

UNIVERZA V LJUBLJANI
EKONOMSKA FAKULTETA

DIPLOMSKO DELO
POSLOVNI VIDIK ODPRTE KODE

Ljubljana, junij 2010

MATEVŽ ŠTRUKELJ

IZJAVA

Študent Matevž Štrukelj izjavljam, da sem avtor tega diplomskega dela, ki sem ga napisal pod mentorstvom dr. Tomaža Turka, in da dovolim njegovo objavo na fakultetnih spletnih straneh.

V Ljubljani, dne _____

Podpis: _____

KAZALO

Uvod.....	1
1 Odprta koda.....	2
1.1 Kriteriji odprte kode	2
1.2 Razlika med odprtokodno in lastniško programsko opremo	3
1.2.1 Stroški.....	4
1.2.2 Varnost	5
1.2.3 Fleksibilnost	5
2 Evropski trg programske opreme.....	5
2.1 Obstoječe stanje na evropskem trgu	5
2.2 Poslovni in tehnološki trendi	7
2.2.1 Spremembe povpraševanja.....	7
2.2.2 Razvoj tehnologije.....	8
2.2.3 Nove tržne strukture	10
2.3 Izzivi evropskega trga programske opreme.....	10
2.3.1 Prepoznavanje človeškega talenta	11
2.3.2 Spodbujanje inovativnosti	11
3 Poslovni model odprte kode	11
3.1 Trženje lastniške in odprtokodne programske opreme.....	12
3.2 Najbolj razširjeni poslovni modeli	13
3.2.1 Zagotavljanje storitev in podpore.....	13
3.2.2 Programska oprema kot storitev	13
3.2.3 Razvoj lastniške programske opreme na podlagi odprtokodne.....	14
4 Poslovne strategije odprte kode	14
4.1 Optimizacijska strategija	15
4.2 Strategija dvojnega licenciranja.....	16
4.3 Naročniška strategija	16

4.4	Svetovalna strategija	16
4.5	Pokroviteljska strategija.....	17
4.6	Gostiteljska strategija.....	17
4.7	Strategija vgrajene programske opreme.....	18
5	Odprta koda – vir konkurenčne prednosti.....	18
5.1	Razlogi za uporabo odprte kode.....	19
5.1.1	Odprta koda pri razvoju lastniške programske opreme	19
5.1.2	Prednosti odprtokodne programske opreme v primerjavi z lastniško.....	19
5.2	Izzivi in priložnosti odprtokodne programske opreme	20
	Sklep	22
	Literatura in viri	24
	KAZALO SLIK	
	Slika 1: Storitveno usmerjena arhitektura	8
	Slika 2: Razpršeni računalniški sistem	9
	Slika 3: Osnovni elementi semantičnega spleta.....	10
	Slika 4: Usmerjenost programske opreme	12
	Slika 5: Hibridni model razvoja programskih rešitev	14
	KAZALO TABEL	
	Tabela 1: Primerjava odprtokodne in lastniške programske opreme	4
	Tabela 2: Največji svetovni proizvajalci programske opreme v letu 2009.....	6

Uvod

Podjetja se ob vstopu na trg srečujejo z vse večjo konkurenco. Konkurenca se povečuje z razvojem gospodarstva in podjetja sili v sprejemanje racionalnih odločitev ter smotrno rabo razpoložljivih sredstev. Hiter razvoj gospodarstva poskrbi, da trgi hitreje prehajajo v zrelo fazo. Na zrelih trgih so se podjetja primorana boriti za tržne deleže, posledično je veliko sredstev potrebno nameniti za razvoj, ki predstavlja gonilno silo inovacij. Treba je omeniti, da danes ni več pomembna le kakovost izdelkov in storitev, podjetja morajo slediti smernicam, ki jih narekuje povpraševanje in v čim večji meri zadovoljiti potrebe kupcev. Eden izmed ključnih dejavnikov je zmožnost prilagajanja razmeram na trgu. Od izdelka si lahko podjetje obeta zaslužek le v primeru, ko pravi izdelek ob pravem času ponudi pravim kupcem.

Globalizacija je še zaostila razmere na trgu. Konkurence nam danes ne predstavljajo le podjetja, ki so v naši geografski bližini, ampak podjetja iz celega sveta, ki se ukvarjajo s podobnimi dejavnostmi. Globalizacijo je omogočil razvoj logistike in transporta, saj blago lahko v enem dnevu dostavimo tudi na najbolj geografsko oddaljene trge.

Trg programske opreme se nahaja v zreli fazi. Podjetja, ki želijo doseči konkurenčno prednost morajo neprestano slediti spremembam povpraševanja in iskati odgovor na vprašanje, na kakšen način kupcu zagotoviti dodano vrednost in ga s tem prepričati v nakup njihove programske opreme oz. uporabe storitev. Za storitvena podjetja je bistvenega pomena, da svojim kupcem zagotovijo kakovostno storitev ob čim nižjih stroških. Pri optimizaciji poslovanja in nižanju stroškov je lahko eden izmed ključnih dejavnikov uporaba odprtokodne programske opreme. Odprta koda koristi tudi proizvajalcem programske opreme, ki lahko z njeno uporabo dosežejo konkurenčno prednost in si povečajo prihodke. Potrebna je primerna izbira strategije in poslovnega modela.

Namen diplomskega dela je raziskati, katere poslovne modele in strategije imajo na voljo podjetja, ki se odločijo za uporabo odprte kode. Cilj diplomskega dela je ugotoviti, ali lahko uporaba odprte kode podjetjem prinese konkurenčno prednost ter predstaviti različne poslovne modele in strategije, ki to omogočajo. Pri raziskavi problemske tematike je uporabljena predvsem tuja strokovna literatura, publikacije in članki, ki zaradi hitrega razvoja informacijske tehnologije in veliko novosti niso starejši od nekaj let.

Diplomsko delo sestavljajo uvod, pet poglavij ter sklep. Začeli bomo s predstavitvijo odprtokodne programske opreme, njenih kriterijev in jo v ključnih točkah primerjali z lastniško programsko opremo. V drugem poglavju se bomo osredotočili na evropski trg programske opreme, preučili bomo trenutno stanje, poslovne in tehnološke trende ter izzive. Sledili bodo najbolj razširjeni poslovni modeli in strategije podjetij, ki pri svojem poslovanju uporabljajo odprtokodno programsko opremo. V petem poglavju bomo navedli razloge za uporabo odprte kode in predstavil izzive in priložnosti, s katerimi se bodo soočila podjetja, ki se ukvarjajo z odprto kodo. V sklepu izpostavimo najpomembnejše ugotovitve in na ta način obravnavano tematiko združimo v celoto.

1 Odprta koda

Termin odprta kodna se navezuje na programsko opremo, ki omogoča prost dostop do izvorne kode. Licence odprtokodnih rešitev dajejo uporabniku možnost uporabe, preučevanja, predelave in proste redistribucije originalne ali predelane verzije. Pri programiranju se velikokrat uporabljajo iste metodologije in prakse kot pri izdelavi lastniške programske opreme (Coppola & Neelley, 2004, str. 2).

Licenčni model odprte kode večkrat vsebuje določene omejitve. Aktivnosti, kot so na primer: podpora, trening in posodobitve lahko izvajajo samo določene skupine ljudi. Razmerja so pogodbeno urejena. V večini primerov je izdelek, ki temelji na odprti kodi, rezultat skupinskega dela. Več posameznikov ali skupin prispeva določen del končnega izdelka (Hiong, 2005, str. 1).

Na odprto kodo se pogosto gleda kot na brezplačno programsko opremo. Enačenje je zmotno. Odprtokodne programske rešitve omogočajo dostop do izvorne kode in njeno spreminjanje, ne pomenijo pa njene brezplačne uporabe. Prav tako ni nujno, da je brezplačna programska oprema zasnovana po načelih odprte kode.

1.1 Kriteriji odprte kode

Pogoj za odprtokodno programsko rešitev ni le dostop do izvorne kode. Programska oprema mora izpolnjevati več kriterijev. Slednje določa Open Source Initiative (v nadaljevanju OSI), ki je neprofitna organizacija. Ukvarja se s promocijo odprte kode in skrbi za njeno licenciranje. OSI poudarja tehnične pomanjkljivosti lastniške programske opreme v primerjavi z odprtokodno.

OSI na podlagi definicije odprte kode izvaja certificiranje licenc odprte kode in s tem zagotavlja osnovo za pravilno uporabo pojma odprta koda. OSI definicija odprte kode zajema veliko manjše število programske opreme. Oznako OSI odprta koda si namreč zagotovi le programska oprema, ki izpolnjuje naslednjih 10 kriterijev (Open Source Initiative):

1. Svobodna distribucija

Licenca ne sme omejevati proste distribucije celotne programske rešitve ali njenih delov združenih v paket. Za program se ne sme zahtevati nikakršna vrsta plačila.

2. Izvorna koda

Program mora vsebovati izvorno kodo, v primeru da je ne, mora biti prosto dostopna. Ni dovoljeno njeno zakrivanje ali kakršnokoli oteževanje dostopa do nje. Oblika izvorne kode mora omogočati spreminjanje.

3. Izpeljana dela

Licenca mora dovoljevati spreminjanje in izdelavo izpeljanih programskih rešitev, le-te morajo biti distribuirane pod enakimi licenčnimi pogoji, kot veljajo za originalno programsko rešitev.

4. Integriteta avtorja izvirne kode

Licenca lahko zahteva, da se izpeljani izdelki distribuirajo pod spremenjenim nazivom ali spremenjeno oznako verzije.

5. Prepoved diskriminacije posameznikov in skupin

Licenca ne sme diskriminirati osebe ali skupine pri uporabi programa.

6. Prepoved diskriminacije posameznih področij dejavnosti

Licenca ne sme omejevati uporabe programa na posameznem področju dejavnosti. Program je lahko uporabljen tako za osebno kot poslovno rabo.

7. Distribucija licence

Licenčne pravice se nanašajo na vsakogar, ki prejme program brez kakršnih koli postopkov.

8. Licenca ne sme biti specifična za produkt

Pravice, vezane na program, ne smejo biti odvisne od programske distribucije. V primeru odstranitve programa iz distribucije in uporabi ali distribuciji naprej pod pogoji licence programa, morajo vse stranke, ki so dobile določen program imeti enake pravice kot tiste, ki so jim bile dodeljene pravice za originalni program.

9. Licenca ne sme omejevati druge programske opreme

Licenca ne sme zahtevati, da morajo biti vsi programi na določenem mediju odprtokodni.

10. Licenca mora biti nevtralna do tehnologije

Licenca ne sme biti omejena na posamezno tehnološko rešitev.

1.2 Razlika med odprtokodno in lastniško programsko opremo

Poglavitna razlika med odprtokodno in lastniško programsko opremo je v nadzoru. Odprtokodne licence dovoljujejo uporabo, predelavo in redistribucijo programske opreme, lastniške licence jih omejujejo. Poglavitni razlog je v znanju, saj se v izvorni kodi vsake programske rešitve skriva veliko znanja ljudi, ki so produkt razvijali. Znanje za podjetja predstavlja investicijo, moč in vir konkurenčne prednosti.

V spodnji tabeli so prikazane razlike med odprtokodno in lastniško programsko opremo. Primerjava je narejena na podlagi ključnih atributov, ki bi morali biti upoštevani pri izbiri in nakupni odločitvi programske opreme. Primerjava zajema tudi celotne stroške lastništva, na podlagi katerih lahko ocenimo stroške, ki bodo nastali v povezavi z nakupom in uporabo programske opreme.

Tabela 1: Primerjava odprtokodne in lastniške programske opreme

Atribut	Odprtokodna programska oprema	Lastniška programska oprema
Licence	Licencirana za zagotavljanje svobodne uporabe, spreminjanja in redistribucije.	Licencirana zaradi zaščite pred spreminjanjem in redistribucijo.
Nadzor	Uporabnik in proizvajalec si delita nadzor.	Proizvajalec ima popoln nadzor nad programsko opremo.
Inovacije	Hitre in raznolike zaradi velikega števila ljudi, ki sodelujejo pri razvoju.	Omejene na višino investicij namenjenih za razvoj.
Tveganje opustitve	Programska oprema je na voljo, dokler služi svojemu namenu.	Odvisno od proizvajalca in njegove uspešnosti.
Celotni stroški lastništva		
Stroški pridobitve	Nizki ali brez stroškov licenc.	Stroški licenc.
Stroški implementacije in vzdrževanja	V večini primerov nizki.	Odvisno od proizvajalca in pogodbenega razmerja.
Stroški obsega	Povečanja števila uporabnikov ne poveča stroška licenc.	Stroški licence se v večini primerov večajo s številom uporabnikov.

Vir: Coppola & Neelley, *Open source – opens learning: Why open source makes sense for education*, 2004, str. 4.

V nadaljevanju podrobneje obravnavamo tri ključne lastnosti, ki jim podjetja pri nakupu oz. uporabi programske opreme namenijo največ pozornosti. Stroški so pomembni zaradi optimizacije poslovanja, varnost zaradi varnosti podatkov in možnosti njihove zlorabe. Fleksibilnost je potrebno upoštevati, ker imajo podjetja edinstvene informacijske potrebe, ki jih želijo zadovoljiti v čim večji meri.

1.2.1 Stroški

Pri izbiri programske opreme moramo biti pozorni na vse stroške, ki nastanejo. Stroška ne predstavljajo le začetna investicija. Fazi nakupa sledijo vpeljava, vzdrževanje in podpora.

Zagovorniki odprte kode trdijo, da je lastniška programska oprema dražja, kar ne velja v vseh primerih. Odgovorni za sprejemanje nakupnih odločitev morajo upoštevati celoten nabor stroškov, ki nastane pri odločitvi za določeno vrsto programske opreme. Priporočljiva je analiza različnih nakupnih možnosti, ki mora zajeti celotne stroške lastništva.

1.2.2 Varnost

Varnost predstavlja zelo občutljivo tematiko na vseh področjih, ne le pri razvoju programske opreme. Podjetja jo v večini primerov postavljajo na prvo mesto. Najbolj občutljiva so podjetja, ki imajo opravka z osebnimi podatki naročnikov in dokumenti, zaupne narave. V omenjenih primerih so stroški lastništva in nakupa programske opreme podrejeni varnosti. Izguba ali kraja podatkov za podjetja predstavlja veliko poslovno škodo in izgubo ugleda.

Trenutno poteka razprava med zagovorniki lastniške in odprtokodne programske opreme, katera programska oprema je varnejša. Splošno težko vrednotimo katera izbira je boljša, vse je odvisno od posameznega izdelka. Izpostaviti je potrebno, da je varnost v prvi vrsti odvisna od načina razvoja programske opreme, njenih nastavitvev in vzdrževanja. Slabo vzdrževana programska oprema nudi nizek nivo varnosti. Pomembni sta še kakovost razvijalcev in uporaba primernih razvojnih orodij (Hiong, 2005, str. 9).

Pri zagotavljanju varnosti programske opreme ima pomembno vlogo testiranje. V naprej je nemogoče predvideti vse možne scenarije napak in zanje najti rešitve. Lahko bi dejali, da so odprtokodne rešitve v prednosti. Zaradi prostega dostopa do izvorne kode lahko hitreje odkrijemo napake, po drugi strani pa lahko ravno dostop do izvorne kode poveča ranljivost.

1.2.3 Fleksibilnost

Na prvi pogled se zdi odprtokodna programska oprema fleksibilnejša od lastniške. Razlog je dostop do izvorne kode, možnost spreminjanja in prilagoditve programske opreme našim potrebam. Ne smemo zanemariti dejstva, da nam brez ustreznih znanj, možnost spreminjanja in prilagajanja izvorne kode ne koristi.

Obstajajo viri za prilagoditev in odpravo napak pri obeh vrstah programske opreme. Uradni popravki in nadgradnje so največkrat na voljo le pri proizvajalcih lastniške programske opreme. Slednjim je v interesu zagotoviti zanesljivost svojih rešitev. Od njihovih produktov je odvisna njihova kredibilnost. Popravki in nadgradnje za odprto kodo prihajajo iz velikega števila virov. V večini primerov so zelo hitro na voljo, vendar pa je vprašljiva njihova kakovost. Pri izbiri si moramo zastaviti vprašanje, ali res potrebujemo programsko opremo, ki nam bo omogočala spremembo izvorne kode in na ta način prilagoditev programske opreme našim potrebam (Hiong, 2005, str. 13).

2 Evropski trg programske opreme

V tem poglavju bomo predstavili evropski trg programske opreme. Začnemo s pregledom trenutnega stanja. Sledi pregled poslovnih in tehnoloških trendov, s katerimi se srečuje panoga programske opreme v današnjem času.

2.1 Obstoječe stanje na evropskem trgu

Proizvodnja programske opreme predstavlja pomemben del evropskega gospodarstva, inovacij, sprememb in razvoja. Poraba za programsko opremo in sorodne storitve znaša

2,6 odstotka bruto domačega proizvoda, kar predstavlja 258 milijard €. Proizvodnja programske opreme ustvarja prihodke z naslova davkov in ustvarja delovna mesta z visoko dodano vrednostjo. Programska oprema je rastoč sektor informacijske in komunikacijske tehnologije. Trg programske opreme letno pridobi 6,9 odstotkov, kar je več od celotne panoge, ki povprečno raste 6 odstotkov letno. Rast je izrazito močna v Centralni in Vzhodni Evropi. Proizvodnja programske opreme je pomembna tudi zaradi ustvarjanja dodatne vrednosti. Letni prihodek na zaposlenega znaša več kot 100.000 € in je med najvišjimi v celotnem gospodarstvu (Industry expert group, 2009, str. 4).

*Tabela 2: Največji svetovni proizvajalci programske opreme v letu 2009**

Mesto	Podjetje	Prihodki od prodaje programske opreme (milijon €)	Celotni prihodki (milijon €)	Delež prihodkov programske opreme (v odstotkih)
1.	Microsoft	33.935	42.476	80
2.	IBM	15.157	71.111	21
3.	Oracle	12.050	15.166	79
4.	SAP	7.963	11.055	72
5.	Ericsson	4.979	18.201	27
6.	Nintendo	4.972	13.646	36
7.	HP	4.284	80.860	5
8.	Symantec	3.906	4.222	93
9.	Nokia Siemens Networks	3.655	14.622	25
10.	Activision Blizzard	3.172	3.453	92

Legenda:* Tečaj 1€ = 1,4573\$

Vir: Global software top 100, The world's largest software companies, 2009.

Zgornja tabela prikazuje 10 največjih svetovnih proizvajalcev programske opreme. V prvi deseterici najdemo tudi štiri evropska podjetja, med katerimi je največje SAP iz Nemčije. Izpostaviti je potrebno, da za nekatera navedena podjetja razvijanje programske opreme ne predstavlja primarne dejavnosti in jim posledično delež prihodkov z naslova programske

opreme predstavlja majhen odstotek celotnih prihodkov. Prvo mesto na navedeni lestvici zaseda Microsoft, najmočnejši predstavnik med proizvajalci lastniške programske opreme. Microsoft je bil do uporabe odprte kode na začetku precej zadržan in ji celo nasprotoval. V zadnjih letih se zanimanje za odprto kodo z njihove strani povečuje, tako da so začeli sodelovati tudi pri odprtokodnih projektih.

2.2 Poslovni in tehnološki trendi

Programska oprema predstavlja visoko tehnološki izdelek, zato ni presenetljivo dejstvo, da se trg programske opreme srečuje s hitrimi spremembami. Pojavljajo se nove tehnologije in poslovni modeli. Podjetja, ki se niso sposobna prilagoditi novim razmeram, so v večini primerov obsojena na propad. Ponudba programske opreme je postala zelo raznolika in obsežna, zato morajo biti podjetja v iskanju konkurenčne prednosti aktivna in se v čim večji meri prilagoditi povpraševanju, ki predstavlja gonilno silo gospodarstva. Podjetja zato iščejo nove načine, ki jim omogočajo doseganje prednosti pred svojimi konkurenti. Kakovost izdelka je sicer izredno važna, vendar se je ob tem potrebno zavedati dejstva, da nam tudi najbolj kakovostni izdelek ne koristi, če se na trgu po njem ne povprašuje.

2.2.1 Spremembe povpraševanja

Do sprememb v povpraševanju po programski opremi je prišlo zaradi več dejavnikov, med katerimi so najpomembnejši (Industry expert group, 2009, str. 19):

- globalizacija,
- spremenjena tržna pričakovanja,
- modularne organizacijske strukture,
- sprememba načina nakupa informacijske tehnologije in
- menjava generacije.

Danes svet velikokrat poimenujemo z besedno zvezo globalna vas. Trendi globalizacije so privedli do soodvisnosti svetovnih gospodarstev. Povezava in stopnja globalizacije je prišla do izraza ob izbruhu zadnje finančne krize, ki še vedno traja, vendar se že kažejo znaki okrevanja. Vsa podjetja, ki poslujejo v svetovnem gospodarstvu delujejo na podoben način in uporabljajo prakse, ki so se uveljavile v letih mednarodnega poslovanja. Geografska oddaljenost ne predstavlja več ovir. Razlog za slednje je dobro razvit sistem logistike in transporta. Podjetja niso več omejena na svoje lokalne trge, zato so prisiljena razmišljati globalno. Globalizacija je med drugim prinesla tudi večjo konkurenčnost, ki sili podjetja v preišljeno ravnanje s svojimi viri in racionalno obnašanje.

Spremenjena tržna pričakovanja so močno povezana s fleksibilnostjo. Ne glede na velikost se morajo podjetja truditi ohraniti svojo zmožnost čim hitrejšega prilagajanja spremembam tržnega povpraševanja. Potrošniki in podjetja začenjajo na programsko opremo gledati kot na storitev, zaradi česar so podjetja prisiljena uvajati nove storitveno naravnane poslovne modele.

V zadnjem desetletju se je zaradi potrebe po inovativnosti in hitrejšega prilagajanja

povpraševanju spremenila tudi organizacijska struktura organizacij, ki se ukvarjajo z razvojem programske opreme. Zaznati je prehod iz tradicionalnih organizacijskih struktur na bolj modularne, zaradi česar na nek način lahko govorimo o podjetju znotraj podjetja. Tak način organizacije potrebuje dober informacijski sistem in informacijske kanale, saj nam informacija koristi le v primeru, ko je pravi čas posredovana na pravo mesto pravi osebi. Modularne organizacijske strukture niso primerne za vse vrste organizacij.

Spremenil se je tudi način kupovanja informacijske tehnologije in programske opreme. Postopek pridobivanja programske opreme je postal manj hierarhičen. Za odločitve o nakupih so zadolženi posebni oddelki, ki morajo v skladu s cilji podjetja izbrati najbolj primerne rešitve.

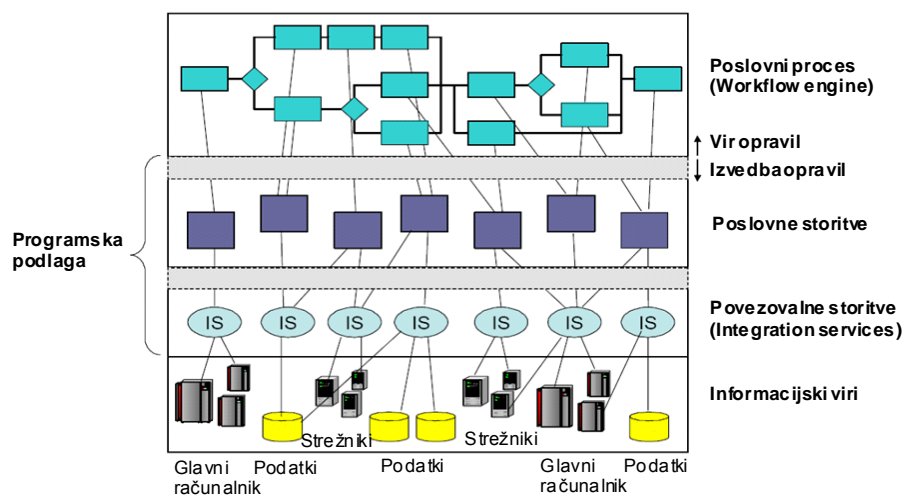
Zadnji dejavnik, ki vpliva na spremembo povpraševanja je sprememba generacije. S prehodom v informacijsko družbo dobivamo ljudi, ki so odraščali ob internetu in informacijski tehnologiji. Nova generacije dela, komunicira in preživlja prosti čas drugače, kot so to počele generacije pred njo.

2.2.2 Razvoj tehnologije

Na razvoj programske opreme vpliva veliko število različnih dejavnikov. Med njimi so najpomembnejši tisti, ki so povezani z drugačnim zaznavanjem programske opreme in razširitev svetovnega spleta. Izpostavimo sledeče tri:

- storitveno usmerjena arhitektura (angl. *Service Oriented Architecture*),
- razpršeni računalniški sistemi (angl. *Cloud Computing*),
- semantični splet (angl. *Semantic Web*).

Slika 1: Storitveno usmerjena arhitektura



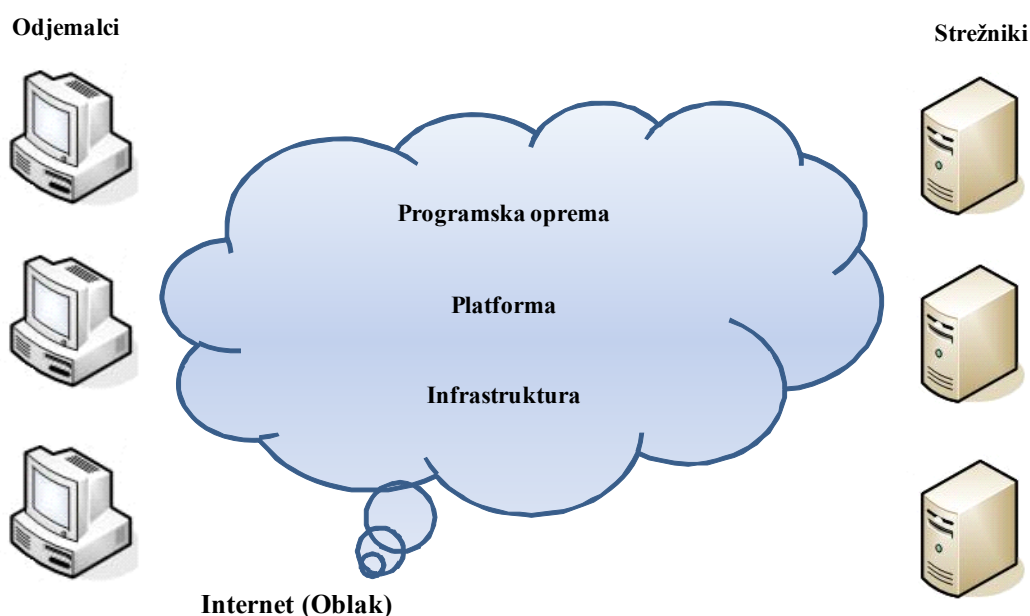
Vir: Keber, *Obvladovanje poslovnih procesov z uporabo ogrodij in storitveno usmerjene arhitekture*, 2006.

Zgornja slika prikazuje storitveno usmerjeno arhitekturo. Pri storitveno usmerjeni arhitekturi govorimo o novi paradigmi razvoja programske opreme. Pristop združuje več neodvisnih programskih rešitev in storitev z namenom učinkovite podpore poslovnim procesom.

Primarni namen arhitekture je omogočanje povezljivosti sistema na aplikativnem nivoju, neodvisno od tehnologije, orodij in operacijskih sistemov, s katerimi se aplikacija dejansko izvede. Storitve povezuje enotni način medsebojne komunikacije (Storitveno naravnana arhitektura (SOA), 2010).

Razpršeni računalniški sistemi predstavljajo nov način zasnove in uporabe oddaljenih računalniških virov. Sistemi uporabnikom omogočajo uporabo zahtevnejših računalniških virov v obliki storitev na zahtevo in so zgrajeni na podlagi raziskav mrežnega in storitvenega računalništva. Računalniški in pomnilniški viri so virtualizirani in za uporabnike dostopni v obliki storitve (Računalništvo v oblaku, 2009).

Slika 2: Razpršeni računalniški sistem



Vir: Glavač, Računalništvo v oblakih in virtualizacija, 2009, str. 27.

Na zgornji sliki je predstavljen razpršeni računalniški sistem. Sestavlja ga več plasti. Definiramo ga lahko kot večplastno strukturo storitev, ki nastopajo v vmesnem sloju med strežniki na eni in odjemalci na drugi strani. Storitve so v obliki storitev na zahtevo. Sloji oz. storitve na zahtevo so med seboj hierarhično odvisni. Prispodoba oblaka je vezana na internet oz. omrežje in s tem na uporabo omrežno dostopnih storitev (Glavač, 2009, str. 39).

Besedna zveza semantični splet je povezana z ogromno količino podatkov na svetovnem spletu, ki so zapisane v računalniku razumljivi obliki. Svetovni splet ni le skladišče statičnih podatkov, omogoča tudi dostop do storitev. Ko imamo podatke pripravljene v računalniku razumljivem jeziku, v naslednjem koraku zagotovimo neposredno komunikacijo preko spleta (Krisper & Lavbič, 2006). Semantičnega spleta ne predstavljajo le spletne strani in povezave med njimi, temveč tudi odnosi med podatki, ki omogočajo strojni opremi razumeti semantične dokumente in podatke. Na ta način se omogoči povezava podatkov pomensko in ne le preko povezav (Web 3.0 – Semantični splet). Cilj semantičnih spletnih storitev je zagotavljanje

povezljivosti tehnologije semantičnega spleta in spletnih storitev (Krisper & Lavbič, 2006).

Spodnja slika prikazuje osnovne elemente semantičnega spleta. Ontologije so slovarji znanj, ki se navezujejo na določena področja. Ustvarjamo jih lahko vsi, vsebujejo pa definicijo pojmov kot so razredi, lastnosti in odnosi med njimi. Meta podatki predstavljajo reference do pojmov, ki so definirani v ontologiji. Inteligentni programi opravljajo analizo meta podatkov in z uporabo ontologije vračajo poizvedbam relevantne podatke (Web 3.0 – Semantični splet).

Slika 3: Osnovni elementi semantičnega spleta



Vir: Web 3.0 – Semantični splet, 2010.

2.2.3 Nove tržne strukture

Tehnološke in poslovne spremembe vplivajo na spreminjanje trga programske opreme. Govorimo o zrelem, razvitem trgu, ki se trenutno nahaja v fazi konsolidacije. Podjetja z novimi poslovnimi modeli želijo doseči konkurenčno prednost in si s tem priboriti večje tržne deleže. Največja sprememba se pojavlja v načinu prodaje programske opreme, ki se v današnjem času trži kot storitev. Za evropske proizvajalce je še vedno značilno, da se preveč osredotočajo na svoje bližnje trge. Prevzem podjetji velja za enega izmed lažjih načinov širitve na tuje trge, ne smemo pa pozabiti, da so tovrstne širitve oz. vstopi na trg zelo tvegani. Na ta način podjetja povečajo svoj krog potencialnih kupcev.

Za trg programske opreme ja značilna razdelitev na segmente. Letno se pojavljajo manjša inovativna podjetja, ki lahko hitro postanejo vodilna v novih segmentih trga. Uspešna mala podjetja so v večini primerov obsojena na prevzem s strani velikih podjetij, kar ni nujno slabo. Nekateri podjetniki namreč ustanavljajo mala, inovativna podjetja s ciljem prodaje in podjetniške žetve.

2.3 Izzivi evropskega trga programske opreme

Evropski trg programske opreme se srečuje z več izzivi, ki jih bo potrebno rešiti, če bomo želeli ostati konkurenčni na svetovnem trgu. V nadaljevanju izpostavimo prepoznavanje človeškega talenta in spodbujanje inovativnosti. Ideje za omenjena izziva smo črpali iz

poročila Industry expert group (2009) z naslovom: Report of an industry expert group on a european software strategy .

2.3.1 Prepoznavanje človeškega talenta

Ključni dejavnik pri razvoju programske opreme predstavljajo razvijalci. Na svetovnih trgih lahko uspejo samo podjetja, ki uspejo prepoznati in uporabiti človeški talent pri razvoju svojih izdelkov. Evropa se v zadnjem desetletju srečuje s pomanjkanjem strokovnjakov na področju informacijske tehnologije, saj se zanimanje za študij računalništva in informatike z leti zmanjšuje. Eden izmed razlogov je slabo definirana karierna pot, ki je možna po končanem študiju. Prav tako ni zanemarljivo dejstvo, da visoko tehnološke panoge zahtevajo veliko dodatnega izobraževanja in učenja novih stvari.

Eden izmed glavnih izzivov evropskega trga programske opreme bo omogočiti okolje, ki bo spodbujalo izobraževanje strokovnega kadra na področju informacijske tehnologije. V naslednjem koraku bo potrebno zagotoviti zadržanje tega kadra in ustvariti inovativnosti prijazno okolje.

2.3.2 Spodbujanje inovativnosti

Spodbujanje inovativnosti je v veliki meri povezano z višino sredstev namenjenih za raziskave in razvoj. Evropa bi veliko pozornosti morala nameniti hitro rastočim podjetjem, ki predstavljajo inkubator za nove inovativne ideje. Prav tako bi lahko več pozornosti namenila skladom, ki se ukvarjajo z iskanjem in financiranjem mladih, obetajočih podjetij. Danes Evropa zaostaja za Združenimi Državami Amerike na večini področij, ki so pomembna za razvoj inovativnosti.

3 Poslovni model odprte kode

Izraz poslovni model se je začel pogosteje uporabljati odkar govorimo o novi ekonomiji. Za omenjeni termin obstaja več različnih definicij. Avtorji se strinjajo v točki, da poslovni model predstavlja način poslovanja, s katerim podjetja ustvarjajo svoje prihodke. Poznamo enostavne in kompleksnejše primere. Primer enostavnega poslovnega modela je proizvodno podjetje, ki svoj proizvod prodaja kupcem in na ta način ustvarja prihodke. S kompleksnejšimi poslovnimi modeli se srečamo v primerih, kjer na prvi pogled ni jasno, na kakšen način se ustvarjajo prihodki. Primer za to so komercialne televizijske postaje (Marc, 2003, str. 21).

Komercialne televizijske postaje svoje prihodke ustvarjajo na več načinov. Predpogoj za uspešno poslovanje predstavlja dober program, ki zagotovi stalni nabor gledalcev. Prihodke komercialne televizije nato ustvarimo s prodajo oglaševalskega prostora. Eden izmed načinov ustvarjanja prihodkov je ustvarjanje resničnostnih oddaj in prenos prireditev, kjer gledalci lahko preko telefonskih klicev in sms sporočil oddajo svoje glasove, ali podarijo sredstva v dobrodelne namene. Na omenjeni način si televizijske postaje zagotovijo provizijo. S pojavom digitalne televizije komercialne televizijske postaje prodajajo pravice za

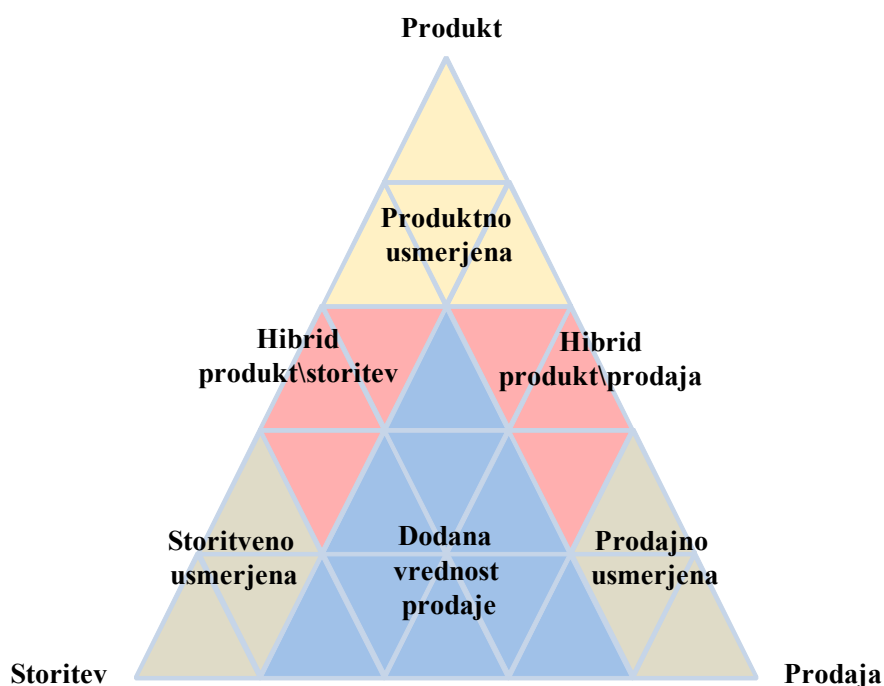
prikazovanje njihovega programa pri ponudnikih internetne in satelitske televizije.

3.1 Trženje lastniške in odprtokodne programske opreme

Podjetja, ki se ukvarjajo z razvojem programskih rešitev, lahko ustvarjajo prihodke na več različnih načinov (Forfas, 2006, str. 25):

- uporabniki kupijo licenco in s tem pravico do uporabe programske opreme,
- zaračunajo za storitve, povezane s programsko opremo,
- zaračunajo provizijo zaradi prenosa intelektualne lastnine na drugo podjetje.

Slika 4: Usmerjenost programske opreme



Vir: Forfas, *Open Source Trends and Business Models*, 2006, str. 25.

Večina razvijalcev lastniške programske opreme se poslužuje tradicionalnega pristopa, kar pomeni, da morajo uporabniki za uporabo programe opreme kupiti licence. Višina proizvajalčevega prihodka je odvisna od števila prodanih licenc. Slabost omenjenega pristopa je težko napovedljiva višina prihodkov in njihova cikličnost. V obdobju nadgradenj in novih izdaj se prihodki zelo povečajo, v zrelem obdobju izdelka pa začnejo prihodki padati.

Podjetja, ki se ukvarjajo z razvojem odprtokodnih rešitev imajo praviloma storitveno naravnane poslovne modele. Stroški pridobitve programske opreme so za uporabnike načeloma nizki. Prihodke si podjetja ustvarijo s svetovanjem in zagotavljanjem podpore. Prihodki so na ta način manj ciklični in lažje napovedljivi, kot pri proizvodnji lastniške programske opreme.

3.2 Najbolj razširjeni poslovni modeli

Poglavje nadaljujemo s predstavitevijo najbolj razširjenih poslovnih modelov odprtokodne programske opreme, ki so: zagotavljanje podpore in storitev, programska oprema kot storitev in razvoj lastniške programske opreme na podlagi odprtokodne.

3.2.1 Zagotavljanje storitev in podpore

Za poslovni model zagotavljanja podpore ni nujno, da vključuje lastništvo programske kode. Podjetja se lahko osredotočijo na zagotavljanje storitev, ki so povezane z odprtokodno programsko opremo. Potreba po podpori se pojavlja pri uporabi skoraj vse programske opreme, večja je v primeru, ko izbrana možnost ne zadovolji vseh potreb kupca in jo je potrebno prilagoditi ali zagotoviti skladnost z obstoječimi sistemi. Tipične storitve so naslednje (Forfas, 2006, str. 30):

- integracija in testiranje odprtokodne programske rešitve,
- prilagoditev odprtokodne rešitve uporabnikovim potrebam,
- zagotavljanje popravkov in posodobitev,
- pravno svetovanje v povezavi z rabo odprte kode.

Tako kot pri vsakem modelu, se tudi v tem primeru srečamo s prednostmi in slabostmi. Ko govorimo o prednostih je potrebno omeniti, da velika podjetja potrebujejo podporo za svojo programsko opremo. Razmerja se urejajo s podpisom pogodb za obdobja od enega do več let. Pri navedenem modelu podjetja lažje predvidijo svoje prihodke in pridejo do sredstev (npr. nižja obrestna mera pri najemu sredstev), ki jih potrebujejo za svoje delovanje. Zagotavljanje podpore ni omejeno le na določen tip programske opreme in se ga lahko uporablja tudi pri lastniški programski opremi, kjer ni podanih omejitev s strani proizvajalca oz. lastnika.

Model ima tudi določene pomanjkljivosti. Uporabniki odprtokodnih programskih rešitev so lahko mnenja, da ne potrebujejo strokovne pomoči. Na voljo je namreč veliko informacij iz različnih informacijskih virov, ki nam delo z odprto kodo zelo olajšajo. Model zahteva stalno bazo naročnikov, ki so pripravljeni plačati za naše storitve. Ko govorimo o zagotavljanju podpore za odprto kodo ne smemo pozabiti na konkurenco, ki nam jo lahko predstavljajo vsa podjetja, ki se ukvarjajo z zagotavljanjem tovrstne podpore.

3.2.2 Programska oprema kot storitev

Koncept programske opreme kot storitve se je pojavil v času, ko so podjetjem začeli naraščati stroški povezani z informacijsko tehnologijo. Nakup programske opreme ne predstavlja celotne investicije. Velik del predstavljajo stroški povezani z implementacijo, svetovanjem pri vpeljavi izbranega sistema, šolanjem uporabnikov in vzdrževanju programske opreme. Prav tako niso zanemarljivi izdatki povezani s strojno opremo (Software as a service (SaaS) definition and solutions, 2010).

Podjetje se namesto za nakup programske opreme odloči za najem pri ponudniku, ki svojo programsko opremo ponuja kot storitev. Podjetje si na ta način zagotovi večjo fleksibilnost,

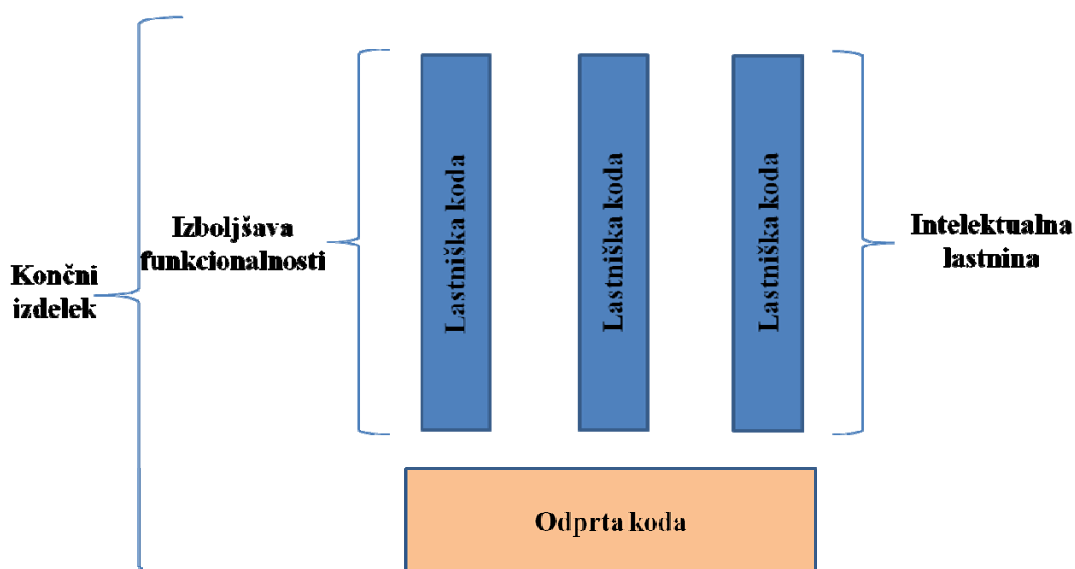
saj teoretično lahko v vsakem trenutku prekinejo sodelovanje z obstoječim ponudnikom in se odločijo za novega. Problemi nastanejo v primeru različnih partnerstev, ki podjetja lahko pogodbeno vežejo na določenega ponudnika programske opreme. Podjetju se ni treba obremenjevati z vzdrževanjem programske opreme, za to skrbi ponudnik storitev. Uporabniki do programske opreme dostopajo preko interneta. Za uporabo programa ni potrebna lokalna inštalacija, vse kar potrebujemo je širokopasovna internetna povezava.

3.2.3 Razvoj lastniške programske opreme na podlagi odprtokodne

Govorimo o hibridnem modelu, ki združuje lastnosti odprte kode z lastniško programsko opremo. Razvijalci na osnovi odprtokodne razvijejo programsko rešitev, ki izboljšuje njeno uporabnost in na ta način ustvarjajo dodano vrednost v očeh uporabnika. Prednost omenjenega modela je, da ohranja poznane ekonomske modele, ki omogočajo visoke stopnje dobičkonosnosti, prav tako na ta način zajamemo večje število potencialnih uporabnikov izdelka. Glavno pomanjkljivost modela predstavlja dejstvo, da konkurenti lahko hitreje izdelajo konkurenčen produkt, kot v primeru prodaje izključno lastniške programske opreme.

Pri izbiri razvoja lastniške programske opreme na podlagi odprtokodne, je potrebno paziti, da so deli programske opreme, ki so razviti po načelih odprte kode jasno ločeni od delov, ki si jih razvojno podjetje lasti. V primeru, da ni jasne ločnice, lahko razvojnim podjetjem odprta koda prinese več težav kot koristi. Pojavijo se namreč vprašanja intelektualne lastnine in pravičnega licenciranja programske opreme. Hibridni model je prikazan na spodnji sliki.

Slika 5: Hibridni model razvoja programskih rešitev



Vir: Forfas, *Open Source Trends and Business Models*, 2006, str. 34.

4 Poslovne strategije odprte kode

Poslovna strategija vsebuje cilje glede proizvodov in trgov za določena poslovna področja in s tem nakazuje prihodnje aktivnosti podjetja. Poslovna strategija je v večini primerov izvedena iz celotne strategije podjetja oz. je z njo usklajena. Proces oblikovanja poslovne

strategije se začne z definicijo poslanstva in ciljev poslovnih enot. Sledi ocenjevanje prednosti in slabosti ter priložnosti in nevarnosti, ki predstavljajo temelj za njeno oblikovanje (Dimovski, Penger & Žnidaršič, 2003, str. 100).

Posamezniki in podjetja imajo danes na izbiro veliko število načinov, s katerimi lahko zadovoljijo svoje informacijske in komunikacijske potrebe. Pri izbiri med posameznimi možnostmi postaja vse privlačnejša uporaba odprto kodnih produktov. Odprta koda podjetjem omogoča izbiro med širokim naborom ponudnikov strojne in programske opreme, ki v največji meri zadosti njihovim potrebam. Uporaba odprte kode lahko predstavlja tudi veliko konkurenčno prednost, pri čemer se je potrebno zavedati, da brez prave strategije, uspešnost poslovanja ni zagotovljena.

Podjetja si z odprto kodo lahko svoje poslovanje izboljšajo na več načinov, pri čemer pa strategije niso namenjene le podjetjem, ki razvijajo odprtokodne programske rešitve, temveč so nekatere povezane zgolj z njeno uporabo. Spodaj navedene in opisane strategije so primerne za vsa podjetja, katerih poslovanje je tako ali drugače povezano z odprto kodo. Podjetja se največkrat ne poslužujejo zgolj ene strategije ampak je njihova krovna strategija mešanica več strategij. Omenjeno velja za večja podjetja, medtem ko se lahko manjša odločijo za točno določeno strategijo in s tem poskusijo osvojiti manjše tržne segmente. Prednost manjših podjetij je, da se lahko hitreje prilagodijo spremembam.

Idejo za izbranih sedem strategij smo dobili na podlagi avtorja Johna Koeniga (2004), ki je v dokumentu z naslovom Open source business strategies predstavil poslovne strategije, ki se jih lahko poslužijo podjetja pri uporabi ali razvoju programske opreme.

4.1 Optimizacijska strategija

Optimizacijsko strategijo je predstavil Clayton Cristensen. Govorimo o zakonu ohranjanja modularnosti. Odprtokodna programska oprema razvita na ta način vsebuje sloj oz. modul, ki se ga da optimizirati in na ta način prilagajati drugim programskim rešitvam. Modul priredimo na način, ki omogoča boljše delovanje drugih aplikacij in tako optimiziramo delovanje celotnega sistema. Na omenjen način ustvarimo dodatno vrednost programske opreme. Primer modularne programske rešitve predstavlja odprtokodni operacijski sistem Linux (Koeing, 2004, str. 3).

Optimizacijsko strategijo lahko uporabljajo vsi razvijalci programske opreme. Eden izmed najbolj znanih predstavnikov je Oracle. Z uporabo modularnega odprtokodnega sistema in njihove programske rešitve za baze podatkov so prišli do rešitve Oracle Real Application Cluster. Sistem omogoča izvajanje in uporabo ene podatkovne baze na več strežnikih. Na ta način so dosegli boljše delovanje in zaradi optimizacije znižali stroške, ki so povezani s strojno opremo. Celoten sistem namreč potrebuje manj virov. Zaradi ustvarjanja večje dodane vrednosti svoje programske opreme, lahko postavijo višje cene, kot v primeru, da optimizacijske strategije nebi uporabljali.

4.2 Strategija dvojnega licenciranja

Podjetja pri uporabi strategije dvojnega licenciranja svoje izdelke ponudijo na dva načina. Programsko opremo lahko ponudijo brezplačno. Pri tem načinu veljajo določene omejitve. Vse spremembe programa morajo biti javno dostopne v izvorni kodi, prav tako programske rešitve ne smejo uporabiti kot komponente v drugih produktih. V primeru, da podjetje za svoj produkt zahteva plačilo, prej omenjene omejitve ne veljajo.

Z uporabo strategije dvojnega licenciranja, si podjetje zagotovi več različnih tokov prihodkov. Dodaten vir namreč predstavljajo dodatne storitve, ki jih ponujajo svojim strankam, kot so na primer: svetovanje, izobraževanje uporabnikov programske opreme, ipd. Z omenjenim pristopom zajamemo večjo bazo potencialnih uporabnikov in kupcev. Brezplačne verzije se s spleta prenesene v velikim številu, kar podjetjem pomaga pri prepoznavnosti izdelkov. Tovrstna strategija podjetjem predstavlja pomembno orodje pri branjenju tržnega deleža.

Predstavniki omenjene strategije je MySQL odprtokodna podatkovna baza. Velja za najbolj razširjeno in najbolj priljubljeno odprtokodno orodje za delo z bazami podatkov. Osnovne verzije so na voljo brezplačno, medtem ko je za napredne verzije, kot je na primer MySQL Enterprise, potrebno plačati.

4.3 Naročniška strategija

Naročniške strategije se poslužuje veliko podjetji, ki se ukvarjajo z razvojem odprtokodnih programskih rešitev. Logika je preprosta, podjetje se s pogodbo veže, da bo izpolnjevalo dogovorjene aktivnosti za čas trajanja naročniškega razmerja. Naročnina v večini primerov ne vključuje le uporabe programske opreme, ampak tudi vse podporne aktivnosti povezane z njo. Za tradicionalni model je značilno, da so nadgradnje drage, podpora omejena, prav tako ponudniki nimajo interesa popravljati programske opreme med nadgradnjami. Uporabniki so torej »obsojeni« na čakanje izida nove verzije.

Omenjeno strategijo je med prvimi predstavilo podjetje Red Hat. Strategija omogoča, da svojim strankam omogočijo nižje stroške in uporabo najnovejše tehnologije. Uporabniki imajo neomejen dostop do popravkov in nadgradenj. Naročniška strategija ima med drugim tudi sledeče prednosti (Red Hat, 2010, str. 2):

- varnost sistema,
- boljše delovanje sistema in dostopna podpora,
- avtomatizirane vsakodnevne vzdrževalne naloge in
- hiter dostop do najnovejših verzij.

4.4 Svetovalna strategija

Z odprto kodo ne služijo le podjetja, ki se ukvarjajo z razvojem odprtokodnih programskih rešitev. Svetovalno strategijo lahko uporabijo vsa svetovalna podjetja, ki delujejo na področju informacijske tehnologije. Svojim strankam zagotovijo pomoč pri integraciji in uporabi

popularnih odprtokodnih rešitev, kot so na primer: MySQL, Apache, Eclipse, itd.

Odprta koda predstavlja priložnosti in nevarnosti za podjetja, ki želijo z njeno uporabo optimizirati svoje delovanje ali znižati stroške. Problemi nastanejo v primeru, ko uporabniki ne poznajo dovolj dobro svojih potreb in trga odprtokodnih programskih rešitev, saj z vpeljavo neustrezne programske rešitve, lahko naredimo več škode kot koristi. Zgodi se, da izbrana informacijska rešitev ne zadovoljuje informacijskih potreb podjetja. Vzroke za nastalo stanje iščemo predvsem v nepoznavanju problematike in nezadostnemu znanju.

V preteklem desetletju, se je uveljavilo več podjetij, ki se ukvarjajo s svetovanjem in pomočjo pri izbiri odprtokodne programske opreme. Vsako podjetje ima namreč specifične potrebe, ki jih je potrebno upoštevati pri izbiri izdelka. Svetovalna podjetja, so v letih delovanja razvila različne tehnike in primere dobre prakse, ki koristijo naročnikom in na nek način predstavljajo garancijo, da bo izbrana rešitev nudila tisto, kar potrebujemo. Tipični model dejavnosti, ki potekajo znotraj procesa svetovanja je sestavljen iz treh faz (Olliance group, 2010):

- ocene stanja,
- izdelave plana in
- implementacije.

4.5 Pokroviteljska strategija

O pokroviteljski strategiji govorimo v primeru, ko podjetje prispeva čas, energijo, razvijalce in programsko kodo za razvoj odprtokodne programske opreme. Obstaja več razlogov, zakaj se podjetja odločajo za omenjeno strategijo. Primer je podjetje IBM, ki se strategije med drugim poslužuje zaradi razširjanja svojih standardov, ki jih želi uveljaviti na trgu programske opreme in s tem doseči večji ugled in prepoznavnost (Koeing, 2004, str. 8).

Uspeh pokroviteljske strategije ni odvisen le od izvorne kode. Pokrovitelj mora zagotavljati doslednost in dobro voditi odprtokodni projekt. V nasprotnem primeru je velika možnost neuspeha. Primer uspešnega projekta predstavlja izdelek Apache spletni strežnik. Microsoft je namreč začel povečevati svoj delež na trgu spletnih strežnikov in lahko bi se zgodilo, da bi se zgodba končala enako kot na trgu internetnih brskalnikov. Microsoft je namreč konec 90-ih let obvladoval 90 odstotkov omenjenega trga. IBM je s pomočjo pokroviteljske strategije Microsoftu preprečil monopolizacijo trga spletnih strežnikov. Priljubljenost strežnika Apache se je še povečala in danes obvladuje trg spletnih strežnikov z več kot 70 odstotnim tržnim deležem.

IBM je zelo dejaven na trgu odprtokodne programske opreme, potrebno pa je omeniti, da večino prihodkov še vedno ustvarjajo s prodajo lastniških izdelkov, kar dokazuje, da odprta koda ni nujno neugodna za prihodke iz naslova lastništva.

4.6 Gostiteljska strategija

Gostiteljske strategije se v največji meri poslužujejo ponudniki sodobnih informacijskih storitev. Uporaba odprte kode ni omejena, prav tako podjetja niso zavezana delitvi in objavi

programske kode. Omejitve veljajo le v primeru redistribucije programske opreme. Podjetja lahko prosto spreminjajo in prilagodijo izvorno kodo tako, da najbolj zadovoljijo svoje potrebe. Uporaba odprtokodne programske opreme jim omogoči znižanje stroškov ob enaki kakovosti in zanesljivosti storitev.

Med velikimi uporabniki odprte kode najdemo tudi dve svetovno znani multinacionalki: Amazon in Google. Omenjeni podjetji sta z migracijo na tehnološke platforme, katerih osnova je odprtokodni Linux, prihranili milijone dolarjev. Stroški so se znižali predvsem na segmentu strojne opreme. Zaradi optimiziranega delovanja sta podjetji potrebovali manj strežniške strojne opreme, kot bi jo v primeru uporabe drugih rešitev.

4.7 Strategija vgrajene programske opreme

Strategijo podjetje lahko uporablja pri vseh elektronskih sistemih, ki za svoje delovanje potrebujejo računalniški čip in niso računalniki. Omenjeni sistemi uporabljajo več različnih vrst čipov, od mikroprocesorjev, do po meri izdelanih čipov. Število elektronskih sistemov se iz leta v leto povečuje, vgrajujejo jih v avtomobile, vlake, stroje, kamere, zabavno elektroniko, omrežne naprave, itd. Programska oprema je v večini primerov naložena na flash ali bralni pomnilnik. Zaradi omejenih kapacitet strojne opreme, se program ob vsakem izvajanju naloži v delovni pomnilnik.

Na trgu vgrajene programske opreme prevladujejo Linux in Unix-like sistemi. Uporabljajo jih v velikem številu velikih in majhnih potrošniških naprav. Postaja prva izbira za proizvajalce, ki se ukvarjajo z izdelavo cenovno ugodnih komunikacijskih produktov. Linux predstavlja platformo, ki je funkcionalna, razširljiva in jo je mogoče uvesti z minimalnimi kapitalskimi stroški. Eno izmed večjih prednosti predstavlja dejstvo, da Linux ni omejen na določene platforme in se zato lahko izvaja na velikem številu različnih sistemov. Slednje nam omogoča manjše stroške strojne opreme, ki nastanejo pri razvijanju in testiranju programskih rešitev. Tako podjetja lahko investirajo več sredstev v segmente, ki bodo ustvarili dodatno vrednost za kupce.

Potreba po vgrajeni programski opremi z leti narašča. Za proizvajalce vgrajene programske opreme je zadnje čase postal zanimiv trg internetne televizije (IPTV). Število uporabnikov IPTV-ja strmo narašča, s čimer se povečuje potreba po kvalitetni programski opremi vgrajeni v elektronske naprave, ki uporabnikom omogočajo sprejemanje signala in prikazovanje na televizijskem sprejemniku.

5 Odprta koda – vir konkurenčne prednosti

V nadaljevanju obravnavamo odprto kodo s stališča konkurenčne prednosti, ki nam jo lahko nudi njena uporaba. Predstavljeni so razlogi za uporabo in izzivi ter priložnosti odprte kode v prihodnosti.

5.1 Razlogi za uporabo odprte kode

Obstaja več razlogov za uporabo odprtokodne programske opreme. Podjetja jo namreč ne uporabljajo zgolj na odprtokodnih projektih, ampak tudi pri razvoju lastniške programske opreme. Odprtokodna programska oprema nam nudi veliko prednosti v primerjavi z lastniško.

5.1.1 Odprta koda pri razvoju lastniške programske opreme

Na tej točki si je smiselno zastaviti vprašanje, zakaj bi se podjetje odločilo za uporabo odprte kode. Odprta koda podjetjem omogoči, da se osredotočijo na stvari, ki jih delajo najboljše. Vsako podjetje, ki se ukvarja z razvijanjem programske opreme je v želji po optimizaciji poslovanja in nižanju razvojnih stroškov postavljeno pred odločitev. Za razvoj izdelka je potrebno veliko programerskega časa, tudi za stvari, ki ne predstavljajo dodane vrednosti in našega programa ne ločijo od druge programske opreme na trgu. Primer je razvoj vmesne programske opreme (angl. *middleware*), ki skrbi za komunikacijo z drugimi aplikacijami. Problema pa ne predstavlja le stroškovna plat lastnega razvoja vmesne programske opreme. Dejstvo je, da interni razvoj nikoli oz. težko doseže zmogljivosti, povezanost in kvaliteto obstoječih odprtokodnih rešitev. Ob uporabi odprtokodne vmesne programske opreme lahko več virov namenimo za izdelavo ključnih delov programa, ki nam bodo na trgu zagotovili diferenciacijo od drugih proizvodov. Uporaba odprte kode v omenjene namene nam pomaga znižati stroške razvoja. Potrebno je upoštevati tudi dejstvo, da programerji radi razvijajo nove stvari in je le malo takih, ki z veseljem delajo na nizko prioritetenih delih programske opreme (Independent Software Vendors, 2009).

Trg programske opreme je v zreli fazi. Imamo veliko število ponudnikov raznovrstne programske opreme, zaradi česar se je z leti zelo povečala konkurenčnost panoge, kar je spodbudilo tekmovanje med ponudniki, kljub povečevanju povpraševanja. Podjetja se srečujejo z nižanjem stroškov v iskanju višjih donosov in konkurenčne prednosti. Odprta koda je pri nižanju stroškov razvoja pomemben faktor. Z uporabo odprte kode lahko razvijamo boljše produkte pri nižjih razvojnih stroških.

5.1.2 Prednosti odprtokodne programske opreme v primerjavi z lastniško

Odprtokodna programska oprema nudi nekaj bistvenih prednosti v primerjavi z lastniškimi proizvodi. Razvijalci lastniške programske opreme velikokrat dajejo prednost vizualnim atributom. Lahko se zgodi, da zaradi tega trpita stabilnost in varnost, pri čemer razvijalci odprtokodnih programskih rešitev največ pozornosti namenijo prav slednjima. Zanje je značilna želja po izpopolnjevanju znanja ter iskanje samopotrditve in ugleda. Zato lahko sklepamo, da bodo pri razvoju programske opreme dali poudarek stvarim, ki bodo izboljšale predvsem delovanje in uporabnost (Primožič, 2005, str. 29).

Skupine in podjetja, ki se ukvarjajo z razvojem odprtokodne programske opreme se držijo načela »izdaj zgodaj, izdaj pogosto«. Zgodaj izdane verzije programske opreme omogočajo pridobitev kvalitetnih povratnih informacij, omogočajo tudi hitrejšo odkrivanje in odpravo

napak (Primožič, 2005, str. 29).

Odperta koda je znana tudi po svoji preglednosti. Zaradi dostopa do izvorne kode se lahko na lastne oči prepričamo, če je programska oprema prilagojena obstoječim standardom in razvita na način, ki bo v prihodnje omogočal spremembe. Lažje lahko podamo sodbo, ali bo programska oprema pokrila naše potrebe ali ne.

Prilagoditev odprtokodne programske opreme nam nudi možnost predelave programa, kar nam teoretično omogoča popolno zadovoljitev naših informacijskih potreb. Potrebno se je zavedati dejstva, da v praksi težko zadovoljimo vse naše potrebe, treba pa se je truditi, da stvari delamo na najbolj optimalen način, ki je možen v danem trenutku. Optimizacija poslovanja na kateri koli točki v končni fazi pomeni nižje stroške in večjo konkurenčnost našega izdelka.

5.2 Izzivi in priložnosti odprtokodne programske opreme

Z odprtokodno programsko opremo se srečujemo dnevno in sicer največkrat ob uporabi interneta, pregledovanju spletnih strani ali pošiljanju elektronske pošte. Ideja odprte kode je, da se ponovna raba izvorne kode ne zaključí na mejah podjetja. Z objavo in delitvijo izvorne kode dosežemo sodelovanje velikega števila razvijalcev z vsega sveta, s čimer dosežemo večjo skupno korist, ki jo nudi določena programska oprema. Odperta koda obstaja že desetletja, njeno uporabo je povečala predvsem prisotnost interneta. Internet je namreč omogočil sodelovanje razvijalcem iz celega sveta. Razpon odprte kode zajema veliko različnih vrst programske opreme, širši javnosti pa so najbolj znani predvsem Linux operacijski sistemi in na strežnikih temelječe aplikacije, kot sta Apache in MySQL (Tuma, 2005, str. 1). V zadnjih nekaj letih so se uveljavili tudi odprtokodni internetni brskalniki in alternativna pisarniška zbirka OpenOffice.org.

Odpertokodna programska oprema je postala zanimiva, ker jo podjetja lahko pridobijo brez večjih stroškov oz. v nekaterih primerih je pridobitev in uporaba povsem brezplačna. Z možnostjo uporabe odprtokodne programske opreme, so se najprej začele spogledovati organizacije in podjetja, ki se pri svojem poslovanju srečujejo s proračunskimi omejitvami. Sledila so jim manjša podjetja, ki so si z uporabo odprte kode uspela znižati stroške poslovanja in tako povečala svojo konkurenčnost. Prava moč odprte kode se pokaže v primeru, ko ima podjetje ali posameznik specifične potrebe po programski opremi. Odperta koda omogoča uporabnikom, da izvorno kodo spreminjajo in jo prilagodijo svojim potrebam. Prednosti ponovne rabe izvorne kode se pokažejo v prihrankih pri razvoju, testiranju in vzdrževanju. Prav tako je potrebno izpostaviti, da se nam ni potrebno lotiti pisanja izvorne kode za celotno programsko opremo, temveč zgolj poiščemo odprtokodno programsko rešitev, ki v čim večji meri zadovoljuje naše potrebe in prilagodimo oz. dogradimo program tako, da ustreza našim željam.

Velik izziv s katerim se srečuje odprtokodna programska oprema je varnost. Težko se opredelimo, katera vrsta programske opreme je varnejša lastniška ali odprtokodna. Oba razvojna pristopa imata svoje prednosti in pomanjkljivosti. Problem nastane, ker so v

večini primerov zagovorniki določene alternative goreči zagovorniki izbrane opcije in tako težko objektivno vrednotijo preučevane lastnosti. Omenjeno dejstvo dela debato zanimivo, na mestu pa je opozorilo, da ne smemo verjeti vsemu, kar slišimo in si skušati z lastnimi raziskavami ustvariti svoje mnenje oz. si pomagamo z mnenji neodvisnih strokovnjakov.

Večina podjetji se pri izbiri vrste programske opreme drži preprostega pravila. Izberejo tisto alternativo, ki pokrije njihove obstoječe potrebe in jim zagotavlja najboljšo vrednost za investiran oz. porabljen denar. Vodje projektov vpeljave programske opreme morajo pripraviti analize vseh možnosti, ki so na voljo in na podlagi rezultatov analize sprejeti ustrezno odločitev. Pri svojem delu, se vodje projektov večkrat srečajo s težavami, povezanimi z odprtokodno programsko opremo. Prva težava, ki se pojavi v večini primerov je identifikacija razpoložljive odprtokodne programske opreme. Odprtokodna programska oprema ima namreč v redkih primerih določen proračun za oglaševanje in marketing, kar ne koristi prepoznavnosti programske opreme na trgu. Mediji pogosto poročajo o vodilnih odprtokodnih produktih, kot sta na primer Linux in Apache, medtem ko je o manj razširjenih odprtokodnih aplikacijah malo govora. Za odprtokodno programsko opremo se moramo torej poslužiti razpoložljivih internetnih virov, pri čemer obstaja več spletnih strani, ki se ukvarjajo s predstavitvijo odprtokodnih programskih rešitev. Dve izmed največjih se nahajata na spletnih naslovih: <http://freshmeat.net/> in <http://sourceforge.net/> (Tuma, 2005, str. 2).

Uporaba odprte kode vse bolj postaja sestavni del razvoja programske opreme. Vse več posameznikov in podjetij se sprašuje, na kakšen način se lahko vključijo v odprtokodno skupnost. Sodelujejo lahko na več načinov. Podjetja, ki uporabljajo odprtokodne razvojne modele, so pogosto v stiku z odprtokodno skupnostjo. Izdane programske rešitve so rezultat sodelovanja. Bolj enostavna oblika sodelovanja in prispevanja je zagotavljanje povratnih informacij in odpravljanje napak v odprtokodnih projektih. Z odkrivanjem in odpravo napak zelo koristimo celotni skupnosti. Z objavo popravkov damo razvijalcem možnost, da si ogledajo nove rešitve in se iz tega nekaj naučijo.

Velika prednost pri razvoju odprtokodnih programskih rešitev je količina znanja, ki nam je na voljo v obliki ljudi, ki so pripravljene sodelovati pri razvoju odprtokodne programske opreme. Pri razvoju lastniških izdelkov v najboljšem primeru sodeluje nekaj skupin programerjev, medtem ko pri razvijanju odprtokodnih programskih rešitev sodeluje veliko večje število ljudi. Potrebno je izpostaviti kvaliteto kadra, ki sodeluje na določenem projektu. V splošnem, bi lahko ocenili, da je pri razvoju lastniške programske opreme kvaliteta kadra višja. Zaposleni se s programiranjem ukvarjajo poklicno, deležni so izobraževanj in so prisiljeni slediti novostim, če želijo ohraniti visoko ceno, medtem ko se sodelovanja na odprtokodnih projektov veliko programerjev poslužuje ljubiteljsko. Treba pa se je zavedati dejstva, da se v velikem številu programerjev, ki se s tem ukvarjajo zaradi svojega veselja, najdemo tudi določen odstotek tistih, ki po svojem znanju prekašajo kolege, ki se z razvojem ukvarjajo profesionalno. Večje število ljudi nam prav tako koristi pri testiranju programske opreme in odpravi napak. Zaradi dostopa do izvorne kode lahko napake odpravijo in nam pošljejo predlagane rešitve. V tej točki se srečamo z novim izzivom. Vsaka predlagana rešitev ni primerna, prav tako je vprašljiva njena kvaliteta. Zagotovljen moramo imeti ustrezen kader, ki

lahko strokovno presodi kvaliteto popravkov ali izboljšav. Podoben način dela lahko uporabimo tudi pri razvoju. Vsak nov del programa ali popravek, more prestati proces preverjanja. Le na ta način lahko sledimo spremembam in ohranjamo začrtano raven kakovosti.

Podpoglavje o izzivih in priložnostih odprtokodne programske opreme zaključimo s priporočili za podjetja in razvijalce. Podjetja morajo medtem ko sprejemajo nakupno odločitev, dobro poznati svoje informacijske potrebe. Temeljito je potrebno preveriti vse možne alternative in na podlagi analize izbrati primerno programsko rešitev. Pri uporabi odprte kode morajo podjetja nameniti pozornost tudi pravnim vprašanjem. Prav tako je potrebno paziti, da se za uporabo odprtokodne programske opreme ne odločimo zgolj zato, ker jo uporabljajo tudi drugi oz. ker je popularno. Upoštevati je potrebno tudi znanje uporabnikov, ki bodo s programsko rešitvijo upravljali. Za odprtokodno programsko opremo je namreč značilno, da zahteva nekoliko več znanja uporabnikov.

Pri razvoju programskih rešitev je uporaba odprte kode smiselna v primerih, ko moduli določene programske rešitve ne predstavljajo novosti in dodane vrednosti. Z uporabo že obstoječe kode skrajšamo čas razvoja in na ta način zmanjšamo stroške. Razvijalci imajo od odprte kode lahko velike koristi. Zaradi dostopa do izvorne kode namreč lahko preverijo, na kakšen način stvari delujejo in na ta način izpopolnjujejo svoje znanje. Odprtokodni projekti so prave zakladnice programerskega znanja. Združijo razvijalce iz celega sveta in spodbujajo delitev znanja in uveljavljenih praks pri razvoju programske opreme.

Sklep

Koncept odprte kode ja na trgu programske opreme prisoten že več desetletij. Večjo veljavo in prepoznavnost je odprtokodna programska oprema dosegla z razvoju širokopasovnega interneta. Razlog, zakaj je odprta koda zares zaživela šele ob razširitvi interneta je preprost. Internet nam danes omogoča komunikacijo in sodelovanje z ljudmi iz celega sveta. Pretok informacij in znanja še nikoli v človeški zgodovini ni bil hitrejši, kot danes. Internet ni postal le učinkovito komunikacijsko sredstvo, temveč z leti postaja velika zakladnica znanja. Danes pri odprtokodnih projektih lahko sodelujejo ljudje iz celega sveta, prav tako internet pomaga pri prepoznavnosti in redistribuciji odprtokodnih programskih rešitev.

Vsa podjetja, ki tekmujejo na razvitih trgih se zavedajo potrebe po optimizaciji poslovanja. V iskanju rasti in višje donosnosti so prisiljena neprestano iskati nove alternative, ki jim omogočajo nižje stroške, višje prihodke in jim zagotavljajo dobro izhodišče v konkurenčnem boju. V današnjem času kvalitetni izdelka ni več garancija za uspešno poslovanje. Na razvitih trgih je večina proizvodov visoke kakovosti, zato se kupci odločajo na podlagi dodatnih lastnosti. Na koncu lahko prevesi tehtnico na določeno stran cena izdelka ali dodatne storitve povezane z nakupom. V vsakem primeru podjetju pomaga, če proizvaja z minimalnimi stroški. Z uporabo odprtokodne programske opreme si lahko poslovanje izboljšajo vsa podjetja, tako tista, ki se ukvarjajo z razvojem programske opreme, kot tista, ki jo le uporabljajo.

Pomemben dejavnik za razvojna podjetja, ki uporabljajo odprto kodo, predstavlja pravilno izbran poslovni model. Uporabniki danes programsko opremo dojemajo drugače, kot pred leti. Veliko vrednost se pripisuje dodatnim storitvam, ki jih nudijo podjetja. Za dodatne storitve so uporabniki pripravljeni plačati, hkrati pa le-te predstavljajo dodano vrednost v očeh kupca. Podjetja, ki se osredotočijo na zagotavljanje teh storitev, večino dohodkov ustvarijo na račun dejavnosti, ki niso strogo povezane s prodajo programske opreme in na ta način ustvarijo dodaten vir prihodkov ter zmanjšajo cikličnost prihodkov, s katero se srečamo pri tradicionalnih pristopih trženja programske opreme. Vse več uporabnikov gleda na programsko opremo kot storitev. Možnost je privlačna predvsem zaradi dejstva, da programske opreme ni potrebno kupiti ampak jo lahko najamemo. Na ta način dosežemo večjo fleksibilnost in lahko hitreje zamenjamo ponudnika v primeru, ko bi bilo to potrebno.

Za uspešno poslovanje je eden izmed ključnih dejavnikov pravilno izbrana strategija. Predstavljene strategije lahko koristijo vsem podjetjem, ki se pri svojem poslovanju srečujejo z uporabo odprte kode. Razvojna podjetja si z uporabo odprte kode znižujejo razvojne stroške in tako povečujejo svojo konkurenčnost na trgu. Odprta koda jim omogoča, da več virov namenijo razvoju tistih delov programske opreme, ki njihovo programsko opremo ločujejo od konkurenčnih rešitev in uporabnikom predstavlja dodano vrednost. Drugim podjetjem uporaba odprte kode pomaga predvsem pri nižanju stroškov programske in strojne opreme. Z nižanjem stroškov poslovanja se podjetjem povečuje konkurenčnost.

Odprta koda se srečuje z veliko izzivi in priložnostmi. Njena uporaba se z leti povečuje. Dva izmed glavnih izzivov sta zagotovitev večje prepoznavnosti in razrešitev vprašanj glede varnosti. V odprti kodi se skriva velik potencial. Pri projektih lahko sodelujejo razvijalci iz celega sveta. Izziv predstavlja izraba znanja, ki je na voljo. Odprtokodni projekti so zahtevni predvsem iz stališča organizacije in koordiniranja virov ter zagotavljanja kakovosti. Po mnenju mnogih odprta koda predstavlja prihodnost programske opreme. Razvoj dogodkov v prihajajočih desetletjih bo pokazal, kdo ima prav.

Pred začetkom raziskovanja predstavljene tematike, smo si zadali cilj. Želeli smo ugotoviti ali lahko uporaba odprte kode podjetjem prinese konkurenčno prednost, pri čemer smo vzporedno s tem, želeli predstaviti tudi različne poslovne modele in strategije, ki jih imajo podjetja pri uporabi odprte kode na voljo. Cilj je bil dosežen. Odprta koda lahko predstavlja ključni dejavnik pri optimizaciji poslovanja in nam s svojimi prednostmi omogoči boljši konkurenčni položaj in konkurenčno prednost pred tekmeci.

Literatura in viri

1. Coppola, C. & Neelley, E. (2004). *Open source – opens learning: Why open source makes sense for education*. Najdeno 5. novembra 2009 na spletnem naslovu <http://www.rsmart.com/assets/OpenSourceOpensLearningJuly2004.pdf>
2. Dimovski, V., Penger, S. & Žnidaršič, J. (2003). *Sodobni management*. Ljubljana: Ekonomska fakulteta.
3. Forfas. (2006, julij). *Open Source trends and business models. Guidelines for key irish stakeholders*. Najdeno 5. novembra 2009 na spletnem naslovu http://www.forfas.ie/media/forfas060724_open_source_trends.pdf
4. Glavač, Z. (2009). *Računalništvo v oblaku in virtualizacija* (diplomsko delo). Maribor: Fakulteta za elektrotehniko, računalništvo in informatiko
5. *Global software top 100 – interim update december 2009*. Najdeno 18. januarja 2009 na spletnem naslovu <http://www.softwaretop100.org/software-top-100/global-software-top-100-interim-update-december-2009>
6. Hiong, G. (2005). *Open source and commercial software*. Najdeno 5. novembra 2009 na spletnem naslovu <http://w3.bsa.org/asia-eng/policy/upload/OS-Commercial-Analysis.pdf>
7. Independent software vendors. (2009). *Open source as a competitive advantage*. Najdeno 5. novembra 2009 na spletnem naslovu <http://www.navicasoft.com/Newsletters/ISV%20Open%20Source.pdf>
8. Industry expert group. (2009, julij). *Report of an industry expert group on a european software strategy*. Playinf to win in the new software market. Version 3.5. Najdeno 15. novembra 2009 na spletnem naslovu ftp://ftp.cordis.europa.eu/pub/fp7/ict/docs/ssai/European_Software_Strategy.pdf
9. Keber, B. (2006). *Obvladovanje poslovnih procesov z uporabo ogrodij in storitveno usmerjene arhitekture*. Najdeno 20. maja 2010 na spletnem naslovu http://miha.ef.uni-lj.si/_dokumenti3plus2/191060/CLANEK6--obvladovanje1.doc
10. Koenig, J. (2004). *Open source business strategies*. Najdeno 5. novembra 2009 na spletnem naslovu <http://kb.cospa-project.org/retrieve/1537/Opensourcebusinessmodels.pdf>
11. Krisper, M. & Lavbič, D. (2006). *Semantične spletne storitve*. Najdeno 2. februarja 2010 na spletnem naslovu [http://amor.fri.uni-lj.si/dejan/delo/bibliografija/\(2006,%20DSI\)%20Semanticne%20spletne%20storitve%20\(Dejan%20Lavbic\).pdf](http://amor.fri.uni-lj.si/dejan/delo/bibliografija/(2006,%20DSI)%20Semanticne%20spletne%20storitve%20(Dejan%20Lavbic).pdf)
12. Marc, M. (2003). *Analiziranje poslovanja podjetja v pogojih elektronskega poslovanja* (magistrsko delo). Ljubljana: Ekonomska fakulteta
13. Olliance group. (2010). *Open source consulting services and solutions*. Najdeno 20. januarja 2010 na spletnem naslovu <http://www.olliancegroup.com/services/index.php>
14. Open source initiative. *The open source defenition*. Najdeno 5. novembra 2009 na

spletnem naslovu <http://opensource.org/docs/osd>

15. Primožič, P. (2005). *Uporaba odprte kode kot osnova za razvoj programske opreme* (diplomsko delo). Ljubljana: Fakulteta za računalništvo in informatiko
16. *Računalništvo v oblaku*. (2009). Najdeno 20. marca 2010 na spletnem naslovu <http://www.aktivno.si/ai/sl/734-racunalnistvo-v->
17. Red Hat. (2010). *Value of a Red Hat subscription*. Najdeno 20. januarja 2010 na spletnem naslovu <http://www.redhat.com/software/subscriptions.html>
18. *Software as a service (SaaS) defenition and solutions*. Najdeno 9. februarja 2010 na spletnem naslovu http://www.cio.com/article/109704/Software_as_a_Service_SaaS_Definition_and_Solutions
19. *Storitveno naravnana arhitektura (SOA)*. Najdeno 28. marca 2010 na spletnem naslovu <http://www.smart-com.si/?m=65&idp=73&n=43>
20. Tuma, D. (2005, januar). *Open source software: Opportunities and challenges*. Najdeno 29. marca 2010 na spletnem naslovu <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.114.1104&rep=rep1&type=pdf>
21. *Web 3.0 – Semantični splet*. Najdeno 25. marca 2010 na spletnem naslovu <http://blog.morphix.si/2007/uporabniske-izkusnje/web-30-%E2%80%93-semanticni-splet/>