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

Julia Velkova, Södertörn University
Revised manuscript, after peer review, 23rd of March 2016.

Abstract
Free software development, and the technological practices of hackers have been broadly recognized as fundamental for the formation of political cultures that foster democracy in the digital mediascape. This article explores the role of free software in the practices of digital artists, animators and technicians who work in various roles for the contemporary digital visual media industries. Rather than discussing it as a model of organising work, the study conceives free software as a production tool and shows how its becomes a politics of finding material security in flexible capitalism. While it also extends creative and craft autonomy of its users, these senses do not mobilise a critical project, but rather embed the values of flexible capitalism in technology production, while nurturing further precarity. Empirically, the article draws on ethnographically collected material from the media practices of digital artists and programmers who engage with two popular free software production tools, Blender and Synfig.

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

1. Introduction

Media practices, such as free and open source software development[1], and the technological experiments of hackers have been broadly recognized as fundamental for the formation of political cultures that foster 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); through writing an independent operating system (Coleman, 2013; Kelty, 2008); creating alternative institutions for intellectual property rights management; to experimenting with digital aesthetics and critical art projects (Morgan, 2013). In all these cases, politics are 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; Kubitschko, 2015).

Occasionally, the public entities and institutions that are brought to the world can inspire broader social groups to embed them in their distinct practices; either directly, or by 'modulating' them through inducing them with own meaning. Repurposing them for other goals, these cultural transformations of free software are often conductors of critique of different aspects of the contemporary, capitalist media systems. Among the most prominent examples of critical modulations is the embracement of free software technologies by activists such as in the case of

1 Free software refers to non-proprietary but licensed computer programs that allow users to alter, share and distribute their source code without needing to request a permission.

Indymedia in order to make a case for alternative journalism (Atton, 2007; 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 as a way to develop critique directed towards dominant regimes of ownership over digital 'materials' or software programs used for making digital visual media (Morgan, 2013).

Despite the richness and important insights of these studies, one of their limitations is their focus on the uses of free software predominantly for political activism, by social movements and creative minorities. However, free and open source software has also been increasingly integrated in the practices of corporate technological manufacturers like IBM and Google, or the Hollywood computer graphics giants of Disney and Pixar. With regards to these developments, Kelty (2013) suggests that corporate uses of free software threaten to make its critical potential 'sterile', by being equally easily put in use for mobilising counter-critical power that strengthens monopolies rather than criticising them. At individual level, free software could also be used instrumentally for 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, 2013). Kelty concludes that 'As open source becomes an instrumentalized kind of politics, the possibility of new beginnings fades.' Thus, the critical potential offered by free software seems to simultaneously flourish among activists, and get neutralised by its uses in the media industries, converting it into a motor for new models of value creation (Barron, 2013).

This article seeks to broaden the scope of knowledge about the role of free software in the politics of digital media production by discussing its relevance for other actors, beyond activists, hackers or large media corporations. Particularly, it explores its value, use and development among computer graphics artists, designers and animators who work in a wide range of roles in small advertising agencies, visual effects and computer game and film production companies for the contemporary digital media industries[2], while occasionally engaging in projects of free culture and independent film making.

The material for this study comes from a larger research project on the media practices of two free software computer graphics communities, those formed around the programs Blender for 3D animation and Synfig for 2D animation. The data has been collected through multi-sited ethnography and qualitative interviews with 35 visual media artists and developers. They took place between 2013 and 2015 and documented the uses of these three programs for, predominantly, open and free cultural production. Yet, in the progress of the larger research project it became clear that the same producers who engage, for a wage, in open cultural production and free software development also work in different roles for the media industries where they put in use the same free software media production tools. Some have worked on large projects such as the LEGO movie; Pixar's short films; or Rovio that produces the Angry Birds franchise. Others work for advertising agencies across Europe; produce animation for educational projects or create independent and free culture films. Oscillating between two supposedly antagonistic fields of media production, by having a relation to the industries and to free culture projects, the empirical material that underpins this article represents a fruitful starting point to explore the broader value of free software as a media production tool beyond its uses for radical politics.

In order to understand what meaning can free software have for the practices of individual digital media artists and small studios in the field of digital visual media production, the first section outlines briefly the work context in this field of media production using the overarching framework of Boltanski and Chiapello (2007) on the moral justifications that motivate society to engage in the

2 Throughout the article I alternate between “digital artists” and “media creators” to refer to this particular group of media producers.

ideology of capitalism; complemented with literature on work in the creative industries. The particular lens through which the value of free software is examined is then brought afore by drawing on Howard Becker's work on 'Art Worlds' (1982/2008) and the role of materiality in creative practice.

The argument developed here is that free software for visual media production is conceived by media creators as a form of material capital that represents a source of security in relation to their creative practice in the highly competitive media production environment. Yet, these senses are not mobilized to serve a broader critical political project, but represent individual pragmatic strategies to extend digital artists' creative autonomy in the media industries, or establish links of equivalence with them, while nurturing further precarity.

2. Media production and free software in the new spirit of capitalism

Digital visual media production occurs today to a large extent in the context of post-Fordist work frameworks that promote 'creativity, reactivity and flexibility' (Boltanski & Chiapello, 2007, p. 90) as core cultural values. In their seminal work on the transformations of capitalism between 1960 and 1990, Luc Boltanski and Eve Chiapello advance the thesis that these values are not universal, but rather manifestations of a new 'spirit' of capitalism. By 'spirit' they refer to a set of normative and moral rules that justify society's engagement in capitalism. These rules need to offer a promise for some form of autonomy and security for individuals while serving the common good. In terms of autonomy, Boltanski and Chiapello mean that there should exist an excitement for people to engage in the process of accumulation even if they will not necessarily be the main benefits of it. Individuals also need to feel some form of security for themselves and their children, while participation in accumulation needs to be justified as serving 'the common good which contributes to producing for everyone' (p. 8) and being just.

The transition to post-Fordism in the 1980s and 1990s is regarded by Boltanski and Chiapello as a specific point in capitalism when its justification apparatus is radically redefined. In terms of autonomy, the core values become the 'the development of oneself and one's employability' (p. 111). The former emerges through the paradigm of constant improvement of skills, reputation, being adaptable, self-organised, and participate in novel and exciting projects. In order to become employable, workers need to know how to engage in a project and to remain 'adaptable, physically and intellectually mobile' (p. 112). Enhanced by networked communications, qualities such as flexibility and adaptabibility are argued to emerge from *activity* and *autonomy*, rather than from obedience and belonging to hierarchical structures. In this way, personal development as an option for 'everybody' serves the ideal of the common good, while contributing to broader processes of value production and its accumulation.

This spirit is particularly identifiable in the contemporary media industries and in the debates about autonomy and control of media work. These industries carry a strong allure to young people and creators promising 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, Martin, & Allen, 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 and adjusting it to market demands. They do so by, first, transferring ever greater responsibilities for personal artistic and technical skill development on individual creators; and second, by embedding creators in institutions of employment and regulatory systems of intellectual property that detach creators from their creations, converting their labor into an object of value extraction (Deuze, 2007; Huws, 2014; Stahl, 2010). In the latter context, free software development has been acknowledged to have a potential to bring change in terms of offering more

efficient and less alienating ways of organising and managing media production (Benkler, 2006; Hesmondhalgh & Baker, 2010). Yet, these alternatives have been questioned in terms of their financial viability, and placed free software in the context of the free labor debates (Terranova, 2004).

In the context of digital media production, free labor has been discussed largely in terms of the unpaid work that media users perform by producing content in various online contexts, a work that gets valorised by the media industries (see for example Bolin, 2012; Hesmondhalgh, 2010; van Dijck, 2009). However, unpaid work has always been integral to certain spheres, such as those of social reproduction (Jarrett, 2016) or cultural production (Hesmondhalgh, 2010, p. 277). In these spheres, free labor can be regarded not only in terms of paid or unpaid, but also of good and bad, just and unjust (Hesmondhalgh & Baker, 2010). For example, the internship systems in the media industries today are largely unjust, but unpaid labor as such has always been part of the process of developing skills, ranging from learning to play on music instruments to programming, computer graphics or game development (Hesmondhalgh, 2010). Rather than being unjust, the latter form of free labor stems from the dependency of cultural production on materiality, and is addressed in part by Howard Becker (1982/2008) in his work on the sociology of art production. I will discuss this dependency later on.

Free software producers are also able to engage in the valorisation of their products (Author, 2015), something which, as Barron (2013) shows, has transformed it from a critical practice to a distilled form of the 'spirit' of contemporary capitalism. Converting technology into global software commons, free software enables autonomy and project mobility for everyone, serving the common good. What it falls short of, Barron concludes, is to guarantee security to those who engage in its development, thus paving the way for new forms of critique.

Indeed, security is what Boltanski and Chiapello dismiss as the new 'spirit' of capitalism not offering enough solutions to. The main security that projects or companies can offer to individuals today is development of personal capital that could help employability in future projects and initiatives.

However, as I will suggest further, free software could represent a specific form of security, that of *material security,* that enables media creators who engage in using and developing it also gain a form of creative autonomy, namely craft autonomy. In order to understand how this happens, it is necessary to take a different perspective on free software and approach it as a *media production tool*, rather than as a *model of organising* work. Hence, this article continues with exploring deeper the relationship between technology and digital media creators, rather than between individuals and the broader organisational structures of media production.

3. Materials for media production

Employability and participation in media projects is largely predicated on the creativity and technical skills of creators. Those, in turn get developed in relation to the materials, or tools that are available to them. In the current 'spirit' of flexible capitalism, media creators need to be adaptable and flexible not only in relation to the organisations or projects that they work on; their possibility to sell their work or develop critique depends also on the flexibility and creative autonomy that the technologies they work with could offer to them.

From this perspective, 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 (1982/2008) offers some insights on how to understand these entanglements in relation to creative 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 forms 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, 1982/2008, p. 3).

In the case of producing specialised media, such as digital visual media, creators need materials that are designed and manufacturerd specifically 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. Despite that manufacturers try to be sensitive to the needs of the creators of a particular medium, 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). Through this argument Becker establishes a link between technical innovation, creative autonomy and the frameworks of creation and distribution of materials. The fewer manufacturers dominate the market, he argues, the more insensitive they get to what artistic minorities want or need. Occasionally artistic minorities can revert to the craft of making their own materials, or of customising existing ones if facing the threat of discontinuing the material against which creators have developed their skill; in the case of wanting more than the available materials can provide; or in the case of lack of available materials to satisfy a creative impulse.

It is in this context, and rather pragmatic considerations about individual strategies to develop creative practice that free software emerges as a tool of high value among media professionals, digital artists and aspiring media workers. The next section substantiates this point through a discussion of the emergence of two popular free software tools for computer graphics production, Blender and Synfig.

4. Crafting technical autonomy: The Blender and Synfig free software projects

The two free software programs discussed here, Blender for 3D animation and sculpting and Synfig for 2D vector animation, were conceived as digital tools that would enable their creators to exercise a greater degree of craftsmanship, innovation and autonomy in the medium. They also represent the free software alternatives for professional animation production such as 3D Studio Max, Adobe After Effects, Anime Studio and Maya.

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 independent 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 bankruptcy in the early 2000:ds, both projects emerged as free software through very particular processes of de-commodification, the details of which I have discussed elsewhere (Author 2015).

In the case of Blender, the need to start developing an independent program 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 author 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 creative ambitions.

Until the mid-1990s, software for computer graphics development was distributed as an add-on to a very expensive hardware that media creators anyway needed to invest in. The computer industry restructured in the end of the 1990s. With computing power getting cheaper and more ubiquitous, companies started developing business models around selling, and more recently renting specialised software for computer graphics production. The changes in the politics of distribution of software for computer graphics production have been experienced as constraining creativity, experimentation and large-scale projects for small studios and individual digital artists:

> [In the 1990s] the hardware costed money, but once you had it, you could do anything – we were getting CDs with Silicon Graphics code! It was proprietary stuff, but it didn't prevent us from making things with it... computer graphics is about openness, because you can build on everyone else's developments. Once you get a patent or close it – people find a way around it. (notes from informal conversation with Ton Roosendaal, May 2015)

Hence, the experiences of material constraints to continue experimenting with computer graphics led Blender's creator to find his way around it by re-licensing his program as free software, as a strategy to retain technological and creative independence and let the program grow by allowing other digital artists to contribute to it:

> 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 (Ton Roosendaal, interview, August 2014) .

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, Konstantin Dmitriev, 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 an independent feature-length animation film.

In search for style, his work had started with proprietary programs 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...'. Facing in this way a constraint to innovate in the medium, instead of trying to adapt his practice to the technical limitations of the tool, he moved to experimenting with free software as a way to adjust technology to the scale of his creative ideas. Initially he tested 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'.

While improving his skills on Blender, Konstantin specialised also in 2D animation, in

parallel with using proprietary programs until their development frameworks collided with his own work process. The manufacturer of the proprietary 2D animation program Moho, discontinued its development under Linux, which had gradually become Konstantin's main platform of work. The impossibility to use this program as a production tool caused a great sense of anxiety:

> I liked that everything (in Moho) 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 (Konstantin Dmitriev, interview, November 2014).

Since then he focused his efforts on studying and developing the free software Synfig which he integrated at the core of his creative practice, and multiple projects ranging from free-lancing work; to education; to independent free culture production (see author, 2015a).

Both Konstantin's and Ton's choices to invest their time in developing Synfig, respectively Blender emerged out of explicitly pragmatic concerns related to the possibilities to create within frameworks of own making and under their own control. This form of engagement with technology has been referred to, in the contexts of free software development and hacker cultures, as forms of establishing 'craft autonomy' (Coleman, forthcoming), one that fosters skill and expertise, but also sensibilities similar to pre-industrial, craft like engagement with technology. It also reproduces the reactions to material constraints that Howard Becker had outlined as common for artistic practice, suggesting that problems related to the possibilities for creative expression with tools of production continue to be a topic of high concern even in digital media production.

Despite the freedom of creative expression which developing autonomous media production tools granted to their creators, it also constrained their autonomy in new ways. In order to fulfill their ideas they needed to motivate more people to adopt these technologies and contribute to the free software projects in order to let them grow in functionality. Both Blender and Synfig faced the problem that instead of developing art projects, they needed to develop frameworks to train or convince other people to use these technologies. As Becker (1982/2008, 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. There is no space in this article to discuss in detail the strategies employed in these cases, but for the present argument it is enough to say that Blender succeeded in creating a large user base to a greater scale than Synfig, and is today embraced by animators, digital artists and technical artists who use it for a broad range of purposes. The usage ranges vary from experimental concept art projects; trough developing 3D printing models, to experiments of novel forms of artistic collaboration; open culture projects; to the production of special effects, games, animation, and simulations for the media industries.

The variety of uses which it finds implies that the public nature of free software represents a source of value and craft autonomy for a broad range of actors and purposes. The next section discusses three main ways in which digital media artists find meaning in these tools, and illustrates how they reconcile craft and creative autonomy with efficiency, independence, ultimately securing *materially* their creativity.

5. 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.

French free-lancing illustrator and digital comic artist, David Revoy, recalls how he used to work with proprietary digital production tools such as Corel Painter, Manga Studio, Photoshop Elements and, CS2. After upgrading to a newer computer and a newer version of a proprietary operating system, all these tools stopped working: 'I had to do a lot of horrible hack to make all my

software run[ning] on it, but it wasn't stable as it was on Xp anymore. I had to reboot almost twice a day' (Revoy, 2013). From a tool that automates and mediates creative expression, media production software can become an artifact with 'agential' (Paasonen, 2014) properties that may, for some time, leave the user powerless. Faced with the choice of either re-purchasing all programs to match the new operating system and hardware, of reverting to the older computer and operating system, or of doing something completely different, Revoy (2013) chose to move to free software:

> I thought all of this circus couldn't work in the long term, and wasn't happy.... I switched my machine to a full open-source system around 2009...thinking, open-source could work on the long term.

The result of this move was not explained in terms of economic gains, but in the qualitative difference related to a new degree of creative autonomy and security gained in relation to technology:

> I really like the independence I get from it: I can install it on laptops, every machine, upgrade, downgrade, fine tuning it. This independence is gold. The con is that I'm now dependent on hardware 'linux' compatible. Which is not easy to find, and not well documented. (Revoy, 2013)

If for David, free software has initially been a way to reduce his material and creative dependency from technological frameworks out of his control, for other media producers, switching to free software has been a way to increase their work efficiency.

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 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 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 course you can customise everything you want, now after Blender 2.5 Which I do, a lot. (Hjalti, animator, interview, August 2014)

Later versions of Blender and Hjalti getting more experienced with it allowed him to adapt it to his own work process in a way which increases his speed of work. In practice this meant adjusting small details, such as the position of his hands which he wanted to keep static while working. Until moving to Blender, whenever he needed to change perspectives on the screen while animating, the program interface would require him to move his hand to the keypad on the right side of his keyboard. He experienced this as a constraint to be efficient:

> I am doing it every 10 seconds. And take one 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. (Hjalti, animator, interview, August 2014)

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.
> (Hjalti, animator, interview, August 2014)

This example shows how free software as a production tool is conductive to frameworks of rationalising production, and personal skill development while maintaining a strong sense of autonomy among its users. This combination ultimately gives a competitive advantage in the work market of the media industries. Of course, possibilities for customization exist in other software too. Yet, in line with Becker's argument about the constraints of materials, the limits to which free software allows creators to adapt technology to their everyday practice depends more on individual technical skill and creative ideas rather than on the production frameworks and affordances set by software manufacturers.

Besides reducing dependencies, and increasing efficiency, many digital artists value free software for its infinite adaptability and extensibility.

In the spring of 2014 one free-lancing animator and one technical artist from Costa Rica worked on a 4-seconds long shot for the teaser of a larger free culture animation film project. The shot was supposed to show a green caterpillar blinking. The animator wanted the pupils of the caterpillar to resemble the facial features 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 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 :)
> (Daniel Salazar, technical artist, 2014)

In this case, the process of making animation has been very similar to hacking. 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, animation director, archived blog post, 2014).

The above examples illustrate how free software strengthens feelings of creative autonomy in their users, by being flexible and adaptable to individual needs of creativity, efficiency and material independence. In their totality, these senses construct free software as a source of individual material security and capital that allows digital artists to gain competitive creative advantage in the post-Fordist media industry frameworks. With the increased transfer of responsibility over skills

development on individual media creators, the choice of technology becomes an investment that can increase media producers mobility and employability in different projects. At the same time, while free software stimulates a craft like engagement with technology, it illuminates how the values of personal self-development, flexibility and security of the new 'spirit' of capitalism get embedded in digital artists's technological choices. The problem which emerges is that the security and autonomy that controlling and extending free software digital tools gives may decrease criticism against some problematic aspects of the post-Fordist production frameworks, such as precarity of labor. As the next section will show, while digital artists strengthen their creativity and material security through free software, they nurture precarity of work further.

6. Tools development as a source of precarity

Once digital artists identify free software as tools of value to them, they employ different strategies to attempt to further shape and adapt the programs to their individual needs. Those artists who are unable to code (and they constitute a majority), or do not want to dedicate time to code, resort to financial and rhetorical means to convince programmers to do the work for them. Below I discuss three dominant ways in which this can happen.

5.1 Hiring a developer

The technical possibility to extend Blender for other purposes than those intended by its original creator emerged from the need of a technical artist to rationalise his process of work in a wealthy media production company. He attempted to find a less costly and more flexible alternative to a professional 3D software 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 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 the 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.
>
> (technical artist, interview, 2014).

The functional extensions made at that time entered the core of Blender, and made it possible for other artists to further develop the program. While representing a contribution to the common good, the possibility that free software opens for an artist to hire a developer changes the status of the artist. From being a wage-taker who sells his work to the industry, the artist can become an employer who creates small, temporary jobs for programmers. By offering programmers temporary projects, digital artists mirror the frameworks of the media industries by outsourcing jobs with the promise of personal development, employability and a wage, for the common good. Hence, while free software represents a source of material security for artists, it becomes a source of work insecurity towards developers, nurturing precarity further.

5.2 Becoming a financial patron of a project

Another common strategy employed by artists to influence the general direction of free software tools development is to become a financial patron to the project. This can happen by making small donations to the free software projects in order to buy developer time to develop the project in the direction they want.

A free-lancing animator from Sweden who specializes in cut-out animation which he sells to the Nordic advertising and film industries explained how he could make small financial donations to Synfig in order to push its development in the direction that he needed. His principle has been to

donate 3-4 per cent of his income from commercial projects to the free software projects he uses, with occasional higher donations in order to set a priority for a specific feature development. He remembered how he paid once a few hundred euros to the project in order to speed up the development of a specific function in Synfig that he needed in his work for the industry.
This form of exercising influence over the broader technical development of the project makes digital artists into patrons who become connecting links between a media project (be it for the industry or not), the creative visions of an artist and the technical community that can be convinced to prioritise the development of a feature.

5.3 Motivating developers

When artists do not have financial means to invest in a project, they resort to rhetorical means to motivate a developer to do the job for them for free, an approach that is the driver of major disputes in the communities formed around the free software tools.

Digital artists can request features and extensions directly from the programs' main developers. Mobilising rhetorics and prototypes of unfinished media projects in order to illustrate the need for improving software in a particular direction, these interactions become the locus of many tensions and conflicts. In some cases feature requests are welcomed and fulfilled, but in most cases they are ignored:

> 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... 'I'm using Blender for an interactive blah blah and it's draw modes don't work for me because...etc' ...people who use Blender for ten minutes and don't like color also post...
>
> (Blender developer, interview, December 2014)

This example illustrates that despite gaining a greater technical autonomy, free software makes digital artists deeply embedded in the social dynamics surrounding the maintenance and development of their tools of choice. Those artists who manage to convince the developers in the importance of their request are usually those who are most active in the media industries and have concrete, urgent needs:

I was already doing graphics that were watched by millions, and I started falling in love with [Blender] because it is so versatile. And plus I really liked the idea that you could change the program....I started asking the question: How easy is it to get camera movements from Blender to After Effects....One of the developers out there got and wrote a little script. Within two weeks you could do this thing that you couldn't do before. And yeah, that's what sold me on Blender. Wasn't the interface, wasn't the toolset, it was just the fact that you could change it. That made the case that even if you are not happy with it, if you argue for your case well, you can actually get changes to it.

(animator, interview, 2015)

Hence, a rhetorical approach anchored in a concrete project for the media industries can become the equivalent of a financial donation as it fulfillment may bring value for a broader range of digital artists, while satisfying individual creative demands.

6. Concluding discussion

The empirical examples discussed in this article show that free software's role in the politics of digital media production should be understood as individual strategies to find material security, and extend personal creative and craft autonomy through technological choices. Approaching free

software as a tool, rather than as a form of organising work allowed to illuminate the degree to which creative autonomy of digital media creators is configured in relation to the affordances, mouldability and degree of control over the programs that they use in their everyday creative practice. The main issues which artists used to struggle with in the past, as described by Becker (1982), such as dependency on materials and the frameworks of their production, changes in the politics of their distributions, and not least, their affordances, remain highly actual in the contemporary digital mediascape.

Drawing on Becker, the article conceived free software as a strategy to develop own materials and independent frameworks of production in response to changes in the political economy of software distribution. In Becker's framework, such an approach has been commonly used by artistic minorities. This article showed that free software can resemble this approach by becoming relevant for a broader range of users, beyond creative minorities, particularly for those working in different roles for the contemporary media industries. Hence, free software as a source of value for digital artists is about meaningful, and not ideological self-realisation (Hesmondhalgh & Baker, 2010, p. 180ff). In the new 'spirit' of capitalism that promotes self-development and outsources the responsibility for skill development on media producers, the ability to shape technology according to distinct creative ideas becomes an individual strategy to remain flexible and competitive.

The specific ways in which artists find meaning from using free software as a production tool are in bringing their form of work to a form of pre-industrial craft, and saturating their work with 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). The senses of craft autonomy developed through free software are, importantly, not mobilised for a broader critical or political project for social change, but are rather pragmatic, rooted in strategies to influencing technological development in ways that benefits one's individual work practice. As a consequence, digital artists inscribe further the values of the new 'spirit' of capitalism, embedding them in the free software tools that they use and develop.

Finally, in crafting security and autonomy for themselves, digital artists do not offset some of the negative effects of flexible capitalism, such as the shift of responsibility for skill, personal development and finding work onto individual creators. Rather, their practices nurture further precarity of labor by becoming employers or patrons of other groups of creative workers, such as hackers and software developers. Further research could fruitfully explore further the practice of media workers hiring developers to code functionality for them, and explore whether such engagements create new hierarchies or forms of exclusion that need critique, or whether they are a source of positive aspects and pleasures of digital work that enhance autonomy and creativity in the digital media industries even further.

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). New York: Routledge.

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

Barron, A. (2013). Free software production as critical social practice. *Economy and Society*, *42*(4), 597–625.

Becker, H. S. (1982). *Art worlds*. 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. (2012). The Labour Of Media Use. *Information, Communication & Society*, *15*(6), 796–814.

Boltanski, L., & Chiapello, È. (2007). *The new spirit of capitalism*. London: Verso.

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.

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.

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

Hesmondhalgh, D. (2010). User-generated content, free labour and the cultural industries. *Ephemera*, *10*(3/4), 267–284.

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.

Lievrouw, L. A. (2011). *Alternative and activist new media*. Cambridge: 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.

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*.

Revoy, D. (2013, May 1). Why I use Open-source? Retrieved 5 October 2015, from http://www.davidrevoy.com/article170/the-choice-of-open-source

Salazar, D. (2014). Caterpillar Making Of – Animating UVs. Retrieved 18 March 2016, from https://gooseberry.blender.org/caterpillar-making-of-animating-uvs/

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.

Terranova, T. (2004). *Network culture: politics for the information age*. London ; Ann Arbor, MI: Pluto Press.

van Dijck, J. (2009). Users like you? Theorizing agency in user-generated content. *Media, Culture & Society*, *31*(1), 41–58.