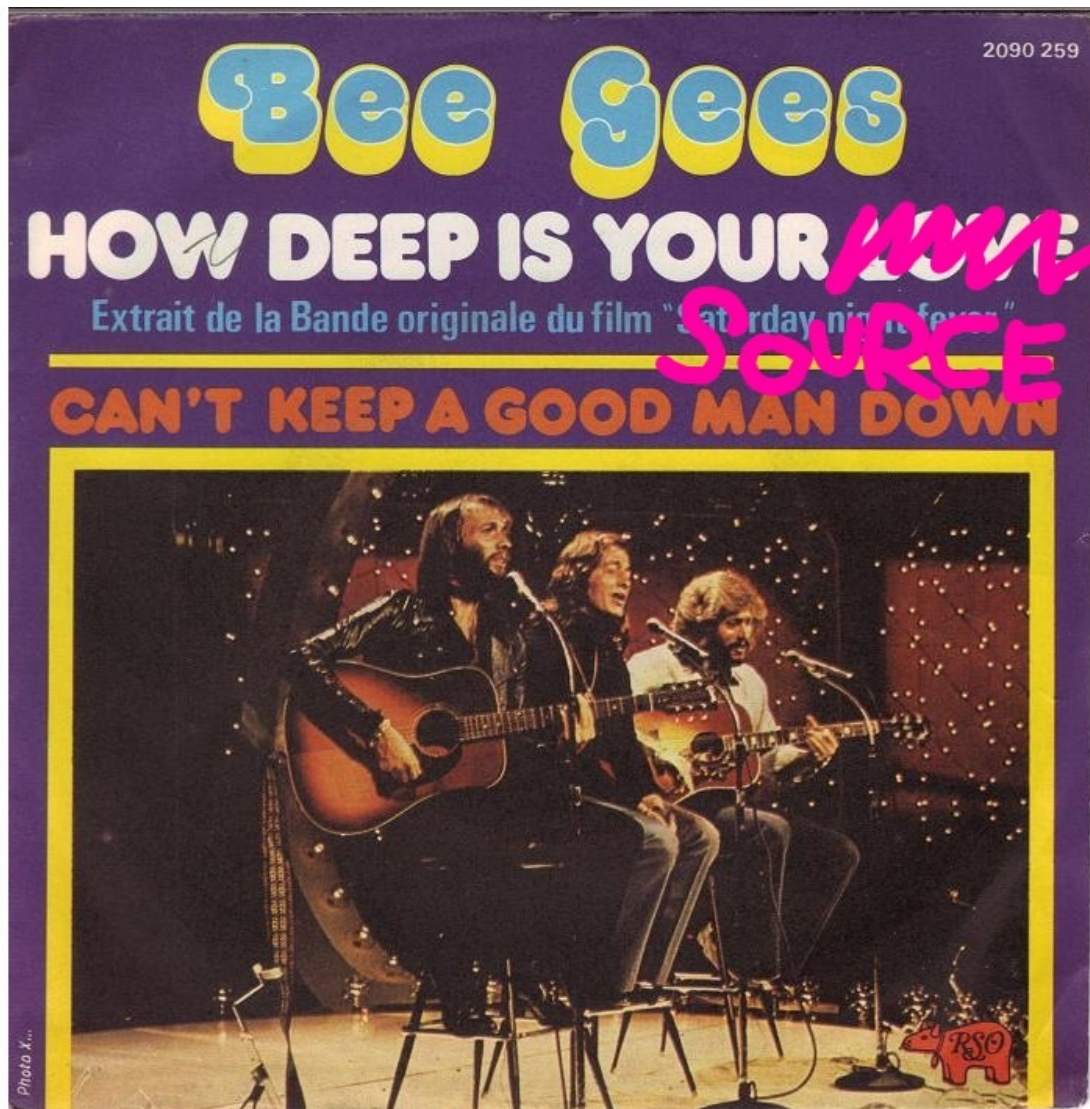


How Deep is your Source?

Aymeric Mansoux - 2013



Introduction

Within the realm of archiving and conservation, free and open licenses are a useful tool to make both the reuse and the preservation of digital art more feasible.[1] Even though it is undeniable that such licenses have an overall positive effect, they do however struggle when it comes to defining the components, materials, and assets that have been involved in the creation of a work. The lingo used in such licenses stems from the world of software and despite its adaptation to cultural works, it still hasn't got rid of the binary nature of its origin. When a software license requires the access to its source, it refers to the "source code," a well-known object in the making of software that is easy to define and identify. While this perfectly fits a particular form of art, in particular art that involves computational processes, it is questionable whether this concept of source code can be literally ported to other digital cultural expressions, such as moving or still digital images, sounds, and by extension multimedia and rich media works.

In this paper, I argue that such works require both a media-specific as well as a metaphorical understanding of what source code is in the context of art, music, and design in order to make free and open licenses valuable for conservation, archiving, and of course in-depth study and reappropriation of the former. Behind this challenge lie the issues of accessibility and control with free cultural expressions and open knowledge, and how these compare with their software-centric parent definitions.

A Brief History of Software Freedom

Source code is a collection of computer instructions written using a human-readable computer language.[2] The source code can then be either compiled in machine code and executed manually to perform some tasks as a stand-alone program or as part of a larger software. It can also be interpreted, in which case the source is both translated and executed on the fly. The role of source code in the production and manipulation of software is essential. Its access allows virtually any modification of the latter, whether it is about adding new features or fixing bugs. This importance is best exemplified with the free software movement, and the creation of the Free Software Foundation (FSF) in 1985. Since its very beginning, the goal of the FSF has been to provide an alternative to the proprietary software model that relies specifically on controlling the access to and distribution of source code and compiled software.

“The word ‘free’ in our name does not refer to price; it refers to freedom. First, the freedom to copy a program and redistribute it to your neighbors, so that they can use it as well as you. Second, the freedom to change a program, so that you can control it instead of it controlling you; for this, the source code must be made available to you.”[3]

The foundation, as well as its early attempts to define software freedom, has been an inspiration for many, including Ian Murdock, the founder of the Debian operating system[4]. This influence was further demonstrated as this project grew and published its own free software guidelines, in which crucial importance was given to source code. [5] Eventually, this guideline was modified a few years later to serve as the Open Source definition[6] of the newly founded Open Source Initiative (OSI) which resulted from the 1998 call to embrace the term “open source.”[7] Yet, probably the most important text written on free software is about the four essential freedoms of program 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.”[8]

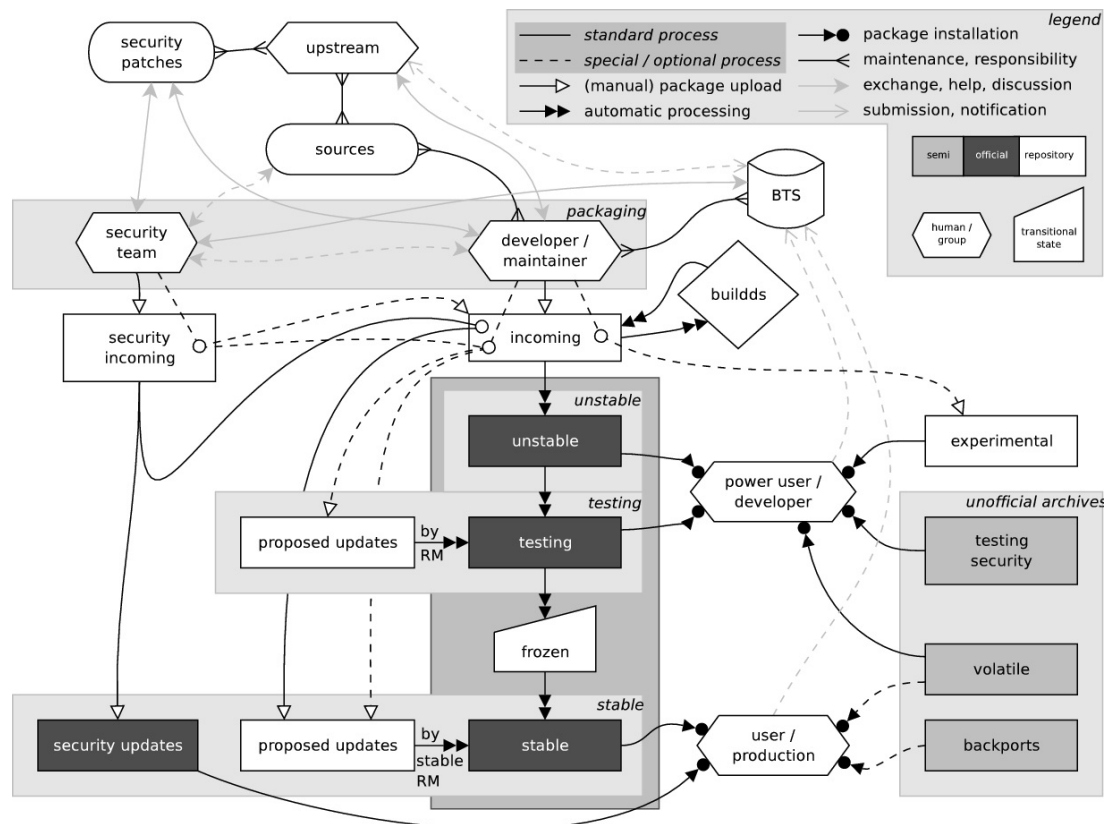
According to Bruce Perens, these freedoms established by Richard M. Stallman were only published on the Free Software Foundation website after the creation of the OSI, “as an alternative to the Open Source Definition”[9]. He suggests, however, that they probably existed prior to their online release. As a matter of fact they even existed as three freedoms,[10] which are now numbered 1, 2 and 3. The freedom 0 was added at a later stage.[11]

These parallel efforts from the FSF, Debian, and the OSI work as both a guideline for new, and a filter for existing contracts specific to intellectual property: the licenses. Indeed, good will, manifestos, and announcements are clearly not enough to enforce consistently these visions and definitions. So in order to enable the latter, the license works as a contract between the copyright holder of the source code (for licenses relying on copyright) and its user. It makes sure that software freedom and openness are ensured once the software is published, and at the same time it provides legal mechanisms in case of non-respect of the license.

In practice the relationship between source code, license, and accessibility is clearly visible in the way the Debian operating system is put together. With a standard installation, Debian provides the usual graphical user interface desktop metaphor similar to MacOS or Windows, and a collection of several free and open source software both for general and specialized tasks. What is interesting is how this operating system is connected to several repositories of software and their respective source code. The latter is distributed as packages, collections of files that can be copied or downloaded from offline and online data storage and installed on one’s computer. More specifically, source packages provide several items: the original source code written by the author(s) of the original software, as well as optional patches to apply on top of it and the license(s) under which these files are published; the metadata of the distributed software, that is its description, category, list of author(s) and maintainer(s), etc; the conditions of its access in the means of technical dependencies. Finally, any changes in these files are logged and stored in the packages themselves. These changes combined with the storage of previous versions of the software, and its source code, as well as the ability to access these at any time literally turn the Debian operating system into a software archaeological excavation site.

In addition, these packages can be mirrored by anyone with enough space and bandwidth, and are currently available on nearly five hundred officially registered

servers across the world.[12] From a user perspective, package managers, or an administrative software, can then be used to install, remove, or upgrade software, and as a consequence maintain one's operating system to one's liking. This process is not unidirectional; users are given the possibility to help and give feedback by writing documentation, submitting bug reports, writing patches for their favorite software, even maintaining such a software themselves by becoming official maintainers and, why not, official Debian developers as well. Most importantly the whole infrastructure can be reappropriated and derived into new projects, new Debian-based operating systems, and software collections.[13] In that sense the Debian infrastructure is not a classic form of dedicated archiving system: instead, here archiving, conservation, distribution, and access are merged into one replicable structure.



Flowchart of the life cycle of a Debian package, 2005, Martin F. Krafft, based on the work by Kevin Mark (CC BY-SA 2.5)

Such a participatory living archive is possible because of the access and distribution of the source code by the means of a techno-legal framework that allows for a certain fluidity and manipulation of information from the public lists where software maintenance is discussed to the private computers where it is effectively manipulated and executed. It is a great example of how source code can become an invitation "for creative practice encouraging collaboration and further development of existing work on the level of contribution, manipulation and recombination, and its further release under the same conditions in the public domain".[14]

Now, whether a work would be code itself, its manifestation through the execution of the compiled code, or a combination of both, it's not a big stretch to see how Debian can be an inspiration to support art that fits within this form.[15] Ultimately, both public and private collectors, venues, and libraries could benefit in a cooperative and distributed resource for free and open source code poetry, net.art, generative art, and software art. Generally speaking, the same would apply for media art where software elements could also be released under free and open source licenses and integrated in distributed infrastructures. In the end, regardless of how one wants to frame software – as art, tool or magical information – the technical and legal benefits of free and open source software are the same, because this is still software that we are dealing with.

This is not a fantasy or wishful thinking: in practice a free and open source GNU/Linux distribution such as Puredyne[16] has been distributing works from artists such as Alex McLean and Martin Howse.[17] Next to works that are already distributed by Debian, such as Electric Sheep,[18] it is possible for media artists releasing their work as free and open source software to be approached by distribution maintainers to help integrate their project within free and open source operating systems.[19] It goes without saying that such software must comply with the distribution's guideline and understanding of user-friendly applications. While there is no trouble for a work like Electric Sheep that effectively runs as a screensaver, the same cannot be said, thankfully, for all software art.[20]

But what about works that are neither code nor software based? While software is

undeniably a cultural expression,[21] all cultural expressions are obviously not software. How would that practically work?

Enter free culture...

Free and Open Content for the Masses

The 2000s saw the birth of several projects and movements that found their inspiration in free and open source software. The transition from software to culture and knowledge is, for instance, clearly visible in the Manifiesto de Hipatia[22] that connects free software to activism through the value of knowledge access. Yet, being driven by different understandings of what freedom and openness mean in the context of culture and knowledge, in practice the more explicit definitions that follow can differ greatly from one to another. These differences are also made visible in their selection of licenses that match their respective intentions and agendas. First of all, possibly the closest adaptation of software freedom to a broader cultural context are the “four kinds of free knowledge”:

“The freedom to use the knowledge, for any purpose (freedom 0).

The freedom to study how the knowledge applies, and adapt it to your needs (freedom 1). Access to the source information is a precondition for this.

The freedom to redistribute knowledge so you can help your neighbor (freedom 2).

The freedom to improve the knowledge, and release your improvements to the public, so that the whole community benefits (freedom 3). Access to the source information is a precondition for this.”[23]

Another one is the Free/Libre Knowledge definition, from the Free Knowledge Foundation (FKF) that has been created to describe what users should be free to do with cultural expressions:

“(0) use the work for any purpose

(1) study its mechanisms, to be able to modify and adapt it to their own needs

(2) make and distribute copies, in whole or in part

(3) enhance and/or extend the work and share the result.”[24]

Yet another are the “essential freedoms,” written as a guide to decide whether or not a license can qualify to a free culture license:

“The freedom to use and perform the work [. . .]

The freedom to study the work and apply the information [. . .]

The freedom to redistribute copies [. . .]

The freedom to distribute derivative works [. . .]”[25]

A further approach is the open content definition, which is an attempt to make a stronger distinction between rework and remix. It is also a twist on the four freedoms, and in this case it has been renamed the “4Rs Framework”:

“Reuse – Use the work verbatim, just exactly as you found it

Revise – Alter or transform the work so that it better meets your needs

Remix – Combine the (verbatim or altered) work with other works to better meet your needs

Redistribute – Share the verbatim work, the reworked work, or the remixed work with others”[26]

Finally, another notable effort is the Open Definition, or Open Knowledge Definition, that “sets out principles to define ‘openness’ in relation to content and data.”[27] The definition, too long to reproduce here, has been directly derived from Perens’ Open Source definition, which, as mentioned previously, was itself derived from Perens’ own work on the Debian’s free software guideline.

With such a strong affiliation both in style and content, the link to their parent software-centric definitions is blatant. In spite of that, if we take a closer look, even though the first attempt in porting the software freedom to knowledge did mention the idea of

source in its definition, the efforts that followed stopped mentioning it.

The reason for this is simple: while computer software is a cultural expression, not all cultural expressions are computer software. Therefore, the computer-specific issue of source code can be seen as not so relevant in a broader approach to free culture and open knowledge.

Towards the Borges Public License

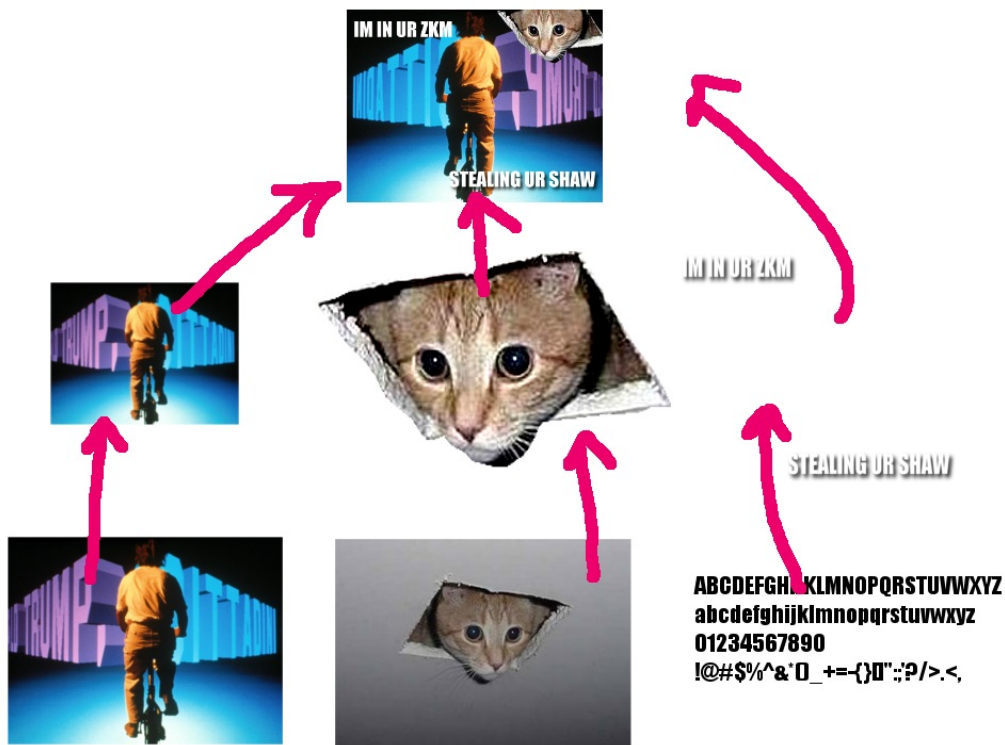
Seeing now that the free culture license definition, the open knowledge definition and other approaches are not a perfect transposition of free and open source software due to the lack of or incomplete approach to defining what is a source, we must then ask ourselves how this can impact free/libre/open content/knowledge/expression/work (FLOCKEW?).

First of all from a simple qualitative perspective, it means that any content is fine to publish and distribute. For instance, a low-resolution, highly compressed, photo or video can be distributed freely under these licenses. While this work perfectly qualifies as FLOCKEW, its value becomes questionable when the high-resolution, raw, or less destructively compressed “original” still remains under monopolistic exclusive rights. Another aspect is what the freedom defined project calls the “practical modifiability” of a work that is licensed with certain free culture licenses: how in practice a work can be appropriated and modified by someone else. For instance, if the licensed work is an image composed of several elements, its practical modifiability is affected if the author decides to publish such an image as a flatten-down work, or if instead, she or he also provides the layers used to make this final image.



Highly compressed, yet free, thumbnail.

Furthermore, there is an unavoidable recursion triggered by the existence of such external “source” files. Indeed, and still using the example of a digital collage, we can ask ourselves what would happen if the layers provided were themselves derived from other “originals”? Shouldn’t they be included as well? What about the font used for a caption or logo, what would be the practical modifiability of a rasterized text layer? Would it make sense to provide the font file? If we want to practically modify the file, beyond the one-dimensional mashup or remix, such elements are very much needed. A last problem arises with the licensing of these media assets. If an author would distribute the “source” of his or her work, this source being a distinct cultural expression itself, the author is free to distribute the material under separate licenses. Is it acceptable then for free content to have its assets under non-free culture licenses? What about license compatibility? Is it acceptable if these external cultural expressions are freely licensed, yet using closed standards from proprietary software?



ZKat sources?

To address such issues Rob Myers imagines what an ideal cultural source would be. He suggests to consider five attributes: transparent, in an easily editable text-based format; full quality, in a standard that permits the recreation of the final format; complete, so that all the materials to produce the distributed work are provided; unencumbered, that is, free of patents and DRM (Digital rights management); structured, as provided in a descriptive format, such as vector graphics.[28] Even though the open definition excludes explicitly software from its definition of knowledge, therefore avoiding the question of source in the distribution of open works, the question of the source is actually covered by the freedom defined project. In fact to be a truly free cultural work, a work must respect four more conditions, one of them is being specific about the idea of source data:

“Availability of source data: Where a final work has been obtained through the compilation or processing of a source file or multiple source files, all underlying source data should be available alongside the work itself under the same conditions. This can be the score of a musical composition, the models used in a 3D scene, the data of a scientific publication, the source code of a computer application, or any other such information.”[29]

However, unlike with free software licenses this condition is not contractual. It is simply part of a guideline to decide whether or not a work can be called a free cultural work. Put differently, an author does not have to respect this condition when using a free culture license. In practice it is thus possible to distribute non-free works with free culture licenses, literally turning the popular understanding of free culture into a messy mix of both free and non-free cultural expressions.

At this point things can start to get fairly confusing. What started as a simple exploration into the transposition of free software to free culture, is now ending up in a maze of concurrent definitions. Each of them pointing to a “Choose Your Own Adventure” labyrinth of licenses, where every step seems to further obfuscate the source of the problem, literally. As a matter of fact Creative Commons even use the misleading term “approved for free cultural works”[30] for its licenses that respect the free culture license definition, whereas it really should say that such or such licenses are free culture licenses; no less, no more.

No matter how annoyingly picky it may seem, the question of cultural sources must be raised. It is essential to do so in order to understand the value and the limits of a FLOCKEW in the context of archiving and conservation. As we see now, without source information and in the case of conservation and reappropriation, the advantage of such practice is limited. Truth is, if there is such a thing as an artwork “source code”, it must contain much more information than remixable elements. Such a source is made of everything that a work contains, from the dirt that feeds its roots to the canopy. A radical approach like this one should not to be understood solely from a paratextual or paramedial perspective, but instead in a very practical way as best exemplified with the PlayStation Harddrive[31] published in 2001. This is a CD-ROM that contains “raw data”

from the hard drive of media artist Joshua Davis; data that is meant to be explored, studied, and reused. While the content is in fact far from being a bitstream copy of his drive, it is nonetheless quite an impressive collection of 3637 files, including many Macromedia Flash “source” files, making the intention behind the project close to the free and open source ethos.[32] In practice this parallel is more of a misunderstanding of how free software and open source operate as the files have been released without any licenses,[33] making the drive fall into the gooey swamp that is unspecified public domain and default copyright laws. Yet its immense positive impact on the Flash community at that time, both as an educational and cultural artifact, is a very direct illustration that without a dump of an artist’s storage device no complete works or biography can be written.[34]

In the end, a thorough publication of properly licensed source materials for works of art is rare and is limited to artists and collectives already close to free and open source software communities. Exposed to such practices in the software they use, some of them eventually apply the same philosophy to their own work, and make many elements of the latter publicly available in repositories, using different software licenses.[35] Such an attitude towards the meticulous sharing of source material is unlikely to become popular due to the complete or partial disappearance of an articulated concept of source in the licensing of FLOCKEW. This is the reason why some free culture licenses, more specifically from Creative Commons, should not be used for software.

Even though there is no such thing as a “bitstream” license yet, some efforts to take into account these issues are worth mentioning. For instance, as early as 2004, the Open Art Network started to work on the Open Art license, also known as the “View Source”, or simply the “Source License.”[36] In this non-free[37] license it is stipulated that “source file/s for the work must remain accessible to the public.”[38] Unfortunately, there is no consideration of the nature of the standard used for such source files. It can be a proprietary format using proprietary software, and would still be, not without problems, considered as a valid source for the open art license.

Another take on the question can be found in Valentin Villenave’s ongoing, yet to date unpublished work on a new license that would solve some of the source issues discussed so far. Villenave is an active member of the Copyleft Attitude[39] community from which the free culture Free Art license was born. In the latter the concept of source is mentioned, but is not clearly defined. This lack of clarity is the point of departure for Villenave’s new license in which the artist would be required to provide all intermediary source materials that were used during the creation of a work of art, including sketches, research, and all versions of these. If at any given time a source element is involved, it must be provided, so as to avoid the situation where what is given access to is in fact a summary of the work and not the work as a whole.[40]

With this extra step we are now witnessing that our problem is expanding even beyond the recursive vertigo triggered by diving into the sources of cultural sources: it is also reaching the context in which these very sources are created. After all, if we are aiming at providing the source code of an artwork, the capture of the creative process is completely relevant.[41] And so without noticing it, the frustration deriving from the lack of definition of artistic sources makes us drift into the analysis and recording of the artistic practice itself. Defining an artistic source is as problematic as defining art, yet our access to increasingly sophisticated legal and technological tools, which can enforce a fine-grained versioned capture of the artistic creation, directly feeds into our desire to sample and make tangible the “participation mystique” of the poet.[42] Indeed, if there is more to these sources than just a flatten-down object, nothing is preventing us from also providing electroencephalographic data, DNA samples, cosmological models, and more, thus transforming the capture of pretty much any phenomenon into the source of a work of art becoming noumenon. If anything at all we might well rename our speculative “bitstream” license the Borges Public License, a real treat for the Babel librarians and their lawyers, but very likely a nightmare for artists agreeing to sign with their blood such a devilish contract in exchange for an immortal openness.

Conclusion

Until a serious copyright reform, or why not complete abolition, takes place, it is undeniable that projects that promote FLOCKEW are very much welcome and needed. Practically speaking the usage of licenses that fit the free culture license definition or the open knowledge definition simplifies a lot and unifies the process of archiving, distribution, and access to culture and knowledge; that is, of course, in the very specific context of works respectful of existing copyright laws.[43]

However, the ability to modify practically, therefore reappropriate or maintain such expressions, should not be taken for granted. This situation puts free culture and open knowledge in a completely different camp to the one of free and open source software. By not taking into account the articulation of sources, free culture and open knowledge can only provide an ersatz of the freedom found in free software. In the end, what

software freedom achieved for the user is disappearing in its cultural transposition as the author is brought back into a position of complete control over the reusability of his or her work by others. This is possible because the author is the sole judge to decide whether or not to provide sources, and what constitutes them.

So even if FLOCKEW licenses are very effective to fluidify information, they have no effect on its actual usability. In that sense, Creative Commons licenses can give a false sense of security. All the issues covered in this paper represent as many potential backdoors waiting to be exploited, as well as many opportunities to misunderstand how this whole legal infrastructure operates. The very use of the cumbersome FLOCKEW acronym is just the tip of the iceberg. At the same time, it is clear that no solutions can be satisfying when it comes to trying to define sources both within a free culture and artistic context. There can only be specific workarounds. Any rational attempt to define artistic sources is simply not working because a general “one size fits all” definition of such sources will always tend to exclude practices rather than include them. Besides this, there is a risk with trying to find a compromise by simplifying and limiting the scope for valid sources of cultural expressions. Doing so would only benefit certain forms of practices by turning art into the byproduct of rationalized transformations – as we can see with remix culture.

The strength of free culture and open knowledge is not so much of being a technical tool that can solve copyright problems. Instead, it should really be appreciated here as a device that makes suddenly tangible the obfuscation and secrecy found in art practices and art preservation. The most important aspects are the dynamics and values that are found in communities involved in sharing cultural expressions. The organization of such communities can effectively help rethink and reinvent the access to culture in the age of distributed infrastructures by opening up the institutional walled gardens of conservation. Museums, art institutions, and archives urgently need to look closer at the different models of sharing in which the preservation of content can be globally scaled and accessed publicly while still benefiting from the love and care of a network of dedicated collectors, instead of building a business model around zombie works, exploiting their every possible permutation, and thus replicating at a different scale the exploitative cultural dead-end found in the relationship between media industries and copyright.

The question still remains, how deep is your source?

Footnotes

[1] Anne Laforet, Aymeric Mansoux, and Marloes de Valk, “Rock, Paper, Scissors and Floppy Disks,” in: Annet Dekker (ed.), *archive2020. Sustainable Archiving of Born-digital Cultural Content*, Virtueel Platform, Amsterdam, 2010, pp. 21–29.

[2] Wikipedia, the free encyclopedia, “Source code,” available online at: http://en.wikipedia.org/wiki/Source_code, accessed 15/08/2012.

[3] Richard M. Stallman, “What Is the Free Software Foundation?,” in: *GNU’s Bulletin*, vol. 1, no. 1, 1989, pp. 8–9, there p.8. Available online at: <http://www.electricsheep.org/>, accessed 07/02/2012.

[4] Ian Murdock, “A Brief History of Debian (Appendix A: ‘The Debian Manifesto’),” 1994, available online at: <http://www.debian.org/doc/manuals/project-history/ap-manifesto.en.html>, accessed 07/02/2012.

[5] Bruce Perens, “Debian Social Contract,” Version 1.0, July 5, 1997, available online at: http://www.debian.org/social_contract.1.0.en.html, accessed 07/02/2012.

[6] Bruce Perens, “The Open Source Definition,” in: Mark Stone, Chris DiBona, and Sam Ockman (eds.), *Open Sources: Voices from the Open Source Revolution*, O’Reilly Media, Sebastopol CA, pp. 171–188.

[7] Eric S. Raymond, “Goodbye, ‘Free Software’; Hello, ‘Open Source’,” 1998, available online at: <http://www.catb.org/~esr/open-source.html>, accessed 07/05/2012.

[8] Richard M. Stallman, “Philosophy of the GNU Project. What Is Free Software?,” 2000, available online at: <http://web.archive.org/web/20000302065400/http://www.gnu.org/philosophy/free-sw.html>, accessed 07/09/2012.

[9] Bruce Perens, “Re:1997 ?,” 2009, available online at: <http://news.slashdot.org/comments.pl?sid=1129863&cid=26875815>, accessed 07/11/2012.

[10] Richard M. Stallman, “Philosophy of the GNU Project. What Is Free Software?,” 1998, available online at: <http://web.archive.org/web/19980126185518/http://www.gnu.org/philosophy/free->

sw.html, accessed 07/10/2012.

[11] Richard M. Stallman, "Philosophy of the GNU Project. What Is Free Software?," 1999, available online at: <http://web.archive.org/web/19990430060825/http://www.gnu.org/philosophy/free-sw.html>, accessed 07/09/2012.

[12] The Debian Project, "Debian Worldwide Mirror Sites," 2011, available online at: <http://www.debian.org/mirror/list-full.en.html>, accessed 11/07/2012.

[13] The Ubuntu operating system is probably the most popular example of such a fork.

[14] Joasia Krysa, and Grzesiek Sedek, "Source Code," in: Matthew Fuller (ed.), *Software Studies: A Lexicon*, MIT Press, Cambridge MA, 2008, pp. 236–243.

Here "public domain" should probably be understood in the sense of a resource commonly put together and publicly accessible, as most free and open source software is not in the public domain. They are copyrighted material and made available through the license mechanism discussed above. However, it might be worth noting that Creative Commons CC0 license, that aims at simplifying the waiving of copyrights and related or neighboring rights, is currently compatible with the FSF General Public License (GPL).

[15] Javier Candeira, "Towards a Permanently Temporary Software Art Factory (Notes for the Sustainability of Software Artifacts)," in: Olga Goriunova (ed.), *Readme 100. Temporary Software Art Factory*, Books on Demand GmbH, Norderstedt, 2006, pp. 105–121.

[16] Puredyne is an USB-bootable GNU/Linux operating system for creative multimedia based on Ubuntu and Debian Live, cf. <http://puredyne.org/about.html>, accessed 06/29/2012.

[17] Anne Laforet, *Le net art au musée. Stratégies de conservation des œuvres en ligne*, Editions Questions Théoriques, Paris, 2011, p. 137.

[18] Scott Draves, "The Electric Sheep Screen-Saver: A Case Study in Aesthetic Evolution Applications of Evolutionary Computing," in: Franz Rothlauf, Jürgen Branke, Stefano Cagnoni, David W. Corne, Rolf Drechsler, Yaochu Jin, Penousal Machado, et al. (eds.), *Applications of Evolutionary Computing*, (Lecture Notes in Computer Science, vol. 3449), Springer, Berlin, Heidelberg, 2005, pp. 458–467. –See also: <http://www.electricsheep.org/>, accessed 07/02/2012.

[19] Laforet, Mansoux, de Valk 2010, pp. 23–34.

[20] In particular, this case was discussed during the "FLOSS+Art" panel during the 2007 edition of the "make art" festival, taking as example whether or not a "user friendly" program exit mechanism should be implemented in a work from Martin Howse, now that the software was distributed in Puredyne.

[21] Matthew Fuller, *Behind the Blip: Essays on the Culture of Software*, Autonomedia, New York, 2003, pp. 11–32.

[22] Mario Teza, Luiz Teza, Diego Saravia, Juan Carlos Gentile, and Luis González, "Manifiesto de Hipatia," Buenos Aires, 2001, available online at: http://www.hipatia.net/index.php?id=manifiesto_es, accessed 07/09/2012.

[23] Ismael Peña-López, "The Four Kinds of Freedom of Free Knowledge," in: *ICTlogy*, no. 1, October 2003, available online at: <http://ictlogy.net/review/?p=12>, accessed 07/09/2012.

[24] Available online at: http://en.wikipedia.org/wiki/Libre_knowledge, accessed 07/09/2012.

[25] The Definition of Free Cultural Works project, "Definition of Free Cultural Works 1.0", 2007, available online at: <http://freedomdefined.org/index.php?title=Definition&oldid=4582>, accessed 07/09/2012.

[26] David Willey, "Open Education License Draft," blog posted August 8, 2007, available online at: <http://opencontent.org/blog/archives/355>, accessed 07/09/2012.

[27] Open Knowledge Foundation, "Open Definition", Version 1.1, 2012, available online at: <http://opendefinition.org/okd/>, accessed 07/09/2012.

[28] Rob Myers, "Cultural Sources", August 26, 2007, available online at: <http://robmyers.org/weblog/2007/08/26/cultural-sources/>, accessed 07/09/2012.

[29] The Definition of Free Cultural Works project, "Definition of Free Cultural Works 1.0," 2007, available online at: <http://freedomdefined.org/index.php?title=>

Definition&oldid=4582, accessed 07/09/2012.

[30] The affiliation is made visible with a graphical badge in the human-readable summaries of their licenses.

[31] Joshua Davis, PrayStation Hardrive (CD-ROM). System Design Limited, 2001.

[32] Matthew G. Kirschenbaum, *Mechanisms: New Media and the Forensic Imagination*. MIT Press, Cambridge MA, 2008, p. 54; Boris Čučković, and Hrvoje Stančić, "Open Source in Art: Originality, Art Process and Digital Preservation," in: *INFuture2009: Digital Resources and Knowledge Sharing, Second International Conference "The Future of Information Science" (INFuture)*, November 4–6, 2009, Proceedings, pp. 157–167.

[33] Mail from Joshua Davis to the author, June 8, 2012.

[34] Florian Cramer, "Peer-to-Peer Services: Transgressing the Archive (and Its Maladies?)" in: Franziska Nori (ed.), *adonnaM.mp3 – Filesharing, the Hidden Revolution in the Internet*, Museum für Angewandte Kunst (MAK), Frankfurt, 2003, available online at: http://www.digitalcraft.org/index.php?artikel_id=502, accessed 07/09/2012.

[35] Chun Lee, *Art Unlimited: An Investigation into Contemporary Digital Arts and the Free Software Movement*, PhD diss., Middlesex University, London, 2008, available online at: eprints.mdx.ac.uk/4963/1/ChunLeeThesis.pdf, accessed 07/09/2012.

[36] Open Art Network, "The Source License," 2004, available online at: <http://web.archive.org/web/20041208023918/http://three.org/openart/>, accessed 07/09/2012.

[37] The license prohibits commercial use.

[38] Available online at: <http://three.org/openart/license/index.html>, accessed 07/09/2012.

[39] Available online at: <http://artlibre.org/licence/lal/en>, accessed 07/09/2012.

[40] Valentin Villenave, "Re: Sources d'une œuvre," 2011, available online at: https://listes.april.org/wws/arc/copyleft_attitude/2011-10/msg00042.html, accessed 07/09/2012.

[41] Annet Dekker, and Jeroen van Mastrigt, "Serious Archiving: Preserving the Intangible by Capturing Processes," in: Annet Dekker (ed.), *archive2020: Sustainable Archiving of Born-digital Cultural Content*, Virtueel Platform, Amsterdam, 2010, pp. 79–87.

[42] Carl Gustav Jung, *Modern Man in Search of a Soul*, trans. W.S. Dell and Cary F. Baynes, Chapter "Psychology and Literature," new edition, Routledge, London, New York, 2001, pp. 155–176.

[43] Aymeric Mansoux, "The Sound of Network Topologies," in: *Neural*, no. 38, 2011, pp. 19–22.