

### Свободен софтуер в науката

Вася Атанасова

Институт по информационни технологии, Българска академия на науките  
[yassia.atanassova@gmail.com](mailto:yassia.atanassova@gmail.com)

Неведнъж са правени паралели между науката и свободния софтуер и са откривани сходствата в принципите, на които функционират ([1], [2]). На пръв поглед, тези две понятия са твърде различни по своята степен на общност, за да бъдат сравними. Но така е само на пръв поглед. Ако се вгледаме по-внимателно, ще осъзнаем, че софтуер вече се разработва за всяка една научна дисциплина, при това за решаването на отделни научни проблеми от всяка област съществуват не само по един софтуерен продукт, а често и цели продуктови гамии на един софтуерен производител или продукти на конкуриращи се производители. Нещо повече, в много научни направления има разработен не само собственически софтуер със затворен код, но и свободен софтуер и софтуер с отворен код.

Нека за начало дефинираме и разтълкуваме понятията, с които боравим.

Собственически софтуер (*proprietary software*) е софтуер, който има известен собственик в лицето на своя производител и този собственик държи изключителните права над негово съдържание, разработка, управление и комерсиализация. За да запази изключителните си права, собственикът не предоставя на клиентите си достъп до програмния код, а само до продукт под формата на изпълним файл или до услуга, предоставяна онлайн (*software as a service*). В този смисъл, закупуването на собственически софтуер означава заплащане от страна на крайния потребител на цена, определена от собственика, за правото на използване на копие при дадени условия, чието неизпълнение в противен случай се третира като нарушение на изключителните права на собственика според законите за авторските и сродните им права. Сред споменатите условия, освен цената, могат да влизат периодът на времетраене на полученото с лиценза право, версията на софтуера, за която е закупено правото за ползване, броят работни станции, на които копието може законно да се инсталира (обикновено една), изискване за онлайн регистрация или друг начин за предоставяне на лични данни за клиента, и т.н. Без значение колко висока е цената на собственическия софтуер, закупуването на лиценза за ползването му не включва и правото на достъп до кода. Клиентът е само краен потребител, който няма възможност да определя функционалностите на софтуера, да го надгражда според потребностите си, или дори само да го опознае и използва идеите от реализацията му за други свои проекти.

Това статукво най-често е напълно задоволително за крайни потребители на софтуер, които нямат необходими познания или неудовлетворени потребности, за да представлява за тях интерес достъпът до програмния код на приложението. Но не така стоят нещата при хората, които не са просто консуматори на готови решения, а

сами разполагат с нужните компетенции и ресурси или имат необходимост да управляват софтуера, вместо да оставят софтуерът да управлява тях. В този момент се явява потребността от несобственически софтуер с незатворен код. Възприето е такъв вид софтуер да се нарича „свободен софтуер“.

Категорията „свободен софтуер“ (*free software*) е приложима над всяко програмно приложение, без значение от програмния език, на който е написано, от хардуерно-софтуерната платформа, над която работи, от научно-приложните задачи, които решава, или от цената, която е платена за него (или пък не е платена). Понятието се дефинира посредством набор от неизключителни условия за използване и разпространение, които са определени от разработчиците на софтуера, които остават носители на авторски права над него. По-конкретно, свободният софтуер дава на потребителите си следните четири свободи:

1. Свобода за използване за всякакви, включително комерсиални цели.
2. Свобода за изучаване на програмния код на приложението.
3. Свобода за модифициране на кода.
4. Свобода за разпространение на свой ред на софтуера.

Всички тези четири свободи са по подразбиране ограничени при несвободния, наречен още собственически софтуер. В същото време, терминът „свободен софтуер“ не е синоним на „безплатен софтуер“ (*freeware*), макар че много често той се явява такъв в част от своите дистрибуции или изцяло. За предоставянето на свободен софтуер също може да се иска възнаграждение, но разликата със собственическия софтуер е, че срещу това възнаграждение потребителят получава и права за използване за всякакви цели, за достъп до кода, за надграждането му, и за разпространяването му, било срещу заплащане или безплатно, било в оригиналния или в променения му вид. За разлика от собственическия софтуер, свободният софтуер не криминализира желанието на потребителите да споделят с други получените знания, удоволствия и удобства (практиката, заклеямена като „софтуерно пиратство“), нито пък криминализира стремежа им да изучават и променят програмния код зад изпълнимия файл (т.нар. „обратно инженерство“, *reverse engineering*). За разлика от собственическия софтуер, свободният софтуер също така не възпрепятства и възможността на потребителите да комерсиализират резултата от труда си, вложен в надграждането на законно придобития програмен код, който те получават в комплект със свободната програма.

В основите си, науката и свободният софтуер най-много си приличат по това, че са немислими без синергията между отделните индивиди, т.е. феномена, при който съвместно хората постигат много повече отколкото всеки един от тях би постигнал поотделно. Една от крилатите фрази за това принадлежи на сър Исак Нютон: „Ако съм могъл да видя по-надалеч, то е защото съм стоял на раменете на гиганти“. Друга мисъл в тази посока изрича Алберт Айнщайн: „Ако аз имам една ябълка и ти имаш една ябълка и, ако ги разменим, всеки от нас ще има по една ябълка, но ако аз имам една идея и ти имаш една идея и ги разменим, всеки от нас ще има по две идеи.“ Както в науката, така и в сферата на свободния софтуер, постиженията на работата в сътрудничество облагодетелстват не само учените, но и обществото като цяло, в това число и икономиката. В по-близко или по-далечно бъдеще, от тези постижения се облагодетелстват дори и онези индивиди и организации, които са били безучастни към напредъка на науката и технологиите, или дори активно са ги възпрепятствали.

Дори и само от така представените дефиниции и тълкувания читателят вече съзира сходствата между свободния софтуер и науката. Но има и още една, по-трудно доловима връзка между тях, и тя произтича от научния метод.

Част от научния метод се състои в изграждането на нови теории и провеждането на експерименти, които да ги проверят и потвърдят на практика. В зависимост от конкретната научна област и метод, за верни се считат експериментално потвърдени хипотези или формално доказани твърдения. Ако дадено твърдение се докаже веднъж формално, то всяко допълнително доказателство няма да добави ново съществено научно познание. Друг научен метод е отхвърлянето на дадена теория чрез провеждането на експеримент или изнамирането на контрапример, който посочва слабост на теорията. Това може да доведе до необходимост от преформулиране на теорията или до пълното ѝ опровергаване, и то независимо от количеството потвърждения, които са съществували до този момент. Един-единствен контрапример може да обезсмисли безкрайно много примери, удостоверявали верността на дадена теория.

Къде тук обаче е връзката със свободния софтуер?

Софтуерът днес вече е и в бъдеще ще става все по-съществена и неотменна част от науката. Не става въпрос за офис пакетите с текстообработващи програми, електронни таблици и програми за мултимедийни презентации, с които учените документират и популяризират своите научни резултати. От гледна точка на научното качество е без значение дали ученият ще ползва за публикацията си най-прост текстов редактор или последната версия на най-скъпия софтуер за професионален предпечат и дизайн.

Връзката между софтуера и научния метод се проявява тогава, когато провеждането на научните експерименти частично или изцяло зависи от софтуера. С други думи, когато софтуерът стане част от инструментариума на научния процес и съответно – част от резултатите от този процес.

Първият по-значим такъв случай е доказателството с помощта на компютър на Теоремата за четирите цвята, която след Последната теорема на Ферма е вторият най-известен останал дълго нерешен проблем в математиката, [3]. Теоремата е печално известна с големия брой неуспешни опити и пропуски във формалното ѝ доказателство, първият от които, на Август де Морган, датира от 1852 година. През 1976 година Кенет Апел и Волфганг Хакен предлагат компютърно доказателство с пълно изчерпване на случаите, но впоследствие възникват множество противоречия и съмнения за коректността на метода им. Въпреки че през 2005 година отпадат и последните съмнения относно доказателството на Апел и Хакен, този и други математически проблеми вече са поставили на дневен ред потребността от изследване на формалната коректност на компютърно реализираните алгоритми, т.нар. верификация на програми.

За всеки човек, който някога е разработвал софтуер, е очевидно, че дори да е известен и формално коректен алгоритъмът, вложен в програмата, това не е гаранция, че винаги също тъй коректно ще е функционирането на програмата, а оттам и резултатите от научните експерименти, провеждани с нея. Необходимо е да се знае и каква е конкретната програмна реализация на алгоритъма. Несвободният софтуер не дава такава информация, понеже той, по дефиниция, не предоставя на потребителите изходния код, от който е компилиран изпълнимият файл на програм-

ното приложение. Несвободният софтуер крие програмния код от потребителите, обявявайки го за своя „търговска тайна“ или „патент“.

В същото време, без непосредствен, пряк, законен достъп до програмния код на софтуера, опитите за извличане на необходимата информация са: обикновено много трудно осъществими, понякога практически невъзможни, никога абсолютно сигурни и винаги изискващи отклоняване, само за тази цел, на много допълнителни човешки ресурси и време. Нещо повече, достъпът до това познание почти винаги се възпрепятства чрез специално проектирани за целта софтуерни и хардуерни средства, чието заобикаляне се обявява за дейност, която нарушава авторските права на фирмата-производител и се преследва по съдебен път. Ограничаването на достъпа до кода, а в добавка и цената на софтуера, допълнително възпрепятства възможността на научната общност да повтори и да се увери в коректността на получените с негова помощ резултати от научните изследвания.

Интересна е ситуацията при два конкуриращи се софтуерни продукта под затворен код, които дори само в един момент връщат различни или противоположни резултати при едни и същи изпитания. Само един такъв случай е достатъчен, за да постави под съмнение всички дотогавашни (съвпадащи) резултати, които всяка от програмите е връщала. И тъй като няма възможност да се провери една от важните брънки по веригата, а именно съдържанието на програмния код на двата продукта, то би се оказало, че целият използван научен метод се крепи само на вяра и доверие.

В това се изразява ефектът на „черната кутия“ при несвободния софтуер [4]. Ученият използва някакъв софтуер, получава някакви резултати, прави някакви научни изводи. Но без пълен достъп до знанията, от които зависи, без пълен контрол над своя научен инструментариум, ученият винаги ще бъде зависим от нечия „търговска тайна“ и няма да разполага с други средства да установи кога потвържденията и опроверженията на научните му хипотези са закономерни или случайни, кога са следствие от самите хипотези, от алгоритмите, използвани за проверка на хипотезите, от реализацията на тези алгоритми, от входните данни или от нещо друго. Ето защо собственическият, несвободен софтуер е несъвместим с научния метод.

## Литература

- [1] “Free Software in Science”, Georg Greve, Free Software Foundation, A National Workshop on Free Software in Science, Union Christian College, 4-6 Jan. 2007, <http://www.space-kerala.org/fsc/index.php/Fsc/FreeSoftwareInScience> (последно посетен на 14 декември 2009)
- [2] “The 'Open-Source Movement' Turns Its Eye to Science”, Vincent Kiernan, Chronicle of Higher Education, 5 Nov. 1999, <http://chronicle.com/free/v46/i11/11a05101.htm> (последно посетен на 14 декември 2009)
- [3] “Last doubts removed about the proof of the Four Color Theorem”, Keith Devlin, MAA Online, Jan. 2005, [http://www.maa.org/devlin/devlin\\_01\\_05.html](http://www.maa.org/devlin/devlin_01_05.html) (последно посетен на 14 декември 2009)
- [4] “Robust Nonproprietary Software”, Peter Neumann, IEEE Symposium on Security and Privacy, Oakland CA May 15-17, 2000, <http://www.csl.sri.com/~neumann/ieee00+.pdf> (последно посетен на 14 декември 2009)