free software report

Volume 2, Number 1

Free Software And The Law

By Michael Tiemann

Legal review is an inevitable process in virtually any commercial activity; the use of software is no exception. While public domain and freely redistributable software have much more liberal copying terms than proprietary software, these terms are often negated by lawyers who are uncomfortable with the implications of using free software.

It is the thankless task of the lawyer to find a way to make any legal problem somebody else's responsibility. Many lawyers are simply unaware of how to legally pass the buck when it comes to free software. This article will address the most common legal issues surrounding free software and hopefully dispel some common misconceptions.

In this issue:

There are three main classes of freely redistributable software: public domain (PD) software, "MIT" or "Berkeley" copyright software, and software covered by the GNU General Public License (GPL) and its derivatives. We will first compare the legal issues unique to each of these types, and then legal issues that are common to all.

Public domain software is by far the simplest and most complex case. First of all, public domain software is not copyrighted (by definition), so there is no way to violate a license by using it, copying it, modifying it, or redistributing it. It is in this seeming simplicity that the lawyer's troubles begin. When somebody claims copyright on a piece of software, the lawyer can satisfy the client and/or the owner that the intended use of the software is consistent with the owner's copyright terms. PD software does not have a clear author or owner, and thus makes lawyers nervous; their legal advice is the last link in the chain.

Much of the PD software available today is written by the United States Government. Such software often carries a large, official-looking banner stating the laws which preclude the federal government from claiming copyright. Companies can make a business by copyrighting PD software and selling their assurance that there won't be a problem. This is in fact very similar to what title insurance companies do with property titles.

The so-called "MIT" and "Berkeley" copyrights are an extension of copyright terms on research papers. The "MIT" copyright became well-known with the distribution of the X Window system, the "Berkeley" copyright with the distribution of

Continued on page 2

Free Software And The Law

Continued from page 1

freely distributable utilities and libraries for BSD UNIX (the basis of Sun's Unix operating system). Most universities offer similar terms: the ability to use, distribute, and modify the sources of the software, provided that the university is not used in advertising or represented as being responsible for any aspect of the software other than owning the copyright.

The software products of these (and other universities) are rightly treated as research, and they are rightly distributed with copyright terms that allow others to extend their work. Some universities (notably Stanford University) has begun to experiment with more restrictive licensing (limitations on commercial exploitation) in hopes of selling rights to work it develops, but this strategy appears to be backfiring. Most universities are supported by a number of commercial, federal, and state programs; by effectively competing with these sponsors, these universities will drive away potential funding and be forced into the decision of turning back to traditional academic research or becoming a full-fledged commercial operation.

The GNU General Public License (GPL) is perhaps the most interesting copyright of all. It states that one may use, distribute, or modify the source (and resulting binaries) with only one restriction: any recipient of the

Continued on page 6

E-Mu Systems And Free Software:

Making Beautiful Noise

By Terri Thatcher

Genesis, Yes, Todd Rundgren, and Prince. Besides huge success, what do all of these bands have in common? E-mu Systems, Inc.

E-mu (Electronic Music) is a company specializing in the design and creation of synthesizers and keyboards used by musicians around the world. Accompanying these highly specialized instruments is E-mu's customized software. This software is an embedded application which ultimately controls sound quality and acoustics. This software was itself created using free software: the GNU C compiler. The software can only be used on E-mu's instruments.

E-mu Systems, Inc. has also been credited with the introduction of digital sampling. Digital sampling converts analogue waves to digital bits which in turn control the pitch, volume and other acoustic parameters by applying digital signal processing (dsp). Their customized dsp chips generate the sound in the instruments. E-mu's specially designed software controls the chips in a musically useful way making it customizable and programmable.

Just one of thousands of companies who have discovered the value in using free software, E-mu was faced with an engineering problem in developing their software and found a successful solution in free software.

The following is an interview with Niel Warren, a senior design engineer at E-mu.

Q: What host/target platforms are you developing on?

A: Hosts are Sun4, SGI Indigo, Sun3, Mac, PC. Targets are Sun 4, PC. Embedded targets are m68000, ns32k, 56000, and 2105.

Q: What proprietary software development tools are/were you using?

A: We were using the Intermetrics compiler.

Q: How long have you used a proprietary development solution?

A: We used a proprietary solution for right around 15 years.

Q: What was the quality of service and support you came to expect from the proprietary software vendor?

A: After a time, we came to expect virtually none.

Q: Did their support and services help or hinder your development productivity?

A: Without proper support, our development efforts were hindered.

Q: On how many host/target platform combinations were you able to use your proprietary development solution?

A: Only one.

Q: What prompted you to re-evaluate using a proprietary development solution?

A: We bought newer, faster computers and could not face paying more money to a an unresponsive company with little, if any, support and a product whose value was questionable.

Continued on page 7

Intellectual Property On The Global Net

By John Perry Barlow

Throughout the history of copyrights and patents, the proprietary interest of thinkers in their ideas has been focused on that point when those ideas were manifested in some physical object, whether book or invention.

In practice, legal protection was extended less to the expression of the ideas than to the tangible conveyance of that expression...less to the wine than the bottle.

Now, as information enters Cyberspace, the native home of mind, these bottles are vanishing. Expression goes on-line and stays there. Increasingly, the goods of the Information Age will exist either as pure thought or something very much like thought: voltage conditions darting around the Net at the speed of light. They will not, at any point, enter a container, whether a sheet of paper or a CD-ROM disk, which one could touch.

It may well be that, without our old methods of physically containing the expression of ideas, we simply don't know how to assure payment for the work we do with our minds. To make matters trickier, this comes at a time when the human mind is replacing sunlight and mineral deposits as the principle source of new wealth.

We are developing an economy in which the primary items of commerce can be infinitely reproduced and instantaneously transferred around the planet, at little or no production cost, with or without the consent of its originator or owner, from whom it might have been stolen without leaving his possession.

Lacking unauthorized volumes to embargo or bootleg pressings to seize, it will be very difficult to enforce existing copyright laws without placing in peril the ultimate source of intellectual property, the free exchange of ideas.

"It is likely that new models of intellectual property will look more like performance than possession."

That is, when the primary article of exchange in an economy looks so much like speech as to be indistinguishable from it, and when the traditional methods of protecting ownership of that article are increasingly irrelevant, increasing the vigor of intellectual property enforcement will inevitably imperil freedom of speech.

Increasingly, the principle threat to our future liberties will not be our government but will more likely be corporate legal departments laboring to protect by force what can no longer be protected by practical efficiency or general social consent.

Further complicating the matter is the fact that along with the physical bottles in which intellectual property protection has resided, digital technology is also erasing the legal jurisdictions of the physical world, and replacing them with the unbounded and still largely lawless seas of Cyberspace.

In Cyberspace, not only are there no national or local boundaries to contain the scene of a crime and determine the method of its prosecution, there are no clear cultural agreements on what a crime might be.

Notions of property, value, ownership, and the nature of wealth itself are about to change more fundamentally than at any time since the Sumerians first inscribed clay tables with cuneiform representations of stored grain. At this point, only a very few people are aware of the enormity of this impending shift.

It is incumbent upon those of us who are considering these questions to begin preparing responses for the legal and social confusion which will erupt as efforts to protect new forms of property with old methods become more obviously futile and, as a consequence, more adamant.

Humanity seems bent on creating a world economy primarily based on these immaterial goods, but without some predictable connection between their originators and a fair reward for the utility or pleasure others may find in them, we are building our future in furor, litigation, and institutionalized evasion of payment except in response to raw force. It will be a return to the Bad Old Days of property.

Continued on page 4

Did Y ou Know?

The Open Software Foundation uses the GNU Language Tools for all of their in-house software development.

Intellectual Property On The Global Net

Continued from page 3

Throughout the darker parts of human history, the possession and distribution of property was largely a military matter. "Ownership" was assured those with the nastiest tools, whether fists or armies, and the most resolute will to use them. Property was the divine right of thugs.

This continued to be the case as long as the origin of wealth was agricultural, but with the dawning of the Industrial Revolution, humanity began to focus as much on means as ends. Tools acquired a new social value and, thanks to their own development, it became possible to duplicate and distribute them in quantity. To encourage their invention, copyright and patent law were developed in most western countries to assure their inventors compensation for the value of their use by others.

In both cases, patent and copyright, the original protection of law was extended to the expression of an idea and not to the idea itself. In the west, as in the east, ideas and facts about the nature of things were still considered to be the collective property of humanity.

A patent, until recently, was literally a description of the form into which materials were to be rendered in the service of some purpose. If the object which resulted from this rendering wasn't entirely new to the world or if rendering were impossible due to some material limitation, the patent was rejected.

Thus the right of invention and authorship adhered to an act or series of actions in the physical world. One didn't get paid for his ideas but for the ability to execute them.

Now all of these constraints are gone.

Did Y ou Know?

Linux is a freely redistributable multi-user multi-tasking operating system that runs on standard 386/486 PC clones and laptops.

Since it is now possible to convey ideas from one mind to another without ever making them physical, we are now claiming to own ideas themselves and not merely their expression. And since it is likewise now possible to create useful tools which never take physical form, we have taken to patent processes, sequences of events, and mathematical formulae.

This constitutes a fundamental revision of the entire body of existing intellectual property law. Everything is up for grabs and once again property adheres to those who can muster the largest armies. The only difference is that this time the armies consist of lawyers.

What is to be done? While there is a certain grim fun to be had in it, dancing on the grave of copyright and patent law will solve little. Ideally, the best thing to do might be nothing at all. In a more perfect world, we might do well to declare a moratorium on both litigation and legislation until we have a clearer sense of the terms and conditions of enterprise in Cyberspace.

One can already see in development new unwritten codes to suit commerce in the immaterial world. For example, while the law requires that every piece of commercial software be paid for, the unenforcability of this requirement has resulted in a generally accepted practice of pirating software to try out before buying it. Most software is now being purchased in the service of ethics or the practical awareness that the failure to buy it will result in its not being produced any longer.

Thus, time assumes the role which space has held in determining value. The value of physical products often resides in their scarcity, or limitation in space. An informational product is generally more valuable the closer the purchaser can place himself to the moment of its expression, a limitation in time.

It is likely that new models of intellectual property will look more like performance than possession. Commercial exchange will be more like ticket sales to a continuous show than the purchase of discrete bundles of that which is being shown.

We will return to something like an oral culture in which there is never a final cut to sell but only a process to place oneself as deeply in the middle of as one can afford.

At last commerce will consist of verbs, not nouns.

John Perry Barlow is a retired Wyoming cattle rancher, a lyricist for the Grateful Dead, and co-founder and Executive Chairman of the Electronic Frontier Foundation.

He is a writer and lecturer on subjects relating to digital technology and society and is a contributing editor to numerous publications, including *Communications of the ACM, MicroTimes, Mondo 2000,* and *Wired.* He is a recognized authority on the social and legal conditions arising in the global network of connected computers.

Who W rites Free Software, And Why?

By Terri Thatcher

The free software movement stems far beyond the organizations we have all become accustomed to hearing about, such as The Free Software Foundation. In addition to organizations such as this, several individuals and companies have taken on free software and incorporated it into a part of their lives. Some have chosen to use free software as the catalyst in their efforts to earn a living while others create and maintain lists of free software on their own time.

Ajay Shah, a Ph. D. student of economics at University of Southern California in Los Angeles, has an enduring interest in Unix programming. He is one of those who has dedicated a portion of his personal time to the maintenance of one of the hundreds of lists of free software available on the Internet.

Shah is the official maintainer of *The Index For Free Source Code For Numerical Computation In C or C++*. This particular list serves as a clearing house for information in the emerging area of FORTRAN-less numerical computation.The list currently includes over 80 software programs designed to appeal to the serious mathematician or to one dabbling in computation as a hobby. When people want something, they simply look it up. It also helps authors get in touch with each other and helps eliminate replication of work that has already been done on individual software programs.

"...why do I do it? Because I think it's useful, and constitutes a good service to the world at a very low cost to me in time and trouble."

Did Y ou Know?

Hitachi America, Ltd. ships a complete set of development tools which includes the GNU C compiler (GCC) and the GNU source-level Debugger (GDB) for the H8/300 family of embedded controllers.

In Shah's own words, "Why do I do it? Because I think it's useful, and constitutes a good service to the world at a very low cost to me in time and trouble." According to Shah, the world of numerical computation is in for a major shift away from FORTRAN and moving towards C and C++. "The popular perception is that good free libraries for numerical computation in C or C++ don't exist." Shah's efforts are putting a damper on that myth.

Shah is quick to point out that his list is certainly not the only such "informative service" available on the net. Traditionally people have looked to the net for source, possibly without realizing the availability of vast amounts of free lists of this sort. Lists are constantly being kept up to date on a myriad of different topics.

The following are interesting examples of some of the types of programs that have made it into Shah's index:

- **kalman.zoo** - A class library for Kalman Filtering.
- **Cfortran** - A set of macros (cfortran.h = 1000 lines) allowing function calls to be made from C to FORTRAN and vica-versa.
- **frac** - Finds rational approximation to floating point value.
- **Octave** - Interactive system for numerical computations, like matlab.

For more information on how you can get access to this list: Send email to `netlib@ornl.gov` saying "send free-c from c." Or, `ftp` into `research.att.com`, `cd` into `netlib/c`, get `free-c`. It's that easy. If you don't have `ftp` access, send email to `ftpmail@-decwrl.dec.com` saying "help". You will get instructions on how to `ftp` via email.

June, 1992

New Y ork University announced that they signed a contract with the Ada 9x Project Office to develop a full Ada 9x compilation system, available at no cost and with source code, as part of the Free Software Foundation's distribution known as the GNU C Compiler - GCC. The project is cosponsored by DARP A and the AJPO (Ada Joint Program Office); two phased releases are expected during 1993.

Free Software
And The Law

Continued from page 2

sources (or derivative works) must also be given these same freedoms. In short, the GPL places a restriction on the restrictions others can impose, and the restriction ensures that any party to the development or distribution of software covered by the GPL has the same freedoms as any other party.

The MIT and Berkeley copyrights make it possible to develop derivative programs that can be made proprietary programs (the X servers from DEC and HP bear this out, as does the Ingres database program that was developed at UC Berkeley). GNU software itself cannot be co-opted in this fashion, but service, support, and packaging of GNU software are all viable commercial endeavors.

Many people have misinterpreted the GPL. Some people think that any use of GNU software implies that all software at their site must then be provided in source form to anybody who asks. This is totally false: the use of Emacs to edit a document does not cause the document to fall under the GPL; the use of GDB to debug a program does not cause that program to fall under the GPL; the use of GNU compilers to compile an application does not cause that application to fall under the GPL.

Copyright law defines a "derivative work of authorship" in such a way that when a program is linked together with a library to create a single work, the work may be governed by the copyright of the library. For this reason, one must fully understand the implications of linking libraries (or software cov-

Continued on page 7

Did Y ou Know?

Many MS-DOS networking packages ship a free software version of a highly portable packet driver from Crynwr Software with their products.

Correction from issue 1.1 of the Free Software Report. In Rich Morin's article, "UNIX Freeware is Different" the GhostScript suite is written in C, not C++. Groff is written in C++.

We want to hear from you!. Please send your inquiries to the editor:
`fsr@cygnus.com`
or send letters to:

Free Software Report
1937 Landings Drive
Mountain View, CA
94043

The Fine Print

More Reading On Free Software and Related Topics

"Wanted: Free Software"
SunExpert Magazine
December 1992

"Lawyers"
Computer Language
November 1992

"Help Writing `grap` Programs"
UnixWorld
September 1992

"Archie, Your Directory for Internet Software"
UnixWorld
September 1992

"Software Control"
SunWorld
February 1993

"Talking Tech with Your Lawyers"
Corporate Computing
November 1992

"Lawyers Stand in the way of Open Systems"
The Sun Observer
February 1993

"The Internet Notebook"
Unix Review
February 1993

E-Mu Systems

Continued from page 2

Q: What prompted you to consider the combination of Cygnus Support and GNU development tools?

A: Our team liked the GNU toolchain and had been using and supporting it in-house for quite some time. Contracting with Cygnus for support freed up much of our developer's time. They were able to use the tools rather than maintain them.

Q: How does support from Cygnus compare to the support you received from your proprietary software vendor?

A: Cygnus provides comprehensive customer support. They cover anything from hand-holding to specific instruction. The tools tend to be less stable than proprietary tools due to their evolutionary nature. The combination of the two make for a completely scalable development solution.

Q: How does the free software model benefit your development team?

A: GNU development tools run on a number of platforms. At E-Mu we do a lot of cross development and embedded development. The great portability of GNU allows us to successfully do embedded applications using only one compiler. Using the GNU compiler on our new machines, a total rebuild went from nearly one hour to under two minutes.

Q: Where does free software fit into your future development solutions?

A: We will definitely continue to use it. The combination of free software and support from Cygnus make the most sense for us. Not only are the GNU tools and compilers the best available, the combination is a scalable solution, saving us time and money.

Free Software And The Law

Continued from page 6

ered by the GPL) into an non-freely redistributable application.

To fail to make the distinction between activities which do or do not create derivative works of authorship leads to decisions that can be very costly, either because of lost opportunity or because of other legal implications.

Beyond the copyright issues, there are issues of patent infringement (which are a very real problem today for anybody writing original software) and restrictions imposed by the Dept. of State (which has the authority to classify software as a munition and therefore restrict its export).

50 years ago, patents and copyrights assured inventors and authors that their innovative ideas and expressions would not be misappropriated (stolen). Today, programmers routinely reinvent previously patented techniques as a matter of daily work because patent examiners did not have the competence to understand prior art, obviousness, and/or the distinction between an algorithm and a process in the software domain. The League for Programming Freedom has a list of over 50 "obvious" software patents which make for sobering reading. While preparing this article, a new one was found for the list: a patent on what amounts to client-server computing. The next time you write an application that runs over a network, you may be at risk.

For more information on Public Domain Software, software covered by the GPL or "MIT/Berkeley" software, please send email to fsr@cygnus.com.

Access Information

Crynwr Software
11 Grant Street
Potsdam, New York 13676
315 268 1925 voice
315 268 9201 fax
nelson@crynwr.com

Cygnus Support
1937 Landings Drive
Mountain View, CA 94043
+1 415 903 1400 voice
+1 415 903 0122 fax
info@cygnus.com

E-mu Systems, Inc.
1600 Green Hills Road
Scotts Valley, CA 95067-0015
408 438 1921 voice
408 438 8612 fax

Free Software Foundation
675 Massachusetts Avenue
Cambridge, MA 02139 USA
+1 617 876 3296
gnu@prep.ai.mit.edu

Free Software Report
1937 Landings Drive
Mountain View, CA 94043
+1 415 903 1400 voice
+1 415 903 0122 fax
fsr@cygnus.com

Hitachi America, Ltd.
Semiconductor & I.C. Division
2000 Sierra Point Parkway
Brisbane, CA 94005-1819
415 589 8300 voice
415 244 7472 fax

Linux - Access Information
In the United States:
tsx-11.mit.edu (18.172.1.2)
directory /pub/Linux
In Europe:
nic.funet.fi (128.214.6.100)
directory /pub/OS/Linux
To acquire updated information:
email: torvalds@kruuna.helsinki.fi

Micro Station Technology
1140 Kentwood Avenue
Cupertino, CA 95014-5808
408 253 3898 voice
408 253 7853 fax
jms@mst.com

Open Software Foundation
11 Cambridge Center
Cambridge, MA 02142
617-621-8700 voice
617-621-0631 fax
romeo@osf.org

The Free Software Report is published quarterly. A 4-issue subscription is available for $60.00 payable in U.S. Dollars through a U.S. bank. Send a check or money order to: **The Free Software Report, 1937 Landings Drive, Mountain View, CA 94043, USA**

Please supply your name and title, company name, and address for receiving the Free Software Report. The most recent and subsequent issues will be mailed as soon as we receive your order.

Name____________________ Title____________________

Company or Organization______________________________

Address____

City________________ State__________ Postcode__________

Country________________ Amount Included: $______________

Comments ____

____

Email__________________________

The Free Software Report
1937 Landings Drive
Mountain View, CA 94043

BULK RATE
U.S. POSTAGE
PAID
MOUNTAIN VIEW, CA
PERMIT NO. 521