

Why Free Software Happens: A Survey of Motivation in the Open Software Community.

Nicholas J Bailey*

July 26, 2001

Abstract

This paper asks the question “What examples of achievement exist where synergy is an important factor, and what motivates the parties to work in this way?” The area of excellence examined is the Free Software movement. Its origins are examined, and examples of notable free software projects are given. Some common misrepresentations relating to this environment are addressed. Finally the types of software licence commonly used by the Open Source and Free Software communities are summarised.

1 Reason for the Open-Source Synergy Theme

Effective dissemination of software and other technologies arising from the kind of projects which the Circus forum addresses is an important part of building an infrastructure upon which artistic and scientific synergies is built. The normal route for dissemination of materials arising from scientific and engineering endeavour is to attempt to market them. However, this may be an inappropriate course of action for the kind of deliverables likely to arise from the Circus-related projects:

- Commercialisation makes sense when the product is monolithic. If, as would probably be the case with Circus project results, the potential use requires not merely to reproduce results exactly but to extend, modify and contribute, the issue of rights ownership becomes unresolvable.
- Notwithstanding the above, it is possible that tools might be made available which artistic users might wish to use. In the event of commercialisation and closure of access to source codes, the

copyright owner is essentially exercising an option to control the activities of the creator. Traditionally, the creator will accept these terms where s/he is unaware of alternative solutions, or no such solutions exist. This is contrary to proper dissemination of artistic works, and instead creates a dependency culture where users expect product support without synergistic input to its evolution.

- The only successful and proven method of permitting users free access to software is that adopted by the Free Software Foundation. The GPL (GNU Public Licence) requires that the source code be made available to all users on demand. It also places a strong copyright on the source code and thus protects the rights of the original authors and contributors while ensuring their work reaches the widest public without any marketing activity.

2 The Origins of the Free Software Movement

The cost of computing hardware used to be significant. Acquisition and maintenance of early mainframe computers was so expensive its cost dwarfed the expense of writing the rather limited software they were capable of running. Even when the complexity of software began to grow, software piracy was not a big issue: the party acquiring the software package would have to purchase a machine on which to run it, and the hardware vendors would recoup their costs.

This environment gave rise amongst software engineers and system administrators of the time to what is

now often referred to rather disparagingly as the “Play Ethic”. Code was frequently shared and solutions to common problems arrived at cooperatively and openly. Pat Kane’s article *Second Sight*¹ which appeared in the Guardian newspaper on March 29th 2001 starts by citing the open-source community and concludes that the play ethic is responsible for much of the general success of the Finnish society and economy.

As corporations began to tighten down on information about their products, citing the protection of intellectual property as their reason for doing so, coopera-

* Centre for Music Technology, The University of Glasgow

¹ <http://www.guardian.co.uk/Archive/Article/0,4273,4161088,00.html>

tion between different groups of programmers became increasingly difficult. It became de rigeur to require non-disclosure agreements before any software source was exchanged, if indeed it was exchanged at all. A group activists formed themselves under the title of the Free Software Foundation.² The FSF works to abolish the closed mentality of commercial institutions by lobbying and by producing its own freely redistributable software.

It is because of the spirit of openness and cooperation between the different communities of users and programmers who make up the open-source movement that it is instructive for the CIRCUS member to be aware of their motivation and goals. Central to the play ethic and to much of the free software and open-source movements is the pursuit of excellence as an end in itself; a motivation surely more often associated in the popular mind with the arts rather than commercial enterprise. In fact, there is a range of motivations within the open-software community, and these are expressed in the various copyrights and licences which have arisen governing the use and distribution of “free” software.

Before examining them, we will survey some of the

counter-attacks often launched at the open-source community which commerce and industry often regards as hostile. However, it is important before doing so to define what exactly is meant by “free” software. The FSF’s Richard Stallman is at pains to distinguish “free” in the sense of “freedom” from “free” in the sense of “costing nothing”. The FSF, he argues is about freedom, and but in fact encourages those adopting its principles to charge as much as they can for their work! His slogan “Free as in speech, rather than free as in beer” is a concise reminder of the difference.

Dr Stallman is also at pains to point out the difference between the free software movement and open source movement. Members of the free software movement believe in total freedom in the application and distribution of source code, and have generated strong legal frameworks to ensure that these freedoms are maintained. Members of the open-source movement believe that the open source idea is a good one because it yields robust and reusable code, but do not focus on the ethical and political issues of open software. Both, however, are members of the same open-software (or free-software) community.

3 Myths

3.1 Free Projects are Always Small Scale

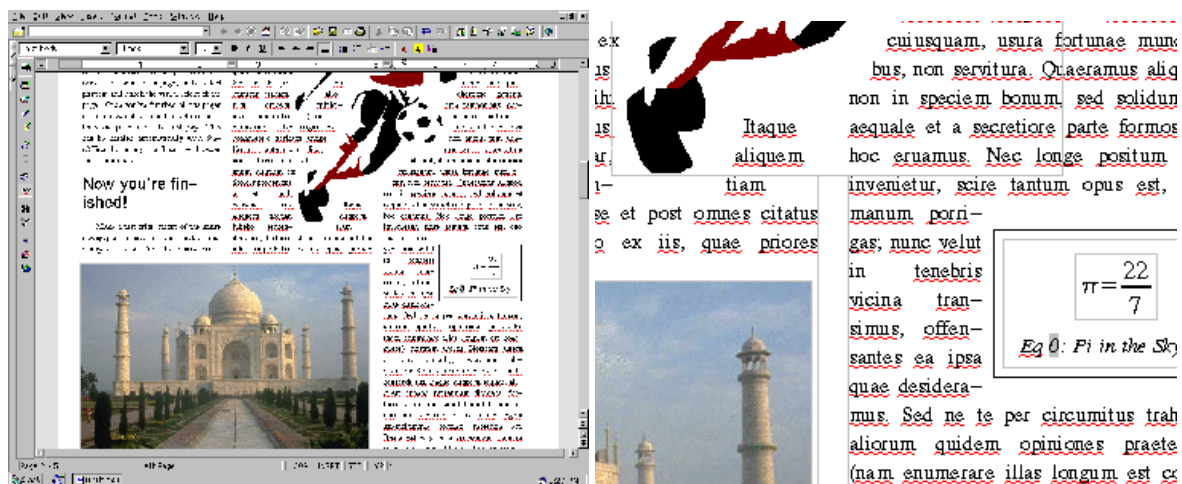


Figure 1: Large-scale Free Software

The open-source community has developed systems which foster cooperative authorship and code maintenance. The machine on which this article was written runs Debian GNU/Linux³ (Linux is the kernel; the rest of the packaged system is largely the work of the Free Software Foundation’s GNU⁴ project, although this is, unreasonably, rarely acknowledged) and

has thousands of installed packages with a total of more than 105000 managed files.

The screen shots in Figure 1 show a session from Sun’s StarOffice⁵ package (overall screen-shot and detail) showing the word processor package flowing text around shaped images and a formula. Sun has recently placed the source code for the office package under the

²<http://www.fsf.org/>

³<http://www.debian.org/>

⁴<http://www.gnu.org>

⁵<http://www.sun.com/staroffice/>

FSF's licence. It continues to sell packaged versions of the software, and support.

This myth may have arisen because the activities of the free-software community are more often than not tool-based. It is considered more useful to write a col-

lection of small programs with well-documented interfaces than to deliver a monolithic system. By exposing the interface at each level, the whole becomes more flexible and general.

3.2 Free Software is Unreliable

Examples of open-source software in high-reliability and critical applications:

- GNU/Linux is used on board the space shuttle;
- The apache web server is the most popular, robust and secure available;
- Almost all of the world's email is routed by sendmail or exim, both of which are free;
- TeX and LaTeX⁶ are the most reliable document typesetting packages currently available: Knuth offers a prize for the discovery of a bug in the TeX code, and now only changed the version number by allowing it to approach π

```
nick@haydn: ~ > tex
This is TeX, Version 3.14159
(Web2C 7.3.1)
**
```

The image shows a screen-shot of the KDE's Konqueror web browser and file manager. Officially still under development, this browser has shown itself more stable than commercial software like Netscape's "Navigator". As of Version 6, Netscape has adopted an open-source policy for development. However, the open-source community has taken some time in adopting this project, and the current versions seem even less stable than the previous commercial product (version 4).

From the CiRCUS perspective, we observe how opening the user-base by encouraging the use of software through open-source methods encourages the community to test and improve the product. The benefits of the cooperative approach do not cease after a work has been made public, but continue as the product is used. By analogy, a freely accessible framework for the dissemination, use of, and participation in artistic works is equally as important as support for the generation of such works in the first place.

3.3 Open-source projects lack proper documentation and are unmaintainable

This myth arises from commercial sources which perceive a threat, real or otherwise, from the open-source community. Their assertion is that rigorous software engineering practice is not used outside of industry.

Experience denies this. Having worked in the software industry and supervised students placed in the in-house software divisions of some very large companies, I have yet to come across a single example of a commercial development which uses a rigorous code versioning system or formal documentation. Anecdotally, there seems to be two reasons for this: the management of software projects often have very limited experience in Computer Science, maybe coming instead from an Engineering or Physics background, and are ignorant of formal development methods; or code is developed by consultants who, in order to ensure future employment, keep supplied documentation to a minimum. There is some poetic justice in the fact that the protagonists of closed-source software are thus one of its first victims!

The image shows a detail from a screen-shot of a GUI CVS front end⁷ designed to operate with the GNU

CVS. GNU CVS is a full-featured code versioning system accommodating development branches and code roll-back. It is used at SourceForge⁸ – a web-fronted repository for open-source projects run by VALinux inc.⁹ For most large projects and many smaller ones, a rigorous to documentation and code versioning is a matter of pride to the developers.

From the CiRCUS perspective, we observe that the motivation to produce works of quality arises from the creator, not from a market-driven competition ethos. Many participators in open-source development regard competition as wasteful, because it implies duplication and that the majority of the work produced be discarded. There are no quality benefits in a market-based system unless the consumer is both fully aware of the requirements, and is competent to judge the products. In the case of artistic output, any innovative work must *ipso facto* be outside of the participants prior awareness, and in the case of closed, complex works or products, the consumer can neither be fully aware of its specification/inspiration nor expected immediately to understand it.

⁶<http://www.ctan.org>

⁷<http://cervisia.sourceforge.net>

⁸<http://sourceforge.net/>

⁹<http://www.valinux.com>

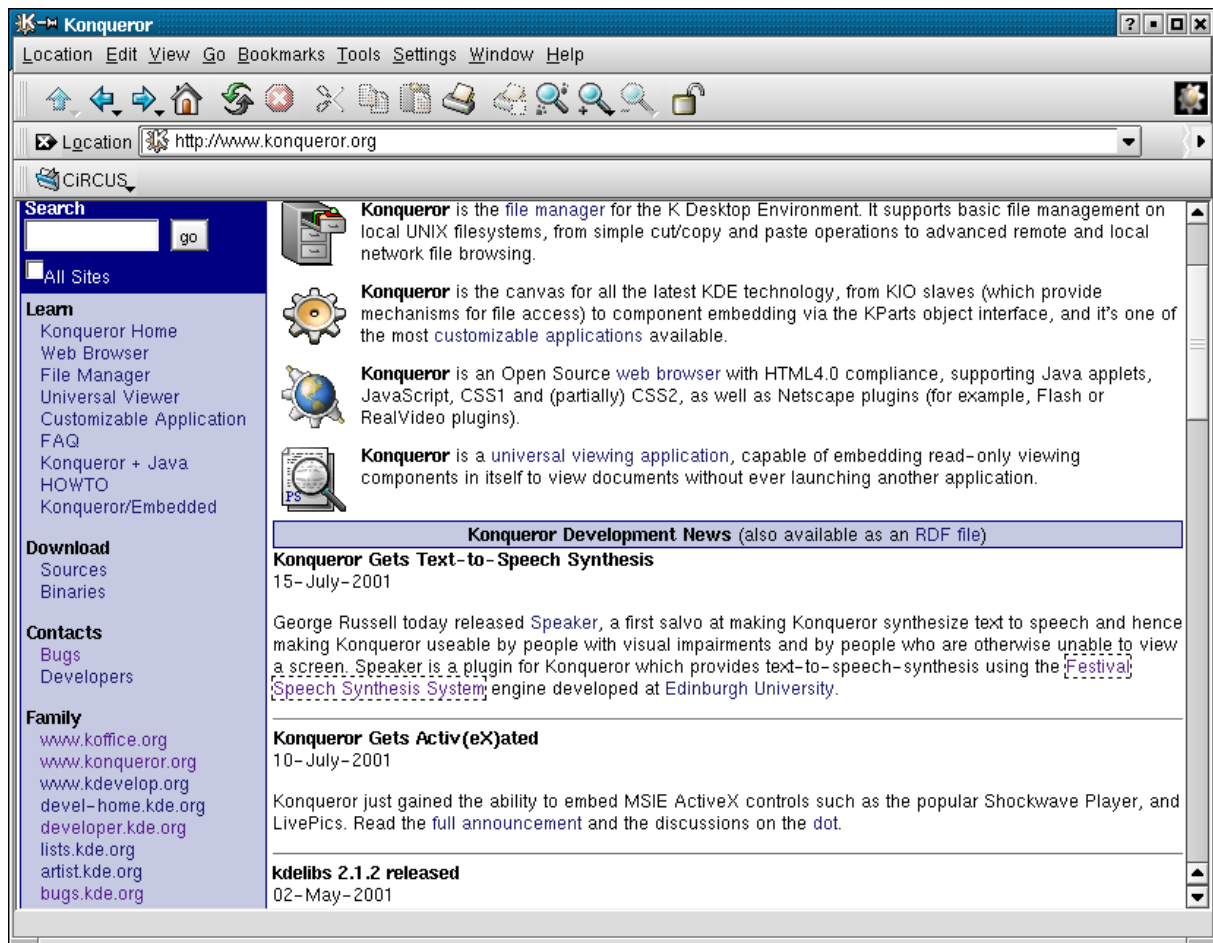


Figure 2: The Konqueror Web Browser

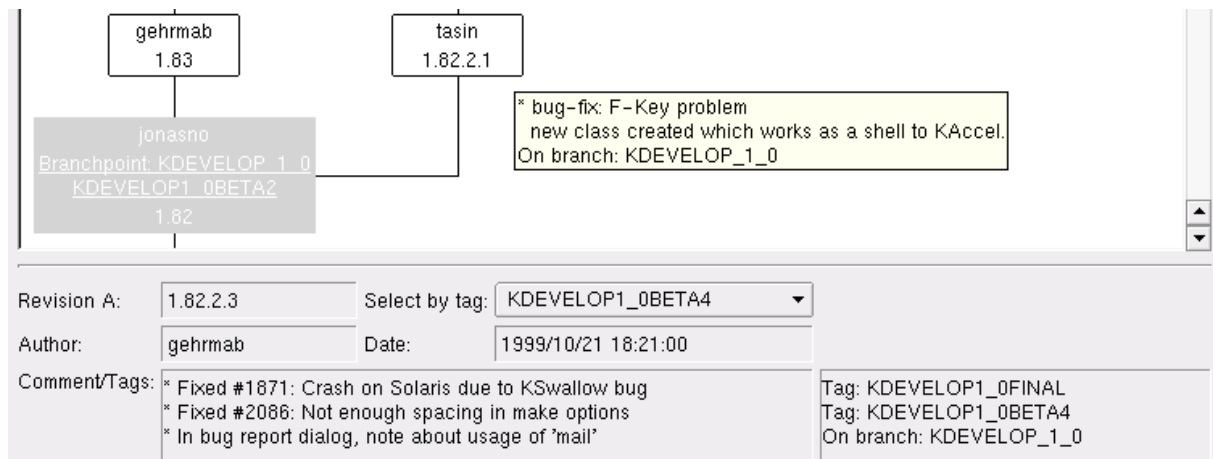


Figure 3: The Cervisia Code Versioning System Front End in Operation

3.4 You can't develop open source products because with no sales, they cannot be supported

There is no restriction of selling either open-source products or support for them.

4 Licences

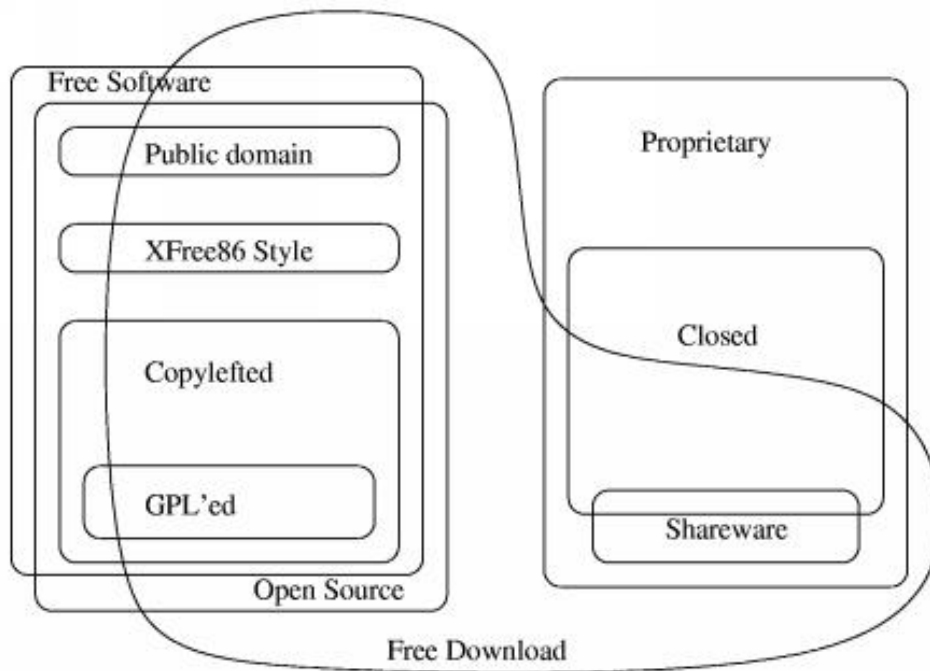


Figure 4: A Taxonomy of Software Licences

The diagram in Figure 4 was drawn by Chao-Kuei and appears on the FSF's Categories of Free and Non-Free Software page.¹⁰ It represents a taxonomy of open- and closed-source software.

The Free Software Foundation's motives are to promote the freedom of others to use software as they please. This motivation has been at the core of the FSF's philosophy since its foundation, but has achieved particular relevance recently by various highly restrictive practices pursued by some hardware and software manufacturers. Notably, some hardware manufacturers (e.g. those manufacturing "WinModems" and some sound cards) began releasing drivers only for Microsoft Windows. Naturally, there is nothing wrong in that, as it is unreasonable to demand they support operating systems used by numbers sufficiently small that they consider the investment unjustifiable. However, the same manufacturers would place restrictive clauses on the use of their hardware, or simply make secret the necessary information for a third party to write software to support their product. It is not clear why they should choose to do this. One has to ask, "will more sound cards or less be sold as a result of increasing

the potential user group?". Such reasoning seems to be beyond the grasp of some corporations, however, and there are still many products, not just from the personal computing arena, which have the same restrictions. For example, Manufacturers of Field-Programmable Gate Arrays (FPGAs - devices contain up to several millions of logic gates which can be wired up entirely through software) stalwartly refuse to issue details of the configuration file formats, thus stunting research using such devices in dynamically reconfigurable computing.

From the FSF's advocacy arose the idea that not only was it stupid to apply these restrictions to software, but that it is wrong morally to do so. Their argument is that there can be no such thing as software theft, because "theft" means "permanently to deprive the owner of use" and copying a piece of software, while infringing copyright, does not stop the owner from using the legally copied software. The FSF developed the concept of a legally binding copyright (referred to as "copyleft") which was designed not to restrict the use of software but protect the freedom of its users. The General Public Licence (GPL¹¹) permits the work to which it is applied to be distributed freely (in

¹⁰<http://www.fsf.org/philosophy/categories.html>

¹¹<http://www.fsf.org/copyleft/gpl.html>

the sense of “free speech”) with or without charge, but is radical in that it operates virally. That is to say, it “infects” derived works which must also be freely redistributable, and requires anybody who uses the derived work to make available its source code under the GPL. In doing so, by providing tools and programs without charge which are of very high quality, the quantity of GPL software has now reached critical mass and is increasing fast.

There exist open-source non-proprietary software licences which are nonetheless incompatible with the GPL. An excellent survey of licences¹² is available from the FSF’s web site. Some authors wish their work to be freely available only to a specific community or for a specific purpose; while this may constitute open-source software in that the source code is made available, it can not be regarded as free software in the FSF sense because a restriction is placed on the freedom of the user to use the software as she or he chooses. Some “shareware” packages fall into this domain.

Of particular interest to the CiRCUS community is the effect of releasing a major piece of software into two different user communities, and watching the behaviour of the groups involved. Dr Mark Spink, formerly a lecturer in the Department of Electronic and Electrical Engineering at Leeds, UK, wrote a significant package to permit the handling of NURBS (Non-Uniform Rational B-Splines). The package was released under the GPL to users of Scilab,¹³ a GPL’ed scientific computation package, and Matlab, a similar but commercial package. Users from both communities sent email providing comment, feedback and bugs reports. However, there was a fundamentally different attitude displayed by the two groups. Those from the Scilab camp were more inclined to suggest solutions to bugs rather than merely report the facts, and to contribute code back for new features. This is quite interesting because although Matlab is not an open source package like Scilab, both communities of users do write extensions called toolboxes which very often include source code. Dr Spink personally believes that the attitude stems from a different perception of the code ownership. In the open source community for which

Scilab can be included, code is seen as a community commodity with a principle author (leader) where suggesting patches and contributing code is not seen as giving good ideas and work away. However, in the closed proprietary world within which Matlab has been developed, ideas and work are seen more as owned by the author, and they can fix their own problems.

The Free Software Foundation has also been involved in the production of the Free Documentation Licence¹⁴ which is used by at least one large-scale commercial project¹⁵. This is possibly of interest to circus members in that it covers documentary content which is closer to some members’ interests than is software. From the preamble:

The purpose of this License is to make a manual, textbook, or other written document “free” in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or non-commercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of “copyleft”, which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

5 Arts and Free Processes

Traditional models of artistic activity are not necessarily compatible with the ideas of interaction. In the conventional process, a composer (or author) produces a work and releases it. Commercial considerations aside, it is not clear why the composer should be entitled to any influence over the work after it has been released, and indeed, the very connotation of the word suggests a freedom and individual existence¹⁶ for the

new work of art.

Performers interpret the released work. Most likely, they will be interested in the composer’s motivations and intentions beyond those literally expressed in the work, and will listen to the composer’s opinions, but their interpretation determines the nature of the performance.

¹²<http://www.fsf.org/philosophy/license-list.html>

¹³<http://www-rocq.inria.fr/scilab/>

¹⁴<http://www.fsf.org/philosophy/license-list.html#DocumentationLicenses>

¹⁵The free encyclopædia project, <http://www.nupedia.com>

¹⁶UK’s Channel 4 recently produced a series, *Beckett on Film*, in spite of the fact that during his lifetime, Beckett consistently refused permission for any of his plays to be filmed, exercising an unusual degree of authorial control over his work. Although arguments could be made against the dumbing-down of the work, did his refusal significantly reduce his potential audience? Is there an obligation of the artist towards the work? or does creation beget duties as well as rights?

The audience experiences the performance. In many cases, reading a work of literature is an example, the audience member and the performer are the same. In music, they are usually different, but this is not an advantage; the performance of chamber works by a small group of amateurs can be at least as rewarding for the participants as being a member of the audience in the concert given by a professional symphony orchestra. This is because of the close interaction of the experience and the interpretation phases of the performance, but it does not change the fact that the performance is actually happening in the head of the listener, and not at the point of contact between the bow and the string nor that of the composer's quill and his manuscript.

In all these processes, except at the purely technical level, the flow of information is essentially mono-directional. Although the performer can change their performance in the light of the sound they hear emanating from their instrument, this is because they are afforded the unique position of being both performer and audience.

If one is to respect the functions of the three stages of delivery in an artistic work, composition, performance and audition, one is to respect the composer's (or in fact the work's?) right to constrain the performance, and the performer's right to interpret, as well as the auditioner's right to perceive. Interaction can (but not necessarily does) interfere with these processes in that the auditioner or performer becomes composer, and changes the work that is individual of any of the three involved in its realisation. Some works actively encourage this process; Jazz or Aleatory music for example, although always within strictly prescribed limits: but for those which do not, any alteration outside of the accepted bounds is a recomposition. Interaction in this sense means the imposition of a compositional process on the performers or participant: a role they may be incapable and/or unwilling to undertake.

The traditional work of art can be described in terms of three attributes of its manifestation: structure; content; appearance. The new media adds a fourth: "behaviour".¹⁷ This enables, or even formalises, the act of interaction. Whereas in the case of the old media, the work is unable to respond to the interaction but is only passively modified by it, the interactive possibilities of multimedia permit the creator to add prescribed behavioural pattern, and to codify, formalise and "require" interaction between the work and the audience.

There is a correlation between the tools, ideals and activities of the open-source community, and the structure, content and appearance of the traditional artwork. Arguably, the computer science background of many of the Open Source Community participants have enabled them to structure their work more formally, strengthening their ideals and activities. Key among the ideals is that of the GPL, which requires freedom of use and sharing of code. Further ideals, such as a pure, fluid and dynamic process of peer review as embodied

in the IETF (Internet Engineering Task Force) further strengthen their cause.

In fact, C.P. Snow's "great divide" between the artistic and scientific process is a misrepresentation and shows an ignorance of what is involved in the act of creating a computer program or designing an electronic circuit. The concept of the divide has become ingrained in society — students of electronic engineering have objected loudly in the past when I have referred to their activities as lacking in artistic merit, yet would expect such feedback were they studying the subject in a design (*art* and design?) college rather than a University department — but it seems to me that the differences in presentation of scientific and artistic communities are to blame rather than any substantial difference. On the one hand, the artistic community values the individual contribution and the purely conceptual part of the creative process to a greater extent, while the scientists write papers which present the work in a logical sequence of steps even though the actual process, especially in a large system will have benefitted from hunches and subjective opinions. To deny this is as ridiculous as to suggest a symphony or a novel is unplanned and the result of pure spontaneous creation. It would be most surprising if the benefits of the hotbed of collaboration which is the free-software community did not have lessons for the arts.

Stallman's speech at MIT this year¹⁸ makes the point that copyrights, like most laws, are dependent upon and arise from a social context. A hermenutic examination of the original purpose of copyright quickly demonstrates that its current usage, essentially to the advantage of publishers before that of the users or originators, is a corruption. Consider this quotation which is just one decade old:

The primary objective of copyright is not to reward the labor of authors, but "[t]o promote the Progress of Science and useful Arts." To this end, copyright assures authors the right to their original expression, but encourages others to build freely upon the ideas and information conveyed by a work. This result is neither unfair nor unfortunate. It is the means by which copyright advances the progress of science and art.

— Justice Sandra Day O'Connor
(*Feist Publications, Inc. v. Rural Telephone Service Co.*,
499 US 340, 349(1991))

The issue of *fair use*¹⁹ arises when corporations are permitted too great a power over dissemination. Unfortunately, the Millenium Copyright Act²⁰ goes far in shattering Justice O'Conner's idealism by essentially removing the right of an owner of an artifact in electronic medium even to lend or show the work to their

¹⁷David Garcia *verbatim*

¹⁸Dr Stallman's speech is available in the open alternative format to "mp3": *ogg vorbis*. The audio file is obtainable from <http://www.gnu.org/philosophy/audio/rms-mit2001.ogg>, and the player from <http://www.vorbis.com/index.psp>

¹⁹The issue of what constitutes fair use in the publication of electronic texts is discussed at <http://www.arl.org/info/frn/copy/fairuse.html>

²⁰<http://www.loc.gov/copyright/legislation/hr2281.pdf>

acquaintances, thus removing all publicity and distribution, along with the associated benefits, into the hands of the publishers.

Formally, patents had nothing to do with copyright. Copyright issues arise when it is essentially free to make copies, such as a recording of some music or photocopying a journal paper. Such copies can be made at the cost of a fraction of that of acquiring the original. As yet, there is no such way to copy a machine or a drug. Nothing has the potential to damage education, scientific research and artistic endeavour more than the proposed introduction of software patents. Even industry stands to be severely damaged by such aggressively antidemocratic, pro-monopolistic regulation.²¹

Significant implications clearly exist if we desire to foster cooperation between the artistic and scientific

communities. The GPL mentality must be adopted, in the understanding that it promotes the visibility and usage of artistic work without compromising its commercial possibilities. The spirit of the Free Software Foundation is already being adopted by some of the music community in the dissemination of musical scores and editions (as for example at Mutopia²²) If there is one lesson to be learned from the history of the Free Software Foundation so far, it is that the importance of getting right the terms of distribution can not be underestimated. If commercial interests are allowed to prevail, the World-Wide Web of the 21st century will be another and more powerful method for distributors to control the content of what we view and hear, and not the engine for freedom of speech as it is popularly described.

A The Design Science Licence

Applying the GPL to non-software products has caused some difficulties, principally due to the ambiguity of some technical terms applied to works such as books and photographs.

Michael Stutz²³ has recently released the Design Science Licence, reproduced below. It is the licence under which this paper is distributed.

DESIGN SCIENCE LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

Copyright 1999-2001 Michael Stutz Verbatim copying of this document is permitted, in any medium.

0. PREAMBLE.

Copyright law gives certain exclusive rights to the author of a work, including the rights to copy, modify and distribute the work (the "reproductive," "adaptative," and "distribution" rights).

The idea of "copyleft" is to willfully revoke the exclusivity of those rights under certain terms and conditions, so that anyone can copy and distribute the work or properly attributed derivative works, while all copies remain under the same terms and conditions as the original.

The intent of this license is to be a general "copyleft" that can be applied to any kind of work that has protection under copyright. This license states those certain conditions under which a work published under its terms may be copied, distributed, and modified.

Whereas "design science" is a strategy for the development of artifacts as a way to reform the environment (not people) and subsequently improve the universal standard of living, this Design Science License was written and deployed as a strategy for promoting the progress of science and art through reform of the environment.

1. DEFINITIONS.

"License" shall mean this Design Science License. The License applies to any work which contains a notice placed by the work's copyright holder stating that it is published under the terms of this Design Science License.

"Work" shall mean such an aforementioned work. The License also applies to the output of the Work, only if said output constitutes a "derivative work" of the licensed Work as defined by copyright law.

²¹<http://www.smets.com/it/policy/useright/useright.pdf>

²²<http://www.mutopiaproject.org/>

²³Stutz was one of the earliest journalists to understand the motivations of the free software community, and explains further at <http://www.dsl.org>

"Object Form" shall mean an executable or performable form of the Work, being an embodiment of the Work in some tangible medium.

"Source Data" shall mean the origin of the Object Form, being the entire, machine-readable, preferred form of the Work for copying and for human modification (usually the language, encoding or format in which composed or recorded by the Author); plus any accompanying files, scripts or other data necessary for installation, configuration or compilation of the Work.

(Examples of "Source Data" include, but are not limited to, the following: if the Work is an image file composed and edited in PNG format, then the original PNG source file is the Source Data; if the Work is an MPEG 1.0 layer 3 digital audio recording made from a WAV format audio file recording of an analog source, then the original WAV file is the Source Data; if the Work was composed as an unformatted plaintext file, then that file is the Source Data; if the Work was composed in LaTeX, the LaTeX file(s) and any image files and/or custom macros necessary for compilation constitute the Source Data.)

"Author" shall mean the copyright holder(s) of the Work.

The individual licensees are referred to as "you."

2. RIGHTS AND COPYRIGHT.

The Work is copyrighted by the Author. All rights to the Work are reserved by the Author, except as specifically described below. This License describes the terms and conditions under which the Author permits you to copy, distribute and modify copies of the Work.

In addition, you may refer to the Work, talk about it, and (as dictated by "fair use") quote from it, just as you would any copyrighted material under copyright law.

Your right to operate, perform, read or otherwise interpret and/or execute the Work is unrestricted; however, you do so at your own risk, because the Work comes WITHOUT ANY WARRANTY -- see Section 7 ("NO WARRANTY") below.

3. COPYING AND DISTRIBUTION.

Permission is granted to distribute, publish or otherwise present verbatim copies of the entire Source Data of the Work, in any medium, provided that full copyright notice and disclaimer of warranty, where applicable, is conspicuously published on all copies, and a copy of this License is distributed along with the Work.

Permission is granted to distribute, publish or otherwise present copies of the Object Form of the Work, in any medium, under the terms for distribution of Source Data above and also provided that one of the following additional conditions are met:

(a) The Source Data is included in the same distribution, distributed under the terms of this License; or

(b) A written offer is included with the distribution, valid for at least three years or for as long as the distribution is in print (whichever is longer), with a publicly-accessible address (such as a URL on the Internet) where, for a charge not greater than transportation and media costs, anyone may receive a copy of the Source Data of the Work distributed according to the section above; or

(c) A third party's written offer for obtaining the Source Data at no cost, as described in paragraph (b) above, is included with the distribution. This option is valid only if you are a non-commercial party, and only if you received the Object Form of the Work along with such an offer.

You may copy and distribute the Work either gratis or for a fee, and if desired, you may offer warranty protection for the Work.

The aggregation of the Work with other works that are not based on the

Work -- such as but not limited to inclusion in a publication, broadcast, compilation, or other media -- does not bring the other works in the scope of the License; nor does such aggregation void the terms of the License for the Work.

4. MODIFICATION.

Permission is granted to modify or sample from a copy of the Work, producing a derivative work, and to distribute the derivative work under the terms described in the section for distribution above, provided that the following terms are met:

(a) The new, derivative work is published under the terms of this License.

(b) The derivative work is given a new name, so that its name or title cannot be confused with the Work, or with a version of the Work, in any way.

(c) Appropriate authorship credit is given: for the differences between the Work and the new derivative work, authorship is attributed to you, while the material sampled or used from the Work remains attributed to the original Author; appropriate notice must be included with the new work indicating the nature and the dates of any modifications of the Work made by you.

5. NO RESTRICTIONS.

You may not impose any further restrictions on the Work or any of its derivative works beyond those restrictions described in this License.

6. ACCEPTANCE.

Copying, distributing or modifying the Work (including but not limited to sampling from the Work in a new work) indicates acceptance of these terms. If you do not follow the terms of this License, any rights granted to you by the License are null and void. The copying, distribution or modification of the Work outside of the terms described in this License is expressly prohibited by law.

If for any reason, conditions are imposed on you that forbid you to fulfill the conditions of this License, you may not copy, distribute or modify the Work at all.

If any part of this License is found to be in conflict with the law, that part shall be interpreted in its broadest meaning consistent with the law, and no other parts of the License shall be affected.

7. NO WARRANTY.

THE WORK IS PROVIDED "AS IS," AND COMES WITH ABSOLUTELY NO WARRANTY, EXPRESS OR IMPLIED, TO THE EXTENT PERMITTED BY APPLICABLE LAW, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

8. DISCLAIMER OF LIABILITY.

IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS WORK, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

END OF TERMS AND CONDITIONS