Ethics in Software Licensing

Christian Howell

STEM 300 – Computer Ethics, West Virginia University at Parkersburg

October 2, 2024

            Software licensing is vital in both proprietary and open-source software environments. Understanding different types of licensing and their impact on developers and businesses can help people make ethical decisions about which license to use in any given scenario. This paper will cover an overview of license types, compare different licenses and their agreements, and review a case study of RedHat’s actions with the GPLv2 license. By the end of this paper, we should be informed well enough to carefully choose a license that meets our ethical and technical standards.

Before we can examine common types of licenses and understand them better, we need to define commercial, copyleft, and permissive licensing. According to an article from Open Source Collection, “Commercial software is created and distributed by a single organization that maintains sole authority over the source code and distribution rights.” [1] With commercial licensing, the end-user cannot view, modify, or publish code that is a derivative of the software, and it is neither ethical nor legal to do so. Some of the benefits of commercial licensing include having access to expert assistance, predictable terms and conditions while using the software, and maintaining a competitive advantage on the market with proprietary features. Some drawbacks include increased liability for the developers, increased risk of piracy, dependency on the manufacturer, and increased costs. Some common examples of commercial licensing are operating systems like Windows 11 and MacOS. In contrast to the restrictive nature of commercial licensing, according to an article written for Britannica, “Copyleft is a general license agreement granted by a copyright owner permitting anyone to use copyrighted property freely but under specific terms.” [2] To clarify, copyleft allows users to make derivative variants of an original piece of software, but with some restrictions set to ensure that the developer’s original intentions and goals with the software remain possible. This means that if someone were to make a derivative software off one with a copyleft license, their software, in turn, would be required to remain under the same license as the original. Some examples of copyleft licenses include the General Public License (GPL) and the Affero General Public License (AGPL), which the GNU Project defines. One step less restrictive than copyleft licensing, Deepak Gupta, a technology entrepreneur, defines permissive licensing as the following: “Permissive licenses offer the most freedom to users. They allow the software to be used, modified, and distributed for any purpose, including commercial use, with minimal restrictions. The main requirement is usually to give credit to the original authors.” [3] Permissive licensing, a form of open-source licensing, allows users to use the application freely for profit or in any way they desire. The only minor restriction is that credit must be given to the original creator when published. This license type works best with software that seeks to adopt widespread use for personal and commercial use. Some examples of permissive licensing include the Apache 2.0 license and the BSD 3 clause.

We have mentioned a few licenses from each category; let us dive into more details about each, including their pros and cons, legal limitations, potential incorporation into other software, and some ethical scenarios that may arise when using the license. First off, let us look at a Windows 11 license. If we scan the agreement, we can define Windows 11 as a commercial license, which means that users must purchase a license to use the operating system to use it legally. Windows 11 comes in two forms for the general user: Home and User. While Windows Home is more affordable, it lacks some features Pro offers. Windows Pro has more tools like BitLocker and Remote Desktop but is more expensive than Home. The legal limitations of Windows are rigorous, including restrictions on transferring licensing between user devices, prohibiting any modification of the software, or reverse-engineering the software. Violating these terms can result in agreement termination or even legal proceedings. As such, Windows cannot be incorporated into any other software, as not only is it an operating system, but it is also not permitted by Microsoft. One example of an ethical but not legal scenario with Windows 11 is with the license transfers. If a user ethically installs Windows on multiple personal machines and only uses one at a time, it would be morally fair to assume that this is a fair use of the software. However, according to the agreement with Microsoft, this action is not allowed and violates their terms. Another license incredibly, like Windows, is Apple’s MacOS license agreement. MacOS is licensed under Apple and their commercial license. It is restricted to only function on hardware manufactured by Apple but can be installed on other hardware using a modified client. One benefit of this is that, much like Windows, it operates incredibly well when used in its intended function on the intended devices, as it has regular security and system updates. The downside of this agreement is that it is not only restricted to Apple devices but that all Apple devices also come at a reasonably hefty premium. The primary legal limitations of the MacOS agreement are about the same as those of Windows but include a specific rule about the operating system only being permitted to run on Apple devices and running it outside of that is a direct violation of the agreement. With this being said, its incorporation into other software is impossible, as it is restricted to Apple hardware and software. An example of an ethical but not legal scenario is where a developer could create a MacOS virtual machine for personal use on non-Apple hardware. While this is ethically sound, as it is with benign intention, it is not permitted by Apple’s MacOS license agreement.

Now that we have looked at commercial licensing let us move into copyleft licensing. GPLv3 (General Public License, Version 3) allows users to use, modify, and publish any form of software. Still, any derivative work must also be under the GPLv3 License to maintain the open-source nature of the original project. This leads to the benefits of a high degree of freedom when it relates to modifying and distributing software. Still, it also deters potential companies from using it as a commercial licensing solution. The only legal limitation that GPLv3 has to offer is that it forces open-source on any software that uses it for licensing, meaning that any commercial or closed-source software would violate this agreement.

An example would be a developer who ethically believes in using a piece of GPLv3 licensed code for their proprietary code, which is ethical but not legally permitted by the GPLv3 license, as that software would also need to be open-source. Another copyleft license is the AGPLv3 (Affero General Public License, Version 3), which functions similarly to the GPLv3 but adds additional protections for network-based applications, like software as a Service (SaaS). This leads to a very apparent benefit of covering more forms of software, but it has the same potential drawbacks that come with the GPLv3 license. This also applies to the legal limitations, where the only additional note needs to be for SaaS products, as they now fall under the core copyleft principles of the GPLv3. This also means that the license allows for even more integrations, especially in the age of web-based software and the general connectivity of the modern Internet. An example of an ethical scenario is where a company provides a SaaS product under the AGPLv3 license but does not release any part of its software or any modifications, as the end user would ‘never need it.’ However, if they choose to use this license, they are both ethically and legally required to publish that code for all; failing to do so would violate the license agreement.

We have now reviewed some common forms of copyleft licensing. Let’s further review some aspects of permissive licensing, starting with Apache 2.0. Apache 2.0 is a permissive licensing form allowing users to freely modify, distribute, and use software for any purpose, including commercial software. This means that a company can use this license for proprietary software or modify open-source software without needing to make their software openly available. This has the benefit of being open to everyone, even companies attempting to profit from their proprietary software. Some of the legal limitations include requiring you to credit the original creator for any software modifications, and the license must be retained within your software. This loose level of legal restrictions means that integration into other software is permitted for open-source and proprietary software, making it an incredibly robust license agreement to implement. An ethical concern with this is that a company could take an utterly open-source piece of code, pass it as their own, and profit from it. While this is legal, it is highly unethical as it could be deemed code plagiarism and undermine open-source software's free and collaborative nature. Much like Apache 2.0, the BSD 3-Clause license allows users to create, modify, or publish any software or code they desire, including for commercial purposes. One note to differentiate between the two licenses is that BSD3 benefits from being brief and is commonly used in academia and industry. This also comes with drawbacks similar to Apache 2.0, as it allows modified software versions to become proprietary. The legal limitations also mimic the Apache 2.0 software license but add the required need for original copyright holders of the software to be credited in any revisions or modifications. This likewise means that integration into other software is extremely fast and straightforward, assuming the software is not a proprietary derivative. One ethical concern from this license is that software can be modified under certain conditions without crediting the original creator. While this, in some instances, is legal, it is not ethical, as it undermines the ethical integrity of the person modifying it and builds distrust within the open-source community.

Now that we have better understood how licenses work and some examples of different types of licenses, let us dive into a case study to apply what we have learned to a real-world scenario. In 2023, Red Hat, a software development company, modified how it distributed its Linux distribution source code. It was based on the GPLv2 license, which requires the developer to share the software's source code under the license openly. Initially, Red Hat publicly made this code available, but they shifted it to only provide the source code to customers who actively paid for their services. This change affected other Linux distributions like Rocky Linux and CentOS, as the open-source framework of RHEL was no longer available to the public. This then begs the question, was their action legal? Well, in short, yes. Red Hat adheres to the GPLv2 license agreement by providing the binaries for all versions of its operating system to paying customers. They are legally not required to provide the source code for the public. While this is deemed legal, we need to consider the ethical implications of Red Hat's decision. Some people argue that this change to their business model compromises the open-source nature of the license and the community surrounding Linux, as Linux was made an open-source alternative to Unix when it was developed. This change also made it more difficult for others to modify the RHEL distribution, which was designed to be more community-driven, leading to more distrust. On the other hand, it could be deemed ethical in one aspect. Red Hat is a business that focuses on developing the Linux distribution RHEL. Shifting their business model to require more payment ensures that they will continue to provide and maintain this piece of software for years to come without the risk of going bankrupt or other financial risks. In summary, while what Red Hat did was legal under the rules of GPLv2, it did harm its reputation as a developer despite its actions being for longer-term ethical reasons.

In conclusion, this paper reviewed commercial licensing practices with direct analysis of Windows and MacOS as examples, copyleft practices by analyzing licensing like the GPLv3 and AGPLv3, and permissive licensing like Apache 2.0 and BSD3. We also took the knowledge from licensing types and examples to ethically and legally analyze the controversy between Red Hat and the GPLv2 license. Software licensing is a challenging balance between ethical considerations and legal restrictions. At the same time, some licenses aim to provide an open development avenue for all, while others focus more on protecting original, proprietary works. By understanding this balance, we can better make ethical decisions on licensing our software, ensuring fair use, and all can use software for good.

# Works Cited

|  |  |
| --- | --- |
| [1] | OpenSourceCollection, "Understanding Software Licenses: Open-Source vs Commercial," OpenSourceCollection, 14 May 2024. [Online]. Available: 2024. [Accessed 2 October 2024]. |
| [2] | K. Friedman, "copyleft," Britannica, 26 August 2024. [Online]. Available: https://www.britannica.com/topic/open-source. [Accessed 2 October 2024]. |
| [3] | D. Gupta, "Open Source Licensing 101: Everything You Need to Know," Meet the Tech Entrepreneur, Cybersecurity Author, and Researcher, 17 June 2024. [Online]. Available: https://guptadeepak.com/open-source-licensing-101-everything-you-need-to-know/. [Accessed 2 October 2024]. |
| [4] | Microsoft, Inc., "Microsoft License Terms," April 2024. [Online]. Available: https://www.microsoft.com/content/dam/microsoft/usetm/documents/windows/11/oem-(pre-installed)/UseTerms\_OEM\_Windows\_11\_English.pdf. [Accessed 2 October 2024]. |
| [5] | Apple Inc., "SOFTWARE LICENSE AGREEMENT FOR macOS Monterey," 17 September 2022. [Online]. Available: https://www.apple.com/legal/sla/docs/macOSMonterey.pdf. [Accessed 2 October 2024]. |
| [6] | GNU, "GNU GENERAL PUBLIC LICENSE," 29 June 2007. [Online]. Available: https://www.gnu.org/licenses/gpl-3.0.html. [Accessed 2 October 2024]. |
| [7] | GNU, "GNU AFFERO GENERAL PUBLIC LICENSE," 19 November 2007. [Online]. Available: https://www.gnu.org/licenses/agpl-3.0.en.html. [Accessed 2 October 2024]. |
| [8] | Apache Software Foundation, " Apache License," January 2004. [Online]. Available: https://www.apache.org/licenses/LICENSE-2.0.html. [Accessed 2 October 2024]. |
| [9] | Open Source Initiative, "The 3-Clause BSD License," 22 July 1999. [Online]. Available: https://opensource.org/license/BSD-3-clause. [Accessed 2 October 2024]. |
| [10] | B. M. Kuhn, "A Comprehensive Analysis of the GPL Issues With the Red Hat Enterprise Linux (RHEL) Business Model," Software Freedom Conservancy, 23 June 2023. [Online]. Available: https://sfconservancy.org/blog/2023/jun/23/rhel-gpl-analysis/. [Accessed 2 October 2024]. |