

Зашто софтвер треба да буде слободан

од [Ричарда Сталмана](#)

Увод

Постојање софтвера неизбежно доводи до питања како би требало доносити одлуке о његовој употреби. На пример, претпоставимо да особа која има примерак програма сретне другу која би желела свој примерак. Да ли им је дозвољено да умножавају програм? Ко би требало да то дозволи? Ти појединци? Или друга страна, звана „власник“?

Градитељи софтвера обично разматрају ова питања под претпоставком да је критеријум одговора увећање њиховог профита. Политичка моћ бизниса водила је до усвајања и овог критеријума и одговора који су понудили градитељи: програми имају власнике, обично фирме повезане са њиховом изградњом.

Ја бих разматрао исто питање са друге тачке гледишта: просперитета и слободе јавности у општем смислу.

Постојећи закон не може да пружи тај одговор — закон се мора прилагодити етици, а не обратно. Ни досадашња пракса не даје одговора, премда може да наговести могуће одговоре. Једини начин за процену је да се види коме ће, зашто и колико помоћи а коме одмоћи истицање власника софтвера. Другим речима, требало би да покренемо анализу трошкова и користи за рачун друштва као целине, узимајући у обзир личне слободе и материјална добра.

У овом есеју ћу описати ефекте власништва и показати да су они штетни. Мој закључак је да је дужност програмера да подстакну друге да деле, расподељују, проучавају и побољшавају софтвер који пишемо: другим речима, да пишу [„слободни“ софтвер](#).⁽¹⁾

Како власници правдају своју моћ

Они којима користи постојећи систем у ком су програми својина нуде два аргумента у прилог њиховим тврдњама да поседују програме: емоционални и економски.

Емоционални аргумент је овакав: „Ја улажем своје срце и душу у овај програм. Он потиче од мене, он је мој!“

Овај аргумент не заслужује озбиљно побијање. Осећање повезаности програмери могу гајити кад год им то одговара — оно је неизбежно. Узмимо, на пример, како углавном ти исти програмери добровољно преписују сва права на велику фирму зарад плате. Емоционална везаност мистериозно нестаје. Насупрот томе, узмимо велике уметнике и уметнице Средњег века, који се чак нису ни потписивали на своја дела. Њима име конкретног уметника није било важно. Оно што им је било важно је обављање посла — и намена којој ће послужити. Ово гледиште је владало стотинама година.

Економски аргумент је следећи: „Хоћу да се обогатим (обично нетачно описано као ‘зарађивање за живот’), и ако ми не дозволите да се обогатим од програмирања, ја нећу програмирати, а онда

вам неће остати ниједан програм!“ Ова претња се обично изриче у облику мудрог пријатељског савета.

Касније ћу објаснити зашто је ова претња блеф. Али прво хоћу да истакнем скривену претпоставку која је видљивија у другачијој формулацији овог аргумента.

Ова формулација почиње поређењем друштвене корисности власничких програма са корисношћу непостојања програма, и закључује да је изградња власничких програма у целини корисна, и као такву је треба подстицати. Овде је заблуда у поређењу само два исхода — власнички софтвер против непостојања софтвера — и претпоставке да не постоје друга решења.

Уз систем софтверских ауторских права, изградња софтвера је обично повезана са постојањем власника који контролише употребу тог софтвера. Све док ова веза постоји, често смо суочени са избором између власничког и никаквог софтвера. Међутим, ова веза није урођена или неизбежна; она је последица избора социјално-правне стратегије који преиспитујемо: избора постојања власништва. Формулисање избора као избора између власничког и никаквог софтвера представља наметање одговора.

Аргумент против власништва

Питање које разматрамо је: „Да ли би изградња софтвера требало да буде повезана са постојањем власника који би ограничили његову употребу?“

Да бисмо ово одлучили, морамо да оценимо учинак који свака од те две активности *посебно* чини на друштво: учинак изградње софтвера (независно од одредби расподеле), и учинак забране његове употребе (претпостављајући да је софтвер изграђен). Ако је једна од ове две активности корисна а друга штетна, било би боље да занемаримо њихову повезаност и обављамо само ону корисну.

Другим речима, ако је ограничавање расподеле програма који је већ изграђен штетно по друштво у целини, тада ће градитељ софтвера који је етичан одбацити ту могућност.

Да бисмо одредили учинак ограничавања дељења, треба да упоредимо друштвену вредност ограниченог (нпр. власничког) програма са вредношћу истог програма који је доступан свима. Дакле, поредимо два могућа света.

Ова анализа узима у обзир и једноставан контрааргумент који се понекад износи, а то је да је „корист давања примерка програма ближњем поништена штетом учињеном аутору“. Овај контрааргумент претпоставља да су штета и корист једнаки по јачини. Анализа укључује поређење те две јачине и показује да је корист много већа.

Да бисмо објаснили овај аргумент, применимо га у другој области: изградњи путева.

Било би могуће финансирати изградњу свих путева уз помоћ друмарина. То би изискивало постојање наплатних рампи на свакој раскрсници. Тај систем би пружио велики подстрек за побољшање путева. Такође би имао врлину да би сви корисници било ког пута морали да за њега плате. Међутим, наплатна станица је вештачка препрека несметаној возњи, јер није последица начина функционисања путева или кола.

Поредећи слободне путеве и путеве са друмаринама по њиховој корисности, налазимо да су (не узимајући у обзир остале разлике) путеви са друмаринама јефтинији за изградњу, јефтинији за коришћење, сигурнији и ефикаснији.⁽²⁾ У сиромашној земљи, друмарине могу да одагнају становништво са путева. Путеви без наплатних станица на тај начин нуде више користи друштву за мање новца; они су друштвено прихватљивији. Дакле, друштво би требало да изабере да финансира путеве на други начин, не преко наплатних станица. Употреба једном изграђеног пута би требало да буде слободна.

Када заговорници наплатних станица предлажу исте *само* као начин за повећање средстава, они извршу постојећи избор. Киосци за наплату повећавају средства, али на одређен начин и деградирају путеве. Пут са друмарином није добар као слободни пут. Обезбеђивање технички савршенијих путева може да не буде побољшање уколико подразумева и замену слободних путева путевима са друмарином.

Наравно, изградња слободног пута кошта, а то јавност мора некако да плати. Међутим, ово не значи неизбежност наплатних станица. Ми који морамо да платимо у сваком случају, више ћемо добити за наш новац куповином слободног пута.

Не тврдим да је пут са друмарином гори него непостојање пута. То би било тачно ако би друмарина била толико висока да скоро нико не би користио пут — али ово је мало вероватна политика сакупљача друмарине. Међутим, све док наплатне станице узрокују значајне губитке и неугодности, боље је повећавати средства на погоднији начин.

Како бих применио исте аргументе на изградњу софтвера, сада ћу показати да постојање „наплатних станица“ за корисне софтверске програме скупо кошта друштво: програми су скупљи за изградњу, скупљи за расподелу, и мање задовољавајући и ефикасни. Из тога следи да би изградња програма требало да буде подстицана на други начин. А онда ћу прећи на објашњавање других начина подстицања и (до неопходне границе) финансирања изградње софтвера.

Штета учињена ометањем софтвера

Претпоставимо, за тренутак, да се програм гради, и да су све уплате неопходне за његову изградњу завршене; тада друштво мора да изабере или да га претвори у власнички или да дозволи слободно дељење и употребу. Претпоставимо да су постојање програма и његова доступност пожељни.⁽³⁾

Ограничења расподеле и измене програма не могу да користе његовој употреби. Она могу само да сметају. Дакле, учинак може бити само негативан. Али колико? И на који начин?

Три различита нивоа материјалне штете су проузрокована таквим ометањем:

- Мање људи користи програм.
- Нико од корисника не може да прилагоди или поправи програм.
- Други градитељи не могу да уче из програма, или да заснују ново дело на њему.

Сваки ниво материјалне штете прати некакав облик психолошке штете. Ово се односи на учинак који људске одлуке имају на њихова даља осећања, ставове и склоности. Ове промене начина

мишљења људи ће даље имати утицај на њихове односе са њиховим ближњима, и могу да узрокују материјалне последице.

Три нивоа материјалне штете уништавају део вредности коју би програм могао да допринесе, али не могу је свести на нулу. Ако униште скоро сву вредност програма, онда писање програма штети друштву махом због труда уложеног у писање програма. А програм који је исплатив за продају мора да обезбеди директну збирну материјану корист.

Међутим, ако узмемо у обзир пратећу психолошку штету, онда не постоји граница штете коју изградња власничког софтвера може да учини.

Ометање употребе програма

Први ниво штете спречава једноставну употребу програма. Примерак програма кошта приближно нули (и можете платити ову цену обављајући посао сами), тако да би на слободном тржишту имао цену приближно једнаку нули. Наплата лиценце представља значајно обесхрабривање за коришћење програма. Ако је широко употребљив програм власнички, далеко мање људи ће га користити.

Лако је показати да је укупан допринос програма друштву умањен везивањем власника за њега. Сваки потенцијални корисник програма, суочен са потребом да плати за употребу, може да одлучи да плати, или може да игнорише употребу програма. Ако одлучи да плати, то представља пренос добара нулте вредности између две стране. Али сваки пут када неко изабере да игнорише употребу програма, то штети тој особи без икакве користи за остале. Збир негативних бројева и нула је увек негативан.

Али ово не умањује количину рада која је потребна за *изградњу* програма. Као резултат, ефикасност целог поступка посматрана кроз испоручено задовољење потреба корисника по часу рада је умањена.

Ово илуструје кључну разлику између примерака програма и аутомобила, столица или сендвича. Ван научне фантастике, не постоји машина за умножавање материјалних предмета. Али програми се могу лако умножавати; свако може да направи жељени број примерака уз врло мало труда. Ово не важи за материјалне предмете јер је материја очувана: сваки нови примерак мора да буде изграђен од сировина на исти начин како је изграђен и први.

Обесхрабривање употребе материјалних ствари има смисла, јер што мање се ствари купује то је мање сировина и рада потребно за њихово стварање. Тачно је да обично постоји почетна цена — цена изградње — која је раздељена на процес производње. Али све док је крајња цена производње значајна, додавање цене изградње не ствара квалитативну разлику. И не захтева ограничавање слободе обичних корисника.

Међутим, наметање цене за нешто што би иначе било бесплатно јесте квалитативна измена. Централизовано наметнута цена за расподелу софтвера постаје моћан извор обесхрабривања.

Штавише, централизована производња какву данас познајемо је неефикасна чак и по испоруци примерака софтвера. Овај систем укључује паковање физичких дискета или трака у сувишне пакете, испоручивање великих количина пакета широм света, и њихово складиштење за продају.

Ова цена је представљена као трошак пословања; у ствари, она је део расипања проузрокованог постојањем власништва.

Штета учињена друштвеној кохезији

Рецимо да ви и ваш ближњи сматрате покретање одређеног програма корисним. Из етичке бриге за вашег ближњег, требало би да осећате даје исправан поредак ствари омогућавање да га обојица користите. Предлог да се употреба дозволи само једном од вас а забрани другоме представља поделу. Ни ви ни ваш ближњи не треба да га сматрате прихватљивим.

Потписивање уобичајеног софтверског уговора о лиценцирању представља издају вашег ближњег: „Обећавам да забраним ближњем да употребљава овај програм, како бих имао свој примерак.“ Људи који тако бирају осећају унутрашњи психолошки притисак да се оправдају смањивањем важности помагања ближњем, па стога испашта дух друштва. Ова психолошка штета је повезана са материјалном штетом обесхрабрења употребе програма.

Многи корисници несвесно схватају да је одбијање дељења погрешно, па одлучују да игноришу лиценце и законе расподељујући програме упркос свему, али често због тога осећају кривицу. Они схватају да морају да крше законе како би били добри према ближњима, али и даље схватају да се закони морају поштовати и зато закључују да је погрешно или срамно чинити добро ближњима. Ово је и психолошки штетно, али се то може избећи ставом да те лиценце и закони немају моралну превагу.

Програмери често трпе психолошке муке знајући да великом броју корисника неће бити дозвољено да користе њихов рад. То води циничном ставу или порицању. Програмер може са ентузијазмом описивати дело које сматра технички занимљивим. Али, када га упитају: „Да ли ћу и ја моћи да га користим?“, одмах спласне, и призна да је одговор одречан. Да би избегао обесхрабреност он или углавном игнорише ову чињеницу или усваја цинични гард који смањује њену важност.

Од Регановог доба до сада, Сједињене државе највише оскудевају не у технолошким иновацијама, већ у вољи да се заједнички ради у корист јавног добра. Бесмислено је развијати ово прво на рачун другог.

Ометање произвољних унапређења програма

Други ниво материјалне штете је немогућност прилагођавања програма. Лакоћа измене софтвера је једна од његових највећих предности над старијом технологијом. Али, већина комерцијалног софтвера је недоступна за измене, чак и пошто купите софтвер. Он је доступан само за узимање или остављање, попут црне кутије — и то је све.

Сваки програм који можете да покренете се састоји од низа загонетних бројева. Нико, па чак ни добар програмер не може на лак начин изменити те бројеве са циљем другачијег функционисања програма.

Обично програмери раде са „изворним кодом“ програма који се пише у програмском језику као што је Фортран или Це. У њима се користе имена да представе коришћене податке и делове програма, а операције се представљају симболима као што су „+“ за сабирање или „-“ за

одузимање. Они су осмишљени како би програмерима помогли да читају и мењају програме. Ево примера; програм за рачунање растојања између две тачке у равни може бити овакав:

```
float
distance (p0, p1)
    struct point p0, p1;
{
    float xdist = p1.x - p0.x;
    float ydist = p1.y - p0.y;
    return sqrt (xdist * xdist + ydist * ydist);
}
```

Заправо, значење тог изворног кода није проблем, него је проблем што изгледа као алгебра, особи која влада тим програмским језиком би било јасно о чему се ту ради. Насупрот томе, ево га исти тај програм у извршној форми у којој се налази на мом рачунару који сам користио док сам ово писао:

1314258944	-232267772	-231844864	1634862
1411907592	-231844736	2159150	1420296208
-234880989	-234879837	-234879966	-232295424
1644167167	-3214848	1090581031	1962942495
572518958	-803143692	1314803317	

Изворни код је користан (макар потенцијално) сваком кориснику програма. Међутим, већини корисника није дозвољено да поседују примерке изворног кода. Обично изворни код власничког програма власник чува у тајности, како нико други не би из њега нешто научио. Корисници примају само датотеке са неразумљивим бројевима које рачунар извршава. Ово значи да само власник програма може изменити програм.

Пријатељица ми је једном причала да је неких шест месеци радила као програмерка у банци пишући програм који је био сличан једном комерцијално доступном програму. Она сматра да је тај комерцијално доступни програм могла да са лакоћом прилагоди њиховим потребама да је имала његов изворни код. Банка је била вољна да јој за то плати, али то јој није било дозвољено — изворни код је био скривен. Зато је морала да се шест месеци бави непотребним послом, који се рачуна у бруто националном дохотку, али је објективно сувишан.

Фирма Ксерокс (*Xerox*) је 1977. поклонила графички штампач -овој Лабораторији за вештачку интелигенцију (ЛВИ). Њега је покретао слободни софтвер коме смо додали многе згодне одлике. На пример, софтвер је обавештавао корисника када би задатак за штампу био завршен. Када је са штампачем било проблема, као што су заглављивање или недостатак папира, софтвер би истог тренутка обавестио све кориснике који су послали задатке за штампу. Овакве могућности су осигуравале ефикасан рад са штампачем.

Касније је Ксерокс поклонио ЛВИ-ју нов, бржи штампач, који је био један од првих ласерских штампача. Покретао га је власнички софтвер који се извршавао на посебном рачунару, и коме нисмо смели да додамо ниједну од наших омиљених могућности. Могли смо да подесимо да се обавештење шаље при слању задатка за штампу том посебном рачунару, али не и када би штампа стварно почела (то одлагање је обично било знатно). Није било начина да се открије да

ли је задатак заиста одштампан — могли смо само да нагађамо. Како нико није обавештаван кад би се папир заглавио, штампач је често чекао на исправку проблема по читав сат.

Системски програмери у ЛВИ-ју су били способни да исправе такве проблеме, и то вероватно исто колико и аутори тог програма. Ксерокс за то није био заинтересован, и изабрао је да нас у томе спречи, па смо били приморани да их прихватимо. Ти проблеми никада нису исправљени.

Већина добрих програмера је искусила ово разочарање. Банка је могла да себи приушти да проблем реши пишући потпуно нов програм, али обичан корисник, без обзира на то колико је вешт, може само да одустане.

Одустајање психолошки штети самопоуздању. Деморалишуће је живети у кући коју не смете преуређивати. То води мирењу са таквим стањем и обесхрабрењу, које се може проширити и на друге делове нечијег живота. Људи који се овако осећају су несрећни и не обављају свој посао добро.

Замислите како би било да се рецепти скривају на исти начин као софтвер. Могли бисте да упитате великог мајстора-кувара: „Како да из овог рецепта избадим со?“, а он би одговорио: „Како се усуђујеш да вређаш мој рецепт, производ моје памети, покушајем да га скрнавиш? Ти немаш право да мењаш мој рецепт и поправљаш га!“

„Али мој лекар ми је рекао да не смем да једем со! Шта да радим? Да ли бисте ви могли да уместо мене одстраните со?“

„Биће ми драго да то урадим. Цена је само 50.000 долара.“ Пошто власник држи монопол на измене, цена је висока. „Међутим, тренутно немам времена. Заузет сам смишљањем новог рецепта за бродске колаче за Војну морнарицу. Можда ћу се позабавити твојим проблемом кроз две године.“

Ометање изградње софтвера

Трећи ниво материјалне штете погађа изградњу софтвера. Изградња софтвера је била еволутивни процес, у коме би неко поново написао делове неког постојећег програма како би додао нове одлике, а онда би неко други поново написао друге делове како би додао неке друге одлике. У појединим случајевима је то трајало и по двадесет година. У међувремену, делови програма су одвајани да би се од њих стварали зачеци других програма.

Постојање власника спречава овакву врсту еволуције, приморавајући програмере да почињу од нуле када развијају неки нови програм. Такође, оно спречава нове програмере да проучавају постојеће програме како би научили корисне технике или чак начине на које је могуће организовати велике програме.

Власници спречавају образовање. Сретао сам одличне студенте рачунарства који никада нису ни видели изворни код неког великог програма. Они су можда добри у писању малих програма, али они не могу почети да стичу разне вештине у писању великих програма уколико не виде како су то урадили други.

У свакој интелектуалној области се могу постићи веће висине стајањем на раменима других. Али то се уопштено више не дозвољава у области софтвера. Можете да стојите само на раменима других *запослених у вашој фирми*.

Ова психолошка штета нарушава дух научне сарадње, који је био толико јак, да су научници сарађивали чак и уколико би им земље биле у рату. На пример, јапански океанографи су сачували њихов рад када су амерички моринци извршили инвазију, и оставили су поруку у којој су од њих затражили да га чувају.

Сукоб због профита је уништио оно што су међународни сукоби поштедели. Данас у многим областима научници не објављују у својим радовима онолико информација колико је потребно да би други поновили њихов експеримент. Они објављују само онолико колико је потребно да би се читаоци дивили томе колико они могу да постигну. Ово свакако важи у рачунарству, где је изворни код програма који се обзнањују обично сакривен.

Није битно како ће дељење бити ограничено

Управо сам разматрао ефекте спречавања људи да умножавају, мењају и заснивају рад на неком програму. Међутим, нисам прецизирао како се та опструкција спроводи, јер то не утиче на закључак. Било да се она спроводи заштитом од умножавања, ауторским правима, лиценцама, шифровањем, ROM картицама или хардверским серијским бројевима, уколико је она *успешна* у спречавању употребе, она је штетна.

Корисници сматрају неке од тих метода мрскијим од других. Ја сматрам да су најмрскији они методи који постижу свој циљ.

Софтвер треба да буде слободан

Показао сам зашто је власништво над програмом, односно моћ да се ограничи његова измена или умножавање, опструктивно. Његове негативне последице су широко распрострањене и битне. Одатле следи да друштво не би требало да дозволи да програми имају власнике.

Други начин да се ово схвати је тај да је слободни софтвер оно што је друштву потребно, а да је власнички софтвер слаба замена за њега. Подстицање такве замене не представља рационалан начин да се обезбеди оно што је потребно.

Вацлав Хавел нас је саветовао да „Радимо за неки циљ зато што је добар, а не зато што има изгледа да успе.“ Делатност која производи власнички софтвер има изгледа да успе, посматрано из његовог уског угла, али то није добро за друштво.

Зашто ће људи изграђивати софтвер

Уколико елиминишемо ауторска права као начин подстицања људи да изграђују софтвер, испрва ће бити изграђено мање софтвера, али ће тај софтвер бити кориснији. Не може се предвидети да ли ће укупно испоручено корисничко задовољство бити мање. Али, уколико и буде такво, или уколико желимо да га у сваком случају повећамо, постоје други начини да се подстакне развој, исто као што постоје и други начини да се финансирају поправке путева, а не само преко

наплатних рампи. Пре него што размотрим како би то могло да се уради, желим да се упитам колико је заиста потребан вештачки подстицај.

Програмирање је забавно

Постоје неке радње које ће мало ко хтети да обавља без новца: на пример, изградња путева. Постоје и области науке и уметности у којима је богаћење мало вероватно, а у које људи улазе због опчињености тим областима или њихове уочене друштвене вредности. Неки од примера су математичка логика, класична музика, археологија и политичко организовање радника. Људи се надмећу, више тужно него грчевито, за неколицину плаћених места, од којих ниједно није превише плаћено. Они чак плаћају, уколико то могу да приште, за могућност да раде у струци.

Таква област може трансформисати саму себе преко ноћи уколико почне да нуди могућност богаћења. Када се један радник обогати, и други захтевају исту могућност. Ускоро сви могу захтевати велике своте новца за оно што су радили из задовољства. После још неколико година, сви који су повезани са том облашћу ће се подсмевати идеји да би посао у тој области требало да се обавља без великих финансијских награда. Они ће саветовати друштвене планере да се постарају да се те награде омогуће, додељујући посебне привилегије, моћи и монополе који су за то неопходни.

Оваква промена се десила у области рачунарског програмирања у осамдесетим годинама прошлог века. Седамдесетих било је новинских чланака и “зависности од рачунара”: како су људи били “стално на мрежи”: и да су их те навике коштале стотине долара месечно. Сматрало се да су људи толико волели програмирање да би због њега растурали бракове. Данас се сматра да се тиме нико не би бавио без велике плате. Људи су заборавили оно што су тад сматрали.

Иако је тачно да ће се у одређено време већина људи бавити одређеном облашћу само за високу плату, то не мора да остане тачно. Точак промена може да се окрене и уназад, уколико га друштво тако окрене. Уколико уклонимо могућност за велико богаћење, онда ће после извесног времена, када људи буду преиспитали своје ставове, они поново бити вољни да се баве одређеном облашћу само због радости достигнућа.

Питање „Како да исплатимо програмере?“ постаје лакше када схватимо да се не ради о финансирању њиховог богатства. Много је лакше финансирати зарађивање за живот.

Финансирање слободног софтвера

Институције које исплаћују програмере не морају да буду софтверске куће. Већ сада постоје многе друге институције које то могу да ураде.

Произвођачима хардвера је од суштинске важности да подрже развој софтвера, чак и ако не могу да контролишу употребу софтвера. Године 1970. је доста њиховог софтвера било слободно јер они нису имали намеру да га ограниче. Данас њихова растућа спремност да се придруже конзорцијумима показује њихов став да власништво над софтвером није оно што им је најважније.

Универзитети спроводе много програмерских пројеката. У данашње време они често продају резултате, али то нису чинили седамдесетих година двадесетог века. Ван сваке сумње,

универзитети би развијали слободни софтвер уколико им не би било дозвољено да продају софтвер. Овакви пројекти би могли да се подрже истоветним државним уговорима и субвенцијама као што су они који сада подржавају развој власничког софтвера.

У данашње време је општа пракса за истраживаче на универзитету да добијају субвенције за развој система, развију га скоро до краја, назову то „завршеним“ и оснују фирме у којима заиста доврше пројекат и оспособе га. Понекад они прогласе недовршену верзију „слободном“. Уколико су темељно искварени, уместо тога они издејствују ексклузивну лиценцу од универзитета. Ово уопште није тајна. То отворено признају сви који су у то умешани. Али чак и да истраживачи нису изложени овом искушењу, они би и тада истраживали.

Програмери који пишу слободни софтвер могу да зарађују за живот продајом услуга које су у вези са софтвером. Мене су ангажовали да прилагодим [ГНУ-ов преводац за Це](#) новом хардверу и да додам проширења корисничке спреге [ГНУ-овом Емаксу](#). (Та побољшања сам понудио јавности када су била завршена.) Такође држим предавања за која добијам хонорар.

У овоме нисам усамљен: данас постоји једна успешна фирма која не обавља другу врсту посла. И неке друге фирме такође обезбеђују комерцијалну подршку за слободни софтвер система ГНУ. То је почетак независне индустрије софтверске подршке, која би могла да веома ојача уколико би преовлађивао слободни софтвер. Она корисницима пружа могућност која код власничког софтвера није доступна, осим веома имућнима.

Нове институције, као што је [Задужбина за слободни софтвер](#), такође могу да финансирају програмере. Већина њених средстава потиче од корисника који купују траке преко поште. Софтвер на тракама је слободан, што значи да је сваком кориснику пружена слобода умножавања и измене софтвера, али и поред тога многи плаћају своје примерке. (Подсетимо се да се „слободни софтвер“ односи на слободу, а не на цену.) Траке наручују и неки корисници који већ поседују примерке, јер сматрају да заслужујемо прилог који нам на тај начин дају. Задужбина такође прима значајне прилоге од произвођача рачунара.

Задужбина за слободни софтвер је добротворна организација, и њен прилог се троши на запошљавање што већег броја програмера. Да је она организована као делатност, која расподељује исти слободни софтвер јавности за исту цену, сада би обезбеђивала пристојан живот њеном оснивачу.

Због тога што је Задужбина добротворна организација, програмери често за њу раде за упола мање него што би могли да зараде другде. Они то раде зато што у њој нема бирократије и зато што им сазнање да употреба њиховог дела неће бити спречавана пружа задовољство. Пре свега, они раде јер је програмирање забавно. Поред овога, добровољци су за нас написали много корисних програма. (Одскора су чак и технички писци почели да раде добровољно.)

Ово потврђује да је програмирање, уз музику и уметност, једна од најфасцинантнијих области. Не морамо да страхујемо да нико неће желети да програмира.

Шта корисници дугују градитељима?

Постоји оправдан разлог зашто би корисници софтвера требало да осећају моралну обавезу да га подрже. Градитељи слободног софтвера дају прилог активностима тих корисника, а правично је и уједно у дугорочном интересу корисника да им они дају средства да са тим наставе.

Међутим, ово се не односи на градитеље власничког софтвера, јер опструкционизам заслужује казну, а не награду.

Тако имамо парадокс: градитељ корисног софтвера има право на подршку корисника, али било какав покушај да се ова морална обавеза претвори у захтев уништава основу те обавезе.

Градитељ може или да заслужи награду или да је захтева, али не и једно и друго.

Верујем да градитељ који је етичан мора, суочен са овим парадоксом, да дела тако да заслужи награду, али би такође требало и да замоли кориснике за добровољне прилоге. Са временом ће корисници научити да подрже градитеље без принуде, на исти начин на који су научили да подрже јавне радио и телевизијске станице.

Шта је софтверска продуктивност?

Програмера би било и када би софтвер био слободан, али можда у мањем броју. Да ли би то било лоше по друштво?

Не би морало да буде лоше. У данашње време у напредним земљама је мање сељака него 1900. године, али се то не сматра лошим по друштво, јер неколицина њих пружа потрошачима више хране него што је то могло мноштво. То се назива повећаном продуктивношћу. Слободном софтверу би било потребно много мање програмера да задовољи потражњу, и то због повећане софтверске продуктивности на свим нивоима:

- Већа распрострањеност сваког програма који се изграђује.
- Могућност да се повећа прилагодљивост постојећих програма уместо писања нових од почетка.
- Боља обука програмера.
- Спречавање удвостручавања градитељских напора.

Они који се противе сарадњи јер би она довела до смањења броја запослених програмера уствари се противе повећању продуктивности. И поред тога они обично прихватају распрострањено мишљење да је индустрији софтвера потребна повећана продуктивност. Како је то могуће?

„Софтверска продуктивност“ може да означава две различите ствари: укупну продуктивност свих изградњи софтвера или продуктивност појединачних пројеката. Укупна продуктивност је оно што би друштво желело да побољша, а најпростији начин да се то уради јесте да се уклоне вештачке препреке које смањују сарадњу. Међутим, они који проучавају област „софтверске продуктивности“ се усредсређују на друго, ограничено значење овог израза, у ком побољшање захтева компликоване технолошке помаке.

Да ли је надметање неизбежно?

Да ли је надметање зарад престижања супарника у друштву неизбежно? Вероватно јесте. Али надметање само по себи није лоше. Оно што је лоше је *борба*.

Постоје многи начини да се надмеће. Надметање се може састојати у покушају да се постигне још више, да се уради више и боље од других. На пример, у старо добро време су постојала надметања програмерских „чаробњака“ — надметања ко ће постићи невероватније ствари на рачунару, или ко може да направи најкраћи или најбржи програм за одређен задатак. Оваква врста надметања може да користи свима, *све док се одржава спортски дух*.

Конструктивно надметање је довољно да подстакне људе на велика дела. Неколико људи се надмеће око тога ко ће бити први човек који је посетио све државе на Земљи. Неки су покушавајући да то ураде потрошили читаво богатство. Али они не подмићују капетане бродова да насуку њихове супарнике на напуштена острва. Они су задовољни пуштањем да бољи победи.

Надметање се претвара у борбу онда када они који се надмећу почну да ометају једни друге уместо да лично напредују, то јест онда када принцип „Нека најбољи победи“ устукне пред принципом „Дај ми да победим, без обзира на то какав сам“. Власнички софтвер је штетан не зато што је облик надметања, већ зато што је облик борбе између грађана нашег друштва.

Сарадња између делатности не мора обавезно да буде борба. На пример, кад се две бакалнице надмећу, сав њихов напор лежи у побољшању њиховог посла, а не у саботирању ривала. Али ово није знак посебне оданости пословној етици. То је пре последица тога да нема много простора за борбу у тој врсти посла изван физичког насиља. Ову црту не деле све врсте делатности. Скривање информација које би свима помогле да напредују представља облик борбе.

Пословна идеологија не припрема људе да одоле искушењу да се боре против конкуренције. Неки облици борбе су забрањени антимонополским законима, законима о оглашавању који захтевају истинитост информација, итд. Међутим, уместо да прошире ово до одбијања борбе у општем смислу, извршни директори измишљају друге облике борбе који нису експлицитно забрањени. Друштвени ресурси се траће на економски облик фракционашког грађанског рата.

„Пресели се у Русију!“

У Сједињеним државама, свако ко заступа било какав став који је различит од најекстремнијег себичног конформизма често трпи овакве оптужбе. На пример, оне су уперене против присталица националног система здравствене заштите, који постоји у свим другим индустријализованим нацијама слободног света. Оне су уперене против заступника друштвеног спонзорисања уметника, што је такође уобичајено у напредним земљама. Идеја о томе да грађани имају било какву обавезу према друштвеном просперитету се у Америци сматра комунизмом. Али колико су те идеје заиста сличне?

Комунизам је, у оном облику у ком је примењен у Совјетском савезу, био систем централизоване контроле, где су све активности биле надгледане, наводно ради општег добра, а уствари за рачун

чланова Комунистичке партије. Опрема за умножавање је била строго надгледана да би се избегло нелегално умножавање.

Амерички систем интелектуалне својине представља централизовану контролу над расподелом програма, и надгледа опрему за умножавање уз помоћ аутоматских шема за заштиту од умножавања да би спречио нелегално умножавање.

Насупрот овоме, ја радим на изградњи система у коме ће људи бити слободни да одлучују о својим поступцима. Конкретно, имаће слободу помагања својим ближњима и слободу измене и побољшања алата које свакодневно користе. То је систем заснован на добровољној сарадњи и децентрализацији.

Према томе, уколико ћемо процењивати ставове по њиховој сличности са руским комунизмом, онда су власници софтвера комунисти.

Питање премиса

У овом тексту претпостављам да корисник софтвера није мање важан од аутора тог софтвера или ауторовог послодавца. Другим речима, њихови интереси и потребе су подједнако важни при одлучивању о томе шта је најбоље предузети.

Ова премиса није општеприхваћена. Многи остају при ставу да је ауторов послодавац суштински важнији од било кога другог. Они, на пример, кажу да је сврха постојања власника софтвера да се ауторовом послодавцу пружи предност коју заслужује, без обзира на то како то може утицати на јавност.

Не вреди покушавати да се ове премисе докажу или оповргну. За доказ су потребне заједничке премисе. Зато је већина онога о чему говорим намењена само онима који деле премисе које ја користим, или су барем заинтересовани за то какве су њихове последице. За оне који сматрају да су власници важнији од било чега другог, овај текст је сасвим небитан.

Зашто би велики број Американаца прихватио премису која одређене људе издиже изнад других? Делом због тога што се сматра да је ова премиса део правничке традиције америчког друштва. Неки су мишљења да преиспитивање премисе представља угрожавање друштвених основа.

Важно је да такви људи знају да ова премиса није део наше правничке традиције. Она то никада није ни била.

Тако, Устав^[1] говори да је сврха ауторских права да „подстакну напредак науке и корисних вештина.“ Врховни суд је ово образложио доносећи пресуду у случају „Фокс Филм против Дојала“ на следећи начин: „Једини интерес Сједињених држава и основни приговор предаји монопола [над ауторским правима] лежи у општим користима рада аутора јавности.“

Од нас се не захтева да се слажемо са Уставом или Врховним судом. (Обоје су својевремено бранили ропство.) Дакле, њихови ставови не осуђују премису о суверености власника. Али, ја се надам да ће свест о томе да је ово екстремна десничарска премиса, а не она која је у складу са традицијом, ослабити њен значај.

Закључак

Ми волимо да мислимо да наше друштво охрабрује помагање ближњем, али сваки пут када наградимо неког за опструкционизам или када смо задивљени богатством које је стекао на овај начин, уствари шаљемо супротну поруку.

Присвајање софтвера је један облик наше опште воље да занемаримо добробит друштва зарад личне користи. Можемо да пратимо ово занемаривање од Роналда Регана до Дика Чејнија, од Ексона до Енрона, од пропалих банака до пропалих школа. Можемо да га меримо бројем бескућника и затвореника. Антидруштвени дух се храни самим собом, јер што више увиђамо да нам други неће помоћи, више нам изгледа узалудно да му помажемо. Тако друштво копни у џунглу.

Ако не желимо да живимо у џунгли, морамо да променимо наш став. Морамо да почнемо да шаљемо поруку да је добар грађанин онај који сарађује када је прикладно, а не онај који је успешан у узимању од других. Надам се да ће покрет за слободни софтвер допринети овоме: макар у једној области, заменићемо џунглу ефикаснијим системом који подстиче добровољну сарадњу и од ње зависи.

Фусноте

1. Реч „слободно“^[2] се односи на слободу, а не на цену. Цена примерка слободног програма може да буде нулта, али и мала или (ретко) веома велика.
2. Питања загађења и закрчења саобраћаја не мењају овај закључак. Ако желимо да поскупимо возњу да бисмо је у потпуности спречили, није добро то учинити преко наплатних станица, које доприносе и загађењу и закрчењу. Порез на бензин је много боље решење. Исто, жеља за повећањем безбедности смањењем брзине није од значаја; пут са слободним приступом побољшава средњу брзину избегавањем заустављања и чекања, при било којем ограничењу брзине.
3. Могли бисмо да сматрамо конкретни рачунарски програм штетном ствари која уопште не би ни требало да буде доступна, као што је то база личних података Лотусова Пијаца, која је повучена из продаје због јавног негодовања. Већина оног што ја тврдим се не односи на овај случај, али нема смисла расправљати треба ли имати власнике или не само на основу тога што ће власник програм учинити мање доступним. Власник га неће учинити *сасвим* недоступним, као што би се желело у случају програма чија употреба се сматра штетном.

Овај есеј је објављен у књизи [Слободни софтвер, слободно друштво: одабрани есеји Ричарда М. Сталмана](#).

Примедбе преводиоца:

1. Мисли се на устав и Врховни суд САД. [CP]
2. У енглеском оригиналу: „free“ — израз који означава и слободу и бесплатност. [CP]