

UPGRADE is the European Journal for the Informatics Professional, published bimonthly at <<http://www.upgrade-cepis.org/>>

#### Publisher

UPGRADE is published on behalf of CEPIS (Council of European Professional Informatics Societies, <<http://www.cepis.org/>>) by **Novática** <<http://www.ati.es/novatica/>>, journal of the Spanish CEPIS society ATI (*Asociación de Técnicos de Informática*, <<http://www.ati.es/>>)

UPGRADE monographs are also published in Spanish (full version printed; summary, abstracts and some articles online) by **Novática**

UPGRADE was created in October 2000 by CEPIS and was first published by **Novática** and **INFORMATIK/INFORMATIQUE**, bimonthly journal of SVI/FSI (Swiss Federation of Professional Informatics Societies, <<http://www.svifsi.ch/>>)

UPGRADE is the anchor point for UPENET (UPGRADE European NETWORK), the network of CEPIS member societies' publications, that currently includes the following ones:

- **Informatik-Spektrum**, journal published by Springer Verlag on behalf of the CEPIS societies GI, Germany, and SI, Switzerland
- **ITNOW**, magazine published by Oxford University Press on behalf of the British CEPIS society BCS
- **Mondo Digitale**, digital journal from the Italian CEPIS society AICA
- **Novática**, journal from the Spanish CEPIS society ATI
- **OCG Journal**, journal from the Austrian CEPIS society OCG
- **Pliroforiki**, journal from the Cyprus CEPIS society CCS
- **Pro Dialog**, journal from the Polish CEPIS society PTI-PIPS

#### Editorial Team

Chief Editor: Rafael Fernández Calvo, Spain, <[rfoalvo@ati.es](mailto:rfoalvo@ati.es)>

Associate Editors:

François Louis Nicolet, Switzerland, <[nicolet@acm.org](mailto:nicolet@acm.org)>

Roberto Carniel, Italy, <[carniel@dgf.uniud.it](mailto:carniel@dgf.uniud.it)>

Zakaria Maamar, Arab Emirates, <[Zakaria.Maamar@zu.ac.ae](mailto:Zakaria.Maamar@zu.ac.ae)>

Soraya Kouadri Mostefaoui, Switzerland,

<[soraya.kouadrimostefaoui@unifr.ch](mailto:soraya.kouadrimostefaoui@unifr.ch)>

#### Editorial Board

Prof. Wolfried Stucky, CEPIS Past President

Prof. Nello Scarabottolo, CEPIS Vice President

Fernando Piera Gómez and

Rafael Fernández Calvo, ATI (Spain)

François Louis Nicolet, SI (Switzerland)

Roberto Carniel, ALSI - Tecnoteca (Italy)

#### UPENET Advisory Board

Hermann Engesser (Informatik-Spektrum, Germany and Switzerland)

Brian Runciman (ITNOW, United Kingdom)

Franco Filippazzi (Mondo Digitale, Italy)

Rafael Fernández Calvo (Novática, Spain)

Veith Risak (OCG Journal, Austria)

Panicos Masouras (Pliroforiki, Cyprus)

Andrzej Marciniak (Pro Dialog, Poland)

**English Language Editors:** Mike Andersson, Richard Butchart, David Cash, Arthur Cook, Tracey Darch, Laura Davies, Nick Dunn, Rodney Fennemore, Hilary Green, Roger Harris, Michael Hird, Jim Holder, Alasdair MacLeod, Pat Moody, Adam David Moss, Phil Parkin, Brian Robson

Cover page designed by Antonio Crespo Foix, © ATI 2006

Layout Design: François Louis Nicolet

Composition: Jorge Llácer-Gil de Ramales

Editorial correspondence: Rafael Fernández Calvo <[rfoalvo@ati.es](mailto:rfoalvo@ati.es)>

Advertising correspondence: <[novatica@ati.es](mailto:novatica@ati.es)>

UPGRADE Newsletter available at

<<http://www.upgrade-cepis.org/pages/editinfo.html#newsletter>>

#### Copyright

© Novática 2006 (for the monograph and the cover page)

© CEPIS 2006 (for the sections MOSAIC and UPENET)

All rights reserved under otherwise stated. Abstracting is permitted with credit to the source. For copying, reprint, or republication permission, contact the Editorial Team

The opinions expressed by the authors are their exclusive responsibility

ISSN 1684-5285

## Monograph of next issue (August 2006) "The 'Bologna Process' and The Informatics Profession"

(The full schedule of UPGRADE  
is available at our website)



The European Journal for the Informatics Professional

<http://www.upgrade-cepis.org>

Vol. VII, issue No. 3, June 2006

2 Editorial  
Farewell to An UPGRADE Founder — *Geoff McMullen* (President of CEPIS)

2 From The Editor's Desk  
Thank You for The Privilege ... — *Rafael Fernández Calvo*

3 British Magazine ITNOW Joins UPENET

### Monograph: Free Software Licenses in Context (published jointly with Novática\*)

Guest Editors: *Luis Fajardo López* and *Jean-Baptiste Soufron*

4 Presentation. The Wide World of Libre/Free Software Licenses —  
*Luis Fajardo-López*

10 The Legal Protection of Computer Programs under Spanish and European Law — *Javier Plaza-Penadés*

13 DRM as A Dangerous Alternative to Copyright Licences — *David Monniaux and Jean-Baptiste Soufron*

16 Ideological Foundations and Real Effects of The Current Model of Intellectual Property — *Eduardo Melero-Alonso*

22 The Legal Framework of Software Licences: Is The Will of The Licensor Subject To Constraints? — *Luis Fajardo-López*

29 Some Legal Reflections on The Use of Libre Software Licences in Public Administrations — *Luis Fajardo-Spínola and Luis Fajardo-López*

33 Free Software in Extremadura, The History of Some Good Reasons — *Luis Millán-Vázquez de Miguel*

38 Creative Commons: Open Content Licenses to Govern Creative Works — *Melanie Dulong de Rosnay*

41 Scientific Publications: The Role of Public Administrations in The ICT Era — *Roberto Di Cosmo*

### UPENET (UPGRADE European NETWORK)

49 From **Informatik Spektrum** (GI, Germany, and SI, Switzerland)  
High Performance Computing  
High Performance Computing Technology, Applications and Business — *Luigi Brochard*

56 From **ITNOW** (BCS, United Kingdom)  
Informatics Profession  
The Changing Landscape — *Karen Price*

\* This monograph will be also published in Spanish (full version printed; summary, abstracts, and some articles online) by **Novática**, journal of the Spanish CEPIS society ATI (*Asociación de Técnicos de Informática*) at <<http://www.ati.es/novatica/>>.

# DRM as A Dangerous Alternative to Copyright Licences

David Monniaux and Jean-Baptiste Soufron



David Monniaux and Jean-Baptiste Soufron, 2006. This article is published under the terms of the license "Creative Commons Attribution 2.5", <<http://creativecommons.org/licenses/by/2.5/deed.en>>

*Copyright Licences are the traditional way for authors to control software but Digital Rights Management (DRM) techniques are becoming increasingly used, especially since new laws to that effect are being adopted around the world. This is altering the delicate balance between the need for authors to obtain revenue from their software, and the need for the public to access and use this software. Ultimately, these two models seem to be in opposition. Instead of being seen as the continuation of copyright rules, DRM could be more appropriately described as a whole new kind of intellectual property.*

**Keywords:** Copyright, DADVSI, DRM, EUCD, Intellectual Property.

## 1 Introduction

Most people see licences as the normal way to implement software copyright. But Digital Rights Management (DRM) techniques - a new-fangled term for copy protection systems - are now legally protected against 'circumvention' in many jurisdictions. We explain here how this protection can, depending on how it is legally worded, introduce a de facto new kind of intellectual property. We will then try to understand how this new kind of intellectual property can be a replacement for Software Licences, and an obstacle to their development.

The 1996 World Intellectual Property Organization Copyright Treaty (WCT)<sup>1</sup>, the 1998 Digital Millennium Copyright Act (DMCA)<sup>2</sup>, the 2001 European Union Copyright Directive (EUCD)<sup>3</sup>, and the 2006 French bill known as DADVSI (*Droit d'Auteur et Droits Voisins dans la Société de l'Information - Copyright and Neighbouring Rights in The Information Society Bill*)<sup>4</sup> all grant specific protection to so-called digital rights management techniques. We contend that this protection introduces a de facto new kind of intellectual property.

## 2 The Legal Protection of Software by Copyright

Authors (or publishers) of computer software are granted a copyright on their work - unauthorized copying of software is illegal in most jurisdictions. This protection is very

similar to that granted to books, photographs, songs etc. (though under *droit d'auteur* legislations, software authors do not typically enjoy as many "moral rights" as authors of other kinds of work). Accordingly, this copyright only covers specific programs, as opposed to the abstract ideas and techniques used in software production.

This is an important distinction to make. Computer software is generally written as source code (a human-read-

### Authors

**David Monniaux** is a researcher at the *Centre National de la Recherche Scientifique* (CNRS), Engineering Dept., in the *Laboratoire d'Informatique de l'École Normale Supérieure* (LIENS), in France. He works in the Abstract Interpretation and Semantics team led by Patrick Cousot. He is also teaching at the Computer Science Dept. of the *École Polytechnique*, Palaiseau, Paris, France. His current interests are most concentrated on computer statistical analysis by abstract interpretation. He is a member of the development team of ASTREE, a program analyzer aiming at proving automatically the absence of run time errors in programs written in the C programming language that is being used for checking the software of Airbus airplanes. <David.Monniaux@ens.fr>.

**Jean-Baptiste Soufron** has a Master's degree in International Business Law from the *Université de Paris I Panthéon Sorbonne*, France, a Postgraduate degree in Contracts and Business Law from *Université Lille II*, France, and a Postgraduate degree in Intellectual Property and Copyright Law from *Université Strasbourg III CEIPA*, France. He is currently working on his PhD with a scholarship position at the *Centre National de la Recherche Scientifique* (CNRS) and the *University of Paris II Panthéon Assas*. He is also teaching FLOSS (Free/Libre and Open Source Software) principles at the *Université de Paris VII*. He is currently working on several Wikipedia projects as well as many FLOSS projects. He has been practicing Intellectual Property Law for the last four years in Tokyo and in Paris. Additional information can be found at <<http://soufron.free.fr>>. <Jean-Baptiste.Soufron@cersa.org>

<sup>1</sup> Available at <[http://www.wipo.int/treaties/en/ip/wct/trtdocs\\_wo033.html](http://www.wipo.int/treaties/en/ip/wct/trtdocs_wo033.html)>.

<sup>2</sup> Available at <<http://thomas.loc.gov/cgi-bin/query/z?c105:H.R.2281.ENR:>>.

<sup>3</sup> Available at <<http://europa.eu.int/eur-lex/lex/LexUriServ/LexUriServ.do?uri=CELEX:32001L0029:EN:HTML>>.

<sup>4</sup> Available at <<http://www.legifrance.gouv.fr/WAspad/UnTexteDeJorf?numjo=MCCX0300082L>>.

able, machine-processable text), which is then translated (or, in technical terms, compiled) into directly executable code. In most cases, the source code does not express novel techniques; yet, it is an original work, covered by a copyright, in the same way that bad novels without any novel ideas are also covered by a copyright. If the software presents interesting techniques, other software producers may decide to implement the same techniques in their own products, provided that they do not copy the software itself but only the ideas that it implements.

In the industrial world, techniques do not give rise to copyright protection, but people can use patents to control their use. Patents grant the inventor of a novel technical technique an exclusive right over the exploitation of that idea, in exchange for the publication of the idea in the patent document. Whereas previously inventors had to keep their inventions as trade secrets if they did not wish competitors to copy them, patents allow inventors to publish their inventions and to grant licences to other suppliers, thus encouraging the spreading of new ideas and new techniques. Also, the duration of patents is limited, typically to 20 years, which is thought to be a good compromise between the general interest of making inventions free for everyone and providing the inventor with a return on investment, all of which is supposed to encourage innovation.

### 3 Patents on Software

The application of patents to computer software techniques is highly controversial. Some forms of "software patents" are allowed by the US Patent and Trademark Office. Software patenting is forbidden as such by European texts, but the European Patent and Trademark Office (EUIPO) is often described as extending the very narrow exceptions to these rules in order to circumvent that European legislation. Nevertheless, in 1995 the European Parliament refused to generalize the EUIPO practices and to adopt similar provisions to the US ones.

There are manifold reasons for opposing software patents:

- Software 'techniques' often consist of algorithms; that is, mathematical descriptions of computing processes. Traditionally, algorithms have been considered in the same way as mathematical formulas - they are not patentable in themselves, though there are ways to work around this restriction.

- The experience with the administration of "software patents" by the US Patent and Trademark office has been highly controversial. In many cases, patents were granted to techniques already known in the state-of-the-art or immediately derivable by a technical person - both of which are normally precluded by patent rules.

- The pace of the computing industry is very fast compared to other industries. Take, for example, medical suppliers: medicines may be used for decades after they are invented, but patents only grant a monopoly over the first couple decades. Furthermore, the lengthy approval procedures subtract from the "exploitable" patent period. How-

ever, in the case of a software invention, 20 years may often extend beyond its period of usefulness.

- Finally, the economics of software are governed by powerful "network effects". This is important with respects to DRMs, and we shall therefore explain these effects in more detail.

In many industries, the intrinsic qualities of a product are overshadowed by its compatibility with what other users have adopted. For instance, the Betamax format for videotapes ultimately lost out to the VHS format. Many have argued it was technically superior, but VHS was more widespread and the differences were not compelling enough to justify an alternate technology. In computing, there is the constant issue of "compatibility" and "interoperability" - meaning the ability of several software or hardware components to work together.

Historically, it has been often the case that hardware manufacturers introduced gratuitous incompatibilities between their hardware and that of other manufacturers (or even between different hardware lines in their own company) in order to lock customers in : for instance, they produced computers that would only work with their own terminals and printers. Patents over connectors, interfaces, and communication protocols may prevent competitors from manufacturing compatible systems.

The same applies to software, especially in today's wired world. We do not generally buy our software in isolation - because of the Web and e-mailing, for example, we want our word processor, our spreadsheet, etc. to be compatible with those used by our colleagues and friends. In the past, there were a number of competing office suites with significant market shares - but today there is only Microsoft Office and, far less commonly used, its libre competitor OpenOffice.org. With regard to operating systems - the essential software infrastructure on computers -, there are only two widespread desktop systems: Microsoft Windows and Apple's MacOS, though the libre competitor Linux is making some inroads. One reason for this is that for many applications one is more or less forced to use Windows or MacOS. For instance, there are proprietary multimedia (audio and video) formats for which documentation is not freely available and whose designers only supply players for Windows and MacOS. If you wish to play such content, the easy solution is to have Windows or MacOS. The same applies to, let's say, documents supplied in the Microsoft Word format - if you require good compatibility, you have to have a copy of Word, which is available only for Windows and MacOS but not for Linux. Thus there are strong network effects that reinforce quasi-monopolies.

### 4 DMR and Software Patents

Despite these difficulties, some software makers manage to produce 'compatible' software - software capable of reading proprietary formats. They often do so by "reverse engineering" software and file formats for compatibility purposes, which is allowed by current European law. Now,

here's the catch with DRMs. DRMs "technical methods" are simply a new word for "copy protection" systems - systems that prevent unauthorized software from playing or copying content. If you wish to produce software compatible with a DRM 'secure' format, you will essentially have to understand how to work around the DRM system. For instance, if the DRM-encumbered file is encrypted for copy protection purposes, the 'authorized' software will have to decipher it, and so will the 'compatible' software.

The original wording of the DADVSI bill criminalized working around DRM protections, without exceptions. Many people became worried that this law would indirectly allow designers of DRM formats to claim exclusive rights to produce software or hardware capable of reading such formats, since any maker of compatible software or hardware could be accused of circumvention of the DRM, which is a felony. Designers of DRM systems would thus enjoy rights equivalent to those of a patent (exclusivity of production of compatible systems) without the limitations (limited duration and obligation of publication), even though software patents had been refused by elected representatives. Because DRM formats are now used in video, audio, and even text, and because of the "network effect", this might have had severe consequences for competition. For instance, neither Apple nor Microsoft produce players for their popular DRM-encumbered formats for the competitor operating system Linux, and the law could have been used to outlaw any 'compatible' player; thus, Linux would have been unable to play the majority of videos downloadable from the Internet, making it unsuitable for many desktop deployments.

Because of this risk, the French Parliament ended up including provisions that made circumvention of DRMs legal for compatibility and security reasons. We have already explained the compatibility issue; the security issue is justified by the actions of some manufacturers whose "copy protection" software was insecure and even tried to spy on the users (see for instance the Sony *rootkit* scandal). Should the original law have been passed, producers of such systems could have prevented the revelation of their flaws by security researchers by threatening prosecution for circumvention.

But that is not to say that there are no flaws in it. The French Constitutional Court has just released its ruling on the DADVSI bill, and there are two ways it can impede the normal development of licences.

Firstly, the Court eliminated the exception that protected software developers who were specifically working on collaborative software, research or file sharing. Given the decision, any French developers working on such software could be sued by DRM producers or copyright holders, even when its software is intended for non-copyrighted contents. So, regardless of whether people use P2P (Peer-to-Peer) software for a distributed business model or just to share Creative Commons-music, it is already illegal. In this sense, DRM is actually a new form of intellectual property. Hierarchically, DRMs are meant to precede software copyright.

The same mechanism is at work again on another issue. The Court accepted a compulsory licence system whereby DRM developers must grant licences of their DRM Software to other DRM producers. The law creates a DRM Regulation Authority that will centralize problems and propose solutions. But as of today, any Software editor, any DRM producer, or any webservice can ask to access the "essential information" needed for interoperability: if Microsoft wants to interoperate with Apple, they can go to the DRM Authority. And the latest thing is that, if the DRM producer refuses to let the plaintiff access its DRM essential information, the Authority has the power to mandate him to communicate them. In other words, if Apple refuses to let Microsoft interoperate, the Authority can mandate Apple to provide them with the information. So much for their copyright. So much for their patents or their DRM. So much for their ability to refuse and negotiate agreements.

This mainly concerns companies like Apple who could be forced to disclose the functioning details of their software, but it is indicative of the current hierarchical move from copyright to DRMs.

One way or another, software is increasingly perceived as an essential facility. The copyright licensing scheme that has prevailed for 30 years is now being challenged by Software Patents and DRMs. And this new hierarchy is challenged in turn by the need for competition.

The current mechanism has been successful in many ways. It has allowed the parallel development of commercial business models under the exclusive licensing of the right to use a program and access its source code, and the development of more complex and modern *Libre/Free* Software and licensing schemes. The development of software patents has been seen as a threat to these two models from the start. But the development of DRM and the relative discrediting of copyright is just beginning to be perceived as a threat too.

### 5 The Need for A Serious Debate

There is certainly a serious debate to be had on the opportunity of allowing "software patents", and, if they are allowed, under what conditions. There is also a debate to be had on the business model of the entertainment industry in the Internet age. But, copyright for music and films should not be used as an excuse to introduce harmful *de facto* patents in other industries, especially when this "new kind of intellectual property" would clearly hamper competition and innovation.