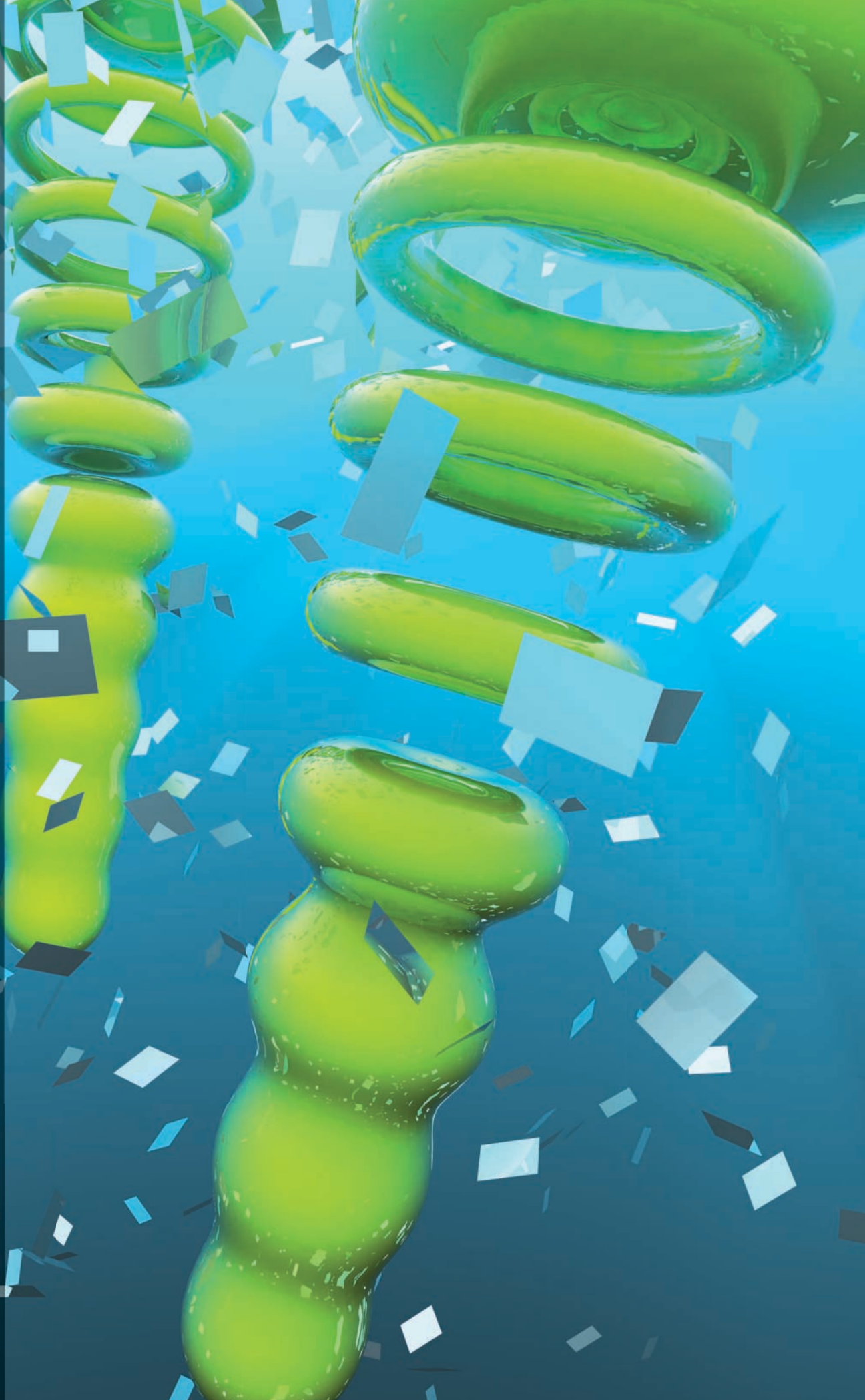


U P O R A B N A

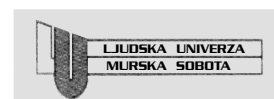
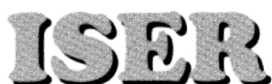
# I N F O R M A T I K A

2011 ŠTEVILKA 1 JAN/FEB/MAR 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 158 državah, ki so vključene v program ECDL. Doslej je bilo v svetu izdanih že več kot 10,8 milijona indeksov, v Sloveniji več kot 16.000 in podeljenih več kot 10.000 spričeval. Za izpitne centre v Sloveniji je usposobljenih 17 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 1 JAN/FEB/MAR LETNIK XIX ISSN 1318-1882

## Uvodnik

## Znanstveni prispevki

Marcel Križevnik, Matjaž B. Jurič: <b>Upravljanje matičnih podatkov kot osnova pri vpeljavi storitveno usmerjene arhitekture</b>	5
Franc Brcar: <b>Izzivi zunanjega izvajanja informatike</b>	15
Simon Vrhovec, Rok Rupnik: <b>Obvladovanje odpora pri projektih informacijskih tehnologij</b>	24

## Strokovni prispevki

Tomaž Dogša, Mirjam Bonačič: <b>Orodja za podporo testiranja parov</b>	38
Tomaž Stenšak, Tomaž Dogša: <b>Migracija wordovih dokumentov</b>	46

## Informacije

Katarina Puc, Niko Schlamberger: <b>Deset let terminološkega slovarja informatike</b>	54
<b>Iz Islovarja</b>	56
<b>Koledar prireditev</b>	59

#### 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 Vukšić, Gregor Hauc, Jurij Jaklič, Milton Jenkins, Andrej Kovačič, Katarina Puc, Vladislav Rajkovič, Heinrich Reineremann, Ivan Rozman, Rok Rupnik, Niko Schlamberger, John Taylor, Mirko Vintar, Tatjana Welzer Družovec

#### Recenzenti

Marko Bajec, Marko Bohanec, Vesna Bosilj Vukšić, 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 Reineremann, 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.

© 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](mailto: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](http://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.

*Spoštovane bralke in spoštovani bralci,*

*zdi se, da so nekatere teme na področju informatike vedno aktualne. Tako se že desetletja pogovarjamo o odnosu med informatiko in poslovanjem, čeprav informatika v poslovanju ne more biti drugega kot njegov del. Druga taka večna tema je kakovost podatkov (in informacij). Na podlagi izkušenj lahko brez težav pritrdim ugotovitvi, ki sem jo nekje prebral, da se menedžment začne praviloma ukvarjati s kakovostjo podatkov, šele ko ta postane problem, ki se odraža v poslovanju. Do takrat razumejo problematiko zagotavljanja kakovosti podatkov kot tehnološki izziv, ki naj ga rešujejo informatiki. Pa ni tako. Tudi na tem področju se uveljavljajo nove tehnologije in (bolj ali manj) novi metodološki pristopi. Obvladovanje matičnih podatkov (angl. Master Data Management), o katerem govori prvi prispevek v tej številki, je zagotovo eden od ključnih pogojev za kakovostne informacije, ki naj bodo na voljo uporabnikom. Tovrstne metodologije in tehnologije so lahko bistven element zagotavljanja kakovosti podatkov in informacij, vendar same po sebi ne morejo rešiti problemov nekakovosti, ki so posledica neurejenosti v poslovanju. Če v organizaciji ne obstaja globalni konceptualni podatkovni model oz. terminološki slovar ključnih poslovnih pojmov, če organizacija ne obvladuje poslovnih procesov in globalnih poslovnih pravil, potem sta integracija podatkov in, širše, kakovost podatkov le nedosegljiva želja. Funkcijski silosi se nujno odražajo v nekakovosti podatkov in informacij, še posebno ko se lotevamo celovitih rešitev (ERP, poslovnointeligentni sistemi).*

*Na vprašanja o zunanjem izvajanju informatike večinoma nimamo enoznačnih in dokončnih odgovorov, zato vam jih tudi ne more dati prispevek Izzivi zunanjega izvajanja informatike, zagotovo pa dodatno osvetljuje problematiko in vam lahko pomaga pri iskanju ustreznih rešitev v vaši organizaciji. Podobno lahko ugotovimo za naslednji prispevek, ki govori o problematiki odpora pri informacijskih projektih. Tokratna strokovna prispevka sta naravnana bolj tehnološko; prvi se ukvarja s testiranjem programov, drugi pa s problemom, s katerim se srečuje vsakdo med nami, to je z migracijo wordovih dokumentov.*

*Želimo vam veliko zanimivega branja v tej številki in se veselimo srečanja z vami aprila na Dnevih slovenske informatike v Portorožu.*

*Jurij Jaklič,  
odgovorni urednik*



**18. konferenca  
Dnevi slovenske informatike**

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

**Najpomembnejša neodvisna  
slovenska IT konferenca**

Prisluhnite vrhunskim slovenskim predavateljem,  
ki bodo predstavili izkušnje in novosti v naslednjih sekcijah:


- Poslovne aplikacije
- Poslovna inteligenca
- Menedžment poslovnih procesov
- Računalništvo v oblaku in SaaS
- 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

Tudi letos podelitev nagrade za najboljši IKT projekt, izbrali bomo tudi najboljši študentski projekt.

*Več informacij poiščite na spletni strani konference [www.dsi2011.si](http://www.dsi2011.si)*

*Pridružite se nam!*

Priveditelj konference

 slovensko  
društvo  
informatika

Organizacija konference

 ipmit

Dodatne informacije:

mag. Jasna Požnenel  
tel.: 01/30 09 810  
e-pošta: [dsi@drustvo-informatika.si](mailto:dsi@drustvo-informatika.si)  
[jasna.poznenel@ipmit.si](mailto:jasna.poznenel@ipmit.si)

# Upravljanje matičnih podatkov kot osnova pri vpeljavi storitveno usmerjene arhitekture

Marcel Križevnik, Matjaž B. Jurič

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

marcel.krizevnik@fri.uni-lj.si; matjaz.juric@fri.uni-lj.si

## Izvleček

Pri vpeljavi storitveno usmerjene arhitekture (SOA – Service Oriented Architecture) se organizacije soočajo s številnimi težavami. Eden izmed glavnih vzrokov za neuspeh projektov SOA je slaba kakovost podatkov, saj so nekateri ključni šifranti (npr. stranke, dobavitelji, produkti) pogosto podvojeni, nekonsistentni, neažurni in nepopolni. Tovrstni problemi lahko ogrozijo uspešnost vpeljave SOA, zato se v zadnjem času vedno več organizacij odloča za vpeljavo celovitega pristopa k upravljanju matičnih podatkov (MDM – Master Data Management), ki predstavlja najnovejši pristop k podatkovni integraciji ter uspešno odpravlja našteje težave. Izkušnje kažejo, da se MDM in SOA odlično dopolnjujeta, saj po eni strani MDM z izboljšanjem kakovosti podatkov poenostavi vpeljavo SOA, po drugi strani pa SOA zagotavlja komponente, ki olajšajo implementacijo storitvenega sloja MDM, tako da je smiselno združiti oba projekta. V prispevku bomo predstavili napreden arhitekturni model SOA-MDM, ki združuje prednosti obeh pristopov in veča raven učinkovitosti in fleksibilnosti celotne arhitekture.

**Ključne besede:** SOA, storitveno usmerjena arhitektura, MDM, upravljanje matičnih podatkov.

## Abstract

### Master Data Management as a Basis for Adoption of Service Oriented Architecture

Companies often face a number of difficulties when adopting SOA (Service Oriented Architecture). One of the most common problems is bad data quality as a result of redundancy, inconsistency, inaccuracy and incompleteness of important business data (such as customers, suppliers and products). Such problems may jeopardize the success of adopting SOA, so recently an increasing number of organizations have decided to introduce Master Data Management (MDM), which represents the latest approach to data integration and successfully eliminates the issues listed. Similarly as MDM simplifies the adoption of SOA, also SOA provides some components which facilitate the implementation of the MDM services layer, so it makes sense to combine both projects. In this article, we introduce an advanced SOA-MDM architectural model which combines the advantages of both approaches and thus improves levels of efficiency and flexibility of the whole architecture.

**Key words:** SOA, Service Oriented Architecture, MDM, Master Data Management.

## 1 UVOD

Pri vpeljavi storitveno usmerjene arhitekture (SOA – Service Oriented Architecture) [9, 10] organizacije pogosto zanemarijo pomen kakovosti podatkov in izpustijo izvedbo podatkovne integracije. Posledica tega so lahko precejšnje težave pri integraciji aplikacij in implementaciji poslovnih procesov. Če vsaka aplikacija uporablja svoj podatkovni vir, se lahko zgodi, da se posamezna stranka nahaja v več šifrantih, v katerih so strukture podatkov različne, zapisi pa so pogosto nekonsistentni, podvojeni, zastareli in nepopolni [6]. Zadnje raziskave kažejo, da se pričakovana stopnja podatkovnih napak v nekaterih organizacijah povzpne do 30 odstotkov [7]. Še slabši so rezultati druge raziskave [16], v kateri je kar

83 odstotkov sodelujočih organizacij trpelo hude posledice zaradi slabe kakovosti ključnih podatkov. Posledice slabe kakovosti podatkov se čutijo na vseh ravneh organizacije in se odražajo v nepravilnem izvajanju kritičnih poslovnih procesov, zmanjšanem vpogledu v izvajanje poslovnih aktivnosti, netočnem delovanju sistemov za podporo odločanju ter visoki porabi sredstev za odkrivanje in popraviljanje netočnih podatkov [1]. Omenjene težave lahko ogrozijo uspešnost vpeljave SOA in učinkovitost organizacije kot celote, zato je pri načrtovanju strategije SOA kot enega izmed začetnih korakov smiselno vključiti tudi izvedbo integracije in konsolidacije vseh ključnih šifrantov.

V prispevku bomo kot rešitev za odpravo opisanih težav predstavili celovit pristop k upravljanju matičnih podatkov (MDM – Master Data Management) [6, 14], ki se dopolnjuje s SOA in zagotavlja dolgoročno kakovost podatkov. MDM predvideva izolacijo ključnih (matičnih) podatkov iz obstoječih podatkovnih virov v centraliziran repozitorij ter vzpostavitev storitvenega sloja, ki omogoča učinkovito uporabo in upravljanje teh podatkov.

V prispevku bomo najprej izpostavili pomen zagotavljanja visoke kakovosti podatkov v organizaciji. V razdelku 3 bomo predstavili osnovne koncepte upravljanja matičnih podatkov (MDM). Nato bomo v razdelku 4 opisali prednosti in slabosti mogočih arhitekturnih stilov MDM ter kako izbrati najbolj ustrezno arhitekturo MDM. V razdelku 5 bomo opisali, kako je mogoče z uporabo principov SOA implementirati učinkovit in fleksibilen sloj matičnih podatkovnih storitev, ki pred poslovnim nivojem skriva vso nizkonivojsko kompleksnost podatkovnega nivoja. V razdelku 6 bomo predstavili predlagani postopek vpeljave MDM, ki povzema dobre prakse in omogoča organizacijam, da se izognejo nekaterim pogostim napakam. V razdelku 7 bomo predstavili sorodne raziskave ter na koncu v razdelku 8 podali sklep prispevka.

## 2 KAKOVOST PODATKOV V SOA

Pojem kakovost podatkov je zelo širok in predstavlja pričakovano raven kakovosti podatkov, ki organizaciji omogoča učinkovito izvajanje poslovnih aktivnosti. Preden se lotimo odprave problema slabe kakovosti podatkov, moramo definirati in razumeti zahteve (dimenzije) kakovosti podatkov [8, 14, 18]:

- **unikatnost:** zahteva, da so entitete zajete in predstavljene unikatno, kar pomeni, da nobena entiteta iz realnega sveta ne sme biti podvojena in da obstaja enolični identifikator za dostop do posamezne entitete;
- **točnost:** stopnja točnosti, s katero entitete v podatkovnem viru predstavljajo objekte iz realnega okolja. Točnost je v praksi zelo težko spremljati, saj potrebujemo za primerjavo podatkovni vir, za katerega smo prepričani, da je točen in ažuren, saj se lahko podatki v realnem svetu spreminjajo zelo hitro;
- **konsistentnost:** zahteva, da so podatki v eni zbirki podatkov konsistentni s podatki v drugi podatkovni zbirki. Podvajanje podatkov je sicer v na-

sprotju z zahtevo po unikatnosti, vendar se temu v realnem primeru včasih ne moremo izogniti, saj vedno ni mogoče izvesti popolne integracije in konsolidacije podatkov. Konsistentnost najpogosteje zagotavljamo z redno sinhronizacijo podatkovnih virov;

- **popolnost:** stopnja popolnosti se nanaša na delež od nič različnih (not null) vrednosti. Pri preverjanju popolnosti je treba podatke razdeliti v tri skupine. Prvo skupino predstavljajo obvezni podatki; to so podatki, ki morajo v vsakem trenutku imeti vpisano vrednost. Primer je, recimo, podatek o imenu osebe. V drugo skupino spadajo opcijski podatki. V zadnjo skupino pa spadajo podatki, ki so za nekatere zapise relevantni, za druge pa ne. Podatek o moči v šifrantu artiklov npr. ni relevanten za objekt klobuk;
- **dostopnost:** časovna pričakovanja v zvezi z dostopnostjo informacij. Merimo jo lahko v času, ki preteče med tem, ko je podana zahteva po podatku, in ko je podatek dejansko na voljo. Za lažje spremljanje dostopnosti priporočamo definiranje dogovora o ravni storitev (SLA – Service Level Agreement);
- **ažurnost:** stopnja ažurnosti podatkov v podatkovnem viru glede na realno stanje, saj se lahko podatki v realnem svetu spreminjajo zelo hitro. Ažurnost lahko merimo kot funkcijo pričakovane frekvence spreminjanja podatkov. Primer je pričakovana pogostost spreminjanja telefonskih števil;
- **skladnost s formatom:** vsak podatkovni model definira nabor pravil, ki definirajo predstavitev in način shranjevanja podatkov. Ta pravila določajo podatkovni tip podatkov, maksimalno dovoljeno dolžino ipd.;
- **referenčna integriteta:** pravila za ohranjanje integritete in konsistentnosti podatkov v podatkovnem viru, ki npr. definirajo, da je zaloga vrednosti tujih ključev enaka uniji vseh primarnih ključev za posamezno tabelo v podatkovni bazi.

Večina organizacij se sooča s težavami pri doseganju teh zahtev, a le peščici jih uspe uspešno odpraviti. Samo vpeljava rešitev ERP (Enterprise Resource Planning) in CRM (Customer Relationship Management) običajno ne prinese zelenih rezultatov, saj te rešitve ne zagotavljajo dolgoročne kakovosti podatkov, poleg tega ne omogočajo integracije vseh ključnih šifrantov. Organizacije želijo predvsem nov pristop, ne pa le še enega šifranta, ki ga je treba vzdrževati. Iz omenjenih razlogov se v zadnjem času vedno



bolj uveljavlja pristop MDM, ki se odlično dopolnjuje s SOA in zagotavlja dolgoročno kakovost podatkov.

### 3 UPRAVLJANJE MATIČNIH PODATKOV

Podatke v organizaciji delimo v tri skupine; to so:

- **transakcijski podatki**, ki se uporabljajo pri vsakodnevem izvajanju aplikacij, imajo izrazito časovno dimenzijo ter predstavljajo rezultat izvedbe posamezne transakcije. Primeri transakcijskih podatkov: naročila, računi, plačila itn. Za razliko od matičnih podatkov količina transakcijskih podatkov raste skoraj premo sorazmerno s poslovno aktivnostjo;
- **analitični podatki**, ki podpirajo sprejemanje poslovnih odločitev. Nahajajo se v podatkovnih skladiščih in se uporabljajo pri podatkovnem rudarjenju. Analitični podatki so shranjeni v velikih tabelah dejstev, ki jih obdajajo ključne dimenzije, kot so stranke, produkti, dobavitelji, računi ipd.;
- **matični podatki**, to so pomembni poslovni objekti, ki jih uporablja več informacijskih sistemov in predstavljajo ključ pri izvedbi transakcij in poslovnih procesov [2]. Ti podatki prav tako predstavljajo dimenzije v podatkovnem skladišču, na podlagi katerih se izvajajo analize za podporo poslovni inteligenci. Najpogostejši primeri matičnih podatkov so šifranti strank, dobaviteljev, produktov, zaposlenih, računov itn. Za uspešno delovanje organizacije je zagotavljanje kakovosti teh podatkov izredno pomembno. Posledice nezadostne kakovosti matičnih podatkov so lahko zelo hude in se izražajo v nepravilnem izvajanju poslovnih aktivnosti, netočnem delovanju sistemov za podporo odločanju, visoki porabi sredstev za odkrivanje in odpravljanje netočnih podatkov, oteženem razvoju novih rešitev itn.

Upravljanje matičnih podatkov (MDM) [6, 14] je celovit pristop k izvedbi podatkovne integracije, ki združuje nabor procesov in orodij za konsistentno definiranje in upravljanje ključnih (matičnih) podatkov. Bistvo MDM je v izolaciji matičnih podatkov iz obstoječih aplikacij v centraliziran repozitorij, ki omogoča celovito upravljanje življenjskega cikla podatkov ter ponuja nabor podatkovnih storitev, s pomočjo katerih lahko do teh podatkov dostopajo vse aplikacije, ki operirajo s temi podatki. MDM se od drugih pristopov k podatkovni integraciji razlikuje predvsem po tem, da zahteva natančno definicijo od-

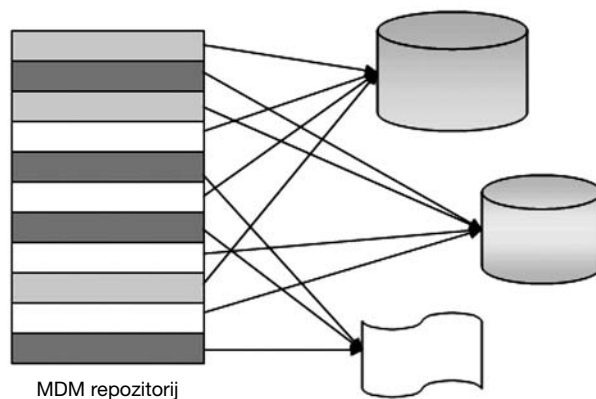
govornosti, vlog in politik za vzpostavitev celovitega upravljanja podatkov (Data Governance), ki se ne zaključijo s polnjenjem matičnega repozitorija, temveč predstavlja ponavljajočo se aktivnost v vsem življenjskem ciklu matičnih podatkov. MDM torej zagotavlja trajno kakovost podatkov ter s tem predstavlja idealno podlago za vpeljavo SOA.

### 4 ARHITEKTURNI STILI MDM

Obstajajo trije glavni arhitekturni pristopi k MDM [14]: navidezni register, transakcijsko vozlišče ter hibridni model. Cilj vseh arhitektur MDM je podpora transparentnosti in skupni uporabi unikatnih predstavitev matičnih podatkov.

#### 4.1 Navidezni register

Arhitekturni stil navidezni register (slika 1) za delovanje uporablja majhen indeks matičnih podatkov, ki vsebuje le minimalen nabor enoličnih identifikatorjev. Register torej vsebuje le kazalnike na dejanske zapise, ki pa se še vedno nahajajo na strani izvornih aplikacijskih podatkovnih virov.



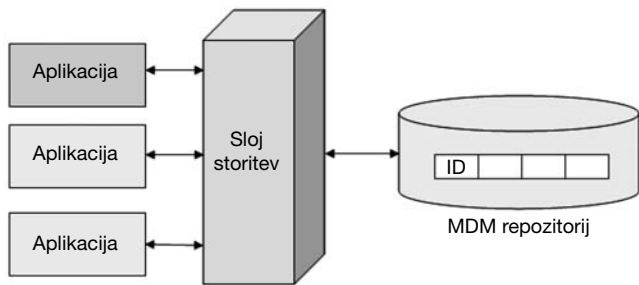
Slika 1 [14]: **MDM arhitekturni stil navidezni register**

Arhitekturni stil navidezni register torej ne predvideva polne integracije podatkov, saj se matični podatki ne nahajajo v centralnem repozitoriju in tako ne moremo govoriti o zlatem zapisu. O konceptu zlatega zapisa lahko govorimo le v primeru, ko so podatki konsolidirani in integrirani ter se nahajajo le v matičnem repozitoriju, ki tako za podatke predstavlja edino verzijo resnice (single version of truth). Prednost uporabe tega arhitekturnega stila je predvsem dejstvo, da obstoječih aplikacij ni treba prilagajati, da bi podatke pridobivale iz centralnega registra

s pomočjo matičnih podatkovnih storitev. Register implementiramo tako, da za vsako entiteto iz realnega sveta v repozitoriju obstaja kvečjemu en zapis, ki vsebuje stolpce s primarnimi ključi, ki kažejo na zapise v podatkovni virih. Če posamezen podatkovni vir ne vsebuje določenega zapisa, je vrednost ključa enaka vrednosti null. Glavna slabost tega pristopa so kompleksna SQL-povpraševanja, ki lahko vključujejo več podatkovnih virov in vračajo podvojene zapise. Ker se matični podatki nahajajo v lokalnih podatkovnih virih, sta dostopnost in ažurnost podatkov visoki. Zaradi dejstva, da se za posamezen objekt iz realnega okolja v podatkovnih virih lahko nahaja več zapisov, pa so stopnje koherentnosti, konsistentnosti in determinizma nizke. Arhitekturni stil navidezni register po navadi predstavlja prvo stopnjo v evoluciji rešitve MDM in ne sme predstavljati končnega cilja, ki je izolacija matičnih podatkov v centralni repozitorij MDM.

### 4.2 Transakcijsko vozlišče

Transakcijsko vozlišče (slika 2) je celovit repozitorij, ki se uporablja za upravljanje vseh aspektov matičnih podatkov. Vsi matični podatki se nahajajo le v tem repozitorju in se ne replicirajo v druge podatkovne vire, zato je treba ustrezno prilagoditi aplikacije za manipuliranje s temi podatki. Za dostop do matičnih podatkov in upravljanje njihovega življenjskega cikla aplikacije uporabljajo matične podatkovne storitve.



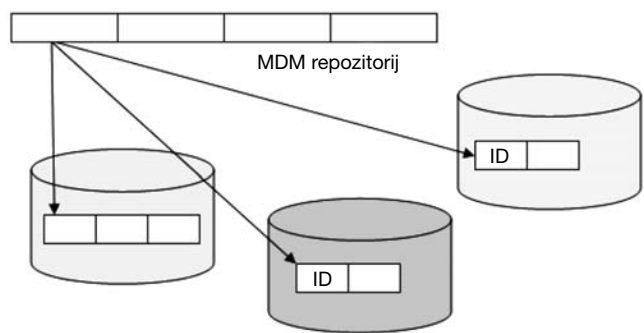
Slika 2 [14]: **MDM arhitekturni stil transakcijsko vozlišče**

Transakcijsko vozlišče funkcioniira kot edina verzija matičnih podatkov, pri čemer so vsi atributi podatkov del matičnega repozitorija. Za vsako entiteto iz realnega sveta obstaja v matičnem repozitoriju kvečjemu en zapis, zato lahko govorimo o konceptu zlatega zapisa. Stopnje dostopnosti, ažurnosti, determinizma in konsistence so visoke. Kljub vsem našte-

tim prednostim pa ima ta arhitekturni stil tudi svoje pomanjkljivosti. Prvo težavo predstavlja dejstvo, da obstoječih aplikacij ni mogoče vedno prilagoditi, saj se lahko zgodi, da nimamo na voljo izvorne kode. Tudi če jo imamo, je lahko prilagajanje aplikacij zelo boleče in zahteva veliko časa in truda. Prav tako se lahko zgodi, da repozitorij MDM postane glavno ozko grlo v arhitekturi.

### 4.3 Hibridni model

Hibridni model (slika 3) predstavlja kompromis med prej predstavljenima arhitekturnima stiloma in lahko poleg enoličnih identifikatorjev za posamezen zapis vsebuje tudi nabor konsolidiranih skupnih podatkovnih atributov; te podatke imenujemo tudi skupni osnovni matični objekti.

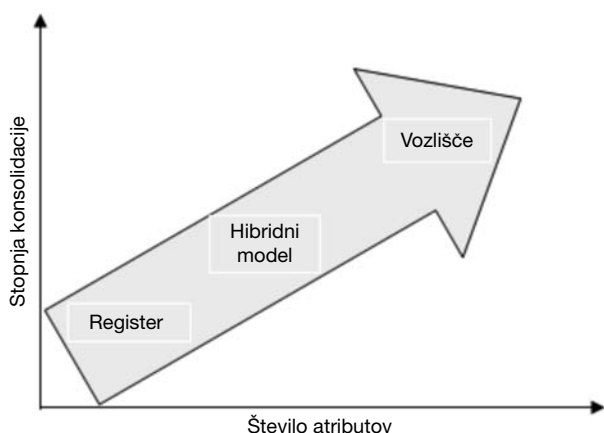


Slika 3 [14]: **MDM arhitekturni stil hibridni model**

Ker se nekateri atributi nahajajo v matičnem repozitoriju, drugi pa v posameznih aplikacijskih virih, je izvedba sinhronizacije še posebno zahtevna, saj ne želimo ogroziti konsistentnosti med podatki v matičnem repozitoriju in podatki v lokalnih podatkovnih virih. Spremembe nad matičnimi objekti se morajo zato propagirati med vsemi lokalnimi kopijami. Izvedba paketne sinhronizacije ob določenih časovnih terminih najpogosteje ne predstavlja ustrezne rešitve, saj ne želimo ogroziti ažurnosti in konsistentnosti podatkov. Vsako spremembo nad matičnimi podatki želimo torej takoj propagirati do lokalnih kopij. Tukaj pa se pojavi problem težavnosti implementacije matičnih podatkovnih storitev, saj je treba ob spremembi ali branju matičnega podatka dostopati do več podatkovnih virov in hkrati skriti vso kompleksnost pred poslovnim nivojem. Tukaj nam je lahko v precejšnjo pomoč SOA, kar bomo podrobneje razložili v razdelku 5.

#### 4.4 Izbira ustreznega arhitekturnega stila

Izbira ustreznega arhitekturnega stila je vse prej kot preprosta. Na prvi pogled se zdi najboljša izbira transakcijsko vozlišče, saj zagotavlja visoko ažurnost in konsistentnost, prav tako pa zagotavlja preprosto implementacijo matičnih podatkovnih storitev. Vendar pa ima tudi ta arhitekturni stil svoje slabosti. Pri izbiri ustrezne arhitekture je treba upoštevati te dejavnike: število atributov, zahtevano stopnjo konsolidacije ter sklopljenost aplikacij z matičnimi podatki. Register je najbolj primeren za okolja, v katerih je število atributov majhno in so aplikacije šibko sklopljene, zahteve po konsolidaciji pa niso previsoke. Prav tako ta pristop zahteva najmanj časa, truda in finančnih sredstev, zato pogosto predstavlja prvo stopnjo pri vpeljavi MDM. Ravno nasprotno velja za transakcijsko vozlišče. Hibridni model pa se, kot prikazuje slika 4, nahaja nekje vmes.



Slika 4: **Izbira ustreznega MDM arhitekturnega stila**

Našteti kriteriji pa seveda niso edino merilo pri odločanju o izbiri ustrezne arhitekture. Upoštevati je namreč treba tudi predvideno kompleksnost storitvenega sloja, mehanizme dostopa ter zahteve glede odzivnosti.

#### 5 IZGRADNJA STORITVENEGA SLOJA MDM

Ko so podatki konsolidirani in je matični repozitorij napolnjen, vpeljava MDM še zdaleč ni končana. Zbrani podatki postanejo uporabni šele, ko sta mogoča njihovo upravljanje in uporaba. Ena izmed pglavitnih nalog MDM je torej vzpostavitev sloja matičnih podatkovnih storitev [3, 13, 14, 23], ki aplikacijam omogoča uporabo matičnih podatkov ter upravljanje

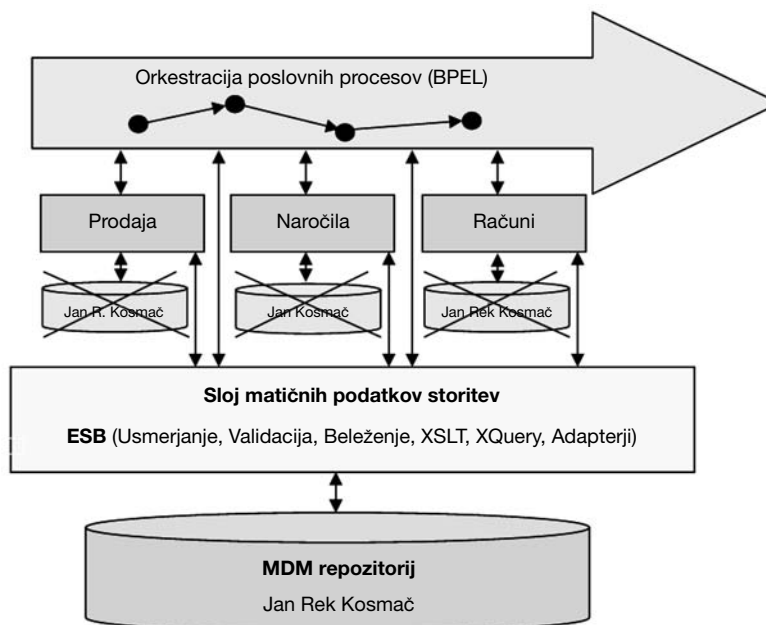
njihovega življenjskega cikla. Te storitve lahko razdelimo v te skupine:

- **kreiranje** matičnega zapisa v centralnem repozitoriju. V primeru arhitekturnega stila navidezni register ali hibridni model vključuje tudi kreiranje zapisa v lokalnih podatkovnih virih. Vstavljanje novega matičnega zapisa lahko vključuje tudi preverjanje skladnosti z definiranimi politikami upravljanja matičnih podatkov. Tako se ob vstavljanju zapisa po navadi preveri kakovost vpisanega podatka ter izvrši preverjanje, če morebiti v matičnem repozitoriju že obstaja podatek za to entiteto. Prav tako se lahko izvrši postopek avtentikacije in avtorizacije (vse osebe nimajo pravice vstavljanja matičnih podatkov). Vsako vstavljanje zapisa se tudi zabeleži skupaj s časovno značko, kar omogoča sledenje sprememb;
- **dostop in uporaba:** branje in spreminjanje matičnih podatkov. V primeru transakcijskega vozlišča gre za preproste operacije branja in spreminjanja podatkov v matičnem repozitoriju. V primeru hibridnega modela ali navideznega registra pa so te operacije bolj kompleksne, saj v transakciji sodeluje več podatkovnih virov. Vsaka sprememba matičnega podatka se zabeleži in preveri se kakovost spremenjenih podatkov;
- **deaktivacija in umik:** ko matični podatek ni več aktualen, ga je treba deaktivirati in umakniti iz uporabe, vzrok za to pa podrobno dokumentirati. Deaktivacija matičnega podatka ne pomeni fizični izbris podatka, temveč ustrezno spremembo statusa;
- **vzdrževanje in popraviljanje:** nabor storitev, ki omogoča paketno preverjanje kakovosti podatkov v matičnem repozitoriju in tako zagotavlja dolgotrajno kakovost podatkov. Preverjanje vključuje odpravo podvojenih zapisov, preverjanje konsistentnosti podatkov ipd.

Glavni namen storitvenega sloja MDM je torej aplikacijam omogočiti čim bolj učinkovito in preprosto manipuliranje z matičnimi podatki. Kot smo že spoznali, je težavnost implementacije matičnih podatkovnih storitev v veliki meri odvisna od izbranega arhitekturnega stila. V primeru transakcijskega vozlišča je vstavljanje, branje in spreminjanje matičnih podatkov preprosto, saj se vsi podatki nahajajo znotraj matičnega repozitorija. Če uporabimo hibridni model, pa še zdaleč ni tako. Ker so v tem primeru matični podatki razpršeni v več podatkovnih virih,

je dostop do njih precej bolj zapleten. Ob vstavljanju zapisa se tako v matični repozitorij vstavi zapis z naborom skupnih atributov. Poleg tega se nov zapis vstavi tudi v vse lokalne podatkovne vire, pri čemer se v posamezni vir zapišejo le aplikacijsko specifični atributi. Pri tem je treba upoštevati, da so lahko podatkovni viri različnih tipov. Celotna operacija se mora izvršiti kot globalna transakcija, izvesti se mora tudi preverjanje kakovosti vnešenih podatkov in zabeležiti se mora sprememba. Preprosta operacija vstavljanja torej postane precej kompleksna. Podobno velja za branje podatkov. Implementacijo sloja podatkovnih storitev MDM pa lahko precej poenostavimo z uporabo pristopa SOA. V tem prispevku predlagamo napredni arhitekturni model SOA-MDM, ki omogoča izgradnjo učinkovitega in fleksibilnega sloja matičnih podatkovnih storitev. Model skriva vso kompleksnost podatkovnega sloja pred poslovnim nivojem in vpeljuje koncept t. i. virtualnega zlatega zapisa, ki je neodvisen od izbranega MDM arhitekturnega stila. Pri izgradnji podatkovnega sloja MDM predlagamo uporabo naslednjih komponent SOA, ki so sestavni del vsake celovite platforme SOA:

- **storitveno vodilo** (ESB – Enterprise Service Bus) [10] s svojimi zmožnostmi usmerjanja zahtev in pretvarjanja med različnimi podatkovnimi formati predstavlja komunikacijsko hrbtenico vsake zrele storitveno usmerjene arhitekture, saj odpravlja neposredne povezave med ponudniki in uporabniki storitev ter zagotavlja šibko sklopjenost, ki je ena izmed glavnih zahtev SOA. ESB nam tako omogoča implementacijo fleksibilnih matičnih podatkovnih storitev, pri čemer se zahteva za branje ali spreminjanje matičnega podatka preusmeri do matičnega repozitorija in v primeru hibridnega modela ali navideznega registra tudi do lokalnih podatkovnih virov. Podatke lahko po potrebi ustrezno pretvorimo, saj ESB podpira transformacije med podatkovnimi formati (uporaba XSLT, XQuery). S pomočjo ESB lahko preprosto implementiramo tudi beleženje vseh sprememb ter validacijo podatkov v primeru spreminjanja ali dodajanja zapisov;
- **sistem za upravljanje s poslovnimi pravili** (BRMS – Business Rules Management System) je sestavni del vsake zrele SOA-platforme in nam omogoča izolacijo poslovnih pravil zunaj kode aplikacij, kar omogoča bolj fleksibilno upravljanje poslovnih pravil. Pri implementaciji storitvenega sloja MDM lahko poslovna pravila uporabimo za validacijo kakovosti podatkov ter za preverjanje, kdo ima pravico branja ali spreminjanja matičnih podatkov. Recimo, da spletna trgovina svojim zvestim strankam dodeli status flzlati uporabnik«, kar jim prinaša dodatne ugodnosti v obliki popustov, in da je eden izmed pogojev za pridobitev tega statusa, da je uporabnik registriran že vsaj tri leta. Matična podatkovna storitev mora torej pred vsakim spreminjanjem statusa uporabnika izvesti ustrezno validacijo ter s klicem poslovnega pravila preveriti, ali uporabnik izpolnjuje zahtevana merila;
- **adapterji** omogočajo preprost dostop do različnih tipov podatkovnih virov (relacijske podatkovne baze, podatkovne baze XML, datoteke, sporočilne vrste) s pomočjo uporabe čarovnikov. Uporaba adapterjev nam lahko precej olajša delo z matičnimi podatki, še posebno ko se ti nahajajo v več podatkovnih virih, ki so lahko različnih tipov;
- **jezik BPEL** (Business Process Execution Language) [15] omogoča orkestracijo spletnih storitev v poslovne procese in je ena izmed najpomembnejših komponent vsake platforme SOA. Pri implementaciji matičnih podatkovnih storitev ga lahko uporabimo skupaj z ESB za definiranje kompleksnejših podatkovnih storitev, ki vključujejo tudi pošiljanje obvestil, uporabniške aktivnosti itn. BPEL lahko npr. uporabimo pri spreminjanju kritičnih matičnih podatkov, pri čemer je npr. za brisanje podatkov predvideno večnivojsko ročno potrjevanje za to zadolženih uporabnikov;
- **dogodkovni mehanizmi** (npr. EDA – Event Driven Architecture) [21] podpirajo šibko sklopljeno komunikacijo, ki sledi vzorcu objavi–naroči. V našem arhitekturnem modelu SOA-MDM predlagamo uporabo dogodkovnih mehanizmov za objavo dogodkov o spremembah nad matičnimi podatki v centralnem repozitoriju MDM in posredovanje teh dogodkov do za to zadolženih podatkovnih storitev, ki te spremembe propagirajo do lokalnih podatkovnih virov. Tako lahko zagotovimo visoko stopnjo konsistentosti med podatki v matičnem repozitoriju ter morebitnimi lokalnimi kopijami matičnih podatkov. Predlagani arhitekturni model SOA-MDM prikazuje slika 5.



Slika 5: **Uvedba storitvenega sloja MDM s pomočjo SOA**

V primeru transakcijskega vozlišča so vsi matični podatki konsolidirani v repozitoriju MDM, v primeru hibridnega modela ali navideznega registra pa se matični podatki nahajajo tudi v lokalnih podatkovnih virih. Storitveni sloj MDM je implementiran s pomočjo pristopa SOA, ki predvideva uporabo storitvenega vodila, jezika BPEL, dogodkovnih mehanizmov, beleženja, uporabe poslovnih pravil, adapterjev itn. Obstoječe aplikacije so tako prilagojene, da do matičnih podatkov dostopajo prek sloja matičnih podatkovnih storitev. Sloj matičnih podatkovnih storitev skriva vso kompleksnost podatkovnega nivoja in zagotavlja pogled virtualnega zlatega zapisa. Rezultat je bistveno izboljšana fleksibilnost, saj imajo razvijalci enovit in konsistenten pogled na matične podatke in se jim ni treba ukvarjati s tem, kje so dejansko shranjeni ti podatki.

## 6 PREDLAGANI POSTOPEK VPSELJAVE MDM

Postopek vpeljave MDM [14] je lahko zelo obsežen in dolgotrajen, zato se ga je treba lotiti premišljeno in z majhnimi koraki. To je še posebno pomembno v primeru, ko se ga lotevamo sočasno z vpeljavo storitveno usmerjene arhitekture. V nadaljevanju je navedenih šestnajst korakov, ki predstavljajo predlagani postopek prehoda in povzemajo številne dobre prakse.

### 1. Pridobitev podpore vodstva organizacije

MDM služi predvsem kot dobra podlaga za doseganje drugih ciljev, kot so npr. vpeljava sistema SOA ali ERP, izboljšano delovanje sistemov za poročanje, izboljšano obvladovanje napak, olajšano sprejemanje poslovnih odločitev, boljša poslovna produktivnost itn. V tej luči je treba projekt najprej predstaviti vodstvu organizacije, ki mora dati zeleno luč za začetek projekta in odobritev sredstev.

### 2. Razvoj osnovnega načrta vpeljave MDM

Projekt vpeljave MDM zahteva strateško planiranje s podrobnimi opisi zahtev za posamezne faze. Te zahteve kasneje služijo kot glavno merilo pri spremljanju uspešnosti postopka vpeljave. Primer takšnega načrta je lahko sestavljen iz teh faz: priprava, izvedba pilotnega POC (Proof of Concept) projekta, začetna namestitvev, prva verzija, prehod itn. Kadar vpeljava MDM spada v okvir vpeljave SOA, je osnovni načrt MDM seveda del osnovnega načrta vpeljave SOA.

### 3. Določitev vlog in odgovornosti

Določiti je treba vloge ter pripadajoče odgovornosti za vse osebe, ki sodelujejo pri projektu. V vsakem trenutku mora vsaka oseba natančno poznati svoje zadolžitve. To je eden izmed ključnih korakov v postopku vpeljave MDM. Ena izmed najpomembnejših vlog je vloga skrbnika podatkov, ki je neposredno odgovoren za spremljanje in zagotavljanje kakovosti podatkov enega ali več šifrantov matičnih podatkov [11, 20].

#### **4. Planiranje projekta**

Ko je pripravljen osnovni načrt vpeljave ter so dodeljene odgovornosti, je treba pripraviti podroben načrt z natančno dodeljenimi odgovornostmi ter roki za končanje posameznih faz, kar predstavlja podlago za vse naslednje korake.

#### **5. Analiza modelov poslovnih procesov**

Poslovni procesi predstavljajo jedro vsake organizacije. Iz tega razloga je treba začeti pri njih, vedno ko želimo izboljšati delovanje organizacije. Tako lahko identificiramo aplikacije, ki sodelujejo pri izvedbi procesov, ter podatke, ki se prenašajo med posameznimi aktivnostmi. Rezultat analize je seznam modelov poslovnih procesov, ki uporabljajo funkcionalnosti, katerih podatkovne vire bi bilo smotno prestaviti v matično okolje.

#### **6. Identifikacija podatkovnih virov za integracijo**

Korak predvideva podrobno preučitev zbranih procesnih modelov z namenom identifikacije ključnih šifrantov, ki predstavljajo potencialne kandidate za izvedbo podatkovne integracije.

#### **7. Obvladovanje podatkov**

Korak narekuje definiranje informacijskih politik, ki odsevajo poslovne potrebe in pričakovanja ter procese za spremljanje skladnosti s temi politikami.

#### **8. Vpeljava upravljanja z metapodatki**

Upravljanje metapodatkov zagotavlja podlago za vse nadaljnje aktivnosti. Korak predvideva vpeljavo metapodatkovnega repozitorija, ki omogoča kontrolo in spremljanje razvoja matičnega okolja, z vzdrževanjem tehle informacij: rezultati analize podatkov, razlaga posameznih podatkovnih tipov, seznam matičnih podatkovnih tipov, podatkovni modeli matičnih podatkov, opis interakcije med aplikacijami in matičnimi podatki, zahteve glede kakovosti podatkov, dostop do podatkov in njihovo upravljanje itn.

#### **9. Analiza matičnih podatkov**

Faza predstavlja pripravo na nadaljnji korak, tj. modeliranje matičnega repozitorija. Na tej stopnji je treba analizirati in konsolidirati različne predstavitve matičnih podatkov, ki se nahajajo v številnih aplikacijskih podatkovnih virih. Rezultati koraka: križno-sistemska podatkovna analiza, zaključeni metapodatkovni profili, poslovna pravila, preslikave in navodila za transformacijo podatkov.

#### **10. Modeliranje matičnih podatkov**

Izdelava podatkovnega modela matičnih podatkov za uporabo v matičnem repozitoriju. Rezultata

modeliranja sta model za izvedbo konsolidacije ter model za trajno shranjevanje matičnih podatkov.

#### **11. Upravljanje kakovosti podatkov**

Eden izmed ključnih ciljev pri vpeljavi matičnega okolja je povečano zaupanje v kakovost podatkov. Za zagotavljanje tega je treba najprej definirati pričakovano raven kakovosti podatkov. Pri zagotavljanju kakovosti podatkov pa si je treba pomagati s posebnimi orodji za razčlenjevanje in standardizacijo podatkov. Pričakovani rezultati koraka: definirane poslovne zahteve glede orodij za izboljševanje kakovosti, formalizacija in implementacija tehnik za trajno zagotavljanje kakovosti podatkov, pridobitev in uporaba orodja za izboljšanje kakovosti podatkov.

#### **12. Izbira arhitekture MDM**

Mogoče arhitekture MDM so opisane v razdelku 4. V tem koraku gre torej za analizo sistema z namenom ugotovitve števila atributov ter stopnje sklopljenosti in zahtevane ravni konsolidacije. Ko izberemo najprimernejšo arhitekturo, je treba preučiti še vse rešitve na tržišču ter nato izbrati najprimernejšo.

#### **13. Integracija in konsolidacija podatkov ter polnjenje matičnega okolja**

Korak predvideva uporabo integracijskih in konsolidacijskih orodij ter polnjenje matičnega repozitorija, ki od sedaj naprej predstavlja edini vir matičnih podatkov.

#### **14. Uvedba storitvenega sloja MDM**

Storitveni sloj sestavlja nabor matičnih podatkovnih storitev, ki omogočajo uporabo in upravljanje matičnih podatkov. Načini implementacije so lahko precej različni, v večini primerov pa se kot najprimernejša izkaže uporaba pristopa SOA, ki je podrobneje opisan v razdelku 5.

#### **15. Izdelava plana prehoda na MDM**

Potrebno je preoblikovanje aplikacij, da začnejo namesto lokalnih podatkovnih virov uporabljati podatke iz matičnega repozitorija. Seveda je treba pred začetkom prehoda izdelati podroben plan, v katerem so postavljeni roki posameznih korakov ter identificirane ključne težave, do katerih lahko pride pri izvedbi prehoda.

#### **16. Vzdrževanje MDM**

Ko imamo postavljeno arhitekturo MDM in so matični podatki že v uporabi, je treba vseskozi spremljati in nadzirati kakovost podatkov, po potrebi dopolniti podatkovni model in storitveni sloj, zagotavljati redno sinhronizacijo podatkov, izdelovati poročila o kakovosti podatkov itn.

## 7 SORODNE RAZISKAVE

Številni avtorji [2, 7, 8, 18] so v svojih delih izpostavili pomen zagotavljanja kakovosti podatkov v organizaciji. Kot verjetno najboljšo rešitev za trajno izboljšanje kakovosti podatkov so mnogi predlagali uporabo pristopa MDM [1, 3, 4, 5, 11, 12, 14]. Avtorji v [19] kritično presojujejo MDM in ugotavljajo, ali gre dejansko za nov pristop ali samo za že znan pristop v novi preobleki. Prispevek sklenejo z ugotovitvijo, da MDM predstavlja evolucijo obstoječih pristopov k podatkovni integraciji (ERP, CRM), ki se niso najbolje obnesli, in da bo treba na tem področju opraviti še precej raziskav. Cleven in Wortmann [5] predlagata ogrodje s štirimi mogočimi strategijami vpeljave MDM, vendar ne omenjata povezave med MDM in SOA. Na možnost uporabe MDM za zagotavljanje kakovosti podatkov v SOA so v svojih delih opozorili številni avtorji [1, 3, 4, 6, 13, 17, 23]. Wang [22] npr. smatra MDM kot najpomembnejšo strateško začetno točko pri vpeljavi SOA. Loshin [14] izpostavlja simbiotsko razmerje med MDM in SOA, pri čemer storitveno orientirani pristop k razvoju poslovnih rešitev temelji na obstoju repozitorija ključnih poslovnih podatkov in je uspeh vpeljave MDM odvisen od možnosti izgradnje fleksibilnega storitvenega sloja MDM. Kalagiriou [12] razlaga, kako si SOA in MDM delita številne razvojne principe, kot so ponovna uporaba, abstrakcija, razvoj na podlagi dobro definiranih pogojev itn. Butler in Pollock [3, 4] omenjata možnost uporabe storitvenega vodila pri izgradnji storitvenega sloja MDM, vendar ne podajata konkretnjših arhitekturnih napotkov, ki bi bili primerljivi z modelom, predstavljenim v tem prispevku. Predlagani arhitekturni model SOA-MDM tako nadgrajuje prispevke omenjenih avtorjev, tako da definira konkretne predloge, kako s pomočjo principov in komponent SOA implementirati fleksibilen sloj matičnih podatkovnih storitev, ki skriva vso kompleksnost podatkovnega nivoja in bistveno poenostavi vpeljavo MDM, ki poteka kot eden izmed začetnih korakov vpeljave SOA.

## 8 SKLEP

V prispevku smo izpostavili pomen zagotavljanja kakovosti podatkov v storitveno usmerjeni arhitekturi. Identificirali smo ključne težave ter kot najboljšo rešitev za njihovo odpravo predstavili pristop celovitega upravljanja matičnih podatkov (MDM). Opisali

smo mogoče arhitekturne stile MDM ter predstavili pomen izgradnje sloja matičnih podatkovnih storitev, ki zagotavlja učinkovito uporabo teh podatkov. SOA in MDM se torej odlično dopolnjujeta, zato je pri prehodu na SOA kot enega izmed začetnih korakov smiselno razmisliti tudi o izvedbi podatkovne integracije s pomočjo pristopa MDM. V prispevku smo predstavili napredni arhitekturni model SOA-MDM, ki omogoča izgradnjo fleksibilnega storitvenega sloja MDM, ki skriva kompleksnost podatkovnega nivoja in vpeljuje koncept virtualnega zlatega zapisa, ki bistveno poenostavi razvoj novih poslovnih rešitev. Najpomembnejše prednosti uporabe predlaganega pristopa so izboljšana kakovost podatkov, izboljšano upravljanje s strankami, konsistentno poročanje, izboljšano obvladovanje tveganj, lažja izdelava analiz in planiranje, olajšan razvoj novih aplikacij, večje zupanje v rezultate sistemov za podporo odločanju, izboljšana operativna učinkovitost ter nižji stroški.

## 9 VIRI IN LITERATURA

- [1] Andreescu, A., Mircea, M. (2008). Combining Actual Trends in Software Systems for Business Management. 9th International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing (CompSysTech '08). Gabrovo, Bolgarija.
- [2] Brunner, J.-S., Ma, L., Wang, C., Zhang, L., Wolfson, D. C., Pan, Y., Srinivas, K. (2007). Explorations in the use of semantic web technologies for product information management. Proceedings of the 16th International Conference on World Wide Web, Alberta, Canada.
- [3] Butler, D. (2007). MDM as a foundation for SOA, Oracle. Dostopno na: <http://www.oracle.com/master-data-management/mdm-foundation-for-soa-white-paper.pdf>.
- [4] Butler, D., Pollock, J. (2008). Data Management: The Missing Link In Your SOA Strategy. SOA Magazine.
- [5] Cleven, A., Wortmann, F. (2010). Uncovering four strategies to approach master data management. 43rd Hawaii International Conference on System Sciences, 2010.
- [6] Dreibelbis, A., Hechler, E., Milman, I., Oberhofer, M., Van Run, P., Wolfson, D. (2008). Enterprise Master Data Management: An SOA Approach to Managing Core Information, IBM Press.
- [7] Fan, W., Geerts, F., Jia, X. (2008). A Revival of Integrity Constraints for Data Cleaning. Proceedings of the 34th International Conference on Very Large Data Bases, Auckland, New Zealand.
- [8] Fox, C., Levitin, A., Redman, T. (1994). The notion of data quality and its quality dimensions, Information Processing & Management, Letnik 30, številka 1.
- [9] Havey, M. (2008). The SOA Cookbook: Design Recipes for Building Better SOA Processes. Packt Publishing, Birmingham.
- [10] Jennings, F., Jurič, M., Sarang P., Loganathan, R. (2007). SOA Approach to Integration, Packt Publishing, Birmingham.
- [11] Joshi, A. MDM Governance: A Unified Team Approach. Cutter IT journal. Letnik 20, 2007. Str. 30–35.

- [12] Kalogirou, J. Master Data Management Meets SOA. SOA World Magazine. April 2007.
- [13] Krizevnik, M., Jurič, M. (2010). Improved SOA Persistence Architectural Model. ACM SIGSOFT Software Engineering Notes, Letnik 35, številka 3.
- [14] Loshin, D. (2008). Master Data Management, 1. izd., Morgan Kaufmann Publishers, Burlington.
- [15] Mathew, B., Jurič, M., Sarang, P. (2006). Business Process Execution Language for Web Services 2nd Edition, Packt Publishing, Birmingham.
- [16] Neil, S. (2007). A new structure for corporate data. Managing Automation.
- [17] Newman, D. (2005). The Essential Building Blocks for Enterprise Information Management. Gartner Inc.
- [18] Power, D. (2007). Clean, Accurate and Synchronized: Overcoming MDM Challenges. Information Management Special Reports, Februar 2007.
- [19] Sammon, D., Frederic, A., Tadhg, N., Carlsson, S. (2010). Making Sense of the Master Data Management (MDM) concept: Old Wine in New Bottles or New Wine in Old Bottles? Proceeding of the 2010 conference on Bridging the Socio-technical Gap in Decision Support Systems: Challenges for the Next Decade. IOS Press Amsterdam.
- [20] Snow, C. Embrace the role and value of master data management. Manufacturing Business Technology. Letnik 26, 2008. Str. 38–40.
- [21] Taylor, H., Yochem, A., Phillips, L. (2009). Event-Driven Architecture: How SOA Enables the Real-Time Enterprise, Pearson Education.
- [22] Wang, R. (2006). Eleven Entry Points To SOA For Packaged Applications. Forrester Research.
- [23] Your SOA can be DOA without MDM (2007). TIBCO Software Inc. Dostopno na: [http://www.information-management.com/white\\_papers/10001609-1.html](http://www.information-management.com/white_papers/10001609-1.html)

■

Marcel Krizevnik je mladi raziskovalec v laboratoriju za integracijo informacijskih sistemov na Fakulteti za računalništvo in informatiko v Ljubljani, kjer pripravlja doktorsko disertacijo. Raziskovalno dela predvsem na področjih storitveno usmerjenih arhitektur in računalništva v oblaku. Sodeluje v številnih raziskovalnih in aplikativnih projektih za industrijo. Je soavtor knjige WS-BPEL 2.0 for SOA Composite Applications with Oracle SOA Suite 11g (Packt Publishing). Udeležuje se številnih konferenc s področja informatike, na katerih predstavlja teme s področij upravljanja poslovnih procesov in storitveno usmerjenih arhitektur.

■

Matjaz B. Jurič je redni profesor na Fakulteti za računalništvo in informatiko Univerze v Ljubljani, kjer vodi laboratorij za integracijo informacijskih sistemov. Ukvarja se z računalništvom v oblaku, storitvenimi arhitekturami, kompozicijo poslovnih procesov, integracijo, elektronskim poslovanjem, spletnimi storitvami in optimizacijo zmogljivosti. Je avtor oz. soavtor knjig WS-BPEL 2.0 for SOA Composite Applications with IBM WebSphere 7, WS-BPEL 2.0 for SOA Composite Applications with Oracle SOA Suite 11g, Oracle Fusion Middleware Patterns, Business Process Driven SOA, SOA Approach to Integration, Best Practices for SOA-based integration and composite applications development, Business Process Execution Language for Web Services (Packt Publishing), .NET Serialization Handbook, J2EE Design Patterns Applied, Professional J2EE EAI in Professional EJB (Wrox Press) in poglavij v knjigah More Java Gems (Cambridge University Press) ter Technology Supporting Business Solutions (Nova Science Publishers), objavjal je v revijah SOA-Web Services Journal, eAI Journal, Java Report, Java Developers Journal in na konferencah, kot so OOPSLA, Oracle Open World, Java Development, BEA Forum, Wrox Conferences idr. 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. Je predsednik nacionalne komisije za inovacije. Leta 2007 mu je SOA World Journal podelil nagrado za najboljšo knjigo s področja SOA, leta 2010 pa je prejel nagrado za najodmevnejši znanstveni članek s področja storitev iz NMS. Ima naziva Java Champion in Oracle ACE Director.



# Izzivi zunanjega izvajanja informatike

Franc Brcar

Revoz, d. d., Belokranjska 4, 8000 Novo mesto

franc.brkar@renault.com

## Izvleček

Menedžment poslovnih procesov, zunanje izvajanje poslovnih procesov in zunanje izvajanje informatike imajo pomembno vlogo v organizacijah za zagotavljanje konkurenčnosti in konkurenčnih prednosti. Z raziskavo analiziramo vlogo zunanjega izvajanja informatike v slovenskih organizacijah. Z anketo pridobljene podatke smo interpretirali z opisno statistiko, frekvenčno statistiko, Wilcoxonovim testom in odvisnim  $t$ -testom. Z rezultati smo dokazali, da se bo stopnja zunanjega izvajanja informatike v slovenskih organizacijah v prihodnosti povečala. Dokazali smo tudi, da bo ta stopnja večja na področju informacijske strojne opreme in nekoliko nižja na področju programske opreme. Ugotovili smo tudi, da slovenske organizacije sledijo trendom na globalnih trgih in se jim prilagajajo. Zunanje izvajanje informatike ocenjujemo kot pomemben pristop pri povečevanju uspešnosti in učinkovitosti organizacij.

**Ključne besede:** informatika, informacijska tehnologija, menedžment poslovnih procesov, zunanje izvajanje poslovnih procesov, zunanje izvajanje informatike.

## Abstract

### Challenges of IT Outsourcing

Business process management, business process outsourcing and IT outsourcing play an important role in organizations helping them ensure competitiveness and competitive advantages. Our study analyzes the role of IT outsourcing in Slovenian organizations. The data was obtained through a survey and interpreted with descriptive statistics, frequency statistics, Wilcoxon's test and dependent  $t$ -test. With the help of the results of the survey, we have demonstrated that the level of IT outsourcing in Slovenian organizations will increase in the future. We have also demonstrated that this rate will increase significantly in the field of hardware and that it will be slightly higher in the field of software. Our conclusion is that Slovenian organizations follow the trends in global markets and adapt to them. We believe IT outsourcing is an important method for increasing the effectiveness and efficiency of organizations.

**Keywords:** informatics, information technology, business process management, business process outsourcing, information technology outsourcing.

## 1 UVOD

**Informatika ima v organizacijah ključno vlogo za zagotavljanje konkurenčnosti, v nekaterih pa celo omogoča konkurenčno prednost. V začetku so se organizacije odločale predvsem za zunanje izvajanje podpornih aktivnosti, danes pa se mnoge odločajo za zunanje izvajanje temeljnih aktivnosti. To je omogočilo spoznanje, da pogosto specializirani dobavitelj opravi aktivnost ceneje, hitreje in bolje od organizacije – pogosto uporabljamo izraz kupec – in to v primeru temeljnih in podpornih aktivnosti. V preteklosti je bilo veliko nasprotnikov zunanjega izvajanja informatike (*Information Technology Outsourcing – ITO*), danes pa je le-ta pogosto predmet teh korenitih organizacijskih sprememb.**

Menedžment poslovnih procesov (*Business Process Management – BPM*) je najustreznejši pristop za zagotavljanje stalnih konkurenčnih prednosti. Za uspešno

implementacijo sta nujna dva temeljna dejavnika in sicer: 1) ureditev procesov in 2) vključenost zaposlenih (Hung, 2006). Temelj vsega dogajanja v organizacijah so ljudje in procesi. Ena od oblik menedžmenta poslovnih procesov je zunanje izvajanje poslovnih procesov (*Business Process Outsourcing – BPO*) in s tem zunanje izvajanje informatike (*Information Technology Outsourcing – ITO*). Osnovni cilj vsake organizacije mora biti: 1) zmanjševanje stroškov, 2) spoštovanje rokov in (3) izboljševanje kakovosti. Izkoristiti morajo vsa razpoložljiva sredstva za povečevanje uspešnosti in učinkovitosti – tudi zunanje izvajanje poslovnih procesov. Brown in Wilson (2005) sta že leta 2005 ocenila globalni trg zunanjega izvajanja poslovnih procesov na 1.000 milijard ameriških dolarjev letno, napovedala pa sta tudi trikratno povečanje v naslednjih nekaj letih.

Namen in cilj raziskave je: 1) ugotoviti, kakšno je stanje na področju zunanjega izvajanja informatike v slovenskih organizacijah, in 2) ugotoviti, kakšen bo trend na tem področju v prihodnosti. V ta namen definiramo dve hipotezi.

- Hipoteza 1 (H1): Stopnja zunanjega izvajanja informatike se bo v slovenskih organizacijah v prihodnosti povečala.
- Hipoteza 2 (H2): Od vseh aktivnosti informatike se bo stopnja zunanjega izvajanja aktivnosti vzdrževanje osebnih računalnikov v prihodnosti najbolj povečala.

V prvem razdelku smo definirali hipotezo. Drugi razdelek je namenjen pregledu literature in dosedanjim raziskovalnim spoznanjem. V tretjem razdelku opišemo raziskovalno metodo in podamo pojasnila, nujno potrebna za razumevanje rezultatov. Četrty razdelek prikazuje rezultate raziskave in podaja ustrezne komentarje.

## 2 PREGLED LITERATURE

### 2.1 Zunanje izvajanje poslovnih procesov

Power, Desouza in Bonifazi (2006) navajajo spremembe, ki so povzročile razmah zunanjega izvajanja poslovnih procesov: 1) pritisk na zniževanje stroškov; 2) osredinjane na glavno dejavnost; 3) potreba po dostopu do znanja in virov; 4) rast globalnega znanja; 5) razvoj informacijskih tehnologij (*Information Technology – IT*) in 6) globalna razporeditev znanja. Pri tem je pomembno tudi zniževanje stroškov telekomunikacij, višja raven informatizacije, višja izobrazbena raven, pojav mobilne telefonije, elektronske pošte, videa, spletnih konferenčnih sistemov in drugih orodij za delo v skupini. Pomemben je dostop do virov in znanja, lastništvo je drugotnega pomena. Aird in Sappenfield (2009) smatrata tehnologijo za dejavnik, ki omogoča pojav razširjenih globalnih organizacij, za katere je značilno tudi globalno zunanje izvajanje poslovnih procesov (*international outsourcing, global outsourcing, offshore outsourcing, offshoring*).

Dyer, Kale in Singh (2001) so definirali štiri pridobitve strateškega zavezištva: 1) izboljšano upravljanje znanja; 2) boljše zunanja podoba organizacije; 3) boljše notranja koordinacija in 4) boljše razmejitve odgovornosti in povečana učinkovitost reševanja problemov. Click in Duening (2005) sta ugotovila pet razlogov za uvedbo zunanjega izvajanja poslovnih procesov: 1) znižanje stroškov; 2) dostop do znanja

dobaviteljev; 3) povečanje fleksibilnosti na trgu; 4) povečanje prilagodljivosti in razširljivosti organizacije in 5) skrajšanje časa vstopa na trgu. Podobno sta Brown in Wilson (2005) ugotovila deset argumentov za zunanje izvajanje poslovnih procesov, to so: 1) hitrejša prenova procesov; 2) dostop do novih tehnologij; 3) pridobitev denarja s prodajami; 4) sprostitev notranjih virov; 5) ponovni razmislek o problemih; 6) osredinjane na glavno dejavnost; 7) sprostitev denarnih sredstev; 8) znižanje operativnih stroškov; 9) zniževanje rizikov in 10) dostop do novih virov. Abraham in Taylor (1996) smatrata nižje stroške dela v razvijajočih se državah za enega od poglobitnih razlogov za pojav internacionalnega zunanjega izvajanja poslovnih procesov. Nasprotno pa Forbath in Brooks (2007) poudarjata, da lahko organizacija z zunanjim izvajanjem poslovnih procesov doseže konkurenčne prednosti, če se ne osredini samo na zniževanje stroškov, temveč izkoristi le-to za stalno ustvarjanje prihodkov.

Proces zunanjega izvajanja poslovnega procesa je zelo zahteven in tvegan, zato mora organizacija natančno definirati in izvesti vseh sedem korakov celotnega življenjskega cikla (Power, Desouza, & Bonifazi, 2006), ki so: 1) definicija strategije; 2) analiza potreb; 3) ocena dobaviteljev; 4) pogajanja in izbira dobavitelja; 5) izvedba projekta oz. prenos procesa; 6) upravljanje odnosov z dobavitelji in 7) izvajanje stalnega napredka ali konec sodelovanja. Če kateri koli korak izvedemo pomanjkljivo ali ga izpustimo, je zunanje izvajanje poslovnega procesa obsojeno na neuspeh. Pobudnik sprememb mora biti najvišje vodstvo organizacije.

Brown in Wilson (2005) navajata, da bi se organizacije leta 2005 najpogosteje odločile za zunanje izvajanje informatike (55 %), administracije (47 %), logistike (22 %), financ (20 %), človeških virov (19 %), proizvodnje (18 %), odnosov s strankami (15 %), prodaje (13 %), infrastrukture (11 %) in prevozov (9 %); pri tem vsi odgovori ne predstavljajo 100 %, ker so nekateri anketiranci odgovorili na več vprašanj.

Deset najpogostejših napak projekta zunanjega izvajanja poslovnega procesa so (Power, Desouza, & Bonifazi, 2006): 1) nezadostna podpora najvišjega menedžmenta; 2) nezadostno znanje iz metodologij zunanjega izvajanja; 3) pomanjkanje komunikacije znotraj organizacije; 4) nepoznavanje poslovnih tveganj; 5) nezadostno interno znanje; 6) najboljši notranji viri niso angažirani; 7) nespoštovanje postopkov;

8) neupoštevanje kulturnih razlik; 9) nezadosten prenos znanja k dobavitelju in 10) slabo upravljanje odnosov z dobaviteljem. Podobno Shi (2007) navaja probleme na strani organizacije: 1) previsoka pričakovanja glede zniževanja stroškov; 2) nezadostna zrelost procesa, ki je predmet zunanjega izvajanja in 3) pomanjkanje poznavanja in konsenza o ciljnem modelu procesa za zunanje izvajanje. Na drugi strani so vzroki za neuspeh na strani dobavitelja: 1) nezadostne kompetence, 2) prevelika mobilnost zaposlenih pri dobavitelju in 3) nepoznavanje varnostnih zahtev in prakse. Na koncu so lahko vzroki za neuspeh tudi v odnosih med organizacijo in dobaviteljem: 1) pomanjkanje detaljnega opisa projekta uvajanja zunanjega izvajanja poslovnega procesa, 2) jezikovne in kulturne razlike, 3) težave pri prenosu znanja, 4) težave pri namestitvi procesa, 5) različen tempo sledenja tehnološkim spremembam, 6) nekompatibilna organizacijska struktura in 7) izguba kontinuitete zaradi kadrovskih premestitev. Aron, Clemons in Reddi (2005) ugotavljajo, da so različna tveganja dejavniki, ki omejujejo razmah zunanjega izvajanja poslovnih procesov, kar pa lahko zmanjšamo s prenovo poslovnih procesov in s sodelovanjem z več dobavitelji.

Gilley in Rasheed (2000) ugotavljata, da ni povezave med zunanjim izvajanjem poslovnih procesov in sposobnostjo organizacije. Do podobnih ugotovitev sta prišla tudi Bengtsson in Dabhilkar (2009). Le-to potrjujejo kontradiktorne ugotovitve mnogih avtorjev, saj eni dokazujejo precejšnje pozitivne učinke, drugi pa ne. Hirschheim in Lacity (2000) sta v svoji raziskavi pri približno polovici organizacij ugotovila prihranke, pri drugi polovici pa ne. Bettis, Bradley in Hamel (1992) poudarjajo, da je zunanje izvajanje poslovnih procesov samo eno izmed orodij menedžmenta. V raziskavi dokazujejo, da lahko pravilna uporaba poveča konkurenčne prednosti, nepravilna uporaba pa jih zmanjša in v skrajnem primeru celo povzroči propad organizacije. Aron in Singh (2005) v svoji študiji povzemata obstoj treh vzrokov za uspeh oz. neuspeh zunanjega izvajanja poslovnih procesov: 1) izbira pravih procesov, 2) nadzor operativnih in strukturnih tveganj in 3) prilagoditev organiziranosti potrebam.

## 2.2 Zunanje izvajanje informatike

Willcocks in Lacity (1995) pravita, da obstaja veliko podobnosti med zunanjim izvajanjem poslovnih pro-

cesov in zunanjim izvajanjem informatike in navajata pet pomembnih razlik med njima: 1) hiter razvoj informacijskih tehnologij, kar povzroča negotovost, 2) ekonomičnost informacijskih tehnologij se hitro spreminja, npr. razmerje med ceno in zmogljivostjo opreme, 3) informacijska tehnologija in procesi so tako prepleteni, da jih je pogosto težko izolirati; 4) zamenjava informacijske tehnologije ali dobavitelja je pogosto tako draga, da skoraj ni mogoča in 5) mnogo organizacij nima ustreznega znanja iz zunanjega izvajanja informatike. Cheon, Grover in Teng (1995) obravnavajo zunanje izvajanje informatike na podlagi štirih teorij; to so: 1) teorija virov, pri kateri organizacija nima zadostnih virov, 2) teorija odvisnosti od virov, saj je vsaka organizacija odvisna od okolja in dobaviteljev, 3) teorija transakcijskih stroškov, po kateri imajo dobavitelji nižje stroške zaradi ekonomije obsega, in 4) teorija stroškov agencij, ki predvsem poudarja stroške in probleme zaradi odnosov med organizacijo in dobaviteljem, tj. agentom. Cullen in Willcocks (2003) poudarjata, da zunanje izvajanje informatike pomeni strateški menedžment oskrbovanja z informacijskimi storitvami in strateško partnerstvo med organizacijo in dobaviteljem ter definirata osem etap uvedbe: 1) zavrnitev tradicionalnih prepričanj, 2) priprava strategije, 3) izbira pravih procesov oz. aktivnosti, 4) konstrukcija zelenega stanja, 5) izbira dobavitelja ali več dobaviteljev, 6) prenos oz. tranzicija, 7) upravljanje zunanjega izvajanja informatike in 8) ponovni razmislek o vseh možnostih.

Saunders, Gebelt in Hu (1997) trdijo, da je uspešno zunanje izvajanje informatike mogoče tudi, če je le-ta temeljna aktivnost oz. temeljni proces organizacije. Opozoriti je treba, da v tem primeru potrebujemo kakovostno pogodbo, ki bo zmanjšala odvisnost organizacije zaradi strateškega pomena informatike in omogočila kontrolo dobavitelja. V primeru zunanjega izvajanja informatike se lahko njena strateška pomembnost še poveča. Nujne so te pogajalske strategije: 1) v pogodbi je treba natančno opredeliti vse vidike dogovora, 2) vsaka organizacija je specifična in zahteva drugačno pogodbo, 3) poiskati moramo tudi neodvisno izvedensko mnenje, 4) v pogodbo je treba vgraditi opcijo o ponovnih pogajanjih zaradi sprememb, 5) temelj sodelovanja naj bo partnerstvo na podlagi skupnih koristi in 6) pogodba naj bo zasnovana tako, da omogoča primerno svobodo oz. prilagodljivost. Tudi Lacity, Willcocks in Feeny (1995) se strinjajo, da se lahko organizacija odloči za delno ali

celotno zunanje izvajanje informatike, tudi če je le-ta kritična za poslovanje. Ob tem pa mora zadržati teh devet kompetenc (Willcocks & Feeny, 2006; Feeny & Willcocks, 1998): 1) vodenje integracije med informatiko in poslovanjem, 2) poznavanje poslovnih procesov, 3) vzdrževanje odnosov med informatiki in uporabniki, 4) planiranje prihodnje informacijske arhitekture, 5) odprava napak, 6) spremljanje novosti na področju informacijskih tehnologij, 7) reševanje pogodbenih nesoglasij, 8) nadzor uresničevanja pogodb in 9) razvoj dobaviteljev. Rottman in Lacity (2006) sta zbrala petnajst najboljših praks za uspešno zunanje izvajanje informatike: 1) učenje o zunanjem izvajanju s pilotskimi projekti; 2) izbira dobavitelja na podlagi poslovnih ciljev; 3) spodbujanje tekmovalnosti med dobavitelji; 4) sodelovanje z več dobavitelji; 5) zaposlene motiviramo tako, da imajo koristi od zunanjega izvajanja informatike; 6) z delitvijo zunanjega izvajanja zaščitimo intelektualno lastnino; 7) omogočiti moramo vstop dobavitelja v organizacijo; 8) odločitev med pavšalno pogodbo in pogodbo po dejanskih stroških; 9) presoja na podlagi zrelostno-sposobnostnega modela (*Capability Maturity Model – CMM*); 10) standardizacija procesov s CMM zaradi hitrejšega prenosa procesa k dobavitelju; 11) implementacija fleksibilnega CMM modela v organizaciji; 12) pravilna izbira zaposlenih pri dobavitelju za delo pri kupcu oz. v organizaciji; 13) prenos znanja k dobavitelju; 14) spodbujanje prenosa znanja med dobavitelji in 15) uporaba modela uravnoveženih kazalnikov (*Balanced Scorecard – BSC*).

### 3 METODA

Osnovna metoda raziskovanja je anketa, osnovno orodje pa statistična analiza. V ta namen smo izdelali anketni vprašalnik, katerega smo poslali na 484 naključno izbranih organizacij. Skladno z Zakonom o gospodarskih družbah (2006, člen 55) smo izbrali največje po kriterijih povprečnega števila zaposlenih delavcev v poslovnem letu, višini čistega prihodka od prodaje in vrednosti aktive. Tako smo zajeli srednje velike in velike proizvodne in storitvene organizacije, ki predstavljajo populacijo. Na vprašalnik je odgovorilo 80 anketiranih organizacij, kar imenujemo vzorec. Vprašalnik, naslovljen na vodje informatike, smo poslali po klasični pošti. Sestava anketirancev, ki so odgovorili na vprašalnik je takšna: 3,8 % glavni direktor, 11,4 % direktor poslovne funkcije, 59,5 % vodja informatike, 13,9 % vodja oddelka v informa-

tiki in 11,4 % druga funkcija ali položaj. Povprečna delovna doba anketirancev je bila 19,4 leta. Sklepamo lahko, da so imeli ustrezna znanja in kompetence za izpolnitev vprašalnika in da so odgovori kakovostni.

Vprašalnik vsebuje tri vprašanja. Prvo vprašanje je, kje se izvajajo aktivnosti informatike in kje se bodo izvajale v prihodnosti, npr. čez tri leta. To vprašanje je razdeljeno na trinajst aktivnosti: 1) tiskalniki – vzdrževanje tiskalnikov; 2) PC – vzdrževanje osebnih računalnikov; 3) strežniki – vzdrževanje strežnikov; 4) mreža – vzdrževanje mreže; 5) planiranje – strateško planiranje informatike; 6) izobraževanje – izobraževanje uporabnikov; 7) podpora – odprava napak in podpora uporabnikom; 8) integracija – integracija in nadgradnja informacijskih sistemov; 9) OS – vzdrževanje systemske programske opreme; 10) ERP – vzdrževanje aplikacij materialno poslovanje (MRP, ERP); 11) plače – vzdrževanje aplikacij plače in kadri; 12) finance – vzdrževanje aplikacij finance in računovodstvo in 13) CRM – vzdrževanje aplikacij odnosi s strankami (*Customer Relationship Management – CRM*). Na voljo so bili štirje odgovori: 1) aktivnost se izvaja interno v organizaciji; 2) aktivnost se izvaja delno v organizaciji in delno pri dobavitelju; 3) aktivnost se v celoti izvaja pri dobavitelju in 4) organizacija nima te aktivnosti. Drugo vprašanje je funkcija ali položaj anketiranca v organizaciji in tretje vprašanje delovna doba anketiranca v letih. Odgovora na prvi dve vprašanji obravnavamo kot kategorne oz. nominalne (*nominal*) spremenljivke, odgovor na zadnje vprašanje pa je razmernostna (*scale*) spremenljivka.

Podatke, pridobljene z anketnim vprašalnikom, smo obdelali s statistično analizo in sicer z opisno statistiko, frekvenčno statistiko, neparametričnim Wilcoxonovim testom (*non-parametric Wilcoxon's signed-rank test*) in parametričnim odvisnim *t*-testom (*parametric dependent t-test*). Statistično značilnost oz. signifikantnost definiramo pri vrednosti 0,05 oz. pri 5 % in označimo s *p*, pri tem *p* pomeni dvostranski test oz. *p*(dvostransko). Ostale pomembnejše oznake so *M* povprečje (*mean*), *Mdn* mediana (*median*), *SD* standardna deviacija (*standard deviation*), *SE* standardna napaka povprečja (*standard error of mean*), *r* velikost efekta (*effect size*), *z* z-vrednosti (*z-scores*) in *CI* interval zaupanja (*confidence interval*). Številске vrednosti so prikazane z natančnostjo dveh decimalnih mest. Statistike so zapisane skladno s priporočili American Psychological Association (APA). Izraza poslovni proces in aktivnost sta uporabljena kot sinonima.

## 4 REZULTATI RAZISKAVE IN RAZPRAVA

### 4.1 Trend zunanjega izvajanja informatike

Rezultati frekvenčne statistike prvega vprašanja so prikazani v tabeli 1. Vsako vprašanje oz. vsaka aktivnost je razdeljena na trenutno izvajanje in izvajanje v prihodnosti, saj želimo ugotoviti trenutno stanje in predvidevanja anketirancev o stanju v prihodnosti. Organizacije interno najpogosteje izvajajo aktivnosti strateškega planiranja informatike in odpravo napak ter podporo uporabnikom. Ocenjujejo, da sta ti dve aktivnosti zelo pomembni za delovanje organizacije in da bi prišlo do težav, če bi ju prenesli k dobaviteljem. Kljub temu pa se bo delež internega izvajanja zmanjšal zaradi delnega zunanjega izvajanja, ki se bo povečal. Zanimivo je tudi, da se bo delež zunanjega izvajanja v celoti zmanjšal pri aktivnosti podpora. Aktivnosti izobraževanje uporabnikov in vzdrževanje systemske programske opreme se najpogosteje izvajata delno v organizacijah in delno pri dobaviteljih. Razlaga tega dejstva je v tem, da organizacije želijo

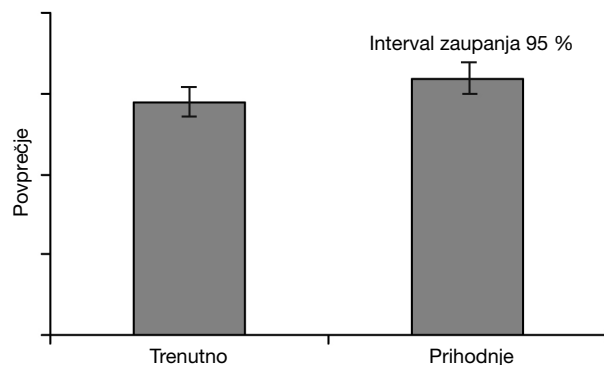
ohranjati nadzor nad tema aktivnostima, po drugi strani pa zahtevata specialistična znanja, katerih pogosto nimajo strokovnjaki v organizacijah. Aktivnosti vzdrževanje tiskalnikov ter vzdrževanje aplikacij plače in kadri se najpogosteje izvajajo pri zunanjih dobaviteljih. Vzdrževanje tiskalnikov je aktivnost, katero lahko relativno preprosto in z malo truda prenesemo k dobaviteljem, zato ta delež ni presenetljiv. Zanimivo pa je, da se organizacije kljub pomislekom zaradi varovanja osebnih podatkov pogosto odločajo za zunanje izvajanje v celoti aktivnosti plače in kadri in ta delež se bo še povečal. Če seštejemo deleže delnega zunanjega izvajanja in zunanjega izvajanja v celoti, vidimo, da imajo velik delež – poleg že omenjenih – še aktivnosti vzdrževanje strežnikov ter vzdrževanje aplikacij finance in računovodstvo. Poleg tega lahko opazimo, da se bo pri večini aktivnosti v prihodnosti povečal delež zunanjega izvajanja v celoti. Aktivnost vzdrževanje aplikacij odnosi s stranikami (CRM) je zaradi svoje specifične namembnosti najmanj razširjena med organizacijami.

Tabela 1: Frekvenčna statistika trenutnega in prihodnjega izvajanja informatike

Aktivnost	Izvajanje	Interno v organizaciji [%]	Delno pri dobavitelju [%]	V celoti pri dobavitelju [%]	Nima te aktivnosti [%]
1) Tiskalniki	Trenutno	21,3	33,8	<b>45,0</b>	0
	Prihodnje	15,0	35,0	<b>50,0</b>	0
2) PC	Trenutno	36,3	33,8	30,0	0
	Prihodnje	22,5	42,5	35,0	0
3) Strežniki	Trenutno	26,3	38,8	33,8	1,3
	Prihodnje	18,8	41,3	38,8	1,3
4) Mreža	Trenutno	37,5	33,8	28,8	0
	Prihodnje	30,0	36,3	33,8	0
5) Planiranje	Trenutno	<b>77,5</b>	18,8	3,8	0
	Prihodnje	<b>73,8</b>	21,3	5,0	0
6) Izobraževanje	Trenutno	28,8	<b>52,5</b>	17,5	1,3
	Prihodnje	22,5	<b>58,8</b>	18,8	0
7) Podpora	Trenutno	<b>58,8</b>	26,3	15,0	0
	Prihodnje	<b>50,0</b>	36,3	13,8	0
8) Integracija	Trenutno	31,3	45,0	23,8	0
	Prihodnje	26,3	46,3	27,5	0
9) OS	Trenutno	25,0	<b>48,8</b>	26,3	0
	Prihodnje	18,8	<b>51,3</b>	30,0	0
10) ERP	Trenutno	23,8	38,8	35,0	2,5
	Prihodnje	20,0	41,3	36,3	2,5
11) Plače	Trenutno	17,5	35,0	<b>47,5</b>	0
	Prihodnje	15,0	36,3	<b>48,8</b>	0
12) Finance	Trenutno	21,3	36,3	42,5	0
	Prihodnje	18,8	38,8	42,5	0
13) CRM	Trenutno	27,5	18,8	17,5	<b>36,3</b>
	Prihodnje	23,8	27,5	23,8	<b>25,0</b>

Z odvisnim *t*-testom naredimo primerjavo povprečij trenutnega izvajanja aktivnosti in izvajanja v prihodnosti. Odgovore ovrednotimo z utežmi. Če organizacija nima aktivnosti, dobi odgovor vrednost 0; če se izvaja interno, dobi vrednost 1; če se izvaja delno interno v organizaciji in delno pri dobavitelju, dobi vrednosti 2, in če se v celoti izvaja pri dobavitelju, dobi vrednosti 3. Vrednosti vseh aktivnosti seštejemo in primerjamo seštevek trenutnega izvajanja s seštevkom predvidenega izvajanja v prihodnosti. Trenutno izvajanje aktivnosti ( $M = 24,51$ ,  $SE = 0,75$ ) se bo v prihodnjih treh letih povečalo ( $M = 25,91$ ,  $SE = 0,75$ ), kar je grafično prikazano na sliki 1. Korelacija med obema spremenljivkama je močna, saj ima korelacijski koeficient vrednosti 0,86 in je statistično značilna pri  $p < 0,001$ . Vrednosti odvisnega *t*-testa so zapisane v tabeli 2, samo statistiko pa zapišemo v obliki  $t(79) = 3,55$ ,

razlika je statistično značilna pri  $p < 0,001$  in  $r = 0,37$ . S tem je potrjena prva hipoteza (H1), da se bo stopnja zunanjega izvajanja informatike v slovenskih organizacijah v prihodnosti povečala.



Slika 1: Trenutno in prihodnje izvajanje informatike

Tabela 2: Primerjava med trenutnim in prihodnjim izvajanjem

Izvajanje	M	SD	SE	95 % CI razlike		t	df	Sig.
				sp	zg			
Prihodnje vs. trenutno	1,40	3,52	0,39	0,62	2,18	3,55	79	0

## 4.2 Izbira aktivnosti za zunanje izvajanje informatike

Ugotoviti želimo, katere aktivnosti bodo organizacije v prihodnosti – v naslednjih treh letih – najpogosteje prenašale k zunanjim dobaviteljem oz. katere aktivnosti bodo najpogosteje doživele organizacijsko spremembo zunanjega izvajanja; katerim aktivnostim se bo najbolj povečala stopnja zunanjega izvajanja. Odgovore na prvo vprašanje obravnavamo kot nominalno spremenljivko, zato uporabimo Wilcoxonov test. Rezultati so prikazani v tabeli 3. Organizacije, ki že imajo zunanje izvajanje aktivnosti vzdrževanje mreže ter odprava napak in podpora uporabnikom, bodo za ti dve aktivnosti najpogosteje ukinile zunanje izvajanje. To je posledica razočaranja nad tem in dokaz, da je zunanje izvajanje informatike zelo zahteven organizacijski proces. Če ta ni temeljito izveden in upravljan, lahko pride od težav. Nasprotno vidimo, da se bo pri vseh aktivnostih stopnja zunanjega izvajanja povečala, najbolj pri aktivnostih vzdrževanje osebnih računalnikov in vzdrževanje aplikacij odnosi s strankami (CRM). Podobno kot aktivnost

vzdrževanje tiskalnikov je tudi aktivnost vzdrževanje osebnih računalnikov relativno nezahtevna za prinos k dobavitelju, število osebnih računalnikov v organizacijah je praviloma veliko in količina dela je velika, vendar manj zahtevna. Zaradi vsega tega je ta aktivnost primerna za zunanje izvajanje. Velik delež aktivnosti vzdrževanje aplikacij odnosi s strankami (CRM) je posledica uvajanja v organizacijah, ki še nimajo teh informacijskih sistemov. Najmanjšo spremembo pričakujemo pri aktivnosti vzdrževanje aplikacij finance in računovodstvo.

Wilcoxonov test aktivnosti vzdrževanje osebnih računalnikov zapišemo v naslednji obliki: trenutna stopnja zunanjega izvajanja je ( $M = 1,94$ ,  $Mdn = 2,00$ ) in se bo v prihodnosti povečala ( $M = 2,13$ ,  $Mdn = 2,00$ ),  $z = 3,09$ ,  $p < 0,001$  – razlika je statistično značilna,  $r = 0,85$ . Podobno lahko statistiko zapišemo tudi za vse ostale aktivnosti. S tem je potrjena druga hipoteza (H2), da se bo stopnja zunanjega izvajanja aktivnosti vzdrževanje osebnih računalnikov od vseh aktivnosti v informatiki v slovenskih organizacijah v prihodnosti najbolj povečala.

Tabela 3: Wilcoxonov test med trenutnim in prihodnjim izvajanjem

Prihodnje vs. trenutno	Negativni red [n]	Pozitivni red [n]	Enakost [n]	Skupaj [n]	Wilcoxonov test	
					z	Sig.
1) Tiskalniki	1	10	69	80	-2,13	0,03
2) PC	1	<b>16</b>	63	80	-3,09	0
3) Strežniki	1	12	67	80	-2,50	0,01
4) Mreža	<b>2</b>	11	67	80	-2,50	0,01
5) Planiranje	1	5	74	80	-1,63	0,10
6) Izobraževanje	0	7	73	80	-2,53	0,01
7) Podpora	<b>2</b>	9	69	80	-1,60	0,11
8) Integracija	1	9	70	80	-1,94	0,05
9) OS	1	10	69	80	-2,14	0,03
10) ERP	1	6	73	80	-1,26	0,21
11) Plače	1	5	74	80	-1,00	0,32
12) Finance	1	4	<b>75</b>	80	-0,71	0,48
13) CRM	0	<b>13</b>	67	80	-3,22	0

Negativni red: prihodnje < trenutno; pozitivni red: prihodnje > trenutno; enakost: prihodnje = trenutno.

Vse odgovore smo ovrednotili z utežmi. Zaradi tega lahko kljub nominalnim spremenljivkam – obravnavamo jih kot razmernostne – naredimo parametrični odvisni *t*-test razlik med trenutnim in prihodnjim izvajanjem posameznih aktivnosti. Namen tega testa je potrditev rezultatov, dobljenih z Wilcoxonovim testom. Rezultati so zelo podobni. Povprečja razlik (*M*) so nominalno majhna, vendar pozitivna, kar pomeni povečano stopnjo zunanjega izvajanja. Tudi s tem testom potrdimo, da je razli-

ka povprečij največja pri aktivnostih vzdrževanje osebnih računalnikov in vzdrževanje aplikacij odnosi s strankami (CRM), najmanjša pa pri aktivnosti vzdrževanje aplikacij finance in računovodstvo, pri katerih razlika tudi ni statistično značilna  $p > 0,05$ . Statistiko odvisnega *t*-testa za aktivnost vzdrževanje osebnih računalnikov zapišemo  $t(79) = 3,32, p < 0,001$  – razlika povprečij je statistično značilna,  $r = 0,35$ . Ostali rezultati *t*-testa so zbrani v tabeli 4. Druga hipoteza (*H2*) je s tem dodatno potrjena.

Tabela 4: Odvisni t-test med trenutnim in prihodnjim izvajanjem

Prihodnje vs. trenutno	M	SD	SE	95 % CI razlike		t	df	Sig.
				sp	zg			
1) Tiskalniki	0,11	0,45	0,05	0,01	0,21	2,24	79	0,03
2) PC	<b>0,19</b>	0,51	0,06	0,07	0,30	3,32	79	0
3) Strežniki	0,12	0,43	0,05	0,03	0,22	2,59	79	0,01
4) Mreža	0,12	0,43	0,05	0,03	0,22	2,59	79	0,01
5) Planiranje	0,05	0,27	0,03	-0,01	0,11	1,65	79	0,10
6) Izobraževanje	0,10	0,34	0,04	0,02	0,18	2,62	79	0,01
7) Podpora	0,07	0,41	0,05	-0,02	0,17	1,62	79	0,11
8) Integracija	0,09	0,40	0,04	0	0,18	1,98	79	0,05
9) OS	0,10	0,41	0,05	0,01	0,19	2,19	79	0,03
10) ERP	0,05	0,35	0,04	-0,03	0,13	1,27	79	0,21
11) Plače	0,04	0,33	0,04	-0,04	0,11	1,00	79	0,32
12) Finance	<b>0,02</b>	0,32	0,03	-0,05	0,10	0,71	79	0,48
13) CRM	<b>0,32</b>	0,81	0,09	0,14	0,50	3,60	79	0

## 5 SKLEP

Menedžment poslovnih procesov, zunanje izvajanje poslovnih procesov in zunanje izvajanje informatike so področja, ki postajajo čedalje pomembnejša za povečanje uspešnosti in učinkovitosti organizacij. Poleg procesov so ključni tudi zaposleni s svojim fizičnim in umskim potencialom; še posebno je pomembna vloga vodij, za katere Kotter (2001) pravi, da morajo biti sposobni: 1) razvijati vizijo, 2) usklajevati sodelavce in 3) jih motivirati ter navdihovati.

Zunanje izvajanje informatike se uveljavlja zadnjih deset let. Zametki so v klasičnih vzdrževalnih pogodbah, pogumnejši začetki so bili z npr. najemom tiskalnikov, danes pa se že mnoge organizacije odločajo za delno ali celotno zunanje izvajanje informatike. Miti, da je zunanje izvajanje informatike neprimerno zaradi strateškega pomena in zaradi zagotavljanja konkurenčnih prednosti na trgu, se rušijo. Tako kot Nike ne izdeluje športnih copat, kot Amazon ne tiska knjig niti jih ne distribuira, kot mnoga farmacevtska podjetja naročajo razvoj zdravil na univerzah, inštitutih ali v specializiranih laboratorijih, kot mnoge organizacije iz svojih vrednostnih verig izločajo temeljne procese, tako se lahko vsaka organizacija odloči za delno ali popolno zunanje izvajanje informatike.

Zunanje izvajanje informatike je zelo zahteven proces, mnogo poskusov se je končalo neuspešno. Pomembno je, 1) da temeljito izvedemo projekt uvedbe zunanjega izvajanja informatike in 2) da dosledno skrbimo za odnose z dobavitelji.

S prvo hipotezo (H1) – da se bo stopnja zunanjega izvajanja informatike v slovenskih organizacijah v prihodnosti povečala – smo dokazali, da slovenske organizacije sledijo dogajanju na globalnih trgih in da se prilagajajo novim ekonomskim razmeram. Večina organizacij bo povečala stopnjo zunanjega izvajanja, nekatere pa se bodo celo odločile za celotno zunanje izvajanje informatike. Na prvi pogled to pomeni ukinitje delovnih mest, vendar to na drugi strani pomeni tudi odpiranje novih delovnih mest pri dobaviteljih. Pomembno je, da oba partnerja – kupec in dobavitelj – napredujeta in da sta ekonomsko uspešna.

Z drugo hipotezo (H2) – da se bo stopnja zunanjega izvajanja aktivnosti vzdrževanje osebnih računalnikov od vseh aktivnosti v informatiki v slovenskih organizacijah v prihodnosti najbolj povečala – smo dokazali, da so najbolj aktivna področja vzdrževanje tiskalnikov, osebnih računalnikov, strežnikov in

mrež, torej na področju informacijske strojne opreme. Področje programske opreme nekoliko zamuja; ugotovili pa smo tudi, da se mnoge organizacije pogumno odločajo za zunanje izvajanje področja plač, človeških virov, financ in računovodstva, torej področij, ki so najzahtevnejša s področja varnosti in varovanja podatkov.

Organizacije bodo v prihodnosti za povečanje uspešnosti in učinkovitosti uporabljale vsa razpoložljiva sredstva in tudi zunanje izvajanje informatike. Slovenske organizacije tradicionalno sodelujejo predvsem z domačimi dobavitelji, vendar lahko pričakujemo, da se bo povečeval delež tujih dobaviteljev. Z raziskavo smo dokazali, da se bo v prihodnosti povečevalo zunanje izvajanje informatike na slovenskem trgu.

## 6 LITERATURA

- [1] Abraham, K. G. & Taylor, S. K. (1996). Firms' Use of Outside Contractors: Theory and Evidence. *Journal of Labor Economics*, 14(3), 394–424.
- [2] Aird, C. L. & Sappenfield, D. (2009). IT the »Enabler« of Global Outsourcing. *Financial Executive*, 25(5), 62–63.
- [3] Aron, R., Clemons, E. K. & Reddi, S. (2005). Just Right Outsourcing: Understanding and Managing Risk. *Journal of Management Information Systems*, 22(2), 37–55.
- [4] Aron, R. & Singh, J. V. (2005). Getting Offshoring Right. *Harvard Business Review*, 83(12), 135–143.
- [5] Bengtsson, L. & Dabhiikar, M. (2009). Manufacturing outsourcing and its effect on plant performance—lessons for KIBS. *Journal of Evolutionary Economics*, 19(2), 231–257.
- [6] Bettis, R. A., Bradley, S. P. & Hamel, G. (1992). Outsourcing and industrial decline. *Academy of Management Executive*, 6(1), 7–22.
- [7] Brown, D. & Wilson, S. (2005). *The Black Book of Outsourcing: How to Manage the Changes, Challenges, and Opportunities* (reprint 2008). Hoboken, NJ: John Wiley & Sons.
- [8] Cheon, M. J., Grover, V. & Teng, J. T. C. (1995). Theoretical perspectives on the outsourcing of information systems. *Journal of Information Technology*, 10(4), 209–219.
- [9] Click, R. L. & Duening, T. N. (2005). *Business Process Outsourcing: The Competitive Advantage*. Hoboken, NJ: John Wiley & Sons.
- [10] Cullen, S. & Willcocks, L. (2003). *Intelligent IT outsourcing: eight building blocks to success* (reprint 2006). Oxford, UK: Elsevier Butterworth-Heinemann.
- [11] Dyer, J. H., Kale, P. & Singh, H. (2001). How to Make Strategic Alliances Work. *MIT Sloan Management Review*, 42(4), 37–43.
- [12] Feeny, D. F. & Willcocks, L. P. (1998). Core IS Capabilities for Exploiting Information Technology. *Sloan Management Review*, 39(3), 9–21.
- [13] Forbath, T. & Brooks, P. (2007). Global Service Providers: Outsourcing's Next Wave. *Financial Executive*, 23(3), 21–24.
- [14] Gilley, K. M. & Rasheed, A. (2000). Making More by Doing Less: An Analysis of Outsourcing and its Effects on Firm Performance. *Journal of Management*, 26(4), 763–790.
- [15] Hirschheim, R. & Lacity, M. (2000). The Myths and Realities of Information Technology Insourcing. *Communications of the ACM*, 43(2), 99–107.



- [16] Hung, R. Y.-Y. (2006). Business Process Management as Competitive Advantage: a Review and Empirical Study. *Total Quality Management* 17(1), 21–40.
- [17] Kotter, J. P. (2001). What Leaders Really Do. *Harvard Business Review*, 79(11), 85–96.
- [18] Lacity, M. C., Willcocks, L. P. & Feeny, D. F. (1995). IT Outsourcing: Maximize Flexibility and Control. *Harvard Business Review*, 73(3), 84–93.
- [19] Power, M. J., Desouza, K. C. & Bonifazi, C. (2006). *The Outsourcing Handbook: How to Implement a Successful Outsourcing Process* (reprint 2008). London, UK: Kogan Page.
- [20] Rottman, J. W. & Lacity, M. C. (2006). Proven Practices for Effectively Offshoring IT Work. *MIT Sloan Management Review*, 47(3), 56–63.
- [21] Saunders, C., Gebelt, M. & Hu, Q. (1997). Achieving Success in Information Systems Outsourcing. *California Management Review*, 39(2), 63–79.
- [22] Shi, Y. (2007). Today's Solution and Tomorrow's Problem: The Business Process Outsourcing Risk Management Puzzle. *California Management Review*, 49(3), 27–44.
- [23] Willcocks, L. P. & Feeny, D. (2006). IT Outsourcing and Core IS Capabilities: Challenges and Lessons at DuPont. *Information Systems Management*, 23(1), 49–56.
- [24] Willcocks, L. P. & Lacity, M. C. (1995). Information Systems Outsourcing in Theory and Practice. *Journal of Information Technology*, 10(4), 203–207.
- [25] Zakon o gospodarskih družbah. (2006). Uradni list RS. Št. 42/2006, 19. april 2006.

■

Franc Brčar je univerzitetni diplomirani inženir strojništva in magister informacijsko-upravljaljskih znanosti. Zaposlen je v podjetju Revoz, d. d. Na začetku je delal kot specialist na področju operacijskih sistemov in baz podatkov. Sledilo je delo na področju uvajanja in vzdrževanja sistemov za računalniško konstruiranje in celovitih rešitev ERP. V zadnjem obdobju se ukvarja s splošnim menedžmentom, menedžmentom informacijskih sistemov, menedžmentom poslovnih procesov in menedžmentom kakovosti.

# Obvladovanje odpora pri projektih informacijskih tehnologij

Simon Vrhovec, Rok Rupnik

Fakulteta za računalništvo in informatiko, Univerza v Ljubljani, Tržaška 25, 1000 Ljubljana  
simon.vrhovec@fri.uni-lj.si; rok.rupnik@fri.uni-lj.si

## Izvleček

Spremembe informacijskih sistemov se običajno implementirajo skozi projekte in programe informacijskih tehnologij. Spremembe informacijskih sistemov so tudi organizacijske spremembe, zaradi katerih se lahko pojavi odpor, ki pa ne ovira vedno procesa spreminjanja. Ob ustreznem obvladovanju odpora in upoštevanju priporočil za ravnanje z njim se lahko celo poveča zavezanost spremembam. Pri obravnavi odpora je pomembno razlikovanje med oblikami odpora in vzroki za njegov pojav, pri čemer so lahko v pomoč v članku predlagani ključni elementi obvladovanja odpora. Na podlagi sorodnih raziskav je v članku predlagan generični model obvladovanja odpora. Članek vsebinsko sklone predlog umestitve obvladovanja odpora v standard za vodenje projektov PMI.

**Ključne besede:** odpor do sprememb, obvladovanje odpora, ravnanje z odporom, razreševanje odpora, informacijski sistemi, vodenje projektov, obvladovanje programov.

## Abstract

### Resistance Management in IT projects

IS changes are commonly implemented in IT projects and programs. IS changes are also organizational changes and as such can cause resistance. Resistance is not necessarily the enemy of change. It can strengthen the commitment to change when managed well and when recommendations for handling resistance are taken into account. It is important to distinguish between forms and causes for resistance when addressing it. The key elements of resistance management proposed in this paper can help identify the causes for resistance. This paper also proposes a generic resistance management model based on related research and its placement within the PMI standard for project management.

**Keywords:** resistance to change, resistance management, resistance handling, resistance resolution, information systems, project management, program management.

## 1 UVOD

**Področje informacijskih tehnologij (IT) je v podjetjih pridobilo velik pomen, informacijski sistemi (IS) pa so dandanes nepogrešljivi del velike večine organizacij. Spremembe informacijskih sistemov se po navadi implementirajo v projektih in programih informacijskih tehnologij. Kljub splošni razširjenosti in izkušnjam se velik del obsežnejših projektov informacijskih tehnologij – med 60 in 70 odstotki – konča neuspešno (Pan, Hackney & Pan, 2008; Washington & Hacker, 2005). Med pomembnimi dejavniki uspeha se pojavlja tudi odpor uporabnikov do sprememb informacijskih sistemov (Aladwani, 2001; Lorenzi & Riley, 2003).**

Odpor je kompleksen pojav, ki je navzoč pri vseh spremembah (Henry, 1997; Proctor & Doukakis, 2003). Do izraza prihaja pri obsežnejših spremembah, saj je sorazmeren z njihovim obsegom (Pardo del Val & Martínez Fuentes, 2003). Odpor je simptom

pomembnejših problemov v ozadju (Waddell & Sohal, 1998). Pojavlja se v različnih oblikah in jakostih, od nedolžnih in težko opaznih simptomov do skrajnih oblik odpora, npr. stavk po vsej državi, kar se je zgodilo leta 1993 v Franciji zaradi novega računalniškega rezervacijskega sistema francoskih železnic (Pan, Hackney, & Pan, 2008).

Odpor sam pa ni edini problem, ki se pojavlja pri spreminjanju informacijskega sistema. Organizacije pogosto nimajo potrebnega strokovnega kadra za implementacijo sprememb informacijskih sistemov, zato jih izvajajo v sodelovanju z zunanjimi izvajalci. Pri tem so nekoliko problematični obstoječi standardi na področju projektnega menedžmenta (PM), saj tega področja ne obravnavajo z zornega kota matične organizacije, ki uporablja izide projektov informacijske tehnologije.

Članek ima takšno strukturo: v drugem razdelku je na kratko predstavljeno ozadje spreminjanja organizacij in opredeljeni relevantni pojmi, v tretjem razdelku je predstavljen pojem odpora, kako ga je treba razumeti in njegovi pozitivni učinki. V četrtem razdelku so obravnavani vzroki za pojav odpora, v petem pa priporočila za ravnanje z odporom. V šestem razdelku so opredeljeni ključni elementi obvladovanja odpora (KEOO), ki so temelj za obravnavo vzrokov in ravnanje z odporom. V sedmem razdelku so predstavljene sorodne raziskave, na katerih je utemeljen v članku predlagani model obvladovanja odpora (MOO). Ta je predstavljen v osmem razdelku, v katerem je predlagana tudi dopolnitev standardov PM in umestitev obvladovanja odpora v standard za vodenje projektov PMI.

## 2 SPREMINJANJE ORGANIZACIJ

V sodobnih organizacijah so spremembe stalnica (Washington & Hacker, 2005), njihova frekvenca pa je v informacijski družbi velika (Arciniega & González, 2008; Chen & Wang, 2007). Organizacije se spreminjajo, da bi se prilagodile zahtevam okolja in povečale svojo konkurenčnost (Arciniega & González, 2008; Pardo del Val & Martínez Fuentes, 2003). Preživijo lahko samo tiste organizacije, ki so se sposobne dovolj hitro prilagajati dinamiki spreminjajočega se okolja (Gareis, 2010). Sprememba preprosto pomeni delati nekaj novega ali drugače, pri tem pa se pojavljajo pritiski v želji po pridobitvi podpore zanjo (Hultman, 2003).

Spremembe se v današnjih organizacijah implementirajo pretežno skozi projekte, obsežnejše spremembe pa tudi skozi programe. Program je skupina medsebojno povezanih in koordinirano vodenih projektov. Njegov namen je zagotoviti koordinacijo vseh projektov programa in s tem pridobiti učinke, ki jih ni mogoče pridobiti z vodenjem posameznih projektov. Programi vključujejo tudi delo zunaj obsega projektov, kar sodi v področje obvladovanja programa (PMI, 2008; PMI, 2009c).

Velika večina sprememb v organizacijah se implementira skozi spremembo njihovih informacijskih sistemov (Lorenzi & Riley, 2003). V sodobnih organizacijah obstaja tesna povezanost med njenim informacijskim sistemom in ljudmi, ki so del te organizacije. Vsaka sprememba informacijskega sistema se zaradi tega prej ali slej odrazi v organizaciji in obratno, zato je pomembno razumevanje tega razmerja.

Da je vsaka sprememba informacijskega sistema tudi organizacijska sprememba, potrjujeta dejstva, da je primarni cilj obstoja informacijskega sistema informiranje ljudi (Yeo, 2002) in da informacijski sistem ne deluje brez ljudi (Lorenzi & Riley, 2003).

Projekt oz. program informacijske tehnologije je projekt oz. program, katerega vsaj eden od ciljev je sprememba informacijskega sistema neke organizacije. Portfelj informacijske tehnologije neke organizacije obsega vse njene projekte in programe informacijske tehnologije. Ob uvedbi portfelja informacijske tehnologije je tako mogoče obvladovati vse spremembe informacijskega sistema na izvedbeni ravni.

## 3 ODPOR DO SPREMEMB

Odpor je pojav, ki v prvi vrsti ovira implementacijo spremembe in povzroča nepredvidene stroške ter zamude. Po drugi strani pa gre za naravni pojav (Bovey & Hede, 2001), ki je v določeni meri navzoč pri vseh spremembah. Lastnosti odpora strokovnjakom omogočajo njegovo predvidevanje, spremljanje in razreševanje.

Odpor je protiutež spremembam in poskuša obdržati obstoječe ravnotežje v organizaciji (*status quo*) (Henry, 1997; Pardo del Val & Martínez Fuentes, 2003; Proctor & Doukakis, 2003; Waddell & Sohal, 1998). Ob primerjavi odpora z vztrajnostjo (Pardo del Val & Martínez Fuentes, 2003; Waddell & Sohal, 1998) je mogoče ugotoviti, da ga ni priporočljivo prezreti za učinkovito upravljanje organizacije. Vztrajnosti se vozniki motornih koles zelo dobro zavedajo, saj jim omogoča stabilnost tako pri vožnji naravnost kot v ovinkih. Po drugi strani pa ravno vztrajnost ovira spremembo naklona motornega kolesa, npr. pri vходу v ovinek. Nizka hitrost zahteva prefinjen nagib, morda le motornega kolesa, visoka pa radikalno spremembo težišča z nagibom telesa krepko v notranjost ovinka. Spremembe naklona motornega kolesa so primerljive z organizacijskimi spremembami. Organizacije potrebujejo stabilnost med spremembami ravno toliko kot motoristi med ovinki. Poznavanje odpora je prvi pogoj za njegovo obvladovanje, ki lahko organizacijam omogoči omenjeno stabilnost.

### 3.1 Razumevanje odpora

Razumevanje odpora se je od začetka njegovega obravnavanja do danes spreminjalo. Sprva je bil odpor označen kot izvor različnih mnenj, ki se pojavlja pri osebah, ki lastne interese postavljajo nad splo-

šne in dobro organizacije (Waddell & Sohal, 1998). Odpor naj bi tako povzročal nezaželene konflikte (Waddell & Sohal, 1998). Takšna obravnava odpora je na prvi pogled smiselna, saj odpor v svojem bistvu povzroča le nepotrebne stroške in zamude pri implementaciji sprememb. Po drugi strani pa namiguje tudi na dejstvo, da je odpor lahko izvor novih idej in vir inovativnosti pri planiranju in implementaciji sprememb.

Kasnejše raziskave so poglobile razumevanje odpora z aplikacijo idej s področij psihologije, sociologije in antropologije (Waddell & Sohal, 1998). Odpor so začeli obravnavati kot kompleksen in večplasten pojav (Waddell & Sohal, 1998). Oblikovalo se je enotno mnenje, da odpor in konflikti, ki jih povzročata le-ta, niso nujno sovražniki sprememb (Waddell & Sohal, 1998). Z ustreznim obvladovanjem odpora in koriščenjem njegovih pozitivnih učinkov lahko odpor celo občutno pripomore k ugodnejšim posledicam sprememb (Waddell & Sohal, 1998).

Odpor ima pogosto neupravičeno negativno konotacijo (Hultman, 2003). Že domneva, da je dobra sprememba tista, ki ni naletela na odpor, prikazuje odpor v negativni luči (Waddell & Sohal, 1998). Lahko je spremeniti nekaj, za kar se nihče ne zanima, občutno težje pa je spremeniti nekaj, za kar se ljudje zanimajo ali se začnejo zanimati (Lorenzi & Riley, 2003).

Kot primer koristnosti in primernosti odpora bi lahko navedli sprejemanje novega zakona v državnem zboru. Če v državnem zboru sprejmejo zakon brez odpora, to še ne pomeni, da bo sprejeti zakon tudi dober, temveč, da zanj ni pretiranega zanimanja. Nasprotovanje opozicije in burna razprava v državnem zboru bi podvrgla zakon preizkušnji, iz katere bi se gotovo rodil bolj robusten in življenjski zakon (Waddell & Sohal, 1998).

Odpor je mogoče obravnavati kot obliko konflikta (Fiedler, 2010), zato je mogoče na odpor aplicirati nekatere njihove značilnosti. Vzroki za konflikte se lahko med njegovim potekom spreminjajo (Pinto, 1996), kar je mogoče ugotoviti tudi za odpor. Ker se lahko tudi vzroki za pojav odpora spreminjajo med njegovim potekom, jih je treba obravnavati s komponento dinamičnosti.

Odpor se lahko pojavi tudi s časovnim zamikom. Obsežnejše spremembe informacijskega sistema zahtevajo določeno dobo uvajanja uporabnikov. Obstaja možnost, da se odpor pojavi šele po uvajanju, ki je

lahko dolgotrajno, ko se razjasni obseg spremembe in njene značilnosti.

### **3.2 Pozitivni učinki odpora**

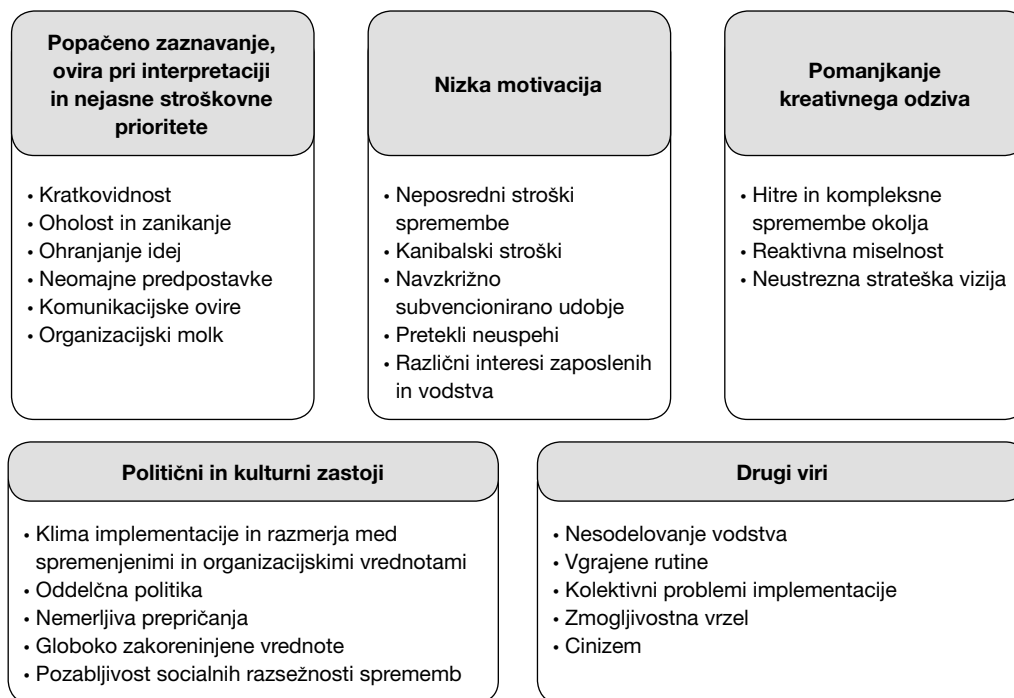
Odpor ima lahko pri naporih spreminjanja pozitivne učinke predvsem pri motivaciji in inovativnosti (Waddell & Sohal, 1998). Na prvi pogled je morda paradoksalna trditev, da odpor lahko pripomore k motivaciji za spremembo. K temu levji delež prispeva pogosta predpostavka, da je odpor nekaj, kar samo ovira spremembe.

V apatičnih in pasivnih delovnih okoljih je težko implementirati dobre spremembe. Spremembe tipično niso kreativne in so slabo implementirane (Waddell & Sohal, 1998). Odpor in konflikti, ki jih povzročajo spremembe, spodbujajo njihovo resno in podrobnejšo obravnavo (Waddell & Sohal, 1998). Pri tem je treba paziti na ustrezno ravnovesje, saj lahko postane prevelik obseg konfliktov bolj problematičen od prvotnih problemov (Waddell & Sohal, 1998). Omenjeno ravnovesje se omenja tudi kot optimalna raven motivacije, ki služi procesu spreminjanja in ugodno vpliva na njegov izid (Waddell & Sohal, 1998).

Odpor spodbuja iskanje alternativnih metod in izidov, da bi se uskladila konfliktna mnenja. Odpor zato postane kritičen vir inovacij v procesu spreminjanja, saj se pretehta in oceni več možnosti (Waddell & Sohal, 1998). Določene rešitve vodstvo pogosto favorizira in zato niso temeljiteje obravnavane. Pod takimi pogoji je sprejemanje vgrajeno, rast in spreminjanje organizacij pa sta omejena na sposobnosti predlagatelja sprememb (Waddell & Sohal, 1998). Raziskave so pokazale tudi, da je veliko menedžerskih odločitev neracionalnih. Menedžerji ne obravnavajo dovolj alternativnih rešitev problema, niti niso te rešitve ocenjene adekvatno (Waddell & Sohal, 1998). Odpor je torej pojav, ki nam preprečuje, da bi se zavzeli za vsako neumnost, ki nam pade na pamet (Waddell & Sohal, 1998).

## **4 VZROKI ZA POJAV ODPORA**

V literaturi se pojavlja več poskusov strukturiranja vzrokov za pojav odpora. Slika 1 prikazuje poskus strukturiranja vzrokov na podlagi njihove medsebojne sorodnosti. Vzroki so strukturirani v pet skupin, katerih celovitost je bila tudi dokazana v empirični raziskavi (Pardo del Val & Martínez Fuentes, 2003).

Slika 1: **Vzroki za pojav odpora**<sup>1</sup>

V omenjeni raziskavi (Pardo del Val & Martínez Fuentes, 2003) je bilo ugotovljeno, da so nekateri vzroki bolj korelirani z obsegom spremembe kot drugi. Ti vzroki so lahko problematični pri obsežnejših spremembah, kljub temu da pri manj obsežnih niso. Med omenjenimi vzroki izstopajo predvsem:

- **globoko zakoreninjene vrednote:** gre za vzrok, ki je sorazmeren z obsegom spremembe. Obstoj globoko zakoreninjenih vrednot in čustvene lojalnosti do načina dela (Pardo del Val & Martínez Fuentes, 2003; Rumelt, 1995) je med najpogostejšimi in najvplivnejšimi vzroki;
- **zmogljivostna vrzel:** vzrok je sorazmeren z obsegom spremembe. Zmogljivostna vrzel se pojavlja, ko imajo posamezniki na voljo premalo kapacitet ali sposobnosti za implementacijo spremembe (Fiedler, 2010; Pardo del Val & Martínez Fuentes, 2003; Proctor & Doukakis, 2003; Rumelt, 1995);
- **oddelčna politika:** vzrok, ki nastopa ne glede na obseg spremembe. Odpor se pojavlja pri oddelkih, katerim sprememba ogroža status ali celo obstoj (Pardo del Val & Martínez Fuentes, 2003; Rumelt, 1995). Večja pripadnost posameznim oddelkom

kot organizaciji kot celoti lahko ovira tudi komunikacijo med posameznimi oddelki (Mihelčič, 2003);

- **kanibalski stroški:** vzrok, ki nastopa ne glede na obseg spremembe. Sprememba lahko izboljša situacijo na eni strani in jo hkrati poslabša na drugi, kar pomeni neko vrsto žrtvovanja (Pardo del Val & Martínez Fuentes, 2003; Rumelt, 1995). Pri tem gre lahko za racionalno podlago ali pa odraža interese določenih skupin (Rumelt, 1995);
- **navzkrižno subvencionirano udobje:** vzrok, ki nastopa ne glede na obseg spremembe. Potreba po spremembah je kompenzirana z visokimi donosi iz drugih dejavnikov, ki niso posledica spremembe (Pardo del Val & Martínez Fuentes, 2003; Rumelt, 1995);
- **nemerljiva prepričanja:** vzrok, ki nastopa ne glede na obseg spremembe. Odpor se pojavi zaradi močnega in neizogibnega nestrinjanja med skupinami glede same narave problema in posledično alternativnih rešitev (Pardo del Val & Martínez Fuentes, 2003; Rumelt, 1995);
- **pomanjkanje kreativnega odziva:** vzroki iz te skupine prihajajo do izraza pri obsežnejših spremembah. Gre za skupino vzrokov, ki so povezani s pomanjkanjem kreativnosti pri iskanju ustrezne strategije izhoda iz težav.

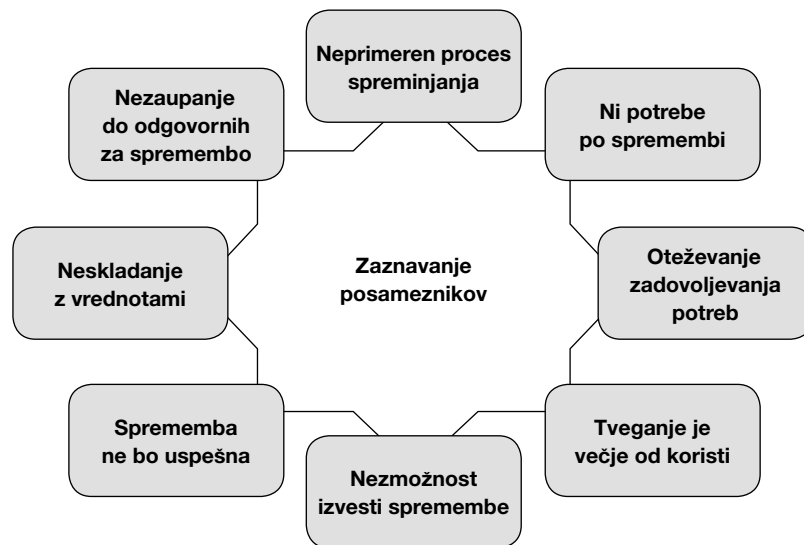
<sup>1</sup> Povzeto po Pardo del Val in Martínez Fuentes (Pardo del Val & Martínez Fuentes, 2003).

Vzroki, strukturirani glede na medsebojno sorodnost, niso najbolj praktični za identifikacijo v konkretnih primerih. Večino informacij o odporu pridobimo neposredno od posameznikov, zato je primernejša obravnavna vzrokov z njihovega vidika.

Posamezniki si ustvarijo lasten subjektivni pogled na realnost na podlagi dejstev, prepričanij in vrednot (Hultman, 2003). Zaznava posameznikov tako določa njihovo realnost (Pinto, 1996), lahko pa bi celo rekli, da vsak posameznik živi v lastnem svetu oz. zaznani realnosti.

Odpor se lahko pojavi zaradi čustvene reakcije posameznika. Ob spremembah se pri posameznikih lahko pojavi občutek nelagodja, pomanjkanja navdu-

šenja in tesnobe, kar se lahko izrazi kot odpor (Arciniega & González, 2008). Bovey in Hede ugotavljata, da posamezniki podzavestno uporabljajo dobro razvite obrambne mehanizme za obrambo pred spremembami in občutkom tesnobe (Bovey & Hede, 2001). Pri tem ni pomembno, ali je vzrok dejansko pravi ali le namišljen. Posamezniki se namreč branijo pred zaznano nevarnostjo, ki temelji tudi na preteklih izkušnjah in strahovih. Posamezniki, ki se zatekajo k uporabi projekcije,<sup>2</sup> so bistveno bolj nagnjeni k odporu (Bovey & Hede, 2001). Problematične čustvene reakcije so sicer lahko tudi posledica osebnih težav, kot sta npr. ločitev in izguba bližnjega (Bovey & Hede, 2001).



Slika 2: Vzroki za pojav odpora z vidika zaznavanja posameznikov<sup>3</sup>

Hultman je najpogostejše vzroke za pojav odpora z vidika zaznavanja posameznikov združil v osem skupin, ki jih prikazuje slika 2:

- **neprimeren proces spreminjanja:** ob soočenju s spremembo imajo posamezniki tipično tri vprašanja: »Zakaj je potrebna sprememba?« »Kako bo vplivala name?« in »Kaj imam od tega?« (Hultman, 2003) Že učinkovit odgovor na navedena vprašanja lahko prepreči ali minimizira odpor (Hultman, 2003). Posamezniki pričakujejo upoštevanje in spoštovanje svojih mnenj (Bovey & Hede, 2001; Hultman, 2003), zato jih je treba vključiti v proces spreminjanja (Henry, 1997; Hultman, 2003). Odpor

se lahko pojavi tudi zaradi nesodelovanja posameznikov pri odločanju, nevšečnega načina predstavitve spremembe, nepravčasnosti spremembe, občutka zavajanja ali manipulacije s strani vodstva in spremembe, ki je bila presenečenje ali pa je bila slaba (Hultman, 2003). Odpor lahko okrepijo tudi prekomerni pritiski (Fiedler, 2010; Henry, 1997; Mihelčič, 2003) in neprimerni ali slabi upravljavski slogi (Waddell & Sohal, 1998);

<sup>2</sup> Projekcija je obrambni mehanizem, za katerega je značilno pripisovanje lastnih nesprejemljivih občutkov, impulzov in misli drugi osebi (Bovey & Hede, 2001). V bistvu gre za prenos krivde in odgovornosti.

<sup>3</sup> Povzeto po Hultmanu (Hultman, 2003).

- **ni potrebe po spremembi:** posamezniki v določenih primerih zaradi različnih vzrokov ne vidijo potrebe po spremembi (Henry, 1997), npr. zaradi njene neotipljivosti (Hultman, 2003). Odpor je močno pogojen tudi z zadovoljstvom do obstoječega stanja, kar povzema miselnost, da spremembe niso potrebne, dokler se kaj ne zalomi (Hultman, 2003);
- **oteževanje zadovoljevanja potreb:** odpor se pojavi, če posamezniki verjamejo, da bo sprememba poslabšala njihov status (Hultman, 2003; Proctor & Doukakis, 2003) ali ogrozila zadovoljevanje osnovnih potreb (Washington & Hacker, 2005). Pri posameznikih se lahko zaradi novih tehnologij pojavi strah, da nekatera znanja ne bodo več potrebna (Mihelčič, 2003), sprememba pa lahko oteži zadovoljevanje potreb posameznikov tudi, če izpostavi pomanjkljivosti njihovega obstoječega dela (Henry, 1997);
- **tveganje je večje od koristi:** tveganje spremlja tako vsako spremembo kot tudi ohranjanje obstoječega stanja (Hultman, 2003). Posamezniki lahko za isto spremembo ali ohranjanje stanja zaznajo različne stopnje tveganja (Hultman, 2003). Odpor se lahko pojavi, če posamezniki ali skupine zaznajo neprimerno stopnjo tveganja (Aladwani, 2001) ali če posamezniki ne zaznajo dovolj (osebnih) koristi spremembe (Proctor & Doukakis, 2003). Nagrade so za posameznike toliko koristne, kolikor jim sami to pripisujejo, zato se lahko pojavi odpor tudi zaradi neustrezne nagrade (Henry, 1997);
- **nezmožnost izvesti spremembe:** odpor se lahko pojavi zaradi dejanske ali zaznane nezmožnosti izvesti spremembe (Hultman, 2003). Dejanska zmožnost je pogojena z znanjem, spretnostmi in potrebnimi viri za izvedbo spremembe (Hultman, 2003), zaznana pa je posledica zaznave posameznikov, ki si ustvarijo mnenje na podlagi svojih izkušenj in drugih osebnih lastnosti. Odpor je torej lahko tudi posledica pomanjkanja samozaupanja posameznikov v lastne sposobnosti (Henry, 1997; Hultman, 2003);
- **sprememba ne bo uspešna:** odpor se lahko pojavi, če posamezniki dvomijo v uspeh spremembe ali ne verjamejo, da je sploh izvedljiva (Hultman, 2003; Waddell & Sohal, 1998);
- **neskladje z vrednotami:** vrednote so posameznikova prepričanja o tem, kaj je pomembno (Hultman, 2003). Odpor se lahko pojavi, če sprememba

prihaja v konflikt z obstoječimi vrednotami posameznikov ali skupin (Arciniega & González, 2008; Henry, 1997; Hultman, 2003; Washington & Hacker, 2005);

- **nezaupanje do odgovornih za spremembo:** odpor se lahko pojavi zaradi občutka, da vodstvo ni odkrito glede spremembe in njenega vpliva (Hultman, 2003; Mihelčič, 2003). Pomanjkanje zaupanja je pogosta, zahrbtna in prodorna težava, saj po nekaterih raziskavah več kot 80 odstotkov uslužbencev ne zaupa svojim nadrejenim (Hultman, 2003), pri čemer igra vidno vlogo tudi uporaba dvomljivih tehnik, npr. manipulacije in prisilne odgovornih (Bovey & Hede, 2001).

## 5 PRIPOROČILA ZA RAVNANJE Z ODPOROM

Za učinkovito obvladovanje odpora je treba izvajati tako preventivne ukrepe kot strategije razreševanja odpora (Fiedler, 2010; Henry, 1997), pri čemer oboje skupaj pojmuje kot strategije za ravnanje z odporom. Strategije za ravnanje z odporom je treba prilagoditi vsakokratni situaciji, saj lahko v nasprotnem primeru učinkujejo ravno nasprotno od pričakovanega. V nadaljevanju so po razdelkih na kratko opisana nekatera v literaturi najpogosteje omenjana priporočila za ravnanje z odporom.

### 5.1 Priprava na spremembo

Pogoj za obvladovanje odpora je poznavanje organizacije, kulture in tehnologije (Lorenzi & Riley, 2003; Proctor & Doukakis, 2003). Priporočljiva je priprava na argumentiran zagovor vizije spremembe (Henry, 1997; Rumelt, 1995). Pri tem je priporočljivo predvideti, kaj bo uporabnike zanimalo o spremembi (Henry, 1997), in na podlagi tega vnaprej pripraviti odgovore na pogosto zastavljena vprašanja (Henry, 1997; Hultman, 2003). Priporočljiva je tudi priprava učinkovitih odgovorov na zbadljive komentarje uporabnikov (Henry, 1997).

Za implementacijo spremembe je treba zagotoviti ustrezne vire (Hultman, 2003). Priporočljivo je tudi aktivno iskanje v dani situaciji primernejših alternativnih načinov uvajanja sprememb (Waddell & Sohal, 1998).

### 5.2 Zgled in podpora spremembi

Priporočljivo je poskrbeti, da višji menedžerji dejansko podpirajo spremembo. Višje organizacijske ravni predstavljajo zgled (Rumelt, 1995), zato mora

biti njihova podpora jasno vidna. Če menedžerji zagovarjajo spremembo, ki se je vidno ne držijo, lahko to vzdržuje odpor (Henry, 1997). Priporočljivo je tudi povečanje zavezanosti spremembi posameznikov z veliko neformalno močjo, npr. oblikovalcev mnenj in »mogočnežev« (Aladwani, 2001; Fiedler, 2010; Lorenzi & Riley, 2003; Mihelčič, 2003). To je v določenih primerih mogoče doseči z oblikovanjem skupine ambasadorjev spremembe, v kateri omenjeni posamezniki aktivno podpirajo spremembe (Fiedler, 2010).

### 5.3 Vključevanje uporabnikov

Sodelovalne tehnike so med najboljšimi načini ravnanja z odporom (Waddell & Sohal, 1998). Zelo so priporočljive, kadar pričakujemo močan odpor (Waddell & Sohal, 1998). Sodelovanje v vseh fazah spreminjanja zmanjšuje odpor in povečuje zavezanost spremembam. Pri tem je pomembno, da vključimo vse relevantne skupine (Fiedler, 2010), pri čemer je različne skupine primerno vključevati v proces spreminjanja v različnih fazah. Sodelovalne tehnike naj ne bi bile le fiktivne. To pomeni, da dejansko upoštevamo in spoštujemo mnenja in interese uporabnikov (Hultman, 2003; Proctor & Doukakis, 2003), se z njimi posvetujemo in skupaj diagnosticiramo probleme (Proctor & Doukakis, 2003). Uporabniki so tudi pomemben vir povratnih informacij (Waddell & Sohal, 1998). Pri sodelovanju z uporabniki je priporočljiva določena mera potrpežljivosti, moramo pa biti pozorni na namerno zavlačevanje, ki lahko nakazuje na druge interese, npr. vzdrževanje moči, in ki ne razrešuje odpora (Henry, 1997).

### 5.4 Motivacija

Pred uvedbo spremembe je priporočljivo ustvarjanje občutka potrebe po spremembi (Hultman, 2003). Odpor je sorazmeren s stabilnostjo obstoječega stanja, zato mu je treba zamajati temelje. To je mogoče storiti s slabšanjem obstoječega stanja ali promocijo spremenjenega. Zavaljo promocije spremenjenega stanja je treba zagotoviti, da sprememba prinaša dovolj koristi za uporabnike (Hultman, 2003; Mihelčič, 2003). Koristi je mogoče v splošnem opredeliti samo okvirno, saj so učinkovite le toliko, kolikor pomenijo posameznikom (Hultman, 2003).

### 5.5 Odprava komunikacijskih ovir

Priporočajo vzpostavitev formalnega in koordiniranelega komunikacijskega sistema (Proctor & Douka-

kis, 2003). Pri tem lahko odigra ključno vlogo informacijska tehnologija, zato priporočajo uporabo le-te, kjer koli je mogoče. Pri komunikaciji je priporočljivo:

- redno komuniciranje (Waddell & Sohal, 1998);
- posredovanje vizije in jasna opredelitev strategije sprememb vsem uporabnikom (Fiedler, 2010; Henry, 1997; Mihelčič, 2003; Proctor & Doukakis, 2003);
- posredovanje ključnih lastnosti spremembe (Proctor & Doukakis, 2003) in motivov zanjo (Fiedler, 2010);
- posredovanje kakovostnih in uporabnih informacij. Prave informacije lahko ugodno vplivajo na odnos uporabnikov do spremembe (Washington & Hacker, 2005);
- posredovanje nedvoumnih informacij. Informacije so namenjene uporabnikom, ki si jih lahko razlagajo tudi drugače od predvidenega (Hultman, 2003);
- celovito posredovanje informacij. Če uporabniki določene informacije ne dobijo, jo predvidevajo, kar lahko povzroča napačno razumevanje in nezaupanje (Proctor & Doukakis, 2003);
- informacije predstaviti ustrezno (Proctor & Doukakis, 2003);
- prirejanje informacij za posamezne ciljne skupine, npr. čas posredovanja informacij, količina in način posredovanja informacij (Fiedler, 2010);
- odprta diskusija oz. dialog med uporabniki in voditeljem sprememb (Fiedler, 2010; Proctor & Doukakis, 2003) ter konzultacije (Waddell & Sohal, 1998);
- upoštevati dejstvo, da so voditelji sprememb bolj zavezani spremembam, če jih razumejo (Washington & Hacker, 2005).

Uporaba informacijske tehnologije skrajšuje pot informacij (Proctor & Doukakis, 2003), kar olajša zagotavljanje preverjenih informacij in omejuje netočne, nepopolne in napačne informacije, ki sicer predstavljajo problem (Hultman, 2003).

### 5.6 Usklajevanje vrednot

Obstoj globoko zakoreninjenih vrednot je med najvidnejšimi vzroki za pojav odpora (Pardo del Val & Martínez Fuentes, 2003). Cilje spremembe je treba še pred njeno implementacijo uskladiti z organizacijsko kulturo (Pardo del Val & Martínez Fuentes, 2003). Dolgoročno gledano je priporočljiva neprestana skrb za usklajenost vrednot posameznikov z vrednotami organizacije (Hultman, 2003). Za doseglo kakovost



stne spreminjajoče se organizacije je potreben razvoj vrednot posameznikov in organizacije, ki povečujejo zavezanost spremembam samim. Zmožnost učinkovitega in kakovostnega spreminjanja bi tako lahko postala ključna vrednota.

### 5.7 Evolucija uporabnikov

Evolucija uporabnikov je ključna za evolucijo in napredek organizacije, saj brez nje ni mogoče uigrati organizacije. Njene aktivnosti naj bi potekale v okolju vzajemnega učenja in podpore, brez strahu (Henry, 1997). Evolucija uporabnikov lahko med drugim vključuje:

- izboljšavo sposobnosti samoocenjevanja (Hultman, 2003),
- krepitev samozaupanja (Hultman, 2003),
- razvoj potrebnih spretnosti in znanj (Hultman, 2003), npr. izobraževanja in mentorstvo (Henry, 1997),
- različna urjenja (Pardo del Val & Martínez Fuentes, 2003),
- informiranje in svetovanje v zvezi z odzivi na spremembe, npr. obrambnimi mehanizmi (Bovey & Hede, 2001),
- razvoj vodstvenih sposobnosti voditeljev sprememb (Proctor & Doukakis, 2003).

### 5.8 Ustvarjanje spremembi naklonjenega okolja

V organizaciji je treba pred implementacijo spremembe vzpostaviti pripravljenost nanjo (Proctor & Doukakis, 2003). Pri tem je treba posvetiti pozornost odpravljanju občutka uporabnikov, da jim je bila oz. jim bo sprememba vsiljena (Proctor & Doukakis, 2003) ali da obstajajo skriti nameni (Mihelčič, 2003; Proctor & Doukakis, 2003). Priporočljiva je stalna skrb za krepitev zaupanja med vsemi zaposlenimi v organizaciji (Hultman, 2003). Zaupanje ni recipročno, saj zaupanje zaposlenih v vodstvo še ne pomeni tudi zaupanja v obratni smeri (Proctor & Doukakis, 2003).

### 5.9 Vključitev zunanjih svetovalcev

Pri obvladovanju odpora, pa tudi sicer pri obsežnejših spremembah informacijskega sistema, je priporočljiva pomoč zunanjih izvajalcev (Lorenzi & Riley, 2003; Mihelčič, 2003). Zunanji svetovalci zagotavljajo objektivni pogled, morda njihova največja vrednost pa je, da lažje pridobijo informacije občutljive narave, ki jih zaposleni sicer neradi posredujejo sodelavcem (Mihelčič, 2003). Prednost zunanjih svetovalcev je tudi v izkušnjah in strokovni usposobljenosti.

## 6 KLJUČNI ELEMENTI OBVLADOVANJA ODPIORA

Analiza odpora in razvoj strategij ravnanja z odporom sta zahtevna, a ključna procesa obvladovanja odpora. Uporaba koncepta ključnih elementov obvladovanja odpora (KEOO) je pri omenjenih procesih lahko v pomoč. To velja predvsem pri identifikaciji vzrokov za pojav odpora, planiranju strategij ravnanja z odporom in planiranju implementacije strategij ravnanja z odporom.

Vzroki za pojav odpora so splet interakcije med ključnimi elementi obvladovanja odpora. Razmerje med njimi in vzroki za pojav odpora je najlažje opisati s preprostim primerom. Če opazujemo dva ključna elementa obvladovanja odpora, uporabnike in spremembo, so lahko vzroki za pojav odpora na strani uporabnikov (npr. interes ključnih uporabnikov, da sprememba ni uspešna) ali na strani spremembe (npr. sprememba je neprimerna). Spremembo lahko uporabniki le zaznajo kot neprimerno (vzrok za pojav odpora je pravzaprav na strani uporabnikov) ali pa je neprimerna tudi objektivno.

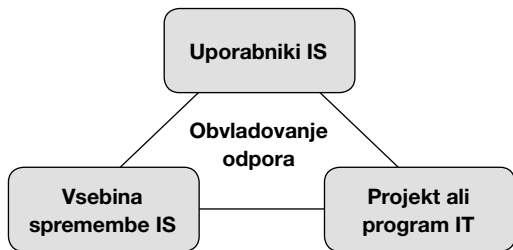
Rezultate analize odpora je mogoče neposredno uporabiti pri razvoju strategij ravnanja z odporom. Za vsak identificirani vzrok je mogoče ugotoviti, na kateri ključni element obvladovanja odpora je treba aplicirati strategijo (uporabnike, spremembo oz. nekakšen kompromis). Uporaba ključnih elementov obvladovanja odpora v procesu razvoja strategij ravnanja z odporom omogoča tudi pregled nad skupnim vplivom izbranih strategij na posamezen ključni element obvladovanja odpora, kar podpira planiranje implementacije strategij ravnanja z odporom.

Vsaka sprememba je edinstvena, zato si zasluži temu primerno obravnavo. Vprašamo se lahko, ali je torej sploh smiselno slediti priporočilom in dobrim praksam. Odgovor je pravzaprav preprost. Dobre prakse in priporočila izhajajo iz konkretnih situacij, ki pa se razlikujejo med seboj. Torej gre za splošne ugotovitve, ki v povprečju delujejo pozitivno. Možnost, da v konkretnem primeru splošne ugotovitve povzročijo odpor, kljub temu obstaja. Na voljo sta dve možnosti:

- tvegati in slepo zaupati priporočilom in dobrim praksam. V povprečju se tak pristop izplača, v posameznih primerih pa se lahko tudi grdo maščuje;
- analizirati ključne elemente obvladovanja odpora, priporočila in dobre prakse ter na podlagi tega razviti ustrezne strategije za ravnanje z odporom.

Tak pristop drastično zniža tveganje, hkrati pa poveča možnosti za uspeh spremembe.

Poznavanje koncepta ključnih elementov obvladovanja odpora olajšuje obravnavo vzrokov za pojav odpora in razvoj strategij za ravnanje z odporom. Omenjene strategije namreč temeljijo na modifikacijah ključnih elementov obvladovanja odpora ali vplivanju nanje.



Slika 3: Ključni elementi obvladovanja odpora

Pri splošni spremembi so ključni elementi obvladovanja odpora vsebina spremembe, proces implementacije spremembe in prizadeti posamezniki oz. skupine. Slika 3 prikazuje ključne elemente obvladovanja odpora pri projektih informacijske tehnologije:

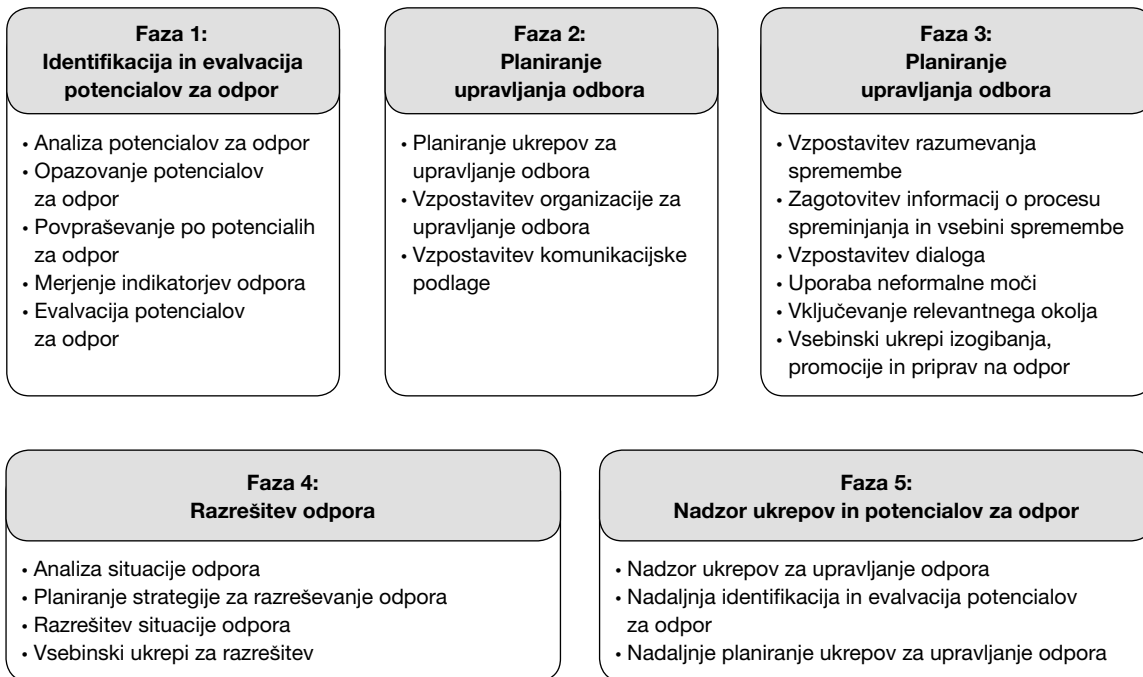
▪ **vsebina spremembe informacijskega sistema:** odpor se lahko pojavi na stiku s projektom ali programom informacijske tehnologije, z uporabniki ali pri skupnem stiku z obema elementoma;

▪ **projekt ali program informacijske tehnologije:** spremembe informacijskega sistema se lahko implementira skozi projekte ali programe informacijske tehnologije (Fiedler, 2010; PMI, 2008);

▪ **uporabniki informacijskega sistema:** posamezniki, ki jih prizadene sprememba, so uporabniki informacijskega sistema. Lahko so notranji (člani organizacije) ali zunanji (poslovni partnerji, stranke idr.), odpor pa se lahko pojavi pri obojih (Fiedler, 2010). Uporabniki informacijskega sistema so večplastni deležniki in lahko obsegajo več hierarhičnih ravni v organizaciji (PMI, 2008).

## 7 PREGLED SORODNIH RAZISKAV

Slika 4 prikazuje Fiedlerjev model obvladovanja odpora, ki obravnava odpor kot projektno tveganje. Faze Fiedlerjevega modela niso strogo zaporedne in predstavljajo le okvirno časovno zaporedje (Fiedler, 2010).



Slika 4: Fiedlerjev model obvladovanja odpora<sup>4</sup>

<sup>4</sup> Povzeto po Fiedlerju (Fiedler, 2010).

Fiedler izpostavlja nekaj omejitev svojega modela (Fiedler, 2010):

- Model ni splošen zaradi raznolikosti mogočih situacij odpora, a postavlja strukturno orientacijo.
- Model ne ponuja vpogleda v vsebinske procese obvladovanja odpora, kot je npr. proces pridobivanja informacij.
- Psihološki in sociološki procesi niso v ospredju. Mogoča bi bila integracija teh pristopov za boljšo identifikacijo odpora in potencialov za odpor.

Na podlagi Fiedlerjevih ugotovitev in primerjave z drugimi modeli lahko opazimo, da model predvideva dva sklopa enakega zaporedja procesov. Prvi sklop je namenjen preventivi in preprečevanju odpora, drugi sklop pa obravnava obstoječe situacije odpora. Prvi sklop sestavljajo procesi:

- **analiza situacije:** zadnji proces prve faze »evalvacija potencialov za odpor«. Evalvacija potencialov za odpor vključuje tudi analizo potencialov za odpor;
- **razvoj strategij:** vsa druga faza. Razvoj strategij vključuje tudi pripravo na implementacijo strategij;
- **implementacija strategij:** vsa tretja faza. V Fiedlerjevemu modelu je vsa tretja faza pravzaprav za konkretni primer prirejena faza implementacije strategij – v splošnem je to rezultat predhodnega procesa »razvoj strategij«.

Drugi sklop sestavljajo procesi:

- **analiza situacije:** prvi proces četrte faze »analiza situacije odpora«;
- **razvoj strategij:** drugi proces četrte faze »planiranje strategije za razreševanje odpora«;
- **implementacija strategij:** tretji proces četrte faze »razrešitev situacije odpora«.

Fiedlerjev model je po našem mnenju mogoče poenostaviti in združiti sklopa. V združenem modelu se zaporedje obravnavanih sklopov odraža v več iteracijah procesa obvladovanja odpora. V začetnih iteracijah je treba posvetiti več pozornosti preventivnim strategijam, če se pojavi odpor, pa tudi procesom drugega sklopa.

Slika 5 prikazuje Hultmanov model premagovanja odpora<sup>5</sup> odpora, ki se vsebinsko pokriva s četrto fazo Fiedlerjevega modela. Hultmanov model velja za splošen

proces implementacije spremembe, torej ga je mogoče aplicirati tudi na spreminjanje skozi projekt ali program.



Slika 5: **Hultmanov model premagovanja odpora**<sup>6</sup>

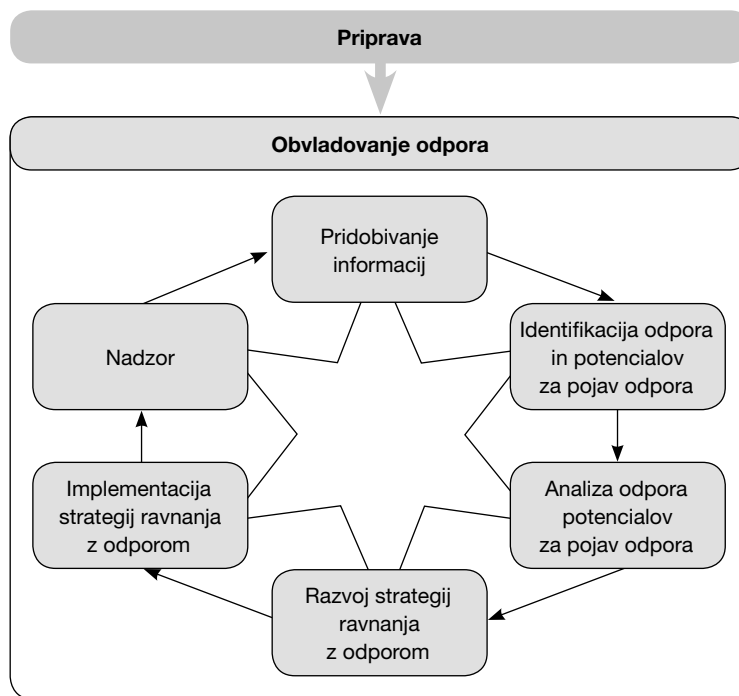
Hultman razločuje pozitivni in negativni odpor, torej odpor, ki se obravnava kot problem, in odpor, ki se obravnava kot rešitev (Hultman, 2003). Hultmanov model naj bi bil primeren le za obravnavo problematičnega odpora (Hultman, 2003). Spremembo je mogoče oceniti samo za nazaj (Hultman, 2003), zato razločitev pravzaprav ni najbolj posrečena. V sedanjosti namreč ni mogoče ugotoviti, kakšnega tipa je odpor, zato tudi ni mogoče z gotovostjo ugotoviti, kdaj naj bi se Hultmanov model sploh uporabljal. Kljub temu ga je mogoče ob upoštevanju koristi, ki jih prinaša odpor, prilagoditi za obravnavo odpora, ne glede na njegov tip.

## 8 MODEL OBVLADOVANJA ODPIORA

Kot nadgradnjo sorodnih raziskav smo v okviru naše raziskave opredelili model obvladovanja odpora, ki temelji na obeh predstavljenih modelih, generičnem modelu obvladovanja tveganj (Fiedler, 2010), modelu obvladovanja tveganj PMI (PMI, 2009a) in priporočilih za ravnanje z odporom. Model obvladovanja odpora tako predstavlja združitev pogledov na odpor z vidika obvladovanja tveganj in s »klasičnega« vidika obvladovanja odpora. Model obvladovanja odpora sestavljajo priprava na obvladovanje odpora in šest procesov obvladovanja odpora, kar prikazuje slika 6.

<sup>5</sup> Izraz razreševanje odpora (*resolution of resistance*) je primernejši v primerjavi s primerljivimi izrazi, kot je npr. premagovanje odpora (*overcoming resistance*), saj vsaj deloma odpravlja negativno konotacijo odpora. Pri vseh izrazih gre sicer za isto vsebino.

<sup>6</sup> Povzeto po Hultmanu (Hultman, 2003).

Slika 6: **Model obvladovanja odpora**

Podobno kot Fiedlerjev model je tudi model obvladovanja odpora samo okvirno časovno zaporedje procesov (Fiedler, 2010). Obvladovanje odpora (podobno kot obvladovanje tveganj) ni enkratno, temveč ponovljivo (iterativno) zaporedje procesov (PMI, 2009a). Vse strategije namreč temeljijo na hipotezah, zato jih je treba prilagajati (Hultman, 2003). Med procesom spreminjanja se spreminjajo tudi okoliščine, količina informacij pa se običajno povečuje, zato je prva iteracija le začetek in nikakor še ne konec obvladovanja odpora (PMI, 2009a).

Za učinkovito obvladovanje odpora je treba ekipi za obvladovanje odpora zagotoviti ustrezen vpliv na ključne elemente obvladovanja odpora. Pri tem je stvar strategije organizacije, kakšen bo vpliv na posamezne ključne elemente obvladovanja odpora. Npr. na vsebino spremembe informacijskega sistema je mogoče vplivati na strateški ravni (npr. izbira med prenovo ali nakupom novega dela informacijskega sistema) ali nekoliko nižji ravni (npr. izbira med različnimi alternativami dela prenove informacijskega sistema).

## 8.1 Priprava

Namen priprave je vzpostavitev pogojev za aktivno obvladovanje odpora; to so:

- določitev jedra ekipe za obvladovanje odpora,
- razvoj splošne strategije obvladovanja odpora,
- integracija procesov obvladovanja odpora,
- določitev grobih okvirov obvladovanja odpora.

Priprava na obvladovanja odpora naj bi se izvedla zgodaj v planiranju projekta, procesi obvladovanja odpora pa naj bi bili integrirani v plan za obvladovanje projekta (PMI, 2009a). Če se med aktivnim obvladovanjem odpora pojavijo potrebe po prilagoditvi v pripravi obravnavanih elementov, jih je treba ustrezno prilagoditi. Že od samega začetka je pomembna promocija obvladovanja odpora pri vodstvu – izpostavljanje pozitivnih učinkov in narave odpora.

## 8.2 Pridobivanje informacij

Fiedlerjev model ne predvideva vsebinskih procesov obvladovanja odpora, kot je npr. pridobivanje informacij, predlaga pa njihovo vključitev med procese obvladovanja odpora (Fiedler, 2010).

Pridobivanje informacij je proces, ki ga ne izpeljemo samo na začetku vsake iteracije obvladovanja odpora. Izvajamo ga neprestano in je sestavni del vseh drugih procesov. Njegov namen je pridobivanje relevantnih informacij za obvladovanje odpora. Ob začetku vsake iteracije ga izvedemo v večjem obsegu in preverimo informacije – preverimo stare in pridobimo nove.

bimo nove. Informacije lahko pridobivamo na podlagi ključnih elementov obvladovanja odpora:

- pridobivanje informacij o vsebini spremembe informacijskega sistema,
- pridobivanje informacij o projektu ali programu informacijske tehnologije,
- pridobivanje informacij o uporabnikih.

Pridobljene informacije morajo biti celovite, da jih lahko uporabimo pri izvajanju v procesu obvladovanja odpora razvitih strategij ravnanja z odporom. Za obvladovanje odpora je ključnega pomena, da so informacije kakovostne (PMI, 2009a) in zanesljive. Najpogosteje informacije pridobivamo z izvedbo intervjujev in delavnic, sicer pa tudi na druge načine z uporabo strokovnega znanja (PMI, 2009a). Pridobljene informacije je treba ocenjevati in preverjati, saj obstaja nevarnost, da bi bile pristranske. Če je mogoče, je treba pristranske informacije identificirati in popraviti ali pa jih pridobiti iz drugega vira (PMI, 2009a).

Ekipo za obvladovanje odpora lahko zavoljo tega procesa razširimo z notranjimi ali zunanjimi informatorji, zadolženimi samo za pridobivanje kakovostnih in zanesljivih informacij.

### **8.3 Identifikacija odpora in potencialov za pojav odpora**

Identifikacija odpora in potencialov za odpor vključuje:

- ustvarjanje seznama odpora in potencialov,
- verifikacijo odpora in potencialov.

Namen identifikacije odpora in potencialov je oblikovati seznam že obstoječega odpora in potencialov ter njihova morebitna verifikacija (Fiedler, 2010). Pri identifikaciji ločevanje med vzroki za pojav odpora in oblikami odpora še ni bistvenega pomena.

### **8.4 Analiza odpora in potencialov za pojav odpora**

Analiza odpora in potencialov za odpor vključuje:

- ocenjevanje odpora in potencialov,
- identifikacijo vzrokov za pojav odpora.

Pri analizi odpora in potencialov je treba jasno ločiti vzroke za pojav odpora od oblik in simptomov odpora (Fiedler, 2010; Hultman, 2003). Ugotovitve analize (npr. nujnost, intenzivnost, ciljne skupine in terminski plan) predstavljajo podlago za razvoj strategij ravnanja z odporom (Fiedler, 2010).

### **8.5 Razvoj strategij ravnanja z odporom**

Razvoj strategij ravnanja z odporom vključuje:

- planiranje strategij ravnanja z odporom,

- planiranje implementacije strategij ravnanja z odporom,
- vzpostavitev organizacije,
- vzpostavitev komunikacijske podlage.

Razvoj strategij ravnanja z odporom določa vsebino procesa »implementacija strategij ravnanja z odporom« in vzpostavlja pogoje za njegovo izvajanje. Dejavniki uspeha razvoja strategij ravnanja z odporom so podobni dejavnikom uspeha planiranja odgovorov na tveganja (PMI, 2009a):

- odkrita komunikacija med ekipo za obvladovanje odpora, lastnikom spremembe in drugimi deležniki;
- jasna opredelitev vlog in odgovornosti;
- integracija procesa implementacije strategij ravnanja z odporom v terminski plan;
- ocenjevanje in odobritev potrebnih virov, proračuna in časa za implementacijo posameznih strategij;
- upoštevanje interakcije med odporom in strategijami;
- razvoj primernih, izvedljivih, učinkovitih in sprejemljivih strategij;
- obravnavanje pozitivnih in negativnih vidikov odpora;
- razvoj splošnih strategij pred podrobnejšimi taktičnimi ukrepi.

### **8.6 Implementacija strategij ravnanja z odporom**

Implementacija strategij ravnanja z odporom je v splošnem nedoločen proces, ki se oblikuje v predhodnem procesu »razvoj strategij ravnanja z odporom«.

Pri implementaciji je ključnega pomena, da jo izvedemo ob pravem času in v primernem obsegu (Hultman, 2003). Primerna strategija ima lahko nasproten učinek od želenega že zaradi neprimerne časa implementacije. Pravi čas je mogoče ugotoviti s pomočjo komuniciranja in povratnih informacij (Hultman, 2003). Primeren obseg, ki je sicer povezan tudi s pravočasnostjo, se osredinja na to, kolikšen delež strategij implementirati v določenem času. Vsak posameznik je omejen z obsegom sprememb, ki jih lahko prenese v določenem času. Implementacija, ki presega omejitve ciljne skupine, ima lahko tako zaradi prevelikega obsega nasproten učinek od želenega (Hultman, 2003).

### **8.7 Nadzor**

Namen nadzora implementacije strategij ravnanja z odporom je ocenjevanje učinkovitosti strategij,

proženje predvidenih aktivnosti implementacije in po potrebi proženje nove iteracije obvladovanja odpora. Tudi pri nadzoru je pomembno pridobivanje informacij na že omenjene načine (Hultman, 2003). Nadzor je treba opravljati redno (PMI, 2009a).

Proces implementacije je mogoče nadzorovati z različnih aspektov, kot so (Fiedler, 2010):

- učinkovitost strategij,
- doseganje ciljev (npr. časovni okvir, predvideni viri),
- odstopanja od predvidevanj,
- potrebe po prilagoditvah,
- drugi učinki, ki zahtevajo novo iteracijo procesa obvladovanja odpora.

Učinek strategij je lahko izrazito pozitiven ali negativen, vendar pa je lahko včasih tudi komaj opazen (Hultman, 2003). Zaradi tega moramo biti pozorni na kakršen koli učinek, saj je lahko tudi najmanjši učinek namig, da strategija deluje (Hultman, 2003).

### **8.8 Umestitev obvladovanja odpora v standard za vodenje projektov PMI**

Umestitev obvladovanja odpora v standarde PM je obsežna naloga glede na relativno široko paleto standardov organizacij, kot so npr. *International Project Management Association* (IPMA, 2006), *Office of Government Commerce* (OGC, 2007; OGC, 2009) in *Project Management Institute* (PMI, 2008; PMI, 2009b; PMI, 2009c). V tem razdelku bomo predstavili le umestitev v standard za vodenje projektov PMI (PMI, 2008), ki ne obravnava odpora na strani uporabnikov. Na podlagi ugotovitev pričujoče raziskave podajamo naš pogled na možne dopolnitve standarda:

- **odpor do sprememb kot projektno tveganje:** odpor do sprememb je po našem mnenju vsekakor treba obravnavati kot projektno tveganje. Fiedler npr. v svojih raziskavah ugotavlja, da je odpor mogoče uspešno obvladovati tudi ob obravnavi odpora kot projektne tveganja, kar avtor potrди tudi s primerom iz prakse (Fiedler, 2010). Za uspešno obvladovanje odpora je ključnega pomena, da ekipi za obvladovanje odpora zagotovimo ustrezne pristojnosti in vzvode. Obvladovanje odpora je vsebinsko del upravljanja sprememb, zato bi morali odpor obvladovati v okviru upravljanja sprememb v matični organizaciji, in sicer v kontekstu projekta;
- **obvladovanje odpora kot novo področje znanja:** standard podaja metodološko-vsebinski del kot

področja znanja (*knowledge areas*). Obvladovanje odpora bi glede na strukturo standarda lahko opredelili tudi kot novo področje znanja.

Na podlagi doslej predstavljenih ugotovitev pričujoče raziskave in na podlagi izkušenj iz prakse ocenjujemo, da je najboljša prva različica dopolnitve standarda za vodenje projektov, torej obravnavanje odpora do sprememb kot projektno tveganje. Menimo, da je v obravnavanem standardu na področju obvladovanja tveganj smiselno podati različna priporočila in prijeme za področje obvladovanja odpora.

Novo področje znanja bi morda lahko bil zanimiv predlog, ki ni povsem brez argumentov. Vendar pa se je treba zavedati, da bi šlo za radikalen poseg v standard, ki bi zahteval zelo široko podporo.

## **9 SKLEP**

Ključni elementi obvladovanja odpora, model obvladovanja odpora in umestitev obvladovanja odpora v standard za vodenje projektov PMI predstavljajo ogrodje obvladovanja odpora pri projektih informacijske tehnologije. Omenjeno ogrodje lahko olajša obvladovanje odpora, nikakor pa ne more nadomestiti strokovne usposobljenosti in »žilice za obvladovanje odpora« članov ekipe za obvladovanje odpora.

Po predstavitvi koncepta ključnih elementov obvladovanja odpora je mogoče opaziti, da so vzroki za pojav odpora obravnavani pomanjkljivo. Nadaljnje raziskave so potrebne v smeri strukturiranja vzrokov na podlagi medsebojne interakcije ključnih elementov obvladovanja odpora. Ustrezno strukturirani vzroki so lahko v pomoč pri obravnavi (identifikaciji in analizi) vzrokov za pojav odpora.

Obvladovanje informatike, ki predstavlja disciplino obvladovanja organizacije, s katero lahko upravljamo spremembe informacijskega sistema na strateški ravni (Korac-Kakabadse & Kakabadse, 2001), se osredinja predvsem na tehnični vidik sprememb informacijskega sistema, pri tem pa vpliv sprememb informacijskega sistema na organizacijo ostaja prezrt. Do problemov prihaja predvsem pri prenovah informacijskega sistema in spreminjanju delovnih procesov, saj je pri tem ključnega pomena usklajeno spreminjanje informacijskega sistema in organizacije. Težave povzročajo predvsem spreminjanje organizacije, saj je zaradi kompleksnosti medsebojnih razmerij težje predvideti, kakšen bo njen odziv. Od odziva organizacije pa je odvisen tudi končni uspeh sprememb informacijskega sistema. Zaradi tega se

na področju informacijske tehnologije pojavlja potreba po integraciji upravljanja sprememb, obvladovanja informatike in projektnega menedžmenta. Gleđano nekoliko širše gre za potrebo po sožitju med tehnologijo in ljudmi, ki jo uporabljajo.

## 10 LITERATURA

- [1] Aladwani, A. M. (2001). Change management strategies for successful ERP implementation. *Business Process Management Journal*, 7 (3), 266–275.
- [2] Arciniega, L. M., & González, L. (2008). Validation of the Spanish-language version of the resistance to change scale. *Personality and Individual Differences*, 46 (2), 178–182.
- [3] Bovey, W. H., & Hede, A. (2001). Resistance to organisational change: the role of defence mechanisms. *Journal of Managerial Psychology*, 16 (7), 534–548.
- [4] Chen, J., & Wang, L. (2007). Locus of control and the three components of commitment to change. *Personality and Individual Differences*, 42 (3), 503–512.
- [5] Fiedler, S. (2010). Managing resistance in an organizational transformation: A case study from a mobile operator company. *International Journal of Project Management*, 28 (4), 370–383.
- [6] Gareis, R. (2010). Changes of organizations by projects. *International Journal of Project Management*, 28 (4), 314–327.
- [7] Henry, P. K. (1997). Overcoming resistance to organizational change. *Journal of the American Dietetic Association*, 97 (10), S145–S147.
- [8] Hultman, K. (2003). Resistance to Change, Managing. *Encyclopedia of Information Systems* (3), 693–705.
- [9] IPMA. (2006). *ICB – IPMA Competence Baseline, Version 3.0*. Zurich: International Project Management Association.
- [10] Korac-Kakabadse, N., & Kakabadse, A. (2001). IS-IT governance: need for an integrated model. *Corporate Governance*, 1 (4), 9–11.
- [11] Lorenzi, N. M., & Riley, R. T. (2003). Organizational issues = change. *International Journal of Medical Informatics*, 69 (2–3), 197–203.
- [12] Mihelčič, M. (2003). *Organizacija in ravnanje*. Ljubljana: Fakulteta za računalništvo in informatiko.
- [13] OGC. (2007). *Managing Successful Programmes*. London: Office of Government Commerce.
- [14] OGC. (2009). *Managing Successful Projects With Prince2*. London: Office of Government Commerce.
- [15] Pan, G., Hackney, R., & Pan, S. L. (2008). Information Systems implementation failure: Insights from prism. *International Journal of Information Management*, 28 (4), 259–269.
- [16] Pardo del Val, M., & Martínez Fuentes, C. (2003). Resistance to change: a literature review and empirical study. *Management Decision*, 41 (2), 148–155.
- [17] Pinto, J. K. (1996). *Power & Politics in Project Management*. Newtown Square, PA: Project Management Institute.
- [18] PMI. (2008). *A Guide to the Project Management Body of Knowledge*. Newtown Square, PA: Project Management Institute.
- [19] PMI. (2009a). *Practice Standard for Project Risk Management*. Newtown Square, PA: Project Management Institute.
- [20] PMI. (2009b). *Standard for Portfolio Management*. Newtown Square, PA: Project Management Institute.
- [21] PMI. (2009c). *Standard for Program Management*. Newtown Square, PA: Project Management Institute.
- [22] Proctor, T., & Doukakis, I. (2003). Change management: the role of internal communication and employee development. *Corporate communications: An International Journal*, 8 (4), 268–277.
- [23] Rumelt, R. P. (1995). Inertia and Transformation. *Resources in an Evolutionary Perspective: Towards a Synthesis of Evolutionary and Resource-Based Approaches to Strategy*, 101–132.
- [24] Waddell, D., & Sohal, A. S. (1998). Resistance: a constructive tool for change management. *Management Decision*, 36 (8), 543–548.
- [25] Washington, M., & Hacker, M. (2005). Why change fails: knowledge counts. *Leadership & Organizational Development Journal*, 26 (5), 400–411.
- [26] Yeo, K. T. (2002). Critical failure factors in information system projects. *International Journal of Project Management*, 20 (3), 241–246.

Simon Vrhovec je zaposlen kot mladi raziskovalec na Fakulteti za računalništvo in informatiko Univerze v Ljubljani, na kateri je diplomiral leta 2010 in nadaljuje s podiplomskim študijem. Pri projektih laboratorija za informatiko na omenjeni fakulteti je sodeloval od leta 2008. Med študijem se je poleg klasičnih študijskih vsebin ukvarjal s preučevanjem medčloveških razmerij, sožitjem med ljudmi in tehnologijo ter z drugimi področji, povezanimi s sodobno družbo. Zanimanje za družboslovna področja je predvsem posledica opravljanja vloge učitelja karateja v letih 2003 do 2010, inštruktorja matematike ter raznovrstnih življenjskih situacij in izkušenj.

Rok Rupnik je zaposlen kot docent na Fakulteti za računalništvo in informatiko Univerze v Ljubljani, na kateri je magistriral leta 1998 in doktoriral leta 2002. Njegovo raziskovalno področje obsega projektno vodenje, metodologije razvoja informacijskih sistemov, odkrivanje zakonitosti v podatkih, strateško planiranje informacijskih sistemov ter mobilne aplikacije in mobilno poslovanje. V svoji karieri je vodil več projektov razvoja informacijskih sistemov. Kot vodja ali koordinator projekta je sodeloval tudi na drugih projektih širšega področja informacijskih sistemov, kot so izdelava strateških planov razvoja informatike za večje poslovne sisteme, izdelava strateških študij, raziskovalni projekti itn. Je član Slovenskega društva Informatika, združenja ACM, združenja AIS (Association for Information Systems) in ustanovitveni član slovenske sekcije PMI (Project Management Institute). Leta 2009 je pridobil certifikat PMP (Project Management Professional), ki ga podeljuje PMI.

# ▣ Orodja za podporo testiranja parov

Tomaž Dogša, Mirjam Bonačić

Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko, Smetanova ulica 17, 2000 Maribor  
tdogsa@uni-mb.si; mirjam.bonacic@uni-mb.si

## Izvleček

V prispevku je na kratko opisana strategija testiranja vseh parov. Strategija testiranja parov je sistematičen postopek izbora vhodnih podatkov, ki predpostavlja, da so v programu takšne napake, katerih navzočnost se bo pokazala pri ustrezni kombinaciji samo dveh vhodnih vrednosti. Zelo pogosto jo uporabljamo pri testiranju raznih računalniških konfiguracij (npr. različni operacijski sistemi, tiskalniki, grafične kartice) in opcij. Sam postopek je že dlje časa znan in zahteva določen napor, če testne primere tvorimo ročno s pomočjo tabel. Ker obstaja na tržišču večje število orodij, s katerimi si lahko pomagamo pri tej strategiji, smo raziskali učinkovitost in dostopnost izbranih brezplačnih orodij.

**Ključne besede:** strategija testiranja vseh parov, kombinatorno testiranje, ortogonalna polja, tvorjenje testnih vzorcev, računalniška orodja.

## Abstract

### Pairwise Testing Tools

In the paper pairwise testing strategy is briefly described. The pairwise testing strategy is a systematic test case design based on the assumption, that presence of a fault will be discovered when particular pairs of input values are used. The procedure has been well known for a long time and demands a particular effort if test cases are designed manually. As there are many commercial tools on the market that support this strategy, we analyzed and compared the efficiency and availability of selected free pairwise testing tools.

**Key words:** pairwise testing strategy, combinatorial testing, orthogonal arrays, test case generation, test case generation tools.

## 1 UVOD

**Če program testiramo z vsemi mogočimi vhodnimi podatki, smo izvedli totalno testiranje in s tem odkrili vse nepravilnosti. Tako lahko tudi dokažemo popolno korektnost programa. Ker je to v večini primerov praktično neizvedljivo oz. predrago, je naloga preverjevalca, da na podlagi raznih testnih strategij tvori manjše število učinkovitih testnih primerov. Manjšanje števila testnih primerov večja nevarnost, da bodo kljub testiranju ostale še neodkrita nepravilnosti. Na ta kompromis pa vpliva tudi pričakovana kakovost programa. Če je zahtevana kakovost velika, moramo ustrezno povečati zahteve glede temeljitosti testiranja. Sistematično načrtovanje testnih primerov<sup>1</sup> je pomembna disciplina, ki omogoča pregled nad temeljitostjo, stroški in učinkovitostjo.**

Najbolj pomembni elementi testnega primera so začetno stanje, vhodni podatki in pričakovano obnašanje. Vhodni podatek je vsak dogodek, vrednost ali stanje zunanjih entitet, ki vpliva na delovanje programa oz. naprave. Glede na število mogočih vrednosti, ki ji lahko zavzame vhodni podatek, jih lahko razdelimo v te skupine: binarni podatek (npr. spol – mo-

ški/ženska, teža – maksimalna/minimalna), majhno število diskretnih vrednosti (npr. operacijski sistem – Windows XP, Windows 7, Linux) in zelo veliko število diskretnih vrednosti (npr. 8-bitno število).

Strategija, ki jo bomo obravnavali podrobneje, je uporabna le za vhodne podatke, stanja ali vrste komponent, ki jih lahko razvrstimo v prvi dve skupini. Hkrati bomo preizkusili dve orodji, ki tvorita testne vzorce po strategiji testiranja vseh parov.

Prispevek je razdeljen na pet razdelkov. Ker strategijo testiranja vseh parov uvrščamo v večjo skupino, ki ji pravimo kombinatorno testiranje, je le-to na kratko predstavljeno v drugem razdelku. Tretji razdelek obravnava temeljne lastnosti in probleme testiranja vseh parov. Iz nabora brezplačnih orodij smo izbrali dva primerka in napravili kratko primerjalno analizo o njihovi uporabljivosti. Rezultati primerjave so v četrtem razdelku.

## 2 KOMBINATORNO TESTIRANJE

Kombinatorno testiranje (angl. combinatorial testing) je testiranje raznih kombinacij vhodnih podatkov. Vsaka kombinacija predstavlja en testni vzorec. Zelo pogosto ga uporabljamo pri testiranju raznih ra-

<sup>1</sup> Več o testiranju programske opreme je v Beizer (1990), Jorgensen (2001), Dogša (1994) ter Bath in McKay (2008).



čunalniških konfiguracij (npr. različni operacijski sistemi, tiskalniki, grafične kartice) in opcij. V nadaljevanju bomo uporabljali samo izraz vhodni podatek,<sup>2</sup> čeprav bodo vse ugotovitve veljale tudi za stanja in vrste komponent.

Tabela 1 prikazuje enostaven program, ki prebere tri vhodne podatke X, Y in Z. Ker lahko vsak od njih zavzame dve vrednosti, je mogočih kombinacij oz. testnih vzorcev  $2 \cdot 2 \cdot 2 = 2^3 = 8$ . Tipičen program ni tako preprost, saj zahteva večje število vhodnih podatkov, kar pomeni, da obstaja zelo veliko mogočih kombinacij.

Tabela 1: Nabor testnih vzorcev za program, ki prebere tri vhodne podatke, od katerih lahko zavzame vsak po dve različni vrednosti

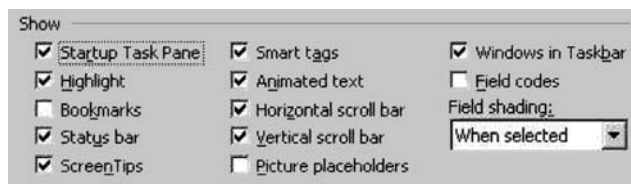
Testni vzorec	Vhod $X = \{1, 2\}$	Vhod $Y = \{A, B\}$	Vhod $Z = \{10, 20\}$
1	1	A	10
2	1	A	20
3	1	B	10
4	1	B	20
5	2	A	10
6	2	A	20
7	2	B	10
8	2	B	20

Najprej bomo izračunali število vseh mogočih kombinacij za poljubno število vhodnih podatkov. Vse vhodne podatke pregledamo in jih razvrstimo glede na število mogočih vrednosti. Naj bo  $m$  največje število mogočih vrednosti nekega vhodnega podatka in  $k(i)$  število vhodnih podatkov, ki lahko zavzamejo  $i$  mogočih vrednosti. Število vseh mogočih kombinacij vrednosti vhodnih podatkov  $N$  izračunamo s pomočjo preproste enačbe:

$$N = \prod_{i=1}^m i^{k(i)}$$

V primeru, ki je prikazan na sliki 1, imamo 12 parametrov, ki lahko zavzamejo dve vrednosti ('da' ali 'ne') in enega, ki zavzame tri (*never, always, when selected*). Če hočemo program preveriti z vsemi mogočimi kombinacijami, moramo tvoriti  $N = 2^{12} \cdot 3^1 = 12.228$  testnih primerov. Iz enačbe je razvidno, da je število kombinacij najbolj občutljivo na spremembo števila mogočih vrednosti. Če bi v prejšnjem zgledu 12 para-

metrov zavzelo namesto dveh tri mogoče vrednosti, bi se število kombinacij povečalo 130-krat. Če bi dodali samo en parameter, pa samo dvakrat. Ker število kombinacij skokovito narašča s številom mogočih vrednosti, je kombinatorno testiranje primerno za tiste sisteme, pri katerih lahko vsak podatek zavzame majhno število mogočih vrednosti. Za druge sisteme ima preverjevalec na voljo niz strategij, kot so mejne vrednosti, prepovedane vrednosti, ekvivalentni razredi itn.



Slika 1: Dialog opcijskih nastavitvev za MS Word (Bach in Schroeder, 2004)

Ukaz DIR (slika 2) pozna 18 različnih vhodnih podatkov (parametrov).<sup>3</sup> Največ vrednosti<sup>4</sup> lahko zavzame stikalo /A. Ker gre za majhno število mogočih vrednosti, ki jih lahko zavzame posamezen parameter, je ta ukaz primeren kandidat za testiranje vseh parov. Število vseh mogočih kombinacij oz. testnih vzorcev je 21,626.880.

```
DIR [drive:] [path] [filename]
[/A[[:]attributes]] [/B] [/C] [/D] [/L] [/N]
[/O[[:]sortorder]] [/P] [/Q] [/R] [/S]
[/T[[:]timefield]] [/W] [/X] [/4]
```

/A : D, H, S, L, R, A, I, -D, -H, -S, -L, -R, -A, -I

/C : /-C

/O : N, E, G, S, D, -N, -E, -G, -S, -D

/T : C, A, W

Slika 2: Vhodni podatki za ukaz DIR

Ker je v večini primerov testiranje programa z vsemi mogočimi kombinacijami vhodnih podatkov praktično neizvedljivo oz. predrago, je naloga preverjevalca, da tvori manjše število učinkovitih testnih primerov. Ena izmed možnosti je, da program, ki ima  $w$  vhodnih podatkov, preverimo z vsemi kombinacijami  $n$ -teric, za katere velja, da je  $n < w$ . Npr. program,

<sup>2</sup> Pogosto se uporabljajo naslednji sinonimi: vhodni podatek = faktor, vrednost = opcija.

<sup>3</sup> Windows 7.

<sup>4</sup> 14 vrednosti + parametra ne navedemo = 15 mogočih vrednosti.

ki ima 8 vhodnih podatkov, preverimo z vsemi kombinacijami trojic. Število najdenih nepravilnosti z večanjem  $n$  narašča, vendar zelo počasi (Kuhn, Wallace in Gallo, 2004). Raziskave so pokazale (Black, 2007; Stobie, 2005), da če namesto vseh mogočih kombinacij uporabimo samo vse mogoče pare ( $n = 2$ ), učinkovitost testnih primerov bistveno ne pade, število testnih primerov pa se izrazito zmanjša. Tako se v praksi najpogosteje odločamo za testiranje parov, saj je dovolj učinkovito, hkrati pa zanj uporabimo veliko manj testnih vzorcev kot za testiranje  $n$ -teric z višjim številom  $n$ . Pri tvorjenju testnih vzorcev si pomagamo s tabelami (ortogonalnimi polji<sup>5</sup>) ali pa s posebnimi računalniškimi programi.

### 3 STRATEGIJA TESTIRANJA VSEH PAROV

Vsaka strategija temelji na določeni predpostavki<sup>6</sup> (angl. bug assumption), ki se nanaša na morebitno nepravilnost. Strategija testiranja parov (angl. pairwise testing, testing all-pairs, 2-way testing) je sistematičen postopek izbora vhodnih podatkov, ki predpostavlja, da so v programu takšne napake, katerih navzočnost se bo pokazala pri ustrezni kombinaciji samo dveh vhodnih vrednosti, neodvisno od ostalih vrednosti. Ta predpostavka o nepravilnosti temelji na dveh možnostih: določena kombinacija ne sme povzročiti nepravilnega delovanja (npr. menjava vrste tiskalnika) ali pa mora vplivati na izhodno vrednost (npr. vrsta bencina na stroške prevoza). Ker ne vemo, pri katerem paru se bo to zgodilo, program preizkusimo z vsemi pari.

Za primer, ki je prikazan v tabeli 1, tvorimo pare med dvema vhodnima podatkom:  $(X, Y)$ ,  $(X, Z)$  in  $(Y, Z)$ . Za vhoda  $X$  in  $Y$  obstajajo natanko štiri pari:  $(1, A)$ ,  $(1, B)$ ,  $(2, A)$ ,  $(2, B)$ . Če ustrezno izbiramo še ostale pare, potrebujemo za naš primer namesto 8 le 4 testne primere (v tabeli 1 so označeni s sivo). S tem smo zmanjšali število testnih primerov za 50 odstotkov. Kot je razvidno iz tabele 1, je vsak par vhodnih podatkov zastopan v vsaj enem od testnih vzorcev. Večje kot je število vhodnih podatkov in njihovih mogočih vrednosti, večja je razlika med številom vseh mogočih kombinacij vhodnih vrednosti ter številom vseh parov. Če imamo npr. štiri vhodne podatke, od katerih lahko zavzame vsak tri vrednosti, obstaja  $3^4 = 81$  kombinacij oz. potrebujemo 81 testnih prime-

rov. Če se odločimo za pare, lahko testiranje izvedemo samo z devetimi testnimi primeri. Za primere, pri katerih obstaja majhno število vseh mogočih kombinacij, lahko vse pare poiščemo ročno, pri večjem številu kombinacij pa je treba uporabiti posebne generatorje testnih primerov.

#### 3.1 Učinkovitost testiranja vseh parov

Kljub relativno majhnemu številu testnih primerov raziskave kažejo, da lahko z njimi odkrijemo večino nepravilnosti na testiranem sistemu, česar ne dosežemo z ročnimi testi (Stobie, 2005; Tai in Lei, 2002). Metoda testiranja vseh parov je zaradi tega v zadnjih letih postala zelo popularna. Kljub tej popularnosti pa nekateri strokovnjaki opozarjajo na prevelika pričakovanja (Bach in Schroeder, 2004).

Učinkovitost metode testiranja parov je v svoji raziskavi potrdil Justin Hunter (Kuhn, Kracker, Lei in Hunter, 2009). V raziskavo je vključil deset različnih projektov, na katerih je primerjal uspešnost testiranja vseh parov z uspešnostjo testiranja z ročnimi metodami. Projekte je vodilo šest družb, ki so testirale aplikacije v razvoju. V vsakem projektu sta sodelovali dve skupini testerjev, ki sta istočasno testirali isto aplikacijo, vendar vsaka z drugimi testirnimi metodami. Prva je uporabljala običajne testirne strategije (npr. specifikacijsko testiranje), druga skupina pa testiranje vseh parov, pri kateri so si pomagali z generatorji testnih primerov.

Rezultati raziskave so pokazali, da je bila učinkovitost testiranja z vsemi pari večja kot pri testiranju z drugimi metodami. Vse nepravilnosti, ki so jih odkrili z ročnim izbiranjem testnih primerov, so bile najdene tudi v skupinah, ki so projekte testirale s testiranjem vseh parov. V petih projektih so te skupine odkrile še dodatne nepravilnosti, ki jih nasprotno skupine niso. Tudi primerjava časa, ki so ga posamezne skupine porabile, gre v korist strategije testiranja vseh parov.

#### 3.2 Pogostost kombinacij

Učinkovitost testiranja lahko povečamo, če poznamo profil uporabe, ki pa le redko ustreza enakomerni porazdelitvi. Kadar so vhodni parametri določene nastavitve sistema, izkušnje kažejo, da uporabniki v večini primerov uporabljajo privzete nastavitve in jih le redko spremenijo. Če spremenijo katero izmed nastavitvev, spremenijo najpogosteje le en parameter ali dva. Za določene kombinacije nastavitvev obstaja

<sup>5</sup> Angl. orthogonal array testing.

<sup>6</sup> Več o tem v Beizer (1990) ali Dogša (2001).

večja verjetnost, da se bodo izvedle, kot za druge (Bryce in Colbourn, 2006). Če ta podatek poznamo, lahko določimo prioriteto izvrševanja testnih primerov. S testiranjem začnemo s kombinacijami, ki se v praksi najpogosteje uporabljajo. Kadar imamo zelo veliko število testnih primerov, lahko izpustimo tiste kombinacije, za katere velja zelo majhna verjetnost uporabe.

### 3.3 Pomen medsebojne povezave vhodnih podatkov

Če je  $n = 1$ , imamo najbolj preprosto strategijo, kar pomeni, da npr. že samo nastavitev enega parametra povzroči odpoved. V raziskavi (Kuhn, Wallace in Gallo, 2004) so odkrili, da je en parameter povzročil 30 do 60 odstotkov nepravilnosti. Pri  $n = 2$  (testiranje vseh parov), pa ta učinkovitost naraste na 70 do 90 odstotkov (slika 3).

Pri odločitvi glede  $n$  je pomembno, da vemo, kateri vhodni podatki vplivajo na določeni izhod. Kadar ocenimo, da je to število majhno (npr. 2), je testiranje vseh parov dobra izbira testirne metode. Kadar pa je to število večje (8 in več), testiranje vseh parov zagotovo ni več dovolj učinkovito (Bach in Schroeder, 2004). V tem primeru je treba povečati  $n$  in si pri načrtovanju testnih primerov pomagati z ortogonalnimi polji.

### 3.4 Problem oraklja

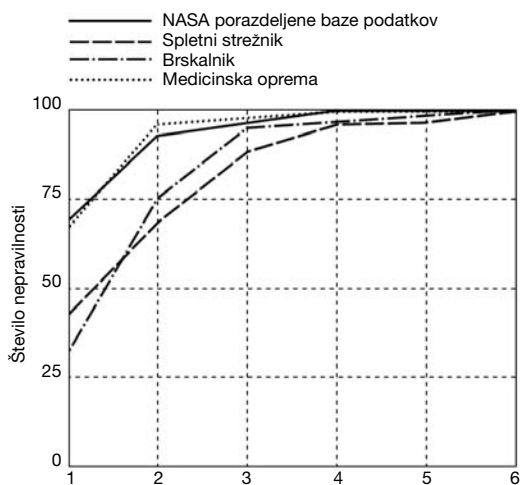
Načrtovanje testnih podatkov (vhodne vrednosti) je mogoče avtomatizirati pri katerem koli kombinacijskem testiranju, kar pa ne velja za pričakovano obnašanje. To je treba določiti s pomočjo oraklja drugače (Kuhn, Lei in Kacker, 2008). S pomočjo tabel ali orodij lahko hitro določimo testne primere, dosti več napora pa moramo vložiti v določitev pričakovanega ob-

našanja. Pogosto se to določevanje ne izvaja, ker preverjevalci predpostavljajo, da bo nepravilnost tako očitna, da jo bodo zlahka opazili.

## 4 GENERATORJI PAROV

V generatorjih, ki na podlagi strategije vseh parov tvorijo testne vzorce, so implementirani različni algoritmi. Tabela 2 prikazuje učinkovitost najpomembnejših orodij, ki je bila izmerjena na istih primerih. Takoj je opazno izrazito znižanje števila potrebnih testnih primerov, npr. s 1020 na 180. V povprečju je dalo najboljše rezultate orodje TestCover, ki ga prodaja podjetje Testcover.com.

Med prostodostopnimi orodji so npr. Jenny (podjetje Jenkins), Allpairs (podjetje Satisfice), Allpairs (podjetje McDowell), Allpairs (podjetje MetaCommunications), PICT (podjetje Microsoft) in Pairwise TestCase Generator (podjetje TestersDesk).



Slika 3: Število nepravilnosti v odvisnosti od števila povezav med vhodnimi podatki (Kuhn, Wallace in Gallo, 2004)

Tabela 2: Primerjava učinkovitosti orodij za generiranje vseh parov (Czerwonka, 2009) za šest zgljedov (prva kolona). V kolonah 3 do 11 je število testnih vzorcev, ki jih je tvorilo določeno orodje.

Zgled – število vhodnih podatkov in njihovih vrednosti	Število vseh mogočih kombinacij	AETG	IPO	Tconfig	CTS	Jenny	TestCover	DDA	AllPairs (McDowell)	PICT	EXACT	IPO-s
$3^4$	81	9	9	9	9	11	9	?	9	9	9	9
$3^{13}$	$1,6 \cdot 10^6$	15	17	15	15	18	15	18	17	18	15	17
$4^{15} 3^{17} 2^{29}$	$7,4 \cdot 10^{25}$	41	34	40	39	38	29	35	34	37	?	32
$4^1 3^{39} 2^{35}$	$5,5 \cdot 10^{29}$	28	26	30	29	28	21	27	26	27	21	23
$2^{100}$	$1,2 \cdot 10^{30}$	10	15	14	10	16	10	15	14	15	10	10
$10^{20}$	$10^{20}$	180	212	231	210	193	181	201	197	210	?	220

V naslednjih razdelkih bomo natančneje primerjali dve prostodostopni orodji: Allpairs (McDowell) («All-pairs Testing» n. d.) in Pairwise TestCase Generator (TestersDesk.com, n. d.), katerih učinkovitost smo preverili na dveh primerih. Za prvi primer smo

vzeli ukaz DIR (slika 2), ki ima 18 vhodnih podatkov. V drugem primeru smo testirali sistem, ki vsebuje dvajset vhodnih podatkov, od katerih lahko vsak zasede 10 mogočih vrednosti.

Tabela 3: **Tabela preslikav za primer ukaza DIR**

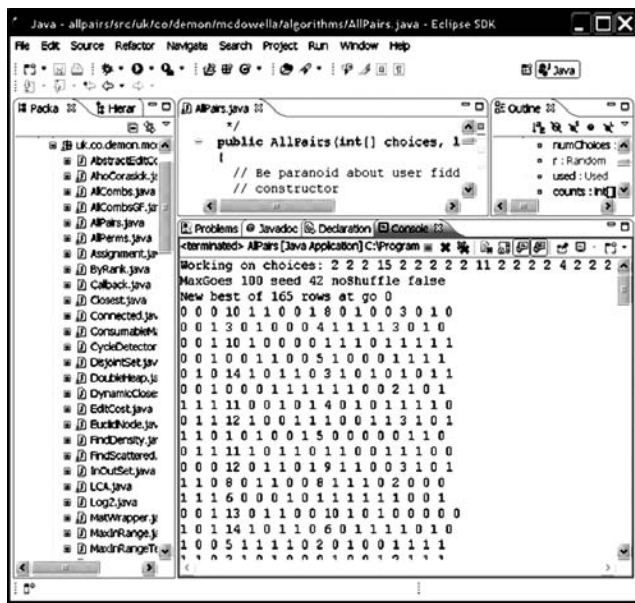
Preslikava																
Parameter		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Indeks	Ime															
1.	[drive:]	Stikalo ni uporabljeno.	Stikalo je uporabljeno.													
2.	[path]	Stikalo ni uporabljeno.	Stikalo je uporabljeno.													
3.	[filename]	Stikalo ni uporabljeno.	Stikalo je uporabljeno.													
4.	[/A[[:attributes]]]	Stikalo ni uporabljeno.	D	H	S	L	R	A	I	-D	-H	-S	-L	-R	-A	-I
5.	[/B]	Stikalo ni uporabljeno.	Stikalo je uporabljeno.													
6.	[/C]	Stikalo ni uporabljeno.	/-C													
...	...	Stikalo ni uporabljeno.	Stikalo je uporabljeno.													
10.	[/O[[:sortorder]]]	Stikalo ni uporabljeno.	N	E	G	S	D	-N	-E	-G	-S	-D				
...	...	Stikalo ni uporabljeno.	Stikalo je uporabljeno.													
15.	[/T[[:timefield]]]	Stikalo ni uporabljeno.	C	A	W											
...	...	Stikalo ni uporabljeno.	Stikalo je uporabljeno.													
18.	[/4]	Stikalo ni uporabljeno.	Stikalo je uporabljeno.													

#### 4.1 AllPairs (MC Dowell)

AllPairs je orodje, ki tvori kombinacije vseh parov. Orodje je izdelano v programskem jeziku java in ne vsebuje grafičnega vmesnika. Zagon orodja AllPairs smo izvedli v orodju Eclipse.

Vhodni podatek aplikacije Allpairs je niz pozitivnih števil, ki predstavljajo preprost opis vhodnih podatkov. Ker ne ponuja možnosti vnosa dejanskih vrednosti vhodnih podatkov, moramo izdelati tabelo preslikav, pri čemer je vsaki vrednosti vhodnega podatka prirejeno celo pozitivno število (glej tabelo 3).

Poglejmo prvi primer, ki ga prikazuje slika 2 in ima 18 vhodnih podatkov. S pomočjo tabele 3 tvorimo niz celih števil, s katerimi zaženemo AllPairs: 2 2 2 15 2 2 2 2 2 11 2 2 2 4 2 2 2. Ta niz je v bistvu opis vhodne domene. Prvi trije vhodni podatki lahko zavzamejo vrednosti 0 ali 1, četrti 0, 1, 2 ... 14, peti zopet 0 ali 1 itn. Program vrne 165 kombinacij oz. testnih vzorcev, ki pokrivajo vse pare vhodnih podatkov testiranega sistema (slika 5).



Slika 5: Rezultat tvorjenja vseh parov s pomočjo orodja AllPairs

Rezultat moramo nato prevesti s pomočjo tabele 3 v konkretne vrednosti vhodnih podatkov. Ta pomanjkljivost je moteča, predvsem kadar je število vhodnih podatkov in njihovih vrednosti veliko.

Z orodjem AllPairs smo tvorili tudi testne vzorce za drugi primer. Nastalo je 198 testnih vzorcev, kar je veliko manj od števila vseh mogočih kombinacij, ki je  $10^{20}$ . Primer nam pokaže, da metoda testiranja vseh parov učinkovito zmanjša število testnih vzorcev.

## 4.2 Pairwise TestCase Generator

Drugo orodje za testiranje vseh parov, ki smo ga preizkusili, je Pairwise TestCase Generator (PTCG). Do orodja dostopamo prek grafičnega spletnega vmesnika (<http://www.testersdesk.com/>). Za brezplačno uporabo tega orodja je potrebna registracija v sistemu. Kot registrirani uporabnik lahko izbiramo med šestimi različnimi programi za generiranje testnih vzorcev, ki uporabljajo različne metode: Pairwise TestCase Generator za testiranje po metodi vseh parov, T-way TestCase Generator, N-wat/Random TestCase Generator, Subset TestCase Generator, Boundary TestCase Generator (BVA++) ter TestCase Permutations Generator.

PTCG omogoča vnos vseh mogočih vhodnih podatkov ter njihovih vrednosti, ki jih vpišemo v navadno besedilno datoteko, tako da ne potrebujemo tabele preslikav kot pri uporabi orodja AllPairs. Zapis vhodnih podatkov mora biti oblikovan tako, kot

prikazuje slika 6. Prednost tega orodja v primerjavi z orodjem AllPairs je v tem, da so lahko vhodni podatki s pripadajočimi vrednostmi zapisani v besedilni obliki. To omogoča lažjo povezljivost z dejanskimi vhodnimi podatki testiranega sistema.

Po vnosu vhodnih podatkov in njihovih pripadajočih vrednosti v PTCG lahko izbiramo med izpisom rezultata na zaslon ter izpisom v formatu CSV.<sup>7</sup> Slika 6 prikazuje prvi način izpisa. Rezultat generiranja testnih vzorcev za primer ukaza DIR je 165.

```
drive:0,drive
path:0,path
filename:0,filename
A:0,D,H,S,L,R,A,I,-D,-H,-S,-L,-R,-A,-I
B:0,B
C:0,-C
D:0,D
L:0,L
N:0,N
O:0,N,E,G,S,D,-N,-E,-G,-S,-D
P:0,P
Q:0,Q
R:0,R
S:0,S
T:0,C,A,W
W:0,W
X:0,X
4:0,4
```

Slika 6: Oblika zapisa vhodnih podatkov ukaza DIR, pripravljenih za uporabo v programu PTCG

Orodje PTCG je za omenjeni primer generiralo isto število testnih vzorcev kot orodje AllPairs. Ob podrobnem pregledu rezultatov obeh orodij pa ugotovimo, da se testni vzorci, ki sta jih generirali orodji, razlikujejo med seboj.

Iz slike 7 je razvidno, da je orodje PTCG v 25. do 28. testnem vzorcu za nekatere vhodne vrednosti zapisalo flany\_value\_of\_X« (pri čemer je X ime vhodnega podatka). To pomeni, da lahko uporabnik sam izbere vrednost vhodnega podatka X, saj bo v vsakem primeru zagotovljena popolna pokritost vseh parov. Aplikacija izračuna tudi število vseh mogočih kombinacij vhodnih podatkov.

<sup>7</sup> Angl. comma-separated values – preprost besedilni format, v katerem so podatki ločeni z decimalno vejico.

Slika 7: Testni vzorci za primer vhodnih podatkov s slike 6, ki jih je tvorilo orodje Pairwise Test Case Generator

Z orodjem PTCG smo generirali tudi testne vzorce za drugi primer z dvajsetimi vhodnimi podatki. Kot rezultat smo dobili 215 kombinacij testnih vzorcev, kar je 17 več kot smo jih dobili s pomočjo orodja

	AllPairs (McDowell)	PTCG
Grafični uporabniški vmesnik	Ne	Da
Vrsta dostopa	Namizna aplikacija	Spletna aplikacija
Možnost vnosa dejanskih vrednosti vhodnih podatkov	Ne (potrebna je izdelava tabele preslikav)	Da
Možnost vnosa vhodnih podatkov iz besedilne datoteke	Ne	Da
Možnost izvoza testnih vzorcev v formatu CSV	Ne	Da
Izračun vseh mogočih kombinacij vhodnih podatkov	Ne	Da
Število generiranih testnih vzorcev za primer ukaza DIR	165	165
Število generiranih testnih vzorcev za primer dvajsetih vhodnih podatkov z desetimi vrednostmi	198	215

Slika 8: Primerjava orodij AllPairs in PTCG

AllPairs. V tem primeru lahko sklepamo, da je algoritem orodja AllPairs učinkovitejši, vendar pa sta obe orodji nalogo opravili zelo uspešno, če rezultat primerjamo s številom vseh mogočih kombinacij testnega primera, ki je  $10^{20}$ .

Primerjava obeh orodij (slika 8) pokaže, da je orodje AllPairs bolj učinkovito, kadar tvorimo testne vzorce za večje število vhodnih podatkov. Na drugi strani je PTCG uporabniku bolj prijazno orodje, saj ima preprost grafični uporabniški vmesnik ter možnost branja vhodnih podatkov iz besedilne datoteke, v katero lahko zapišemo dejanske vrednosti vhodnih podatkov, torej ne potrebujemo tabele preslikav. PTCG je dostopna prek spleta, medtem ko je AllPairs namizna aplikacija, za katero potrebujemo okolje java.

## 5 SKLEP

Učinkovito testiranje zahteva, da ocenimo kakovost sistema v čim krajšem času ter s čim nižjimi stroški. Ker imajo različni sistemi svoje specifične značilnosti, ne obstaja univerzalna metoda testiranja, ki bi bila najboljša rešitev za vse sisteme. Metoda testiranja vseh parov je kljub sistematičnosti preprosta in v mnogih primerih zelo učinkovita rešitev za testiranje sistemov z več diskretnimi vhodnimi podatki, ki so medsebojno le rahlo povezani. Da bi znali metodo testiranja vseh parov uporabiti pravilno in učinkovito, moramo poznati predpostavke o napakah oz. nepravilnostih, katerih navzočnost lahko odkrijemo s to strategijo. Na tržišču obstaja niz brezplačnih orodij, ki podpirajo načrtovanje parov oziroma n-teric. Njihova uporabljivost je na takšni ravni, da predstavljajo učinkovit pripomoček za načrtovanje testnih vzorcev.

## 6 LITERATURA

- [1] *All-Pairs Testing*. <http://www.mcdowella.demon.co.uk/allPairs.html>, dostopno 26. 8. 2010.
- [2] Bach, J., Schroeder, P. (2004, oktober). *Pairwise testing: A best practice that isn't*. 22nd Pacific Northwest Software Quality Conference, Portland, 175–193.
- [3] Bath, G., McKay, J. (2008). *The Software Test Engineer's Handbook: A Study Guide for the ISTQB Test Analyst and Technical Analyst Advanced Level Certificates*, Rocky Nook Inc.
- [4] Beizer, B. (1990). *Software Testing Techniques*, Van Nostrand Reinhold.
- [5] Black, R. (2007). *Pragmatic Software Testing: Becoming an Effective and Efficient Test Professional*. New York: Wiley.
- [6] Bryce, R. C., Colbourn, C. J. (2006, oktober). *Prioritized interaction testing for pair-wise coverage with seeding and constraints*. Information and Software Technology, 48(10), 960–970.

- [7] Cohen, D. M., Dalal, S. R., Fredman, M. L., Patton, G. C. (1997). *The AETG system: An approach to testing based on combinatorial design*. IEEE Transactions On Software Engineering, 23(7).
- [8] Colbourn, C. J., Cohen, M. B., Turban, R. C. (februar, 2004). *A Deterministic Density Algorithm for Pairwise Interaction Coverage*. Proc. IASTED International Conf. Software Engineering, Innsbruck Austria, 245–252.
- [9] Czerwonka, J. (2009). *Pairwise Testing: Combinatorial Test Case Generation*. <http://www.pairwise.org/>, dostopno 26. 8. 2010.
- [10] Czerwonka, J. (2006, oktober). *Pairwise testing in real world*. In PacificNorthwest Software Quality Conference, 419–430.
- [11] Dogša, T. (2001). *Splošen koncept načrtovanja testnih primerov*, Uporabna informatika, šte. 2, letnik IX, 2001, 75–79.
- [12] Dogša, T. (1994). *Verifikacija in validacija programske opreme*, Tehniška fakulteta Maribor.
- [13] Jorgensen, Paul C. (2001). *Software testing – A Craftsman's Approach*, CRC Press LLC.
- [14] Kuhn, R., Kacker, R., Lei, Y., Hunter, J. (2009, avgust). *Combinatorial Software Testing*. Computer, 42(8), 94–96.
- [15] Kuhn, R., Lei, Y., Kacker, R. (2008, maj/junij). *Practical Combinatorial Testing: Beyond Pairwise*. IT Professional Magazine. Washington, 10(3), 19.
- [16] Kuhn, D. R., Wallace, D. R., Gallo, A. J., Jr. (2004, julij). *Software Fault Interactions and Implications for Software Testing*. IEEE Transactions on Software Engineering, 30(6).
- [17] *Pairwise TestCase Generator from TestersDesk.com*. [http://www.testersdesk.com/pairwise\\_testersdesk.html](http://www.testersdesk.com/pairwise_testersdesk.html), dostopno 26. 8. 2010.
- [18] Stobie, K. (2005, februar). *Too darned BIG to test*. New York: ACM. Queue, 3(1), 30–37.
- [19] Tai, K. C., Lei, Y. (2002, januar). *A test generation strategy for pairwise testing*. IEEE Transactions on Software Engineering. New York, 28(1), 109.

■

Tomaž Dogša je izredni profesor na Fakulteti za elektrotehniko, računalništvo in informatiko Univerze v Mariboru, kjer predava na dodiplomski in podiplomski stopnji in vodi Center za verifikacijo in validacijo sistemov. Na raziskovalnem področju se ukvarja predvsem s tehnologijo V&V oz. testirnimi orodji.

■

Mirjam Bonačič je asistentka za področje informatike na Fakulteti za elektrotehniko, računalništvo in informatiko Univerze v Mariboru, kjer sodeluje pri predmetih s področja podatkovnih baz. Hkrati je študentka magistrskega študija na omenjeni fakulteti. Aktivno se udeležuje konferenc in je avtorica in soavtorica številnih člankov v domačih in tujih revijah.

# Migracija wordovih dokumentov

Tomaz Stensak<sup>1</sup>, Tomaz Dogša<sup>2</sup>

<sup>1</sup> BSH, d. o. o., tomaz.stensak@bshg.com

<sup>2</sup> Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko, tdogsa@uni-mb.si

## Izvelek

Digitalni dokumenti bodo vsaj deloma nadomestili papir samo v primeru, ko bo zagotovljeno, da bodo berljivi prostorsko in časovno neodvisno. Berljivost starejših digitalnih dokumentov ohranjamo z ustrežno migracijo. V članku so predstavljeni rezultati testiranja migracije dokumentov, ki so bili tvorjeni s starejšimi različicami urejevalnika Word. Pri migraciji dokumentov, tvorjenih z urejevalnikom *Word 5.5* (različica za DOS), se je popolnoma ohranilo le povprečno 20 odstotkov obravnavanih lastnosti. Testiranje je pokazalo, da je za te dokumente najbolj primerna postopna migracija. Zelo uspešno je bilo mogoče migrirati dokumente, ki so bili tvorjeni z urejevalnikom *Word 2.0* (različica za Windows), pri katerih se je popolnoma ohranilo povprečno 89 odstotkov obravnavanih lastnosti.

**Ključne besede:** migracija, digitalni dokumenti, vzdrževanje dokumentov, ohranjanje dokumentov, formati dokumentov.

## Abstract

### Migration of Word Documents

Digital documents will, at least partially, replace paper documents only if they remain readable in the same way regardless of time and space, which means that old documents need to be transformed. Readability of older digital documents can be kept by proper migration. In this article are presented the results of old Word documents migration tests. On average, after migration of documents created in *Word 5.5* (version for DOS) only 20 % of document properties were completely preserved. The test shows that gradual migration is most suitable for these documents. The migration of documents created in *Word 2.0* (version for Windows) was very successful with on average 89 % of properties being completely preserved.

**Key words:** migration, digital documents, document maintenance, document preservation, document formats.

## 1 UVOD

**Prvi digitalni dokumenti so bile preproste besedilne datoteke, ki so vsebovale samo golo besedilo. Z razvojem urejevalnikov se je kompleksnost strukture dokumentov neprestano večala. Zaradi vključevanja raznih objektov (slik, formul itd.) v dokument se format zapisa še danes neprestano spreminja. Word 2003 je poznal sedem različnih formatov, verzija 2007 pa jih ima že petnajst. Ker število dokumentov skokovito narašča, jih je treba ustrezno hraniti in omogočiti preprosto iskanje, branje in urejanje. Ali bo mogoča uporaba sedanjih oz. še starejših dokumentov tudi po daljšem obdobju (npr. čez 20 let)? Povprečnemu uporabniku takrat današnja programska in strojna oprema ne bosta več uporabni niti dostopni.**

Načinov ohranjanja digitalne dokumentacije je več. Najbolj preprosta metoda je prenos na papir, ki pa od treh zahtev (branje, prostorsko in časovno neodvisno iskanje, urejanje) zagotavlja samo branje. Postopek, pri katerem dokumente, ki so bili tvorjeni v starem formatu, pretvorimo v novejši format, imenujemo migracija. Migracija je eden najbolj priljubljenih in razširjenih načinov ohranjanja digitalnih dokumentov [2].

Rezultat idealne migracije je dokument, ki je berljiv, ima ohranjeno integriteto v semantičnem in oblikovnem smislu in ga je mogoče ponovno urejati. Če je ciljni format kompatibilen z izvornim, se prenesejo vse pomembne lastnosti digitalnega dokumenta v novo različico. Tako je ohranjena verodostojnost in integriteta dokumenta.

Ker je treba učinkovitost vsake migracije predhodno preveriti, smo analizirali učinkovitost migracije dokumentov, ki so bili tvorjeni s starejšimi različicami urejevalnika besedil Word. Uspešnost migracije odvisna tudi od strukture dokumentov. Dokument, ki vsebuje samo preprosto besedilo, lahko v večini primerov brez težav prenesemo v novejši format. Zato smo za testne dokumente izbrali takšne, ki so bili sestavljeni iz treh segmentov: slogovno oblikovanega besedila, drugih objektov (slik, formul itd.) ter podatkov o dokumentu (avtor, datum nastanka itd.). V nadaljevanju so najprej predstavljeni primeri iz splošne prakse.



## 2 MIGRACIJA V PRAKSI

### 2.1 Problematika migracij digitalnih dokumentov

Migracija je periodično pretvarjanje digitalnih vsebin iz ene strojne ali programske konfiguracije v drugo ali iz predhodnih generacij računalniških tehnologij v kasnejše. [8] Glavni problem migracije je, da zahteva specializirano preverjanje parov formatov izvornih in končnih datotek dokumentov. Le tako jo je mogoče izvesti kontrolirano in zanesljivo. [2]

Za dokumente, ki jih je treba ohranjati dlje časa, je potrebnih več migracijskih ciklov. Pri tem se pojavi vprašanje, kdaj začeti z migracijskimi cikli in kako pogosto jih izvajati. Migracija dokumentov je potrebna vsakič, ko to zahteva sprememba tehnologije ne glede na to, ali je v tistem trenutku znano, ali bodo določeni dokumenti še uporabni ali ne. Poleg tega je treba zagotoviti tudi ustrezno migracijo metapodatkov, ki se navezujejo na dokumente. Vse to otežuje pripravo stroškovnega modela brez katerega migracija ni izvedljiva. Migracija je zahtevna tudi glede zagotavljanja kadrov, razen kadar jo je mogoče izvajati popolnoma avtomatizirano. [2]

Migracija digitalnih dokumentov je večkrat kritizirana zaradi neznanih učinkov, ki jih lahko ima proces na verodostojnost in integriteto dokumentov. Pri vsakem kopiranju lahko pride do napake zaradi hroščev v programu, nepravilnega ravnanja z datotekami ali zaradi mehanske okvare podatkovnih nosilcev. Kljub vsemu izsledki raziskav kažejo, da je migracija s pravilno oceno tveganja primerna metoda za dolgoročno hranjenje večine digitalnih dokumentov. [2]

### 2.2 Raziskave in projekti na temo migracije digitalnih dokumentov

V zadnjih desetih letih je bilo na področju migracij digitalnih dokumentov izvedenih veliko raziskav, kar je omogočilo identifikacijo glavnih problemov ter tveganj. Prihodnji razvoj je tako naravnani v iskanje najprimernejših rešitev.

Eden pomembnejših projektov je bil CAMiLEON. Začeli sta ga skupaj univerzi Michigan (ZDA) in Leeds (VB) leta 1999. Ena izmed nalog je bila priprava smernic uporabe emulacije in migracije za ohranjanje digitalnih dokumentov. [9]

Projekt InterPARES (The International Research on Permanent Authentic Records in Electronic Systems) je osnovan na univerzi British Columbia v

Kanadi. Med drugim se ukvarja s pripravo smernic kot podlago za standarde, strategije, politiko, ki zagotavljajo dolgoročnost digitalnih dokumentov in možnost, da lahko uporabniki zaupajo v njihovo verodostojnost. [10] Kljub temu da se njihovi dokumenti ne nanašajo neposredno na migracijo, je predlagane smernice in modele mogoče neposredno uporabiti za pripravo ustrezne strategije migracije digitalnih dokumentov.

Nizozemski narodni arhivi (National Archives of the Netherlands) so leta 2000 vzpostavili projektno skupino Digital Preservation Testbed. Ukvarja se z raziskovanjem različnih metod dolgoročnega ohranjanja digitalnih dokumentov. [14] V letih 2000 do 2003 so izvajali projekte testiranja migracij besedilnih dokumentov, baz podatkov in preglednic v različne formate. Rezultati so pokazali, da migracija s povratno združljivostjo ni primerna za dolgoročno ohranjanje digitalnih dokumentov (50 let in več). Migracija besedilnih dokumentov v XML se je izkazala kot zelo dobra, kadar so dokumenti tvorjeni na podlagi vnaprej pripravljenih predlog. Pri migraciji besedilnih dokumentov v PDF so ugotovili, da je to najboljša strategija, če pri tvorbi dokumentov niso bile uporabljene vnaprej pripravljene predloge. Migracija v XML je bila spoznana kot najboljša strategija za migracijo baz podatkov in preglednic. [11]

Narodni arhivi Avstralije (National Archives of Australia) se od leta 1994 aktivno ukvarjajo z raziskovanjem in razvojem rešitev za dolgoročno hranjenje digitalnih dokumentov. [12] Njihove raziskave na področju migracije so osredinjene na besedilne datoteke in HTML-datoteke, ki so del projekta, ki se ukvarja s spletnimi dokumenti PANDORA (Preserving and Accessing Networked Documentary Resources of Australia). [13]

Našteti projekti in organizacije so eni pomembnejših in so pogosto omenjeni v literaturi, ki obravnava migracijo digitalnih dokumentov. V nadaljevanju bodo predstavljene vrste migracij, ki se pogosto uporabljajo.

## 3 VRSTE MIGRACIJ

Pojavlja se več vrst migracije. V literaturi, ki je trenutno na voljo, se najpogosteje pojavljajo:

- migracija s pomočjo povratne združljivosti (backward compatibility [3]),
- migracija s pomočjo medobratovalnosti (interoperability [3]) in

- migracija v standardne formate (migration to standard formats) [1].

Odločitev, katera oblika je najprimernejša, je odvisna od dolžine obdobja hranjenja dokumentov. Upoštevati je treba vse zahteve za ohranitev verodostojnosti in integritete, kar je ključna zahteva pri upravljanju z digitalnimi dokumenti [4].

### 3.1 Migracija z uporabo povratne združljivosti

Migracija z uporabo povratne združljivosti omogoča preslikavo in pravilno reprodukcijo dokumenta, ki je bil tvoren v starejši različici aplikacije, z uporabo njene kasnejše različice. Migracija je najlažja in najcenejša, kadar razvijalec originalne programske opreme zagotavlja pretvornike v okviru novejših različic programov, npr. Word 2007 lahko bere datoteke tvorjene v Word 95.

Bistvena nevarnost te vrste migracije je, da programi običajno omogočajo migracijo le za dve ali tri predhodne generacije in ne za vse predhodne različice. Poleg tega razvijalci niso dolžni zagotavljati povratne združljivosti in je to odvisno od njihove poslovne strategije. [2]

Običajno je treba izvesti migracijo na novo različico vsakih nekaj let, odvisno od tega, kako pogosto izhajajo nove različice programske opreme. Povratna združljivost je primerna strategija za kratkoročno hranjenje. Primeri takšnega načina migracije sicer ne kažejo večjih težav, če med različicami urejevalnikov ni preteklo preveč časa. S tem verodostojnost ter integriteta večinoma nista ogroženi. [1]

### 3.2 Migracija z uporabo medobratovalnosti

Migracija z uporabo medobratovalnosti pomeni, da so datoteke (dokumenti) prenosljive z ene platforme na drugo in jih je še vedno mogoče prikazati enako. Datoteko je mogoče brati in urejati z uporabo različnih verzij iste aplikacije, ki deluje na različnih operacijskih sistemih. Datoteke, pri katerih uporabljamo za opis besedila samo prvih 127 znakov v tabeli ASCII, so zelo prenosljive. Ker pri njih z migracijo skorajda ni težav, so podlaga za razne označevalne jezike (npr. HTML, RTF). Drugi primer medobratovalnosti je prenos med podobnimi programskimi aplikacijami. Tretja oblika medobratovalnosti temelji na uporabi programskega vmesnika za pretvorbo. To pomeni, da je dokument, tvoren v enem urejevalniku in shranjen v izmenljivem formatu, npr. RTF (Rich Text Format [5]), berljiv v drugem urejevalniku besedil.

Medobratovalnost zaradi številnih pomanjkljivosti ni zanesljiv način za hranjenje digitalnih dokumentov. Uporabna je lahko kot začasna rešitev do uvedbe dolgoročnejšega načina [1].

### 3.3 Migracija s pretvorbo v standardni format

Migracija s pretvorbo v standardni format je migracija lastniškega formata (običajno zaprt in v binarni obliki) v format, ki temelji na javnem standardu in je odprt. Digitalni dokumenti niso več odvisni od originalne programske opreme, s katero so bili tvorjeni. Posledično niso več izpostavljeni nevarnosti zastaranja programske opreme.

Formati so lahko formalni, odprte narave ali akreditirani pri organizacijah za standardizacijo (ISO, NEN, W3C) [2]. Eden takšnih standardnih formatov je XML (Extensible Markup Language [6]). XML je bil razvit z namenom povečanja medobratovalnosti jezikov SGML in HTML [6]. Določeni formati temeljijo na standardih, ki niso uradno priznani, so pa tako razširjeni, da jih uporablja veliko število uporabnikov [2]. Takšna standardna formata sta na primer PDF (Portable Document Format), preden je postal odprti standard, ter format RTF (Rich Text Format [5]).

Format PDF je leta 2008 uradno postal odprti standard, poznan kot ISO 32000. Datoteke v tem formatu so berljive in jih je mogoče natisniti na platformah, kot so Mac OS, Microsoft Windows, UNIX, in na mnogih mobilnih platformah. Datoteke PDF ohranjajo izgled kot izvorniki in ohranijo besedilo, risbe, video, 3D-slike, načrte, fotografije ipd. [7].

Format RTF je metoda kodiranja besedila in grafike za enostaven prenos med aplikacijami. Trenutno so na voljo prevajalniki, ki omogočajo prenos dokumentov med različnimi aplikacijami, ki delujejo v okoljih DOS, Microsoft Windows, OS/2, Macintosh in Power Macintosh [5].

Tudi pri migraciji v standardne formate se pojavljajo težave. Na izbiro je veliko standardov in težko je določiti, kateri je najprimernejši. Poleg tega se standardi spreminjajo in prihajajo nove različice. To pomeni, da je treba tudi v tem primeru migracijo izvajati večkrat. [2]

Kljub vsemu je pretvorba v standardne formate primeren pristop za hranjenje dokumentov. Takšen način omogoča tako povratno združljivost kot medobratovalnost [1].

## 4 TESTIRANJE MIGRACIJE STAREJŠIH WORDOVIH DOKUMENTOV

Glede na izkušnje uporabnikov je znano, da se določene lastnosti pri migraciji starejših dokumentov vedno ne ohranijo popolnoma. Namen testiranj je bil ugotoviti, kakšna migracija omogoča čim bolj popolno ohranitev verodostojnosti in integritete obravnavanih dokumentov.

Izvedena sta bila dva testa. V prvem smo opazovali uspešnost različnih migracij dokumentov, ki so bili tvorjeni z urejevalnikom Word 5.5, ki je deloval v okolju DOS. V drugem testu smo opazovali uspešnost različnih migracij dokumenta, ki je bil tvorjen z urejevalnikom Word 2.0, ki deluje v okolju Windows.

### Glavne lastnosti testnih dokumentov

V okviru testiranj smo opazovali, kako uspešna je migracija enajstih najpomembnejših lastnosti in elementov testnih dokumentov:

- slike, kopirane iz grafičnega programa prek odložišča,
- slike, vstavljene kot hiperpovezave,
- polja s samodejnim posodabljanjem (številke strani, številke slik),
- tabele,
- standardne oblike pisav,
- šumniki,
- posebni znaki,
- enačbe,
- sprotne opombe,
- slogi besedil,
- povzetek dokumenta.

### Testni dokumenti:

- dokumenti, tvorjeni z Word 5.5 (datoteke: testni\_dokument\_1.doc, testni\_dokument\_2.doc in testni\_dokument\_3.doc),
- dokument, tvorjen z Word 2.0 (datoteka: testni\_dokument\_4.doc).

### Uporabljena programska in strojna oprema:

- Word 5.5 (angleška različica za DOS),
- Word 2.0 (angleška različica za Windows),
- Word 95 (angleška različica 7),
- Word 2007 SP2 (slovenska različica 12.0.6535.5002),
- Adobe reader 9, različica 9.3.1,
- program za pretvorbo dokumentov v format pdf: FreePDF XP 3.26,

- operacijski sistem: MS Windows Xp; Home Edition; različica 2002; servisni paket 3; slovenska različica,
- procesor: Intel Core™ 2; 1,66GHz, 1GB RAM.

### Merila za ovrednotenje uspešnosti rezultatov migracije:

- **popolnoma uspešna migracija:** vsi elementi in lastnosti v ciljnem formatu so popolnoma enaki kot v izvornem;
- **delno uspešna migracija:** določen element ali lastnost v ciljnem formatu nista popolnoma enaka kot v izvornem (npr. vrsta pisave), vendar s tem ni ogrožena njegova integriteta;
- **neuspešna migracija:** določen element ali lastnost v ciljnem formatu se tako razlikuje od izvornega, da je ogrožena njegova integriteta.

### Programi za preverjanje uspešnosti migracij

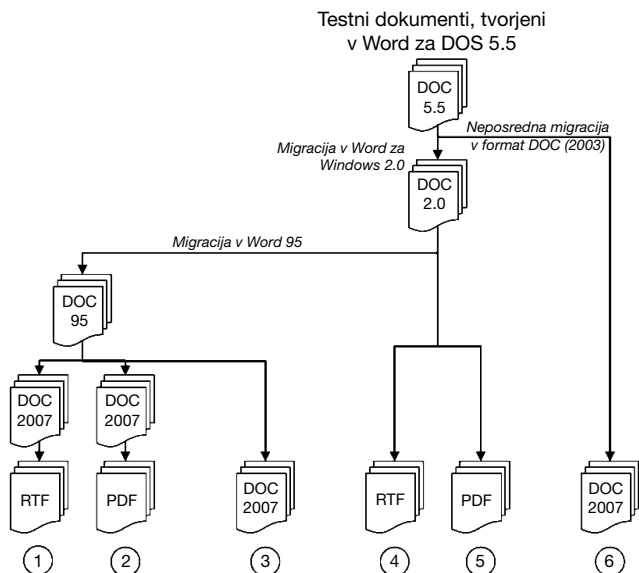
Uspešnost migracije smo spremljali na podlagi ohranjenih lastnosti v ciljnem formatu glede na izvorni dokument. Dokumente v novem formatu smo odpirali s programi:

- format DOC; Word 2007,
- format RTF; Word 2007,
- format PDF; Adobe reader 9.

## 4.1 Testiranje migracije dokumentov, tvorjenih z urejevalnikom Word 5.5

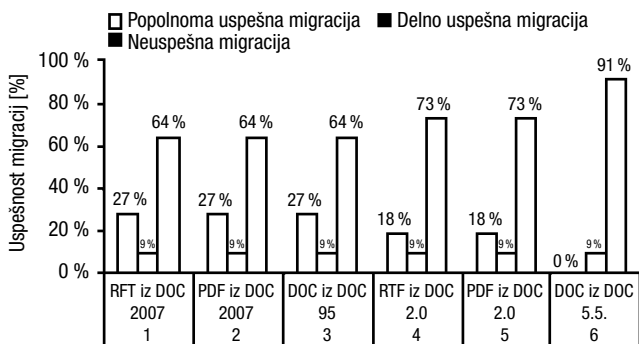
V prvem testu smo opazovali uspešnost neposredne in postopne migracije dokumentov, tvorjenih v urejevalniku Word 5.5. Dokumente smo v obeh primerih pretvorili v formate DOC (različica 2007), PDF (različica 1.3) in RTF. Glede na to, da urejevalniki v osnovi ne omogočajo pretvorbe v format PDF, je bil uporabljen brezplačni program FreePDF XP, ki deluje kot navidezni tiskalnik. Različica formata PDF po migraciji je 1.3. Neposredna migracija v formata RTF in PDF neposredno iz Word 5.5 z razpoložljivo programsko opremo ni bila mogoča, zato je bila izvedena postopna migracija najprej v format Word 2.0. Potek prvega testa prikazuje slika 1.

Urejevalnik Word 2007 ne omogoča neposrednega odpiranja datotek, ki so bile tvorjene v urejevalniku Word 2.0 brez poseganja v register Windows. Glede na to, da za povprečne uporabnike poseg v register ni priporočljiv, je bilo treba te dokumente predhodno pretvoriti še v format DOC MS Word 95.



Slika 1: **Potek testiranj migracije dokumentov, tvorjenih v Word 5.5**

Rezultati testiranja so pokazali, da je bila migracija slik v prvem testu neuspešna. Ohranil se je le del slik, kar bistveno vpliva na integriteto dokumentov. Polja s samodejnim posodabljanjem so se ohranila le pri postopni migraciji najprej v Word 2.0 in nato še v Word 95. V ostalih primerih je bila migracija neuspešna. V vseh primerih je bila neuspešna migracija tabel, šumnikov, posebnih znakov in povzetka dokumenta. Pri tem je treba upoštevati, da je bila tabela v testnem dokumentu sestavljena iz posebnih znakov in razmikov med vrednostmi s pomočjo tabulatorjev. Edina lastnost, ki je bila delno uspešna v vseh primerih, je ohranitev standardne oblike pisav brez upoštevanja šumnikov. V vseh primerih razen pri neposredni migraciji v format DOC 2007 je bila popolnoma uspešna migracija slogov besedil in sprotnih opomb.



Slika 2: **Uspešnost migracij dokumentov, tvorjenih v Word 5.5, v nekatere trenutno aktualne formate**

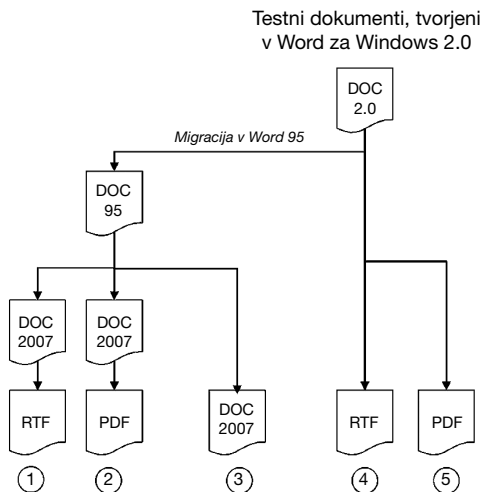
Kot prikazuje slika 2, delež neuspešnih migracij prvega testa v nobenem primeru ni manjši od 64 odstotkov.

Največ neuspešnih pretvorb posameznih lastnosti je bilo v primeru neposredne migracije izvornih dokumentov v Word 2007, in sicer 91 odstotkov. Popolnoma uspešna migracija je bila le pri 27 odstotkih obravnavanih lastnosti pri postopni migraciji originalnih dokumentov v treh korakih, tj. najprej v Word 2.0, nato v Word 95 in nazadnje v ciljni format. Pri postopni migraciji izvornih dokumentov v dveh korakih, tj. najprej v Word 2.0 in nato v končni format, je bil delež lastnosti, ki so prestali migracijo popolnoma uspešno, 18 odstotkov.

Zgornji rezultati kažejo, da v primerih, ko preteče daljše časovno obdobje med posameznimi različicami programske opreme, lahko pričakujemo večji delež neuspešnih migracij.

#### 4.2 Testiranje migracije dokumentov, tvorjenih z urejevalnikom Word 2.0

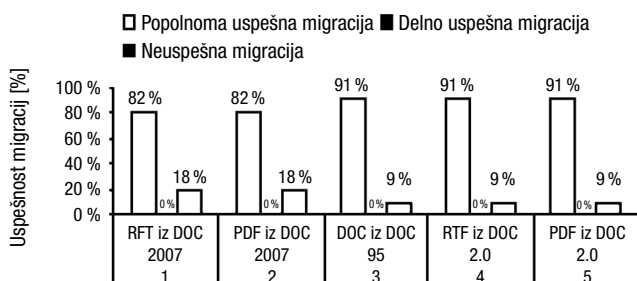
V drugem testu smo opazovali uspešnost neposredne migracije dokumentov, ki so bili tvorjeni v urejevalniku Word 2.0. Tudi v tem primeru so bili dokumenti pretvorjeni v iste formate kot v prvem testu. Potek drugega testa prikazuje slika 3.



Slika 3: **Potek testiranj migracije dokumentov, tvorjenih v Word 2.0**

Migracija v drugem testu je bila neuspešna le pri pretvorbi šumnikov, slogih besedila in povzetku dokumenta. V vseh drugih primerih je bila popolnoma uspešna.

Šumniki se niso ohranili v primeru migracije iz formata DOC Word 2.0 v Word 95 in nato v RTF, PDF ter format DOC 2007. Slogi besedil se niso ohranili pri migraciji testnega dokumenta iz formata DOC 2.0 v format RTF. Naslovi posameznih poglavij na isti strani so se pomaknili na začetek strani, kar je povzročilo prekrivanje besedila. S tem je ogrožena integriteta dokumenta. Migracija povzetkov dokumenta je bila neuspešna v primerih migracije v format PDF, saj se vpisane lastnosti ne ohranijo. S tem je ogrožena verodostojnost dokumenta.



Slika 4: Uspešnost migracij dokumenta, tvorjenega v Word 2.0, v nekatere trenutno aktualne formate

Rezultate uspešnosti migracije v različne formate v drugem testu prikazuje slika 4. Najnižji delež popolnoma uspešnih migracij obravnavanih lastnosti v tem testu je bil 82 odstotkov pri migraciji iz formata Word 2.0 v Word 95, nato v Word 2007 in nato v format RTF ali PDF. V drugih primerih je delež popolnoma uspešnih migracij 91 odstotkov.

Največjo težavo pri migraciji so predstavljali šumniki, in sicer v primerih, da je postopna migracija vključevala tudi pretvorbo v format Word 95. V teh treh primerih je bila migracija šumnikov neuspešna.

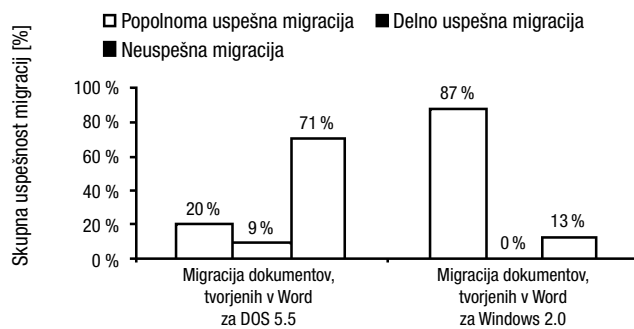
Rezultati drugega testa kažejo, da je neposredna migracija dokumenta Word 2.0 v formata RTF in PDF bolj uspešna kot v primeru postopne migracije.

Pri migraciji v format Word 2007 je bila potrebna postopna migracija v format Word 95, saj urejevalnik Word 2007 ne omogoča neposredne migracije dokumentov iz formata Word 2.0.

### 4.3 Primerjava uspešnosti migracij digitalnih dokumentov, tvorjenih v Word 5.5 in v Word 2.0

Rezultati uspešnosti migracij obeh testov, ki jih prikazuje slika 5, kažejo, da je bila migracija obravnavanih lastnosti bistveno bolj uspešna v primeru, ko je bil dokument tvorjen v urejevalniku Word 2.0, kot

v primeru dokumentov, tvorjenih v Word 5.5. Pri migraciji dokumentov, tvorjenih v Word 5.5, se je popolnoma ohranilo le povprečno 20 odstotkov obravnavanih lastnosti. Pri migraciji dokumenta, tvorjenega v Word 2.0, se je popolnoma ohranilo povprečno 87 odstotkov obravnavanih lastnosti.



Slika 5: Primerjava uspešnosti migracij testnih dokumentov, tvorjenih v Word 5.5, in testnega dokumenta, tvorjenega v Word 2.0

## 5 SKLEP

Rezultati prvega testa kažejo, da je za dokumente, tvorjene v formatu Word 5.5, najbolj učinkovita postopna migracija v format Word 2007.

V drugem testu je bil obravnavani dokument tvorjen v urejevalniku Word 2.0, ki deluje v operacijskem sistemu Windows. Najbolj uspešna je bila neposredna migracija v standardna formata RTF in PDF. Enak rezultat je pokazala posredna migracija v format DOC 2007. Najmanj učinkovita je bila posredna migracija v formata RTF in PDF.

Rezultati testov kažejo, da je uspešnost migracije dokumentov, tvorjenih v Word 5.5, slabša kot uspešnost migracije dokumentov, tvorjenih v Word 2.0.

## 6 LITERATURA IN VIRI

- [1] ICTU: From digital volatility to digital permanence, Preserving text documents (version 1.0); ICTU, Hag, december 2003; <http://www.digitaleduurzaamheid.nl> (dec. 2009).
- [2] ICTU: Migration: Context and Current Status, ICTU, Hag, december 2001; <http://www.digitaleduurzaamheid.nl/bibliotheek/docs/Migration.pdf> (dec. 2009).
- [3] Slovensko društvo Informatika, Slovar informatike (iskanje ustreznih prevodov); <http://www.islovar.org> (dec. 2009).
- [4] William E. Neale, IBM FileNet Records Manager: Assuring Records Integrity, IBM corporation, 2008; <ftp://ftp.software.ibm.com/software/data/ECM/compliance/fn-records-integrity-wp.pdf> (dec. 2009).
- [5] Microsoft Corporation, Rich Text Format (RTF) Specification, version 1.6, maj 1999; <http://msdn.microsoft.com/en-us/library/aa140277%28office.10%29.aspx> (dec. 2009).

- [6] World Wide Web Consortium, Extensible Markup Language (XML) Version 1.0, december 1997; [www.w3.org/TR/PR-xml-971208.rtf](http://www.w3.org/TR/PR-xml-971208.rtf) (dec. 2009).
- [7] Adobe Systems, <http://www.adobe.com/products/acrobat/adobepdf.html> (dec. 2009).
- [8] CPA/RLG, Preserving Digital Information: Report of the Task Force on Archiving of Digital Information, maj 1996; <http://www.clir.org/pubs/reports/pub63watersgarrett.pdf> (sept. 2010).
- [9] <http://www2.si.umich.edu/CAMILEON/reports/mor/index.html> (sept. 2010).
- [10] <http://www.interpares.org/> (sept. 2010).
- [11] Caroline van Wijk; Migration Research; Koninklijke Bibliotheek; april 2006; [http://www.kb.nl/hrd/dd/dd\\_projecten/Starting\\_Point\\_Migration\\_Research.pdf](http://www.kb.nl/hrd/dd/dd_projecten/Starting_Point_Migration_Research.pdf) (sept. 2010).
- [12] NLA, Digital Preservation: The Australian Experience; oktober 2000; <http://www.nla.gov.au/nla/staffpaper/dw001004.html> (sept. 2010).
- [13] <http://pandora.nla.gov.au/> (sept. 2010).
- [14] <http://en.archief.nl/> (sept. 2010).

■

Tomaž Stenšak je diplomiral leta 2002 na mariborski Fakulteti za elektrotehniko, računalništvo in informatiko na univerzitetnem študiju Gospodarsko inženirstvo, smer Močnostna elektrotehnika. V okviru magistrskega študija raziskuje področje vzdrževanja tehniške dokumentacije v elektroenergetskih podjetjih.

■

Tomaž Dogša je izredni profesor na Fakulteti za elektrotehniko, računalništvo in informatiko Univerze v Mariboru, kjer predava na dodiplomski in podiplomski stopnji in vodi Center za verifikacijo in validacijo sistemov. Na raziskovalnem področju se ukvarja predvsem s tehnologijo V&V oz. s testirnimi orodji.

# Slovensko društvo INFORMATIKA –

## Sekcija za operacijske raziskave in Fakulteta za informacijske študije Novo mesto

vas vabita na mednarodni simpozij iz operacijskih raziskav  
**The 11th International Symposium on Operations Research (SOR'11)**  
Dolenjske Toplice, 28.–30. septembra 2011

### Cilji srečanja

Področje operacijskih raziskav in njihovih aplikacij v ekonomijo, poslovne znanosti, organizacijo, proizvodnjo, ekologijo itn. se v svetu in pri nas zelo hitro razvija. Da bi sledili najnovejšim dognanjem na tem področju v svetu in doma, je treba najmanj vsako drugo leto izpeljati mednarodni simpozij s tega področja. Glavni namen simpozija je popularizacija operacijskih raziskav in stimulacija k novim raziskavam. Na mednarodnem simpoziju iz operacijskih raziskav The 11<sup>th</sup> International Symposium on Operations Research in Slovenia (SOR'11) pričakujemo izmenjavo izkušenj, pretok novih spoznanj in rešitev v mednarodnem in slovenskem okviru, identifikacijo praktičnih problemov ter operativni pristop k tržni ekonomiki.

Tridnevni program simpozija se bo odvijal v **tematskih sekcijah**:

- ▶ **Metodologija in tehnike operacijskih raziskav** (kombinatorična optimizacija, teorija odločanja, strateške igre, linearno programiranje, celoštevilsko programiranje, večkriterialno odločanje, mrežno planiranje in grafi, nelinearno programiranje, numerične metode, simulacija, statistika, stohastični procesi, vektorska optimizacija itn.)
- ▶ **Aplikacije operacijskih raziskav** v agronomiji, bančništvu, ekologiji, ekonomskih sistemih, energiji, varovanju okolja, financah, proizvodnji, zalogah, transportu itd.
- ▶ **Informatika in računalništvo v operacijskih raziskavah** (umetna inteligenca, sistemi za podporo odločanja, ekspertni sistemi, informacijski sistemi, računalniški programi s področja operacijskih raziskav itn.)

Vsak programski sklop bodo začeli  
**vabljeni predavatelji:**

**Prof. dr. Erling D. Andersen**

MOSEK ApS

Copenhagen, Danska

Naslov predavanja: *Ten years of experience with conic quadratic optimization*

**Prof. dr. Walter Gutjahr**

University of Vienna, Department of Statistics

and Decision Support Systems

Dunaj, Avstrija

Naslov predavanja: *Multi-Objective Combinatorial Optimization under Uncertainty*

**Prof. dr. Horst W. Hamacher**

University of Kaiserslautern, Department  
of Mathematics

Kaiserslautern, Nemčija

Naslov predavanja: *Operations research methods in the planning, control, and adaptation of evacuation plans*

**Prof. dr. Arie M. C. A. Koster**

Lehrstuhl II für Mathematik, RWTH Aachen

Aachen, Nemčija

Naslov predavanja: *Robust Optimization of Telecommunication Networks*

**Prof. dr. Zrinka Lukač**

University of Zagreb, Faculty of Economics &  
Business

Zagreb, Hrvaška

Naslov predavanja: *Metaheuristic Optimization*

**Prof. dr. Nenad Mladenović**

School of Information systems and Mathematics,  
Brunel University

Uxbridge, West London, Velika Britanija

Naslov predavanja: *Variable neighbourhood search metaheuristic: recent applications*

**Prof. dr. Ulrich Pferschy**

University of Graz, Department of Statistics and  
Operations Research

Graz, Avstrija

Naslov predavanja: *Mathematical Models and Solutions for Network Design Problems*

Kot običajno bo tudi letos izšel **zbornik recenziranih prispevkov**, ki je citiran v številnih mednarodnih bazah. Po končanem simpoziju pa bodo organizatorji simpozija izdali še posebno številka revije **CEJOR** (revija SCI), v kateri bodo objavljeni izbrani in razširjeni prispevki iz SOR'11, ki bodo ustrezali recenzentskim merilom revije.

### Vabilo udeležencem

Na konferenco vabimo vse, ki pri svojem delu razvijajo ali uporabljajo operacijske raziskave. Udeležence, ki želijo na simpoziju predstaviti svoje prispevke, prosimo, naj prispevek pošljejo organizatorju skladno z vabilom. Vabilo in druge izčrpne informacije o simpoziju dobite na naslovu: <http://sor11.fis.unm.si>. Dodatne informacije pa lahko dobite tudi na naslovu: [sor11@fis.unm.si](mailto:sor11@fis.unm.si).

## Deset let terminološkega slovarja informatike

Slovensko društvo INFORMATIKA je že ob odločitvi za izdajanje strokovne revije *Uporabna informatika* zavzelo stališče, da mora kot stanovska organizacija informatikov skrbeti za strokovni jezik informatike. Posledica je bila ustanovitev sekcije za jezik, ki si je za prvo in glavno nalogo zadala ustvariti slovenski terminološki slovar izrazja informatike. Strokovne odličnosti namreč ni brez odličnega strokovnega jezika. Letos poteka deseto leto od prve javne predstavitve Islovarja.

Aprila 2001 smo na posvetovanju Dnevi slovenske informatike predstavili spletni terminološki slovar informatike, ki ga je pripravila jezikovna sekcija Slovenskega društva INFORMATIKA. Čas za objavo takega slovarja je bil ugoden: uporaba interneta je bila že uveljavljena, objavljenih je bilo že nekaj bolj ali manj obširnih slovarjev računalništva, društvo je že več let izdajalo strokovno revijo *Uporabna informatika*.

Na povabilo članom društva, naj se pridružijo sekciji, se je prijavilo več zainteresiranih – članov društva in drugih –, od katerih so nekateri v sekciji aktivni še danes. Članstvo v sekciji se je potem spreminjalo, kakor se to pač dogaja povsod: eni so prihajali, drugi so se umikali. Zdaj je v skupini trideset urednikov. To so ljudje različnih profilov, večina so visokošolski učitelji, nekaj je prevajalcev, nekaj ljudi iz prakse.

Spletna rešitev ponuja urednikom številne možnosti: iskanje po slovenskem ali angleškem izrazu, po raznih kriterijih, vpogled v zgodovino dogajanja, komentarje, razprave, povezave na koristne naslove in korpus izrazja informatike, zvočni zapis izraza, takojšnje spremembe in različne možnosti urejanja (dodajanje, dopolnjevanje,

spreminjanje) slovarskih sestavkov. Poleg tega omogoča aktivno sodelovanje tudi uporabnikom Islovarja, kar sicer ni ravno pogosta praksa.

Pomembni mejniki razvoja so:

- ustanovitev jezikovne sekcije Slovenskega društva INFORMATIKA junija 2000,
- odločitev sekcije o izdaji terminološkega slovarja na sestanku decembra 2000,
- predstavitev terminološkega slovarja na posvetovanju DSI aprila 2001,
- opredelitev strategije – julija 2003,
- začetek dela slovaropisne skupine oktobra 2003,
- z glasovanjem urednikov dogovorjeno ime Islovar – februarja 2004,
- izdaja poskusnega snopiča – aprila 2004,
- nova izdaja Islovarja – novembra 2004.

Islovar je prosto dostopen na naslovu <http://www.islovar.org>. Na dan 1. marca 2011 je bilo v Islovarju 5.685 izrazov, od tega je bilo 4.795 opremljenih z razlago, registriranih je bilo 1.374 uporabnikov. Islovar se veliko uporablja (ca. 20.000 iskanj v enem mesecu) in tudi citira.

Če ocenimo naše preteklo delo, smo z razmeroma skromnimi sredstvi napravili zelo veliko. V Sloveniji ni podobnega terminološkega slovarja, pa tudi v tujini bi težko našli enak dosežek. S primerno podporo in znanjem pa bi verjetno to, kar imamo danes, naredili veliko hitreje. V prvih letih smo morali prestati naporno učenje ljudi, ki so se lotili resnega dela brez pravega znanja. Edino, kar smo imeli, je bilo prepričanje, da delamo nekaj koristnega. Niti najmanj si nismo predstavljali, kako



obsežno in dolgotrajno bo naše delo. Naš namen je bil predvsem izdaja slovarja v papirni obliki, pa tudi odprtost, spletna dostopnost ter možnosti iskanja po slovenskih iztočnicah ali angleških ustreznica, opremljenih s slovenskimi razlagami.

Danes vidimo zadevo drugače: to je dolgoročen projekt, ki bo predvidoma trajal še več let. Vedno znova srečujemo nove pojme, ki se uveljavljajo v informacijski tehnologiji. Odpirajo se nova področja v informatiki. Menimo, da bi Islovar moral ostati ažuren, zato je precejšnja verjetnost, da bo ostal kar na spletu. Pravzaprav smo v preteklih letih izoblikovali model razvoja terminološkega slovarja, ki bi bil lahko koristen tudi za druga področja.

Ves čas nas je v okviru možnosti podpiralo Slovensko društvo INFORMATIKA, Ekonomska fakulteta Univerze v Ljubljani nas gosti na strežniku. Do sedaj nam je Ministrstvo za kulturo odobrilo

sofinanciranje že nekaj manjših projektov razvijanja Islovarja, vendar je bila večina dela opravljena prostovoljno. Prav prispevek naših urednikov, ki so redno zaposleni drugje in imajo veliko drugih obveznosti, nas potrjuje v prepričanju, da delamo nekaj, kar je koristno za vse. Lepo javno priznanje pomena Islovarja je določilo, ki smo ga zasledili že v več javnih razpisih s področja informatike in ki bi ga lahko povzeli takole: »Strokovni izrazi v tem razpisu imajo tak pomen, kot je definiran v Islovarju.«

Ob deseti obletnici Islovarja se zahvaljujemo vsem, ki so kakor koli prispevali k njegovemu razvoju – predsednikom sekcije za jezik, urednikom in vsem drugim, ki so ta pomembni projekt podprli s prispevki v denarju ali z delom.

*Katarina Puc  
Niko Schlamberger*

## Iz Islovarja

Islovar je spletni terminološki slovar informatike, ki je prosto dostopen na naslovu <http://www.islovar.org>. Izraze lahko komentirate, tako da se prijavite v poglavju Nov uporabnik, poiščete izraz, ki ga želite komentirati, in zapišete svoj komentar ter predlog spremembe. V tej številki revije objavljamo pomensko zbirko, ki smo jo sestavili na temelju izraza podatek.

**dekodiranje** -a s (angl. *decoding*)

razbiranje sporočila iz zaporedja znakov, upoštevajoč dogovorjeno kodo (1); prim. kodiranje

**dekodirati** -am *dov., nedov.* (angl. *decode*)

razbirati sporočila iz zaporedja znakov, upoštevajoč dogovorjeno kodo (1); prim. kodirati

**dekompresija podatkov** -e -- ž (angl. *data decompression*)

gl. raztezanje podatkov

**dekomprimiranje datoteke** -a -- s (angl. *file decompression*)

gl. raztezanje datoteke

**dekomprimiranje podatkov** -a -- s (angl. *data decompression*)

gl. raztezanje podatkov

**DTE** DTE-ja [dətəé, -êja] m *krat.* (angl. *data terminal equipment*)

gl. terminalna naprava

**EDI** EDI-ja [édi] m *krat.* (angl. *electronic data interchange*)

gl. računalniško izmenjavanje podatkov

**izgúbno stískanje** -ega -a s (angl. *lossy compression*)

stiskanje, pri katerem se del podatkov namerno zavrže

**izvajalna podatkovna báza** -e -e -e ž (angl. *operational data store, krat. ODS*)

pomožna podatkovna baza, v kateri se podatki iz različnih virov obdelujejo in pripravijo za uporabo

**izvòz podátkov** -ôza -- m (angl. *data export*)

prenos podatkov(2) v zapisu<sup>1</sup>, tako da jih lahko prebere drug program; prim. uvoz podatkov

**knjížnica podátkov** -e -- ž (angl. *library of data*)

gl. podatkovna knjižnica

**kóda** -e ž (angl. *code*)

1. pravila za oblikovanje zapisa<sup>4</sup>; sin. kod; prim. šifra<sup>1</sup> (1), izvorna koda, psevdokoda, strojna koda
2. gl. geslo

**kodirani** -a -o prid. (angl. *encoded*)

ki je spremenjen z uporabo kode (1)

**kodiranje** -a s (angl. *encoding*)

pretvarjanje sporočila v zaporedje znakov, upoštevajoč dogovorjeno kodo (1); sin. zakodiranje; prim. dekodiranje

**kodirati** -am *nedov.* (angl. *encode*)

pretvarjati sporočila v zaporedje znakov, upoštevajoč dogovorjeno kodo (1); sin. zakodirati; prim. dekodirati

**kompresija datoteke** -e -- ž (angl. *file compression*)

gl. stiskanje datoteke

**kompresija podatkov** -e -e ž (angl. *data compression*)

gl. stiskanje podatkov in komprimiranje podatkov

**komprimiranje datoteke** -a -- s (angl. *file compression*)

gl. stiskanje datoteke

**komprimiranje podatkov** -a -- s (*angl. data compression*)

vsak od postopkov za prekodiranje podatkov (2), ki zmanjšuje redundanco in s tem obseg podatkov (2); sin. stiskanje podatkov, kompresija podatkov; prim. raztezanje podatkov

**nabòr podatkov** -ôra -- m (*angl. dataset, data set*)  
skupek podatkov z določeno vsebino

**naprava za shranjevanje podatkov** -e -- -- -- ž (*angl. storage device*)  
gl. pomnilnik (1) in pomnilniška naprava

**ODS ODS-ja** [odəsə] m *krat.* (*angl. operational data store*)  
gl. izvajalna podatkovna baza

**odzípati** -am *dov.* (*angl. unzip*) *žarg.*  
raztegniti datoteko, datoteke, stisnjene v datoteke s končnico zip; prim. zazipati

**ohránjanje podatkov** -e -- s (*angl. data retention*)  
trajna hramba podatkov (2), navadno določena s predpisi

**osébní podatki** -ih -ov m (*angl. personal data*)  
podatki, ki se nanašajo na posameznika

**podátek** -tka m (*angl. data*)  
1. znano dejstvo o določeni stvari, predmetu, pojavu, ki je temelj za sklepanje  
2. to dejstvo v taki obliki, da se lahko obdeluje z računalnikom

**podátkovna knjižnica** -e -e ž (*angl. data library*)  
tematska zbirka nizov podatkov z orodji za vpogled, namenjena raziskovalnemu delu; sin. knjižnica podatkov

**podátkovna shrámba** -e -e ž (*angl. data storage, data store*)  
kar se uporablja za shranjevanje podatkov

**podátkovno skladišče** -ega -a s (*angl. data warehouse*)  
podatkovna baza, v kateri se zbirajo podatki o poslovanju organizacije, namenjeni analizi in odločanju; prim. sprotna analitična obdelava podatkov

**podróčno podátkovno skladišče** -s (*angl. data mart*)  
podatkovno skladišče, prirejeno za posamezno poslovno področje

**prekodírati** -am *dov.* (*angl. recode*)  
ponovno kodirati že kodirano sporočilo z drugo kodo (1)

**pridobívanje podatkov** -a -- s (*angl. data acquisition*)  
avtomatizirano zbiranje podatkov in prenos podatkov v računalniški sistem; prim. zajemanje podatkov, zbiranje podatkov

**računálniško izmenjávanje podatkov** -ega -a -- s, *krat.* RIP (*angl. electronic data interchange, krat. EDI*)  
prenašanje podatkov med informacijskimi sistemi organizacij brez posredovanja človeka

**razšírjanje podatkov** -a -- s (*angl. 1. data distribution, 2. data decompression*)  
1. razpošiljanje podatkov na več naslovov  
2. gl. raztezanje podatkov

**raztégni podátke** -em -- *dov.* (*angl. data decompress*)  
prekodirati prvotno stisnjene podatke (2) v njihovo izvorno obliko; prim. stisniti podatke

**raztézanje datotéke** -a -- s (*angl. file decompression*)  
raztezanje podatkov (2) v datoteki; sin. dekomprimiranje datoteke; prim. stiskanje datoteke

**raztézanje** -a s (*angl. data decompression*)  
vsak od postopkov za prekodiranje prvotno stisnjenih podatkov (2), ki obnovi njihovo izvorno obliko; sin. dekomprimiranje podatkov, razširjanje podatkov (2); prim. stiskanje podatkov, komprimiranje podatkov

**redundánca** -e ž (*angl. redundancy*)  
lastnost znaka, sistema znakov, da prenaša določeno obvestilo z več znaki, prvinami, kot je nujno potrebno

**RIP RIP-a** [ríp] m *krat.* (*angl. electronic data interchange, krat. EDI*)  
gl. računalniško izmenjavanje podatkov

**SCADA** SCADA [škáda] m *krat.* (*angl. supervisory, control and data acquisition*)  
sistem, namenjen nadzorovanju in krmiljenju tehnološkega procesa z računalnikom

**splétno podatkovno skladišče** -ega -ega -a s (*angl. data webhouse*)  
podatkovno skladišče, v katerem se zbirajo predvsem podatki o uporabi spletnih strani

**stískanje datotéke** -a -- s (*angl. file compression*)  
stiskanje podatkov (2) v datotekah; sin. kompresija datoteke, komprimiranje datoteke; prim. raztezanje datoteke

**stískanje podátkov** -a -- s (*angl. data compression*)  
vsak od postopkov za prekodiranje podatkov (2), ki zmanjšuje redundanco in s tem obseg podatkov (2); sin. komprimiranje podatkov, kompresija podatkov; prim. raztezanje podatkov

**stísiniti podátke** -em *dov.* (*angl. compress data*)  
prekodirati podatke (2) s postopkom za zmanjšanje redundance in s tem obsega podatkov (2); prim. raztegniti podatke

**stísjnjeni** -a -o *prid.* (*angl. compressed, zipped*)  
ki se nanaša na tako prekodiranje podatkov (2), ki zmanjšuje redundanco in s tem obseg podatkov (2), npr. stisnjeni podatki, stisnjena datoteka

**podátkovni terminál** -ega -a ž (*angl. data terminal equipment, krat. DTE*)  
vhodna in/ali izhodna naprava za vnos, izpisovanje ali prikazovanje podatkov, ki je praviloma oddaljeno povezana z računalnikom

**uvòz podátkov** -óza m (*angl. data import*)  
prevzem podatkov (2), ki so bili zapisani z drugim programom; prim. izvoz podatkov

**zajémanje podátkov** -a -- s (*angl. data capture*)  
zbiranje podatkov in prenos podatkov v računalniški sistem; prim. pridobivanje podatkov, zbiranje podatkov

**zakodíran** -a -o *prid.* (*angl. encoded*)  
gl. kodirani

**zakodíranje** -a s (*angl. encoding*)  
gl. kodiranje

**zakodírati** -am *dov.* (*angl. encode*)  
gl. kodirati

**zapís<sup>4</sup>** -a m (*angl. record*)  
predstavitev podatka (1) v zaznavni obliki

**zazípati** -am *dov.* (*angl. zip*) *žarg.*  
stisniti datoteko, datoteke v datoteko s končnico zip; prim. odzipati

**zbíranje podátkov** -a-- s (*angl. data collection*)  
evidentiranje podatkov iz virov informacij; prim. pridobivanje podatkov, zajemanje podatkov

**zíp** -a m (*angl. zip*)  
zapis<sup>1</sup> stisnjene arhivske datoteke

**zípati** -am *nedov.* (*angl. zip*) *žarg.*  
stiskati datoteko, datoteke v eno datoteko s končnico zip

## Koledar prireditev

ID @ Munich Training Session, ID Management State-of-the-Art	5.-7. april 2011	München, Nemčija	<a href="http://www.smart-university.eu/program-id-management-state-of-the-art.htm">http://www.smart-university.eu/program-id-management-state-of-the-art.htm</a>
18. konferenca Dnevi slovenske informatike	18.-20. april 2011	Portorož, Slovenija	<a href="http://www.dsi2011.si">www.dsi2011.si</a>
9th Eastern European eGov Days: eGovernment in Times of Economic Callanges	(8.) 9.-11. maj 2011	Ljubljana, Slovenija	<a href="http://www.ocg.at">www.ocg.at</a> ; <a href="http://www.cegd.eu">www.cegd.eu</a>
11th International multidisciplinary scientific Geo-Conference & Expo SGEM 2011	19.-25. junij 2011	Albena, Bolgarija	<a href="http://www.sgem.org">www.sgem.org</a>
10th International Symposium Economy & Business	3.-7. september 2011	Sunny Beach resort, Bolgarija	<a href="http://sciencebg.net/economy-and-business.php">http://sciencebg.net/economy-and-business.php</a>
NFC World Congress	19.-21. september 2011	Nica, Francija	<a href="http://www.nfcworldcongress.com">http://www.nfcworldcongress.com</a>
Smart Event 2011 – The future of Digital Security Technologies – conferences: e-Smart, Smart Mobility, World e-ID	21.-23. september 2011	Sophia-Antipolis, Francija	<a href="http://www.smart-event.eu">http://www.smart-event.eu</a>
The 11th International Symposium on Operations Research (SOR'11)	28.-30. september 2011	Dolenjske Toplice, Slovenija	<a href="http://sor11.fis.unm.si">http://sor11.fis.unm.si</a>
MPP 2011 – Konferenca Management poslovnih procesov 2011	19.-20. oktober 2011	Ljubljana, Slovenija	<a href="http://www.process-conference.org">http://www.process-conference.org</a>
First IFIP CIO Forum	1.-4. november 2011	Shenzhen, Kitajska	<a href="http://www.worldcioforum.com">www.worldcioforum.com</a>

## Pomembni spletni naslovi

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

# 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.
<b>Domači naslov</b>	
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
- upravna 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

# Izpitni centri ECDL



## Znanstveni prispevki

Marcel Križevnik, Matjaž B. Jurič

**Upravljanje matičnih podatkov kot osnova pri vpeljavi storitveno usmerjene arhitekture**

Franc Brcar

**Izzivi zunanjega izvajanja informatike**

Simon Vrhovec, Rok Rupnik

**Obvladovanje odpora pri projektih informacijskih tehnologij**

## Strokovni prispevki

Tomaž Dogša, Mirjam Bonačič

**Orodja za podporo testiranja parov**

Tomaž Stenšak, Tomaž Dogša

**Migracija wordovih dokumentov**

## Informacije

Katarina Puc, Niko Schlamberger

**Deset let terminološkega slovarja informatike**

**Iz slovarja**

**Koledar prireditev**

ISSN 1318-1882

