Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Chancellerie fédérale ChF
Transformation numérique et gouvernance de l'informatique TNI

14.09.2026

Em002-1 Guide pratique pour les logiciels open source dans l’administration fédérale

Version 2.0-0be65ca

Table des matières

1. 1. Utilisation du présent guide 3
1.1. Intention / parties prenantes 3
2. Définitions 5
3. Potentiel et défis 7
3.1. Potentiel des logiciels open source 7
3.2. Défis liés à l'OSS 9
4. Constellations d'utilisation et de travail avec l'OSS 11
4.1. Degrés de maturité pour l'OSS dans l'administration fédérale 11
4.2. Utilisation de logiciels open source non modifiés 13
4.3. Développement avec des composants open source et publication du code source 13
4.4. Participation à des projets ouverts (contribution) 13
5. Caractéristiques et choix des licences open source 14
6. Acquisition d'OSS 14
7. Repérabilité des solutions OSS 14
7.1. Catalogue OSS 14
7.2. OSS Directory 14
7.3. alternativeTo 15
7.4. Dépôts open source 15
7.5. Open Hub 16
7.6. Dépôts de code particuliers 17
8. Modèles de support pour l'utilisation de l'OSS 17
8.1. Utilisation sans support professionnel 17
8.2. Utilisation avec support interne 18
8.3. Utilisation avec support d'un fournisseur externe 18
9. Point de contact 19
10. Annexes 19
10.1. Modifications par rapport à la version précédente 19
10.2. Références 19
10.3. Abréviations 19

10.4. Modèles d’affaires avec l’OSS 19

10.4.1. Services et produits fondés sur l’OSS 19

10.4.2. Services pour l’OSS 20

10.4.3. Abonnements 20

10.4.4. Double licence 20

> **NOTE** Le présent document est un projet en cours d'élaboration et fait partie des outils destinés à soutenir l'administration fédérale dans la publication de code open source. Pour de plus amples informations, voir le README principal.

1. 1. Utilisation du présent guide

Conformément à l'article 9 de la loi fédérale sur l'utilisation des moyens électroniques pour l'exécution des tâches des autorités (LMETA), la Confédération est tenue de publier le code source des logiciels qu'elle développe ou fait développer pour accomplir ses tâches. [1] Le secteur TNI de la Chancellerie fédérale a déduit de ce mandat légal des objectifs généraux, exposés dans les Lignes directrices stratégiques pour les logiciels open source.

Le présent document constitue un guide pratique relatif aux exigences juridiques et stratégiques. Il vise à soutenir les unités administratives fédérales dans l'utilisation, l'acquisition et la publication de logiciels open source. Il offre à toutes les personnes intéressées une vue d'ensemble complète du sujet et renvoie à des informations complémentaires figurant dans les autres outils.

Selon vos connaissances préalables et vos besoins, vous pouvez toutefois lire les guides de manière sélective avant de passer aux autres outils. L'aperçu suivant vous aidera à trouver le point de départ approprié. En cas de doute, il est recommandé de lire d'abord le guide.

1.1. Intention / parties prenantes

Introduction générale aux logiciels open source

- Chapitre Définitions
- Chapitre Potentiel et défis
- Chapitre Repérabilité des solutions OSS
- Em002-5 Aide-mémoire Outils OSS
- Em002-3 Guide de licences OSS
- Em002-6 FAQ sur l'OSS

Introduction générale à la LMETA pour les décideurs

- Em002-5 Aide-mémoire Outils OSS
- Em002 Lignes directrices stratégiques pour les logiciels open source dans l'administration fédérale

Acquisition uniquement de logiciels destinés à être utilisés

- Chapitre Degrés de maturité pour l'OSS dans l'administration fédérale
- Chapitre Utilisation de logiciels open source non modifiés
- Em002-7 Aspects stratégiques des achats et OSS
- Aide-mémoire Acquisition de logiciels et art. 9 LMETA [KBB-MB]
- Directives OSS dans les acquisitions [BBL-WL]

Nouveau développement informatique et publication

1. RS 172.019 : https://www.fedlex.admin.ch/eli/cc/2023/682/fr

- Chapitre Développement avec des composants open source et publication du code source
- Chapitre Participation à des projets ouverts (contribution)
- Em002-2 Instructions pour la publication d'OSS
- Em002-2.1 Liste de contrôle Évaluation préalable OSS
- Em002-2.2 Liste de contrôle Analyse et préparation OSS
- Em002-2.3 Liste de contrôle Validation et publication OSS
- Em002-4 Guide des communautés OSS
- Em002-4.1 Liste de contrôle Communauté OSS

Acquisition de logiciels à créer (prestation de services ou ouvrage)

- Chapitre Degrés de maturité pour l'OSS dans l'administration fédérale
- Chapitre Développement avec des composants open source et publication du code source
- Chapitre Participation à des projets ouverts (contribution)
- Em002-7 Aspects stratégiques des achats et OSS
- Boîte à outils de l'OFCL, y compris les directives Open Source dans les acquisitions
- Plateforme d'apprentissage et de modèles de l'administration publique du CCMP (www.perimap.admin.ch), y compris Acquisition de logiciels et art. 9 LMETA [KBB-MB]
- Directives OSS dans les acquisitions [BBL-WL]
- Em002-2 Instructions pour la publication d'OSS
- Em002-2.1 Liste de contrôle Évaluation préalable OSS
- Em002-2.2 Liste de contrôle Analyse et préparation OSS
- Em002-2.3 Liste de contrôle Validation et publication OSS
- Em002-4 Guide des communautés OSS
- Em002-4.1 Liste de contrôle Communauté OSS

Clarifications juridiques en vue de la publication

- Em002-6 Foire aux questions sur la publication d'OSS (FAQ-OSS)
- Em002-3 Guide de licences OSS

Principes de travail pour les cheffes et chefs de projet

- Documents HERMES
- Em002-5 Aide-mémoire Outils OSS

Contributions logicielles à des tiers et collaborations

- Em002-4 Guide des communautés OSS

- Em002-4.1 Liste de contrôle Communauté OSS
- Em002-7 Aspects stratégiques des achats et OSS
- Em002-3 Guide de licences OSS

Le schéma suivant donne une vue d'ensemble des documents pertinents pour l'OSS au sein de l'administration fédérale.

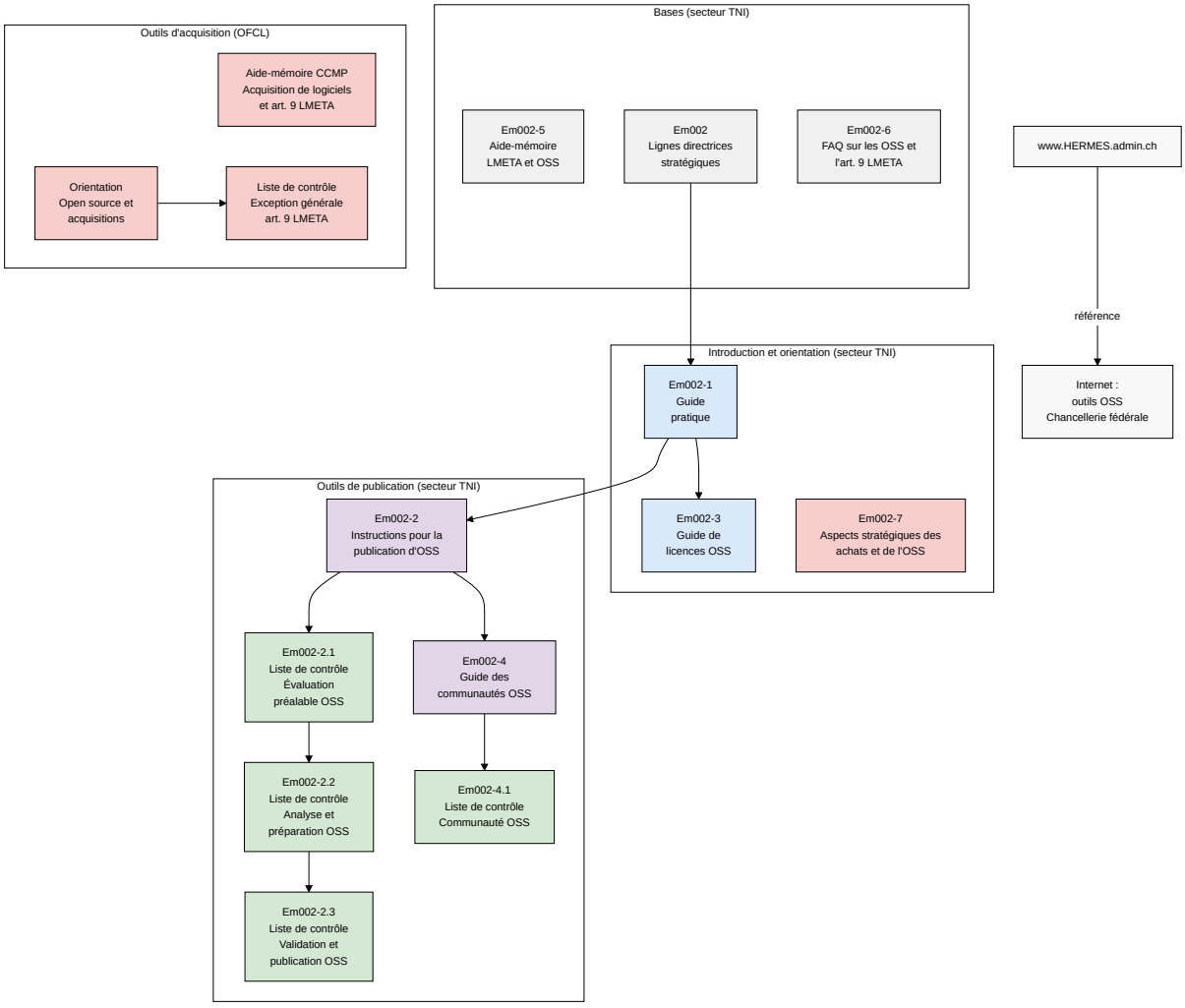


Figure 1 : Vue d'ensemble des outils OSS en lien avec l'art. 9 LMETA

2. Définitions

Logiciel open source

Un logiciel est considéré comme un logiciel open source (OSS) lorsqu'il est publié sous l'une des quelque 80 licences reconnues par l'Open Source Initiative OSI *[OSI2024]*. Concrètement, la définition de l'open source énonce dix critères que toutes les licences open source doivent remplir *[Pe1999]*. Ceux-ci peuvent pour l'essentiel se résumer en quatre points :

1. Le logiciel peut être utilisé à toutes fins.
2. Le code source du logiciel est librement accessible.
3. Le logiciel peut être copié et diffusé sans restriction.
4. Le logiciel peut être modifié et diffusé sous forme modifiée à certaines conditions.

Logiciel libre

Avant le terme « open source », la Free Software Foundation (FSF) a introduit dans les années 1980 la notion de « logiciel libre ». Sur le fond, le logiciel libre remplit les conditions du logiciel open source, mais vise à garantir que le logiciel reste toujours librement accessible et, dans la mesure du possible, ne soit pas intégré à des logiciels propriétaires. [2]

Logiciel propriétaire

Dans le cas d'un logiciel propriétaire, un fournisseur développe le logiciel et vend à l'utilisateur une licence d'utilisation. En règle générale, l'utilisateur ne connaît pas le code du logiciel et ne peut ni le rediffuser ni le modifier. Il peut uniquement utiliser le logiciel selon les conditions de licence (p. ex. End-User Licence Agreement, EULA) moyennant le paiement de redevances de licence. Le logiciel peut par exemple n'être autorisé que pour un certain nombre d'utilisateurs ou un certain nombre de processeurs. [3] En contrepartie des frais de maintenance généralement annuels, le fournisseur s'engage à corriger les erreurs dans un délai raisonnable et à développer le logiciel en continu.

Licence

Une licence est un document qui contient des prescriptions contraignantes relatives à l'utilisation et à la diffusion d'un logiciel. Dans le domaine de l'open source, on utilise généralement des licences conformes à l'OSI. [4]

Communauté

La notion de communauté dans le contexte des logiciels open source est définie largement. Il peut s'agir d'un écosystème informel ou d'une structure dotée d'une gouvernance qui développe le logiciel. Le document Em002-4 Guide des communautés OSS traite de ce sujet. Les membres de la communauté peuvent gérer le produit en commun, le développer, le tester, le traduire, donner leur avis ou simplement utiliser le logiciel.

Droits de tiers

Même pour les logiciels open source, des droits de protection peuvent exister (droit d'auteur, droit des marques, droit des brevets) et être invoqués par des tiers, p. ex. lors de l'utilisation de code source créé par des tiers.

Exceptions selon la LMETA

Les exceptions prévues à l'art. 9, al. 1, LMETA sont les droits de tiers et les motifs relevant de la sécurité. Les deux sont traités au chapitre 3 de Em002-2 Instructions pour la publication de logiciels open source.

Code source

En informatique, le code source (ou texte source) est le texte lisible par l'être humain d'un programme informatique, rédigé dans un langage de programmation. [5]

Publication

La publication désigne en premier lieu la mise à disposition du code source sous une licence ouverte conforme à l'OSI. Pour des raisons pratiques, elle comprend généralement aussi la documentation et les instructions automatisées permettant de construire l'application à partir du code source. Les tests automatisés et les documents de test en font aussi habituellement partie. Pour qu'une communauté puisse se développer, une plateforme est utilisée, laquelle assume alors plusieurs fonctions d'une plateforme de développement (build, test, support).

Souveraineté numérique

Nous utilisons la définition du rapport du DFAE sur la souveraineté numérique [St2024] : « La souveraineté numérique d'un État ou d'une organisation comprend nécessairement le contrôle complet des données stockées et traitées ainsi que la décision autonome de savoir qui peut y

2. Logiciel libre — Wikipédia
3. Logiciel propriétaire — Wikipédia
4. Open Source Initiative
5. Code source — Wikipédia

accéder. Elle comprend en outre la capacité de développer, de modifier, de contrôler et de compléter de manière autonome des composants et des systèmes technologiques. »

Dépôt (repository)
Un dépôt est un emplacement de stockage utilisé par un outil de gestion de versions pour les fichiers et les métadonnées relatives à la base de code. Les dépôts permettent à plusieurs contributeurs de travailler sur les mêmes fichiers et d'enregistrer différentes versions. La plupart permettent en outre le suivi des problèmes, la gestion des versions, les compilations automatisées et la documentation.

Fork
Un fork [6] d'un projet open source est le fait de mettre en place un développement indépendant du projet initial.

Contributor Licence Agreement (CLA)
Un Contributor Licence Agreement [7] est un document qui décrit les conditions auxquelles de la propriété intellectuelle peut être apportée à un projet ou à une entreprise.

Source code management (SCM)
La gestion du code source [8], également appelée gestion de versions, est un outil qui gère efficacement les modifications et les numéros de version, en particulier dans le développement logiciel. Le logiciel libre Git [9] s'est imposé pour la gestion des SCM distribués.

Open source software development (OSSD)
Dans le développement de logiciels open source (OSSD) [10], outre la publication du code source, l'ensemble du processus de développement se déroule publiquement. Tout, des exigences (issues) au code source, est transparent. L'OSSD s'appuie sur des outils de communication publics (listes de diffusion, forums, etc.), un système de gestion de versions (Git), des listes d'erreurs et de fonctionnalités, une feuille de route et des outils de développement.

Software Package Data Exchange (SPDX)
Software Package Data Exchange (SPDX) [11] décrit un format standard pour les Software Bill of Materials (SBOM), dans le but de faciliter le traitement correct des logiciels libres ou open source.

Effet copyleft
Si un logiciel est placé sous une licence comportant une clause de copyleft, toute modification ou extension du code source doit être publiée à nouveau sous la licence du logiciel open source modifié (voir aussi Em002-3 Guide de licences OSS).

3. Potentiel et défis

Les logiciels open source recèlent un potentiel énorme dans le monde informatique actuel. Parallèlement, l'utilisation de solutions open source comporte aussi des défis. Ces deux aspects sont expliqués plus en détail dans la section suivante. Les indications se fondent notamment sur les résultats de l'Étude Open Source Suisse 2018, 2021 et 2024 de l'Université de Berne, dans laquelle les personnes interrogées ont donné des informations sur les avantages et les inconvénients reconnus des logiciels open source.

3.1. Potentiel des logiciels open source

Les points suivants décrivent le potentiel qui peut être réalisé lors de l'utilisation et de la publication de logiciels open source.

6. Fork (développement logiciel) — Wikipédia
7. Contributor License Agreement — Wikipédia
8. Gestion de versions — Wikipédia
9. Git — Wikipédia
10. Open-source software development — Wikipédia
11. Software Package Data Exchange — Wikipédia

1. **Souveraineté numérique**
   Il s'agit ici de la capacité d'utiliser et de contrôler des services numériques. La souveraineté numérique s'étend sur l'ensemble du cycle de vie d'un système numérique. Les logiciels open source favorisent la souveraineté numérique, car ils garantissent l'interchangeabilité, la liberté de conception et l'influence. [12]

2. **Pas de redevances de licence**
   L'utilisation de logiciels open source n'entraîne aucun coût de licence. Lors de l'acquisition de progiciels open source standard complexes, il peut toutefois être judicieux d'acheter des abonnements de support, qui sont payants.

3. **Économies grâce à la coopération avec d'autres utilisateurs**
   Comme un logiciel placé sous licence open source peut être utilisé et développé sans restriction, les coûts des développements ultérieurs peuvent être partagés ou des développements complémentaires existants d'autres unités administratives peuvent être repris selon le principe « développer une fois — utiliser plusieurs fois ». Il est en même temps possible de profiter de l'expérience et des développements des autres.

4. **Constitution de communautés et partage des connaissances**
   Les logiciels open source facilitent la constitution de communautés et le partage des connaissances, par exemple entre les différents niveaux fédéraux en Suisse. Les logiciels peuvent être développés en commun, les erreurs corrigées et les expériences individuelles partagées. L'échange accru de connaissances entre différentes unités administratives conduit à une meilleure compréhension de ce sur quoi travaillent les autres, de sorte que les doublons sont évités et que les meilleures solutions peuvent se diffuser.

5. **Moindre dépendance vis-à-vis des fabricants**
   Le vendor lock-in (dépendance envers les fournisseurs de logiciels) est considéré comme très élevé en informatique. L'utilisation de logiciels sous licence open source réduit le vendor lock-in, car l'exploitation, la maintenance, le support, le développement ultérieur et d'autres prestations pour les logiciels open source peuvent faire l'objet d'appels d'offres ouverts. Au lieu de réacquérir entièrement un système complet en raison d'un vendor lock-in, des extensions ou des prestations de cycle de vie pour une solution OSS peuvent aussi faire l'objet d'appels d'offres distincts. Cela permet d'économiser des coûts de migration considérables.

6. **Standards ouverts et interopérabilité**
   Avec les logiciels open source, la compatibilité avec d'autres solutions logicielles et systèmes informatiques (interopérabilité) est généralement plus élevée qu'avec les logiciels propriétaires. Les solutions open source utilisent en outre presque exclusivement des formats de données ouverts, ce qui permet de les échanger facilement avec d'autres systèmes.

7. **Transparence sur la structure du logiciel**
   Comme le logiciel est également disponible sous forme de code source, sa qualité peut être vérifiée, par exemple par des revues externes. De plus, de la documentation peut être créée à partir du code source (p. ex. en vue de nouveaux appels d'offres pour des prestations de développement ou du remplacement d'un logiciel open source en fin de vie).

8. **Sécurité et confiance grâce à la transparence**
   En raison du caractère ouvert du code source, la sécurité des logiciels open source peut être supérieure à celle des logiciels propriétaires. En outre, il est nettement plus difficile d'intégrer des portes dérobées et d'autres failles dans un logiciel open source, ce qui accroît la confiance dans le logiciel.

---

12. Nous utilisons la définition de *[St2024]* : « La souveraineté numérique d'un État ou d'une organisation comprend nécessairement le contrôle complet des données stockées et traitées ainsi que la décision autonome de savoir qui peut y accéder. Elle comprend en outre la capacité de développer, de modifier, de contrôler et de compléter de manière autonome des composants et des systèmes technologiques. »

9. Qualité et modularité accrues du code

La qualité des logiciels open source peut être supérieure à celle des logiciels propriétaires, car la motivation d'écrire du bon code est probablement plus grande lorsque les développeurs savent que leur code source sera publié. De plus, les solutions open source tendent à être fortement modulaires, de sorte que des modules individuels peuvent être facilement remplacés et les autres modules continuer à être utilisés. [13]

10. Adaptation aisée

L'accès au code source permet aux utilisateurs de réaliser eux-mêmes des développements ultérieurs ou de les confier à des fournisseurs externes. Le logiciel peut ainsi être rapidement adapté à leurs propres besoins.

11. Innovation et intégration rapides

Les logiciels open source font l'objet d'un développement rapide et continu par la communauté. Ainsi, les nouvelles technologies et les nouveaux standards de données sont souvent publiés sous forme de bibliothèques de programmation open source. Cela permet la réalisation rapide de solutions logicielles innovantes.

12. Attractivité de l'employeur

L'utilisation de technologies open source modernes et la collaboration informelle avec des communautés internationales favorisent la motivation des collaborateurs et accroissent ainsi l'attractivité de l'employeur, ce qui facilite à son tour le recrutement de jeunes professionnels qualifiés.

3.2. Défis liés à l'OSS

Sont décrits ci-après les défis typiques rencontrés dans la pratique avec les logiciels open source, ainsi que quelques pistes de solution. Nous n'abordons pas ici les défis généraux qui concernent tous les logiciels (p. ex. la nécessité de vérifier la cybersécurité et d'assurer un support approprié pour les logiciels critiques).

1. Coûts de migration élevés dus aux dépendances existantes

Le remplacement d'un logiciel propriétaire par une solution open source peut être très coûteux si l'application est étroitement intégrée au système informatique existant par des interfaces ou d'autres dépendances. Ces coûts de migration font que l'introduction d'un logiciel open source n'est souvent rentable qu'à la fin du cycle de vie du logiciel propriétaire.

2. Fonctionnalités manquantes ou absence de solution open source adaptée

Lors de l'acquisition de logiciels standard, il peut arriver qu'il n'existe aucune alternative open source à la solution propriétaire (précédente), ou seulement une alternative fonctionnellement insuffisante. Comme piste de solution, il est possible de développer les fonctionnalités manquantes conjointement avec d'autres utilisateurs du logiciel open source et de les ajouter au projet open source. [14]

3. Coûts d'intégration plus élevés

Les économies à court terme résultant de l'absence de licences lors de l'utilisation de logiciels open source sont souvent compensées par des coûts d'intégration plus élevés. Il est donc

13. Un autre aspect de sécurité est qu'il est plus facile de documenter les bibliothèques utilisées (y compris les dépendances) que dans un logiciel propriétaire. Cela signifie à son tour que, lorsque des vulnérabilités apparaissent, elles peuvent être rapidement identifiées et corrigées (voir aussi SBOM).

14. Les responsables de projet ou d'application sont compétents pour de telles nouvelles fonctionnalités. Le projet détermine exactement comment procéder. Em002-2 et Em002-4 aident à prendre des décisions détaillées. Selon l'ampleur de la fonctionnalité, il peut ne pas être judicieux de parcourir les instructions dans leur intégralité. Les aspects du droit des marchés publics relatifs aux ressources doivent être respectés.

important de reconnaître que l'utilisation professionnelle de logiciels open source n'est pas « gratuite », mais génère bel et bien des coûts internes ou externes.

4. Responsabilités et support peu clairs

Les logiciels open source sont souvent critiqués pour l'absence de support et de maintenance assurés par des entreprises externes. Il existe toutefois désormais une multitude de fournisseurs proposant des prestations commerciales (support à long terme, maintenance continue, développement ultérieur, responsabilité et garantie, etc.) pour les solutions open source. Les différents modèles d'affaires liés aux logiciels open source sont expliqués à l'annexe D et les différents modèles de support au chapitre 8. Une vue d'ensemble actuelle des prestataires open source et de leurs services figure dans l'Open Source Directory (https://www.ossdirectory.com) ou dans l'Étude OSS 2024 [15] *[OSS2024]*.

5. Marché restreint avec peu de fournisseurs

Comme certaines applications open source, telles que les applications de bureau, ne nécessitent généralement que peu ou pas de support et de maintenance, le marché des fournisseurs open source de support et de maintenance peut être restreint. Dans ce cas, un appel d'offres portant sur des prestations de services pour logiciels open source peut contribuer à créer un marché qui accroît la concurrence entre les fournisseurs. L'existence d'une large communauté de développement pour le logiciel concerné et la qualité de la documentation de développement revêtent une importance pratique lors de l'appel d'offres pour de telles prestations.

6. Faible visibilité

Les communautés des projets open source se concentrent avant tout sur le développement du produit et rarement sur le marketing. À l'inverse, les fabricants de logiciels propriétaires investissent massivement dans la commercialisation et la vente de leurs produits. Ce manque de publicité pour les logiciels open source donne souvent la fausse impression qu'il n'existe pas d'alternative aux produits propriétaires. Il existe cependant des plateformes telles qu'alternativeTo [16] qui offrent une vue d'ensemble des solutions répondant à des tâches précises (plus d'informations sur les alternatives OSS au chapitre « Repérabilité des solutions OSS »).

7. Manque d'acceptation par les utilisateurs finaux

En raison d'interfaces utilisateur différentes, de fonctions manquantes, d'une faible convivialité et du peu de publicité pour les logiciels open source, certains produits open source souffrent d'un manque d'acceptation de la part des utilisateurs finaux. Des offres appropriées de communication, de documentation et de formation peuvent contrer ce défi. Le fait qu'un logiciel open source puisse être développé selon les besoins des utilisateurs peut également conduire à une meilleure acceptation.

8. Expertise interne faible ou inexistante

Les solutions open source évoluent rapidement et exigent en même temps une compréhension technique approfondie. Il peut donc arriver que les collaborateurs internes disposent de peu ou pas d'expertise sur certains logiciels open source. Le savoir-faire OSS interne peut être développé par la formation continue et des possibilités d'auto-apprentissage via des sources en ligne, la constitution de wikis internes, etc.

9. Évaluation difficile de l'avenir

Il est souvent difficile, pour une personne extérieure à une communauté open source, de reconnaître comment le projet évoluera à l'avenir. Il est donc important de pouvoir évaluer de manière réaliste le développement futur d'une solution open source. Le chapitre « Open Hub » présente à cet égard la plateforme Open Hub, qui permet d'évaluer l'activité et l'hétérogénéité de la communauté de développement. Cela permet de mieux apprécier le développement futur d'un projet open source.

10. Incertitudes juridiques

La multitude de licences open source différentes et le faible nombre de décisions judiciaires rendues à ce jour sur des questions d'interprétation peuvent parfois engendrer des incertitudes

15. Étude OSS
16. https://alternativeto.net/

juridiques avec les logiciels open source. Le présent guide pratique vise à donner une vue d'ensemble des principales licences open source ainsi que de leurs caractéristiques et compatibilités. Des sources pour des réponses approfondies à des questions juridiques figurent dans les documents Em002-3 Guide de licences OSS et Em002-6 FAQ sur l'OSS et l'art. 9 LMETA.

11. Offre trop vaste de logiciels open source
Le nombre de produits logiciels open source a fortement augmenté ces dernières années. Les utilisateurs potentiels de logiciels open source déplorent dès lors la multitude de solutions open source existant sur le marché. C'est pourquoi le chapitre « Repérabilité de… » présente deux plateformes, alternativeTo et Open Hub, qui permettent de comparer les solutions open source.

12. Charge de coordination liée au développement conjoint
Lorsque l'administration fédérale participe à des projets communs et y contribue, il en résulte une charge de coordination au sein de ces collaborations, mais aussi lorsqu'une communauté doit être associée. Idéalement, cette charge supplémentaire doit malgré tout en valoir la peine grâce à une solution améliorée, une meilleure acceptation ou une participation accrue de tiers.

4. Constellations d'utilisation et de travail avec l'OSS

4.1. Degrés de maturité pour l'OSS dans l'administration fédérale

Lors de l'utilisation et du développement d'OSS au sein de l'administration fédérale, il existe différentes constellations concernant les obligations liées aux licences open source. Un modèle de maturité typique [17] pour l'introduction de l'open source dans les organisations peut se présenter ainsi :

1. Denial — aucune utilisation de logiciels open source, ou utilisation inconsciente
2. Consumption — utilisation passive de logiciels open source
3. Participation — engagement auprès des communautés open source
4. Contribution — contributions pragmatiques à des projets open source
5. Leadership — engagement stratégique en faveur de l'open source pour créer de la valeur [18]

Constellation	Conséquence
La simple **utilisation** de logiciels open source existants (sans modification du code).	Il n'existe **aucune obligation** de partager le code source.
Le **développement entièrement nouveau** d'un logiciel destiné à être publié sous une licence open source.	**Dans ce cas, il existe une liberté totale quant à la licence sous laquelle le logiciel est publié.** Pour la Confédération, il convient de se référer au document Em002-3 Guide de licences OSS. Les documents Em002-2 Instructions pour la publication d'OSS et Em002-4 Guide des communautés OSS doivent être respectés.

17. Modèle de maturité inspiré de https://todogroup.org/resources/guides/a-guide-to-outbound-open-source-software/#maturity-levels
18. La création n'est pas mentionnée, car l'art. 9 LMETA l'exige directement.

Constellation	Conséquence
Développement ultérieur (interne ou externe) avec des composants open source existants (avec ou sans effet copyleft [19]), tant que le logiciel est utilisé **exclusivement au sein de la même organisation** et n'est pas rediffusé.	Les composants OSS peuvent être combinés à volonté avec des composants tiers. La licence n'entraîne en principe aucune obligation de publication, tant que le logiciel est utilisé exclusivement au sein de la même organisation et n'est pas diffusé (exception : AGPL). Comme il n'est pas juridiquement établi de manière définitive jusqu'où s'étend la notion de « diffusion au sein de la même organisation », ce scénario ne s'applique aux composants OSS à effet copyleft que dans des cas exceptionnels. Dans ces cas, consultez le service juridique de votre organisation. Voir Em002-3 Guide de licences OSS et Em002-6 FAQ sur l'OSS et l'art. 9 LMETA. Les documents Em002-2 Instructions pour la publication d'OSS et Em002-4 Guide des communautés OSS doivent être respectés. **Comme une obligation de publication découle de l'art. 9 LMETA, les modifications doivent être publiées à nouveau. Le plus simple est de le faire sous forme de modifications du projet open source existant.**
Développement ultérieur (interne ou externe) **exclusivement avec** des **composants** open source existants **sans effet copyleft**, lorsque le logiciel doit être diffusé à l'extérieur (p. ex. aux cantons).	Les composants OSS peuvent être combinés à volonté avec des composants tiers, et les licences n'entraînent aucune obligation de publication (voir aussi Em002-3 Guide de licences OSS). **La publication est toutefois obligatoire dans le cadre de l'art. 9 LMETA. Idéalement, elle devrait se faire via le projet existant (pas de fork).** Les documents Em002-2 Instructions pour la publication d'OSS et Em002-4 Guide des communautés OSS doivent être respectés.
Développement ultérieur (interne ou externe) avec des composants open source existants **à effet copyleft**, lorsque le logiciel doit être diffusé à l'extérieur (p. ex. aux cantons).	**Il existe une obligation de publication** en raison de la licence copyleft (voir aussi Em002-3 Guide de licences OSS concernant l'effet copyleft). Les documents Em002-2 Instructions pour la publication d'OSS et Em002-4 Guide des communautés OSS doivent être respectés.
Contribution à un projet open source existant.	**Il existe en principe une obligation de publication selon la LMETA. La gouvernance et la licence du projet s'appliquent.** Les documents Em002-2 Instructions pour la publication d'OSS et Em002-4 Guide des communautés OSS doivent être respectés.
Développement conjoint d'un projet nouveau ou existant avec une communauté et partage des coûts.	**Il existe en principe une obligation de publication selon la LMETA.** Les documents Em002-2 Instructions pour la publication d'OSS, Em002-3 Guide de licences OSS et Em002-4 Guide des communautés OSS servent **à mettre en place le projet et sa gouvernance.**

19. Explication du terme « copyleft » dans Em002-3 Guide de licences OSS

Constellation	Conséquence
Toutes les variantes de développement ou de développement ultérieur sont réalisées **par un fournisseur ou un tiers mandaté.**	**Il existe en principe une obligation de publication selon la LMETA.** Sur la base de Em002-2 Instructions pour la publication d'OSS, Em002-3 Guide de licences OSS et Em002-4 Guide des communautés OSS. Si possible, les conditions de la publication devraient être créées dès le processus d'acquisition et, au besoin, les exigences relatives à la validation du logiciel communiquées au fournisseur ou au tiers.

4.2. Utilisation de logiciels open source non modifiés

Lors de l'utilisation de logiciels open source, il n'est en principe pas déterminant de savoir sous quelle licence open source ils sont publiés, car toutes les licences open source (voir chapitre 2 « Définitions ») autorisent l'utilisation illimitée des logiciels open source, indépendamment du nombre de postes de travail, d'utilisateurs connectés simultanément, de serveurs et de processeurs, etc. sur lesquels le logiciel est utilisé.

Lors de l'utilisation de logiciels open source non modifiés, **aucune obligation de publication** ne naît, même pour les programmes placés sous une licence à effet copyleft strict (exception : licence AGPL, Affero General Public Licence).

Tant que le logiciel open source est introduit et exploité par des prestataires internes à la Confédération, cela peut se faire **sans procédure de marché public**, car « la licence OSS à elle seule ne coûte en règle générale rien au service acquéreur et n'est donc pas en soi pertinente du point de vue des marchés publics ».

Si un prestataire externe est mandaté pour la maintenance, le support et d'autres prestations relatives au logiciel open source, les exigences du droit des marchés publics doivent être respectées. Différentes variantes de support sont décrites au chapitre 9. Il est en outre important d'utiliser des critères d'aptitude et d'adjudication appropriés lors de l'acquisition de logiciels open source. Les bases à ce sujet sont exposées au chapitre 8.

4.3. Développement avec des composants open source et publication du code source

Compte tenu de l'obligation de publication prévue à l'art. 9 LMETA, il convient de suivre ici la procédure décrite dans Em002-2 Instructions pour la publication d'OSS. Les documents Em002-3 Guide de licences OSS et Em002-4 Guide des communautés OSS doivent également être consultés si nécessaire. L'objectif est de publier le projet de manière contrôlée à l'aide des trois listes de contrôle *Em002-2.1*, *Em002-2.2*, *Em002-2.3*.

4.4. Participation à des projets ouverts (contribution)

Les réflexions possibles concernant la contribution de l'administration fédérale à des projets ouverts déjà existants sont exposées au chapitre 4.2 du guide BITKOM *[BITKOM2024]*. [20] Selon l'importance du projet pour l'administration fédérale, il convient d'examiner quelle part de responsabilité l'office concerné veut et peut assumer.

La participation à des projets ouverts et le développement direct dans des projets ouverts peuvent également s'appuyer sur les documents relatifs à la publication. L'accent est mis sur Em002-4 Guide des communautés OSS et la liste de contrôle correspondante *Em002-4.1*.

20. Voir https://www.bitkom.org/Bitkom/Publikationen/Bitkom-Leitfaden-zu-Open-Source-Software-20.html, chapitre 4.2.

5. Caractéristiques et choix des licences open source

Les caractéristiques et le choix des licences open source sont décrits dans le document Em002-3 Guide de licences OSS.

6. Acquisition d'OSS

Les questions stratégiques relatives aux acquisitions et à l'OSS sont traitées dans Em002-7 Aspects stratégiques des achats et des logiciels open source.

Le Centre de compétence des marchés publics de la Confédération (CCMP) a publié un nouvel aide-mémoire intitulé *Acquisition de logiciels et art. 9 LMETA [KBB-MB]* ainsi que deux documents « Critères types — acquisition LMETA et open source » et « Textes contractuels types — développement logiciel ».

Une aide supplémentaire est disponible sur l'intranet de l'OFCL (accessible uniquement sur le réseau fédéral) avec les *directives Open Source dans les acquisitions [BBL-WL]* et la *liste de contrôle pour l'exception forfaitaire selon l'art. 9 LMETA [BBL-CL]*.

L'aptitude de l'open source à un usage en entreprise est traitée dans le document *Selection Criteria for Enterprise-Ready Open Source Software [Gu2024]* [21] ou dans le guide Bitkom. [22]

7. Repérabilité des solutions OSS

Différentes plateformes permettent aux organisations intéressées de trouver et d'analyser des logiciels open source avant leur utilisation ou leur acquisition. Les données ainsi obtenues constituent une base pour décider de l'utilisation ou de l'acquisition d'un logiciel open source.

Une compilation des exigences minimales et, le cas échéant, une analyse de marché devraient servir dans tous les cas.

7.1. Catalogue OSS

Les autorités fédérales sont tenues de publier le code source des logiciels qu'elles développent ou font développer pour accomplir leurs tâches. Le catalogue OSS [23] recense tous les logiciels développés par des autorités fédérales depuis le 1er janvier 2024 et publiés en open source. Il offre une vue d'ensemble des logiciels disponibles et facilite la recherche et la réutilisation de solutions open source.

Si un nouveau dépôt ou une nouvelle organisation est créé, cela doit être annoncé par courriel à opensource@bk.admin.ch en vue de l'inscription au catalogue OSS.

Catalogue OSS : https://www.opensource.admin.ch

7.2. OSS Directory

L'OSS Directory exploité par CH Open [24] facilite la recherche de fournisseurs qualifiés offrant un support professionnel pour les logiciels open source.

21. Voir *[Gu2024]*
22. Voir Recommandations pratiques pour les logiciels open source | Guide 2025 | Bitkom e. V., chapitre 3.3
23. https://www.opensource.admin.ch/
24. https://www.ossdirectory.com/fr/home

Les fournisseurs open source montrent, au moyen de témoignages de réussite, quels utilisateurs (clients) ils ont aidés à mettre en œuvre des produits open source déterminés. Le Directory propose en outre des actualités open source issues de plateformes existantes, des actualités phares sélectionnées, des événements, des communautés, des offres d'emploi et des articles de partage de connaissances. L'OSS Directory est disponible en allemand, anglais, français et italien. Toutes les entrées peuvent être ajoutées et mises à jour directement par les utilisateurs enregistrés.

OSS Directory : https://www.ossdirectory.com

7.3. alternativeTo

Un outil pratique pour trouver des alternatives à un logiciel est « alternativeTo - Crowdsourced software recommendations ». [25] Comme son nom l'indique, les alternatives et leurs évaluations ont été créées par crowdsourcing, c'est-à-dire par les contributions de nombreux utilisateurs individuels. alternativeTo distingue quatre types de solutions logicielles, dont seul le premier est considéré comme un logiciel open source au sens de l'art. 9 LMETA :

Free open source
: Logiciel publié sous une licence open source.

Free
: Logiciel gratuit (freeware) dont le code source n'est toutefois pas librement disponible et ne peut pas être modifié.

Freemium
: Ce type de logiciel propose une version gratuite et une version premium, toutes les fonctionnalités importantes étant déjà disponibles dans la version gratuite. Cette catégorie ne comprend pas les logiciels pouvant être testés gratuitement pendant une période déterminée (p. ex. une version d'essai de 30 jours), qui relèvent de la catégorie « Commercial ».

Commercial
: Il s'agit de logiciels propriétaires pour lesquels des redevances de licence sont perçues en cas d'utilisation.

7.4. Dépôts open source

La plateforme de développement open source la plus populaire au monde à l'heure actuelle, avec plus de 30 millions d'utilisateurs et plus de 100 millions de dépôts de code source, est GitHub. [26] Aujourd'hui, pratiquement toutes les entreprises informatiques, en particulier les fabricants de logiciels propriétaires, ainsi que de nombreux autres types d'entreprises et d'organisations, sont présents sur GitHub. De plus en plus d'autorités publiques — une douzaine environ en Suisse [27] — publient en outre leurs propres logiciels open source sur « GitHub and Government » à l'adresse https://government.github.com. Cette démarche peut également être réalisée avec d'autres dépôts.

Exemples d'autres dépôts :

- GitLab
- Bitbucket
- SourceForge
- LaunchPad

25. https://alternativeto.net/
26. https://octoverse.github.com
27. https://government.github.com/community/#switzerland

GitHub Insights permet de consulter de nombreuses statistiques pertinentes d'un projet open source sur GitHub :

Pulse
Vue d'ensemble des activités récentes d'un projet open source sur GitHub : résumé des principales informations sur l'intensité du développement, l'hétérogénéité de la communauté, les signalements ouverts, les améliorations (pull requests), etc.

Contributors
Affichage des développeurs qui ont été actifs et à quel moment. Il s'agit d'un indicateur important pour savoir si tout dépend d'une seule personne ou si une communauté plus large se trouve derrière le projet.

Commits
Affichage des contributions apportées à ce projet open source et de la période concernée.

Code frequency
Visualisation de la quantité de code source ajoutée ou supprimée et du moment où cela s'est produit.

Dependency graph
Dépendances du projet open source vis-à-vis d'autres logiciels open source (p. ex. pertinent pour identifier les failles de sécurité et les mises à jour).

Network
Affichage indiquant quand quel développeur a contribué à quelle branche de développement.

Forks
Liste de toutes les copies du projet open source sur GitHub. Indicateur de la popularité et de la diffusion du logiciel open source.

7.5. Open Hub

S'il s'agit de collecter des informations sur une solution open source qui n'est pas nécessairement développée sur GitHub, Open Hub de Black Duck constitue une bonne option. [28] Pour environ un demi-million de projets open source, une multitude d'informations importantes sont résumées de manière claire :

Project Summary
Une brève description du projet open source.

In a Nutshell
Les faits les plus importants sur un projet open source, tels que le nombre de commits, de contributeurs et de lignes de code, ainsi que le langage de programmation le plus utilisé, la date du premier commit et de la dernière modification. Une appréciation de la base de code et de la taille de l'équipe de développement est en outre indiquée (p. ex. « Mozilla Firefox has a well-established, mature codebase maintained by a very large development team with stable Y-O-Y commits »).

Quick Reference
Contient des informations sur l'organisation, des liens vers le projet et le code, ainsi que des renvois vers des projets similaires.

Licences
Indique sous quelle(s) licence(s) le projet open source est placé et quelles conséquences y sont associées.

Project Security
Fournit des indications sur la sécurité et les vulnérabilités du projet open source.

---

28. https://www.openhub.net/

Code
Un graphique montrant le nombre de lignes de code au fil du temps, ventilé par langage de programmation.

Activity
Un graphique montrant le nombre de commits par mois. Un résumé des 30 derniers jours et des 12 derniers mois est également fourni.

Community
Un graphique montrant le nombre de contributeurs actifs par mois. Une évaluation du projet sur une échelle de cinq étoiles est également affichée.

Open Hub permet également de comparer différents projets open source. [29] Cela donne rapidement une vue d'ensemble du projet qui a la communauté la plus active, la durée de développement la plus longue ou la licence la plus appropriée.

7.6. Dépôts de code particuliers

Les grandes organisations disposent de leurs propres dépôts de code. Il convient de mentionner en particulier le dépôt allemand destiné aux autorités, opencode.de. [30] Les dépôts d'autres administrations publiques présentent ici un intérêt particulier.

8. Modèles de support pour l'utilisation de l'OSS

Les logiciels open source déjà disponibles sur le marché peuvent en principe être utilisés de trois manières :

1. Utilisation sans support professionnel
2. Utilisation avec support interne
3. Utilisation avec support d'un fournisseur externe.

Ces trois types d'utilisation et leurs avantages et inconvénients sont brièvement expliqués ci-après. Lequel de ces scénarios est le plus judicieux dans un cas donné doit être décidé au cas par cas. Le modèle de support approprié dépend de la pertinence stratégique du logiciel open source, de l'intégration technique et des ressources en personnel disponibles.

Pour les logiciels critiques, le support doit dans tous les cas être assuré de manière professionnelle, en interne ou en externe. Le cycle de vie prévu devrait également jouer un rôle dans la planification du support. Il se peut aussi que le support soit obtenu auprès de plusieurs fournisseurs.

8.1. Utilisation sans support professionnel

Les logiciels open source sont disponibles sur Internet et peuvent être téléchargés, installés et utilisés gratuitement.

Avantages

- Faibles coûts
- Mise en œuvre rapide

Inconvénients

- Aucun support garanti

29. https://www.openhub.net/p/_compare
30. https://opencode.de/en/home

- Aucune prétention en responsabilité

Risque et couverture
Risque élevé : il n'existe ni contrats de support ni garanties, et l'organisation dispose de peu ou pas de compétences en développement.

Domaine d'application typique
Logiciels open source standard pouvant être mis à jour indépendamment d'autres systèmes.

8.2. Utilisation avec support interne

Une entreprise ou une organisation du secteur public développe des compétences et des ressources sur des solutions open source déterminées en vue d'une utilisation à long terme. Cette approche est particulièrement répandue dans les domaines critiques pour l'activité.

Avantages

- Grande flexibilité grâce au savoir-faire interne
- Pas de dépendance envers un fournisseur

Inconvénients

- Investissement et temps importants pour développer les compétences
- Coûts fixes internes plus élevés pour le personnel

Risque et couverture
Risque moyen : le support dépend du savoir-faire et de la disponibilité de l'informatique interne.

Domaine d'application typique
Développements propres, logiciels open source utilisés de manière stratégique et pour lesquels un savoir-faire approfondi est disponible.

8.3. Utilisation avec support d'un fournisseur externe

Un fournisseur open source externe est associé afin d'accompagner de manière professionnelle le déploiement et la maintenance du logiciel open source. Cette approche est surtout utilisée dans les domaines critiques pour l'activité, où un savoir-faire approfondi du logiciel doit être immédiatement disponible. Cela peut aussi englober une partie ou l'intégralité de la publication.

Avantages

- Accès direct au savoir-faire des développeurs open source
- Corrections et extensions sur une base contractuelle
- Choix entre différents fournisseurs open source
- Engagement, couverture contre les risques de conformité

Inconvénients

- Coûts externes liés aux fournisseurs open source
- Dépendance en matière de savoir-faire envers le fournisseur open source

Risque et couverture
Risque faible : la garantie est fournie selon le cahier des charges ou le service level agreement

Domaine d'application typique
: Logiciels open source critiques pour l'activité, pour lesquels peu ou pas de compétences internes de développement sont disponibles

9. Point de contact

Il n'existe au sein de l'administration fédérale aucun point de contact unique faisant office d'Open Source Programme Office. En principe, les offices fédéraux sont eux-mêmes responsables de la mise en œuvre.

Les demandes générales concernant les outils OSS du jeu de documents Em002 peuvent être adressées au secteur TNI de la Chancellerie fédérale : opensource@bk.admin.ch.

10. Annexes

10.1. Modifications par rapport à la version précédente

- Chapitre 4.1 : degrés de maturité ajoutés
- Chapitre 6 : acquisition d'OSS complétée et adaptée avec les nouveaux documents de l'OFCL
- Chapitre 7.1 : nouveau catalogue OSS de la Confédération (opensource.admin.ch)
- Autres modifications et améliorations rédactionnelles mineures

10.2. Références

Les références relatives au jeu de documents Em002 figurent dans les Em002 Lignes directrices stratégiques.

10.3. Abréviations

Une liste d'abréviations figure dans le document principal Em002. Un glossaire figure dans le document « Em002-6 Foire aux questions sur la publication d'OSS (FAQ-OSS) ».

10.4. Modèles d'affaires avec l'OSS

Le logiciel open source n'est pas en soi un modèle d'affaires car, contrairement au logiciel propriétaire, il n'est pas possible de vendre un logiciel placé sous licence open source. Il existe néanmoins différentes possibilités pour les entreprises d'exploiter des modèles d'affaires fondés sur les logiciels open source. Ainsi, si un support externe professionnel doit être obtenu pour une solution open source, l'acquisition des prestations commerciales correspondantes est nécessaire. Les quatre modèles d'affaires les plus courants pour les logiciels open source sont expliqués ci-après. D'autres aspects de ces modèles et d'autres encore sont exposés dans le guide BITKOM *[BITKOM2023]*.

10.4.1. Services et produits fondés sur l'OSS

Les entreprises peuvent proposer des services commerciaux tels que l'hébergement web ou le cloud computing sur la base de logiciels open source, ce qui serait nettement plus coûteux avec des logiciels propriétaires. C'est pourquoi la plupart des start-up, des portails en ligne et des fournisseurs de commerce électronique construisent aujourd'hui leurs plateformes sur des logiciels open source. D'autres entreprises technologiques, telles que les entreprises de télécommunications, les fournisseurs de streaming ou même les fabricants de logiciels propriétaires, intègrent des logiciels open source dans leurs produits logiciels et matériels ainsi que dans leurs services en ligne. Cela permet aux entreprises de continuer à proposer des solutions innovantes, difficilement réalisables avec des logiciels propriétaires.

10.4.2. Services pour l’OSS

Les fournisseurs open source fournissent des prestations pour des logiciels open source sélectionnés. Ils disposent de développeurs open source expérimentés et peuvent donc proposer du support, de la maintenance, de l’exploitation, du développement, du conseil, de la formation et d’autres prestations pour les logiciels open source. Celles-ci peuvent être obtenues auprès du fournisseur open source dans le cadre d’un mandat ou d’un contrat d’entreprise. De telles prestations pour logiciels open source peuvent faire l’objet d’appels d’offres publics, car il n’y a pas de vendor lock-in. Ce qui importe pour de tels appels d’offres, c’est la prise en compte de critères appropriés afin que les prestataires réellement compétents soient sélectionnés (voir chapitres 0 et 8, *[BITKOM2023]* chapitre 3.3 et *[Gu2024]*).

10.4.3. Abonnements

Lorsque des prestations pour logiciels open source sont fournies sous une forme standardisée et récurrente, à la manière d’un service level agreement (SLA), on parle d’abonnements. Dans le cadre de tels abonnements, les entreprises garantissent par exemple des mises à jour de sécurité continues, un support par courriel ou par téléphone, la compatibilité avec d’autres produits logiciels et matériels au moyen de certifications, des prestations de maintenance à long terme et une protection contre les prétentions juridiques (droit d’auteur, brevets). En contrepartie, les clients paient des abonnements par poste de travail ou par CPU, à l’instar des redevances de licence ou d’utilisation pour les logiciels propriétaires. Contrairement aux logiciels propriétaires, les abonnements pour les logiciels open source ne constituent toutefois pas une condition d’utilisation du logiciel, mais simplement un moyen de rémunérer la valeur ajoutée des prestations fournies par le fournisseur open source.

10.4.4. Double licence

Si une entreprise détient la propriété intellectuelle d’une solution logicielle et que tous les composants open source intégrés sont placés sous une licence permissive, une double licence ou une licence multiple peut être appliquée. Cela permet à l’entreprise qui développe de publier le logiciel sous une licence open source copyleft et de le vendre en même temps sous une licence propriétaire. Cette version commerciale est souvent appelée version Enterprise et comprend généralement certaines fonctionnalités supplémentaires, telles que des intégrations d’interfaces exclusives ou l’autorisation pour les acheteurs d’intégrer le logiciel dans leurs propres produits propriétaires. Selon l’ampleur des restrictions de la version open source, la prudence est de mise lors de l’utilisation de logiciels sous double licence, car l’acquisition de la version Enterprise peut s’avérer inévitable, le vendor lock-in devenant alors aussi élevé qu’avec un logiciel propriétaire classique.