

Chapter XXXI

FLOSS Legal and Engineering Terms and a License Taxonomy

Darren Skidmore
Monash University, Australia

ABSTRACT

This chapter introduces the reader to terms relevant to understanding free/libre and open source licenses, some of the relevant legal, and relevant software engineering terms that are useful in understanding the issues in FLOSS. Then a brief history of FLOSS licenses is given before introducing a taxonomy to help understand the types of licenses which are available in the FLOSS domain. A brief description to think about differing views of the usage and users of FLOSS is given in conclusion.

INTRODUCTION

The purpose of this chapter is to explain some of the issues which free/libre open source software (FLOSS) licenses are attempting to address, although it should be noted at the outset that these also apply to any type of software license. The chapter firstly discusses the legal terms applicable in intellectual property with an emphasis on FLOSS. To complement the legal issues, discussion turns to software terms and their definitions as part of software development and engineering. Having defined the two areas pertaining to the FLOSS licenses, a brief history is given before discussing a taxonomy of FLOSS Licenses. The chapter concludes with a brief discussion on how the view of user of the FLOSS may change the need for a type of license. The purpose of

explaining the legal and software engineering terms is because if a person does not have a background in these areas, then it is unclear as to why licenses, or debate about the outcome of licenses, are being made. The taxonomy is given to help readers understand that there are several license types, and to assist them in their choice of a license or in understanding the outcomes attached to a license.

As more organisations adopt or consider FLOSS, there is a greater need to understand at a more generic level the broad aims or outcomes of the effects of the licenses. The Commonwealth of Massachusetts compares 52 different licenses (Commonwealth of Massachusetts, 2004), the Open Source Initiative (OSI) compares over 58 (Open Source Initiative, 2004b), while the Free Software Foundation (FSF) lists and comments

on almost 100 (Free Software Foundation, 2005a), and the ifrOSS lists over 180 different open source licenses (ifrOSS, 2005). Each independent license has different conditions and outcomes; legally it is important to understand the clauses in a specific license, but before that there is a need to understand the broad aims of the license, and to match that to the organisational needs and requirements, of an application. Certainly in terms of ICT governance a taxonomy of FLOSS licenses helps to match the organisational strategic and tactical aims with the operational choice of the specific license. There has been comment that there are too many open source licenses (Skidmore, 2006), with the OSI looking into the proliferation of licenses (Open Source Initiative, 2005), and comment that it is not that there should be less licenses, but that there should be a cleaning up of the terms used and agreement on how to word specific desired outcomes (Rosen, 2005).

The term FLOSS is used to describe free/libre and open source software. The word *libre* is specifically included to emphasis the concept is about freedom, rather than price. Also within this chapter, when referring to an application, the term can include a computer program, which could be a word processor, a Web browser, or an email program, but also an operating system such as Linux. Although there is a distinct difference in terms of what functions these different types of programs do, the issues in licensing are, in the main, similar. The term FLOSS means F/LOS-software, so when taking about the software just the acronym FLOSS is used.

Before discussing the licenses, it is important to firstly explain some of the legal issues which are trying to be addressed by the licenses, not all licenses with each of these issues, some of the legal issues did not really exist when some were created, or were not considered by the authors of the licenses. Complementary to the legal issues are the software engineering terms which also influence the licenses. Certainly debate about

new or updating FLOSS licenses are aimed at issues in software engineering or software usage which are being practiced now. In reading the chapter readers of course should be aware of the changes in the last four decades in both software engineering and in legal jurisprudence which, because they are constantly changing, do have implications for the FLOSS licenses.

BACKGROUND: LEGAL TERMS

The legal terms in Table 1 are not an exhaustive list of either the terms or of the scope of these terms, nor is the full complexity of the issues of applying to software treated, however the list does include the more important terms and states some of the more critical issues. Table 1 gives an overview of the terms that will be covered in this chapter. Although it is possible to sell an entire program, the most common form of transfer in software development is that of licensing (von Krogh & von Hippel, 2003), which is why the aspects pertaining to licensing are focussed on in this chapter. There are several works which are dedicated to the legal issues in software and explanations in depth of the issues (Rosen, 2004; St. Laurent, 2004; Välimäki, 2005).

Table 1. List of legal terms discussed

Intellectual Property terms
Author / Owner
Copyright / Patent
Trademark
Derivative Work
License / Contract
Jurisdiction terms
Choice of Forum
Choice of Law
Other Legal terms
Consumer Warranty
Export Control
Distribution
Written Language
Reasonable and Non Discriminatory Licensing (RAND)

15 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/floss-legal-engineering-terms-license/21204

Related Content

Implementing an Open Source ePortfolio in Higher Education: Lessons Learned Along the Way
Stein Brunvard, Gail R. Luera, Tiffany Marraand Melissa Peet (2011). *Free and Open Source Software for E-Learning: Issues, Successes and Challenges* (pp. 132-146).

www.irma-international.org/chapter/implementing-open-source-eportfolio-higher/46312

Requirements to Class Model via SBVR: RECM via SBVR TOOL

Murali Mohananand Imran Sarwar Bajwa (2019). *International Journal of Open Source Software and Processes* (pp. 70-87).

www.irma-international.org/article/requirements-to-class-model-via-sbvr/233514

Open Source Technology and Ideology in the Nonprofit Context

Jonathan Peizer (2007). *Handbook of Research on Open Source Software: Technological, Economic, and Social Perspectives* (pp. 468-479).

www.irma-international.org/chapter/open-source-technology-ideology-nonprofit/21209

Developing a Dynamic and Responsive Online Learning Environment: A Case Study of a Large Australian University

Janet Buchan (2011). *Free and Open Source Software for E-Learning: Issues, Successes and Challenges* (pp. 92-109).

www.irma-international.org/chapter/developing-dynamic-responsive-online-learning/46309

Using Open Source to Building and Accessing Learning Objects and Online Courses

Christos Bourasand Maria Nani (2007). *Open Source for Knowledge and Learning Management: Strategies Beyond Tools* (pp. 266-297).

www.irma-international.org/chapter/using-open-source-building-accessing/27815