

Affinity Video
Peter Westenberg

This article is a speculative journey through desires and considerations connected to online services for video. In writing it I have tried to stay close to my own practice as a videomaker. I choose to work with open source tools that support a free and participatory web environment, leaving the definition of these terms open for further discussion. From that angle I formulated some personal expectations that will serve as a backbone for this text. Although they might seem obvious at first glance, even on this basic level many existing platforms leave a lot to be desired for. First of all a platform for video should facilitate exchange of- and access to video files. It should provide tools and solutions for sharing work in collaborative processes. A platform should respect rights of users and let users retain control over their own video-material. A web platform should allow participation in the system.

As an artist I work with video within social contexts: I often collaborate with groups of people in artistic projects that investigate the notion of publicness. These projects can take the shape of explorations, guided tours or laboratory like spaces. A few years back I produced a series of weekly web streamings with inhabitants of a housing block. The shows we broadcasted varied from experimental documentaries to art performances, from a demonstration by the local Emotion Freedom Therapist to kids who organized their own Idols election, taking care of their own streaming.¹ On the broadcasting side, we worked with Quicktime broadcaster. On the receiving side numerous different softwares existed on a multitude of different types of computers in the neighborhood. The incompatibility issues were countless: missing plug ins, flaky bandwidth, uninstallable media players and so on. It became clear that the soft- and hardware we used to stream with, and the bits, processor power, screen resolutions and connection speeds on the receiving end, were as much part of the 'video work' as the content of the broadcasts. I became interested in working with open standards and started to work with open source software because it seemed to me that in situations like these, the core of the problems is often rooted in conflicting commercial software, pervaded with patents, temporary licenses, non-open standards and corporate codecs. In stead of continuing to react ad hoc to the problems of the moment, it made more sense to me to start using software which proposes long term sustainable solutions, even when this means that on the short term some of the work might become less comfortable.

¹ <http://www.worldwidewestwijk.nl>

World Wide Westwijk was part of 'Uit + Thuis Videomagazijn', a public workspace for video in a residential area in the Dutch city Vlaardingen which I ran for two years in the framework of the temporary cultural zone 'De Strip' initiated by artist Jeanne Van Heeswijk.

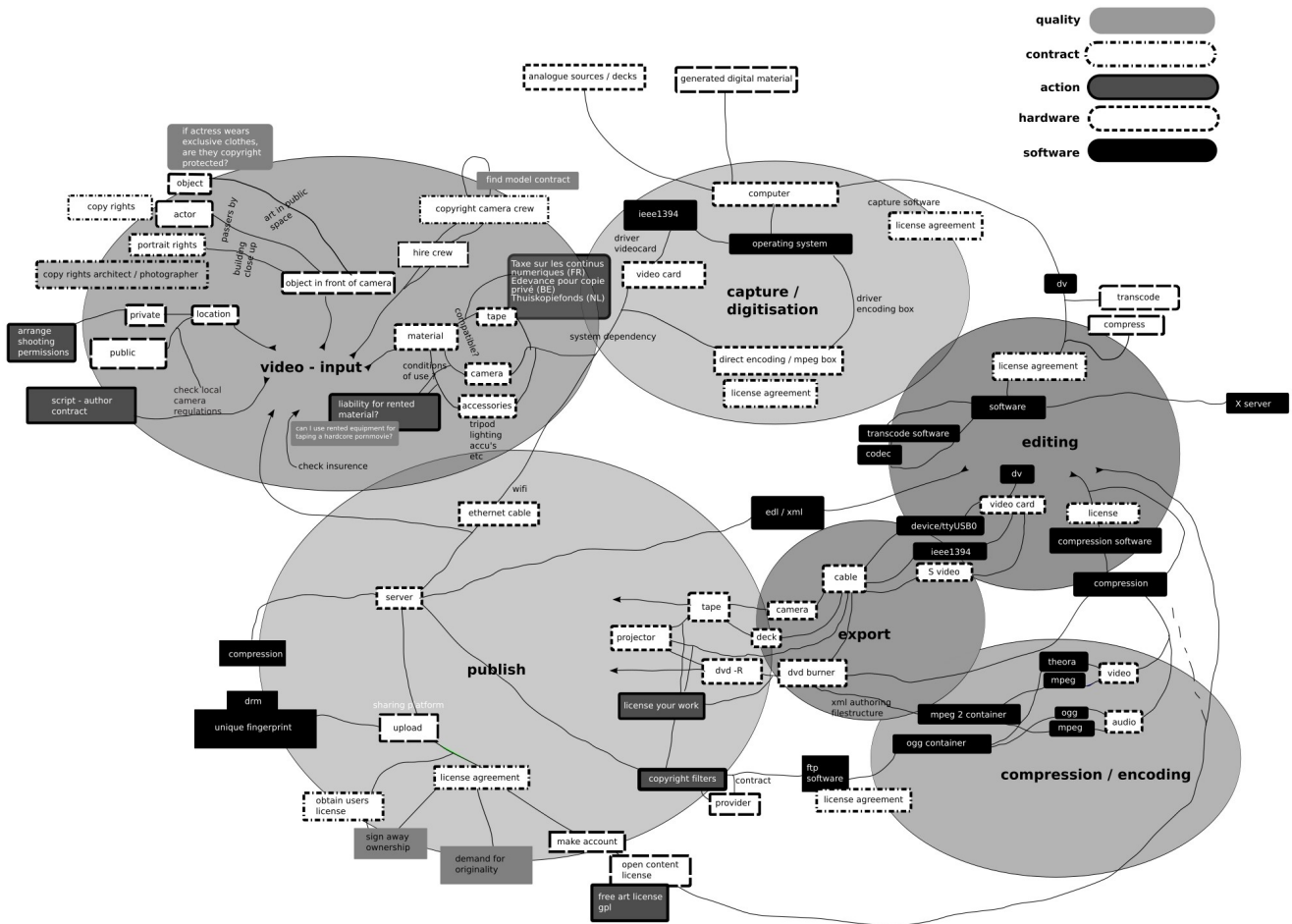


fig. 1: Workflow sketch.

To gain insight in how using non-proprietary software changes my practice, I am mapping my working process in a schema that lists all stages of work required for a video project; from shooting to publishing online. The aim of this drawing is not only to map existing routes, but also to try to imagine new avenues; to invent an infrastructure based on free software, open standards and protocols. A workflow consists of a chain of small tricks, techniques, tools, actions, movements. Once we have found our favorite way of working and repetitively execute these separate steps, we are beginning to embody them, fusing them into one comfortable fluent process that no longer feels artificial. But this comfort of 'knowing how things work' poses the risk of rendering us insensitive for other perspectives, for other ways of doing things.

When academic writers apply brackets, text mark ups and other types of annotations to refer to external sources, meanings and connotations, they break up the fluency of reading. The act of interrupting a comfortable stream of information helps us to remind that words are never neutral. Can we find ways to do a similar thing to our workflows? Can we re-validate inefficiencies, glitches and detours as valuable

"stumbling blocks" which help us to remind that using software is neither natural nor a neutral act?

Let's zoom into the part of the workflow that deals with putting video online. But first let me list the tools I use when working on a video-project: I work on a Linux Ubuntu operating system. To capture video I use the software Kino, for basic edits Kino and also Kdenlive, for complex editing I turn to Cinelerra. Editing sound is done with Audacity and Ardour, coding digital video with FFmpeg and Mencoder. VLC and Mplayer are my favorite mediaplayers. Icecast, DVgrab, ffmpeg2theora and Oggvorbis are piped together on the command line for streaming video² I use the open formats Ogg/Vorbis for audio and Theora for video.³

Exchange and access to video files

When you exchange files with others through the web you need to agree on a common protocol. A 'protocol' can be understood as a "type of controlling logic that operates outside institutional, governmental and corporate power"⁴. Decentralized systems built on protocols which have to be accepted by all the participants, a protocol is formal and has to be internalized by all users / machines. If there is no central authority in a system, control manifests itself in the protocol. The contradiction at the heart of a protocol is that it has to standardize in order to liberate. To put it more extreme: "It has to be fascistic and unilateral to be utopian"⁵.

When we exchange with friends, colleagues and like-minded our protocols of exchange are not only technical but also social: HTML, IP, P2P, HTTP, FTP, IEEE1394, mix with Trust, Friendship, Affinity, Communitarity. How do we recognize our friends online? By comparing our profiles? By following hyperlinks, are friends those who invite us to be part of their 'community' ?

Commercial services for video apply an elementary principle of magic, a basic skill known to every magician: they draw attention to something else than what they are actually doing. Their words and appearances suggest approachability, they welcome us to a community. We are promised new friends; we are being spoken to in the language of social softwares. But we have to remember that the core activity of all commercial companies, no matter what type of services they offer, is making money. In order to determine what type of platform we are dealing with we have to ask a few questions: Who runs the service and what is its true aim? Who controls the rules of the game? Who benefits most from the existence of the platform?

When I join an online service, I am asked to agree with its policies and conditions. The Terms of Use (or Terms of Service) document specifies the conditions for use of the website and the services it offers, comparable to a contract between house-owner and tenant. Next to this, Terms of Use documents often also specify how users are expected to behave among themselves. Nothing wrong with members of the same household or users arranging their communal life or web behaviour in a document. But does a landlord telling you what you can and cannot do fit the picture of a '(virtual) environment in which people can work together and can form (virtual) communities'?⁶

² A good startingpoint for tips on working with open source tools is: Flossmanuals.net
<http://en.flossmanuals.net/TheoraCookbook/FfmpegStreaming>

³ For specifications of open source codecs for audio and video see: www.xiph.org

⁴ Alexander Galloway, *Protocol: How Control Exists After Decentralization*, Cambridge, MIT press, 2004

⁵ Alexander Galloway, 2004

⁶ From Wikipedia: "Social software can be defined as software which enables the online interactions between people, which facilitates virtual relationships or creates virtual environments in which people can collaborate or built virtual communities." http://nl.wikipedia.org/wiki/Sociale_software.

Terms of Use documents, license agreements, marriage certificates, open content licenses, leases, and other contracts for use or cohabitation do not guarantee happy lives, friendship, love, success. They arrange forms of control or giving up control. When working on a collective project, such as the development of a piece of software, or the making of a film, collaboration⁷ is often a productive and fruitful necessity. When we work together with people we do not like, it is a good idea to describe our exchange protocol in a document.

An easy way to optimize exchange of files is to publish work under an open content license. Copyleft licenses such as the General Public License and the Free arts License (Licence Art Libre) enable our worst enemies to use our work and allows us to benefit from their work. By accepting reciprocity as a social procedure for regulating data transmissions, the term 'sharing' gains new meaning. In stead of utilizing 'real world' nostalgic familiar and friendly connotations of the word, as is often done by commercial video 'sharing' services, open content licenses infuse the act of sharing with an understanding of the characteristics of relationships in network environments. I will sign any contract that reads: You are allowed freedoms as long as you pass on those freedoms to others.⁸

I expect an online video service to give me access to the videofiles. Let's try to be a bit more precise in what we mean with 'access'. In order to be able to work with online video files I will need physical access, in the shape of a direct link to the file. I will need legal access: I need permission of the right holder(s) for copying, downloading, remixing etc. Having visual access, being able to see the files in a preview, index thumbnails or otherwise, would help greatly.

Generally speaking commercial services are not keen on providing direct access / links to files, more likely they will offer code to embed the video, or have you watch it through their own web interface. Avoiding responsibility for possible copyright infringements or other forms of potential abuse by users is less of an issue for platforms that require uploaded files to be under open content licenses⁹. Open content files found on Archive.org, Lulu.tv, TheoraSea.org and Blip.tv¹⁰ are downloadable and point to the licenses that specify the rights granted for further use.

An easy way to find legally accessible files is offered by CCsearch; a search engine developed by Creative Commons which looks specifically for CC licensed files which can legally be used, remixed, altered, sampled, copied.¹¹ Open content licenses do not protect my exclusive rights to the object, but they protect the rights of the object to be freely used and multiplied by everybody. Everybody means: including me. A digital object deserves to be copied, changed and redistributed, whether people like

⁷ Collaboration carries a connotation of working with the enemy, ... as Florian Schneider remarks in his text o, the subject. "It means working together with an agency with which one is not immediately connected."
<http://summit.kein.org/node/190>

⁸ From the GPL 3.0:
"To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others."
<http://www.gnu.org/licenses/gpl-3.0.txt>

⁹ Lawrence Liang, *A Guide To Open Content Licences*, Rotterdam, Piet Zwart Institute, 2004.
http://pzwart.wdka.hro.nl/mdr/research/liang/open_content_guide

¹⁰ Recently Blip.tv was voted best online video platform recently by the magazine PC World, leaving YouTube and other big companies way behind.
<http://www.pcworld.com/article/id,136101-page,1/article.html>

¹¹ The Creative Commons search engine can be found at: <http://search.creativecommons.org/>

each other or not. Attaching an open content license protects me from applying warm notions of human relationships as if they were reliable protocols for exchange, which they are not.

Most online services offer video's in a Flash Video format. If you do not want to equip your computer with proprietary software this can be tricky: FLV is not open and therefore not fully supported by open source software. Several existing initiatives aim to improve this situation: Web videos can be coded in Ogg/Theora, a format that can be played in browsers using the Java applet Cortado. By embedding this applet in a website, viewers can access the video without the need for a locally installed media player supporting the correct formats on their computers.¹²

The Gnu project Gnash is another promising alternative: an open source Flash player under heavy development which can be compiled for most operating systems and architectures.¹³

Sharing the work process

One of my favorite ways to exchange work in process is by relaying Edit Decision List. EDLs are lightweight text files, in which the decisions made during an editing process are saved. When several people working together all have access to the same source videos and use the same software, the EDL is the only thing they need to exchange to know of each others edits. The structure of EDLs such as the one exported by the video editing software Kino, can be easily understood and rewritten with a simple text editor.



fig. 2: A screenshot of a project in Kino.

¹² More info on Cortado: <http://www.flumotion.net/cortado/>

¹³ More info on Gnash: <http://www.gnu.org/software/gnash/>

```

<?xml version="1.0"?>
<smil xmlns="http://www.w3.org/2001/SMIL20/Language">
  <body>
    <seq>
      <video src="werk/hasselt/[Video]/7.dv" clipBegin="00:00:00.000" clipEnd="00:00:05.160"/>
    </seq>
    <seq>
      <video src="werk/hasselt/[Video]/6.dv" clipBegin="00:00:00.000" clipEnd="00:00:07.840"/>
    </seq>
    <seq>
      <video src="werk/hasselt/[Video]/10.dv" clipBegin="00:00:00.000" clipEnd="00:00:05.400"/>
    </seq>
  </body>
</smil>

```

fig. 3: The same Kino project saved as a SMIL formatted XML.

This is all great when working with a few colleagues, but it might be more exciting to integrate the distributive potential of the web in the decision making and production phase of a film project. Co-authorship on synopsis and scenario. Exchange of ideas, sketches, try outs. Remote filming. Distributed editing and decision making processes. That potential lies for instance in combining Content Managing Systems with scripts and the EDL output of video editing software.¹⁴ Using a CMS to produce bins, clips, edits and sequences, instead of an editing software that runs locally, replaces the authoritarian scheme of the responsible editor in chief with a more diverse, versatile and accessible (I won't say democratic) approach towards film making.

Imagine also the inclusion of wiki-power for collective scripting¹⁵ and the management skills of Version Control Systems¹⁶ for helping decide between concurring edits, commenting to commits and help negotiate final versions.

Control over your own video

To what extent do I keep control over my own work when I upload it to an online web service? To find out more I have to return to the Terms of Use document. Here we find who owns the material on the site and who will be allowed to use it under which conditions. In most cases the uploading party remains the formal owner of the work, s/he signs a license granting the platform certain rights; among others enabling them to show and view the work. This goes for all platforms, commercial or not, but each platform specifies conditions differently. In some cases granting rights to the platform is very

¹⁴ Echo Chamber Project is a collaborative documentary in development about the behaviour of American media in the days before the invasion of Iraq. It is experimenting with connecting the SMIL and XML generating functions of the CMS Drupal to Apple's Final Cut Pro's XML format.

<http://www.echochamberproject.com/collaborativefilmmaking>

Based on the Echo Chamber flowchart, Deptford.tv is currently developing a system involving only Free software: using the XML output of Open Source timeline editor Cinelerra.

<http://deptford.tv/>

¹⁵ An example of a movie project using wiki's for distributed scriptwriting is A Swarm of Angels.
<http://aswarmofangels.com/>

¹⁶ Version control systems are software programs that keep track of all work and all changes in a set of files, and allow several (potentially widely-separated) developers to collaborate. Examples of well known Version Control Systems are: CVS: Concurrent Versioning System: an open-source version control system.

SVN stands for Subversion; a version control system which is used to maintain current and historical versions of files such as source code, web pages, and documentation.

GIT is a distributed version control system focused on speed, effectivity and real-world usability on large projects.

close to donating your rights of ownership to the platform. Quoted from the Terms of Use of Facebook:¹⁷ “By posting User Content to any part of the Site, you automatically grant (...) to the Company an irrevocable, perpetual, non-exclusive, transferable, fully paid, worldwide license (with the right to sublicense) to use, copy, publicly perform, publicly display, reformat, translate, excerpt (in whole or in part) and distribute such User Content for any purpose on or in connection with the Site or the promotion thereof, to prepare derivative works of, or incorporate into other works, such User Content, and to grant and authorize sublicenses of the foregoing.”

Suppose I agree to such a license, and I click the Yes button, at that moment I also agree to a short sentence which can easily be overseen: “YouTube (or any other service) reserves the right to amend these Terms of Service at any time and without notice, and it is your responsibility to review these Terms of Service for any changes.” The Terms will also inform me that when major changes will be made to the document, the platform will notify me.

You rent a flat. The lease contract specifies that the landlord can change the contract at any time without notice. It is your responsibility to check back regularly to see if you still agree with the contract. If you do not agree with it, you have no way to influence its content and your only option is to end the arrangement. Which other contract specifying that sort of conditions would you click 'I agree' to? Marriage? Provider? Telephone? Mortgage? Online ordering? Oh yes, one other thing: ... How often do I have to re-read the Terms to see changes occur? How do I spot changes in these very elaborate documents? How do I distinguish minor changes from major changes? And why have I never been notified of such a major change?

In response to these questions, Alexandre Dulaunoy and Michael Noll developed GooDiff¹⁸: a system which records the history of changes made to legal documents of big companies including: Google, Yahoo, Ebay and Six Apart. GooDiff scans documents from services such as: Gmail, del.icio.us, Flickr, YouTube and Facebook. This tool allows us to see all changes that were made to the Terms of Use document of YouTube since GooDiff was launched in 2006. Within a year, more than 300 changes were made to this document, replacing literally every single word in the document to which I had eagerly clicked 'Yes I agree'. GooDiff paints a radically different image of the service than is suggested by YouTube itself. No longer can I believe its cosy, friendly, social words and (inter)face. YouTube turns out to be a two faced monster, showing an active policy of rewriting contracts, taking full commercial benefit of web 2.0 potential.

Participate in the system

Last on my shortlist of expectations is that I would like the platform to enable me to participate in its system. If a platform gives the impression of being a social tool, that suggestion should be supported and ratified by how its machinery operates and should not be contradicted by its legal and technical structure. Opening the source code is a way to give users an opportunity to verify that structure. Permitting others to develop new projects upon the source code and allowing changes and contributions to that code is a convincing participatory methodology which can be very beneficial to the project

¹⁷ Quoted from the version of the Terms of Use available from the Facebook website at time of writing.
<http://www.facebook.com/terms.php>

¹⁸ Homepage GooDiff: <http://www.goodiff.org/>. Read how the project was started on Michael Nolls blog:
<http://www.michael-noll.com/blog/2006/03/18/goodiff-project-upnrunning/>

itself: the history of the open source movement shows that developers feel challenged to improve interesting and valuable code.

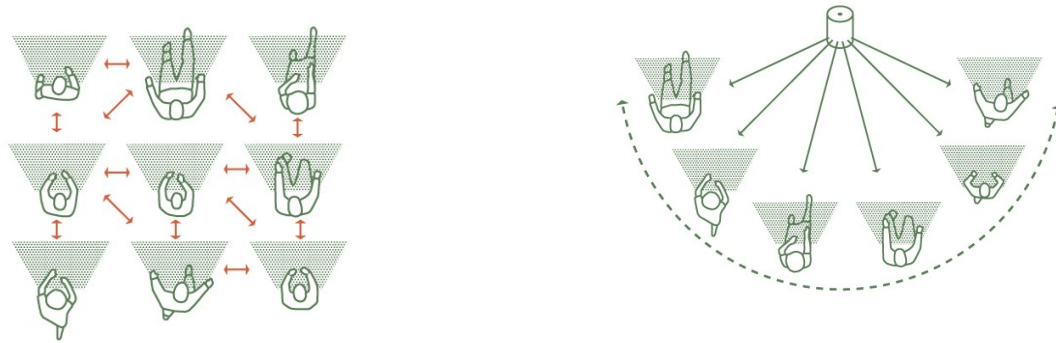


fig. 4: Left-P2P exchange, Right: Central control system

Applying exchange protocols such as P2P, seems an obvious way to structurally letting participants have a share in distributive platforms. The schematic representation of a P2P network indicates that data lives inbetween its nodes, it is received and made available by the nodes. Switching off one node does not make data inaccessible, increasing the amount of active nodes certainly makes exchange faster. The responsibility for powering the exchange operations lies with the participating nodes.

In a server / client model power and authority are not distributed, but centralized in the server.

Let's freely adapt some Foucauldian¹⁹ notions of control to a central server video service: Who owns the server, possesses control (the company defines the conditions). The user of the network is monitored but does not have capabilities to oversee the system. (User uploads merchandise for company to trade with) S/he is the object of information (company collects user profiles), but is not a partner in communication (corporate censorship). The server functions as a panopticum from which processes are being monitored and directed. Websites such as YouTube thrive on a conscious and permanent state of visibility in which users engage themselves, (flagging, commenting) confirming the automatic functioning of power.

Of course I am comparing two completely different entities: A prison model is not a server system, and YouTube is not a jail. But the resemblance between the diagram of a central server and the panopticum are too obvious to ignore. Now this makes me wonder: Is my desire to participate in a system sufficient? Aren't prisoners participating in the very system by which they are detained?

When reflecting on the censoring aspects of television, Pierre Bourdieu remarks that television producers and spectators are engaging in a symbolic, non physical form of violence that can exist

¹⁹Michel Foucault, *Discipline and Punish: the Birth of the Prison*, trans. Alan Sheridan, 1977, New York, Random House, pp. 195-228

"He is seen, but he does not see; he is the object of information, never a subject in communication. (...) Hence the major effect of the Panopticon: to induce in the inmate a state of conscious and permanent visibility that assures the automatic functioning of power. So to arrange things that the surveillance is permanent in its effects, even if it is discontinuous in its action; that the perfection of power should tend to render its actual exercise unnecessary. (...) In view of this, Bentham laid down the principle that power should be visible and unverifiable. Visible: the inmate will constantly have before his eyes the tall outline of the central tower from which he is spied upon. Unverifiable: the inmate must never know whether he is being looked at at any one moment; but he must be sure that he may always be so."

<http://foucault.info/documents/disciplineAndPunish/foucault.disciplineAndPunish.panOpticism.html>

thanks to the silent approval of those who are undergoing it and those who exercise it, but only when neither of the parties are consciously contributing.²⁰ According to Bourdieu, the price for performing on television is that you allow yourself to be censored. Among others, this has to do with three aspects of the television system: subjects are fixed, the conditions of communication are predetermined and the time you get to speak is limited. Let's draw an analogy with YouTube. Recently YouTube has put a limit on the uploaded clips. The uploaded content is framed by 'community guidelines' and can be flagged as 'inappropriate' by the 'community'. The conditions for communications are defined by the company.

Silently approving to a not very sympathetic system is definitely not what I am looking for. I have to be clearer in my desire: the system I want to engage in should itself be participatory. Before embarking on a 'participatory' project I have to learn to be more critical: if my participation is not answered by offering me for example a voice in the structure of the system, or a share in its benefices or revenues, my commitment is probably not valued as participation, but perceived as a gift. I should refuse to work with parties that make their systems look open and accessible when in fact they are fishing for input of clients without having to pay for it.

That is why I do not like using the term 'sharing' platform when talking about a YouTube type of service. To me, the term 'sharing' implies more than watching videos through a corporate website. It suggests taking a political standpoint by translating intentions of sharing to all aspects of your system, applying corresponding standards, protocols, terms and conditions. Our world is not a binary one. Relations and power structures are usually not transparent and clearly identifiable, which makes operating with clean hands virtually impossible. In preparing this article I have signed quite a few Terms of Use documents which I do not agree to at all. Despite the fact that our practices as cultural workers are embedded in a diffuse reality, I think it is worth while taking up the challenge of critically assessing systems we engage in, attempting to influence them for the better and to support the development of sustainable alternatives.

Credits:

This article frames in a long term practice based research on the use of FLOSS tools for design and artistic purposes by Constant; a Brussels based organisation for Art and Media. The weblogs Open Source Video and Open Source Publish offer good starting points for further reading on this subject. <http://www.constantvzw.org>

fig. 4: Illustration by Pierre Huyghebaert, Open Source Publish

© www.ospublish.constantvzw.org Licence Art Libre 1.3 (LAL 1.3) <http://artlibre.org/licence/lal>

© This work is free, you can copy, spread, and modify it under the terms of the Free Art License 1.3 (FAL 1.3): <http://artlibre.org/licence/lal/en/>

²⁰ Pierre Bourdieu, Sur la télévision, Parijs, Liber - Raisons d'Agir, 1996, pp. 17

Allen, Danny. 'Blip.tv Beta', *PC World*, october 2007, <http://www.pcworld.com/article/id,136101-page,1/article.html>

A Swarm of Angels, <http://aswarmofangels.com/>

Bourdieu, Pierre. *Sur la télévision*, Parijs, Liber - Raisons d'Agir, 1996, pp. 17

Creative Commons search engine, <http://search.creativecommons.org/>

Deptford.tv, <http://deptford.tv/>

Echo Chamber Project, <http://www.echochamberproject.com/collaborativefilmmaking>

Facebook, 'Terms of Use', <http://www.facebook.com/terms.php>

Flossmanuals.net, <http://en.flossmanuals.net/TheoraCookbook/FfmpegStreaming>

Foucault, Michel. *Discipline and Punish: the Birth of the Prison*, trans. Alan Sheridan, New York, Random House, pp. 195-228, 1977.

Galloway, Alexander R. *Protocol, How Control Exists After Decentralization*, Cambridge: MIT press, 2004

Gnu.org, 'GPL 3.0', <http://www.gnu.org/licenses/gpl-3.0.txt>

GooDiff, <http://www.goodiff.org/>, <http://www.michael-noll.com/blog/2006/03/18/goodiff-project-upnrunning/>

Liang, Lawrence. *A Guide To Open Content Licences*, Rotterdam, Piet Zwart Institute, 2004.

Schneider, Florian. 'Collaboration', *Summit: non aligned initiatives in education culture*, <http://summit.kein.org/node/190>

Wikipedia, 'Social software', http://nl.wikipedia.org/wiki/Sociale_software.

World Wide Westwijk, <http://www.worldwidewestwijk.nl>