

UNIVERZA V LJUBLJANI
EKONOMSKA FAKULTETA

MAGISTRSKO DELO

**ODPR TOKODNE PROGRAMSKE REŠITVE
V POSLOVNIH MODELIH PODJETIJ**

Ljubljana, september 2011

MARKO KONIČ

IZJAVA

Študent Marko Konič izjavljam, da sem avtor tega magistrskega dela, ki sem ga napisal v soglasju s svetovalcem prof. dr. Tomažem Turkom, in da v skladu s 1. odstavkom 21. člena Zakona o avtorskih in sorodnih pravicah dovolim njegovo objavo na fakultetnih spletnih straneh.

V Ljubljani, dne 29. 9. 2011

Podpis: _____

KAZALO

UVOD.....	1
1 FENOMEN ODPORTE KODE	8
1.1 Zgodovina odprtokodnih programskih rešitev	8
1.2 Odprtokodni programi	11
1.3 Odprtokodne skupnosti	13
1.4 Značilnosti odprtokodnih projektov in razvoj	15
1.4.1 Razvojni proces	16
1.4.2 Vodenje in koordiniranje	18
1.5 Motivacija odprtokodnih programskih razvijalcev	20
1.6 Motivacija podjetij	21
1.7 Različni pristopi v odnosu podjetje – skupnost	25
1.8 Izzivi v odnosu podjetje – skupnost.....	26
2 ODPRTOKODNI PRISTOPI V PODJETJIH.....	27
2.1 Notranji odprtokodni pristopi	28
2.1.1 Kdaj je odprtokodni notranji pristop primeren.....	29
2.1.2 Odprtokodne programske rešitve ali komercialni programi.....	29
2.1.3 Odprtokodne programske rešitve za poslovne namene	32
2.1.4 Uporaba odprtokodnih programskih rešitev v organizacijah	33
2.2 Zunanji odprtokodni pristopi	34
2.2.1 Kdaj je odprtokodni zunanji pristop primeren.....	35
2.2.2 Koristi od odprtokodnega zunanjega pristopa	36
3 POSLOVNI MODELI	37
3.1 Koncept poslovnih modelov	37
3.2 Struktura poslovnih modelov	40
3.3 Poslovni modeli, ki za osnovo uporabljajo odprtokodne programske rešitve	47
3.4 Struktura poslovnega modela, uporabljenega v nalogi	49
3.4.1 Ponudba	50
3.4.2 Viri.....	51
3.4.3 Odnosi.....	51
3.4.4 Dohodkovni model	52

3.5	Ocena različnih modelov in primerjava med njimi	52
4	ANALIZA UPORABNOSTI ODPRTOKODNIH PROGRAMSKIH REŠITEV V PODJETJIH	54
4.1	Utemeljitev strukture analize	54
4.2	Odprtokodni poslovni modeli podjetij	58
4.2.1	Hit d.d.	58
4.2.2	Typo3	60
4.2.3	Netgem	63
4.2.4	IBM	65
4.2.5	Nokia Symbian	68
4.2.6	Iskra Avtoelektrika d.d.	71
4.2.7	SugarCRM	74
4.2.8	Bimeli d.o.o.	76
4.3	Raziskovalni model	78
4.4	Raziskovalno vprašanje	78
4.5	Rezultati in analiza podatkov	78
5	SKLEP	86
6	LITERATURA IN VIRI	89
7	PRILOGE	

KAZALO SLIK

Slika 1:	Grafični prikaz čebulne strukture odprtokodne programske skupnosti	15
Slika 2:	Združevanje strategij, poslovnih modelov in procesnih modelov na različnih nivojih	38
Slika 3:	Grafični prikaz poslovnega modela, ki ga predlagata Hedman in Kalling	41
Slika 4:	Grafični prikaz komponent poslovnega modela po Osterwalderju	42
Slika 5:	Grafični prikaz koncepta poslovnega modela po Hamelu	44
Slika 6:	Grafični prikaz poslovnega modela, ki temelji na odprtokodnih programskih rešitvah	80

KAZALO TABEL

Tabela 1: Mejniki pri soustvarjanju odprtokodnih rešitev	9
Tabela 2: Taksonomija motivacij posameznih programerskih razvijalcev	21
Tabela 3: Taksonomija motivacij podjetij	24
Tabela 4: Stebri in komponente poslovnega modela po Osterwalderju.	41
Tabela 5: Komponente poslovnega modela po Hamelu.	43
Tabela 6: Elementi poslovnega modela po Altu in Zimmermannu	44
Tabela 7: Elementi poslovnega modela po Afuahu in Tucciju.	46
Tabela 8: Primerjava med različnimi poslovnimi modeli.	53
Tabela 9: Odprtokodni pristopi v podjetjih	79

UVOD

Problematika magistrskega dela. Podjetja, ki želijo preživeti v današnjem dinamičnem in spreminjajočem se okolju, so zavezana k uporabi najrazličnejših strategij, taktik in orodij, ki jim pomagajo pri doseganju konkurenčne prednosti. Pregled ustreznih virov (Tapscott; 2006, Stallman, 2008; Bonaccorsi&Rossi, 2003; Hecker, 1999; Wichmann, 2002) kaže, da postaja odprta koda vse pomembnejša in uporabnejša možnost za zagotavljanje potrebnih programskih rešitev in informacijskih storitev. V gibanje se vključuje tudi vse več javnih uprav in njihovih institucij. Vpliv odprtokodnih programskih rešitev ter modeli uporabe in razvoja prinašajo veliko sprememb v informacijsko tehnologijo (v nadaljevanju IT), v industrijo in s tem v celotni poslovni sektor. Programska oprema, ki deluje na osnovi odprtokodnih programskih rešitev, je že začela spreminjati pravila na trgih informacijske tehnologije, kar kaže na to, da bodo tista podjetja, ki bodo prva pri uporabi te tehnologije, imela določene konkurenčne prednosti.

Odprtokodne programske rešitve definirajo trije ključni sestavni deli (Open Source Initiative, 2010):

Razvojna metodologija za razvoj programskih rešitev uporablja pristop interaktivnih skupnosti in medsebojnega pregledovanja. Odprtokodna programska oprema se razvija v skupnostih, ki jih tvorijo posamezniki s celega sveta. Ti se med seboj ne poznajo, vendar jih združuje zavezanost k istemu cilju. Taki projektni skupini učinkovito uspeva opravljati tako težko nalogo, kot je izgradnja visokokakovostne programske opreme.

Licenciranje zagotavlja prost dostop do izvorne kode v skladu z eno od številnih licenc, ki jih avtorizira OSI¹ (Open Source Initiative, 2010). Za vse licence (BSD², GPL³, LGPL⁴, MPL⁵) velja, da omogočajo uporabnikom ogled in spreminjanje izvorne kode in natančno specificirajo pogoje spreminjanja in razširjanja kode, saj se ta vedno vrne skupnosti odprte kode.

Odprtokodna skupnost, ki jo sestavljajo razvijalci, uporabniki in ponudniki programskih rešitev, skrbi za vzdrževanje programske opreme. Skupnost uporabnikov pa ni namenjena samo podajanju predlogov za izboljšave in poročanju o napakah v programski opremi. Njen namen je predvsem pomagati odstranjevati napake v programski opremi ter izboljševati, nadgrajevati in razširjati programsko kodo. Odprtokodne skupnosti so danes eden najbolj uspešnih in hkrati najslabše razumljenih primerov visokokakovostnega in učinkovitega sodelovanja na internetu. Druge skupnosti bi lahko ogromno pridobile z razumevanjem delovanja odprtokodnih skupnosti. Značilnosti odprtokodnih programskih

¹ Open Source Initiative

² Berkeley Software Distribution

³ General Public Licence

⁴ Library General Public Licence

⁵ Mozilla Public Licence

rešitev so: svobodnost, dostopnost izvorne kode programa, možnost prilagajanja in nadaljnja distribucija spremenjene izvorne kode, podpora odprtih standardov in protokolov ter hitro rastoča množica uporabnikov in s tem posredno tudi razvijalcev. Predpogoj za uresničevanje ideje zagovornikov odprtokodnih programskih rešitev in povečanje njihove popularnosti je bil prav gotovo pojav interneta. Z enostavnim dostopom do spleta pa se je možnost deljenja programov in skupnega razvoja programske opreme razvijalcev s celega sveta približala široki množici ljudi. Od takrat naprej imajo odprtokodne programske rešitve vse opaznejšo vlogo v svetovni informacijski industriji. Kljub temu danes še vedno mnogi ne sprejemajo odprtokodnih programskih rešitev, ker naj bi šlo za slabo napisane programe, ki jih uporabljajo hekerji in izkoriščevalci brezplačnih programov. Pri tem prednjačijo predvsem velike multinacionalke, za katere predstavljajo odprtokodne programske rešitve velikega konkurenta. Na drugi strani pa med velikimi multinacionalkami obstajajo tudi izjeme, ki so v odprtokodnih programskih rešitvah videle svojo priložnost. Tako je na primer IBM v odprtokodne programske rešitve (predvsem v operacijski sistem Linux) vložil precej svoje tehnologije, z namenom da bi se ta uporabljala širše, s čemer bi popularizirali druge svoje izdelke, ki uporabljajo enako tehnologijo (Open Source Initiative, 2010).

Večina odprtokodnih programskih rešitev je na internetu na voljo brezplačno. Nizki stroški nakupa so še vedno očitni razlog za izbiro odprtokodnih programov, a se jih kljub temu veliko podjetij odloči uporabiti za izboljšanje storitev (Madanmohan & De'Rahul, 2004, str. 66). Na voljo je veliko odprtokodnih produktov, ki predstavljajo alternativo komercialnim, tudi v poslovne namene. Tako se je veliko družb odločilo za njihovo uporabo (na primer Windows XP alternativa Linux; Microsoft Office alternativa Open office; .Net alternativa php; Sql alternativa Mysql) (Wichmann, 2002).

Odprtokodne programske rešitve, spletne storitve, dnevniki, wiki, internet 2.0 in druge tehnologije prihajajo v podjetja in zato se menedžerji IT pripravljajo na povsem nove oblike poslovanja. Številni menedžerji IT to primerjajo z vremenom (Tapscott, 2007), češ da vsi razpravljajo o tem, v resnici pa nihče ničesar ne ve. Takšno mnenje je zmotno, ker novi internet zares prinaša za menedžerje IT, za same informacijske tehnologije in za podjetja nove modele z daljnosežnimi implikacijami. Nedvomno se večina odprtokodno nastalih spletnih dnevnikov, wikijev, klepetalnic in »osebne oddajanja« že danes uporablja za medsebojne pogovore, sodelovanja in razpravljanja. Tisto, kar prihaja in bo kmalu tu, pa je nič manj kot nov način poslovanja. Zahvaljujoč novemu spletu, podjetja začnejo razumevati, oblikovati, razvijati in distribuirati tako izdelke kot storitve na povsem nov način. Mnenje, da mora podjetje privabljati, razvijati in obdržati najboljše in najpametnejše znotraj svojih meja, postaja zastarelo. S strmim upadanjem stroškov sodelovanja podjetja zdaj lahko uporabljajo ideje, inovacije in enkratna mišljenja iz velikanskega globalnega rudnika talentov in tako ustvarjajo odprtokodne programske rešitve (Tapscott, 2007).

Enciklopedije (Wikipedija), operacijski sistemi (Linux), internetni brskalnik (FireFox, Chrome), pisarniški programi (openoffice), podatkovne baze (MySQL), oblikovanje športnih srajc (threadless.com) in podobni kompleksni izdelki nastajajo z uporabo odprtokodnih programskih rešitev s sodelovanjem na tisoče, celo milijone ljudi. Medtem ko se posamezni menedžerji IT bojijo rasti množičnih spletnih skupnosti, ki jih povezujejo odprtokodne programske rešitve, pametni vodje opravljajo tisto, kar Tapscott imenuje wikinomija (Wikinomics) ali odprtokodni način ustvarjanja programskih rešitev, da bi izkoristili te kolektivne sposobnosti in talente ter spodbudili inovacije, rast in uspeh. Bolj kot podjetja vidijo koristi od odprtokodnega množičnega sodelovanja, tem bolj postaja verjetno, da bo ta novi način organiziranja dela končno nadomestil tradicionalne (z)družbe kot primarne ekonomske mehanizme ustvarjanja bogastva (Tapscott & Williams, 2006, str. 2).

Najbolj znan je primer kanadskega podjetja GoldCorp, ki je bilo tik pred propadom, ker geologi niso znali natančno določiti, kolikšne so zaloge zlata v njihovem rudniku. Vodstvo se je odločilo geološke podatke o rudniku razkriti svetu in je ponudilo 575.000 dolarjev nagrade za najboljše metode in ocene. Rezultat je bil izjemen, saj se je na natečaj prijavilo veliko število strokovnjakov in podjetje je identificiralo 50 lokacij za izkop. Podjetje je imelo takrat letni promet 100 milijonov dolarjev, danes pa ga ima več kot 8 milijard dolarjev. Iz primera je razvidno, da tudi neračunalniška podjetja lahko koristno izkoristijo odprtokodno sodelovanje (Tapscott & Williams, 2006, str. 7).

Googlu je z odprtokodnimi programskimi rešitvami uspelo postati podjetje z najbolj cenjeno blagovno znamko na svetu. Sodeč po raziskavi družbe Millward Brown Optimor, je Google po vrednosti tržne znamke na prvem mestu, s skupno ocenjeno vrednostjo 86,1 milijarde dolarjev (Google Grabs No. 1 Brand Rank, 2008).

Prilagodljivost informacijskega sistema in aplikacij, ki jih podjetja uporabljajo za podporo poslovanju, je pomemben dejavnik prilagodljivosti podjetja. Med splošno sprejete načine za doseganje tega cilja spadajo rešitve, ki temeljijo na intenzivni rabi spletnih storitev in odprtokodnih standardov. Tudi storitveno usmerjena arhitektura (SOA) uporablja koncept poslovne logike aplikacij, ki jih je treba v ta namen izpostaviti v obliki spletnih storitev (Web Services). Torej ni dvoma, da odprtokodne programske rešitve, spletne storitve in storitveno usmerjena arhitektura pomenijo naslednjo veliko spremembo v svetu računalništva in informatike. Gre za nadgradnjo v smeri celostnega obvladovanja življenjskega kroga poslovnih storitev, in ne samo obvladovanja posameznih procesov, potrebnih za njihovo dobavo in podporo. Google uporablja tehnologijo spletnih storitev že nekaj časa. V svoje aplikacije si lahko z njegovo pomočjo integriramo množico uporabnih pripomočkov, združenih v skupino Google Web APIs, na primer Google Search, Maps, Calendar in druge, ki so brezplačni. S tem dokazujejo, da so danes spletne storitve

enostavne, varne in poceni rešitve za probleme, ki so se še do nedavnega zdeli nepremostljivi (Google Web APIs, 2008).

Tapscottova in Williamsova (2006, str. 19) ugotovitev je, da v kapitalizmu osnovni način konkuriranja ni cena; podjetje lahko pridobi konkurenčno prednost predvsem z uvajanjem novih izdelkov, tehnologij, oskrbnih virov in novih načinov organizacije. Zato mora skrbeti za nenehno spreminjanje notranjih ekonomskih struktur in starih načinov poslovanja. Najbolj znan primer spletne trgovine v Združenih državah Amerike (v nadaljevanju ZDA) je gotovo Amazon, katere poslovni model temelji predvsem na kakovosti storitev. Amazon je tako s pomočjo odprtokodnih programskih rešitev Perl⁶, Linux⁷, Apache⁸ ne samo odkrila vrzel na tržišču, ampak tudi prisilila večje etabrirane knjigarne z dolgoletno tradicijo, kot je na primer Barnes & Noble, v spreminjanje poslovnih modelov. Sodobni avtorji se tako v nekaterih primerih odpovedujejo klasičnemu načinu publiciranja v tiskani obliki in tako ne potrebujejo ne publicistov ne posrednikov, ki bi zagotovili, da njihov material prispe do končnega potrošnika (Volovšek, 2001, str. 3).

Poslovni modeli se lahko implementirajo na različne načine. Podjetje lahko vključi več takšnih modelov v svojo poslovno strategijo. Poslovni modeli, ki temeljijo na odprtokodnih programskih rešitvah, se rojevajo hitro, nove in zanimive različice se bodo pojavljale tudi v prihodnosti. Struktura poslovnih modelov, ki temeljijo na uporabi odprtokodnih programskih rešitev, pa mora upoštevati osnovne značilnosti odprtkodnega poslovanja. V tem kontekstu je treba izpostaviti skupnosti, ki v poslovni mreži podjetja oblikujejo popolnoma nove interesne skupine in odnose do njih. Zaradi tega morajo podjetja svoje interese uskladiti z interesi skupnosti, saj le tako lahko oblikujejo koristne vzajemne odnose. Za strukturo poslovnega modela, ki temelji na uporabi odprtokodnih programskih rešitev, je pomemben tudi dohodkovni model, ki naj bi razkril, kako podjetje pridobiva denar.

Namen in cilj magistrskega dela. Namen naloge je izdelati strukturo za opis poslovnih modelov, ki so se do danes razvili v podjetjih.

Cilji magistrskega dela so ugotoviti:

- kako podjetja uporabljajo odprtokodne programske produkte,
- kako podjetja uporabljajo razvojno prakso,
- kakšni poslovni modeli obstajajo v podjetjih, ki poslujejo z odprto kodo.

Na podlagi namena in ciljev smo oblikovali raziskovalno vprašanje:

⁶ Odprtokodni programski jezik

⁷ Odprtokodni operacijski sistem

⁸ Odprtokodni spletni strežnik

Kakšni poslovni modeli obstajajo v podjetjih, ki temeljijo na odprtokodnih programskih rešitvah. Zelo malo je napisano o odprtokodnih programskih rešitvah v poslovnih modelih, a je raziskovanje na to temo zelo pomembno, saj so podjetja odprtokodne programske rešitve v veliki meri že sprejela. Prisotne so v številnih sodobnih informacijskih sistemih, s pomočjo katerih podjetja ne delujejo več na stacionarnih računalnikih, temveč selijo poslovanje na internet. Internet preobraža naravo in sam proces dela, vzpodbuja interaktivnost in distribuiranost. Mrežna organiziranost postaja vse bolj učinkovita, še posebej z naraščanjem hitrosti v telekomunikacijah. Pri odprtokodnih programskih rešitvah je oblika produkcije mrežne organiziranosti enako pomembna kot koda sama, saj gre za kreativni proces. Odprtokodne programske rešitve so vseobsežen družbeni koncept, ki ni omejen na področje programske opreme, ampak je uporaben za produkcijo in distribucijo znanja v najrazličnejših domenah.

V sodobnem poslovanju je čedalje več neposrednega poslovanja, ki izzove odzive potencialnih kupcev na porabniških in medorganizacijskih trgih. Temeljna podlaga za neposredno poslovanje je baza podatkov odjemalcev, ki omogoča poslovanje preko elektronskih poti. Sodobno poslovanje si lahko predstavljamo kot novo poslovno usmeritev novih spremenjenih okoliščin, ki jim botrujejo odprtokodne programske rešitve, spletne storitve, storitveno usmerjena arhitektura, dnevniki, wiki, internet 2.0 in druge tehnologije. Te tehnologije so ponovno odkrile tudi nekatere obstoječe, a manj priljubljene modele prodajnega procesa, kot so na primer dražbe, ki so se uporabljale v glavnem za cenitev kmetijskih pridelkov, finančnih instrumentov in unikatnih umetniških in starinskih predmetov.

S tem pa je povezano dejstvo, da v novih okoliščinah klasični poslovni modeli ne zadoščajo več. Nastalo je precej poslovnih modelov, ki so specifični za nove razmere. Podjetja za izkoriščanje novih poslovnih priložnosti in ustvarjanje vrednosti uporabljajo nove poslovne modele in dinamične strategije, ki delujejo predvsem na treh področjih: povečanju prihodkov, zmanjšanju stroškov in izkoriščanju nemerljivih sredstev. Menjajo se odnosi, kar pomeni nove zahteve od vseh zaposlenih. Te vloge predstavljamo v osrednjem poglavju o poslovnih modelih.

Uporabnost odprtokodnih programskih rešitev v poslovnih modelih se razlikuje od podjetja do podjetja. V nalogi bomo to opredelili s kvalitativno raziskavo uporabe odprtokodnih programskih rešitev v poslovnih modelih podjetij. Ustvariti želimo pregled uporabljenih struktur poslovnih modelov in ugotoviti njihove podobnosti in razlike. Predpostavljamo, da odprtokodnih programskih rešitev v poslovnih modelih vsa podjetja ne uporabljajo v enaki meri in na enak način.

Metode dela. Ugotovitve številnih raziskovalcev (Tapscott, 2007; Stallman, 2008; Bonaccorsi & Rossi, 2003; Morris & Allen, 2005; Rappa, 2004; Hecker, 1999), ki se

ukvarjajo z odprtokodnimi programskimi rešitvami, spletnimi storitvami, internetom 2.0, poslovnimi modeli, storitveno usmerjeno arhitekturo, novo ekonomijo⁹ in drugimi tehnologijami, bomo uporabili kot osnovo za ugotavljanje dejavnikov, ki pomembno vplivajo na prihodnost poslovnih modelov, ki temeljijo na odprtokodnih programskih rešitvah. Za proučevanje bomo uporabili kvalitativno raziskovanje, s katerim bomo ustvarili pregled različnih struktur poslovnih modelov, ki temeljijo na odprtokodnih programskih rešitvah. Vprašalnik je prilagojen podjetjem, ki so odprtokodne programske rešitve integrirale v poslovanje, in tistim, ki odprtokodne programske rešitve prodajajo. Anketiranci bodo izbrani zaposleni s širokim znanjem o poslovanju podjetja. Podatke bomo pridobili s pomočjo elektronske pošte. Sestavljen je iz petih sklopov vprašanj o ponudbi, virih, odnosih in dohodkovnem modelu, s katerimi bomo sestavili strukturo vsakega posameznega poslovnega modela podjetja.

Pregled vsebine magistrskega dela. V uvodu bomo na kratko predstavili področje raziskovanja, problem, namen, cilje, metodologijo in strukturo magistrskega dela.

V drugem poglavju bomo predstavili zgodovino odprte kode, njen razvoj skozi čas in kdo so bili njeni pobudniki in utemeljitelji. Nadaljevali bomo z definicijo odprte kode in glavnimi značilnostmi najbolj popularnih licenc (zakonitih dokumentov), na podlagi katerih je grajeno prosto oziroma odprtokodno programje. Nato se bomo osredotočili na odprtokodne skupnosti, v katerih se razvija prosto programje. V okviru tega poglavja bomo odgovorili na vprašanja, kdo so ljudje, ki razvijajo prosto programje, zakaj to počnejo oziroma kakšni so njihovi motivi. Sledili bodo odprtokodni projekti, kjer bomo predstavili razvojni model odprtokodnih projektov, razložili bomo, kdo so uporabniki prostega programja in kakšne so lahko njihove vloge pri njegovem razvoju. Prav tako bomo predstavili tipičen razvojni cikel odprtokodnih projektov. Nadaljevali bomo z motivacijo posameznih razvijalcev in podjetij pri uporabi odprtokodnih programskih rešitev. Poglavje bomo zaključili z razumevanjem različnih pristopov in izzivov v odnosih podjetja do skupnosti.

V tretjem poglavju se bomo srečali z odprtokodnimi pristopi podjetij. Predstavili bomo dva različna pristopa pri uporabi odprtokodnih produktov, odprtokodnih praks in odprtokodnega fenomena v širšem pomenu besede. Pri prvem pristopu podjetja uporabljajo obstoječe odprtokodne programske rešitve in komponente, ki jih integrirajo kot del produkta ali pa jih uporabijo v notranjih poslovnih projektih. Poimenovali smo ga notranji odprtokodni pristop (Fink, 2003, str. 203); ugotavili bomo, kdaj je primeren za uporabo v podjetju ter proučili izbire med odprtokodnimi in komercialnimi programi. Nadaljevali bomo z odprtokodnimi programskimi rešitvami, ki jih podjetja realizirajo v poslovne namene. Drugi pristop zahteva več proaktivnosti na strani podjetja pri razvoju odprtokodnih programskih rešitev, poimenovali smo ga zunanji odprtokodni pristop (Fink, 2003, str. 192).

⁹ Namesto izraza nova ekonomija se pogosto pojavljajo tudi izrazi internetna ekonomija, digitalna ekonomija, mrežna ekonomija in informacijska ekonomija.

Tudi pri tem pristopu bomo ugotovili, kdaj je za podjetje primeren, in proučili koristi, ki jih bomo z njim imeli.

V četrtem poglavju bomo predstavili, kaj je poslovni model in kaj je poslovni model, ki temelji na odprtokodnih programskih rešitvah. V tem poglavju bomo predstavili razumevanje izraza in koncepta poslovnega modela. Predstavitev bo podana po pregledu literature (Hedman & Kalling, 2003; Osterwalder, 2004; Morris & Allen, 2005; Rappa, 2004), vendar dodajamo, da imajo avtorji v literaturi različne poglede na pojem in na sestavne dele »poslovnih modelov«. Prav z določitvijo sestavnih delov lahko poslovne modele uporabimo kot orodje za poslovno načrtovanje, s katerim si menedžerji pomagajo pri razumevanju in opisovanju poslovne logike njihovega podjetja (Osterwalder 2004, str. 14). Poslovni modeli, ki za osnovo uporabljajo odprtokodne programske rešitve, niso še tako razširjeni. Hecker's (1999) in Raymond (2001, str. 63) nam ponujata najboljšo klasifikacijo poslovnih modelov, ki kot osnovo uporabljajo odprtokodne programske rešitve. Teoretični del strukture poslovnega modela, uporabljenega v nalogi, bomo povzeli po različnih avtorjih (Osterwalder, 2004; Morris & Allen, 2005; Rappa, 2004). In ker je treba doseči različnih opise poslovnih modelov, in ne le klasificirati poslovnih modelov v predefinirane opise kot piše Hecker (1999), bomo iz različnih poslovnih modelov sestavili strukturo, ki je primerna za to vrsto študije in temelji na odprtokodnih programskih rešitvah. Tako bomo v nalogi reducirali poslovni model na štiri bistvene elemente: ponudba, viri, odnosi, prihodkovni model, s katerim bomo zajeli celoten pogled poslovnih modelov. V literaturi je kar nekaj različnih poslovnih modelov, katerih struktura vsebuje različne elemente, ki se razlikujejo od avtorja do avtorja, po drugi strani pa so si po določenih elementih zelo podobni. V veliko poslovnih strukturah so elementi preveč na široko obrazloženi z istim pomenom, samo poimenovani so drugače.

V petem poglavju bomo predstavili rezultate raziskave o trenutnem stanju uporabnosti odprtokodnih programskih rešitev v poslovnih modelih podjetij. Z izbrano strukturo poslovnega modela, ki je sestavljena iz 4 elementov: ponudbe, virov, odnosov in dohodkovnega modela, bomo zgradili in opisali poslovne modele podjetij. Za proučevanje bomo uporabili kvalitativno raziskovanje, s katerim bomo ustvarili pregled različnih struktur poslovnih modelov, ki temeljijo na odprtokodnih programskih rešitvah. Vprašalnik bo prilagojen podjetjem, ki so odprtokodne programske rešitve integrirala v poslovanje, in tistim, ki odprtokodne programske rešitve prodajajo. Anketiranci bodo izbrani zaposleni s širokim znanjem o poslovanju podjetja. Podatke bomo pridobili s pomočjo elektronske pošte. Odgovorili bodo na 4 sklope vprašanj o ponudbi, virih, odnosih in dohodkovnem modelu ter s tem sestavili strukturo poslovnega modela podjetja.

1 FENOMEN ODPORTE KODE

Fenomen odprtokodnega gibanja predstavlja izziv za veliko tradicionalnih teorij v ekonomiji, informatiki, poslovni strategiji in pri upravljanju informacijske tehnologije. Razvoj odprtokodnih programskih rešitev so projekti internetnih skupnosti programerjev, ki prostovoljno sodelujejo pri razvoju programskih rešitev. Te so prosto dostopne vsem, saj zagotavljajo prost dostop do njihove izvirne kode. To gibanje je spremenilo osnovno naravo programske industrije (Ghosh, 2005, str. 24). Na strani ponudbe so se zgodile temeljne spremembe v razvoju, mehanizmih nagrajevanja, delegiranju razvojnega dela in v poslovnih modelih, ki prikazujejo, kako je mogoče doseči določen dobiček. Povpraševanje kot nakup in razvoj možnosti, ki so običajno na voljo podjetjem, je bilo dopolnjeno z novimi alternativnimi odprtokodnimi programskimi rešitvami (Fitzgerald, 2006, str. 587).

1.1 Zgodovina odprtokodnih programskih rešitev

Ideja odprte kode ima svoje začetke v 60. letih 20. stoletja in je tesno povezana z zgodovino Unixa¹⁰. Kot izraz pa ga je šele leta 1998 skovala Inicijativa za odprto kodo. Zamisel o odprtokodnih programskih rešitvah ni takoj postala priljubljena. Bruce Perens in Eric Raymond (Raymond, 1998) sta se strinjala, da je del problema povezan s Stallmanom, ki je terminologijo prostega programja razlagal kot brezplačno programje, nevedoč, da bo pri poslovnih ljudeh imelo negativen odziv. Perens in Raymond sta ustanovila OSI¹¹ in začneta gibanje odprtokodnih programskih rešitev, ki se je razlikovalo od FSF-ja¹² predvsem po filozofskih razlogih. Označba odprtokodne programske rešitve je bila ustvarjena, da bi izboljšali sprejemanje odprtokodnih programskih rešitev v računalniških podjetjih. Prednost pojma odprtokodne programske rešitve je, da poslovni svet ni želel uporabljati besedne zveze brezplačne programske rešitve in da so se komercialno usmerjeni programerji želeli oddaljiti od Stallmanove ideološke filozofije v zvezi s prostim programskim gibanjem (Markus, Marville & Agres, 2000, str. 17).

S pojavom interneta v začetku leta 1990 so se začele mrzlično širiti odprtokodne dejavnosti. Število razvijalcev, skupnosti in projektov je naraščalo zelo hitro, tudi sodelovanje med trgovskimi družbami in odprtokodnimi skupnostmi je postala vsakdanjost. Danes odprtokodne programske rešitve ponujajo zanimivo alternativo komercialnim programskim rešitvam, ki jo uporabljajo v zasebne in poslovne namene (Lerner & Tirole, 2000, str. 6).

¹⁰ Operacijski sistem

¹¹ Inicijativa za odprto kodo – Open Source Initiative

¹² Gibanja prostega programja – Free Software Foundation

Tabela 1: Mejniki pri soustvarjanju odprtokodnih rešitev

1983	Richard Stallman začne GNU projekt, ambiciozen poskus za oblikovanje celotnega odprtokodnega operacijskega sistema, ki temelji na Unixu. Richard Stallman je ustvaril GNU General Public License (GPL), ki omogoča vsakomur prenos, spreminjanje, distribucijo in tudi plačilo pristojbine za GNU izvirno kodo, ki jo zahteva proces, »copylefting«. Glavno določilo GNU projektov je, da se vse spremembe izvirnih kod ali vsak novi razvoj programskih rešitev deli z razvijalci skupnosti.
1987	Larry Wall objavi na Usenetu (omrežje) prvo različico programskega jezika Perl, ki je temeljil na Unixovem okolju za skeniranje, manipuliranje in tiskanje tekstovnih datotek. Prvo verzijo objavi pod GPL licenco, vendar Wall meni, da so ti pogoji preveč restriktivni in napiše lastno distribucijo pravil, ki jih imenuje »Artistic License«.
1989	Študent univerze v Helsinkih, Linus Torvalds, napiše alternativo Minixu, Unix varianto, ki bi se uporabljala za poučevanje računalniške znanosti in bi delovala na osebnih računalnikih. Linux je rojen. Podjetje Cygnus Solutions opravlja komercialne prilagoditve in podporo za namestitve GNU orodij.
1992	Torvalds se odloči za »copyleft« Linux v čast dela GNU projekta, ki prispeva k njegovemu operacijskemu sistemu.
1993	Ko so CD-romi pridobili na popularnosti, postanejo tudi brezplačne programske opreme bolj razširjene.
1993	FreeBSD 1.0, program, ki temelji na Unixu, razvitem na različnih točkah AT & T, Novella, in UC Berkeley, se sprostí na Netu (mreži) in na CD-romih. Uporabniške licence za program Perl ne zahtevajo, da razvijalci predložijo svoje spremembe izvirne kode nazaj v skupnost.
1994	Bryan Sparks ustanovi podjetje Caldera. Ustanovitveni kapital dobi od nekdanjega glavnega izvajalskega šefa Novella, Raya Noorda. Družba v zasebni lasti, s 50 zaposlenimi, zavzema Linux in preprodaja različne javne storitve in aplikacij. Perl 5 izdajo z dodatki, ki dajejo Perl programerjem veliko bolj prilagodljiv okvir za dodajanje novih funkcij.
1995	Ekipa programerjev se odloči, vzeti izvirno kodo nacionalnega Centra za računalniške aplikacije spletnega strežnika, ga dopolniti in ponuditi javnosti. Preimenujejo ga v Apache spletni strežnik zaradi popravkov in potrebnih nadgradenj. Izide FreeBSD 2.0.

»se nadaljuje«

»nadaljevanje«

1996	Izvorna koda za Apache je v celoti spremenjena. Apache prehiti NCSA kot najbolj priljubljeni spletni strežnik z 29 % trga. Tako kot velja za GNU projekte in Linux, ekipa programerjev vzdržuje in posodablja jedro izvirne kode. Za Apache licenca ne zahteva, da uporabniki predložijo spremembe izvirne kode nazaj v skupnost.
1997	Prva Perl uporabniška konferenca s cca. 1.000 udeleženci v San Jose.
1998	Microsoft je pod velikim pritiskom zaradi internetnega brskalnika, saj je Netscape Communications napovedal, da bo izvorna koda prihajajočega programa Communicator 5,0 na voljo vsakomur, da si jo bo lahko naložil, spremenil in razmnožil. Tržni delež spletnega strežnika Apache doseže 50 % trga.
1999	OpenOffice je prvotno razvit kot lastniška programska aplikacija StarOffice, ki ga je razvilo nemško podjetje StarDivision, od katerega Sun Microsystems odkupi kodo. 19. julija 2000 Sun Microsystems naznani, da je bila izvorna koda StarOffice na voljo za prenos v skladu z LGPL in z the Sun Industry Standards Source License (SISSL)
2002	MediaWiki je spletni računalniški program tipa wiki, ki je bil razvit za potrebe proste spletne enciklopedije Wikipedije in zdaj predstavlja infrastrukturo vseh spletnih strani v okviru Fundacije Wikimedia, vseh wikijev, ki jih gosti servis Wikia, ter mnogih drugih. Poleg tega ga mnoga podjetja uporabljajo v namen internega upravljanja z znanjem oziroma splošneje kot sistem za upravljanje vsebin (CMS).
2003	Mozilla Firefox (včasih Phoenix in Firebird, različica v Debianu Iceweasel) je brezplačen odprtokodni grafični spletni brskalnik, ki je na voljo za več operacijskih sistemov in preveden v več svetovnih jezikov. Razvilo ga je podjetje Mozilla Corporation s prostovoljci. Firefox, prej imenovan Phoenix ali Firebird, je preprostejša različica Mozille; ta je nastala na osnovi Netscape Navigatorja.
2008	Google Chrome je odprtokodni spletni brskalnik, ki ga razvija Google. Za izrisovanje na spletnih straneh se uporablja odprtokodna tehnologija WebKit.
2009	Android je odprtokodni programski jezik in operacijski sistem za pametne telefone ter ostale prenosne naprave. Zgrajen je na Linuxovem jedru.

1.2 Odprtokodni programi

Izraz odprta koda se nanaša na odprtokodne programe, za katere je na voljo izvorna programska koda, ki jo lahko prosto uporabljamo, modificiramo in distribuiramo v skladu z odprtokodnimi licencami (Open Source Initiative, 2010). SourceForge.net je največji spletni portal, ki ponuja odprtokodne programske rešitve z več kot 230.000 programerskimi projekti in ima 2 milijona registriranih uporabnikov.

Definicija odprte kode (Open Source Initiative, 2010) zahteva, da:

- omogoča svobodno redistribuiranje – odprto kodo lahko redistribuira kdorkoli brezplačno ali proti plačilu;
- je izvorna programska oprema dostopna uporabniku – licenca mora dovoljevati distribucijo tako v prevedeni, kakor tudi v izvorni kodi;
- licenca dovoljuje spremembe osnovne kode in izvedene oblike nove kode;
- kljub temu da mora biti izvorna koda dostopna, lahko njeni avtorji zahtevajo, da se morebitne spremembe jasno ločijo od originalne kode in tako ohranijo ločnico med prvotno in modificirano kodo (na primer v obliki popravkov ali različnih verzij);
- licenca ne sme omejevati katerekoli osebe ali skupine;
- licenca ne sme biti omejena glede na področje dela, v okviru katerega se programska koda uporablja;
- mora biti distribucija licenc enakovredna za vse uporabnike, brez dodatnih omejitev;
- se licenca za isto programsko kodo ne sme razlikovati, če se uporablja v kombinaciji z drugo programsko opremo;
- licenca ne sme omejevati uporabe druge programske opreme;
- mora licenca biti tehnološko nevtralna.

Po mnenju Fitzgeralda (2006, str. 593) lahko definicija odprtokodnih programskih rešitev pomeni produkt, proces ali gibanje. Vse tri sklope bomo obravnavali v tem poglavju.

Licence so pomemben in neločljiv del odprtokodne programske opreme. Danes je zelo težko obdržati pregled nad različnimi licencami in njihovimi komaj opaznimi razlikami. Tako GNU projekt kot Iniciativa za odprto kodo (OSI) sta ustvarili širok seznam licenc, ki ustrezajo kriterijem odprte kode. Ta seznam se stalno povečuje. Nekatere od bolj znanih licenc so: GPL, LGPL, BSD, MPL, IBM Public Licence.

GPL (angl. General Public Licence, v nadaljevanju GPL) zagotavlja svobodo pri uporabi, kopiranju in distribuciji programske opreme. Dovoljena sta tudi spreminjanje programske opreme ali uporaba programske kode. Vendar je distribucija programske opreme, ki temelji

na programski opremi, licencirani po GPL, dovoljena le pod licenco GPL. To pa pomeni, da tovrstne programske opreme ni mogoče prodajati in ni mogoče omejevali njene uporabe. Spremembe ali dodatkov k programski opremi pa ni treba javno objaviti. Spremembe, ki jih neko podjetje naredi za lastne potrebe, lahko ostanejo skrivnost. Obveza do objavljanja modifikacij dodatkov nastane takrat, ko se oseba, ki je spremenila programsko opremo odloči, da jo bo razpečevala. Glavni razlog njene popularnosti je njen virusni efekt, to je koncept copyleft.

LGPL (angl. Library General Public Licence, v nadaljevanju LGPL) se razlikuje od GPL v enem pomembnem vidiku: dovoljuje povezovanje z drugimi programi. Namenjena je programskim knjižnicam, torej programom s funkcijami, ki jih je mogoče izkoristiti v drugih programih. Uporaba tovrstnih programskih knjižnic v nobenem smislu ne omejuje programa, ki jih uporablja. Gre torej lahko za komercialni izdelek in podobno. Spremembe knjižnice pa so lahko distribuirane le v skladu z licenco LGPL, ki je v tem pogledu enaka licenci GPL, kar pomeni, da morajo biti razpečavane z izvorno kodo in brez omejitev uporabe.

BSD (angl. Berkeley Software Distribution, v nadaljevanju BSD) licenca je zelo preprosta. Dovoljuje uporabo programske opreme, distribucijo izdelka in izvorne kode. Dovoljeno je spreminjanje in vključevanje v drugo programsko opremo brez omejitve. Edina zahteva je, da se navede imena vseh avtorjev v izvorni kodi in dokumentaciji programa. Za razliko od GPL licence BSD omogoča, da se spremembe naredijo zasebne. Drugače povedano, lahko se spreminja pridobljena izvorna koda BSDlicenciranega programa in se ga potem prodaja v binarni obliki, ne da bi se poleg distribuirala tudi spremenjena izvorna koda. To je še vedno odprta koda, saj definicija odprte kode ne zahteva, da imajo izpeljana dela originalno licenco.

MPL (angl. Mozilla Public Licence, v nadaljevanju MPL) je licenca, ki dovoljuje brezplačno uporabo in distribucijo programske opreme. Razvila jo je združba Netscape Corporation, ko je objavila svoj brskalnik Netscape Navigator v odprti kodi. Vsakdo, ki razpečava spremembe ali dodatke k izdelku mora dovoliti enake pravice za njegov kos programske opreme in kombinacijo izvirnega dela in njegove rešitve. Pogoj za distribucijo je tudi, da je izvorna koda na voljo, kar pomeni, da je lahko vključena v distribucijo, lahko pa je na voljo na kakšni spletni strani. Definicija modifikacije programa v MPL je, da je to vsaka sprememba v datoteki, ki je del originalnega programa in vsaka nova datoteka, v katero je bil prepisan določen del izvorne kode originalnega programa. Veliko podjetij je sprejelo variacijo MPL za svoje programe. Tako so nastale licence: Netscape Public Licence, Interbase Licence, Nokia Open Source Licence itd.

Brezplačne programske rešitve in odprtokodne programske rešitve se pogosto obravnavajo kot sinonimi. Vendar pa obstajajo nekatere razlike med njimi v zvezi z licencami in ideologijo o tem, kako je treba razvijati programske rešitve (Fitzgerald, 2006, str. 594).

Brezplačne programske rešitve so običajno licencirane pod splošno javno licenco GNU (GPL), medtem ko odprtokodne programske rešitve uporabljajo GPL ali kakšno drugo licenco. Še ena razlika med temi pogoji je, in sicer da so brezplačne programske rešitve vedno tudi odprtokodne programske rešitve, odprtokodne programske rešitve pa niso vedno brezplačne programske rešitve (Fitzgerald, 2006, str. 596). V tej študiji izraz odprtokodne programske rešitve pomeni tako brezplačne kot odprtokodne programske rešitve.

1.3 Odprtokodne skupnosti

Raymond (2001, str. 19) je ponazoril novo formo razvoja s »katedralo in bazarjem«. Lastniški razvoj predstavlja katedralo: masivno, zaprto, počasno in spoštljivo. Odprtokodni razvoj predstavlja bazar: je fleksibilen, odprt za nove ideje in pristope, hitrejši in zelo neodvisen. Skupnosti so začele rasti s polno paro v zgodnjih 90. letih z Linuxom in s komercializacijo interneta.

Odprtokodne skupnosti sestavljajo podobno misleči uporabniki, ki uporabljajo in razvijajo programske sisteme, katerih izvorna koda je prosto dostopna. Ti prihajajo iz različnih delov sveta in med seboj sodelujejo s pomočjo interneta in ostalih interaktivnih orodij. Ne delajo razlik med podjetji in individualnimi uporabniki. Kultura odprtokodnih skupnosti je sodelovanje.

Skupnosti lahko sestavljajo uporabniki odprtokodnih programskih rešitev, ki zgolj poročajo o napakah v programski kodi, a v večini primerov skupnosti sestavljajo razvijalci, ki aktivno prispevajo k dejanski izvorni kodi (Von Krogh, Sebastian & Karim, 2003, str. 1219). Tako odprtokodne programske projekte sestavlja veliko raznovrstnih uporabnikov, ki prispevajo k dejanski izvorni kodi: organizacije, podjetja, posamezniki in strokovnjaki. Markus in ostali podrobneje razmejijo uporabnike odprtokodnih skupnosti na dve ravni. Prvo stopnjo sestavljajo razvijalci izvorne kode, drugo pa podjetja, ki izvorno kodo uporabljajo komercialno (Markus et al., 2000, str. 21).

Včasih odprtokodne skupnosti primerjamo z virtualnimi organizacijami, saj oboje predstavljajo avtonomne uporabnike, ki komunicirajo in sodelujejo za doseganje skupnih ciljev. Razvijalci odprtokodnih programskih rešitev imajo v skupnostih različne vloge: lahko so glavni razvijalci, lastniki, programerji, varnostni administratorji, recenzenti ali končni uporabniki. Skupnostim različnih programskih rešitev prispevajo tako, da sporočijo njihove popravke prek spletnih strani, forumov, e-poštnih sporočil in novic. Po Gallivanu (2001, str. 279) so skupnosti sestavljene najmanj iz treh vlog posameznikov:

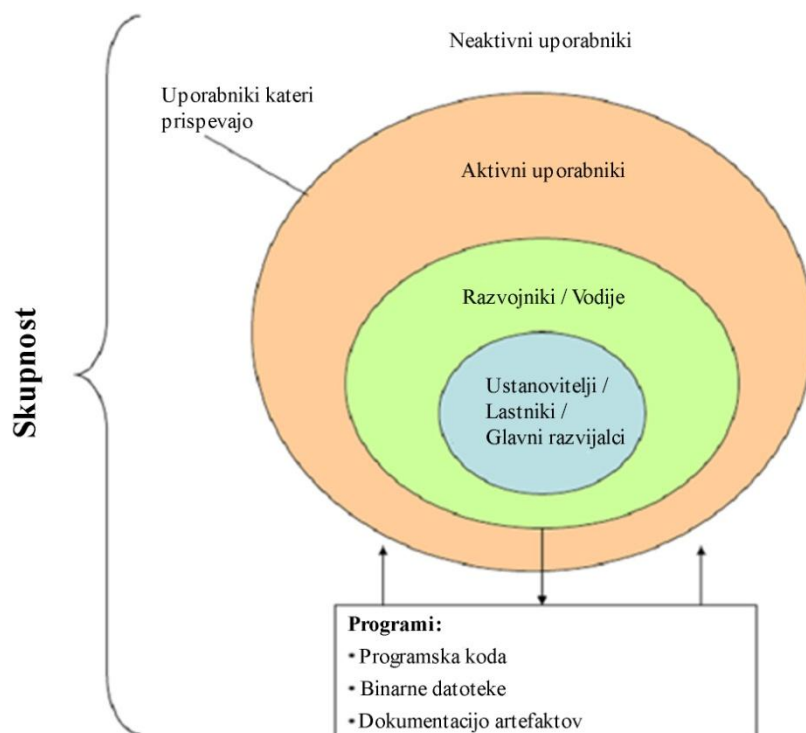
- vodje projektov, ki imajo splošno odgovornost za odprtokodne programske projekte;
- razvijalci, ki programirajo in predložijo kodo za odprtokodne programske projekte;

- posamezniki, ki uporabljajo programsko opremo in izvajajo testiranja, identificirajo napake in poročajo o morebitnih problemih.

V komercialnih programerskih družbah ostanejo vloge, ki jih določi projektni vodja, dalj časa nespremenjene, za razliko od vlog v odprtokodnih programskih skupnostih, kjer se konstantno spreminjajo. V slednjem primeru vloge niso vnaprej dodeljene in člani lahko vplivajo na vlogo ki jo želijo pri razvoju programskih rešitev. Odprtokodni programski projekti in skupnosti se običajno začnejo z individualnim pristopom ali majhnimi skupinami, ki imajo idejo, kaj bi razvijali v intelektualne, osebne ali poslovne namene. Ustanovitelji projekta običajno postanejo takoimenovani projektni lastniki ali upravitelji, ki prevzamejo odgovornost za vodenje projektov. Večina projektov se začne z alfa različico, ki opisuje glavne funkcionalnosti programskih rešitev in se tudi kot prva objavi na internetu ter je tako vsem na voljo (von Hippel & von Krogh, 2003, str. 211). Če projekt vzbudi zanimanje in privabi nove uporabnike v skupnost, bodo ti novi uporabniki dali v zameno svoje prispevke izvorni kodi programskih rešitev. Ta rastoči cikel raste z obstoječo skupnostjo, ki se ji vedno znova pridružujejo novi uporabniki, ki so zainteresirani prispevati svoj delež. Novi uporabniki lahko izvorno kodo v nadaljevanju izboljšujejo, spreminjajo ali ji dodajajo nove funkcionalnosti, da bi programske rešitve čimbolj prilagodili svojim lastnim potrebam.

Odprtokodne programske skupnosti so pogosto ponazorjene s podobo čebulne strukture, ki temelji na stopnji prispevka. Na sredini čebule je najmanjša skupina, imenovana glavni razvijalci, ki je odgovorna za večino razvoja kode programskih rešitev ter nadzira načrtovanje in razvoj projekta (Flosscom.net, 2008). Glavni razvijalci morajo biti vztrajni in sposobni doseči temeljne cilje. Obstajajo številne študije ki dokazujejo, da večino izvirne kode ustvarijo glavni razvijalci, ki ustvarijo skoraj 80 % vseh prispevkov kode. Ustanovitelji in lastniki so tudi del jedra skupnosti in odločajo o tem, katere prispevke bi bilo treba vključiti in katere je treba obravnavati. V naslednjem obroču so razvijalci in vodje. Prispevajo, spreminjajo, pregledujejo in predložijo popravljeno izvorno kodo, ki jo nato pregleda še ožja skupina. V tretjem obroču so aktivni uporabniki, ki ne prispevajo k izvorni kodi, vendar pa uporabljajo najnovejše različice programskih rešitev, testirajo programe, pišejo poročila o napakah, ustvarjajo nove ideje, ki zahtevajo nove funkcije. Na obrobnihih plasteh čebule pa so neaktivni uporabniki, ki uporabljajo programske rešitve, ne da bi kar koli prispevali ali poročali o projektu na forumih ali po e-pošti (Flosscom.net, 2008).

Slika 1: Grafični prikaz čebulne strukture odprtokodne programske skupnosti



Vir: FLOSSCOM.net, Diagram skupnosti, 2008, str .4.

Skozi čas se lahko vloga posameznega uporabnika spreminja. Na začetku se lahko projektu pridružimo kot neaktivni uporabnik, kasneje začnemo prispevati s poročanjem o napakah, z udeležbo na projektnih forumih in tako začnemo prispevati v izvirno kodo. Več kot bomo prispevali, višje bomo rangirani in postali bomo del izbrane skupine glavnih razvijalcev. Vsi uporabniki si ne želijo postati del glavnih razvijalcev, ampak si želijo le uporabljati programske rešitve, spet drugi želijo samo občasno prispevati prispevke. V kolikor si nobeden od novih uporabnikov ne prizadeva postati del glavnih razvijalcev z rednimi prispevki, je razvoj skupnosti ogrožen ali pa se lahko celo prekine, če se kateri od glavnih razvijalcev odloči zapustiti skupnost.

1.4 Značilnosti odprtokodnih projektov in razvoj

Pri odprtokodnih projektih poleg programskih rešitev veliko pozornosti pridobijo tudi vzporedni dogodki, kot so odprtokodne razvojne prakse, procesi, prostovoljstva, vzporedni inženiring in hiter razvoj aplikacij (Gallivan, 2001, str. 286). Razvojni procesi odprtokodnih programskih rešitev se v veliki meri razlikujejo od tradicionalnih. Glavne razlike so (Raymond, 2001, str. 26):

- da odprtokodne programske rešitve razvija potencialno veliko število prostovoljnih programerjev,

- da delo ni dodeljeno in ga ljudje opravljajo le zato, ker si to želijo,
- da ni podrobnega načrta,
- da ni projektnega plana, urnika ali seznama rezultatov.

Hertel, Niedner & Herrmann S. (2003, str. 1161) pravi, da kljub neformalni organizacijski strukturi obstajajo nekateri pogoji za uspešen razvoj odprtokodnih programskih rešitev:

- delegativna in participativna načela vodenja z jasnimi pristojnostmi,
- modularna projektna struktura, ki zmanjšuje nepotrebno zapletenost,
- vzporedna politika razvoja, ki omogoča hiter razvoj in stabilen delovni sistem,
- motiviranje s kreditnimi točkami za prispevke razvijalcev in dokumentacije del,
- jasna pravila in norme skupnosti, ki so zapisana na internetnih straneh,
- preprosta in zanesljiva komunikacijska orodja, ki so na voljo po celem svetu.

1.4.1 Razvojni proces

Tradicionalni življenjski cikel razvoja programskih rešitev je sestavljen iz različnih faz, katerih cilj je razvoj programskih rešitev. Eden bolj znanih tradicionalnih modelov razvoja programskih rešitev je slapovni model, ki je sestavljen iz sedmih faz: analiza zahtev, načrtovanje, izvajanje, testiranje, integracija in vzdrževanje (Petersen & Wohlin & Baca, 2009, str. 389). V nasprotju s tradicionalnim razvojem programskih rešitev odprtokodni razvoj ni sestavljen iz zgoraj navedenih faz, ampak poteka povsem nasprotno. Končni izdelek se ne izda ob koncu razvoja življenjskega cikla, ampak na njegovem začetku (Feller & Fitzgerald, 2002, str. 101).

Raymond (2001, str. 19) navaja, da je razvoj tradicionalnih programskih rešitev zapleten, ponazarja ga s katedralami, kjer skupina nadarjenih programerjev dela v izolaciji in izda proizvod, ko je pripravljen. Kot pravo nasprotje Raymond navede razvoj odprte kode, ki jo primerja z velikim bazarjem, kjer se srečujejo številni programerji. Veliko število uporabnikov, zgodnje in pogoste izdaje so ključni del razvoja programskih rešitev v slogu bazarja. V bazarju je pogled na odpravljanje napak programskih rešitev zanemarljiv pojav, ker se nove različice izdajajo pogosto in strokovnost razvijalcev ni vprašljiva. Napake so sposobni odkriti in popraviti zelo hitro. V katedrali pa je odkrivanje napak in razvojnih problemov veliko težje, saj se pojavijo šele po dolgih intervalih, ko programske rešitve izidejo v javnost.

Ni univerzalne strukture, ki bi določala, kako naj poteka razvoj odprtokodnih programskih rešitev, a vendar je veliko odprtokodnih projektov, ki imajo podobnosti v razvojnem procesu. Projekti se običajno začnejo, ko si nekdo želi nekaj razviti, da bi izpolnil določene potrebe. Ko so izdelani razvojni okviri za odprtokodne programske rešitve, se v javnost izda izvorna koda. Če postane projekt zanimiv, se bodo začeli pojavljati novi razvijalci, ki

bodo prispevali izvirno kodo in z razvojnimi prispevki se bo prvotna različica postopoma razvijala (von Hippel & von Krogh, 2003, str. 211).

Raymond (2001, str. 26) navaja 19 smernic za ustvarjanje dobrih odprtokodnih programskih rešitev:

1. Vsako dobro delo programskih rešitev se začne z razvijalčevo lastno željo po razvoju.
2. Dobri programerji vedo, kaj naj napišejo. Najboljši vedo, kaj novega naj napišejo ali ponovno uporabijo.
3. Kljub dobremu planiranju boste kakšno rešitev zavrgli.
4. Če imate pravi odnos, vas bodo našle zanimive težave.
5. Ko izgubite zanimanje za program, je vaša dolžnost, da ga predate naslednikom.
6. Obravnavati vaše uporabnike kot sorazvijalce, da vaša pot do hitrega izboljšanja kode in učinkovitega odpravljanja napak ne bo težavna.
7. Zgodnje sprostitev. Pogoste sprostitev. Prisluhnite vašim strankam.
8. Zaradi dovolj velikega beta testa in sorazvijalske osnove bo skoraj vsak problem hitro odpravljen, saj je popravilo jasno vsakemu razvijalcu.
9. Dobre podatkovne strukture in slaba razvojna koda deluje veliko bolje kot pa dobra razvojna koda in slabe podatkovne strukture.
10. Če obravnavate beta preizkuševalce kot vaš najbolj dragocen vir, se bodo ti odzvali in postali vaš najbolj dragocen vir.
11. Dobra stvar dobrih idej je njihovo priznavanje s strani vaših uporabnikov.
12. Najbolj presenetljive in inovativne rešitve prihajajo iz spoznanj, da je bil vaš koncept problema napačen.
13. Perfekcije (v oblikovanju) ne dosežemo, ko ni ničesar več za dodati, ampak ko ni ničesar več za odvzeti.
14. Vsako orodje bi moralo biti uporabno v pričakovani smeri razvoja, ampak resnično odlično orodje je uporabno, kakor niste nikoli pričakovali.
15. Pri pisanju programskih rešitev vseh vrst je potrebnih čim manj motenj. Pri programiranju nikoli ne zavrzite podatkov, razen če to od vas zahtevajo!
16. Ko niste blizu rešitvi, je lahko vaš prijatelj orodje za odkrivanje napak.
17. Varnostni sistem je le toliko varen, kot je varovana njegova skrivnost. Pazite na psevdonimske skrivnosti.
18. Da bi rešili zanimiv problem, moramo začeti z iskanjem problema, ki je zanimiv za nas.
19. Internet je vsaj tako dober kot razvojni koordinator in ve, kako voditi brez prisile, v stilu več glav več ve.

Opensolaris¹³ identificira pri procesu odprtokodnih programskih rešitev šest razvojnih stopenj. **Prvi korak** je programiranje, pri katerem uporabniki projekta prispevajo kodo ali pa odpravljajo napake. V **drugem koraku** strokovnjaki pregledajo prispevke, da bi izboljšali kakovost prispevkov. **Tretji korak** se imenuje test pred izdajo, s katerim glavni

¹³ Odprtokodni delovni proces, potreben za razvoj Solaris operacijskega sistema.

razvijalci, ki so odgovorni za prispevke in za to, da se programiranje z izidom končne različice ne ustavi, testirajo kodo, ki deluje samostojno in v povezavi z drugimi datotekami. V **četrtem koraku** lastniki projekta odločijo, kdaj je razvojna različica pripravljena na izdajo. V nekaterih primerih lahko tudi lastniki prenesejo pooblastila na določene razvijalce, ki lahko spremembe vključijo neposredno v razvojno izdajo. V **petem koraku** so programske rešitve izpostavljene odpravljanju napak, kjer lahko vsi razvijalci in uporabniki poročajo in odpravljajo napake, ki jih najdejo. V **zadnjem koraku** postanejo prispevki del produkcijske izdaje, ki preidejo v stabilno produkcijo. Ta je običajno sestavljena iz dveh različnih korakov. Pri prvem koraku preidemo iz razvoja k produkciji, v drugem koraku pa se izvorna koda zapre tako, da je možno le odpravljati napake. V fazi produkcije lahko glavni razvijalci to ustavijo, če ugotovijo, da programska rešitev ni pripravljena na izdajo.

1.4.2 Vodenje in koordiniranje

Ena od značilnosti razvoja odprtokodnih programskih rešitev je pomanjkanje formalnega vodenja. Običajno pobudniki projekta postanejo tudi lastniki, ki prevzemajo odgovornost za vodenje projekta. Ti bi morali imeti pomembnejše vloge, kot je na primer nadzor nad prispevki, ki se vključujejo v izvorno kodo. Nekateri avtorji so predlagali, naj ima Linus Torvalds vodilno vlogo v skupnosti Linux (Raymond, 2001, str. 38). Pobudniki imajo pogosto precej omejeno vlogo in nimajo formalne avtoritete nad skupnostjo. Poleg tega razvoj odprtokodnih programskih rešitev vključuje več zadev, ki pomenijo precejšnje izzive upravljanja tudi za formalna podjetja. **Prvič**, razvijalci niso zaposleni, kar pomeni, da sta izguba zaposlitve ali znižanje plač neučinkovita za razvijalce, ki ne želijo več prispevati svojih rešitev. **Drugič**, število razvijalcev na projektu je lahko več kot pol milijona in na veliko projektih so prav vsi prostovoljci dobrodošli. Organizacija velikanske množice ni enostavna naloga in vodje projektov niso vedno večši, da bi izbrali ljudi, katerim bi dodelili posebne naloge. Vendar pa veliko odprtokodnih projektov (Apache, MySQL, php, Linux, Google ...) dela presenetljivo dobro in imajo dobro strukturirano upravljanje modelov in jasno opredeljeno vodstvo (Markus et al., 2000, str. 20).

Za koordiniranje razvoja odprtokodnih programskih rešitev so Markus in ostali identificirali štiri medsebojno povezane mehanizme (Markus et al., 2000, str. 21): vodenje članstva, pravil, sankcioniranje in ugled. Bistven pogoj za delovanje teh štirih mehanizmov upravljanja je niz skupnih vrednot in domnev o tem, kako stvari delujejo. Z **vodenjem članstva** odprtokodnih projektov skušamo omejiti potencialno velike in morebiti nesposobne delovne sile. Oblikujemo skupnosti, ki so dovolj velike, da so obvladljive in visoko kakovostne za opravljanje dela. Na primer vodje projektov lahko ocenjujejo, ali so razvijalci sposobni za te posebne vloge ali ne. Imajo pa tudi pooblastila, da lahko izključijo člane, ki ne upoštevajo pravil skupnosti. Odprtokodne programske licence predstavljajo **pravila**. Različne licence lahko na primer tudi omejujejo uporabo odprtokodnih

programskih rešitev za komercialne namene. Naslednja stvar je drugačen način vodenja postopkov za razpravo in glasovanje o pomembnih vprašanjih. Večina teh dejavnosti poteka na internetu. Tako se v Linux skupnosti glasovanje uporablja kot mehanizem usklajevanja. Glasovanje uporabljajo za prisilne odločitve in za testiranje soglasja. Skupnost spremlja svoje člane in kaznuje tiste, ki kršijo njene norme. Ti mehanizmi delujejo tudi v odprtih skupnostih, saj je v njih močan socialni pritisk, in člani skrbijo, kaj drugi mislijo o njih. **Sankcioniranje** se lahko zgodi zaradi pošiljanja neustrezne ali sovražne e-pošte, pošiljanja velikega števila nezaželenih sporočil z uporabo e-pošte in sprenevedanja članov, ki se ne odzivajo na e-pošto. Ob soočanju s sankcijami člani pogosto prostovoljno zapustijo skupnost. Nenazadnje je želja članov, da upravljalni mehanizem ohrani dober **ugled**. Strah pred izgubo ugleda motivira razvijalce, da delajo dobro delo in na način, da sledijo pravilom skupnosti.

Tudi Egyedi in Wendel de Joode (2006, str. 74) sta preučevala mehanizme koordiniranja v odprtokodnih skupnostih. Predlagata, da se koordiniranje v skupnostih odprtokodnih programskih rešitev opira na čredni mehanizem, regulativne mehanizme, operativne mehanizme in koordinacijski organ. **Čredni mehanizem** pomeni, da bodo priljubljeni odprtokodni projekti postali še bolj priljubljeni. Razvijalci odprtokodnih programskih rešitev si pogosto želijo pridobiti ugled in zato aktivno sodelujejo v projektih, ki imajo veliko občinstvo. **Regulativni mehanizmi usklajevanja** se nanašajo na licence, sporazume o sodelovanju in blagovne znamke. Nekateri skupnosti zahtevajo, da razvijalci podpišejo sporazum o sodelovanju, preden lahko neposredno prispevajo k izvorni kodi. Da bi privabljele nove razvijalce, jih znane blagovne znamke prepričujejo, da razvijalci odprtokodnih programskih rešitev ne razvijajo več novih različic. **Operativno usklajevanje** mehanizmov vključuje orodja, kot je sočasno verzioniranje sistemov(CVS) in priročnikov. **Koordinacijski organ** se sklicuje na koordinativne učinke distribucije (na primer Debian¹⁴, Red Hat¹⁵ in Suse¹⁶) in vloge lastnikov v odprtokodnih skupnostih. Lastniki so organ za sprejemanje odločitev ali za odobritev novih prispevkov.

Odpravljanje napak in **zavestno prispevanje k izgradnji skupnosti** so tudi mehanizmi koordiniranja v odprtokodnih skupnostih. Prav **odpravljanje napak** daje zelo pomemben prispevek in določa dnevni red vprašanj, na katera je treba odgovoriti. **Zavestna izgradnja skupnosti** pomeni, da znotraj skupnosti obstajajo smernice (na primer bonton za komuniciranje) in struktura za usklajevanje dela (Iannacci, 2005, str. 4).

¹⁴ Odprtokodni operacijski sistem.

¹⁵ Odprtokodni operacijski sistem.

¹⁶ Odprtokodni operacijski sistem.

1.5 Motivacija odprtokodnih programskih razvijalcev

Razvijalci odprtokodnih programskih rešitev imajo veliko različnih razlogov, zato da razvijajo, popravljajo in spreminjajo kodo brezplačno. Glavne razloge za sodelovanje pri odprtokodnem razvoju razdelimo na tri motivacijska področja: **socialno, ekonomsko in tehnološko** (Bonaccorsi & Rossi, 2006, str. 3).

Nekateri zagovorniki razvoja odprtokodnih programskih rešitev trdijo, da programerji odprtokodnih programskih rešitev niso motivirani z monetarnimi spodbudami, ampak z njihovimi osebnimi preferencami, hobiji in prizadevanji za povečanje blaginje drugih. Zdi se, da so razvijalci odprtokodnih programskih rešitev veliko bolj verjetno motivirani zaradi **socialnih** dejavnikov kot običajno zaposleni razvijalci programske opreme. Altruizem in vzajemnost so pomembni dejavniki, ki spodbujajo programerje, da sodelujejo v odprtokodnih projektih programskih rešitev. Zato si želijo biti del skupnosti in se opredeliti kot člani skupnosti. Odprtokodni programerji verjamejo, da je ponujati brezplačne programske rešitve prava stvar. To je lahko tudi eden od načinov za boj proti lastniškim podjetjem programske opreme (Stallman, 1983). Navdušenje in zanimanje za programiranje sta tudi razloga za motivacijo programerjev (Raymond, 2001, str. 53).

Kot smo ugotovili, so altruizem, ugled in ideologija pomembni motivatorji, a ne morejo biti indiferentni do **ekonomskih** vprašanj. Kar pomeni, da kljub temu, da veliko razvijalcev odprtokodnih programskih rešitev dela brez denarnega nadomestila, še ne pomeni, da vsi programerji delajo brezplačno. Glede na ugotovitve 81 posameznikov, ki so sodelovali v odprtokodnih projektih, je bilo 16 % vprašanih neposredno plačanih za svoje prispevke (Bonaccorsi & Rossi, 2006, str. 7). Poleg tega so nekatere licence sestavljene v skladu z odprtokodno definicijo, a omogočajo izgradnjo komercialnega odprtokodnega okolja. Razvijalci lahko dodajo nove funkcije in funkcionalnosti kodi v razvojnem okviru odprtokodnega projekta in se tako vključijo v programska podjetja, ki bi v nadaljnjem razvoju zaposlila razvijalca in mu zagotovila poklicni in strokovni napredek. Priložnost za komercialno korist od odprtokodnih programskih rešitev zagotavlja močno spodbudo za mnoge posameznike in podjetja. V odprtokodnem svetu finančna nadomestila plače ali pogodbene pristojbine nadomestijo deleži v skupno proizvedenem bogastvu. Programerji odprtokodnih programskih rešitev udeležbo v skupnostih obravnavajo kot naložbo za prihodnost. Po njihovem mnenju bi lahko v prihodnje dodatni zaslužek ustvarjali s prodajo sorodnih proizvodov ali storitev, s povečanjem človeškega kapitala (spretnosti, sposobnosti in znanja) ali pa bi delo na projektih odprtokodnih programskih rešitev obravnavali kot učinkovit način za dokazovanje sposobnosti in spretnosti podjetjem.

Ena izmed **tehnoloških** spodbud za sodelovanje v razvoju odprtokodnih programskih rešitev je, da poskušamo zapolniti nezapolnjen trg programskih rešitev. Odprtokodni projekti lahko nastanejo, ker ne obstaja poseben program za opravljanje določenih nalog.

Tipičen primer je programski jezik Perl, ki ga je razvil sistemski administrator, ki je videl potrebo po združitvi hitrosti in neposrednosti Unix lupine za razvoj zapletenih programov, kar je značilno za programske jezike visoke ravni, kot je programski jezik C. Učenje je tudi pomemben dejavnik za motivacijo tistih, ki prispevajo. Na primer, pri razvoju podatkovne baze MySQL je učenje iz branja odgovorov na vprašanja eden od najpomembnejših motivatorjev za udeležence. Učenje je možno tudi s pomočjo konstantnih povratnih informacij, ki jih predlagatelji dobijo od drugih članov skupnosti.

Tabela 2: Taksonomija motivacij posameznih programerskih razvijalcev

Motivacijsko področje	Mikro nivo	Reference
Socialno področje	Zabavno programiranje	Torvalds & Diamond (2001)
	Altruizem	Raymond (2001), Bergquist & Ljungberg (2001)
	Občutek pripadnosti skupnosti	Raymond (2001)
	Boj proti lastniškim programom	Stallman (1983)
Ekonomsko področje	Denarne nagrade	Lerner & Tirole (2002)
	Nizki oportunitetni stroški	Bonaccorsi & Rossi (2003), Lakhani & von Hippel (2003)
	Ugled med vrstniki	Delle & David (2003)
	Koristi v karieri	Lerner & Tirole (2001)
Tehnologija	Učenje	Lakhani & von Hippel (2003)
	Prispevki in povratne informacije iz skupnosti	Raymond (2001), Bonaccorsi & Rossi (2003)
	Najnovejša neuveljavljena tehnologija	David & Pfaff (1998)
	Lastno zanimanje za razvoj	Feller & Fitzgerald (2002)

Vir: A. Bonaccorsi in C. Rossi, Comparing Motivations of Individual Programmers and Firms to Take Part in the Open Source Movement, 2006, str. 5.

1.6 Motivacija podjetij

Motivacija podjetij za sodelovanje v razvoju odprte kode ni tako široko raziskana kot motivacija posameznih razvijalcev. Tudi motivacijo podjetij razdelimo na tri motivacijska področja: **socialno, ekonomsko in tehnološko** (Bonaccorsi & Rossi, 2006, str. 12).

Motivacija podjetij za sodelovanje v razvoju odprte kode ni tako široko raziskana kot motivacija posameznih razvijalcev, a vseeno je vedno več podjetij, ki želijo sodelovati pri fenomenu odprtokodnih pojavov, bodisi z uporabo odprtokodnih programskih rešitev bodisi s sprostitev izvorne kode lastniških programov in razvojne prakse odprtokodnih programskih rešitev, z zagotavljanjem finančne podpore odprtokodnim projektom ali z nudenjem izkušenih programerjev za odprtokodne projekte. Takšno odprtokodno gibanje prinaša povsem nove načine ustvarjanja dohodkov in s tem se pojavijo tudi novi poslovni modeli. Podjetja lahko na primer pridobijo dohodke s prodajo sorodnih izdelkov ali s ponudbo povezanih storitev, kot je na primer podpora. Spodbujanje prodaje strežnikov je IBM realiziral z motivacijsko odločitvijo, da bo vlagal v Linux. Za podjetja je jasno, da sodelujejo pri odprtokodnem gibanju zato, da bi posredno ali neposredno izboljšala finančni položaj. Zato sta glavna razloga za sodelovanje v odprtokodnem razvoju povečanje prihodkov ali zmanjšanje stroškov. Posledično imajo glavno vlogo pri motivaciji podjetij **ekonomskimi vidiki**.

Večina odprtokodnih programskih rešitev je na voljo na spletnih straneh (na primer sourceforge.net, opensource.org...), kjer so prosto dostopni za prenos, uporabo in spremembe. Nizki stroški nakupa so še vedno očitni razlog za izbiro odprtokodnih programov, a se jih kljub temu veliko podjetij odloči uporabiti za izboljšanje storitev (Madanmohan, 2004, str. 66). Veliko je tudi odprtokodnih programskih rešitev, ki ponujajo alternativo komercialnim proizvodom. »Open-office« je izzval komercialnega tekmeca »Microsoft office« v poslovni rabi in zelo uspel, saj ga je veliko podjetij začelo uporabljati (Wichmann, 2002). Podjetja lahko izkoristijo prednosti odprtokodnih skupnosti s sprostitev izvorne kode njihovih lastniških programskih rešitev. Tako je Nokia sprostila v javnost izvorno kodo platforme za mobilni telefon S60 in poskusila pritegniti člane skupnosti za sodelovanje pri inovacijah in razvojnem delu. Podjetja lahko tako delijo kodo in znanja s skupnostjo in uporabljajo povratne informacije in prispevke članov skupnosti (Feller & Fitzgerald, 2002, str. 117). Za podjetja je lahko strateški premik tudi, da vključijo odprtokodni sistem »Linux« v strojno opremo. Na primer, podjetje Asus ima na trgu strojne opreme močne tekmece in je zato na trg poslal najcenejši mini prenosni računalnik Asus Eee PC, ki temelji na Linux operacijskem sistemu, ki stane le 199 dolarjev, kar je veliko ceneje za uporabnike, saj najcenejši konkurenčni računalnik z operacijskim sistemom Windows XP stane 259 dolarjev (Glyn, 2008). S tem lahko podjetja morebiti pridobijo tržni delež od svojih konkurentov in celo možnost za povečanje prodaje nekaterih sorodnih izdelkov ali storitev. Glede na empirične ugotovitve Lerner in Tirole (2002, str. 204) se zdi, da so odprtokodni projekti dobili nov zagon, zlasti če bi nastopali proti prevladujočim podjetjem.

Ena izmed tehnoloških spodbud za sodelovanje v razvoju odprtokodnih programskih rešitev je, da podjetja poskušajo zapolniti nezapolnjen trg programskih rešitev. Z naslednjo spodbudo skuša OSI (Open Source Initiative, 2010) spodbuditi standardizacijo z razvojem

odprtokodnih programskih rešitev, saj je trg programske in strojne opreme zasičen z velikim številom različnih proizvajalcev in izdelkov. Združljivost med temi proizvodi pa predstavlja velik izziv. Na primer vektorski program za oblikovanje Corel Draw¹⁷ lahko namestimo le na operacijskem sistemu Windows, na Linuxih in OSX¹⁸ se to ne da. Združljivost je treba upoštevati pri sprejemanju odločitev o informacijski infrastrukturi, saj če se v bodoče izkaže, da je treba zamenjati operacijski sistem Windows z Linuxi, nas bo to drago stalo. Proizvajalci programske opreme pa bodo morali podpirati vse večje število kombinacij med programskimi rešitvami (na primer operacijski sistemi) in strojno opremo. Če bi odprtokodne skupnosti zagotovile paket ustreznih programov za poslovno rabo, bi podjetjem ne bilo treba biti boja z vprašanji o združljivosti z različnimi proizvajalci. Tako bi se lahko osredotočili na ključne sposobnosti, ki bi bile ustvarjanje poslovnih in ne programskih rešitev. Prav zato je odprtokodno gibanje še posebej zanimivo za velika podjetja, kot so Sun Microsystems, IBM, Oracle, ki želijo postati aktivni udeleženci pri razvoju odprtokodnih programskih rešitev, da bi le to lahko izkoristili v svoj interes. Po drugi strani pa bi s podporo odprtokodnim projektom podjetja lahko vnaprej prepoznavala razvojne standarde tehnologij, ki so v lasti konkurence.

Zdi se, da so zaradi socialnih dejavnikov razvijalci odprtokodnih programskih rešitev veliko bolj motivirani kot običajni zaposleni razvijalci programskih rešitev. HP uporablja razvojne prakse odprtokodnih programskih rešitev tudi znotraj podjetja, saj lahko s pomočjo socialnih dejavnikov vzpostavijo odprtokodne tehnike sodelovanja, delijo izvorno kodo v podjetju in ponovno razvijajo programske rešitve. Na ta način je podjetje HP racionaliziralo razvoj programskih rešitev, spodbudilo ponovno uporabo obstoječih programskih rešitev, poenostavilo razvojne probleme, zmanjšalo stroške podpore za 80 % in pospešilo razvojne cikle. Ta pristop zahteva sodelovanje med zaposlenimi, kar pomeni, da mora družba motivirati zaposlene za upravljanje procesa sprememb. Druga možnost je, da podjetje uvede nove delovne prakse, za katere pa ni nujno, da jih tudi sprejme. To je v skladu z odprtokodnimi načeli. Če zaposleni niso seznanjeni z odprtokodnim pojavom in vidijo v tem zgolj nov način dela, niso motivirani za zamenjavo obstoječih delovnih navad, do sprememb čutijo odpor. Če pa so zaposleni seznanjeni z odprtokodnim pojavom, ga lažje sprejmejo in njihovo delo temelji še vedno na prostovoljnosti.

Ob vključevanju v odprtokodne projekte podjetja vključijo odprtokodne zamisli v komercialne programske rešitve. Tako je Google na primer izdal mobilni telefon G1, ki izrablja Android operacijski sistem, ki temelji na Linux kodi. S priključevanjem skupnosti pri razvoju projekta Google Android nadarjenim programerjem to predstavlja osebni izziv, istočasno pa privabijo tudi programerje, ki želijo delati v intelektualno zahtevnem okolju.

¹⁷ Oblikovalski program

¹⁸ Operacijski sistem podjetja Apple

Eucalyptus je odprtokodna programska infrastruktura za implementiranje »računalništva v oblaku« za grozde. Trenutni vmesnik za Eucalyptus je združljiv z vmesniki Amazon EC2, S3, in EBS, vendar je infrastruktura namenjena podpori več vmesnikov na strani klientov. Eucalyptus se izvaja z uporabo splošno razpoložljivega orodja Linux in osnovnih spletnih storitev, zaradi česar je enostaven za namestitev in vzdrževanje.

Tabela 3: Taksonomija motivacij podjetij

Motivacijsko področje	Mikro nivo	Reference
Ekonomsko	Neodvisni od cene in politike licenc velikih programskih podjetij	Lerner & Tirole (2002)
	Spodbujanje novih programskih modelov za potrošniško naravnane storitve (pridobivanje denarja z dopolnilnimi storitvami)	Feller & Fitzgerald (2002), Lerner & Tirole (2002)
	Pridobivanje dohodka s prodajo posrednih izdelkov, povezanih s produktom	Feller & Fitzgerald (2002), Lerner & Tirole (2002)
	Ponudba inovacij (z raziskavo in razvojem dejavnosti odprtokodne skupnosti)	von Krogh et al.(2002)
	Najem dobrih ID specialistov	Fink (2003), Lerner (2002)
Socialno	V skladu z vrednotami odprtokodne skuposti (da se ne izda zaupanja odprtokodnih razvijalcev)	Lerner & Tirole (2002)
	Deliti izvorno kodo in znanje s skupnostjo (za vzdrževanje sodelovanja)	Lerner & Tirole (2002)
	Zavedati se, da programska oprema ne sme biti lastniška dobrina (za zmanjševanje tržne moči velikih programskih podjetij)	Feller & Fitzgerald (2002)
Tehnološko	Uporaba povratnih informacij in prispevkov razvijalcev odprtokodnih skupnosti za zmanjševanje stroškov razvoja in izboljšanje programskih rešitev	Lerner & Tirole (2002), Hecker (2000)
	Uporaba povratnih informacij in prispevkov skupnosti uporabnikov za preizkušanje in izboljšanje programskih rešitev	von Hippel (2002), Fink (2003)
	Zmanjšanje stroškov strojne opreme	Feller & Fitzgerald (2002)

»se nadaljuje«

»nadaljevanje«

Motivacijsko področje	Mikro nivo	Reference
	Pospeševanje standardizacije	Fink (2003), Lerner (2002)
	Obravnavanje vprašanj varnosti	Fink (2003)

Vir: A. Bonaccorsi in C. Rossi, 2006, *Comparing Motivations of Individual Programmers and Firms to Take Part in the Open Source Movement*, str. 14.

1.7 Različni pristopi v odnosu podjetje – skupnost

Po izdaji izvirne kode sta naslednji velik korak podjetij (IBM, Sun, Nokia ...) skupnost in želja po čim večjem sodelovanju uporabnikov in razvijalcev v skupnosti. Želijo si, da bi bile njihove skupnosti vsaj tako dobre kot MySQL-ove, Linux-ove in podobno uspešne odprtokodne skupnosti.

Da bi podjetja pritegnila čim večje število razvijalcev, ki bi brezplačno prispevali k skupnosti, bi morala tudi sama razkriti, kako spodbujajo in sodelujejo s skupnostjo. Pri podjetju IBM to počnejo na različne načine: od denarnih nagrad, napredovanj v ožje projekte, do zaposlitve, izobraževanj itd. Interakcija med podjetjem in skupnostjo je lahko problematična, saj kot vemo, je skupnost motivirana s socialnimi dejavniki, med tem ko so podjetja motivirana predvsem s kapitalom. Bonaccorsi in Rossi (2006, str. 20) sta predlagala tipologijo štirih različnih pristopov v odnosu podjetje – odprtokodna skupnost: **podjetja, orientirana k skupnosti, podjetja ki niso orientirana k skupnosti, oportunistična podjetja, neopazno orientirana podjetja k skupnosti.**

Pristop **podjetja, orientirana k skupnosti**, pomeni, da obstaja razumevanje med podjetjem in skupnostjo. Podjetje poskuša sodelovati in razvijati samo skupnost. Lahko pa pride do napetosti med podjetjem in skupnostjo, če cilji in strategija podjetja niso v skladu s cilji in strategijo skupnosti, saj člani skupnosti nimajo nobene formalne povezave s podjetjem. Da bi se izognili tem napetostim, je pomembno, da management podjetja neposredno sodeluje pri razvoju skupnosti, saj s tem podjetje pridobi legitimnost, da lahko vpliva na skupnost. Podjetja bi morala spoštovati norme in vrednote skupnosti ter pridobljene koristi od skupnosti vrniti nazaj v skupnost. V tem kontekstu bi lahko podjetja izdala del izvirne kode, ki je bila interno razvita v skupnosti ali zagotovila dobro delujočo infrastrukturo, ki bi omogočala izvajanje različnih nalog in spodbujala interakcijo. Za podjetja je koristno, da razkrijejo njihove namene o motivaciji, saj več kot skupnost ve o motivaciji podjetja, boljši bodo predlogi za podjetje.

Podjetja, ki niso orientirana k skupnosti ne delijo vrednot s skupnostjo in ne delujejo v skladu s pravili odprtokodnih skupnosti.

Oportunistični pristop pomeni, da se podjetja osredotočajo samo na lastne koristi, in ne na interese skupnosti. Podjetja iščejo le koristne vloške, ne da bi spoštovala norme, vrednote in pravila skupnosti. Podjetja, ki se ukvarjajo z odprtokodnimi programskimi rešitvami nenamerno izberejo ta pristop, saj povzroča škodo skupnosti in ne zagotavlja trajnostnega poslovnega modela.

Pristop **neopazno orientirana podjetja k skupnosti** je vmesni pristop med oportunističnim pristopom in pristopom podjetja, orientirana k skupnosti. To pomeni, da podjetja pridobivajo koristi od skupnosti brez povzročanja škode. Tveganje, povezano s pristopom neopazno orientirana podjetja k skupnosti, pa je to, da se spreobrne v oportunistični pristop. Na ta način lahko podjetje krši osnovne norme, vrednote in pravila, kar negativno vpliva na skupnost.

1.8 Izzivi v odnosu podjetje – skupnost

Veliko je podjetij (IBM, Google, Sun Microsystems), ki se zavedajo, kako veliki so izzivi pri sodelovanju s skupnostjo. Za ustvarjanje in ohranjanje odnosov s skupnostjo morajo biti pripravljena nameniti znatne vire, kot sta čas in denar. Podjetje Google v ta namen organizira srečanje »Summer of Code«. To srečanje je bilo vzpostavljeno iz več razlogov:

1. da bi navdušili mlade razvijalce za sodelovanje pri razvoju odprte kode;
2. da bi študentom računalništva in sorodnih ved zagotovili plačano poletno delo, povezano z njihovimi akademskimi interesi;
3. izpostavitve teh študentov razvojni odprtokodni distribuciji;
4. več pisanih programskih rešitev in
5. za pomoč odprtokodnim projektom, da lahko identificirajo nadarjene nove razvijalce.

Od leta 2005 program oglašuje, da je povezan z več kot 2.500 študenti razvijalci odprtokodnih projektov, ki večinoma ne tekmujejo neposredno z Googlovimi poslovnimi interesi (Google Summer of code, 2009). Sredstva niso edini predmet razprave, saj je treba spoštovati tudi norme in vrednote odprtokodnih skupnosti. Podjetja se lahko soočijo s precejšnjimi težavami, tudi če kršijo pravila in norme, saj so člani odprtokodnih skupnosti na to zelo občutljivi. Kadar se ne sledi etičnemu kodeksu, lahko člani skupnosti preprosto prekinijo sodelovanje ali migrirajo v druge projekte (Tapscott, 2007). Tudi preveč komercialno usmerjene dejavnosti lahko spodbudijo razvijalce, da zapustijo skupnosti. Če pa razvijalci razumejo, da imajo njihove ideje velik komercialni zaslužek, jih lahko to odvrne od tega, da bi brezplačno prispevali. Podjetja naj bi za ohranitev dobrih odnosov s skupnostjo ne uporabljala odprtokodnih programskih rešitev za komercialne namene. Če pa že, bi morala pripraviti jasna poročila, katere kose programskih rešitev uporabljajo iz

različnih skupnosti in kako se jih uporablja v povezavi z interno razvito izvirno kodo (Lerner & Tirole, 2002, str. 230).

Z namenom da bi se skupnost aktivno razvijala, morajo podjetja razmišljati, kako privabiti razvijalce programskih rešitev. Razvijalci niso samodejno zainteresirani in tudi ne samodejno prispevajo k razvoju projekta. Veliko je projektov, ki tekmujejo za pozornost in zanimanje razvijalcev in uporabnikov. Tako morajo podjetja zagotoviti zanimive izzive in zabavne projekte za razvijalce. Poleg tega morajo podjetja imeti možnost, da razvijajo izdelke, ki so v interesu prodaje, uporabnikov in storitev za uporabnike. Pomembno pa je tudi, da so podjetja sposobna uskladiti različne interese o poteku dela, saj ko je dosežen razvojni cilj in postane skupni primarni poudarek prodaja, je vedno težje sodelovati s skupnostjo (Bonaccorsi & Rossi, 2006, str. 16).

Izzivi so zelo povezani z licencami in kontrolami lastništva. V okviru odprtokodnih programskih rešitev so licence pomembne za podjetja, saj določajo in omejujejo uporabo obstoječih modulov programskih rešitev, ki jih je razvila skupnost. To omejuje možnosti za uporabo komponent odprtokodnih programskih rešitev v povezavi z interno razvito izvirno kodo. Kot je bilo že omenjeno, mnogi programerji prispevajo k razvoju odprtokodnih programskih rešitev, da bi pridobili ugled med svojimi vrstniki. To ni mogoče, če podjetje ukrade kodo in jo vstavi v svoje lastniške programe, saj potem razvoj kode ni več pod budnim očesom skupnosti (Bonaccorsi & Rossi, 2006, str. 18). Posledično programerji raje delajo na projektih z GPL licenco, saj ta preprečuje podjetjem krajo njihove kode. Prav tako so pomembna vprašanja lastništva, ko se ukvarjamo s skupnostmi. Interesi podjetij in skupnosti lahko brez težav privedejo do konfliktov v smislu lastništva. Najboljši način za opredelitev lastništva je z uporabo različnih licenc (Lerner & Tirole, 2002, str. 229).

2 ODPRTOKODNI PRISTOPI V PODJETJIH

Podjetja pri izkoriščanju odprtokodnih programskih rešitev uporabljajo različne pristope. Ti se med seboj razlikujejo, saj je odprtokodno gibanje omogočilo nastanek povsem novih poslovnih modelov. V tej študiji bomo predstavili dva različna pristopa pri uporabi odprtokodnih produktov, odprtokodnih praks in odprtokodnega fenomena v širšem pomenu besede.

Pri prvem pristopu podjetja uporabljajo obstoječe odprtokodne programske rešitve in komponente, ki jih integrirajo kot del produkta ali pa jih uporabijo v notranjih poslovnih projektih. Poimenovali smo ga notranji odprtokodni pristop (Fink, 2003, str. 203). Nokia je na ta način izdala mobilni telefon, model N900, ki ima integriran operacijski sistem Linux. Ta pristop je med podjetji najbolj razširjen način uporabe odprtokodnih programskih rešitev, kar je razvidno iz velikega števila komercialnih podjetij, ki danes uporabljajo odprtokodne programske rešitve in tehnologije. Pri tem pristopu večina podjetij zasleduje

zmanjševanje stroškov, čeprav se število tistih podjetij, ki stremijo k boljši kakovosti, tudi povečuje.

Drugi pristop zahteva več proaktivnosti na strani podjetja pri razvoju odprtokodnih programskih rešitev, poimenovali smo ga zunanji odprtokodni pristop (Fink, 2003, str. 192). Pri tem pristopu podjetja izdajo obstoječo lastniško izvirno kodo programskih rešitev. Veliko je podjetij, kot so IBM, HP, Sun, Oracle, ki izdajajo lastniško kodo njihovih programskih rešitev. Prvo večje komercialno podjetje, ki je izdalo lastniško kodo, je bilo podjetje Netscape, ki je leta 1998 izdalo kodo spletnega brskalnika Netscape Navigator, danes brskalnik FireFox podjetja Mozilla. Podjetja lahko izdajo obstoječe programske rešitve kot odprtokodne ali pa ustanovijo odprtokodne skupnosti za razvoj odprtokodnih programskih rešitev. Podjetja se lahko že v fazi razvoja programskih rešitev odločijo, da bodo kasneje sprostila izvirno kodo javnosti. Lahko izdajo izvirno kodo celotne programske rešitve ali samo njen del, pri čemer obdržijo platformo za nadzor zaprto. Na ta način lahko podjetja nadzirajo, kako se bodo programske rešitve licencirale. Pri drugem pristopu podjetja pogosto vzporedno z izdajo izvirne kode njihovih programskih rešitev ponudijo uporabnikom tudi dopolnilne storitve. Na ta način lahko programske rešitve ponudijo brezplačno in dohodke črpajo od prodaje vzporednih storitev.

Pristopa se med seboj ne izločujeta, kar pomeni, da lahko podjetja izkoristijo in uporabijo oba pristopa skupaj. Prvi pristop je še posebej prilagodljiv, saj se velikokrat uporablja v kombinaciji z drugim pristopom. V nadaljevanju bomo podrobneje predstavili oba pristopa.

2.1 Notranji odprtokodni pristopi

Odprtokodne programske rešitve so danes realnost in podjetja to s pridom izkoriščajo, kar je povsem razumljivo. Nedavne študije in ocene (Netcraft, 2008, Paul, 2008, Keizer, 2008) kažejo na vsestransko sprejetje odprtokodnih programskih rešitev v podjetjih in pri fizičnih osebah. Odprtokodni spletni strežnik Apache ima približno 54 % tržni delež (Netcraft, 2008), spletni brskalnik Firefox ima skoraj 30 % tržni delež v Evropi (Paul, 2008) in približno 20 % po vsem svetu (Keizer, 2008), in to kljub temu, da je spletni brskalnik Internet Explorer podjetja Microsoft del operacijskega sistema Windows, ki je nameščen na večini osebnih računalnikov po vsem svetu. Komercialna podjetja lahko znatno prihranijo pri stroških in izboljšajo kvaliteto delovanja, če izberejo odprtokodne programske rešitve namesto komercialnih. Odprtokodne programske rešitve ponujajo zanimivo **alternativo** komercialnim programskim rešitvam. Odprtokodni operacijski sistem Linux je na primer alternativa operacijskemu sistemu Windows, odprtokodna pisarniška orodja Openoffice so alternativa Microsoft Office. Podjetja lahko odprtokodne programske rešitve vključijo v njihove končne izdelke. Zavedati se moramo, da uporaba odprtokodnih programskih rešitev ni tako enostavna in preprosta, kot se zdi na prvi pogled.

Podjetja morajo skrbno premisliti, ali je notranji odprtokodni pristop primeren za njihovo delovanje.

2.1.1 Kdaj je odprtokodni notranji pristop primeren

Fink (2003, str. 203) je sestavil seznam poslovnih razlogov, ko je uporaba notranjega odprtokodnega pristopa smiselna. Prva dva razloga sta povezana s poslovnimi standardi. Fink navaja, da če je implementacija ključnega poslovnega standarda odprtokodna in je pomembna za poslovanje podjetja, potem je uporaba tega standarda dobra stvar. Z uporabo tega standarda se lahko izognemo ponovnemu implementiranju. Enako velja za obstoječo tehnologijo, ki so jo že razvile odprtokodne skupnosti. Uporaba nekaterih prodornih tehnologij odprtokodnih programskih rešitev postaja vse pogostejše učinkovit način za povečanje proizvodnje v prihodnosti. Odprtokodne rešitve so prav tako lahko učinkovit način, da se podjetja osredotočijo na razpoložljiva sredstva z višjo dodano vrednostjo zmogljivosti. Podjetja morajo poskrbeti le, da so licence odprtokodnih programskih rešitev in tehnologija primerni. Nekatere licence zahtevajo od podjetja, da vrne vse spremembe in izboljšave skupnosti, na primer Novel. Zato morajo podjetja zagotoviti, da ni nobenih ovir, če na primer združimo GPL kodo in lastniško kodo. Notranji pristop je tako najprimernejši za podjetja, ki potrebujejo dobro razvite, zelo prilagojene programske rešitve s točno določenim proračunom in rokom izvedbe. To velja tudi za tiste, ki nimajo tehničnih zmogljivosti, da bi prispevali rešitve, in za tiste, ki niso pripravljeni na vključitev v proces razvoja programskih rešitev.

Po mnenju Finka obstajajo okoliščine, ko se je treba izogibati uporabi odprtokodnih programskih rešitev znotraj podjetij in znotraj projektov. Odprtokodni projekti ne smejo zadovoljevati le nujnih potreb, ampak morajo biti skladni z dolgoročnimi cilji poslovanja podjetja. Primernost odprtokodnih programskih rešitev za celotno tehnično arhitekturo podjetja mora biti preveriti in sprejeti tehnično osebje. Podjetja se morajo zavedati, da lahko odprtokodne skupnosti izdajo nove, a nepravilne različice programskih rešitev. Če podjetja potrebujejo popoln nadzor nad rokom izdaje in natančne načrte izdelave, potem notranji odprtokodni pristop ni primeren. Po drugi strani pa lahko podjetja sama prispevajo ali sodelujejo s skupnostjo in na ta način poskušajo odpraviti nepravilnosti in pridobiti vzvod za pravilno izdajo programskih rešitev.

2.1.2 Odprtokodne programske rešitve ali komercialni programi

Odprtokodne programske rešitve so postale popularne v veliko podjetjih, kot so na primer IBM, HP, Sun, Oracle, Asus, Dell. Obstaja veliko prednosti, zaradi katerih se podjetja odločijo za odprtokodne programske rešitve. Z uporabo odprtokodnih programskih rešitev je mogoče izkoristiti njihove prednosti, ne da bi sodelovali s skupnostjo. Sodelovanje s skupnostjo ni obvezno, prav tako ni obvezno zagotavljanje povratnih informacij in

sprememb skupnosti. Pridobivanje odprtokodnih programskih rešitev je zelo preprosto in enostavno, saj si jih lahko brezplačno prenesemo iz številnih spletnih strani, kot sta na primer sourceforge.com in opensource.org. Prost dostop omogoča razvijalcem, da preizkusijo programske rešitve, ne da bi morali skozi formalni postopek nakupa. Odprtokodne programske rešitve je mogoče pogosto dobiti tudi od več dobaviteljev, ki pogosto delujejo na različnih arhitekturah strojne opreme. Tako operacijski sistem Linux deluje na strežnikih podjetja Cisco, kot tudi podjetja HP. To vodi k izbiri in prožnosti za stranke. V nadaljevanju bomo primerjali odprtokodne programske rešitve z lastniškimi programskimi rešitvami v smislu stroškov, varnosti, zanesljivosti, razvoja, vzdrževanja in podpore.

Zmanjševanje stroškov. Najočitnejša stvar, povezana z odprtokodnimi programskimi rešitvami, je ta, da je večina od njih brezplačnih. Začetni stroški odprtokodnih programskih rešitev se gibljejo od nič navzgor. Odprtokodne programske rešitve so v nasprotju z lastniškimi na voljo brezplačno, ponujajo prost dostop do izvorne kode in nadaljnjo brezplačno pregledovanje in distribuiranje. Vendar pa je prepričanje, da nastanejo odprtokodne programske rešitve brez stroškov, napačno. Izraz »brezplačno« je namenjen izražanju »svobode« in ne »brez stroškov«. Začetni stroški nakupa odprtokodnih programskih rešitev so res lahko enaki nič, a se to hitro spremeni, ko začnemo računati vse stroške, ki jih ima podjetje pri njihovi uporabi, in ugotovimo, da to ni tako preprosto vprašanje. Nabavna vrednost predstavlja le majhen del vseh stroškov, ki jih ima podjetje pri uporabi odprtokodnih programskih rešitev. Izračun vseh stroškov, ki jih ima podjetje pri uporabi odprtokodnih programskih rešitev zahteva upoštevanje številnih področij: nakup programske opreme, postavitve (vgradnja), vzdrževanje, stroški nadgradnje, nakup strojne opreme, stroški vzdrževanja, stroški podpore in administrativni stroški. Čeprav se številna podjetja odločajo za uporabo odprtokodnih programskih rešitev zaradi zmanjšanja stroškov, je treba opozoriti, da začetni stroški pridobitve ne odtehtajo drugih stroškov. Zato ni nujno, da so brezplačne programske rešitve v smislu vseh stroškov cenejša izbira od lastniških programskih rešitev (Fitzgerald, 2006, str. 595).

Varnost in zanesljivost. Veliko je razprav o varnosti in zanesljivosti odprtokodnih programskih rešitev. Prodajalci lastniških programskih rešitev na primer financirajo, producirajo in objavljajo poročila, ki podpirajo trditev, da zaprti lastniški sistemi ponujajo vrhunsko varnost v primerjavi z odprtokodnimi programskimi rešitvami. Zagovorniki lastniških programskih rešitev menijo, da so le-te boljše, saj izvorna koda ni na voljo, kar pomeni, da je varnostne luknje izjemno težko ali celo nemogoče najti. Vendar so številne študije pokazale, da so te izjave netočne, in številne študije kažejo, da bi bilo treba povečati varnost s pomočjo odprte kode. V primeru lastniških programskih rešitev morajo podjetja zaposlovati tiste, ki bodo testirali in razvijali. Tudi podjetja, ki so pripravljena nameniti sredstva za testiranje proizvodov, ne morejo natančno simulirati vseh pogojev, pod katerimi se bodo ti uporabljali. Proizvodi odprtokodnih programskih rešitev pa so

preizkušeni na veliko večjem naboru pogojev, s pomočjo globalne skupnosti uporabnikov. Na primer Hoepman in Jacobs (2007, str. 80) trdita, da je uporaba odprtokodnih programskih rešitev nujen pogoj, da bi gradili sisteme, ki bi bili bolj varni. Po njihovem mnenju odprtje izvorne kode omogoča neodvisno oceno izpostavljenosti sistema in tveganje, povezano z uporabo sistema, omogoča popravljanje napak na lažji način in sili razvijalce programskih rešitev, da bi več napora namenili kakovosti kode. Proces razvoja odprtokodnih programskih rešitev omogoča proizvodnjo bolj varne programske opreme, ker se opredeljene varnostne težave sporočijo v skupnost in so odpravljene v veliko krajšem času od zaprtih programskih rešitev. Zato širši in lažji dostop do izvorne kode dejansko poveča verjetnost za odkrivanje in odpravljanje napak in težav. Obstaja tudi veliko razprav o vprašanjih zanesljivosti odprtokodnih programskih rešitev. Odprtokodne skupnosti vsebujejo praktično neomejeno število razvijalcev, ki lahko dostopajo do izvorne kode, odpravljajo napake in dodajajo izboljšave. Razlog, zakaj bi odprtokodne programske rešitve bile bolj zanesljive kot lastniške programske rešitve, najbolje obrazloži Raymond (2001, str. 30): »Zadostno število oči in vse napake so odpravljene«. Raymond ugotavlja, da se pomanjkljivosti odkrijejo in določijo hitreje pri odprtokodnih projektih kot v zaprtih lastniških projektih. Prav tako Raymond ugotavlja, da pri odprtokodnih projektih na splošno nastane manj napak kot pri zaprtih projektih in pomanjkljivosti so odkrite in odpravljene v zelo kratkem času.

Kljub argumentom glede varnosti in zanesljivosti odprtokodnih programskih rešitev ni dokončnega odgovora, ali so odprtokodne programske rešitve bolj varne in zanesljive kot lastniške programske rešitve. Zagotovo obstajajo lastniški sistemi, ki so bolj varni in zanesljivi kot odprtokodni, obstajajo pa tudi odprtokodni izdelki, ki so boljši od lastniških. Analitični argumenti niso prepričljivi za nobenega od pristopov. Na splošno so tako odprtokodni kot lastniški sistemi približno enaki v smislu varnosti in zanesljivosti.

Razvoj, vzdrževanje in podpora. Odprtokodne programske rešitve se za razvoj in vzdrževanje opirajo na razvojno skupnost. Če se v razvojno skupnost aktivno vključi veliko število razvijalcev, je lahko razvoj programskih rešitev zelo hiter in kakovosten. S pogostimi izdajami in s pomočjo velikih skupnosti je mogoče odkriti in določiti napake zelo hitro. Poleg tega je lahko vzdrževanje takojšnje, če je skupnost aktivna in redno izdaja nove različice. Tako lahko programske rešitve, ki jih razvijajo odprtokodne skupnosti, prekašajo svoje komercialne nasprotnike.

Tehnična podpora je običajno del lastniške programske opreme. Uporabnik se s težavo obrne na podporni center, prodajalec pa mu je dolžan nuditi pomoč. Vendar pa so stopnje podpore vsebinsko zelo slabe in počasne, saj mnoga podjetja zmanjšujejo sredstva za pomoč strankam. Institucionalni uporabniki odprtokodnih programskih rešitev imajo na voljo tri možnosti za podporo. Prva možnost je zagotoviti podporo znotraj podjetja. Včasih se to šteje za preveč tvegano, še posebej v primerih, ko ni pravih internih izkušenj

(Wheeler, 2007). Mnoga podjetja, kot so IBM, SUN, Red Hat, nudijo podporo in usposabljanje za odprtokodne programske rešitve. Podjetja, ki ponujajo podporo za nekatere odprtokodne programske rešitve, so prav tako lahko del razvojne skupnosti. Podjetja, ki zaposlujejo razvijalce skupnosti jasno dokazujejo dobro razumevanje razvoja programske opreme in lahko potencialno hitro odpravijo težave. Tretji način za organiziranje podpore je biti odvisen od razvoja in uporabnikov skupnosti (Wheeler, 2007).

Vzdrževanje in podpora se med produkti zelo razlikujeta, predvsem sta odvisni od skupnosti, ki stoji za produktom. Lahko sta hitri in koristni, če je skupnost aktivna in pripravljena odgovoriti na zastavljena vprašanja, lahko pa skupnost ne uspe zagotoviti rednih novih izdaj in podpore oziroma niti ni dolžna zagotoviti ne vzdrževanja ne pomoči. Zato bi morala podjetja oceniti skupnosti pred sprejetjem odprtokodnih programskih rešitev. Pri ocenjevanju skupnosti morajo biti podjetja pozorna na število projektov, kakovost dokumentacije, tekoče aktivnosti, nove različice, vtičnike, udeležbo v skupnostih, hitrost odgovorov na zastavljena vprašanja (Madanmohan, 2004, str. 67).

2.1.3 Odprtokodne programske rešitve za poslovne namene

Februarja 2009 je bilo v skupnosti sourceforge.net registriranih več kot 230.000 projektov odprtokodnih programskih rešitev z več kot 2 milijona registriranih uporabnikov (Sourceforge.net, 2010). Za te projekte velja, da razvijajo eno specifično rešitev ali komponento, ki se razlikuje v stopnji razvoja in zrelosti. Na splošno je mogoče trditi, da je za vsak izdelek lastniških programskih rešitev na voljo ustrezna odprtokodna programska rešitev. Prav tako je na voljo veliko odprtokodnih programskih rešitev za poslovno rabo, čeprav jih razvijalci odprtokodnih programskih rešitev ne uvrščajo na prvo mesto. Številne aplikacije, kot so Linux, Apache, Mysql, so primerne tako za domačo kot za poslovno rabo. Poleg tega pa obstajajo številne odprtokodne programske rešitve, kot so na primer SugarCRM, Compiere ERP, CRM Business Solution, ki so razvite samo za poslovno rabo. Te v večini primerov vodijo podjetja s komercialnimi interesi (Rotz, 2007).

Rotz (2007) razvrsti programske rešitve v štiri različne kategorije: **operacijski sistemi in infrastruktura, razvoj aplikacij in infrastrukture, infrastrukturne rešitve in poslovne aplikacije**. Vsaka od teh kategorij zajema veliko odprtih orodij, ki izzivajo komercialna orodja. **Operacijski sistemi in infrastruktura** vključujejo veliko odprtokodnih izdelkov, ki so celo nadomestki za tržne proizvode. Mednje spadajo na primer Linux operacijski sistem, spletni strežnik Apache in spletni brskalnik Firefox. **Razvoj aplikacij in infrastrukture** zajema orodja, okvire in sestavne dele, potrebne za izvajanje in delovanje aplikacij: na primer programski jezik php, MySQL in PostgreSQL podatkovne baze, strežnik Apache, aplikacijski strežnik JBoss, Tomcat Servlet motor, razvojno okolje Eclipse, CVS Subversion kontrolni sistem za revizijo. **Infrastrukturne rešitve** običajno ne

koristijo končnemu uporabniku, ampak so vzvod za druge rešitve. Pomagajo podjetjem pri upravljanju z vsebinami, z identitetami, z dostopom, z rešitvami za telefonijo. Odprtokodne programske rešitve v tej kategoriji so Typo3 in Joomla za upravljanje z vsebinami, MediaWiki rešitev za sodelovanje pri urejanju enciklopedije in trixbox, Asterisk platforma za VOIP klice. Med **poslovne aplikacije** spadajo programske rešitve, kot so Compiere ERP + CRM Business Solution, SugarCRM, pisarniški paket OpenOffice in Gimp program za slikovno manipulacijo.

Iz zgoraj navedenih primerov je razvidno, da lahko odprtokodne programske rešitve ponujajo pravo alternativo tudi poslovnim rešitvam. Nekaterim odprtokodnim aplikacijam uspeva konkurirati in nekaterim celo nadomestiti komercialne tekmece. Po drugi strani pa se na trgu pojavlja vedno večje število odprtokodnih programskih rešitev, kar pomeni tudi izzive za komercialna podjetja. Podjetja morajo ugotoviti, katere aplikacije obstajajo, katere so tiste najbolj uspešne, kako dobro so podprta, katere funkcionalnosti ponujajo in kako se jih lahko povezuje z drugimi odprtokodnimi ali lastniškimi programskimi rešitvami.

2.1.4 Uporaba odprtokodnih programskih rešitev v organizacijah

Walli, Gynn in Rotz (2005) so izvedli raziskavo z zbiranjem podatkov s pomočjo spletne ankete, katere spletni naslov je bil razdeljen med več kot 40.000 strokovnjakov informacijske tehnologije in med 512 poslovnih podjetij v ZDA. Njihov namen je bil ugotoviti razširjenost odprtokodnih programskih rešitev v podjetjih v ZDA. Ugotovili so, da velika večina (87 % od 512) podjetij uporablja odprtokodne programske rešitve. Vsa velika podjetja (s prihodki nad 1 milijardo dolarjev) so uporabljala odprtokodne rešitve. Med srednje velikimi in velikimi organizacijami (prihodki več kot 50 milijonov dolarjev) je zelo razširjena uporaba odprtokodnih operacijskih sistemov za poslovne namene in spletnih brskalnikov. 74% vprašanih ima Linux operacijski sistem in 70 % jih uporablja odprtokodni brskalniki Firefox. V velikih organizacijah najpogosteje uporabljajo (58 %) orodja odprtokodnih rešitev za razvoj programskih rešitev (odprtokodni programski jezik, na primer php) in (57 %) aplikacijske strežnike (na primer Apache). Tudi sistemi za upravljanje podatkovnih baz (na primer MySQL) so bili v uporabi v večini podjetij (51 %). Najbolj priljubljena orodja v srednje velikih podjetjih so sistemi za upravljanje podatkovnih baz in orodja za razvoj programskih rešitev. Razlike so se pojavile tudi med različnimi poslovnimi subjekti. Telekomunikacijska podjetja zelo dobro izkoriščajo odprtokodne rešitve, medtem ko je raziskava za finančne institucije, kot so banke in zavarovalnice, pokazala najnižjo stopnjo uporabnosti. Drugi poslovni subjekti, kot so industrija, vlada, zdravstvo, informacijska, strojna oziroma programska oprema, pa so se uvrstile v zgoraj navedeni panogi.

Ghosh in Glott (2005, str. 8) sta izvedla raziskavo v javni upravi v 13 evropskih državah in ugotovila, da 78 % uporablja odprtokodne programske rešitve. Prav tako sta ugotovila, da obstaja povpraševanje po produktih odprtokodnih programskih rešitev: 70 % uporabnikov odprtokodnih programskih rešitev in 38 % neuporabnikov želi povečati uporabo odprtokodnih programskih rešitev in 22 % vprašanih si želi v celoti zamenjati lastniške programske rešitve na svojih sistemih z odprtokodnimi programskimi rešitvami. Odprtokodne programske rešitve se pogosto uporabljajo za delna opravila, kot dopolnilo, in ne kot osnovna programska rešitev. Uporaba odprtokodnih programskih rešitev kot osnovne programske rešitve je bila zelo redka. Odprtokodne programske rešitve igrajo pomembno vlogo v povezavi s spletnimi aplikacijami, kjer imajo ključno vlogo na področju operacijskih sistemov in aplikacijskih orodij, medtem ko je pri stacionarnih računalnikih še precej neizkoriščenega potenciala. Najbolj priljubljeni izdelki so bili GNU/Linux operacijski sistem, podatkovna baza MySQL, spletni strežnik Apache in spletni brskalnik Firefox.

2.2 Zunanji odprtokodni pristopi

Drugi odprtokodni pristop (za podjetja) je tako imenovan zunanji odprtokodni pristop. Ta se nanaša na programske rešitve, ki se trenutno prodajajo pod lastniškimi licencami in prehajajo na odprtokodne licence (Fink, 2003, str. 192). Na ta način je podjetje Nokia izdalo izvorno kodo operacijskega sistema Symbian in ustanovilo skupnost za razvoj. Podjetja pričnejo z odprtokodnimi projekti z izdajo izvorne kode obstoječih programskih rešitev skupnosti ali z vzpostavitvijo odprtokodne skupnosti za razvoj programskih rešitev. Izdaja izvorne kode je podlaga za nadaljnji razvoj programskih rešitev. S tem lahko podjetja kontrolirajo licenciranje programske rešitve, saj podjetje, ki je interno razvilo projekt programskih rešitev z izdajo svoje izvorne kode javnosti, uporabi odprtokodne licence.

Zunanji odprtokodni pristop ponuja zanimive možnosti za programerska podjetja. Namesto da zaprto izvorno kodo zadržujejo, jo lahko v celoti ali le delno sprostijo v javnost. Podjetja lahko na ta način pridobijo posredne dohodke s prodajo sorodnih produktov. West (2003, str. 1278) je preučeval različne odprtokodne strategije v okviru platform. Po njegovem mnenju imajo podjetja raje strategijo lastniških platform, ker zagotavljajo boljše marže in boljše ovire za imitiranje. Ugotavlja, da je lastniška zaprta strategija na voljo le enemu ali dvema vodilnima trgoma. Za mnoge razvijalce programskih rešitev uporaba strategije lastniških platform postane nemogoča zaradi kombinacije tehničnih in ekonomskih razlogov. Tako je odprtokodna strategija dobra priložnost za številne razvijalce, saj odpravlja prodajne možnosti konkuriranja na podlagi odprtokodnih implementacij, vse dokler je izvorna koda vsem na voljo. Podjetjem ni treba v celoti izdati izvorne kode, West predlaga dve različni hibridni odprtokodni strategiji: **Odprtje določenih delov** in **delno odprtje**. Če se podjetje odloči za **odprtje določenih delov**

programskih rešitev, nad njimi spremeni nadzor in ohrani popoln nadzor nad drugimi deli, ki domnevno zagotavljajo večje možnosti za diferenciacijo in konkurenco. Če se pa podjetje odloči za **delno odprtje**, s tem razkrije določene dele programskih rešitev in tako zagotovi dodatno vrednost za stranke, istočasno pa oteži konkurentom neposredno vzpostavitev. V nekaterih primerih, ko so izdani le deli programskih rešitev in ključni deli ostanejo lastniški, projekti zunanjega odprtokodnega pristopa spominjajo na lastniški razvoj. Ta strategija zahteva ustrezno uporabo licenc, na primer dvojno licenciranje (na primer MySQL).

2.2.1 Kdaj je odprtokodni zunanji pristop primeren

Lerner in Tirole (2002, str. 225) sta nanizala dva pogoja za ugotovitev, kdaj je smiselno, da podjetja razvijajo lastniške rešitve pod odprtokodnimi licencami. Prvi pogoj je, da podjetja s tem povečajo dohodke na dopolnilnih segmentih, kot so svetovanje, vzdrževanje in izobraževanje. Drugi pogoj pa je ta, da ustvarijo vsaj toliko dohodka, kot bi ga ustvarili, če bi razvojno kodo obdržali lastniško. Lerner in Tirole menita, da je ta pristop zanimiv predvsem za manjša podjetja, ki ne uspejo konkurirati na njihovem primarnem trgu, ali ko zaostajajo za vodilnimi in jih to lahko privede do prenehanja delovanja na tem trgu. Pristop je primeren tudi za podjetja, katerih cilj je izboljšati svoj konkurenčni položaj s pomočjo skupnosti, ali pa za podjetja, ki preprosto potrebujejo zunanjo pomoč za razvoj in prilagoditev programskih rešitev.

Fink (2003, str. 193) je predstavil nekaj dejavnikov, ki jih je potrebno upoštevati pred uporabo odprtokodnega zunanjega pristopa. Po njegovem mnenju je odprtokodni zunanji pristop primeren za podjetja, ki imajo na trgu občutno slabšo ponudbo od konkurence. Za izboljšanje konkurenčne prednosti lahko podjetja ponudijo njihove rešitve brezplačno. Odprtokodni zunanji pristop omogoča podjetjem, da lahko konkurirajo podjetjem z lastnimi rešitvami in ne kot eden od kolaborantov. Če podjetja sodelujejo pri razvoju in nastopijo s skupnim produktom na trgu, obstaja možnost da bo produkt uspešen. Zavedati se je potrebno, da podjetja, ki uporabljajo ta pristop, morajo biti prepričana, da imajo dobre možnosti, da bodo imela koristi od tega in da nobeno drugo podjetje ne more monopolizirati ali si prisvojiti vseh zaslug (Markus et al., 2000, str. 24).

Odprtokodni zunanji pristop je primeren tudi za podjetja, katerih glavna dejavnost niso samo programske rešitve. Če je primarna dejavnost vezana na strojno opremo, prodajo storitev in prodajo podpore, ima smisel izdati izvirno kodo. V sodelovanju s skupnostjo lahko podjetja nadaljujejo z zniževanjem stroškov in na ta način lažje dočakajo boljše čase na trgu. Vendar to ni vedno tako in zato morajo podjetja izvesti natančne izračune stroškov in oceniti potencial skupnosti. Po mnenju Fink (2003, str. 194) je lahko odprtokodni zunanji pristop koristen, če ima podjetje temeljno infrastrukturno tehnologije, ki omogoča razvoj drugih izdelkov in rešitev v portfelju podjetja. Odprtokodne programske rešitve se

lahko uporabljajo kot metoda za podjetja, ki ustvarjajo prodorne tehnologije ali pa sprejemajo to kot standard. Odprta koda ima lahko tudi spodbujevalni efekt za sodelovanje in se lahko uporablja kot način za delo s partnerji in tekmeci na zelo velikih projektih.

Fink (2003, str. 196) predstavlja tudi nekatere okoliščine, ko izbira odprtokodnega zunanjega pristopa ni dobra. Po njegovem mnenju podjetja ne bi smela izdati izvirne kode, če so njihovi produkti boljši od konkurence ali pa z njimi ustvarjajo velike ovire za vstop konkurentov. Podjetja morajo narediti tudi izračun koristi, stroškov in notranjih virov za ta pristop. Če odprtokodni pristop ne ustvarja večjih dohodkov in zahteva velike količine notranjih virov, potem morda ne bo optimalna rešitev. Po mnenju West-a (2003, str. 1279) predstavlja odprtokodni zunanji pristop tudi nekatere **tehnične, motivacijske in pravne izzive** za podjetja. **Tehnični izzivi** se pojavijo, ko zunanja skupnost ni seznanjena s spremembami na obstoječi tehnologiji in le-ta nima možnosti biti del razvojne skupine od začetka življenjskega cikla produkta. Lahko se pojavijo tudi težave pri motivaciji, če člani skupnosti ne ustvarijo občutka lastništva tehnologije. **Motivacija** je pogosto ključni element za člane skupnosti in se v veliko primerih tudi razvije, saj so običajno razvijalci v večini primerov del razvoja od samega začetka. Projekti odprtokodnega zunanjega pristopa se lahko soočijo tudi z bitko za nadzor projekta v prihodnosti, če se vizija, cilji in prednostne naloge razlikujejo med podjetjem in skupnostjo. Podjetja morajo skrbno preučiti, pod katerimi odprtokodnimi licencami bodo izdali programske rešitve, saj si člani skupnosti lahko želijo imeti **pravne** pravice do uporabe, spreminjanja in distribuiranja kode, ki so jo pomagali razviti.

2.2.2 Koristi od odprtokodnega zunanjega pristopa

Podjetja lahko pri razvoju ogromno pridobijo z uporabo odprtokodnih programskih rešitev. Vendar pa je treba poudariti, da so koristi dolgoročne, potreben je čas, da odprtokodni sistem doseže samozadostni nivo. Podjetja z uporabo odprtokodnega zunanjega pristopa pridobijo naslednje koristi (Bonaccorsi & Rossi, 2006, str. 5): **zmanjšanje razvojnih stroškov, boljšo distribucijo produktov in izboljšanje kakovosti produktov.**

Odprtokodni zunanji pristop pripomore k **zmanjšanju razvojnih stroškov**, ker člani odprtokodnih skupnosti prispevajo k razvoju (Bonaccorsi & Rossi, 2006, str. 14). Skupnosti lahko poleg razvoja izvirne kode ustvarjajo tudi nove in inovativne ideje. Naslednja korist, ki jo pridobimo z izdajo izvirne kode, je **boljša distribucija produktov**, saj ni stroškov za pridobitev odprtokodnih programskih rešitev. Podjetja, ki izdajo izvorno kodo, lahko izboljšajo konkurenčni položaj s pomočjo aktivnega razvoja skupnosti ter z izdajo alternativnih rešitev z nizkimi stroški, ki pritiskajo na konkurente, da bi znižali svoje cene (Bonaccorsi & Rossi, 2006, str. 15). Konkurenti z lastniškim razvojem pa ne morejo izkoriščati odprtokodnih komponent za njihove izdelke zaradi licenčnih omejitev odprtokodnih programskih rešitev. **Izboljšanje kakovosti produktov** je naslednja

ugodnost, ki jo pridobimo z odprtokodnim zunanjim pristopom, ker le dobre programske rešitve vodijo do večjih skupnosti, klahko razvijajo rešitve za daljno prihodnost (Bonaccorsi & Rossi, 2006, str. 18). Bonaccorsi in Rossi (2006, str. 22) menita, da odprtokodni zunanji pristop pomaga še posebej malim podjetjem, ki jim ni treba narediti celotnega razvoja z lastnimi sredstvi. Če mala podjetja niso plačilno sposobna za ureditev celotnega razvojnega in inovacijskega dela, lahko izdajo izvorno kodo in pustijo skupnosti določen del razvoja. Sodelovanje v odprtokodnih projektih spodbuja zanimanje širše javnosti, iz katere lahko podjetja pripravijo strategijo za izboljšanje celostne podobe podjetja. Za uspeh podjetij v odprtokodnem zunanjem pristopu sta potrebna čas in denar. Treba je vključiti marketing in zaposliti potencialne razvijalce, ki integrirajo svoje prispevke v projekte in razvijajo takšne programske rešitve, ki so kompatibilne tako s komercialnimi kot z nekomercialnimi rešitvami. Podjetja se morajo zavedati, da niso vsi odprtokodni projekti uspešni (Lerner & Tirole, 2001, str. 824). Člani skupnosti lahko svobodno izbirajo projekte, v katerih želijo sodelovati. Če podjetja izdajo programske rešitve, ki ne spodbujajo zanimanja med programerji, obstaja veliko tveganje, da projekt ne bo dobil nobenega prispevka od skupnosti.

3 POSLOVNI MODELI

Poslovne modele je mogoče razumeti kot abstraktne predstavitve poslovnih logik podjetij (Osterwalder, 2004, str. 14). Poslovna logika določa, kaj podjetja ponujajo in komu oziroma kako ustvarjajo prihodke. Na poslovne modele lahko gledamo tudi kot na posrednike med strategijami in poslovnimi procesi podjetij. Poslovni modeli so prevodi strategij v načrte podjetij, kako pridobiti denar. Poslovni procesi pa določajo, kako se izvajajo poslovni modeli v praksi (Osterwalder, 2004, str. 15). V literaturi obstaja veliko različnih definicij poslovnih modelov. Poleg tega obstaja tudi veliko načinov formalizacije poslovnih modelov. V tem poglavju bomo predstavili različne definicije in strukture poslovnih modelov.

3.1 Koncept poslovnih modelov

Izraz poslovni model ni nov pojem, saj se je pojavil že leta 1950, ko so iz podjetij McDonald's in Toyota prišli novi poslovni modeli. Leta 1960 so bili inovatorji Wal-Mart in Hipermarketi, 1970 so se pojavili novi poslovni modeli v podjetjih FedEx in Toys R Us, 1980 pri podjetjih Blockbuster, Home Depot, Intel in Dell Computer, 1990 pa pri podjetjih Southwest Airlines, Netflix, eBay, Amazon.com, in Starbucks (Business model, 2010). Uporaba izraza se je pogosteje začela uporabljati leta 1990 in osvojila naklonjenost strokovne literature v tem tisočletju. Izraz je mogoče najti tudi v številnih različicah, kot so »novi poslovni modeli«, »e-poslovni modeli« in »internetni poslovni modeli« (Osterwalder, 2004, str. 23).

Opredelitve poslovnih modelov se med seboj precej razlikujejo in zajemajo zelo širok spekter opredelitev, od tega, kako podjetja ustvarjajo dohodke, do strukture samih podjetij. Osterwalder meni, da se izraz poslovni model velikokrat uporablja za izražanje le enega dela poslovnega modela – na primer dohodkovnega modela, organizacijske oblike itd. Poleg tega Osterwalder ugotavlja, da ljudje, ko govorijo o poslovnih modelih, v resnici govorijo o treh različnih stvareh: o **komponentah poslovnih modelov, operativnih poslovnih modelih in o spremenljivih poslovnih modelih**. Komponente poslovnih modelov so deli aktualnega poslovnega modela, glavni cilj operativnega poslovnega modela je ustvarjanje vrednosti, glavni cilj spremenljivega poslovnega modela pa je ugotoviti, kako se bo podjetje spreminjalo skozi čas in ostalo profitabilno. Pri tako širokem spektru pa pogosto prihaja tudi do neustreznih opredelitev poslovnih modelov. Napačne opredelitve zasledimo v raziskavi o industriji, kjer se je izraz poslovni model uporabljal kot poslovno-procesni model (Gordijn & Akkermans & Vliet, 2000, str. 46).

Nejasnosti se pojavljajo tudi pri enačenju poslovnega modela s poslovno strategijo, ko podjetja enostavno povzemajo iz poslovnih strategij poslovne modele in jih predstavljajo kot načrte za pridobivanje denarja. Združevanje strategij, poslovnih modelov in procesnih modelov Osterwald razlaga kot podobne probleme na različnih nivojih (Osterwalder, 2004, str. 14).

Slika 2: Združevanje strategij, poslovnih modelov in procesnih modelov na različnih nivojih



Vir: A. Osterwalder, *The Business Model Ontology*, 2004, str. 14.

V kontekstu e-poslovanja, e-trgovanja in e-tržnic so poslovni modeli tisti, o katerih se največ razpravlja, a imajo najmanj razumljive vidike in pogoje (Alt & Zimmerman, 2001; Rappa, 2004). Po mnenju Alta in Zimmermana (2001, str. 4) je razumevanje poslovnih modelov pogosto nespecifično in implicitno (nenatančno). Meni tudi, da znanstvene in neznanstvene publikacije podajajo preveč široko razumevanje poslovnih modelov. Z razcvetom interneta je postala beseda poslovni model zelo popularna, kar se je izkazalo v napihnjenem konceptu poslovnih modelov, ki so s časom postali staromodni, a razlog ni bil v konceptu, temveč v njegovem izkrivljanju in zlorabi. Porter (2001, str. 73) pa kritizira koncept poslovnih modelov, ker so nejasni in se sklicujejo na ohlapno pojmovanje tega, kako podjetja poslujejo in ustvarjajo dohodke.

Zaradi raznolikosti definicij smo proučili literaturo obstoječih poslovnih modelov in analizirali koncepte različnih publikacij poslovnih modelov.

Med najbolj znanimi je **Timmersova (1998, str. 4)** opredelitev. Poslovni model opredeljuje kot strukturo tokov proizvodov, storitev in informacij, ki poleg poslovnega procesa določa tudi sodelujoče subjekte in njihove naloge ter način, s katerim bodo sodelujoči subjekti dosegli največjo korist ob uporabi določenih sredstev. Po Timmersovem mnenju poslovni modeli sami po sebi ne zagotavljajo razumevanja, kako uresničiti poslovno poslanstvo. Zato Timmers predstavlja tudi tržni model, ki je kombinacija poslovnega modela in strategije trženja.

Hamel (2000). Poslovni model opredeljuje skozi poslovne koncepte, ki opisujejo ideje, kako poteka poslovanje. Poslovni modeli so preprosto poslovni koncepti, ki so dani v prakso.

Tapscott in Williams (2006). Poslovni model je temeljna struktura dejavnosti, ki določa, kako se razporejajo viri za ustvarjanje diferencirane vrednosti.

Winter in Szulanski (2001). Poslovni model je oblikovanje številnih soodvisnih komponent. Dinamika poslovnega modela nas opozori, ko se strategija življenjskega cikla in internacionalizacija spremenita.

Weil in Vitale (2001). Poslovni modeli so opisi vlog in razmerij med podjetjem, potrošniki, kupci, dobavitelji, ponudniki, ki opredelijo glavne tokove proizvodov, informacij, denarja in koristi za udeležence.

Morris in Allen (2005). Poslovni model je sestavljen iz povezanih nizov za sprejemanje odločitev o komponentah za ustvarjanje konkurenčne prednosti na določenem trgu.

Afuah in Tucci (2003). Poslovni model je eden od treh dejavnikov uspešnosti podjetja. Ostala dva dejavnika sta okolje, v katerem deluje podjetje, in spremembe, ki vplivajo na poslovne modele in njihovo okolje. Če je poslovni model dobro oblikovan, daje podjetju konkurenčno prednost v dejavnosti, v kateri nastopa, in omogoča, da podjetje pridela večji dohodek kot njeni konkurenti. Poslovni model bi moral vsebovati odgovore na vrsto vprašanj: katere cene ponuditi strankam, katerim strankam zagotoviti določeno ceno, kako oceniti ceno, kakšno strategijo izbrati za zagotavljanje določene cene, kako zagotoviti ceno in kako vzdrževati koristi od zagotovljene cene.

Amit in Zott (2001). Poslovni modeli prikazujejo sodelovanje vsebin, struktur in upravljanja za ustvarjanje vrednosti za izkoriščanje poslovnih priložnosti. Poslovni model prikazuje **transakcijo vsebin**, kar pomeni izmenjavo blaga ali informacij s potrebnimi viri

ali zmogljivostmi, ki omogočijo izmenjavo. **Struktura transakcij** se nanaša na stranke, ki sodelujejo pri izmenjavi, in na njihove medsebojne odnose. **Upravljanje transakcij** predstavlja načine za pretok informacij, virov in blaga pod določenimi pravili, s katerimi pridemo do pravne oblike podjetja.

Bossidy, Charan in Burck (2002). Poslovni model je sestavljen iz več komponent, potrebnih za oblikovanje poslovanja. Mogoče jih je kategorizirati v tri skupine: zunanje okolje, finančni cilji in poslovne dejavnosti.

Margretta (2002). Poslovni model je set hipotez, ki pojasnjujejo, kako ustvariti poslovno vrednost za vse udeležence. Poslovni model mora tako odgovoriti na različna vprašanja: kdo je kupec, kakšna je vrednost kupca, način poslovanja in kaj je glavi ekonomski cilj, ki pojasnjuje, kako porazdeliti cene med kupce za primerne stroške?

Shafer, Smith in Linder (2005). Poslovni model predstavlja poslovno temeljno upravljalno logiko, ki se odraža s strateškimi vrednostnimi odločitvami. S pomočjo te strategije se ustvarjajo vrednote in dobiček.

Rappa (2004). Poslovni model lahko preprosto definiramo kot način opravljanja poslov, s katerim se lahko podjetje samo vzdržuje in ustvarja dohodke. Poslovni model določa, kako podjetje pridobiva denar z opredelitvijo njegove pozicije v vrednostni verigi.

Hill in Jones (2008). Poslovni model odraža menedžersko strateško razmišljanje, kako oblikovati strategijo, da bi bilo podjetje integrirano v sistem na način, da bi pridobilo konkurenčno prednost, uspešnost in rast dobička.

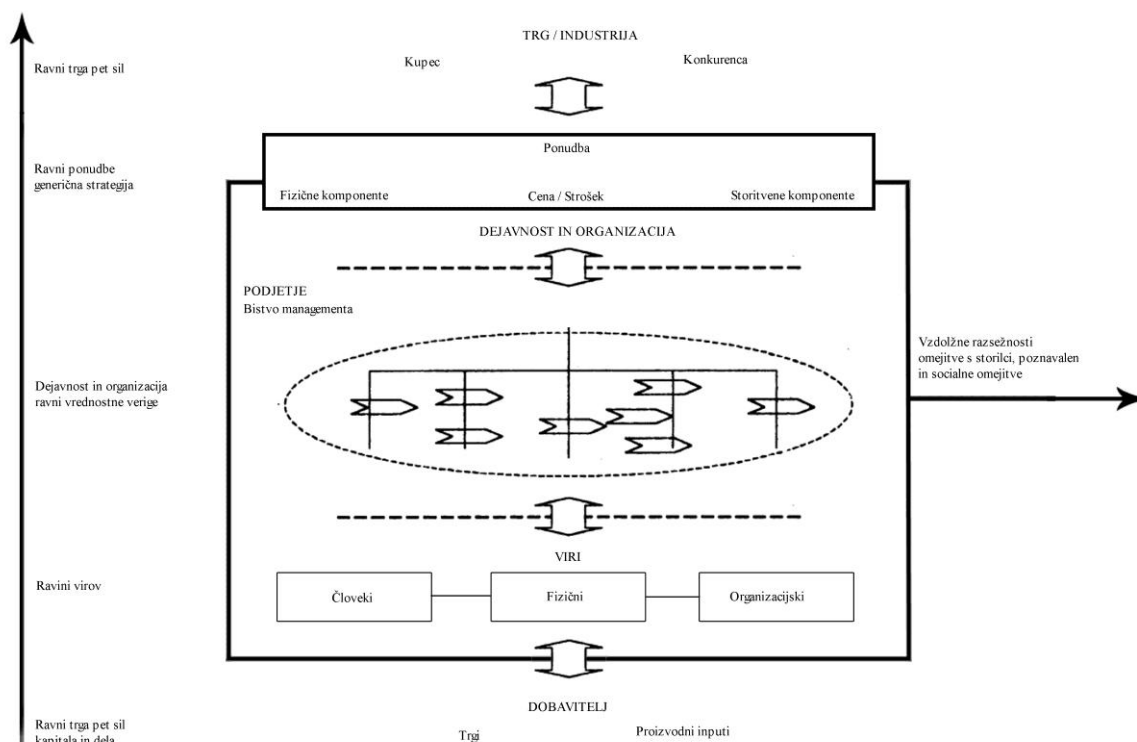
Poslovni model je po našem mnenju analitično orodje, ki omogoča raziskovalcem in strokovnjakom, da bi dobili celosten pogled na poslovanje podjetja. Konceptualno pa si lahko zamislimo poslovni model kot sistem različnih komponent, ki omogočajo podjetjem, da zgradijo ponudbo, uporabljajo različne vire in odnose, s katerimi nudijo strankam in partnerjem boljše storitve ali izdelke kot konkurenti in to odrazijo v dohodkovnem modelu.

3.2 Struktura poslovnih modelov

Veliko avtorjev je opredelilo, kateri elementi sestavljajo strukturo poslovnih modelov. Poslovni modeli se lahko uporabljajo kot orodje za poslovno načrtovanje in menedžerjem pomagajo pri razumevanju in opisovanju poslovne logike njihovega podjetja (Osterwalder, 2004, str. 30). Namen tega dela naloge je iz literature povzeti različne opise komponent, ki tvorijo različne strukture poslovnih modelov. V literaturi se komponente poslovnih modelov zelo razlikujejo glede na njihovo vsebino in pravila. Nekatere strukture so zelo natančne, druge pa so samo sezname različnih konceptov.

Hedman in Kalling (2003, str. 52) predlagata poslovni model, ki vključuje sedem elementov: kupce, konkurente, ponudbo, dejavnost in organizacijo, vire, dobavitelje, proizvodni input in vzdolžni proces. Sedma komponenta, vzdolžni proces, je bila vključena v model za kritje dinamike skozi čas poslovnega modela ter za kritje kognitivnih in kulturnih omejitev, ki jih obvladujejo menedžerji. Na poenostavljen način model združuje podjetje tako, da vključuje notranje vire, ki pretvarjajo faktorje v sredstva ter prek dejavnosti v strukturi ponujajo produkte in ponudbo na trgu.

Slika 3: Grafični prikaz poslovnega modela, ki ga predlagata Hedman & Kalling



Vir: J. Hedman in T. Kalling, *The business model concept: theoretical underpinnings and empirical illustrations*, 2003, str. 53.

Osterwalder (2004, str. 43) predlaga štiri ključne stebre: produkt, uporabniški vmesnik, upravljanje infrastrukture in finančni vidik, ki se naprej delijo na 9 elementov poslovnega modela:

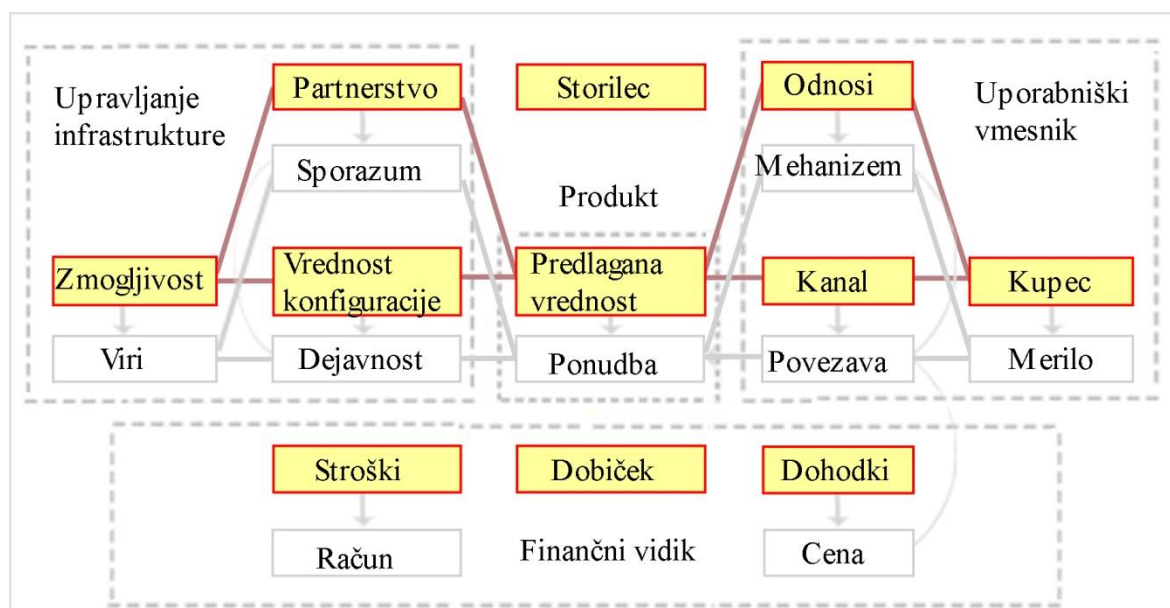
Tabela 4: Stebri in komponente poslovnega modela po Osterwalder-ju

Stebri	Elementi poslovnega modela	Opis
Produkt	Predlagana vrednost	Splošen pregled izdelkov in storitev podjetja, ki predstavljajo vrednost za kupca.

Uporabniški vmesnik	Ciljne stranke	Predstavljajo segment kupcev, za katere podjetje ponudi določeno vrednost.
	Distributivni kanal	Sredstvo za vzpostavljanje stikov s kupci.
	Odnosi	Odnosi opisujejo, kakšno razmerje je vzpostavljeno med podjetjem in stranko.
Upravljanje infrastrukture	Vrednost konfiguracije	Opisuje ureditev dejavnosti in virov, ki so potrebni za ustvarjanje vrednosti za kupca.
	Zmogljivost	Predstavlja možnost izvršitve ponovljivih vzorcev, ki so potrebni za ustvarjanje vrednosti za kupca.
	Partnerstvo	Partnerstvo je prostovoljni kooperacijski sporazum med dvema ali več podjetji za ustvarjanje vrednosti za kupca.
Finančni vidik	Struktura stroškov	Zastopa v denarju vsa sredstva zaposlenih v poslovnem modelu.
	Dohodkovni model	Opisuje, kako podjetje ustvarja denar z različnimi dohodkovnimi tokovi.

Vir: A. Osterwalder, *The Business Model Ontology*, 2004, str. 43.

Slika 4: Grafični prikaz komponent poslovnega modela po Osterwalderju



Vir: A. Osterwalder, *The Business Model Ontology*, 2004, str. 44.

Hamel (2000, str. 20) podaja zelo zanimiv predlog poslovnega modela. Predstavlja ga kot enostavni poslovni koncept, ki je dan v prakso in za katerega razvije številne elemente. Poslovni model, ki ga predlaga Hamel zajema štiri različne elemente: glavno strategijo, podporo strategiji, vrednostno mrežo in odnose s kupci. Ti koncepti so med seboj povezani s tremi mostovi in so naprej razčlenjeni na različne podelemente. Pregled nad temi elementi lahko vidimo na Sliki 5.

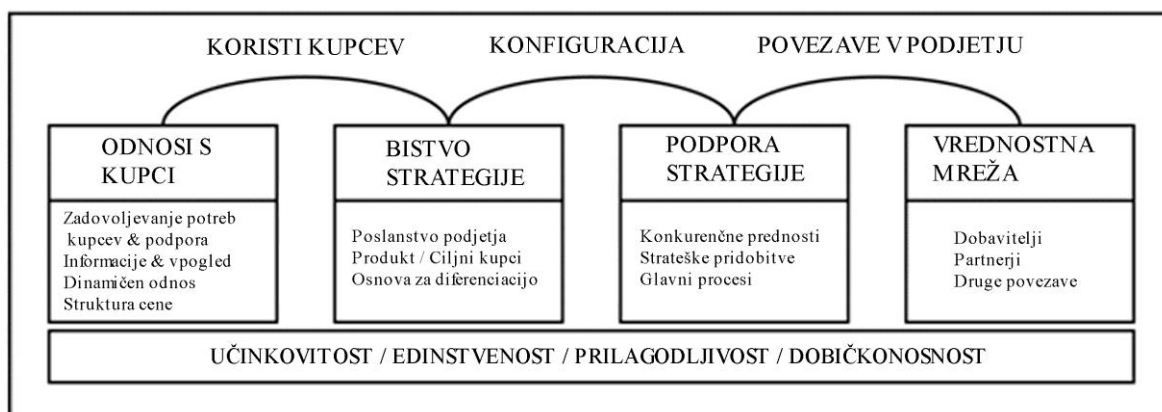
Tabela 5: Komponente poslovnega modela po Hamelu

Elementi poslovnega modela		Opis
Elementi	Bistvo strategije	Definira poslanstvo podjetja, ki opredeljuje, zakaj je ustvarjen poslovni model in kakšni so cilji, ki jih mora doseči. Dalje definira produkt, ciljne kupce in v kakšnem segmentu podjetje tekmuje. Nazadnje pa definira, kako podjetje tekmuje drugače kot konkurenca.
	Podpora strategiji	Ta element vsebuje konkurenčne prednosti podjetja: kaj podjetje ve in njegove edinstvene sposobnosti. Najprej specificira strateške pridobitve, kot so na primer infrastruktura, znamke in patenti. Ta element izpostavi glavne procese v podjetju, razloži, kaj podjetja delajo.
	Odnosi s kupci	Pri tem elementu se združuje več različnih podelementov. Najprej zadovoljevanje potreb kupcev in podpora, s čemer avtor misli, kako in preko katerih kanalov kontaktiramo stranke. Drugič, informacije in vpogled definira vse znanje, ki je zbrano o kupcih. Tretjič, dinamičen odnos se nanaša na naravo vplivanja med proizvajalcem in kupcem. Zadnji podelement je struktura cene, ki nam razlaga, kaj in koliko zaračunamo kupcu.
	Vrednostne mreže	Pod ta element sodi celotna mreža, ki obdaja podjetje, njegove elemente in vire podjetja. Podelementi so: dobavitelji, partnerji in ostale povezave. Partnerji ponavadi dodajajo kritične dopolnitve končnemu produktu ali rešitvi, na drugi strani pa imamo dobavitelje, ki dobavljajo produkte. Pomemben podelement vrednostne mreže so tudi ostale povezave, ki so v nekaterih primerih pomembnejše od dobaviteljev in partnerjev.
ve med ele	Konfiguraci	Bistvo te povezave je, da se osredotoča na

	ja	edinstven način, v katerem se sposobnosti, premoženje in procesi združijo v medsebojnem odnosu za podporo določene strategije.
	Koristi kupcev	Povezava med bistvom strategije in odnosom do kupcev. Definira skupek določenih ugodnosti, ki so ponujene kupcu.
	Povezave v podjetju	Ta povezava se nanaša na odločitve, ki so bile sprejete v podjetju, in odločitve, ki vplivajo na vrednostno mrežo.

Vir: G. Hamel, *Leading the revolution*, 2000, str. 20.

Slika 5: Grafični prikaz koncepta poslovnega modela po Hamelu



Vir: G. Hamel, *Leading the revolution*, 2000, str. 20.

Poleg elementov poslovnega modela in povezav med elementi Hamel opredeljuje še štiri dejavnike, ki vplivajo na zmožnost podjetja, da uspe. Poslovni koncept mora biti učinkovit in edinstven pri zagotavljanju koristi strank. Koncept poslovanja mora biti močan, ustrezati elementom in izkoristiti mora dobiček za ustvarjanje potencialnega nadpovprečnega donosa.

Alt in Zimmerman (2001, str. 5) ločita šest generičnih elementov poslovnih modelov:

Tabela 6: Elementi poslovnega modela po Alt-u in Zimmermann-u

Elementi poslovnega	Opis
---------------------	------

modela	
Vizija	Pomemben del poslovnega modela je razvijanje razumevanja vizije na splošno, strateških ciljev in vrednostnih predlogov, vključno z osnovnimi proizvodi ali storitvami.
Struktura	Struktura določi funkcijo različnim vpletenim agentom. Hkrati se osredotoči na kupce in produkte.
Procesi	Procesi omogočajo podrobnejši pogled na poslanstvo in strukturo poslovnega modela. Pokažejo nam elemente ustvarjanja vrednostnih procesov.
Dohodek	Dohodek je del poslovnega modela, v katerem nas zanimajo stališča v zvezi s profitom in stroški.
Pravne zadeve	Pravne zadeve vplivajo na vse vidike poslovnega modela in vizije na splošno.
Tehnologija	Tehnologija omogoča, da poslovni modeli, ki temeljijo na IT-ju sploh obstajajo. Po drugi strani pa ima tehnologija močan vpliv tudi na oblikovanje poslovnega modela.

*Vir: Povzeto po R. Altu in H. Zimmermannu,
Introduction to Special Section – Business Models, 2001, str. 5.*

Mahadevan (2000, str. 9) je študiral poslovne modele, predvsem tiste, ki temeljijo na internetnem poslovanju. Po mnenju Mahadevana so poslovni modeli edinstvena mešanica treh tokov, ki so ključnega pomena za poslovanje. Ti tokovi so **vrednostni tokovi za poslovne partnerje in kupce, tok dohodkovni in logistični tok**. **Vrednostni tokovi** obravnavajo dolgoročno vzdržnost poslovnega predloga in pogosto določajo okvire za določanje tokov dohodkov podjetja. **Dohodkovni tokovi** so uresničitve vrednostnih predlogov v kratkem času in načrti za zagotavljanje dohodkov podjetja. **Logistični tokovi** se ukvarjajo z različnimi vprašanji, povezanimi z zasnovo dobavne verige poslovanja.

Osnovni koncept Kindströma (2005) je bil reducirati poslovni model na tri bistvene elemente: **strateški položaj, operativno platformo in ponudbo**. **Ponudba** je sestavljena iz produktov in storitev, ki jih podjetje ponuja na trgu. **Operativne platforme** so notranji viri, procesi, strukture in tehnologije, ki jih podjetje uporablja za uspešno delovanje na trgu. To je potrebno načrtovati tako, da lahko podjetje doseže konkurenčno prednost in optimalno delovanje. **Strateški položaj** nam prikazuje, kako trdno smo pozicionirani na trgu z našo ponudbo. Trg je še posebej specifičen v industriji zaradi kupcev, dobaviteljev, prodajalcev in konkurentov. Zato da je podjetje uspešno, mora doseči dinamiko in sinergijo med elementi. Kindström to opiše kot dinamično izmenjavo med strateškim in operativnim položajem.

Prav tako so Petrovic, Kittl in Teksten (2001, str. 5) proučevali poslovne modele v okviru e-poslovanja. Poslovne modele so razdelili v sedem podmodelov: **vrednostni model, model virov, model proizvodnje, model odnosov s strankami, dohodkovni model, model kapitala in tržni model**. **Vrednostni model** opisuje, kateri temeljni proizvodi so

dostavljeni stranki. **Model virov** opredeljuje, kako so elementi potrebni za proces preoblikovanja ter kako jih prepoznati in pridobiti zahtevane količine. **Model proizvodnje** opisuje, kako so elementi združeni v proces preoblikovanja od izvora do rezultata. **Model za odnose s strankami** opredeljuje, kako doseči stranke, jim ustreči in jih ohraniti. Sestavljen je iz treh podmodelov: distribucijskega modela, tržnega modela in storitvenega modela. **Dohodkovni model** določa, kaj, kdaj, zakaj in kako podjetje prejme nadomestilo v zameno za izdelke. **Kapitalski model** opisuje, kako pride do finančnih virov za ustvarjanje dolga, strukture kapitala in kako se ta denar uporablja za sredstva in obveznosti. **Tržni model** opisuje logiko izbire ustreznega okolja, v katerem naj deluje podjetje.

Afuah in Tucci (2003, str. 57) sta raziskala internetne modele in opredelila osem elementov.

Tabela 7: Elementi poslovnega modela po Afuah in Tucci-ju

Elementi poslovnega modela	Opis
Vrednost za kupca	Vrednost za kupca opisuje, v čem se njihov izdelek razlikuje od izdelka, ki ga ponujajo konkurenčni proizvajalci. Ali ga proizvajajo po nižjih cenah?
Obseg aktivnosti podjetja	Podjetje mora definirati, kakšno vrednost ponuja kupcu, kakšen je razpon produktov in storitev, ki so vključeni v to vrednost.
Cenitev vrednosti	Način, kako podjetje oceni svojo ponudbo, strategijo oblikovanja cen.
Vir dohodka	Določajo vire dohodkov in dobička podjetja ter kdo in kdaj plačuje za kakšno vrednost. Definirati morajo tudi marže na vsakem trgu ter ugotoviti smisel njihovega delovanja.
Povezanost aktivnosti	Povezanost aktivnosti nam pove, katere aktivnosti in kdaj jih moramo izvesti, da lahko podjetje ponudi večjo vrednost kupcu. Pojasni nam, kako so dejavnosti povezane.

Izvedba	Določa organizacijsko strukturo, sistem (na primer sistem nagrajevanja in sistem za pretok informacij), ljudi in okolje za izvedbo izdelka ali storitve. Obenem pa mora biti definiran tudi odnos med vsakim izmed prej naštetih elementov posebej.
Sposobnost	Sposobnost determinira, katera sredstva (opredmetena, neopredmetena ali človeški viri) podjetje potrebuje, ko gradi vrednost kupca. Poglavitne sposobnosti podjetja lahko ustvarijo konkurenčno prednost, saj omogočajo podjetju, da ponudi svojim strankam večjo vrednost kot konkurenti.
Konkurenčnost	Določa, kako zagotoviti, da bodo tudi v prihodnje ustvarjali denar in ohranjali konkurenčno prednost.

Vir: A. Afuah in C. Tucci, *Internet Business Models and Strategies*, 2003, str. 57.

3.3 Poslovni modeli, ki za osnovo uporabljajo odprtokodne programske rešitve

V literaturi obstaja le manjše število opisov poslovnih modelov, ki temeljijo na odprtokodnih programskih rešitvah, vendar pa so v primerjavi s splošnimi strukturami še v razvoju. Zaenkrat so to strogi opisi različnih načinov uporabe odprtokodnih programskih rešitev. V predhodnem poglavju smo predstavili splošne poslovne modele in komponente, s katerimi je mogoče opisati in opredeliti strukture poslovnih modelov. Uporabljamo jih lahko za analizo različnih poslovnih modelov ter za ocenjevanje, ali so poslovni modeli za podjetja primerni ali ne. Za našete strukture je irelevantno, ali poslovni modeli temeljijo na odprtokodnih programskih rešitvah ali ne. Seveda ima vsak poslovni model svoje značilnosti in je primeren za analizo točno določenih podjetij iz točno določenih perspektiv. Vse našeto je treba upoštevati pri odločanju o tem, katero strukturo bomo uporabili.

Rast in interes za poslovne modele, ki temeljijo na odprtokodnih programskih rešitvah sta ključna faktorja za nastanek nekaterih opisanih struktur poslovnih modelov, ki so več kot le razvojne strategije za podjetja. Najvplivnejša klasifikacija je prav gotovo Hecker's, ki temelji na Raymondovem delu in na Koenigovih možnih taksonomijah odprtokodnih strategij. Po analizi 120 različnih odprtokodnih poslovnih modelov podjetij jih je Daffara razvrstil v 6 glavnih skupin. Odprtokodni poslovni modeli imajo številne podobnosti s Tapscottovimi inovativnimi poslovnimi modeli, ki izumljajo nove vrednostne možnosti, spreminjajo pravila konkurence in aktivirajo ljudi ter sredstva na nepričakovano raven učinkovitosti. Hecker (1999) je opisal osem različnih poslovnih modelov, ki jih lahko uporabimo za odprtokodne programske rešitve. To je verjetno najbolj znan primer razvrstitve odprtokodnih poslovnih modelov, a je tudi ta še vedno precej omejen:

- temelji na strogih razredih, tako da ne upošteva poslovnih modelov zunaj te kategorizacije.
- osredotoča se na zunanji odprtokodni pristop in izključuje notranji odprtokodni pristop.
- ne upošteva drugih razlogov razen denarnih za sodelovanje v odprtokodnem gibanju in se v glavnem osredotoča na denarni tok med podjetjem in njenim okoljem.

Kritik Heckersovega poslovnega modela (1999) navaja, da so to le idealni poslovni modeli in podjetja jih lahko le združujejo ter ustvarjajo nove poslovne modele.

V nadaljevanju bomo predstavili Heckersov (1999) generični poslovni model:

- **Podpora prodajalcem:** To so podjetja, ki dajejo brezplačne programske rešitve in prodajajo podporo zanje. Dohodki prihajajo iz medijske distribucije, trženja blagovne znamke, izobraževanj, svetovanj, prilagojenega razvoja in poprodajne podpore (primer Red Hat, Open Source Initiative).
- **Izguba vodilnih:** V tem modelu se izdajo odprtokodne programske rešitve kot izguba vodilnih za pozicioniranje komercialnih programskih rešitev na trgu. Cilj je ustvariti interes skozi odprtokodne programske rešitve. Prihodki se ustvarjajo iz prodaje komercialnih programskih rešitev. To je počel Netscape (Open Source Initiative, 2010).
- **Zamrznitev velikega zaslužka:** To je uporaben model za podjetja, ki prodajajo predvsem strojno opremo. S priložitvijo odprtokodnih programskih rešitev strojni opremi lahko dajo produktu večjo vrednost za kupca. Dellovi prenosni računalniki z operacijskim sistemom Linux, HP strežniki z Apachejem in Silicon Graphics podpirajo in vgrajujejo Sambo (Open Source Initiative, 2010).
- **Dodatna oprema:** Prodaja dodatne opreme – knjig, združljive strojne opreme, celotnih sistemov z vnaprej nameščenimi odprtokodnimi programskimi rešitvami. Enostavni dodatki, kot so Linuxove lutke (pingvini), skodelice za kavo, razne potiskane majice in resnejši dodatki (knjige) predstavljajo velik uspeh podjetij, kot so O'Reilly Associates in SSC (Open Source Initiative, 2010).
- **Storitveni model:** Ustvarja prihodke od prodaje s tem povezanih storitev. Odprtokodne programske rešitve se razvijajo za podporo poslovnim storitvam.
- **Licenciranje blagovnih znamk:** Razvije odprtokodne programske rešitve in nato proda blagovno znamko drugemu podjetju, ki izdeluje produkte. Druga podjetja potem zaračunajo za uporabo blagovnih znamk.
- **Prodaj, sprosti kodo - model:** najprej proda programske rešitve kot običajno zaprte, komercialne programske rešitve in zatem na neki točki pretvori programske rešitve v odprtokodne.
- **Franšizing programskih rešitev:** Združuje več predhodnih modelov, zlasti licenciranje blagovnih znamk in podporo prodaji. Pri tem pristopu podjetja dovoljujejo drugim, da uporabljajo njihove blagovne znamke pri ustvarjanju pridruženih podjetij,

ki razvijajo programske rešitve po meri za točno določena geografska območja in trge. Podjetja lahko ponujajo za dodatno plačilo tudi franšize z usposabljanjem in storitvami.

Podobno Koenigsova (2004, str. 3) taksonomija vključuje naslednje strategije:

- **Optimizacija:** Finančni vzvod tehnologiji z dodajanjem vrednostnih plasti informacijski tehnologiji.
- **Dvojne licence:** V skladu s strategijo dvojnih licenc lahko programsko podjetje ponuja brezplačno uporabo svoje programske opreme z nekaterimi omejitvami ali pa ponuja proti plačilu komercialno distribucijo pravic in večji nabor funkcij.
- **Naročnina:** naročnina zagotavlja vzdrževanje za končnega uporabnika za eno leto in daje pravice do podpore konfiguracije, posodobitev tehnologije, kadar in če so na voljo v času trajanja naročnine. Sun ponuja StarOffice in velik del razvojnih in programskih dodatkov za naročnino ter priznava, da razvijalci imajo raje naročnino ali članstvo.
- **Svetovalne storitve:** eno od podjetij, ki nudi svetovanje v odprtokodnem prostoru, je podjetje 10X programska oprema, ki zagotavlja svetovanje podjetjem za integracijo odprtokodnih programskih rešitev vključno z MySQL, Apache, JBoss, Tomcat in Eclipse).
- **Pokroviteljstva:** Prispevajo k odprtokodnim programskim rešitvam, da bi jih povzdignili na višjo raven ali odpravili nelojalno konkurenco (Microsoft zasenči Netscape s 95 % deležem trga na področju brskalnikov, HP podpira več kot 60 odprtokodnih projektov, da bi zagotovil orodja, pripomočke in rešitve strankam za lažjo namestitve ali prilagoditev proizvodov).
- **Gostovanje:** Notranja uporaba odprtokodnih programskih rešitev za gostovalne platforme. Podjetja, kot so Salesforce.com, eBay in Google, so v programskem poslovnem svetu, vendar ne prodajajo njihovih programskih rešitev, temveč nudijo uporabo ali najem. Google ima več kot 100.000 Linuxovih strežnikov in ima načrte za povečanje strežniške infrastrukture.

Daffara (2007) opredeljuje naslednjih 6 skupin:

- **Dvojno licenciranje:** Iste programske rešitve se lahko distribuirajo pod GPL ali komercialnimi licencami.
- **Razcep odprtokodnih / komercialnih licenc:** Model temelji na osnovi odprtokodnih programskih rešitev s plačljivimi komercialnimi vtičniki.
- **Blagovna znamka:** Enako kot Heckersov model blagovnih znamk.
- **Produktni strokovnjaki:** Podjetja s strokovnim znanjem o produktih odprtokodnih programskih rešitev, s katerimi ustvarjajo dohodke od storitev, kjer so glavne storitve izobraževanje in svetovanje.
- **Ponudniki platform:** Ponudniki podpore, integracije in storitev tehnoloških platform, izbor in svetovanje.

3.4 Struktura poslovnega modela, uporabljenega v nalogi

Cilji magistrskega dela nam predstavljajo izhodišče za sestavo koncepta strukture poslovnega modela. S študijo primera podjetij želimo oblikovati pregled struktur različnih poslovnih modelov, ki temeljijo na odprtokodnih programskih rešitvah, in ne zgolj poizkusiti uvrstiti poslovne modele v vnaprej določene razrede kot v primeru Hecker 1999. Poleg tega je zelo pomembno, da so strukture tako splošne in prilagodljive, da lahko sprejemajo različne poslovne modele, ki temeljijo na odprtokodnih programskih rešitvah. Kot je predstavljeno v predhodnem poglavju, obstajata vsaj dva različna pristopa: notranji odprtokodni pristop in zunanji odprtokodni pristop. Struktura poslovnega modela se ne sme osredotočiti le na določene pristope ali na določene ravni poslovanja, kot na primer e-poslovanje. Tudi pri strukturah poslovnih modelov, ki temeljijo na odprtokodnih programskih rešitvah moramo biti še posebej pozorni na značilnosti skupnosti in odnose z njimi. Pri uvajanju poslovnih modelov, ki temeljijo na odprtokodnih programskih rešitvah, skupnosti formirajo popolnoma novo interesno skupino poslovne mreže podjetja. Zato je pomembno, da podjetje prilagodi svoje interese interesom skupnosti, da se ustvarijo dobri obojestranski odnosi. Dohodkovni model je še posebej zanimiv za poslovne modele, ki temeljijo na odprtokodnih programskih rešitvah, saj mora struktura poslovnega modela razkriti, kako podjetje pridobiva dohodke.

Na voljo ni veliko primernih struktur, ki bi upoštevale zgoraj omenjene zahteve. Nekatere strukture, predstavljene v poglavjih 4.2 in 4.3, so osredotočene na točno določena področja poslovanja, kot sta na primer e-poslovanje (na primer Mahadevan, 2000; Petrovich et al., 2001; Afuah & Tucci 2003) ali marketinško usmerjena stališča (na primer Hamel, 2000). Tako da so posamezni elementi primerni za uporabo v navedenih strukturah, nekateri elementi pa so neprimerni ali odveč pri študiji poslovnih modelov, ki temeljijo na odprtokodnih programskih rešitvah. Tako velja, da bi nekatere strukture potrebovale kar nekaj sprememb, preden bi jih lahko uporabljali za analizo poslovnih modelov, ki temeljijo na odprtokodnih programskih rešitvah. Teoretični del strukture poslovnega modela, uporabljenega v nalogi, bomo povzeli po različnih avtorjih (Osterwalder, 2004, Morris & Allen, 2005, Rappa, 2004, Kindström, 2005). In ker je treba doseči opise različnih poslovnih modelov, in ne le klasificirati poslovnih modelov v predefinirane opise, bomo iz različnih poslovnih modelov sestavili strukturo, ki je primerna za to vrsto študije in temelji na odprtokodnih programskih rešitvah. Tako bomo v nalogi reducirali poslovni model na štiri bistvene elemente: ponudba, viri, odnosi, dohodkovni model, s katerim bomo zajeli celoten pogled poslovnih modelov. V nadaljevanju je vsak element podrobneje predstavljen.

3.4.1 Ponudba

Opredelitev ponudbe je podobna opredelitvi ponudbe uveljavljene literature poslovanja in upravljanja. Ponudba opredeljuje celotno ponudbo produktov in storitev podjetij svojim strankam in glavne vrednote za stranke. Bistvene koristi ali vrednosti, ki jih stranka kupi,

so lahko na primer značilnosti proizvoda, kakovost ali blagovna znamka (Kindström, 2005, Kotler 1996, str. 68). Danes vse več podjetij sodeluje z različnimi strateškimi partnerji. Zato je pomembno, da se ponudba obravnava kot storitev teh povezanih akterjev. V okviru poslovanja podjetij ponudba pomeni, katere programske rešitve uporabljajo oziroma ponujajo podjetja, katere dodatne storitve so prisotne v ponudbi in kako ima podjetje urejeno distribucijo produktov do njihovih strank. Programske licence so tudi del ponudbe, saj imajo velik vpliv na podjetja, ki ponujajo ali uporabljajo odprtokodne programske rešitve.

3.4.2 Viri

Razvoj ponudbe temelji na uspešnih in učinkovitih virih poslovnega procesa, ključnih skupnostih in partnerjih podjetja. Cilj virov je, ugotoviti, kateri so ključni viri in zmogljivosti, ki jih mora imeti podjetje pri različnih vrstah poslovnih modelov, ki temeljijo na odprtokodnih programskih rešitvah. Vire lahko razdelimo na opredmetene in neopredmetene, med katere sodijo človeške zmogljivosti, ki se nanašajo na znanje in sposobnosti upravljanja, ravnanja in izvajanja v podjetju (Hitt & Ireland & Hoskinsson, 2001, str. 16). Pri odprtokodnih poslovnih modelih se zmogljivosti podjetij odražajo predvsem v njihovem uspehu združevanja notranjih in zunanjih virov za opravljanje dejavnosti. Viri in zmogljivosti se delijo na notranje in zunanje vire in zmogljivosti. Notranji viri vplivajo na podjetje znotraj le tega medtem, ko zunanji viri so pridobljeni iz različnih odprtokodnih skupnosti.

3.4.3 Odnosi

Prav tako kot ponudba in viri tudi odnosi v magistrski nalogi veljajo za nepogrešljiv del poslovnega modela. Timmers (2003) trdi, da se je fokus ustvarjanja vrednosti iz notranjih dejavnikov preusmeril k ustvarjanju vrednosti z zunanjimi odnosi. Odnosi predstavljajo pomemben element v vrednosti mreži, zlasti pri razvoju in distribuciji ponudb. Normann in Ramirez (1995, str. 17) definirata ustvarjanje vrednosti kot »sistem ustvarjanja vrednosti, znotraj katerega različni igralci – dobavitelji, poslovni partnerji, kupci – delujejo skupaj z namenom soustvarjanja vrednosti. Podjetja se morajo odločiti, katere dejavnosti bodo opravljala sama in katere dele bodo izvajala v sodelovanju s poslovnimi partnerji. Za sodelovanje podjetij z odprtokodnimi skupnostmi imajo zelo pomembno vlogo odnosi, na podlagi katerih temeljijo interakcije med podjetji in skupnostmi. Če podjetja želijo spodbuditi skupnosti, morajo ustvariti pogoje za komunikacijo in sodelovanje z njimi. Interakcija ni pomembna v primeru, ko se podjetje odloči zgolj za uporabo rezultatov skupnosti. S sodelovanjem s skupnostmi se morajo podjetja zavedati vpliva skupnosti na

način dela podjetja. Tako podjetje ne bo sprejemalo strogih rokov in načrtov in morda podpora ne bo na voljo vedno, kadar je to potrebno.

3.4.4 Dohodkovni model

Četrty element poslovnega modela je dohodkovni model, ki vključuje tako prihodke od prodaje kot druge vire financiranja. Vsi trije obravnavani endogeni elementi (ponudba, viri in odnosi) poslovnega modela omogočajo, usmerjajo in omejujejo nastajanje dohodkovnega modela. Kindström (2005) trdi, da obstajajo najmanj štiri različni zunanji dejavniki, ki vplivajo na dohodkovni model. Ti dejavniki so kupci, konkurenčno okolje, tehnološka infrastruktura in finančno okolje. Nekoliko so podobni Porterjevemu (2001, str. 66) modelu petih sil ki, določa pet sil, ki vplivajo na celotno podjetje: kupci, konkurenti, dobavitelji, novi udeleženci in nadomestki. Dohodkovni model opisuje, kakšna je prodajna vrednost produkta ali storitve, zajete s strani podjetja. Prodajna vrednost se nanaša na Raymondovo (2001, str. 117) klasifikacijo gospodarskih vrednosti programskih rešitev. Programske rešitve imajo dva različna ekonomska vira vrednosti: **uporabno vrednost** in **prodajno vrednost**. **Uporabna vrednost** programske rešitve predstavlja ekonomsko vrednost kot orodje, med tem ko **prodajna vrednost** predstavlja ekonomsko vrednost kot prodajni produkt. Dohodkovni modeli so lahko neposredni ali posredni (Kindström, 2005). Neposredni dohodkovni modeli so posledica uporabe programskih rešitev, medtem ko so posredni dohodkovni modeli nastali s tretjimi osebami ali pa na kakšen drugačen način. V magistrski nalogi bomo obravnavali stroške in prihranke kot del dohodkovnega modela, saj je mogoče, da podjetja ne poskusijo ustvariti neposrednih ali posrednih dohodkov, ampak pridobiti dohodke z zmanjševanjem stroškov kot izbira poslovnega modela.

3.5 Ocena različnih modelov in primerjava med njimi

V literaturi je kar nekaj različnih poslovnih modelov, katerih struktura vsebuje različne elemente, ki se razlikujejo od avtorja do avtorja, po drugi strani pa so si po določenih elementih zelo podobni.

V veliko poslovnih strukturah so elementi preveč na široko obrazloženi z istim pomenom, samo poimenovani so drugače. Poleg tega je lahko en element v nekaterih strukturah razdeljen na dva, zelo strogo določena elementa. Nekatero strukturo vključujejo tudi zunanje dejavnike (na primer Hedman & Kalling, 2003: konkurenti), vendar je večina struktur osredotočenih na notranje dejavnike. Nekatero strukturo vsebujejo elemente za strategijo in poslanstvo (Alt & Zimmerman, 2001: poslanstvo; Hamel 2000; ključna strategija). Tudi procesi se upoštevajo pri nekaterih strukturah (Alt in Zimmerman, 2001: procesi; Hedman & Kalling, 2003: vzdolžni procesi). Vsi poskušajo opredeliti, kaj podjetje ponuja svojim strankam, katere so dejavnosti in viri, potrebni za proizvodnjo ponudbe, in

kako se ustvarjajo dohodki. Element, ki se največkrat pojavi v obravnavanih poslovnih modelih, je dohodkovni model.

Tabela 8: Primerjava med različnimi poslovnimi modeli

Avtor	Elementi
Hamel (2000)	Bistvo strategije, podpora strategiji, odnosi s kupci, vrednostne mreže.
Mahadevan (2000)	Vrednostni tokovi za poslovne partnerje in kupce, tok dohodkov in logistični tok.
Afuah & Tucci (2001)	Vrednost za kupca, obseg aktivnosti podjetja, cenitev vrednosti, vir dohodka, povezanost aktivnosti, izvedba, sposobnost, konkurenčnost.
Alt & Zimmerman (2001)	Poslanstvo, struktura, procesi, dohodek, pravne zadeve, tehnologija.
Petrovic et al. (2001)	Vrednostni model, model virov, model proizvodnje, model odnosov s strankami, dohodkovni model, model kapitala in tržni model.
Hedman & Kalling (2003)	Kupci, konkurenti, ponudba, dejavnost in organizacija, viri, dobavitelji, proizvodni input in vzdolžni proces.
Osterwalder (2004)	Produkt (predlagana vrednost), uporabniški vmesnik (ciljne stranke, distributivni kanal, odnosi), upravljanje infrastrukture (vrednost konfiguracije, zmogljivost, partnerstvo), finančni vidik (struktura stroškov, dohodkovni model).
Kindström (2005)	Strateški položaj, operativna platforma in ponudba.

Ker v nalogi ne želimo klasificirati poslovnih modelov v predefinirane opise, smo iz različnih zgoraj navedenih elementov poslovnih modelov sestavili strukturo, ki je primerna za to vrsto študije.

Strukturo poslovnega modela smo skrčili na štiri bistvene elemente: ponudba, viri, odnosi, dohodkovni model, s katerim bomo zajeli celoten pogled poslovnih modelov. S poslovnim modelom smo tako določili, kaj podjetja ponujajo svojim strankam, kateri notranji, zunanji viri in odnosi so potrebni za proizvodnjo te ponudbe in kako podjetje ustvarja dohodke s to ponudbo. Ker poslovni modeli temeljijo na odprtokodnih programskih rešitvah nismo neposredno upoštevali zunanjih dejavnikov, kot so konkurenti, substituti, dobavitelji in tržni pogoji, pač pa smo jih posredno, pri ponudbi in dohodkovnem modelu podjetij. Prav tako nismo neposredno upoštevali strategije in poslanstva v poslovnem modelu, kar je običajno izhodišče, ko podjetje načrtuje svojo ponudbo, zato da bi bila ponudba v skladu s poslanstvom in strategijo. S poslovnim modelom nismo neposredno pozorni na ustreznost ponudbe na določenem trgu in s tem ne upoštevamo, ali je ponudba v skladu s poslanstvom in strategijo. To je mogoče razumeti kot omejitev poslovnega modela, kar moramo upoštevati pri razlagi rezultatov v nalogi. Po drugi strani pa poslovni model kreira obsežno strukturo za analizo notranjih dejavnikov, ki vplivajo na programske rešitve poslovnega modela. V okviru odprtokodnih programskih rešitev so še posebej zanimivi viri in odnosi.

Podjetja, ki uporabljajo odprtokodne programske rešitve, se srečujejo s povsem novimi načini dela in z novimi izzivi. Podjetja lahko pridobijo brezplačne vire za razvoj programskih rešitev od zunanjih udeležencev, a morajo za to vložiti veliko truda za vzpostavitev dobrih odnosov s skupnostmi, da jih lahko motivirajo in vplivajo nanje. S pomočjo odprtokodnih programskih rešitev lahko podjetja kreirajo povsem nove dohodkovne modele, kar je v našem poslovnem modelu predstavljeno s četrtem elementom. Poslovni model ima dobro strukturo za oblikovanje celostnega pogleda na modele poslovanja podjetij in upošteva elemente, ki so še posebej pomembni za poslovne modele, ki temeljijo na odprtokodnih programskih rešitvah.

4 ANALIZA UPORABNOSTI ODPRTOKODNIH PROGRAMSKIH REŠITEV V PODJETJIH

Uporaba odprtokodnih programskih rešitev v podjetjih je odvisna od različnih dejavnikov, zato pri njihovi uporabi upajajo notranje ali zunanje pristope. Zelo raznolika je tudi praksa pri sestavi razvojne skupnosti, ki se posledično odraža v vlogi notranjih in zunanjih virov. Kako vplivata na uporabo teh rešitev strategija in ponudba, kakšen je odnos med razvijalci in skupnostjo, med podjetjem in skupnostjo, kaj podjetja motivira, kdo izdaja izvirno kodo, kako poteka komunikacija med vsemi člani, s kakšnimi tveganji se srečujejo in kakšnim vplivnim dejavnikom so podvrženi, so vprašanja, na katera lahko odgovorimo s pomočjo analize uporabnosti odprtokodnih programskih rešitev v podjetjih.

4.1 Utemeljitev strukture analize

Cilj magistrskega dela je ugotoviti, kako podjetja uporabljajo odprtokodne programske produkte, razvojno prakso in kakšni poslovni modeli obstajajo v teh podjetjih. S pomočjo kvalitativne raziskave želimo v tej študiji ustvariti razumljiv pogled na primere odprtokodnih poslovnih modelov podjetij. Kardoff (2004, str. 3) je pojem kvalitativno raziskovanje opredelil kot skupni pojem za zelo različne teoretične, metodološke in metodične pristope k družbeni različnosti. Kvalitativni pristop raziskave je po Denzinu (2000, str. 5) primeren, ko želimo pridobiti vpogled v določene fenomene ali pojave, za katere je bilo opravljenih le malo raziskav in niso natančno znane pomembne spremenljivke. Podpira dejstvo za izbiro kvantitativnega pristopa, ker je le malo formalne literature o poslovnih modelih podjetij, ki uporabljajo odprtokodne prakse. Po Denzinu so kvalitativni podatki zelo bogati in podrobni, primerni za oblikovanje analize, ki se osredotoča na uporabo teh podatkov za kreiranje celovitega vsebinskega načina. To je v skladu s cilji naše študije.

Kvalitativni pristop je v tej raziskavi razlagalen. Razlagalno-pojasnjevalna raziskava predpostavlja, da je dostop do realnosti mogoč le s pomočjo tako imenovanih socialnih gradenj, kot so jezik, zavest in skupni interesi. Pri razlagalnem pristopu so primarni viri

podatkov ljudje in njihove razlage, dojemanja, pomeni in razumevanja. Po Masonovem mnenju (2002, str. 56) za razlagalno raziskavo ni treba sklicevati skupnih srečanj, saj lahko na primer podpremo študijo z uporabo metode anketiranja. Tudi drugi viri podatkov pridejo prav, možna so na primer različna besedila. Razlagalni pristop je za to študijo primeren, saj je cilj oblikovati celosten pogled na raziskovalni problem.

Študija primera je bila izvedena s pomočjo celostnih pluralnih (angl. multiple case study) primerov Yin (2002, str. 150). Yin (2002, str. 1) definira študijo primera kot empirično študijo, za katero je značilno, da raziskovalec raziskuje sodoben pojav v realnem življenju, pri čemer meje med pojavom in kontekstom pojava niso vedno jasno razvidne in pri katerem se uporabljajo različni viri podatkov. Študija primera je idealna, ko bi radi vedeli, »kako« ali »zakaj« je bilo raziskovalno vprašanje postavljeno o proučevanem fenomenu, nad katerim raziskovalci nimajo nadzora. Študija primera v svoji osnovi opisuje in pojasnjuje, a je njen namen lahko tudi, da poskuša odkriti vzročne odnose med različnimi zadevami (Yin, 2002, str. 1). Raziskovalno vprašanje v tej študiji se glasi »Kakšni poslovni modeli obstajajo v podjetjih, ki temeljijo na odprtokodnih programskih rešitvah«, in je opisne narave. Zaradi teh dejavnikov se študija primera šteje za najbolj sprejemljivo metodo raziskovanja.

Celostni pristop je koristen, če raziskava ne vsebuje nobene logične podenote, ki jo je treba analizirati (Yin, 2002, str. 45). Teoretična struktura poslovnega modela v tej študiji primera opredeljuje poslovne modele podjetij s štirimi elementi: ponudba, viri, odnosi in dohodkovni model. Namen strukture niso natančne definicije poslovnih modelov, ampak – razlaga poslovnih modelov z uporabljenimi elementi. S to študijo želimo ustvariti pogled na različne strukture poslovnih modelov, ki obstajajo v podjetjih. Vsi primeri so analizirani kot ločene enote in ne vsebujejo nobenih pod enot. Zato je izbira celovite pluralne študije primerna.

Po Yinu (2002, str. 150) mora biti v primeru celovite pluralne izbire vsak primer skrbno izbran tako, da napoveduje podobne rezultate (dobesedna replikacija) ali pa da daje nasprotujoče si rezultate iz predvidljivih razlogov (teoretična replikacija). V tej študiji lahko različne primere razdelimo v dve različni kategoriji: notranji odprtokodni pristop (kako podjetja uporabljajo odprtokodne programske rešitve) in zunanji odprtokodni pristop (sprostitev izvorne kode programskih rešitev javnosti). Zato so rezultati verjetno drugačni. To je v skladu s konceptom teoretične replikacije.

Ticehurst (2000, str. 95) povzema po Easterby-Smithu, Thorpeju in Loweju (1991) naslednje prednosti kvalitativnih metod:

- Raziskovalcu omogočajo, da razume in pojasni osebne izkušnje posameznika, ki so lahko interne ali eksterne.

- Osredotočene so na človeško razumevanje in interpretacijo, bolj kot na ugotovitve zunanjih vzrokov ali zakonov obnašanja.
- Uporabljajo se lahko za ozko usmerjene ali holistične, celostne študije, ko se popolnoma razišče situacija.
- Raziskovalcu dovolijo, da raziskuje raziskovalni problem z vidika udeleženca.
- Navadno so predstavljene v pripovedni obliki, bolj kot v statistični formi, kar jih naredi bolj zanimive za bralce, ki niso večši statistike.
- Statistične metode so koristne za raziskovanje osebnih sprememb čez čas in v času.
- Prikrito znanje ima pomembno vlogo pri analiziranju raziskovalnega problema in se lahko naprej razvija preko induktivnega procesa.
- Čustva raziskovalca, tako kot njegovo utemeljevanje, lahko pomagajo oblikovati raziskavo.
- Kvalitativni raziskovalci lahko uporabljajo kombinacijo tehnik in celo izumljajo metode, da zagotovijo široko razumevanje socialne situacije.
- Kvalitativne metode se osredotočajo na zadeve, ki zanimajo človeka, kar je pomembno za menedžerje.

Zbiranje podatkov je potekalo s pomočjo anketnega vprašalnika, ki je bil poslan po elektronski pošti. Kvalitativni način je bil uporabljen, ker smo želeli ustvariti širok, raznolik, kompleksen in celovit način zbiranja podatkov, ki je v skladu z Masonom (2002, str. 65).

Poslovni model ima širok koncept in globoko konceptualizacijo, zato bi bilo težko, da bi bil zasnovan le s pomočjo zunanjih informacij. Za oblikovanje celovitega pogleda sta potrebna poglobljeno zbiranje podatkov in analiza. Kot posledica tega smo se odločili, da bomo podatke pridobili s pomočjo kvalitativnega vprašalnika.

Anketiranci so bili izbrani tako, da smo lahko oblikovali celosten pogled na problem raziskovanja, zato je bilo zaželeno, da ima vsak anketiranec celovit pregled nad poslovanjem podjetja, odnose z odprtokodno skupnostjo in širše razumevanje odprtokodnih programskih rešitev.

Vsi vprašalniki so sestavljeni iz vprašanj, ki izhajajo iz predlagane strukture poslovnega modela. Vsa vprašanja so odprtega tipa. Odprto vprašanje ne podaja vnaprej oblikovanih odgovorov, zato lahko anketiranec odgovarja popolnoma svobodno, s svojimi besedami. Pri zaprtem vprašanju anketiranec izbere odgovor(e) med vnaprej ponujenimi možnostmi, ki jih je oblikoval raziskovalec glede na raziskovalni cilj. Anketiranci so imeli možnost zahtevati pojasnila, če se je pojavilo nerazumevanje besed ali vprašanja.

Vprašanja so bila zastavljena na podlagi vprašalnika (Priloga 1), odgovori so bili prosti in tako dolgi, kot je kdo želel. Anketiranci niso bili pozvani na noben drug način. Anketiranci so se vzdržali ideji, da so odgovori pravilni ali napačni.

Vseh osem vprašalnikov je bilo uspešno izpolnjenih in so doprinesli želene podatke za študijo.

Analiza podatkov je potekala v treh fazah. V okviru prve faze smo na podlagi zbranih odgovorov sestavili spise, zbrane s pomočjo vprašalnika. Potem ko smo spise sestavili, smo jih poslali, tako da so jih lahko prebrali in po potrebi dopolnili. Le trije anketiranci so svoje spise dopolnili, drugi so se z njimi strinjali. Trije anketiranci so bili po elektronski pošti naprošeni, da dodatno pojasnijo nekatere njihove odgovore.

Po oblikovanju spisov je bilo odločeno, da dodatne ankete niso potrebne. Na osnovi prejetih anketnih vprašalnikov je bilo mogoče oblikovati celosten pogled za vsak primer posebej.

Po opravljeni prvi fazi smo v drugi fazi obdelali podatke in ugotovitve razvrstili glede na strukturo poslovnega modela. Cilj je bil pridobiti ustrezne informacije za ugotovitev, kakšni poslovni modeli obstajajo v teh podjetjih in oblikovati bogato razumevanje in vpogled v uporabljene poslovne modele.

V zadnji, tretji fazi, smo rezultate analizirali na podlagi izbrane strukture poslovnega modela in poiskali skupne točke oziroma podobnosti med posameznimi primeri ter jih poskusili razvrsti v kategorije.

Zanesljivost in veljavnost raziskave so kriteriji, ki jih upoštevamo že takrat, ko načrtujemo raziskavo. Saunders, Lewis in Thornhill (1997, str. 164) svetujejo, da zanesljivost v fenomenološki raziskavi preverimo z vprašanjema, ali bi ponovno (enaka) raziskava prinesla enake rezultate in ali bi drugi raziskovalci v drugih okoljih uporabili podobne metode raziskovanja. Cilj zanesljivosti je zmanjšati napake in pristranskosti v študiji primera. Za zagotovitev tega, Yin (2002, str. 34) priporoča uporabo načrta študije primera in razvoj baze podatkov študije primera.

Zaželeno je, da ima študija primera načrt, ki ga zasleduje v vseh okoliščinah, a še posebej je to primerno za pluralne primere. Načrt je glavni kriterij za povečevanje zanesljivosti raziskave študije primera in vodilo pri njenem izvajanju. Sestavni deli načrta študije primera so razdeljeni na naslednja poglavja: pregled projekta, opredelitev primera (fenomena), raziskovalno vprašanje, vzorec, metoda zbiranja podatkov, obdelava, analiza in zaključek. Ta struktura je v skladu z Yinovim modelom. Vzpostavljena je bila tudi baza podatkov študije primera za pomoč pri obravnavanju zanesljivosti raziskave. Ta vsebuje anketni vprašalnik, ki je v skladu s konceptom strukture poslovnega modela ter z drugimi materiali, ki so povezani s projektom. Poleg tega baza podatkov vsebuje tudi povzetke odgovorov in kratke opise vsakega podjetja.

Pri izvedbi raziskave so bili upoštevani naslednji kriteriji za preverjanje veljavnosti študije primera (Yin, 2002, str. 35): **veljavnost konstrukta, notranja veljavnost in zunanja veljavnost**. **Notranja veljavnost** je primerna samo za obrazložitev vzročne študije in zato ni pomembna za to študijo. **Veljavnost konstrukta** obravnava obseg operativnih ukrepov, ki se nanašajo na raziskovanje pojava. Za pravilno izgradnjo veljavnosti moramo zagotoviti naslednje tri kriterije: zagotovitev nepristranskosti raziskovalca z uporabo več virov dokazov v podjetju, vzpostavitev verige dokazov ter pregled osnutka ugotovitev s ključnimi informatorji. Študija je bila narejena sistematično za zagotavljanje celovitosti in veljavnosti podatkov. Poleg anket smo pregledali tudi nekaj poročil študij primerov in drugih materialov, povezanih s projektom. Struktura dokazov je bila vzpostavljena z zasnovo študije primera in načrtom študije primera. Na koncu so bili primeri preverjeni še s strani ključnih informatorjev iz podjetij.

Zunanja veljavnost se navezuje na možnost posploševanja, tako analitičnega kot teoretičnega. Namen te študije ni statistično posploševanje, ampak analitično posploševanje. Ideja analitske posplošitve je, da posplošimo posebne vrste rezultatov za nekatere večje teorije (Yin, 2002, str. 37). Ena od omejitev te študije je, da je vzorec majhen. Primere lahko posplošeno razdelimo v dve kategoriji: notranji pristop in zunanji pristop. Štirje primeri predstavljajo notranji pristop in štirje primeri predstavljajo zunanji pristop. Primerjave znotraj vsake kategorije so možne, a so nekateri primeri precej različni, zlasti v kategoriji zunanjega pristopa. Zato je obseg posploševanja omejen. Če bi študija vključevala več primerov podjetij, obstaja možnost, da bi se pojavili novi načini uporabe odprtokodnih programskih rešitev. Cilj te študije ni oblikovati celotne zbirke poslovnih modelov, ki temeljijo na odprtokodnih programskih rešitvah, ampak prikazati osem različnih načinov za uporabo odprtokodnega fenomena. Vsak od primerov predstavlja določen poslovni model in skupaj ustvarijo širok pogled na to, kako lahko podjetja uporabljajo odprtokodne programske rešitve. Poleg tega smatramo podatke, ki se nanašajo na primere, da so zadovoljivo poglobljeni. To pomeni, da novi primeri ne bi ustvarili novih spoznanj in tako smo na podlagi zbranih podatkov uspeli oblikovati celovite poglede na posamezne primere.

4.2 Odprtokodni poslovni modeli podjetij

4.2.1 Hit d.d.

Skupino Hit sestavlja 11 družb, ki so po inovativnosti na področju zabave postale vodilne v Evropi. Med te sodi: 7 igralniško-zabaviških centrov, 1 turistični resort, 1 igralnica, 2 igralna salona, 11 hotelov, 1 potovalna agencija, več kot 2.300 hotelskih ležišč, 2000 sedežev v konferenčnih centrih, več kot 4000 igralnih avtomatov, 215 igralnih miz, 2.883 zaposlenih (konec decembra 2009). Geslo Hitovih igralniško-zabaviških centrov in

hotelov je prav zabava, ki jo spremljajo velika izbira iger na igralnih mizah in avtomatih, vrhunski koncerti, prvovrstna kulinarična ponudba, razkošne nastanitve, wellness ponudba ter vse, kar lahko gostom omogoča najboljše počutje. Krovna družba skupine je družba Hit, d.d. (v nadaljevanju Hit), s sedežem v Novi Gorici, ki nadzoruje 10 odvisnih družb, od katerih jih šest deluje v Sloveniji, ostale štiri pa se nahajajo na Hrvaškem, v Bosni in Hercegovini, v Črni gori in v Srbiji.

Ponudba. V podjetju Hit uporabljajo odprtokodne programske rešitve v naslednji sestavi: operacijski sistem Linux, spletni strežnik Apache, programski jezik PHP, podatkovna baza MySql ter orodje za obvladovanje spletnih vsebin TYPO3. Njihove dodatne storitve so prisotne v obliki podpore, vzdrževanja, razvoja in svetovanja za vse navedene programske rešitve v prvem vprašanju. Njihove odprtokodne programske rešitve so brezplačne, za dodatne storitve, kot so podpora, vzdrževanje, razvoj in svetovanje, pa Hit plačuje mesečni pavšal. Z večjo angažiranostjo so pri razvoju deležni drugačne ponudbe pri nadgradnjah, svetovanju in podpori. Zelo dobro sodelujejo pri razvoju vtičnikov spletnega sistema za urejanje spletnih vsebin TYPO3. Nove različice prilagajajo po načelu »izdajaj zgodaj, izdajaj pogosto«. Pogoste izdaje verzij programa omogočajo pridobivanje kvalitetnih povratnih informacij zelo različnih uporabnikov, prav tako se hitreje odkrijejo napake. Za odprtokodne programske rešitve so uporabljene naslednje licence: Operacijski sistem: Linux GNU General Public License, Spletni strežnik: Apache Apache License 2.0, Podatkovna baza: MySql GPL licence, Programski jezik: PHP Licence v3.01, Sistem za urejanje vsebin CMS: TYPO3 GPL (General Public License). Prosto programje ima običajno manj prijazne uporabniške vmesnike. To se izkaže kot slabost pri uporabnikih, ki niso večji dela v računalniškem okolju. Podpora je namenjena razvijalcem, in ne končnim uporabnikom z zelo tehnično dokumentacijo. Poleg tega programi pogosto niso tipa »vstavi in poženi«. Pogoji za to, da odprtokodni projekt zaživi, je ta, da mora privabiti dovolj veliko skupnost zelo delavnih razvijalcev.

Viri. V Hitu so ključni notranji viri, bogate izkušnje na igralniškem trgu in dobra blagovna znamka. Potrebne vire za raziskave in razvoj vtičnikov, dodatkov in dokumentacij odprtokodnih programskih rešitev ima Hit v podjetju. Iztočnice črpajo tudi preko sodelovanja s skupnostmi. V trženjskem oddelku so viri potrebni za prodajne in promocijske aktivnosti. Za spletno pojavnost in prodajo skrbi ekipa treh zaposlenih, ki ima široko znanje o spletni pojavnosti in prodaji. Imajo tudi znanje o odprtokodnih programskih rešitvah, saj jih uporabljajo dnevno in tako pripomorejo k razvoju novih vtičnikov, ki jim olajšajo delo. Iz odprtokodnih skupnosti pridobivajo predvsem programerske vire, ki lahko specifično programsko rešitev razvijejo v zelo kratkem času. Pri manjših skupnostih so neskladja pogostejša, z večanjem skupnosti pa se procent zmanjšuje. Največ anomalij prihaja pri sestavi dokumentacije, uporabi licenc, nespoštovanju dogovora o datumu realizacije programskih rešitev itd.

Odnosi. Njihovi ključni partnerji pri razvoju odprtokodnih programskih rešitev so Typo3, php, MySQL, Apache in Ubuntu (Linux). Ključni komercialni akterji so Microsoft, Oracle ter vsi ostali ponudniki komercialnih rešitev za internetno pojavnost in prodajo. Ključni partnerji (Google, Iopublicita, GDS, booking.com) opravljajo oglaševalsko dejavnost v poslovnem okolju, v odprtokodnih skupnostih pa razvojno funkcijo. Hit ima veliko interakcij z odprtokodnimi skupnostmi. Po njihovem mnenju imajo skupnosti rade podjetja, ki se ne eksplicitno ukvarjajo z razvojem programskih rešitev, ampak jih le dopolnjujejo in uporabljajo. Odprtokodne skupnosti zelo vplivajo na tehnološko in posledično tudi na poslovno odločanje, zato sodelujejo na različnih odprtokodnih projektih (TYPO3, Ubuntu, php, MySQL ...), ki razvijalcu omogočajo uporabo najnovejših tehnologij, ki se odražajo na boljšem poslovanju. Z odprtokodnimi skupnostmi komunicirajo z uporabo temu prilagojenih spletnih strani ali internetnih aplikacij (Webex ...). Hit ima s skupnostmi razvite neke vrste načrte za razvoj vtičnikov, popravkov in dokumentacij. Njihov namen je ustvariti vizijo in smernice za povečanje uporabe odprtokodnih programskih rešitev v poslovne namene in znotraj podjetja Hit. S pomočjo odprtokodnega sistema za obvladovanje spletnih vsebin Typo3 potekajo vsa on-line povpraševanja na spletnih straneh podjetja Hit, ki so ključnega pomena pri ugotavljanju ROI-ja in realizaciji dohodkov.

Dohodkovni model. V podjetju Hit so glavni vir dohodkov on-line nakupi (nakupi igralniških, spa in hotelskih paketov, rezervacije poker udeležbe, vstopnic za koncerte ...) italijanskih, avstrijskih in slovenskih gostov, zato je spletna prodaja prilagojena vsakemu posebej. Osnova za izračun cene programskih rešitev se izračuna na osnovi realiziranih on-line nakupov. Njihovo plačilo je provizija od prodaje on-line nakupov. Med stroške sodita razvoj in prilagoditev na uporabo odprtokodnih programskih rešitev. V Hitu na to ne gledajo kot na strošek ampak kot razvoj lastnih človeških virov ter obenem tudi vložek v prihodnost.

4.2.2 Typo3

Typo3, katerega korenine segajo v leto 1998, velja za eno najzmogljivejših odprtokodnih rešitev na področju urejanja spletnih vsebin. Namenjen je tako malim, kot tudi velikim podjetjem, ki potrebujejo prilagodljiv in razširljiv sistem ter veliko število že pripravljenih modulov, s katerimi lahko dodatno povečajo uporabnost sistema in funkcionalnost lastne spletne strani. Typo3 je narejen z zelo popularnim odprtokodnim skriptnim jezikom PHP, kar omogoča neslutene možnosti prilagodljivosti in nadgradnje. Typo3 med drugim, zaradi njegove funkcionalnosti in bistveno znižanih stroškov vzdrževanja, uporabljajo tudi spletne strani državne uprave Republike Slovenije.

Ponudba. Njihova glavna ponudba je programska rešitev za upravljanje z internetnimi vsebinami Typo3 (CMS orodje) s ciljnimi trgi v Nemčiji in z ostalimi trgi po svetu. V

njihovi ponudbi so prisotne dodatne storitve, kot sta svetovanje in podpora, in jih ponujajo v obliki članstva, ki se deli v 4 skupine: bronasto, srebrno, zlato in platinasto. CMS orodje je brezplačno, uporabnik, če želi, plača le članstvo. **Bronasto članstvo:** enkratna pristopnina 100 €, letni pavšal 100 €. Primerno je za posameznike, ki uporabljajo Typo3 redno, saj s tem pridobijo možnost, da prispevajo rešitve nazaj v skupnost in tako pripomorejo k temu, da so projekti uspešnejši. S tem pridobijo tudi podporo pri Typo3 projektih, popuste pri certificiranjih, konferencah in dogodkih, udeležbo na letnem srečanju T3A, obveščeni so o vseh tekočih aktivnostih skupnosti. **Srebrno članstvo:** enkratna pristopnina znaša 500 €, letni pavšal pa 700 €. Članstvo je primerno za majhna podjetja, ki ponujajo Typo3 storitve ali redno uporabljajo Typo3 kot končni uporabnik. To članstvo je temeljna zahteva za prisotnost na splošnem seznamu typo3.org svetovanj. Poleg tega člani iz Nemčije prejmejo tudi brezplačno enoletno naročnino na revijo t3n. **Zlato članstvo:** enkratna pristopnina znaša 500 €, letni pavšal pa 2000 €. Primerno je za srednje velika podjetja, ki ocenijo, da jim srebrno članstvo ne ponuja dovolj. Zlato članstvo vsebuje vse, kar ponuja srebrno članstvo in še kupon za brezplačno udeležbo na kakršnem koli Typo3 dogodku. **Platinasto članstvo:** enkratna pristopnina je 500 dolarjev, letni pavšal 10.000 dolarjev. Primerno za podjetja, ki ponujajo več kot Typo3 storitve in so prihranila na tisoče evrov na licencah, ker uporabljajo in ponujajo odprtokodni sistem Typo3. Člani pridobijo vse koristi predhodnih članstev ter še dodatno oglasno pasico na typo3.org, s povezavo na njihovo spletno stran ter tri kupone za brezplačno udeležbo na kakršnem koli Typo3 dogodku. Typo3 CMS uporablja licenco GNU General Public License. Vse oblike storitev distribuirajo preko spletne strani. Verzije se spreminjajo na šest mesecev. Uporabniki, ki se želijo prepričati, da je verzija stabilna, počakajo do trenutka, ko verzija dobi oznako stabilna. Lahko pa sodelujejo in skupnosti poročajo o napakah v času alfa verzij. Nadgradnjo CMS orodja opravimo kar prek spleta tako, da pod rubriko nadgradnja enostavno naložimo novo verzijo. Nadgradnje so in vedno bodo brezplačne.

Viri. Ključni viri so jedro ekipe Typo3 in skupnost Typo3. Jedro Typo3 je namenjeno razvoju in vzdrževanju osrednjega dela CMS orodja. Na voljo so tudi podskupine (v4 in v5) za večje branže in projektni timi, ki delajo na glavnih ciljih za specifične verzije. Viri na področju raziskav in razvoja skrbijo za organiziranost Typo3 raziskav in razvojnih projektov. Njihov glavni cilj je poskrbeti za skladen in dolgoročno usmerjen razvoj. Vršijo nadziranje in koordiniranje razvoja Typo3 in flow3, sprejemajo končne odločitve o strateških ali spornih vprašanjih za jedro ekipe, podpirajo razvojne projekte z zagotavljanjem znanja in kontaktov, podpirajo in nadzorujejo skupnosti za raziskave in razvoj, dajejo smernice za programiranje in pripravo časovnega načrta. Iz odprtokodnih skupnosti pridobivajo vire vseh vrst za njihove potrebe, tako programske kot organizacijske, strateške in marketinške. Typo3 je velika in rastoča skupnost z vedno manj neskladji. Dokumentacija pri vtičnikih jim predstavlja največji odstotek neskladij, saj jih v nekaterih primerih razvijalci ne priložijo, a se odstotek manjša.

Odnosi. Če ne bi bilo skupnosti, ne bi bilo CMS orodja Typo3. Za poslovno in tehnološko odločanje skrbi skupnost, ki je sestavljena iz uprave, upravnega odbora, skupščine. Skrbijo za pospeševanje razvoja Typo3, organizacijo prireditev za obveščanje in izobraževanje skupnosti, certificiranje za zagotavljanje kakovosti storitev ter komuniciranje s člani in širšo javnostjo za širjenje znanja in usposobljenosti za uporabo Typo3 programske opreme. Njihovi ključni partnerji pri razvoju in raziskavah so skupine uporabnikov po celem svetu: München, Stuttgart, Frankfurt, Berlin, Danska, Švica, Avstrija, Francija, Nova Zelandija, ZDA, Norveška, Indija. Ključni partnerji pri začetnem financiranju pa so: AOL media , CS2 – Creative Solutions GmbH , AlterNET Internet BV ... Med ključne partnerje za obstoj Typo3 pa so apache, MySQL, php, ki predstavljajo del paketa za hitro instalacijo. Typo3 deluje le v paketu z Apache, MySQL in php. Skupnost raste iz leta v leto, in zavedajo se, da je od nje odvisna prihodnost. Zato se od leta 2002 se enkrat letno TYPO3 navdušenci sestajajo na Typo3 Snowboard Tour, to je njihov največji družabni dogodek. Enkrat letno prirejajo tudi konferenco T3CON in T3DD. Ključni komercialni akterji so Share Point Portal (Microsoft), Drupal, Joomla, SilverStripe, Oracle in vsi ostali ponudniki CMS orodij. Partnerji v poslovnem okolju in odprtokodnih skupnostih opravljajo različne aktivnosti. Typo3 partner vključuje zvezo vodilnih ponudnikov Typo3 storitev v Nemčiji. Več kot 40 strokovnjakov iz devetih lokacij po vsej Nemčiji ponujajo res popolne storitve z vrsto storitev od spletne agencije do samega sistema za internet, intranet, extranet, in IT spletne rešitve. Partnerji t3n.de magazin, typo3 artplan(21) webconcepts & solutions, digiparden, fishfarm netsolutions, naw.info net&works Internet Services, network.publishing GbR,punkt.de, saltation GmbH & Co. KG. Komunikacije z odprtokodno skupnostjo potekajo s pomočjo spletne strani typo3.org, seznamom elektronske pošte in IRC klepeta. Tu se srečujejo ljudje v virtualnem prostoru, da si pomagajo med seboj in razpravljajo o novih razvojnih programih. Obstaja tudi Blog, imenovan Buzz, ki vam lahko prikaže drugačen pogled na Typo3 skupnost.

Dohodkovni model. Typo3 je najbolj iskan CMS na evropskem trgu in se v celoti financira z donacijami in članarino. Več kot 300.000 podjetij po celem svetu, med njimi tudi nekaj zelo velikih, kot so Deutsche Telekom, Basf, TUI, Sixt itd, in nekatere velike organizacije, kot so Greenpeace, UNICEF, UNESCO in WWF. Tudi te organizacije uporabljajo Typo3 in pridobivajo koristi iz popolnoma funkcionalne CMS rešitve brez plačila licenc. Typo3 je odprtokodno orodje, ki je odvisno od prispevkov večje skupnosti, tako tehnične kot finančne. Za poenostavitev postopka doniranja in z namenom spodbujanja čim večjega kroga ljudi k darovanju in rasti projekta je bila vzpostavljena spletna stran typo3.org. Sprejemajo kakršnekoli donacije različnih sredstev, do članstva združenja. Članarina je edini način za pripravo proračunskega načrta in zagotavljanje nadaljnjega učinkovitega razvoja. Članarina znaša od 100 € do 10.000 €, kar omogoča podjetjem vseh velikosti možnost, da redno prispevajo k razvoju Typo3. Izračun cen posameznih vtičnikov pa ni smislen, saj skupnost deluje po načelu dam – daš. Izračun

osnove za ceno programskih rešitev osnovnega CMS orodja je nemogoč, saj je to orodje, ki nastaja kot izdelek enega posameznika, ki je ugotovil, da ni poslovnež, ampak programer in če je želel naprej programirati, je program odprl skupnosti in ustanovil skupnost Typo3. Strošek uporabe odprtokodnih programskih rešitev je minimalen, z aktivnim članstvom podjetja v skupnostih ter s sodelovanjem pri raziskavah in razvoju. Na ta način je podjetje seznanjeno s tekočim razvojem in smernicami razvoja za dolgoročni načrt proračunskih sredstev.

4.2.3 Netgem

Netgem zagotavlja tehnologijo, znanje in prilagodljiv poslovni model za pomoč telekomunikacijskim operaterjem pri razvoju inovativnih konvergenčnih storitev s kombinacijo najboljšega, kar nudijo televizija, internet in multimedijske vsebine v domače omrežje. Strojne in programske rešitve Netgema omogočajo telekomunikacijskim operaterjem, da se razlikujejo od svojih konkurentov z integracijo več virov, postavitvijo po meri, razširljivostjo, stroškovno učinkovitostjo in inovativno platformo.

Njihovi izdelki (na primer Siol Box) uporabljajo odprtokodne programske rešitve, v tem primeru operacijski sistem Linux. Netgem uporablja in integrira odprtokodne programske rešitve v svoje izdelke, a zelo malo prispeva ustreznim odprtokodnim projektom. Netgem bi rad imel določen vpliv na odprtokodne skupnosti, vendar je splošno znano, da je edini način ta, da sodeluje na odprtokodnih projektih s prispevanjem razvojne kode. V ta namen je Netgem začel razvijati odprtokodno platformo v oblaku za razvoj orodij, s katerimi bodo lahko telekomunikacijska podjetja razvijala njihove aplikacije in storitve v lokalnem ekosistemu. Platforma v oblaku je eden od načinov za zagotavljanje dragocenih povratnih informacij odprtokodnim projektom. Cilj projekta je, da bi s pomočjo odprtokodne platforme v oblaku Netgem omogočil telekomunikacijskim podjetjem različna orodja za izboljšavo njihovih produktov. Z odprtokodnim projektom želijo opozoriti na morebitne pomanjkljivosti produktov in usmeriti razvijalce k spremembam projektov v smeri, v kateri bi želel Netgem. Poleg tega Netgem namerava izkoristiti vse notranje informacije, ki jih je pridobil od IPTV tehnologije.

Ponudba. Njihova ponudba zajema celotno IPTV tehnologijo za internetne ponudnike, kot na primer tudi Siol, ki se prav tako poslužuje Netgemovih rešitev. Rešitve vključujejo tako programsko kot strojno opremo, ki se ne prodaja ločeno. Na primer, za Siol Box se programska oprema ne prodaja ločeno od strojne opreme. Poleg tega Netgem zagotavlja tudi vzdrževanje in podporo. Netgem mora prilagoditi izdelke različnim trgov, saj so telekomunikacije zelo regulirana dejavnost, kar pomeni da je trebaspoštovati nacionalne zahteve, ki veljajo za produkte.

Netgem uporablja odprtokodne programske rešitve kot del končnih proizvodov. Programska oprema, ki jo na koncu ponudi na trg ni odprtokodna rešitev. Netgem uporablja odprtokodne programske rešitve v svojih izdelkih Netbox = Siol Box. V primeru Siol Boxa je uporabljen operacijski sistem Linux. Pri uporabi odprtokodnih programskih rešitev kot del svojih izdelkov Netgem vedno pridobi programsko opremo od distributerjev. V primeru operacijskega sistema Linux je bil pridobljen od distributerja Linux. Netgem ponudi prvostopenjsko podporo svojim strankam, Linux distributer pa ponudi drugo stopnjo podpore. V primeru, da Netgem ne more rešiti določenih problemov z izdelki svojih strank, se te lahko obrnejo na distributerja operacijskega sistema Linux.

Viri. Netgem je eno izmed vodilnih ponudnikov IPTV opreme v svetu. Menijo, da so ključnega pomena prav njihovi notranji viri, bogate izkušnje na trgu telekomunikacij in dobra blagovna znamka. Netgem pridobiva nekatere vire tudi pri odprtokodnih skupnostih. Tako pri končnih produktih kot pri razvoju Netgem uporablja veliko odprtih programskih rešitev. Netgem je na primer strateški razvijalec odprtokodnega oblaka, kjer bodo lahko telekomunikacijska podjetja razvijala aplikacije in storitve v lokalnem ekosistemu za diferenciacijo od konkurence. Prednost uporabe odprtokodnih orodij v razvoju je, da ni vprašanj v zvezi z licencami. Pri Netgemu je situacija nekoliko drugačna, saj uporabljajo odprtokodne komponente pri končnih proizvodih. Z vključevanjem odprtokodnih programskih rešitev v končne produkte mora biti Netgem zelo previden pri uporabi licenc in preveriti, kaj določene licence dovoljujejo in kaj prepovedujejo in to uskladiti. Licence štejejo kot največje povračilo pri uporabi odprtokodnih programskih rešitev. Zaradi teh licenčnih vprašanj Netgem poskuša uporabljati odprtokodne programske rešitve kot take, ne da bi jih spreminjali. V primeru ko morajo spremeniti komponente, to prepustijo distributerju, da opravi vse potrebno v zvezi s spremembami.

Pri Netgemu menijo, da so odprtokodni sestavni deli cenejši za uporabo kot lastniška programska oprema, vendar to ni glavni razlog za uporabo odprtokodnih programskih rešitev. Netgem vidi v odprtokodnih skupnostih in komponentah prihodnosti rešitev, saj gospodarske družbe postajajo vedno manjše in njihov obstoj je dolgoročno negotov.

Odnosi. Netgem nima veliko interakcije z odprtokodnimi skupnostmi. Menijo, da skupnosti ne obožujejo velikih podjetij in v njihovem primeru se ne posebno zanimajo za njihove produkte. S skupnostjo komunicirajo predvsem preko posameznikov, ki delajo pri njih, in so obenem tudi člani posameznih skupnosti. Ti člani ne predstavljajo Netgema, ko sodelujejo v skupnostih, saj je po francoski zakonodaji v lasti podjetja vse, kar zaposleni počnejo v prostem času, če je to povezano z delom v službi.

Pri Netgemu imajo projekte in pobude za oblikovanje tesnejših odnosov z odprtokodnimi skupnostmi. Imajo spletno stran, kjer so objavljene vse informacije v zvezi z udeležbo Netgema v odprtokodnem projektu. Na spletni strani je na voljo razvojna koda z

odprtokodno licenco. Netgem je tudi eno izmed podjetij, ki podpirajo razvoj odprtokodnih rešitev OSDL-ja in v ta namen prispevajo sredstva razvijalcem odprte kode, ker se zavedajo pomembnosti Linuxa v telekomunikacijah in v podjetjih. S partnerji, vpletenimi v IPTV tehnologijo, je Netgem razvil neke vrste načrt za prihodnje distribuiranje IPTV-ja tehnologije. Njihov namen je ustvariti vizijo in smernice za povečanje IPTV-ja, za zadovoljevanje potreb tako ponudnikov interneta kot tržnih segmentov. Na ta način želijo zagotoviti, da bi IPTV še naprej učinkovito in na enoten način tudi v prihodnosti podpiral zahteve internetnih ponudnikov.

Objavili so tudi izvorno kodo poslovne programske opreme imenovano Home Cloud software platform. Home Cloud software platform je orodje, s katerim bodo lahko telekomunikacijska podjetja razvijala aplikacije in storitve v lokalnem ekosistemu. Cilj projekta je internetnim ponudnikom pomagati izboljšati produkte za zahtevne internetne uporabnike. Netgem želi uporabiti to platformo v oblaku za spodbujanje razvoja projektov odprte kode v želeno smer. Namenjen je medijskim strežnikom in večpredstavnim rešitvam s poudarkom na povezanem TV-ju. Na ta način si prizadevajo okrepiti sodelovanje v odprtokodnih projektih in okrepiti tudi odprtokodne sposobnosti, ki jih podjetja potrebujejo. Menijo, da je projekt odprtokodna platforma v oblaku eden od načinov za spodbujanje in zahvala odprtokodnim skupnostim.

Dohodkovni model. Njihov glavni vir dohodkov so IPTV rešitve, ki jih prodajajo velikim internetnim ponudnikom (na primer Siol). Na splošno so ta pogajanja ki se ukvarjajo z omrežnimi rešitvami, tako velika, da stranke običajno ne plačajo v enem obroku.

Dohodke ustvarjajo tudi s prodajo podpore in vzdrževanja. Uporabljajo odprtokodne komponente, a te nimajo pomembne vloge pri zagotavljanju finančne uspešnosti, ker njihova uporaba povzroča določene dodatne stroške, kot so provizije, ki jih morajo plačati distributerjem.

4.2.4 IBM

IBM je dokazal, da odprtokodni poslovni model ni samo možen, temveč tudi zelo dobičkonosen posel. IBM je uveljavljeno podjetje v strežniški industriji, ki je sprejelo odprtokodne programske rešitve. Njihova linija izdelkov vključuje strežnike, kot so System x, p, i, z, Power sistemi, Blade center in sisteme za shranjevanje. IBM je bil nekoč glavni igralec na področju stacionarne opreme in je zato razvijal in prodajal lastniške programske rešitve vključno z lastnim operacijskim sistemom (AIX, OS/390), ki ga je nadomestil Linux, in spletne storitve, ki ne delujejo na lastniškem strežniku Lotus Domino Go, ampak na odprtokodnem spletnem strežniku Apache. S pojavom interneta se je IBM moral prilagoditi novemu okolju, kjer delujejo strežniki, spletne storitve, velike podatkovne baze in ogromen pretok medijskih vsebin na osebne računalnike. Podjetja povprašujejo po zanesljivih strežnikih za delovanje njihovih aplikacij, saj čas, ko strežnik ne deluje, pomeni

izgubljene prodajne priložnosti, jezne kupce in izgubo produktivnosti za podjetja. Največja prednost IBM-a je ta, da prodaja celotne rešitve skupaj s strojno opremo, za kar nudi tudi izobraževanje, podporo in vzdrževanje.

Ponudba. Ponudba zajema široko paleto strojne in programske opreme. IBM je eno izmed največjih podjetij na področju ponudbe strojne in programske opreme. Med njihove prednosti štejejo vodilni položaj na trgu, dobro blagovno znamko in ugled. Programske rešitve so vedno del strojne opreme. Te so predhodno nameščene v tovarni ali pri naročniku. Strežniška strojna oprema temelji na operacijskem sistemu Linux, spletne storitve pa na Apache spletnem strežniku. IBM skrbi za sistem 10 let. To včasih zaplete poslovanje zaradi velike baze namestitev. To pomeni, da ne morejo spremeniti arhitekture sistema in programskih rešitev, saj morajo nove verzije delovati tudi na starejših sistemih. IBM uporablja odprtokodne prakse pri razvoju programskih rešitev za strojno opremo. Različne programerske skupine programirajo programske rešitve za systemske skupine, ki integrirajo programske rešitve v končne sisteme. IBM ima notranje dobavne verige za komponente programskih rešitev. Komponente so posredovane s pomočjo intraneta, elektronske pošte, ftp-ja in CD-ja. Poslovne enote izberejo komponente, ki jih želijo uporabiti in nato prilagoditi ali nastaviti. Poleg tega si skupine med seboj ponujajo storitve. Te storitve so na primer podpora in vzdrževanje in zato obstajajo različni sporazumi med različnimi notranjimi skupinami. IBM uporablja odprtokodni operacijski sistem Linux in Apache spletni strežnik, ki je implementiran v strojno opremo.

Viri. Med svoje ključne notranje vire štejejo zaposlene, ki imajo znanje o strojni opremi in razvoju programskih rešitev, še posebej za strežniško, strojno in programsko opremo. IBM je primer zaposlenih, katerih naloga je ugotoviti, če obstajajo kakšne funkcije, ki lahko pripomorejo k boljšemu opravljanju dela končnega uporabnika sistema. Na podlagi povratnih informacij iz prakse lahko programerji določene funkcionalnosti izboljšajo oziroma naredijo nove dodatke. Za IBM je zelo pomembno slediti razvoju različnih tehnologij, kot so razvoj platform, operacijskih sistemov, spletnih strežnikov, podatkovnih baz. Dell na primer načrtuje vstop na strežniško področje z novimi tehnologijami in IBM želi videti, ali bo Dell zagotovil funkcionalnosti v svoje tehnologije, ki bi jih lahko IBM uporabil.

Notranji viri ponujajo različne sestavne dele programskih rešitev poslovnim enotam. Pri uporabi notranjih virov se kažejo dobre možnosti za ponovno uporabo sestavnih delov in izogibanje dvojnemu delu. To vodi k izboljšanju učinkovitosti razvoja. Notranji pristop dela omogoča izdelavo celotnega paketa izdelkov skupaj. Ideja je, da ena skupna platforma določa splošne funkcionalnosti za vse poslovne enote in posamezne poslovne enote lahko nato izdelajo razširitve na tej platformi, ki temelji na njihovem strokovnem znanju. Zato je razvoj programskih komponent potekal v poslovni enoti, ki je imela največ znanja o komponenti. Posledica tega je izboljšanje kakovosti. Poleg tega se bo splošno razumevanje

in znanje o celotnem sistemu izboljšalo zaradi skupnega razvoja. Različne skupine sodelujejo pri razvoju določenih funkcij in tako lahko spoznajo celotno funkcionalnost in natančno tehnično implementacijo.

Prav tako je težje zagotoviti kakovost programske opreme, če različne poslovne enote razvijajo samostojne dele za platformo. Problem je ,kako zagotoviti, da bodo vse poslovne enote prispevale k razvoju in ne samo uporabljale rezultatov. Običajno se pričakuje od razvijalca programske komponente ali sestavnega dela, da poskrbi tudi za vzdrževanje. To lahko privede do situacije, ko nekatere poslovne enote namenijo veliko sredstev za razvoj in vzdrževanje sistema, druge enote pa zgolj uporabijo rezultate.

Odnosi. Notranja skupina je primarno razvojno in inovacijsko naravnana. Obstaja veliko različnih mehanizmov komunikacije znotraj skupine in z odprtokodnimi skupnostmi. Neformalno sodelovanje poteka s pomočjo omrežij za sodelovanje, saj je pri skupnem razvoju veliko neformalnih skupin in komunikacij med njimi.

Pomembnejše zadeve pa se rešujejo na formalnih sestankih tako z voditelji ekip kot z operativnimi skupinami. Formalna zasedanja o tehnoloških zadevah potekajo tudi med arhitekti sistemov. Cilj je jasno sporočilo in izmenjava informacij z vsemi zainteresiranimi skupinami. Na primer, skupina, ki skrbi za platformo, ima mesečni operativni sestanek, ki se ga lahko vsakdo udeleži. Informacije o sestanku so objavljene na intranetu, tako da lahko vsakdo dostopa in jih prebere. Na ta način lahko druge skupine v podjetju dobijo vse potrebne informacije v zvezi s platformo.

Eden od načinov komunikacije med različnimi skupinami podjetij poteka s pomočjo tako imenovanega predstavnika. Ti notranji predstavniki zastopajo različne skupine z različnimi dejavnostmi in komunicirajo med sabo po določenem komunikacijskem kanalu. IBM ima elektronske poštne sezname, katerim se lahko vsakdo pridruži. Poleg tega ima IBM tudi partnersko mrežo. To so predvsem dobavitelji delov strojne opreme, kot na primer Intel in AMD. Tudi odprtokodne skupnosti programirajo sestavne dele programskih rešitev za IBM.

IBM namenja posebne ukrepe za spodbujanje vključevanja v odprtokodne skupnosti. Odkar je IBM postal sponzor finala ACM-ICPC World leta 1997, ko je na tekmovanju sodelovalo 840 ekip, se je udeležba povečala na več kot 7.100 ekip, ki vključujejo več deset tisoč najboljših študentov in profesorjev na področju računalniških disciplin iz več kot 1800 univerz iz skoraj 90 držav, iz 6 kontinentov. IBM-ovo sponzorstvo finala ACM-ICPC World je pomembna komponenta številnih podjetij, akademskih iniciativ, ki so namenjeni za spodbujanje odprtokodnih programskih sposobnosti za razvoj bolj konkurenčne informacijske tehnologije, ki je zmožna peljati svetovne inovacije in gospodarsko rast.

V IBM menijo, da bo sodelovanje pri razvoju motiviralo in spodbudilo razvijalce. S sodelovanjem razvijalci postanejo odgovorni za nekatere komponente in tako bolje razumejo delovanje med različnimi elementi. To razumevanje in sodelovanje na področju razvoja bo okrepilo članstvo in razvoj skupnosti. Z razvojnim delom se ukvarjajo usmerjevalne skupine, na podlagi operativnih sestankov se določi, kaj se razvija, vendar pa je doslej bil razvoj posledica poslovnih potreb po novih funkcionalnostih. Razvoj je prav tako pod nadzorom z različnimi komunikacijskimi mehanizmi. V skladu z notranjim pristopom bi morale poslovne skupine najprej ugotoviti, če je kakšna skupina že razvila ustrezno komponento, preden se poda v lasten razvoj. Če ni na voljo nobena primerna komponenta, se je treba odločiti, ali se gre v lasten razvoj skupaj z drugimi skupinami ali z odprtokodno skupnostjo. Poslovne skupine bi se morale izogibati lastnega razvoja, še posebej v primerih, ko je nekdo že razvil ustrezno komponento ali če ima katera skupina oziroma skupnost več strokovnega znanja, da to stori. IBM priznava, da je notranji pristop pri uvajanju odprtokodnih programskih rešitev prinesel tudi določene izzive. Prvi izziv je, da ni nikakršne centralne skupine, ki bi usklajevala razvoj. To je eden od načinov, kako notranje skupine sprejemajo odločitve. Drugi izziv je ta, da morajo poslovne skupine sodelovati in doseči ravnotežje med različnimi notranjimi skupinami in odprtokodnimi skupnostmi pri razvoju novih komponent. Na primer, ko skupina, ki razvija platformo, doda nove funkcionalnosti na platformo, se lahko pojavi problem, da poslovna skupina ne bo sprejela novih funkcij tako gladko kot drugi. To je razlog, zakaj mora skupina, ki razvija platformo, upoštevati sposobnosti različnih poslovnih skupin, da lahko enostavno sprejmejo spremembe.

Dohodkovni model. IBM je leta 2001 vložil 1 milijardo dolarjev v operacijski sistem Linux in ugotovil, da je bil vložek dobičkonosen, saj je v letu 2002 zaslužil 1,5 milijarde dolarjev od operacijskega sistema Linux, ki ga je vgradil v strežnike. Več kot 6300 njihovih strank uporablja operacijski sistem Linux na njihovi strojni opremi. Bolj kot IBM podpira Linux, bolj ga uporabniki uporabljajo in več kot ga uporabljajo, več smisla ima za podjetja da se odločijo za uporabo Linuxovih rešitev. To koristi IBM-u, ki je premierni ponudnik Linuxovih rešitev za strežnike, svetovalne storitve in programske aplikacije, ki delujejo na platformi Linux. Odprta koda omogoča IBM-u znižanje stroškov. IBM ima sedaj na tisoče programerjev iz celega sveta, ki delajo brezplačno, da bi bil Linux bolj stabilen. Povedati je treba, da je veliko programskih rešitev že na voljo brezplačno tako za Linux kot za Apache. IBM se je odločil za uporabo operacijskega sistema Linux, takoj ko je bilo na voljo dovolj brezplačnih programskih rešitev ki jih lahko IBM ponudi njihovim strankam brez dodatnih stroškov za IBM. Vseh programskih rešitev ni treba več razviti znotraj podjetja, kar pomeni, da se določena vsota denarja sprosti in posledično lahko IBM investira v druga področja raziskav in razvoja.

4.2.5 Nokia Symbian

Leta 1998 so Ericsson, Nokia in Motorola ustanovili partnersko podjetje Symbian za razvoj programske rešitve, ki bi omogočala konvergenco med dlančnikom in mobilnim telefonom. 10 let kasneje je podjetje Symbian vztrajalo za priljubljenim operacijskim sistemom Symbian za mobilne telefone, kar je tudi spodbudilo Nokio, da je odkupila lastniške deleže ostalih podjetij. Nokia je leta 2008 za preostali 52 % lastniški delež plačala okrog 264 milijonov eurov. Kot 100 % lastnik se je Nokia odločila, da bo operacijski sistem Symbian ponudila brezplačno v obliki odprtokodne programske rešitve. Eden od najpomembnejših razlogov za takšno poslovno odločitev Nokie je bil ta, da so druga podjetja želela obdržati zaprto kodo za operacijski sistem Symbian. Dokler je bila programska rešitev vzdrževana s strani podjetja Symbian, izvorna koda ni bila vidna, rezultati niso bili vedno zaupljivi s strani drugih podjetij. Ko pa so operacijski sistem ponudili kot odprtokodno rešitev, kjer je vsakdo imel priložnost, da je kodo videl in sodeloval pri njenem, se je zaupanje v sistem povečalo. To bilo pričakovano, posledično so se odpravile tudi težave s komunikacijo, kar je znižalo stroške podpore. Poleg tega so bili še trije drugi razlogi, zakaj je prišlo do odločitve za odprtokodno rešitev. Eden od teh je bil porazdeljen razvoj med tremi podjetji, ki so bila na različnih lokacijah. Leta 2000 se je razvoj začel v Veliki Britaniji, na Kitajskem in v Indiji, leta 2005 pa so se razvojne aktivnosti izvajale na Japonskem, v Koreji ter v Združenih državah Amerike. Delitev razvoja na različne strani in med različna podjetja je imela velik vpliv na učinkovitost dela. Razvoj programskih rešitev je na primer potekal v različnih državah, kot zaključen zaprt del kjer programska koda ni bila vidna.

S sprejetjem odprtokodnega načina dela z objavo izvorne kode v odprto okolje (SourceFroge) so se pomanjkljivosti zmanjšale. Drugi pomemben razlog je bil v namenu, ustvariti veliko in aktivno skupnost in uporabniško skupino. S sprostitvijo programske rešitve je bilo moč Symbian operacijski sistem naložiti iz mnogih spletnih strani, kot so sourcefroge.net, symbian-freeware.com, zato se je odziv uporabnikov zelo povečal. Tretji razlog je bil poizkus v vključitvi več podjetij v razvoj Symbiana in s tem zmanjšati razvojne stroške.

Ponudba. Ponudba Nokie vključuje odprtokodni operacijski sistem Symbian za pametne mobilne telefone. Cilj tega projekta je ustvariti uporabniku prijazen operacijski sistem, ki bi dovolil telekomunikacijski industriji, zmanjšati težave pri združevanju. Drugi cilj je v tem, da bi ustvarili veliko skupnost za testiranje Symbiana. Operacijski sistem Symbian je integriran v različne mobilne telefone različnih proizvajalcev, kot so Nokia, Sharp, Motorola, Samsung, LG, in Sony Ericsson. Symbian operacijski sistem je bil zasnovan za generacijo pametnih mobilnih telefonov. Uvaja nove funkcije, novo arhitekturo, 2D in 3D grafiko, podporo za zunanje zaslone preko HDMI, boljši izkoristek baterije, boljša je tudi združljivost z drugimi napravami, še posebej z izmenljivimi mediji. Ciljni trg Symbiana somobilna telekomunikacijska industrija in podjetja, ki ponujajo mobilne telekomunikacijske programske rešitve. Symbian je operacijski sistem, ki je najpogostejše

uporabljen v pametnih telefonih. Nokia ponuja v sklopu Symbian fundacije dopolnilne storitve, kot so podpora uporabnikom, in različne vtičnike kot dodatke operacijskemu sistemu. Tako se lahko osredotočijo na različne ciljne skupine in jim ponudijo prilagojene vtičnike za lažje upravljanje. Poleg tega Nokia ponuja na spletni strani skupnosti Symbian bogat nabor testiranj, različnih video in tekstovnih scenarijev, forum, kjer lahko uporabniki Symbiana komunicirajo z odprtokodno skupnostjo. Izvorno kodo operacijskega sistema Symbian si je mogoče prenesti z različnih spletnih strani, kot so sourcefroge.net, symbian-freeware.com ... Symbian je licenciran pod licenco Eclipse Public License.

Viri. Pri Nokii menijo, da so ključni njihovi notranji viri ter njihovo znanje o operacijskem sistemu Symbian. Nokia predstavlja aktivni del razvoja Symbiana z močnim strokovnim znanjem o programiranju. Imajo zaposlene v Veliki Britaniji, na Kitajskem, v Indiji, na Japonskem, v Koreji in Združenih državah Amerike, ki delajo na tem projektu. Poleg zaposlenih sodelujejo tudi s partnerji, ki pomagajo pri razvoju: Sonim Technologies, Samsung... Pomanjkljivost premajhnega števila programerjev z dobrim znanjem so odpravili s tem, ko so ponudili operacijski sistem v odprtokodni različici. S tem so veliko lažje pridobili nove vire za projekte Symbian.

Izvorna koda Symbiana je odprta. Pri Nokii vidijo prednost tega pristopa v tem, da ljudje vedno bolj zaupajo v odprtokodne programske rešitve. To pa zato, ker uporabniki lahko vidijo, da ni nobenih skritih funkcij in kako je sistem implementiran. Doslej Nokia ni pridobila veliko virov za razvoj iz odprtokodne skupnosti, čeprav je izvorna koda Symbian na voljo že eno leto. Symbian skupnost šteje cca. 200 podjetij iz različnih področij, kot so izdelovalci naprav (na primer Nokia, Fujitsu), finančne družbe (na primer Visa), mobilni operaterji (na primer China Mobile, Vodafone, AT&T), programerska podjetja, strokovno storitvena podjetja itd. Podjetja poročajo o napakah, razvijajo nove vtičnike in jih tudi preizkušajo. Skupnost v bistvu pomaga pri testiranju programskih rešitev. Symbian in partnerji na podlagi povratnih informacij programske rešitve potem izboljšajo. Aktivna skupnost pripomore z uporabo programskih rešitev v različnih situacijah in zagotavljanjem povratnih informacij k izboljšanju njihove kakovosti. Pri Nokii pomanjkanja sodelovanja ne jemljejo kot grožnjo, saj menijo, da je vsaka udeležba zunanjih uporabnikov koristna, saj jim s tem ni treba narediti celotnega dela v lastni režiji. Imajo aktivni razvoj skupnosti Symbian Foundation, kjer razvojne vire prispevajo Nokia, Sony Ericsson, Motorola, Texas Instruments, Vodafone, Fujitsu, Qualcomm, STMicroelectronics in AT&T.

Odnosi. Pri Nokii razvijajo programske rešitve v sodelovanju z več kot 200 podjetij, kot so Nokia, Sony Ericsson, Motorola, Texas Instruments, Vodafone, Fujitsu, Qualcomm, Visa, STMicroelectronics in AT&T, ki vsa tudi sodelujejo v Symbian Foundation. Dvakrat do največ trikrat na leto ima usmerjevalni odbor Symbian Foundation sestanek, na katerem se odločajo o tem, kako bodo izpeljali preostanek leta. Vse teme sestankov so predhodno na voljo tudi na internetu, tako da lahko vsi vidijo vprašanja in smer, v kateri bo potekala

razprava. Pri Nokii menijo, da tudi če nekdo od partnerjev nenadoma zapusti projekt, to projekta ne bo ogrozilo. Po drugi strani pa je projekt z vsakim izstopom partnerja težje upravljati.

Komuniciranje s skupnostjo uporabnikov poteka preko spletne strani Symbian Foundation. Na spletu lahko uporabniki poročajo o ugotovljenih napakah, zastavljajo vprašanja ter razpravljajo o različnih temah. Preden je Symbian postal odprtokodni operacijski sistem, so poročila o napakah prispevali le Nokia, Sony Ericsson, Motorola. Danes, ko je Symbian postal odprtokodna programska rešitev, prihajajo poročila iz številnih podjetij in s strani različnih uporabnikov. Nokii veliko pomeni njihova odzivnost in vpletenost v skupnosti, saj menijo, da na ta način pripomorejo k temu, da je skupnost aktivna in tako izkazujejo njihovo resnost do skupnosti. Če se na primer pojavi vprašanje o določeni napaki v sklopu operacijskega sistema, se zavzemajo, da se nanj odgovori v enem dnevu.

Pri Nokii organizirajo različne konference in zabave za ljudi, ki so zainteresirani za Symbian. Tako poskušajo spodbujati uporabo operacijskega sistema in pritegniti razvijalce, da sodelujejo pri razvoju kode. Celoten proces razvoja in načrt morata biti jasna in odprtokodna, da se lahko omogoči sodelovanje. Veliko podjetij se zanima za Symbian, ker veliko mobilnih telekomunikacijskih podjetij uporablja ta operacijski sistem. S sodelovanjem pri razvojnih projektih bi podjetja lahko izvedela več o Symbianu. Pri Nokii nudijo tudi pomoč, podporo in svetovanje vsem, ki se zanimajo zanj.

Dohodkovni model. Operacijski sistem Symbian je brezplačen, zato ne ustvarja neposrednih dohodkov. Poleg podpore nudijo tudi svetovanje za pametne telefone. Trenutno je to edini vir dohodka. Pri podjetju Nokia Symbian proučujejo tudi možnost, da bi ponudili tudi komercialne storitve v sklopu operacijskega sistema, a do danes tega še niso izvedli. Razvoj se financira iz drugih poslovnih skupin. Symbian ima pri njih neke vrste posreden vpliv na dohodke, saj se navezujejo predvsem na večjo publiciteto, kar ima lahko pozitiven vpliv na njihove skupne dohodke. Drug posreden vir dohodkov izhaja iz zmanjšanja težav z operacijskim sistemom. S pomočjo skupnosti se lahko izboljša kakovost operacijskega sistema Symbian in posledično tudi stroški storitev in podpore.

4.2.6 Iskra Avtoelektrika d.d.

Iskra Avtoelektrika (v nadaljevanju Iskra) je globalna proizvajalka zaganjalnikov, generatorjev, električnih pogonskih sistemov, mehatronskih sistemov in delov za avtomobilsko, logistično in drugo industrijo, ki razvija, proizvaja in trži globalno, z lastno proizvodno in prodajno-distribucijsko mrežo. Iskra je implementirala odprtokodni intranetni portal, imenovan e-portal. E-portal je internetno orientirana programska rešitev in orodje za sodelovanje. Projektni portali, kot je e-portal, so uspešni za gostovanje projektov odprtokodnih programskih rešitev ter za zagotavljanje orodij za distribucijo

razvoja. E-portal zagotavlja generični komplet orodij za notranji pristop in prožnost pri prilagajanju na procese podjetja. Prvi demo e-portal je bil uveden leta 2001, z namenom povečanja notranje dejavnosti podjetja. Da postanejo programske rešitve prepoznavne, je treba poslovne koristi doseči od interesov poslovnih enot in z omogočanjem uporabe rešitev onstran organizacijskih omejitev.

Ponudba. Torej, ponudba je intranet portal, imenovan e-portal, ki omogoča skupinski nadzor in upravljanje razvoja produktov. Cilj portala je okrepiti notranje dejavnosti, kot je razvoj Iskrinih produktov. S tem pristopom si v Iskri prizadevajo predvsem za učinkovito ponovno uporabo obstoječih sredstev ter stroškovno učinkovit nadaljnji razvoj produktov. Za posamezne poslovne enote je prav tako koristno, saj se lahko osredotočijo na njihove ključne kompetence in storitve globalnega pomena, kot sta na primer vzdrževanje in usposabljanje, druga vprašanja prepustijo enoti za podporo. E-portal je orodje z različnimi uporabniki, kot so poslovne enote, ki lahko odločajo, kako bodo to orodje uporabljali. Iskra je prepoznala tri različne možnosti uporabe portala. V prvi vrsti je to notranja uporaba, kjer imajo poslovne enote njihova projektna sredstva in razpoložljiva izhodna sredstva, do katerih ima lahko dostop kdorkoli v podjetju.

E-portal je bil razvit na osnovi potreb notranjih skupin. Projektna skupina skrbi za razvojne projekte e-portala. Njeni člani so glavni razvijalci iz različnih poslovnih skupin in skrbijo za nadaljnji razvoj ter odločajo o implementaciji novih orodij. Če poslovna enota želi uporabljati orodje, ki ni na voljo v e-portalu, ga lahko predlagajo razvojni skupini in le-ta jo bo implementirala v portal. Na podlagi različnih potreb uporabnikov obstajajo različna orodja e-portala. Z orodjem za preverjanje verzij v e-portalu enostavno nadziramo verzije samega portala in verzije ostalih produktov. Portal ponuja tudi orodja, ki omogočajo sodelovanje med skupinami ter sistem licenciranja. Ta ni problematičen, dokler se uporablja zgolj interno. V nasprotnem primeru, pri prodaji za zunanjo uporabo, pa je treba pridobiti dovoljenje.

Največje koristi z uporabo odprtokodnih rešitev pri Iskri vidijo predvsem v zelo nizkih stroških implementacije, saj veliko ljudi pozna odprtokodna orodja. Lamp je odprtokodni standard, uporabljen v različnih šolah, tako da je olajšano tudi delo s podizvajalci, ki že poznajo ta orodja. Med prednosti pri uporabi lastnega razvoja za programske rešitve sodita tudi vedno dosegljivi pomoč in podpora. E-portal se nadgrajuje dvakrat letno. E-portal uporablja celotna skupina v podjetju.

Viri. V Iskri so implementirali troslojni model za nudenje pomoči in podpore. Glavni podporni model temelji na trinivojski hierarhiji tako za uporabnike znotraj podjetja kot za celotno skupino. Pri uporabniku podjetja so to projektni vodja e-portala (3. stopnja), vodja oddelka (2. stopnja) in uporabniki (1. stopnja). Nivoji v podporni skupine podjetja so napredna podpora (3. stopnja), aplikacijska podpora in operacijski center (2. stopnja) ter

tehnična podpora (1. stopnja). Glavni namen takšne organiziranosti je v tem, da različni nivoji v podjetju razpravljajo med seboj in da so z različnimi zahtevami ali ponavljajočimi se težavami seznanjeni tudi višji nivoji. Vse enote imajo tudi prilagojena usposabljanja glede na njihove potrebe. Ko se na določenem projektu nekaj razvija, se najprej preveri, če je že na voljo ustrezna komponenta v e-portalu. Pri tem lahko sodelujejo različne skupine, ki lahko komponente prispevajo ali si jih med seboj delijo. Težava notranjih virov in odprtokodnih programskih rešitev je v tem, da morajo razpolagati z jasnimi vmesniki in zelo dobro dokumentacijo za ponovno uporabo pri nadaljnjem razvoju. Prav tako je treba biti pri projektih prepričan, da je omenjena komponenta primerna za posamezen projekt. Iskra se poslužuje tudi zunanjih virov. To še posebej velja za razvoj programskih rešitev. Približno polovico delovnih ur opravijo podizvajalci. Podizvajalci nudijo tudi podporo 1. in 2. stopnje, kjer imajo nekateri izmed njih tudi dostop do e-portala. Zaradi tega so potrebni tudi viri s področja zagotavljanja varnosti, ki preverjajo, če razvoj poteka po standardih Iskre. Odprtokodne komponente vgrajujejo tudi v aplikacije in operacijske sisteme. Zato tudi potrebujejo specialiste, ki so primerno usposobljeni, da lahko vključijo različne komponente v različne aplikacije. Aktivno tudi spremljajo razpoložljive odprtokodne komponente. Vse odprtokodne komponente uporabljajo takšne, kot, in jih ne dodatno razvijajo zaradi vprašanj dovoljenja in varnosti.

Odnosi. Razvoj e-portala temelji na povratnih informacijah uporabnikov. Zbrane informacije prediskutirajo projektni administratorji in ključni uporabniki, saj dobro poznajo poslovanje, poznajo orodja in lahko pomagajo drugim pri uporabi e-portala. Poleg tega obstajajo tudi skupni sestanki, ki temeljijo na usmerjanju razvoja. V osnovi se razvoj izvaja po zahtevah uporabnikov. Podporna organizacija skrbi za razvoj in vzdrževanje ter podporo uporabnikom, ki temeljijo na dogovorjenih sporazumih s poslovnimi enotami. Ključni uporabniki sodelujejo s končnimi uporabniki, podporno organizacijo ter projektnimi administratorji, ki upravljajo z različnimi notranjimi viri. Med njimi je veliko neformalne komunikacije, kot je na primer e-pošta.

V poslovnih enotah so ključne osebe, ki poskušajo promovirati e-portal in delujejo kot promotorji e-portala. Te osebe imajo tudi zelo pomembno vlogo v večini poslovnih enot, ker tam ni formalnega menedžmenta, ki bi lahko sprejel odločitve za uporabljanje e-portala. Zanimanje za e-portal izvira po hierarhiji od spodaj navzgor, nekaj pa je tudi takšnih, ki so se prostovoljno odločili za uporabo e-portala. Trenutno je samo ena oseba, ki v podjetju skrbi za komercialne aktivnosti, kot je letni proračun za prihodnje leto. To je namerno odločeno, ker ne želijo centralizacije vodenja, ampak stremijo k decentralizaciji, kot to počnejo v odprtokodnih skupnostih. Trenutno tudi ne stimulirajo skupnosti, se pa zavedajo, da bo to potrebno. Skupnost vidijo bolj kot enega izmed projektov, na katerega nočejo vplivati. Eden od načinov za komuniciranje s skupinami je tudi redna vsakoletna raziskava o zadovoljstvu, ki se uporablja, da bi dobili širše razumevanje o potrebah uporabnikov.

Dohodkovni model. Poslovne enote skrbijo za stroške e-portala. Ti stroški temeljijo na uporabnikih. Poslovne enote na primer plačujejo določeno vsoto na uporabnika za uporabo e-portala mesečno. Ti stroški so ocenjeni na vsake pol leta in so procentualno razdeljeni med poslovne enote na podlagi števila uporabnikov. Razvojni projekti imajo svoj proračun. Tukaj so stroški zaračunani mesečno skupinam, ki sodelujejo na projektih. Pojavi se tudi možnost brezplačne uporabe, saj nekatere poslovne enote ne sodelujejo pri razvoju, ampak samo pri končni uporabi. V Iskri so naredili oceno stroškov med e-portalom in komercialnimi produkti, kot je SharePoint, kjer se je e-portal izkazal kot cenejši. Zaupajo bolj v interne storitve in to je tudi eden izmed razlogov za implementacijo lastnega portala. Med prednosti štejejo tudi ustvarjanje lastnih orodij, saj so tako bolj upoštevane njihove specifične potrebe. Stroški uporabe odprtokodnih programskih rešitev predstavljajo tako cenejšo varianto kot uporaba komercialnih programskih rešitev.

Pri uporabi odprtokodnih programskih rešitev začetnih stroškov ni, a kljub temu uporaba ni brezplačna. Stroški prihajajo iz različnih virov, kot pri uporabi komercialnih rešitev. Pri odprtokodnih programskih rešitvah je tudi potrebno več dela za integracijo komponent, dokumentacija je lahko manj obširna kot pri komercialnih produktih, pa tudi podpora je lahko vprašljiva oziroma dražja. V podjetju verjamejo, da je uporaba odprtokodnih komponent bolj fleksibilna, saj omogoča uporabo in vgradnjo zgolj tistih komponent, ki temeljijo na njihovih potrebah. Poleg tega so te komponente velikokrat tudi bolj napredne kot pri komercialnih produktih.

4.2.7 SugarCRM

SugarCRM je programski paket za upravljanje odnosov s strankami, ki je prisoten tako v odprtokodni brezplačni različici kot v komercialni. Ponuja bogate zmogljivosti za podporo poslovnih procesov, ki služijo za povečanje učinkovitosti na področju marketinga, prodaje, izboljšanja zadovoljstva kupcev ter ponuja možnost pregleda o učinkovitosti podjetja. Podjetje SugarCRM ponuja osnovno različico programa brezplačno. Ta vsebuje vrsto uporabnih funkcij, ki lahko izboljšajo poslovanje podjetja, prav tako pa služijo kot testno okolje, da lahko podjetje predvidi, ali je izbira CRM sistema ustrezna. Programski paket je zasnovan na tehnologiji LAMP. Več kot 800 podjetij zaupa odprtokodnim različicam Sugar Professional in Sugar Enterprise. Poleg tega ima na tisoče podjetij odprtokodno različico Sugar CRM, za kar je SugarCRM najbolj priljubljena in uspešna poslovnoodprtokodna CRM rešitev na svetu.

Ponudba. SugarCRM ponuja trgu odprtokodno CRM¹⁹ programsko rešitev za upravljanje odnosov s strankami v mikro, majhnih, srednjih in velikih podjetjih. Uporabnik lahko izbira med SugarCRM brezplačno različico, Sugar Professional različico za 360 dolarjev

¹⁹ Upravljanje odnosov s strankami

na uporabnika letno ter Sugar Enterprise različico za 600 dolarjev na uporabnika letno. Prilagojen je za različne trge s kar desetimi jezikovnimi različicami. Programska rešitev je zelo enostavna za namestitvev in uporabo, prilagodljiva je lahko za potrebe tržnih raziskav, tržnega komuniciranja, prodaje, podpore strankam ter poročanja. Poleg programske rešitve SugarCRM ponuja tudi storitve svetovanja, ki so sestavljene iz usposabljanja, podpore in vzdrževanja. Z različico Sugar Professional uporabnik pridobi: integracijo z MS Outlookom ter funkcije, kot so napoved prodaje, upravljanje z dokumenti, upravljanje s pogodbami, brezžični dostop, upravljanje poslovnih tokov, poročanje o aktivnostih. Z različico Sugar Enterprise pa uporabnik pridobi naslednje: Oracle 10G podpora, Oracle 9i podpora, offline sinhronizacija, SQL poročanje. Brezplačni SugarCRM se izdaja z licenco GPL (angl. General Public License), Sugar Professional in Sugar Enterprise pa sta komercialni različici. Brezplačna programska rešitev se distribuira na spletni strani. Glavne prednosti in razlogi za izbiro omenjene programske rešitve so nižji stroški ter posredno hitrejša vračanje investicij, različne možnosti postavitve oziroma gostovanja sistema ter večja prilagodljivost aplikacije potrebam podjetja zaradi odprte kode. Slabost pa se kaže v razumevanju podpore, saj je namenjena zgolj razvijalcem programskih rešitev, in ne uporabnikom.

Viri. SugarCRM je eden izmed vodilnih ponudnikov CRM programskih rešitev v svetu s 250.000 člani skupnosti, 25.000 razvijalci in z več kot 850 projekti. Med ključne vire sodi Sugar univerza, ki ponuja storitve od hitrega pregleda do podrobnega usposabljanja, kar omogoča, da iz SugarCRM uporabnik pridobi največ. Sugar Wiki je zbirka znanj, namigov in trikov za odpravljanje težav, namestitvev, konfiguracijo in uporabo. Sugar Forum pa omogoča pogovor in interakcijo z drugimi člani Sugar skupnosti o različnih temah. On-line dokumentacija je enostavno dosegljiva uporabniška in administratorska dokumentacija ter dokumentacija o razširitvah. Portal za podporo vsebuje pogosta vprašanja, povzetke, prenose za stranke, pregled podpore pri vstopu; iskanje napak – poročila o napakah v datotekah ter zahteva po novi funkciji. Demo Center omogoča izgradnjo privatnega demo primera (uporabno za hitro testiranja funkcij).

Odnosi. Ključni del SugarCRM predstavlja tudi odprtokodna skupnost uporabnikov, ki jo sestavljajo številni uporabniki in razvijalci po celem svetu. Njihove povratne informacije in prispevki pripomorejo k sooblikovanju produkta in so neprecenljiv vir pri razvoju in napredovanju produkta. SugarCRM kot podjetje je to prepoznalo in razvilo nekaj orodij za interakcijo s skupnostjo in ostalimi uporabniki po celem svetu. To so: **Sugar forumi in Wikiji.** Najlažji način za sodelovanje s Sugar skupnostjo je Sugar spletni forum. Tukaj se srečujejo SugarCRM tim, končni uporabniki, partnerji, razvijalci, ki razpravljajo o vprašanjih glede produkta, dostopne so informacije za prihajajoče vtičnike in izide zadnjih verzij ter informacije o tem, kako drugi uporabljajo SugarCRM v podjetjih. Nadzor nad forumom imata SugarCRM tim in skupnost, tako da je sodelovanje zelo enostavno. **SugarWiki** je zelo pomemben vir za nasvete in trike. Vodilno vlogo pri Wiki ima

predvsem skupnost, ki zagotavlja tudi uradno dokumentacijo produkta. **SugarExchange** je prostor, kjer lahko vsak Sugar uporabnik, ki želi razširiti Sugar funkcionalnosti, izbira med velikim številom vtičnikov, tem in jezikovnih različic, ki jih zagotovijo člani in partnerji Sugar skupnosti. SugarExchange zajema tako brezplačne kot plačljive komponente. SugarForge je razvijalski del SugarExchange, ki zagotavlja razvojno in projektno orodje za SugarCRM aplikacijo. Na SugarForge obstaja več kot 600 aktivnih projektov v več kot 80 jezikovnih različicah, ki so brezplačni za prenos. Sugar razvijalska cona je prostor za razvijalce SugarCRM. Na tem mestu imajo razvijalci povezavo na uradno razvijalsko literaturo SugarCRM, razvijalski forum, skupne vaje o prilagoditvah programske rešitve. Vključuje tudi dnevnik, ki ga ureja SugarCRM tim, ki zagotavlja razvijalcem vpogled v nove prihajajoče razvijalske funkcionalnosti. Večje kot je število sodelujočih v skupnosti, večja bo verodostojnost SugarCRM na tržišču in močnejše bodo konkurenčne sposobnosti. Živahna in trdna skupnost razvijalcev je zgodba skoraj vsakega uspešnega podjetja programskih rešitev. Več kot prispevaš, več dobiš nazaj.

Dohodkovni model. SugarCRM ustvarja dohodke iz produktov, kot sta Sugar Professional ter Sugar Enterprise, za katere uporabnik plačuje letno najemnino, ter z zagotavljanjem podpornih storitev. Dohodke komercialnih produktov dosežemo na tri načine: z on-line produkti, ki so na voljo kupcu v obliki gostovanja (Sugar On-Demand), s produkti na ključ, ki zagotavljajo tako strojne kot programske rešitve (Sugar Cube), ter s produkti, ki si jih enostavno prenesemo k sebi z dodatno prednostjo, da je izvorna koda napisana v PHP-obliki, ki je na voljo uporabniku (Sugar On-Permise). Proti plačilu se lahko uporabniki gibljejo med zgoraj omenjenimi možnostmi postavitve glede na njihove potrebe. Temeljno izhodišče za določitev cen proizvodov je uporabnik na leto. Uporabnik dejansko plačuje naročnino ali najem produkta SugarCRM. Dohodke od storitev prav tako prejmemo na tri načine:

- odpodpornih storitev: standardna podpora se plačuje z naročnino za odprtokodno različico produkta, ki je del paketa lastniške različice; razširjena podpora je na voljo za dodatne stroške;
- od storitev implementacije: ponudba nekaj deset implementacij je na voljo v enodnevnem paketu do polne namestitve in prilagoditev;
- od ter profesionalnih storitev: zajemajo 12 storitev, ki pokrivajo večino dogodkov, ki bi si jih želeli CRM uporabnik izvajati.

4.2.8 Bimeli d.o.o.

Bimeli d.o.o. je podjetje, specializirano za razvoj odprtokodnih spletnih rešitev, spletnih strani in spletne komunikacije. Nenehno išče nove in nove rešitve, kako splet izkoristiti za doseganje čim boljših poslovnih rezultatov. Gradi na prepričanju, da je za uspeh projekta potrebno dobro sodelovanje med izvajalcem in naročnikom. Stranka investira, z željo po izboljšanju svojega poslovanja, njen cilj je zaslužiti na dolgi rok. Da bi sodelovanje z

naročniki še izboljšali, transparentnost in organiziranost dela pa še povečali, so se pri podjetju Bimeli odločili za razvoj odprtokodnih spletnih rešitev, ki bi njihove storitve še lažje približale naročnikom.

Ponudba. Ponudba podjetja temelji na izdelavi odprtokodnih spletnih rešitev (spletne strani, spletne predstavitve, vtičniki ...). Programske rešitve so na voljo pod GPL licenco ali pod drugimi licencami takrat, ko se GPL možnosti ne da aplicirati za uporabo. Za doseganje čim večjega števila uporabnikov so v podjetju sprejeli sistem dvojnega licenciranja. Pod licenco GPL se distribuirajo brezplačne različice odprtokodnih spletnih rešitev, prodajna različica pa se distribuira pod komercialno licenco, kjer so spletne rešitve in tehnologija zaprtega tipa. Zagotavljajo spletne rešitve za ponudnike programskih rešitev, razvojne partnerje, podjetja, posameznike ter odprtokodne skupnosti. Ciljna trga sta slovenski in italijanski trg spletnih rešitev in komunikacij. Uporabnik pridobi odprtokodno spletno rešitev, ki jo lahko sam spreminja in uporablja, plačuje pa storitve, povezane s spletnimi rešitvami, kot so podpora, nadgradnje in izobraževanje. Spletne rešitve so prilagojene posameznim segmentom uporabnikov, ki se razlikujejo po željah in potrebah.

Viri. Ključni vir podjetja je vodja razvoja, ki je začel s programiranjem spletnih rešitev leta 2003, pred tem je bil zaposlen v podjetju Hit, kjer so razvijali spletne rešitve za on-line igralnico. Tudi njegov sodelavec, kasneje soustanovitelj podjetja, je sodeloval pri načrtovanju projektov v obdobju 2008 - 2011 in akumuliral znanja o spletnih rešitvah. Z licenciranjem spletnih odprtokodnih programskih rešitev z odprtokodno licenco je podjetje preneslo nekatere notranje intelektualne vire v odprtokodne skupnosti, s pomočjo katerih so pridobili možne stranke v vlogi razvijalcev in navdušencev za podporo njihove ponudbe. Tudi licenciranje njihovih spletnih rešitev z odprtokodno licenco je pripomoglo k dobremu sodelovanju z odprtokodno skupnostjo, saj so dobri odnosi zelo pomembni pri vodenju odprtokodnih projektov. Notranji programerji so ključni element v poslovnem modelu podjetja. Večino izvorne kode Bimeli spletnih rešitev prispevata ključna zgoraj omenjena vira. V podjetju stremijo k zaposlitvi večjega števila programerjev za povečanje njihovih notranjih programerskih virov. Poleg povečevanja notranjih programerskih virov podjetje sistematično investira v strokovno upravljanje človeških virov za ustvarjanje dobičkonosnega podjetja.

Odnosi. Sodelovanje temelji na osebnih odnosih med ključnimi posamezniki, kar lahko navedemo kot glavni dejavnik uspeha v začetnih fazah razvoja spletnih programskih rešitev Bimeli. Odprtokodno vzdušje in izmenjava znanja med soustanovitelji podjetja, spletnih rešitev in komunikacij so dobra osnova za širitev internetnega omrežja odprtokodnih skupnosti. Število odnosov se hitro množi z rastočimi internetnimi skupnostmi in z institucionalizacijo odprtokodnega gibanja. Partnerji podjetja Bimeli so zunanji izvajalci storitev, podjetja s področja odprtokodnih programskih rešitev, poslovno-raziskovalne ustanove in drugi strateški partnerji. Odnosi s temi podjetji temeljijo na dvo

ali večstranskih komercialnih dejavnostih. Poleg tega si v podjetju želijo sodelovati z javnimi organizacijami in raziskovalnimi inštituti, saj postaja poznavanje odprtokodnih spletnih rešitev vedno večje. Razmerja v odprtokodnih skupnostih so večplasten pojav. Skupnost uporabnikov pripomore k ustvarjanju odprtokodnih spletnih rešitev, usposabljanju in predstavitvam. Poleg tega razvijajo aplikacije za različne odprtokodne projekte. Podjetje veliko pridobiva od odprtokodnih skupnosti, a vendar celoten razvoj naredijo v podjetju sami. Podjetje je kakorkoli naredilo ogromen prispevek v odprtokodni skupnosti tako z licenciranjem spletnih rešitev, kot tudi z odprtokodnimi programskimi rešitvami.

Dohodkovni model. Podjetje Bimeli stremi k prodaji spletnih rešitev in s tem povezanih storitev, kot so podpora, izobraževanje in svetovanje. Projekti so odprtokodni, a usmerjeni v profit. Podjetje ponuja odprtokodne spletne rešitve pod licenco GPL brezplačno ter prodajo spletnih rešitev pod lastniško licenco za stranke tam, kjer GPL možnost ni idealna opcija (primer vključitev Adobe Flash tehnologije v razvoj zaprtega tipa). V podjetju prejmejo večino dohodkov iz naslova prodaje lastniških spletnih rešitev, manj iz drugih virov dohodka, kot so odprtokodne spletne storitve. Njihov glavni vir dohodka so tako komercialni uporabniki.

4.3 Raziskovalni model

S kvalitativnim raziskovanjem, ki nam omogoča pridobitev vpogleda v določene fenomene ali pojave, za katere je bilo opravljenih le malo raziskav in niso natančno znane pomembne spremenljivke (Denzin, 2000, str. 5), smo skušali ustvariti pregled različnih poslovnih modelov, ki temeljijo na odprtokodnih programskih rešitvah. Vprašalnik je sestavljen iz petih sklopov vprašanj o ponudbi, virih, odnosih in dohodkovnem modelu, s katerimi bomo sestavili strukturo vsakega posameznega poslovnega modela podjetja.

4.4 Raziskovalno vprašanje

Uresničevanje poslovnih ciljev z uporabo odprtokodnih programskih rešitev je zelo kompleksno. Z empiričnim raziskovanjem bomo odgovorili na raziskovalno vprašanje, kakšni poslovni modeli obstajajo v podjetjih, ki temeljijo na odprtokodnih programskih rešitvah.

4.5 Rezultati in analiza podatkov

Podjetja, vključena v študijo, razkrivajo več zornih kotov, kako lahko uporabijo odprtokodne programske rešitve. Kot že navedeno, literatura ne ponuja veliko opisov struktur in poslovnih modelov, ki bi bili primerni za analizo poslovnih modelov, ki temeljijo na odprtokodnih programskih rešitvah. Verjetno najbolj znan primer klasifikacije odprtokodnih poslovnih modelov je Heckersov (1999). A ta razvrstitev ni primerna za analizo primerov, saj v tej študiji ne upošteva nobenih drugih razen denarnih razlogov za sodelovanje v odprtokodnem gibanju. Po Heckerju (1999) so poslovni modeli osredotočeni predvsem na denarni tok med podjetjem in njegovim okoljem.

Štirje primeri se do določene mere poenotijo s Heckerjevimi modeli, to so izguba vodilnih, dodatna oprema in podpora prodajalcem, kjer sta cilj uporaba odprtokodnih programskih rešitev ter ustvarjanje dohodkov od prodaje izdelkov programskih opreme, nekaterih fizičnih izdelkov ali od prodaje podpore. Omenjena podjetja, kot so IBM, SugarSRC, Netgem in Bimeli, uporabljajo odprtokodne programske rešitve, z namenom razvoja končnih proizvodov, ki ustvarjajo njihove dohodke. Zato ti poslovni modeli veljajo za neke vrste kombinacijo med poslovnimi modeli: izguba vodilnih, dodatna oprema in podpora prodajalcem. Heckersovi (1999) poslovni modeli v večini primerov niso direktno aplikativni in se ne uporabljajo kot osnova za analizo. Tako smo analizo različnih poslovnih modelov opravili na podlagi sprejetih osnov odprtokodnega pristopa, ki opisuje, kako podjetja uporabljajo odprtokodne programske rešitve.

V študiji sta bila opredeljena dva pristopa: zunanji odprtokodni pristop in notranji odprtokodni pristop. V Tabeli 9 smo razvrstili primere poslovnih modelov na podlagi sprejetih odprtokodnih pristopov.

Tabela 9: Odprtokodni pristopi v podjetjih

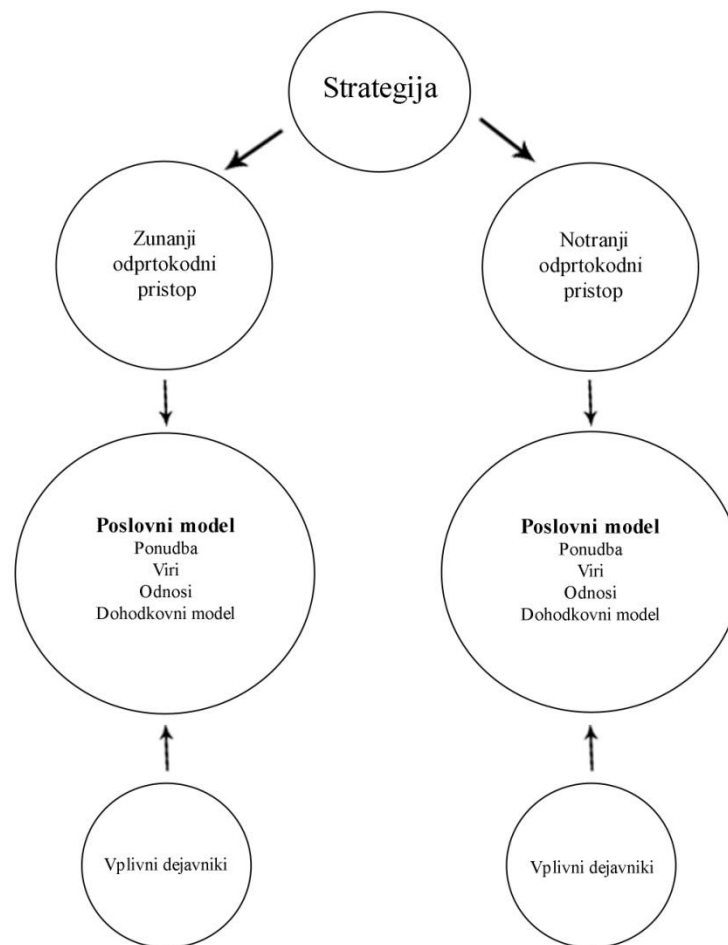
Študija	Odprtokodni pristop
Hit d.d.	Notranji pristop
Typo3	Zunanji pristop
Iskra Avtoelektrika d.d.	Notranji pristop
IBM	Notranji pristop
SugarCRM	Zunanji pristop
Netgem	Notranji pristop
Nokia Symbian	Zunanji pristop
Bimeli d.o.o.	Zunanji pristop

Glavne ugotovitve študije so, da značilnosti poslovnih modelov določamo z dvema dejavnikoma: z **odprtokodnim pristopom** in **vplivnimi dejavniki**. Ta dva dejavnika različno vplivata na poslovne modele. **Prvič**, na podlagi študije razkrijemo, da sprejetje odprtokodnega pristopa določa zelo visoko raven značilnosti poslovnega modela, saj

podjetja, ki uporabljajo enake pristope, imajo veliko podobnosti v svojih poslovnih modelih. Še posebej zunanji odprtokodni pristop močno vpliva na značilnosti poslovnih modelov. **Drugič**, na poslovne modele, ki se izvajajo v praksi, vplivajo določeni dejavniki, zaradi katerih so lahko poslovni modeli, ki temeljijo na zunanjem odprtokodnem pristopu ali notranjem odprtokodnem pristopu drugačni. Te dejavnike smo poimenovali **vplivni dejavniki**.

Poleg tega je sprejetje določenega odprtokodnega pristopa odvisno od strategije, ki opredeljuje dolgoročne cilje podjetja. Zato morajo podjetja izbrati tak odprtokodni pristop, ki podpira in je v skladu z njihovo strategijo. To pomeni, da se strategija odločanja in vplivov usmerja glede na to, kakšen odprtokodni pristop smo izbrali. Z drugimi besedami, s strategijo najprej opredelimo cilje podjetja in nato lahko izberemo odprtokodni pristop za doseg te ciljev. Seveda je možno tudi, da nobeden od odprtokodnih pristopov ni v skladu s strategijo podjetja, potem pa mora podjetje poiskati druge načine za doseganje svojih strateških ciljev. Glavne ugotovitve te študije so predstavljene na naslednji sliki, ki jo bomo podrobneje predstavili v nadaljevanju.

Slika 6: Grafični prikaz poslovnega modela, ki temelji na odprtokodnih programskih rešitvah



Podjetja, ki so sprejela **notranji odprtokodni pristop** za izkoriščanje odprtokodnih programskih rešitev so Hit, d.d., Iskra Avtoelektrika, IBM in Netgem. Iskra uporablja odprtokodne programske rešitve kot take in jim zato ni treba skrbeti glede licenc. Povsem drugače je v Netgemu, IBM-u in Hitu, kjer je vključevanje odprtokodnih sestavnih delov v tržne izdelke videti veliko bolj zapleteno, saj mora biti podjetje zelo previdno z licencami in mora poskrbeti, da je vključevanje odprtokodnih programskih rešitev primerno v vseh pogledih. Eden od načinov za preprečitev morebitnih težav je pravilna uporaba licenc ali posluževanje distributerjev, kot to počnejo v Netgemu za operacijski sistem Linux, kjer spreminjajo odprtokodne rešitve in zagotavljajo podporo. Na splošno se odprtokodne programske rešitve štejejo za cenejše kot lastniške programske rešitve, vendar njihova uporaba ni brezplačna. Še posebej, če upoštevamo podporo, vzdrževanje in stroške integracije.

Typo3, SugarCRM, Nokia Symbian in Bimeli uporabljajo **zunanji odprtokodni pristop**. Vsi uporabljajo zunanji odprtokodni pristop na nekoliko drugačen način. Glavni cilj podjetja Bimeli d.o.o. ni v tem, da bi pridobilo razvojne vire iz odprtokodnih skupnosti, kar pa ne velja za Typo3, SugarCRM in Nokia Symbian, ki poskušajo pridobiti zunanje vire

ravno iz njih. Typo3, SugarCRM, Nokia Symbian so izdali izvorno kodo in poskušajo privabiti nove razvijalce. Cilj podjetja Nokia Symbian je oblikovati aktivno razvojno skupnost in znižati stroške. Nokia Symbian je imela tako imenovan konzorcijski model, kjer so razvijalci podjetja. Pri podjetju Nokia Symbian se zavedajo, da gre za veliko tveganje v primeru, da bi katero od podjetij zapustilo projekt, in to je tudi eden od razlogov, zakaj si želijo večje baze razvijalcev. Nokia Symbian ni prav uspešna pri oblikovanju razvoja skupnosti. Bimeli d.o.o. je v začetku odprtokodnega življenjskega cikla in načrtuje izdati javnosti določene spletne rešitve kot odprtokodne. Njihov pristop se razlikuje od pristopa podjetja Nokia Symbian, saj načrtujejo javnosti izdati le določene rešitve in določene ohraniti zaprte. Pri Typo3 in SugarCRM se zavedajo velikosti njihovih skupnosti, zato je njihov cilj še dodatno motivirati skupnost, da bo še več prispevala in vedno izboljševala svoje rešitve.

Pri poslovnih modelih, ki temeljijo na **zunanjem odprtokodnem pristopu**, je **ponudba** odvisna od ciljev podjetij. Ponudba programskih rešitev je lahko objavljena pod odprtokodno licenco in sproščena v javnost v celoti ali delno, odvisno od tega, koliko nadzora želi podjetje obdržati. Če podjetje oceni, da nekateri deli programskih rešitev zagotavljajo konkurenčno prednost, potem izdajo izvorno kodo v javnost le delno. Typo3 in Nokia Symbian sta sprostili izvorni kodi svojih programskih rešitev v celoti, med tem ko so pri SugarCRM in Bimeli d.o.o. te sprostili le delno. Če je izvorna koda v celoti objavljena, verjetno ne ustvarja neposrednih denarnih tokov. Včasih podjetja nimajo nobenih denarnih ciljev. Na primer Nokia Symbian želi z uporabo zunanjega odprtokodnega pristopa povečati članstvo skupnosti. Na ta način želi Nokia Symbian, da bi lahko spodbudili skupnosti in pridobili razvojne vire ter končno dosegli prihranke pri stroških. Aktualna ponudba v zunanjem odprtokodnem pristopu ni programska rešitev sama. V mnogih primerih podjetja izdajo izvorno kodo in oblikujejo njihovo ponudbo v sklopu programskih rešitev. Tako ponudbo storitev svetovanja ponujajo v SugarCRM, Typo3 in Nokia Symbian. V primeru Typo3, SugarCRM in Nokia Symbian so sprostili izvorno kodo programskih rešitev, da bi pridobili dodatne vire za razvoj iz skupnosti. Pri SugarCRM in Typo3 menijo, da so z zunanjim odprtokodnim pristopom ustvarili programske rešitve, povečali uporabnost produktov ter pridobili na prepoznavnosti blagovnih znamk.

Eden od glavnih **virov** zunanjega odprtokodnega pristopa je običajno odprtokodna skupnost. Pogosto podjetja upajo, da bodo pridobila razvojne vire iz odprtokodnih skupnosti, ko le-ta izdajo izvorno kodo programskih rešitev v javnost. Včasih podjetja nimajo dovolj virov za razvoj programskih rešitev in upajo na skupnost, da bi pomagala pri tej nalogi. Vendar pa uspešno izvajanje zunanjega odprtokodnega pristopa ni enostavno. Ko so pri Nokii sprostili izvorno kodo operacijskega sistema Symbian, so si želeli, da jim bo skupnost lahko pomagala pri razvoju, vendar so ugotovili, da pritegniti posamezne razvijalce ni tako enostavna naloga. Da bi pritegnili razvijalce in razvili aktivno razvojno

skupnost, bi morala podjetja upoštevati značilnosti odprtokodnega gibanja in različne motivacijske faktorje članov skupnosti. Razvojno skupnost lahko sestavljajo podjetja, posamezni razvijalci ali pa kombinacija teh. V primeru Nokie več podjetij skupaj razvija operacijski sistem Symbian. Poleg podjetij bi Nokia želela pridobiti pomoč posameznih razvijalcev pri razvoju. Poleg razvoja lahko pomagajo izboljšati in testirati operacijski sistem. Typo3 in SugarCRM imata aktivno skupnost uporabnikov, ki poroča o napakah in na ta način prispeva k izboljšanju programskih rešitev. Običajno podjetja izdajo izvorno kodo in predstavljajo glavni del razvoja, prispevajo v razvoj in ga usmerjajo. To velja tudi za podjetji Typo3 in SugarCRM. Tako so notranji viri pogosto najpomembnejši del zunanjega odprtokodnega pristopa.

Za zunanji odprtokodni pristop so najpomembnejši **odnosi** med podjetjem in skupnostjo. Če skupnost sestavljajo tudi druga podjetja, to pomeni, da so si ta precej blizu glede odnosov in sodelovanja. To je posledica podobnih ciljev in motivacij podjetij. Položaj je nekoliko drugačen, če je skupnost sestavljena samo iz posameznikov. Zdi se, da se lahko podjetja soočajo z velikimi izzivi pri oblikovanju razvoja skupnosti. Nokia za operacijski sistem Symbian ni pridobila velikega števila prispevkov skupnosti, medtem ko se pri Typo3 in SugarCRM postavlja vprašanje, če bi lahko sploh obstajala brez skupnosti. Podjetjema, kot sta SugarCRM in Typo3, je uspelo pri razvoju skupnosti, a se kljub temu zavedata, da se sodelovanje s skupnostjo močno razlikuje od tradicionalnega sodelovanja. Člani skupnosti imajo lahko različne cilje kot podjetja, motivirajo jih lahko različni dejavniki in lahko si želijo razvoja v drugo smer kot podjetja. Poleg tega lahko posamezni razvijalci svobodno izbirajo, ali želijo sodelovati ali ne, saj imajo možnost zapustiti projekt v vsakem trenutku. To je razlog, zakaj podjetja poskušajo poiskati ustrezne načine za podporo, motivacijo in usmerjanje skupnosti. Pomembno je tudi povedati, da veliko odprtokodnih razvijalcev ni naklonjenih podjetjem, ki izkoriščajo njihove proizvode. Zato morajo podjetja vzpostaviti dober odnos s skupnostmi in na ta način poskušati vplivati nanje. V primerih Typo3 in SugarCRM je veliko komunikacije med podjetjema in skupnostmi, kar se odraža v velikem interesu za sodelovanje pri razvoju in podpori. Nokia Symbian je skušala biti prav tako odprta za skupnost, da bi pritegnila razvijalce, a je bil odziv precej majhen.

V obravnavani literaturi je glavni cilj poslovnih modelov, ki temeljijo na odprtokodnih programskih rešitvah, ustvariti **dohodke** z odprtokodnimi programskimi rešitvami (Hecker, 1999). Edini primer v študiji, ki je v skladu s to obliko modela, je operacijski sistem Nokia Symbian. Nokia je izdala izvorno kodo operacijskega sistema Symbian v javnost, da bi si na ta način povečala prodajo strojne opreme. Zanimiva lastnost drugih primerov zunanjih odprtokodnih pristopov pa je ta, da njihov glavni cilj ni pridobivanje neposrednih denarnih sredstev z odprtokodnimi programskimi rešitvami, ampak zmanjšanje stroškov in povečanje posrednih dohodkov z uporabo odprtokodnega gibanja. Nokia za Symbian želi, da bo razvoj skupnosti lahko pomagal pri razvoju programske opreme in na ta način

zmanjšal Nokii delovne obremenitve in tudi izboljšal proizvode, tako da se bodo stroški servisa in podpore zmanjšali. Poleg tega menijo, da zunanji odprtokodni pristop lahko spodbudi pozitivne odzive javnosti na operacijski sistem Symbian. Za Typo3, SugarCRM in Bimeli pa ni glavni cilj ustvariti neposredne dohodka z odprtokodnimi programskimi rešitvami, ampak ponuditi posredne storitve, kot so svetovanje, podpora, usmerjanje skupnosti, izobraževanje, da bi podjetja v čim večji meri povečala koristi od odprtokodnih programskih rešitev in posledično zmanjšala tudi stroške.

Pri **notranjem odprtokodnem pristopu ponudbo** razdelimo na dva dela: ponudbo za zunanji trg ter ponudbo za interno uporabo v podjetju. Netgem in IBM uporabljata odprtokodne programske rešitve v svoji strojni opremi, ki jo prodajajo zunanjemu trgu kot končni izdelek. V obeh primerih podjetji uporabita notranji odprtokodni pristop za ponudbo končnih izdelkov zunanjemu trgu. Interna uporaba odprtokodnih programskih rešitev pa se uporablja v podjetjih Hit d.d. in Iskri. V Hitu so se odločili za kombinacijo odprtokodnih programskih rešitev Linux, Apache, Mysql, php in Typo3. V podjetju Iskra so se odločili za uporabo odprtokodnih programskih rešitev, in sicer Linux, Apache, Javascript in xml za delovanje njihovega intranet portala.

Pri notranjem odprtokodnem pristop glavne **vire** predstavljajo zaposleni v podjetjih in njihovo znanje. Iskra je poleg notranjih virov dovolila dostop do intraneta tudi nekaterim zunanjim partnerjem. Prav tako je Hit dovolil dostop do urejevalnika vsebin svojih internetnih strani, vendar gre v obeh primerih bolj za lažje sodelovanje s partnerji oziroma hčerinskimi družbami kot pa za posebno notranje sodelovanje pri razvoju. Za notranji odprtokodni pristop imajo razvite programske komponente pomembno vlogo, saj sta s tem IBM in Netgem izboljšala ponovno uporabo programske opreme. Zato lahko te programske komponente štejemo kot notranje vire.

Racionalizacija razvoja programske opreme in ponovna uporaba obstoječih sredstev so bili eden od glavnih dejavnikov za sprejetje notranjega odprtokodnega pristopa v Hitu in Iskri. Z notranjim odprtokodnim pristop skušajo Hit, Netgem, IBM in Iskra izkoriščati obstoječe vire na najbolj učinkovit način.

Za notranji odprtokodni pristop so potrebni različni **odnosi** med zaposlenimi, poslovnimi enotami in projekti. Del komunikacije je formalen in poteka na različnih sestankih ter okroglih mizah. Poleg tega obstajajo številni neuradni komunikacijski mehanizmi, kot so e-poštni sezname, Skype, spletni forumi in neformalni pogovori iz oči v oči. Vendar pa je bilo nekaj razlik med IBM-om in Netgemom. V podjetju Iskra ne želijo, da bi uporaba intraneta bila obvezna za vse poslovne enote, zato se je lahko vsaka poslovna enota odločila, ali ga bo uporabljala ali ne. Z drugimi besedami, poslovne enote se lahko same odločijo, ali želijo uporabiti obstoječo programsko opremo in sodelovati z drugimi enotami. Iskra niti ni poskušala spodbuditi skupnosti, da bi ohranila intranet v največji

možni meri, v odprtokodnem okolju. Tveganje pri tem pristopu od spodaj navzgor je v tem, da se člani skupnosti obrnejo v pasivne uporabnike, namesto da bi bili aktivni razvijalci. Netgem pa se je odločil za pristop od zgoraj navzdol, da razvoj programske opreme deluje kot celota v skladu s svojim novim notranjim odprtokodnim pristopom. Zato je sodelovanje med poslovnimi enotami zagotovljeno in Netgem usmerja s pomočjo sodelovanja v oblaku skupnost z opredelitvijo delovnih postopkov.

Dohodkovni model pri notranjem odprtokodnem pristopu lahko razdelimo na zunanje in notranje modele dohodkov. Zunanja dohodkovna modela, kot sta IBM in Netgem, se nanašata na stroške in dohodke celotnega podjetja, medtem ko se notranji model dohodkov nanaša predvsem na delitev stroškov v podjetju. Cilj pri zunanjem dohodkovnem modelu je zmanjševanje stroškov s pomočjo učinkovitejše razvojne prakse in na ta način povečevanje dobička. Tako IBM kot Netgem sta racionalizirala razvoj programskih rešitev, izboljšala sodelovanje med poslovnimi enotami in povečala ponovno uporabo obstoječih sredstev. Iskra in Hit se soočata z nekaterimi izzivi notranjega dohodkovnega modela. Notranji odprtokodni pristop zahteva, da poslovne enote in projekti sodelujejo med seboj in da so programske rešitve za ponovno uporabo. Vendar pa imajo vse poslovne enote in projekti svoje proračune in načrte, ki omejujejo njihovo pripravljenost za sodelovanje, saj bi za dele za ponovno uporabo potrebovali več sredstev. To je razlog, zakaj podjetja uporabljajo notranji odprtokodni pristop, s katerim morajo ustvariti politiko, ki motivira člane poslovnih enot in projektnih skupin, da si prizadevajo za najboljše celotnega podjetja, in ne le za optimizacijo svojega dobička.

Podjetja imajo tako pri notranji kot pri zunanji uporabi odprtokodnega pristopa različne cilje. Nekatera podjetja sprostijo izvorno kodo v celoti, druga pa le delno. To pomeni, da obstajajo razlike v poslovnih modelih, čeprav se uporablja enak pristop. Te dejavnike bomo imenovali »**vplivni dejavniki**«, ker imajo nižji vpliv na značilnosti poslovnih modelov. Na podlagi študije primerov se zdi, da imata tako zunanji kot notranji odprtokodni pristop svoje značilne dejavnike, ki morebiti vplivajo na poslovne modele. Nekateri dejavniki so prav tako enaki.

Pri **zunanjem odprtokodnem pristopu** med vplivne dejavnike štejemo **cilje, strateško naravnost programskih rešitev, nadzor, dohodke in stroške**. Cilji podjetja imajo velik vpliv na poslovne modele in na to, kako se zunanji odprtokodni pristop uporablja. Z uporabo zunanjega odprtokodnega pristopa podjetja poskušajo pridobiti razvojne vire iz skupnosti ali pa promovirati standardizacijo. **Strateška naravnost programskih rešitev** pomeni, v kolikšni meri se sprostijo izvorna koda programskih rešitev. Če se programska rešitev obravnava kot vir konkurenčne prednosti, je v želji podjetja, da je sprostitev izvorne kode le delna. Tako lahko podjetja sprostijo izvorno kodo določenih sestavnih delov in hranijo diferencirane komponente, ki jih pustijo zaprte. Enake argumente je mogoče uporabiti tudi za vprašanja glede **tveganja**. Če se podjetje boji, da mu bo nekdo ukradel

izvorno kodo in ponudil konkurenčno ponudbo, se lahko odloči, da izda le nekatere dele programske opreme in diferencirane dele ohrani zaprte. Dohodkovna logika podjetja prav tako vpliva na poslovni model. Če je glavni razlog za sprejetje zunanjega odprtokodnega pristopa ustvarjanje neposrednih **dohodkov**, potem so programske rešitve najverjetneje videti kot način za prodajo produktov ali storitev. Kot je pokazala ta študija, pa podjetja ne ustvarjajo vedno neposrednih dohodkov. Podjetja lahko na primer poskušajo doseči dohodke pri stroških s pomočjo porazdeljenega razvoja.

Pri **notranjem odprtokodnem pristopu** vplivni dejavniki vključujejo **cilje, izbrane odprtokodne prakse, nadzor, stroške in delitev stroškov**. Cilji podjetja imajo velik vpliv na to, kako se uporablja notranji odprtokodni pristop. Cilji so lahko na primer racionalizacija razvoja programske opreme, ustvarjanje družine proizvodov ali ponovna uporaba programske opreme. **Izbrane odprtokodne prakse** nato odločijo, kako se odprtokodni pojav uporablja v podjetju. Podjetja lahko posnemajo odprtokodni razvojni proces, hierarhično strukturo razvoja skupnosti ali brezplačno delitev programske opreme. Tako je tudi z **nadzorom**; če podjetja želijo zagotoviti, da je notranji odprtokodni pristop podobno sprejet v celotnem podjetju, je treba, da odločitve potekajo od zgoraj navzdol in vnaprej določajo delovne prakse. V večini primerov notranji odprtokodni pristop zahteva, da poslovne enote sodelujejo druga z drugo, saj skupne delovne prakse povzročajo stroške, ki lahko zmanjšajo pripravljenost poslovne enote k sodelovanju. Zato so podjetja razvila različne načine za pravično delitev stroškov in nagrajevanje za prispevke. Te različne prakse so nato vplivale na poslovni model.

5 SKLEP

Odprtokodne programske rešitve v poslovnih modelih podjetij prinašajo nove izzive v podjetja, ki se odločijo za njihovo uporabo.

Odprtokodne programske rešitve, ki bodo delovale v interesu vseh udeležencev poslovanja, bodo tiste prave. Postati bodo morale še prijaznejše, bolj zaželenne s strani uporabnikov, uživati bodo morale njihovo zaupanje. S pomočjo odprtokodnih programskih rešitev bo poslovanje podjetij obstajalo v ravno pravem razmerju za vsako podjetje posebej, saj bo vsako podjetje samo odločalo, kateri odprtokodni pristop je najprimernejši za njihovo uporabo.

Z večjim številom raziskav o odprtokodnih programskih rešitvah v poslovnih modelih bodo iz anonimnih prešle v znane odprtokodne strukture poslovnih modelov s specifičnimi željami posameznega podjetja, strukture poslovnih modelov pa bodo postale zadovoljevalec njihovih specifičnih želja. Razvoj se bo iz podjetja, ki je trenutno (še) najpomembnejše in v večji meri določa razvojne aktivnosti, preneslo na skupnost. Ta bo prevzela aktivnejšo vlogo in podjetja ji bodo morala slediti, sicer bo izbrala drugega

ponudnika. Zato vsak odprtokodni pristop z namenom uporabe v poslovnem modelu podjetja teži k drugačnemu, posebnemu pristopu. Odprtokodne programske rešitve v poslovnih modelih podjetij so razpete med naraščajočo specializacijo in individualizacijo podjetja ter ekonomičnostjo, ki zahteva univerzalnost, množičnost in poenotenje programskih rešitev.

Za odprtokodne programske rešitve je značilno, da je programska koda odprta in da uporabnik nadzira razvoj oziroma vpliva na njegov potek, sama skupnost pa določa smer in tempo razvoja. Vnaprej določena časovna razporeditev, kot v primeru komercialnih programskih rešitev, ni potrebna. Odprtokodne programske rešitve omogočajo interakcijo med podjetjem in skupnostjo. Tako je uporabnik odprtokodnih programskih rešitev dobil možnost biti aktiven soudeleženec množičnega razvojnega procesa in ni več razosebljen del ciljnega občinstva. Razvoj s pomočjo odprtokodnih skupnosti ima v nekaterih primerih mnogo značilnosti razvoja v podjetju.

Za zaposlene bo zelo pomembno razumevanje novih načinov uporabe odprtokodnih programskih rešitev v poslovnih modelih, ki uporabnikom omogočajo izbiro zelenih odprtokodnih programskih rešitev. Prav razumevanje nove spremenjene vloge odprtokodnih programskih rešitev v poslovnih modelih in s tem tudi strategije podjetja je temeljni pogoj, ki lahko v podjetju ustvari razmere, ki omogočajo uresničevanje celostnega poslovanja.

Rezultatov študije ne moremo preprosto posplošiti na vse poslovne modele podjetij, lahko pa jih upoštevamo, ko sprejemajo poslovni model, ki temelji na odprtokodnih programskih rešitvah, saj odprtokodne programske rešitve ne ponujajo enostavnih možnosti apliciranja v poslovne modele. Podjetja morajo skrbno premisliti, kateri pristop je primeren zanje glede na trg, trenutne razmere na trgu in glede na strategijo podjetja, če želijo ustvariti ustrezne dohodke ali prihranke stroškov z uporabo odprtokodnih programskih rešitev.

Notranji odprtokodni pristop je najbolj pogosto uporabljen odprtokodni pristop. Na podlagi tudi drugih študij je razvidno, da veliko podjetij interno uporablja kakšno odprtokodno rešitev. Če se odprtokodna programska rešitev uporablja kot taka, ni nobenih težav z licencami. V kolikor pa podjetje želi spreminjati sestavne dele ali jih uporabiti kot del pri njihovih komercialnih programskih rešitvah, mora biti previdno z licencami. Poleg tega je treba pri sprejemanju notranjega odprtokodnega pristopa podjetja opozoriti na zaželeno udeležbo v skupnosti in natančno izračunati stroške vključevanja takega pristopa. Že res, da stroškov za pridobitev programskih rešitev ni, moramo pa upoštevati vzdrževanje, podporo in spremembe, ki v večini premerov povzročajo določene stroške.

Pri uporabi zunanjega odprtokodnega pristopa morajo biti podjetja zelo predvidna, da ne izgubijo konkurenčne prednosti z izdajo izvirne kode javnosti. Določene programske

rešitve niti nimajo odprtokodnega potenciala. Če posamezni razvijalci menijo, da so programske rešitve nezanimive ali da podjetje zgolj izkorišča njihove prispevke, bodo po vsej verjetnosti prenehali prispevati. Če se podjetje odloči, da bo sprostilo programske rešitve kot odprtokodne, potem se mora odločiti, ali bo izvirno kodo sprostilo v celoti ali delno in če ta vključuje tudi razvojno arhitekturo programskih rešitev, tako da omogoči zunanjim, da lahko prispevajo. S sprostitvijo izvirne kode mora podjetje oblikovati strategijo o tem, kako bo pritegnilo razvijalce, jih motiviralo k sodelovanju in usmerjalo tako, da bo cilj družbe dosežen.

Poslovni modeli bodo uporabljali različne možnosti, ki jih bodo ponujale odprtokodne programske rešitve, vendar bodo te različne možnosti usklajene in bodo skušale podjetjem ponuditi tisto, kar bodo želela in pričakovala. Menimo, da bodo meje med posameznimi elementi poslovnega modela bolj zabrisane kot doslej. Razvoj bo moral biti integriran in iz solističnega pristopa, ki je trenutno običajna praksa, bo prešel v skupinski. Pobuda ali vsaj privolitev na začetno pobudo iz podjetja za razvoj bo morala priti s strani porabnikov ali drugih deležnikov.

Podjetja odprtokodne programske rešitve uporabljajo, vendar pa ne izkoriščajo vseh možnosti, ki jih te rešitve že sedaj ponujajo. Zanimivo pa je, da pri tem vseeno velika večina podjetij meni, da uporabniki vse več uporabljajo odprtokodne programske rešitve za svoje lastne potrebe oziroma v prostem času. Zato pričakujemo, da se bodo v prihodnje v podjetjih v večji meri začeli zavedati in uporabljati možnosti, ki jih prinašajo odprtokodne programske rešitve. Hkrati pa se bodo svojih večjih možnosti začeli zavedati tudi uporabniki, ki bodo pridobili vlogo podjetjem enakovrednega soudeleženca razvojnih procesov.

6 LITERATURA IN VIRI

1. Afuah, A., & Tucci, C. L. (2003). *Internet Business Models and Strategies*. Boston: McGraw Hill.
2. Alt, R. & Zimmerman, H. D. (2001). Introduction to Special Section – Business Models. *Electronic Markets - The International Journal*, 11(1), 3–9.
3. Amit, R., & Zott, C. (2001). Value Creation in E-business. *Strategic Management Journal*, 22(6), 493–520.
4. Ašanin, G. (2000). Od celibata in dominacije do integriranih organizacijskih komunikacij. *Akademija MM*, 4(6), 21–28.
5. Bonaccorsi, A. & Rossi, C. (2003). Why Open Source can succeed. *Research Policy*, 32(7), 1243–1258.
6. Bonaccorsi, A. & Rossi, C. (2006). *Comparing Motivations of Individual Programmers and Firms to Take Part in the Open Source Movement: From Community to Business*. Pisa: Laboratory of Economics and Management.
7. Bossidy, L., Charan, R., & Burck, C. (2002). *The discipline of getting things done*. New York: Random House.
8. Business model. (2010). V *Wikipedia*. Najdeno 28. maja 2010 na spletnem naslovu http://en.wikipedia.org/wiki/Business_model
9. Chesbrough, H., & Rosenbloom, R. S. (2002). The role of the business model in capturing value from innovation: evidence from Xerox Corporation's technology spin-off companies. *Industrial and Corporate Change*, 11(3), 529–555.
10. Daffara, C. (2007). Business models in FLOS-based software companies. *MIT*. Najdeno 2. februarja 2010 na spletnem naslovu <http://opensource.mit.edu/papers/OSSEMP07-daffara.pdf>
11. Denzin, N. K., & Lincoln, Y. S. (2000). *Handbook of Qualitative Research* (2nd ed.). Thousand Oaks, CA: Sage Publications.
12. *Dobičkonosna odprta koda*. Najdeno 2. julij 2008 na spletnem naslovu http://old_www.lugos.si/arhiv/prispevki/dok/
13. Egyedi, T.M., & van Wendel de Jode, R. (2006). Standardization and Other Coordination Mechanisms in Open Source Software. *Information Technology Standards and Standardization Research*, 1(1), 71–91.
14. Feller, J., & Fitzgerald, B. (2002). *Understanding Open Source Software Development*. London: Addison - Wesley.
15. Fink, M., (2003). *Business and Economics of Linux and Open Source*. New Jersey: Prentice Hall.
16. Fitzgerald, B. (2006). The Transformation of Open Source Software. *MIS Quarterly*, 30(3), 587–598.

17. FLOSSCOM.net. (2008). *Education transfer report*. Najdeno 21. avgusta 2010 na spletnem naslovu <http://openedworld.flossproject.org/index.php/flosscom-project-2006-to-2008>
18. Gallivan, J. M. (2001). Striking a balance between trust and control in virtual organization. *Information Systems Journal*, 11(4), 277–304.
19. Ghosh, R. A. (1998). What Motivates Free Software Developers: Interview with Linus Torvalds. *Firstmonday*. Najdeno 23. junija 2008 na spletnem naslovu http://www.firstmonday.org/issues/issue3_3/torvalds/index.html.
20. Ghosh, R. A. (2005). Understanding free software developers: Findings from the FLOSS study. V *Perspectives on Free and Open Source Software* (str. 23–46). Boston: MIT Press.
21. Ghosh, R. A., & Glott, R. (2005). *Free/Libre and Open Source Software: Policy Support. Results and policy paper from survey of Government authorities*. Maastricht: University of Maastricht.
22. Glyn M. (2008). Why falling Flash prices threaten Microsoft. *The guardian*. Najdeno 27. december 2009 na spletnem naslovu <http://www.guardian.co.uk/technology/2008/mar/06/opensource.olpc>
23. *GNU projekt*. Najdeno 3. maja 2008 na spletnem naslovu <http://www.gnu.org/home.html>
24. *Google Grabs No. 1 Brand Rank*. (2008). Najdeno 6. avgusta 2008 na spletnem naslovu http://googlewatch.eweek.com/content/google_financials/google_grabs_no_1_brand_rank.html
25. *Google Summer of code*. (2009). Najdeno 3. maja 2009 na spletnem naslovu <http://code.google.com/intl/sl/soc/>
26. *Google Web APIs*. (2008). Najdeno 6. junij 2008 na spletnem naslovu <http://code.google.com>
27. Gordijn, J., Akkermans H. & Vliet, H. (2000). Buisness Modelling is not Process Modelling. V *Conceptual Modeling for E-Business and the Web* (str. 40–51). Salt lake City: Springer-Verlag.
28. Hamel, G. (2000). *Leading the revolution*. Boston: Harvard Business School Press.
29. Hecker, F. (1999). Setting up shop: the business of Open-Source software. *Hecker.org*. Najdeno 19. maja 2008 na spletnem naslovu <http://www.hecker.org/writings/setting-up-shop.html>
30. Hedman, J., & Kalling, T. (2003). The business model concept: theoretical underpinnings and empirical illustrations. *European Journal of Information Systems*, 12(1), 49–59.
31. Hertel, G., Niedner, S., & Herrmann S. (2003). Motivation of software developers in Open Source projects: an Internet-based survey of contributors to the Linux kernel. *Research Policy*, 32(7), 1159–1177.

32. Hill, C., & Jones, G. (2008). *Strategic management theory*. New York: Houghton Mifflin Company.
33. Hitt, A. M., Ireland, R. D., & Hoskinsson, E. R. (2001). *Strategic Management*. South-Western Cincinnati. OH: College Publishing Inc.
34. Hoepman, J. H., & Jacobs, B. (2007). Increased Security Through Open Source. *Communications of the Association for Information Systems*, 50(1), 79–83.
35. Iannacci, F. (2005). *Coordination Processes in Open Source Software Development: The Linux Case Study*. London: Department of Information Systems.
36. *IBM in European Union Launch Joint Research Initiative for Cloud Computing*. Najdeno 9. avgusta 2008 na spletnem naslovu <http://www-03.ibm.com/press/us/en/pressrelease/23448.wss>
37. *Kaj je odprtokodna programska oprema?* Najdeno 5. maja 2008 na spletnem naslovu http://www.coks.si/index.php5/Vse_o_Odprti_kodi
38. Kardoff, E. (2004). *A Companion to Qualitative Research*. London: Sage Publications.
39. Keizer, G. (2008). Firefox on track to crack 20% share in July. *Computerworld*. Najdeno 21. avgusta 2010 na spletnem naslovu http://www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=9091959&intsrc=hm_list
40. Kindström, D. (2005). The Integration of E-business into Mature and Established Companies. *Liu.diva-portal.org*. Najdeno 25. decembra 2009 na spletnem naslovu <http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-31178>
41. Koenig, J. (2004). *Open Source Business Strategies*. In: Riseforth, Inc.
42. Kotler, P., & Armstrong, G. (1996). *Principles of marketing*. Englewood Cliffs, NJ: Prentice-Hall.
43. Lakhani, K., & von Hippel, E. (2003). How Open Source Software Works: Free User-to-User Assistance. *Forthcoming in Research Policy*, 32(6), 923–943.
44. Lerner, J., & Tirole, J. (2000). The Simple Economics of Open Source Software. *Nber.org*. Najdeno 19. maj 2010 na spletnem naslovu <http://www.webcitation.org/5SxWmySWj>
45. Lerner, J., & Tirole, J. (2001). The Open Source movement: key research questions. *European Economic Review*, 45(4), 819–826.
46. Lerner J. & Tirole J. (2002). Some Simple Economics of Open Source. *Journal of Industrial Economics*, 50(2), 197–234.
47. Madanmohan, T. R., & De' Rahul. (2004). *Open Source Reuse in Commercial Firms*. *IEEE Software*, 21(6), 62–69.
48. Magretta, J. (2002). *What management is: How it works and why it's everyone's business*. New York: The Free Press.
49. Mahadevan, B. (2000). Business Models for Internet-based e-Commerce: An anatomy. *California Management Review*, 42(4), 0–32.

50. Markus, M. L., Manville, B., & Agres. (2000). What Makes a Virtual Organisation Work?. *Sloan Management Review*, 42(1), 13–26.
51. Mason, J. (2002). *Qualitative researching*. California: Sage Publications.
52. Morris, M. S. M., & Allen J. (2005). The entrepreneur's business model: toward a unified perspective. *Journal of Business Research*, 58(6), 726–735.
53. Netcraft. (2008). *Web Server Survey*. Najdeno 15. maja 2010 na spletnem naslovu http://news.netcraft.com/archives/2008/01/28/january_2008_web_server_survey.html
54. *New to Open source IBM*. Najdeno 10. oktobra 2008 na spletnem naslovu <http://www.ibm.com/developerworks/opensource>
55. Normann, R., & Ramirez, R. (1995). *Designing Interactive Strategy. From Value Chain to value Constellation*. New York: John Wiley&Sons.
56. *Open Source Initiative*. (2010). Najdeno 21. avgusta 2010 na spletnem naslovu <http://www.opensource.org/>
57. Osterwalder, A. (2004). *The Business Model Ontology – A Proposition In a Design Science Approach*. Lausanne: Universite de Lausanne.
58. Paul, R. (2008). All the rage in Europe: Firefox market share climbs higher. *Arstechnica.com*. Najdeno 15. maja 2010 na spletnem naslovu <http://arstechnica.com/news/ars/post/20080427-all-the-rage-in-europe-firefox-marketshare-climbs-higher.html>
59. Petersen, K., Wohlin, C., & Baca, D. (2009). The Waterfall Model in Large-Scale Development. *V Product-Focused Software Process Improvement (str. 386–400)*. Berlin: Springer-Verlag.
60. Petrovic, O., Kittl, C., & Teksten, R. D. (2001). Developing Business Models for eBusiness. *Citeseer*. Najdeno 19. april 2010 na spletnem naslovu <http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.24.9466&rank=1>
61. Porter, M. (2001). Strategy and the Internet. *Harward BusinessReview*, 79(3), 63–78.
62. Rappa, M. (2004). Business models on the Web. *Digitalenterprise.org*. Najdeno 18. avgusta 2007 na spletnem naslovu <http://digitalenterprise.org/models/models.html>
63. Raymond, E. S. (1998). Goodbye, »free software«; hello »open source«. *Eric S. Raymond's Home Page*. Najdeno 10. maja 2008 na spletnem naslovu <http://www.catb.org/esr/open-source.html>
64. Raymond, E. S. (2001). *The cathedral & the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*. Sebastopol, CA: O'Reilly & Associates.
65. Rotz, V. B. (2007). Realize the Benefits of Open Source. *LWN*. Najdeno 15. junija 2008 na spletnem naslovu <http://lwn.net/images/pdf/optaros-catalog-2007.pdf>
66. Saunders, M., Lewis, P., & Thornhill, A. (1997). *Research Methods for Business Students* (5th ed.). London: Pitman.
67. Shafer, S. M., Smith, H. J., & Linder, J. C. (2005). The power of business models. *Business horizons*, 48(3), 199–207.

68. *Sourcefroge.net*. (2010). Najdeno 12. april 2010 na spletnem naslovu <http://Sourcefroge.net>
69. Stallman, R. (1983). The GNU Manifesto. *GNU*. Najdeno 15. julija 2010 na spletnem naslovu <http://www.gnu.org/gnu/initial-announcement.html>
70. Stallman, R. M. (2002). *Free Software Free Society*. Boston: GNU Press.
71. Stallman, R.M. (2008). V *Wikipedia*. Najdeno 15. maja 2008 na spletnem naslovu http://sl.wikipedia.org/wiki/Richard_Matthew_Stallman
72. Tapscott, D. (2007). Smart Enterprises. *Smartenterpriseexchange.com*. Najdeno 18. junij 2008 na spletnem naslovu <http://www.smartenterprisemag.com/articles/2007spring/perspectives.jhtml>
73. Tapscott, D., & Williams, A. D. (2006). *WIKINOMICS How Mass Collaboration Changes Everything*. New York: First published by Portfolio.
74. Ticehurst, W. G. (2000). *Business research methods – a managerial approach*. Frenchs Forest, NSW: Longman.
75. Timmers, P. (1998). Business Models for Electronic Markets. *Journal on Electronic Markets*, 8(2), 3–8.
76. *Tutorials and projects in open source IBM*. Najdeno 7. avgusta 2008 na spletnem naslovu <http://www.ibm.com/developerworks/opensource>
77. Volovšek, Š. (2001). Application Service Providers: spremenjeni pogoji na tržišču programske opreme. *Drustvo-informatika.si*. Najdeno 10. junija 2008 na spletnem naslovu <http://www.drustvo-informatika.si>, 10.06.2008.
78. Von Hippel, E., & von Krogh, G. (2003). Open source software and the Private-Collective Innovation Model: Issues for Organization Science. *Organization Science*, 14(2), 209–223.
79. Von Krogh, G., Sebastian S., & Karim, R. L. (2003). *Community, Joining, and Specialization in Open Source Software Innovation: A Case Study*.
80. Walli, S., Gynn, D., & Rotz, V. B. (2005). The growth of Open Source Software in Organisations. *Novell.com*. Najdeno 23. julija 2010 na spletnem naslovu http://www.novell.com/news/press/pressroom/wp_optaros_oss_usage_in_organizations.pdf
81. Weil, P., & Vitale, M. R. (2001). *Place to space: Migrating to eBusiness Models*. Boston: Harvard Business School Press.
82. West, J. (2003). How open is open enough? Melding proprietary and open source platform strategies. *Research Policy*, 32(7), 1259–1285.
83. Wheeler, D. A. (2007). How to Evaluate Open Source Software / Free Software. *Dwheeler.com*. Najdeno 25. junija 2010 na spletnem naslovu http://www.dwheeler.com/oss_fs_eval.html
84. Wichmann, T. (2002). Firms' Open Source Activities: Motivations and Policy Implications. *BERLECON RESEARCH*. Najdeno 26. julija 2008 na spletnem naslovu http://www.berlecon.de/studien/downloads/200207_FLOSS_Activities.pdf

85. Winter, S. G., & Szulanski, G. (2001). Replication as strategy. *Organization Science*, 12(6), 730–743.
86. Yin, R.K. (2002). *Case study research: Design and methods*. (3rd ed.). Thousand Oaks: Sage Publications.

7 PRILOGA

Priloga 1:

Pozdravljeni, moje ime je Marko in sem obiskovalec magistrskega študija na Ekonomski fakulteti v Ljubljani. V okviru svojega magistrskega dela z naslovom **ODPR TOKODNE PROGRAMSKE REŠITVE V POSLOVNIH MODELIH PODJETIJ** želim odgovoriti na raziskovalno vprašanje »Kakšni poslovni modeli obstajajo v podjetjih, ki temeljijo na odprtokodnih programskih rešitvah«. Vaše sodelovanje v anketi mi bo v veliko pomoč pri izdelavi strukture poslovnih modelov, ki so se do danes razvili v podjetjih. Anketa bo trajala do 30 minut.

Ponudba

Katere programske rešitve uporabljate oziroma ponujate?

Kateri je ciljni trg programskih rešitev?

Katere dodatne storitve so prisotne v ponudbi?

Kaj uporabnik plača in kaj s tem pridobi?

Katere licence so uporabljene pri programskih rešitvah?

Ali za različne segmente uporabnikov obstajajo različne ponudbe?

Kakšni so načini distribuiranja programskih rešitev?

Katere so prednosti in pomanjkljivosti odprtokodnih komponent?

Kako prilagajate različne verzije in nadgradnje programskih rešitev?

Viri

Kateri so ključni viri in zmogljivosti, ki jih mora imeti podjetje?

Kateri viri so potrebni za raziskave in razvoj programskih rešitev?

Kateri viri so potrebni za prodajne(trženjske) aktivnosti?

Katere vrste virov pridobivate iz odprtokodnih skupnosti?

Kako bi opisali možnosti za sodelovanje z viri odprtokodnih skupnosti?

Kako bi opisali neskladja pri sodelovanju z viri odprtokodnih skupnosti?

Odnosi

Kako dojemate odprtokodne skupnosti?

Kako odprtokodne skupnosti vplivajo na vaše tehnološko in poslovno odločanje?

Kdo so vaši ključni partnerji v odprtokodnih skupnostih? Zakaj?

Kakšne vrste odnosov imate s člani odprtokodnih skupnosti?

Kdo so ključni komercialni akterji v vašem poslovnem okolju (omrežju)?

Katere aktivnosti opravljajo partnerji v poslovnem okolju in odprtokodnih skupnostih?

Kako komunicirate z odprtokodnimi skupnostmi?

Kako spodbujate vključevanje v skupnosti?

Dohodkovni model

Kateri so glavni viri dohodkov?

Kdo so vaši odjemalci?

Kakšna je vaša osnova za izračun cene programskih rešitev?

Kdaj prejmete plačilo?

Kolikšen je strošek uporabe odprtokodnih programskih rešitev?