

Univerza v Ljubljani
Pravna fakulteta

FIDUCIARY LICENCE AGREEMENT
NOVI IZZIVI

DIPLOMSKA NALOGA

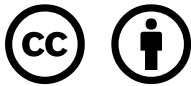
Avtor: Matija Šuklje
Mentor: prof. dr. Klemen Podobnik

Ljubljana, februar 2016

Zahvaljujem se svojemu mentorju za vse nasvete in pomoč pri pisanju te diplomske naloge ter odobritev tako kompleksne teme.

Hvala tudi društvu Free Software Foundation Europe, v katerem sem se ukvarjal z dokumentom, ki je bil podlaga za moje delo.

Z vso srčnostjo se zahvaljujem tudi svoji družini, še posebno mami, za dolgoletno podporo in neomajno zaupanje.



To delo je licencirano pod licenco **Creative Commons Priznanje avtorstva 4.0 Mednarodna**

Kopija licence je na tej povezavi:

<http://creativecommons.org/licenses/by/4.0/>

Aktualna različica te diplomske naloge in njena izvorna koda v ConT_EXt tipografskem jeziku sta na naslednjih povezavah:

<http://matija.suklje.name>

<https://gitlab.com/fla-llm-thesis/fla-llm-thesis>

Povzetek	4
Ključne besede	4
Summary	5
Keywords	5
1 Uvod	6
1.1 Terminologija	6
1.2 Zgodovinski kontekst in pregled alternativ	8
1.2.1 Licenčna pogodba med razvijalci	9
1.2.1.1 Model licenciranja <i>inbound</i> = <i>outbound</i>	10
1.2.1.2 Razvijalčev certifikat o izvoru	11
1.2.2 Pogodba o prenosu avtorskih pravic	12
1.3 Kaj je FLA in zakaj se uporablja	15
2 Kratka analiza uporabe FLA	17
2.1 Kdo oz. za kaj se uporablja FLA v praksi	17
2.1.1 Aeonice	18
2.1.2 Bacula	20
2.1.3 KDE	30
2.1.4 SELF Project & Consortium	33
3 Kompatibilnost s slovensko jurisdikcijo	36
4 Problemi in predlagane izboljšave	37
4.1 Kompatibilnost z več jurisdikcijami	37
4.2 Patenti	39
4.3 Blagovne znamke	42
4.4 Izbor izhodnih licenc	44
4.5 Delodajalci	45
4.6 Berljivost dokumenta	47
4.6.1 Laikom prijazen povzetek	49
5 Sklep	52

A	Glosar	53
B	FLA, version 1.2	57
C	Literatura	62

POVZETEK

Glavno pravno področje, ki se ukvarja z računalniškimi programi in tudi prostim programjem, je avtorsko pravo. V sodobnem svetu se programje razvija kolaborativno prek interneta, kar prinaša svojstvene izzive pri urejanju avtorskih pravic na programju.

V ta namen je bila ustvarjena Fiduciarna licenčna pogodba oz. na kratko FLA (ang. Fiduciary Licensing Agreement) kot posebna oblika prenosa avtorskih pravic na neko zaupanja vredno osebo (fiduciarja), da se poenostavi upravljanje avtorskih pravic na prostem programju.

Zob časa ni prizanesel niti FLA in zato je bil glavni namen te diplomske naloge analiziranje prej nepredvidenih problemov, ki zadevajo FLA, in njihovih morebitnih rešitev.

V uvodu so na kratko razloženi vsebina in cilji FLA, ki je umeščena v zgodovinski okvir s pregledom najbolj razširjenih prijemov problema raztresenih avtorskih pravic.

Nato je na podlagi štirih zelo različnih prostih projektov analizirano, kako se FLA uporablja v praksi – Aeonix, Bacula, KDE in SELF Project/Platform.

Za tem je prikazan preizkus, ali je FLA veljavna v slovenskem in tujih pravnih sistemih. Za tuja pravna sistema sem uporabil belgijski in indijski primer.

Na koncu so predstavljeni moji predlogi, kako naj bi se FLA dopolnila za lažje spopadanje s sodobnimi izzivi, kot so patenti, blagovne znamke, potreba po spremembi izhodne licence ter berljivost in razumljivost besedila.

KLJUČNE BESEDE

avtorsko pravo, prenos pravic, prosto programje, Free Software, FOSS, odprta koda, licence, programska oprema, blagovne znamke, patenti

SUMMARY

The main legal branch responsible for computer programs and therefore also Free Software (or Open Source) is copyright. In the modern world software is being developed collaboratively via the internet, which brings its own challenges when it comes to copyright management.

The Fiducary Licence Agreement (short: FLA) was created with the goal to address these issues. As a special kind of copyright assignment onto a trustworthy person (fiduciary) its aim is to simplify copyright management of Free Software.

The FLA has had its fair share of challenges during the past few years and as such the main aim of this masters thesis is analysing unforeseen problems and suggesting potential solutions.

Firstly, I summarise FLA's content and goals, as well as describe the history around it and the other popular instruments of copyright management.

Then follows the analysis of how four very different projects have been using the FLA in practice – Aeonic, Bacula, KDE and SELF Project/Platform.

After which I test whether the FLA is valid in Slovenia and abroad. Belgian and Indian legal systems serve as examples for issues found abroad.

Finally, I present my suggestions how to modify the FLA in order for it to easier tackle the modern challenges it faces, such as patents, trade marks, change of the project's outbound licence, as well as readability of the text itself.

KEYWORDS

copyright, copyright assignment, Free Software, FOSS, open source, licence, software, trade marks, patents

1 UVOD

Povod za to diplomsko nalogo je bilo večletno delo s FLA, saj sem leta 2010 kot koordinator pravnega moštva postal odgovoren za ta dokument v društvu Free Software Foundation Europe (na kratko: FSFE). Do leta 2016 sem tako odgovarjal na vprašanja glede FLA, predaval o tej temi in prepričeval razvijalce, naj jo podpišejo. Sčasoma sem odkril več pogledov na ta sicer precej kratek dokument, ki je v **prilogi B**.

Svoje izkušene sem razširil s svojimi poznejšimi raziskavami in to strnil v to diplomsko nalogo. Njeni izsledki pa bodo podlaga tudi za odločitev FSFE, kako naprej s FLA. Zato lahko ponosno trdim, da ima naloga tudi uporabno vrednost za širšo prostoprogramsko skupnost.

1.1 TERMINOLOGIJA

Ker se ta diplomska naloga ukvarja s tematiko, za katero ni veliko literature v slovenskem jeziku, je terminologija neizbežna težava.

Podobno kot doc. dr. Matija Damjan v svoji doktorski disertaciji¹ v tej nalogi uporabljam *prosto programje* kot *terminus technicus* za programje, ki je licencirano pod eno izmed prostih ali odprtih licenc. Ta je tudi sicer najbolj relevanten vir v domačem jeziku o pravnih vidikih prostega programja in preostalih prostih vsebin. Poleg tega me v to pojmovanje sili že izbira teme, saj FSFE, kot avtor in upravljavec FLA izrecno govori o Free Software² oz. prostem programju ter poudarja, da je izvorni termin edini primerni, ker bolje odslkava prostost oz. svoboščine uporabnika (ang. Free as in Freedom). Čeprav se pogovorno, predvsem zaradi lepšega zvena besedne zveze, pogosto za isto materijo uporablja pozneje skovani termin Open

¹ M. Damjan, PROSTA LICENCA KOT PRAVNI TEMELJ ODPRTE VSEBINE (2007), predvsem glej poglavje 2.4.6 Terminološka dilema.

² FSFE, What is Free Software?; URL: <https://fsfe.org/about/basics/freesoftware.html> (dne: 27. 12. 2015).

Source oz. odprta koda, zadnji navadno implicira bolj tehnične in manj družbene značilnosti takega programja³.

Definicija prostega programja je, da mora vsakdo imeti pravico, da tako programje 1) uporablja za vsak namen; 2) prouči, kako deluje, in ga spremeni po svojih potrebah in željah; 3) deli z drugimi programje v nespremenjeni obliki, kot ga je dobil, ter 4) deli z drugimi svojo, spremenjeno različico programa.

Že v naslovu in tudi v diplomski nalogi se pogosto pojavljajo termini *fiducija*, *fiduciary* ter *beneficiar*. Poudariti moram, da pri tem uporabljam samostojni termin, kot je zapisan v besedilu Fiduciary Licence Agreement, in ne npr. po SPZ⁴. Razprava o posrečenosti in skladnosti terminologije originalnega analiziranega besedila v primerjavi s slovenskim civilnim pravom ni predmet tega dela.

Opozarjam tudi, da se nekateri termini, uporabljeni v tej nalogi, še niso uveljavili v slovenskem jeziku, predvsem zato, ker se v računalniški terminologiji uporabljajo predvsem angleški izvorniki. Upam, da bo **Glosar v prilogi A** v pomoč bralcu in poznejšemu razvoju jezika. Nekaj terminov v njem sem moral izumiti tudi sam. Glede okornosti nekaterih se opravičujem in se nadejam, da bo glosar služil kot motivacija za kovanje boljših različic.

³ Na tem mestu je treba omeniti, da se zadnja leta relevantne organizacije – predvsem OSI in FSFE – aktivno trudijo za zmanjševanje te razlike in zbliževanje obeh taborov.

⁴ Stvarnopravni zakonik (SPZ), Ur. l. RS, št. 87/2002, 91/13.

1.2 ZGODOVINSKI KONTEKST IN PREGLED ALTERNATIV

Živimo v svetu, v katerem so računalniki povsod – od pametnega telefona, ki ga nosimo vedno s seboj, do naprav, kot sta televizija in avtomobil, do interneta. Programska oprema je tisti del računalnika, ki obdeluje podatke in prikazuje rezultate. Bolj poetično rečeno, če je strojna oprema računalnikov *corpus*, je programska oprema njegov *animus*⁵.

Primarno pravno zaščito za programsko opremo daje avtorsko pravo⁶. Bernska konvencija⁷ je prva, ki je definirala programsko opremo kot avtorsko delo.

V slovenski zakonodaji je programska oprema opredeljena v 2. oddelku Zakona o avtorski in sorodnih pravicah⁸, ki definira »računalnišk[e] program[e] [kot] program[e] v vsaki izrazni obliki, vključno s pripravljanim gradivom za njihovo izdelavo.«⁹

Če so v času Bernske konvencije programe večinoma pisali posamični avtorji, je zdaj precej drugače. Programska oprema je postala v desetletjih čedalje kompleksnejša. Ne le, da sodobni programi pogosto zavzemajo na tisoče in milijone vrstic programske kode, pogosto so tudi proizvod večletnega sodelovanja več avtorjev z vsega sveta. Na večjih projektih, kot so LibreOffice, jedro Linux in namizje KDE Plasma, pogosto istočasno dela na stotine, tudi tisoče razvijalcev.

Če se avtorji niso domenili drugače, je v večini primerov tak program v soavtorstvu vseh njegovih avtorjev, ki skupaj uživajo nedeljivo avtorsko pravico na tem delu¹⁰. Posledično se morajo vsi soavtorji strinjati z vsakršnim izvajanjem avtorskih pravic in njihovim upravljanjem¹¹.

Kadar so avtorske pravice tako razpršene med več avtorjev, sta seveda tudi izvajanje in zaščita pravic precej težje izvedljiva. Že objava objektne ali programske

⁵ Pomislek, kje je točna ločnica med strojno in programsko opremo, je za to diplomsko nalogo irelevanten. Poteka pa o tem v računalniški stroki zanimiva debata, rezultati pa bodo verjetno imeli tudi pravne posledice, predvsem za podjetja. Zavzemam se za stališče, da je ločnica močno zabrisana in da gre le za praktično odločitev, kdaj in kakšen program je primerneje »zapeči« v strojni in kdaj ga zapisati v spomin kot programsko opremo.

⁶ Za zdaj ignorirajmo možnost patentov na programski opremi. Temu problemu je namenjeno **poglavje 4.2**.

⁷ Bernska konvencija za varstvo književnih in umetniških del (MSBKVK), Ur. l. SFRJ, MP, št. 14/75, 4/86; Ur. l. RS št. 25/92; Ur. l. RS. MP št. 9/92 in 3/2007.

⁸ Zakon o avtorski in sorodnih pravicah (ZASP), Ur. l. RS, št. 16/2007, 68/08, 110/13 in 56/15.

⁹ 1. odstavek, 111. člen ZASP.

¹⁰ 1. odstavek, 12. člen ZASP.

¹¹ 2. odstavek, 12. člen ZASP.

kode namreč spada med ekskluzivne pravice, o katerih bi se morali (so)avtorji strinjati in jo izvajati skupno.

Ker zakonodaja ni bila kos taki situaciji, so gospodarske družbe in nepridobitne organizacije začele uporabljati različna pravna orodja za poenostavitev in centralizacijo pravic. V nekaterih redkih primerih velja to celo za skupnost razvijalcev, ki so brez formalne pravne osebnosti, a so kljub temu poenotili pravila prispevanja k projektu, ki so širše od (izhodne) licence.

Med njimi so najpogostejši naslednji instrumenti – vsaj v kontekstu prostega programja. To so *pogodbe o prenosu avtorskih pravic* (ang. Copyright Assignment Agreement, od tu CAA) in *licenčne pogodbe med razvijalci* (ang. Copyright/Contributor Licensing Agreement, od tu CLA) programa. Poleg teh dveh tipov sporazumov o sodelovanju (ang. Contributor Agreement, od tu CA) obstajajo tudi rešitve z drugačnim pristopom k isti problematiki, kot je npr. razvijalčev certifikat o izvoru (ang. Developer's Certificate of Origin, od tu DCO).

Čeprav so vsi ti instrumenti namenjeni reševanju enakih problemov – namreč temu, da lahko več razvijalcev pripomore k istemu delu in je končni produkt izdan pod enotno licenco –, se razlikujejo v ciljih in metodah.

To poglavje naj služi le kot kratek uvod v različne instrumente, ki jih prostoprogramska skupnost uporablja za urejanje pravic pri projektu. Več o tem – pogosto zelo pozicionirano – so pisali drugi avtorji. Za hiter in nevtralen pregled sta najbolj primerna vira ifrOSS¹² in M. Michlmayr¹³. Za bolj vsebinsko analizo pa sta priporočljiva M. Meeks¹⁴ (z vidika izkušenega razvijalca) in pa S. F. Jakob¹⁵ (s pravnega vidika). Od slovenske literature je na voljo doktorska naloga M. Damjana¹⁶.

1.2.1 LICENČNA POGODBA MED RAZVIJALCI

Najpreprostejši način, kako omogočiti, da bo končni proizvod objavljen pod enotno licenco, je, da se razvijalci, ki prispevajo svoje delo, vnaprej strinjajo, pod katero

¹² ifrOSS, Licence Center; URL: <http://www.ifross.org/en/license-center> (dne: 27. 12. 2015), poglavji E. in F.

¹³ M. Michlmayr, Open Source Contributor Agreements: Some Examples; URL: <https://fossbazaar.org/content/open-source-contributor-agreements-some-examples/> (dne: 28. 12. 2015).

¹⁴ M. Meeks, Some thoughts on Copyright Assignments; URL: <https://people.gnome.org/~michael/blog/copyright-assignment.html> (dne: 25. 12. 2015).

¹⁵ S. F. Jakob, A Qualitative Study on the Adoption of CAA and CLA in FOSS, v: JIPITEC, 5 (2014) 2, str. 105–115.

¹⁶ M. Damjan, PROSTA LICENCA KOT PRAVNI TEMELJ ODPRTE VSEBINE (2007), glej poglavje 8.4 Koncentriranje pravic na prostem programju.

licenco bodo objavili svoje prispevke in da bo s tem skupno delo objavljeno pod enako licenco.

Obstaja več načinov, kako se to izvaja, vendar gre s pravnega vidika pri vseh le za to, da vsi avtorji objavljajo svoj prispevek ali *a)* pod isto licenco, kot je končna licenca projekta/programa; ali pa *b)* pod bolj permisivno licenco, in s tem omogočijo spremembo izhodne licence.

Ker je v obeh primerih avtorstvo še vedno zelo razpršeno, ima ta način še vedno dve veliki pomanjkljivosti. Pri uveljavljanju avtorskih pravic na programu (pred sodiščem) je v večini jurisdikcij velika razlika, ali si imetnik izključnih pravic ali ne – brez njih je avtorskoppravna sodba zelo težko izvedljiva. Tudi če bi bilo v prihodnosti treba spremeniti licenco programa, je to mogoče le z veliko truda. Med večjimi projekti je to uspelo le VLC¹⁷, pri katerem je moral glavni razvijalec prepričati približno 230 prispevkarjev, da so se strinjali s spremembo licence, pod katero naj bo objavljeno njihovo avtorsko delo. To je bilo delo takih herkuleanskih razsežnosti, da so o tem poročali tudi mediji¹⁸.

1.2.1.1 MODEL LICENCIRANJA *inbound = outbound*

Pri licenciranju programske opreme se pogosto govori o dveh licencah – vhodni in izhodni. *Vhodna licenca* (ang. *inbound licence*) je tista, ki jo z vidika opazovanega subjekta dobi ta od izvirnega avtorja oz. licencodajalca iz povratnega toka. *Izhodna licenca* (ang. *outbound licence*) pa je tista, ki jo opazovani subjekt nudi tretjim osebam naprej v sprejemnem toku.

Najpreprostejši model licenciranja oz. licenčne pogodbe deluje tako, da vzdrževalec projekta sprejema samo prispevke prispevkarjev, ki so bili preneseni pod vhodno licenco, ki je enaka licenci, pod katero je celoten projekt objavljen oz. na voljo tretjim osebam – torej izhodno licenco projekta.

Ker gre pri tem za izenačitev vhodne in izhodne licence, se ta sistem tudi imenuje model licenciranja *inbound = outbound* – termin, ki ga je skoval R. Fontana¹⁹.

¹⁷ VideoLAN, VideoLAN – Uradna stran VLC predvajalnika, odprto kodnega video ogrodja!; URL: <https://www.videolan.org/vlc/> (dne: 01. 09. 2015).

¹⁸ N. Willis, Relicensing VLC from GPL to LGPL; URL: <https://lwn.net/Articles/525718/> (dne: 01. 09. 2015).

¹⁹ R. Fontana, The trouble with Harmony: Part 1; URL: <https://opensource.com/law/11/7/trouble-harmony-part-1> (dne: 22. 12. 2015).

Zaradi svoje preprostosti je zelo razširjen, vendar ima podobne težave kot vse licenčne pogodbe. Program ali skupnost, ki ga razvija, lahko postane večja in kompleksnejša. Zato se pogosto pojavi potreba po bolj prilagodljivi rešitvi.

1.2.1.2 RAZVIJALČEV CERTIFIKAT O IZVORU

Skupnost, ki razvija jedro Linux²⁰, ni bila nikoli naklonjena birokraciji in pravnim komplikacijam. Glede na številčnost razvijalcev in količino kode, ki jo napišejo vsak dan, je to dokaj razumljivo. Po uradnih podatkih Linux Foundationa²¹ naj bi pri razvoju jedra Linux namreč sodelovalo na tisoče razvijalcev (vsaj 6000 od leta 2005 do 2010) in prispevalo s hitrostjo približno 5 popravkov na uro – torej 120 popravkov na dan.

Da bi poenostavili pogoje in proces vhodnega licenciranja, so uvedli t. i. *Developer's Certificate of Origin*²² (slo. Razvijalčev certifikat o izvoru; kratko: DCO).

Vsak razvijalec, ko pošilja svoj prispevek v centralni repozitorij projekta jedra Linux, digitalno podpiše, da je ali *a*) sam avtor kode, ki jo prispeva; ali *b*) je kodo, ki jo prispeva, sicer dobil drugje, vendar ima pravice, da jo skupaj s svojimi popravki prispeva; ali *c*) je kodo dobil od tretje osebe, ki jamči za katerokoli izmed teh treh opcij²³. V vseh treh primerih razvijalec s svojim digitalnim podpisom potrjuje²⁴, da ima pravice licencirati prispevano kodo pod licenco, ki je navedena v datoteki.

Gre torej za zelo tehnično rešitev primarno pravnega problema. To je tudi razlog, zakaj postaja DCO v razvijalskih krogih čedalje bolj popularna.

²⁰ Linux je jedro popularnega prostega operacijskega sistema. Na strežnikih in namiznih računalnikih se jedro Linux najpogosteje kombinira skupaj z orodji GNU, ki potem skupaj tvorijo operacijski sistem GNU/Linux. Vzporedno je na ugnezdjenih napravah najpogostejši operacijski sistem BusyBox/Linux, na telefonih pa Android/Linux.

²¹ J. Corbet, G. Kroah-Hartman, A. McPherson, Linux Kernel Development – How Fast it is Going, Who is Doing It, What They are Doing, and Who is Sponsoring It; URL: http://www.linuxfoundation.org/docs/lf_linux_kernel_development_2010.pdf (dne: 04 . 02. 2016).

²² Linux Foundaton, Developer's Certificate of Origin 1.1; URL: <http://developercertificate.org/> (dne: 22. 12. 2015).

²³ Alineja c) je potrebna, ker ima pri razvoju tako kompleksnega projekta, kot je jedro Linux, pogosto le peščica ljudi dostop neposredno pošiljati kodo. To pomeni, da delujejo ti kot skrbniki projekta, ki pregledujejo kodo vseh preostalih prispevkarjev, preden jo potrdijo in vključijo v projekt.

²⁴ Tehnično gre sicer pri tem podpisu za posebno polje v sistemu Git, ki se imenuje »Signed-off-by« in vključuje ime in e-poštni naslov (npr. Signed-off-by: Matija Šuklje <matija@suklje.name>). Poleg tega pa obstaja tudi varni digitalni podpis, s katerim prispevkar zagotavlja, da je ta prispevek res poslala oseba, ki ga je podpisala.

Različni avtorji se ne strinjajo popolnoma, ali gre pri tem za poenostavitev CLA ali zgolj za obljubo oz. pisno potrdilo. Tako B. M. Khun²⁵ npr. meni, da gre pri DCO za *inbound=outbound* CLA, A. Metzger²⁶ pa v svoji analizi pravi, da z DCO razvijalec zgolj pisno potrdi, da ima primerne pravice. Strogo pravno-tehnično morda sicer res pri tem ne gre za licenčno pogodbo, vendar je v praksi (delno tudi zaradi socialnih norm) učinek enak kot pri licenčni pogodbi *inbound = outbound*. Zaradi tega sam štejem tudi bolj socialne rešitve, kot so DCO, Debian Social Contract²⁷, Fedora Project Contributor Agreement²⁸ in Mozilla Foundation Committer's Agreement,²⁹ kot CLA. Nasprotno jih klasificira ifrOSS³⁰ kot pogodbe o prispevanju (ang. Committer Agreements). Razmejitev med licenčnimi pogodbami med razvijalci na eni in pogodbami o prispevanju na drugi strani opravičujejo s tem, češ da prve vsebujejo avtorskopravno licenco, zadnje pa le napotujejo na zunanje, že obstoječe, proste licence (npr. Apache-2.0³¹, GPL-3.0 itd.). Menim, da je to razlikovanje odveč.

1.2.2 POGODBA O PRENOSU AVTORSKIH PRAVIC

Do zdaj so bile prikazane le različne oblike licenciranja – tako ozko kot potrditev oz. izjava o licenciranju (npr. DCO) ali pa popolna ekskluzivna licenca. Pri *pogodbi o prenosu avtorskih pravic* (angleško: copyright assignment agreement; kratko: CAA) pa gre seveda za točno to – prenos vseh avtorskih pravic na delu.

Pravno je situacija – vsaj teoretično – v tem primeru najpreprostejša. Avtor, kot cedent, s pogodbo prenese svoje pravice na cesionarja.

²⁵ B. M. Khun, Project Harmony (and »Next Generation Contributor Agreements«) Considered Harmful; URL: <http://ebb.org/bkuhn/blog/2011/07/07/harmony-harmful.html> (dne: 22. 12. 2015).

²⁶ A. Metzger, Internationalisation, v: SCRIPTed, 10 (2013) 2, str. 177–206.

²⁷ Debian, Debian Social Contract; URL: http://www.debian.org/social_contract (dne: 28. 12. 2015).

²⁸ Red Hat, Fedora Project Contributor Agreement; URL: https://fedoraproject.org/wiki/Legal:Fedora_Project_Contributor_Agreement (dne: 28. 12. 2015).

²⁹ Mozilla Foundation, Mozilla Foundation Committer's Agreement v2.1; URL: <https://static.mozilla.com/foundation/documents/commit-access/committers-agreement.pdf> (dne: 28. 12. 2015).

³⁰ ifrOSS, Licence Center; URL: <http://www.ifross.org/en/license-center> (dne: 27. 12. 2015), poglavje F.

³¹ Apache License 2.0

Ta navidezna preprostost odnosa je tudi eden izmed razlogov, zakaj se predvsem gospodarske družbe odločajo za tak način ureditve pravic na prostoprogramskem projektu. Drugi razlog je pogosto to, da temelji na CAA eden bolj razširjenih poslovnih modelov, t. i. strategija multi-licencing – torej, da se isti program nudi pod (praviloma) močno licenco copyleft brezplačno, a istočasno tudi ločeno pod zaprto, plačljivo licenco³². Tretji razlog je v tem, da si želijo gospodarske družbe imeti čim bolj popoln nadzor nad razvojem in distribucijo. Mogoče si je tudi zamisliti, da so prenosi pravic potrebni zaradi knjigovodstva in obveznosti do delničarjev.

Prvi tip je torej izključno CAA z vidika gospodarske družbe, drugi tip cesionarja, ki se pogosto prenese avtorske pravice na prostoprogramskem projektu, pa so nepridobitne nevladne organizacije (kratko: NVO). Te imajo navadno precej drugačen interes na programju in gre za to, da z njimi dosegajo javne interese. Tipičen primer sta npr. FSFE in KDE e.V., ki sta zavezani, da programi, ki so jima zaupani, ostanejo prosti.

Z vidika skupnosti razvijalcev določenega prostoprogramskega projekta je seveda razlika med tem, ali prenašajo svoje pravice na gospodarsko družbo ali NVO, precejšnja, saj sta interesa obeh tipov cesionarjev skoraj diametralna. FSFE je izrazila tudi svoje skrbi glede t. i. modela »Open Core«,³³ ki je pogosto razlog, zakaj se zahteva CAA.

Pogosto je kot glavni razlog, zakaj naj bi avtorji razvijalci prenesli svoje avtorske pravice za projekt na neko centralizirano entiteto, omenjena preprosta možnost relicenciranja pod drugo licenco in s tem prilagoditev projekta pravnemu in tehnološkemu napredku. Kot drugi resni razlog je ta, da se s prenosom pravic izogne negotovosti, ki sledi neizbežni smrti in dedovanju avtorskih pravic razvijalca. Pogosto navajajo tudi uspešnejše varstvo pravic avtorjev. Predpostavlja se, da bo imetniku vseh avtorskih pravic na delu lažje uspelo pred sodiščem kot pa zgolj enemu izmed stotih ali tisočih soavtorjev³⁴.

CAA torej prinaša pravno varnost za cesionarja, vendar imajo razvijalci – centri tudi veliko razlogov za skrb.

Ob pomanjkanju dodatnih klavzul prenos vseh avtorskih pravic na tretjo osebo oropa dejanske avtorje tako rekoč vseh pravic in nadzora nad njihovim delom.

³² Najpopularnejša, vendar ne edina, različica dual- oz. multi-licencing sistema je t. i. »open core«, pri katerem so osnovne funkcije na voljo pod licenco copyleft, dodatne pa samo proti plačilu za zaprto licenco.

³³ G. Greve, What makes a Free Software company?; URL: <https://fsfe.org/freesoftware/enterprise/freesoftwarecompany.en.html> (dne: 28. 12. 2015).

³⁴ V tej logiki je čutiti vzporednico s kolektivnim upravljanjem pravic. Kolektivno upravljanje kolaborativnih prostih del bi lahko bila zanimiva tema za poznejše raziskave.

Pogosto gre pri prostoprogramskih projektih za delo zagrizenih prostovoljcev in zato ti (razumljivo) nikakor ne zaupajo takim ekstremnim posegom. Ker nimajo več pravice o (so)odločanju, jih najpogosteje skrbi, da se bo njihovo delo nehalo deliti pod prosto licenco, kakršna je bila določena, ko so pristopili k projektu. Poleg tega je birokracija, potrebna za sklepanje takih pogodb – in v določenih jurisdikcijah kot na primer v Franciji vzdrževanje –, za marsikoga prevelik vstopni prag v skupnost in se zato že vnaprej vzdrži prispevanja v skupen razvoj.

Slovenska zakonodaja³⁵ in tudi nekatere druge jurisdikcije ne dopuščajo prenosa avtorskih pravic. Da se doseže podoben želeni učinek, številne CAA vključujejo ekskluzivno CLA kot varovalko za jurisdikcije, pri čemer polna CAA ni mogoča³⁶.

³⁵ Neprenosljive so moralne pravice, definirane v 16. členu ZASP.

³⁶ Poleg Slovenije tak prenos ni mogoč še vsaj v Avstriji, Nemčiji in na Madžarskem.

1.3 KAJ JE FLA IN ZAKAJ SE UPORABLJA

V **poglavju 1.2** sem nakazal, da imajo CLA in CAA svoje probleme – eni v pravni varnosti, drugi v zaupanju avtorjev. Da bi ponudila prostoprogramski skupnosti rešitev, je leta 2003 FSFE v sodelovanju z ifrOSS³⁷ napisala Fiduciarno licenčno pogodbo oz. FLA.

Fiduciarna licenčna pogodba (ang. Fiduciary Licence Agreement, kratko: FLA) je posebna oblika prenosa avtorskih pravic na neko zaupanja vredno osebo (fiduciar), ki je praviloma, vendar ne nujno, pravna oseba. Historično je šlo navadno za FSFE, vendar se je zadnjih letih razvilo v smer preference po prenašanju na specializirane organizacije in fundacije, ki se ukvarjajo s specifičnim prostim projektom³⁸.

Celotno besedilo aktualne različice – *FLA 1.2* – je na voljo na spletni strani³⁹ in tudi priloženo k diplomski nalogi v **prilogi B**.

Pri FLA gre primarno za CAA oz. v jurisdikcijah, kjer to ni mogoče, za ekskluzivno CLA. Hkrati pa je posebnost tega dokumenta v tem, da ščiti predvsem interes avtorja oz. beneficiarja⁴⁰, in ne poznejšega imetnika (materialnih) avtorskih pravic oz. fiduciarja⁴¹. Že od prvih osnutkov je bila prav ta garancija glavni motiv za nastanek FLA. To je tudi zaslediti iz prvih omemb iz leta 2003 v spletnih novicah FSFE⁴² in tudi v člankih glavnega avtorja FLA, A. Metzgerja, v ifrOSS⁴³ in Linux Magazin⁴⁴.

Glavni cilj FLA je, da omogoča prejemniku pravic – fiduciarju – vse pravice, ki jih potrebuje, da lahko vzdržuje prosti projekt ter ga brani pred morebitnimi (avtorsko-)pravnimi zlorabami in sodnimi postopki.

Da lahko to izpolni, mora tak prejemnik/fiduciar imeti možnost, da:

³⁷ nem. Institut für Rechtsfragen der Freien und Open Source Software, oz. ang. Institute for Legal Questions on Free and Open Source Software

³⁸ Tipičen primer je KDE e.V., ki drži avtorske pravice na projektih KDE. Več o tem v **podpodpoglavju 2.1.3**.

³⁹ FSFE, Fiduciary License Agreement (FLA); URL: <https://fsfe.org/activities/ftf/fla.html> (dne: 15. 12. 2014).

⁴⁰ V klasičnem slovenskem pravnem jeziku bi tako osebo imenovali cedent ali fiduciant.

⁴¹ V klasičnem slovenskem pravnem jeziku bi tako osebo imenovali cesionar.

⁴² FSFE, FSF Europe releases Fiduciary Licence Agreement V1.0; URL: <http://mail.fsfeurope.org/pipermail/press-release/2003q1/000049.html> (dne: 28. 12. 2015).

⁴³ A. Metzger, FSF Europe und ifrOSS verbessern Rechtsschutz für Freie Software: Fiduciary License Agreement veröffentlicht; URL: <http://www.ifross.org/artikel/fsf-europe-und-ifross-verbessern-rechtsschutz-fuer-freie-software-fiduciary-license-agreemen> (dne: 28. 12. 2015).

⁴⁴ A. Metzger, FLA: Zu treuen Händen, v: Linux Magazin, (2003) 5, str. 75–77.

- relicencira programje pod novo prosto licenco, v odgovor na spremembe tehničnega napredka in prava;
- če ima projekt tudi komercialni del, zagotavlja stabilno razmerje med komercialnim interesom ter interesom prostega programja in skupnosti;
- varuje programje pred zlorabo, v skrajnem tudi po sodni poti;
- varuje avtorje pred sodnimi postopki in drugimi pravnimi sredstvi naperjenimi zoper njih tako, da prevzame tveganje nase.

Pri tem moralne pravice ostanejo avtorjem, kar besedilo FLA tudi izrecno določa.

Najlažji način je seveda, da pridobi vse avtorske pravice na programju. Pri tem razvijalci/avtorji programja in tudi širša skupnost vlagata veliko zaupanja v takega prejemnika. FLA se uporablja v situacijah, ko je tako zaupanje že prej vzpostavljeno, hkrati pa za razne eventualitete v prihodnosti vključuje fiduciarno obveznost, ki onemogoča zlorabo.

V državah, v katerih popolni prenos avtorskih pravic ni mogoč – med drugim tudi v Sloveniji –, pa beneficiar na fiduciarja prenese izključno in globalno licenco, ki je sestavljena iz naslednjih pravic:

- pravice razmnoževanja v originalni in modificirani obliki;
- pravice distribuiranja v originalni in modificirani obliki;
- pravice dajanja na voljo javnosti prek podatkovnih omrežij, predvsem interneta, ter omogočanje shranjevanja – v originalni in modificirani obliki;
- pravice dovoliti tretjim osebam predelavo ali drugačno spremembo programa oz. take spremembe opraviti sam.

2 KRATKA ANALIZA UPORABE FLA

2.1 KDO OZ. ZA KAJ SE UPORABLJA FLA V PRAKSI

FLA uporabljajo manjši projekti⁴⁵ in tudi večji⁴⁶. Za večje projekte je uporaba FLA seveda bolj smiselna. Tudi vsebinsko in organizacijsko se razlikujejo, kot je razvidno v nadaljevanju.

Januarja 2014 sem v anketi izprašal vse štiri projekte, ki so uradno vključeni v FSFE Fiduciary Programme⁴⁷. Vsi štirje se med seboj precej razlikujejo. Izsledki so navedeni v tem poglavju.

Vprašanja iz ankete:

- Kako se imenuje projekt?
- Pod katerimi licencami je projekt objavljen?
- Katero različico FLA uporablja projekt in ali je besedilo FLA modificirano?
- Ali projekt zahteva, da razvijalec sklene FLA, preden sme prispevati k projektu?
- Prenašajo razvijalci s podpisom FLA svoje avtorske pravice (oz. ekskluzivno licenco) na FSFE ali kako tretjo osebo (na katero)?
- Kdaj je projekt nastal in od kdaj uporablja FLA?
- Se projekt še razvija ali je zamrl?
- Če je zamrl, zaradi katerih razlogov?
- Ali se vam zdi, da FLA izpolnjuje svoje naloge?

⁴⁵ Aeonix ima npr. samo 5 razvijalcev.

⁴⁶ KDE pa ima več kot 1800 razvijalcev (statistika iz 2010, verjetno je aktualno število večje).

⁴⁷ FSFE, Fiduciary Programme; URL: <https://fsfe.org/activities/ftf/fiduciary.html> (dne: 09. 01. 2016).

- So razvijalci sprejeli FLA brez večjih zadržkov in ali jim je besedilo dovolj razumljivo?
- Če je projekt uporabil FLA, da je spremenil licenco, zasledoval kršitev avtorskih pravic in podobno, kakšne so bile izkušnje?
- Kakšne so izkušnje projekta z birokratskim delom, povezanim s FLA?

2.1.1 AEONIC

Prvi projekt je prvi po abecedi in najmanjši od vseh omenjenih. Anketa je potekala po e-pošti in IRC z vodjo projekta in glavnim razvijalcem, Anastasiosom Hatzisom.

Aeonic⁴⁸ je poenoteni jezik modeliranja (UML) za razvoj spletnih in podatkovnih aplikacij. Projekt se je začel kot zaprto programje v podjetju HATZIS Edelstahlbearbeitung GmbH, ki ga je nato hotelo objaviti kot prosto programje z imenom PySwarm. Pozneje se je projekt preimenoval v OpenSwarm⁴⁹ in na koncu v Aeonic. Po navedbah glavnega razvijalca se je projekt začel leta 2007 kot zaprt program in objavil kot prosto programje leta 2009 s FSFE kot fiduciarjem pod FLA in pa GPL-2.0+⁵⁰ kot licenco. Nadaljnje raziskave so pokazale, da je projekt obstajal že leta 2006⁵¹ FLA je bila podpisana že leta 2007⁵².

Zadnja objavljena različica Aeonic je 0.7.1, in sicer pod licenco GPL-2.0+. Med pisanjem te naloge Anastasios Hatzis in FSFE sodelujejo v smeri, da objavijo novo različico 0.9 pod GPL-3.0. Razvoj se je sicer zaradi tehničnega preobrata začasno ustavil leta 2011, vendar naj bi leta 2016 izšla nova različica na novih tehničnih temeljih.

Podpis FLA je pogoj za prispevanje kode v projekt Aeonic in nosilec avtorskih pravic je FSFE.

⁴⁸ A. Hatzis &al., Aeonic – Model Driven Server Apps; URL: <http://sourceforge.net/projects/aeonic/> (dne: 01. 01. 2016).

⁴⁹ PySwarm in OpenSwarm sta trenutno imeni dveh povsem drugih projektov, ki očitno nimata nobene povezave z Aeonicom.

⁵⁰ GNU General Public License, različica 2 oz. po presoji licencojemalca katerakoli poznejša različica GPL. Najnovejša je različica 3.

⁵¹ A. Hatzis &al., Aeonic [prvi commit]; URL: <http://sourceforge.net/p/aeonic/code/1/> (dne: 01. 01. 2016).

⁵² FSFE, [FSFE PR][EN] FSFE becomes the legal guardian of the OpenSwarm Project; URL: <http://mail.fsfeurope.org/pipermail/press-release/2007q1/000165.html> (dne: 01. 01. 2016).

A. Hatzis je navedel *FLA 1.2* kot tisto različico, ki jo projekt uporablja, ter dodal, da je bila modificirana glede na uradno različico FSFE. Natančnejši pregled prikazuje uporabo različice 1.1 pred tem. Primerjava med najstarejšo *FLA 1.1* za takrat še PySwarm in originalno različico kaže naslednje spremembe:

- Vnaprej je izpolnjeno polje za subjekt pogodbe v §2, in sicer pravi:

»*All contribution to the pyswarm project*«

- V §3.3 je drugačen prvi stavek:

»*FSFE shall only exercise the granted rights and licences in accordance with the principles of Free Software.*«

namesto kot v 1.2 različici originala:

»*FSFE shall only exercise the granted rights and licences in accordance with the principles of Free Software as defined by the Free Software Foundations.*«

(poudarek moj).

Ob preimenovanju projekta v OpenSwarm je FLA ostala enaka in omenjene razlike prav tako.

Ko se je projekt končno preimenoval v Aeonice, je tudi prevzel *FLA 1.2* v identični obliki, kot je originalna različica FSFE⁵³. Ta različica je tudi tista, ki je še aktualna in na voljo na spletni strani projekta⁵⁴.

Vodja projekta pravi, da po njegovih izkušnjah FLA izpolnjuje svoje naloge in da jim je bil v precejšnji pomoč.

Prav tako naj bi razvijalci Aeonice brez težav sprejeli besedilo. Sicer jim je vodja projekta prej pojasnil osnove dokumenta in posledice podpisa, vendar je menil, da bi to moral tako ali tako za vsako licenco ali pogodbo.

Glede birokratskih ovir je Aeonice navedel zgolj en problem. In sicer naj bi eden izmed razvijalcev ne dobil odgovora FSFE, ko je podpisal in poslal FLA. V tem primeru so rešili težavo tako, da je ta razvijalec podpisal podobno pogodbo s podjetjem HATZIS Edelstahlbearbeitung GmbH in s tem posredno spravil svojo kodo pod FLA, saj je ta gospodarska družba že na začetku podpisala FLA s FSFE. Glavni razvijalec je predlagal, naj v prihodnje razvijalci pošiljajo podpisane FLA

⁵³ A. Metzger, G. C. F. Greve, S. M. Coughlan, Fiduciary License Agreement, version 1.2; URL: <https://fsfe.org/activities/ftf/FLA.en.pdf> (dne: 15. 12. 2014).

⁵⁴ FSFE, Fiduciary License Agreement (version 1.2) for Aeonice; URL: http://aeonice.sourceforge.net/get/Aeonice_FLA_en_v1.2.pdf (dne: 01. 01. 2016).

njemu in da jih bo on potem posredoval FSFE v nadaljnje obravnavanje. S tem bi si rad zagotovil boljši vpogled v stanje in s tem pospešil proces sprejemanja prispevane kode in dokumentacije.

Razen prej omenjenega zapleta niso imeli večjih birokratskih težav s FSFE kot fiduciarjem.

Projekt Aeonic tudi nikoli ni bil vpleten v sodne procese ali druge situacije, pri katerih bi bilo treba uporabiti celoten skupek avtorskih pravic oz. ekskluzivnih pravic proti tretji osebi.

Čeprav Aeonic 0.9 še ni izšel kot paket, je ta koda v repozitoriju SVN že prelicencirana v GPL-3.0+. To je A. Hatzis lahko storil brez privolitve FSFE kot fiduciarja in imetnika (ekskluzivnih) pravic, saj je bil projekt izrecno objavljen z opcijo, ki omogoča, da se licenca lahko spremeni v katerokoli različico GPL, ki je novejša od 2.0:

»[...] This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.«

(poudarek moj)

2.1.2 BACULA

Naslednji projekt, ki je bil sprejet v FSFE Fiduciary Programmu, je imel od vseh najbolj pestro življenje pod FLA. Pretežno zato, ker je vmes iz projekta Bacula.org⁵⁵ zrasla gospodarska družba Bacula Systems⁵⁶, ki je tržila isto kodo pod zaprto licenco.

Projekt Bacula se je začel leta 2000, ko sta ga Kern Sibbald in John Walker začela razvijati kot zaprtokodni sistem za ustvarjanje in urejanje varnostnih kopij. J. Walker in K. Sibbald sta dva izmed soustanoviteljev enega največjih računalniških podjetij svojega časa, Autodesk. Leta 1999 sta bila že upokojena in nista našla nobenega programa za varnostne kopije na sistemu GNU/Linux, ki bi jima bil všeč. Posledično sta se odločila, da ga bosta napisala sama in ga poimenovala

⁵⁵ Bacula.org, Bacula – Open Source Backup, Enterprise ready, Network Backup Tool for Linux, Unix, Mac, and Windows; URL: <http://bacula.org/> (dne: 09. 01. 2016).

⁵⁶ Bacula Systems, Bacula Systems – Backup, restore and disaster recovery; URL: <http://www.baculasystems.com/> (dne: 09. 01. 2016).

Bacula. Zgodbo zgodovine Bacule je K. Sibbald povedal leta 2005 v intervjuju⁵⁷ in leta 2013 na lastnem blogu⁵⁸.

Aprila 2002 je K. Sibbald javno objavil⁵⁹ Bacula kot projekt na SourceForge⁶⁰, takrat popularnem portalu za razvijanje prostih programov. Licenca za veliko večino kode je bila GPL-2.0 z izjemami za povezovanje kode z OpenSSL in MS Windows ter nekaj deli pod LGPL-2.1. Dokumentacija je bila objavljena pod GFDL-1.2⁶¹.

Novembra 2006 je postal Bacula prvi projekt, ki je prenesel svoje avtorske pravice prek FLA na FSFE⁶². V ta namen sta podpisala FSFE in K. Sibbald kot predstavnik projekta Bacula tudi memorandum o soglasju, ki podrobneje ureja razmerje med obema. Med drugim določa, da ima (glavni) administrator projekta Bacula zadnjo besedo pri izbiri proste licence, pod katero naj bo projekt objavljen.

Nato je K. Sibbald leta 2008 skupaj s še sedmimi ustanovil podjetje Bacula Systems⁶³, da bi lahko zagotovil vzdržen razvoj programja Bacula. Namen podjetja je bil služiti z zagotavljanjem storitev strankam, ki bi rade uporabljale Baculo, in s tem financirati razvoj programja Bacula. Debata o ustanovitvi podjetja naj bi potekala javno že od leta 2006 in bila FSFE znana.

Zaradi birokratskih zapletov v letih 2008 in 2009 na strani FSFE, ki so bili posledica selitve pisarne v drugo državo ter pomanjkanja kadra, se je K. Sibbald odločil prositi razvijalce, da FLA podpišejo kar z njim. Po lastnih navedbah je bil v to primoran, da bi lahko nadaljeval razvoj projekta. Istočasno pa naj bi ekskluzivne pravice posredno stekle spet v FSFE prek FLA, ki ga je K. Sibbald leta 2006 podpisal s FSFE. To interpretacijo so leta 2013 potrdili FSFE, Bacula Systems in K. Sibbald v tristranskem sporazumu⁶⁴ (a več o tem pozneje).

⁵⁷ N. Valentine, The Bacula Philosophy; URL: <http://www.linuxdevcenter.com/pub/a/linux/2005/04/07/bacula.html> (dne: 10. 01. 2016).

⁵⁸ K. Sibbald, The Hidden Cost of Creating Open Source Code; URL: <http://blog.bacula.org/110/> (dne: 11. 01. 2016).

⁵⁹ K. Sibbald, [Bacula-announce] Bacula 1.17; URL: <http://sourceforge.net/p/bacula/mailman/message/587465/> (dne: 10. 01. 2016).

⁶⁰ Bacula.org, Bacula – SourceForge project; URL: <http://sourceforge.net/p/bacula/> (dne: 10. 01. 2016).

⁶¹ GNU Free Documentation License, različica 1.2

⁶² FSFE, [FSFE PR][EN] FSFE becomes the legal guardian of the Bacula Project; URL: <http://mail.fsfeurope.org/pipermail/press-release/2006q4/000161.html> (dne: 10. 01. 2016).

⁶³ Bacula Systems, Bacula Systems – Backup, restore and disaster recovery; URL: <http://www.baculasystems.com/> (dne: 09. 01. 2016).

⁶⁴ FSFE, Bacula Agreement – FSFE Legal; URL: <https://fsfe.org/activities/ftf/bacula-agreement.en.html> (dne: 10. 01. 2016).

Julija 2010 je K. Sibbald po prejšnjem posvetovanju s FSFE in FSF spremenil licenco iz GPL-2.0 v AGPL-3.0 in obdržal iste izjeme za povezovanje kode.

Ker se prvotni poslovni model ni obnesel, se je Bacula Systems 2011 odločila spremeniti svojo ponudbo. Interna koda se je ločila na *Bacula Community Edition*, ki je ostala pod isto prosto licenco, in zaprtokodno *Bacula Enterprise Edition*. Zadnjo je Bacula Systems v novem poslovnem modelu tržil kot samostojen produkt in vanjo vključeval zmožnosti, ki jih v odprtokodni različici ni.

Situacija se konkretno zakomplicira leta 2012, ko se je del skupnosti z enim izmed poslovnih partnerjev podjetja Bacula Systems na čelu po tiho odcepil⁶⁵ in ustvaril odcepljeni projekt (ang. fork) z imenom Bareos in istoimensko podjetje – Bareos GmbH. Cepitev ni bila opravljena strokovno, na obeh straneh so bile storjene napake, ki so privedle do sodnega obračuna med obema podjetjema. V to situacijo je bil poklican FSFE kot imetnik pravic v večini kode obeh projektov.

Pogajanje je bilo dolgotrajno in naporno za vse vpletene, kar se je pokazalo⁶⁶ najprej v pisnem tristranskem soglasju⁶⁷ leta 2013 med društvom FSFE, podjetjem Bacula Systems in K. Sibbaldom (kot predstavnikom projekta Bacula). Glavne točke tristranskega dogovora so razčistile razmerja, kdo je imetnik katerih avtorskih pravic, in dolgoročno vzdržljivost projekta Bacula. Za širšo skupnost je verjetno najpomembnejša zaveza Bacule Systems, da mora najpozneje po petih letih vso kodo, ki jo lahko objavi brez kršitve pravic drugih, iz *Bacula Enterprise Edition* objaviti kot prosto programje v *Bacula Community Edition*. Pomembne so tudi varovalke za primer, če bi podjetje propadlo. V tem primeru pridobi FSFE neekskluzivne pravice na *Bacula Enterprise Edition* in je zavezan, da ga objavi pod prosto licenco. Podobno velja, če bi prenehal obstajati FSFE, ko pa pridobi (nazaj) avtorske pravice na Bacula K. Sibbald.

Več mesecev so tudi uspešno odpravljali avtorskopravne napake v odcepljenem projektu Bareos. Kljub temu se sprtima stranema ni uspelo izogniti sodni poti, saj je bil glavni spor povezan z *Bacula Enterprise Edition* in kršitvijo poslovne tajnosti (oz. pogodbe o nerazkritju). Na koncu je kljub temu prišlo do zunajsodne poravnave⁶⁸.

⁶⁵ K. Sibbald, Why Forking is Bad; URL: <http://blog.bacula.org/why-forking-is-bad/> (dne: 11. 01. 2016).

⁶⁶ K. Sibbald, Bacula Status Report (25. November 2013); URL: <http://blog.bacula.org/status1/> (dne: 11. 01. 2016).

⁶⁷ FSFE, Bacula Agreement – FSFE Legal; URL: <https://fsfe.org/activities/ftf/bacula-agreement.en.html> (dne: 10. 01. 2016).

⁶⁸ M. Außendorf, What about copyright issues between Bacula and Bareos?; URL: https://www.bareos.org/en/faq/copyright_bacula_bareos.html (dne: 11. 01. 2016).

Na tej podlagi je maja 2014 v dogovoru s FSFE in posvetovanjem s FSF projekt Bacula znova spremenil licenco⁶⁹. In sicer se je spremenila licenca dokumentacije v CC-BY-SA-4.0⁷⁰. Koda se je obdržala pod AGPL-3.0, vendar so izjeme in dodatna dovoljenja znova napisali in prečistili. Cilj novih dovoljenj in omejitev je bil zagotoviti navedbo avtorstva glavnega razvijalca K. Sibbalda, tudi v izvedenih delih in odcepljenih projektih, ter zaščiti blagovno znamko in patente.

Pozneje se je leta 2015 K. Sibbald odločil, da bi raje vodil svoje avtorske pravice na projektu Bacula sam. Razloga sta bila dva: 1) upravljanje avtorskih pravic v situaciji, v kateri sta njhova imetnika dva subjekta (FSFE in K. Sibbald), se je izkazalo za komplicirano; 2) menil je tudi, da ga FSFE proti Bareosu ni podpiral toliko, kot je pričakoval. Težava je bila v tem, da sta bila na strani Bacule in Bareosa podpisnika FLA s FSFE in je torej moral FSFE ščititi interes obeh sprtih strani. K. Sibbald je marca istega leta prosil, naj se prekine FLA med njim in FSFE. Temu je FSFE ustregel⁷¹. V novem tristranskem dogovoru med FSFE, Baculo Systems in K. Sibbaldom avgusta istega leta so se dogovorili glede načina prenosa pravic in bolj dodelali cilj vseh treh podpisnic, da se bo *Bacula Community Edition* aktivno razvijal in bo vedno objavljen kot prosto programje. O tem poročata K. Sibbald⁷² in FSFE⁷³.

Na podlagi tega dogovora sta K. Sibbald in FSFE stopila v stik z vsemi avtorji, ki so podpisali FLA s FSFE ter jih vprašala, kaj naj se zgodi z njihovimi pravicami. Na voljo so jim bile dane opcije, da se ali *a*) prenesejo pravice, ki so jih prenesli na FSFE na K. Sibbalda, ali *b*) prenesejo pravice nazaj na avtorja, ali pa *c*) pravice obdrži FSFE.

Avtorjev, ki so v več letih podpisali FLA s FSFE za kodo in jo prispevali projektu Bacula, se je zbralo okoli 60. Javni seznam vodi FSFE⁷⁴. Do dneva pisanja te diplomske naloge se je samo pet avtorjev izjasnilo, da bi radi, da FSFE obdrži njihove pravice⁷⁵. Vsi drugi so se strinjali s prenosom pravic na K. Sibbalda ali

⁶⁹ Bacula.org, Licenses – Bacula; URL: <http://blog.bacula.org/general/bacula-copyright-trademark-and-licenses/> (dne: 09. 01. 2016).

⁷⁰ Creative Commons: Priznanje avtorstva – Deljenje pod istimi pogoji, 4.0

⁷¹ FSFE, Statement on changed relations between the FSFE and Kern Sibbald; URL: <https://fsfe.org/news/2015/news-20150414-01.en.html> (dne: 10. 01. 2016).

⁷² K. Sibbald, Bacula Status Report 27 August 2015; URL: <http://blog.bacula.org/bacula-status-report-27-august-2015/> (dne: 10. 01. 2016).

⁷³ FSFE, New role for the FSFE in the Bacula project; URL: <https://fsfe.org/news/2015/news-20150817-01.en.html> (dne: 10. 01. 2016).

⁷⁴ FSFE, Fiduciary Licence Agreement List – Bacula; URL: <https://fsfe.org/activities/ftf/fla-list-bacula.en.html> (dne: 10. 01. 2016).

⁷⁵ Trije izmed njih so bili avtorji Bareosa.

izrecno ali molče po preteku 60-dnevnega roka od prejete dopisa. Med pisanjem naloge situacija še ni povsem razrešena, vendar hitro napreduje.

Prenos avtorskih pravic na FSFE je torej najprej potekal neposredno, tako da so razvijalci podpisali FLA s FSFE. Pozneje pa so podpisovali t. i. »[modificirane] FLA« (pozneje CAA) s K. Sibbaldom in s tem posredno prenašali svoje pravice na FSFE, saj je FLA, ki jo je K. Sibbald podpisal s FSFE, vključevala prenos pravic, ki jih je sam pridobil od razvijalcev. Navednice so tu na mestu, ker K. Sibbaldove spremembe FLA odstranijo pomembne varovalke, ki so sicer integralni del tega dokumenta in ki opravičujejo ime »Fiduciarni«.

Pred drugimi vprašanji še o zgodovini *FLA* in »[modificirane] *FLA*« kot pravnem dokumentu tega projekta:

- Leta 2006, torej na začetku, je projekt Bacula uporabljal *FLA* za prenos pravic na FSFE. Različica *FLA* je bila 1.1.
- Okoli leta 2009 je K. Sibbald začel prositi razvijalce, naj podpisujejo »[modificirano] *FLA*« kar neposredno z njim. V ta namen je vzel besedilo *FLA 1.1* in ga prikrojil:
 - odstranil je prvi odstavek preambule, ki govori o pomembnosti prostega programja v sodobni družbi in graja zaprto oz. imetniško programje;
 - za fiduciarja je namesto FSFE napisal sebe;
 - v §3 [*Fiduciary*]'s Rights and Re-Transfer of Non-Exclusive Licence je odstranil §3.3 – največjo varovalko v *FLA* –, ki sicer zavezuje fiduciarja, da mora objaviti kodo pod prosto licenco ter dodatno zavaruje beneficijarja (fiducianta) s tem, da izrecno omogoča avtomatični prenos pravic nazaj, če bi se fiduciar prekršil zoper to zavezo;
 - kot pristojno sodišče je določil stvarno pristojno sodišče v švicarskem mestu Lozana, v katerem prebiva tudi sam (prej je bil kot *forum* določeno nemško mesto München);
 - prav tako je za veljavno zakonodajo določil švicarsko zakonodajo (pred tem je bila izbrana nemška zakonodaja).

To novo pogodbo o prenosu avtorskih pravic je poimenoval »*Fiduciary Licence Agreement (Version 2.0)*«. Kot je razvidno, ta dokument ni imel veliko opravka s fiduciarnem zaupanjem in zato ni vreden tega imena.

- Leta 2010 se je pojavila ›[modificirana] FLA 3.0‹⁷⁶, ki se od ›[modificirane] FLA 2.0‹ razlikuje zgolj v tem, da je v preambuli izrecno zapisano, da je eden izmed ciljev pogodbe ta, da se omogoča K. Sibbald, da prispevano kodo objavi in distribuira tudi pod drugimi licencami in v drugih projektih.
- Različice ›[modificirane] FLA 4.0‹ mi ni uspelo najti.
- Januarja 2014, ko sem prejel odgovor ankete, je bila aktualna različica ›[modificirana] FLA 5.0‹⁷⁷. Ta je predvsem pomembna, saj gre za prvo različico po tristranskem dogovoru FSFE – Bacula Systems – Bacula. Če jo primerjamo z ›[modificirano] FLA 3.0‹, je razen kozmetičnih popravkov K. Sibbald naredil naslednje spremembe:
 - v dokumentu ni več govora o »Beneficiary«, torej beneficiarju, temveč se pojavlja samo še »Contributor«, torej prispevkar;
 - v §1 Grant je vrinjen nov odstavek §1.1, ki se glasi:

»(1) In the event that the author or owner of the contributed code has already signed an exclusive license to another entity or person, each use of the term ›License‹ shall mean ›non-exclusive license‹ everywhere within this agreement, otherwise the term ›License‹ shall mean ›exclusive license‹.«

Predstavljam si, da je ta odstavek K. Sibbald dodal kot poskus, da bi se razčistilo avtorsko-pravno razmerje med FSFE, K. Sibbaldom in prispevkarjem/beneficiarjem. V tej fazi je bilo že običajno, da je prispevkar k projektu Bacula podpisal nekoč že FLA s FSFE in pozneje ›[modificirano] FLA‹ z K. Sibbaldom (ne nujno v tem vrstnem redu).
 - med pravice, ki jih prispevkar prenese na K. Sibbalda, je v §1.2, eksplicitno dodal kot 5. alinejo pravico, da sme uporabljati, reproducirati, redistribuirati in ustvarjati izvedena dela programja pod drugimi licencami – torej tudi pod zaprtimi (neprostimi) licencami:

»5. the right to use, reproduce, redistribute and make derivative works of the Software under other licenses.«

⁷⁶ Bacula.org, Fiduciary License Agreement (Version 3.0) [kot arhivirano dne 10. 6. 2013 na Internet Archive]; URL: <https://web.archive.org/web/20130510151653/http://bacula.org/en/FLA-bacula.en.pdf> (dne: 12. 01. 2016).

⁷⁷ Bacula.org, Fiduciary License Agreement (Version 5.0) [kot arhivirano dne 7. 2. 2014 na Internet Archive]; URL: <https://web.archive.org/web/20140207093545/http://www.bacula.org/en/FLA-bacula.en.pdf> (dne: 10. 01. 2016).

- v §2 *Subject Matter* je vnaprej izpolnjeno polje in bolj specificirano, da gre za programsko kodo in dokumentacijo, ki jih prispevkar prispeva v Bacula.org projekt – t.j. *Bacula Community Edition*;
 - prav tako v §3 *Kern's Rights and Re-Transfer of Non-Exclusive license* je v §3.2 nadaljnje specificiral, da prispevkar spet pridobi neekskluzivno licenco zgolj na lastnem prispevku (in ne na celotnem skupnem delu), kar je skladno z interpretacijo originalne FLA;
 - v §4 *Miscellaneous* je nadalje opredelil, kako naj se izvajajo avtorske pravice v primeru smrti ene ali druge stranke v §4.1 ter v novem vrinjenem §4.2, da postane ›[modificirana] FLA‹ veljavna že ob podpisu prispevkarja. Za zadnje domnevam, da je poskušal pospešiti in poenostaviti postopek prenosa pravic, da ta ne bi oviral razvoja. Za spremembe v §4.1 pa ne vidim nobenega razloga, saj je bila originalna klavzula povsem primerna in je v praksi določala isto;
 - nazadnje je v §4 *Miscellaneous* dodal kot §4.3 ločitveno klavzulo.
- Različice ›[modificirane] FLA 6.0‹ mi ni uspelo najti.
 - Konec leta 2014 je projekt Bacula dobil ›[modificirano] FLA 7.0‹⁷⁸. Ta različica je bila (razen imena) tudi usklajena prej skupaj s FSFE. Poleg kozmetičnih popravkov in večje interne koherence terminologije so pomembne naslednje spremembe:
 - v §1 *Grant* in §4 *Miscellaneous* sta spremenjena §1.1 ter §4.2, tako da naj podpis te ›[modificirane] FLA‹ izrecno nadomesti prejšnje ›[modificirane] FLA‹ med podpisnikoma, če gre za isti osebi (torej, če je pravice prenašal K. Sibbaldu);
 - analogno temu novi vrinjeni §1.2 govori o tem, da prispevkar, če je prej podpisal FLA s FSFE, izrecno prenaša zgolj neekskluzivne pravice na K. Sibbalda;
 - §2 *Subject Matter* po novem pokriva tudi skripte in izrecno navaja, da se nanaša (tudi) na vso kodo, skripte in dokumentacijo, ki jo je avtor prispeval

⁷⁸ Bacula.org, Fiduciary License Agreement (Version 7.0) [kot arhivirano dne 18. 12. 2014 na Internet Archive]; URL: <https://web.archive.org/web/20141218084223/http://www.bacula.org/downloads/FLA-bacula.en.pdf> (dne: 12. 01. 2016).

pred podpisom pogodbe – po mojem mnenju tega ne bi bilo treba izrecno pisati, saj bi razumna interpretacija FLA rekla isto;

- v §3 *Kern's Rights and Re-Transfer of Non-Exclusive license* v §3.1 izrecno piše, da ima s tem K. Sibbald pravico tretjim osebam izdajati podlicence – te sicer v originalni FLA niso bile izrecno omenjene, saj se je ukvarjala samo z ekskluzivnimi pravicami in je zato smotrno, da je K. Sibbald to dodal. Po zakonu⁷⁹ sme podlicence namreč dajati zgolj pridobitelj izključne licence⁸⁰;
 - v §3.1 razširi možnost, da pridobitelj njene pravic izvaja tudi pred sodiščem, še na pravice prispevkarjev oz. podpisnikov ›[modificirane] FLA‹ – to se mi ne zdi potrebno, saj pridobi te pravice že prej s prenosom ekskluzivne licence oz. avtorske pravice (kjer je to mogoče). Tožbe na podlagi neizključnih pravic so verjetno na precej šibkih nogah, besedilo pa se mi ne zdi dovolj specifično, da bi z njim lahko že vnaprej pooblastili K. Sibbalda avtorji, ki so prenesli zgolj neizključne pravice in katerih izključne pravice ležijo pri FSFE;
 - nazadnje v §4 *Miscellaneous* sodišče, pristojno za spore iz te pogodbe še bolj konkretno določeno z navedbo naziva sodišča.
- Februarja 2015 je K. Sibbald na prošnjo FSFE preimenoval svojo pogodbo za prenos avtorskih pravic, da bi se izognil njeni zamenjavi z originalno FLA. Zato je izdal *Copyright Assignment Agreement (Version 8.0)*⁸¹, ki je razen nekaj kozmetičnih popravkov identična ›[modificirani] FLA 7.0‹, le ime je spremenjeno.
 - Od avgusta 2015 – in ob pisanju te naloge aktualna – je *Copyright Assignment Agreement (Version 9.0)*⁸². Vsebinsko pomembne spremembe, ki so bile prej usklajene s FSFE, pa so naslednje:
 - v §1 *Grant*, §1.2 je zdaj bolj konkretno določeno, kako s podpisom te CAA prenese avtor, ki je pred tem prenesel svoje izključne pravice na FSFE, te pa na K. Sibbalda. V ta namen se izrecno sklicuje na tristranski dogovor

⁷⁹ V Sloveniji določa (pod)licence Obligacijski zakonik (OZ), Ur. l. RS, št. 97/2007.

⁸⁰ 721. člen OZ.

⁸¹ Bacula.org, Copyright Assignment Agreement (Version 8.0) [kot arhivirano dne 17. 3. 2015 na Internet Archive]; URL: <https://web.archive.org/web/20150317080150/http://www.bacula.org/downloads/CAA-bacula.en.pdf> (dne: 12. 01. 2016).

⁸² Bacula.org, Copyright Assignment Agreement (Version 9.0); URL: <http://www.bacula.org/downloads/CAA-bacula.en.pdf> (dne: 10. 01. 2016).

FSFE – Bacula – Bacula Systems, podpisan avgusta 2015. V nadaljevanju je na kratko povzet;

- dodana je alineja §1.3.6, s katero avtor dovoljuje, da K. Sibbald prenese vse izključne in neizključne licence na pravno osebo, ki bi jo ustvaril in ki naj bi ga nasledila v pravicah in obveznostih iz te CAA. K. Sibbald je v pogovoru razkril, da razmišlja o ustanovitvi fundacije, ki bi po njegovi smrti nadaljevala prost razvoj projekta Bacula in bila prostovoljski antipol podjetju Bacula Systems;
- §2 *Subject Matter* je spet razširjen, da izrecno pokriva tudi tiste prispevke, ki so nastali po podpisu te CAA – tudi v tem primeru menim, da je ta dodatek odvečen;
- v §3 *Kern's Rights and Re-Transfer of Non-Exclusive license* je vrinjen nov odstavek §3.3, ki pravi, da bodo vsi prispevki, ki ne bodo sprejeti za vključitev v projekt Bacula, uničeni in posledično ne bodo predmet te CAA. Nasprotno so vsi sprejeti prispevki podpisanega avtorja pokriti s to CAA. Posledice tega so dvojne, in ne nujno takoj razvidne brez poznavanja ozadja in namena te klavzule: 1) avtor prispevka, ki ni bil sprejet v projekt, obdrži vse pravice na prispevani kodi in dokumentaciji in jih zato lahko uporabi kjerkoli; in 2) zaradi tristranega dogovora se ne sme uporabljati kode in dokumentacije, ki je bila sprejeta v projekt Bacula, v *Bacula Enterprise Edition*, ne da je hkrati tudi v *Bacula Community Edition*;
- prav tam je tudi vrinjen nov odstavek §3.4, ki jasno zapisuje, da morajo biti vsi prispevki, ki so pokriti s to CAA, objavljeni pod prosto licenco⁸³. Izrecno pa tudi omenja, da so lahko isti prispevki hkrati licencirani pod drugimi (tudi neprostimi) licencami. Ta klavzula CAA približa spet namenu originalne FLA, kar je bil tudi namen dolgoletnega pogajanja s FSFE.

Kot je razvidno iz tega zgodovinskega pregleda, je s pravnega vidika razvoj projekta Bacula zelo pester. V njegovi zgodovini pa so se spreminjale izhodne licence projekta, imetništvo projekta in pogodbe, s katerimi so se prenašale avtorske pravice oz. izključne vhodne licence (v neki fazi celo neizključne vhodne licence).

Da bi to zmešnjavo odpravili, je bilo potrebne veliko volje in truda na strani K. Sibbalda, Bacule Systems in FSFE. Ker sem sodeloval pri pogajanju, predstavljam

⁸³ Bolj specifično pravi, da morajo prispevki ostati prosto programje in odprta koda kot ta dva termina definirata FSF in OSI.

aktualno stanje projekta, in ne zgolj situacije, ki se je pokazala po opravljeni anketi januarja 2014.

Januarja 2014 je bila torej koda objavljena pod AGPL-3.0, skupaj z nekaterimi izjemami in dodatnimi dovoljenji, dokumentacija pa pod GFDL-1.2. Stanje ob pisanju diplomske naloge je, da je koda objavljena pod AGPL-3.0, vendar z malce drugačnimi izjemami in dodatnimi dovoljenji; dokumentacija pa je po novem pod CC-BY-SA-4.0.

FLA je bila za projekt Bacula vedno obvezna – v kakršnikoli obliki je že bila – in tako tudi ostaja. Izjema so samo popravki kode, ki so krajši od 10 vrstic. Trenutna situacija je, da projekt uporablja K. Sibbaldovo lastno CAA 9.0 – ki sicer izhaja iz FLA – in z njo velika večina razvijalcev prenaša pravice oz. izključne licence na K. Sibbalda. Peščica razvijalcev pa še vedno preferira, da FSFE drži njihove pravice. Razmerje med FSFE, projektom Bacula in Baculo Systems nadaljnje opredeljujeta tristranska dogovora iz let 2013 in 2015.

Projekt je še zelo živ, čeprav je vmes doživel vsaj dve odcepitvi projekta⁸⁴.

K. Sibbald meni, da ima FLA več napak. Za nekatere trdi, da so zgolj stilske oz. da se vidi, da besedila ni napisal nekdo, čigar materni jezik je angleščina. V to kategorijo spadajo npr. to, da naj v *FLA 1.2* ni v vseh členih enako konsistentno napisano, da sme fiduciar programje (in dokumentacijo) objaviti pod prosto in tudi pod zaprto licenco. Druge napake so bile povezane predvsem z nesrečno situacijo, da so razvijalci podpisovali FLA (in CAA) s FSFE in tudi K. Sibbaldom.

Kljub temu pravi K. Sibbald, da so razvijalci, ki so prispevali k projektu Bacula, brez večjih pomislekov podpisali FLA (in pozneje CAA). Kljub temu meni, da »mogoče eden izmed 10.000 razvijalcev razume razliko med izključnimi in neizključnimi licencami«.

Kot pravi K. Sibbald v anketi, je Bacula doživela spremembe licence in tudi bila vpletena v tožbo glede kršitev avtorskih pravic in licenc – zadnje sicer kot Bacula Systems. Glede spremembe licence navaja, da mu je prenos pravic na FSFE upočasnil in otežil spremembe. To je pričakovati, saj je cilj FSFE prek FLA zagotovljati, da programje ostane prosto. K. Sibbaldov pogled na to pa je nekje odstopal od pogleda FSFE, zato pogajanje pogajanje. V povezavi s kršitvami (licence in) avtorskih pravic je K. Sibbald izrazil svoje nezadovoljstvo s FSFE in navedel, da je imel občutek, kot da FSFE ne ščiti njegovih interesov kot beneficiarja. Treba je poudariti, da je šlo za spor med dvema beneficiarjema in je FSFE (neformalno in delno uspešno) mediirala z namenom, da bi se na obeh straneh odpravile kršitve.

⁸⁴ Drugi odcepek je Burp, ki pa za to diplomsko nalogo ni relevanten.

Pomembno je tudi poudariti, da se je sodni proces sam skoncentriral predvsem na *Bacula Enterprise Edition* in na kršenje poslovne tajnosti. Pri ničemer od obojega ni mogla FSFE pomagati, saj na tem delu kode ni imela avtorskih pravic. Da bi po §3.1 podelila K. Sibbaldu mandat, da zastopa tudi FSFE (in s tem njene avtorske pravice) nasproti Bareosu, pa se FSFE ni zdelo upravičeno. K. Sibbaldu tudi ni bilo všeč, da licenca AGPL dopušča, da se njegovo ime odstrani iz glave datotek, ki vključujejo kodo. Tako je tudi po interpretaciji FSF, ki je avtor licence AGPL. Razlog je v tem, da se kot nujni del oznake licence v glavah kode štejeta le oznaka, kdo je imetnik avtorskih pravic, in navedba licence. Vse drugo sme pridobitelj licence spreminjati – torej tudi dodatno omembo, kdo je glavni avtor/razvijalec. Večino teh pomislekov, če ne celo vseh, se je do pisanja diplomske naloge že rešilo ali so v sklepni fazi reševanja. Je pa treba priznati, da je trajalo precej več časa, kot bi si povprečni računalniški razvijalec želel.

Med problemi, ki jih K. Sibbald vidi v FLA, je ta, da ker odcepljeni projekt Bareos ne zahteva podpisa FLA ali CAA, lahko ta vzame kodo iz *Bacula Community Edition*, dokler se drži licence. Nasprotno pa ni res, saj Bacula zahteva podpis CAA, od razvijalcev konkurence pa nikakor ne more pričakovati, da bodo prenesli avtorske pravice ali izključno licenco nanj. Pri tem je treba poudariti, da te težave ne bi bilo oz. bi bila precej manjša, če ne bi poslovni model Bacula Systems temeljil na t. i. modelu *dual-licencing*, v katerem trži *Bacula Enterprise Edition* pod zaprto licenco, njen pretežni del pa sestavlja koda *Bacula Community Edition*. Tako pa se je Bacula Systems sam skopal v to luknjo.

Primeru Bacula se je nekako razvil kot stresni test FLA – v njem so se pojavile nekatere pomanjkljivosti dokumenta in napake v upravljanju FLA. Po drugi strani pa so se potrdile tudi nekatere glavne namere – kot npr. zagotoviti, da se programje nadaljuje kot prosto programje –, pa čeprav tega glavni razvijalec morda tisti trenutek ni (več) želel.

2.1.3 KDE

Projekt KDE⁸⁵ se je začel leta 1996 kot odgovor na probleme takratnih namizij in grafičnih programov za GNU/Linux oz. sisteme Unix. V tistem času sta bili najboljši opciji CDE⁸⁶ – Common Desktop Environment – in knjižnica Motif. V obeh

⁸⁵ KDE, KDE – Experience Freedom!; URL: <https://kde.org> (dne: 14. 01. 2016).

⁸⁶ KDE je bila najprej kratica za Kool Desktop Environment. Šala je bila skrita v tem, da zaradi zmešnjave s CDE imena projekta ni(so) črkovali z besedo »Cool«. Pozneje se je projekt preprosto preimenoval v K Desktop Environment. Od leta 2009 pa se uporablja samo kratica KDE, saj vključuje več kot le namizje.

je prvotni avtor KDE Matthias Ettrich videl različne probleme. Med največjimi, da so se grafični vmesniki vseh teh programov med seboj preveč razlikovali. V ta namen je torej začel KDE, ki je prerastel to, da bi bil le eden najzmogljivejših in popularnih prostih namizij in se prelevil v širši projekt, ki ima pod svojim okriljem poleg standardnega nabora namiznih rešitev tudi vmesnik za krmiljenje 3D-tiskalnikov, napreden urejevalnik filmov, profesionalne astronomske programe in celo portal za e-izobraževanje šolskih predmetov. Projekt je torej leta 2016 še zelo živ.

FLA je KDE začel uporabljati šele po 12 letih, ko je avgusta 2008⁸⁷ prevzela *FLA 1.2*, s to razliko, da je za fiduciarja veljala KDE e.V.⁸⁸ namesto FSFE. To je bila logična poteza, saj je že od leta 1997 obstajalo društvo KDE e.V.⁸⁹ z namenom podpirati projekt KDE pravno, finančno in organizacijsko.

Ob pisanju diplomske naloge uporablja KDE prirejeno različico *FLA 1.3.5*. Vse razlike so na kratko povzete na spletni strani⁹⁰ in storjene v posvetovanju s FSFE.

Tako je med veljavno različico⁹¹ ter *FLA 1.2* razlika samo v osebi fiduciarja, projektu primerno prirejeno polje za vnos, kateri prispevki se smatrajo kot predmet pogodbe, ter posebna pravila o tem, kdaj in kako se lahko koda in dokumentacija relicencirajo. Ta pravila so zapisana v posebnem pravilniku – imenovanem *FLA Relicensing Policy*⁹² (slo. FLA politika za relicenciranje oz. kratko FRP).

FLA Relicensing Policy je poleg KDE Free Qt⁹³ Foundation⁹⁴ s pravnega vidika najbolj inovativna rešitev tega projekta. Namen in cilj tega dokumenta sta, da dopolni FLA s tem, da zelo natančno specificira, pod katerimi licencami se sme katere dele projekta KDE relicencirati in kakšen je postopek za tako spremembo.

⁸⁷ KDE e.V., KDE e.V. – FSFE welcomes KDE's adoption of the Fiduciary Licence Agreement (FLA); URL: <https://ev.kde.org/announcements/2008-08-22-fsfe-welcomes-fla.php> (dne: 14. 01. 2016).

⁸⁸ KDE e.V., KDE e.V.; URL: <https://ev.kde.org> (dne: 14. 01. 2016).

⁸⁹ nem. »eingetragener Verein« pomeni registrirano društvo

⁹⁰ KDE, KDE e.V. – Fiduciary Licensing Agreement; URL: <https://ev.kde.org/rules/fla.php> (dne: 14. 01. 2016).

⁹¹ FSFE, Fiduciary License Agreement (version 1.3.5) [za KDE]; URL: <https://ev.kde.org/resources/FLA.pdf> (dne: 15. 01. 2016).

⁹² KDE e.V., FLA Relicensing Policy [za KDE]; URL: <https://ev.kde.org/resources/FRP.pdf> (dne: 14. 01. 2016).

⁹³ Qt je knjižnica, na kateri temelji KDE, in zato obstajajo posebni dogovori z vsakokratnim podjetjem, ki jo ima v lasti.

⁹⁴ KDE, KDE Free Qt Foundation; URL: <https://www.kde.org/community/whatiskde/kdefreeqtfoundation.php> (dne: 16. 01. 2016).

FRP deli licence na *šibke* (LGPL, BSD-2-Clause, MIT), *močne* (GPL) in *druge* (specifična varianta GFDL-1.2). KDE knjižnice⁹⁵ morajo biti pod šibko licenco, vse druge kode pod močno licenco, dokumentacija pa pod preostalimi dovoljenimi licencami.

Če bi KDE e.V. ocenila, da mora spremeniti licenco na določenem delu kode, ki jo je pridobila s FLA, mora slediti naslednjim pravilom:

- KDE e.V. mora vložiti razumen trud v kontaktiranje avtorjev, beneficiarje.
- Strinjanje vseh avtorjev/beneficiarjev, katerih kodo se poskuša relicencirati, je potrebno za spremembe, kjer je licenca v isti skupini.
- Če v razumnem roku (30 dni) katerikoli od vprašanih avtorjev⁹⁶ eksplicitno nasprotuje tej spremembi licence, ne sme KDE e.V. vsaj 60 dni predlagati nove licence za isto kodo.
- Če bi KDE e.V. rada spremenila licenco kode, ki spada v eno skupino v licenco iz druge skupine – npr. da bi spremenila licenco knjižnice kdepimlibs iz šibke v močno –, je potrebno spletno glasovanje skupščine društva.
- Če bi KDE e.V. rada spremenila licenco dokumentacije ali predlagala kako nepredvideno spremembo, se tudi mora opraviti spletno glasovanje skupščine društva.

FRP je zelo stabilen pravni dokument, saj mora KDE e.V. njegove spremembe doseči na skupščini z isto večino, kot je potrebna za spremembe statuta društva. Zato so lahko beneficiarji KDE e.V. popolnoma mirni, saj vedo, da se trenutna licenca njihovih prispevkov in tudi seznam potencialnih naslednjih licenc ne moreta kar arbitrarno spremeniti čez noč.

Ker je projekt KDE tako pozno prevzel FLA in ker ima na tisoče razvijalcev⁹⁷, so ocenili, da bi bilo preveč zahtevati od vseh, da podpišejo FLA in prenesejo svoje avtorske pravice oz. izključne licence na KDE e.V.. Čeprav ne zahtevajo podpisa FLA od razvijalcev⁹⁸, jim pa to toplo priporočajo. Zanimivo je, da sem poleti 2015 na Academy⁹⁹ zvedel, da morajo zaradi potrebe spremembe licence nekega paketa

⁹⁵ Po novem se imenujejo KDE Frameworks, ampak v FRP še niso preimenovane. KDE e.V. sem že kontaktiral glede te pomanjkljivosti.

⁹⁶ Ta veto ima samo avtor sam oz. pravna oseba, ne pa njegovi dediči ali drugi pravni nasledniki.

⁹⁷ KDE po večini meril šteje za drugi največji projekt prostega programja – takoj za jedrom Linux.

⁹⁸ Leta 2015 naj bi približno petina avtorjev KDE podpisala FLA.

⁹⁹ Academy je vsakoletna celotedenska konferenca za razvijalce in uporabnike KDE, na kateri sem tudi že dvakrat predaval in vodil delavnice o FLA – leta 2013 v Bilbao in leta 2015 v A Coruñi.

prositi prispevkarje, ki niso podpisali FLA, da dajo glavnemu vzdrževalcu tega specifičnega paketa *carte blanche* pisno dovoljenje, da lahko celotno kodo spremeni v primerno licenco.

V pogovorih z nekdanjim predsednikom KDE e.V. Aaronom Seigom in predsednico Lydio Pintscher se je potrdilo, da KDE e.V. meni, da je FLA zelo dober dokument, ki jim poenostavi delo zagotavljanja preživetja projekta in ga prilagodi izzivom v prihodnosti. Kolikor jim je znano, naj bi FLA bila razumljiva in veljavna v vseh jurisdikcijah, v katerih imajo razvijalce – sam sem sicer pozneje izvedel, da je v Indiji problem, a več o tem pozneje v **podpoglavju 4.1**. Pravijo tudi, da hranijo skenirane FLA in s tem nimajo težav. Prav tako so zelo zadovoljni s sodelovanjem s FSFE.

Glede razvijalskih sprejemaj FLA jim je težje, saj nimajo vzvoda, da sprejemajo samo prispevke, ki so predmet FLA. Zato, kot pravijo, je zelo pomembno, da razvijalcem približajo FLA in jim razložijo vsebino in tudi, zakaj je za preživetje njihovih prispevkov znotraj projekta tak prenos potreben. Seigo je v najinem pogovoru leta 2013 šaljivo omenil, da razvijalci vedno radi iščejo luknje in napake, zato moraš biti dobro pripravljen, da imaš dober odgovor. Kakor so se sami naučili iz tega, je pametno imeti strokovnjaka na konferencah¹⁰⁰. Še bolje pa bi bilo, če bi imeli nekoga zaposlenega, čigar glavna naloga bi bila odgovarjanje na taka vprašanja in urejanje potrebne birokracije.

KDE e.V. pravi, da mu v vseh teh letih ni bilo treba izkoristiti svoje moči za spremembo licence ali tožiti koga zaradi kratenja avtorskih pravic.

Projekt KDE je v tem pregledu torej izstopal zaradi dveh lastnosti – 1) podpis FLA ni pogoj za prispevanje; in 2) uvedli so specifična interna pravila za spremembo licence – FRP.

2.1.4 SELF PROJECT & CONSORTIUM

SELF Project¹⁰¹ je bil edini, od katerega nisem dobil odgovora, saj je projekt že zaključen in zgolj arhiviran. Odgovore na vprašanja sem moral izslediti na spletu in s poizvedbami znotraj FSFE, ki je (bil) član SELF Consortium in imetnik avtorskih

¹⁰⁰ Zadnja leta sem za to odgovoren jaz v vlogi koordinatorja pravnega moštva FSFE in tudi dolgotrnega uporabnika KDE ter celo podpisnika FLA s KDE e.V.

¹⁰¹ SELF Consortium, SELF Project – Sharing knowledge about Free Software; URL: <http://www.selfproject.eu> (dne: 28. 12. 2015).

pravic projekta. Dokumentacijo projekta¹⁰² sem presenetljivo zasledil na straneh Free Knowledge Instituta¹⁰³, ki sicer ni bil član SELF Consortium, vendar se je razvil¹⁰⁴ stran iz projekta SELF in zato vodi njegov arhiv.

SELF¹⁰⁵ projekt je nastal s pomočjo evropskih sredstev in ga je vodil SELF Consortium. Konzorcij je sestavljalo sedem organizacij s treh celin, in sicer: Internet Society Nederland, Universitat Oberta de Catalunya, Free Software Foundation Europe, University of Goteburg, Internet Society Bulgaria, Fundacion Via Libre in Homi Bhabha Centre for Science Education.

Projekt SELF je bil mednarodni projekt, ki je razvil načrt in prototip platforme za sodelovanje pri sestavi prostih (oz. odprtih) izobraževalnih virov in gradiva za usposabljanje o prosti programski opremi in odprtih standardih. Zasnova platforme je bila zgrajena kot mehanizem za ocenjevanje, prilagajanje, ustvarjanje in prevajanje tega gradiva po navdihu Wikipedije. Imela pa je nekaj inovativnih prilagoditev, specifičnih kolaborativnemu pisanju učnih gradiv.

Projekt se je začel leta 2006 in se *de facto* končal leta 2012. Glavni razlog za njegov konec je bil, da ga je financirala Evropska komisija le do 2008¹⁰⁶, potem pa jim je počasi začelo primanjkovati vetra v jadrih. Koda se je razvijala še do leta 2010 in je še vedno na voljo¹⁰⁷.

Fiduciar je bila FSFE, in sicer je šlo za *FLA 1.2* – torej aktualno različico, kot jo analiziram v tej diplomski nalogi. Iz dokumentacije projekta¹⁰⁸ izhaja, da je bila FLA modificirana glede na uradno FSFE *FLA 1.2*. Vendar natančnejši pregled pokaže, da mu je v praksi vsebinsko identičen. Razlika je zgolj v tem, da je v formularju že vnaprej zapisano, da so objekt prenosa oz. licence vsi prispevki k projektu SELF (ang. original: »All contributions to the SELF project«).

¹⁰² FKI, The SELF Project; URL: <http://freeknowledge.eu/projects/self> (dne: 30. 12. 2015).

¹⁰³ FKI, Free Knowledge Institute; URL: <http://freeknowledge.eu> (dne: 30. 12. 2015).

¹⁰⁴ FKI, FKI: Strategy/Track record; URL: http://wiki.freeknowledge.eu/index.php/FKI:Strategy/Track_record (dne: 30. 12. 2015).

¹⁰⁵ Kratica SELF pomeni znanost, izobraževanje in učenje v svobodi/prostosti (ang. original: Science, Education and Learning in Freedom)

¹⁰⁶ FSFE, Projects – SELF: Science, Education and Learning in Freedom; URL: <https://fsfe.org/campaigns/self/self.en.html> (dne: 27. 12. 2015).

¹⁰⁷ SELF Consortium, SELF Platform; URL: <https://savannah.nongnu.org/projects/self-platform> (dne: 28. 12. 2015).

¹⁰⁸ FSFE, SELF Legal Policy, Workpackage 4; URL: http://freeknowledge.eu/sites/freeknowledge.eu/files/D3_WP4-v1.0.pdf (dne: 30. 12. 2015).

Programje, ki je bilo napisano za ta projekt, je bilo objavljeno pod GPL-2.0+¹⁰⁹ in razvijalčev podpis FLA je bil prvi pogoj, da je smel kodo prispevati v centralni repozitorij projekta.

Dokumentacija in pa učno gradivo sta bila objavljena primarno pod GFDL z opcijo objave učnega gradiva pod kako drugo licenco – npr. licenci CC-BY¹¹⁰ in CC-BY-SA¹¹¹ pa sta bili izrecno omenjeni v projektni dokumentaciji. Vendar zaradi disfunkcionalne spletne strani, na kateri se gradivo hrani, ne morem preveriti dejanskega stanja. Podpis FLA ni bil zahtevan za prispevanje gradiva, je pa obstajala možnost, da se avtor gradiva kljub temu odloči prenesti svoje avtorske pravice prek FLA na FSFE.

¹⁰⁹ GNU General Public License, različica 2 oz. po presoji licencojemalca katerakoli poznejša različica GPL. Trenutno je najnovejša različica 3.

¹¹⁰ Creative Commons: Priznanje avtorstva

¹¹¹ Creative Commons: Priznanje avtorstva – Deljenje pod istimi pogoji

3 KOMPATIBILNOST S SLOVENSKO JURISDIKCIJO

FLA je bila že v osnovi pisana tako, da bi bila čim širše uporabljiva in globalno veljavna. Tako že v konceptiji upošteva oba sistema – *droit d’auteur* in *copyright*.

Tako v slovenskem pravnem sistemu sicer po §1.1 v §1 *Grant* ne bo upoštevan prenos avtorskih pravic, ampak izključna licenca oz. prenos izključnih materialnih pravic na fiduciarja. To je predvideno tudi že v dokumentu, v katerem je Slovenija izrecno omenjena v opombi kot ena izmed držav, v kateri naj se aplicira ta izjema.

Sicer v dokumentu ni videti ničesar, kar bi bilo v nasprotju z ZASP ali OZ. Tudi M. Damjan¹¹² ne navaja nobenih problemov s slovensko zakonodajo.

Težava bi kvečjemu nastopila, če bi beneficiar poskušal prenesti »materialne pravice na *vseh* svojih prihodnjih delih«¹¹³ (moj poudarek). Vendar dokument nikoli ni bil namenjen prenosu vseh del, ampak zgolj del na specifičnem projektu. Sicer bi bilo mogoče, da bi stranki obrazec izpolnili tako, da bi bila njuna FLA nezakonita po ZASP, vendar bi bila taka uporaba precej neverjetna.

¹¹² M. Damjan, *PROSTA LICENCA KOT PRAVNI TEMELJ ODPRTE VSEBINE* (2007), str. 292–295, poglavje 8.4.4. FSF Europe: Fiduciary Licence Agreement.

¹¹³ 79. člen, 3. alineja ZASP.

4 PROBLEMI IN PREDLAGANE IZBOLJŠAVE

V naslednjih poglavjih so natančneje opredeljeni posamezni mogoči problemi oz. popravki.

Prvo podpoglavje (**podpoglavje 4.1**) je namenjeno kratkemu pregledu, ali FLA deluje po pričakovanju svojih avtorjev tudi v drugih jurisdikcijah. Za tem sta poglavji posvečeni vprašanju, ali je treba dokument razširiti tudi na patente (**podpoglavje 4.2**) in blagovne znamke (**podpoglavje 4.3**). Na podlagi teh spoznanj in novih okoliščin, v **podpoglavju 4.4** predlagam spremembo seznama licenc, pod katerimi naj bi se zadevni program smelo izdati. Za tem je **podpoglavje 4.5**, ki na kratko obravnava problematiko prispevkov k skupnemu programu/delu prispevkarja v delovnem razmerju in znotraj delovnega časa. Med zadnjimi problemi, ki jih analiziram, je forma dokumenta (**podpoglavje 4.6**).

4.1 KOMPATIBILNOST Z VEČ JURISDIKCIJAMI

Specifike nekaterih zakonodaj po svetu glede prenosa avtorskih pravic se zelo razlikujejo – zato bo v nekaterih primerih treba FLA dopolniti, da bodo tudi razvijalci/avtorji iz teh držav lahko prenesli svoje pravice.

Kljub Bernski konvenciji, TRIPS¹¹⁴ in WCT¹¹⁵ ima avtorsko pravo po svetu precej lokalnih specifik.

Avtorji FLA so že v zgodnjih osnutkih poskušali upoštevati različne avtorsko-pravne sisteme in zato je v besedilu precej izjem in opomb, ki stranke vodijo po specifikah določenih jurisdikcij.

Izrecno so tako omenjene Nemčija, Avstrija, Slovenija in Madžarska kot primeri držav, v katerih prenos avtorskih pravic ni mogoč. V tem primeru se po §1.1 prenaša izključna licenca.

¹¹⁴ Agreement on Trade-Related Aspects of Intellectual Property Rights

¹¹⁵ WIPO Copyright Treaty

Drugje v besedilu je tudi omenjena Francija, kot primer jurisdikcije, kjer ni mogoče prenesti pravic na bodočem delu. Tako morajo francoski beneficiarji redno obnavljati FLA, če želijo, da so njihovi prispevki z njo pokriti.

Pri raziskovanju sem našel jurisdikciji, pri katerih nastopi težava s trenutnim besedilom FLA.

Prva je Belgija, v kateri, kot poroča Y. van den Brande v svoji analizi FLA¹¹⁶, ni mogoče prenašati avtorskih pravic za nedoločen čas. Tako bi bilo primerneje, da se namesto časovno neomejenega prenosa pravic oz. licence omeji ta prenos oz. licenco na »čas trajanja avtorskih pravic na delu«. Taka rešitev bi še vedno izpolnjevala cilje in lastnosti FLA ter hkrati tudi zadovoljila belgijsko zakonodajo. Y. van den Brande omenja tudi zanimivo specifično prenosov avtorskih pravic po belgijskem pravu, ki zahteva, da se pri tem označi tudi žanr dela. Očiten problem je, kako določiti žanr – to priznava tudi Y. van den Brande.

Druga je Indija, v kateri je prav tako problem s časovno omejenostjo prenosa¹¹⁷. V tem primeru pa gre za težavo, da če ni eksplicitno in nedvoumno napisano, da gre za globalen in časovno neomejen prenos pravic, indijsko avtorsko pravo postavlja *presumptio iuris*, da se teritorij nanaša zgolj na Indijo in da gre za prenos pravic samo za rok petih let. Po njihovem poteku avtorske pravice stečejo nazaj k avtorju. To je določeno v naslednjih dveh odstavkih §19 *Mode of assignment* indijskega zakona o avtorskih pravicah¹¹⁸.

»(5) *If the period of assignment is not stated, it shall be deemed to be five years from the date of assignment.*

(6) *If the territorial extent of assignment of the right s is not specified, it shall be presumed to extend within India.*«

Trenutno besedilo FLA v §1 ter §3.4 po mojem mnenju izpolnjuje, kar zahteva indijska zakonodaja. Vendar bi bilo kljub temu smotrno besedilo predelati tako, da bi bila razsežnost prenosa vidna že na prvi pogled. Smiselno bi bilo tudi kar uporabiti omejitev, ki jo predlaga Y. van den Brande, saj bi morala časovna omejitev na celotno trajanje avtorskih pravic biti zadostna za vse namene.

¹¹⁶ Y. van den Brande, FLA, v: *International Free and Open Source Law Review*, 1 (2009) 1, str. 9–14.

¹¹⁷ Na ta problem sem naletel na *Academy 2013*, ko mi je med podpisovanjem FLA s KDE e.V. indijski razvijalec (in pozneje dober znanec) Vishesh Handa omenil to težavo. Zahvaljujem se dragemu kolegu Aahitu Gabiju, ki mi je namignil, kje naj najdem relevantno indijsko zakonodajo.

¹¹⁸ 5. in 6. odstavek, člen 19 *Copyright Act, 1957* (*Copyright Act, 1957*), *The Official Gazette [of India]*, št. 14/1957, 1999/49, 2012/27.

4.2 PATENTI

Patenti so v zadnjih dveh desetletjih žal čedalje večja nevarnost za prosto programje. Najbolj boleče na njih je, da v nasprotju z avtorskimi pravicami sploh ni treba priti z izumom v stik, da lahko patent kršiš. Poleg tega obstoj patentov na isti stvari pogosto izniči izjeme avtorskega prava, na katere bi se sicer lahko upravičeno naslanjali – s tem kombinacija teh dveh monopolnih pravic podre kakršnokoli ravnotežje.

Zavest o njihovem obstoju se vidi tudi v tem, da se eksplicitne patentne klavzule pojavljajo tudi v najbolj popularnih prostih licencah: Apache-2.0, GPL-3.0 in MPL-2.0¹¹⁹. Prva eksplicitna patentna klavzula se je pojavila že leta 1998 v MPL-1.1¹²⁰. Pred tem pa so se pojavljale tudi implicitne patentne klavzule (npr. v licencah MIT¹²¹ in BSD¹²²) ali pa zgolj omemba problematike v preambuli (kot npr. v GPL-2.0) v prostih licencah prve generacije, kot jih imenuje M. Bain.

Patentne klavzule v licencah sicer pomagajo rešiti težavo izhodne licence, ne pa vhodne s strani prispevkarjev. Več o vplivih patentov na proste licence opisujeta M. Damjan¹²³ in pa M. Bain¹²⁴. Zdaj je na vrsti problematika patentov z vhodne strani projekta.

M. Damjan je identificiral tri neposredne grožnje¹²⁵ patentov prostemu programju in M. Bain gre tu še v večje detajle. Dve izmed treh groženj sta zunanji, in sicer:

- če prosto programje krši že obstoječe patente proizvajalcev lastniškega programja;
- če proizvajalci lastniškega programja naknadno patentirajo izume, ki so bili prvotno razviti znotraj prostega programja;

Za prvi problem obstajajo rešitve kot pristop k patentnim skupnostim, kot je npr. Open Invention Network¹²⁶.

¹¹⁹ Mozilla Public License Version 2.0

¹²⁰ Mozilla Public License Version 1.1

¹²¹ MIT License

¹²² BSD licenc je več. Najbolj uporabljeni med njimi sta BSD 3-clause »New« or »Revised« License ter BSD 2-clause »Simplified« License

¹²³ M. Damjan, PROSTA LICENCA KOT PRAVNI TEMELJ ODPRTE VSEBINE (2007), poglavje 7.

¹²⁴ M. Bain, PATENTS AND FOSS (2013).

¹²⁵ M. Damjan, PROSTA LICENCA KOT PRAVNI TEMELJ ODPRTE VSEBINE (2007), str 251.

¹²⁶ OIN, Open Invention Network – Home; URL: <https://www.openinventionnetwork.com/> (dne: 27. 1. 2016).

Za drugi problem obstajajo rešitve, kot so t. i. obrambne objave¹²⁷, da pozneje patentov za ta izum sploh ne more vložiti.

Ostaja nam torej še vedno tretji problem, in sicer:

- če razvijalci prostega programja sami patentirajo svoje izume.

Tega se lahko rešimo v pogodbi o prenosu pravic na projekt – in torej v FLA.

FLA zdaj ne rešuje tega problema, bi bilo pa to smotrno.

ContributorAgreements.org¹²⁸ ponuja dve možnosti za vključevanje patentne problematike v prenos pravic na projekt oz. fiduciarja. Prva je po vzoru IBM, Red Hat in Google zgolj obljuba, ki jo podpiše prispevkar, da ne bo izvajal pravic iz patenta:

»3.1 Pledge of Identified Patents

Whereas You are aware of certain of Your own patents, which You desire to identify and disclose as the following patents and patent applications (>Pledged Patents<):

Number or Application Number: _____

Country: _____

You hereby pledge to all developers, distributors and users of the Contribution or the Material to which this Agreement applies (>Pledge Recipient<) that You and your Affiliates will not bring a lawsuit or other legal proceeding against a Pledge Recipient for patent infringement under any Pledged Patents based on the Pledge Recipient's making, having made, using, selling, offering for sale, importing or otherwise transferring the Contribution or the Material. You will require any person or entity to whom You sell or transfer any of the Pledged Patents to agree, in writing, to abide by the pledge and to place a similar requirement on any subsequent transferees to do the same.

3.2 Revocation of Patent Pledge

You reserve the right to revoke the patent pledge stated in section 3.1, as if it never existed with regard to any Pledge Recipient, who files a lawsuit or other legal proceedings for patent infringement or who has a direct financial interest in such a lawsuit or other legal proceedings against You or your Affiliates.«

¹²⁷ OIN, Defensive Publications; URL: <http://www.defensivpublications.org/> (dne: 27. 1 . 2016).

¹²⁸ ContributorAgreements.org, ContributorAgreements.org; URL: <http://contributoragreements.org/> (dne: 23. 11. 2015).

Ta možnost je seveda bolj zanimiva za podjetja, ki nerada licencirajo svoje patente. Ker pa je cilj FLA, da se čim bolj zaščiti prostoprogramski projekt, pa bi neizključna patentna licenca bila bolj smotrna, saj se v tem primeru patentne klavzule v prostih licencah še toliko bolj izrazijo in projekt dodatno varujejo.

Predlagam torej, da se v prihodnje besedilo FLA vključi v spodnjo – ali njej enakovredno – klavzulo o patentni licenci iz ContributorAgreements.org:

»3.1 Patent License

Subject to the terms and conditions of this Agreement You hereby grant to us a worldwide, royalty-free, non-exclusive, perpetual and irrevocable (except as stated in Section 3.2) patent license, with the right to transfer an unlimited number of non-exclusive licenses or to grant sublicenses to third parties, to make, have made, use, sell, offer for sale, import and otherwise transfer the Contribution and the Contribution in combination with the Material (and portions of such combination). This license applies to all patents owned or controlled by You, whether already acquired or hereafter acquired, that would be infringed by making, having made, using, selling, offering for sale, importing or otherwise transferring of Your Contribution(s) alone or by combination of Your Contribution(s) with the Material.

3.2 Revocation of Patent License

You reserve the right to revoke the patent license stated in section 3.1 if we make any infringement claim that is targeted at your Contribution and not asserted for a Defensive Purpose. An assertion of claims of the Patents shall be considered for a ›Defensive Purpose‹ if the claims are asserted against an entity that has filed, maintained, threatened, or voluntarily participated in a patent infringement lawsuit against Us or any of Our licensees.«

Taka rešitev se mi zdi povsem mogoča tudi po slovenski zakonodaji¹²⁹.

¹²⁹ Zakon o industrijski lastnini (ZIL-1), Ur. l. RS, št. 51/2006, 100/13.

4.3 BLAGOVNE ZNAMKE

Prostoprogramska skupnost se je zavedela pomembnosti blagovnih znamk in pravilnega ravnanja z njimi, najpozneje s pojavom problemov, ki sta jih imela Mozilla in Arduino s svojimi blagovnimi znamkami.

Mozilla Foundation¹³⁰ je znana predvsem po svojem prostem brskalniku Firefox. Pred kakim desetletjem, ko je postal zares popularen, je več različnih združb začelo ponujati programe pod blagovno znamko Firefox (ime in logotip), ki pa niso bili (povsem) originalni. Kršitve so bile različne – od finančnih goljufij, ko so poskušali prenesti program, do modificiranih brskalnikov, ki so bili polni zlonamerne kode. V ta namen je Mozilla Foundation ustvarila posebno politiko za svoje blagovne znamke¹³¹ in hujše kršitelje tudi kazensko preganjala. Treba pa je omeniti, da vsi iz prostoprogramske skupnosti niso zadovoljni z Mozillino politiko, vendar poglobljanje v to presega okvire te diplomske naloge.

Arduino¹³² (oz. v Evropi zdaj Genuino) je najbolj razširjen prosti oz. Open Source Hardware¹³³ mikrokrmilnik na svetu. Nastal je leta 2005 iz sodelovanja najprej treh in pozneje petih razvijalcev. Ko sta iz tega zaradi poslovnih nesoglasij nastali dve podjetji – Arduino LLC v Združenih državah Amerike ter Arduino SRL v Italiji – in se je izkazalo, da je Arduino SRL za hrbtom preostalih partnerjev registriral blagovno znamko v Italiji, je nastal velik problem¹³⁴.

Ta dva primera dobro nakazujeta, kako je za uspešen projekt – pa naj si bo to programska ali strojna oprema, komercialno ali zastonjsko – lahko vitalnega pomena blagovna znamka¹³⁵.

Niti ni potrebno niti primerno prenašati blagovne znamke v FLA ali katerikoli drugi CAA ali CLA. Razlog je zelo preprost¹³⁶ – namreč blagovna znamka ščiti

¹³⁰ Mozilla Foundation, Mozilla – Skupaj gradimo boljši internet; URL: <https://www.mozilla.org/s1> (dne: 22. 1. 2016).

¹³¹ Mozilla Foundation, Mozilla Trademark Policy; URL: <https://www.mozilla.org/en-US/foundation/trademarks/policy/> (dne: 21. 1. 2016).

¹³² Arduino, Arduino; URL: <https://www.arduino.cc> (dne: 22. 01. 2016).

¹³³ Definition of Free Cultural Works, OSHW; URL: <http://freedomdefined.org/OSHW> (dne: 22. 01. 2016).

¹³⁴ N. Willis, A trademark battle in the Arduino community; URL: <https://lwn.net/Articles/637755/> (dne: 21. 01. 2016).

¹³⁵ Kot domač primer lahko služi Kiberpipa, ki si je leta 2013 zagotovila nadaljnji obstoj s pomočjo blagovne znamke, ki sta jo nosila dva od soustanoviteljev, in ne organizacija, ki ji je Kiberpipa prvih 12 let pripadala.

¹³⁶ In hkrati nakazuje, zakaj je uporaba nadpomenke »intelektualne pravice« povsem zgrešena za tako različne pravice.

povsem drugi vidik prostega programja kot avtorske pravice in patenti. Pri zadnjih dveh gre za zaščito vsebine, pri znamki pa za prepoznavnost oz. identiteto¹³⁷ produkta. Tudi relevantni subjekti so različni. Če je za nastanek prostega programa treba združiti prispevke več različnih avtorjev (in morda izumiteljev) v skupno delo, gre pri znamkah za enoznačno oznako enega dela. Iz tega sledi, da ni niti pričakovati, da bo vsak (ali sploh kateri) razvijalec/prispevkar prispeval tudi blagovno znamko. Zato taka klavzula za FLA ni primerna. Če se registrira ali prenaša blagovna znamka za projekt, naj se to stori posebej.

Je pa seveda pametno, da ima prostoprogramski projekt svoje blagovne znamke urejene in tudi pravila oz. politiko, kdo in kako jih lahko uporablja. Pri tem je potrebnega veliko občutka, da bi se izognili pretirani previdnosti in s tem negativno vplivali na neposredne uporabnike projekta in širšo skupnost. Paziti je namreč treba, da npr. prosta avtorskopravna licenca projekta zapoveduje uporabo znamke¹³⁸, hkrati pa jo licenca blagovne znamke prepoveduje – v taki situaciji bi namreč prejemnik licence nujno ali *a)* se izognil kršitvi blagovne znamke in s tem kršil avtorsko pravico ali pa *b)* se izognil kršitvi proste licence in s tem avtorske pravice in bi posledično kršil blagovno znamko. O tem, kako naj se prostoprogramski projekti lotijo blagovnih znamk in na katere pasti je treba paziti – predvsem, da zaradi proste licence ne izgubi blagovne znamke –, govori N. Shemtov¹³⁹. Zanimivo bo tudi slediti novemu projektu FOSSmarks¹⁴⁰, ki razvijalcem nudi znanje in pomoč glede blagovnih znamk.

¹³⁷ Tu leži morda edino stičišče med glavnimi moralnimi avtorskimi pravicami in znamko – izvor in identiteta originatorja. Tudi v skupnostih, ki se ukvarjajo s prostimi vsebinami, je mogoče pogosto občutiti močno povezanost avtorjev s svojim delom, čeprav so ga sicer pripravljene dati javnosti povsem na voljo.

¹³⁸ Taka praksa, čeprav ni priljubljena, ni prepovedana niti po GPL in jo FSF izrecno dovoli. Kot primer lahko navedem, da ima projekt Bacula dodano v AGPL-3.0 klavzulo, ki zahteva, da se navede blagovna znamka Bacula®.

¹³⁹ N. Shemtov, *Trade Marks and FOSS*, v: *FOSS – POLICY, LAW & PRACTICE* (2013), str. 109–140.

¹⁴⁰ P. S. Chestek & al., *FOSSmarks*; URL: <http://fossmarks.org/> (dne: 21. 01. 2016).

4.4 IZBOR IZHODNIH LICENC

Kot se je izkazalo predvsem pri Baculi (v poglavju 2.1.2) in KDE (v poglavju 2.1.3), je pri večjih projektih izziv ne samo zbiranje avtorskih pravic v eno (praviloma pravno) osebo, ampak sčasoma tudi sprememba (izhodne) licence.

FLA se v različici 1.2, kot jo je napisala FSFE, tega ne dotika in prepušča to problematiko povsem fiduciarju. S tem, da mu seveda lahko nudi strokovne nasvete, če jih fiduciar želi.

KDE e.V. se je odločila za način relicenciranja¹⁴¹, ki dobro zadene ravnotežje med srednjo in dolgoročno predvidljivostjo na eni strani ter dolgoročno prilagodljivostjo na drugi, s tem ko je sestavila *FLA Relicensing Policy*¹⁴². V tem dokumentu določa dovoljene licence in tudi proces, kako se bo relicenciralo, koliko se bodo pri tem kljub temu upoštevale želje originalnih avtorjev / prispevkarjev in nazadnje, kakšen je postopek za spremembe.

Tak način urejanja problema morebitnega relicenciranja je primer dobre prakse. Zdi se mi, da bi ga morali prevzeti vsi (vsaj večji) projekti, ki uporabljajo FLA.

Seveda tudi FRP, kot vse rešitve, ni brez svojih težav. Ker je FRP del FLA in za določenega beneficiarja veljata tisti različici teh dokumentov, kot jih je podpisal, se lahko zgodi, da za različne prispevkarje veljajo različna pravila o (re)licenciranju. Zato je pametno 1) voditi seznam beneficiarjev in (vseh)različic FLA & FRP, ki so jih podpisali, in po potrebi pozvati, naj podpišejo novo različico; ter 2) dobro premisli med vsako spremembo FRP, da se ne uvede različic, ki bi bile med seboj nekompatibilne.

Če smo že pri analizi FRP¹⁴³ za projekt KDE, se je mogoče posvetiti še temu, kar bi se mi zdelo smotrno popraviti v njem. Menim, da bi blo bolj skladno s prostoprogramsko terminologijo, da se v §2 *Definitions* preimenuje skupino licenc »Weak« v »Weak & Non-copyleft« ter jim doda LGPL-3.0(+). Temu primerno bi bilo smotrno tudi skupino »Strong« preimenovati v »Strong copyleft« ter mednje dodati AGPL-3.0(+). Med preostale licence bi bilo smotrno dodati CC-BY-SA 4.0 in/ali CC-BY 4.0 ter ukiniti GFDL zaradi nekompatibilnosti z GPL. Kode pod GFDL (npr. iz dokumentacije) namreč ni mogoče uporabiti v kodi, ki je pod GPL, kodo iz CC-BY 4.0 in CC-BY-SA 4.0 pa je.

Na koncu je tudi smotrno izpostaviti dobro prakso, da KDE e.V. vodi seznam sprememb FLA in FRP na svoji spletni strani¹⁴⁴.

¹⁴¹ Več o tem v poglavju 2.1.3.

¹⁴² KDE e.V., FLA Relicensing Policy [za KDE]; URL: <https://ev.kde.org/resources/FRP.pdf> (dne: 14. 01. 2016).

¹⁴³ Analiziram trenutno aktualno različico *FLA Relicensing Policy (Version 1.3.4)*.

4.5 DELODAJALCI

Težava pri prenašanju avtorskih pravic je včasih, da njihov imetnik ni (več) avtor sam. Tipičen primer so dela, opravljena po naročilu, ali pa med delovnim razmerjem.

To različne jurisdikcije rešujejo različno, zato je pomembno, da se tega zavedajo vodja prosto programskega projekta oz. fiduciar, razvijalec oz. beneficiar in pa seveda avtor dokumenta, ki je namenjen prenosu teh pravic.

FLA vsebuje v ta namen v poglavju §1 *Grant* odstavek §1.3, v katerem je zapisano:

»In some countries, the law may provide that the employer is deemed to be the owner of the rights on materials developed by an employee in the course of his or her employment, unless the parties have agreed otherwise. The Beneficiary is aware of these provisions, and therefore warrants, represents and guarantees that the Subject Matter is free of any of his or her employer's exclusive exploitation rights.«

S tem je beneficiar opomnjen, naj pazi, ali je z delodajalcem uredil potrebne pravice za prenos na fiduciarja.

Prenosu pravic z delodajalca na fiduciarja je prirejen tudi obrazec na začetku pogodbe:

»the owner of the exclusive licence,

_____ (Company or name)

_____ (HQ or addr.)

_____ (managing director)

- *acquired by virtue of a contract date as of _____*
- *contracting party: _____*
- *acquired as employer in the context of a work and service relationship*

– hereinafter referred to as ›Beneficiary‹ –«

To možnost je veliko razvijalcev in njihovih delodajalcev uporabilo za prenos pravic predvsem za projekta KDE in Bacula.

¹⁴⁴ KDE, KDE e.V. – Fiduciary Licensing Agreement; URL: <https://ev.kde.org/rules/fla.php> (dne: 14. 01. 2016).

Sklenemo lahko torej, da glede vprašanja delodajalcev FLA zagotavlja dovolj zanesljivosti v teoriji in praksi. Le v originalnem besedilu bi bil lahko obrazec preglednejši – več o tem pa v **poglavju 4.6**.

4.6 BERLJIVOST DOKUMENTA

FLA je uporabna samo, če jo razvijalci sprejmejo in podpišejo. Prvi pogoj za podpis pa je seveda, da jo razumejo in se z njeno vsebino strinjajo.

Kot je razvidno iz **poglavja 2.1** so večji projekti, kot sta Bacula in KDE, imeli precej dela, da so razložili vsebino FLA in implikacije podpisa.

Namen preambule je bil tudi v tem, da je pomagala potencialnim podpisnikom razumeti namene in cilje dokumenta. Sicer je tu v precejšno pomoč, vendar so avtorji podcenjevali naravno nagnjenost ciljne skupine k temu, da poskuša najti napake v vsem – tudi kadar jih najde tam, kjer jih ni.

Predvsem bi se dalo poenostaviti definicije predmeta pogodbe: *prispevka, kode in dokumentacije*. To je npr. lepše definirano v ContributorAgreements.org¹⁴⁵, kjer so definirani naslednji pojmi:

- » ›Contribution‹ means any original work of authorship (software and/or documentation) including any modifications or additions to an existing work, Submitted by You to Us, in which You own the Copyright. If You do not own the Copyright in the entire work of authorship, please contact Us at [...]. «
- » ›Copyright‹ means all rights protecting works of authorship owned or controlled by You, including copyright, moral and neighboring rights, as appropriate, for the full term of their existence including any extensions by You. «
- » ›Material‹ means the software or documentation made available by Us to third parties. When this Agreement covers more than one software project, the Material means the software or documentation to which the Contribution was Submitted. After You Submit the Contribution, it may be included in the Material. «
- » ›Submit‹ means any form of physical, electronic, or written communication sent to Us, including but not limited to electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, Us, but excluding communication that is conspicuously marked or otherwise designated in writing by You as ›Not a Contribution‹. «
- » ›Documentation‹ means any non-software portion of a Contribution. «

Posodobitev definicij v FLA, da bi bila bolj podobna tej, bi po mojem mnenju pripomogla h kakovosti dokumenta.

¹⁴⁵ ContributorAgreements.org, ContributorAgreements.org; URL: <http://contributoragreements.org/> (dne: 23. 11. 2015).

Smotrno bi bilo tudi premakniti klavzulo o časovni in krajevni neomejenosti prenosa pravic iz §3.4 neposredno v §1 *Grant*, kjer je sicer ta prenos definiran.

Razen teh si, kar zadeva vsebino, težko predstavljam spremembe, ki bi besedilo naredile lažje razumljivo za laike, hkrati pa še vedno zagotavljale isti kompleks pravic in varovalk.

Vsekakor pa se je izkazalo, da je predavanje dober način, ki bi FLA približal razvijalcem/potencialnim podpisnikom. Da je mogoče FLA predstaviti v 10 minutah¹⁴⁶ in celo v 5 minutah¹⁴⁷, sem izkusil tudi sam. Seveda je še boljši stik s posameznikom, ki mu lahko razložiš in tudi odgovoriš na vprašanja. To delata Bacula in KDE s pomočjo FSFE.

Težave s formo FLA sem izkusil aprila 2014, ko sem tudi sam podpisal FLA¹⁴⁸ s KDE. Čeprav sem takrat že več let do potankosti poznal vsebino besedila in večkrat predaval o tem, moram priznati, da sem se moral med izpolnjevanjem obrazca nekajkrat ustaviti in dobro premisliti.

Glavna težava je nastala že na drugi strani dokumenta pri izpolnjevanju osebnih podatkov. Navodila so za količino možnosti preskopa, težko je tudi ugotoviti, katere se sme v kakšnih kombinacijah prečrtati. Npr. če sem se izjavil kot podjetje oz. »the owner of the exclusive licence«, ali smem prečrtati katero od danih možnosti, kako sem prejel to licenco. Tudi ni takoj jasno, da prav tam podalineeji »acquired by virtue of a contract date as of« in pa »contracting party:« zahtevata vnos manjkajočih podatkov, saj manjka prazna črta.

Smotrno bi bilo tudi normirati obrazec za specifikacijo predmeta pogodbe – avtorskega dela, ki se prenaša/licencira. Tudi tu ima ContributorAgreements.org zanimivo rešitev, saj po tem, ko definira »Contribution« in »Submit«, se v celotnem dokumentu zgolj sklicuje na to definicijo. Tako je vsa vsebina, ki jo podpisnik takega CAA/CLA pošlje nasprotni stranki, avtomatično prenesena nanjo. Izjema je le, če pri prenosu oz. komunikaciji vsebine označi, da ne gre za »Contribution«. To je sicer lahko dvorezni meč in je odvisno od projekta, ali bi bilo tako *opt-out* rešitev bolj ali manj smotrno uporabiti kot *opt-in* rešitev v FLA. Alternativno bi bilo v veliko pomoč že, če bi bil obrazec specifičen glede na projekt in tam dopuščal samo možnosti prispevanja, ki jih projekt uporablja.

¹⁴⁶ M. Šuklje, Learn More About the FLA [video posnetek predavanja na Academy 2013]; URL: <https://conf.kde.org/en/Akademy2013/public/events/75> (dne: 17. 01. 2016).

¹⁴⁷ M. Šuklje, FLA – We Meet Again! What is FLA Good For and Why You Should Sign It [video posnetek predavanja na Academy 2015]; URL: <https://conf.kde.org/en/akademy2015/public/events/221> (dne: 17. 01. 2016).

¹⁴⁸ Bolj konkretno gre za *Fiduciary Licence Agreement (Version 1.3.4)* projekta KDE.

Kot dober primer nam služi obrazec iz §2 *Subject Matter*, kot ga je določila KDE e.V. (skupaj s FSFE) v svoji FLA 1.3.5:

»All contributions made to the source code repository maintained by KDE e.V. (at the time of this writing, *svn.kde.org* and *git.kde.org*) by the Beneficiary under the account name

on or before the date of this document.«

V pogovorih z razvijalci in organizacijami, ki uporabljajo FLA, se je izkazalo, da obstaja tudi potreba po večji zavesti o tehničnih podrobnostih, kako se pravilno licencira programska koda. Vendar to presega meje te diplomske naloge.

4.6.1 LAIKOM PRIJAZEN POVZETEK

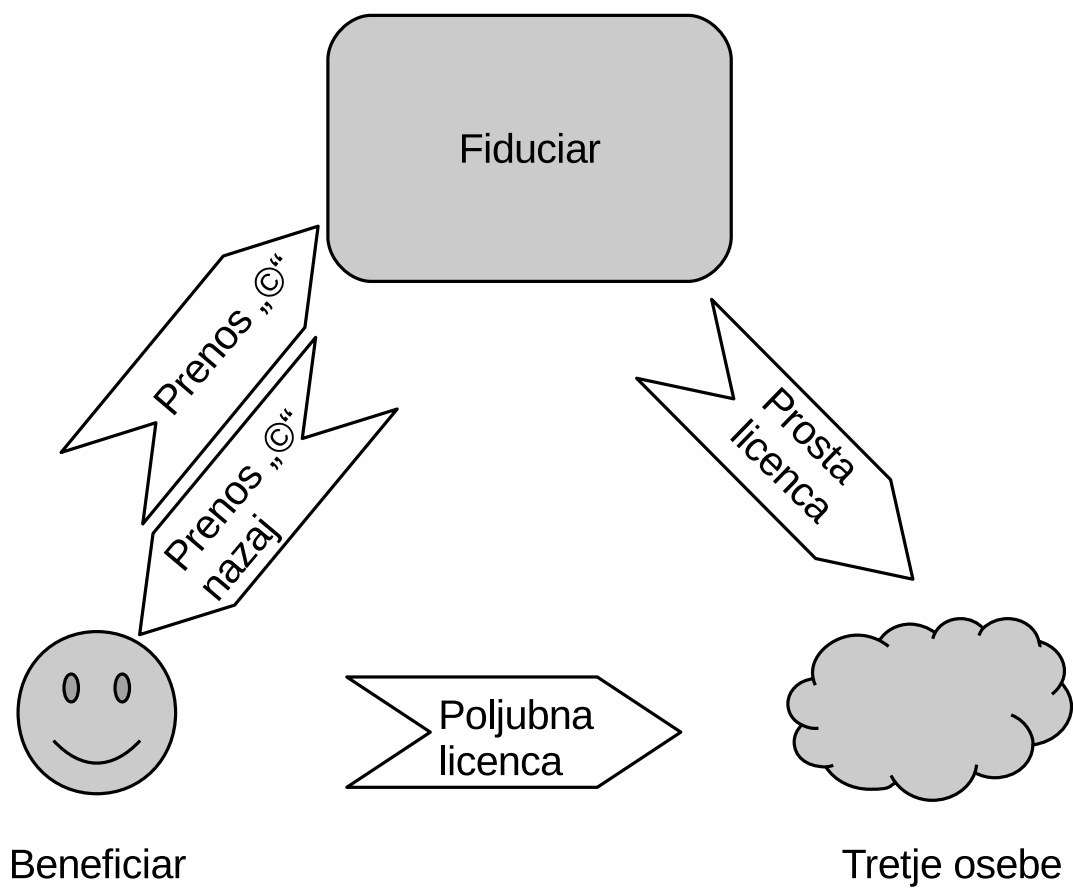
Ciljno občinstvo so razvijalci programske opreme – v povprečju ti kot (praviloma) pravni laiki ne berejo radi pravnih besedil in so *a priori* skeptični do prenosa svojih avtorskih pravic. Za to je delno razlog korporativna kultura zaprtega programja, ki razvijalcem praviloma zanika avtorstvo¹⁴⁹.

Za razvijalce bi bilo primerno poleg dokumenta ponuditi tudi vizualno pomoč, s katero je lažje razložiti razmerja v FLA.

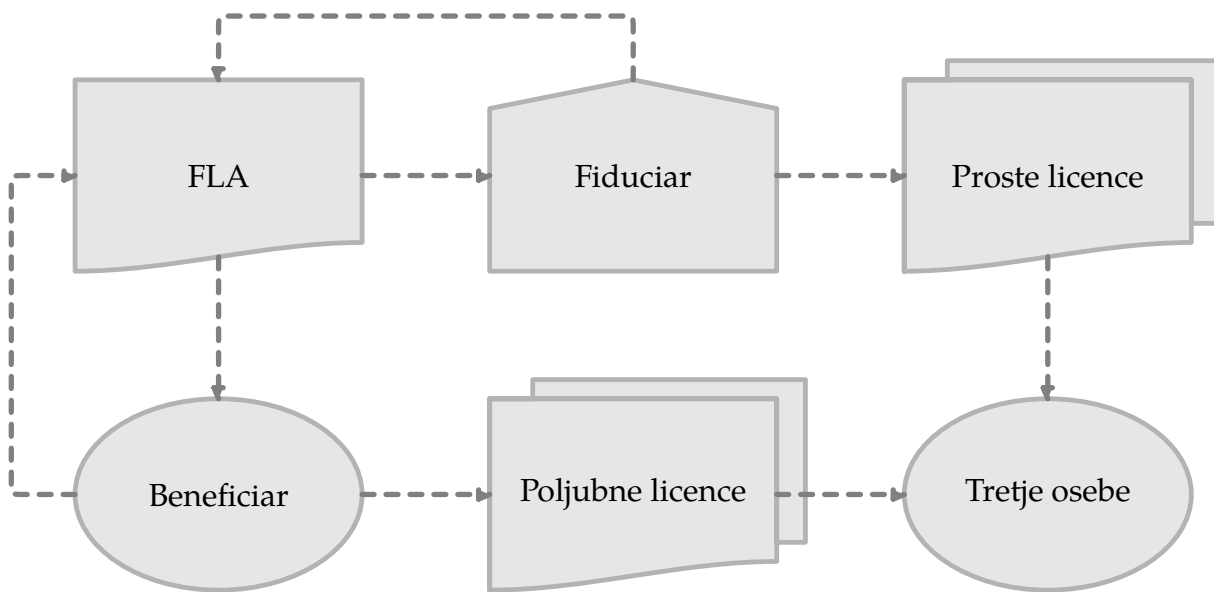
Pri predavanjih o FLA redno uporabljam infografiko kot na **sliki 4.1**, da lažje razložim razmerja med akterji v FLA. Shema je sicer zelo poenostavljena, vendar se je izkazala kot zelo uporaben pripomoček za razlago tega sicer razmeroma kompleksnega razmerja.

Naslednji mogoči način – in morda razvijalcem bližje – bi bila tudi metoda shematskega prikaza, kot na **sliki 4.2**. Osebnost se mi sicer zdi manj pregledna, vendar bi bila morda komu bolj domača, saj uporablja standardne elemente shematskega prikaza.

¹⁴⁹ Za preprost eksperiment poskusite ugotoviti, kdo vse je *avtor* (ne imetnik materialnih pravic!) programa, v katerem berete to nalogo. Če ne gre za prosti program, zelo dvomim, da boste našli vsaj eno osebno ime.



Slika 4.1 Infografika, ki jo uporabljam na predavanjih o FLA.



Slika 4.2 Shematski prikaz FLA.

5 SKLEP

Leta 2016 ima tako *Fiduciary Licence Agreement* še vedno svoje mesto v prostoprogramskem ekosistemu. Sicer se lahko marsikaj nauči iz novejših CAA/CLA, kot npr. ContributorAgreements.org, vendar je glede svoje politične pozicije še vedno unikatna. Če bi jo hoteli zamenjati za kak drug dokument, bi ta zelo težko pridobil enako zaupanje skupnosti.

Kot je prikazano v **podpoglavju 2.1**, se je FLA izkazala uporabna za male in velike projekte.

Na podlagi izkušenj teh projektov in **poglavju 4** te diplomske naloge pa predlagam, da se spremeni besedilo FLA tako, da je uporabno v še več jurisdikcijah in vključuje vhodno (neizključno) patentno licenco in ne nazadnje, da je lažje berljivo – tudi z uporabo infografik.

Po drugi strani pa se je izkazalo, da blagovne znamke niso predmet, s katerim naj se FLA ukvarja.

Kot dobro prakso zelo toplo priporočam uporabo *FLA Relicensing Policy* v projektih, ki uporabljajo FLA. Tudi predavanja in podobni načini približanja vsebine dokumenta potencialnim podpisnikom/beneficiarjem so zelo dober nasvet.

Na tem področju so še vedno mogoče zelo zanimive raziskave. Smotno bi bilo ugotoviti, ali lahko ContributorAgreements.org zagotovijo enako ravnotežje kot Fiduciary Licence Agreement. Poleg tega je veliko neraziskanih možnosti še odprtih za avtomatizacijo in drugačne poenostavitve agregiranja avtorskih pravic. V to področje spada vprašanje, ali je lahko kako Developer's Certificate of Origin celostna ali zgolj delna rešitev. Prav tako bi bil zanimiv problem za nadaljnjo predvsem interdisciplinarno raziskavo, ali se da globalno avtomatizirati CAA/CLA.

A GLOSAR

AGPL	GNU Affero General Public License
beneficiar	Avtor oz. prispevkar, ki prek FLA prenaša svoje pravice oz. licenco na fiduciarja. (ang. beneficiary)
CA	Pogodba o sodelovanju. Pogosto se uporablja kot nadpomenka za CAA in CLA. (ang. Contributor Agreement)
CAA	Pogodba o prenosu avtorskih pravic. Z njo avtor prenese (vse) svoje avtorske pravice na cesionarja. (ang. Copyright Assignment Agreement)
CLA	Licenčna pogodba med razvijalci. Z njo avtor prenese licenco na prejemnika. V kontekstu prostega programja je ideja v tem, da se vsi (so)avtorji nekega programja strinjajo, pod katero vhodno licenco prispevajo svoje delo. (ang. Copyright/Contributor Licence Agreement)
DCO	Razvijalčev certifikat o izvoru. Tehnični način, kako razvijalec zelo preprosto izjavi, da ima zadostne pravice, da svoje delo prispeva pod določeno licenco. Nekateri prostoprogramski projekti poskušajo z DCO nadomestiti CLA, saj je vzdrževanje precej lažje. (ang. Developer's Certificate of Origin)
FLA	Fiduciarna licenčna pogodba. Posebna tipska CLA/CAA, ki jo je napisala FSFE in vsebuje zelo močne zavore in kontrole v korist prispevkarjev. Predmet te diplomske naloge. (ang. Fiduciary License Agreement)

fiduciar	Subjekt, na katerega beneficiar s FLA prenaša svoje prispevke na skupnem projektu. Fiduciar je po FLA zavezan, da se projekt drži pod prostoprogramsko licenco. (ang. fiduciary)
FOSS	Kratica, ki se pogosto uporablja kot nevtralna oznaka za prosto programje in odprto kodo. (ang. Free and Open Source Software)
FRP	FLA politika za relicenciranje. Dokument, ki dopolnjuje FLA s tem, da predpisuje licence, pod katerimi sme fiduciar objaviti prispevke, ki so mu bili zaupani pod FLA, ter postopek re- ali podlicenciranja. (ang. FLA Relicensing Policy)
FSFE	Free Software Foundation Europe
GFDL	GNU Free Documentation License
Git	Git je trenutno najbolj razširjen sistem za porazdeljeno dokumentiranje sprememb in izdajanje različic datotek (ang. code versioning system). Je novejši in bolj dodelan kot Subversion (SVN) ali CVS.
GNU	GNU's Not Unix. Eden prvih in pomembnejših prostoprogramskih projektov. Njegov cilj je ustvariti povsem prostoprogramski operacijski sistem.
GPL	GNU General Public License
IRC	Eden najstarejših spletni klepetov in internetni protokol zanj. V prostoprogramskih krogih je še vedno zelo popularen za komunikacijo. Pogosto imajo prostoprogramski projekti vsak svoj kanal na IRC. (ang. Internet Relay Chat)
izhodna licenca	Z vidika opazovanega subjekta ali projekta je to licenca, s katero daje pravice do uporabe kode drugim v svojem povratnem toku. (ang. outbound licence)
izvorna koda	Programska koda, v človeku berljivi obliki. (ang. source code)

odcepljen projekt	Projekt se včasih znajde na precepu, ko se skupnost razvijalcev razdvoji. To se lahko zgodi zaradi različnih razlogov: tehničnih, osebnih ali drugih nesoglasij. Ko se en tak projekt loči od prvotnega, rečemo, da se odcepi. Tak projekt imenujemo odcepljeni projekt. (ang. fork, forked project)
odprta koda	Drugo ime za prosto programje. Zagovorniki tega termina pogosto bolj poudarjajo tehnične prednosti modela razvoja. (ang. Open Source Software)
popravek	Tehnični termin za enoto prispevka, kadar prispeva spremembo glede na prejšnje stanje. Praviloma gre pri tem za datoteko *.patch, ki jo sestavljajo vrstice kode, ki so dodane al zbrisane iz prejšnje različice kode. (ang. patch)
povratni tok	Subjekt(i) višje v dobavni verigi projekta. Npr. za Android je jedro Linux povratni tok, saj Android temelji na Linuxu. (ang. upstream)
pricepitev	Pogosto se – predvsem pri uporabi sistemov za porazdeljeno dokumentiranje sprememb in izdajanje različic datotek – iz odcepljenega projekta ali odcepljene veje razvoja koda znova pricepi v glavno vejo. (ang. merge)
prispevek	Tako se v kontekstu kolaborativnega razvoja in ustvarjanja imenuje delo, ki ga posamezni prispevkar prispeva v skupen projekt oz. skupno delo. (ang. contribution)
prispevkar	Prispevkar oz. prispevajoča stranka je tista oseba, ki v skupni projekt prispeva svoj del v celoto – skupno avtorsko delo. Odvisno od kvalitete in kvantitete njenega prispevka lahko posledično pridobi soavtorstvo na delu ali pa zgolj primakne majhen kamenček v mozaik (npr. s korekcijo majhnega a problematičnega zatipka, ki ni dovolj velik, da bi bil avtorsko delo). (ang. contributor)
projekt	Predvsem v svetu prostega programja označujemo projekt kot program, ki ima neki cilj in skupnost okoli njega, ki ga uporablja

in razvija. Ni ravno posrečen termin, saj je cilj projektov, da se zaključijo; prosto programski projekti pa so namenjeni ravno nasprotnemu – da se razvijajo in vzdržujejo naprej, da izpolnjujejo svoj cilj. Kljub tej pomanjkljivosti je ta termin globoko zasidran v tem okolju.
(ang. project)

prosto programje Prosto programje je programje, ki vsakemu legalnemu uporabniku oz. prejemniku licence zagotavlja pravice, da tako programje 1) uporablja za vsak namen; 2) prouči, kako deluje, in ga spremeni po svojih potrebah in željah; 3) deli z drugimi programje v nespremenjeni obliki, kot ga je dobil sam; ter 4) deli z drugimi svojo, spremenjeno različico programa. Zagovorniki tega termina (v nasprotju s terminom odprta koda) poudarjajo pomen prostega programja za svoboščine v sodobnem svetu.
(ang. Free Software)

razvijalec Oseba, ki razvija programje.
(ang. developer)

sprejemni tok Subjekt(i) nižje v dobavni verigi projekta. Npr. za jedro Linux je Android sprejemni tok, saj Android gradi na Linuxu.
(ang. downstream)

SVN Apache Subversion je sistem za dokumentiranje sprememb in izdajanje različic datotek. Je nadgradnja sistema CVS.

vhodna licenca Z vidika opazovanega subjekta ali projekta je to licenca, s katero dobi pravice do uporabe kode od subjektov v svojem povratnem toku.
(ang. inbound licence)

UML Poenoteni jezik modeliranja. Računalniški standard, ki poenoti modeliranje računalniških sistemov, procesov in drugega.
(ang. Unified Model Language)

zapis prispevka Prispevek oz. popravek, kot je oddan in zapisan v sistemu za dokumentiranje spremembe in izdajanje različic datotek (npr. Git). Dokumentiran je skupaj (vsaj) z imenom in e-poštnim naslovom prispevkarja (praviloma avtorja) in kratkim opisom prispevanega popravka.
(ang. commit)

B FLA, VERSION 1.2

Fiduciary Licence Agreement

(Version 1.2)

COPYRIGHT (C) 2002-2007 FSFE, E.V.,
TALSTRASSE 110, 40217 DUSSELDORF, GERMANY
EVERYONE IS PERMITTED TO COPY AND DISTRIBUTE VERBATIM COPIES
OF THIS LICENSE DOCUMENT, BUT CHANGING IT IS NOT ALLOWED.

Preamble

Access to software determines participation in a digital society. To secure equal participation in the information age, the Free Software Foundation Europe (FSFE) pursues and is dedicated to the furthering of Free Software, defined by the freedoms to use, study, modify and copy. Independent of the issue of commercial exploitation, it is proprietary, freedom-diminishing licensing that works against the interests of people and society at large, which is therefore rejected by FSFE.

The purpose of this agreement is to ensure the lasting protection of Free Software by making FSFE the fiduciary of the author's interests. It empowers FSFE – and its sister organisations – to uphold the interests of Free Software authors and protect them in court, if necessary.

FSFE is given the right to relicense the software as necessary for the long-term legal maintainability and protection of the software. The agreement also grants the author an unlimited amount of non-exclusive licences by FSFE, which allow using and distributing the program in other projects and under other licences.

The contracting parties sign the following agreement in full consciousness that by the grant of exclusive licence to the Free Software Foundation Europe e.V. and by the administration of these rights the FSFE becomes trustee of the author's interests for the benefit of Free Software.

Agreement

Between (please cross out unsuitable alternatives)

- the author

 ----- (Occupation, D.o. Birth)
 ----- (Street)
 ----- (ZIP, City, Country)
 ----- (Pseudonym, Email)

- the owner of the exclusive licence,

----- (Company or name)
 ----- (HQ or addr.)
 ----- (managing director)

- acquired by virtue of a contract date as of
- contracting party:
- acquired as employer in the context of a work and service relationship

- hereinafter referred to as "Beneficiary" –

and the

Free Software Foundation Europe, e.V.,
 Talstrasse 110
 40217 Dsseldorf
 Germany

- hereinafter referred to as "FSFE" –

the following agreement is entered into:

§ 1 Grant

(1) Subject to the provision of § 2, Beneficiary assigns to FSFE the Copyright in computer programs and other copyrightable material world-wide, or in countries where such an assignment is not possible,¹ grants an exclusive licence, including, inter alia:

1. the right to reproduce in original or modified form;

¹Countries where assignments of the copyright in a work are impossible include, but are not limited to, Germany, Austria, Slovenia and Hungary.

2. the right to redistribute in original or modified form;
3. the right of making available in data networks, in particular via the Internet, as well as by providing downloads, in original or modified form;
4. the right to authorize third parties to make derivative works of the Software, or to work on and commit changes or perform this conduct themselves.

(2) Beneficiary’s moral or personal rights remain unaffected by this Agreement.

(3) In some countries, the law may provide that the employer is deemed to be the owner of the rights on materials developed by an employee in the course of his or her employment, unless the parties have agreed otherwise. The Beneficiary is aware of these provisions, and therefore warrants, represents and guarantees that the Subject Matter is free of any of his or her employer’s exclusive exploitation rights.

§ 2 Subject Matter

The rights and licences granted in § 1 are subject to all “Software” and “Documentation.” For the purpose of this Agreement, “Software” shall mean all computer programs, copyrightable sections of computer programs or modifications of computer programs that have been developed or programmed by Beneficiary and that are specified in this Agreement below or that are listed in Exhibit A attached to this Agreement and dated and signed by the contracting parties. Likewise, “Documentation” shall refer to all manuals and documentation written by Beneficiary alongside and usually distributed with the “Software” and are similarly specified below or listed in Exhibit A:

Except in countries where such an assignment is not possible,² the rights [and licences] granted under this agreement by Beneficiary shall also include future developments, future corrections of errors or faults and other future modifications and derivative works of the software that Beneficiary obtains copyright ownership. Excluded from this provision are modifications that are not derived from the subject matter and that have to be regarded as independent and original software.

§ 3 FSFE’s Rights and Re-Transfer of Non-Exclusive Licence

(1) FSFE shall exercise the granted rights and licences in its own name. Furthermore, FSFE shall be authorized to enjoin third parties from using the software and forbid any unlawful

²Countries where assignments of the copyright in a future work are impossible include, but are not limited to, France.

or copyright infringing use of the Software, and shall be entitled to enforce all its rights in its own name in and out of court. FSFE shall also be authorized to permit third parties to exercise FSFE's rights in and out of court.

(2) FSFE grants to Beneficiary a non-exclusive, worldwide, perpetual and unrestricted licence in the Software. This right's [and licence's] scope shall encompass and include all the rights [and licences] specified in § 1. Furthermore, FSFE grants to Beneficiary additional non-exclusive, transferable license to use, reproduce, redistribute and make available the Software as needed for releases of the Software under other licences. This re-transfer shall not limit the scope of FSFE's exclusive licence in the Software and FSFE's rights pursuant to § 1.

(3) FSFE shall only exercise the granted rights and licences in accordance with the principles of Free Software as defined by the Free Software Foundations. FSFE guarantees to use the rights and licences transferred in strict accordance with the regulations imposed by Free Software licences, including, but not limited to, the GNU General Public Licence (GPL) or the GNU Lesser General Public Licence (LGPL) respectively. In the event FSFE violates the principles of Free Software, all granted rights and licences shall automatically return to the Beneficiary and the licences granted hereunder shall be terminated and expire.

(4) The transfer of the rights and licences specified in § 1 shall be unrestricted in territory and thus shall apply world-wide and be temporally unlimited.

§ 4 Miscellaneous

(1) Regarding the succession of rights in this contractual relationship, German law shall apply, unless this Agreement imposes deviating regulations. In case of the Beneficiary's death, the assignment of exclusive rights shall continue with the heirs. In case of more than one heir, all heirs have to exercise their rights through a common authorized person.

(2) Place of jurisdiction for all legal conflicts arising out of or in connection with this Agreement is Munich, Germany.

-----, -----

(Beneficiary)

(FSFE)

C LITERATURA

MONOGRAFIJE

Damjan, Matija: PROSTA LICENCA KOT PRAVNI TEMELJ ODPRTE VSEBINE – DOKTORSKA DIZERTACIJA, Univerza v Ljubljani, Pravna fakulteta, Ljubljana 2007.

ZBORNIKI

Bain, Malcolm: Patents and FOSS, v: FREE AND OPEN SOURCE SOFTWARE – POLICY, LAW, AND PRACTICE (ur. Shemtov, Noam in Walden, Ian), Oxford university press, Oxford 2013, str. 141–180.

van den Brande, Ywein: Belgium, v: THE INTERNATIONAL FREE AND OPEN SOURCE SOFTWARE LAW BOOK (ur. van den Brande, Ywein; Coughlan, Shane Martin; Jaeger, Till), Open Source Press GmbH, München 2014, str. 31–52.

Gardler, Ross: Open Source and Governance, v: FREE AND OPEN SOURCE SOFTWARE – POLICY, LAW, AND PRACTICE (ur. Shemtov, Noam in Walden, Ian), Oxford university press, Oxford 2013, str. 37–68.

Shemtov, Noam: Trade Marks and FOSS, v: FREE AND OPEN SOURCE SOFTWARE – POLICY, LAW, AND PRACTICE (ur. Shemtov, Noam in Walden, Ian), Oxford university press, Oxford 2013, str. 109–140.

Walden, Ian: Open Source as Philosophy, Methodology, and Commerce: Using Law with Attitude, v: FREE AND OPEN SOURCE SOFTWARE – POLICY, LAW, AND PRACTICE (ur. Shemtov, Noam in Walden, Ian), Oxford university press, Oxford 2013, str. 1–36.

ČLANKI

- van den Brande, Ywein: The Fiduciary Licence Agreement: Appointing legal guardians for Free Software Projects, v: *INTERNATIONAL FREE AND OPEN SOURCE LAW REVIEW*, 1 (2009), 1, str. 9–14.
- Damjan, Matija: GNU GPL – licenca ali pogodba?, v: *PRAVNIK*, 61 (2006), 3–4, str. 329–351.
- Damjan, Matija: Izšla je 3. verzija licence GPL, v: *PRAVNA PRAKSA*, 26 (2007), 37, str. II–VIII.
- Damjan, Matija: Problemi pravnega varstva računalniških programov, v: *PODJETJE IN DELO*, XXXIV (2008), str. 471–502.
- Engelhardt, Tim: Drafting Options for Contributor Agreements for Free and Open Source Software: Assignment, (Non)Exclusive Licence and Legal Consequences. A Comparative Analysis of German and US Law, v: *SCRIPTED*, 10 (2013), 2, str. 148–176.
- Guadamuz, Andres in Rens, Andrew: Comparative Analysis of copyright assignment and licence formalities for Open Source Contributor Agreements, v: *SCRIPTED*, 10 (2013), 2, str. 207–230.
- Jakob, Sylvia F.: A Qualitative Study on the Adoption of Copyright Assignment Agreements (CAA) and Copyright License Agreements (CLA) within Selected FOSS Projects, v: *JIPITEC*, 5 (2014), 2, str. 105–115.
- Maracke, Catharina: Editorial: Copyright Management for Open Collaborative Projects – Inbound Licensing Models for Open Innovation, v: *SCRIPTED*, 10 (2013), 2, str. 140–148.
- Metzger, Axel: FLA – Die Treuhänderische Lizenzvereinbarung der FSF: Zu treuen Händen, v: *LINUX MAGAZIN* (2003), 5, str. 75–77.
- Metzger, Axel: Internationalisation of FOSS Contributory Copyright Assignments and Licenses: Jurisdiction-Specific or »Unported«?, v: *SCRIPTED*, 10 (2013), 2, str. 177–206.

PRAVNI VIRI

BERNSKA KONVENCIJA ZA VARSTVO KNJIŽEVNIH IN UMETNIŠKIH DEL (MSBKVK), Ur. l. SFRJ, MP, št. 14/75, 4/86; Ur. l. RS št. 25/92; Ur. l. RS. MP št. 9/92 in 3/2007.

COPYRIGHT ACT, 1957 (Copyright Act, 1957), The Official Gazette [of India], št. 14/1957, 1999/49, 2012/27.

OBLIGACIJSKI ZAKONIK (OZ), Ur. l. RS, št. 97/2007.

STVARNOPRAVNI ZAKONIK (SPZ), Ur. l. RS, št. 87/2002, 91/13.

ZAKON O AVTORSKI IN SORODNIH PRAVICAH (ZASP), Ur. l. RS, št. 16/2007, 68/08, 110/13 in 56/15.

ZAKON O INDUSTRIJSKI LASTNINI (ZIL-1), Ur. l. RS, št. 51/2006, 100/13.

SPLETNE STRANI

Arduino; ARDUINO, URL: <https://www.arduino.cc> (dne: 22. 01. 2016)

Außendorf, Maik; WHAT ABOUT COPYRIGHT ISSUES BETWEEN BACULA AND BAREOS?, URL: https://www.bareos.org/en/faq/copyright_bacula_bareos.html (dne: 11. 01. 2016)

Bacula.org; BACULA – OPEN SOURCE BACKUP, ENTERPRISE READY, NETWORK BACKUP TOOL FOR LINUX, UNIX, MAC, AND WINDOWS, URL: <http://bacula.org/> (dne: 09. 01. 2016)

Bacula.org; BACULA – SOURCEFORGE PROJECT, URL: <http://sourceforge.net/p/bacula/> (dne: 10. 01. 2016)

Bacula.org; COPYRIGHT ASSIGNMENT AGREEMENT (VERSION 8.0) [KOT ARHIVIRANO DNE 17. 3. 2015 NA INTERNET ARCHIVE], URL: <https://web.archive.org/web/20150317080150/http://www.bacula.org/downloads/CAA-bacula.en.pdf> (dne: 12. 01. 2016)

Bacula.org; COPYRIGHT ASSIGNMENT AGREEMENT (VERSION 9.0), URL: <http://www.bacula.org/downloads/CAA-bacula.en.pdf> (dne: 10. 01. 2016)

Bacula.org; FIDUCIARY LICENSE AGREEMENT (VERSION 3.0) [KOT ARHIVIRANO DNE 10. 6. 2013 NA INTERNET ARCHIVE], URL: <https://web.archive.org/web/20130510151653/http://bacula.org/en/FLA-bacula.en.pdf> (dne: 12. 01. 2016)

Bacula.org; FIDUCIARY LICENSE AGREEMENT (VERSION 5.0) [KOT ARHIVIRANO DNE 7. 2. 2014 NA INTERNET ARCHIVE], URL: <https://web.archive.org/web/20140207093545/http://www.bacula.org/en/FLA-bacula.en.pdf> (dne: 10. 01. 2016)

Bacula.org; FIDUCIARY LICENSE AGREEMENT (VERSION 7.0) [KOT ARHIVIRANO DNE 18. 12. 2014 NA INTERNET ARCHIVE], URL: <https://web.archive.org/web/20141218084223/http://www.bacula.org/downloads/FLA-bacula.en.pdf> (dne: 12. 01. 2016)

Bacula.org; LICENSES – BACULA, URL: <http://blog.bacula.org/general/bacula-copyright-trademark-and-licenses/> (dne: 09. 01. 2016)

Bacula Systems; BACULA SYSTEMS – BACKUP, RESTORE AND DISASTER RECOVERY, URL: <http://www.baculasystems.com/> (dne: 09. 01. 2016)

Bottomley, James E. J.; THE DCO, PATENTS AND OPENSTACK, URL: <http://blog.hansenpartnership.com/the-dco-patents-and-openstack/> (dne: 14. 01. 2016)

Bottomley, James E. J.; A MODEST PROPOSAL ON THE DCO, URL: <http://blog.hansenpartnership.com/a-modest-proposal-on-the-dco/> (dne: 14. 01. 2016)

Chestek, Pamela S.; Malujda, Rafał; Ghalamkarizadeh, Anthonia; Davis, Weston; FOSSMARKS, URL: <http://fossmarks.org/> (dne: 21. 01. 2016)

ContributorAgreements.org Core Team; CONTRIBUTORAGREEMENTS.ORG, URL: <http://contributoragreements.org/> (dne: 23. 11. 2015)

ContributorAgreements.org Core Team; CONTRIBUTORAGREEMENTS.ORG FAQ, URL: <http://contributoragreements.org/faq> (dne: 23. 11. 2015)

Corbet, Jonathan; COMMUNITY CONTRIBUTIONS AND COPYRIGHT ASSIGNMENT, URL: <https://lwn.net/Articles/359013/> (dne: 23. 12. 2015)

Corbet, Jonathan; Kroah-Hatman, Greg; McPherson, Amanda; LINUX KERNEL DEVELOPMENT – HOW FAST IT IS GOING, WHO IS DOING IT, WHAT THEY ARE DOING, AND WHO IS SPONSORING IT, URL: http://www.linuxfoundation.org/docs/lf_linux_kernel_development_2010.pdf (dne: 04. 02. 2016)

Debian; DEBIAN SOCIAL CONTRACT, URL: http://www.debian.org/social_contract (dne: 28. 12. 2015)

Definition of Free Cultural Works; OSHW, URL: <http://freedomdefined.org/OSHW> (dne: 22. 01. 2016)

Fontana, Richard; THE TROUBLE WITH HARMONY: PART 1, URL: <https://opensource.com/law/11/7/trouble-harmony-part-1> (dne: 22. 12. 2015)

Fontana, Richard; THE TROUBLE WITH HARMONY: PART 2, URL: <https://opensource.com/law/11/7/trouble-harmony-part-2> (dne: 22. 12. 2015)

Free Knowledge Institute; FREE KNOWLEDGE INSTITUTE, URL: <http://freeknowledge.eu> (dne: 30. 12. 2015)

Free Knowledge Institute; FKI: STRATEGY/TRACK RECORD, URL: http://wiki.freeknowledge.eu/index.php/FKI:Strategy/Track_record (dne: 30. 12. 2015)

Free Knowledge Institute; THE SELF PROJECT, URL: <http://freeknowledge.eu/projects/self> (dne: 30. 12. 2015)

Free Software Foundation Europe; BACULA AGREEMENT – FSFE LEGAL, URL: <https://fsfe.org/activities/ftf/bacula-agreement.en.html> (dne: 10. 01. 2016)

Free Software Foundation Europe; [FSFE PR][EN] FSFE BECOMES THE LEGAL GUARDIAN OF THE BACULA PROJECT, URL: <http://mail.fsfeurope.org/pipermail/press-release/2006q4/000161.html> (dne: 10. 01. 2016)

Free Software Foundation Europe; [FSFE PR][EN] FSFE BECOMES THE LEGAL GUARDIAN OF THE OPENSWARM PROJECT, URL: <http://mail.fsfeurope.org/pipermail/press-release/2007q1/000165.html> (dne: 01. 01. 2016)

Free Software Foundation Europe; FIDUCIARY LICENSE AGREEMENT (FLA), URL: <https://fsfe.org/activities/ftf/fla.html> (dne: 15. 12. 2014)

Free Software Foundation Europe; FIDUCIARY LICENCE AGREEMENT LIST – BACULA, URL: <https://fsfe.org/activities/ftf/fla-list-bacula.en.html> (dne: 10. 01. 2016)

Free Software Foundation Europe; FIDUCIARY LICENSE AGREEMENT (VERSION 1.2) FOR AEONIC, URL: http://aeonic.sourceforge.net/get/Aeonic_FLA_en_v1.2.pdf (dne: 01. 01. 2016)

Free Software Foundation Europe; FIDUCIARY LICENSE AGREEMENT (VERSION 1.3.5) [ZA KDE], URL: <https://ev.kde.org/resources/FLA.pdf> (dne: 15. 01. 2016)

Free Software Foundation Europe; NEW ROLE FOR THE FSFE IN THE BACULA PROJECT, URL: <https://fsfe.org/news/2015/news-20150817-01.en.html> (dne: 10. 01. 2016)

Free Software Foundation Europe; FIDUCIARY PROGRAMME, URL: <https://fsfe.org/activities/ftf/fiduciary.html> (dne: 09. 01. 2016)

Free Software Foundation Europe; FSF EUROPE RELEASES FIDUCIARY LICENCE AGREEMENT V1.0, URL: <http://mail.fsfeurope.org/pipermail/press-release/2003q1/000049.html> (dne: 28. 12. 2015)

Free Software Foundation Europe; PROJECTS – SELF: SCIENCE, EDUCATION AND LEARNING IN FREEDOM, URL: <https://fsfe.org/campaigns/self/self.en.html> (dne: 27. 12. 2015)

Free Software Foundation Europe; SELF LEGAL POLICY, WORKPACKAGE 4, URL: http://freeknowledge.eu/sites/freeknowledge.eu/files/D3_WP4-v1.0.pdf (dne: 30. 12. 2015)

Free Software Foundation Europe; STATEMENT ON CHANGED RELATIONS BETWEEN THE FSFE AND KERN SIBBALD, URL: <https://fsfe.org/news/2015/news-20150414-01.en.html> (dne: 10. 01. 2016)

Free Software Foundation Europe; WHAT IS FREE SOFTWARE?, URL: <https://fsfe.org/about/basics/freesoftware.html> (dne: 27. 12. 2015)

Greve, Georg C. F.; WHAT MAKES A FREE SOFTWARE COMPANY?, URL: <https://fsfe.org/freesoftware/enterprise/freesoftwarecompany.en.html> (dne: 28. 12. 2015)

Hatzis, Anastasios; Kumar, Animesh; Spealman, Calvin; Jackson, Hugh; Rooker, Kenneth; AEONIC – MODEL DRIVEN SERVER APPS, URL: <http://sourceforge.net/projects/aeonic/> (dne: 01. 01. 2016)

Hatzis, Anastasios; Kumar, Animesh; Spealman, Calvin; Jackson, Hugh; Rooker, Kenneth; AEONIC [PRVI COMMIT], URL: <http://sourceforge.net/p/aeonic/code/1/> (dne: 01. 01. 2016)

Institute for Legal Questions on Free and Open Source Software; LICENCE CENTER, URL: <http://www.ifross.org/en/license-center> (dne: 27. 12. 2015)

Kempf, Jean-Baptiste; HOW TO PROPERLY(?) RELICENSE A LARGE OPEN SOURCE PROJECT - PART 1, URL: <http://www.jbkempf.com/blog/post/2012/How-to-properly-relicense-a-large-open-source-project> (dne: 01. 09. 2015)

Kempf, Jean-Baptiste; HOW TO PROPERLY(?) RELICENSE A LARGE OPEN SOURCE PROJECT - PART 2, URL: <http://www.jbkempf.com/blog/post/2012/How-to-properly-relicense-a-large-open-source-project-part-2> (dne: 01. 09. 2015)

Khun, Bradley M.; PROJECT HARMONY (AND »NEXT GENERATION CONTRIBUTOR AGREEMENTS«) CONSIDERED HARMFUL, URL: <http://ebb.org/bkuhn/blog/2011/07/07/harmony-harmful.html> (dne: 22. 12. 2015)

Khun, Bradley M.; WHY YOUR PROJECT DOESN'T NEED A CONTRIBUTOR LICENSING AGREEMENT, URL: <http://ebb.org/bkuhn/blog/2014/06/09/do-not-need-cla.html> (dne: 22. 12. 2015)

Linux Foundation; DEVELOPER'S CERTIFICATE OF ORIGIN 1.1, URL: <http://developercertificate.org/> (dne: 22. 12. 2015)

KDE; KDE – EXPERIENCE FREEDOM!, URL: <https://kde.org> (dne: 14. 01. 2016)

KDE; KDE E.V. – FIDUCIARY LICENSING AGREEMENT, URL: <https://ev.kde.org/rules/fla.php> (dne: 14. 01. 2016)

KDE; KDE FREE QT FOUNDATION, URL: <https://www.kde.org/community/whatiskde/kdefreeqtfoundation.php> (dne: 16. 01. 2016)

KDE e.V.; FLA RELICENSING POLICY [ZA KDE], URL: <https://ev.kde.org/resources/FRP.pdf> (dne: 14. 01. 2016)

KDE e.V.; KDE E.V., URL: <https://ev.kde.org> (dne: 14. 01. 2016)

KDE e.V.; KDE E.V. – FSFE WELCOMES KDE'S ADOPTION OF THE FIDUCIARY LICENCE AGREEMENT (FLA), URL: <https://ev.kde.org/announcements/2008-08-22-fsfe-welcomes-fla.php> (dne: 14. 01. 2016)

Meeks, Michael; SOME THOUGHTS ON COPYRIGHT ASSIGNMENTS, URL: <https://people.gnome.org/~michael/blog/copyright-assignment.html> (dne: 25. 12. 2015)

Metzger, Axel; Greve, Georg C. F.; Coughlan, Shane Martin; FIDUCIARY LICENSE AGREEMENT, VERSION 1.2, URL: <https://fsfe.org/activities/ftf/FLA.en.pdf> (dne: 15. 12. 2014)

Metzger, Axel; FSF EUROPE UND IFROSS VERBESSERN RECHTSSCHUTZ FÜR FREIE SOFTWARE: FIDUCIARY LICENSE AGREEMENT VERÖFFENTLICHT, URL: <http://www.ifross.org/artikel/fsf-europe-und-ifross-verbessern-rechtsschutz-fuer-freie-software-fiduciary-license-agreemen> (dne: 28. 12. 2015)

Michlmayr, Martin; OPEN SOURCE CONTRIBUTOR AGREEMENTS: SOME EXAMPLES, URL: <https://fossbazaar.org/content/open-source-contributor-agreements-some-examples/> (dne: 28. 12. 2015)

Mozilla Foundation; MOZILLA – SKUPAJ GRADIMO BOLJŠI INTERNET, URL: <https://www.mozilla.org/sl> (dne: 22. 1. 2016)

Mozilla Foundation; MOZILLA FOUNDATION COMMITTEE'S AGREEMENT v2.1, URL: <https://static.mozilla.com/foundation/documents/commit-access/committees-agreement.pdf> (dne: 28. 12. 2015)

Mozilla Foundation; MOZILLA TRADEMARK POLICY, URL: <https://www.mozilla.org/en-US/foundation/trademarks/policy/> (dne: 21. 1. 2016)

Open Invention Network; DEFENSIVE PUBLICATIONS, URL: <http://www.defensivepublications.org/> (dne: 27. 1. 2016)

Open Invention Network; OPEN INVENTION NETWORK – HOME, URL: <https://www.openinventionnetwork.com/> (dne: 27. 1. 2016)

Red Hat; FEDORA PROJECT CONTRIBUTOR AGREEMENT, URL: https://fedoraproject.org/wiki/Legal:Fedora_Project_Contributor_Agreement (dne: 28. 12. 2015)

SELF Consortium; SELF PLATFORM, URL: <https://savannah.nongnu.org/projects/self-platform> (dne: 28. 12. 2015)

SELF Consortium; SELF PROJECT – SHARING KNOWLEDGE ABOUT FREE SOFTWARE, URL: <http://www.selfproject.eu> (dne: 28. 12. 2015)

Sibald, Kern; [BACULA-ANNOUNCE] BACULA 1.17, URL: <http://sourceforge.net/p/bacula/mailman/message/587465/> (dne: 10. 01. 2016)

Sibald, Kern; BACULA STATUS REPORT (25. NOVEMBER 2013), URL: <http://blog.bacula.org/status1/> (dne: 11. 01. 2016)

Sibald, Kern; BACULA STATUS REPORT 27 AUGUST 2015, URL: <http://blog.bacula.org/bacula-status-report-27-august-2015/> (dne: 10. 01. 2016)

Sibald, Kern; WHY FORKING IS BAD, URL: <http://blog.bacula.org/why-forking-is-bad/> (dne: 11. 01. 2016)

Sibald, Kern; THE HIDDEN COST OF CREATING OPEN SOURCE CODE, URL: <http://blog.bacula.org/110/> (dne: 11. 01. 2016)

Šuklje, Matija; FIDUCIARY LICENSE AGREEMENT – LESSONS LEARNED [VIDEO POSNETEK PREDAVANJA NA FOSDEM 2014], URL: <https://archive.fosdem.org/2014/schedule/event/fla/> (dne: 17. 01. 2016)

Šuklje, Matija; FIDUCIARY LICENSE AGREEMENT – KAJ TO JE IN KAJ SMO SE NAUČILI [VIDEO POSNETEK PREDAVANJA V KIBERPIPI 2014], URL: http://video.kiberpipa.org/pot_suklje_fl/ (dne: 17. 01. 2016)

Šuklje, Matija; LEARN MORE ABOUT THE FLA [VIDEO POSNETEK PREDAVANJA NA AKADEMY 2013], URL: <https://conf.kde.org/en/Akademy2013/public/events/75> (dne: 17. 01. 2016)

Šuklje, Matija; FLA – WE MEET AGAIN! WHAT IS FLA GOOD FOR AND WHY YOU SHOULD SIGN IT [VIDEO POSNETEK PREDAVANJA NA AKADEMY 2015], URL: <https://conf.kde.org/en/akademy2015/public/events/221> (dne: 17. 01. 2016)

VideoLAN; VIDEO LAN – URADNA STRAN VLC PREDVAJALNIKA, ODPRTO KODNEGA VIDEO OGRODJA!, URL: <https://www.videolan.org/vlc/> (dne: 01. 09. 2015)

Valentine, Nathan; THE BACULA PHILOSOPHY, URL: <http://www.linuxdevcenter.com/pub/a/linux/2005/04/07/bacula.html> (dne: 10. 01. 2016)

Willis, Nathan; A TRADEMARK BATTLE IN THE ARDUINO COMMUNITY, URL: <https://lwn.net/Articles/637755/> (dne: 21. 01. 2016)

Willis, Nathan; RELICENSING VLC FROM GPL TO LGPL, URL: <https://lwn.net/Articles/525718/> (dne: 01. 09. 2015)