

## ЧАСТИЧНА АВТОМАТИЗАЦИЯ НА ЛИЦЕНЗИОННО СЪОТВЕТСТВИЕ ПРИ ИЗПОЛЗВАНЕ НА СОФТУЕР С ОТВОРЕН КОД

**Владимир Славов**

„Босх Диджитал“ ЕООД, България

***Резюме:** Софтуерът се ползва с авторскоправна закрила. Следователно, за да бъде използван от лице, различно от носителя на авторското право, е необходимо правно основание, каквото например е лицензът. Специфична категория софтуер е т.нар. софтуер с отворен код, който в някои индустрии достига до 85% от използвания софтуер. Това, което го прави специфичен, е специалната категория лицензи, под които се предоставя. Тези лицензи предоставят широк набор от права на лицензополучателя, но предвиждат и задължения в определени ситуации, като например момента на предоставяне на копия от лицензирания софтуер или неговите производни произведения на трето лице. Това създава необходимост организациите да изградят процеси, чрез които да оценят дали и до каква степен софтуерни компоненти с отворен код под определени лицензи могат да бъдат използвани в конкретни бизнес контексти. Колкото повече софтуерни компоненти с отворен код една организация използва, толкова повече нараства необходимостта процесът да бъде подпомогнат от автоматизация, в резултат от което ролята на юриста в този процес значително се променя.*

***Ключови думи:** софтуер, софтуер с отворен код, софтуерно лицензиране, лицензионно съответствие, правна автоматизация, процес*

## PARTIAL AUTOMATION OF OPEN SOURCE SOFTWARE LICENSE COMPLIANCE

**Vladimir Slavov**

Bosch Digital LTD, Bulgaria

***Abstract:** Software is subject to copyright protection. Therefore, in order to use a piece of software, any entity other than the copyright holder must rely on a legal basis, such as a license. Open source software is a specific kind of software which, in some industries, reaches up to 85% of all used software. What distinguishes open source software is the special category of licenses it relies on. These open source*

*licenses grant a broad range of rights to the licensee, but they also require the fulfillment of obligations in certain cases, such as when a copy of the software or a derivative work thereof is distributed to third parties. This creates a necessity for organizations to develop processes that evaluate whether, and to what extent, components under particular open source licenses can be used in certain business contexts. The more open source software components an organization uses, the greater the need to enhance the process with automation. As a result, the role of legal professionals in this process changes significantly.*

**Key words:** *software, open source software, software licensing, license compliance, legal automation, process*

## **ЦЕЛ И СТРУКТУРА НА ИЗСЛЕДВАНЕТО**

Настоящата статия поставя в центъра на изследването въпроса дали и доколко е възможно една традиционно правна дейност да бъде автоматизирана посредством организационни процеси и софтуерни инструменти. Конкретно ще бъде разгледан проблемът с привиждането на дейността на една организация<sup>1</sup> в съответствие с изискванията на лицензите на използвания от нея софтуер с отворен код (OSS),<sup>2</sup> чийто носител на авторското право е трето лице. За целта, на първо място, ще бъдат разгледани някои от основните характеристики на софтуера с отворен код, степенята му на проникване в дейността на организациите в световен мащаб и правните специфики на лицензите, под които той се предоставя. В резултат от това ще бъде обосновано, че внедряването на автоматизация е не само възможно, но и необходимо, като ще бъдат представени примерни процеси за анализ и оценка на OSS лицензите без и с автоматизация, както и примерни инструменти, които могат да бъдат използвани. Специално внимание ще бъде обърнато на това какви са границите на автоматизацията и как тя променя ролята на юриста.

## **ЗА СОФТУЕРА С ОТВОРЕН КОД И НЕГОВИТЕ ЛИЦЕНЗИ**

Софтуерът с отворен код е широко разпространен в различни индустрии, в т.ч. софтуерната индустрия и разработваните от нея продукти. Интеграцията му е напреднала до такава степен, че според проучване от 2022 г. в някои индустрии OSS достига до 85% всички използ-

---

<sup>1</sup> В настоящата статия под „организация“ следва да се разбира всяко юридическо лице, лицензополучател по смисъла на съответния софтуерен лиценз, независимо от естеството на дейността му.

<sup>2</sup> Оригиналният термин на английски за софтуера с отворен код е Open Source Software, с общоприетото съкращение OSS, което ще бъде използвано и в тази статия.

ван софтуер.<sup>3</sup> На практика това означава, че по-голямата част от софтуера, който тези организации използват, включително в своите продукти, не е разработен от тях самите, а от трети лица, следователно използването му е възможно единствено в съответствие с условията на конкретните лицензи, под които този OSS се предоставя.<sup>4</sup>

Това е така, тъй като софтуерът е обект на авторскоправна закрила.<sup>5</sup> Той съществува в две форми – изхоген код (още познат като сорс код, от английски език *source code*) и бинарен код. Сорс кодът е сбор от инструкции, написани на разбираем за човека език за програмиране, обикновено като текст.<sup>6</sup> Това е формата, в която програмистът пише компютърна програма. В резултат на процес, наречен компилация,<sup>7</sup> сорс кодът се превръща в бинарен код, съставен от нули и единици, който може да бъде прочетен и изпълнен от компютър. Всички програми, които използваме на нашите лаптопи, както и всички приложения на нашите телефони, са компютърни програми, съществуващи под формата на бинарен код. И двете форми на софтуера (сорс кодът и бинарният код) са обект на авторскоправна закрила.<sup>8</sup>

Софтуерът с отворен код е специфичен вид софтуер, чийто сорс код е свободно достъпен, включително в интернет, и който се предоставя под специална категория софтуерни лицензи.<sup>9</sup> Тези лицензи са

---

<sup>3</sup> Срв. Whiting, D. Why Do Enterprises Use and Contribute to Open Source Software? (<https://www.linuxfoundation.org/blog/blog/why-do-enterprises-use-and-contribute-to-open-source-software>).

<sup>4</sup> Разбира се, изключение прави софтуерът с отворен код, чийто носител на авторско право е същата организация, която го използва. Той ще бъде по-малък или по-голям процент от използвания OSS, в зависимост от това доколко организацията участва в неговата разработка.

<sup>5</sup> Чл. 3, ал. 1, т. 1 ЗАПСП изрично предвижда авторскоправна закрила за компютърните програми, когато са създадени в резултат на творческа дейност.

<sup>6</sup> Вж. [https://bg.wikipedia.org/wiki/Изхоген\\_код](https://bg.wikipedia.org/wiki/Изхоген_код).

<sup>7</sup> Вж. <https://bg.wikipedia.org/wiki/Компилятор>. Изключение правят програмните езици, които се интерпретират.

<sup>8</sup> В този смисъл Решение от 22 декември 2010 г., *Bezpečnostní softwarová asociace*, C-393/09, т. 34 – 35.

<sup>9</sup> Едно уточнение, свързано с лицензионния механизъм на софтуера с отворен код: Обикновено това става чрез добавянето от страна на носителя на авторското право на текстов файл, който се съхранява на същото място като останалите файлове на софтуера и излага условията, под които трето лице може да го използва. Вероятно в някои юрисдикции би се повдигнал въпросът дали в такава хипотеза изобщо е налице валидно сключен договор за използване на софтуера. Този въпрос, макар да е интересен от теоретична гледна точка, е практически лишен от смисъл. Освен ако използването на софтуера от страна на третото лице не попада в някое от предвидените в закона гонустими без

силно стандартизирани, тъй като всички те следват дефиниция,<sup>10</sup> чиито изисквания трябва да покриват, за да се считат за OSS лицензи. Дефиницията се състои от 10 точки, които могат да бъдат синтезирани като необходимост съответният лиценз да дава на потребителя на софтуера<sup>11</sup> право да го използва за всякаква цел, право да го изследва, право да го променя и право да споделя копия, в т.ч. на производни произведения.<sup>12,13</sup> Тези права поначало се предоставят без насрещно задължение за плащане, неексклузивно и за неограничен период от време. „Пазителят“ на дефиницията, регистрирана в Калифорния, САЩ, е корпорация с обществена полза (*public benefit corporation*), наречена *Open Source Initiative*.<sup>14</sup> Тя има сертификационен механизъм за OSS лицензи.<sup>15</sup> Тези лицензи, които са оценени като покриващи изискванията на дефиницията, могат да бъдат открити на уебсайта на организацията.<sup>16</sup>

### ИЗПОЛЗВАНЕ НА СОФТУЕР С ОТВОРЕН КОД В ОРГАНИЗАЦИЯТА

Съществуват два основни начина, по които софтуерът с отворен код може да се използва.

Първият начин е като потребителят сваля софтуера на своето устройство и просто го използва като самостоятелна апликация. Приме-

---

съгласието на носителя на авторското право действия, като например тези по чл. 24 Вр. с чл. 23 ЗАПСП, то е поначало възможно единствено съгласно предвидените в текстовия файл условия, които наричаме лиценз. Следователно при един хипотетичен правен спор за неправомерно използване на чужд обект на авторско право никое лице не би изложило твърдението, че правоотношение не съществува и че то не е обвързано от условията на лиценз, тъй като веднага би се повдигнал въпросът на какво основание изобщо го използва.

<sup>10</sup> The Open Source Definition (Annotated) (<https://opensource.org/definition-annotated>).

<sup>11</sup> В тази статия терминът „потребител“ се използва в смисъла, който му придава софтуерната терминология, т.е. ползвател на софтуера, а не в смисъла, който му се придава в потребителското право – срв. § 13, т. 1 ДР Закон за защита на потребителите.

<sup>12</sup> Още познати като четирите свободи на свободния софтуер – вж. What is Free Software? (<https://www.gnu.org/philosophy/free-sw.en.html>). Въпреки че съществуват леки разминавания между 10-точковата система на OSI и четирите свободи на свободния софтуер, и двете системи рисуват доста сходна картина на това какво означава OSS.

<sup>13</sup> Правото на лицензополучателя да създава производни произведения, когато е получил за това разрешение от носителя на авторското право върху оригиналното произведение, е предвидено в чл. 18, ал. 2, т. 8 ЗАПСП.

<sup>14</sup> Вж. <https://opensource.org/about>.

<sup>15</sup> От това обаче не следва заключението, че един лиценз не може да бъде OSS лиценз, ако не е сертифициран като такъв от OSI.

<sup>16</sup> Вж. OSI Approved Licenses (<https://opensource.org/licenses>).

ри за такъв софтуер са популярният редактор за текст на Microsoft Visual Studio Code,<sup>17</sup> софтуерът за превръщане на електронни книги от един формат в друг формат, Calibre<sup>18</sup>, и много други. Този начин на използване на софтуер с отворен код поначало не представлява проблем от правна гледна точка. Както бе посочено, OSS лицензите предоставят правото на лицензополучателя да използва софтуера за всякаква цел.

По-голям интерес за настоящата статия представлява вторият начин на използване на софтуера с отворен код, а именно като компонент<sup>19</sup> в продукт с цифрови елементи.<sup>20</sup> Макар OSS лицензите да не поставят ограничения върху начина, по който лицензираният софтуер може да се използва, те предвиждат конкретни задължения, които лицензополучателят трябва да изпълни в определени ситуации. За по-голямата част от лицензите това е моментът, в който лицензополучателят предостави на трето лице копие от софтуера или производно на него произведение.<sup>21</sup>

Повечето задължения до голяма степен се припокриват в различните лицензи. Те включват задължение да се предостави на третото лице информация, че софтуерният продукт съдържа съответния компонент с отворен код, под какъв лиценз се предоставя той и кой е неговият носител на авторското право. Типично задължение за някои OSS лицензи е свързано с т. нар. Копилефт ефект,<sup>22</sup> който по същество се свежда до задължение копия от производни произведения да се пре-

---

<sup>17</sup> Вж. <https://github.com/Microsoft/vscode>, публикуван под MIT лиценз.

<sup>18</sup> Вж. <https://github.com/kovidgoyal/calibre>, публикуван под GLP-3.0 лиценз.

<sup>19</sup> В света на софтуера с отворен код има няколко понятия за този случай, които се използват до голяма степен взаимнозаменяемо – компонент (component), библиотека (library) и dependency. Същественото е, че основният софтуер не може да работи правилно без съответния OSS компонент, от който зависи.

<sup>20</sup> Терминът „продукт с цифрови елементи“ се използва в Предложение за регламент на Европейския парламент и на Съвета относно хоризонтални изисквания за киберсигурност за продукти с цифрови елементи и за изменение на Регламент (ЕС) 2019/1020 и е дефиниран в чл. 3, т. 1) като „всеки софтуерен или хардуерен продукт и неговите решения за обработване на данни от разстояние, включително софтуерни или хардуерни компоненти, които ще бъдат пуснати на пазара поотделно“. Терминът не е от първостепенно значение за целите на тази статия, но е полезен, тъй като обхваща добре различните продукти, съдържащи софтуер, които достигат до пазара. В настоящия труд „продукт с цифрови елементи“ и „софтуерен продукт“ ще бъдат използвани като взаимнозаменяеми термини.

<sup>21</sup> Изключения от този принцип са лицензи като AGPL-3.0, които предвиждат задължения и при предоставяне на достъп до софтуера на потребител посредством мрежа – вж. Art. 13 от същия, GNU Affero General Public License (<https://www.gnu.org/licenses/agpl-3.0.de.html>).

<sup>22</sup> От английски *copyleft*, което е игра на думи върху термина за авторско право *copyright* – вж. What is Copyleft? (<https://www.gnu.org/licenses/copyleft.en.html>).

гоставят под същия лиценз като оригиналното произведение. Самият лиценз включва задължение да се предостави сорс кодът на софтуера. По този начин се създава механизъм, който препречва възможността софтуерът с отворен код да се превърне в комерсиален софтуер със затворен код.

Самият Копилефт ефект може условно да бъде разделен на силен и слаб Копилефт ефект, в зависимост от това дали лицензът предвижда механизъм, чрез който ефектът може принципно да бъде ограничен единствено до компонента с отворен код, или се разпростира до всички производни произведения, използващи съответния компонент. Съществуват обаче и някои по-нетипични задължения, като например специфичното изискване на BSD-4.0 лиценза<sup>23</sup> рекламни материали, свързани с производни произведения, да предоставят информация, че съответният софтуер включва компонент, разработен от съответната организация, носител на авторското право върху компонента под BSD-4.0 лиценз.

Към 17 ноември 2024 г. популярната база данни за софтуерни лицензи ScanCode LicenseDB<sup>24</sup> съдържа общо 2356 лиценза, повечето от които могат да бъдат счестени за една или друга категория OSS лицензи. Най-често използваните лицензи обаче са далеч по-малко на брой.<sup>25</sup>

Всичко това цели да покаже, че, от една страна, OSS лицензите са силно стандартизирани и до голяма степен се припокриват в задълженията, които предвиждат. От друга страна, задълженията, които съответният лиценз предвижда, могат да бъдат приемливи в един бизнес контекст и неприемливи в друг. Като се вземе предвид колко голямо е количеството софтуер с отворен код, използван днес в различни индустрии, неизбежно се стига до необходимостта да бъде изградена система, чрез която един компонент бива оценяван за неговата пригодност за ползване, в зависимост от лиценза му и съответния бизнес контекст, в който се планира да бъде използван.

## **СИСТЕМИ ЗА ОЦЕНКА НА OSS ЛИЦЕНЗИТЕ**

В случай, че една организация цели да използва софтуер с отворен код в свой продукт, логично се повдига въпросът как следва да се извърши оценката на съответните OSS лицензи, за да бъде взето решение дали съответният компонент може да бъде интегриран и при какви условия.

---

<sup>23</sup> BSD 4-Clause „Original“ or „Old“ License (<https://spdx.org/licenses/BSD-4-Clause.html>).

<sup>24</sup> <https://scancode-licensedb.aboutcode.org>

<sup>25</sup> Вж. например изследване на OSI за най-популярните OSS лицензи през 2023 г., разделени по програмен език: Vidal, N. The most popular licenses for each language in 2023 (<https://opensource.org/blog/the-most-popular-licenses-for-each-language-2023>).

Ако организацията не притежава голям опит с дейностите, свързани със софтуера с отворен код и неговите лицензи, процесът вероятно би изглеждал по следния начин:<sup>26</sup> екипът, разработващ софтуерния продукт, взема решение да използва определен компонент с отворен код. Екипът обръща внимание, че софтуерният компонент се предоставя под определен лиценз и решава да се допита до юрист, който да разясни дали е възможно използването на този компонент и ако да – какви условия трябва да бъдат спазени и какви задължения трябва да бъдат изпълнени.

Юристът, от своя страна, трябва да се запознае със съдържанието на лиценза и да разтълкува неговите клаузи. В този процес юристът би установил, че предвидените от лиценза задължения зависят от бизнес контекста, в който лицензираният софтуер ще бъде използван. Следователно той трябва, от своя страна, да се обърне към екипа и да събере фактите за това как точно ще се използва съответният компонент. Веднъж изпълнил тези две стъпки – подробна интерпретация на лиценза и анализ на фактите относно предвиденото използване на лицензирания софтуер – юристът може да направи заключение дали съответният компонент може да бъде използван в съответния контекст и какви задължения трябва да бъдат изпълнени.

Тук следва да бъде направено едно уточнение. Макар всички OSS лицензи да позволяват използването на лицензирания софтуер за всякакви цели, не всички бизнес контексти правят възможно изпълнението на лицензионните задължения. Така например, ако екипът разработва приложение, което е предвидено да бъде със затворен код, да работи на устройството на потребителя и да бъде предоставено под комерсиален лиценз на организацията, не би било възможно това приложение да ползва компонент под OSS лиценз, който изисква всякакви производни произведения да бъдат лицензирани под същия този OSS лиценз, тъй като това би довело до лицензионен конфликт. Изправена пред такава ситуация, организацията би следвало да избере между това да промени своя бизнес модел или просто да не използва съответния компонент, като по всяка вероятност по-лесният избор би бил вторият.

Ако процесът бъде представен като схема,<sup>27</sup> би изглеждал по следния начин:

---

<sup>26</sup> Този хипотетичен процес презюмира, че служителите в организацията са поне в минимална степен обучени относно някои от основните положения на авторското право, в т.ч. обстоятелството, че софтуерът е обект на авторскоправна закрила и използването на софтуер, чийто носител на авторското право е трето лице, се нуждае от правно основание, каквото е лицензът.

<sup>27</sup> Схемите в тази статия са създадени с draw.io (<https://github.com/jgraph/drawio>).

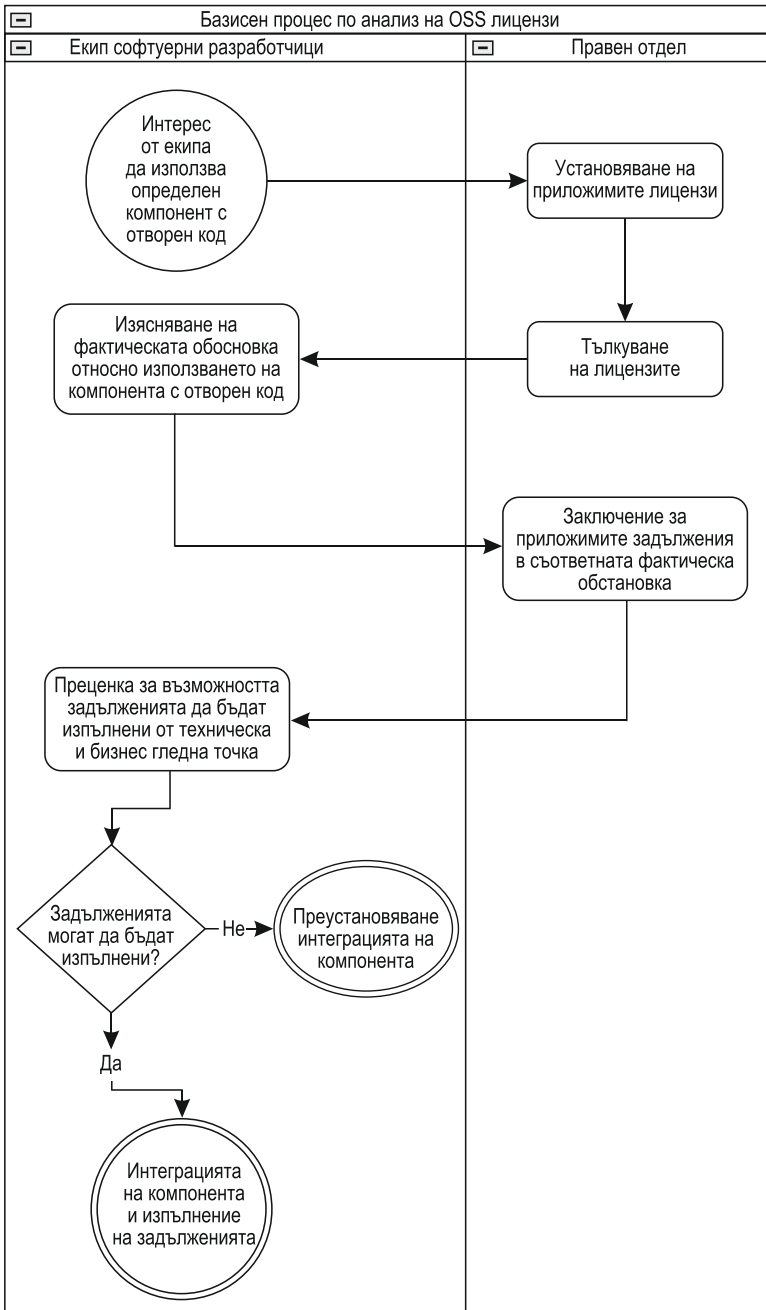


Схема 1: Базисен процес по анализ на OSS лицензи

Този процес би бил напълно разумен за една организация, която първо започва да използва софтуер с отворен код в своите продукти и трябва да изпълни предвидените в лицензите задължения, но не работи добре на по-големи мащаби. Той би работил, докато използваните компоненти с отворен код са сравнително малко на брой, но ако става въпрос за стотици или дори хиляди компоненти в различни продукти, би било изключително трудно и за софтуерния екип, и за правните специалисти да минават през него за всеки един компонент, с което се увеличава и възможността за човешка грешка. Като се прибавят и модерни практики за разработка на софтуер като CI/CD,<sup>28</sup> при които разработваният софтуерен продукт се променя често и нови версии се публикуват на кратки времеви интервали, което би могло да означава и промяна в използваните компоненти с отворен код, съвсем се достигат лимитите на процеса.

Тук проличава ясно необходимостта от алтернативен процес, който съответства както на мащабите на използваните компоненти с отворен код в софтуерните продукти, така и на модерните практики в разработката на софтуер. Това означава да е в голяма степен автоматизиран. Тази автоматизация ще трябва, първо, да вземе предвид съответния бизнес контекст; второ, да събере информация за всички използвани компоненти и техните лицензи; и трето, да приложи предварително зададени машинночетими правила към тези лицензи, като предостави информация на екипа кои лицензи могат да се използват, кои не следва да се използват и кои могат да се използват след консултация с правен експерт (например поради наличието на нестандартни задължения, чието изпълнение не може да бъде автоматизирано).

В този алтернативен процес ролята на юриста значително се променя. Неговата компетентност става релевантна в две ситуации: първо, при първоначалното тълкуване на лиценза и класификацията му в системата;<sup>29</sup> и второ, в случаите, в които конкретен лиценз предвижда такива задължения, които трудно биха могли да бъдат обработени автоматично, и следователно, за да се избегне създаването на правен риск, изискват човешка намеса.

Представен схематично, процесът би изглеждал по следния начин:

---

<sup>28</sup> Вж. <https://en.wikipedia.org/wiki/CI/CD>.

<sup>29</sup> В случая под „система“ следва да се разбира съвкупността от софтуерни инструменти, които се използват в процеса по автоматизация. Очакването е, че такава система поддържа функционалност за вхождане на конкретен лиценз под определена категория, като в този момент той става „познат“ на системата и за обработката му могат да се настроят различни машинночетими правила.

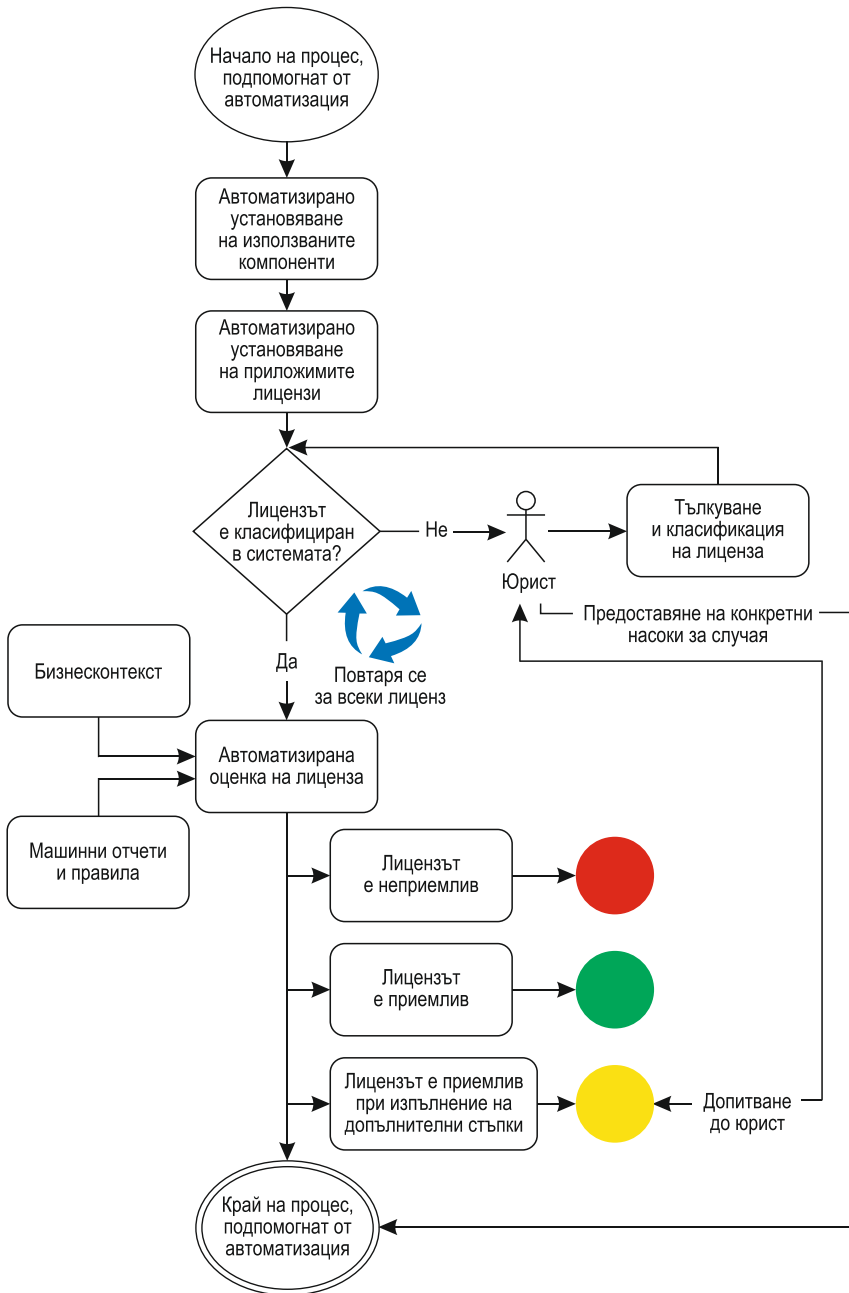


Схема 2: Процес по анализ на OSS лицензи, подпомогнат от автоматизация

Тук следва да се направят две уточнения. Първото от тях е, че при лицензи, които са приемливи при изпълнение на допълнителни стъпки, би могъл да се създаде още един филтър преди допитване до юриста, като например предварително зададени писмени насоки към екипа, упоменаващи как следва да действат, за да изпълнят задълженията на лиценза. Това, разбира се, зависи от степенята на търпимост на организацията към правен риск.

Второто уточнение е, че този процес се развива на етапа на разработката на софтуера и може да се повтаря толкова, колкото често екипът счита за необходимо. Резултат е поетапното изчистване на компоненти под неприемливи лицензи или използването на алтернативни такива преди софтуерът да напусне рамките на организацията и съответно да се задействат клаузите на лицензите, предвиждащи изпълнението на определени задължения.

Тази статия няма за основна цел да се фокусира върху конкретни технологични решения, които могат да бъдат използвани в такъв процес, а и е трудно да се говори за тях, без да се навлиза в технически детайли. Все пак такива съществуват както под формата на комерсиални продукти със затворен код, така и като софтуер с отворен код. От втората категория следва да се отбележи популярният *OSS Review Toolkit* (ORT)<sup>30</sup> проект, поддържащ както функционалност за анализ на използваните компоненти с отворен код и техните лицензи, така и за конфигуриране на машинночетими политики спрямо тези лицензи под формата на код. ORT анализира софтуера на нивото на сорс кода, като разчита на съответната система за управление на пакети (*package manager*),<sup>31</sup> за да установи кои са използваните компоненти с отворен код. За тяхното сканиране, с цел намиране на информация за приложимите софтуерни лицензи и носители на авторското право<sup>32</sup>, може да се използват други инструменти, като например ScanCode.<sup>33</sup>

---

<sup>30</sup> Вж. <https://github.com/oss-review-toolkit/ort>, публикуван под Apache-2.0 лиценз; вж. също съвърнатата имплементация на ORT, разработвана в рамките на Eclipse Aopsis (<https://github.com/eclipse-apoapsis>), също публикувана под Apache-2.0 лиценз.

<sup>31</sup> Вж. [https://en.wikipedia.org/wiki/Package\\_manager](https://en.wikipedia.org/wiki/Package_manager)

<sup>32</sup> Става въпрос за т.нар. Copyright Notice, който посочва носителя или носителите на авторското право. Може да бъде открит във файловете на съответния компонент с отворен код и чието възпроизвеждане е необходимо за изпълнение на задълженията, предвидени в съответните OSS лицензи.

<sup>33</sup> Вж. <https://github.com/aboutcode-org/scancode-toolkit>, публикуван под Apache-2.0 лиценз.

Анализът на сорс кода не е единственият възможен механизъм. Други инструменти могат да работят на различни принципи, като например чрез анализ на бинарни файлове.<sup>34</sup>

## ЗАКЛЮЧЕНИЕ

Използването на софтуер с отворен код в софтуерен продукт изисква спазването на приложимите лицензи. В една организация тази дейност следва да бъде формализирана в процес. Какъв точно ще бъде този процес зависи от нуждите на организацията и степента ѝ на търпимост към правен риск. При един по-базисен процес правният специалист би заемал една по-централна роля, взаимодействаща си тясно с екипа от софтуерни разработчици, за да изчисти правните въпроси, свързани с използването на компонента с отворен код, преди той да бъде интегриран в съответния продукт. Колкото по-автоматизиран става процесът, толкова по-периферна става ролята на правния специалист. Така например в илюстрирания в Схема 2 процес неговата роля е свързана с тълкуване и класифициране на непознати на системата лицензи, както и за съдействие в случаите, в които не може да бъде направена автоматизирана оценка на пригодността на определен лиценз, а оттам – и на лицензирания компонент да бъде използван в съответния бизнес контекст.

В Схема 2 проличават също границите на автоматизацията. Въпреки високата степен на стандартизация на OSS, при лицензите, в зависимост от съответния бизнес контекст все пак е необходима човешка намеса. Относно тези случаи най-добротото, което може да се очаква от автоматизацията, е да ги идентифицира и да насочи към тях вниманието на съответните специалисти.

С напредъка на технологиите неизбежно ще се променят правно-релевантните процеси, което влече промени и в ролята на юриста. И въпреки че модерните процеси, подпомогнати в голяма степен от автоматизация, ще намалят случаите, в които е необходима човешка намеса, те същевременно ще поставят акцента върху ценния принос на експерта, в т.ч. правния експерт, в справянето с нови и нестандартни ситуации, които засяга остават откъд способностите на софтуерните технологии.

---

<sup>34</sup> Вж. [https://en.wikipedia.org/wiki/Binary\\_file](https://en.wikipedia.org/wiki/Binary_file)