Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Cancelleria federale CaF

**Trasformazione digitale e governance delle TIC TDT**

14.09.2026

# Em002-6 FAQ su OSS e art. 9 LMeCA

Versione 2.0-33a0da2

# Indice

> **NOTA** Il presente documento è una bozza in continua evoluzione e fa parte degli ausili destinati a sostenere l'Amministrazione federale nella pubblicazione di codice open source. Per ulteriori informazioni si veda il README.

# 1. Situazione iniziale e scopo

## 1.1. Situazione iniziale e scopo delle FAQ

In relazione agli ausili per la pubblicazione di software open source emergono regolarmente alcune domande. La raccolta seguente risponde alle domande più frequenti e intende migliorare la comprensione di singoli temi e aspetti.

## 1.2. Struttura e articolazione

Le FAQ sono articolate per tema:

2. Concetti e definizioni
3. Ruoli e responsabilità
4. Processo
5. Ausili
6. Questioni giuridiche
7. Aspetti generali

# 2. Concetti e definizioni

## 2.1. Software open source

*Che cosa si intende per software open source?*

Questo concetto è definito in Em002-1 Guida pratica per il software open source nell'Amministrazione federale.

## 2.2. Software e licenze

*Che cosa si intende per software?*

Questo concetto è definito in Em002-1 Guida pratica per il software open source nell'Amministrazione federale. Per le licenze si veda Em002-3 Guida alle licenze OSS.

*Che cosa si intende per licenze?*

Si tratta qui di licenze software. Ulteriori informazioni sono riportate in Em002-3 Guida alle licenze OSS.

*I modelli di IA sviluppati internamente vanno considerati software e quindi sottoposti ai requisiti giuridici dell'art. 9 LMeCA? E che dire dei dati di addestramento?*

Open source indica software il cui codice sorgente è liberamente accessibile e modificabile a condizioni minime. Vi rientrano anche i modelli di IA addestrati internamente (compresi algoritmo, procedura di addestramento e codice).

I modelli di IA sono addestrati sulla base di dati, eventualmente Open (Government) Data. Per Open Data si intendono insiemi di dati che possono essere liberamente utilizzati, condivisi ed

elaborati. I modelli di IA i cui dati di addestramento si basano su Open (Government) Data sono considerati software ai sensi della LMeCA e devono pertanto essere pubblicati anch'essi.

Se tuttavia i dati di addestramento contengono dati personali (degni di particolare protezione) o informazioni classificate, si può rinunciare alla pubblicazione per motivi di sicurezza. Nel pubblicare un modello di IA come open source, i dati di addestramento non devono necessariamente essere pubblicati; ciò corrisponde tuttavia allo spirito dell'open source ed è vivamente raccomandato dalla comunità open source. Per ulteriori informazioni sull'IA rinviamo alla Rete di competenze per l'intelligenza artificiale (CNAI).[1]

*Gli strumenti con funzionalità di IA generativa (p. es. GitHub Copilot) devono essere verificati prima dell'impiego affinché non siano violati diritti di proprietà intellettuale di terzi o della Confederazione?*

I contenuti generati da un modello di IA non possono essere protetti dal diritto d'autore, poiché manca un'attività creativa. I contenuti generati dall'IA che attingono a opere protette di terzi possono tuttavia violare i diritti di tali terzi. Inoltre, poiché gli output dell'IA sono notoriamente inaffidabili, dovrebbero essere utilizzati solo dopo un attento esame caso per caso.

Gli operatori di sistemi di IA possono inoltre riservarsi nelle loro condizioni d'uso il diritto di utilizzare gli input per addestrare il sistema. Ciò può a sua volta comportare che gli input o parti di essi vengano mostrati a terzi. Laddove ciò non possa essere escluso, occorre garantire che gli input non violino i diritti della Confederazione o di terzi.

Si raccomanda di esaminare e autorizzare singolarmente l'impiego di strumenti di IA. Gli strumenti non autorizzati non possono essere utilizzati, ma i collaboratori possono chiedere l'autorizzazione di uno strumento. Spesso è opportuno disciplinare tali questioni contrattualmente con il fornitore; alcuni fornitori offrono anche abbonamenti speciali con cui rispettano i diritti di terzi, p. es. DeepL.

### 2.3. Base legale nella LMeCA

*Qual è la base legale per la pubblicazione di software open source?*

Secondo l'art. 9 LMeCA[2] le autorità federali dell'Amministrazione federale centrale devono divulgare il codice sorgente del software che sviluppano o fanno sviluppare. Ogni persona può utilizzare, sviluppare ulteriormente o modificare il software senza dover pagare emolumenti.

Si può rinunciare alla pubblicazione del codice sorgente soltanto per diritti di terzi o per motivi di sicurezza.

Ulteriori informazioni sono riportate in Em002-2 Istruzioni per la pubblicazione di software open source.

*Che cosa si intende per motivi di sicurezza?*

I motivi di sicurezza ammessi dall'art. 9 LMeCA e il modo di trattarli sono definiti nel capitolo 3.2 di Em002-2 Istruzioni per la pubblicazione di software open source.

*Che cosa sono i diritti di terzi?*

I diritti di terzi ai sensi dell'art. 9 LMeCA e il modo di trattarli sono definiti nel capitolo 3.3 di Em002-2 Istruzioni per la pubblicazione di software open source.

*Le eccezioni possono essere limitate nel tempo?*

I motivi che giustificano un'eccezione ai sensi dell'art. 9 LMeCA (diritti di terzi e motivi di sicurezza) possono venir meno con il tempo.

1. Cfr.: https://cnai.swiss/ nonché art. 10 LMeCA
2. RS 172.019

In tal caso il codice sorgente del software deve essere pubblicato.

*Quali conseguenze giuridiche occorre considerare in caso di violazione dell'art. 9 LMeCA?*

Vi sono due tipi di violazioni:

1. sono violati diritti di terzi;
2. un'autorità federale rifiuta di pubblicare software open source (p. es. invocando motivi di sicurezza non fondati).

Risposta al punto 1:

Si tratterà regolarmente di una violazione contrattuale e potranno essere fatte valere pretese di risarcimento danni secondo il Codice delle obbligazioni.

Risposta al punto 2:

Ciò non ha conseguenze dirette. Se un terzo vuole poter utilizzare il codice sorgente di un'autorità federale, deve presentare all'autorità competente una domanda fondata sulla legge sulla trasparenza (LTras; RS 152.3).

In presenza di un forte interesse può essere opportuno valutare una pubblicazione nell'interesse di entrambe le parti. Non sussiste alcun obbligo di pubblicazione retroattiva, che resta però possibile su base volontaria.

*In caso di danno causato da software open source possono essere fatte valere pretese di responsabilità?*

In linea di principio la responsabilità dovrebbe essere esclusa nei contratti. Non è tuttavia possibile escludere la responsabilità per negligenza grave o colpa. In tali casi un'autorità federale potrebbe essere chiamata a rispondere secondo la legge sulla responsabilità (LResp; RS 170.32).

*A partire da quale data il software deve essere pubblicato?*

Secondo l'art. 9 cpv. 1 LMeCA deve essere pubblicato il software sviluppato dopo il 1° gennaio 2024.

Per il software sviluppato prima del 1° gennaio 2024 la pubblicazione dovrebbe essere valutata quando sono previste modifiche rilevanti.

In presenza di un interesse considerevole da parte di terzi, anche il software esistente può essere pubblicato su base volontaria (eventualmente con partecipazione ai costi). In tal caso occorre valutare anche una community (cfr. Em002-4).

Per gli script di piccole dimensioni la pubblicazione non è di regola l'opzione migliore. Possono esistere repository di strumenti che passano poi insieme attraverso il processo di pubblicazione.

Per le unità amministrative dell'Amministrazione federale decentralizzata sottoposte alla LMeCA l'obbligo di pubblicazione vige dal 1° maggio 2025. Sono previste eccezioni all'obbligo di pubblicazione per il software sviluppato senza fondi federali e per il software di ricerca (art. 3 cpv. 1 OMeCA).

*È possibile esigere la pubblicazione retroattiva di software sviluppato prima del 1° gennaio 2024?*

La legge non prevede una pubblicazione retroattiva.

Inoltre, i contratti conclusi prima del 1° gennaio 2024 possono contenere diritti di terzi che ostano a una pubblicazione.

Se è prevista una nuova versione principale (major secondo il Semantic Versioning[3]), occorre valutare la pubblicazione dell'intero software. In caso contrario dovrebbe essere pubblicato almeno il codice sorgente delle nuove funzionalità.

*Su quale base un terzo dovrebbe chiedere la pubblicazione del codice sorgente?*

Un terzo dovrebbe presentare all'autorità competente una domanda fondata sulla LTras.

## 3. Ruoli e responsabilità

### 3.1. Ruolo della Cancelleria federale

*Chi mette a disposizione gli ausili di base per l'attuazione?*

Il settore TDT della Cancelleria federale mette a disposizione delle autorità federali ausili sotto forma di raccolta di documenti relativa a Em002. Ne fanno parte le direttive strategiche e altre guide pratiche, comprese le FAQ, nonché liste di controllo volte a garantire un'attuazione conforme al diritto.

Gli ausili OSS sono aggiornati dal settore TDT.

*La Cancelleria federale è competente per la concessione di eccezioni (p. es. per motivi di sicurezza)?*

Non esiste un organo centrale che decide sulle eccezioni all'obbligo di pubblicazione di cui all'art. 9 LMeCA. Ogni autorità federale è responsabile della propria attuazione conforme al diritto.

I dipartimenti possono emanare disposizioni.

La lista di controllo per l'eccezione generale secondo l'art. 9 LMeCA [BBL-CL] dell'UFCL dovrebbe essere compilata.

### 3.2. Ruolo delle unità amministrative

*Chi è responsabile dell'attuazione operativa delle pubblicazioni OSS?*

Ogni autorità federale (p. es. ufficio, unità amministrativa) che sviluppa o fa sviluppare software è responsabile in proprio dell'intero processo di pubblicazione.

I dipartimenti possono stabilire disposizioni vincolanti e meccanismi di controllo nel loro ambito di competenza, p. es. per evitare danni reputazionali.

Si raccomanda di designare per ogni ufficio federale una persona responsabile dell'OSS, p. es. il responsabile informatico (IM), l'enterprise architect, l'ISBO o il DSBO, al fine di garantire processi uniformi ed efficienti. In assenza di una designazione, la responsabilità di un'attuazione conforme spetta di regola alla direzione dell'ufficio.

Tale persona potrebbe poi essere consultata anche per la concessione di eccezioni all'obbligo di pubblicazione.

### 3.3. Collaborazione con fornitori terzi

*Come vanno pubblicati gli sviluppi interni di un'autorità federale?*

Il diritto d'autore sorge in capo alle persone fisiche che hanno sviluppato il codice sorgente, ma di regola viene trasferito al datore di lavoro tramite il contratto di lavoro. Nei contratti con fornitori di servizi esterni è importante delimitare chiaramente i nuovi sviluppi e convenire la cessione o il

3. Cfr.: https://it.wikipedia.org/wiki/Controllo_versione

trasferimento contrattuale del diritto d'autore. Infine, per gli sviluppi interni comuni le parti coinvolte devono accordarsi sulla licenza open source del codice sorgente con cui il software è pubblicato come progetto open source autonomo. La pubblicazione sotto una licenza open source non significa rinunciare al diritto d'autore: questo resta in capo ai rispettivi titolari. Solo sulla base del loro diritto d'autore essi possono difendersi giuridicamente contro violazioni di licenza o concedere, parallelamente alle licenze OSS, licenze più estese.

### 3.4. Comunicazione di dati personali

*A quali condizioni possono essere pubblicati dati personali, ad esempio degli sviluppatori, nell'ambito di software open source?*

Non esiste una regola generale al riguardo. La decisione spetta al progetto.

In ogni caso vanno rispettate le disposizioni applicabili in materia di protezione dei dati. Secondo l'articolo 36 capoverso 1 LPD gli organi federali possono comunicare dati personali solo se dispongono di una base legale.

L'articolo 36 capoverso 2 LPD prevede tra l'altro le seguenti eccezioni:

(b) la persona interessata ha acconsentito alla comunicazione;

(e) la persona interessata ha reso i suoi dati accessibili a chiunque e non si è opposta espressamente al trattamento.

Nella maggior parte dei casi è opportuno ottenere preventivamente il consenso delle persone interessate, che di regola hanno esse stesse interesse a che il loro lavoro sia visibile pubblicamente. Se il consenso non viene dato o se altri motivi si oppongono alla pubblicazione, il codice sorgente dovrebbe essere pubblicato senza dati personali, con il solo riferimento all'autorità federale interessata.

Tale consenso dovrebbe essere ottenuto da fornitori e collaboratori nell'ambito dei processi di acquisto e di assunzione.

### 3.5. Acquisti

*Dove trovo la documentazione relativa agli acquisti di software open source?*

Con la versione 2.0 è stato messo a disposizione l'ausilio Em002-7 Aspetti strategici degli acquisti e software open source.

Il Centro di competenza per gli acquisti pubblici della Confederazione (CCAP) ha pubblicato un nuovo promemoria intitolato Acquisto di software e art. 9 LMeCA [KBB-MB] nonché due documenti «Criteri tipo – acquisto LMeCA e open source.docx» e «Testi contrattuali tipo – sviluppo di software.docx».

Ulteriori risorse sono disponibili sulla piattaforma di apprendimento e di modelli degli acquisti pubblici (www.perimap.admin.ch).

Ulteriore sostegno è disponibile sull'intranet dell'UFCL (accessibile solo sulla rete della Confederazione) con la direttiva Open source negli acquisti [BBL-WL] e la lista di controllo per l'eccezione generale secondo l'art. 9 LMeCA [BBL-CL].

Le relative informazioni si trovano sulle pagine intranet dell'UFCL (disponibili solo all'interno della Confederazione).

Link: cassetta degli attrezzi dell'UFCL per gli acquisti informatici

# 4. Processo

## 4.1. Istruzioni

*Esistono istruzioni centrali su come trattare il software open source nell'Amministrazione federale?*

Sì, il settore TDT della Cancelleria federale mette a disposizione una documentazione completa a tale scopo.

Va osservato che questi ausili costituiscono una raccomandazione TIC e non una prescrizione vincolante.

Link agli ausili OSS

*Come vanno pubblicati gli sviluppi interni di un'autorità federale realizzati nell'ambito di collaborazioni? Come vanno trattati i diritti esistenti?*

Nello sviluppo di software il diritto d'autore sorge in capo alle persone fisiche che hanno sviluppato il codice sorgente.

Se la persona che sviluppa è legata da un rapporto di lavoro, gli sviluppi appartengono al datore di lavoro (art. 332 CO).

Nei contratti con fornitori di servizi esterni è quindi importante delimitare chiaramente i nuovi sviluppi e convenire la cessione o il trasferimento contrattuale del diritto d'autore. Infine, per gli sviluppi interni comuni le parti coinvolte devono accordarsi sulla licenza open source del codice sorgente con cui il software è pubblicato come progetto open source autonomo.

La pubblicazione sotto una licenza open source non significa rinunciare al diritto d'autore: questo resta in capo ai rispettivi titolari.

Solo sulla base del loro diritto d'autore essi possono difendersi giuridicamente contro violazioni di licenza o concedere, parallelamente alle licenze OSS, licenze più estese.

I diritti originari dovrebbero pertanto, se possibile, spettare sempre alla Confederazione per i nuovi sviluppi o gli sviluppi successivi. Per gli altri partecipanti occorre utilizzare Contributor Licence Agreement (CLA) o un Developer Certificate of Origin (DCO). Le relative istruzioni sono riportate in Em002-3 Guida alle licenze OSS, Em002-2 Istruzioni per la pubblicazione di software open source e Em002-4 Guida alle community OSS.

*Le modifiche devono essere pubblicate?*

Secondo l'art. 9 LMeCA: sì, se sono state sviluppate dopo il 1° gennaio 2024.

Ciò dovrebbe però avvenire in modo da generare anche un beneficio. Il documento Em002-2 Istruzioni per la pubblicazione di software open source informa sul processo.

*Quando la pubblicazione è «sufficiente» ai sensi della LMeCA?*

L'art. 9 LMeCA non è specifico al riguardo e lascia un ampio margine di manovra. Il codice sorgente deve essere accessibile al pubblico.

I requisiti della legge sono per esempio già soddisfatti con la semplice pubblicazione del codice sorgente in un file zip su un sito web.

Per ottenere un beneficio, anche mediante la pubblicazione della documentazione ecc., ciò dovrebbe avvenire in un repository di codice secondo le buone pratiche. Informazioni al riguardo sono riportate in Em002-2 Istruzioni per la pubblicazione di software open source.

*In futuro i beneficiari di prestazioni prescriveranno ai fornitori di prestazioni come pubblicare il software e come garantirne la qualità?*

Sia per i fornitori sia per i beneficiari di prestazioni valgono le regole della rispettiva autorità federale.

È tuttavia opportuno integrare i relativi ausili nei punti adeguati dei processi e dei quality gate esistenti.

Sulla base della propria esperienza di sviluppo, il fornitore di prestazioni può formulare proposte per i suoi beneficiari di prestazioni e trattare la questione in modo uniforme (così come il processo di sviluppo è trattato in modo uniforme).

## 4.2. Scelta della licenza

*Sotto quale licenza open source deve essere sviluppato il software?*

Non esistono prescrizioni rigide sulla scelta della licenza, purché siano rispettate le condizioni di licenza dei componenti software utilizzati nel software da sviluppare.

Per gli sviluppi completamente nuovi occorre scegliere un tipo di licenza che consenta una base ampia e sostenibile per gli sviluppi successivi.

A tal fine è importante che la licenza in questione sia ampiamente accettata nella community di sviluppatori pertinente.

**Si raccomanda pertanto l’impiego di AGPL (con copyleft) e MIT.**

Queste due licenze rappresentano, per così dire, i due estremi dello spettro delle licenze.

L’AGPL, con un copyleft molto forte, impone durevolmente il principio «public money – public code». In caso di sviluppi successivi, segnatamente di soluzioni proprietarie, ciò si attenua in direzione della LGPL.

La MIT è invece una licenza molto liberale, con la quale è possibile quasi tutto.

Le due licenze coprono l’intero spettro. Se, nel caso di licenze permissive, si attribuisce particolare importanza al fatto di non menzionare la Confederazione, la BSD-3 è da preferire alla MIT.

Per una scelta più dettagliata delle licenze rinviamo a Em002-3 Guida alle licenze OSS.

## 4.3. Criteri di qualità per il software open source

*Il software open source deve soddisfare determinati criteri di qualità?*

Di regola valgono tutti i criteri di qualità applicabili a qualsiasi altro software. L’open source rende soprattutto tutto più trasparente. Per rispettare la licenza, i criteri di una «buona» pubblicazione e le eventuali regole della community sono necessari alcuni punti supplementari. Questi sono illustrati in Em002-2.2 Lista di controllo analisi e preparazione OSS e Em002-4.1 Lista di controllo community OSS.

*Quali requisiti sono previsti per i quality gate?*

Non sono previste prescrizioni centrali. Sono determinanti le regole dell’autorità federale (unità amministrativa) che sviluppa il software.

A parte i requisiti generali non esistono requisiti particolari per la qualità del software open source. HERMES sarà tuttavia aggiornato in vista dell’art. 9 LMeCA.

La pubblicazione è il modo in cui l'Amministrazione federale si presenta al pubblico. Se ciò non avviene con cura e professionalità, l'immagine pubblica di un'autorità federale può rapidamente risentirne.

*Come sono disciplinate le direttive di sviluppo?*

A parte i requisiti generali contenuti negli ausili non esistono prescrizioni particolari sullo sviluppo di software open source. Indicazioni sono riportate in Em002-2.2 Lista di controllo analisi e preparazione OSS e Em002-4 Guida alle community OSS.

*Il software open source deve essere pubblicato in un luogo determinato?*

Attualmente la Confederazione non gestisce un proprio repository centrale. Le autorità federali sono quindi in linea di principio libere nella scelta del luogo di pubblicazione. Poiché molte autorità federali utilizzano già GitHub, questa piattaforma è attualmente la scelta preferita. Esiste inoltre un account GitHub centrale (swiss) che può essere utilizzato.

GitHub – swiss/index: una panoramica delle attuali organizzazioni di repository.

Ciò vale in particolare quando un'autorità federale non dispone ancora di un account GitHub («organizzazione») o non utilizza un'altra piattaforma come GitLab o BitBucket.

Si prega di rivolgersi all'unità competente del settore TDT (opensource@bk.admin.ch) per verificare se una pubblicazione tramite l'account GitHub centrale (swiss) sia opportuna e possibile.

Le autorità federali dovrebbero in ogni caso conservare una copia locale del codice. È inoltre opportuno che ogni autorità federale disponga di una strategia corrispondente e di corrispondenti regole interne.

Ulteriori raccomandazioni sono riportate in Em002-2 Istruzioni per la pubblicazione di software open source.

### 4.4. Documentazione

*Quali requisiti di documentazione valgono per la pubblicazione di software open source?*

Gli ausili messi a disposizione dal settore TDT della Cancelleria federale informano sui requisiti giuridici; cfr. gli ausili OSS.

Si raccomanda di compilare le liste di controllo OSS e di archiviarle centralmente in ogni autorità federale (unità amministrativa).

### 4.5. Contributi dell'Amministrazione federale a progetti

*Un'unità amministrativa può e deve contribuire con codice a un progetto «upstream» e soddisfa in tal modo i requisiti dell'art. 9 LMeCA?*

- Se i progetti sono open source, l'art. 9 LMeCA è soddisfatto. I progetti devono però anche rimanere open source. In caso contrario il codice deve essere pubblicato nuovamente.
- I contributi upstream hanno il vantaggio che la manutenzione avviene nel progetto upstream e che il beneficio per la community è massimizzato.

  Esistono i seguenti svantaggi: minore controllo e influenza; può essere necessario firmare un CLA o un DCO (cfr. Em002-4 capitolo 3.3). I contributi non compaiono inoltre nel publiccode.yml né negli elenchi della Confederazione. Un vantaggio del tutto diverso è che con tali contributi gli sviluppatori continuano a crescere all'interno dell'ecosistema.

- Finché i rischi e l'onere restano gestibili (contenuto del CLA, raccolta del codice), i contributi upstream sono da accogliere favorevolmente.
- Al contributo si applica allora la licenza del progetto upstream. Ciò corrisponde alla procedura descritta in Em002-3.
- **I fork di progetti upstream sono da evitare:** l'intero onere di manutenzione e aggiornamento ricade allora sull'autorità federale. Un fork curato in modo insufficiente rappresenta inoltre un rischio per la sicurezza. Per la community il beneficio è pressoché nullo.
- I contributi a progetti upstream dovrebbero essere in linea di principio valutati positivamente e discussi attivamente dai responsabili di progetto delle autorità federali.
- Se il progetto upstream appartiene all'Amministrazione federale e si tratta di contributi di terzi, Em002-7 capitolo 9 fornisce ulteriori indicazioni.

### 4.6. Supporto

*L'autorità federale è tenuta a fornire supporto per il software pubblicato?*

No, ma può farlo se lo desidera.

Secondo l'art. 9 cpv. 5 e 6 LMeCA può anche riscuotere emolumenti per il supporto.

*Come deve procedere un'autorità che intende fatturare il supporto?*

Per motivi di risorse ciò non fa parte dell'attuale progetto di messa a disposizione di ausili. Se necessario, questo aspetto potrebbe essere integrato negli ausili in una tappa successiva. Per ora ogni autorità federale lo disciplina autonomamente. (Cfr. anche Em002-4 Guida alle community OSS)

## 5. Ausili

### 5.1. Istruzioni

*Esistono istruzioni centrali sull'impiego di software open source nell'Amministrazione federale?*

Sì, il settore TDT della Cancelleria federale mette a disposizione una documentazione completa nell'ambito di Em002 Direttive strategiche per il software open source nell'Amministrazione federale.

Il software open source è trattato in generale in Em002-1 Guida pratica per il software open source nell'Amministrazione federale. La pubblicazione ai sensi dell'art. 9 LMeCA è trattata in Em002-2 Istruzioni per la pubblicazione di software open source.

### 5.2. Liste di controllo

*Gli ausili comprendono anche liste di controllo per la pubblicazione di software open source?*

Sì, il settore TDT della Cancelleria federale mette a libera disposizione le seguenti liste di controllo:

- Em002-2.1 Lista di controllo valutazione preliminare OSS
- Em002-2.2 Lista di controllo analisi e preparazione OSS
- Em002-2.3 Lista di controllo rilascio e pubblicazione OSS
- Em002-4.1 Lista di controllo community OSS

Si veda inoltre la **lista di controllo per l'eccezione generale secondo l'art. 9 LMeCA** [BBL-CL] dell'UFCL.

### 5.3. Questioni di sicurezza

*Esistono rischi di sicurezza specifici del software open source?*

In questo contesto si citano ad esempio gli attacchi alla catena di fornitura, le vulnerabilità zero-day e gli attacchi di typosquatting.

Tali questioni di sicurezza devono essere considerate per il software open source e i relativi acquisti, ma anche in generale per lo sviluppo di software e per l'acquisto di software commerciale. Al più possono risultare leggermente amplificate se il progetto OSS impiegato è stato poco accurato nell'integrazione di librerie e container.

A medio termine è necessaria una strategia generale per attenuare questi rischi (tramite l'NCSC/UFCS). Se necessario devono essere messe a disposizione fonti sicure per container e librerie.

In generale tutti i responsabili in ambito informatico devono conoscere questi possibili vettori d'attacco. Possibili tentativi sono:

- Attacchi alla catena di fornitura: `XZ_Utils_backdoor` (https://en.wikipedia.org/wiki/XZ_Utils_backdoor), Codecov (https://blog.gitguardian.com/codecov-supply-chain-breach/)
- Vulnerabilità zero-day: Log4Shell (https://www.ibm.com/de-de/think/topics/log4j)
- Typosquatting: https://en.wikipedia.org/wiki/Typosquatting

*La pubblicazione del codice sorgente aumenta il rischio di una copia malevola di un'app? (p. es. un'app di phishing o un'app falsa)*

No. Trattenere il codice sorgente non offre alcuna protezione efficace contro le app false ed è inoltre incompatibile con l'articolo 9 LMeCA, che sancisce il principio dell'open source per l'Amministrazione federale.

Il rischio che un terzo realizzi un'app visivamente identica per raccogliere dati sussiste indipendentemente dalla pubblicazione del codice sorgente. Poiché i sistemi di design della Confederazione sono pubblicamente disponibili, l'aspetto di un'app può essere riprodotto da un aggressore senza accesso al codice sorgente. Le app possono inoltre essere decompilate per estrarre elementi come i loghi dalla versione pubblicata.

Sicurezza e fiducia sono pertanto garantite non dalla segretezza, bensì dalla verifica e dalla sicurezza crittografica:

- **Firme digitali:** le app ufficiali devono essere firmate crittograficamente; ciò garantisce tecnicamente che un'app sia un originale non modificato della Confederazione. Il sistema operativo dello smartphone verifica questa firma.
- **Verifica negli app store:** gli app store di Apple e Google offrono procedure per verificare che gli editori siano organi statali ufficiali. Google, ad esempio, contrassegna esplicitamente le app della Cancelleria federale (CaF) come app governative, dando agli utenti fiducia nella loro autenticità.
- **Build riproducibili:** mediante build indipendenti e riproducibili è possibile dimostrare tecnicamente che il codice sorgente pubblicato corrisponde esattamente all'app disponibile nello store.

Un’eccezione generale per motivi di sicurezza ai sensi dell’articolo 9 LMeCA non è pertanto applicabile alle app.

Raccomandazione:
Anziché trattenere il codice sorgente, l’attenzione dovrebbe concentrarsi sui processi degli app store e sul monitoraggio. I canali ufficiali devono essere mantenuti aggiornati e le app false segnalate tempestivamente affinché siano rimosse dai gestori delle piattaforme (Apple e Google).

*È opportuno che i servizi richiedenti all’interno dell’Amministrazione federale si uniscano per acquistare software open source?*

Sì, senz’altro. Ciò può avvenire direttamente e in modo informale tra unità amministrative.

Il problema è piuttosto che le unità amministrative conoscono poco le esigenze reciproche.

Una soluzione potrebbe consistere nel fatto che le unità amministrative tengano un elenco «most wanted» di strumenti (o librerie) open source da sviluppare o mettere a disposizione e condividano tale elenco.

Un elenco gestito centralmente potrebbe anche aiutare ad analizzare meglio il fabbisogno di componenti open source e a valutare meglio il grado di maturità dell’unità amministrativa. In definitiva sarebbe opportuno un artifact repository.

La tenuta di un simile elenco sarebbe poi compito di un OSPO, qualora ne venisse introdotto uno. Nel migliore dei casi ciò potrebbe avvenire anche tramite Amministrazione digitale svizzera per l’intera struttura federale.

# 6. Questioni giuridiche

## 6.1. Impiego dell’inglese

*La Confederazione può utilizzare licenze nella lingua originale inglese?*

Secondo l’art. 9 cpv. 4 LMeCA occorre utilizzare, ove possibile e opportuno, testi di licenza affermati a livello internazionale.

Insistere su licenze OSS in lingua tedesca vanificherebbe in larga misura lo scopo dell’art. 9 cpv. 4 LMeCA, poiché esistono pochissimi testi di licenza affermati a livello internazionale per i quali una versione in una delle lingue ufficiali della Svizzera (tedesco, francese e italiano secondo l’art. 70 cpv. 1 Cost.) sia riconosciuta come determinante per l’interpretazione.

Questa situazione è analoga a quella dei corsi di studio in lingua inglese presso scuole universitarie come il PF di Zurigo, dove l’inglese è ammesso come lingua d’insegnamento – fatte salve le tre condizioni dell’articolo 36 della Costituzione federale (Commentario di San Gallo alla Costituzione federale, 4a ed. 2023, art. 70 N 29; devono cioè sussistere una base legale, un interesse pubblico e la proporzionalità affinché l’inglese sia ammesso).

Su questa base l’art. 9 cpv. 4 LMeCA va interpretato nel senso che costituisce la base legale per l’impiego di licenze in lingua inglese, poiché le licenze internazionali sono redatte prevalentemente in inglese. La disposizione deve pertanto riferirsi a queste.

L’interesse pubblico di cui all’art. 9 LMeCA risiede tra l’altro nello scambio con le rispettive community di sviluppatori.[4] Nel settore informatico tali community sono tipicamente internazionali e utilizzano l’inglese come lingua di lavoro. Limitare le licenze a quelle in lingua tedesca renderebbe quantomeno difficile, se non impossibile, la cooperazione con le community

4. FF 2022 804, pag. 66

internazionali, poiché un software così licenziato non sarebbe praticamente utilizzato a livello internazionale. Ciò contrasterebbe anche con lo scopo dell'articolo 9 LMeCA.

Inoltre, le licenze OSS affermate a livello internazionale sono per definizione valide in tutto il mondo. Il legislatore intendeva evidentemente rendere possibile l'utilizzo internazionale del software pubblicato. Limitare le pubblicazioni OSS a licenze in lingua tedesca restringerebbe fortemente l'utilizzabilità internazionale del software pubblicato.

A ciò si aggiunge che l'inglese è saldamente affermato come lingua standard nella sfera commerciale del settore informatico, di cui fanno parte anche le pubblicazioni OSS. L'impiego di licenze in lingua inglese rappresenta pertanto una limitazione ragionevole e proporzionata dei diritti dei partner contrattuali. Le licenze in lingua inglese sono inoltre idonee e necessarie per raggiungere gli obiettivi perseguiti.

## 6.2. Esclusione della responsabilità

*La maggior parte delle licenze OSS contiene clausole di esclusione della responsabilità. È sufficiente o sono necessarie misure supplementari? Come ci si comporta in caso di software difettoso? (tenendo conto della legge sulla responsabilità)*

Le licenze applicabili escludono la responsabilità per negligenza lieve. **La responsabilità non può essere esclusa per negligenza grave o dolo.** I rischi pratici di responsabilità sono tuttavia minimi.

Distribuire software modificato sotto una licenza incompatibile con quella del software originario (ad esempio sotto licenza MIT senza copyleft laddove il codice originariamente utilizzato era sotto licenza GPL con copyleft) viola la licenza originaria. In assenza di tutela della buona fede nel diritto della proprietà intellettuale, ciò non consente né agli altri utenti di utilizzare improvvisamente il software originario sotto la nuova licenza, né fa sorgere un obbligo azionabile di offrire la nuova licenza alle condizioni della licenza originaria. L'unica conseguenza è la violazione della licenza originaria, che può comportare pretese di risarcimento danni (analogia di licenza) nei confronti di un'autorità federale.

Questo rischio può tuttavia essere gestito mediante una verifica sistematica secondo le guide, come è oggi prassi standard nell'industria del software. Concretamente la concessione di licenze richiede la creazione di una distinta di tutti i componenti software preesistenti confluiti nel nuovo software, che deve essere rigorosamente rispettata al momento della concessione delle licenze.

Laddove esistano diritti di terzi, l'art. 9 cpv. 1 LMeCA non richiede comunque la pubblicazione del software (o eventualmente solo delle sue parti di nuova scrittura; ciò andrebbe chiarito nelle guide).

Secondo l'articolo 11 della legge sulla responsabilità, la responsabilità della Confederazione è retta dal diritto civile quando essa agisce come soggetto di diritto civile. Questo è il caso secondo l'art. 9 cpv. 2 LMeCA; una responsabilità supplementare secondo la legge sulla responsabilità è di conseguenza esclusa.

## 6.3. Software permissivo

*Se un software si basa su una libreria o su un componente di programma pubblicato sotto una licenza permissiva, il software principale (senza la libreria) può essere pubblicato sotto una licenza non permissiva?*

Sì, ciò non pone problemi. Si vedano le indicazioni contenute nella guida Em002-3.

## 6.4. Disposizioni di licenza

*Un software può essere utilizzato da più uffici? Quali disposizioni di licenza valgono all'interno dell'Amministrazione federale? L'Amministrazione federale è considerata un gruppo di società? (Ciò riguarda il software open source di terzi non modificato.)*

La distribuzione di software sotto licenze permissive è di regola priva di problemi.

Con le licenze copyleft si pone tuttavia la questione se la distribuzione all'interno dell'Amministrazione federale centrale o decentralizzata faccia scattare obblighi di copyleft e rischi così di compromettere la riservatezza degli sviluppi propri della Confederazione. Nella consegna a istituzioni dell'Amministrazione federale decentralizzata dotate di propria personalità giuridica ciò è effettivamente il caso, mentre non vale all'interno dell'Amministrazione federale centrale, poiché tutti i dipartimenti agiscono sotto la stessa personalità giuridica.

Le società di un gruppo sono giuridicamente indipendenti ma restano sotto il controllo economico della società madre. La dottrina giuridica considera la consegna di codice licenziato con copyleft a una società del gruppo come fatto che fa scattare obblighi di copyleft. Poiché le società di un gruppo ricevono spesso software open source nello sviluppo di software, esse necessitano di un proprio diritto d'uso; la consegna a una società del gruppo fa quindi scattare obblighi di copyleft.

Per contro, la consegna all'interno dell'Amministrazione federale centrale non fa scattare obblighi di copyleft. Per l'Amministrazione federale decentralizzata, dove esistono personalità giuridiche distinte, vale quanto sopra: la consegna fa scattare obblighi di copyleft e il software deve essere offerto in forma di codice sorgente alle stesse condizioni di licenza del software originario. Un'istruzione di non consegnare il software a terzi violerebbe la licenza e potrebbe contravvenire anche all'art. 9 LMeCA.

## 6.5. Rapporto tra LTras e LMeCA

*In quale rapporto stanno la legge sulla trasparenza (LTras) e la LMeCA?*

Secondo l'art. 6 LTras i documenti ufficiali devono essere consegnati. È documento ufficiale ai sensi dell'art. 5 cpv. 1 LTras ogni informazione registrata su un supporto qualsiasi, che si trova in possesso di un'autorità e concerne l'adempimento di un compito pubblico. Il software open source rientra in linea di principio nella LTras. La LTras si applica quando una persona chiede a un dipartimento o a un ufficio il codice sorgente per uso proprio. L'ufficio deve divulgare il codice sorgente alla persona interessata, salvo che si applichino le eccezioni di cui all'articolo 7 LTras. Se l'ufficio rifiuta di divulgare il codice sorgente, si applica la procedura LTras. Diverso è il caso in cui qualcuno esiga che l'ufficio pubblichi il codice sorgente sul proprio sito web. In tal caso la procedura non segue la LTras bensì la LMeCA. Poiché la LMeCA non prevede essa stessa una procedura, deve bastare il diritto amministrativo generale. Questo offre la possibilità di una decisione, di un atto materiale o di una decisione di accertamento generale dell'ufficio interessato. Tale decisione può poi essere impugnata dinanzi al Tribunale amministrativo federale.

Se da un lato sono ipotizzabili obblighi di divulgazione per software sviluppato da un'autorità federale ai sensi della LTras (qualora siano soddisfatte le relative condizioni), va osservato che la divulgazione di software secondo la LTras non concede automaticamente una licenza OSS. L'utilizzo di software divulgato secondo la LTras si limita pertanto agli scopi fissati dalla LTras, come la consultazione del codice o la sua esecuzione a fini di osservazione. La divulgazione secondo la LTras non comprende tuttavia una licenza per l'utilizzo regolare del software o come base per lo sviluppo di ulteriore software (dettagli in T. Poledna/S. Schlauri/S. Schweizer, Rechtliche Voraussetzungen der Nutzung von Open-Source-Software in der öffentlichen Verwaltung, insbesondere des Kantons Bern [trad.: Presupposti giuridici dell'utilizzo di software open source nell'amministrazione pubblica, in particolare nel Cantone di Berna], Zurigo 2017, https://carlgrossmann.com/?ddownload=11748, N 393 segg.).

La decisione di licenziare un software come OSS, anche se esso rientra nella LTras, continua a seguire la LMeCA; non sussiste tuttora alcun diritto al rilascio di una licenza.

È importante notare che il software sviluppato prima del 1° gennaio 2024, pur non essendo sottoposto alla LMeCA, deve comunque essere consegnato secondo la LTras.

### 6.6. OSS e protezione dei dati

*I dati personali degli sviluppatori possono essere pubblicati senz'altro? Quali aspetti della protezione dei dati vanno considerati?*

Dal punto di vista della protezione dei dati la pubblicazione di dati personali (p. es. nomi o indirizzi e-mail) è problematica se i collaboratori non vi hanno preventivamente acconsentito. La comunicazione di dati personali può infatti essere evitata in linea di principio mediante la pseudonimizzazione o l'anonimizzazione delle voci del repository, il che sarebbe richiesto dal principio della minimizzazione dei dati e della proporzionalità (art. 6 cpv. 2 LPD).

Tale consenso dovrebbe tuttavia essere di regola facile da ottenere, poiché i collaboratori hanno spesso interesse a mostrare pubblicamente le proprie capacità (così Poledna/Schlauri/Schweizer, N 68, con rinvii). Il consenso può anche essere implicito attraverso l'uso dei nomi nel repository, quantomeno in assenza di un'istruzione contraria del datore di lavoro.

Ciononostante è opportuno disciplinare nei contratti la responsabilità del fornitore per i dati personali dei suoi collaboratori.

## 7. Aspetti generali

*È rilevante che il software open source sia utilizzato a scopo commerciale o non commerciale?*

Le licenze open source di regola non distinguono tra utilizzo commerciale e non commerciale. Il software open source può quindi essere utilizzato per qualsiasi scopo, comprese le applicazioni commerciali. I fornitori commerciali cercano spesso di integrare componenti open source in prodotti proprietari. Ciò è ammesso solo se i componenti open source non sono soggetti a una licenza con effetto copyleft (cfr. anche Em002-3 Guida alle licenze OSS).

*Esistono restrizioni sulle organizzazioni ammesse per le community?*

In definitiva occorre sempre esaminare il singolo caso: chi c'è dietro e quali impegni vengono assunti?

Finché non circola denaro e non vengono assunti impegni formali, la situazione è relativamente semplice e priva di problemi.

Sono possibili affiliazioni ad associazioni, così come una collaborazione informale.

È possibile delegare le affiliazioni a eOperations o a organizzazioni analoghe.

Cfr. anche Em002-4 Guida alle community OSS, capitolo 3.1

*In che rapporto sta la norma ISO 5230 con le guide?*

La norma ISO 5230:2020 è stata esaminata e considerata nell'elaborazione delle guide. L'elenco seguente riporta i requisiti della norma e il modo in cui sono stati affrontati nelle guide:

**3.1 Programme foundation**

- Le «Em002 Direttive strategiche per il software open source nell'Amministrazione federale» e le strategie richiamate costituiscono il quadro di riferimento. L'attuazione compete ai rispettivi dipartimenti e uffici.

**3.2 Relevant tasks defined and supported**

- La guida «Em002-2 Istruzioni per la pubblicazione di software open source» e le liste di controllo definiscono compiti concreti per la pubblicazione. Attualmente non esiste un organo centrale nell'Amministrazione federale.

#### 3.3 Open source content review and approval

- Il processo di pubblicazione è descritto nell'ausilio «Em002-2 Istruzioni per la pubblicazione di software open source». Esso rinvia anche all'esistenza di una Software Bill of Materials (SBOM).
- L'ausilio «Em002-3 Guida alle licenze OSS» informa in modo completo sulla concessione di licenze.

#### 3.4 Compliance artifact creation and delivery

- Il processo di pubblicazione è descritto nell'ausilio «Em002-2 Istruzioni per la pubblicazione di software open source». Esso rinvia anche all'esistenza di una Software Bill of Materials (SBOM).
- L'attuazione non è descritta più in dettaglio e avviene nei dipartimenti o negli uffici.

#### 3.5 Understanding open source community engagements

- L'ausilio «Em002-4 Guida alle community OSS» descrive come costituire e curare una community. Per ogni progetto occorre definire e attuare un concetto.

#### 3.6 Adherence to the specification requirements

- Il capitolo finale enuncia requisiti per le organizzazioni o i progetti che intendono confermare esplicitamente il rispetto dei requisiti OpenChain.

La norma ISO delinea ulteriori ambiti di responsabilità di un Open Source Programme Office (OSPO) che non sono coperti dalle guide.

## 8. Allegato

### 8.1. Abbreviazioni

**ADS** Amministrazione digitale svizzera (www.digital-public-services-switzerland.ch/strategy)

**AO** Application owner

**BP** Beneficiario di prestazioni

**CaF** Cancelleria federale

**CCAP** Centro di competenza per gli acquisti pubblici della Confederazione

**CLA** Contributor Licence Agreement

**DCO**
Developer Certificate of Origin

**FP**
Fornitore di prestazioni

**FSF**
Free Software Foundation

**HERMES**
*Handbuch der Elektronischen Rechenzentren des Bundes*, un metodo per lo sviluppo di sistemi (www.hermes.admin.ch)

**IA**
Intelligenza artificiale

**LMeCA**
Legge federale concernente l'impiego di mezzi elettronici per l'adempimento dei compiti delle autorità

**LTras**
Legge sulla trasparenza

**ODIC**
Organo direzione informatica della Confederazione (fino al 2021)

**OSI**
Open Source Initiative

**OSPO**
Open Source Programme Office

**OSS**
Software open source

**OSSD**
Sviluppo di software open source

**UFCL**
Ufficio federale delle costruzioni e della logistica

**UO**
Unità organizzativa (di regola un ufficio federale)

**SPDX**
Software Package Data Exchange

## 8.2. Glossario

Ulteriori termini sono reperibili in TERMDAT, la banca dati terminologica dell'Amministrazione federale.

**Ambito di applicazione**
Si intendono le categorie di software per l'automazione d'ufficio (BA) secondo [A029] e altre classificazioni.

**Branch**
Un ramo di sviluppo di un software open source.

**Collaborazione**
Il termine «collaborazione» deriva dalla parola latina «collaborare», che significa «lavorare insieme».
Descrive un modo di lavorare in cui più persone o team operano insieme verso un obiettivo comune, apportando le proprie competenze e risorse. Al centro vi sono lo scambio reciproco, la trasparenza e la condivisione delle conoscenze.
Secondo il Gabler Wirtschaftslexikon, la collaborazione indica anche la «cooperazione di un'impresa con i suoi clienti e fornitori, ricorrendo alle moderne tecnologie dell'informazione per integrare i processi aziendali interni e interaziendali».

**Contribution**
Una contribution OSS è un contributo allo sviluppo e al miglioramento di software open source (OSS). Ciò può avvenire sotto forma di codice, documentazione, test, riscontri o altri tipi di contributi. (Google AI)

**Contributor Licence Agreement (CLA)**
Un Contributor Licence Agreement (CLA), noto anche come Contributor Agreement, è un documento che stabilisce le condizioni alle quali la proprietà intellettuale può essere apportata a un progetto o a un'iniziativa. Di regola si tratta di un progetto software sotto licenza open source (Wikipedia).

**Core developer**
Sviluppatore di un progetto open source che dispone di diritti di accesso come committer al repository.

**Analisi di mercato**
Con un'analisi di mercato specifica si rilevano ed elaborano informazioni sul mercato d'acquisto attuale e potenziale. Scopo dell'analisi di mercato è riconoscere e comprendere le strutture del mercato e in particolare la struttura dell'offerta sotto tutti gli aspetti rilevanti: situazione dei prezzi (alti, bassi, oscillazioni), dimensione del mercato, distribuzione geografica dei fornitori potenziali e degli attori principali ecc. (cfr. https://perimap.admin.ch/goto_perimap_file_38221_download.html).
Possibili fonti di software open source sono riportate in Em002-1 Guida pratica OSS al capitolo 7.

**OSS**
Software open source

**Prodotto**
Qui utilizzato come sinonimo di «applicazione».

Il termine deriva da ITIL. In ITIL (Information Technology Infrastructure Library) un prodotto è una configurazione di risorse concepita per generare valore per un cliente o un'organizzazione. A differenza di un servizio, che è tipicamente un processo continuo e interattivo, un prodotto è un'entità statica.

I prodotti possono essere software, hardware, dati o altre risorse messe a disposizione dall'organizzazione. (Google AI)

**Standardizzazione dei prodotti**
Nel presente documento la standardizzazione si riferisce in primo luogo a SD120, il servizio standard TIC per l'automazione d'ufficio, nonché ad altre standardizzazioni del settore TDT.

**Subscription**
Acquisendo software in abbonamento si stipula una subscription. Ciò significa che il software non viene acquistato.
Nella maggior parte dei casi ciò comprende non solo l'autorizzazione a utilizzare il software, ma anche tutte le prestazioni e il supporto pertinenti.

**Progetto upstream**

Upstream è un termine dello sviluppo software distribuito (spesso open source) e indica la direzione di una patch verso la sua origine (upstream), ossia verso gli sviluppatori o i maintainer originari del software o verso il progetto originario. Può trattarsi anche di librerie software.