

## WHAT DOES “OPEN” MEAN IN DIGITAL ARTS?

FLORIAN CRAMER

### CONCERNING THE TOPIC OF MY SPEECH

I have been kindly invited here to speak about “open source and open art”. If digital art would be free in the sense of Free Software, it would not only have to be available for free use and redistribution, but it also would have to be built and or at least viewable with Free Software. Which means: No use of Flash, no GIF files, no video DVDs, no mp3, no QuickTime, no Macromedia Director, no Quark XPress desktop publishing and so on. Such art would look and feel different because it would use different technical platforms, in some cases even technical restraints. Instead it would have to use open formats and free tools like ASCII, LaTeX, ogg vorbis, Perl or Python, centering its own distribution radically upon the possibility to be spread, viewed and modified with entirely free means, at the expense even of technically smoother or more integrated proprietary standard solutions.

Therefore, I would like to modify the topic “Open Source and Open Art” into the following question: “What does ‘open’ mean (and imply) in digital arts?”. In the digital arts – but not only there –, “openness” has a double meaning: It could, on the one hand, mean free public distribution of art or public access to it, or it could mean openness of its technical platform. (Let me elaborate on that.)

### ARTISTIC OPENNESS

If we speak about freedom and openness in digital information, the model clearly is Free (or Open Source) Software which has received increasing attention in artistic net cultures over the last few years. What is Free Software in relationship to Open Source? Both terms mean technically the same – i.e. copylefted software –, they only stress different aspects of it: freedom of usage and openness of the sourcecode. “Free Software” is the older, more political term associated with Richard Stallman, the Free Software Foundation and its GNU project, “Open Source” was coined in 1998, according

---

*Date:* November 2nd, 2002.

to its initiators, as a more business-compatible “marketing pitch for Free Software”.

The Free Software Foundation defines Free Software through four criteria, or, freedoms granted to its users:

- The freedom to run the program, for any purpose (freedom 0).
- The freedom to study how the program works, and adapt it to your needs (freedom 1). Access to the source code is a precondition for this.
- The freedom to redistribute copies so you can help your neighbor (freedom 2).
- The freedom to improve the program, and release your improvements to the public, so that the whole community benefits. (freedom 3). Access to the source code is a precondition for this.

Obviously, these freedoms are specific to algorithmic computer software, but not less obviously they can easily be generalized as (a) unrestricted freedom of using a work, (b) unrestricted freedom of studying the work, (c) unrestricted freedom of redistributing the work and (d) unrestricted freedom of modifying the work. The same principles are valid of “Open Source”. So Free Software at first doesn’t have any implications about the nature of its development process. It can be a community process, but – as many examples of Free Software show – it doesn’t have to be. On the other hand, its principles of course exist to allow and not restrict community authorship. Free Software implies, btw., a positive and strict notion of authorship, because it values attribution. A Free Software developer can make his or her authorship/coding skills publicly visible as opposed to a programmer of proprietary software. The best known programmers in the world like Donald Knuth, Andrew Tanenbaum, Richard Stallman and Daniel Bernstein are Free Software programmers. By itself, proprietary software is much less individual and much more an anonymous product than Free Software.

How does Free Software relate to art?

When we discuss Free Software and art, the implication often seems to be as if artists had to catch up with Free Software. Quite the reverse is true though if we look, for example, at Situationism. From 1958 onwards, publications of the Situationist International included the following copyright notice:

All texts published in *Situationist International* may be freely reproduced, translated and edited, even without crediting the original source.

In other words, the Situationist copyleft grants all the four freedoms of the Free Software definition – free usage, study, redistribution and modification. It therefore fully qualifies as a Free Software or Open Content license according to the Free Software Definition, the Debian Free Software Guidelines and the Open Source Definition of the Open Source Initiative.

Beyond the mere distribution policy, Situationist poetics themselves were centered around appropriation and modification of found material, which the Parisian situationists defined as “détournement”. A simple example was a situationist film which consisted of a kung-fu movie overdubbed with marxist theorizing. “Détournement” thus was a hybrid of Brecht’s epic theatre with its method of “estrangement” and collage techniques of 20th century avant-garde art including futurism, dadaism, surrealism and pop art.

In the late 1980s, a whole subcultural art discourse plagiarized the Situationists by making plagiarism its own main topic and slogan. Its main voice was the fanzine SMILE which could be published by anyone. (Picture SMILE 8). The name SMILE was a travesty of the General Idea’s Canadian arts magazine FILE which itself had been travestied by the Mail Art zines BILE and VILE. Since 1983, SMILE appeared in more than one hundred issues, each of them often plagiarizing the other. They were created by several dozen publishers who usually assumed the collective identities Monty Cantsin or Karen Eliot. Finally, the whole multiple name concept of Monty Cantsin and SMILE was plagiarized by the Italian Luther Blissett project in the 1990s which made the name of an English soccer player its phantom identity.

The institutional and media conditions of the Festivals of Plagiarism are almost emblematically visible on a photograph shot on the Festival of Plagiarism in Glasgow 1989: It is situated in an alternative gallery, the machine in the foreground is a xerox copier, and the exhibition consisted of xeroxed pamphlets, newspaper collages and self-made t-shirts. What is not visible on the photograph are other media like VHS video and audio tapes. In other words: All media and technologies are analog. As an interesting detail, computers have been used only in order to create analog artefacts. Although this is a Festival of Plagiarism, no one seems to have realized or reflected the radical implications of digital media for copy culture.

Through the gallery space and the use of media, the festival situates itself clearly in the visual arts. On the other hand, the media and technologies used are emblematic of subculture and non-professional art.

- The xerox copier is the multiplication technology of fanzine culture
- Collages and xerox copies are the main media of Mail Art which itself remained largely a network of amateur artists

- Audio tapes were the typical medium of self-produced underground music in the 1980s.

So in the end, this plagiarist art remained in a ghetto where it only recycled itself. It never succeeded to be aggressive, institutionally challenging plagiarism as an appropriation of official codes because it didn't succeed to grasp and therefore plagiarize the discourse of the official art world.

Of course, there are much older examples of "plagiarist" poetics in art or at least of a collective circulation and usage of artistic works: the novels of Rabelais, Cervantes and Sterne for example, the proliferation of musical themes in Renaissance music and Renaissance workshop painting.

If plagiarism could be defined as simulated novelty, then its opposite is fake as simulated historicity. Almost all religions and gnostic doctrines are founded on documents whose years of origins were manipulated to make them older as they actually were. (This is even true for the New Testament.) The names of prophets were collectively used sometimes over several centuries. Hermes Trismegistus and Christian Rosenkreutz could be called the precursors of Monty Cantsin and Luther Blissett, and that analogy can be found in SMILE issues themselves.

The 1980s plagiarists were aware of historical connections when they plagiarized the Situationists which in turn drew upon Lautréamont, the author of "Les chants de Maldoror", and a passage in his 1886 work "Poésies":

Plagiarism is necessary. Progress implies it. It approaches the sentence of one author, takes its expressions, removes a false idea and replaces it with a better one.

Already in 1886, Lautréamont does not simply conceive of plagiarism simply as doubling or accumulation of information, but – if you like – as a critical process of collective peer review and improvement of work. His saying could be smoothly plagiarized, detoured, appropriated into a principle of Free Software development: „Free Software is necessary. Progress implies it. It approaches the works of one programmer, takes its code, removes a false construct and replaces it with a better one.“ This exactly describes the way Free Software like the Linux kernel is being developed. On the other hand, Free Software also has its own specific notion of plagiarism which applies when somebody illegally puts free code into a piece of proprietary software.

In that perspective, Lautréamont's wording "plagiarism" appears to be somewhat odd, if nevertheless to the point. After all, the notion of "plagiarism" is, on a historical scale, quite modern. In the first century A.D. the latin poet Martial accused another writer of "plagium", i.e., "kidnapping"

of his verse. For 1500 years, this saying remained metaphorical, as the term “plagiarism” was coined as late as in the 17th century (according to dictionaries). In other words: Before 1600, the concept of “plagiarism” did not exist. That it was invented at the same time as copyright, is of course no accidental. Since its invention, plagiarism and thus all practices that defy “intellectual property” have become the other, the negative implication of copyright (and vice versa).

What is copyright? – Copyright is a political regulation of the publishing industry, not of the users. It is an exemption granted for a limited period of time, at the moment until 70 years after the death of the author. Most artworks and writings we know thus are no longer copyrighted. (But don’t ask me why museums still have “reproduction rights” on Renaissance paintings.)

All digital arts are code-based. Since the invention of writing and mathematics, the particular quality of code has been that it can be copied and transcribed without information loss in the reproduction process. Because of this, the equation  $1+1=2$  has survived without falsification over a couple of milleniums, although it has been recoded into various notation schemes including Roman, Arabic and binary numbers.

So far, I have spoken only of copyright and copyleft strategies as regards to so-called “content”, i.e. information itself, not the technological carriers of information. But if we speak about electronic and digital arts, we have to particularly focus on the formats, protocols and generators of information.

#### TECHNICAL OPENNESS

If we look at the material and immaterial carriers of traditional arts, we see that they all could be called “open platforms”: Canvas, paper and books, marble, musical score notation, language. Their manufacturing and internal structure was open and transparent in that they weren’t proprietarized by monopolies. Any manufacturer or artist can produce them either commercially or noncommercially.

Among the materials listed, the closest analogy to software and digital data formats might be languages and musical score notation, since they are symbolic and not material. Both written languages – like English and German – or musical score notation are open, publicly documented standards. Equivalents of closed platform are harder to imagine: A composer for example could design a cryptic, nonstandard notation system, teach it only verbally to musicians whom he would force to sign a non-disclosure agreement and

never pass along their knowledge. The result would be, obviously, that his music would never be performed anymore after his death.

In electronic information technology, most hardware standards are open. This applies to TV standards like PAL and NTSC, video standards like VHS, the audio CD format or FM radio. You don't have to pay patent fees or royalties for soldering your own radio receiver or burning your self-recorded audio cd. Free software programmers can write audio cd playing software for PCs without having to pay patent fees either.

But the same is no longer true for formats like video DVD or even mp3 audio, both of which are encumbered with patents or proprietary cryptography schemes. The situation has changed all the more with the growing importance of software for the display of digital information: While noone used software players when the audio CD was introduced, many people prefer software DVD video players to consumer hardware DVD players. In addition, software-only formats become increasingly popular, thus making digital media formats significantly more fluid and innovative. For video, the DiVX became popular in a very short time, it was improved without regards to data compatibility, and it can supersede older formats or codecs, because people aren't forced to buy new hardware, but just need to upgrade their software to play the new format. Which means: file formats and encoding schemes become the equivalent of paper, canvas, score notation, etc. But this means that information carriers are no longer stable for centuries or even milleniums, but change in increasingly rapid cycles.

The other problem is that proprietary data formats create a vendor lock-in. A company can now have the monopoly on the *access* to data just by controlling the file format and keeping its internals secret. Before the popularization of the Internet, file formats were a secondary issue because files on personal computers were just seen as secondary devices for eventually delivering a work on computer-independent output media such as paper, video tape or audio cd. Since the popularization of the Internet, file exchange increasingly supersedes those output media. If a single company controls the file format, it owns the information carrier and thus has monopoly control over both the production and perception of works created in this format.

A prime example is Macromedia Flash, others are QuickTime, Microsoft Office file formats, Visual Basic and browser-proprietary HTML. Macromedia Flash intentionally blurs the distinction between the file format, the authoring application and the viewer plugin by calling them all the same and forcing artists and their audience to accept that short-circuit. The company can arbitrarily decide which computing platforms to support and which not

– excluding for example, users of the GNU/Linux and BSD operating systems on non-Intel hardware –, can arbitrarily decide on the licensing of both the authoring and the viewing software, and it can arbitrarily decide to kill off the format at any time. Artists working in Flash thus must invest an enormous amount of trust and confidence in the company which in fact controls the availability of their work. – An extreme example of such forced user loyalty is the programmer-artist Netochka Nezvanova and her absurdly proprietary video manipulation software “NATO”; if anybody criticizes Netochka Nezvanova in public, she revokes their software licenses.

The historical example of a proprietary piece of software that obscured the distinction of authoring software and file format was Apple’s HyperCard program. In the late 1980s, HyperCard was extremely popular among digital artists as a combined hypertext, multimedia and programming environment. Although most artistic work done in HyperCard was public domain and could be modified up to the level of its scripting algorithms, it is factually lost today after Apple discontinued developing HyperCard and older versions no longer run on the present MacOS. Unlike in the transition from VHS or UMatic to DVD, there is no way of transferring the old HyperCard data into a newer format like, for example, script-generated HTML. This explains why Free Software and open standards are attractive for digital artists because they provide the only alternative to vendor lock-in. If HyperCard had been Free Software according to the definition I gave at the beginning of my talk, even artists without programmer’s skills could have taken the sourcecode of the program and commission, perhaps against payment, other programmers to port and maintain the code for today’s operating systems and PC architectures, or they could commission a software converter from HyperCard to script-generated HTML. I am afraid that we will see the same story happening again with Flash. The work of what Lev Manovich calls the “Generation Flash” is doomed because it is encumbered into non-open software and will vanish with it. This example also shows why free software is not about freedom for programmers, but about freedom for users.

(But we need to differentiate: Free software vs. proprietary software is not the same as open standards vs. proprietary data formats. An example: The jpeg image format is an open, publicly document, patent- and royalty-free standard. You can create, display and modify jpeg images either with Free Software (like the Gimp) or with proprietary software (like Photoshop), you can create, display and modify a standard-compliant web page either with Free Software (like Emacs and Mozilla) or with a proprietary software (like Windows Notepad and Internet Explorer). Since Free

Software is fully transparent in its sourcecode and includes no restriction of usage, the file formats it creates are always open by definition.)

#### THE BOTTOM LINE

Free Software enables digital art to be put into public use and therefore is in self-interests. Change of aesthetics is not always obvious, but can be substantial if one switches to only free tools and values data transparency. One shouldn't glorify hackers as the better digital artists – their notions of art are mostly conservative, based on a classicist notion of beauty and good craft. In the end, "Openness" in art is above all an ethical question. I would highly resist making a quality distinction between open art as good and proprietary art as bad. Example: Netochka Nezvanova. Mallarmé. Lautréamont said that poetry should be made by all, but his works suggest the opposite. Art can be particularly good when its morally questionable.

Critical of Foucault and Foucaultian critics like Kittler: Functionalizing people, denying subjectivity. Subjectivity does exist and is important. Free Software comes from a scientific and engineering tradition, not from art.