

# Faculty Guide to Open-Source Software Licenses



**Penn Center for Innovation**  
**2026**

---

## Table of Contents

I. Introduction .....	2
II. Open Source Software Licensing Introduction and Guide to Types of Licenses .....	3
A. Assessing Intent for Public Availability .....	4
B. Deciding which OSS license to use .....	6
→ Copyleft/Restrictive Licenses .....	6
→ Permissive Licenses .....	7
→ Existing Community .....	8
→ Other Circumstances Requiring Consideration .....	8
C. Penn recommendations .....	10
D. Applying a License to Your Repository .....	10
E. Best Practices .....	10
F. References .....	10
III. The Chart .....	11
A. Creating New OSS .....	11
B. Consulting and Making Use of the OSS Chart .....	11
Appendix A .....	13
Appendix B .....	17

---

## I. Introduction

Open Source Software (“OSS”) is commonly used at many institutions for IT operations and research, offering benefits such as cost-effectiveness, collaboration, flexibility and community; however, the implementation of OSS and distribution of new code through an OSS license requires consideration of restrictions and risks related to intellectual property (“IP”) and commercialization prospects.

While an OSS license may be appropriate, and beneficial in many cases, it is important that distribution of Penn developed IP under an OSS license be considered only in cases where the proposed license complies with all relevant policies, and with explicit recognition, and mitigation, of any risks to Penn's IP portfolio. Because Penn’s IP is potentially implicated when using, modifying, or distributing software pursuant to OSS licenses, it is crucial that faculty members, researchers and staff understand the implications of an OSS license selection.

This guide is intended to provide faculty members, researchers, and staff an overview of the restrictions and risks associated with OSS licenses and offer criteria to consider when assessing how to release newly created software technologies at Penn.

For questions and further information, contact PCI at:

[pciinfo@pci.upenn.edu](mailto:pciinfo@pci.upenn.edu) or call 215-7-INVENT.

For faculty members with a Technology Licensing Officer (TLO) assigned to assist them, please first [contact your designated TLO](#).

## II. Open Source Software Licensing Introduction and Guide to Types of Licenses

Open Source Software licenses are software licenses that allow software code to be used, modified, and shared while providing protection for the developers and for Penn. There are 2 main OSS license repositories - The Open Source Initiative (OSI) and the Free Software Foundation (FSF). The OSI and the FSF are international, non-profit organizations that approve and maintain licenses, and in some cases enforce them. OSS licenses are reviewed and approved by either the OSI, a standards body that maintains the [Open Source Definition](#) and [Approved Licenses](#), or the FSF which solely maintains the GPL based licenses. The FSF also has useful information about OSS licenses, especially in the context of license compatibility.

- Open Source Initiative (or OSI): <https://opensource.org>
- Free Software Foundation (FSF): <https://www.fsf.org>

The two organizations have slightly different definitions of OSS and the licenses vary in goals and restrictions:

- OSI: <https://opensource.org/docs/osd>
- FSF: <https://www.gnu.org/philosophy/free-sw.html>

There are many different OSS licenses; however, in general, they fall into 2 major categories – permissive and restrictive – that align fairly closely with the 2 different standards bodies.

- 1) **Restrictive (FSF):** The restrictive licenses generally have ongoing obligations that can limit future uses of the software. These restrictive licenses are frequently referred to as “copyleft” or “restrictive” or “viral” and require users who modify the software to release any modifications made to the software under the same license (also referred to as a “hereditary license requirement”). They are referred to as viral because the terms of such licenses (e.g., the GPL licenses) are dominant and are carried over to any code with which they are combined. Thus, when any GPL code is incorporated into non-GPL code, the resulting combined software code **must** be licensed under the GPL license, regardless of the license terms of the non-GPL code. Also included in certain restrictive licenses are patent grants (Detailed in Section II.B.).
- 2) **Permissive (OSI):** Permissive licenses permit freedom to use the software for a variety of purposes, including within commercial products. There are no viral or hereditary license requirements with any of the permissive licenses – you may use a different license to release software received or previously distributed under a permissive license. Examples include the BSD, and MIT licenses.
- 3) **Creative Commons Licenses:** Other intellectual property licenses you may be familiar with, such as Creative Commons (CC), are not appropriate for software; however, they can be used for data, written content or other copyrightable works *other* than software.

A matrix of the most popular OSS licenses is included at the end of this guide for easy reference as well as a FAQ. [PCI Contact](#) When embarking on a new project involving the use, modification or creation of software, it's important to first decide on the level to which you wish the output of the project to be made available to the public *and* under which specific license. These initial considerations will set the stage for the project's future development, distribution, and impact.

---

## A. Assessing Intent for Public Availability:

### 1. Project Goals and Audience:

- **Define the primary objectives** of your project. Is it aimed at solving a specific problem, advancing knowledge, or providing a tool or resource for a broader audience?
- **Identify your intended audience.** Is the project meant for the academic community, commercial exploitation, industry professionals, the public, or a specific group?

### 2. Open Development and Sharing:

- **Consider the benefits of open development.** Openly developed projects can benefit from community contributions, peer review, and widespread use and recognition.
- **Assess the potential for collaboration.** Projects intended for open development often facilitate greater collaboration and knowledge sharing within and beyond Penn.

### 3. Impact on Licensing Choices:

- **Understand how public availability influences licensing.** An intent to make the project publicly available typically leans towards open-source licensing, which allows others to use, modify, and distribute your project more freely, but could limit the commercial applicability of the project.
- **Evaluate alignment with OSS principles.** If public availability is a goal, consider how OSS licenses can support principles like transparency, collaboration, and community engagement.

### 4. Role of PCI:

- **Determine the need for PCI involvement.** If your project involves potentially patentable innovations, complex IP considerations, or an output that could have commercial application or potential, PCI's expertise will be crucial.
- **Disclosure.** Complete a software disclosure form and submit to PCI:  
[https://pci.upenn.edu/wp-content/uploads/2025/04/UPENN\\_Copyright-and-Software-Disclosure-Form\\_Dec-2024-3.docx](https://pci.upenn.edu/wp-content/uploads/2025/04/UPENN_Copyright-and-Software-Disclosure-Form_Dec-2024-3.docx)
- **Engage with PCI.** PCI is available to assist to help balance open access with preserving options for commercial development and IP protection, if applicable.

### 5. Documenting Your Decision:

- 
- **Clearly document the intent for public availability.** This documentation will guide future decisions and help in communicating the project's goals to stakeholders, collaborators, and potential users.
  - **Review and Reassess Over Time.** As the project evolves, regularly revisit the decision on public availability to ensure it remains aligned with the project's evolving scope and objectives.

---

## B. Deciding which OSS license to use:

If you have concluded that you want to distribute the software publicly, the next step is determining which OSS license to select. This choice is important and can impact commercial applications for your project and any future development plans. Different OSS licenses offer varying degrees of protection to the creator of the software, the creator's employer, or the downstream user or modifier.

### → Copyleft/Restrictive Licenses

Choose a copyleft or restrictive license for the protection it offers against proprietary software development and the risk that someone will take your code and limit full access to the source code.

Reasons to choose a copyleft license:

- GPL licensed code has been incorporated into the new code. The newly created code **MUST** be distributed under a GPL license and any new code written will automatically be GPL licensed (this is dictated by the terms of the GPL license and will be enforced by the FSF).
- You see an opportunity to develop an open-source service or consulting business around the software (an example would be Red Hat software which provides consultation and services for Linux software applications).
- You don't want people using the code in proprietary applications and making the source code unavailable.

With a copyleft license any derivative works of your code, including bug fixes, enhancements, etc. *must* also be released under the same license, thus ensuring that your contributions (and anyone else's contributions) to the open-source community remain freely available and accessible in perpetuity. Once software is released under this license, it cannot be "un-released" and made private again. This viral or hereditary provision prevents companies from taking advantage of your work and incorporating it into proprietary software without giving back to the community. By choosing a copyleft license, you can safeguard the integrity and sustainability of the open-source ecosystem; however, due to the patent grant language, the use of the GPL license can limit the commercialization opportunities and could impact Penn's patenting decisions.

A patent grant embedded in an OSS license means that anyone who uses, modifies, or distributes the software is also granted a license to any patents that cover the software. If Penn has granted an exclusive patent license that covers the software, this could cause conflicts with that license.

A patent-granting license is one that, in addition to granting a license to the copyright, also grants licenses to any *related* patents held by the licensor and embodied in the code that is being released. The grant of a license to the related patents may not be what was initially intended when the software was released and may have unintended consequences with respect to commercializing the software.

---

GPL is a hereditary or viral license meaning that any new code that uses or incorporates GPL-licensed code *must* also be licensed under the same GPL terms – even if just a few lines of code or a short routine is all that is incorporated. This can cause compatibility issues if GPL-licensed code is incorporated unknowingly by a developer and can impact future plans for the code. **Caution should be used whenever any outside or existing code is used on a project and anytime code is incorporated, it must be documented to ensure future compliance with any existing licenses.**

For copyleft/reciprocal licenses, **PCI recommends the GPLv2** because it's widely used and understood, making it easy to find resources and support. Penn does NOT support the release of software under the GPLv3 because it has much broader patent granting language than GPLv2.

## → Permissive Licenses

Choose a permissive license to foster wider adoption and collaboration and to preserve all options for future commercial uses of your code.

Reasons to choose a permissive license:

- You'd like as many people as possible to use the software.
- You work in open science and want people to be able to reproduce your research results.
- You want everyone to be able to use the code with no restrictions, including commercial users.
- Your project has an outside funder, such as NASA or the NSF, that requires permissive licensing with no other restrictions applied to the software.

Permissive licenses do not impose restrictions on derivative works, allowing companies and individuals to incorporate code into their proprietary projects without requiring them to release their own code under the same license. Anyone can use the code and change the license under which the code was released at any time. Generally, the only restriction is that attribution of all contributors to the code must be included.

This broader reach for a permissive license can significantly increase the impact and fuel innovation in the software development community. Additionally, permissive licenses typically simplify the process of sharing and contributing to the code, encouraging more active participation from developers worldwide. By opting for a permissive license, access to the code is democratized and its potential to benefit the broader software ecosystem is maximized.

The most commonly used permissive licenses are MIT, BSD-3-clause, and Apache 2.0. The Apache 2.0 license includes an explicit patent grant (so it should be used carefully and only after prior consultation with PCI), but the MIT and BSD-3-clause licenses do not contain such explicit patent grants.

---

## → Existing Community

When a project is related to an existing OSS community, or code from an existing community has been incorporated into a project, it can make ongoing adoption easier if the same license is used to release the new project.

To find the community's preferred license, look for a file called LICENSE or COPYING, and skim the project's README. If license cannot be identified, ask the developers or anyone who maintains the software. Depending on the original project's license, using the same license might be a requirement (i.e. GPL). Some communities have strong preferences for certain licenses. If you want to participate in one of these, it will be easier to use their preferred license, even if you're starting an entirely new project with no existing dependencies.

## → Other Circumstances Requiring Consideration

- The new project is related to technology for which patents have been filed
- The new project is related to technology licensed by Penn to a company

Please contact PCI to discuss your options: [PCI Contact](#). Publishing source code under an OSS license can negatively affect existing and future intellectual property rights and may cause Penn to breach its obligations under existing license agreements.

Some questions to consider when choosing an OSS license are:

- 1. Should a license be applied to software under development?** No software should be released without a license applied to the code if only to protect the creators' work and ensure attribution for the creator and the lab. Additionally, having a license ensures that you and Penn are protected from liability associated with use of the software by others.
- 2. Do all initial authors/contributors need to agree on the type of OSS license?** Everyone who might be impacted by the choice of license, such as your dean, department chair, and other authors, should be aligned on how best to protect and distribute the software.
- 3. Will the chosen OSS license facilitate or hinder third-party contributions or use?** Is broad distribution the sole goal or does the software present a commercial opportunity?
- 4. Do the funders of the work on the software require a certain form of software license?** Many NSF grants, for example, require distribution of software under specific OSS licenses.
  - Examine the funding agreement to understand the sponsor's requirements, if any, on intellectual property, OSS licensing, and publication of research. Identify any specific provisions related to OSS development, such as requirements for sharing results, restrictions on licensing, or obligations to release software or research outputs as open source. PCI can

---

interpret sponsor requirements and ensure that the chosen license aligns with both sponsor's and Penn policies.

- Select an OSS license that complies with the sponsor's policies. This might involve choosing licenses that are permissive or restrictive, depending on the sponsor's stipulations. PCI can assist with selection of the appropriate OSS license.
- Seek clarification from the sponsor if their policies on OSS are not clear or if there are any ambiguities regarding licensing and IP rights. PCI can assist with seeking such clarification.

**5. Could the software present a commercial opportunity or be the basis for a startup?** What might be the potential market for the software? Does the software have commercial benefits? PCI can assist with making this determination.

**6. Could the software be made available under multiple licenses?** For example, could the software be released under an MIT license but a 'premium' version of the software is made commercially available? Contact PCI for more detailed information and discuss such a dual licensing scheme.

**7. Have you consulted the "Open Source Software at Penn" Chart?** See Appendix A.

**It is important to add a license to any software before you release the software. If you have questions or want to discuss your options, please contact PCI - [PCI Contact](#)**

---

## C. Penn recommendations:

**The first OSS licenses that should be considered are the BSD or MIT licenses.** These licenses protect the institution and the authors of the software appropriately and keep the license language simple and direct. Importantly, these licenses also preserve certain commercialization options for the software. Other restrictive licenses such as the GPLv2 may be used after consultation with PCI; however, Penn strongly discourages the use of GPLv3.

## D. Applying a License to Your Repository

Once a license has been chosen, the next step is to ensure anyone looking at the code repository knows which license has been applied:

1. Create a license file named LICENSE or LICENSE.md and place it in the root directory of the repository.
2. Copy and paste the license text into the LICENSE file.
3. Commit the new file to the repository and push the changes.

## E. Best Practices

1. Educate anyone who is doing coding in the lab on the importance of being aware of the risk of incorporating OSS code and how to use the associated chart (Appendix A).
2. Require disclosure and documentation of all OSS code that is incorporated into any software being developed in the lab, the license under which the code was obtained and the source of the code (e.g. MIT license, obtained from github.)
3. Prior to releasing any software, contact PCI to discuss the proposed release. PCI can provide context and information so the software creators are all aligned on the plans for releasing the software.

## F. References

- A. Penn Policy <https://pci.upenn.edu/inventors/how-it-works-commercializing-innovation/policies/>
- B. OSS links
  - Open Source Initiative <https://opensource.org/>
  - Free Software Foundation <https://www.fsf.org/>
  - CopyLeft Currents – a blog about open source software <https://www.fsf.org/>
- C. Copyright Law of the United States of America and Related Laws Contained in Title 17 of the United States Code - <http://www.copyright.gov/title17/>

---

## III. The Chart

The OSS Chart in Appendix A contains detail that should assist in reviewing OSS license options. It includes a decision flow chart that can assist software developers in deciding on the most appropriate OSS license to use when releasing new software.

### A. Creating New OSS.

1. When new software is created that is not built upon a prior OSS project or community, and the software creators desire to release the software via an OSS license, the selected OSS license used best serve the project and the interests of Penn. For example, while BSD is a commonly used OSS license it may not be the best license to use in all cases and there may be a more appropriate license.
2. PCI has drafted an alternative OSS license, Appendix B, which is available to developers. It allows for distribution to non-commercial entities and restricts commercial use. This can be very useful if you wish to share the code with universities and other non-profit organizations and allowing them to build upon your code while preserving the ability to disseminate the software through commercial channels.

### B. Consulting and Making Use of the OSS Chart.

1. Use and distribution of software via an OSS license (or redistribution of third-party OSS) *must* comply with the terms of relevant OSS license(s). Not all OSS licenses are alike! Some OSS licenses (so-called “permissive” OSS licenses such as the BSD and MIT licenses) will allow licensees to use, modify, and redistribute the code with limited restrictions. Other OSS licenses require the licensee to distribute any modified or unmodified code under the same form of license (“Hereditary” or “Viral” is one term for these types of licenses). Other OSS licenses not only require the licensor to grant a license to its copyrights embodied in the code to the licensee, but also to grant a license, and all subsequent users, a license to any necessary patents owned by the licensor (whether such patents exist at the time the software was released or in the future) For purposes of this guide, we’ll call these Patent-Granting licenses. Examples of Patent-Granting licenses include GPL v.3.0 and Apache 2.0.
2. The OSS Chart, attached, provides an at-a-glance view of the most frequently used OSS licenses and the circumstances under which developers might use. The OSS licenses are listed in order of preference with complications highlighted as you move down the chart. Note the following footnotes: “H” indicates that the license is “Hereditary”, and “PG” indicates that the license is “Patent-Granting”.

- 
3. The color assigned to any given OSS license on the OSS Chart is intended to provide merely a high-level assessment of the relative risk related to patent grants of using a particular OSS license -- it does not consider the commercial value of the software nor whether OSS licensing and/or which particular OSS license is an appropriate licensing scheme given the specific software at hand. The red boxes are intended to highlight those OSS licenses that will be considered only in rare circumstances wherein the benefits to Penn and researchers are significant and the associated risks with the patent granting language are acceptable to the department and Penn.

# Appendix A

## OSS License Use and Redistribution of Modified & Unmodified OSS Chart

OSS Licenses	Internal use (at Penn)		External distribution (outside Penn)			
	OSS Unmodified	OSS with Bug Fixes or Otherwise Modified	OSS Unmodified	Bug Fixes	OSS Otherwise Modified	OSS function call only
<b>BSD</b> (Berkeley Software Distribution) <i>Must retain © notice</i>	Low Risk	Low Risk	Low Risk	Low Risk	Low Risk	Low Risk
<b>MIT</b> <i>Must include original © notice</i>	Low Risk	Low Risk	Low Risk	Low Risk	Low Risk	Low Risk
<b>Apache 1.x (not 2.0 or later)</b> <i>Must reproduce original © notice</i>	Low Risk	Low Risk	Low Risk	Low Risk	Low Risk	Low Risk
<b>GPL 1.x / 2.x, or LGPL 2.x</b> (GNU General Public License) – but <b>not</b> 3.0 or “any later version” of GPL <b>H</b>	Low Risk	Low Risk	Low Risk	Low Risk	Low Risk	Low Risk
<b>Educational Community License v2.0 (ECL-2.0)</b> <b>PG, H*</b>	Low Risk	Low Risk	Low Risk	Low Risk	Low Risk	Low Risk
<b>Apache 2.0</b> <b>PG, H*</b>	Low Risk with Internal Use Notice	Low Risk with Internal Use Notice	Low Risk	Low Risk	<b>Moderate Risk</b>	Low Risk
<b>Eclipse Public License 1.0 and 2.0</b> <b>PG, H</b>	Low Risk with Internal Use Notice	Low Risk with Internal Use Notice	Low Risk	Low Risk	<b>Moderate Risk</b>	Low Risk
<b>MPL 2.0</b> (Mozilla Public License) <b>PG, H</b>	Low Risk with Internal Use Notice	Low Risk w/ Bug Fixes ONLY	Low Risk	Low Risk	<b>Moderate Risk</b>	<b>Moderate Risk</b>
<b>Affero General Public License (AGPL) 3.0</b> <b>PG, H</b>	Low Risk with Internal Use Notice	Low Risk with Internal Use Notice	<b>High Risk</b>	<b>High Risk</b>	<b>High Risk</b>	<b>Moderate Risk</b>
<b>GPL 3.0 or LGPL 3.0</b> (GNU General Public License) <b>PG, H</b>	Low Risk with Internal Use Notice	Low Risk with Internal Use Notice	<b>High Risk</b>	<b>High Risk</b>	<b>High Risk</b>	<b>High Risk</b>
<b>Other OSS Licenses</b>	<b>Contact Licensing Office</b>					

GREEN [Low Risk]

YELLOW [Moderate Risk]

RED [High Risk]

**Legend** – colors are intended to guide the delegated authority through levels of potential risk regarding these OSS licenses. All OSS

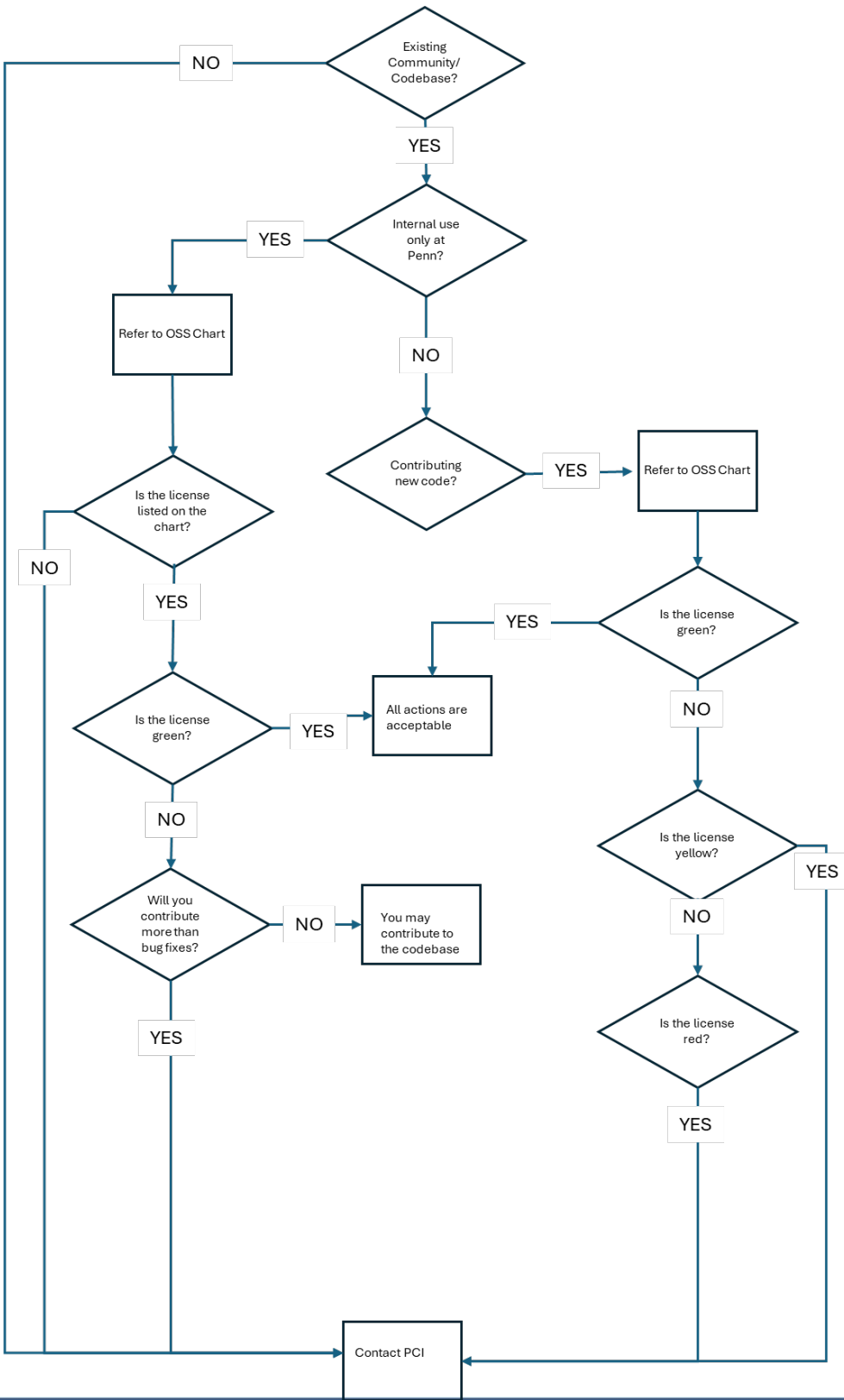
---

licensing of Penn-owned copyrights **requires** approval of the appropriate delegated authority, regardless of the “risk” categories.

*PG* is a “patent-granting” OSS license

*H* is a “hereditary” OSS license (sometimes referred to as viral). \*The heredity requirement may be overridden for derivative works in accordance with section 4 of Apache 2.0 and ECL 2.0 (i.e., it is possible to add your own copyright statement to your modifications).

**\*\*Used and modified with the permission of the Regents of the University of California.**





---

## Appendix B

Example Alternative License for distribution of OSS for non-profit and academic use only

PENN ACADEMIC SOFTWARE LICENSE AGREEMENT FOR  
"NAME OF SOFTWARE"

1. This is a legal agreement ("Agreement") between \_\_\_[name and address]\_\_\_\_\_ ("RECIPIENT" or "you"), and THE TRUSTEES OF THE UNIVERSITY OF PENNSYLVANIA ("PENN"). Penn controls "name and description of the software" ("Software") which was developed in the laboratory of or by [authors] at Penn. This Agreement shall be effective as of the date the Software is downloaded ("Effective Date").
2. By accepting, receiving, and using Software, including any accompanying information, materials or manuals you are agreeing to be bound by the terms of this Agreement.
3. PENN grants to RECIPIENT a royalty-free, nonexclusive, and nontransferable license to copy, reproduce, perform, display, and make derivative works of the Software furnished hereunder, upon the terms and conditions set out below.
4. RECIPIENT agrees to use the Software solely for internal research or non-commercial purposes and shall not distribute or transfer the Software to another location or to any other person without prior written permission from PENN. Nothing in this license grants any rights for commercial use.
5. RECIPIENT is encouraged to provide feedback on the Software or any updates or bug fixes back to the laboratory of \_\_\_\_\_ at \_\_\_\_\_.
6. RECIPIENT acknowledges that the Software is still in the development stage and that it is being supplied to Recipient **as is, with all faults and without any accompanying documentation, services, support or improvements from PENN.**
7. PENN does not grant any licenses under any PENN patent or patent application by this Agreement. The rights and licenses granted in this Agreement are limited to the scope expressly granted. Accordingly, except for the rights expressly granted under this Agreement, no right, title, or interest of any nature whatsoever is granted whether by implication, estoppel, reliance, or otherwise, by Penn to Recipient. All rights with respect to any know-how, patent or other intellectual property rights that are not specifically granted herein are reserved to the owner thereof.
8. Title and copyright to the Software and any derivatives and any associated documentation shall at all times remain with PENN, and RECIPIENT agrees to preserve same.
9. RECIPIENT agrees not to reverse engineer, reverse assemble, reverse compile decompile, disassemble, or otherwise attempt to re-create the source code for the Software. RECIPIENT acknowledges that any programs created based on the Software will be considered a derivative of Software and owned by PENN.
10. RECIPIENT may not further distribute Software without express written permission of PENN; however, if permission to transfer the Software is given, RECIPIENT warrants that RECIPIENT will not remove or

---

export any part of the Software from the United States except in full compliance with all United States export regulations and other applicable laws.

11. RECIPIENT will use the Software in compliance with all applicable laws, policies and regulations including, but not limited to, any approvals, informed consent and patient confidentiality principles.
12. **PENN MAKES NO REPRESENTATIONS AND EXTENDS NO WARRANTIES OF ANY KIND, EITHER EXPRESS OR IMPLIED OTHER THAN ARE SET OUT IN THIS AGREEMENT.**
13. **THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS, CONTRIBUTORS, AND THE TRUSTEES OF THE UNIVERSITY OF PENNSYLVANIA "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER, CONTRIBUTORS OR THE TRUSTEES OF THE UNIVERSITY OF PENNSYLVANIA BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.**
14. RECIPIENT will indemnify, hold harmless, and defend PENN against any claim of any kind arising out of or related to the exercise of any rights granted under this Agreement or the breach of this Agreement by RECIPIENT.
15. This Agreement may be terminated by either party upon thirty (30) days written notice to the other party. In the event of termination, RECIPIENT shall destroy or return immediately all Software and all copies thereof to PENN.
16. The parties further waive any right to challenge the admissibility or authenticity of this document in a court of law based solely on the absence of an original signature.
17. Any notice, request, approval or consent required or permitted to be given under this Agreement shall be in writing and directed to a party at its address shown below or such other address as such party shall have last given by notice to the other party. A notice will be deemed received: if delivered personally, on the date of delivery; if mailed, five (5) days after deposit in the United States mail; if sent via courier, one (1) business day after deposit with the courier service.
18. **Governing Law.** This Agreement shall be governed by and interpreted in accordance with the laws of the Commonwealth of Pennsylvania, excluding application of any conflict of laws principles that would require application of the law of a jurisdiction outside of the Commonwealth of Pennsylvania.
19. If a dispute arises between the parties concerning any right or duty under this Agreement, then the parties will confer, as soon as practicable, in an attempt to resolve the dispute. If the parties are unable to resolve the dispute amicably, then the parties will submit to the exclusive jurisdiction of, and venue in, the state and Federal courts located in the Eastern District of Pennsylvania with respect to all disputes arising under this Agreement.

