

Chapter XXVII

The Road of Computer Code Featuring the Political Economy of Copyleft and Legal Analysis of the General Public License

Robert Cunningham
Southern Cross University, Australia

ABSTRACT

This chapter examines the development of open source computer software with specific reference to the political economy of copyleft and the legalities associated with the General Public License (GPL). It will be seen that within the context of computer software development the notion of copyleft provides an important contrast to more traditional uses of copyright. This contrast symbolizes political, economic, and social struggles which are contextualized within this chapter. As the GPL is an important legal embodiment of copyleft, its legalities are preliminarily explored so as to determine its future potential. While there is some scope to further refine the legal strength of the GPL, it will be seen that it remains a strong and subversive legal instrument which will continue to underlie open source initiatives in the years to come.

INTRODUCTION

This chapter has two distinct objectives. Firstly to survey the political economic foundation of copyleft as it applies to open source computer software, and secondly, to provide some preliminary legal analysis in relation to the General Public License (GPL) which legally embodies copyleft principles. The political economic dimension of

the chapter embraces a philosophical approach on the basis that “philosophy offers nuance where there was none” (Lehman, 1999, p. 239). In relation to the GPL legal analysis, it should be noted by way of disclaimer that the commentary constitutes legal analysis, not legal advice.

The chapter begins its philosophical exploration by giving a brief overview of copyright as it applies to the language of computer software,

specifically source code. This is followed by a discussion that contrasts closed source and open source software development. It will be seen that this contrast is grounded in a political, economic, and social struggle which, almost classically, fits into the right/left political divide. This divide is made explicit by elucidating the anarchist tendencies of open source software development via a discussion of the “tragedy of the anticommons” theory. The political divide is also drawn upon by way of juxtaposing the traditional notion of copyright with the open source notion of copyleft.

Copyleft is an innovative concept derived by the open source movement which draws upon traditional copyrights in an unconventional manner so as to maximize information flow. It is enshrined within Provision 2(b) of the GPL and remains an important legal mechanism of the open source movement. Those readers that are primarily interested in the legal analysis, as opposed to the political economic dimensions of copyleft, are encouraged to turn directly to the second half of this chapter. It will be seen in this part, and thereafter, that the GPL raises a plethora of interesting legal issues specifically arising from its duality as a license *and* as a contract. While the wholesale enforceability of the GPL escapes the parameters of this chapter, the license/contract duality of the license, which has led to it commonly being referred to as a contractual license, is touched upon in order to uncover potential latent legal issues. It will be seen that implicit within the GPL discussion is the apparent self-enforcing nature of the license which makes it at once a strong and subversive legal instrument.

BACKGROUND

Copyright and Computer Software

Copyright is one of the important pillars of the international intellectual property right (IPR) regime. Its practical effect is broad in relation to

both its application and its subject matter. As St. Laurent (2004) explains, application of copyright is automatically inferred in a broad manner to the point that a drawing of a flower on a café napkin is copyrighted simultaneously with its creation and, generally speaking, becomes the sole property of its creator. The drawing of the flower cannot be displayed, copied or otherwise commercially exploited by any person other than the creator for the life of the copyright. Under copyright law, no person other than the creator can create “derivative works,” which are works that depend upon or develop from the original, copyrighted work. In many cases there is no need for registration of the right as it automatically attaches to every novel expression of an idea, whether through text, sounds or imagery for the period of the life of the copyright. In countries such as the USA and Australia this is the life of the creator plus 70 years.

Over and beyond the broad *application* of copyright, *subject matter* is also far-reaching touching upon a wide-range of endeavours; from the mundane such as timetables and betting coupons; to the truly artistic such as films, literature and music; to technology-based products such as television broadcasts and computer software (Caenegem, 2001). It is the latter form that is the subject of this chapter. Although computer *software* has escaped a statutory definition in many Anglo-American countries, Justice Gibbs in the Australian High Court did define the notion of computer *program* broadly in *Computer Edge Pty Ltd. Vs. Apple Computer Inc.* (1986) 161 CLR 171 at 178-179 as:

a set of instructions designed to cause a computer to perform a particular function or to produce a particular result. (emphasis added)

The “instructions” manifest in what is called computer code which generally exists in three formats: flowcharts, source code, and object code (Carstens, 1994). Programmers initially draft a

13 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:
www.igi-global.com/chapter/road-computer-code-featuring-political/21200

Related Content

Open Source E-Learning Systems: Evaluation of Features and Functionality

Phillip Olla (2007). *Handbook of Research on Open Source Software: Technological, Economic, and Social Perspectives* (pp. 638-648).

www.irma-international.org/chapter/open-source-learning-systems/21222

A Model of Cultural Competence in Open Source Systems

Doris Wright Carroll (2015). *Open Source Technology: Concepts, Methodologies, Tools, and Applications* (pp. 802-811).

www.irma-international.org/chapter/a-model-of-cultural-competence-in-open-source-systems/120943

Organizational Influencers in Open-Source Software Projects

Roland Robert Schreiber (2023). *International Journal of Open Source Software and Processes* (pp. 1-20).

www.irma-international.org/article/organizational-influencers-in-open-source-software-projects/318400

Open Source Software Usage in Education and Research: Network Traffic Analysis as an Example

Samih M. Jammoul, Vladimir V. Syuzev and Ark M. Andreev (2021). *Research Anthology on Usage and Development of Open Source Software* (pp. 273-288).

www.irma-international.org/chapter/open-source-software-usage-in-education-and-research/286578

Classification of Software Defects Using Orthogonal Defect Classification

Sushil Kumar, SK Muttooan and V. B. Singh (2022). *International Journal of Open Source Software and Processes* (pp. 1-16).

www.irma-international.org/article/classification-of-software-defects-using-orthogonal-defect-classification/300749