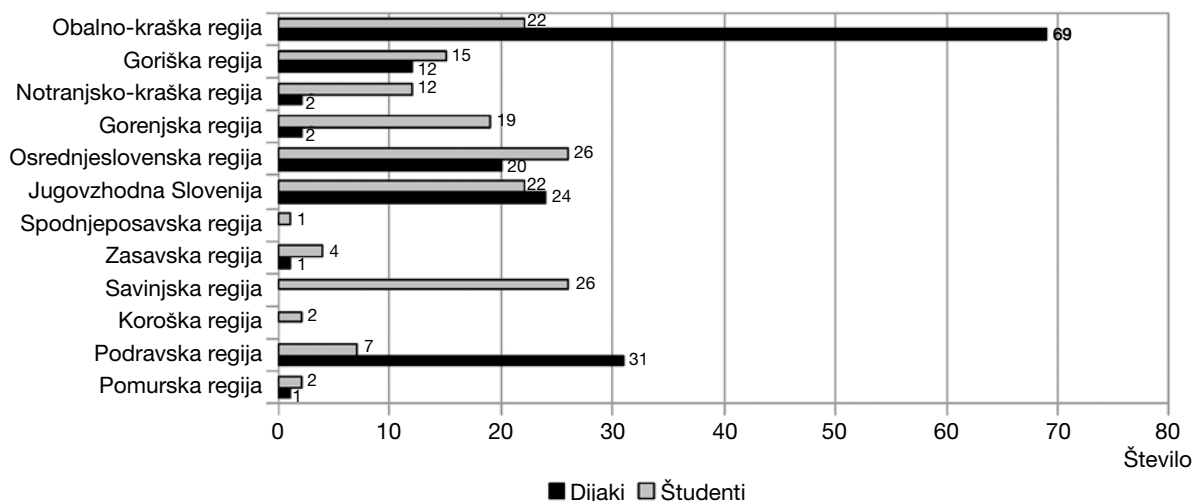
Vrsta končane srednje šole pri študentih je povezana tudi s študijskim programom, saj 71,5 % anketirancev študira na visokošolskih strokovnih študijskih programih, za vstop v katere se zahteva že opravljena poklicna matura. Za vstop v univerzitetne študijske programe morajo kandidati za vpis imeti opravljeno splošno maturo ali – v nekaterih programih – poleg poklicne mature še dodatni predmet iz splošne mature. Skoraj vsi študenti (92,4 %) so vpisani v redni študij.

Med anketiranimi študenti jih 81,6 % študira na družboslovnih študijskih programih, 10,1 % pa na področju tehnike.

Anketiranci prihajajo iz vseh dvanajstih statističnih regij. Med dijaki jih 42,6 % prihaja iz Obalno-kraške regije ter 19,1 % iz Podravske regije, nekoliko manj iz Jugovzhodne Slovenije (14,8 %) in Osrednjeslovenske regije (12,3 %). Pri študentih so regije bolj enakomerno zastopane – po 16,5 % jih prihaja iz Osrednjeslovenske in Savinjske regije ter po 13,9 % iz Obalno-kraške regije in iz Jugovzhodne Slovenije. Podrobnosti so razvidne iz slike na strani 152.

¹¹ <http://www.facebook.com/>.

¹² Za pripravo in distribucijo vprašalnika smo uporabili odprtokodno rešitev LimeSurvey – <http://www.limesurvey.org/>.



Za potrebe nadaljnjih statističnih obdelav smo razpršene podatke po regijah združili v tri skupine:

- prva skupina – *vzhod*: Pomurska, Podravska, Koroška in Savinjska regija – 21,6 % anketirancev,
- druga skupina – *osrednji del*: Osrednjeslovenska, Gorenjska, Zasavska, Spodnjeposavska regija in Jugovzhodna Slovenija – 37,2 % anketirancev,
- tretja skupina – *zahod*: Notranjsko-kraška, Goriška, Obalno-kraška regija – 41,3 % anketirancev.

3.2 Uporaba računalnika

V raziskavi smo proučevali obseg opravil, ki jih v posameznih standardiziranih računalniških rešitvah znajo izvesti anketiranci. V seznam opravil so vključena le osnovna opravila, kot npr. oblikovanje pisave, nastavitve robov, delo s slogi, izdelava kazal itd. V seznam opravil vključena opravila so vsebovana v učnih načrtih srednjih šol, saj je informatika neposredno ali posredno vključena v vse srednješolske programe. V gimnazijski program je informatika vključena kot obvezni (70 ur) in izbirni predmet (210 ur) (Wechtersbach idr., 2008). Dijaki lahko informatiko izberejo tudi kot izbirni predmet splošne mature (Maturitetni izpitni katalog, 2009: 60–61).

V srednjem poklicnem tehniškem izobraževanju je za predmet informatika predvidenih 68 ur (Najdič idr., 2007). Katalog znanja za informatiko za programe srednjega poklicnega tehniškega izobraževanja predvideva, da si dijaki v času srednješolskega izobraževanja pridobijo znanja, potrebna za pridobitev spričevala ECDL¹³. Predmeta informatika v prenov-

ljenem programu Ekonomski tehnik ne najdemo, je pa iz ciljev programa mogoče razbrati, da se dijaki »usposobijo za uporabo sodobne informacijske tehnologije« (prav tam). Vključevanje informacijske in komunikacijske pismenosti v programe srednjega poklicnega izobraževanja izhaja tudi iz koncepta vključevanja ključnih kompetenc (Najdič, 2009). Vsebine informacijske in komunikacijske tehnologije se pri ekonomskem tehniku vključujejo v modul Poslovni projekti (modul 1).

Poleg opravil smo anketirance spraševali po imenu programa, ki ga uporabljajo za opravljanje posamezne skupine opravil, npr. za urejanje besedil. Glede na monopolni položaj Microsoftove¹⁴ programske opreme v slovenskih šolah ne preseneča, da večina anketirancev (88,1 %) besedila ureja z wordom. Podobna situacija je tudi pri drugih skupinah opravil. Tako poleg worda uporabljajo še excel (84,1 %) in powerpoint (84,8%). Odprtokodne programe¹⁵ iz zbirke OpenOffice.org¹⁶ ali LibreOffice¹⁷ uporablja od 5,6 % (impress) do 8,4 % (writer). Calc uporablja 6,9 % anketirancev.

Zanimivo je, da odprtokodne rešitve pogosteje uporabljajo sedanji ali bivši dijaki splošnih gimnazij. Razlike so statistično značilne (Sig. < 0,05).

Kot smo omenili, so anketiranci poznavanje opravil ocenjevali na tristopenjski lestvici. Ocene po posameznih skupinah smo sešteli in razporedili v

¹⁴ <http://www.microsoft.com/>.

¹⁵ V odprtokodnih pisarnah (OpenOffice.org, LibreOffice ali NeoOffice) z writerjem pišemo besedila, s calcom urejamo preglednice, impress pa nam omogoča pripravo elektronskih prosojnic.

¹⁶ <http://www.openoffice.org/>.

¹⁷ <http://www.libreoffice.org/>.

¹³ Evropsko računalniško spričevalo (angl. ECDL – European Computer Driving Licence) – <http://www.ecdl.si/>.

pet enako velikih razredov. Za boljše razumevanje rezultatov analize podatkov predstavimo še nekaj dejstev.

Ker so pri urejevalniku besedil ocenjevali 28 opravil, in so opravilo, ki ga ne znajo opraviti, ocenili z oceno 1, je posamezen anketiranec za urejevalnik besedila zbral najmanj 28 točk. Takšen uporabnik ne zna opraviti nobenega opravila. Spretnost izvedbe opravila je prinesla tri točke, kar pomeni, da je anketiranec, ki zna v urejevalniku besedil opraviti vsa predlagana opravila, predvidena tudi z učnim načrtom srednjih šol, lahko zbral do 84 točk. Izračunov povprečja v tako izpeljanih razredih ne moremo obravnavati kot običajna povprečja, saj so npr. v razred tri uvrščeni anketiranci, ki so opravila znali opraviti in so to pozabili, oz. je delež opravil, ki jih znajo opraviti, precej nizek. Torej uvrstitev v razred tri ne odraža povprečno pismenega uporabnika, saj takšen uporabnik programa ne uporablja na način, kakor naj bi se uporabljal. Vse preveč uporabnikov namreč urejevalnik besedil uporablja kot pisalni stroj. Povprečno oceno štiri dosegajo anketiranci, ki npr. znajo opraviti polovico predvidenih opravil, po-

lovico opravil pa so znali opraviti, vendar so jih pozabili, oz. znajo opraviti dve tretjini opravil, tretjine opravil pa ne znajo izvesti.

Če našo raziskavo primerjamo z zahtevami spričevala ECDL,¹⁸ ko je za opravljen izpit treba zbrati najmanj 75 % točk,¹⁹ bi spričevalo ECDL pridobili le anketiranci, ki so uvrščeni v četrti ali peti razred. Če torej pogledamo podatke iz preglednice 4, lahko predpostavimo, da bi povprečni anketiranec (stolpec Skupaj v preglednici 4) pridobil spričevalo ECDL modula 3 (Obdelava besedil, brez 3.5 Serijsko tiskanje), modula 6 (predstavitev) in del modula 7 (7.4 Elektronska pošta). Če pa gledamo povprečne točke vseh opravil, potem povprečni dosežek dijaka 4,0 in študenta 4,1, upoštevajoč 75-odstotni uspeh za pridobitev spričevala ECDL iz obravnavanih modulov, je osip 46,3 odstotka dijakov in 36,7 odstotka študentov (deleže dijakov in študentov pod povprečno oceno 4,0). Vprašanje je seveda, kako bi se dijaki oz. študenti resnično odrezali na izpitu ECDL, saj naše ocene temeljijo le na subjektivni presoji anketirancev.

Povprečne ocene po posameznih skupinah programov prikazujemo v preglednici 4.

Preglednica 4: Sposobnost uporabe računalnika

Program	Dijaki		Študenti		Skupaj	
	M	SD	M	SD	M	SD
Urejevalnik besedil	4,3	0,8	4,3	0,8	4,3	0,8
Urejevalnik preglednic	3,5	1,2	3,8	1,2	3,7	1,2
Program za pripravo elektronskih prosojnic	4,3	1,2	4,4	1,1	4,4	1,1
Program za odjem e-pošte	4,1	1,0	4,1	1,0	4,1	1,0
Povprečje vseh opravil	4,0	0,8	4,1	0,8		
Splošna ocena sposobnosti uporabe računalnika	3,7	0,9	3,7	0,8	3,7	0,8

V anketi so anketiranci poleg ocen poznavanja posameznih opravil ocenili tudi svoje povprečne sposobnosti uporabe računalnika (preglednica 4). Ocenjevanje sposobnosti uporabe računalnika je potekalo na petstopenjski lestvici.²⁰ Med splošno oceno sposobnosti uporabe računalnika in razvrstitvijo poznavanja opravil v razrede obstajajo pozitivne statistično značilne povezave²¹ – z urejevalnikom besedil (0,50), z urejevalnikom preglednic (0,39), s progra-

mom za pripravo e-prosojnic (0,30) in s programi za odjem e-pošte (0,45).

Zanimivo je, da med srednješolsko in študentsko populacijo ni razlik – obe skupini anketirancev sta svoje sposobnosti uporabe računalnika ocenili s povprečno oceno (M) 3,7. Tudi standardni odklon (SD) je pri obeh skupinah enak (0,8).

Med obema vzorcema anketirancev ni razlik pri uporabi programa za odjem e-pošte (M = 4,1) in pri urejevalniku besedil (M = 4,3). Razlike pa so pri programu za urejanje preglednic, pri katerem so študenti nekoliko bolj večji uporabniki (M = 3,8) kot dijaki (M = 3,5). Je pa v tej skupini razpršenost podatkov kar precej velika (SD = 1,2).

¹⁸ <http://www.ecdl.si/>.

¹⁹ <http://www.ecdl.si/?gr1=kJec&gr2=kDsp>.

²⁰ 1 = slab uporabnik računalnika, 5 = odličen uporaben računalnika.

²¹ V oklepajih so navedene vrednosti statistično značilnih korelacijskih koeficientov.

Zanimivo je, da je uporaba programa za pripravo e-predstavitve v negativni statistično značilni povezavi s srednjo šolo, kar pomeni, da so svoje sposobnosti bolje ocenili anketirani dijaki gimnazij (-0,15). Statistično značilne razlike med šolami in spretnostmi uporabe računalnika smo opazili tudi pri urejevalniku preglednic (Sig. = 0,03), pri katerem najvišje ocene dosegajo dijaki strokovnih gimnazij in dijaki programov 3 + 2 (M pri obeh skupinah je 4,0), v primerjavi s splošnimi gimnazijci in dijaki štiriletnih strokovnih šol (M pri obojih je 3,5). Če podatke študentov in dijakov opazujemo ločeno, je statistično značilne razlike pri dijakih mogoče najti pri urejevalniku besedil (Sig. = 0,01), pri katerem se najvišje uvrščajo dijaki štiriletnih strokovnih šol (M = 4,8), nato splošni gimnazijci (M = 4,3), dijaki strokovnih gimnazij (M = 4,2) in na koncu dijaki programov 3 + 2 (M = 4,0), ki so najmanj večji uporabe programa za e-predstavitve (M = 3,3).

V skupini anketiranih študentov so statistično značilne razlike pri uporabi programa za preglednice (Sig. = 0,00), kjer izstopajo dijaki programov 3 + 2 (M = 4,3). Ta podatek je zanimiv predvsem zato, ker med dijaško populacijo ta skupina uporabnikov zaostaja za dijaki strokovnih gimnazijcev (M = 3,5, pri študentih M = 4,0). Tako da trditev, da obstajajo razlike med srednjimi šolami, lahko potrdimo le delno, saj obstajajo razlike pri uporabi nekaterih programov, ne pa razlike na splošno.

V uvodu predstavljeno trditev H1, da so dijaki enako večji uporabniki računalnika kot študenti, lahko potrdimo, saj med obema skupinama anketirancev ni statistično značilnih razlik.

Potrdimo lahko tudi drugo trditev (H2), da v šoli pridobljene sposobnosti uporabe računalnika ne zadostujejo za pridobitev spričevala ECDL, saj bi spričevalo ECDL – vsaj s področja proučevanih spretnosti ob predpostavki, da subjektivno mnenje anketirancev vsaj približno odraža dejanske sposobnosti – pridobila le polovica anketiranih dijakov (53,7 %) in malo več anketiranih študentov (63,3 %).

Poskušali smo ugotoviti, ali so spretnosti uporabe posameznih programov povezane s kakšno drugo proučevano spremenljivko, predvsem s spolom in starostjo. Za vso skupino anketirancev nismo – tako kot v raziskavi med študenti poslovnih šol (Sulčič, 2011) – ugotovili nobene statistično značilne povezanosti med spolom in spretnostmi uporabe posameznih programov. Je pa zanimivo, da ko podatke opa-

zujemo ločeno za dijaško in študentsko populacijo, opazimo statistično značilne povezave s spolom pri uporabi urejevalnika besedil (- 0,19) in pri splošni oceni sposobnosti uporabe računalnika (- 0,27). Obe povezavi sicer nista močni, sta pa statistično značilni. Tako naše trditve (H3), da obstajajo statistično značilne razlike med sposobnostjo uporabe računalnika in interneta in spolom ne moremo ne potrditi in ne zavrniti.

Pogledali smo še razlike med regijami. Statistično značilne razlike med skupinami regij smo opazili pri urejevalniku besedil (Sig. = 0,02), pri čemer zahodni del (M = 4,2) statistično značilno zaostaja za vzhodnim (M = 4,4) in osrednjim delom Slovenije (M = 4,4). Statistično značilne razlike med regijami so se pokazale tudi pri uporabi programa za e-predstavitve (Sig. = 0,02), pri čemer osrednji del Slovenije (M = 4,2) zaostaja za zahodom (M = 4,3), predvsem pa za vzhodom (M = 4,7).

Na primerjavo med regijami smo pogledali še z ločitvijo anketirancev na srednješolsko in visokošolsko populacijo. Pri dijakih se statistično značilne razlike (Sig. = 0,00) pokažejo le pri uporabi programa za e-predstavitve, pri kateri osrednji del države (M = 3,7) zaostaja za zahodnim (M = 4,4) in vzhodnim (M = 4,8). Pri študentih pa statistično značilnih razlik ni opaziti (Sig. > 0,05). Na podlagi prikazanega trditev H4, da obstajajo razlike med regijami, lahko potrdimo le za spretnosti pri urejevalnikih besedil in pri programu za pripravo e-predstavitve, nikakor pa je ne moremo potrditi za urejevalnik preglednic in odjemalec e-pošte.

Računalniška in internetna pismenost naraščata z leti učenja informatike (0,22). Povezava je statistično značilna.

3.3 Način komuniciranja

Za informacijsko družbo in aktivno sodelovanje v njej je pomemben način komuniciranja. Anketirancem smo v presojo ponudili šest različnih načinov (poti) komuniciranja, katerih pogostost uporabe pri komunikaciji s prijatelji so ocenjevali na petstopenjski lestvici.²²

²² 1 = ne komunicirate, 2 = redko, 3 = včasih, 4 = pogosto, 5 = redno.

Preglednica 5: Način in pogostost komuniciranja

Način komuniciranja	Dijaki			Študenti			Skupaj		
	M	Ne komunicira (%)	Redno (%)	M	Ne komunicira (%)	Redno (%)	M	Ne komunicira (%)	Redno (%)
E-pošta	2,3	23,5	3,7	3,0	8,1	12,7	2,6	15,9	8,1
Objave na facebook zidu	3,3	16,0	25,3	3,0	20,9	17,1	3,1	18,4	21,3
Sporočilni sistem facebook	3,3	14,2	22,8	2,9	22,8	16,5	3,1	18,4	19,7
Programi za neposredno sporočanja (npr. MSN, Skype)	3,0	24,7	22,8	2,4	31,0	7,6	2,7	27,8	15,3
Mobilni telefon	4,4	1,9	63,0	4,7	0,6	75,3	4,5	1,3	69,1
Oseбно	4,6	0,6	72,2	4,6	1,3	69,0	4,6	0,9	70,6

Opomba: 1 = ne komunicira, 5 = komunicira redno.

Med študenti je najbolj priljubljeno komuniciranje prek mobilnega telefona ($M = 4,7$), takoj za tem pa osebni način komuniciranja ($M = 4,6$). Kar 75,3 odstotka anketirancev mobilni telefon redno uporablja za komunikacijo s prijatelji, le 0,6 odstotka pa tega načina komunikacije ne uporablja (preglednica 3). Osebna komunikacija je najbolj priljubljena med dijaki ($M = 4,6$). Pri elektronskih načinih komuniciranja socialno omrežje facebook prevzema vlogo programov za neposredno sporočanje (npr. MSN messenger ali skype), pri dijakih celo bolj kot pri študentih (preglednica 3). Razlika med dijaki in študenti je pri sporočilnem sistemu facebook statistično značilna ($\text{Sig.} = 0,01$).

Zanimivo je, da 1,3 odstotka anketiranih študentov s prijatelji ne komunicira osebno.

E-pošte dijaki ($M = 2,3$) ne uporabljajo za redno komunikacijo, pa čeprav smo pri pregledu statističnih podatkov (slika 1) videli, da je to najpogostejše uporabljena spletna storitev. Pri študentih je komunikacija prek e-pošte še prisotna ($M = 3,0$), vendar jo vse bolj zamenjuje socialno omrežje facebook. Razlika pri uporabi e-pošte med dijaki in študenti je statistično značilna ($\text{Sig.} = 0,00$).

Pri proučevanju načinov komuniciranja smo iskali medsebojne statistično značilne povezave med proučevanimi spremenljivkami.

Tako e-pošto pogosteje uporabljajo boljši uporabniki računalnika (korelacijski koeficient 0,12) in standardiziranih računalniških rešitev (0,23), starejši uporabniki (0,33), ženske (0,15) ter uporabniki, ki niso imeli spletnih učilnic ali pa teh niso pogosto uporabljali ($-0,13$). Vse povezave so statistično značilne.

Medtem ko se e-pošta še uporablja med študenti, pa smo zasledili statistično značilne povezave med komuniciranjem na facebooku in starostjo ter spo-

lom – mlajši anketiranci, predvsem ženske, pogosteje komunicirajo prek objav na facebook zidu in prek sporočilnega sistema facebook (povezava s starostjo oboje $-0,20$, povezava s spolom – oboje 0,15). Objave na facebook zidu so pogostejše pri anketirancih, ki obiskujejo (so obiskovali) srednje šole z interaktivnimi spletnimi stranmi (0,14), kar kaže na medsebojno povezanost spletnih storitev – uporaba ene spletne storitve spodbuja k uporabi drugih spletnih storitev. S tem lahko potrdimo tudi trditev H5, saj komunikacija prek socialnega omrežja facebook vedno bolj prevzema vlogo, ki sta jo včasih imela e-pošta ali sistemi za neposredno sporočanje.

Medtem ko ženske raje komunicirajo prek socialnega omrežja facebook, pa je komuniciranje prek sistemov za neposredno sporočanje bolj priljubljeno med moškimi ($-0,11$) in med starejšimi anketiranci ($-0,13$). Je pa zanimivo, da sisteme za neposredno sporočanje uporabljajo anketiranci, ki obiskujejo (so obiskovali) srednje šole z interaktivnimi spletnimi stranmi (0,14) in kjer spletne učilnice uporabljajo bolj intenzivno (0,16).

Če pri sposobnosti uporabe računalnika in interneta nismo ugotovili statistično značilnih razlik med spoloma, pa smo te ugotovili pri načinih komuniciranja – ženske pogosteje uporabljajo e-pošto in komunicirajo prek socialnega omrežja facebook, moški pa prek sistemov za neposredno sporočanje.

4 SKLEPNA RAZMIŠLJANJA

Slovenija je glede e-vključenosti leta 2010 v povprečju držav EU 27, pa čeprav je še leta 2009 zaostajala za njimi za dve odstotni točki. Seveda še vedno zaostajamo za povprečjem držav EU 15. Kljub primerljivi e-vključenosti slovenskih prebivalcev pa podatki Eurostata kažejo zaostajanje slovenskih upo-

rabnikov računalnika in interneta v starostni skupini šestnajst do štiriindvajset let v usposobljenosti uporabe računalnika in interneta, saj slovenski uporabniki v povprečju dosegajo komaj 1,0-odstotni delež uporabnikov, ki znajo opraviti eno ali več opravil na računalniku oz. 94,6-odstotni delež uporabnikov interneta, ki znajo opraviti enak obseg opravil na internetu. Se pa boljša usposobljenost kaže med mladimi v starosti šestnajst do štiriindvajset let, med katerimi znajo skoraj vsi (97 %) opraviti vsaj eno opravilo na računalniku in na internetu. Res pa je, da sam dostop do interneta in deklarativna uporaba računalnika in interneta še ne kaže dejanske sposobnosti uporabe računalnika in interneta, ki so pomembne za enakopravno sodelovanje v družbi znanja, čeprav je večji delež slovenskih uporabnikov interneta v primerjavi z deležem držav EU 27 in EU 15, ki meni, da imajo zadosten obseg e-spretnosti za ohranitev zaposlitve (36 % v skupini 16 do 74 let in 63 % v skupini med 16. in 24. letom).

Slovenska mladina (od 14. do 24. leta) pozitivno vpliva na statistične podatke o rabi računalnika in interneta v Sloveniji, saj so med mladimi redki posamezniki, ki še niso uporabljali računalnika in interneta. Čeprav SURS med redne uporabnike uvršča tiste, ki so napravo/omrežje/storitve uporabljali v zadnjih treh mesecih, je iz statističnih podatkov razvidno, da večina teh uporabnikov (več kot 80 oz. več kot 90 odstotkov med mladimi) internet uporablja tako rekoč vsak dan. Najpogosteje uporabljena storitev interneta – tako v celotni slovenski populaciji, kot med mladimi – je e-pošta, čeprav je naša raziskava, izvedena na vzorcu 320 dijakov in študentov, pokazala, da sta komunikacija prek e-pošte in komunikacija prek sistemov za neposredno sporočanje najmanj priljubljena načina komuniciranja. Mladina najpogosteje komunicira prek mobilnega telefona. Ta podatek je pomemben za razvijalce e-vsebin (tudi izobraževalnih), ki naj ponudijo čim več vsebin ravno prek mobilnega telefona.

Pri primerjavi statističnih podatkov med vso populacijo uporabnikov interneta in mladimi preseneča podatek, da več kot polovica mladine (55,4 %) redno uporablja storitve e-uprave, kar je skorajda na ravni vse slovenske populacije (59,5 %). Podatek kaže na pripravljenost mladine za e-poslovanje.

Raziskava med dijaki in študenti je pokazala, da kljub splošni prepričanosti o visoki ravni e-spretnosti (90 oz. 84 % mladih zna na računalniku oz. na in-

ternetu izvesti tri opravila in več), bi jih spričevalo ECDL po v raziskavi zbranih podatkih in zahtevah za pridobitev le-tega, dobilo le 53,7 odstotka dijakov in 63,6 odstotka študentov.

Najpogosteje uporabljen način komunikacije med mladimi je socialno omrežje facebook – tako objave na zidu kot izmenjava sporočil prek sporočilnega sistema. Socialno omrežje facebook je zanimivo predvsem za mlajše uporabnike, ki imajo v povprečju tudi najvišje število prijateljev.

Ker nekatere raziskave ugotavljajo, da so moški pogostejši uporabniki računalnikov in interneta ter da imajo večji interes za uporabo računalnika in interneta (Podovšovnik Axelsson, 2009: 193–194), smo povezanost uporabe računalnika in interneta s spolom študentov poskušali preveriti tudi v naši raziskavi. Na proučevanem vzorcu nismo zasledili statistično značilne povezave. Smo pa statistično značilne povezave s spolom zasledili pri načinih komuniciranja – ženske pogosteje komunicirajo prek e-pošte in prek objav na facebook zidu. Moški pa so pogostejši uporabniki sistemov za neposredno sporočanje.

Za konec bi izpostavili v slovenskem prostoru še neupravičeno velik delež Microsoftove programske opreme, predvsem MS Office. Slovenski uporabniki bi lahko vsa opravila, ki smo jih predvideli z raziskavo, izvedli v odprtokodni rešitvi OpenOffice.org ali LibreOffice. Učitelji bi s promocijo odprtokodnih rešitev vplivali na zmanjšanje deleža piratstva v državi, pa tudi na ozaveščanje odnosa do avtorskih pravic. Nesporno dejstvo je, da večina uporabnikov ne nabavi licenčne programske opreme. Zanimivo je, da se Slovenija s 46 odstotki piratstva²³ ne uvršča v seznam tridesetih držav z najnižjo stopnjo piratstva (države s 43 odstotki in manj nelegalno naložene programske opreme). So pa na ta seznam npr. uvrščene Češka (37 %), Madžarska (41 %), Španija (42 %) in Slovaška (43 %) (BSA, 2010: 7).

V prihodnje bi bilo zanimivo ponoviti raziskavo še na večjem vzorcu anketirancev, s čemer bi si ustvarili bolj reprezentativno sliko o e-pismenosti mladine, pa tudi starejše slovenske populacije. Tako bi lahko zasledovali dogajanje na področju e-pismenosti in sprejemali ukrepe za njeno izboljšanje in s tem bolj enakopravno vključitev v informacijsko družbo, družbo znanja. Ugotavljamo, da računalniške in internetne pismenosti ne moremo jemati kot samo po

²³ http://portal.bsa.org/globalpiracy2009/pr/pr_slovenia.pdf.

sebi razumljeno posledico uporabe računalnika in interneta, temveč moramo v razvoj e-spretnosti vložiti določen napor – tako prek vključevanja teh spretnosti v šolski kurikulum kot tudi v ustrezno usposobljenost učiteljev, ki te veščine poučujejo v šolah.

5 LITERATURA

- [1] —. (2009). *Maturitetni izpitni katalog za splošno maturo leta 2011*. Ljubljana: Državna komisija za splošno maturo.
- [2] BSA – Business Software Alliance. (2010). *Seventh Annual BSA/IDC Global Software Piracy Study*. Dostopno: <http://portal.bsa.org/globalpiracy2009/index.html> (24. 1. 2011).
- [3] Chaffey, D. (2007). *E-Business and E-Commerce Management: Strategy, Implementation and Practice*. 3rd ed. Harlow: Prentice Hall.
- [4] EC – European Commission. (2010). *Europe's Digital Agenda*. Dostopno: http://ec.europa.eu/information_society/digital-agenda/index_en.htm (18. 1. 2011).
- [5] EC – European Commission. 2007. *Key Competences for Lifelong Learning – European Reference Framework*. Dostopno: <http://www.scribd.com/doc/33445618/Key-Competences-for-Lifelong-Learning-%E2%80%93-A-European-Framework> (21. 1. 2011).
- [6] e-inclusion – Digital Literacy – European commission Working Paper and Recommendations from Digital Literacy High-Level Expert Group. e-Inclusion Ministerial Conference & Expo, 30th November–2nd December 2008, Vienna, Austria. Dostopno: http://ec.europa.eu/information_society/eeurope/i2010/digital_literacy/index_en.htm (21. 1. 2011).
- [7] Eurostat. 2011. *Information Society Statistics*. Dostopno http://epp.eurostat.ec.europa.eu/portal/page/portal/information_society/data/database (20. 1. 2011, 23. 6. 2011).
- [8] i2010 – High Level Group. *Benchmarking Digital Europe 2011–2015, a conceptual framework*. Issue No. 27, October, 2009. Dostopno: http://ec.europa.eu/information_society/eeurope/i2010/benchmarking/index_en.htm (21. 1. 2011).
- [9] Najdič, F., M. Podbršček in V. Osojnik. (2007). *PTI – Katalog znanja: Informatika*. Ljubljana: Zavod RS za šolstvo.
- [10] Najdič, F. (2009). *Koncept vključevanja ključne kvalifikacije informacijsko-komunikacijsko opismenjevanje v izobraževalne programe srednjega poklicnega izobraževanja*. Center RS za poklicno izobraževanje in Zavod RS za šolstvo. Dostopno: <http://www.cpi.si/strokovna-podrocja.aspx> (15. 6. 2011).
- [11] Podovšovnik Axelsson, E. (2009). *Socialno-psihološki faktorji in družbene determinante računalniške in internetne pismenosti med slovenskimi osnovnošolskimi maturanti*. Ljubljana: Pedagoški inštitut.
- [12] Slovensko društvo Informatika. *Islovar*. II. izdaja. Dostopno: <http://www.islovar.org/> (21. 1. 2011).
- [13] Sulčič, V. (2011). Računalniška in internetna pismenost, vstopnica za informacijsko družbo. V *Nove razmere in priložnosti v informatiki kot posledica družbenih sprememb*, 18. konferenca Dnevi slovenske informatike, Portorož, Slovenija, 18.–20. april 2011.
- [14] SURS – Statistični urad RS. *Informacijska družba – Uporaba informacijsko-komunikacijske tehnologije*. Dostopno: http://www.stat.si/tema_ekonomsko_infdruzba_informacijsko.asp (19. 1. 2011).
- [15] Wechtersbach, R., Batagelj, V., Krapež, A. (2008). *Učni načrt: Informatika – Gimnazija; Splošna, klasična, strokovna*. Ljubljana: Zavod RS za šolstvo.

■

Viktorija Sulčič, izredna profesorica za področje poslovne informatike, je zaposlena na Fakulteti za management Univerze na Primorskem v Kopru. Raziskovalno se ukvarja s proučevanjem uporabe informacijske in komunikacijske tehnologije na področju izobraževanja. Zanimajo jo tako učinki uporabe kot tudi ovire, da informacijske in komunikacijske tehnologije ne uporabljamo bolj intenzivno. Na fakulteti je članica raziskovalnega programa 2009–2013. Kot pobudnica skupnosti Moodle uporabnikov v Sloveniji (www.moodle.si) organizira letne moodle.si konference. V projekt E-šolstva je vključena kot avtorica seminarja za sodelovalno delo v spletni učilnici Moodle.

Integracija procesiranja kompleksnih dogodkov in storitveno usmerjenih arhitektur

Martin Potočnik, Matjaž B. Jurič

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, Tržaška cesta 25, 1000 Ljubljana

martin.potocnik@fri.uni-lj.si; matjaz.juric@fri.uni-lj.si

Izvelek

Inteligentni in agilni moderni informacijski sistemi se morajo v vsakem okolju pravilno in učinkovito odzivati na preproste in kompleksne dogodke, saj so ti pomembni za poslovanje. Zaradi dinamičnosti distribuiranega okolja se povečuje kompleksnost informacijskih sistemov, hkrati pa pomembnost in število dogodkov strmo naraščata. Ker učinkovito procesiranje predvsem kompleksnih dogodkov postaja vedno bolj težavno, a hkrati pomembno, je v članku predstavljena konceptualna rešitev, ki temelji na konvergenci storitveno usmerjenih in dogodkovno gnanih arhitektur. Ker obstoječe platforme omogočajo procesiranje le preprostih dogodkov (kompleksni dogodki se obravnavajo v ločenih informacijskih rešitvah), predlagamo model komponente, ki omogoča celovito procesiranje kompleksnih dogodkov na ravni storitvenega vodila. Predlagana komponenta je povezljiva z obstoječimi sistemi za upravljanje s poslovnimi pravili in sistemi za spremljanje poslovnih aktivnosti ter ponuja zmožnost definiranja, zaznavanja in odzivanja na kompleksne dogodke. Zagotavlja centraliziran nadzor in upravljanje nad vsemi dogodki, kar omogoča učinkovit razvoj naprednih, visoko agilnih in inteligentnih informacijskih sistemov.

Ključne besede: procesiranje kompleksnih dogodkov, storitveno usmerjene arhitekture, dogodkovno gnane arhitekture, storitveno vodilo.

Abstract

Integration of Complex Event Processing into Service Oriented Architecture

Most of today's information systems do not fulfill all goals in sense of event responsiveness. As the dynamism of distributed environments is rapidly increasing, efficient event processing is becoming more and more important. In this paper we present the convergence of Service Oriented Architecture and Event Driven Architecture which are two different paradigms that address complex integration challenges. Their convergence provides necessary capabilities for the development of intelligent, agile and flexible information systems. Because the existing SOA platforms only support processing of simple events (complex event processing is done by external dedicated systems) we propose a new approach to complex event processing in SOA. We define a Service Component Architecture (SCA) compliant component that can be integrated into Enterprise Service Bus (ESB) and can leverage existing Business Rule Management Systems and Business Activity Monitoring systems. The proposed solution offers us a comprehensive platform for complex event processing, centralized event management and event governance.

Key words: Complex Event Processing, Service Oriented Architecture, Event Driven Architecture, Enterprise Service Bus.

1 UVOD

V zadnjih letih je razmah tehnoloških platform omogočil razvoj inteligentnih, agilnih in skalabilnih informacijskih sistemov, s katerimi se lahko organizacije prilagajajo potrebam dinamičnega poslovanja. Dinamičnost distribuiranega okolja predstavljajo številne spremembe oz. dogodki, na katere se morajo informacijski sistemi odzivati pravilno. Danes že uveljavljena in sprejeta pristopa – SOA (Service Oriented Architecture) in EDA (Event Driven Architecture) – delno rešujeta težave agilnosti in kompleksnosti, vendar se z naraščanjem pomembnosti in števila dogodkov, na katere se želimo odzivati pravočasno in pravilno, kompleksnost spet povečuje. Za procesiranje pre-

prostih dogodkov obstajajo pristopi in tehnološke rešitve, ki so danes že uveljavljene; problem je procesiranje kompleksnih dogodkov oz. CEP (Complex Event Processing), ki je zahtevno, a izredno zaželeno, saj ravno kompleksni dogodki vsebujejo pomembne informacije oz. novo znanje. Ker celovite in učinkovite rešitve za definiranje, analizo, procesiranje ter shranjevanje kompleksnih dogodkov na platformi SOA še ni, predlagamo model »komponente CEP«, ki to omogoča. V nadaljevanju so najprej predstavljene storitveno usmerjene arhitekture in dogodkovno gnane arhitekture, nato so opisane dosedanje raziskave in znanstvena dognanja. Tretji razdelek identificira in opisuje problematiko procesiranja kompleksnih dogodkov.

V četrtem razdelku sta predstavljena predlagani pristop k integraciji procesiranja kompleksnih dogodkov v platformo SOA in model predlagane komponente CEP. Peti razdelek vsebuje potencialni znanstveni doprinos in sklep.

1.1 Storitveno usmerjene arhitekture

Danes že uveljavljeni in sprejeti koncept storitveno usmerjenih arhitektur (SOA) predstavlja funkcionalne entitete obstoječih aplikacij, kot šibko sklopljene in ponovno uporabljive storitve. To množico interoperabilnih storitev, ki so uporabljive v različnih poslovnih domenah in informacijskih rešitvah, s kompozicijo ali orkestracijo povezujemo v poslovne procese. Kompozicija in orkestracija storitev se običajno realizira z jeziki, kot je Business Process Execution Language (BPEL) (Alves et al., 2006). Če bi kompozicija ali orkestracija storitev le-te povezovala neposredno, bi arhitekture takšnih informacijskih sistemov postale izredno kompleksne in nepregledne. Zato SOA vpelje storitveno vodilo ali ESB (Enterprise Service Bus), ki je vmesna programska oprema ali platforma, ki omogoča dinamično povezovanje in posredovanje storitev ter nadzor nad njihovimi interakcijami, saj odstrani neposredne povezave med ponudniki in uporabniki storitev. Tako nam zagotavlja šibko sklopljenost, povečano fleksibilnost, celovit pogled na povezljivost, varnost ter primerno dostavo storitev. Storitveno vodilo nam torej ponuja abstrakten pogled na storitve, saj zagotavlja poenostavljen vmesnik za komuniciranje in integracijo storitev.

Večina današnjih platform SOA temelji na specifikaciji SCA (Service Component Architecture), ki definira način izgradnje in model združevanja komponent v tehnološko neodvisne kompozite, iz katerih gradimo aplikacije. Vsaka komponenta je atomarna entiteta in vsebuje storitve (predstavljajo poslovno logiko oz. funkcionalnosti), reference (omogočajo dostop do storitev drugih komponent), specifične lastnosti (za definiranje obnašanja ipd.) in povezave (za interakcijo) (Beisiegel & Blohm, 2007).

Ponudniki in uporabniki storitev med seboj komunicirajo z uporabo natančno definiranih vmesnikov, prek katerih lahko storitve pošiljajo in prejemajo sporočila. Klicatelj oz. odjemalec proži specifično operacijo na določeni storitvi ali komponenti, ki obravnava in izvede prejeto zahtevo. Dvosmerne operacije vključujejo tudi odgovor, ki ga storitev kot rezultat vrne odjemalcu. Tak stil interakcije običajno povzroča

tesnejšo sklopljenost in nižji nivo ponovne uporabe (Erl, 2005). Te slabosti lahko premostimo z uporabo koncepta EDA, ki je opisan v naslednjem razdelku.

1.2 Dogodkovno gnane arhitekture in procesiranje kompleksnih dogodkov

Koncept dogodkovno gnanih arhitektur (EDA) je arhitekturni pristop, ki temelji na generiranju, zaživanju, sprejemanju in obravnavanju dogodkov. V splošnem je dogodek vsaka akcija, ki se zgodi zunaj naše aplikacije in jo lahko obravnavamo v njej. V kontekstu večjih informacijskih sistemov pa je dogodek definiran kot sprememba stanja poljubnega elementa poslovnega procesa, ki ima potencialni vpliv na tok izvajanja ali rezultat poslovnega procesa (Levina & Stantchev, 2009). Dogodki so lahko preprosti ali kompleksni. Primeri preprostih dogodkov so npr. izdaja računa, rezervacija letalske vozovnice in sporočilo od temperaturnega senzorja. Kompleksni dogodki so dogodki, ki nastanejo kot rezultat več preprostih dogodkov in običajno niso vidni. Primer je npr. nakup več letalskih vozovnic v kratkem časovnem intervalu za specifično destinacijo ali nepričakovano upadanje naročil za določeni izdelek. Za distribucijo dogodkov potrebujemo dogodkovno vodilo, ki je lahko realizirano kot MOM (Message Oriented Middleware) – JMS (Java Message Service) in MQ (Message Queue), informacijska rešitev za procesiranje dogodkov – Oracle CEP ali IBM WebSphere Business Events, ali kot storitveno vodilo, ki podpira propagacijo dogodkov.

Koncept EDA temelji na šibko sklopljenih komponentah, ki med seboj komunicirajo po vzorcu objavnararoči, kar pomeni, da je vsak dogodek dostavljen vsem prejemnikom, ki so naročeni na ta dogodek. To implicira strogo delitev med generatorji dogodkov ali dogodkovnimi izvori in prejemniki dogodkov ali dogodkovnimi ponori, ki so poleg dogodkovnega senzorja in sistema za procesiranje dogodkov osnovne komponente dogodkovno gnane arhitekture. Dogodkovni senzor je odgovoren za pravilno zaznavanje dogodkov, sistem za procesiranje dogodkov pa vse dogodke evalvira in se nanje odzove z vnaprej definirano akcijo, ki je običajno definirana s poslovnimi pravili.

EDA definira tri tipe dogodkov s pripadajočimi mehanizmi za njihovo procesiranje:

- preprosti dogodek (SEP – Simple Event Processing) – nanaša se na specifično spremembo stanja,

ki lahko povzroči eno akcijo ali več. V sistemu za procesiranje preprostih dogodkov (SEP) se vedno procesira le en sam dogodek;

- dogodkovni tok (ESP – Event Stream Processing) – sistem za procesiranje toka dogodkov spremlja dogodke v nekem časovnem okviru, ki mu pravimo drsno okno. Dogodki se ne procesirajo individualno, ampak se ob prejemu vsakega novega dogodka ovrednoti množica vseh aktualnih dogodkov (tistih, ki so znotraj drsnega okna);
- kompleksni dogodek (CEP – Complex Event Processing) – sestavljen je iz več preprostih dogodkov, ki v sistem (ne) pridejo bodisi v določenem številu, časovnem intervalu ali določenem zaporedju. Sistem za procesiranje kompleksnih dogodkov poleg ESP omogoča tudi zaznavo kompleksnih vzorcev v množici vseh dogodkov.

V članku se usmerjamo na procesiranje kompleksnih dogodkov, saj so funkcionalnosti SEP in ESP že omogočene na današnjih platformah. CEP je tehnologija za filtriranje, koreliranje, agregiranje in analiziranje dogodkovnih podatkov, ki lahko izvirajo iz vseh plasti organizacije, kar nam omogoča, da pridobimo nova znanja iz distribuiranih sistemov (Eckert, Michael & Bry, 2009; Paschke, 2008). Stroji za procesiranje kompleksnih dogodkov za definiranje povezav in odvisnosti med dogodki uporabljajo klasifikacijo, poslovna pravila in filtre. Definiranje kompleksnih dogodkov je običajno realizirano s pomočjo sistemov za upravljanje s poslovnimi pravili (BRMS – Business Rule Management System), ki omogočajo nadzor in upravljanje procesiranja dogodkov v času izvajanja. Ko stroj za procesiranje kompleksnih dogodkov zazna kompleksni dogodek, lahko dogodek posreduje vsem naročenim komponentam ali pa se odzove neposredno (npr. klic specifične storitve, pošiljanje sporočila, zagon poslovnega procesa).

CEP nam omogoča razvoj varnejših in robustnejših informacijskih sistemov, saj se dogodki analizirajo in ovrednotijo pred vplivom na sistem. Tako npr. deluje programska oprema za upravljanje z letalom, ki pilotu ne pusti, da povzroči nevarno stanje v sistemu (dogodke, ki bi lahko povzročili nevarno stanje, stroj za procesiranje dogodkov zavrne).

1.1 Konvergenca SOA in EDA

Klasični pristop SOA ne zadošča potrebam in zahtevam modernega poslovnega okolja, saj njegova dinamičnost zahteva visoko agilno arhitekturo z

avtonomnimi, šibko sklopljenimi komponentami in zmožnostjo odzivanja na dogodke. Na drugi strani nam EDA omogoča to dinamičnost, vendar sama ni primerna za celovito informacijsko arhitekturo, saj, prvič, ne omogoča najpogosteje uporabljenega načina komuniciranja – sinhrono dvosmerne komunikacije s sporočili (zahteva–odgovor), in drugič, povezave med komponentami so lahko nejasne in prikrite, kar povzroča težave pri upravljanju in obravnavanju napak.

Konvergenca SOA in EDA je torej smiselna; SOA 2.0 ali dogodkovno gnana SOA združuje dobre lastnosti obeh paradig, saj razširja storitveno usmerjeni pristop z zmožnostmi dogodkovno gnane koncepta in tako ponuja te prednosti:

- šibka sklopljenost med komponentami,
- boljše arhitekturna fleksibilnost,
- boljši vpogled v poslovne aktivnosti in poslovne procese,
- nadzor oz. spremljanje poslovnih aktivnosti in odzivnost na okolje (procesiranje dogodkov) v skoraj realnem času,
- zaznavanje in optimalno odzivanje na preproste in kompleksne dogodke.

Zavedati se moramo, da je dogodkovno gnana SOA ali SOA 2.0 arhitekturni pristop oz. koncept in da celovita SOA 2.0 platforma še ne obstaja.

2 PREGLED DOSEDANJIH RAZISKAV

V zadnjem desetletju je veliko avtorjev dognalo, da je za razvoj modernih informacijskih sistemov potrebna konvergenca SOA in EDA oz. razširitev platform SOA z dogodkovno gnanimi koncepti. Woods in Mattern (Woods & Mattern, 2006) sta natančno identificirala razloge in potencialne prednosti združevanja SOA in EDA. Taylor (Taylor, Yochem, Phillips, & Martinez, 2009) je potrdil smiselnost konvergence obeh konceptov ter dobro opisal njune ključne komponente. Michelson (Michelson, 2006) je definiriral in opisal razlike med SOA in EDA in izpostavil relacijo med EDA in CEP. Marechaux (Marechaux, 2006) je opisal SOA in EDA kot dva različna koncepta in kot arhitekturni vzorec za podporo dogodkom (proženje, detekcija, distribucija) v arhitekturi SOA predlagal storitveno vodilo. Laliwala in Chaudhary (Laliwala & Chaudhary, 2008) sta prav tako prišla do spoznanja, da obstoječe platforme SOA ne podpirajo vseh konceptov dogodkovno gnanih arhitektur. Predlagala sta dogodkovno gnano SOA, ki temelji

na spletnih storitvah in semantičnem spletu. Tako bi lahko s pomočjo upravljalca dogodkov, stroja za kompozicijo in izvajanje, ontološkega strežnika, mrežnih storitev ter poslovnih pravil avtomatizirala dogodkovno gnane poslovne procese. Wieland (Wieland, Martin, Kopp, & Leymann, 2009) je odkril, da sta SOA in EDA edinstvena arhitekturna stila, ki se v industriji večinoma uporabljata ločeno. Definiral je tudi metodo za razvoj poslovnih procesov SEODA, ki v BPEL propagira dogodke. Jurič (Juric, 2010) je za podporo konceptov EDA v SOA prvi predlagal razširitve WSDL in BPEL. Z razširitvami za definiranje dogodkovnih izvorov in ponorov WSDL lahko definiramo in uporabljamo dogodke na ravni spletnih storitev. BPEL je avtor razširil z dodatnimi aktivnostmi, ki omogočajo dogodkovno gnano izvajanje poslovnih procesov. Vsi predlagani koncepti nam omogočajo asinhrono komunikacijo z vzorcem objavi-naroči in procesiranje preprostih dogodkov.

V zadnjih letih so bile predlagane, razvite in izdane številne specifikacije, kot sta WS-Eventing (Box et al., 2004) in WS-Notification (P. Niblett & S. Graham, 2005). WS-Eventing definira dogodkovno storitev z naborom operacij, ki spletni storitvi omogočajo asinhrono oz. dogodkovno gnano komunikacijo z drugimi storitvami. WS-Notification sestoji iz treh specifikacij, s pomočjo katerih si lahko spletne storitve izmenjujejo sporočila: WS-BaseNotification (Steve Graham, Hull, Murray, & Event-driven, 2006), ki definira interakcijo med generatorji in prejemniki obvestil, WS-BrokeredNotification (Liu, 2006), ki definira vmesnike za posrednike obvestil, in WS-Topics (Peter Niblett, 2006), ki definira hierarhično področje za teme. Niblett in Graham sta naredila celovit pregled specifikacije WS-Notification (P. Niblett & S. Graham, 2005), Huang in Gannin pa sta kmalu za tem primerjala WS-Eventing in WS-Notification (Huang, 2006). Leta 2006 so HP, IBM, Intel in Microsoft predlagali specifikacijo WS-EventNotification, ki združuje WS-Eventing in WS-Notification (Kevin Cline et al., 2006). Procesiranje preprostih dogodkov in asinhrono komunikacijo na ravni SCA je s pomočjo razširitev predlagal Beisiegel (Beisiegel & Vorthmann, 2009).

Vse omenjene specifikacije nam omogočajo uporabo in procesiranje le preprostih dogodkov. Celovit pristop k vpeljavi procesiranja kompleksnih dogodkov v SOA še ni bil opisan ali predlagan.

3 PREDSTAVITEV PROBLEMA

Veliko število današnjih informacijskih sistemov predstavljajo distribuirani informacijski sistemi, kar pomeni visoko raven kompleksnosti, ki je posledica kompleksnosti samih poslovnih procesov in okolja, ki vpliva na izvajanje le-teh. Procesna usmerjenost oz. upravljanje poslovnih procesov (BPM – Business Process Management) in SOA zmanjšujeta to kompleksnost, pomembnost kompleksnih dogodkov, želja po optimizaciji, dinamičnosti in učinkovitem odzivanju informacijskih sistemov na okolje (dogodke), pa jo, žal, zvišujejo. Z večanjem števila storitev in senzorskih naprav (Internet of Things) se število dogodkov povečuje, kar implicira pomembnost integracije, konsolidacije in inteligentne obravnave distribuiranih dogodkov. Problem današnjih platform SOA je, da delno podpirajo koncepte dogodkovne usmerjenosti (SOA 2.0 ali Event-driven SOA), ne podpirajo pa konceptov procesiranja kompleksnih dogodkov (podpirajo SEP). CEP zajema metode, tehnike in orodja za procesiranje dogodkov v času izvajanja in omogoča izpeljavo kompleksnih dogodkov, ki nosijo obogatene informacije in znanje. Pri realizaciji sodobnih dinamičnih in distribuiranih informacijskih sistemov, ki podpirajo dogodkovno vodene koncepte, se tako še vedno soočamo z izzivi:

- Kako upravljati z velikim številom distribuiranih dogodkov?
- Kako prepoznati kompleksne vzorce iz množice preprostih dogodkov?
- Kako pridobiti znanje iz meddogodkovnih povezav?
- Kako pridobiti zmožnost napovedovanja dogodkov?
- Kako učinkovito implementirati in uporabljati poslovna pravila, ki bi podpirala tudi CEP?
- Kako učinkovito integrirati CEP, sisteme za upravljanje s poslovnimi pravili (BRMS) in sisteme za spremljanje poslovnih aktivnosti (BAM)?

Zaradi omenjenih problemov danes vse platforme SOA rešujejo CEP z ločenimi produkti (npr. IBM WebSphere Business Events ali Oracle Complex Event Processing). Kljub omenjenim informacijskim rešitvam, ki omogočajo CEP, je razvoj inteligentnih in agilnih informacijskih sistemov SOA, ki uporabljajo CEP, izredno zahtevno. Za opisane probleme predlagamo rešitev, ki je opisana v naslednjem razdelku.

4 PREDLAGANA REŠITEV

Predlagamo inovativen pristop k integraciji CEP in SOA. Naš cilj je vpeljati možnost celovitega procesiranja kompleksnih dogodkov, med katere ta semantično spada – na hrbtnico vsake platforme SOA oz. na storitveno vodilo (ESB). Ko govorimo o celovitem procesiranju kompleksnih dogodkov, mislimo na definiranje, zaznavanje, analiziranje, obravnavanje, predobdelavo, obdelavo, konsolidiranje, filtriranje, klasificiranje, agregiranje, integriranje, usmerjenje in posredovanje dogodkov.

4.1 Vpeljava procesiranja kompleksnih dogodkov v storitveno vodilo s komponento CEP

Marechaux (Maréchaux, 2006) je v svojem članku pokazal, da je storitveno vodilo idealen arhitekturni vzorec za procesiranje dogodkov, saj pomeni centralizirano entiteto za komunikacijo med storitvami, procesi, aplikacijami, človeškimi opravili, hkrati pa strogo razdvoji ponudnike in uporabnike storitev, kar je ključnega pomena za dogodkovno gnano (asinhrono objavi–naroči) komunikacijo.

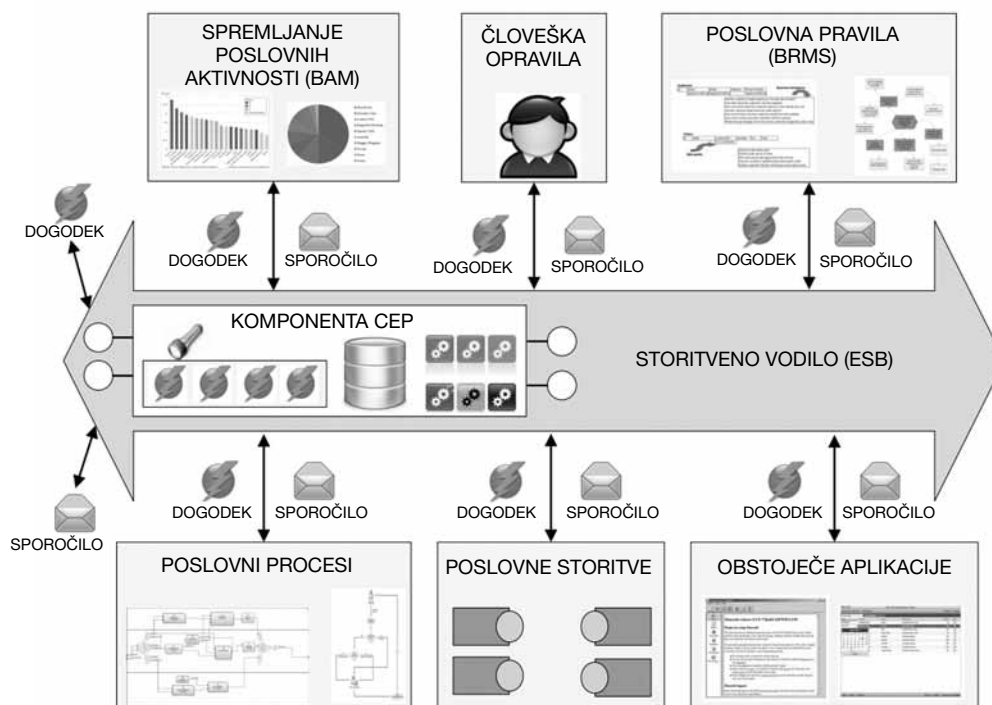
Za vpeljavo procesiranja kompleksnih dogodkov predlagamo t. i. avtonomno in celovito komponento CEP, ki je skladna s specifikacijo SCA (Beisiegel

& Blohm, 2007). Skladnost s specifikacijo SCA je pomembna, saj želimo, da se da komponento CEP preprosto integrirati v obstoječe platforme SOA (na storitveno vodilo), ki temeljijo na SCA. Predlagani model je tipičen kompozit SCA, ki vsebuje osem komponent. Vse komponente in njihove medsebojne povezave so opisane v naslednjem razdelku.

Kot lahko vidimo na sliki 1, predlagana komponenta CEP ni povezana na storitveno vodilo kot sistemi za upravljanje s poslovnimi pravili (BRMS) ali sistemi za spremljanje poslovnih aktivnosti (BAM), ampak je popolnoma integrirana v ogrodje storitvenega vodila. Vsebuje:

- lastno vrsto za dogodke,
- podatkovni vir za shranjevanje,
- mehanizme za definiranje in procesiranje kompleksnih dogodkov in
- mehanizme za integracijo z zunanjimi storitvami in sistemi.

Komponenta uporablja interne podatkovne sheme, ki so primerne za učinkovito definiranje in procesiranje kompleksnih dogodkov ter shranjevanje letih. Vmesniki so definirani generično, kar omogoča integracijo z eksternimi sistemi prek storitvenega vodila.



Slika 1: Integracija komponente CEP v storitveno vodilo

Ker sisteme BRMS in BAM danes pogosto uporabljajo, menimo, da je zmožnost integracije komponente CEP z njimi pomembna. V ta namen komponenta vsebuje posebne vmesnike in programabilne transformacijske module oz. vtičnike, prek katerih jo lahko povežemo z obstoječimi BRMS in BAM. Transformacijski vtičniki so ključnega pomena, saj zagotavljajo pravilno dvosmerno preslikavo definicij kompleksnih dogodkov in drugih podatkov v primerne podatkovne formate. To nam omogoča, da kompleksne dogodke in odzive nanje definiramo z obstoječimi sistemi BRMS in da za spremljanje poslovnih aktivnosti in kompleksnih dogodkov (kompleksni KPI – Key Performance Indicators) lahko uporabljamo obstoječe sisteme BAM.

4.2 Predlagani model komponente CEP

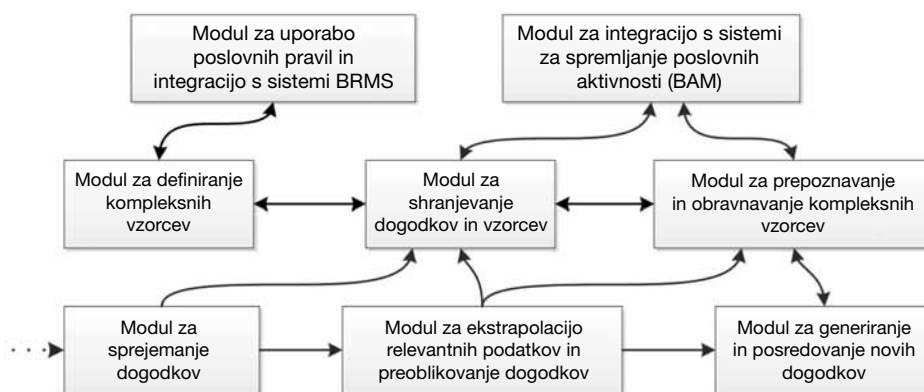
Za učinkovito vpeljavo celovitega procesiranja kompleksnih dogodkov (definiranje, zaznavanje, analiziranje, obravnavanje, predobdelava, obdelava, konsolidiranje, filtriranje, klasificiranje, agregiranje, integriranje, usmerjenje in posredovanje kompleksnih dogodkov) v storitveno vodilo predlagamo konceptualni model kompozita SCA, ki ga sestavlja osem modulov:

- *modul za sprejemanje dogodkov* – ta modul s storitvenega vodila sprejema vse dogodke in jih posreduje potrebnim modulom ali zunanjim sistemom;
 - *modul za ekstrapolacijo relevantnih podatkov in preoblikovanje dogodkov* – ta modul iz dogodkovnih podatkov izlušči pomembne podatke, ki so potrebni za optimalno procesiranje. Dogodke za posredovanje ali boljšo obdelavo lahko modul tudi preoblikuje;
 - *modul za definiranje kompleksnih vzorcev* – ta modul služi za konstrukcijo vzorcev, s katerimi opišemo kompleksne dogodke (vzorci je mogoče definirati tudi v zunanjih sistemih BRMS). Za preprosto in učinkovito definiranje kompleksnih vzorcev lahko uporabimo posebne strukture. K temu modulu spada tudi grafični vmesnik, ki pa je lahko realiziran poljubno, ker je definiranje vzorca realizirano kot človeško opravilo z natančno specifičnim vmesnikom. Tako za definiranje kompleksnih dogodkov oz. vzorcev dosežemo popolno tehnološko neodvisnost;
 - *modul za shranjevanje dogodkov in vzorcev* – ta modul je odgovoren za shranjevanje vseh dogodkov in definiranih vzorcev. Shranjevanje dogodkov bo potekalo na dva načina:
 - kratkoročno shranjevanje bo namenjeno ESP (procesiranje dogodkovnih tokov) z drsečim oknom. Procesirajo se zadnji n-dogodki ali dogodki v določenem časovnem intervalu;
 - dolgoročno shranjevanje (v podatkovno bazo) je namenjeno arhiviranju vseh dogodkov. To omogoča kasnejšo analizo in prepoznavo kompleksnih vzorcev.
- Vzorci (njihovi formalni opisi oz. definicije) se prav tako shranjujejo na dva načina. Aktivni vzorci (tisti, ki jih trenutno prepoznavamo) so shranjeni v pomnilniku, pasivni vzorci pa so arhivirani v podatkovni bazi. Vsi vzorci se shranjujejo na dva načina:
- v internem formatu, ki ga definira komponenta CEP;
 - v originalnem formatu, ki ga definirajo zunanji sistemi BRMS (original potrebujemo za dvosmerno sinhronizacijo);
- *modul za uporabo poslovnih pravil in integracijo z BRMS sistemi* – ta modul omogoča integracijo z zunanjimi sistemi BRMS, kar pomeni možnost uporabe poslovnih pravil, ki so definirana zunaj našega sistema. Modul je prek vtičnikov sposoben sprejemati poslovna pravila različnih formatov in jih transformirati v interno shemo kompleksnih dogodkov. Tako omogočimo uporabo že definiranih pravil, hkrati pa lahko uporabniki za definiranje kompleksnih dogodkov in njihovih posledic uporabljajo obstoječe sisteme;
 - *modul za integracijo s sistemi za spremljanje poslovnih aktivnosti (BAM)* – ker je spremljanje poslovnih aktivnosti za poslovanje izredno pomembno, komponenta CEP omogoča integracijski modul za uporabo zunanjih sistemov BAM. Modul informacije o kompleksnih dogodkih tako lahko posreduje različnim sistemom BAM;
 - *modul za prepoznavanje in obravnavanje kompleksnih vzorcev* – ta osrednji modul CEP je odgovoren za prepoznavanje kompleksnih aktivnih vzorcev. Ko je zaznan kompleksni dogodek, modul poskrbi za pravilno obravnavo (npr. sproži določeno storitev, človeško opravilo ali proces, dogodek posreduje v druge sisteme ali ga samo zabeleži);
 - *modul za generiranje in posredovanje novih dogodkov* – ta modul generira dogodke in jih posreduje na pravilne končne točke. Iz vhodnih podatkov je sposoben generirati dogodek pravilnega tipa in ga posredovati zunanjim ali internim sistemom.

Največ ga uporablja modul za prepoznavanje in obravnavanje dogodkov.

Na sliki 2 je prikazana shema predlaganega modela komponente CEP. Vhod v komponento je spodaj levo – v modulu za sprejemanje dogodkov. Ta modul zazna dogodek in ga posreduje v modula za shranjevanje dogodkov in za inteligentno predobdelavo (ekstrapolacija relevantnih podatkov in preoblikovanje dogodkov v primernejšo obliko). Zadnji modul shrani preoblikovani dogodek, hkrati pa ga preda modulu za prepoznavanje in obravnavanje kompleksnih vzorcev. Za proženje vnaprej definiranih akcij poskrbi modul za generiranje in posredovanje novih dogodkov. Pomemben del predlagane komponente je definiranje kompleksnih vzorcev, ki se posredno lahko vrši v modulu za uporabo poslovnih pravil in integracijo s sistemi BRMS, neposredno pa v modulu za definiranje kompleksnih vzorcev. Modul za integracijo s sistemi BAM poskrbi za učinkovito in transparentno komunikacijo z zunanjimi sistemi za spremljanje poslovnih aktivnosti. Za boljše razumevanje ponazorimo delovanje modela s primerom. V obsto-

ječem sistemu BRMS definiramo kompleksni vzorec, ki ga sestavlja število transakcij (na uporabnika), ki je za 30 odstotkov višje od povprečja, v časovnem intervalu 90 minut, vsaka od transakcij pa presega znesek 20 evrov. Ko nastopi kompleksni dogodek, želimo poklicati določeno spletno storitev. Ta vzorec se v modulu za definiranje kompleksnih vzorcev transformira v primerno obliko in shrani v podatkovno bazo. Ko v sistem začnejo prihajati dogodki, jih modul za sprejemanje pošlje v obdelavo in jih shrani. Modul za prepoznavanje in obravnavanje kompleksnih vzorcev ob nastopu vsakega dogodka preveri, ali so izpolnjeni pogoji za nastop kompleksnega dogodka – če so, poskrbi za pravilno obravnavno. Ko v sistem prispe prva transakcija, ki presega mejni znesek, se zaradi časovnega intervala zažene časovnik, ki teče največ 90 minut. Če v tem intervalu število transakcij preseže mejo, modul za prepoznavanje in obravnavanje pokliče ustrezno storitev in shrani kompleksni dogodek. Sistem BAM lahko ta dogodek prikaže na poslovnem portalu in tako omogoča učinkovit nadzor in upravljanje.



Slika 2: Shema predlaganega modela

Za konvergenco storitveno usmerjenih sekvenčnih poslovnih procesov in naključno zaznanih kompleksnih dogodkov lahko uporabimo mehanizme za obravnavo dogodkov, s katerimi se lahko pravilno odzivamo na dogodke. Predlagana komponenta CEP nam omogoča, da pred dejanskim vplivom dogodkov na poslovni proces na inteligenten način obravnavamo večje število preprostih in kompleksnih dogodkov (neodvisno od poslovnih procesov). S tem

omogočimo optimalnejši razvoj poslovnih procesov, saj kompleksno logiko za obravnavanje dogodkov prestavimo v centralizirano komponento, ki nam zagotovi prepoznavanje kompleksnih dogodkov. Poslovni procesi tako postanejo bolj pregledni in inteligentni, saj se lahko odzivajo na kakršne koli dogodke. Ker predlagana komponenta hrani vse dogodke in vzorce, lahko nad temi podatki izvajamo analize (poslovna inteligenca) in pridemo do novih znanj,

s katerimi lahko optimiziramo poslovne procese in definiramo vzorce za nove kompleksne dogodke ali kompleksne kazalnike uspešnosti (KPI).

5 SKLEP

V članku sta predstavljena koncepta storitveno usmerjenih arhitektur in dogodkovno gnanih arhitektur, ki sta dva arhitekturna pristopa za integracijo informacijskih sistemov. Izpostavili smo razlike v komunikaciji in pomanjkljivosti obeh pristopov ter pokazali, da je njuna konvergenca smiselna in nujna za razvoj modernih agilni in inteligentnih sistemov. Predstavili smo dosedanje raziskave s področja SOA in EDA in prišli do spoznanja, da trenutne arhitekture oz. platforme ne omogočajo procesiranja kompleksnih dogodkov. Ker CEP postaja za optimalno izvajanje poslovnih procesov izrednega pomena, menimo, da bi procesiranje kompleksnih dogodkov moralo biti del same platforme SOA 2.0 (Event-Driven SOA). Za integracijsko platformo, ki bi omogočala vpeljavo CEP v SOA, smo identificirali storitveno vodilo (ESB), saj ta ponuja idealno ogrodje za prejemanje in distribucijo podatkov. Predlagali smo avtonomno komponento CEP, ki omogoča celovito procesiranje kompleksnih dogodkov (definiranje, zaznavanje, analiziranje, obravnavanje, predobdelava, obdelava, konsolidiranje, filtriranje, klasificiranje, agregiranje, integriranje, usmerjenje in posredovanje kompleksnih dogodkov). Predlagana komponenta CEP je skladna s specifikacijo SCA, visoko konfigurabilna in preprosta za integracijo v obstoječe platforme SOA oz. njihova storitvena vodila. Model komponente sestavlja osem modulov oz. komponent SCA, ki so odgovorne za posamezne funkcionalne zahteve procesiranja kompleksnih dogodkov. Spredlaganim pristopom ponujamo nov model za procesiranje kompleksnih dogodkov v obstoječih in razvijajočih se platformah SOA. S centraliziranim nadzorom in upravljanjem preprostih in kompleksnih dogodkov ter z učinkovitim procesiranjem le-teh so omogočeni krajši čas razvoja zaradi preproste arhitekture, višja stopnja agilnosti in odzivnosti na okolje in boljši pregled nad poslovnimi aktivnostmi in procesi.

6 VIRI IN LITERATURA

- [1] Alves, A., Arkin, A., Askary, S., Bloch, B., Curbera, F., Golland, Y., Kartha, N. et al. (2006). *Web services business process execution language version 2.0. Language* (pp. 1–264). Retrieved from <http://www.citeulike.org/group/2540/article/1233272>.
- [2] Beisiegel, M., & Blohm, I H. (2007). *SCA Service Component Architecture – Assembly Model Specification*. Cite-seer. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.168.5919&rep=rep1&type=pdf>.
- [3] Beisiegel, M., & Vorthmann, S. (2009). *SCA Extensions for Event Processing and Pub/Sub*. Retrieved from http://www.osea.org/download/attachments/35/SCA_Assembly_Extensions_for_Event_Processing_and_PubSub_V1_0.pdf?version=1.
- [4] Box, D., Cabrera, L. F., Critchley, C., Curbera, Francisco, Ferguson, D., Graham, Steve, Hull, D. et al. (2004). *Web Services Eventing (WS-Eventing)*. Retrieved from <http://www.w3.org/TR/2009/WD-ws-eventing-20090317/>.
- [5] Eckert, Michael, F. & Bry, C. (2009). Complex Event Processing (CEP). *Informatik-Spektrum*, 32(2), 163–167.
- [6] Erl, T. (2005). *Service-oriented architecture: concepts, technology, and design* (p. 792). Prentice Hall PTR. Retrieved from <http://scholar.google.com/scholar?hl=en&btnG=Search&q=ititle:Service-oriented+architecture:+concepts,+technology,+and+design#0>.
- [7] Graham, Steve, Hull, D., Murray, B. & Event-driven, T. (2006). *Web Services Base Notification 1.3. October*. Retrieved from http://docs.oasis-open.org/wsn/wsn-ws_base_notification-1.3-spec-os.pdf.
- [8] Huang, Y. (2006). A comparative study of web services-based event notification specifications. *2006 International Conference on Parallel Processing Workshops (ICPPW'06)*, 7–14. Ieee. doi:10.1109/ICPPW.2006.5.
- [9] Juric, M. B. (2010). WSDL and BPEL extensions for Event Driven Architecture. *Information and Software Technology*, 52(10), 1023–1043. Elsevier B.V. doi:10.1016/j.infsof.2010.04.005.
- [10] Kevin Cline, Cohen, J., Davis, D., Ferguson, D. F., Kreger, H., Mccollum, R., Murray, B. et al. (2006). *Toward Converging Web Service Standards for Resources, Events, and Management*. Changes (pp. 1–9).
- [11] Laliwala, Z. & Chaudhary, S. (2008). Event-driven Service-Oriented Architecture. *2008 International Conference on Service Systems and Service Management*, 1–6. Ieee. doi:10.1109/ICSSSM.2008.4598452.
- [12] Levina, O. & Stantchev, V. (2009). Realizing Event-Driven SOA. *2009 Fourth International Conference on Internet and Web Applications and Services*, 37–42. Ieee. doi:10.1109/ICIW.2009.14.
- [13] Liu, L. (2006). *Web Services Brokered Notification 1.3. October*. Retrieved from http://docs.oasis-open.org/wsn/wsn-ws_brokered_notification-1.3-spec-os.pdf.
- [14] Maréchaux, J.-L. (2006). *Combining Service-Oriented Architecture and Event-Driven Architecture using an Enterprise Service Bus*. Retrieved from <http://www.ibm.com/developerworks/webservices/library/ws-soa-eda-esb/index.html>.
- [15] Michelson, B. M. (2006). *Event-Driven Architecture Overview*. Architecture, 8. doi:10.1571/bda2-2-06cc.
- [16] Niblett, P. & Graham, S. (2005). Events and service-oriented architecture: The OASIS Web Services Notification specification. *IBM Systems Journal*, 44(4), 869–886. doi:10.1147/sj.444.0869.
- [17] Niblett, Peter. (2006). *Web Services Topics 1.3. October*. Retrieved from http://docs.oasis-open.org/wsn/wsn-ws_topics-1.3-spec-os.pdf.

- [18] Paschke, A. (2008). Design patterns for complex event processing. *2nd International Conference on Distributed Event-Based Systems (DEBS'08)*. Retrieved from <http://arxiv.org/abs/0806.1100>.
- [19] Taylor, H., Yochem, A., Phillips, L. & Martinez, F. (2009). *Event-Driven Architecture: How SOA Enables the Real-Time Enterprise*. (M. Taub, G. Doench, M. Thurston, K. Hart, B. Hassir, & K. Annett, Eds.) (p. 308). Addison-Wesley.
- [20] Wieland, M., Martin, D., Kopp, O. & Leymann, F. (2009). SO-EDA: A Methodology for Specification and Implementation of Applications on a Service-Oriented Event-Driven Architecture. In W. Abramowicz (Ed.), *Proceedings of the 12th International Conference on Business Information Systems BIS 2009 Poznan Poland April 27-29 2009* (Vol. 21, pp. 193–204). Springer Verlag. doi:10.1007/978-3-642-01190-0_17.
- [21] Woods, D. & Mattern, T. (2006). *Enterprise SOA: Designing IT for Business Innovation* (p. 452). O'Reilly.

■

Martin Potočnik je diplomiral na Fakulteti za elektrotehniko, računalništvo in informatiko Univerze v Mariboru in se kot raziskovalec zaposlil v laboratoriju za tehnologije komuniciranja. Kasneje je postal mladi raziskovalec in doktorski študent Fakultete za računalništvo in informatiko Univerze v Ljubljani, kjer je zaposlen v laboratoriju za integracijo informacijskih sistemov. Raziskovalno se ukvarja predvsem z integracijskimi izzivi, storitveno usmerjenimi arhitekturami in računalništvom v oblaku. Sodeluje tudi v aplikativnih industrijskih projektih ter se aktivno udeležuje številnih konferenc s področja informatike.

■

Matjaž B. Jurič je redni profesor na Fakulteti za računalništvo in informatiko Univerze v Ljubljani, kjer vodi laboratorij za integracijo informacijskih sistemov. Je avtor oz. soavtor štirinajstih knjig. Sodeloval je pri številnih projektih doma in v tujini, med drugim tudi pri razvoju RMI-IIOP, sestavnega dela platforme Java 2, in je član BPEL Advisory Boarda. Leta 2007 je od SOA World Journal dobil nagrado za najboljšo knjigo s področja SOA, leta 2010 pa nagrado za najodmevnejši znanstveni članek s področja storitev iz NMS. Ima naziva Java Champion in Oracle ACE Director.

Uporaba odprtokodnih in prosto dostopnih programskih orodij – primer dobre prakse

Mirica Šafran, Mateja Škornik, Dejan Skok, Borut Jereb
Univerza v Mariboru, Fakulteta za logistiko, Mariborska 7, 3000 Celje
mirica.safiran@gmail.com; mateja.skornik@fl.uni-mb.si; logistika.skok@gmail.com; borut.jereb@fl.uni-mb.si

Izvleček

Informacijska podpora poslovnim procesom je v modernem svetu najpomembnejši dejavnik uspešnega poslovanja. Številna programska orodja ponujajo celovite rešitve in pogostokrat se v praksi izkažejo za odlično investicijo. V ospredje pa vedno bolj nezadržno stopajo t. i. programska orodja odprte koda. Filozofijo in uporabnost odprtokodnih in prosto dostopnih programskih orodij smo preizkusili s projektom – nastala je knjiga *Programi za logistiko* (izdana pod licenco Creative Commons), v kateri smo zbrali, analizirali in testirali šestnajst različnih programskih orodij z vidika njihove uporabe na izbranem logističnem primeru šestih različnih dimenzij problematike upravljanja komponent vmesnega skladišča (upravljanje procesov, vizualizacija, prostorsko planiranje, odločanje, napovedovanje in statistične analize). Avtorji knjige želimo s projektom nadaljevati, prevesti knjigo v tuje jezike in spodbuditi javnost k soustvarjanju t. i. »žive« knjige. To potrjujejo tudi rezultati ankete, v kateri so anketiranci v veliki meri potrdili pripravljenost pri soustvarjanju knjige oz. razširjanju tovrstnih programskih orodij.

Ključne besede: odprta koda, prosto dostopna programska orodja, logistika, Planner, Dia, Zint, Google Zemlja, Asdn, Quantum GIS, Qcad, Simple Warehouse Mapper, Petersen, Lindo, GnuCash, GSS World, Scilab, OpenOffice.org, PSPP, Dexi.

Abstract

Freeware and Open Source Software Tools Application – an Example of Good Practice

The most important factor for business success in modern world is information support for business process. Software packages which offer comprehensive solutions are excellent investments. Lately the so called open source software gets increasingly into the focus. With a project (which results in the book *Programs for Logistics*) we test philosophy and usability of freeware and open source software. The book *Programs for Logistics* was released under the Creative Commons license. We collected, analyzed and tested 16 different open source software tools in terms of their use on selected logistics cases (intermediate storage) of 6 different dimensions (process management, visualization, spatial planning, decision making, forecasting and statistics analyzing). The continuation of the project – the book being translated into foreign languages and encouraging the public to co-creation of the so called 'living' book – is the desire of all authors. With the survey and its results we confirmed that willingness to co-create and spread the idea of free software is huge.

Keywords: open source, free software tools, logistics, Planner, Dia, Zint, Google Earth, Asdn, Quantum GIS, Qcad, Simple Warehouse Mapper, Petersen, Lindo, GnuCash, GSS World, Scilab, OpenOffice.org, PSPP, Dexi.

1 UVOD

Informacijska podpora s posameznimi poslovnimi aplikacijami je temelj uspešnega vodenja poslovnih procesov podjetja. V veliki meri se uporabljajo orodja iz zbirke plačljivih programskih orodij, uporabniki le-teh pa se mnogokrat ne zavedajo prednosti, ki jo ponuja uporaba prosto dostopnih programskih orodij.

V okviru laboratorija za informatiko na Fakulteti za logistiko Univerze v Mariboru je izšla knjiga, v kateri opisujemo uporabo prosto dostopne programske

rešitve. Knjiga je namenjena inženirjem, študentom, predavateljem in drugim, ki se srečujejo s potrebo po reševanju predvsem logističnih, vendar v splošnem inženirskih izzivov. Z njo želimo pokazati, kako si je mogoče z uporabo prosto dostopne programske opreme učinkovito olajšati delo. Gre za pristop, v katerega tisoči iz t. i. »skupnosti« vlagajo najboljše, kar premorejo. Prispevajo predvsem zaradi svojih prepričanj, delo pa največkrat opravijo v okviru svojega prostega časa. Bolj kot kakršen koli zaslužek jih

motivira priložnost, da uresničijo svoje zamisli. Prav zaradi tega so rešitve posebne – v sebi imajo zrna genialnosti.

Eden izmed bistvenih ciljev izvedenega projekta je osveščanje in večja uporaba odprtokodnih programskih orodij. Zavedamo se potreb po povezanosti logističnega znanja z informacijsko tehnologijo, zato spodbujamo k raziskovanju in reševanju praktičnih problemov, z namenom dosega širšega pogleda na določeno problematiko.

2 IZZIV ODPRTE KODE

Odprta koda je že dolgo zakoreninjena v hekerski etiki. Raymond [1] kot hekerja opredeli vsako osebo, ki uživa v raziskovanju programskih sistemov. Posamezniki, ki so sestavljali MIT's Artificial Intelligence Laboratory (raziskovalni laboratorij), so bili obsedeni z raziskovanjem programskih sistemov. Leta 1961 so pridobili prvi PDP-1¹ »mini« računalnik, ki so ga uporabljali za reševanje različnih problemov, znanstvene raziskave, matematične formulacije ipd. MIT-ovi hekerji so s pomočjo računalnika razvili programsko opremo, ki so jo prosto razdelili drugim razvijalcem PDP. Sodelujoči v laboratoriju so se pridružili še drugim pomembnejšim laboratorijem – združeni so postali uspešno središče razvoja programske opreme.

Stallman, ustanovitelj in vodja organizacije Free Software, ki spodbuja razvoj in uporabo prostega programiranja, je leta 1983 začel z izvajanjem projekta GNU – svoboda prostega programiranja, na stotine drugih programerjev pa je začelo ustvarjati nove, brezplačne različice vseh večjih programov operacijskega sistema Unix.² Linus Torvalds je leta 1994 izdal prvo verzijo Linux 1.0, izdano pod licenco GNU. Linux je »klon« operacijskega sistema Unix, ki temelji na orodjih GNU, številnih drugih aplikacijah in drugih paketih programske opreme. V svetu so poznane številne distribucije Linuxa. Ena izmed najbolj znanih je Ubuntu Linux. Ubuntu skupnost dovoljuje brezplačen dostop do vse programske opreme, uporabnost programskih orodij v lokalnem jeziku in svobodo za prilagajanje programske opreme na kakšen koli ustrezen način [2].

¹ PDP-1 je bil prvi računalnik, izdelan v seriji Digital Equipment Corporation's PDP. Poznan je kot najpomembnejši računalnik pri ustvarjanju hekerske kulture.

² Unix je operacijski sistem, prvotno razvit leta 1969 v skupini računalniških zanesenjakov iz podjetja Bell Labs.

Pojem odprte kode je bil v veliki meri sprejet zaradi dvoumnosti brezplačne programske opreme. Za formalizacijo in natančno določitev razdelitev licenčnih pogojev so se razvile večlicenčne pogodbe. Razvoj odprte kode se začne kot marketinška kampanja za prosto programiranje. Pojem prosto programiranje ne pomeni »brezplačno« v finančnem pomenu, ampak se nanaša na »prostost uporabnika« pri uporabi programske opreme. Odprta koda dovoljuje vsakršno uporabo in spreminjanje programske opreme ter njeno razširjanje v spremenjeni ali nespremenjeni obliki in tako omogoča, da programska oprema postaja kakovostnejša in zanesljivejša. Njena izvorna koda je dostopna vsakomur. Velik del internetne infrastrukture vključuje odprtokodno programsko opremo, zato se ne smemo čuditi, da je razvoj odprte kode sovpadel z internetno rastjo. Njeni dosežki so v relativno kratkem času doprinesli k številnim pomembnim napredkom. Še vedno ima veliko možnosti izboljšav predvsem na področju promocije, razširjanja in izobraževanja med končnimi uporabniki.

3 IZVEDBA PROJEKTA

Začetna faza

Eden izmed prvih pomembnih korakov izvedbe projekta – uporaba odprtokodnih in prosto dostopnih programskih orodij – je bilo vprašanje, komu je projekt pravzaprav namenjen: samo študentom ali tudi strokovnjakom. Dvom se je v trenutku razblinil, ko smo spoznali, da v praksi obstaja premalo tovrstne strokovne literature, ki bi imela ne le teoretično, temveč tudi praktično veljavo. Projekt smo začeli z upanjem, da izdelamo oprijemljivo literaturo, ki bo kot pripomoček služila inženirjem, študentom, predavateljem in drugim, ki se srečujejo s potrebo po reševanju predvsem logističnih, vendar v splošnem inženirskih izzivov. Nikakor ne moremo mimo informacijskih znanj, ki so pomemben dejavnik pri izvedbi projekta. Informatika in logistika sta namreč nerazdružljiva partnerja do uspeha, kar potrjuje dejstvo, da je dandanes izvedba logističnih procesov neizvedljiva brez ustrezne informacijske podpore. Inženirji logistike si svojega dela ne morejo več predstavljati brez uporabe sodobnih informacijskih sistemov in orodij, s katerimi lahko uspešneje razrešijo marsikateri logistični problem. Vsakodnevno planiranje, organizacija, kontroliranje, odločanje

ipd. so ključne aktivnosti, katerih uporaba ustreznih orodij v pravem času je bistvenega pomena. Presoditi je treba, kdaj in zakaj je neko programsko orodje primerno za izbrani problem.

V času krize in nenadnih sprememb si ne moremo privoščiti napak, ki bi ogrozile in morda povzročile zastoj ali propad poslovanja. Potrebna je nenehna optimizacija logističnih procesov, minimizacija stroškov, nenehno predvidevanje, optimalno planiranje ipd. Vse to je mogoče doseči z uporabo prosto dostopnih in odprtokodnih orodij. Tudi v težavnejšem, poslovnem svetu se je uporaba tovrstnih orodij izkazala kot zelo učinkovita. Po besedah Kositra [4] je odprta koda prihodnost informacijske družbe. Ponuja obilo prednosti, predvsem pa priložnosti, saj ljudem omogoča dostop do informacijskih tehnologij ter sodelovanje in uspeh v informacijski družbi.

V okviru projekta o prosto dostopnih programskih orodjih je v laboratoriju za informatiko na Fakulteti za logistiko konec leta 2010 nastala knjiga *Programi za logistike*, objavljena pod licenco Creative Commons – »priznanje avtorstva« – »nekomercialno« – »brez predelav« (verzija 2.5 in več). Licenca uporabnikom dovoljuje reproduciranje, distribuiranje, dajanje v najem in priobčitev javnosti izvirnega (torej brez predelave) avtorskega dela pod pogojem, da navedejo avtorja in da ne gre za komercialno uporabo. V celoti in po posameznih poglavjih je dostopna na spletnem naslovu <http://labinf.fl.uni-mb.si/p4l>. Knjiga je napisana v jeziku latex, ki s svojo kakovostjo in zanesljivostjo bistveno olajša nadaljnje delo.

Pri opisovanju se avtorji opirajo na poenostavljen logistični primer upravljanja vmesnega skladišča pri proizvodnji avtomobilov. Primer, ki ga sicer uporabljamo tudi pri študijskem procesu, opisuje skladišče z avtomobilskimi pnevmatikami, platišči in vijaki. Skladišče polnimo s komponentami treh dobaviteljev, vsak uporablja svoje transportno sredstvo (vlak, tovornjak s prikolico in manjše dostavno vozilo) s svojimi specifikami, med katerimi je zelo pomembna časovna dimenzija dobav. Praznimo ga v relativno rednih časovnih intervalih, z viličarji, ki zagotavljajo nemoteno oskrbo z avtomobilskimi deli v proizvodnji. Skladišče ima omejeno zmogljivost. Za podjetje pomenijo skladiščeni avtomobilski deli vezan kapital, ki se s časom dinamično spreminja. Količine se nikoli ne smejo spustiti pod minimalne vrednosti, saj bi bila v tem primeru ogrožena vsa proizvodnja.

Prednosti in pomanjkljivosti

Problema obvladovanja logističnih tokov se lahko inženirji in podjetja lotijo različno, z različnimi pristopi in znanji. Uporaba prosto dostopnih programskih orodij in njihova integracija v poslovne procese omogoča sistematično reševanje izbranega logističnega problema. Prednost tovrstnih orodij je v njihovi dostopnosti, enostavnosti, funkcionalnosti in uporabi na različnih logističnih sistemih in primerih. Z njihovo uporabo lahko bistveno izboljšamo obvladovanje notranjih in zunanjih logističnih tokov.

Vsak izvedeni projekt ima svoje prednosti in pomanjkljivosti. Ena izmed ključnih posebnosti projekta je, da ni namenjen trženju finančnih sredstev, temveč prosti uporabi na trgu. Dva izmed bistvenih ciljev sta podrobnejša osredinjenost na posamezna področja in pogostejša uporaba prosto dostopnih in odprtokodnih programskih orodij v praksi. Že od vsega začetka je glavna predpostavka knjige njeno nenehno izboljševanje, zato vabimo vse, ki imajo predloge v zvezi s knjigo, da pišejo na spletni naslov programizalogistike@gmail.com. Vse predloge bomo skrbno proučili in uporabna spoznanja vnesli v naslednje verzije. Posredovalci popravkov se s pošiljanjem predlogov strinjajo z naslednjo izjavo: »Če bo uredniški svet publikacije upošteval moje predloge sprememb publikacije in bodo te dodane v novejšo verzijo publikacije, se odpovedujem vsem materialnim avtorskim pravicam, ki izhajajo iz mojega avtorskega dela in se strinjam z objavo mojega imena med avtorji publikacije.«

Pomanjkljivost, ki jo trenutno vidimo v obravnavani tematiki, je premajhna osveščenost ljudi o tovrstnih programskih orodjih. Še vedno obstajajo negativni predsodki o kakovosti, zanesljivosti in varnosti tovrstnih orodij.

Uporabniki programskih orodij bodo prostodostopna programska orodja uporabili kot pripomoček pri reševanju problemov v praksi. Poudariti je treba, da so tovrstna programska orodja dostopna na spletu, ki jih lahko preprosto prenesemo in uporabljamo. Brezplačne programske rešitve so v praksi specializirane predvsem za določena področja in ne zajemajo širših področij kot plačljive, zato je izvedeni projekt povezovalni člen med različnimi programskimi orodji za reševanje posameznih segmentov problema, s čimer jih približamo plačljivim.

Večfunkcionalnost

Poudariti je treba večfunkcionalnost in praktičnost orodij. Mnogo let so ta dostopna na spletnih straneh, vendar so kljub temu mnogim nepoznana. Pogosto potrebujemo orodje pri reševanju študijskih nalog, za delo na projektu, vendar so plačilni pogoji mnogim nedosegljivi. Opisana programska orodja pa so dostopna vsakomur. Če je program napisan samo za eno okolje in brez odprtokodne licence, izgubi pomen odprtokodnega programa v točki delovanja na različnih platformah (npr. Windows ali Linux) in dostopa do izvorne kode. Za programska orodja, ki prvotno ne delujejo v vseh okoljih, obstaja rešitev – program Wine. Z njegovo uporabo vsa izbrana programska orodja, napisana za okolje Windows delujejo tudi v Ubuntu. Je program, ki uporabnikom operacijskih sistemov Linux ponuja kompatibilnost delovanja programskega orodja, napisanega za okolje Windows.

4 REZULTATI

Uporabljena programska orodja

V knjigi opisujemo šestnajst programov, vezanih na opisani primer. Ti predstavljajo šest različnih dimen-

zij problematike upravljanja komponent vmesnega skladišča. Vsako poglavje je mogoče prebrati neodvisno. Obdelali smo področja upravljanja procesov (projektno vodenje, diagrami poteka, sledljivost izdelkov), vizualizacije (vizualizacija oskrbne verige, vizualizacija geografskega področja in uporaba različnih prostorskih slojev GIS), prostorskega planiranja, odločanja (na podlagi iskanja optimalnih vrednosti, večparameterskega odločitvenega modeliranja, simulacij in finančnih tokov), napovedovanja verjetnega razvoja ter navsezadnje statistične analize.

Vsako poglavje na kratko opisuje teoretično ozadje posamezne dimenzije problema, katerega upravljanje podpremo z ustreznim orodjem. Nadaljujemo z opisom problema skozi optiko obravnavane dimenzije. Sledi kratek opis programskega orodja, njegov primer pa je vezan na primer vmesnega skladišča. Na koncu vsakega poglavja je dodan kratek povzetek napisanega in seznam uporabljenih virov. Pred vsemi poglavji je uvod s kratkim opisom prosto dostopnih programskih orodij in primera, ki predstavlja rdečo nit knjige. Knjigo končuje daljša tabela s prevodi, uporabljenimi v posameznih poglavjih, in dodatki, s primeri programov in izpisi programskih orodij.

Tabela 1: **Predstavljena programska orodja v knjigi *Programi za logistike***

Programska orodja	
Planner – projektno vodenje	Petersen – optimalni izračuni
Dia – diagrami poteka – vizualizacija procesov	Lindo – numerične operacije
Zint – implementacija črtne kode v poslovanju izbranega podjetja	Dexi – večparametrski odločitveni modeli
ASDN – integracija oskrbne verige, oblikovanje industrijskih logističnih mrež	GnuCash – vodenje financ
Google Zemlja – načrtovanje poti z digitalnim zemljevidom	GPSS World – simulacijsko orodje
Quantum GIS – analiza geoprostorskih podatkov	Scilab – napovedovanje
Qcad – 2D vizualizacija prostorov in procesov	OpenOffice.org Preglednica – alternativa Microsoft Excel
Simple Warehouse Mapper – 3D simulacijsko programsko orodje za vizualizacijo	PSPP – analiza podatkov

Prvo programsko orodje, ki smo ga podrobneje opredelili in podkrepili s primerom, je Planner, ki je v pomoč pri projektne vodenju. Opredeljuje prikaz procesa načrtovanja, nadziranja in poročanja aktivnosti v podjetju. V nadaljevanju uporabimo programsko orodje Dia, s katerim izdelamo diagram poteka, ki prikazuje prehode iz ene aktivnosti na drugo (naročanje, dostava, izmenjava dokumentov, grobi in fini prevzem platišč, reklamacije in skladiščenje). Obvladovanje oskrbne verige in zagotovitev

sledljivosti komponent na vhodni strani sistema in izdelkov na izhodni strani sistema, predvsem v procesu transporta in skladiščenja, zahteva poznavanje in uporabo standardov označevanja in identifikacije logističnih enot, kar omogoča programsko orodje Zint, s katerim prikažemo možnosti implementacije črtne kode v poslovanju izbranega podjetja.

S programskih orodjem ASDN predstavimo integracijo oskrbne verige – od dobavitelja komponent do izbranega podjetja, saj je združevanje posameznih

segmentov nadvse pomembno. V vsakdanjih situacijah se srečujemo s problemom mobilnosti in pomanjkanjem časa. Pogosto se zgodi, da se nenadoma odločimo za poslovno potovanje v tujino, čeprav niti ne vemo, kako bomo prispeli tja. V teh primerih je lahko v pomoč programsko orodje Google Zemlja, s katerim prikažemo načrtovanje poti z digitalnim zemljevidom. V nadaljevanju s programskim orodjem Quantum GIS analiziramo geoprostorske podatke in vizualno opredelimo primernost lokacije za izgradnjo skladiščnega objekta z upoštevanjem tveganj.

Področje prostorskega planiranja vključuje tri programska orodja. Orodja Qcad pripomorejo k vizualizaciji prostorov ali procesov, s pomočjo katerih prikažemo preprost model ureditve in razporeditve skladiščnega prostora. Uporabimo še 3D simulacijsko programsko orodje Simple Warehouse Mapper, s katerim preprosto uredimo in razporedimo skladiščni prostor.

Odločitve so pomembno področje na vseh področjih poslovanja. S pomočjo programskega orodja Petersen prikažemo izračun optimalne dostave komponent od dobavitelja do izmišljenega podjetja Open Storage. Zaradi velikega števila numeričnih operacij s programskim orodjem Lindo predstavimo problem izbire transporta platišč, ki jih je treba dostaviti v določenem roku. Z uporabo programskega orodja Dexi predstavimo odločitveni model, s katerim določimo optimalni izbor nakupa viličarja. Pri poslovanju nikakor ne moremo mimo financ. Programsko orodje GnuCash uporabimo za vodenje računa pri investicijah nakupa sodobne infrastrukture. Z uporabo GPSS World simulacijskega orodja je mogoče predvideti učinke različnih variant modeliranja in izbranega realnega kompleksnega sistema.

Z napovedovanjem ocenjujemo verjetne razvoje v prihodnosti. Z orodjem Scilab izvedemo napoved prodaje avtomobilov za petletno obdobje. Uporabimo še programsko orodje Open.Office.org Preglednica, ki nas na prvi pogled spominja na Microsoft Excel, s katerim vodimo stanje zalog komponent. Na koncu dodamo še programsko orodje PSPP, s katerim izvedemo analizo podatkov, zbranih s pomočjo anketnega vprašalnika.

Izzivi v prihodnosti

Vsako leto si želimo pripraviti vsaj dve novi ali dopolnjeni izdaji, po potrebi tudi več. Želimo si, da bi knjiga s pomočjo »skupnosti« postala tako dobra, da

bi jo bilo smiselno prevesti tudi v druge jezike in bi ji s tem odprli mnogo večje tržišče, kar bi imelo za posledico tudi večjo verjetnost posredovanja predlogov za izboljšavo. Koncept knjige temelji na ideji »žive« knjige, ki zahteva neprestano dopolnjevanje in spreminjanje vsebine.

Januarja smo izdali prenovljeno verzijo knjige, v katero smo dodali novo vsebino, programsko orodje WordPress, namenjeno oblikovanju spletnih vsebin za potrebe organizacije. Prav tako je v pripravi vsebina o tehnologiji GPS oz. GNSS (TangoGPS, GPS Analyzer), s katero bomo prikazali številne možnosti njene uporabe na področju logistike in logističnih sistemov, temelječe na prosto dostopnih orodjih s ciljem zagotovitve sledljivosti, monitoringa in morebitnih nadaljnjih optimizacijskih procesov.

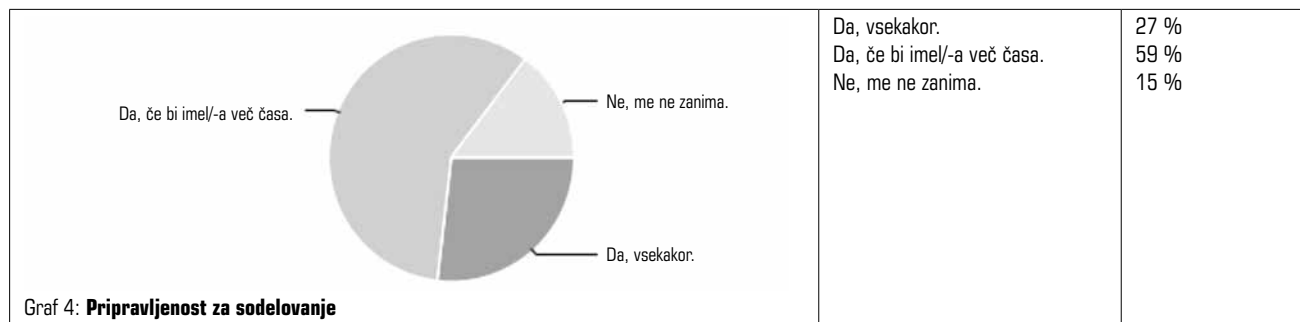
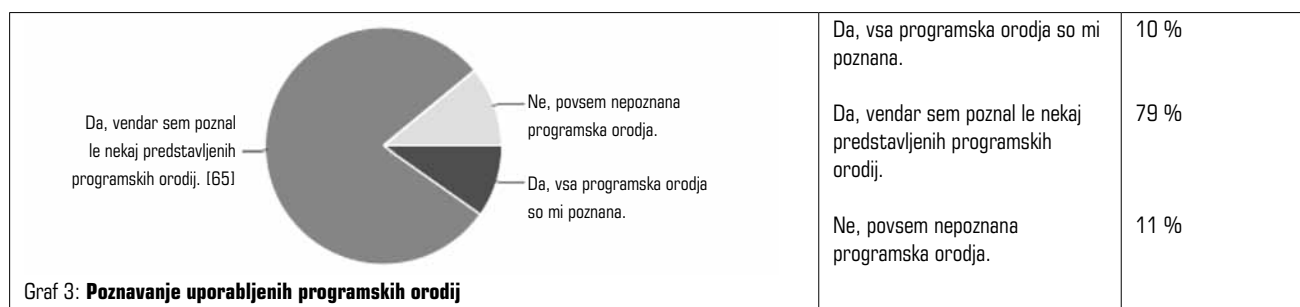
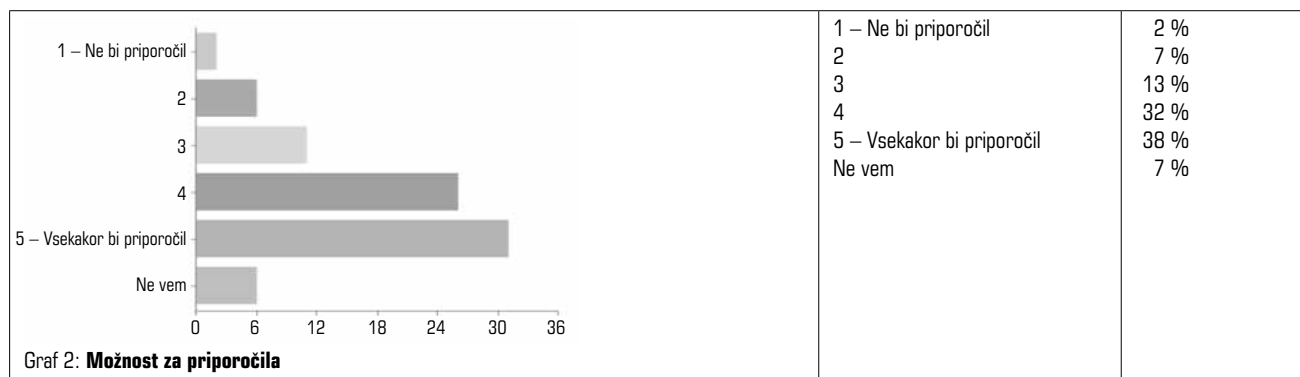
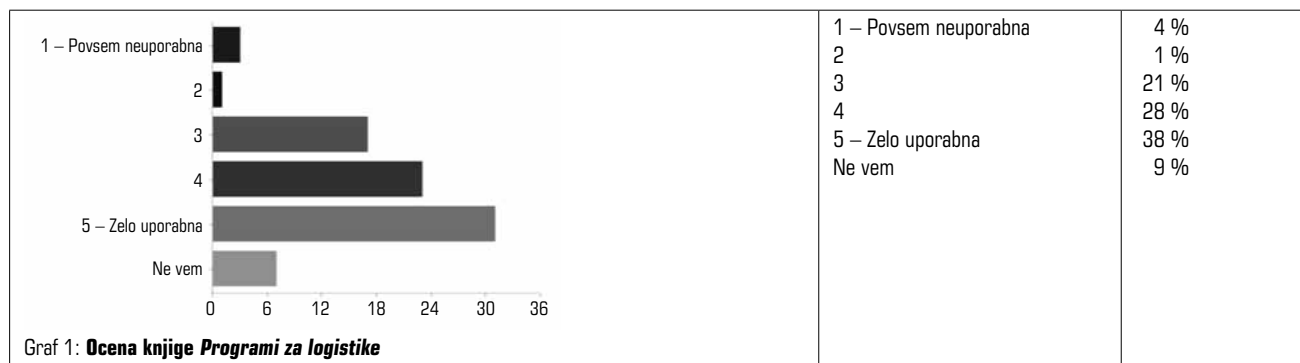
Mnenja uporabnikov (anketni vprašalnik o uporabnosti knjige *Programi za logistike*)

Na podlagi anketnega vprašalnika smo med študenti in pedagoškim osebjem Univerze v Mariboru izvedli kvantitativno raziskavo, katere namen je bil ugotoviti uporabnost knjige *Programi za logistike* in osnovne demografske značilnosti uporabnikov. Raziskava naj bi tako pojasnila stališča z vidika uporabnosti knjige glede na nekatere demografske lastnosti uporabnikov.

Rezultati

Analiza razkriva uporabnost knjige in pripravljenost anketiranih za sodelovanje pri nadaljnjem razvoju knjige. Odgovarja na vprašanja: 1. Kako ocenjujete uporabnost knjige *Programi za logistike* za vaše potrebe (služba, študij ipd.)? 2. V kolikšni meri bi knjigo *Programi za logistike* priporočili svojim prijateljem, znancem, sorodnikom? 3. Ali so vam bila (v povprečju) predstavljena programska orodja v knjigi že poznana? 4. Ali bi bili pripravljeni prispevati in razširjati tovrstna odprtokodna – prosto dostopna programska orodja?

Kar 66 odstotkov anketiranih ocenjuje uporabnost knjige s 5 oz. 4 (po lestvici 1 – povsem neuporabna, 5 – zelo uporabna), 4 odstotki anketiranih jo ocenjujejo kot povsem neuporabno. 70 odstotkov anketiranih bi knjigo priporočilo prijateljem, znancem, sorodnikom, medtem ko se le 9 odstotkov vprašanih ne nagiba k priporočilu. Zanimivo je, da kar 79 odstotkov anketiranih pozna vsaj nekaj predstavljenih programskih orodij, 86 odstotkov pa bi jih bilo pripravljeno prispevati in razširjati tovrstna odprtokodna – prosto dostopna programska orodja.



Anonimna anketa je potekala februarja 2011 med 82 naključnimi uporabniki z Univerze v Mariboru. Anketirani so bili k izpolnjevanju ankete povabljeni osebno: posameznikom smo poslali e-poštna sporočila s pojasnitvijo namena našega pristopa in povezavo na anketni vprašalnik. Vprašani so odgovarjali v posredni interakciji z izvajalcem testiranja prek Google dokumentov. Kvantitativna študija meri te demografske lastnosti uporabnikov: spol, starost, dokončana izobrazba in status.

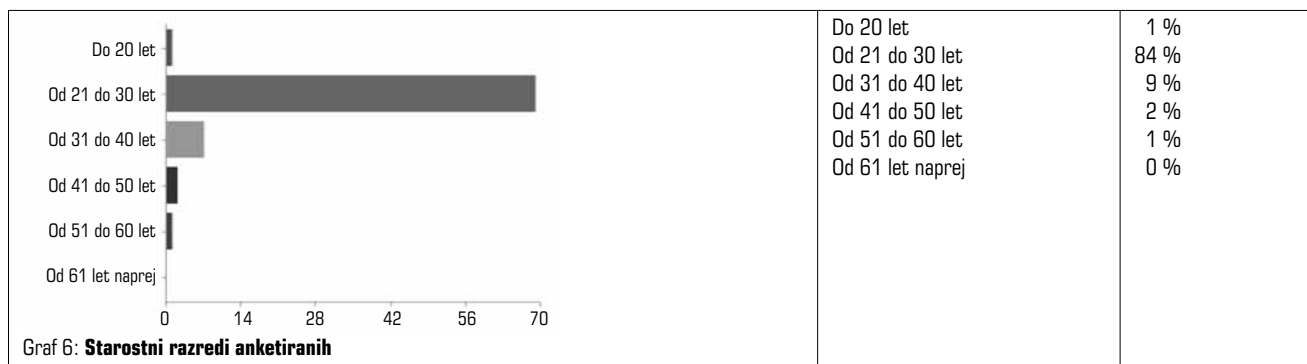
Pri sklepanju smo morali biti previdni. Ker smo prek e-pošte dejavno pozivali anketirance, se raziskava nanaša na selektivno vzorčenje. Izbira lokacije ni bila naključna, saj smo se usmerili na študente in pedagoško osebje Univerze v Mariboru. Skušali smo vključiti čim več različnih fakultet, vendar smo izpustili tiste, za katere smo bili mnenja, da jih knjiga ne bi zanimala. Kljub temu lahko iz zbranih podatkov izberemo nekaj zanimivih sklepov.

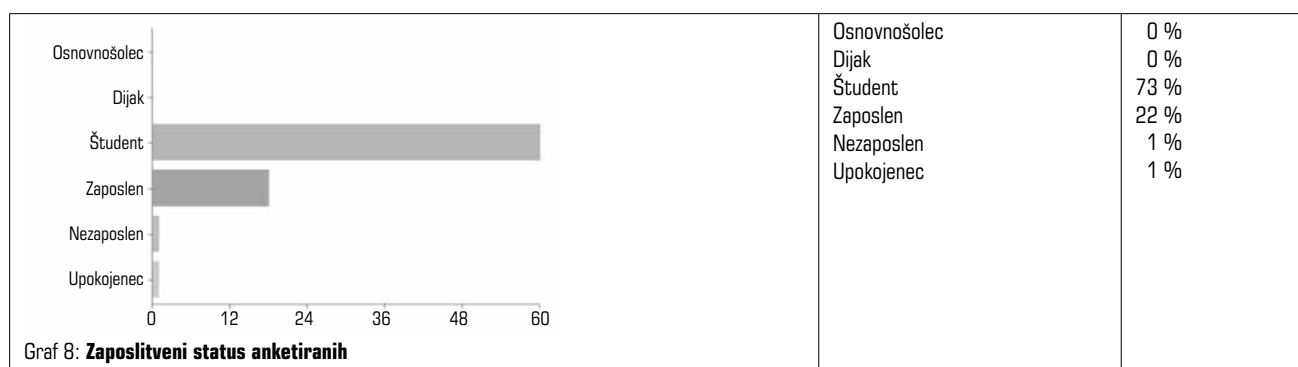
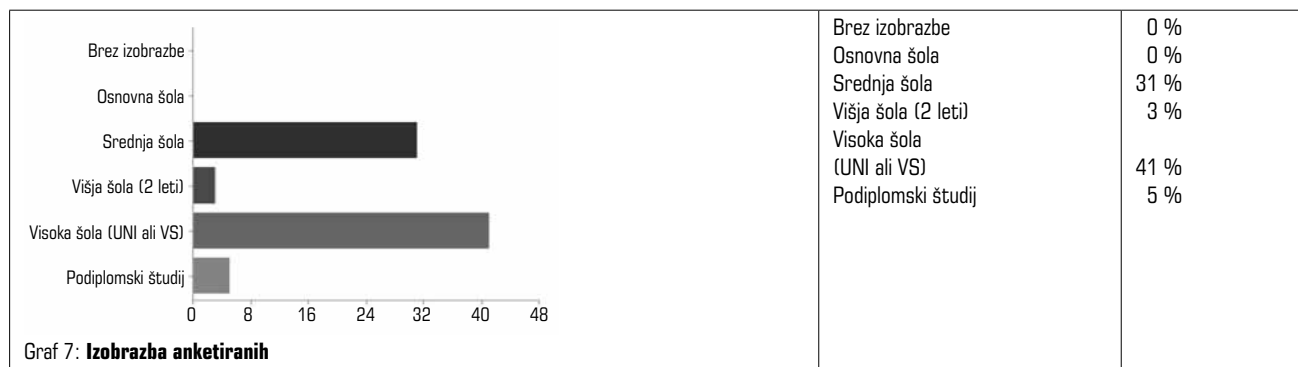
Med izvajanjem ankete lahko vpliva na veljavnost sklepov kar nekaj vidikov. Predvidevamo, da so anketiranci skušali odgovarjati z minimalnim trudom,

saj vprašalnik vsebuje le vprašanja zaprtega tipa. Alternative, med katerimi so lahko izbirali, so bile vedno navedene v istem vrstnem redu. Kot rezultat obstaja tveganje zaporednega efekta: alternative, ki so navedene kot prve, so izbrane pogosteje kot tiste pod njimi. Pri sklepanju bi tako bilo treba pretehtati mogoče posledice tega učinka in preveriti, ali lahko vplivajo na alternativno razlago rezultatov.

Demografski kazalniki

V raziskavi je sodelovalo 82 ljudi oz. 52 žensk (63 %) in 28 moških (34 %); 2 anketirana podatkov o spolu nista navedla. Anketirani so najpogosteje študentje (73 %), imajo doseženo največ visoko šolo (univerzitetni ali visokošolski program) (50 %). Najbolj zastopana starostna skupina je med 21 in 30 leti. Rezultati zaradi majhnega vzorca niso povsem nezanesljivi, vendar so okvirno podali oceno in mnenje izvedenega projekta. Mogoče je, da so si anketiranci kakšno vprašanje razlagali povsem drugače, kot smo si ga zamislili. Pomembno je tudi, koliko so bili anketiranci dejansko pripravljene sodelovati in da ankete niso vzeli le »za šalo«.





5 SKLEP

Uporabnost odprtokodnih in prosto dostopnih programskih orodij se je izkazala tudi v težavnejšem, poslovnem svetu kot odlična informacijska podpora pri odločanju, planiranju, nadzorovanju. Številna programska orodja z oznako odprta koda so v prvi vrsti odlična informacijska podpora logističnim procesom. Prednost orodij je njihova preprostost in brezplačnost, slabost pa omejene možnosti uporabe. Kanček iznajdljivosti s kombinacijo različnih odprtokodnih in prosto dostopnih programskih orodij kar nazorno predstavlja knjiga *Programi za logistike*. Uporabnost knjige smo potrdili še z anketo, ki smo jo izvedli med pedagoškimi delavci in študenti Univerze v Maribo-

ru. Njihova pripravljenost pri soustvarjanju »žive« knjige pa vsekakor daje pozitivne obete za nadaljnje delo. Verjamemo, da se je večmesečno delo izplačalo.

6 VIRI IN LITERATURA

- [1] JOHNSON, Kim: Open-Source Software Development, Department of Computer Science. University of Calgary Alberta, Canada T2N 1N4. Objavljeno na spletni strani: <http://chinese-school.netfirms.com/computer-article-open-source.html>.
- [2] DRUMMOND, John G.: Open Source Software and Documents: A Literature and Online Resource Review, 2000.
- [3] KOSITER, Andrej: Odprtokodne rešitve so tudi v najzahtevnejših podjetniških okoljih, *Finance*, 25. november 2010, št. 292, str. 22–23.
- [4] JEREJ, Borut, SKOK, Dejan, ŠAFRAN, Mirica, ŠKORNIK, Mateja: *Programi za logistike*, Fakulteta za logistiko, Univerza v Mariboru, 2010.

Mirica Šafran je absolventka podiplomskega študija Logistika sistemov na Fakulteti za logistiko Univerze v Mariboru, na kateri je diplomirala leta 2008 z diplomsko nalogo Zavarovanje poštnih storitev. S člani laboratorija za informatiko se ukvarja z raziskovanjem in uporabo odprtokodne in proste programske opreme za logistike, pripravo strokovnih gradiv ipd. Raziskovalno dejavnost krepi v okviru Inštituta za logistiko, delujočega na Fakulteti za logistiko Univerze v Mariboru. Med njene uspehe prištevamo priznanje Fakultete za logistiko na področju raziskovalne dejavnosti (projekt Optimizacija transporta bioloških vzorcev v SBC, izdelan v okviru skupine LARS) ter pohvalo Tutor študent 2009. Prejela je pohvalo podjetja Mercator za sodelovanje na projektu, izvedenem v okviru skupine LARS.

■

Mateja Škornik je leta 2009 s področja interaktivnega grafičnega oblikovanja diplomirala na Fakulteti za elektrotehniko, računalništvo in informatiko Univerze v Mariboru. Med študijem je uspešno službovala v podjetju Cetis, d. d., kjer si je nabrala dragocene izkušnje. Je asistentka pri predmetih s področja informatike na Fakulteti za komercialne in poslovne vede in doktorska študentka na Fakulteti za logistiko, na kateri pripravlja doktorsko disertacijo s področja upravljanja tveganj. Na raziskovalnem področju se ukvarja s standardizacijo, zakonodajo, informacijskimi tveganji, tveganji v logistiki, odprto kodo, pripravo strokovnih gradiv ipd.

■

Dejan Skok je podiplomski študent na Fakulteti za logistiko Univerze v Mariboru, na kateri je z odliko diplomiral z diplomsko nalogo Zasnova sledenja blaga v podjetju Bosio, d. o. o. V tem podjetju je uspešno služboval in si nabiral dragocene izkušnje. V laboratoriju za informatiko Fakulteti za logistiko Univerze v Mariboru se ukvarja z raziskovanjem na področjih logističnega pristopa k obvladovanju masovnih dogodkov (simulacije, množična strežba, logistični tokovi, sledenje ipd.), sledenja logističnih objektov (GPS), priprave strokovnih gradiv. Med njegove uspehe lahko prištejemo različna priznanja in dosežke: Eureka Ideje!mladih 2009 – zlata plaketa in tretje mesto, najboljši diplomant v občini Štore leta 2009, najboljši študent prvega letnika magistrskega študija 2009/2010 na Fakulteti za logistiko Univerze v Mariboru.

■

Borut Jereb je predavatelj na Fakulteti za logistiko Univerze v Mariboru. Leta 1991 je uspešno zagovarjal doktorat s področja računalniških znanosti na Univerzi v Ljubljani. Od leta 1991 do 1992 je kot vabljeni profesor raziskoval in poučeval na Oregon State University. Po vrnitvi v Slovenijo si je skoraj dve desetletji kot svetovalec in kot vodja v podjetjih in v javnem sektorju pridobil veliko praktičnih izkušenj na področju informatike. Trenutno se ukvarja predvsem s področjem dokumentnih sistemov, standardizacije, zakonodaje in upravljanja tveganj v informatiki.

Poročilo o konferenci DSI 2011

Aprila 2011 je v Portorožu potekala že 18. konferenca Dnevi slovenske informatike – letos pod geslom »Nove razmere in priložnosti v informatiki kot posledica družbenih sprememb«.

Na slovesni otvoritvi konference je udeležence pozdravil predsednik Slovenskega društva Informatika Niko Schlamberger, ki je v svojem nagovoru izrazil zadovoljstvo, da je konferenca dočakala polnoletnost, in poudaril pomen informatike, ki se kaže tudi v tem, da kriza relativno manj ogroža družbe iz te kot iz drugih dejavnosti.

Slavnostni govornik Samo Hribar Milič, generalni direktor Gospodarske zbornice Slovenije, je v svojem nastopu navedel več podatkov, ki se nanašajo na gospodarska gibanja. Informatiko je ocenil za dejavnost, brez katere gospodarskim družbam ni več moč preživeti.



Samo Hribar Milič, generalni direktor GZS

V sklepnem delu slovesnega začetka konference je bila podeljena **i-nagrada** za najboljši IKT-projekt. Merila so bila stroga in tako glavna nagrada ni bila podeljena, na drugo mesto pa je bil zaradi inovativnosti in uspešnega pristopa do potreb kupcev, manjših podjetij in zasebnikov uvrščen projekt MOX družb Marg Inženiring, d. o. o., Unija računovodski servis, d. d., in Adacta, d. o. o.

Priznanje Slovenskega društva Informatika za delo in dosežke je prejel Tomaž Turk z Ekonomske fakultete Univerze v Ljubljani za razvojne dosežke spletnega terminološkega slovarja Islovar.

Tekmovanja za **najboljši študentski projekt 2011** na letošnji konferenci DSI so se udeležili študentje računalništva in informatike z ljubljanske in mariborske univerze, ki so tudi predstavili svoje rezultate. Za najboljši študentski projekt 2011 je bil izbran projekt 2ndSight avtorjev Luke Topolovca, Žana

Markana, Tineta Poštuvana in Blaža Magdiča s Fakultete za elektrotehniko, računalništvo in informatiko Univerze v Mariboru. Drugo mesto je pripadlo projektu CloudDocs – socialno omrežje za izmenjavo dokumentov avtorja Roberta Dukarića s Fakultete za računalništvo in informatiko Univerze v Ljubljani, tretje mesto pa je zasedel projekt z naslovom Distribucija avtorja Mateja Lakote, prav tako z ljubljanske fakultete.

Udeleženci so na konferenci lahko dobili odgovore na ključne usmeritve informatike ter vpogled v najpomembnejše poslovne in tehnološke trende, ki so jih predstavili vabljeni tuji predavatelji podjetij Oracle, Microsoft, SAP, IBM in JBoss – Red Hat.

Izjemnega odziva sta bili deležni predavanja Jacoba Morgana o tem, kako lahko uporabimo socialna omrežja za generiranje poslovne vrednosti, ter Gartnerjevega analitika Claudia Da Rolda, ki je govoril o tem, ali pomeni izvajanje storitev v oblaku prihodnost storitev informacijske tehnologije.



Jacob Morgan



Claudio Da Rold

Navajamo nekaj vsebinskih poudarkov in ugotovitev konference.

- Na področju poslovne inteligence je ugotovljeno, da so v vseh okoljih neprestane potrebe po poročanju, na oblikovanje rešitev pa vplivajo predvsem dane možnosti in zahteve posameznih okolij. Razvoj in znanstvene raziskave odpirajo in ponujajo še nove možnosti, katerih uveljavitev bomo lahko spremljali v naslednjih letih.
- Na področju menedžmenta poslovnih procesov je bilo predstavljenih več prispevkov, ki so obravnavali nekatera teoretična spoznanja s področja procesne usmerjenosti ter njihovo uporabo v praksi. Predstavljenih je bilo nekaj metodoloških izhodišč ter ključnih dejavnikov uspeha pri projektih prenove in informatizacije poslovanja, med

katerimi je večina predavateljev poudarjala pove-zavo s strategijo in nujno potrebo po vključenosti najvišjega vodstva organizacij.

- Na področju informacijske varnosti in upravljanja tveganj je bilo ugotovljeno, da se uporabniki informacijskih sistemov prepočasi prilagajajo novim razmeram in priložnostim v informatiki kot posledici družbenih sprememb, saj slepo sledijo trendom rabe socialnih omrežij, ne da bi se pri tem zavedali vseh nevarnosti, ki jih le-ta prinašajo. Enako velja za rešitve v oblaku, pri katerih uporabniki nimajo zadostnega nadzora nad svojimi podatki. S tem se povečuje nevarnost za zlorabe, ki jih je vedno več in so vedno bolj ciljane na posameznika. Posameznik se ne zaveda dovolj nevarnosti, ki prežijo nanj, ali pa jih celo zavestno prezira. Organi pregona dobijo premalo prijav zlorab, malo jih je uspešno preiskanih, še manj pa je tistih, ki so obsojeni. Za ustrezno sistemsko rešitev problema bi bilo treba osveščati uporabnike, da bi ločili med dovoljenim in prepovedanim, ustrezno zavarovali svoje podatke ter prijavljali zlorabe organom pregona.
- Na področju podpore odločanju in operacijskih raziskav so bile predstavljene praktične rešitve, aplikacije večkriterialnega in skupinskega odločanja na področju kmetijstva in prehrane. Cilji in odločevalci so pogosto protislovni in tako z modeli za podporo odločanju želimo najti kompromis med cilji in odločevalci. Globalno želimo doseči učinkovitejšo izrabo virov, kakovosti dela in oblikovati okolje, ki je sprejemljivo za družbo kot celoto. Prikazane so bile izkušnje in spoznanja ter smeri razvoja modelov ob upoštevanju negotovosti odločanja. Izpostavljene so bile prednosti, slabosti in priložnosti uporabe modelov, seveda ob podpori informatike na številnih področjih, v sektorjih gospodarstva in ob problemih. Predstavljene so bile informacijske, optimizacijske in zlasti tehnološke rešitve, ki služijo kot podpora odločanju na področjih računalniške strežbe, kakovosti podatkov, strojništva in procesnih tehnologij.
- Na področju informatike v javni upravi je ugotovljeno, da so ponovno začeli vzpostavljati skupna organizacijska telesa, katerih neobstoj je bil zaradi preteklih reorganizacij pogosto ključna težava pri zagotavljanju transparentnosti izvajanja projektov in procesov sodelovanja. Kljub temu

prihaja do uspešnih projektov sodelovanja, kar je bilo predstavljeno na primeru rešitve za podporo uporabnikom, pri kateri sta združili vsebine obstoječih rešitev ministrstvi za javno upravo in za pravosodje ter tako oblikovali skupno rešitev za podporo uporabnikom z bistveno nižjimi stroški investicije kot v primeru samostojnega razvoja. Še vedno pa zaradi sektorske organiziranosti informacijske in komunikacijske tehnologije ostaja veliko težav pri izvajanju projektov s tega področja, predvsem na ravneh pripravljanja ter spremljanja izvedbe. Zaradi tega so zelo dobrodošli periodične strategije in akcijski načrti, ki zagotavljajo vsaj nekaj pregleda nad dogajanja na področju informacijske in komunikacijske tehnologije. Rdeča nit sekcije je bilo poenotenje delovanja služb informacijske in komunikacijske tehnologije, potreba po skupnem usklajevanju ter informiranju o tekočih in prihodnjih projektih ter zagotavljanje možnosti večkratne uporabe že vzpostavljenih rešitev. Izpostavljena je bila potreba po primerno organiziranem sodelovanju v okolju slovenske državne in javne uprave. Iz prispevkov je bil viden tudi pomen inovativne uporabe razpoložljivega programja in računalniških naprav. Izpostavili bi lahko pametna mesta, tehnologijo RFID uporabo informacijskotehnoloških rešitev v kulturi in znanju ter statistične rešitve, ki pomenijo predvsem prihranek časa in finančnih sredstev.

- Na okrogli mizi »Nove razmere v informatiki: ali so storitve informacijske tehnologije v Sloveniji dovolj cenjene?« so sodelujoči najprej ocenili stanje v Sloveniji na področju informacijske in komunikacijske tehnologije. Gospodarska in finančna kriza močno vpliva na podjetja s tega področja, saj se zmanjšujejo investicije v informacijske in komunikacijske tehnologije tako v gospodarstvu kot v javni in državni upravi. V gospodarstvu v teh kriznih časih vodstva podjetij in naročnikov še vedno smatrajo investicije v opremo in storitve informacijske in komunikacijske tehnologije le kot strošek, ne pa kot investicijo, in torej možnost povečevanja konkurenčnosti in uspešnosti njihovih podjetij. Posebno poglavje sta državna in javna uprava, v katerih se v zadnjih letih zmanjšujejo investicije v informacijske in komunikacijske tehnologije.

Sodelujoči so razpravljali predvsem o rešitvah za izboljšanje kriznih razmer. Razpravljalci so ugo-

tovili, da so nujno potrebne spremembe tako v gospodarstvu in javni upravi kot v podjetjih s področja informacijske in komunikacijske tehnologije. Dolgoročna rešitev je usmerjenost v izvoz rešitev in storitev informacijske in komunikacijske tehnologije, saj je slovenski trg premajhen. Za povečanje izvoza se morajo podjetja povezati in sodelovati, predvsem pa se usmeriti v razvoj in trženje produktov. Neizkoriščena možnost je tudi vzpostavitev javno-zasebnega partnerstva s sodelovanjem uporabnikov in ponudnikov, ki se v tujini uveljavlja, v Sloveniji pa ne znamo izkoristiti priložnosti, ki jih omogoča. Vsi govorniki so soglašali, da kljub kriznim razmeram obstaja dovolj možnosti za nadaljnji razvoj. Treba je le poiskati priložnosti in z agilnim ter inovativnim pristopom uveljavljati rešitve informacijske in komunikacijske tehnologije v gospodarstvu in javni upravi ter v tujini.

Poudarki konference, na kateri je predstavilo svoje prispevke devetih vabljenih vrhunskih tujih plenarnih predavateljev, petdesetih vabljenih slovenskih predavateljev in več kot sto drugih predavateljev in avtorjev prispevkov, so:

1. Že 18. konferenca DSI je s 160 referati prikazala pravo bogastvo vsebin in aktualnih tematik.
2. Rdeča nit konference »Nove razmere in priložnosti v informatiki kot posledica družbenih sprememb« je sporočilo družbam s področja informatike.
3. Informatika v vseh subjektih zasebnega in javnega prava igra ključno vlogo za njihovo boljšo odzivnost in večjo kokurenčnost.
4. Informatiko je treba ustrezno umestiti v vseh organizacijah, če hočejo biti uspešne.

5. Za uspešnost gospodarstva in slovenske informatike je Slovenija postala premajhna, naša prizadevanja morajo biti usmerjena evropsko in globalno.
6. Poslanstvo SDI, ki ga izpolnjujemo na konferenci, je tudi promoviranje strokovne odličnosti.



Utrinek s predavanj

Konferenco DSI 2011 smo uspešno končali. Zahvaljujemo se vsem udeležencem za obisk, avtorjem za bogastvo ter raznolikost vsebin in programskemu svetu, programskemu odboru, organizacijskemu odboru ter pokroviteljem za pomoč pri izvedbi.

Več informacij in utrinkov s konference najdete na spletni strani <http://www.dsi2011.si>.

*Slovensko društvo INFORMATIKA
Programski in organizacijski odbor DSI
Fotografije Janez Strah*

Prireditelj konference

..... slovensko
..... društvo
..... informatika

Organizacija konference



ds



**dnevi
slovenske
informatike**



**„Nove razmere in priložnosti v informatiki
kot posledica družbenih sprememb“**

Kongresni center Grand hotel Bernardin
Portorož | Slovenija | 18. - 20. april 2011

Iz Islovarja

Islovar je spletni terminološki slovar informatike, ki ga ureja jezikovna sekcija Slovenskega društva Informatika na naslovu <http://www.islovar.org>. V tej številki revije objavljamo nekaj novejših izrazov s področja varovanja. Izraze lahko komentirate, tako da se prijavite v poglavju Nov uporabnik, poiščete izraz, ki ga želite komentirati, in zapišete svoj komentar ter predlog spremembe.

béla optimizácija -e -e ž (*angl.* white hat SEO)
optimizacija spletnih strani v skladu s pravili spletnih iskalnikov; *prim.* črna optimizacija, siva optimizacija

béli klobúk -ega -a m (*angl.* white hat)
kdor preizkuša varnost računalniških sistemov z vdiranjem; *prim.* vdiralec

črna optimizácija -e -e ž (*angl.* black hat SEO)
optimizacija spletnih strani s kršenjem pravil spletnih iskalnikov; *prim.* bela optimizacija, siva optimizacija

črni klobúk -ega -a m (*angl.* black hat)
kdor vdira v tuje računalniške sisteme z namenom nepooblaščno uporabljati podatke ali programe; *prim.* vdiralec

eskalácija privilegijev -e -- ž (*angl.* privilege escalation)
nepooblaščna pridobitev večjih privilegijev v operacijskem sistemu, programski opremi

hékanje -a s (*angl.* hacking) *žarg.*
1. spreminjanje strojne in/ali programske opreme na izviren, samosvoj način za doseganje novih, nepredvidenih zmožnosti
2. vdiranje v tuje računalniške sisteme

hékat -am nedov. (*angl.* hack) *žarg.*
1. spreminjati strojno in/ali programsko opremo na izviren, samosvoj način za doseganje novih, nepredvidenih zmožnosti
2. vdirati v tuje računalniške sisteme

héker -ja m (*angl.* hacker)
1. računalniški zanesenjak, ki spreminja strojno in/ali programsko opremo na izviren, samosvoj način za doseganje novih, nepredvidenih zmožnosti
2. *gl.* vdiralec

hékerski málček -ega -čka m (*angl.* script kiddie)
heker (1), ki še nima veliko izkušenj

krája klíkov -e -- ž (*angl.* clickjacking, click-jacking, UI redressing)
krajaj podatkov ali prevzem nadzora nad

brskalnikom preko zlonamerne, zavajajoče spletne strani

kréker -ja m (*angl.* cracker)
kdor zlomi zaščito programske opreme z namenom nepooblaščne uporabe in distribucije; *prim.* razbijalec gesel (1)

kriminalna prográmska opréma -e -e -e ž (*angl.* crimeware)
programska oprema, narejena za izvedbo kriminalnih dejanj v računalniških omrežjih; *prim.* zlonamerno programje

optimizácija splétnih straní -e -- -- ž (*angl.* search engine optimization, krat. SEO)
dejanja, ki izboljšujejo rang strani (1)

ráng straní -a -- m (*angl.* page rank, PageRank, krat. PR)
položaj spletne strani v izpisu zadetkov, ki ga določajo algoritmi spletnih iskalnikov

razbijálec gèsel -lca -- m (*angl.* password cracker)
1. kdor razbija gesla; *prim.* kreker
2. orodje, s katerim se lahko najde ustrezno geslo

SEO -- [séo] m krat. (*angl.* search engine optimization)
gl. optimizacija spletnih strani

síva optimizácija -e -e ž (*angl.* grey hat SEO)
uporaba optimizacije spletnih strani mimo pravil spletnih iskalnikov; *prim.* črna optimizacija, bela optimizacija

skrékati -am dov. (*angl.* crack)
zlomiti zaščito programske opreme z namenom nepooblaščne uporabe in distribucije

vdirálec -lca m (*angl.* cracker, computer security hacker)
kdor vdira v tuje računalniške sisteme; sin. heker (3); *prim.* beli klobuk, črni klobuk

Izbor pripravlja in ureja
Katarina Puc s sodelavci Islovarja.

Pristopna izjava

za članstvo v Slovenskem društvu INFORMATIKA

Pravne osebe izpolnijo samo drugi del razpredelnice

Ime in priimek	
Datum rojstva	
Stopnja izobrazbe	srednja, višja, visoka
Naziv	prof., doc., spec., mag., dr.
Domači naslov	
Poštna št. in kraj	
Ulica in hišna številka	
Telefon (stacionarni/mobilni)	

Zaposlitev člana oz. člana - pravna oseba

Podjetje, organizacija	
Kontaktna oseba	
Davčna številka	
Poštna št. in kraj	
Ulica in hišna številka**	
Telefon	
Faks	
E-pošta	

Zanimajo me naslednja področja/sekcije*

- jezik
- informacijski sistemi
- operacijske raziskave
- seniorji
- zgodovina informatike
- poslovna informatika
- poslovne storitve
- informacijske storitve
- komunikacije in omrežja
- softver
- hardver
- upravná informatika
- geoinformatika
- izobraževanje

podpis

kraj, datum

Pošto društva želim prejemati na domači naslov / v službo.

Članarina znaša: 18,00 € - redna

7,20 € - za dodiplomske študente in seniorje (ob predložitvi dokazila o statusu)

120,00 € - za pravne osebe

Članarino, ki vključuje glasilo društva – revijo **Uporabna informatika**, bom poravnal sam / jo bo poravnal delodajalec.

DDV je vključen v članarino.



Naročilnica

 na revijo UPORABNA INFORMATIKA

Naročnina znaša: 35,00 € za fizične osebe

85,00 € za pravne osebe – prvi izvod

60,00 € za pravne osebe – vsak naslednji izvod

15,00 € za študente in seniorje (ob predložitvi dokazila o statusu)

DDV je vključen v naročnino.

ime in priimek ali naziv pravne osebe in ime kontaktne osebe

davčna številka, transakcijski račun

naslov plačnika

naslov, na katerega želite prejemati revijo (če je drugačen od naslova plačnika)

telefon/telefaks

elektronska pošta

Podpis

Datum

Koledar prireditev

The 11th International Symposium on Operations Research (SOR'11)	28.-30. september 2011	Dolenjske Toplice, Slovenija	http://sor11.fis.unm.si
XVIII. festival informatičkih dostignuća – INFOFEST 2011	2.-8. oktober 2011	Budva, Črna gora	www.infofest.com
Informacijska družba – IS'2011 – vzgoja in izobraževanje v informacijski družbi	14. oktober 2011	Ljubljana, Slovenija	http://www.smart-event.eu
MPP 2011 – Konferenca Management poslovnih procesov 2011	19.-20. oktober 2011	Ljubljana, Slovenija	http://www.process-conference.org
First IFIP CIO Forum	1.-4. november 2011	Shenzhen, Kitajska	www.worldcioforum.com
7th European Computer Science Summit – ECSS 2011	7.-9. november 2011	Milano, Italija	http://www.ecss2011.polimi.it
EEF 2011 – European Employment Forum	22.-23. november 2011	Bruselj, Belgija	www.europeanemploymentforum.eu
3. konferenca Informatika v javni upravi	28.-29. november 2011	Brdo pri Kranju, Slovenija	www.iju2011.si

Pomembni spletni naslovi

- IFIP News: <http://www.ifip.org/images/stories/ifip/public/Newsletter/news> ali www.ifip.org → Newsletter
- IT Star Newsletter: www.itstar.eu
- ECDL: www.ecdl.com
- CEPIS: www.cepis.com

Dostop do dveh tujih strokovnih revij

- Revija **Upgrade** (CEPIS) v angleščini (ISSN 1684-5285) je dostopna na spletnem naslovu: <http://www.upgrade-cepis.org/issues/2008/4/upgrade-vol-IX-4.html>.
- Revija **Novática** (CEPIS) v španščini (ISSN 0211-2124) je dostopna na spletnem naslovu: <http://www.ati.es/novatica/>.

Znanstveni prispevki

Jaka Žgajnar, Emil Erjavec, Stane Kavčič, Lidija Zadnik Stirn
Uporaba metod operacijskih raziskav pri načrtovanju kmetijske proizvodnje

Robert Dukarič, Matjaž B. Jurič
Migracija obstoječih aplikacij na platforme za računalništvo v oblaku

Viktorija Sulčič
Računalniška in internetna pismenost slovenske mladine

Martin Potočnik, Matjaž B. Jurič
Integracija procesiranja kompleksnih dogodkov in storitveno usmerjenih arhitektur

Strokovni prispevki

Mirica Šafran, Mateja Škornik, Dejan Skok, Borut Jereb
Uporaba odprtokodnih in prosto dostopnih programskih orodij – primer dobre prakse

Informacije

Poročilo o konferenci DSI 2011

Iz Islovarja

Koledar prireditev

ISSN 1318-1882

