



# **Open Source Compliance:**

## **Open Source Lizenzoptionen**

vom 17.12.2021

überarbeitet am 20.12.2021

<b>1. ZIEL – OPEN SOURCE POLICY:</b> .....	<b>2</b>
<b>2. HERANGEHENSWEISE UND AUFBAU DIESER STELLUNGNAHME:</b>	<b>3</b>
<b>3. OSS AUS DER SICHT DES ANBIETERS:</b> .....	<b>3</b>
3.1. <i>Die Idee einer Open Source Policy:</i> .....	3
3.2. <i>Parameter, die in die Kategorisierung einfließen:</i> .....	4
3.3. <i>OSS-Matrix :</i> .....	7
3.4. <i>Empfehlung:</i> .....	8
3.5. <i>Exkurs: Software der Kommission:</i> .....	9
3.6. <i>OSS in der Entwicklung:</i> .....	9
3.7. <i>OSS bei Weitergabe:</i> .....	10
<b>4. OSS AUS DER SICHT DES ANWENDERS:</b> .....	<b>10</b>
4.1. <i>Haftungs- und Gewährleistungsfragen:</i> .....	10
4.2. <i>Vertragliche Absicherung:</i> .....	11
4.3. <i>Technische Absicherung:</i> .....	11
<b>5. EXECUTIVE SUMMARY:</b> .....	<b>11</b>

## 1. Ziel – Open Source Policy:

Der Einsatz von Open Source Software (in der Folge „OSS“) etabliert sich zunehmend in der Softwarebranche. Mit der Digitalisierung des Alltags nimmt der Einsatz von OSS zusätzlich zu. Nach Erhebungen des Scantool-Anbieters „Synopsys“ basiert **durchschnittlich über 57 % einer Codebasis auf OSS!**<sup>1</sup> Über 84 % der Unternehmen geben an, dass OSS in wichtigen bzw unternehmenskritischen Bereichen eingesetzt wird.<sup>2</sup> Das zeigt deutlich, dass die Open Source Bewegung unaufhaltsam ins Rollen gekommen ist und mittelfristig die Softwarebranche massiv beeinflussen wird. Eindrucksvoll bringt dies die Studie „*The State of Enterprise on Open Source 2020*“<sup>3</sup> des Unternehmens “Red Hat” zum Ausdruck. Demnach sinkt die Bedeutung von proprietärer Software signifikant. Waren im Jahr 2019 noch 55 % der genutzten Software in den befragten Unternehmen proprietär, sank dieser Wert im Jahr 2020 in den befragten Unternehmen auf 42 %. Der Anteil an Open Source Software stieg hingegen in diesem Zeitraum von 36 % auf 44 %. Studien zeigen weiters, dass Investitionen in Open Source Software einen positiven wirtschaftlichen Effekt haben.<sup>4</sup>

Open Source wird als Möglichkeit gesehen, Softwarelösungen, Wissen und Expertise zu teilen. Dies wiederum kann zu einer Kostenreduktion und Bereicherung der Gesellschaft beitragen. Die Europäische Union sieht den Einsatz von OSS aus diesen Gründen als **äußerst positiv**.<sup>5</sup> Die Europäische Union sieht Open Source Software weiters als wesentlichen Beitrag zur Erreichung einer digitalen Souveränität. Gerade im Bereich der öffentlichen Verwaltung werden vielfältige Einsatzmöglichkeiten diagnostiziert. Konkret nahm die Kommission am 21.10.2020 eine neue Strategie für OSS an, mit der die Verwendung quelloffener Software gefördert wird.<sup>6</sup> Am 8.12.2021 wurde der Beschluss der Kommission über die Open-Source-Lizenzierung und die Weiterverwendung von Software der Kommission verlautbart.

Mit der steigenden rechtlichen Relevanz von OSS rückt die Frage in den Fokus, wie OSS **rechtskonform** implementiert und weitergegeben werden kann. Die Gestaltung einer **Open Source Policy** ist dabei als zentrales Element einer Open Source Compliance zu nennen.

---

<sup>1</sup> Vgl Jaeger/Metzger, Open Source Software, 5. Auflage, S 20.

<sup>2</sup> Vgl Suchomski in Schneider, Handbuch EDV-Recht, S 1131.

<sup>3</sup> <https://www.redhat.com/en/enterprise-open-source-report/2020> (Stand 7.4.2020). Es wurden 950 IT-Führungskräfte befragt.

<sup>4</sup> Vgl Studie: *The impact of Open Source Software and Hardware on technological independence, competitiveness and innovation in the EU economy*, wonach: “On a cumulative basis, the study estimates that, up to 2018, the contribution of OSS to EU GDP, and contributions of EU employees to OSS, yield a cost-benefit ratio of slightly above 1:10. After taking into account hardware and other capital costs of the 260,000 EU contributors to OSS, the cost-benefit ratio is still slightly above 1:4” (S. 16).

<sup>5</sup> Vgl European Commission, 21.10.2020, Open Source Software Strategy 2020-2023.

<sup>6</sup> Vgl Mitteilung an die Kommission, Open-Source-Software-Strategie 2020-2023, Offen Denken, 21.10.2020, C (2020) 7149 final.

Die Open Source Policy soll eine **Orientierung** bieten. Orientierung für Entscheidungsträger im Management, den Einkauf, IT-Sicherheitsbeauftragte und vor allem Softwareentwickler.

## 2. Herangehensweise und Aufbau dieser Stellungnahme:

Bei Gestaltung der Open Source Policy sollen zwei verschiedene Positionen eingenommen werden. Jene des (Software-)Anbieters einerseits und jene des (Software-)Anwenders andererseits.

Zunächst werden aus der Sicht des Anbieters die gängigsten OSS-Lizenzbedingungen im Sinne eines „**Ampel-Systems**“ einer von drei verschiedenen Kategorien zugeordnet. Jeder dieser Kategorien löst einen unterschiedlichen **Prozess** aus. Eine wichtige Rolle spielen dabei die **Parameter**, die in die Kategorisierung einfließen.

Schließlich wird zu differenzieren sein, ob die OSS ausschließlich intern eingesetzt oder diese auch an Dritte **weitergegeben** werden soll. Auch bei der internen Entwicklung ist zu berücksichtigen, wie mit bereits existierenden OSS-Komponenten umzugehen ist.

Abschließend wird die Perspektive gewechselt: Aus der Sicht eines Anwenders wird aufgezeigt, was dieser beim Erwerb von OSS zu beachten hat. Hier wird beschrieben, wie dieser bestmöglich eine **rechtliche Absicherung** beim Einsatz von OSS erreichen kann.

## 3. OSS aus der Sicht des ANBIETERS:

### 3.1. Die Idee einer Open Source Policy:

Die Funktion einer Open Source Policy ist, den involvierten Akteuren (Stakeholdern), hier insbesondere den Entwicklern, eine Orientierung zu bieten. Eine **Open Source Policy**<sup>7</sup>, oder auch Open Source-Richtlinie, regelt, welche OSS die Entwickler unter welchen Voraussetzungen und in welcher Form einsetzen können (und welche nicht).<sup>8</sup> Ohne sich tiefergehend mit der Materie auseinandersetzen zu müssen, erhalten die Stakeholder eine **Handlungsempfehlung** was zu geschehen hat, wenn OSS-Komponenten unter einer bestimmtem Vertragswerk (zB MIT oder GNU GPL-Lizenz) lizenziert sind.

Die Open Source Policy manifestiert dabei drei Aspekte: (i) Die **Parameter** (vgl Punkt 3.2), die in die (ii) **OSS-Matrix** (vgl Punkt 3.3) einfließen und (iii) die **Prozesse**, die sich aus dieser Kategorisierung ableiten.

---

<sup>7</sup> Einer Studie von Bitkom<sup>7</sup> zufolge verfügen rund 20 % der befragten Unternehmen über eine Open Source Policy.

<sup>8</sup> Vgl *Fischer/Hilber* in Intveen/Gennen/Karger, Handbuch des Softwarerechts, S 361.

Mögliche Prozesse wären, dass eine Qualifikation in die Kategorie „*restriktiv*“ zur Folge hat, dass Komponenten unter gewissen Open Source Lizenzbedingungen nur nach Rücksprache mit der Geschäftsleitung (Vorstand, Geschäftsführung) implementiert werden dürfen. Eine Qualifikation unter die Kategorie „*liberal*“ hingegen würde dazu führen, dass die Entwickler diese Komponenten auch ohne weitere Rücksprache implementieren dürfen.<sup>9</sup>

**Wichtiger Hinweis I:** Ein Denken in Schemen und Klassifizierungen führt nicht immer zu den besten Ergebnissen. Daher sollte die erste Frage stets sein: Soll anlassbezogen von dieser Kategorisierung abgewichen werden?

**Wichtiger Hinweis II:** Ein anlassbezogener Ansatz ist weiters deshalb geboten, da nicht sämtliche Softwareprojekte für OSS geeignet sind. Als Beispiel sind etwa Softwareanwendungen im Bereich des Militärs zu nennen, wo eine Vertraulichkeit des Quellcodes regelmäßig geboten ist.<sup>10</sup>

**Wichtiger Hinweis III:** Die Praxis zeigt, dass eine Policy alleine keinen Effekt erzielt. Wichtig ist, dass die Policy mit einem Prozess kombiniert wird, sodass deren Inhalt an den relevanten Stellen (insbesondere bei den Entwicklern) tatsächlich „ankommt“.

### 3.2. *Parameter, die in die Kategorisierung einfließen:*

Aus meiner – rechtlich eingefärbten – Sicht<sup>11</sup> sollten folgende Parameter bei der Kategorisierung der Open Source-Lizenzbedingungen anlassbezogen<sup>12</sup> berücksichtigt werden:

- Die „Intensität“ des **Copylefts**: Der Copyleft-Effekt<sup>13</sup> ist eine Klausel, die sicherstellt, dass Weiterentwicklungen der Software **unter denselben Bedingungen** der Lizenz wieder freigegeben werden. Dahinter steht ein ganz wesentliches Grundprinzip, welches dafür sorgen soll, dass geänderte Open Source Software innerhalb des Open Source Anwendungsbereichs verbleiben und nicht als kommerzielle, proprietäre Software vertrieben werden können. Der Copyleft ist insofern eine Schutzklausel, die sicherstellt, dass Weiterentwicklungen einer

---

<sup>9</sup> Was aber nichts daran ändert, dass deren Einsatz dokumentiert und beim Vertrieb berücksichtigt werden muss.

<sup>10</sup> Vgl in diesem Zusammenhang Artikel 4 lit b des Beschlusses der Kommission vom 8.12.2021 über die Open-Source-Lizenzierung und die Weiterverwendung von Software der Kommission. Demnach gilt dieser Beschluss nicht für Software der Kommission, wenn die Veröffentlichung oder gemeinsame Nutzung ihres Quellcodes ein tatsächliches oder potenzielles Risiko für die Sicherheit der Informationssysteme oder Datenbanken der Kommission oder anderer Organe, Einrichtungen oder sonstiger Stellen der EU darstellen würde.

<sup>11</sup> Damit soll zum Ausdruck gebracht werden, dass auch bspw die Meinung eines Softwareentwicklers evaluiert und gegebenenfalls berücksichtigt werden sollte.

<sup>12</sup> Damit soll zum Ausdruck gebracht werden, dass eine Einstufung in die Stufe 3 „*restriktiv*“ keinesfalls als „schlecht“ interpretiert werden darf. Evtl ist die Implementierung einer Open Source Komponente unter bspw die GNU GPL 3.0. geboten oder aber legt der Entwickler Wert darauf, dass die entwickelten, abgeleiteten Komponenten ebenfalls diesen Open Source Lizenzbedingungen unterliegen.

<sup>13</sup> Das originelle Wortspiel Copyleft als Gegenstück zum Copyright wird übrigens *Richard Stallman* zugeschrieben.

Software unter denselben Bedingungen der Open Source Lizenz wieder freigegeben werden.<sup>14</sup> Die Intention solcher Copyleft-Klauseln liegt darin, die freie Nutzbarkeit der Software auch für weiterentwickelte Versionen sicherzustellen.<sup>15</sup> Diese grundsätzliche Entscheidung kann im Rahmen einer Open Source Strategie festgelegt werden.

- Die „Verträglichkeit“ mit anderen Lizenzen („**Lizenzkompatibilität**“): Eine Inkompatibilität ist bei der Anwendung mehrerer Copyleft-Lizenzen dann anzunehmen, wenn eine abgeleitete Arbeit unter mehreren **sich widerstreitenden Copyleft-Lizenzen** eingesetzt wird. Beispielsweise gelangt die *Free Software Foundation* zu der Ansicht, dass die BSD-4-Clause mit der GPL-2.0. inkompatibel ist.<sup>16</sup> Zwei der häufigsten Open Source Lizenzen, nämlich Apache-2.0 und GPL dürfen nach der Einschätzung der Lizenzgeber ebenfalls nur beschränkt in ein und demselben Projekt eingesetzt werden. Daher ist für jeden Einzelfall zu prüfen, ob eine Open Source-Lizenz die Nutzung einer anderen Open Source-Lizenz gestattet.<sup>17</sup> Schließlich wäre es unzulässig, Softwarekomponenten so zu verbinden, dass sie gleichzeitig unter zwei sich widersprechenden Bedingungen lizenziert sein müssen.<sup>18</sup>
- OSS und **Patentrecht**: „Patente“ sind dadurch charakterisiert, dass sie „neu und geheim“<sup>19</sup> sind, weshalb deren vertrauliche Behandlung höchste Priorität zukommt. Das dies nur schwer in Einklang mit der „Philosophie“ von Open Source gebracht werden kann, ist offensichtlich. Die Open Source Community hegt daher eine grundsätzlich ablehnende Haltung gegenüber dem Patentwesen.<sup>20</sup> Diese kritische Haltung manifestiert sich deutlich in der Präambel der GPL-2.0: „*Schließlich und endlich ist jedes freie Programm permanent durch Softwarepatente bedroht*“.
- OSS und **Tivoisierung**: Inhaber von geistigen Rechten an Software stehen vor der Herausforderung, ihr digitales Eigentum vor unberechtigten Eingriffen zu schützen. Im Zuge des Digital Rights Managements werden daher technische Maßnahmen umgesetzt, um unberechtigte Nutzungen zu verhindern. Unter Tivoisierung versteht man einen Aspekt des Digital Rights Managements, nämlich **technische Schutzmaßnahmen** für Software, welche die Austauschbarkeit von Software-Komponenten beschränken.<sup>21</sup> Das Verbot der Tivoisierung wird daher auch als „DRM-Verbot“ bezeichnet. Da derartige Maßnahmen die Nutzung von Software beschränken, stehen sie in einem Spannungsverhältnis zum Einsatz von OSS. So regelt

---

<sup>14</sup> Vgl *Galetzka* in *Galetzka/Jun/Roßmann*, *Praxishandbuch Open Source*, S 76.

<sup>15</sup> Vgl *Jaeger* in *Redeker*, *Handbuch IT-Verträge*, 1.20., S 5.

<sup>16</sup> Vgl *Galetzka* in *Galetzka/Jun/Roßmann*, *Praxishandbuch Open Source*, S 128.

<sup>17</sup> Vgl *Völkel/Kremer* in *Intveen/Gennen/Karger*, *Handbuch des Softwarerechts*, S 440.

<sup>18</sup> Vgl *Jaeger* in *Redeker*, *Handbuch IT-Verträge*, 1.20., S 8.

<sup>19</sup> Vgl § 3 Patengesetz.

<sup>20</sup> Vgl *Jaeger/Metzger*, *Open Source Software*, S 241.

<sup>21</sup> Vgl *Hackel* in *Galetzka/Jun/Roßmann*, *Praxishandbuch Open Source*, S 258.

etwa die GPL-3.0, dass Software nicht dazu eingesetzt werden soll, um Maßnahmen des Digital Right Managements zu fördern.

- Risiko der **Rechtsverfolgung**: Obgleich die Rechtsdurchsetzung von Open Source Lizenzbedingungen – vor allem in Österreich – noch in ihren Kinderschuhen steckt, kann konstatiert werden, dass gerade im Bereich des Einsatzes des Linux Kernels und von WordPress-Plug-Ins die Wahrscheinlichkeit einer rechtlichen Auseinandersetzung zunimmt. „Streitgeneigte“ Lizenzen sind daher mit größerer Vorsicht zu genießen.
- **SaaS- bzw ASP-Kompatibilität**: Ein praktisch immer wichtigeres Kriterium ist, ob die Software als SaaS- bzw ASP-Anwendung vertrieben werden darf. In einigen moderneren Lizenzversionen, wie der GPL-3.0, AGPL-3.0, LGPL.3.0 und der MPL-2.0, wird SaaS speziell berücksichtigt und in allen genannten Lizenzen auch entsprechende Nutzungsrechte eingeräumt. Ältere Lizenzen jedoch erwähnen eine SaaS bzw ASP-Anwendung hingegen nicht. Dies ist damit zu begründen, dass diese Technologie erst Mitte/Ende der 1990er Jahre wirtschaftlich an Relevanz gewann.<sup>22</sup> Ob diese älteren OSS Lizenzbedingungen eine Nutzung via ASP- bzw SaaS gestatten, ist durch tiefere Recherche sowie durch eine ergänzende Vertragsauslegung zu klären. Teilweise wird gar die Meinung vertreten, dass die cloudbezogene Verwendung von OSS in proprietärer Software keinen Copyleft auslöst und auch dem Anwender keine weiteren Pflichten der Software in der Cloud auferlegt werden dürfen.<sup>23</sup>
- **Rechtswahlklausel und Gerichtsstand**: Wenngleich die Wirksamkeit, gerade von Gerichtsstandvereinbarungen, in Lizenzbedingungen generell als fraglich beurteilt werden kann, sollte auch dieser Aspekt berücksichtigt werden. Einige Lizenzbedingungen, etwa die „CeCILL“, die „deutsche Frei Software Lizenz“ oder die „EURL“ sehen Rechtswahl- und Gerichtsstandsklauseln dezidiert vor.
- Spezialfall „**Software der Kommission**“: Hinsichtlich „Software der Kommission“ ist der Beschluss der Kommission vom 8.12.2021 über die Open-Source-Lizenzierung und die Weiterverwendung von Software der Kommission zu beachten. Als Software der Kommission ist dabei zu verstehen<sup>24</sup>:
  - Software, an der die Kommission im Namen der Union die Rechte des geistigen Eigentums besitzt und

---

<sup>22</sup> Vgl Jaeger/Metzger, Open Source Software, S 140.

<sup>23</sup> Vgl Völkel/Kremer in Intveen/Gennen/Karger, Handbuch des Softwarerechts, S 434.

<sup>24</sup> Vgl Art 2 des Beschlusses der Kommission vom 8.12.2021 über die Open-Source-Lizenzierung und die Weiterverwendung von Software der Kommission

- Software im Besitz eines Dritten, die im Rahmen einer Open-Source-Lizenz zur Verfügung steht und von der Kommission oder einem Dritten auf Verlangen der Kommission geändert wurde.

### 3.3. OSS-Matrix :

Unter Berücksichtigung der unter Punkt 3.2 dargestellten Parameter ergibt sich bei Anwendung eines Ampel-Systems nachstehende **OSS-Matrix** (aus der Sicht eines Anbieters).<sup>25</sup> Die Kategorisierungen reichen dabei von „restriktiv“, über „neutral“ bis hin zu „liberal“<sup>26,27</sup> Die jeweilige Kategorisierung löst dann wiederum, wie in Punkt 3.1 beschrieben, einen gewissen, noch zu definierenden, Prozess aus.

#### **Kategorie „restriktiv“:**

- GPL (GNU General Public License v.2.0/3.0)
- AGPL (GNU Affero General Public License v1.0/3.0)
- EPL-1.0 (Eclipse Public License)
- European Union Public License
- Deutsche Freie Software Lizenz
- Open Software License

#### **Kategorie „neutral“:**

- Mozilla Public License
- Common Development and Distribution License
- GNU Lesser General Public License
- Microsoft Public License
- EPL-2.0 (Eclipse Public License)<sup>28</sup>
- Apache-2.0<sup>29</sup>
- LGPL (GNU Lesser General Public License)

#### **Kategorie „liberal“<sup>30</sup>:**

- BSD License (2-Clause, 3-Clause)

---

<sup>25</sup> Vgl *Jaeger/Metzger*, Open Source Software, S 24 ff; Vgl *Jun/Roßmann* in Galetzka/Jun/Roßmann, Praxishandbuch Open Source, S 348; *Meeker*, Open Source for Business; *Suchomski* in Schneider, Handbuch EDV-Recht, S 1140; Vgl *Jaeger* in Redeker, Handbuch IT-Verträge, 1.20., S 5.

<sup>26</sup> **Wobei „liberal“ keinesfalls bedeutet, dass rechtliche Aspekte (Offenlegungspflichten, Copyrightvermerk etc) bei einer Weitergabe übersehen werden dürfen.**

<sup>27</sup> Bewusst wurde von einer Kategorisierung *à la* Rot/Gelb/Grün oder Gefährlich/Mittel/Ungefährlich abgesehen. Eine solchen Kategorisierung würde nämlich eine (unterschwellige) Bewertung auslösen. Doch gibt es gegenständlich kein „gut“ und kein „schlecht“. Entscheidend ist vielmehr, welcher Zweck angestrebt werden soll.

<sup>28</sup> Im Gegensatz zu Version 1.0. muss nicht jede Bearbeitung unter EPL-2.0 lizenziert werden, sondern nur insoweit, als diese unter die Definition von „Modified Work“ fällt.

<sup>29</sup> Hintergrund dieser Bewertung ist ein Lizenzkonflikt mit GPL.

<sup>30</sup> Im Falle einer intendierten ASP- oder SaaS-Anwendung sollte im Vorfeld auch in diesen Fällen recherchiert werden, ob Argumente ersichtlich sind, dass diese OSS Lizenzbedingungen eine derartige Nutzung verbieten.

- MIT-License
- Zlib License
- Universal Permissive License

#### 3.4. *Empfehlung:*

Auf die Frage, welche Open Source Lizenzbedingungen zu empfehlen sind, muss mit einer Gegenfrage geantwortet werden: *Was ist dem Entwickler der Software wichtig?* Es kann gar nicht stark genug betont werden: **Die Auswahl der Open Source Lizenz hängt primär davon ab, welche Ziele der Entwickler, der Auftraggeber bzw künftige Inhaber der Werknutzungsrechte („Lizenzgeber“) verfolgt.**

Geht es ihm darum, die OSS möglichst zu verbreiten, dann sind Open Source Lizenzbedingungen mit einem starken Copyleft, wie etwa die GNU GPL 3.0, zu empfehlen.

Soll hingegen sowohl der Anbieter als auch der Anwender möglichst viel Freiraum haben, so ist in Richtung der „permissiven Lizenzen“ zu tendieren wie etwa den Lizenzbedingungen: „MIT“.

Die MIT Lizenz wurde vom *Massachusetts Institute of Technology* entworfen und ist mit ca 26 % die meistgenutzte Open Source Lizenz. Die MIT Lizenz ist eine einfache Non-Copyleft-Lizenz. Die Rechteklausel ist sehr weitgehend formuliert („*to deal in the Software without restriction, including without limitation to use,...*“) und zeigt klar die Intention des Lizenzgebers, jegliche Art der Nutzung zu gestatten.<sup>31</sup>

Grundsätzlich spricht auch nichts dagegen, eigene, nach seinen Vorstellungen „maßgeschneiderte“, Lizenzen zu kreieren. Einen Überblick über mögliche Lizenzen kann dem Lizenzcenter des ifrOSS<sup>32</sup> entnommen werden. Dabei ist jedoch darauf hinzuweisen, dass etwa 93 % der gesamten verfügbaren Open Source Software unter lediglich zehn verschiedenen Lizenzen angeboten werden.<sup>33</sup> Letztlich ist aber die Eigenkreation von (vor allem nationalen) Open Source Lizenzbedingungen aber kritisch zu betrachten, da diese mangels Bekanntheitsgrad **nur zurückhaltend oder gar nicht akzeptiert** werden.

---

<sup>31</sup> Vgl Jaeger/Metzger, Open Source Software, S 92.

<sup>32</sup> Vgl <https://ifross.github.io/ifrOSS/Lizenzcenter> (abgerufen am 30.11.2021).

<sup>33</sup> Vgl Jaeger/Metzger, Open Source Software, S 24.



### 3.5. *Exkurs: Software der Kommission:*

Hinsichtlich Software der Kommission<sup>34</sup> sollte<sup>35</sup> grundsätzlich eine Lizenzierung unter die „**EUPL**“, die European Union Licence in der aktuellen Version (Version 1.2) sowie allen künftigen Versionen der Kommission erfolgen.

Die EUPL wurde 2007 von der Europäischen Kommission veröffentlicht. Sie liegt mittlerweile in der Version 1.2 aus dem Jahr 2016 vor. Sie wurde von der IDABC („*Interoperable Delivery of European eGovernment to public Administrations, Business and Citizens*“), einem Programm der Europäischen Gemeinschaft entwickelt. Die bekannten Open Source Lizenzen wurden von der IDABC als unzureichender verworfen, weil keine offiziellen Sprachfassungen existieren und die Ausrichtung auf das US-Recht als unpassend empfunden wurde bzw als rechtlich problematisch, was etwa die Klauseln für den Haftungsausschluss betrifft.<sup>36</sup>

Die EUPL ist als **strenge Copyleft-Lizenz** konzipiert.<sup>37</sup>

### 3.6. *OSS in der Entwicklung:*

Im Bereich der Entwicklung von Software muss sich der Entwickler zunächst eine Übersicht verschaffen, ob bereits Open Source Komponenten eingesetzt werden. Dabei sollte er sich nicht nur auf die Aussagen der Entwickler verlassen, sondern zusätzlich Scanner wie etwa von *fossology*, *BlackDuck Protex*, *White Source* oder *Reverera* einsetzen.<sup>38</sup>

Wenn nun Komponenten eingesetzt worden sein sollten, die unter einer Lizenz mit starkem Copyleft stehen und die selbst entwickelten Software-Elemente – stark vereinfacht dargestellt – eng mit diesen Komponenten „verzahnt“ sind, so ist der Entwickler in seiner Auswahl der Lizenz gar nicht mehr frei. Vielmehr müssten die Lizenzbedingungen der implementierten Open Source Komponenten übernommen werden (sogenannter „*viraler Effekt*“).<sup>39</sup>

Mit anderen Worten: Die Frage, unter welche Open Source Lizenzbedingungen der Anbieter die entwickelte Software stellen möchte, kann **sich immer nur auf die selbst entwickelten Software-Elemente** beziehen.

---

<sup>34</sup> Zu diesem Begriff siehe Punkt 3.2..

<sup>35</sup> Vgl Art 5 des Beschlusses der Kommission vom 8.12.2021 über die Open-Source-Lizenzierung und die Weiterverwendung von Software der Kommission.

<sup>36</sup> Vgl *Jaeger/Metzger*, Open Source Software, S 309.

<sup>37</sup> Vgl *Jaeger/Metzger*, Open Source Software, S 309.

<sup>38</sup> Vgl *Jun* in Galetzka/Jun/Roßmann, Praxishandbuch Open Source, S 145.

<sup>39</sup> Um einen solchen „viralen Effekt“ zu vermeiden, sollte bereits präventiv auf eine technische Trennung dieser „infektiösen Bestandteile“ (etwa im Wege einer Modularisierung; dynamischer Links etc), geachtet werden.

### 3.7. OSS bei Weitergabe:

Ausdrücklich ist darauf hinzuweisen, dass die **bloße Nutzung** von OSS keine Verpflichtungen des Anwenders auslösen. Diese werden erst dann schlagend, wenn diese Software **vertrieben/weitergegeben** wird.<sup>40</sup> Eine Verbreitung findet nach herrschender Meinung statt, wenn ein Vervielfältigungsstück der Software in der Öffentlichkeit angeboten oder in Verkehr gebracht wird. Für ein „Inverkehrbringen“ reicht die Überlassungen an einen einzelnen Dritten aus.<sup>41</sup>

Die erste Frage im Zuge der Open Source Compliance sollte daher immer sein, ob überhaupt eine Intention besteht, die Software weiterzugeben. Eine Verbreitung von Open Source Software an ein konzernangehöriges Unternehmen stellt nach **überwiegender**, wohl aber nicht unbestrittener, **Ansicht** im Übrigen eine pflichtauslösende Weitergabe dar.<sup>42</sup>

## 4. OSS aus der Sicht des ANWENDERS:

### 4.1. Haftungs- und Gewährleistungsfragen:

Verstöße gegen Open Source Lizenzen sind als Urheberrechtsverletzung zu qualifizieren<sup>43</sup>. Unterlassungsansprüche, Beseitigungsansprüche, ein Anspruch auf angemessenes Entgelt, Schadenersatzansprüche bis hin zu Urteilsveröffentlichungen oder strafrechtlichen Sanktionen sind die möglichen Konsequenzen. Der lizenzwidrige Vertrieb führt nicht nur zu Ansprüchen der Urheber und Rechtsinhaber, sondern auch zu wettbewerbsrechtlichen Abwehransprüchen der Konkurrenten.

Nach ganz herrschender Meinung handelt es sich bei der Einräumung von Open Source Lizenzen um einen Lizenzvertrag mit **schenkungsvertragsrechtlichen Elementen**, wenn die Open Source Software unentgeltlich überlassen wird.<sup>44</sup> Dies hat vor allem Auswirkungen für das Gewährleistungsrecht und Haftungsregelungen. Das Gewährleistungsrecht nach den §§ 922 ff ABGB bezieht sich ausdrücklich nur auf entgeltliche Verträge – „*einem geschenkten Gaul schaut man nicht ins Maul*“.<sup>45</sup> Allerdings kann der Open Source Lizenzgeber dann zur Haftung herangezogen werden, wenn er wissentlich eine fremde Sache verschenkt (vgl § 945 ABGB). Zudem kann eine Haftung im Falle groben Verschuldens<sup>46</sup>, jedenfalls aber bei Vorsätzlichkeit angenommen werden.<sup>47</sup>

---

<sup>40</sup> Vgl *Kulke* in Galetzka/Jun/Roßmann, Praxishandbuch Open Source, S 62.

<sup>41</sup> Vgl *Jaeger/Metzger*, Open Source Software, S 121.

<sup>42</sup> Vgl *Breunung/Roßmann* in Galetzka/Jun/Roßmann, Praxishandbuch Open Source, S 169.

<sup>43</sup> Vgl LG München I, 19.5.2004, 21 O 6123/04.

<sup>44</sup> Vgl *Jaeger/Metzger*, Open Source Software, S 200; *Kulke* in Galetzka/Jun/Roßmann, Praxishandbuch Open Source, S 68.

<sup>45</sup> An dieser Stelle sei darauf hingewiesen, dass die weitgehenden Gewährleistungs- und Haftungsausschlüsse in Open Source Lizenzbedingungen regelmäßig nicht mit den Bestimmungen des §§ 864a und 879 Abs 3 ABGB in Einklang gebracht werden können (vgl ua *Suchomski* in Schneider, Handbuch EDV-Recht, S 1174).

<sup>46</sup> Vgl *Bollenberger/P. Bydlinkski*, ABGB-Kurzkommentar, 6. Auflage, S 1194.

<sup>47</sup> Vgl *Jaeger/Metzger*, Open Source Software, S 208.

#### 4.2. Vertragliche Absicherung:

Wie oben unter Punkt 4.1 dargestellt, kann die Nichtbeachtung von Open Source Lizenzbedingungen eine Urheberrechtsverletzung begründen. Die Konsequenz daraus ist, dass **Open Source Compliance sowohl bei der Entwicklung von Software als auch bei deren Anschaffung bzw Anmietung** beachtet werden sollte. Auf Seiten des Schadenspotentials sollte kalkuliert werden, welche Auswirkungen ein erfolgreich geltend gemachter Unterlassungsanspruch hätte. Ein solcher Anspruch sorgt bis zum Austausch der Software mindestens zu einem Auslieferungsstopp, unter Umständen auch zu einem Produktrückruf.

Unterlassungs-, Beseitigungs- und Bereicherungsansprüche bestehen dabei **verschuldensunabhängig**. Das bedeutet, dass die Nutzer einer Software belangt werden können, obgleich diese von der rechtswidrigen Nutzung der Software keine Kenntnis haben! Einem Anwender ist daher anzuraten, vor Anschaffung (bzw Anmietung) einer Lizenz der Software genau zu prüfen, ob die Software allfälligen Open Source Lizenzen Genüge tut. Eine Möglichkeit wäre dabei, den Lieferanten dazu verpflichten, vor der Weitergabe der Software einen Open-Source-konformen Zustand herzustellen. Werden daher Software-Produkte erworben, so ist es sinnvoll, in den Vertrag entsprechende Klauseln bezüglich Open Source Software aufzunehmen.<sup>48</sup> Dabei ist jedoch darauf hinzuweisen, dass von der Rechtsprechung ein strenger Maßstab angelegt wird, die auch eine **Prüfungs- und Erkundigungspflicht** beinhaltet. In diesem Zusammenhang ist auf die (deutsche) Rechtsprechung hinzuweisen, wonach: „...ein Anbieter sich nicht auf die Zusicherung seiner Lieferanten verlassen darf, die Ware verletze keine Rechte Dritter“.<sup>49</sup>

#### 4.3. Technische Absicherung:

Um auf der sicheren Seite zu sein, ist daher zu empfehlen, neben einer vertraglichen Absicherung im Sinne des Punktes 4.2 auch entsprechende technische Due-Diligence-Maßnahmen umzusetzen. Diese Due-Diligence-Maßnahmen können von einer Befragung der Entwickler bis hin zum Einsatz spezieller Scanner-Software (siehe dazu Punkt 3.6) reichen.

### 5. Executive Summary:

Sowohl Statistiken als auch die Praxis machen deutlich, dass Open Source Software zunehmend an Relevanz gewinnt (siehe Punkt 1). Mit deren steigenden Bedeutung gerät auch die Frage der rechtlichen Compliance verstärkt in den Fokus.

---

<sup>48</sup> Vgl *Kendziorra* in *Galetzka/Jun/Roßmann*, *Praxishandbuch Open Source*, S 153

<sup>49</sup> *LG Hamburg*, CR 2013, 498.

Eine Open Source Policy bildet dabei einen wichtigen Pfeiler. Anhand spezifischer Parameter (siehe Punkt 3.2) werden die gängigsten Open Source Lizenzbedingungen in eine von drei Kategorien eingestuft (siehe Punkt 3.3). Die so erstellte Open Source Matrix soll eine schnelle Übersicht ermöglichen.

Wünscht der Anbieter von OSS eine möglichst liberale Anwendung und Implementierung der von ihm entwickelten Software, ist eine Lizenzierung unter die Lizenz „MIT“ empfehlenswert (siehe Punkt 3.4). Im Falle des Einsatzes von Software der Kommission ist regelmäßig eine Lizenzierung unter die „EUPL“ geboten (siehe Punkt 3.5).

Im Zuge der Entwicklung der Software ist darauf zu achten, dass Open Source Komponenten vollständig erfasst und anschließend entsprechend rechtlich gewürdigt werden (siehe Punkt 3.6). Zu beachten ist dabei, dass die Einhaltung der Open Source Bedingungen erst bei der Weitergabe (siehe Punkt 3.7) der Open Source Komponenten schlagend werden.

Nachdem eine Verletzung von Open Source Lizenzbedingungen gravierende rechtliche Konsequenzen nach sich ziehen kann (siehe Punkt 4.1), ist dem Anwender von Open Source Software zu empfehlen, vor deren Anschaffung oder Implementierung entsprechende vertragliche (siehe Punkt 4.2) und technische (siehe Punkt 4.3) Absicherungsmaßnahmen zu ergreifen.

Ich stehe bei Fragen gerne zur Verfügung und verbleibe

mit besten Grüßen

Dr. Tobias Tretzmüller, LL.M. (IT-LAW), B.A.  
Rechtsanwalt