Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Bundeskanzlei BK
Digitale Transformation und IKT-Lenkung DTI

14.09.2026

Em002-3 Leitfaden OSS-Lizenzierung

Version 2.0-1339ccb

Inhaltsverzeichnis

1. Management Summary 3

2. Einleitung 5

3. Urheberrecht und Lizenzen: Grundlagen 5

4. OSS-Lizenzen 6

4.1. Grundlagen 6

4.2. Open-Source-Lizenzen 6

4.3. Copyleft versus permissiv 7

4.3.1. Allgemeines 7

4.3.2. Open-Source-Lizenzen mit starkem Copyleft 7

4.3.3. Open-Source-Lizenzen mit schwachem Copyleft 8

4.3.4. Permissive Open-Source-Lizenzen 8

4.4. Kompatibilität von Open-Source-Lizenzen 9

5. Lizenzen für die reine Nutzung 11

6. Lizenz für eigene Projekte bzw. Creation und für Contributions 12

7. Anwendungsfälle und geeignete Lizenzen 14

8. Besondere rechtliche Themen 18

8.1. Internationale Aspekte 18

8.2. Haftungs- und Gewährleistungsausschluss 18

8.3. Copyleft und Lizenzierung zwischen Stellen der Bundesverwaltung 19

8.4. Lizenzierung von Patenten 19

8.5. Duale Lizenzierung 19

8.6. Nachträgliche Änderung von Lizenzen 20

8.7. Nennung von Beitragenden (Mitarbeitende und Dritte) 20

8.8. Urheberrecht, Lizenzen für generative KI zur Codeerstellung 21

8.9. Contributor Licence Agreements (CLA) und Developer Certificate of Origin (DCO) 21

8.10. Probleme mit (L)GPL-3.0 und IoT 23

8.11. Rechtsstellung der Dokumentation 23

9. Weitere Informationen zu Rechtsfragen 24

10. Anhang 25
10.1. Änderungen gegenüber der Vorversion 25
10.2. Referenzen 25
10.3. Abkürzungen 26
10.4. Beispiele von Freigaben und ihrer Lizenz 26
10.5. D.1 Loom 26
10.6. D.2 trustbroker.swiss 26
10.7. D.3 Geocat.ch 27
10.8. D.4 GWEN / ampycloud / c4dl-multi / dvas /… 27
10.9. D.5 Anwendung EMSG (Erhaltungsmanagement in Siedlungsgebieten) 27
10.10. D.6 Anwendung Covid-Zertifikat 28
10.11. D.7 GovCert-Website 28
10.12. D.8 Verschiedene Bibliotheken zu Open Data (z. B. Linked Data) 28
10.13. D.9 Apache FOP angepasst für archivierbares PDF 29
10.14. D.10 Rechtskonforme elektronische Eingabe mit eKomm 29

Haftungsausschluss: Dieses Dokument ist ein laufend weiterentwickelter Entwurf und Teil der Hilfsmittel, welche die Bundesverwaltung bei der Veröffentlichung von Open-Source-Code unterstützen. Weitere Informationen finden Sie im README.

---

1. Management Summary

Es wird empfohlen, bei einem Projektstart in der Bundesverwaltung strategisch eine der folgenden beiden Lizenzen festzulegen:

- **AGPL-Lizenz**: Copyleft-Effekt erwünscht. Geänderter Code soll in jedem Fall wieder der Allgemeinheit zufliessen und kann dann auch von der Bundesverwaltung wieder übernommen werden. Es wird empfohlen, die neuste Version der Lizenz zu verwenden. So lässt sich der Grundsatz «public money – public code» am besten erfüllen. Rechte Dritter, insbesondere bei Weiterentwicklungen, können dies auf LGPL abschwächen.
- **MIT-Lizenz**: Dritten soll es möglich sein, auf Basis des Codes der Bundesverwaltung proprietäre Anwendungen zu entwickeln. Wird ausdrücklich eine Non-Endorsement-Klausel verlangt, so ist BSD-3-Clause vorzuziehen.

Zur Lizenzwahl kann die folgende Entscheidungshilfe verwendet werden:

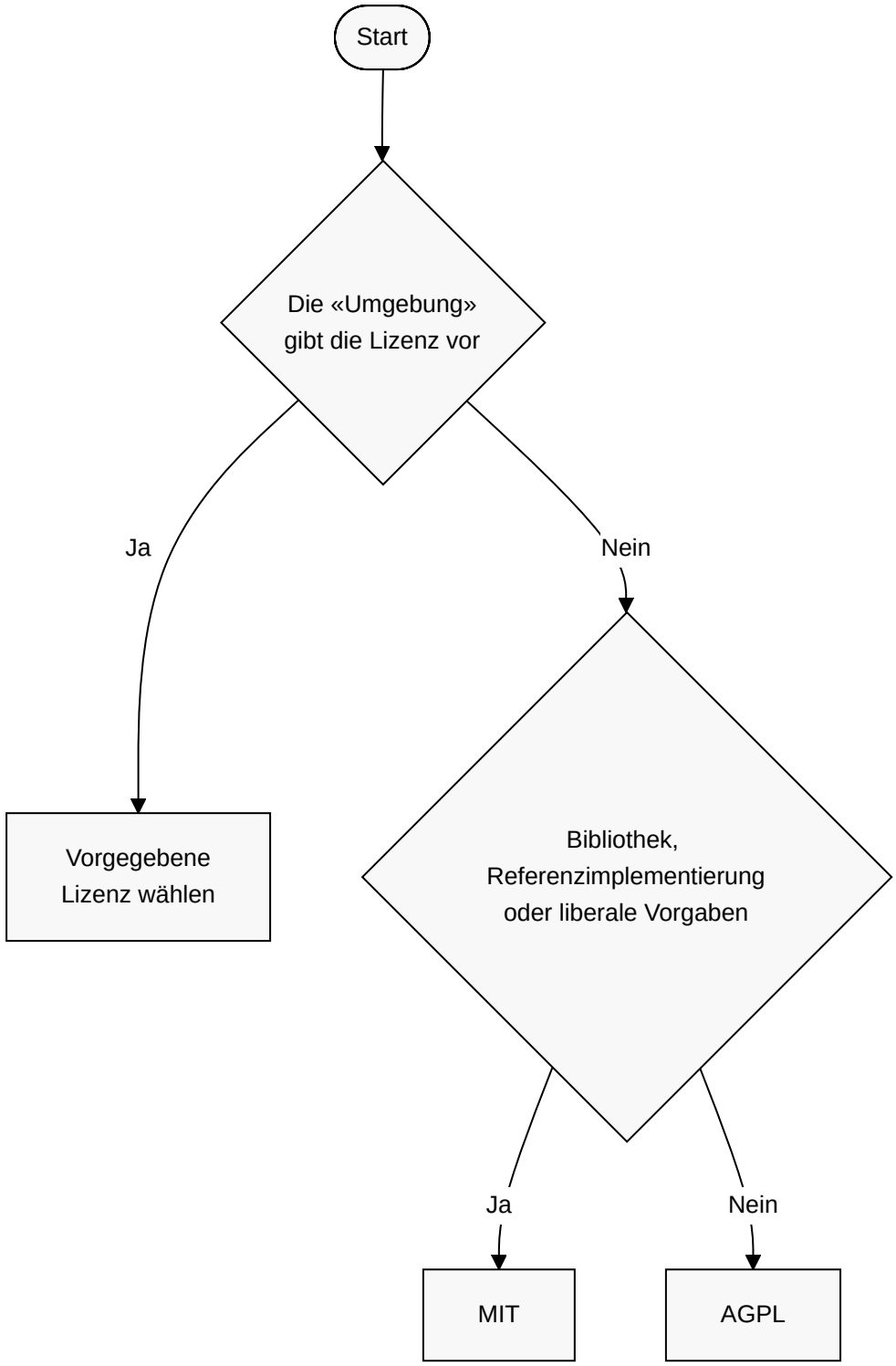


Abbildung 1: Vereinfachte Entscheidungshilfe zur Lizenzwahl

In gewissem Sinne bilden diese beiden Lizenzen die beiden Extreme im Spektrum der OSS-Lizenzen. Es kann durchaus sinnvoll sein, für ein konkretes Projekt eine besser geeignete Lizenz zu wählen. Dieser Leitfaden soll dabei konkrete Unterstützung bieten.

Ergibt die Analyse der Softwarebibliotheken, dass eine andere Lizenz verwendet werden muss, so weisen die Entwickelnden die Projektleitung darauf hin. Die Analyse ist in Em002-2 Anleitung zur Veröffentlichung von Open-Source-Software dargelegt.

Bei der Mitarbeit an bestehenden Projekten sind grundsätzlich alle OSI-kompatiblen Lizenzen akzeptabel.

2. Einleitung

Artikel 9 des Bundesgesetzes vom 17. März 2023[1] über den Einsatz elektronischer Mittel zur Erfüllung von Behördenaufgaben (EMBAG) sieht vor, dass die Bundesbehörden Software, die sie entwickeln oder entwickeln lassen, als Open-Source-Software veröffentlichen und lizenzieren müssen.

Dieses Dokument verfolgt die folgenden Ziele:

- Einführung in die Themen Urheberrecht an Software und OSS-Lizenzen.
- Eine Darstellung, welche OSS-Lizenzen für den Einsatz in der Bundesverwaltung unproblematisch sind. In diesem Dokument nicht genannte Lizenzen und Software unter solchen Lizenzen dürfen ohne zusätzliche Prüfung der Vertragsbestimmungen durch den zuständigen Rechtsdienst nicht verwendet werden.
- Unterstützung bei der Lizenzwahl für die Entwicklung oder Beschaffung eines Projekts nach Art. 9 EMBAG.

3. Urheberrecht und Lizenzen: Grundlagen

Das Urheberrecht begründet sowohl Persönlichkeitsrechte als auch Vermögensrechte (Verwertungsrechte). Zu den Persönlichkeitsrechten gehört beispielsweise das Recht auf Anerkennung der Urheberschaft oder das Recht zu bestimmen, ob und wann das Werk veröffentlicht wird. Zu den Verwertungsrechten gehört beispielsweise das Recht, Werkexemplare herzustellen oder zu verbreiten.

Das Bundesgesetz vom 9. Oktober 1992[2] über das Urheberrecht und verwandte Schutzrechte (URG) sieht eine Reihe weiterer Persönlichkeits- und Verwertungsrechte vor (siehe Art. 8 ff. URG). Diese Rechte stehen der Urheberin oder dem Urheber unmittelbar mit der Schaffung des Werks zu.

Art. 2 URG legt fest, unter welchen Voraussetzungen urheberrechtlicher Schutz gewährt wird. Ein geschütztes Werk im Sinne des Gesetzes liegt nur vor, wenn diese Voraussetzungen erfüllt sind.

Die Voraussetzungen sind:

- Das Werk muss geschaffen worden sein; das blosse Auffinden genügt nicht. Forschungsdaten etwa, also Ergebnisse von Experimenten, lassen sich nicht als Werke einstufen.
- Die Schöpfung muss «geistig» sein, also von einem Menschen stammen.
- Das Werk muss in wahrnehmbarer Weise verwirklicht worden sein. Die blosse Idee hinter einem Werk ist nicht geschützt, sondern nur seine äussere Form, z. B. ein Ausdruck eines Textes auf Papier oder die Aufführung eines Musikstücks.
- Individualität: Schliesslich muss das Werk ein gewisses Mindestmass an Individualität aufweisen. Ein Werk, das jede und jeder gleich gestalten würde, ist nicht geschützt.

Geschützt werden können ganz unterschiedliche Werke, etwa literarische oder wissenschaftliche Texte, Musikwerke, fotografische, filmische und andere visuelle oder audiovisuelle Werke.

Nach Art. 2 Abs. 3 URG sind auch Computerprogramme Werke, sofern die genannten Voraussetzungen erfüllt sind.

1. SR 172.019
2. SR 231.1

Die Urheberrechte stehen mit ihrer Entstehung allein der Urheberin oder dem Urheber zu. Nur sie oder er kann einer anderen Person untersagen, die entsprechenden Handlungen vorzunehmen. Die Urheberin oder der Urheber kann ihre bzw. seine Rechte jedoch nicht nur selbst ausüben:

- Sie oder er kann die Rechte an Dritte übertragen (z. B. durch Verkauf). Die neue Rechtsinhaberin oder der neue Rechtsinhaber kann diese Rechte anschliessend gleich geltend machen wie die Urheberin bzw. der Urheber selbst.
- Alternativ kann die Urheberin oder der Urheber ihre bzw. seine Rechte Dritten auch lizenzieren. Vereinfacht gesagt ist eine Lizenz ein Vertrag, mit dem die Rechtsinhaberin oder der Rechtsinhaber einem Dritten die Nutzung des Werks erlaubt. Die Urheberin oder der Urheber behält die Rechte, verzichtet jedoch gegenüber der Lizenznehmerin bzw. dem Lizenznehmer auf deren Durchsetzung.

Wird ein Computerprogramm im Arbeitsverhältnis geschaffen, so erhält die Arbeitgeberin eine ausschliessliche Lizenz zu dessen Nutzung (Art. 17 URG). Den Arbeitnehmenden verbleiben damit im Wesentlichen nur die Persönlichkeitsrechte an der Software (bzw. je nach Rechtsauffassung nur deren Kernbereich). Dies entspricht der Rechtslage nach dem Bundespersonalgesetz.

4. OSS-Lizenzen

4.1. Grundlagen

OSS zeichnet sich im Wesentlichen durch die folgenden Merkmale aus (nach einer Definition der Open Source Initiative _[Pe1999]_[3]):

- Die unbeschränkte und unentgeltliche Weiterverbreitung der Software ist erlaubt;
- Die Software liegt im Quellcode vor;
- Änderungen an der Software und deren Weiterverbreitung unter derselben Lizenz sind grundsätzlich erlaubt;
- Keine Personen oder Personengruppen dürfen von der Nutzung der Software ausgeschlossen werden, und keine Anwendungsbereiche dürfen ausgeschlossen werden (insbesondere nicht die kommerzielle Nutzung); und
- Die Verbreitung der Software zusammen mit anderer Software (etwa Closed-Source-Software) darf nicht eingeschränkt werden.

Die häufigsten Motive für den Einsatz von OSS sind zunächst die Kosteneinsparungen durch die Nutzung des bereits beträchtlichen Pools frei verwendbarer Software. Unternehmen, die häufig OSS einsetzen, können oft erhebliche Teile ihres IT-Budgets durch den Einsatz von OSS einsparen.

Frei verfügbarer Code lässt sich zudem leicht an die eigenen Bedürfnisse anpassen. Überdies vermeidet die Wahl von OSS einen Vendor-Lock-in.[4]

4.2. Open-Source-Lizenzen

Open-Source-Lizenzen sind in erster Linie normale Lizenzverträge für Computerprogramme.

Anders als normale Lizenzverträge kommen OSS-Lizenzverträge jedoch ohne Weiteres zustande, sobald nur eine der in der Lizenz beschriebenen freien Nutzungsformen ausgeübt wird (etwa durch Kopieren, Weiterverbreiten oder Ändern von Software).

3. https://opensource.org/
4. https://de.wikipedia.org/wiki/Lock-in-Effekt

4.3. Copyleft versus permissiv

4.3.1. Allgemeines

Eine Open-Source-Lizenz nach der OSI kann eine «Copyleft»-Bestimmung enthalten. Änderungen an entsprechend lizenzierter Software müssen wieder unter denselben Bedingungen angeboten werden, die für die ursprüngliche Software galten. Wer solche Änderungen vornimmt, muss bei der Verbreitung der Open-Source-Software in Form von maschinenlesbarem Objektcode den Empfangenden auch den Quellcode anbieten.

In der Formulierung der Free Software Foundation schützt der Copyleft-Effekt die Freiheit der Entwickelnden und stellt damit sicher, dass einmal freie Software immer frei bleibt.

Bei Copyleft-Lizenzen findet eine Art Austausch zwischen der Community und den nutzenden Unternehmen statt. Die Nutzenden, die im Gegenzug für die Vorteile aus dem bestehenden Softwarepool verpflichtet sind, ihre eigenen Entwicklungen wieder unter die jeweilige Lizenz zu stellen, tragen so zur Weiterentwicklung des Softwarepools bei. Im Ergebnis profitieren beide Seiten.

Die Copyleft-Bestimmung hat einen gewissen «Ansteckungseffekt»: Wird copyleft-lizenzierte Software in zuvor proprietäre Software integriert, so muss die ursprünglich proprietäre Software unter einer kompatiblen Open-Source-Lizenz offengelegt werden.

Beim Einsatz von copyleft-lizenziertem Quellcode ist daher darauf zu achten, ihn nur dort zu integrieren, wo die entstehende Software unter einer Open-Source-Lizenz veröffentlicht werden kann und soll.

Enthält eine Open-Source-Lizenz keine Copyleft-Bestimmung (permissive Open-Source-Lizenz), so kann die Lizenz für überarbeitete Fassungen des Codes frei gewählt werden. Insbesondere ist es auch möglich, Änderungen wieder unter eine proprietäre Softwarelizenz zu stellen und den entsprechenden Quellcode in proprietäre Software zu integrieren.

Im Wesentlichen lassen sich Open-Source-Lizenzen in **die folgenden Kategorien** einteilen:

1. Open-Source-Lizenzen mit **starkem Copyleft**;
2. Open-Source-Lizenzen mit **schwachem Copyleft**; und
3. **permissive** bzw. **Non-Copyleft**-Open-Source-Lizenzen.

4.3.2. Open-Source-Lizenzen mit starkem Copyleft

Bei Verwendung einer Open-Source-Lizenz mit starkem Copyleft müssen neue, von der ursprünglichen Software abgeleitete Versionen, sofern sie an Dritte weitergegeben werden, unter die Bedingungen der ursprünglichen Lizenz gestellt und diesen Dritten als Quellcode zur Verfügung gestellt werden.

Als massgebende Open-Source-Lizenzen mit starkem Copyleft gelten am Markt die folgenden beiden:

- GNU General Public Licence (GPL); ein grosser Teil der OSS ist heute darunter lizenziert
- GNU Affero General Public Licence (AGPL)

Der Hauptunterschied zwischen GPL und AGPL betrifft die Art der Nutzung, die das Copyleft auslöst. Bei GPL-Software muss der geänderte Quellcode nur angeboten werden, wenn die neue Softwareversion Dritten als ausführbares Programm angeboten wird (z. B. als mobile App). Wird die Software nur über das Internet zur Verfügung gestellt (aus der eigenen Cloud von eigenen Servern), etwa in Form von «Software as a Service» (SaaS) oder einer Programmierschnittstelle (API), so löst dies kein Copyleft aus.

Steht die Software unter der AGPL, so muss der geänderte Quellcode auch dann mitgeliefert werden, wenn die Funktionalität der Software über eine Website oder eine Programmierschnittstelle angeboten wird. Die AGPL ist damit noch strenger als die GPL.

Bei Lizenzen mit starkem Copyleft betrifft das Copyleft nicht nur das jeweilige Softwaremodul (Bibliothek), sondern das gesamte Softwareprogramm, in das ein copyleft-lizenziertes Softwaremodul eingebettet sein kann. Der erwähnte «Ansteckungseffekt» greift somit.

4.3.3. Open-Source-Lizenzen mit schwachem Copyleft

Auch Lizenzen mit schwachem Copyleft verlangen, dass Änderungen an ihrem Quellcode den empfangenden Dritten unter der ursprünglichen Open-Source-Lizenz freigegeben werden.

Anders als Lizenzen mit starkem Copyleft bewirken sie jedoch nicht, dass andere abgrenzbare Softwarekomponenten (andere Bibliotheken oder das Hauptprogramm selbst) mit ihrer Lizenz «infiziert» werden. Dies erlaubt die Integration von Open-Source-Software mit schwachem Copyleft in proprietäre Software oder in Software mit anderen OSS-Lizenzen, ohne dass diese unter der ursprünglichen Lizenz freigegeben werden muss.

Dies sind die am häufigsten verwendeten Open-Source-Lizenzen mit schwachem Copyleft:

- GNU Lesser General Public Licence (LGPL)
- Mozilla Public Licence 2.0 (MPL)
- Common Development and Distribution Licence (CDDL)
- Eclipse Public Licence (EPL)[5]
- Microsoft Reciprocal Licence (Ms-RL)
- European Union Public Licence (EUPL)[6]

4.3.4. Permissive Open-Source-Lizenzen

Lizenzen ohne Copyleft-Effekt zeichnen sich dadurch aus, dass sie den Lizenznehmenden keine Vorgaben zur Lizenzierung ihrer abgeleiteten Software machen und daher weder ihr neuer Quellcode noch Änderungen an der Open-Source-Software den empfangenden Dritten offengelegt werden müssen. Dies ermöglicht die Entwicklung proprietärer Softwareprodukte durch die Integration von Software unter permissiven Open-Source-Lizenzen.

Beispiele permissiver Lizenzen sind:

- MIT Licence
- Apache Licence 2.0
- Berkeley Software Distribution (BSD) Licence (BSD 2-Clause und BSD 3-Clause)
- Microsoft Public Licence (Ms-PL)

---

5. Neben der Eclipse Public Licence (EPL-1.0) gibt es auch die Eclipse Distribution Licence (EDL-1.0). + Während die EDL der BSD-3-Clause entspricht, ist die EPL eine Lizenz mit schwachem Copyleft.
6. Die EUPL selbst hat ein starkes Copyleft, doch ist aufgrund der Öffnungsklausel ein Übergang zu Lizenzen mit schwachem Copyleft möglich, sodass der Schutz begrenzt ist.

4.4. Kompatibilität von Open-Source-Lizenzen

Programme bestehen in der Regel aus einer Vielzahl von Softwarekomponenten und -modulen, die auf unterschiedliche Weise miteinander verbunden sein können. In der Softwareentwicklung werden bestehende Open-Source-Komponenten häufig in proprietäre, interne Anwendungen und Lösungen integriert. Sobald diese von aussen über eine Website, APIs, Softwareverteilung oder auf andere Weise zugänglich werden, ist auf die Kompatibilität der jeweiligen Open-Source-Lizenzen zu achten, denn dann kann das Copyleft greifen.

Dürfen abgeleitete Werke, die unter einer bestimmten Lizenz stehen, nur unter denselben Lizenzbedingungen verbreitet werden, so hat dies zur Folge, dass andere Open-Source-Lizenzen mit anderen oder widersprüchlichen Lizenzpflichten für abgeleitete Werke nicht verwendet werden können.

Manche Open-Source-Lizenzen enthalten sogenannte Öffnungsklauseln, die die Verwendung des darunter lizenzierten Codes in Projekten erlauben, die unter anderen Lizenzen stehen. So erlaubt etwa die LGPL, Version 2.1, auch die Verwendung des Codes unter der GPL. Die GPLv3 enthält unter anderem eine Kompatibilitätsklausel für die AGPL und die Apache License 2.0. Die EUPL ist sehr offen gestaltet und enthält im Anhang eine lange Liste kompatibler Lizenzen, insbesondere solcher mit schwachem Copyleft wie der LGPL. Dies hat zur Folge, dass der in der Lizenz eigentlich vorgesehene starke Copyleft-Schutz indirekt abgeschwächt wird.

Die folgende Grafik bildet die Abhängigkeiten der wichtigsten Open-Source-Lizenzen ab (Goldstein 2018). Die Darstellung beruht auf der Visualisierung von David A. Wheeler, der die Kompatibilitäten der verschiedenen Lizenzen und ihrer Versionsnummern analysiert hat (Wheeler 2007), und wurde zusammen mit Angaben von https://www.gnu.org/licenses/license-list.de.html weiterentwickelt.[7]

Die Grafik ist wie folgt zu lesen: Code nach einer früher in der Kette aufgeführten Lizenz kann in Software übernommen werden, die unter einer später in der Kette aufgeführten Lizenz veröffentlicht wird. So kann etwa Code unter der modifizierten BSD unter eine LGPL- oder GPL-Lizenzvariante gestellt werden, aber nicht umgekehrt. Public Domain bedeutet, dass das Werk keinen urheberrechtlichen Beschränkungen unterliegt.

7. Eine weitere, erweiterte Darstellung verwendet GEANT in seinem Confluence: «Software Licence Selection and Management in GÉANT - GEANT Software Development Support - GÉANT federated confluence», https://wiki.geant.org/pages/viewpage.action?pageId=1032978710. Dort finden sich einige zusätzliche Lizenzen.

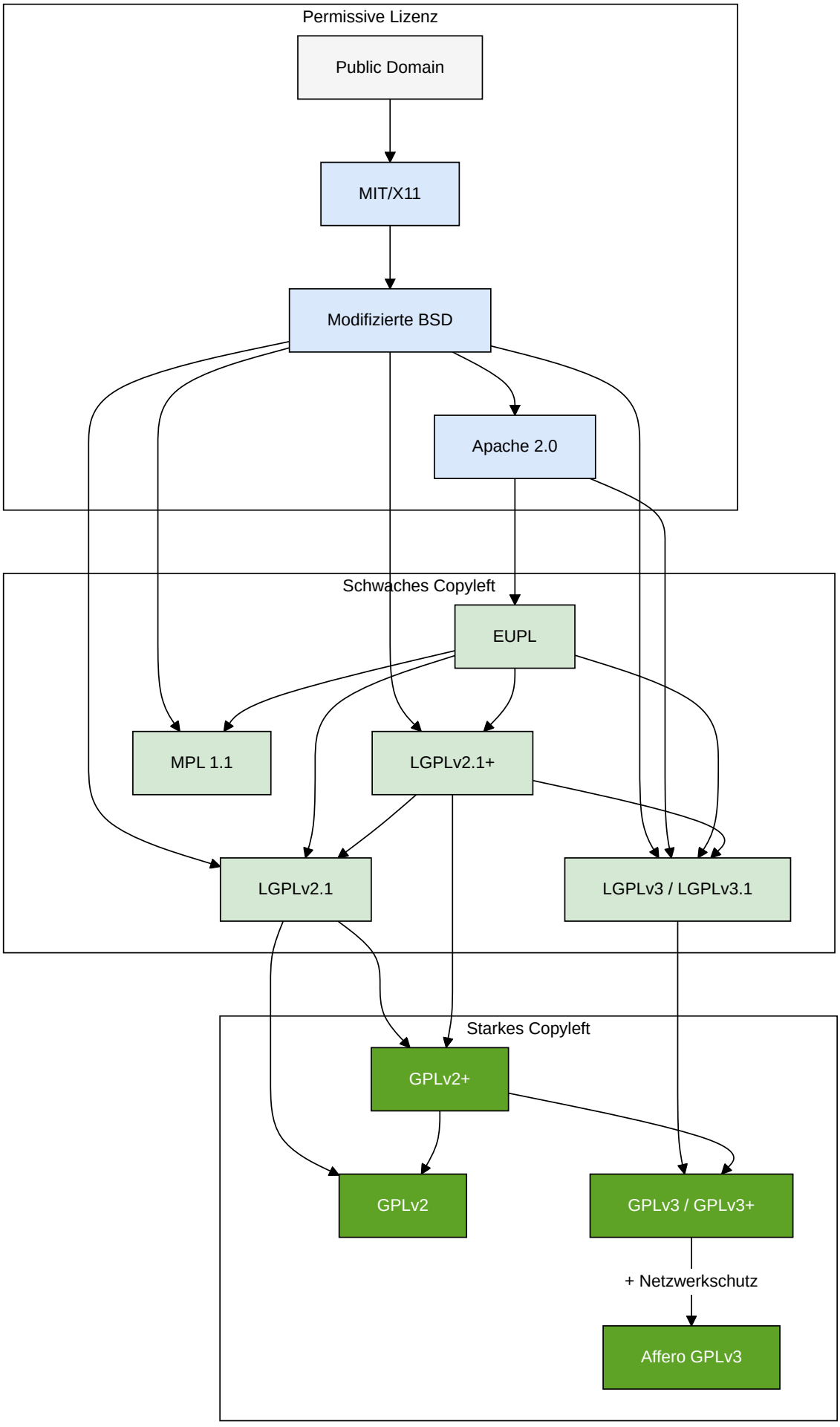

Permissive Lizenz
Public Domain
MIT/X11
Modifizierte BSD
Apache 2.0
Schwaches Copyleft
EUPL
MPL 1.1
LGPLv2.1+
LGPLv2.1
LGPLv3 / LGPLv3.1
Starkes Copyleft
GPLv2+
GPLv2
GPLv3 / GPLv3+
+ Netzwerkschutz
Affero GPLv3

Abbildung 2: Kompatibilität von Open-Source-Lizenzen (in Anlehnung an die Grafik von David A. Wheeler, 2007 und https://www.gnu.org/licenses/license-list.de.html)

Wichtig: Da Lizenzkompatibilitäten stets relevant sind, sollten die Entwicklungsteams bzw. Leistungserbringer die Entwicklungs-Stacks mit Bibliotheken vorgängig auf mögliche Kompatibilitätsprobleme prüfen. Das heisst: Bevor der Einsatz einer Bibliothek durchgewinkt wird, sollte neben Sicherheit und Homogenität im Entwicklungs-Stack geprüft werden, ob die Lizenz der Bibliothek mit den massgebenden, von der Bundesverwaltung bevorzugten Lizenzen kompatibel ist.

5. Lizenzen für die reine Nutzung

Dieses Kapitel behandelt die Situation, in der Software unter einer OSS-Lizenz genutzt werden soll, diese Software jedoch nicht verändert werden soll oder Änderungen nur intern genutzt werden.

Die folgenden Lizenzen sind bei der Nutzung von Software durch Bundesbehörden grundsätzlich unproblematisch. Die entsprechenden Bedingungen, namentlich ein allfälliges Copyleft, sind jedoch einzuhalten.

Unter Umständen bestimmt die Lizenz eines bereits bestehenden Softwaremoduls die möglichen Lizenzen für das Gesamtprodukt (siehe Kapitel 4.4).

Manche Lizenzen sind in Kombination inkompatibel (siehe Abbildung 2).

Lizenz	Copyleft	Besonderheiten
MIT	Nein	Bevorzugte Lizenz der Bundesverwaltung, wenn eine Weiterverbreitung angestrebt wird und Dritten erlaubt werden soll, daraus wieder proprietäre Software zu entwickeln.
Apache 2.0	Nein	Apache ist eine sehr verbreitete Lizenz, bekannt durch die Projekte unter der Apache Foundation. Nutzung der Marken der Lizenzgebenden ausgeschlossen, ausser zur Beschreibung des Werks.
GPL v3	Ja	Dies ist DIE klassische Copyleft-Lizenz. Das Copyleft gilt auch, wenn GPL-lizenzierte Teile in ein Gesamtprogramm eingebunden werden. Verbot von Digital Rights Management. Besondere Kompatibilitätsregeln.
LGPLv3	Schwach	Reduzierte Copyleft-Lizenz, eigens für Softwarebibliotheken entwickelt. Anders als die GPL erlaubt die LGPL, geschlossenen (also proprietären) Code unter bestimmten Bedingungen mit LGPL-Code zu kombinieren. Beispiele sind Standardbibliotheken wie glibc. Die LGPL erlaubt, das Programm, als dessen Teil die Bibliothek verbreitet wird, unter die GPL zu stellen.

Lizenz	Copyleft	Besonderheiten
Affero GPLv3 (AGPL)	Ja	Auch die SaaS-Nutzung löst Copyleft aus. Im Übrigen analog zur GPLv3. Diese Lizenz schliesst die Lücke der GPL, die dazu führte, dass bei der Nutzung für SaaS-Angebote keine Freigabe nötig war. Mit dieser Lizenz kann die Bundesverwaltung sicherstellen, dass Erweiterungen des Codes auch bei SaaS der Allgemeinheit zugutekommen, und unterstützt den Grundsatz **«public money – public code»** am besten.
BSD-3	Nein	Allgemeine, liberale Lizenz. Namen der Autorinnen und Autoren dürfen nicht zur Bewerbung abgeleiteter Werke verwendet werden (Werbeklausel)
European Union Public Licence (EUPL)	Schwach	Besondere Lizenz, die für die EU verwendet wird. Besonders geeignet als Grundlage für europäische Projekte. Lizenz unter Berücksichtigung des europäischen Rechts. Freigabe unter bestimmten kompatiblen Lizenzen ausdrücklich erlaubt.

Tabelle 1: Für die Bundesbehörden unproblematische Lizenzen (nach *[Sc2024]*)

Andere Lizenzen dürfen verwendet werden. In diesen Fällen kann es nötig sein, den Rechtsdienst des jeweiligen Amtes beizuziehen.

6. Lizenz für eigene Projekte bzw. Creation und für Contributions

Dieses Kapitel betrifft die folgenden Fälle:

- Weiterentwicklung / Contribution
  - bei der Bundesbehörden bestehende Software weiterentwickeln; oder
  - neue Software entwickeln, dabei aber auf bestehende OSS-Softwarebibliotheken zurückgreifen; und
  - die entsprechende Software Dritten zur Verfügung stellen. Zu den Dritten zählen auch Organisationen der dezentralen Bundesverwaltung, sofern sie eine eigene Rechtspersönlichkeit haben. Werden die Weiterentwicklungen nur intern genutzt und nicht Dritten zur Verfügung gestellt, so gilt nur Lizenzen für die reine Nutzung.
- Eigene Projekte / Creation / Veröffentlichung: Dieses Kapitel betrifft Fälle, in denen die Bundesbehörden neue Software entwickeln. Dies dürfte selten sein, da die meisten Entwicklungen auf bestehende Open-Source-Softwarebibliotheken zurückgreifen.

  Für vollständige Neuentwicklungen sollte ein Lizenztyp gewählt werden, der eine breite und nachhaltige Grundlage für Weiterentwicklungen ermöglicht. Wichtig ist dabei, dass die betreffende Lizenz in der massgebenden Entwickler-Community breit akzeptiert ist.

Die Kompatibilitätsprüfung erfolgt nach Em002-2 Anleitung zur Veröffentlichung von OSS, Kapitel 7.4. Die Kompatibilität der Lizenzen zeigt Abbildung 2.

Die folgenden Dimensionen sollten berücksichtigt werden:

- Anwendungsfall
- Gewünschte Prüfung Em002-2.1 Checkliste OSS-Vorabklärung
- Einfachheit des Community-Aufbaus Em002-4 Leitfaden OSS-Community
- Einfachheit der Anwendung
- Rechtssicherheit
- Verbreitung

Lizenzen nach Kapitel Lizenzen für die reine Nutzung sind zu bevorzugen.

Die folgenden **Copyleft-Lizenzen** sollten von den Bundesbehörden vorrangig verwendet werden, insbesondere wenn eine nachhaltige offene Entwicklung angestrebt wird und der vererbende Copyleft-Effekt gelten soll. Damit wird sichergestellt, dass von der Öffentlichkeit bezahlte Software und sämtliche Ableitungen davon offen bleiben. Dies erfüllt den Grundsatz «**public money – public code**» am besten und ist **die bevorzugte Empfehlung für die Bundesverwaltung.**

Empfohlene Lizenzen in absteigender Reihenfolge:

- **AGPL v3**
- GPL v3
- LGPL 3.0
- European Union Public Licence (EUPL)

Die folgenden Lizenzen sollten von den Bundesbehörden verwendet werden, wenn **kein Copyleft gewünscht** oder erforderlich ist (in absteigender Reihenfolge):

- **MIT**
- Apache Licence 2.0
- BSD v3

Bei der Mitarbeit an bestehenden Projekten können zudem problemlos die folgenden Lizenzen verwendet werden:

- Mozilla Public Licence
- Microsoft Public Licence

Soll eine andere Lizenz verwendet werden, so wird empfohlen, dies in der Em002-2.3 Checkliste OSS-Freigabe und -Veröffentlichung kurz zu begründen.

Nach Art. 9 Abs. 4 EMBAG sollen in jedem Fall international etablierte Lizenzen verwendet werden.

Weitere Werkzeuge zur Prüfung von Lizenzkompatibilitäten sind:

- ORT Toolkit (https://oss-review-toolkit.org/)
- Black Duck (www.blackducksoftware.com),
- FOSSA (www.fossa.com),
- das Open Source Licence Comparison Grid (https://www.cmu.edu/cttec/forms/opensourcelicensegridv1.pdf)
- FOSSology (https://www.fossology.org).
- OpenCode-Prozess: https://opencode.de/en/knowledge/general-conditions/standardised-open-source-licenses

7. Anwendungsfälle und geeignete Lizenzen

Die folgende Tabelle und die folgende Grafik zeigen, wie anhand unterschiedlicher Anwendungsfälle geeignete Lizenzen für Open-Source-Software ausgewählt werden können.

Anwendungsfall	Lizenz(en)	Begründung
Bestehendes Projekt nutzen und dazu beitragen	**Lizenz gemäss Vorgabe des Projekts**	Aus rechtlichen Gründen erforderlich.
Die Lizenz ist durch die Verwendung bestehender Teile vorgegeben (Lizenzen mit Copyleft)	**Mit den Teilen kompatible Lizenz verwenden.**	Aus rechtlichen Gründen erforderlich. Hinweis: Eine Reihe von Bibliotheken ist unter mehreren Lizenzen verfügbar. Nur eine der Lizenzen muss übereinstimmen. Einer der Zwecke der Mehrfachlizenzierung ist es, sicherzustellen, dass Bibliotheken sowohl mit als auch ohne Copyleft genutzt werden können.
Bestehendes Ökosystem mit bevorzugter Lizenz (z. B. handelt es sich um ein Plug-in für Software, die unter MIT-Lizenz veröffentlicht ist. Dann wäre die Verwendung einer MIT-Lizenz am zweckmässigsten, da alle anderen Nutzenden dies erwarten und sich die Software in das Ökosystem einfügt)	**Lizenz verwenden, wenn sie auf der Liste steht, ansonsten möglichst dieselbe Lizenz verwenden**	Die Akzeptanz in der Community ist zentral.

Anwendungsfall	Lizenz(en)	Begründung
Ziel ist eine möglichst breite Verbreitung, z. B. bei einer Referenzimplementierung[8] (z. B. bei Software, die auch in kommerziellen Lösungen eingesetzt werden soll).	**MIT** BSD v3 Apache Licence 2.0 LGPL v3	In diesem Fall sollte kein Copyleft verwendet werden.
Änderungen am Code sollen an die Bundesbehörden zurückfliessen	**AGPL v.3** GNU GPL v.3 EUPL	Die AGPL erzwingt die Freigabe durch Dritte auch bei der Nutzung in der Cloud, GPL und EUPL nicht.
SaaS-Lösungen sollen vom Rückfluss von Codeänderungen nicht ausgenommen sein	**AGPL v.3**	Die AGPL erzwingt die Freigabe durch Dritte auch bei der Nutzung in der Cloud.
Eine einfache Zusammenarbeit mit der Community ist wichtig.	**AGPL v.3** GNU GPL v3 Apache Licence 2.0	GPL wird bevorzugt, um die Software frei zu halten. Apache enthält Regeln für die Community-Governance, zusätzliche Regeln für Beiträge (man darf keine andere Lizenz als Apache angeben; der Beitrag als solcher ist nicht näher geregelt, besteht also ein CLA, so geht dieses vor), eine Patentlizenz sowie Regeln zur Patentlizenzierung. Insbesondere auch, dass jemand, der Patente gegen eine andere beitragende Person geltend macht, seine Patentlizenz unter der Apache-Lizenz verliert. Massgebend sind die Regeln, welche die Apache Foundation für Beiträge festlegt. Apache ist daher sinnvoll, wenn die Projekt-Governance in erster Linie nach Apache aufgebaut werden soll.
Kleine, universell nutzbare Komponente (Bibliothek) oder kleines Stück Software, bei dem Nachhaltigkeit oder Pflege durch eine Community nicht wichtig ist	**MIT** Apache Licence 2.0 BSD v3 LGPL v3	Die LGPL sieht ein Copyleft nur für die Komponente selbst vor, nicht jedoch für das gesamte Projekt, in das die Komponente eingefügt wird. Die anderen Lizenzen sind permissiv.

8. Um beispielsweise ein Gesetz leichter umsetzbar zu machen, kann eine Referenzimplementierung von den Bundesbehörden in Auftrag gegeben werden. Siehe z. B. https://ech.ch/de/ech/ech-0238/1.0

Anwendungsfall	Lizenz(en)	Begründung
Es muss einfach und permissiv sein; Kontrolle ist nicht wichtig	**MIT** Apache Licence 2.0 BSD v3	Maximal permissiv
Es ist für die Bundesbehörden wichtig, was mit dem Code geschieht, und sie möchten möglichst viele Verbesserungen erhalten.	**AGPL v.3** GNU GPLv3	Copyleft
Es soll verhindert werden, dass die Software proprietär wird und als Konkurrenzprodukt am Markt positioniert wird	**AGPL v.3** GNU GPLv3	Copyleft
Es soll verhindert werden, dass die Bundesbehörden von einzelnen Lieferanten abhängig werden («Vendor-Lock-in»)	**AGPL v.3** GNU GPLv3	Das Copyleft verhindert, dass ein Lieferant mit der Zeit eine Alleinstellung erlangt, indem er die Software des Bundes in einem proprietären Produkt nutzt, was für den Bund wiederum zu einer Abhängigkeit von diesem Lieferanten führen könnte.

Tabelle 2: Je nach Anwendungsfall zu verwendende Lizenz für die Freigabe neuer Software nach Art. 9 EMBAG **(fett = empfohlen, sofern keine Gründe dagegensprechen)**

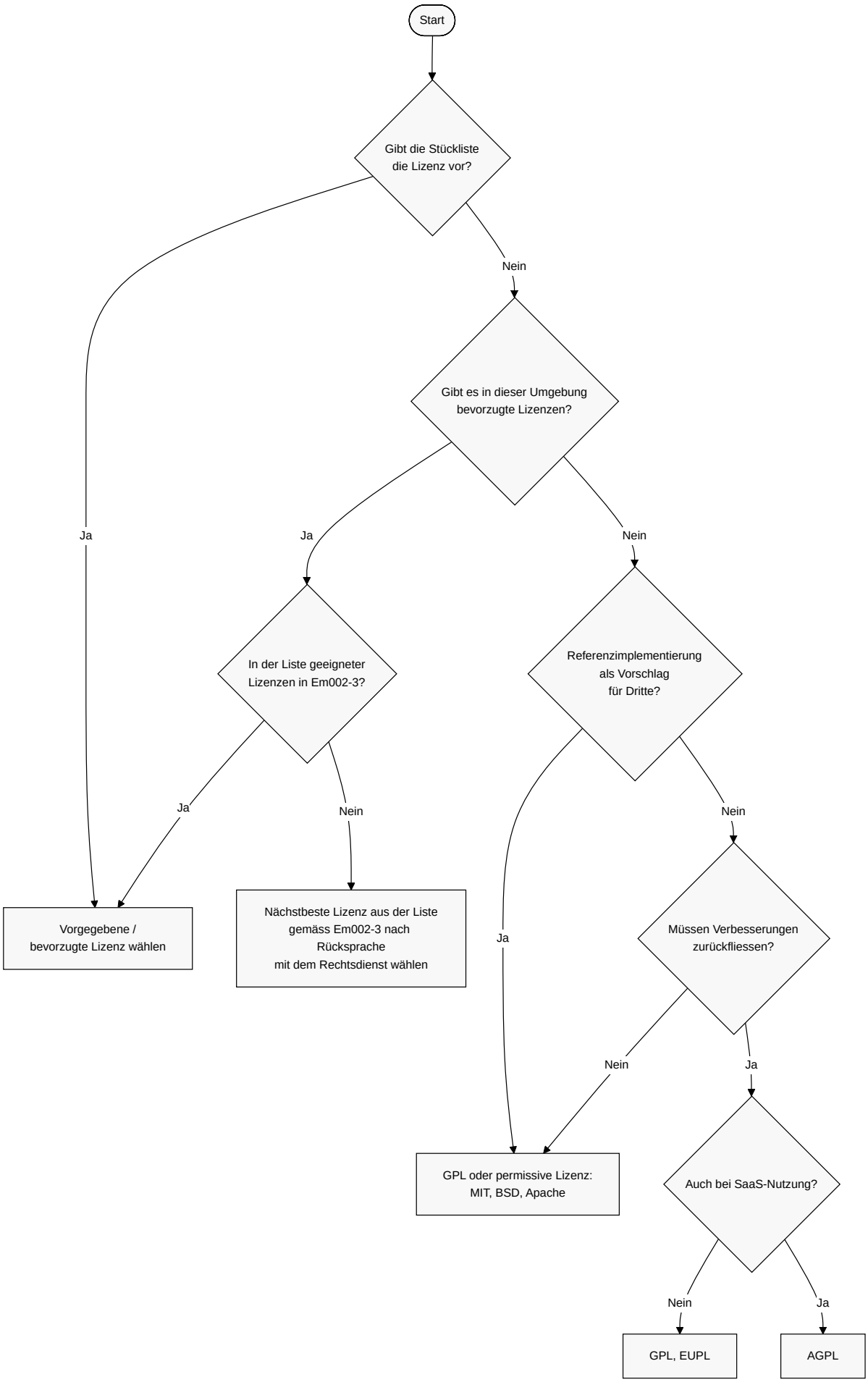

Start
Gibt die Stückliste die Lizenz vor?
Ja
Nein
Gibt es in dieser Umgebung bevorzugte Lizenzen?
Ja
Nein
In der Liste geeigneter Lizenzen in Em002-3?
Ja
Nein
Vorgegebene / bevorzugte Lizenz wählen
Nächstbeste Lizenz aus der Liste gemäss Em002-3 nach Rücksprache mit dem Rechtsdienst wählen
Referenzimplementierung als Vorschlag für Dritte?
Ja
Nein
GPL oder permissive Lizenz: MIT, BSD, Apache
Müssen Verbesserungen zurückfliessen?
Nein
Ja
Auch bei SaaS-Nutzung?
Nein
Ja
GPL, EUPL
AGPL

Abbildung 3: Erweiterter Entscheidungsbaum zur Lizenzwahl in der Bundesverwaltung (falls nicht AGPL oder MIT)

Die Stückliste in Abbildung 3 ist die Software Bill of Materials (SBOM) aller bereits bestehenden Softwarebibliotheken und Teilprogramme, die in die Gesamtlösung eingeflossen sind. Jede Bibliothek hat ihre eigene Lizenz. Je nachdem, wie diese mit anderen kompatibel ist (siehe Kapitel 4.4), ist die mögliche Auswahl an Lizenzen für die Veröffentlichung eingeschränkt.

Ein letzter Punkt kann den Entscheid ebenfalls beeinflussen: **Der Wechsel von einer restriktiveren zu einer weniger restriktiven Lizenz ist nachträglich einfacher**. Im umgekehrten Fall ist ein Fork[9] kaum zu vermeiden.

8. Besondere rechtliche Themen

8.1. Internationale Aspekte

Die wesentlichen Rechtsfragen von OSS in der Praxis sind überwiegend lizenzvertraglicher Natur.

Bei lizenzrechtlichen Fragen gilt zunächst das von den Parteien gewählte Recht (die Mozilla Public License MPL etwa enthält in Ziffer 8 eine Rechtswahl zugunsten des Rechts der beklagten Partei).

Fehlt eine Rechtswahl, was bei allen übrigen hier vorgestellten Lizenzen der Fall ist, so gilt in der Regel das Recht des Staates der lizenzgebenden Partei. Lizenzieren Bundesbehörden also Software von ausländischen Lizenzgebenden, so gilt das Recht des jeweiligen Landes. Sind mehrere Beitragende beteiligt, so kann dies zu komplexen Situationen führen. Umgekehrt gilt bei allfälligen Streitigkeiten zwischen ausländischen Lizenznehmenden und den Bundesbehörden als Lizenzgeberin Schweizer Recht.

Die Frage, welche Aspekte vertraglicher und welche urheberrechtlicher Natur sind, lässt sich nicht allgemein beantworten und kann im Einzelfall recht komplex sein. Die Übertragbarkeit des Urheberrechts oder von Teilen davon oder der Umfang zulässiger Rechtseinräumungen gehören etwa zum Urheberrecht. Hier ist das Recht des jeweiligen Schutzlandes anzuwenden. Eine abweichende Rechtswahl ist nicht möglich (zum Ganzen siehe Jaeger/Metzger, 449 ff.).

8.2. Haftungs- und Gewährleistungsausschluss

Open-Source-Lizenzen schliessen regelmässig jede Gewährleistung und Haftung so weit aus, wie dies rechtlich zulässig ist. In der Praxis treten selten Haftungs- oder Gewährleistungsfragen auf; die damit verbundenen Risiken sind daher gering.

Es ist davon auszugehen, dass bei korrekter Lizenzierung die in allen hier behandelten Lizenzen vorhandenen Haftungsausschlussklauseln greifen, sodass nur die Haftung für grobe Fahrlässigkeit und Vorsatz, allenfalls für Personenschäden, bestehen bleibt.

Regeln zu allgemeinen Geschäftsbedingungen bleiben vorbehalten, etwa wie in Deutschland; soweit jedoch die Bundesbehörden als Lizenzgeberin Haftungssubjekt sind, gilt Schweizer Recht (siehe 8.1 oben), das die genannten Haftungs- und Gewährleistungsausschlüsse im B2B-Verkehr zulässt.

Umgekehrt bedeutet dies aber auch, dass bei Programmfehlern ein Vorgehen gegen die Autorinnen und Autoren des enthaltenen OSS-Codes in der Regel schwierig ist.

Das übliche Vorgehen beim Einsatz von OSS besteht daher regelmässig darin, sich mit den üblichen IT-Sicherheitsmassnahmen gegen Fehler abzusichern und die Fehlerbehebung über

9. Zur Definition siehe Em002-1 Praxisleitfaden für Open-Source-Software in der Bundesverwaltung.

Wartungsverträge sicherzustellen. Ein Rückgriff auf OSS-Beitragende besteht nicht.

8.3. Copyleft und Lizenzierung zwischen Stellen der Bundesverwaltung

Das Copyleft gilt bei der Weitergabe zwischen verschiedenen Rechtsträgern (beispielsweise innerhalb eines Konzerns oder zwischen Stellen der dezentralen Bundesverwaltung mit eigener Rechtspersönlichkeit).

Das Copyleft gilt nicht bei der Weitergabe zwischen Stellen desselben Rechtsträgers (beispielsweise innerhalb der zentralen Bundesverwaltung).

8.4. Lizenzierung von Patenten

Softwarepatente sind nach europäischer Rechtsauffassung nur in engen Grenzen möglich. Entscheidend ist, dass die Software einen technischen Beitrag leistet, also ein konkretes technisches Problem ausserhalb des Computers löst, auf dem sie läuft. Beispiele wären die Motorsteuerung in Autos oder die Steuerung eines Roboters. Software als solche, die nur auf einem Computer läuft, aber keine solche Aussenwirkung hat, ist in Europa grundsätzlich nicht patentierbar. Damit fallen die meisten Softwareprogramme nicht in den Anwendungsbereich des Patentrechts.

In anderen Teilen der Welt, insbesondere in den USA, werden Softwarepatente etwas grosszügiger erteilt. Doch auch in den USA wird eine abstrakte Idee oder ein Geschäftsmodell nicht allein dadurch zur patentierbaren Erfindung, dass sie in Computersoftware umgesetzt wird.

Bei manchen Lizenzen werden allfällige Patente der lizenzgebenden Partei nicht mitlizenziert. Dies sind die GPL2, die BSD-Lizenzen und die MIT-Lizenz.

Bei Projekten, die auch ausserhalb Europas genutzt werden könnten oder die die genannte Form der Aussenwirkung aufweisen, empfiehlt es sich, die patentrechtlichen Aspekte im Einzelfall abzuklären.

8.5. Duale Lizenzierung

Die Urheberin oder der Urheber kann Software gleichzeitig unter verschiedenen Lizenzen lizenzieren. Dies wird duale Lizenzierung genannt.

Eine interessante Form der dualen Lizenzierung liegt vor, wenn die Rechtsinhaberin die Software als OSS mit starkem Copyleft lizenziert (was deren Integration in proprietäre Software verhindert) und interessierten Lizenznehmenden zugleich eine *kostenpflichtige* Lizenz anbietet, die ihnen die Integration der Software in ihre proprietäre Software erlaubt.

Weil interessierte Lizenznehmende so vermeiden können, ihre proprietäre Software als OSS freizugeben, sind sie bereit, für die Lizenz zu bezahlen. Siehe auch den Anhang zu Em002-1 Praxisleitfaden für Open-Source-Software in der Bundesverwaltung.

Für die Bundesbehörden selbst dürfte die duale Lizenzierung in den meisten Fällen uninteressant sein. Solange die Urheberrechte bei den Bundesbehörden liegen (was etwa nach den allgemeinen Geschäftsbedingungen des Bundes geregelt ist), können die Bundesbehörden allein über eine allfällige duale Lizenzierung entscheiden.

Werden Wartung und (Weiter-)Entwicklung an Dritte ausgelagert, so kann dies anders sein. Liegen die Rechte ausnahmsweise bei diesen und haben sich die Bundesbehörden nur das Recht auf Erteilung einer OSS-Lizenz mit Copyleft vorbehalten, so könnte der Dritte versucht sein, eine duale Lizenzierung vorzunehmen. Soll dies verhindert werden, so sollte es vertraglich ausgeschlossen werden.

8.6. Nachträgliche Änderung von Lizenzen

Zunächst ist festzuhalten, dass OSS-Lizenzen grundsätzlich unveränderlich und unwiderruflich sind. Lizenznehmende, die den Lizenzvertrag *vor* einer Änderung abgeschlossen haben, können den Code weiterhin auf Grundlage der alten Lizenz nutzen, verändern und an Dritte weitergeben.

In der Praxis führt eine Lizenzänderung oft zu einem Fork einer neuen Version der Software unter der alten Lizenz, der von den bisherigen Lizenznehmenden eigenständig weiterentwickelt wird.

Die Einführung einer neuen Lizenz für Software ist nur möglich, wenn alle Urheberrechtsinhabenden der Software zustimmen. Änderungen sind in der Praxis vorgekommen, etwa beim Wechsel von einer permissiven zu einer Copyleft-Lizenz.

8.7. Nennung von Beitragenden (Mitarbeitende und Dritte)

Wird ein Computerprogramm im Arbeitsverhältnis geschaffen, so erhält die Arbeitgeberin eine ausschliessliche Lizenz zu dessen Nutzung (Art. 17 URG). Den Arbeitnehmenden verbleiben damit im Wesentlichen nur die Persönlichkeitsrechte an der Software (bzw. je nach Rechtsauffassung nur deren Kernbereich). Dasselbe gilt für Dritte, die Software für die Bundesbehörden entwickeln, sofern die entsprechenden AGB[10] des Bundes gelten, welche die entstehenden Rechte an die Bundesbehörden übertragen.

Dennoch stellt sich die Frage, ob Beitragende (Mitarbeitende, Dritte) das Recht oder die Pflicht haben sollen, sich in Open-Source-Repositories als Autorinnen und Autoren von Beiträgen zu erkennen zu geben.

Ein Argument für die Bundesbehörden, OSS einzusetzen, ist das Angebot attraktiver Arbeitsplätze: OSS bietet Beitragenden die Möglichkeit, ihr Wissen öffentlich zu zeigen und sich so einen Ruf aufzubauen. Ihre Fähigkeiten werden überprüfbar, werden von Peers bestätigt, und wer in einem OSS-Projekt Einfluss gewonnen hat, kann dies bei der Stellensuche als Hebel nutzen.

Ein weiteres Argument könnten die Persönlichkeitsrechte sein, deren Kernbereich der ursprünglichen Urheberin bzw. dem ursprünglichen Urheber weder vertraglich noch gesetzlich entzogen werden kann. Dieser Kernbereich ist bei Computerprogrammen jedoch klein; zu verlangen, dass die Entwicklerin oder der Entwickler eines Computerprogramms auch ein Recht auf Nennung haben sollte, ginge wohl zu weit.

Zudem setzt ein Austausch zwischen Fachpersonen in der Community voraus, dass sie von anderen einzeln angesprochen werden können.

Je nach Projekt kann es daher sinnvoll sein, Beitragenden die Möglichkeit zu geben, sich in der Community als Beitragende zu erkennen zu geben.

Aus Sicht des Datenschutzes ist ein Zwang dazu zu vermeiden, wobei der Austausch in der Community keine Offenlegung von Namen voraussetzt – Pseudonyme sind eher die Regel als die Ausnahme. Wollen Beitragende ihre Namen nicht offenlegen, so sollte ihnen daher die Möglichkeit gegeben werden, unter einem Pseudonym aufzutreten.

Es wird empfohlen, die Namen der Entwickelnden in den Commits zu nennen, sofern das Projekt dem nicht entgegensteht.

In jedem Fall sollten die Bundesbehörden als Rechtsinhaberinnen auftreten.

10. https://www.bkb.admin.ch/bkb/de/home/themen/agb.html

8.8. Urheberrecht, Lizenzen für generative KI zur Codeerstellung

Hier beziehen wir uns nur auf Code, der mithilfe generativer KI erstellt wurde. Weitere Informationen zu diesem Thema finden sich in Em002-6 FAQ zu OSS* in Kapitel 2.2. Von einem LLM erstellter Code ist selbst nicht urheberrechtlich geschützt. Die vom LLM verwendeten Trainingsdaten können jedoch urheberrechtlich geschützt sein. Entspricht der Code 1:1, so sollte auch die Lizenz übernommen werden. Wurde der Code stückweise erstellt und anschliessend von den Entwickelnden verändert, so ist das Risiko gering, dass er als abgeleitete Software gilt. Zu beachten ist, dass bestimmte LLM eine entsprechende Kennzeichnung des Codes verlangen; allfällige vertragliche Vereinbarungen dazu sind einzuhalten.

8.9. Contributor Licence Agreements (CLA) und Developer Certificate of Origin (DCO)

Arbeiten mehrere Autorinnen und Autoren am Code eines Open-Source-Projekts, so erhalten sie gemeinsame Rechte am entstehenden Code.

Die Lizenzierung des Codes an Dritte erfolgt über die OSS-Lizenz, wobei zwischen jeder nutzenden Person und jeder Autorin bzw. jedem Autor ein eigener Lizenzvertrag entsteht (Lizenzbündel). Diese Konstellation kann zu einer unübersichtlichen Rechtslage führen, etwa in internationalen Verhältnissen (siehe 8.1 oben). Entscheiden sich Unternehmen, ihre Software unter eine OSS-Lizenz zu stellen und Beiträge Dritter anzunehmen, so haben sie mitunter das Bedürfnis, eine gewisse Kontrolle über den Code zu behalten.

So kann es etwa interessant sein, die Möglichkeit offenzuhalten, das Projekt unter eine neue Lizenz zu stellen, oder man möchte unter Copyleft lizenzierten Code einer dualen Lizenzierung unterziehen und benötigt dafür mehr Rechte, als man von den Beitragenden über eine einfache OSS-Lizenz erhielte (siehe 8.5 oben). Contributor Licence Agreements (mitunter auch Copyright Transfer Agreements) sollen solche Probleme lösen.

Ein **Contributor Licence Agreement (CLA)** ist eine **rechtsverbindliche Vereinbarung**, welche die **Bedingungen für Projektbeiträge** festlegt.

Beitragende sind verpflichtet, dieser Vereinbarung zuzustimmen, bevor Inhalte zum Projekt beigetragen werden können. Im Wesentlichen räumen die Beitragenden dem Projekt und seinen Nutzenden das Recht ein, die geleisteten Beiträge zu **nutzen, zu verändern und weiterzuverbreiten**. Zudem wird bestätigt, dass die Beiträge **eigenständig verfasst oder von Dritten legitimiert wurden**.

Das **Developer Certificate of Origin (DCO)** ist ein 2004 von der Linux Foundation **eingeführtes Prüfsystem**, das dem Contributor Licence Agreement (CLA) im Zweck ähnlich ist und ein geeignetes alternatives Instrument darstellt. Die Beitragenden bestätigen für jeden Projektbeitrag, dass **(a) sie befugt sind, den Beitrag zu leisten, (b) der Beitrag von einer mit dem Projekt kompatiblen Lizenz erfasst ist und (c) sie der Verbreitung und Nutzung des Beitrags als Open Source zustimmen**.

Das Hauptinstrument von CLA ist entweder die Abtretung der Urheberrechte von den Beitragenden an die Hauptentwicklerin bzw. den Hauptentwickler oder die tragende Organisation (bei Copyright Transfer Agreements) oder die Einräumung einer möglichst weitgehenden, in der Regel unwiderruflichen Lizenz durch die Beitragenden an die Hauptentwicklung (z. B. hat Apache ein CLA definiert). Auch Beiträge an geschlossene Projekte unterliegen häufig einem CLA.

CLA sind in der OSS-Community umstritten[11], unter anderem weil sie Lücken im Copyleft öffnen können. Daher gibt es inzwischen auch CLA, welche die Hauptentwicklung bei der Erteilung einer neuen Lizenz für den Code des Projekts einschränken.

11. Beispielsweise: https://ben.balter.com/2018/01/02/why-you-probably-shouldnt-add-a-cla-to-your-open-source-project/

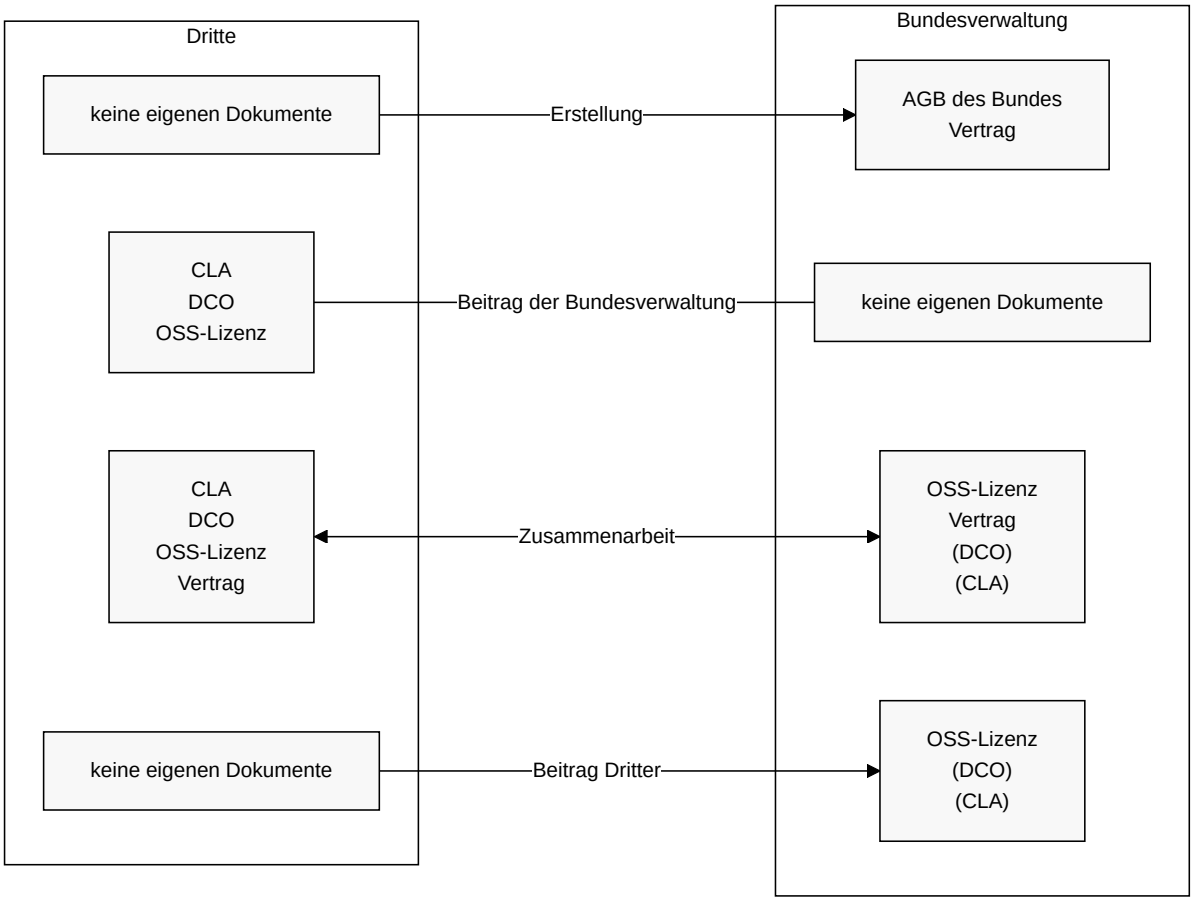


Abbildung 4: Übertragung von Quellcode und Rechten in verschiedenen Konstellationen

Legende:

- Die Kästchen zeigen, **wo die Rechte definiert sind**.
- **Erstellung** und **Beitrag Dritter**: Die Bundesverwaltung hat die Gestaltungsmacht.
- **Zusammenarbeit**: Die Dokumente werden zwischen den Parteien vereinbart.
- **Beitrag der Bundesverwaltung**: Die Bundesverwaltung hat keine Gestaltungsmacht, eine Prüfung ist nötig.
- Die Rechtsdokumente sind von oben nach unten nach Priorität geordnet; Dokumente in Klammern sind eher zu vermeiden.

Für **Eigenentwicklungen** (Creation) bestehen keine Probleme: Nach Art. 17 URG ist die Bundesbehörde Inhaberin der Nutzungsrechte an Software, die von ihren eigenen Mitarbeitenden geschaffen wurde. Bei **Drittentwickelnden** sollten sich die Bundesbehörden die Rechte sichern (beispielsweise durch Verwendung der entsprechenden AGB des Bundes). Damit ist hier kein CLA oder DCO nötig.

Wo Bundesbehörden Hauptentwicklerinnen sind und Beiträge sowie Zusammenarbeiten annehmen, sollte auf die Verwendung eines CLA verzichtet werden. Stattdessen können Digital Certificates of Origin (mit einem Beispiel erläutert in Em002-4 Leitfaden OSS-Community, Kapitel 3.3) verwendet werden. Grundsätzlich sollten Beiträge an Projekte von Bundesbehörden auf der OSS-Lizenz aufbauen, um die Hürde für Beiträge nicht weiter zu erhöhen.

Wird im Einzelfall nach entsprechender Prüfung dennoch ein Bedarf für ein CLA festgestellt, so sollte die Bundesbehörde das Apache-CLA verwenden.[12] In diesem Fall sollten dieser Entscheid und das CLA in den Schlussbemerkungen sowohl der Em002-2.1 Checkliste OSS-Vorabklärung als auch der Em002-4.1 Checkliste OSS-Community vermerkt werden.

Tragen Bundesbehörden zu Drittsoftware bei, die ein CLA verlangt, so muss der Rechtsdienst im Einzelfall prüfen, ob die Bundesbehörde das bestehende CLA annehmen kann.[13] Die folgenden Punkte sind zu bestimmen:

- Hat eine andere Bundesbehörde das CLA bereits unterzeichnet?
- Wer prüft das CLA?
- Wer unterzeichnet es? (Die unterzeichnende Person muss befugt sein, für den Bund gegenüber Dritten zu unterschreiben)
- Wo wird es abgelegt?
- Wie wird das CLA oder DCO im Projekt dokumentiert?
- Wie wird sichergestellt, dass der freigegebene Code veröffentlicht bleibt? (Dies bleibt nach Art. 9 EMBAG in der Verantwortung der Bundesbehörde, siehe [KBB-MB].)

Mögliche Kriterien für die Prüfung sind:

- Vereinbarkeit mit Art. 9 EMBAG: Bleibt der Code offen?
- Sieht die Bundesbehörde Risiken für sich?
- Sieht die Bundesbehörde einen Nutzen für sich (z. B. muss der Code nicht allein gepflegt oder weiterentwickelt werden) oder für andere (innerhalb des Föderalismus der Schweiz oder europaweit lässt sich ohne Mehraufwand eine breitere Wirkung erzielen)?
- Sind die eingeräumten Rechte angemessen?
- Werden der Bundesbehörde keine (unnötigen) Pflichten auferlegt?
- Entspricht das CLA im Wesentlichen dem Apache-CLA?

8.10. Probleme mit (L)GPL-3.0 und IoT

[BITKOM2023] weist auf ein Problem bei der Verwendung von (L)GPL-3.0-Lizenzen im Zusammenhang mit geräteseitiger Software hin: Solche Geräte müssten die Installation eigener bzw. neuer Softwareversionen zulassen. Dies kann aus Sicherheitsgründen unerwünscht sein.

8.11. Rechtsstellung der Dokumentation

Nach Art. 5 URG sind «Entscheidungen, Protokolle und Berichte von Behörden und öffentlichen Verwaltungen» nicht urheberrechtlich geschützt.

Dies geht weiter als CC-0, weil die Rechtsinhaberin das Werk nicht durch weltweiten Verzicht auf alle Urheber- und verwandten Schutzrechte in die Gemeinfreiheit entlassen muss. Es ist bereits von

12. https://www.apache.org/licenses/contributor-agreements.html
13. Für eine bestimmte Empfängerin müsste die Bundesverwaltung die Unterzeichnung eines CLA nur einmal prüfen. In anderen Organisationen wird dies daher als Angelegenheit auf hoher Ebene behandelt, wobei unterzeichnete CLA in der Vertragsablage aufbewahrt werden. CLA dürfen die Pflichten der Bundesbehörde nach Art. 9 EMBAG nicht unterlaufen.

Gesetzes wegen gemeinfrei. Die Botschaft zu Art. 5 URG hält fest: «Die Bestimmung lässt den urheberrechtlichen Schutz nach wie vor für eine ganze Reihe von Werken zu, die aus amtlicher Tätigkeit oder im Zusammenhang damit entstanden sind.

Unterlagen verwaltungsinterner Studienkommissionen und Arbeitsgruppen, Gutachten oder Zeitschriften von Bundesämtern etwa fallen nicht unter die Norm. An deren freier Verbreitung besteht kein überwiegendes Interesse, weil sie die Rechtsstellung der Bürgerin und des Bürgers nicht beeinflussen.»

Damit lässt sich argumentieren, dass für viele der massgebenden Dokumente und die von Bundesbehörden im Zusammenhang mit Open-Source-Software veröffentlichte Dokumentation kein urheberrechtlicher Schutz besteht. Das Kriterium ist, ob das Dokument die Rechtsstellung der Bürgerin und des Bürgers berührt und damit vom urheberrechtlichen Schutz ausgenommen ist. Ist dies nicht der Fall, so sollte für die Dokumentation eine geeignete Lizenz (CC-0[14], CC-BY, CC-BY-SA oder allenfalls LGPL) verwendet werden.

Dennoch wäre insbesondere bei internationaler Nutzung der Dokumentation für viele Nutzende die Lage unklar.

Wir erachten es daher als zweckmässig, wenn wie bei diesem Dokument sicherheitshalber ebenfalls eine entsprechende Freigabe aufgeführt wird.

9. Weitere Informationen zu Rechtsfragen

Weitere Informationen zu Open-Source-Software, zu spezifischen Lizenzmerkmalen und zu vertieften rechtlichen Aspekten finden sich in zahlreichen Publikationen, die nachfolgend vorgestellt werden (siehe auch die Referenzen am Ende des Dokuments).

- Der Text *«Open Source Software im EMBAG, Analyse des neuen Art. 9 des Bundesgesetzes über den Einsatz elektronischer Mittel zur Erfüllung von Behördenaufgaben»* von Rika Koch und Simon Schlauri, im Tagungsband der IT-Beschaffungskonferenz 2023 in Bern, bietet zusätzliche Orientierung zur Umsetzung von Art. 9 EMBAG.

- Der *Leitfaden Open-Source-Software 2.0* des deutschen BITKOM behandelt rechtliche Fragen rund um Open Source ausführlich *[BITKOM2023]*. Er beruht jedoch auf der Rechtslage in Deutschland.

- Wolfgang Straub untersucht in seinem Buch *Softwareschutz: Urheberrecht, Patentrecht, Open Source* die rechtlichen Einzelheiten des Copylefts mit Bezug zum Schweizer Urheberrecht und geht der Kompatibilität von Open-Source-Lizenzen nach [St2011]. Das Kapitel zur Open-Source-Software (sowie deutsche Übersetzungen verschiedener Open-Source-Lizenzen unter Berücksichtigung der Schweizer Rechtsterminologie) sind unter *www.it-recht.ch* frei verfügbar.

- Till Jaeger und Axel Metzger beantworten in ihrem umfassenden Buch *Open Source Software – Rechtliche Rahmenbedingungen der Freien Software [JaAx2016]* zahlreiche Rechtsfragen rund um Open-Source-Software vertieft. Sie stützen ihre Analyse jedoch auf die Rechtslage in Deutschland.

- Kropp Jonathan/Bauer Alexander, Open Source Compliance und Litigation, CB 2019 S. 285 ff., 285.

- Reymond Michel José, Questions de responsabilité civile et contractuelle soulevées par la distribution de «logiciels libres» (open source), SZW 2022 S. 69 ff.

Zudem informieren verschiedene Online-Portale ausführlich über die Merkmale und Fragen einzelner Open-Source-Lizenzen.

14. https://www.creativecommons.ch/

- Auf der GitHub-Plattform https://choosealicense.com lassen sich die gewünschten Ziele für ein Open-Source-Projekt auswählen, worauf die passende Open-Source-Lizenz vorgeschlagen wird.
- Unter https://opensource.guide/legal/ stellt GitHub einen Online-Leitfaden bereit, der auf spezifische Rechtsfragen eingeht.
- Unter https://www.gnu.org/licenses/license-list.de.html finden sich Kurzkommentare zu verschiedenen Lizenzen mit Angaben zur Kompatibilität, insbesondere zur GPL.
- Unter https://opensource.org/faq behandelt die Open Source Initiative zahlreiche Fragen und Antworten zu den rechtlichen Aspekten von Open-Source-Lizenzen.
- Unter https://copyleft.org/guide findet sich ein ausführlicher Leitfaden, der die Einzelheiten des Copylefts erläutert.
- Unter https://tldrlegal.com werden die wichtigsten Open-Source-Lizenzen danach zusammengefasst, was die jeweilige Lizenz erlaubt («can»), was sie verbietet («cannot») und was sie vorschreibt («must»).
- Unter https://opensource.com/tags/licensing werden laufend Artikel zu aktuellen Lizenzfragen veröffentlicht.
- Unter https://www.ifross.org/faq-haeufig-gestellte-fragen hat das private «Institut für Rechtsfragen der Freien und Open Source Software» in Berlin (ifrOSS) zahlreiche Antworten auf häufig gestellte Rechtsfragen veröffentlicht.

Die Einhaltung von Open-Source-Lizenzen wird heute in erster Linie mit Softwarewerkzeugen umgesetzt (zu weiteren Informationen zur Open-Source-Compliance siehe auch Fröhlich-Bleuler *[Fr2012]* sowie Kuhn, Williamson und Sandler *[KuWiSa2008]*). So sind etwa verschiedene Open-Source-Werkzeuge der Linux Foundation unter dem Begriff «fossology» unter https://www.fossology.org dokumentiert und veröffentlicht. Auch kommerzielle Anbieter wie Black Duck[15] oder FOSSA[16] bieten verschiedene proprietäre Lösungen an, mit denen die Kompatibilität der verwendeten Open-Source-Lizenzen geprüft werden kann.

10. Anhang

10.1. Änderungen gegenüber der Vorversion

- Management Summary ergänzt.
- AGPL- und MIT-Lizenz werden neu als vorzugswürdig empfohlen.
- Neu in Kapitel 8.8: Urheberrecht für mit KI erzeugten Code.
- Kapitel 8.9 CLA und CDO stark erweitert.
- Verschiedene redaktionelle Verbesserungen und Präzisierungen.
- Mehr Beispiele aufgeführt.

10.2. Referenzen

Siehe Em002 Strategische Leitlinien für Open-Source-Software in der Bundesverwaltung.

15. https://www.blackducksoftware.com
16. https://fossa.com

10.3. Abkürzungen

Siehe Em002 Strategische Leitlinien für Open-Source-Software in der Bundesverwaltung und Em002-6 FAQ zu OSS.

10.4. Beispiele von Freigaben und ihrer Lizenz

Künftig wird diese Liste durch eine Sammlung von Metadaten über publiccode.yml ersetzt.

10.5. D.1 Loom

URL
https://gitlab.com/swiss-armed-forces/cyber-command/cea/loom

Bundesbehörde
Cyber Command

Beschreibung
Loom ist eine leistungsfähige, einfach zu bedienende Dokumentensuchmaschine. Sie automatisiert die Indexierung konfigurierter Datenquellen, führt OCR durch, extrahiert Inhalte und Metadaten, ermöglicht Tagging und bietet leistungsfähige Such- und Bedienmöglichkeiten.

Jahr
2025

Lizenz
MIT

Begründung
-

10.6. D.2 trustbroker.swiss

URL
https://github.com/trustbroker-swiss/trustbroker.swiss

Bundesbehörde
BK, BIT

Beschreibung
Trust Broker Swiss stellt Föderationsdienste zwischen Relying Parties (Anwendungen, Service Provider, andere IAM-Systeme oder Policy Enforcement Points) und Identity Providern (IdP, auch Claims Provider genannt) bereit und nutzt vertrauenswürdige Attributspeicher, um authentifizierte Nutzende anzureichern.
Es ermöglicht Identity-/Claims-Providern und Relying Parties, Informationen über eine dritte Partei auszutauschen, welche die IdP-Spezifika verbirgt und eine einheitliche oder zumindest zusätzlich verifizierte Identität bereitstellt.

Jahr
2024

Lizenz
AGPL

Begründung
Codeänderungen sollen stets offen bleiben und zurückfliessen.

Für den Trust Broker wurde eine sehr detaillierte Prüfung der Lizenzfrage durchgeführt. Bei sehr grossen Projekten kann sie in dieser Detailtiefe erfolgen. Alle kompatiblen Lizenzen, die mit der Software Bill of Materials vereinbar sind, wurden in einem Lizenzpool aufgeführt.

10.7. D.3 Geocat.ch

URL
https://github.com/geonetwork/core-geonetwork

Bundesbehörde
Swisstopo

Beschreibung
GeoNetwork ist eine Kataloganwendung zur Verwaltung raumbezogener Ressourcen. Sie bietet leistungsfähige Funktionen zur Metadatenbearbeitung und -suche sowie einen interaktiven Webkartenbetrachter. Sie wird derzeit in zahlreichen Initiativen für Geodateninfrastrukturen weltweit eingesetzt.

Jahr
2012

Lizenz
GPL 2.0

Begründung
-

10.8. D.4 GWEN / ampycloud / c4dl-multi / dvas /…

URL
https://meteoswiss.github.io/ampycloud/

Bundesbehörde
MeteoSchweiz

Beschreibung
Python-Paket zur Bestimmung des Bedeckungsgrads und der Basishöhe von Wolkenschichten anhand von Ceilometer-Daten.

Jahr
2023

Lizenz
Verschiedene Lizenzen, FSS

Begründung
-

10.9. D.5 Anwendung EMSG (Erhaltungsmanagement in Siedlungsgebieten)

URL
https://github.com/astra-emsg/ASTRA.EMSG

Bundesbehörde
MeteoSchweiz

Beschreibung
EMSG ist eine C#-GIS-Webanwendung, die von der Schweizer Bundesverwaltung entwickelt wurde, um das Erhaltungsmanagement städtischer Strassensysteme zu verwalten.

Jahr
2017

Lizenz
BSD

Begründung
-

10.10. D.6 Anwendung Covid-Zertifikat

URL
https://github.com/admin-ch/CovidCertificate-Documents

Bundesbehörde
BIT

Beschreibung
Schweizer Anwendung für COVID-Zertifikate

Jahr
2020

Lizenz
MIT

Begründung
So permissiv wie möglich, damit alle den Code nutzen können.

10.11. D.7 GovCert-Website

URL
https://github.com/govcert-ch/website

Bundesbehörde
NCSC

Beschreibung
Quellcode der Website für das Computer Emergency Response Team (GovCERT) der Schweizer Bundesverwaltung

Jahr
2023

Lizenz
MIT

Begründung
-

10.12. D.8 Verschiedene Bibliotheken zu Open Data (z. B. Linked Data)

URL
https://github.com/SwissFederalArchives

Bundesbehörde
BAR

Beschreibung
Das Repository des Schweizerischen Bundesarchivs auf GitHub bietet Zugang zu den Quellcodes unserer Anwendungen. Es erlaubt Ihnen, eigene Forks zu erstellen und über die Schnittstelle «Issues» Fehler zu melden oder erweiterte Funktionalitäten vorzuschlagen.

Jahr
2023

Lizenz
AGPL, Apache, MIT

Begründung
Opendata und Open-Source-Software gehen in dieselbe Richtung. Die AGPL erlaubt maximale Offenheit.

10.13. D.9 Apache FOP angepasst für archivierbares PDF

URL
https://xmlgraphics.apache.org/fop/

Bundesbehörde
IGE

Beschreibung
Das IGE wollte eine Version von Apache FOP, die archivierbare PDF erzeugen kann. Dazu wurde bei einem der Kernentwickler (Jeremias Märki) eine Erweiterung in Auftrag gegeben. Die Anpassungen flossen direkt in das Projekt ein.

Jahr
2007

Lizenz
Apache

Begründung
Erweiterungen wurden an einem bestehenden Open-Source-Projekt vorgenommen.

10.14. D.10 Rechtskonforme elektronische Eingabe mit eKomm

URL
https://launchpad.net/ekomm

Bundesbehörde
IGE

Beschreibung
Das IGE wollte Software für die elektronische Eingabe.

Jahr
2009

Lizenz
Apache

Begründung
Das IGE wollte, dass der Anbieter (Glue) die Freigabe vornimmt, und hat diese als permissiv bewilligt (Integration in eigene Anwendungen).