Free software beyond radical politics: negotiations of creative and craft autonomy in digital visual media production

Draft of article under revision. Not to be cited without permission.
Version: 15 Jan 2016
Author: Julia Velkova

Abstract
Free and open source software development, and the technological practices of hackers have been broadly recognized as fundamental for the formation of political cultures and fostering democracy in the digital mediascape. This article seeks to broaden the scope of knowledge about the role of free software for political engagement by discussing its relevance in the practices of other actors, beyond activists and hackers. The study explores its role and uses in the practices of media creators such as free-lancing digital artists, animators and technicians who work in various roles for the contemporary digital visual media industries. Drawing on ethnographically collected material about the media uses of three popular free software tools, Blender for 3D animation and sculpting, Synfig for 2D vector animation and Krita for digital painting, the study shows that free software in the context of media work contexts enables creators to notably extend their sense of creative autonomy, skills and creative expressivity, yet paradoxically this sense does not lead to resistence or critical engagement, but strengthens even more some of the imaginaries and allure that the creative industries have while not responding to some of its major flows such as precarity of labor.

Keywords
digital visual media, free and open source software, material politics, craft autonomy, media industries

1. Introduction

Media practices, such as free and open source software development, and the technological experiments of hackers have been broadly recognized as fundamental for the formation of political cultures and fostering democracy in the digital mediascape. Their relevance for political agency today is expressed through the ability of actors who take part in these practices to reconfigure "the material politics of cultural action" (Coleman, 2013, p. 185), primarily through introducing new “entities” into the world (Söderberg, 2011, p. 23) and by making them public (Kelty, 2008). These entities can be anything from material objects that take the form of open hardware such as self-made 3D printers (Söderberg, 2014), writing an independent operating system (Coleman, 2013; Kelty, 2008), or of experimenting with digital aesthetics and critical art projects (Morgan, 2013). In all these cases, politics is practiced primarily through creatively engaging with building, modifying and maintaining technological equipment, an activity that both resembles public demonstrations of technical expertise and a way of arguing about technology, with and through it (Kelty, 2008, p. 5; see also Kubitschko, 2015).

The entities that are brought to the world can in some cases be taken by other individuals or social groups who either embed them in their own practices directly, or "modulate" and “translate” them by inducing them with own meaning and repurposing them for other goals, often with the aim to express critique to different aspects of the contemporary capitalist media systems. Among the most prominent examples of such translations is the embracement of free software technologies by activists such as in the case of Indymedia in order to make a case for alternative journalism (Atton, 2007; Coleman, 2013, p. 193; Lievrouw, 2011); its uses by the open data movement (Baack, 2015); for creating alternative social media networks (Gehl, 2015), or for expressing civic disobedience in novel albeit disruptive forms such as through the pranks and hoaxes of the Anonymous (Coleman, 2015). Free software also plays an important role among artistic minorities and digital artists where it is embraced in order to criticize the ownership conditions of digital "materials" or software programs available to consumers while experimenting with digital aesthetics and the materiality of digital technologies (Morgan, 2013).

Despite the richness and important insights of these studies, one of their limitations is their focus on the uses of free software for predominantly radical, activist or explicitly political purposes by social movements and creative minorities. However, free and open source software has been increasingly integrated in other practices, such as those of corporate technological manufacturers, ranging from IBM and Google to large computer graphics corporations such as Disney and Pixar. With regards to these developments, Christopher Kelty (2013) expressed a concern that they threaten to make free software's potential "sterile", by being equally easily put in use for fusing a counter-critical power that strenghtens monopolies rather than criticising them; it could also be used instrumentally for individual technical career advancement: "for a great many software developers, toiling as they do in the richer veins of freelance precarity, it meant not having to rebuild the same damn thing over and over again with every upward career move". Kelty concludes that "As open source becomes an instrumentalized kind of politics, the possibility of new beginnings fades."

The extension of the scope of uses and applications of free software development beyond radical politics requires to study these forms in order to understand their role for cultural expressions and sense-making practices in a broader range of actors in the media and political landscape today.

This article seeks to broaden the scope of knowledge about the role of free software development in the politics of digital media production by discussing its relevance and value for other actors, beyond activists and hackers. In particular, it explores its role and uses in what could be seen as a middle range of uses, those that stand between the two polarities outlined by most of the studies where on the one extreme stand hackers, activists and critical artists, and on the other end stand big technological and media corporations. The way I do so is by focusing on the uses of free software in the work practices of individual computer graphics artists, designers and animators who work in a wide range of roles in small advertising agencies, visual effects and computer game and films production companies. The empirical focus here is therefore on micro, individual perspectives yet it is of relevance in order to initiate a broader discussion about the integration of free software in the production dynamics of the media industries.

The material for this study comes from a bigger research project on the media practices of two free software computer graphics communities, Blender for 3D animation and Synfig for 2D animation, and to a minor degree Krita for digital painting. The data has been collected through a two-year long multi-sited ethnography of the engagements with free software of digital media artists, designers, and programmers in different digital media productions, commercial and for public benefit, who are in the process of developing careers in the media industries. The ethnographic data is complemented with qualitative interviews with 35 visual media artists and developers, numerous informal conversations with other digital media producers who use free software in their practices, as well as online publications written by or about them. An immediate limitation which follows from this material is that it does not address the perspectives of visual media producers who use popular proprietary packages in their work. Nevertheless, I believe that for the purposes of this article the material is indicative enough in order to discuss some complexities and imaginaries of using free software in more "mainstream" or ubiquitous forms of creative practice.

In order to understand what meaning can free software have for the practices of smaller and individual enterprises and practices in the field of digital visual media production, in the first section of this article I outline briefly the work context in this field of media production drawing primarily on literature on work in and for the contemporary creative industries. The particular context in which free software becomes relevant is then discussed by drawing on Howard Becker's work on "Art Worlds"(1982) and the role of materiality in creative practice. With its help I introduce the emergence of the three by day major free software programs that have found use among digital media producers, Blender, Synfig, and Krita, which I extend with a distinction of three major types of their uses among broader range of users. The argument that I develop is that free software in these contexts enables creators to notably extend their sense of creative autonomy, skills and creative expressivity, yet paradoxically this sense does not lead to resistence or critical engagement, but strengthens even more some of the imaginaries and allure that the creative industries have while not responding to some of its major flows such as precarity of labor.

2. Work in the digital media industries: between autonomy and control

Studies of the work of artists, craft workers and not least fans of the computer games industry (Deuze, 2007; Deuze, Martin, & Allen, 2007), television production, music recording and magazine publishing (Banks, 2010b; Hesmondhalgh & Baker, 2010), radio production (Stiernstedt, 2013), as well as the Hollywood animation film industry (Stahl, 2010) have indicated an inherent tension between autonomy and the control of creative labor in these industries as they are embedded in capitalist, neoliberal systems of production. On one hand, work in these industries carries a strong allure to young people and creators which comes from the promise of work of greater social status, autonomy, personal expression, flexibility and self-actualisation (Mayer, 2014). In order to stimulate the creativity of their workers, many media companies adopt anti-corporate work culture and on occasions enables creators to develop the reputation of being an "auteur" (Deuze et al., 2007), a celebrity (Hesmondhalgh, 2009), or a person with a broader public recognition (Mayer, 2014). At the same time, the organisational frameworks of production are dependent on constant rationalisation and effectivisation of labor in order to accelerate production and reduce costs, thus constraining the autonomy of creators in order to adjust creative works to market demands. Respectively, all of the studies mentioned above point that the work conditions in these industries make the work of artists, designers, crafts people and other "creatives" ever more precarious; by transferring ever greater responsibilities for personal artistic and technical skill development on individual creators, while putting creators at one remove from their creations by embedding them in institutions of employment and regulatory systems of intellectual property thereby converting them into an object of value extraction.

A question that these accounts raise is why do media producers and creative workers not respond more systematically and explicitly to these tensions.

One reason, according to sociologist of work, Ursula Huws (2014, pp. 101–125) is that control can happen in multiple ways and dimensions in creative work, and the real and illusionary senses of autonomy absorb resistance or make it too complicated, and at times even controversial. For example, Huws points out that media workers often experience a strong identification with the objects that they create, and a strong sense of self-fulfillment, a feeling which creates an illusory sense of autonomy which remains even after their work has been sold, and the intellectual property rights been transferred. Creative work also often carries a meaning, an ideological content or potential for social impact which puts creators in front of ethical choices about how their work is carried out (p. 112). An emphasis on the extent of personal meanings and affective relationships that creative work produces has been articulated more recently in great detail by the arguments of feminist critique of digital labor (Jarrett, 2016), and cultural studies perspectives on the media industries (see for example Bolin, 2011).

Yet, as Huws emphasises, these personal meanings and sensibilities are under a constant pressure by different forms of control and constraints that exist at multiple levels; at first place by the art-commerce relationship, but then also by the constraintsimposed by the market, by clients, by existing formal and informal rules, forms of management based on results or peer-control. All these forms of control are also in a perpetual change, each of which brings about some forms of resistence. Yet, on a larger scale, the multiplity of levels and ways at which autonomy can simultaneously be experiened and constrained hampers broader resistence. Individual creators develop strategies to accommodate the tensions between autonomy and control, creation and expropriation of their work through what Banks (2010a) and Ward (Ward, 2015, p. 215) refer to as forms of "negotiated autonomy". The latter refers to processes of creating subjective meanings in creative practice in a "quotidian struggle to try to mediate, manage or reconcile the varied opportunities and constraints of the art—commerce relation", in which the primary concern is finding meaningful self-expression within, rather than by directly confronting capitalism (Banks, 2010a, p. 262).

In the contexts outlined above, free software development has been acknowledged to have a potential for change in terms of offering more efficient and less alienating ways of organising and managing work (Benkler, 2006; Hardt & Negri, 2009; Hesmondhalgh & Baker, 2010). Yet, as the concerns expressed by Kelty and mentioned in the introduction suggest, these ways have been embraced most successfully by large commercial actors in the sphere of media production, and in result act as ways to strenghten monopolies rather than bringing much change.

I suggest though that free software could play a role at another dimension of creative practice, not so much in terms of organising production, but in terms of fostering the creative imaginaries of media producers at more individual level.

Creative practice is inherently embedded in materiality and the circles of its production and distribution. In this sense, creators of media are not only integrated in structures of employment, nation state politics, or networks of peers, but also in the specific logics of technology with which they daily interact and in which they are embedded at multiple levels. The sociology of art proposed by Howard Becker (1982) offers some insights in how to understand these entanglements in relation to autonomy. In his discussion of art as collective action, Becker emphasises that the choice of materials of creators affects the work that they do (p. 71). Materiality, or tools and materials, form a crucial part of the production of artistic works: “Musical instruments, paints and canvas, dancers' shoes and costumes, cameras and film - all these have to be made and made available to the people who use them to produce art works” (Becker, 1982, p. 3)

In the case of producing specialised media, such as for example visual media, creators need materials that are designed and manufacturerd specially for them. Becker argues that since the manufacturing of specialised items is so technical a specialty, the artists who use them cannot in most cases produce the items themselves. At the same time, he points out, that manufacturers try to be sensitive to the needs of the creators of this medium and their products tend to satisfy most of the workers in the medium, but they may fail to satisfy those who try to innovate in the medium: "How much conventional materials constrain an artist depends on how monopolistic the market is", he argues (p. 73). The fewer the manufacturers dominate the market, the more insensitive they get to what artistic minorities want or need (73). Under the threat of discontinuing the material against which creators have developed their skill or practice, in case of wanting more than the available materials can provide or in the case of lack of available materials, creative minorities can revert to the craft of making their own materials, or of customising existing ones.

It is in these contexts, and rather pragmatic considerations about individual ways to develop their creative practice that free software emerges as most valued and used among media producers and aspiring media professionals of the media industries. I illustrate this point through a brief review of the emergence of three currently popular among media creators visual media tools, Blender, Synfig and Krita, and then present three major ways in which they have been further embraced and developed by other media producers.

3. Crafting technical autonomy: The Blender, Synfig and Krita free software projects

Blender, a program for 3D animation and sculpting; Krita for digital painting, and Synfig for 2D vector animation are computer programs that represent the non-proprietary but licensed software alternatives for computer graphics and animation manipulation such as 3D Studio Max, Photoshop Element, Adobe After Effects, Anime Studio and Maya. These programs are used today to a greater or lesser extent in a broad range of industries and media practices. The usage ranges from the more experimental type such as conceptual art, or making designs for 3D printing, to ones where the programs are used for the production of comic books, illustrations, special effects, games, animation, and simulations.

The 3D animation software Blender and the 2D Synfig were initiated by two different industrial designers, one living in Europe and the other one in the US, who had ambitions to make large-scale animation projects of Hollywood class. Despite having notable differences in their focus of specialisation, and being incepted at different points of time, with Blender having roots in the late 1980s and Synfig in the mid-1990s, both have been conceived as in-house programs developed within two small commercial animation studios. After facing a bankruptcy in the early 2000:ds, both projects emerged as free and open source software through very particular processes of de-commodification (for more details, see Author,2015; Neus 2002).

The reasons behind starting independent software projects for computer graphics were related to experiences of strong material constraints to fulfill a creative impulse. In the case of Blender, the need to start developing an own program, even prior to its de-commodification emerged from the ambition of its founder, Ton Roosendaal, to align with the industrial practices of 3D technological development:

> "3D is specialist...it is so specialist....any big studio who does animation – or visual effects – they depend for the most of it on their own, in-house software development. They are not going to buy all their applications – and even when they buy some stuff, they want to have the code. Because they can't depend on a software, submit a bug, then wait for two weeks for a bug fix to come in while a thousand people are waiting, right? That's kind of....at that level your IT, your information systems have to be under control...." (Ton Roosendaal, interview, 2014)

Blender was developed by the wish of its creator to have complete control over the development, changes and possible extensions of a computer program, "a digital tool", that would enable its creator to adapt it and mould it to his own needs. Until the mid-1990s, software for computer graphics development used to be seen by digital media producers as an add-on to a very expensive hardware that they needed to invest in. The computer industry restructured in the end of the 1990s and with computing power getting cheaper and more ubiquitous, companies started developing business models around selling specialised computer software for digital media production. In the course of this market reconfiguration, Blender's creator re-licensed his program as free software, in order to continue developing it and yet preserve its possibilities to mould the main tool of his work: "open source is about developing your own software. So the best model [to develop computer graphics]....okay, not the best, the Blender open source model is the in-house software model".

Similar concerns drove the development of Synfig. Its founder Robert Quattlebaum. wanted to rationalize one of the most laborous tasks in 2D animation creation, tweening, and adapt the software to his own creative ambitions:

> Our goal was to write a tool that could be used for the production of feature-film quality 2D animation...In traditional animation, the senior animators use the storyboards to create the keyframes for each shot. The junior animators then use these keyframes as guides for making all of the frames in between--which is called tweening. Tweening is a time-consuming and labor-intensive (and thus expensive) process. However, it is also rather mechanical. So that was the original idea from day one---the elimination of the tweening process... While Synfig has been used in production, the animators using it had the benefit of having the primary developer sitting behind them. That counts for a lot. (OS News, 2006)

After its de-commodification in the mid-2000ds, and conversion into a free software project, Synfig's development was driven forward primarily by one self-taught animator from the city of Gorno-Altaysk in Southern Siberia, Russia. For him, Synfig, represented a technology that with some further development could fulfill his large-scale creative ideas of making a feature-length animation film. In search for style, his work had started with proprietary packages such as 3D Studio Max, but after some time he experienced a limitation in scale: "the more I was complicating a scene, the less controllable it was becoming...". He moved to experiment with Blender, an experience he describes as largely affective: "What shocked me in Blender the first time I used it was that it had layers...layers existed in many other types of programs at that time, but not in 3D...this was so daring, to do layers in a 3D program, I had never seen such thing before". Yet, he realised that for his own purposes there would be needed very substantial hardware investments which were not possible to make in his context. Instead, he invested his effort in specialising further in 2D animation that required less computer power, in parallel with still using proprietary packages until their development frameworks collided with his own work process. The manufacturer developing the 2D animation program he was using at the time, Moho, discontinued its development under Linux, which had gradually become Konstantin's main platform of work. The impossibility to use his program of choice under his operating system of choice caused a great sense of anxiety: “I liked that everything there was under my control. But nobody was supporting it...then I realised what dependencies was proprietary software creating. It is not about the cost, it is about the dependency". Since then he focused his efforts on studying and developing Synfig which he integrated at the core of his work practice as an extra-curriculum teacher of animation in his town, a free-lancing animator of commercials and educational material, and in his independent animation projects. Practicing in this way, workwise, a form of negotiated autonomy, the choice to go about Synfig came out of explicitly pragmatic concerns related to the possibilities to work within frameworks of own making and under his own control. This form of engagement with technology has been referred to, in the contexts of free software development and hacker cultures, as practicing "craft autonomy" (Coleman, forthcoming), one that fosters skill and expertise, but also sensibilities similar to pre-industrial, craft like engagement with technology.

Despite the freedom of creative expression which developing own, autonomous technologies and media production infrastructures granted to its creators, it also constrained their autonomy in new ways. In order to fulfill their ideas they needed to cooperate with more people who would also be able to work with the same technologies. In these cases both creators faced the constraint that instead of developing art projects, they needed to develop frameworks to train or convince other people to use these technologies. There is no space in this article to discuss in detail the strategies employed in these cases, but generally the approaches they embraced were to convince more creators through practice, training material distribution and recruitment of artists that these technologies were worth learning (they are partially addressed in Author, forthcoming). As Becker (1982, p. 74) points out, when creators go about developing own materials, they need to spend time in developing their material precursors and knowledge frameworks instead of dedicating their work on making art.

Blender succeeded in this endeavour to a greater scale than Synfig. After almost a decade of intense development and multiple versions, the computer program has become embraced by more free-lancing animators and creators who work for the media industries. Before moving on to discussing how it is used and valued in these contexts, it should be mentioned that even free software programs developed by hackers from within the free software community could get popularity and get integrated in the commercial work practices of media producers.

A notable example is the emergence of Krita, a program for digital painting on tablets. It started as a project of two hackers one of whom faced the problem of not having a suitable program to draw on a tablet using non-proprietary tools. In its initial years Krita evolved in tight connection to the free software movement, but after several years of struggles for its place and existence with no progress in functionality or broader practical use, one of the hackers refocused its development towards satisfying the needs of professional free-lancing illustrators. As he admits, this involved a change in mentality:

> It's fun to write software the right way...and in free software nobody minds how long you will take, as long as you have gotten no users. So we had to get out of that mentality and into the mentality of - we are making application that will make users happy...We actually lost one of our developers in that process, because he is more academically interested and he wants to figure out new approaches to new problems, but not finish up and polish his work. That's fine, but we wanted to have a polished application.

The next section discusses some of the ways in which digital media artists make sense of these tools, and how the integration of their uses in the practices of free lancers, animators, etc for the creative industries drives further their technological development.

4. Sensibilities of craft

Every media creator has a unique work process. The more they develop their skill, the stronger connection is established to the tools they use, as this skill is shaped through practice which is anchored in the materialities of technologies, even in the case of digital ones. For animators and digital artists who work with computer graphics, the software that they use and its continuous development plays a crucial role for their skill development, one that they can sell on the work market, but also, as argued in the previous section, triggers largely positive sensibilities of craft-like autonomous work.

Hjalti, an animator from Iceland who has been working for many years in the advertising industry encountered Blender by chance after many years of use of the popular package 3D Studio Max. He adopted Blender in his practice by chance, out of the wish to collaborate on a commercial campaign with a colleague of his who had it as a tool of his choice. He discusses his initial experience of learning the new tool in his practice as an agony that has been worth it:

> I was throwing my keyboard at the screen for the first couple of weeks or whatever, but once you get over it, you start to realise why it makes sense. Why pressing G is already moving an object... instead of like having a widget that you press on ...it's because it's faster. It just cuts a lot of steps out of the way. Which adds up. So you start doing things a little faster. And smoother. And then of coruse you can customise everything you want, now after Blender 2.5 Which I do, a lot.

What Blender brought to his work practice was understood initially in terms of efficiency, of speeding up the work process. Later versions of the program and him getting more experienced with it developed gradually into a sense of craft autonomy, one that emerges through the small details that constitute important parts of the creative practice of media producers. In Hjalti's creative practice, such a detail is the position of his hands. He wants to keep his hands static while working. Yet, whenever he needs to change perspectives on the screen while animating, the program interface requires him to move his hand to the keypad on the right side of his keyboard. He feels that this is slowing him down, breaking his concentration:

> I am doing it every 10 seconds. And take 1 second to let go of my mouse, I am losing valuable time, you know, after 15 hours or whatever... and it also breaks your concentration. Because your eye, your thought process has to go into that motion, instead of just keep going, doing what you are supposed to be doing.

After version 2.5 of Blender it was easier for its the users to customise their work processes to a great degree. Hjalti used this possibility to assign his own commands in such a way that he would no longer need to move his hands away from the keyboard while working. Such a seemingly minor detail was very important for him for experiencing a sense of craft:

> That's when it becomes really beautiful. When the tool itself doesn't become a hurdle, you are just doing something, and its an extension of you... So you can do something, you can adjust something, it's intutive. It is mussle memory. Which is really awesome.

Of course, possibilities for customization exist in other software. Yet the degree to which this is possible through free software depends more on the individual media creators' technical skill and creative ideas and less on the production frameworks and affordances set by the their manufacturers. For example, one free-lancing animator and one technical artist from Costa Rica worked in the spring of 2014 to complete a 4-seconds long shot for the teaser of a larger animation film project. The shot should show a green caterpillar blinking. The animator wanted the pupils of the caterpillar to resemble not that of an insect but that of the main character of the animation film. They were using Blender for this production task, and they found out that it did not have the technical capacity to animate what they needed in the way they wanted in order to create the desired effect. The technical artist came up with a concept of how the problem could be solved, and delved into the code of the program: "I started hacking a python script to automate this ^_^. At about 3:00am it actually worked!'. He shared the script and the technical details online with the following comment: "Beware it's a production script and as such it doesn't have a nice UI or anything and you might need to change a couple of names in the first few lines :)". In this case, the process of making animation has been very similar to hacking. Indeed, hackers, artists and free software developers have come to be described as "craftspeople" who have resisted to the general decline of craft in the Western mainstream economies that came with the dominance of the rationalisation of labor and the dominance of Fordist styles of production (Coleman, forthcoming). A common metaphor which was frequently used among the digital artists who were interviewed was to compare working with free software as being similar to the work of painters from pre-industrial craft production: "It is more like the old painters who made their paint themselves. Mixing the ingredients and building their paint themselves", one comic illustrator explained. The possibilities to craft and mould their own tools of work blurs in these cases the separation between art and craft, techne and poiesis:

> "Free software matches very good with the artistic idea because no artist wants to be locked into what they can do - a lot of the process of making art is about making the tools." (Bassam Kurdali, Tube open-source animation project).

The examples above occur, however, in very specific contexts, such as when the digital tool that is used is mature enough so that it requires just a little extension, and there is a person available nearby with programming skills who could quickly implement the idea in practice. In a certain sense these situations represent an ideal case, when in most other cases media creators need to find other ways to mould their tools, such as when they are unable to program. The media creators interviewed had two other well established strategies of how to solve these problems when using Blender, Synfig, Krita or combinations of them, namely to hire someone with technical skills to develop the desired feature for them; or to motivate some of the main developers of the programs to implement their features. In some cases, combinations of both, including coding and hacking the programs were applied in order to get the technical features needed. These strategies are discussed in the next section.

5. Shaping and adapting digital tools

In the case of Blender, the technical possibility to extend the software itself emerged as a result of a technical artist hiring a developer to program such a functionality based on a momentary need of a small company he used to work for. The company had been making substantial financial investments in specialized software for 3D manipulation for its own media production needs when the technical artist attempted to find a less costly alternative that could satisfy the production needs of the company: "I got Blender and I started extending it", he remembers. He admits that his programming skills were not good, so he hired a programmer from Montreal in Canada to come to Australia to to do the extension for him:

> Well, no, I didn't know how to program, like – I was, I was artist, so – I was okay, making stuff with mouse. But I knew some programmers so I got them to program...I hired them to program. I had one of the Blender developers come over to my house, doing internship with me, so I got him to program so it was like – the artist and developer thing happening. Which is really cool, I saw that happen myself... (technical artist, interview 2014)

The functional extensions made at that time entered the core of Blender, and made it possible for other creators, such as the animators from Costa Rica to make smaller individual extensions further. Much of the development of features gets paid by different small media production companies, being an example that free software is not necessarily anti-capitalist or anti-commercial, but its value comes from the infinite range of possibilities of its customisation.

Another strategy applied is by paying directly to a developer from within the specific program's developer community. A free-lancing animator from Sweden does cut-out animation and uses a combination of proprietary and free software tools in his work. He works on animations for the advertising and film industries in Sweden. He admits that he does not program or code, but could pay others or try to push the development of software projects in the direction that he needs. For some time he has been trying to replace a proprietary software package that he uses, Animé Studio, with the free software Synfig, by pushing its features development. His principle is generally to donate, as he says, 3-4 per cent of his income from commercial projects to the free software projects he uses, but occasionally he makes higher donations in order to put a feature he wants on priority for development. Once he payed a few hundred euros in order for the Synfig team to develop a feature that he needed faster than planned, he remembers. The possibilities to directly influence the development of the tools is what is valued highest by creators. It comes though at a cost:

> if you are going to create a file of specific format that is for a specific distrbution format - there are commercial tools for that, click on a button and get the file but with the open source tools you need to spend a day researching the specifications of that format and doing a script for that...so it can take some more time, but I think it is worth it... You need to research quite a lot to get stuff done.

A third approach which media producers use in adapting free software tools to their own practices is by requesting features and extensions directly from their main developers. These interactions lead to many tensions and conflicts. In some cases feature requests are welcomed and fulfilled, but in most cases they are ignored. As one developer admits:

> we get far far more requests then we even have time to read. also, these requests vary in quality. people may explain features in detail, which we already have....people ask for very specific stuff you understand... "Im using Blender for an interactive blah blah and Its draw modes don't work for me because...etc" ...people who use blender for 10min and don't like color also post.

The suggestions and discussions of features among artists and how things should be can get substantial size. A debate about the user interface of Blender that started in 2013 features by now 1661 comments containing various proposals for modifications and wishes from artists of how the program should look like and behave. The volume of the discussion points at the importance that media creators give to the features of their tools, and the convenience to work with them. Yet, the multiplicity of wishes raises the need of their management, to the extent that they are directed towards a small group of core developers who decide whether to implement them or not. In this case artists need to negotiate and argue for their features, and convince a developer of the value of each proposal. Such dynamics exists in other free software programs too. A digital painter, and illustrator, requested a feature from the program Krita through an extensive proposal described in detail on his website, and forwarded to the developers of the program. The proposal illustrated visually the problem in a real work scenario, the envisioned solution by the artist followed by proposals for further customizations and two examples of other usage cases. The document concludes in the spirit of an academic paper: "I studied many 'perspective features' on other 2D softwares and studied videos tutorials to produce this article. I hope this document will inspire or help (or even only influence a bit) the development of a solid perspective tool "[1]. This example has been later used by Krita developers to show how features requests should ideally look like. It suggests that the developers of these programs expect a lot of knowledge from the user about the program, and a solid argumentation for the case and its applicability beyond the individual work practice, so that it is of value for more people.

These forms of negotiating the paths of development of the main tools in the work of media creators put other constraints to autonomy, by forcing media creators to integrate deeper into the dynamics of interpersonal social relations and rhetorics of arguing in order to pursue their goals of craft autonomy and technological development.

6. Conclusions

User engagements with technology are caught up in a constant tension between control and powerlessness, between freedom and dependency, with networks, devices and software embodying different potentialities for action (Paasonen, 2014). The empirical examples discussed in this study indicate that media producers who use, modify, repurpose and study free software for the purposes of producing digital visual content for the media industries, or of similar aesthetics, do so largely because of pragmatic rather than ideological concerns. They do not embrace free software in their practices in order to express critique, defend democratic values, or develop creative alternative spheres of media production. Rather, they do so largely in order to negotiate a form of autonomy, craft autonomy, that allows them to improve their skill, style and expressivity in more craft-like manner, one that resembles an attitude described by Peter Dormer as: "you get the best out of the computer and its software if you are able to drive the tool rather than being driven by it" (Dormer, 1997, p. 146). In these cases, creative or craft autonomy is essentially about influencing the development of tools in a way that benefits one's individual work practice rather than having broader ideals or ambitions about social change. Free software, admittedly, offsets in these cases some of the negative effects of the increasing move of responsibility for skill and ideas development onto individual creators; yet it does not reify among its users the opposition between commercial or not, capitalist production or not. It is rather a means to an end. Paradoxically, while increasing the autonomy of media workers to craft their own work and develop their skill elevating it to industry standards of efficiency and aeshtetics, these craft-like engagements with technologies also further increase the allure of the media industries to which creators see that they could contribute with ever more original expressivity and increased efficiency. This paradox ultimately illuminates yet one more dimension of negotiating autonomy and control in creative work, and one more way in which media creators respond to changes of technological frameworks and rationalisation of production.

This study showed that free software is highly relevant for more groups beyond hackers, critical artists and social movements, yet for other purposes than those that it has traditionally been associated with. The limited scope of the study, to three specific free software programs and their uses among digital visual media creators raises the need to see what other imaginaries can free software development trigger within other technological approaches and practices, and in other contexts of different scale, such as in corporate contexts, or in other spheres of digital media production. It also shows the need to see how creators who keep working with more mainstream and proprietary technological frameworks negotiate their technical and craft autonomy, and skill development, which future studies can fruitfully engage in.

References

Atton, C. (2007). Alternative media in practice: The Indymedia network. In K. Coyer, T. Dowmunt, & A. Fountain (Eds.), *The alternative media handbook* (pp. 71–78). London ; New York: Routledge.

Author (forthcoming)

Author (2015)

Baack, S. (2015). Datafication and empowerment: How the open data movement re-articulates notions of democracy, participation, and journalism. *Big Data & Society*, *2*(2). http://doi.org/10.1177/2053951715594634

[1] See see http://www.davidrevoy.com/article159/design-ideas-for-a-new-krita-perspective-tool

Banks, M. (2010a). Autonomy Guaranteed? Cultural Work and the ‘Art–Commerce Relation’. *Journal for Cultural Research*, *14*(3), 251–269. http://doi.org/10.1080/14797581003791487

Banks, M. (2010b). Craft labour and creative industries. *International Journal of Cultural Policy*, *16*(3), 305–321. http://doi.org/10.1080/10286630903055885

Becker, H. S. (1982). *Art worlds*. Berkeley, Calif.; London: University of California Press.

Benkler, Y. (2006). *The wealth of networks how social production transforms markets and freedom*. New Haven: Yale University Press.

Bolin, G. (2011). *Value and the media: cultural production and consumption in digital markets*. Farnham ; Burlington, VT: Ashgate.

Coleman, G. (Forthcoming). Hackers. In B. Peters (Ed.), *Keywords: A Vocabulary of Information Society and Culture*. Princeton University Press. Retrieved from http://culturedigitally.org/2014/10/hackers-draft-digitalkeywords/

Coleman, G. (2013). *Coding freedom: the ethics and aesthetics of hacking*. Princeton: Princeton University Press.

Coleman, G. (2015). *Hacker, Hoaxer, Whistleblower, Spy: The Many Faces of Anonymous*. London: Bloomsbury.

Deuze, M. (2007). *Media work*. Cambridge: Polity.

Deuze, M., Martin, C. B., & Allen, C. (2007). The Professional Identity of Gameworkers. *Convergence: The International Journal of Research into New Media Technologies*, *13*(4), 335–353. http://doi.org/10.1177/1354856507081947

Dormer, P. (1997). *The culture of craft: status and future*. Manchester, New York: Manchester University Press.

Gehl, R. W. (2015). The Case for Alternative Social Media. *Social Media + Society*, *1*(2), 1–12. http://doi.org/10.1177/2056305115604338

Hardt, M., & Negri, A. (2009). *Commonwealth*. Cambridge, Mass: Belknap Press of Harvard University Press.

Hesmondhalgh, D. (2009). *The cultural industries* (2. ed., reprinted). Los Angeles: SAGE.

Hesmondhalgh, D., & Baker, S. (2010). *Creative labour: media work in three cultural industries*. Abingdon, New York: Routledge.

Huws, U. (2014). *Labor in the global digital economy: the cybertariat comes of age*. New York: Monthly

Review Press.

Jarrett, K. (2016). *Feminism, labour and digital media: the digital housewife* (1 Edition). New York: Routledge.

Kelty, C. M. (2008). *Two bits: the cultural significance of free software*. Durham: Duke University Press.

Kelty, C. M. (2013). There is no free software. *Journal of Peer Production*, (3). Retrieved from http://peerproduction.net/issues/issue-3-free-software-epistemics/debate/there-is-no-free-software/

Kubitschko, S. (2015). Hackers' media practices: Demonstrating and articulating expertise as interlocking arrangements. *Convergence: The International Journal of Research into New Media Technologies*, *21*(3), 388–402. http://doi.org/10.1177/1354856515579847

Lievrouw, L. A. (2011). *Alternative and activist new media*. Cambridge, UK ; Malden, MA: Polity.

Mayer, V. (2014). Creative Work is Still Work. *Creative Industries Journal*. http://doi.org/10.1080/17510694.2014.892286

Morgan, T. (2013). Sharing, hacking, helping: Towards an understanding of digital aesthetics through a survey of digital art practices in Ireland. *Journal of Media Practice*, *14*(2), 147–160. http://doi.org/10.1386/jmpr.14.2.147_1

OS News. (2006). Interview with Synfig's Robert Quattlebaum. Retrieved from http://www.osnews.com/story/13241

Paasonen, S. (2014). As Networks Fail: Affect, Technology, and the Notion of the User. *Television & New Media*. http://doi.org/10.1177/1527476414552906

Söderberg, J. (2011). *Free software to open hardware: critical theory on the frontiers of hacking. Doctoral dissertation.* Gothenburg: University of Gothenburg.

Söderberg, J. (2014). The Cunning of Instrumental Reason: Reproducing Wealth Without Money One 3D printer at a time. In J. Söderberg & Maxigas (Eds.), *Book of Peer Production* (pp. 46–58). Gothenburg; Århus: NSU Press.

Stahl, M. (2010). Cultural Labor's 'Democratic Deficits': Employment, Autonomy and Alienation in US Film Animation. *Journal for Cultural Research*, *14*(3), 271–293. http://doi.org/10.1080/14797581003791495

Stiernstedt, F. (2013). *Från radiofabrik till mediehus: medieförändring och medieproduktion på MTG-radio. Doctoral dissertation*. Örebro: Örebro universitet.

Ward, J. (2015). *Cultural Labour in the Context of Urban Regeneration: Artists' Work in Margate and*

Folkestone. Doctoral dissertation. Thesis submitted to the University of Kent.