



Guidelines on Usage of Creative Commons and Open-Source Licensing for Co-Creation Projects for Societal Benefit

Version 1.1. March 2026

Outline and purpose of the document: These *Considerations and Guidelines on Usage of Creative Commons and Open-Source Licensing for Co-Creation Projects* are part of the IMPAC3T-IP Co-creation toolkit. The document is designed to help co-creation teams, especially those dealing with societal challenges and human-centred projects, choose the proper approaches to protect their intellectual assets while achieving societal benefit and active usage of their results. It explains the advantages, disadvantages of each approach, the specific conditions each can be used, and provides a brief on how to understand, manage, and leverage intellectual property rights (IPR) across content and code. As this guide is designed to provide a basic framework to use Creative Commons (CC) and Open Source Software (OS) licences in co-creation settings, when implementing asset protection we recommend seeking more in-depth guides and instructions, both from and outside the IMPAC3T-IP toolkit.

Target readership: This document is intended for project leads and coordinators in co-creation initiatives, individual researchers and software contributors, community and charity partners, municipal and public sector stakeholders and SMEs. These entities may be involved in creating, disseminating, or managing project outputs, including but not limited to training materials, reports, datasets, audiovisual content, software and scripts, documentation, and design files. The information could also be useful for technology transfer (TT) and knowledge transfer (KT) professionals when advising co-creation project participants.

Level: This document assumes a basic understanding of the co-creation process and knowledge of academic research and/or software development, but no prior expertise in intellectual assets (IA) or IPR. Co-creation teams and other co-creation stakeholders with little or no experience in CC and OS licences are advised to consult legal professionals to ensure compliance with applicable laws, funder mandates, and institutional policies.

Focus: The focus of this document is to explain the approach where CC and OS licences are used to protect and disseminate co-creation results, as well as provide basic guidance on copyright ownership, access rights and reuse conditions for content and OS licences for software.

Scope: The scope of this document is to provide teams with a basic understanding of the CC and OS approaches, enabling them to consider these as options for sharing co-created project results.

Disclaimer: This guide reflects the authors' best understanding of Creative Commons, open-source licensing, and co-creation practices at the time of writing. As these fields continue to evolve, it should be used only as general guidance and not as a substitute for professional legal, IP, or data-protection advice. The authors, IMPAC3T-IP, and project partners accept no responsibility for outcomes arising from its use.

Authors / Contact	
Lead	Natalia Zhuravleva, University College Dublin, Ireland
Contributions by	NovaUCD, University College Dublin, Ireland
Contact	info@impac3tip.eu

Index

Acronyms	1
1. Why Open Licensing is a consideration in ‘Co-creation for Societal Challenges’	2
1.1. What are Creative Commons licences?	3
1.1.1. Limitations of Creative Commons licences	4
1.2. What are Open-Source licences?.....	5
1.2.1. Challenges and risks of open-source models	5
2. Choosing and Managing Open Licences.....	6
2.1. Choosing between CC and OS licences	6
3. Implementation checklist.....	7
Annex I. Examples of Open Societal Projects	8
Annex II. Managing Software-Related OS Risks	9

Acronyms

BSD	Berkeley Software Distribution
CC	Creative Commons
EC	European Commission
EUPL	European Union Public Licence
GPL	Generic Public Licence
IA	Intellectual Assets
IPR	Intellectual Property Rights
KT	Knowledge Transfer
NC	Non-Commercial
ND	No-Derivatives
OS	Open Source
SME	Small and Medium Enterprises
SPDX	Software Package Data Exchange
TT	Technology Transfer

1. Why Open Licensing is a consideration in ‘Co-creation for Societal Challenges’

Co-creation techniques, where diverse groups of stakeholders come together to create a solution to a problem, are increasingly used to respond to societal challenges. Such challenges could include tackling traffic congestion to improve air-quality and benefit public health, or exploring responsible use of AI technologies to safeguard vulnerable users (see Annex 1 for examples of real projects). Co-creation teams for such projects may include end users and researchers from different disciplines alongside civil stakeholders such as a municipal authority. Projects may also require involvement from commercial enterprises to build technical solutions, such as software.

Co-creation initiatives in the social sciences and related domains, including some environmental and well-being initiatives, routinely generate a mosaic of outputs, such as data sets, reports, articles, training materials, approaches, software scripts and device designs. When such assets remain under traditional “all-rights-reserved” copyright or proprietary licenses, every re-use request turns into a negotiation. Collaborative momentum slows, and adaptation and further development of the project results becomes challenging. Implementation and social impact of the project results is restricted.

At the same time, uncontrollable dissemination of results without licensing risks inappropriate usage that could lead to loss and harm for the final users or objects, especially in areas without strict government regulations (‘Inappropriate Usage’).

Creative Commons (CC) and Open-Source Software (OS) licenses offer ready-made, internationally recognized legal frameworks, adapted to legal systems worldwide, that let teams share their results to achieve impact. They allow teams to publish early, iterate quickly and invite new contributors and collaborators, while defining clear boundaries for attribution, commercial use or code redistribution.

Although both CC and OS are already well-established, co-creation teams often involve participants with general or superficial knowledge of the approaches. Participants in societal challenges projects often do not consider Intellectual Property (IP), or do not plan for dissemination of results, believing that obtaining an excellent output itself is enough. Those who, for example, deal with OS for software development, may not transfer that experience to social projects.

Another issue may occur when teams without capacity for enterprise software development subcontract the code development part of the project (for example, creating an application). If the resulting code is owned by the developer, this inhibits use in further work involving project results, or on similar projects, resulting in repeated development costs. In fact, improperly formalising relationships with subcontractors or suppliers, perhaps due to a lack of access to legal support at the right time can be terminal to the entire co-creation project. A policy on the use of CC and OS in projects addressing social challenges may therefore avoid the generation of barriers when formalising the contractors' work.

Generally, large European funding programmes protect projects from such problems by utilising generic sets of documentation for the project results. However, for smaller and local/ regional initiatives, concerns still often arise.

For projects that work with digital resources (which are pervasive today), it is important to note that many digital tools currently depend on OS libraries. In 2024 96% of commercial code bases contained OS components, and 77% of the code itself is OS¹. Therefore, any new product or platform that a co-creation team builds can inherit corresponding OS obligations.

Overall, CC and OS approaches can accelerate upfront adoption and further development of the results, whether mandated to the project or not. Downstream users and external stakeholders are aided to easily understand what they can do with an asset. Also, open licenced assets may become a platform for extending usage, fuelling further research and increasing potential social value.

1.1. What are Creative Commons licences?

CC licences are legal tools specifying in an easy-to-understand way how creative works can be used by the public under copyright law. They offer creators a practical approach to protect creative work while still enabling the sharing and reuse for the benefit of society.

CC occupies an intermediate position between copyright, where any usage or reproduction of the work requires permission of the copyright holder, and the public domain, where the work is free to use, share, or adapt for any purposes and without any permission required. A straightforward system of well-recognised symbols used in CC gives creators the possibility to set easily understandable conditions that others can follow.

Each license type codes specific license conditions - such as requiring attribution, excluding commercial uses, or prohibiting derivative works, which can be combined to fit the project's needs.



CC provides a way to balance openness with revenue generation. For more details refer to *IMPAC3T IP Creative Commons Guidelines*².

<u>Licence Characteristics</u>
Creative Commons 0 – CC 0 <ul style="list-style-type: none">• No credit required (No Attribution)• The work can be remixed (No usage mandates)• The work can be sold (no Commercialisation restrictions)
Creative Commons Attribution – CC BY <ul style="list-style-type: none">• Credit to the author must be provided• The work can be remixed (No usage mandates)• The work can be sold (no Commercialisation restrictions)

¹ 9th Edition "Open Source Security and Risk Analysis" Report, Synopsys Ltd, 2024. Available to download at: https://static.carahsoft.com/concrete/files/1617/1597/8665/2024_Open_Source_Security_and_Risk_Analysis_Report_WRAPPED.pdf (Last accessed 09.02.2026)

² See the IMPAC3T IP toolbox at: impac3tip.eu/toolbox/

<p>Creative Commons Attribution - CC BY ND</p> <ul style="list-style-type: none"> • Credit to the author must be provided • The work cannot be remixed; it can only be used in the original way • The work can be sold commercially.
<p>Creative Commons Attribution – CC BY SA</p> <ul style="list-style-type: none"> • People are free to share (that is, copy and redistribute) and adapt (remix, transform, and build upon) the material, even for commercial purposes, as long as any derivative works are distributed under the same CC BY-SA license.
<p>Creative Commons Attribution - CC BY ND</p> <ul style="list-style-type: none"> • You must give credit to the author • The work cannot be remixed; it can only be used in the original way • The work cannot be sold commercially.
<p>Creative Commons Attribution - CC BY ND</p> <ul style="list-style-type: none"> • You must give credit to the author • The work can be remixed. • The derivative works should be shared on same licensing terms as the original. • The work cannot be sold commercially.

1.1.1. Limitations of Creative Commons licences

CC licences are very useful for open dissemination, but they are not a good fit for every result:

- **Not suitable for software:** CC licences are not designed for code or patents and can clash with standard open-source software licences³. For software, using an established domain-specific permissive open-source licence (e.g. MIT, BSD, or GNU – see section 1.2 below) is usually clearer and safer than any CC licence.
- **Irrevocableness:** Creative Commons licences are irrevocable. Once a licence is granted, the permissions it confers are permanent and cannot be withdrawn. This guarantees that anyone who has accessed or used a work under a CC licence retains those rights indefinitely, even if the creator later decides to stop sharing the work or to apply more restrictive terms.
- **No-Derivatives (ND)** licences do not allow sharing modified versions. This stops translation, updates, corrections, or local adaptation without permission. Don't use this attribution if you prefer your results to be adapted for a wider impact or become a base for further development. Note that all permissions require your time and resources for your legal support.
- **Non-Commercial (NC)** attributes may prevent further development and dissemination of results, and some people may decide to avoid further development of such work while they can be unsure what counts as “commercial use”. This may discourage reuse of the project results in education or data-driven innovation.

³ <https://opensource.org/licenses>

- **Problems for open data:** NC and ND restrictions often conflict with open-data policies, text-and-data mining, or commercial reuse in socially useful services, where very open tools like CC0 are widely used.

1.2. What are Open-Source licences?

Open-Source (OS) licences are legal tools that define how software and other code-based materials can be used, studied, modified, and shared. They work within copyright law but switch the default from “all rights reserved” to “reuse allowed under clear conditions,” so that collaboration and reuse are possible without needing to ask for permission each time. In practice, most OS licences fall into two main families: permissive licences, which allow later proprietary reuse with few conditions, and copyleft licences, which require that modified versions remain open under the same (or a compatible) licence.

Permissive open-source licences set only a small number of conditions for using and distributing software. They usually require you to keep the original copyright notice and licence text, but they do not require you to publish your own changes or future versions under the same terms. This means that code under a permissive licence can later be reused in proprietary products, including closed-source software, as long as the basic conditions are respected. Typical examples of permissive free-software licences are the MIT License⁴ and the BSD (Berkeley Software Distribution) 3-Clause License⁵. When distributing software (both modified and unmodified), permissive licences allow the distributor to keep the modified source code private.

Copyleft licences are more “share-alike” in nature. They allow users to run, study, modify, and share the software, but they also require that any distributed modified version is licensed under the same, or a compatible, copyleft licence. In practice, this means you cannot take copyleft-licensed code, improve it, and then release the result only under a proprietary licence. When you distribute software under a copyleft licence, you must also provide access to the corresponding source code under copyleft terms. Well-known examples include the GNU GPL (General Public License) family described by the Free Software Foundation⁶ and similar “strong” or “reciprocal” licences.

To choose an appropriate open-source licence, you can use tools such as ChooseALicense.com⁷, which compares popular licences and helps match them to your goals (for example, allowing closed-source reuse, requiring sharing of improvements, or supporting commercial use).

The European Commission (EC) actively promotes open licensing and has created the European Union Public Licence⁸ (EUPL) for software and related works. The EUPL is a copyleft licence drafted under EU law and available in all official EU languages. It was originally designed for distributing software developed by EU institutions but can be used by anyone who wants a legally solid open-source licence, allowing interoperability with other major copyleft licences.

1.2.1. Challenges and risks of open-source models

Even when open-source licences are chosen carefully, projects still face several practical problems:

⁴ <https://mit-license.org/>

⁵ <https://opensource.org/license/bsd-3-clause>

⁶ <https://www.gnu.org/licenses/copyleft.en.html>

⁷ <https://choosealicense.com/>

⁸ <https://eupl.eu/> or see https://joinup.ec.europa.eu/community/eupl/og_page/eupl

- **Financial sustainability:** Open-source projects often struggle to build a stable and predictable cash flow. Income usually depends on services such as consulting, support contracts, custom development, or training, not on licence fees. Some projects use a freemium model⁹ or dual licensing, being a free open-source version plus a paid “enterprise” version. For more detail, see the IMPAC3T IP Guidelines on business models for Galleries, Libraries, Archives, and Museums (GLAM)¹⁰
- **Skills gaps for working with open-source repositories:** Teams may lack the skills and time to work properly with OS code repositories, for example understanding branch models, how to make pull/merge requests, or how to follow project contribution guidelines¹¹
- **Procurement and funding processes:** Public or private procurement procedures and funding schemes are usually designed around buying or licensing proprietary IP. They may expect a clear “licence of IP” as the main output or asset. This can create friction when a project wants to use open-source licences.
- **Risks of publishing high-risk components.** Some elements are too risky to be released under OS licences. Examples include code or tools that could be misused or repurposed for harmful activities, or components that embed personal or confidential data. In such cases, open publication can create legal, ethical, or cybersecurity risks.
- **See Annex II for more guidelines on managing software-related risks**

2. Choosing and Managing Open Licences

2.1. Choosing between CC and OS licences

Decision factor	Favours CC	Favours OS
Asset type	Text, images, video, datasets, training materials	Software source code, executable scripts, firmware
Desired freedoms	Let others read, share, translate, adapt or remix content (with optional limits on commercial use or derivatives)	Let others read, share, translate, adapt or remix content (with optional limits on commercial use or derivatives)
Community norms	Arts, education, heritage, open data	Software development, DevOps, data-science code
Patent considerations	Not covered	Some licences (e.g. Apache-2.0, GPL-3.0) include patent-grant clauses
Effort and compliance	You want light-touch management: attribution and maybe NC/ND, almost no tracking of components or	You accept more process: tracking dependencies and licences, sometimes publishing source code, managing compatibility

⁹ <https://www.investopedia.com/terms/f/freemium.asp>

¹⁰ See the IMPAC3T IP toolbox at: impac3tip.eu/toolbox/

¹¹ <https://opensource.guide/how-to-contribute/>

	updates	
Typical project goal	Raise awareness, share knowledge and training materials, support advocacy or community education	Build, share, or scale software tools, platforms, or services that others can run, extend, or integrate

Quick licence-selection workflow

1. **Classify the asset** – code, data, documentation, design?
2. **Clarify goals** – widest possible adoption (permissive) vs. guaranteed openness of derivatives (copyleft).
3. **Check incoming dependencies** – their licences may impose compatibility limits.
4. **Use a licence chooser tool:**
 - CC Licence Chooser for content¹²
 - ChooseALicense.com guide for software¹³
 - SPDX licence list¹⁴ for full texts and identifiers
5. **Document the decision** in the project README or data management plan.
6. **Educate contributors** on their responsibilities (e.g. attribution strings, code copyright notices).

3. Implementation checklist

Step	What to do	Description
1	Map your assets	Prepare an early inventory of key outputs (code, data, documents, training materials, branding, etc.).
2	Choose the licence(s)	Use the licence-selection workflow to decide which Creative Commons or open-source licences apply to each asset.
3	Set contributor terms	Put in place a simple contributor/collaboration agreement to clarify joint ownership, assignment, and reuse.
4	Publish with clear metadata	Release code in a public repository such as GitHub ¹⁵ and other materials in open repositories such as Zenodo ¹⁶ or Figshare ¹⁷ , with licence details in the metadata and README files.
5	Define governance processes	Agree who reviews and merges changes, who approves new releases, and who handles legal or takedown notices.
6	Review and update regularly	Schedule periodic checks for dependency updates, security issues, and changes in laws or policies.

¹² <https://creativecommons.org/choose>

¹³ <https://choosealicense.com/>

¹⁴ <https://spdx.org/licenses/>

¹⁵ <https://github.com/>

¹⁶ <https://zenodo.org/>

¹⁷ <https://figshare.com/>

Annex I. Examples of Open Societal Projects

Here are some examples of projects which use open licences and co-creation approaches for societal benefits.

1. The **Little Window** project¹⁸ is an example of open-source technology developed for social good. It was created by the feminist organisation Chayn as a chatbot to support people experiencing domestic and partner violence. The tool was built with input from survivors and volunteers, and the software community helped to improve and maintain it. This mix of lived experience, technical expertise, and open code shows how co-creation can be used to design safer, more responsive services for very vulnerable groups.

2. The **Open Climate Campaign**¹⁹ is a global partnership between Creative Commons, SPARC, and EIFL that aims to make climate and biodiversity research open by default. It works with funders, universities, libraries, and environmental organisations to change policies, licences, and workflows so that articles, data, and educational materials are shared under open licences. This is a good example of co-creation at policy and infrastructure level: many stakeholders negotiate how knowledge is shared, reused, and credited to speeding up climate action.

3. The **BLOOM**²⁰ large language model was built by more than a thousand researchers from over 250 institutions worldwide, as part of the BigScience initiative. Instead of being a closed commercial system, BLOOM is released as an open-access AI model with a specific Responsible AI License that allows broad research and innovation but restricts clearly harmful uses, such as large-scale surveillance or targeted misinformation. This shows how co-creation and community governance can be combined with tailored licensing to balance openness with safeguards in the AI field.

4. **Bioconductor**²¹ is a long-running open-source platform for bioinformatics and genomic data analysis. It brings together an international community of researchers, clinicians, and developers who co-create hundreds of R packages for processing, visualising, and sharing biological data. The project uses open licences, transparent development on public repositories, and shared training materials to support both academic and commercial users. In this way, Bioconductor connects research, healthcare, and industry and helps turn scientific methods into reusable tools for the wider life-science community.

¹⁸ <https://github.com/chaynHQ/little-window>; Further reading see: <https://blog.chayn.co/why-chayn-took-down-its-chatbot-in-2020-and-what-weve-learned-about-culturally-aware-chatbots-a9587cf80df8>

¹⁹ openclimatecampaign.org

²⁰ huggingface.co/bigscience/bloom

²¹ www.bioconductor.org

Annex II. Managing Software-Related OS Risks

Open-source software is not simply “free”; it creates ongoing responsibilities. Teams need to manage security, licence compliance, and provenance of code, data, and design files throughout the project life cycle – not just at the point of release. Recent audits²² show that 86 % of proprietary applications contain at least one high-risk OSS vulnerability.

Risk management for open source has three main dimensions:

- **Security and maintenance:** keeping components up to date, monitoring known vulnerabilities, and planning how to respond to critical issues.
- **Licensing and compatibility:** understanding which licences are in use, how they interact, and whether they are compatible with your distribution model (for example, copyleft vs. permissive).
- **Provenance and openness:** checking that inherited assets (code, datasets, design files) are genuinely open, and that reuse conditions are clear and respected.

See the following steps to reduce risks:

- Keep a **software bill of materials (SBOM)** for all third-party components, including version numbers and licence information. This makes it easier to respond when a vulnerability or licence issue is identified, in compliance with the Cyber Resilience Act²³.
- Run **regular vulnerability scans** and dependency checks using standard tools integrated into your CI/CD or release process. Plan time in each development cycle to update or replace risky components.
- **Check that datasets**, media assets, and design files inherited from other projects are truly reusable and not “faux-pen” (formally “open”, but with hidden or unclear restrictions). Keep links to the original source and a copy of the licence terms.
- Allocate time and budget for **basic governance**: recording key components, reviewing licences before release, and answering questions from partners or users about reuse conditions. For larger or long-running projects, assign a person or small group to oversee open-source and licensing questions.
- **Record major decisions** about component choice, licence selection, and exceptions. This helps new team members understand the constraints and makes it easier to justify choices to funders, partners, or auditors.

²² <https://www.radicalcompliance.com/2025/02/25/study-open-source-software-risks-are-rampant/>

²³ <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32024R2847>.

info@impac3tip.eu

www.impac3tip.eu



**Funded by
the European Union**

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or HADEA. Neither the European Union nor the granting authority can be held responsible for them.